Evaluation of Robot Path Planning Algorithms in Global Static Environments: Genetic Algorithm Vs Ant Colony Optimization Algorithm

Nohaidda Sariff and Norlida Buniyamin

Abstract—**This paper presents the application of Genetic Algorithm and Ant Colony Optimization (ACO) Algorithm for robot path planning (RPP) in global static environment. Both algorithms were applied within global maps that consist of different number of free space nodes. These nodes generally represent the free space extracted from the robot map. Performances between both algorithms were compared and evaluated in terms of speed and number of iterations that each algorithm takes to find an optimal path within several selected environments. The effectiveness and efficiency of both algorithms were tested using a simulation approach. Comparison of the performances and parameter settings, advantages and limitations of both algorithms presented herewith can be used to further expand the optimization algorithm in RPP research area.**

***Index Terms*—Mobile Robot, Robot Path Planning, Global Path Planning Algorithm**

I. INTRODUCTION

PATH planning is an important task in autonomous mobile robot to enable the robot navigation system to identify a safe path (without colliding with obstacles) to goal. Path lanning research covers a wide area of robotics research that includes path planning in static [1-6] and dynamic environments [7-9]. In static environment, the environment is assumed to be known and the global path can be generated in advanced based on the provided map. This is known as a model-based approach which is implemented offline. In a dynamic environment, the mobile robot will respond to unexpected situation and the map is updated to create a new path in responds to the environment change. This is known as sensor based or online approach where the new path is constructed once the sensor detects that there are changes in the environment. However, the research presented herewith focused on path planning in static environment.

Generally, there are two main elements involved in the process of global path planning, which are, the mapping and the path planning algorithms (PPAs). The environment will be mapped in the initial process. The existence of these maps will simplify the process of finding the optimal path when the PPAs are applied to the maps. Example of maps created and used by previous researchers for their RPP purposes is C-Space such as generalized cones[10], cell decomposition; grid map [2, 4, 11], graph; visibility [12, 13] and MAKLINK graph [7].Within these maps, the area of feasible (no obstacles) and non feasible nodes (obstacles) have been configured before PPAs are applied to find the paths. The selection and utilization of maps in a research must be compatible with the selected PPA and this depends on the objectives and applications of the research itself.

Manuscript received December 28, 2009. This work was supported in part by the University Technology Mara (UiTM). Performance Evaluation of Robot Path Planning Algorithms in Global Static Environments: Genetic Algorithm Vs Ant Colony Optimization Algorithm.

Nohaidda Binti Sariff is with the University Technology Mara (UiTM), Shah Alam, Selangor, 40450 MALAYSIA (phone: 03-55436103; fax: 03-55435077; e-mail: nohaiddasariff@yahoo.com).

Norlida Buniyamin, is a lecturer at the Faculty of Electrical Engineering, University Technology Mara (UiTM), Shah Alam, Selangor, 40450 MALAYSIA (phone: 03-55436082, 03-55435077; fax: 03-55436082;
e-mail: nbuniyamin@salam.uitm.edu.my).

The evolution of PPAs shows improvements of PPA from one generation to the next generation[14]. The later generation of PPAs has been created to be more adaptive and able to work within the robot environment itself. Several traditional PPA approaches have been developed by previous researchers such as artificial potential fields [15-18], neural networks [19], heuristic algorithms [1, 11, 20] and etc. Each method has its own strength and limitations over others in certain aspects of path planning itself. The evolution of traditional approaches produced a widely used heuristic algorithm known as A* algorithm [1, 21]. This algorithm is widely used not only in static environments but it subsequently evolved to D* algorithm [11, 22] which is then widely used in partial known [22, 23] and dynamic environments[24].

However, since appearances of artificial intelligences [25, 26], the problem of path planning has been view as one of an optimization problem. The idea of utilizing artificial technologies in developing intelligent agent for RPP purposes has grown since then. Currently, researchers use neural networks [19], evolutionary computation [27-29] and swarm intelligence [30-33] in RPP research. This newer approach enables the identification of optimal robot path that satisfies optimization criteria such as shortest and fast with small computational time. The wide application of PPAs such as Genetic Algorithms (GA) [2, 4, 34-40] and Ant Colony Optimization (ACO) algorithm [7, 8, 41-43] in RPP research is because it can produce the optimal path effectively compared to the traditional approaches.

The research presented herewith investigates the effectiveness of GA and ACO applied in different complexity of free space nodes in a global map[44]. The performances of the algorithms were evaluated in terms of speed and number of iterations when both algorithms were applied in global maps with different complexity of free space nodes. The global maps that were constructed consists of different number of obstacles, different location of obstacles, or different type of mapping algorithms used which produced different complexity of nodes. It is hoped that the findings from this research can be used by researchers in the RPP optimization area.

This paper discusses GA and ACO algorithm construction including mapping of simple environments and complex environments that were used in this case study. Then the comparison of performances (using simulated data) between GA and ACO in terms of speed and number of iterations in different maps is presented. The pattern of the parameter settings for both algorithms in different complexity of environment as identified is also included. Finally the discussion of the usefulness of these algorithms in RPP research was discussed.

II. RESEARCH BACKGROUND

This section covers the utilization of GA and ACO in RPP research areas where different cases have different methods of presenting the solution to find an optimal robot path.

