Solving an Agricultural Robot Routing Problem with Binary Particle Swarm Optimization and a Genetic Algorithm

Mohd Saiful Azimi Mahmud, Mohamad Shukri Zainal Abidin, Zaharuddin Mohamed
Department of Control and Mechatronics, School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia Skudai,
Johor, Malaysia.
Email: msazimi2@live.utm.my, shukri@utm.my, zahar@fke.utm.my

***Abstract*—Agricultural robotics has become increasingly popular among agricultural researchers as an alternative to the use of human workers in the future. However, the operational cost of agricultural mobile robots must be competitive with the cost of hiring human workers. In agricultural mobile robot navigation, it is difficult to determine an optimized sequential route with a minimal distance. This paper employs binary particle swarm optimization (PSO) and a genetic algorithm (GA) to find the shortest routing path for spraying operations in a greenhouse. The agricultural robotics routing problem has been expressed in terms of the traveling salesman problem, which is commonly used in operational research. To solve the routing problem, an objective of a total path length was measured based on the path computed using a probabilistic roadmap path planner. The results indicated the performance of the GA was better for solution quality and computational time, while binary PSO performed better with respect to convergence time.**

***Index Terms*—agriculture, particle swarm, genetic algorithm, routing**

I. INTRODUCTION

In the past decade, enormous technological changes have occurred in the field of agricultural robotics. Tasks in agriculture, such as cultivation [1], inspection [2], spraying [3], transplanting [4], and selective harvesting [5] has been conducted by using mobile robot application. Therefore, autonomous navigation for agricultural robotics plays an important role in minimizing operational costs by reducing the number of agricultural workers.

A 4-wheel autonomous robot was developed in [6] that initialized the navigation system by collecting data stored in nodes that were distributed in a vineyard. It used a predefined way-point route between the grapevine rows to evaluate the sensor node location. Then, the recorded and georeferenced received signal strength indicator (RSSI) was used for analyzing and mapping, to create a new route for the mobile robot. This method required the consideration of the nodes in the vineyard to determine the optimal route with minimum travel distance.

Optimization methods such as particle swarm optimization (PSO) were applied in [7] to solve a sugarcane harvester routing problem. That study expressed the harvester's routing problem in terms of the traveling salesman problem (TSP) to minimize the objective function. The PSO-based algorithm employed in [7] demonstrated an improvement in minimizing travel distance and maximizing the amount of harvested sugarcane. From these previous studies, the application of an optimization algorithm to solve agricultural problems has been shown as an effective method for minimizing operational costs associated with certain agricultural tasks.

Many algorithms have been applied to the TSP, such as binary PSO (binary PSO) [8-9], genetic algorithms (GAs) [10-12], tabu searches (TSs) [13-14] and ant colony optimization (ACO) [15-16], with binary PSO and GAs being the most common for solving the TSP. Binary PSO is a modified version of basic PSO that has been modified for the TSP, as the basic PSO can only solve continuous-type problems, whereas the TSP is a discrete-type problem. PSO is also known for its rapid convergence, and was thus selected for this study.

Several improvements have been made in the application of GAs for the TSP, such as modification of the crossover and mutation operator [10-11] and hybrid algorithm [12]. Some improvements have resulted in increased computational time due to incremental increases in computational complexity. Despite all the improvements that has been made, the basic GA with a simple crossover is sufficient to solve the TSP problem with a single objective. The high-quality solutions obtained with GAs encourage researchers to apply them to many optimization problems.

This paper presents the application of binary PSO and a GA to solve the agricultural mobile robot routing problem. They are used to minimize the objective function that represents the total distance of the route. The difference between this study and previous applications to the TSP is that the directed graph representation is replaced by real routes generated by a probabilistic roadmap. The performance of each optimization algorithm was compared based on convergence time, solution quality, and computational time.

doi: 10.18178/ijmerr.7.5.521-527

II. Materials and Methods

A. Problem Overview

In agriculture, fungicide spraying is necessary to preserve the health of plants, and the mobile robot must travel to the selected crop with a minimum travel cost. Typically, the mobile robot travels based on the crop rows. However, there are some agricultural environments that divide the row into several different sections to identify shortcuts for the mobile robot path. Fig. 1 depicts the use of row division to account for shortcuts.

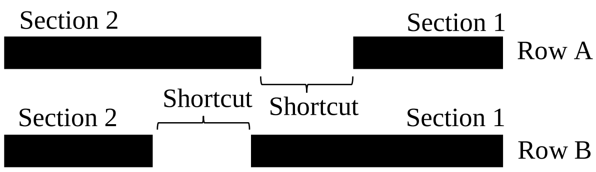

Figure 1. Row division in crop rows

Based on Fig. 1, rows A and B have been divided into two different sections. To minimize the travel distance for the traveling route, the row-based traveling method was not effective, as shortcuts and available to travel between rows. To solve the robot routing problem for spraying operations, the optimization method is used to minimize the agricultural robot's travel distance.

Fig. 2 shows the process used to solve the mobile robot routing problem. In Fig. 2, the process begins by initializing the crop points, and a path planning process is then executed using a probabilistic roadmap. Thereafter, the fitness function for the total distance traveled is calculated, and the fitness function is optimized using binary PSO and a GA, respectively.

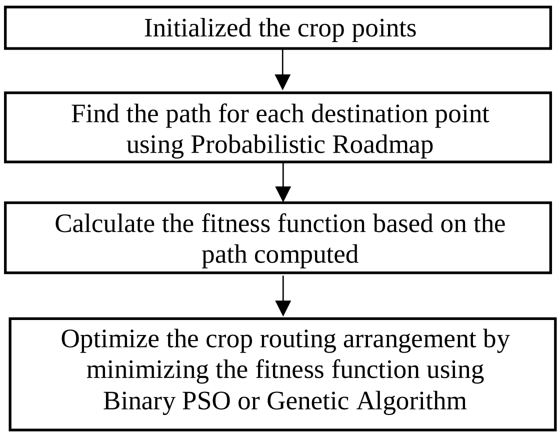

Figure 2. Process in generating optimum travel distance

To initialize the crop point, the farmer observed and identified the selected crops that had been affected by disease. In this paper, a case of Powdery Mildew has been used, which is a disease that does not critically affect plant condition because some nutrients can be absorbed from the plant if it is left untreated, though the quality of the fruit or vegetable does eventually suffer [17]. To avoid the spreading of the disease via wind, the affected area must be treated using an organic fungicide [18]. Therefore, a mobile robot was used to spray the fungicide, as this process can present a significant threat to human health due to the poisonous nature of the chemicals.

Probabilistic roadmaps can be used to solve motion planning problems [19]. The process of computing the path consists of two different phases: a learning phase and a query phase. The roadmap was formed in the C-space of the robot and was stored as an undirected graph R in the learning phase [20]. A random free configuration was then generated and added to a node, N. For every new node, many nodes from the current N node were selected, and each of them was connected using a local planner. To increase its distance from c, a new node, Nc attempts to connect to c.

In the query phase, paths were found between the input start and goal configuration using a roadmap that was constructed during the learning phase. When the start configuration s and the goal configuration g was computed, s and g will attempt to connect with the two nodes of R, $\tilde{s}$ and $\tilde{g}$, using feasible paths, P_s and P_g. The feasible path addresses obstacle avoidance using Dijkstra's search algorithm.

A detailed explanation of the fitness function and constraints is presented in Section B. The framework used for binary PSO is discussed in Section C, and the framework for the GA is discussed in Section D.

B. Model Formulation

In this paper, the simulation environment was generated by taking exact measurements based on the real greenhouse environment shown in Fig. 3.

Figure 3. Real greenhouse environment

The real environment was then redesigned in SolidWorks and then simulated in MATLAB using Simulink3D animation [21]. Fig. 4 presents an aerial view of the generated simulation environment. A binary occupancy grid was then computed based on the aerial image in Fig. 4 to differentiate the obstacles and free space throughout the environment.

Fig. 5 shows the generated binary occupancy grid for the environment where the white area indicates free space and the black area indicates obstacles that include the selected crops that need to be treated. The coordinates for each point sequence are presented in Table 1.