A. Genetic Algorithm

Since its appearance 1975 [27, 28], GA has been used in solving many robot path planning optimization problems. GA is a search technique inspired by evolutionary biology where it work is based on principle of the fittest of the chromosomes. With its ability to work with parallel search techniques, the use of GA contributed to the success in many robot path planning research. For example, Gihan Nagib et al [4] proposed the use of GA to find robot path based on map of free space nodes. Kuzuo Sugihara [35], R. Ramakrishnan [37] also proposed the used of GA with different encoding techniques to ensure GA can find optimal path without depending on the feasible nodes given in the map. Yangrong Hu [45] modified classical GA by incorporating the domain knowledge into specialized operator to improve GA performances when it works in environments that consists of obstacles. From the above literature study, it can be concluded that GA performances depends on the way the solution is encoded in chromosomes, accuracy of the fitness function and variation of genetic operators which determine the whole process of GA.

B. Ant Colony Optimization Algorithm

ACO, compared to GA is a newer optimization method. Introduced by Marco Dorigo [30] in approximately 1992, the application of this algorithm in RPP research increased rapidly as it is a powerful tool for solving hard combinatorial optimizations problem. ACO was inspired by analogy of behavior of real ants, when looking for foods. Tan Guan Zheng [7] proposed the use of ACO to find robot path based on map of MAKLINK graph. Hao Mei [8] combined ACO with Artificial Potential Field to produced the path planning in dynamic environment. Gengqian et al [41] have proven that ACO can find optimal path in their grid map by proposing its own probability equation. However, a literature study shows that the application of ACO to solve RPP problems has not been explored in detail.

III. Research Methodology

Fig. 1 below illustrates the proposed method applied within this research. In the beginning stage, the robot environment needs to be mapped using an appropriate global map as discussed in section A below. This map will create an output of nodes represented by x-y coordinates. Then, GA and ACO will start to initialize the population of path using its own approaches from start to goal by using all the provided nodes including the start, goal and all intermediate nodes. During the initialization, the integer number representing each node will be used. However, during the evaluation, the real x-y coordinates will be used. At the end of the process, the optimal path will be found.

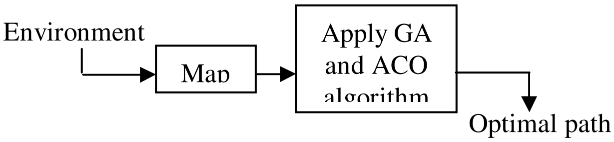

Fig.1. Proposed Method

A. *Environment Modeling*

In this research, a 2D grid map as shown in Fig. 2 below was used. The free space nodes (white grid) represent the area the robot can traverse including the robot size. The obstacles area (black grid) represents the boundary of obstacles with the safety region and the yellow grid represents the feasible free space nodes that can be traversed by the robot. However, during construction of the map used for this study, the obstacles was assumed to have been eliminated and only a route of feasible nodes is left as shown in Figure 4,5 and 6.

By assuming the 10 X 10 cm grid map is the size of the map, three different complexities of free space nodes have been developed. A simple environment consists of 12 numbers of feasible free space nodes (Fig. 4), average complexity with 22 numbers of nodes (Fig. 5) and complex environment have 63 numbers of nodes (Fig. 6). These feasible free space nodes have been located randomly within the grid map for this research study. By using these maps, the algorithm will asses all feasible paths using the available feasible nodes as shown in population of path as depicted in Fig. 3 below. However during implementation, there will be non feasible nodes produced during the generation of the optimal path.