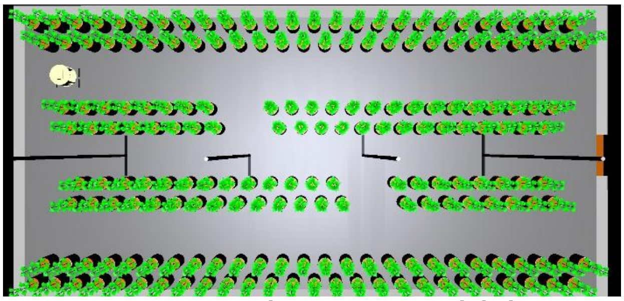

Figure 4. Generated environment in Simulink3d

Based on Fig. 5, the coordinate number was arranged and labeled based on the row sequence, where the red dot indicates the destination point (node) that must be reached by the mobile robot.

The binary occupancy grid $G \in \{V, E\}$ in Fig. 5 consist of a set of nodes $V = \{1,2, \dots, n\}$ and a set of edges E. The node in Fig. 5 represents the crop destination point that must be included in the spraying operation. The edges, E in a binary occupancy grid represent the path computed using a probabilistic roadmap between each destination point. A non-negative number of d_{ij} represent the distance between each point between nodes i and j.

In this paper, only one objective was considered for the fitness function. The fitness function for the total route distance is expressed:

$$f(x) = min\left\{\sum_{i,j \in A_{ij}} d_{ij}\right\} \quad (1)$$

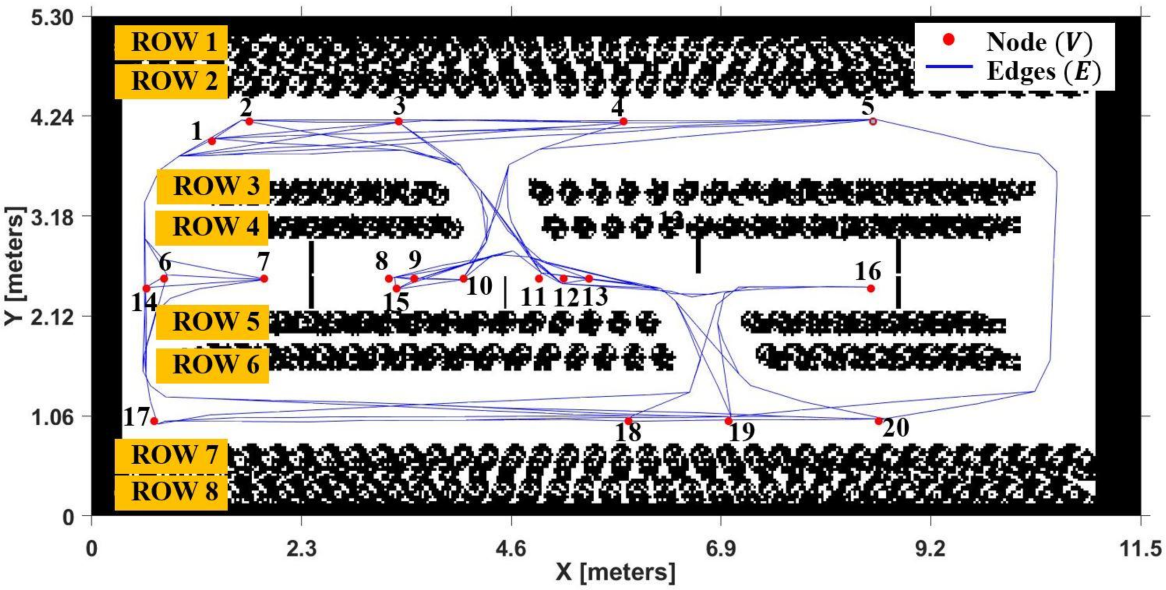

Figure 5. Binary occupancy grid generated

where A_{ij} is the set of edges between nodes i and j. The total distance between the sequence of nodes is calculated based on the generated path from a probabilistic roadmap. The usual approach to solving the TSP employs a directed graph, but is not applicable to the problem in this study due to the obstacle avoidance operation that is not included in the directed graph method. Therefore, a probabilistic roadmap is used to generate a path without colliding with any obstacles.

TABLE I. COORDINATE SEQUENCE AND LABEL

Node (n)	Coordinate (x, y)
1	(1.44, 4.00)
2	(1.89, 4.24)
3	(3.69, 4.24)
4	(5.98, 4.24)
5	(9.05, 4.24)
6	(0.87, 2.67)
7	(2.07, 2.67)
8	(3.57, 2.67)
9	(3.87, 2.67)
10	(4.47, 2.67)
11	(4.87, 2.67)
12	(5.17, 2.67)
13	(5.47, 2.67)
14	(0.66, 2.37)
15	(3.67, 2.37)
16	(9.05, 2.37)
17	(0.75, 1.04)
18	(6.05, 1.04)
19	(7.05, 1.04)
20	(9.15, 1.04)

For the purposes of this study, the constraint is expressed as follows:

$$x(1) = x(end) = 1 \quad (2)$$

where the constraint represents the requirement to navigate starting from node 1, then return to node 1 after completing the task. The end notation indicates the final sequence for the path in the generated route.

C. Binary PSO

A bio-inspired PSO approach was proposed by [22], which consisted of the population-based algorithm that performed a parallel search on a solution space. It is well-known as a great optimization algorithm used to solve continuous-type problems. However, to solve a discrete-type problem, adaptation of the original PSO is required and has been proposed by others [23-25].

In this paper, a modified version of binary PSO proposed by [26] was used to solve the mobile robot routing problem. In binary PSO, each particle represents its position in binary values which are 0 or 1. Then, each particle will then mutate from 1 to 0 or 0 to 1, depending on its velocity. The binary PSO steps used in this paper can be summarized as follows:

1. Initialize the swarm X_i.The position of particles is randomly initialized and the elements of $X_i$are selected randomly based on permutation arrangement in maximum number of node n.
2. Decode the sequence of particle elements from binary to real numbers.
3. Evaluate the performance F for each particle using the current particle position.
4. Encode the particle into a sequence of binary number.
5. Compare the performance of each particle to its best performance so far, P_{ibest}:
   If$F\left(X_i(t)\right) < F(P_{ibest})$
   $F(P_{ibest}) = F\left(X_i(t)\right)$
   $P_{ibest} = X_i(t)$
6. Compare the performance of each individual to the global best particle, P_{gbest}:
   If$F\left(X_i(t)\right) < F\left(P_{gbest}\right)$
   $F\left(P_{gbest}\right) = F\left(X_i(t)\right)$
   $P_{gbest} = X_i(t)$
7. Change the velocity of the particle based on equation:
   $v_i(t+1) = wv_i(t) + c\varphi_1\left(p_{ibest} - x_i(t)\right) + s\varphi_1\left(p_{gbest} - x_i(t)\right)$ (3)
8. Move each particle to a new position using equation:
   $$x_i(t+1) = \begin{cases} 1 & \textbf{If } v_i(t+1) = max(v_i(t+1)) \\ 0 & \text{otherwise} \end{cases}$$
   (4)
9. Go to step 2 and repeat until maximum iteration, I_{max} occurs.

Based on the steps for implementing PSO,φ_1 is the inertia weight, c represents the cognitive and s is the social coefficient. The algorithm used in this study was modified from the original version proposed in [26] by adding some particle encoding, decoding, and constraints. Particle encoding and decoding are needed to represent the particle as a binary and real number for particle updating and fitness evaluation, respectively. Element checking is also conducted to avoid any repeating destinations during particle updating.

D. Genetic Algorithm

The second algorithm used in this paper is a GA. This algorithm consists of three different phases in the search mechanism: evaluation of the fitness in each chromosome, selection of the parent chromosomes, and lastly the mutation and recombination operator for the parent chromosome to produce offspring. Steps used to employ a GA in solving the routing problem can be summarized as follows:

1. Create an initial population of chromosomes (generation 0) that consists of parent's population containing permutation of maximum number of n nodes.
2. Evaluate the fitness of each chromosome (parent).
3. Randomly select two parent chromosomes from the current population using a fitness cost sorting.
4. In crossover operation, exchange the first two route arrangement inside the first parents' chromosomes using a modified crossover to create the first offspring.
5. Repeat step 4 by exchange the first two route arrangement inside the second parents' chromosomes to create the second offspring.
6. Fill in the remaining chromosome structure in the offspring that has not been filled in crossover.
7. Repeat steps 4 and 5 until all parents (N_p) and all offspring (N_{off}) are created.
8. Replace the old population of chromosomes with the new one.
9. Evaluate the fitness of each chromosome in the new population (parents and offspring).
10. Sort the chromosomes (parents and offspring) based on the fitness function cost and select the best of them.
11. Return to Step 2and repeat until maximum iteration, I_{max} occur.