IV. Units

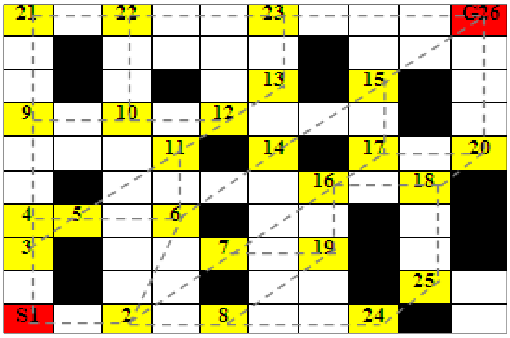

Fig. 2. Global feasible map with obstacles

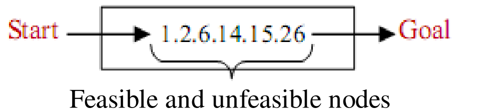

Fig. 3. A sample of path population consists of feasible nodes of Fig. 2

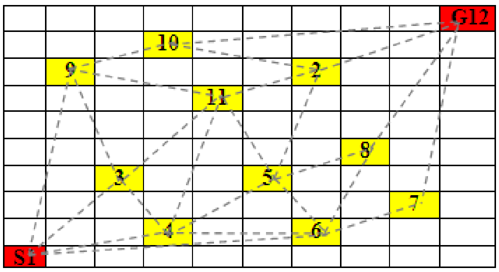

Fig. 4. Simple free space map

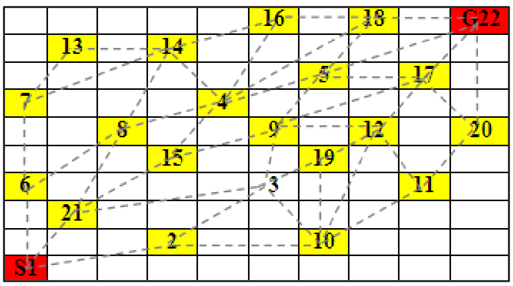

Fig. 5. Average free space map

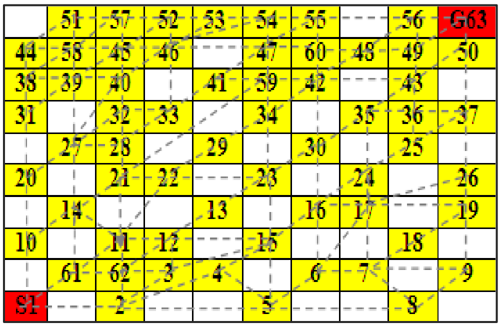

Fig. 6. Complex free space map

B. Genetic Algorithm Design for RPP

The outline of GA is given in Fig. 7. The initial solutions of the RPP problem will initialize in population randomly. In the first case, the population will initialize based on the feasible nodes provided in the global map only. With the complete population, the fitness is evaluated by using the formula below:

$$\text{Fitness node}=\sqrt{(x2-x1)^2+(y2-y1)^2} \tag{1}$$

$$\text{Total Fitness}=\begin{cases} \sum \text{Fitness node} & ;\text{Feasible} \\ 100 & \text{Unfeasible} \end{cases} \tag{2}$$

After the fitness of each population has been evaluated, it will be ranked using an elitism approach. The shorter path will be represented with a high fitness value and will be selected to be carried forward to the next generation while the long path represented with a low fitness value will be eliminated and removed from the population. The good parents, which are carried forward to the next generation will produce the diversity of population that consists of a good child from the genetic operators process. Then, this process is repeated until all of the GA population found the same optimal path with no difference of the fitness value where the distance is equal to 0. It is at this moment, that the solution converges. However, type of GA and important parameters specifications related with GA used in this experimental research is defined in Table I below:

TABLE I
GA PARAMETER SPECIFICATIONS

GA Properties	Parameter
Type of GA	Classical GA
Chromosomes type	Fixed length chromosomes
Population size	Varies, depends on cases
Chromosomes length	Varies, depends on cases
Selection type	Elitism
Crossover type	Two point crossover
Mutation type	Flip bit
Crossover rate	0.75
Mutation rate	0.75
Convergence criteria	Cmax-Cmin≤0.00001

C. Ant Colony Optimization Design for RPP

The ACO algorithm used in this experiment is the Ant System (AS) algorithm as proposed by Marco Dorigo [46]. However, equation (3), a new heuristic equation of the state transition rules, which is more suitable for the applications of this research, was used. The evaluation fitness and ACO parameter setting was created based on the requirements of this research.

The design of AS for robot path planning was divided into three important rules which are state transition rules, local update rules and global update rules. In the beginning, ants will determine the next node to be visited by using the state transition rules based on heuristic and the amount of pheromone laid down by the ants as shown in equation (3) below:

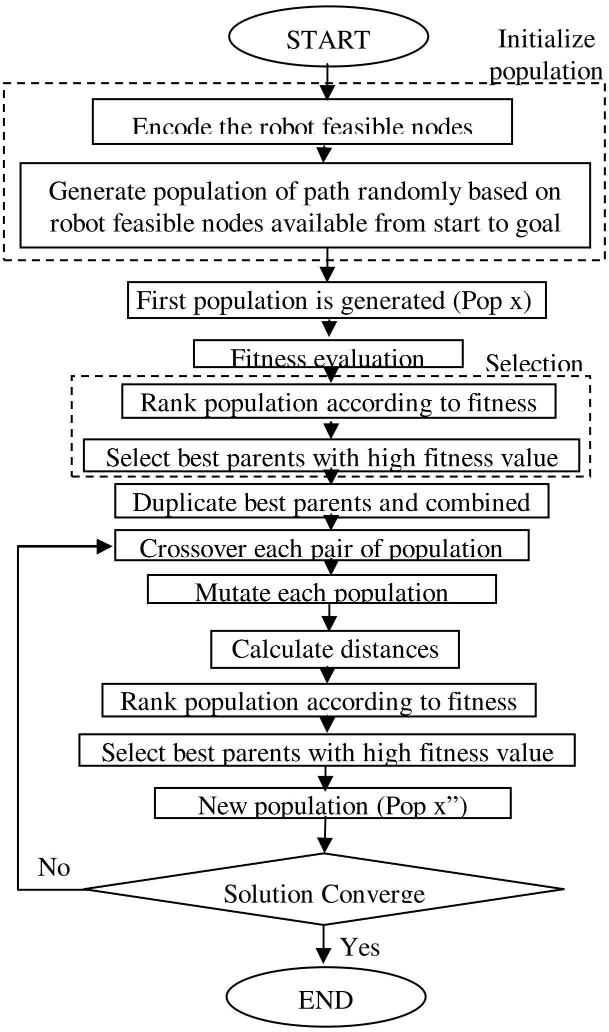

Fig. 7. Outline of GA for RPP of a mobile robot

Probability ij=heuristic * pheromone (3)

=[(1/distance between vector start to subpath and start to perpendicular subpath with reference goal)$^{\beta}$ * (trail/$\sum$ trail)$^{\alpha}$]

* β=heuristic coefficient, α=pheromone trail coefficient

An accurate value of distance by heuristic equation and the higher amount of pheromone of the visited node will be obtained by the ants that have higher probability to choose that nodes. Within these rules, the ants can balance between the exploration and exploitation from the relatives' coefficient provided, known as alpha and beta. During the construction of the path, the pheromone will be reduced locally by the given evaporation rate by using the formula of update local rules below:

$$\text{Tij (new trail)} \leftarrow (1-\rho) * \tau\text{ij (old trail)}, \qquad (4)$$

* ρ=evaporation rate

After all the ants complete the path to goal, then the process of global updating is applied where ants will deposits its pheromone based on the path distance.

$$t_{ij} \leftarrow t_{ij} + \sum \Delta t_{ij}^{k} \qquad (5)$$

Δt_{ij}^{k} = amount pheromone of ant m deposits on the path it has visited. It's defined as below:

$$\Delta t_{ij}^{k} = \begin{cases} Q/C^{k} & \text{;if arc (i,j) belongs to path } P^{k} \\ 0 & \text{;otherwise} \end{cases} \qquad (6)$$

where Q is number of nodes and C^{k} is the length of path P^{k} built by the ants

The amount of pheromone will continuously be updated until it attracts more ants from the next generation to follow the shorter path. Finally, the optimal robot path is found by using behavior of ants' concept as shown in Fig. 8 below.

The parameter specifications of ACO utilized in this experiment is shown in Table II below.

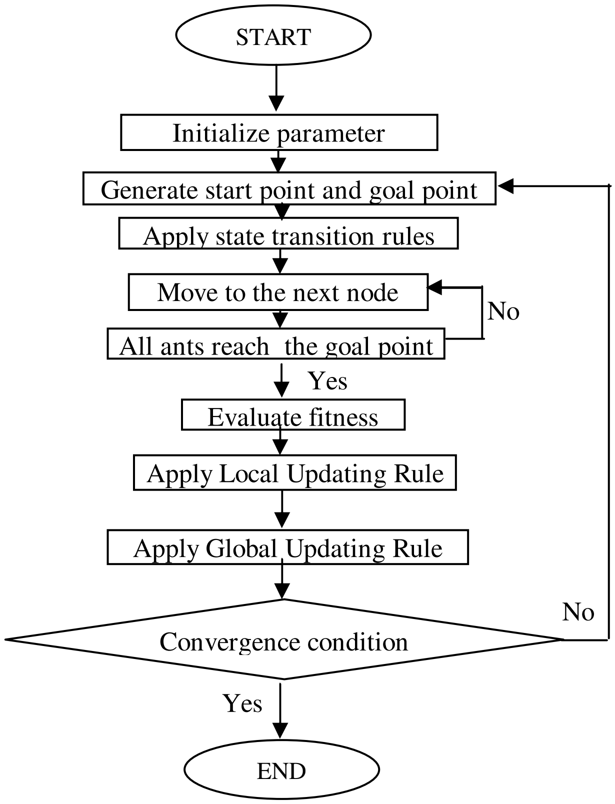

Fig. 8. Outline of ACO for RPP of a mobile robot

TABLE II
ACO PARAMETER SPECIFICATIONS

ACO Properties	Parameter
Population of ants	Varies, depend on cases
Length of ants junction	Varies, depend on cases
Pheromone trail coefficient,β	5
Heuristic coefficient,α	5
Evaporation rate,ρ	0.5
Convergence condition	Cmax-Cmin≤0.00001

V. EXPERIMENTAL RESULTS & DISCUSSION

A. GA performances in different complexity of free space nodes

The parameter settings of the population size and length are based on the requirement of GA to determine the optimal path for each case study as shown in Table III below. The optimal path and the path cost found in ten test runs are recorded in Table IV below. In addition, the results of GA performances evaluated based on time and number of iteration required to obtain the optimal path is shown in Table V. The optimal path found by GA for each type of complexity environment as illustrated in the MATLAB workspace area are shown in Figures. 9, 10 and 11.

TABLE III
GA PARAMETER SPECIFICATIONS

Environment	12 nodes	22nodes	63 nodes
Population Size	50	50	200
Length	8	15	20

TABLE IV
PATH AND PATH COST FOUND BY GA

Environment	12 nodes	22nodes	63 nodes
Path	1.3.11.2.12	1.21.15.9.17.22	1.61.11.22.29.34.42. 48.49.63
Average Path Cost	13.053	13.142	14.136

TABLE V
COMPUTATION TIME AND ITERATION

Environment	12 nodes	22nodes	63 nodes
Average time (sec)	72.1244	242.7488	2144.84
Average iteration	7.7	9.5	21.8

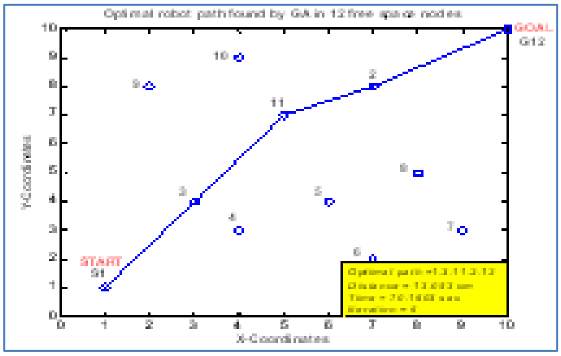

Fig. 9: Optimal path found in 6th generation (12 nodes)