In the steps provided, the crossover type used in this implementation is the modified crossover designed in [27]. The modified crossover operation is illustrated in Fig. 6.

Parent 2	5	7	2	6	1	4	3
Parent 2	2	1	4	6	5	3	7
Offspring 1	5	7	4	6	2	3	1
Offspring 2	2	1	5	6	7	4	3

Figure 6. Modified crossover mutation

In this method, a cut position is chosen at random on the first parent chromosome. In this case, the first two chromosomes are selected (gray box). Then, an offspring will be created by appending the second parent chromosome to the initial segment in the first parent. The duplicates number occurs in the offspring will be replaced randomly with the number, which is not included in the chromosome (light blue box).

III. Results and Discussion

This section presents results in terms of a comparison between the implementation of binary PSO and GA to solve the robot routing problem for spraying application in agriculture. The comparisons were conducted to select the most applicable algorithm for the agricultural spraying operation.

A. PSO and Genetic Algoritm Parameter Setup

To ensure the parameters being used in this experiment would provide an accurate and correct result, data samples were used with binary PSO and the GA validate good results with minimal computational cost. Table II shows the parameter used in Binary PSO, and Table III shows the parameter used in the GA, respectively.

TABLE II. Parameter used in Genetic Algorithm

Parameter	Value
Maximum iteration(I_{max})	200
Number of chromosomes (Parents) (N_p)	500
Number of chromosomes (Children) (N_{off})	400
Type of crossover	non-uniform

TABLE III. Parameter used in Binary PSO

Parameter	Value
Maximum iteration (I_{max})	200
Number of Particles(X_i)	100
Social coefficient(S)	3.5
Cognitive coefficient (C)	0.5

Based on Table II and III, the parameters were selected by executing the algorithm repeatedly to solve the proposed problem until satisfactory results were obtained.

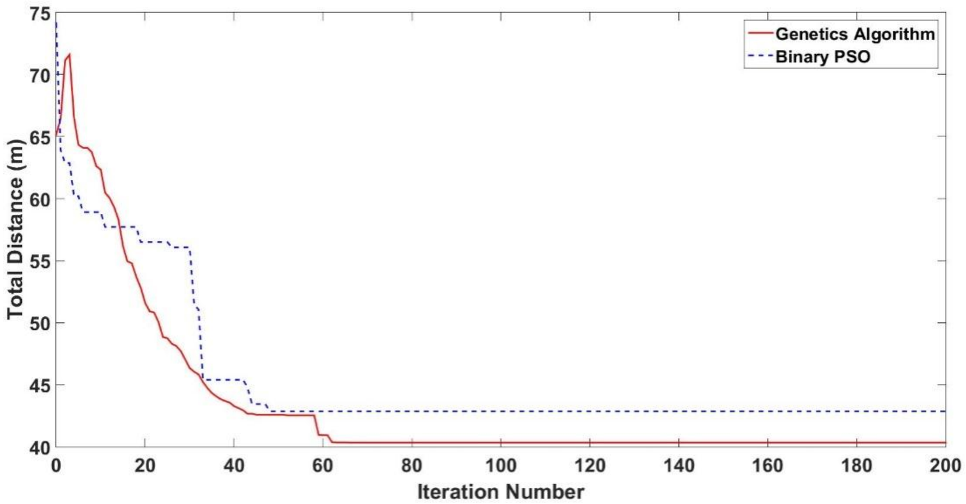

Figure 7. Total distance cost in each iteration

B. Performance Comparison

In this section, a performance comparison between the application of Binary PSO and the GA was conducted based on convergence time, computational time, and solution quality. Fig.7 shows the response for the total distance cost in every iteration for binary PSO and the GA, respectively.

As Fig. 7 indicates, the GA with a modified crossover provided better solution quality with a total distance of 40.34 m compared to binary PSO which had a total distance of 42.87 m. However, in terms of convergence time based on iteration number, binary PSO displayed a better convergence time at the 48th iteration, and the GA converged at the 66th iteration. Table IV provides the computational time between the algorithms execution.