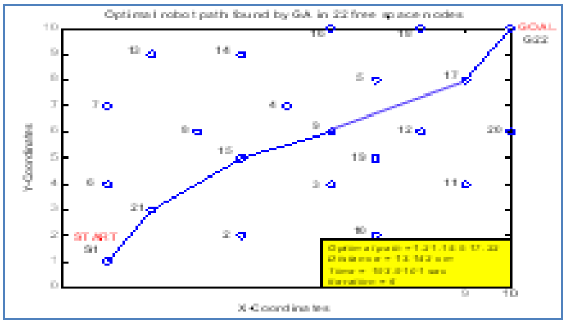

Fig. 10: Optimal path found in 8th generation (22 nodes)

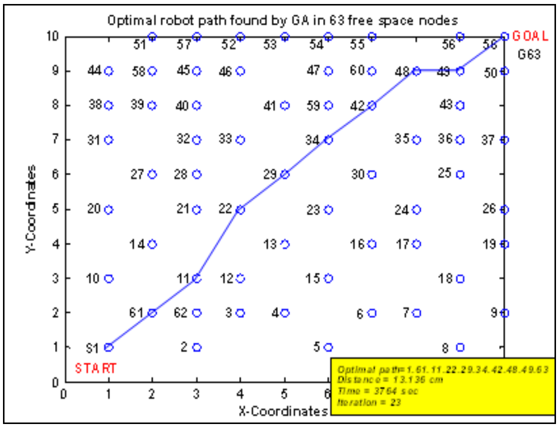

Fig. 11: Optimal path found in 23h generation (63 nodes)

Based on the results tabulated in Table V, performances of GA in terms of time and number of iteration it takes to find the optimal path increases as the number of free space nodes increase. Average time GA takes to converge in ten test runs is around 72 seconds for 12 nodes, 242 seconds for 22 nodes and 2145 seconds for 63 nodes. The percentage of time increment for each case is 3.38% increment from 12 nodes to 22 nodes and 8.823% increment from 22 nodes to 63 nodes. In addition, the number of iteration also increased similar to the increment of time. Starting from an average number of 7 iterations for 12 nodes it then increased to 10 iterations for 22 nodes and 22 for 63 nodes.

Thus, it can be concluded that an increment in the number of nodes within the environment will affect the process of optimization, i.e. increase the time and number of iteration that GA takes to find an optimal path.

The main factors that will affect the time and number of iteration of GA were the increment of the chromosomes length which is proportional to the increment of the number of nodes. The higher the number of feasible nodes, the longer length of chromosomes is required to allocate intermediate nodes within the chromosomes in order to produce the complete population of path to goal. With enough length of chromosomes, GA can perform effectively to cross and mutate the chromosomes to produce many population of optimal path until the solution converged.

However, due to the increment of length, GA needs more time to initialize the population of path and to calculate the fitness of each chromosome for every single bit of chromosome that represent the distance value from one node to another node. In addition, variety type of child population has been produced from the population of parents which have long length chromosomes. With this variation of child, the process of finding an optimal path in the next generation became difficult based on the quality of the child itself. This causes GA, to require more time and additional number of iteration to find the optimal path to goal.

Another reason for the increment of time is because of the increment of the size of population when the number of nodes increased. The size of population need to be increased when the number of nodes increases to ensure the number of suboptimal path within the population in each generation is enough to produce the next quality child for the next generation. This is to guarantee GA can find the optimal path when the solution is converges. In an environment with simple complexity, the less number of nodes will produce less number of variety child which will definitely drive GA to converge efficiently. However, in a complex environment there are more nodes and thus the possibility to have variety type of population is high which will make it more difficult to find the optimal path. To solve this problem, the number of population also should be increased to get the suboptimal path in each generation easier and thus guarantee that the GA can find an optimal path while the solution converges.

Thus it can be concluded that when the number of feasible nodes increases (i.e. complexity increases), the settings of parameters such as the population size and length of chromosomes should also be increased. This will then lead to the increment of time and number of iterations the GA takes to find the optimal path. This indicates that GA is more suitable for simple and average complex environment where the time and number of iterations will be less compared to that of the times and iteration number required for a complex environment. However, although it takes a longer time and higher number of iterations, GA is a robust optimization algorithm as it can find the optimal path quite efficiently in environments of different complexity.

B. ACO performances in different complexity of free space nodes

For the ACO algorithm test runs, the parameter settings of the population size and length for each case study were set similar with GA to ensure that the comparison is valid. Table VI are the average results from 10 tests runs that shows the average time and no of iterations required by the ACO to find the optimal path. The optimal path found by ACO is the same as found by GA for each case.

TABLE VI
COMPUTATION TIME AND ITERATION

Environment	12 nodes	22nodes	63 nodes
Average time(sec)	9.337	32.566	814.819
Average no. of iteration	1.9	3.1	3.5

As shown in Table VI, the average convergent time required by ACO are around 9 seconds for environments with 12 nodes, 33 seconds with 22 nodes and 815 seconds for the complex environment with 63 nodes. This shows that when the number of nodes increased, the time taken by the ants to find optimal path will also increase. This is because the calculation of probability to choose the next node will also increase when the ACO is faced with multiple number of iteration within the environment. Another reason is the increase in multiple adjacent nodes to be traversed by ants. This then will increase the number of possible nodes to goal thus causing the ACO to require additional time and additional number of iterations to converge as shown in table VI.