TABLE IV. Computational Time Comparison

Algorithm Implementation	Computational time (s)
Genetic Algorithm	27
Binary PSO	76

Based on Table IV, application of Binary PSO to solve an agricultural mobile robot routing problem offers a longer computational time compared to the GA, with values of 76 and 27, respectively. The difference of 49 seconds in computational time is indicative of binary PSO's higher level of computational complexity compared to the GA. The higher computational time of binary PSO was primarily due to the encoding and decoding process in the algorithm. In binary PSO, particle encoding is necessary to encode the crop routes in binary values. Then, particle decoding is needed to decode the binary value into the crop sequence as a real number to calculate the total distance. Both particle encoding and decoding was necessary for each iteration, which led to a longer computational time for the execution.

In the application of the GA, the delay in total distance convergence as if calculated the optimal was due to the randomness of the modified crossover. In modified crossover, the chromosomes are exchanged by the selected part to enhance the search for an optimal solution, although it simultaneously degrades the exploitation of an optimal solution.

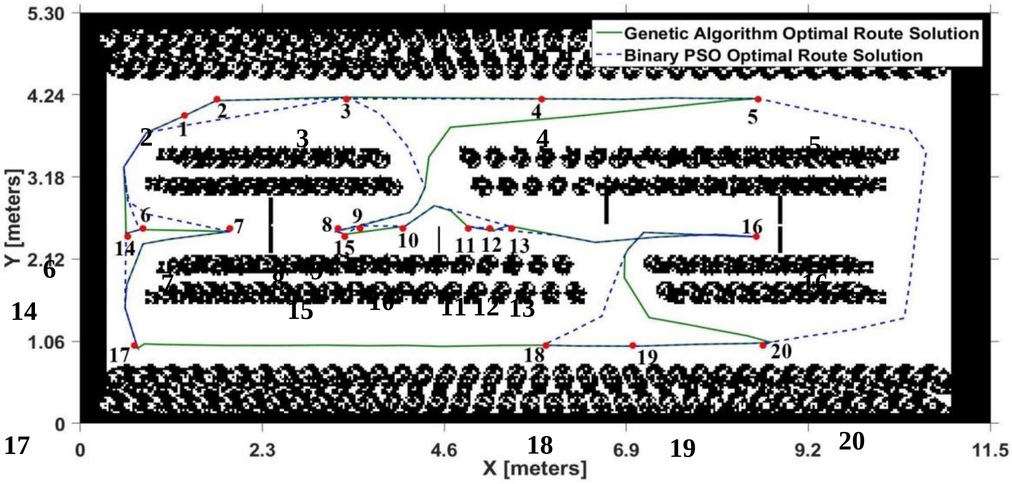

Figure 8. Optimal Computed Route in the Greenhouse

Thus, the search method in modified crossover computed as part of the GA provided a better result with respect to solving the mobile robot routing problem. Fig. 8 shows the computed path based on the optimal crop routes that was found with both algorithms, and Fig.9 shows the optimal crop routes found with binary PSO and the GA, respectively.

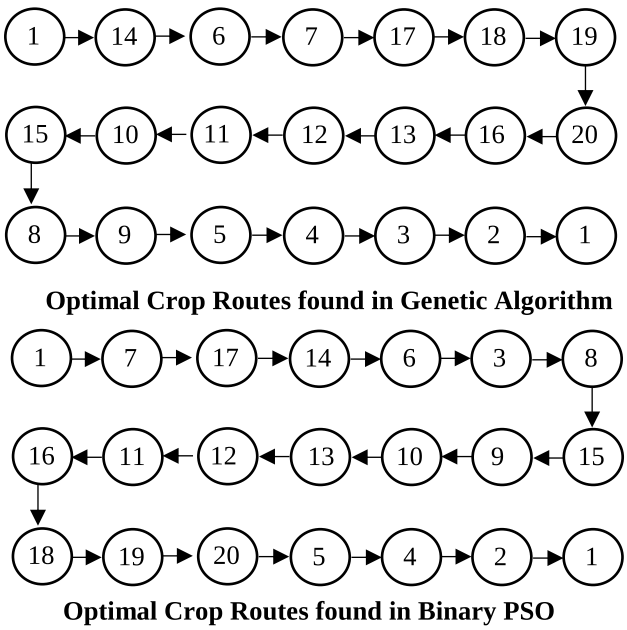

Figure 9. Optimal Crop Route Sequence

Based on Fig. 8, it is clear that the optimal route that was selected by binary PSO, which covered a larger area compared to that of the GA, which is consistent with the results found in Fig. 7, where the total distance of the route found by the GA was lower than the route found by binary PSO.