C. Comparative study of GA and ACO in different complexity of free space nodes

Table VII below shows the performances of GA and ACO. As shown, the time and number of iterations for both GA and ACO increases as the number of feasible nodes increases (increase complexity). ACO seems to perform better. ACO has the robustness of the optimization algorithm compared to GA for this RPP purpose where it can still maintain the performances of time and number of iteration it takes to find optimal path in three different complexity of the environment as shown at Fig. 12 and Fig. 13.

TABLE VII
PERFORMANCES OF GA AND ACO IN DIFFERENT COMPLEXITY OF ENVIRONMENT

Number of nodes	12 nodes		22 nodes		63 nodes	
Algorithm	GA	ACO	GA	ACO	GA	ACO
Time (sec)	72.1244	9.337	242.7088	32.566	2144.84	814.819
No. of Iteration	7.1	1.9	9.5	3.1	21.8	3.5

One of the main reasons for ACO's good performance is due to the behavior of the algorithm i.e. how it works to initialize the population. The way GA initialize the population is based on random approaches where the next node to be visited is randomly chosen from an adjacent feasible node. Due to this random process, the algorithm need to go through the process of selection and some other process to choose the optimal node which will contribute to increment of time and number of iterations especially when the number of nodes increases. It contrast, the ACO utilize the state transition rules which is efficient and enables it to skip to the process of finding the optimal path thus reducing the time and number of iterations required to find the optimal path in either a simple environment or complex environment.

In addition, the quality of population in each generation will also influence algorithm performances. In ACO, the global and local updating approaches is efficient because it will improve the number of optimal path in each generation, this will make the solution faster with fewer number of iteration. However, this differs with GA where GA will carry the good population to produce the next child in the next generation however there is no guarantee to improve the solution rapidly as the child is produced in random approach. Thus the child may come from good categories or worst categories which will then influence the population of the next generation. Hence, this will affect the time and number of iterations that GA takes to find the solution will be greater than ACO's.

Furthermore, when the number of nodes increases, the size of chromosomes length also need to be adjusted based on the requirement in each case. Small number of nodes only needs a small number of lengths to allocate the paths within the chromosomes while the complex numbers of nodes will need a complex number of chromosomes. The increment of length will affect the whole process of GA to find optimal path because the population of the next child in each generation will be produced based on the cross over and mutation process. During this process, the point to be cross and mutate will be determined randomly based on the length of the chromosomes. The more the length, the more possibility of GA to have a variety of population which will cause the process of finding the optimal path more challenging and simultaneously will contribute to the increment of time and number of iteration. It is different with ACO where the length of chromosomes size will not influence the next ant's population as it is based on heuristic and pheromone value carried by the previous ants. Therefore, length will increase the time ants need to traverse from one node to another node and not influences the next node to be traversed by the ants.

With the increment of length usually GA also need to increase the population in order to get the optimal path. This will cause the process of GA to find path to become slower. In contrast with GA, for ACO, it is not necessary to increase the population because it will not affect the process. Thus this helps ACO to minimize the time and number of iterations.

Based on the reason and the results obtained above, it can be concluded that ACO is practical to be use either in small or complex number of nodes compared to GA. ACO can find the optimal path and satisfy the optimization criteria at a faster rate than GA for all environment type.

In addition, it is easier to set the parameters for ACO compared to GA. In ACO, only the length needs to be changed in each case, however for GA, when the environment complexities are changed, there is a need to ensure the balancing of the value of population, length and convergence criteria. From here it shows that the applications of ACO algorithm in RPP can be expanded by applying this algorithm in any type of global map that consists of multiple numbers of nodes, i.e. global map created with complex obstacle environment, global map created in a large indoor or outdoor environment etc. ACO can also be used to optimize not only global paths but also local paths. However, although GA is not as robust as ACO in optimizing the solution of RPP problem, it can still be used in applications that do not require fast solution as GA is able to provide solutions to path planning in complex environments but at a slower rate.

VI. CONCLUSIONS

GA and ACO were successfully implemented to find optimal path that satisfies the optimization criteria in the global static environments. Both algorithms performances were evaluated to determine the effectiveness of both algorithms when it is applied in global static environment of different complexities. Findings from this research proved that ACO performance in terms of speed is much faster compared to GA while the number of iteration required is less for ACO compared to GA in each different complexity of feasible nodes in the global maps. In addition, the adjustment of ACO parameters to adapt to environments of different complexities of feasible nodes is also much easier compared to GA. Advantages and limitations of both algorithms can be further explored to expand the applications of both optimization algorithms in robot path planning research area.

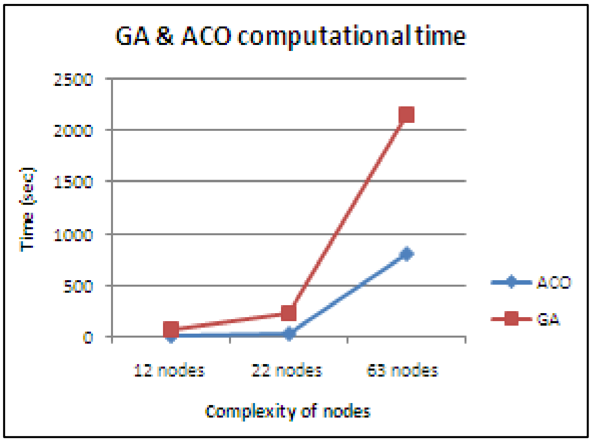

Fig. 12. GA & ACO computation time

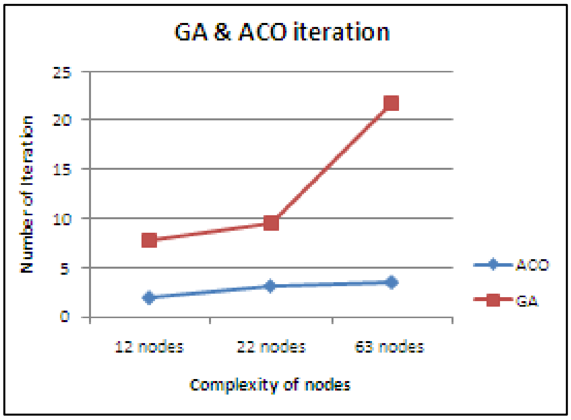

Fig. 13. GA & ACO iteration

ACKNOWLEDGMENT

The authors gratefully acknowledge University Teknologi MARA (UiTM) for supporting this research.

REFERENCES

[1] Charles.W.Warren, "Fast path planning using modified A* Method," *IEEE Transactions on Systems,Man and Cybernetics*, pp. 662-667, 1993.

[2] J. Tu, X.Yang.S, "Genetic Algorithms Based Path Planning for a Mobile Robot," *Proceedings of the 2003 IEEE International Conference on Robotics & Automation, Taipei, Taiwan*, pp. 1221-1226, 2003.

[3] C. Alexopoulos and P. M. Griffin, "Path planning for a mobile robot," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, pp. 318-322, 1992.

[4] G. Nagib and W. Gharieb, "Path planning for a mobile robot using genetic algorithms," *International Conference on Electrical, Electronic and Computer Engineering, 2004 (ICEEC '04)*, pp. 185-189, 2004.

[5] C. H. H. Du Xin, Gu Wei Kang, "Neural Network and Genetic Algorithm Based Global Path Planning in A Static Environment," *Journal of Zhejiang University Science*, vol. 6A, pp. 549-554, 2005.

[6] Y. R. Hu, Simn X.Yang, "A Knowledge Based Genetic Algorithm for Path Planning of Mobile Robot," *Proceedings of the 2004 IEEE International Conferences on Robotics and Automation New Oriens*, pp. 4350-4355, 2004.

[7] T. Guan Zheng, He Huan,Sloman Aaron, "Ant Colony System ALgorithm for Real Time Globally Optimal Path Planing of Mobile Robots," *Proceedings of ACTA AUTOMATICA SINICA*, vol. 33, pp. 279-285, 2007.

[8] H. Mei, Yantao Tian, Linan Zu, "A Hybrid Ant Colony Optimization Algorithm for Path Planning of Robots in Dynamic Environment," *International Journal of Information Technology*, vol. Vol 12,No 3, pp. 78-88, 2006.

[9] C. Tsai-Yu, K. Szu-Wen, and J. Y. J. Hsu, "A two-phase navigation system for mobile robots in dynamic environments," *Proceedings of the IEEE International Conference on Intelligent Robots and Systems '94 (IROS '94)*, vol. 1, pp. 306-313, 1994.

[10] R. A.Brooks, "Solving the find path problem by good representation of free space," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 13(2), pp. 190-197, 1983.

[11] A. Stentz, "Optimal and efficient path planning for partially-known environments," presented at Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on, 1994.

[12] A. Zelinsky, "A mobile robot exploration algorithm," *IEEE Transactions on Robotics and Automation*, vol. 8(6), pp. 707-717, December 1992.

[13] A. Zelinsky , Shinichi Yuta, "A unified approach to planning, sensing and navigation for mobile robots," *Third International Symposium on Experimental Robotics Kyoto, Japan*, pp. 28-30, Oct 1993.

[14] N. Sariff, Norlida Buniyamin, "An Overview of Autonomous Robot Path Planning Algorithms," *4th Student Conference on Research and Development (Scored 2006), Shah Alam, Selangor*, pp. 184-188, June 2006.

[15] O. Khatib, "Real time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5(10), pp. 90-98, 1985.

[16] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation* vol. 16, pp. 615-620, 2000.

[17] J. Barraquand, Bruno Langlois,Jean Claude Latombe, "Numerical potential field techniques for robot path planning," *IEEE Transactions of Systems, Man and Cybernetics*, vol. 22(2), pp. 224-241, 1992.

[18] Y. Koren, J.Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," *IEEE Proceedings of Conference Robotics and Automation*, pp. 1398-1404, 1991.

[19] Z. Ying, W. Cheng-Dong, and L. Meng-Xin, "Rough set and genetic algorithm in path planning of robot," presented at Machine Learning and Cybernetics, 2003 International Conference on, 2003.

[20] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, pp. 100-107, 1968.

[21] K. Chin, Fan and L. Po-Chang, "Solving find path problem in mapped environments using modified A* algorithm," *IEEE Proceedings of Systems, Man and Cybernetics, IEEE Transactions on*, vol. 24, pp. 1390-1396, 1994.