IV. Conclusion

In this paper, a comparison between the performance of binary PSO GA application was conducted to solve a single-objective agricultural mobile robot routing problem. The routing problem was formulated based on the traveling salesman problem, which is commonly used in operational research. The evaluation has been conducted based on convergence time, solution quality, and computational time. Use of the GA to solve the agricultural mobile robot routing problem demonstrated better performance, and found an optimal solution with respect to distance traveled and computational time. Therefore, this approach has significant value as a method for solving the mobile robot routing problem in agriculture with an optimal solution.

Acknowledgment

The authors are grateful to the Universiti Teknologi Malaysia, the Ainuddin Wahid Scholarship and the Ministry of Higher Education (MOHE), for their partial financial support through their research funds, Vote No. R.J130000.7823.4F759.

References

[1] B. S. Shivaprasad, M. N. Ravishankara, B. N. Shoba, “Design and implementation of seeding and fertilizing agriculture robot,” *International Journal of Application or Innovation in Engineering & Management*, vol. 4, no. 6, pp. 251-255, June 2014.

[2] N. Shalal, T. Low, C. McCarthy, N. Hancock, “Orchard mapping and mobile robot localization using on-board camera and laser scanner data fusion – Part B: Mapping and localization,” *Computers and Electronics in Agriculture*, vol. 113, pp. 267-278, November 2015.

[3] G. Zaidner and A. Shapiro, “A novel data fusion algorithm for low-cost localization and navigation of autonomous vineyard sprayer robots,” *Biosystems Engineering (Advances in Robotic Agriculture for Crops)*, vol. 146, pp. 133-148, May 2016.