[22] A. Stentz, "Optimal and Efficient Path Planning for Partially-Known Environments," *In Proceedings IEEE International Conference on Robotics and Automation*, pp. 1-8, 1994.

[23] S. Singh, Reid Simmons, Trey Smith, Anthony Stentz, Vandi Verma, Alex Yahja & Kurt Schwehr, "Recent Progress in Local and Global Traversability for Planetary Rovers," *In Proceedings, IEEE Conference on Robotics and Automation*, pp. 1-7, 2000.

[24] A. Yahja, Sanjiv Singh, Anthony Stentz, "An efficient on-line path planner for outdoor mobile robot," *Journal of Robotics and Autonomous Systems* vol. 32, pp. 129-143, 2000.

[25] M. Netnevitsky, *Artificial Intelligence:A guide to intelligent Systems*: 2nd Edition, Addison-Wesley, 2002.

[26] J. Stuart, R. Norvig,P. Norvig, *Artificial Intelligence - A Modern Approach*: Prentice Hall Series, 1995.

[27] S. N.Sivanandam, N.Deepa S, *Introduction to Genetic Algorithms*, vol. S.N.Sivanandam, S.N.Deepa: Springer-Verlag Berlin Heidelberg 1-425, 2008.
[28] D. E.Goldberg, "Genetic and evolutionary algorithms come on age," *Communications ACM*, vol. 37, pp. 113-119, 1994.
[29] P. J. F. Fleming, "Genetic algorithms in control systems engineering: a brief introduction," *IEE Colloquium on Genetic Algorithms for Control Systems Engineering*, pp. 1-5, 1993.
[30] M. Dorigo and L. M. Gambardella, "Ant colony system: A Cooperative Learning Approach to the Traveling Salesman Problem(TSP)," *Evolutionary Computation, IEEE Transactions on*, vol. 1, pp. 53-66, 1997.
[31] Y. Qing Qin, De Bao Sun, Ning Li and Yi Gang Cen, "Path Planning For Mobile Robot Using The Particle Swarm Optimization With Mutation Operator," presented at Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 2004.
[32] M. Plucinski, *Methods of Artificial Intelligence and Intelligent Agents (Chapter 3: Application of the ant colony algorithm for the path planning)*: Wiley.Inc, 2004.
[33] X. Chen, Yangmin Li, "Smooth Path Planning of a Mobile Robot Using Stochastic Particle Swarm Optimization," *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation*, pp. 1727-1727, 2006.
[34] K. H.Sedighi, Kaveh Ashenayi, Theodore W.Manikas, "Autonomous Local Path Planning for a Mobile Robot Using Genetic Algorithm," *Proceeding 2004 IEEE Congress on Evolutionary Computation (CEC 2004)*, pp. 1338-1345, 2004.
[35] K. Sugihara, Smith.J, "GA for adaptive motion planning of an autonomous mobile robot," *Proc. 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97)*, pp. 138-143, 1997.
[36] J. Tu, Simon X.Yang, "Genetic Algorithms Based Path Planning for a Mobile Robot," *Proceedings of the 2003 IEEE International Conference on Robotics & Automation, Taipei, Taiwan*, pp. 1221-1226, 2003.
[37] R. Ramakrishnan and S. Zein-Sabatto, "Multiple path planning for a group of mobile robot in a 2-D environment using genetic algorithms," *IEEE Conference Proceedings SoutheastCon 2001*, pp. 65-71, 2001.
[38] H. E. Neda Shahidi, Marziye Abdollahi and Caro Lucas, "Memetic Algorithm Based Path Planning For a Mobile Robot," *International Journal of Information Technology Volume 1 Number 4 2004*, pp. 174-177, 2004.
[39] X. Hu, Cunxi Xie, "Niche Genetic Algorithm for Robot Path Planning," *Third International Conference on Natural Computation (ICNC 2007)*, pp. 1-5, 2007.
[40] L. Gengqian, L. Tiejun, P. Yuqing, and H. Xiangdan, "The ant algorithm for solving robot path planning problem," presented at Information Technology and Applications, 2005. ICITA 2005. Third International Conference on, 2005.
[41] Z. Dongbin, Yi Jianqiang, "Robot Planning with Ant Colony Optimization Algorithms," *Proceedings of the 25th Chinese Control Conference*, pp. 1460-1465, 2006.
[42] F. Gholami and M. J. Mahjoob, "An Investigation of Parameters in Ant Colony Optimization for a Path Optimization Algorithm," *Proceeedings of the 2007 IEEE International Conference on Mechatronics and Automation*, pp. 463-468, 2007.
[43] N. Sariff, Norlida Buniyamin, "Comparative Study of Genetic Algorithm and Ant Colony Optimization Algorithm in Global Static Environment of Different Complexities," *2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation(CIRA 2009), Daejeon, Korea*, pp. 132-137, 15th -18th December 2009.
[44] L. Shuhua, T. Yantao, and L. Jinfang, "Multi mobile robot path planning based on genetic algorithm," presented at Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on, 2004.
[45] M. D. a. T. Stutzle, *Ant Colony Optimization*: The Bradford Book,The MIT Press Cambridge.Masachusetts, London, England.pp 1-305., 2004.

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

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), U.K. She is now an Associate Professor at UiTM and a Fellow of the Institution of Engineers, Malaysia. Her current research interest is in Industrial Automation and Robotics, Knowledge Management and Engineering Education.