[4] A. Gallakota, "Agribot- A multipurpose agricultural robot," presented at the 2011 Annual IEEE India Conference, Hyderabad, India, December 16-18, 2011.
[5] V. F. Tejada, M. F. Stoelen, K. Kusnierek, N. Heiberg, A. Korsaeth, "Proof-of-concept robot platform for exploring automated harvesting of sugar snap peas," *Precision Agriculture*, vol. 18, no. 6, pp. 952-972, December 2017.
[6] D. Reiser, D. S. Paraforos, M. T. Khan, H. W. Griepentrog, M. Vazquez-Arellano, "Autonomous field navigation, data acquisition and node location in wireless sensor networks," *Precision Agriculture*, vol. 18, no. 3, pp. 279-292, June 2017.
[7] K. Sethanan and W. Neungmatcha, "Multi-objective particle swarm optimization for mechanical harvester route planning of sugarcane operation," *European Journal of Operational Research*, vol. 252, pp. 969-984, January 2016.
[8] X. H. Shi, Y.C. Liang, H.P. Lee, C. Lu, Q. X. Wang, "Particle swarm optimization-based algorithms for TSP and generalized TSP," *Information Processing Letters*, vol. 103, pp. 169-176, March 2007.
[9] D. C. Dang, R. N. Guibadj, A. Moukrim, "An effective PSO-inspired algorithm for the team orienteering problem," *European Journal of Operational Research*, vol. 229, no. 2, pp. 332-344, September 2013.
[10] N. Mohd Razali, "An efficient genetic algorithm for large scale vehicle routing problem subject to precedence constraints," *Procedia- Social and Behavioral Sciences*, vol. 195, pp. 1922-1931, July 2015.
[11] F. Liu and G. Zheng, "Study of genetic algorithm with reinforcement learning to solve the TSP," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6995-7001, April 2009.
[12] S. Maity, A. Roy, M. Maiti, "An intelligent hybrid algorithm for 4- dimensional TSP," *Journal of Industrial Information Integration*, vol.17, pp. 39-50, March 2017.
[13] H. Jia, Y. Li, B. Dong, H. Ya, "An improved tabu search approach to vehicle routing," *Procedia- Social and Behavioral Sciences*, vol. 96, pp. 1208-1217, November 2013.
[14] O. Pedro, R. Saldanha, R. Camargo, "A tabu search approach for the prize collecting travelling salesman problem," *Electronic Notes in Discrete Mathematics*, vol. 41, pp. 261-268, June 2013.
[15] M. L. Ibanez and C. Blum, "Beam-ACO for the travelling salesman problem with time windows," *Computers & Operations Research*, vol. 37, no.9, pp. 1570-1583, September 2010.
[16] J. Yang, X. Shi, M. Marchese, Y. Liang, "An ant colony optimization method for generalized TSP problem," *Progress in natural Science*, vol. 18, no.10, pp. 1417-1422, November 2008.
[17] Y. Lin, P. Ruiliang, Z. Jingcheng, W. Jihua, Y. Hao, "Using high spatial resolution satellite imagery for mapping powdery mildew at a regional scale," *Precision Agriculture*, vol. 17, no. 3, pp. 332-348, November 2015.
[18] J. C. Zhang, R. L. Pu, J. H. Wang, W. J. Huang, L. Yuang, J. H. Luo, "Detecting powdery mildew of winter wheat using leaf level hyperspectral measurements," *Computers and Electronics in Agriculture,* vol. 85, pp. 13-23, July 2012.
[19] E. L. Kavraki, P. Svetska, J. C. latombem H. M. Overmars, "Probabilistic roadmap for path planning in high dimensional configuration spaces," *IEEE Transaction on Robotics and Automation*, vol. 12, no. 4, pp. 566-579, August 1996.
[20] C. Siddharta, K. Vladlen, "Smoothed analysis of probabilistic roadmaps," *Computational Geometry*, vol. 42, pp. 731-747, October 2009.
[21] M. S. A. Mahmud, M. S. Zainal Abidin, Z. Mohamed, "Virtual reality based mobile robot navigation in greenhouse environment," *Advances in Animal Biosciences*, vol. 8, no. 2, pp. 854-859, July 2017.
[22] J. Kennedy, R. C. Eberhart, "Particle swarm optimization," presented at the IEEE International Conference on Neural Networks, Perth, Western Australia, 27 November- 1 December, 1995.
[23] J. Kennedy, R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," presented at the IEEE International Conference on System, Man and Cybernetics, Orlando, Florida,12-15 October, 1997.
[24] T. R. Machado and H. S. Lopes, "A hybrid particle swarm optimization model for the travelling salesman problem," Adaptive and *Natural Computing Algorithms*, vol. 1, pp. 255-258, August 2005.
[25] W. Pang, K. Wang, C. Zhou, L. Dong, "Fuzzy discrete particle swarm optimization for solving particle swarm optimization problem," presented at the Fourth International Conference on Computer and Information Technology, Wuhan, China, 16-18September, 2004.
[26] H. Izakian, B. T. Ladani, A. Abraham, V. Snasel, "A discrete particle swarm optimization approach for grid job scheduling," *International Journal of Innovative Computing, Information and Control,* vol. 6, no.9, pp.1-15, September 2010.
[27] L. Davis, "Applying adaptive algorithms to epistatic domains," presented at the International Joint Conference on Artificial Intelligence, Los Angeles, CA, 18-23 August, 1985.

Mohd Saiful Azimi Mahmud received his B. Eng in Electrical Engineering, majored in Control and Mechatronics from Universiti Teknologi Malaysia (UTM) in 2015. Currently, he is pursuing his studies in Doctor of Philosophy of Electrical Engineering and his interest is in multi-objective optimization for agricultural mobile robot navigation system.

Mohamad Shukri Zainal Abidin received his B. Eng in Electrical Engineering from Universiti Teknologi Malaysia (UTM) in 1998, Msc in Electrical Engineering from Universiti Teknologi Malaysia (UTM) in 2001 and PhD in Agriculture Engineering from Tokyo University of Technology, Japan in 2014. Currently, he is a Senior Lecturer at Universiti Teknologi Malaysia (UTM) in Control and Mechatronics Department, Faculty of Electrical Engineering and his current research interest involve the adaptive control strategies in fibrous capillary irrigation system and agricultural robotics.

Zaharuddin Mohamedreceived his B. Eng in Electrical, Electronics and Systems from Universiti Kebangsaan Malaysia (UKM) in 1993, Msc in Control Systems Engineering from The University of Sheffield in 1995 and PhD in Control Systems Engineering from The University of Sheffield in 2003. Currently, he is an Associate Professor at Universiti Teknologi Malaysia (UTM) in Control and Mechatronics Department, Faculty of Electrical Engineering and his current research interest involve the control of Mechatronics systems, flexible and smart structures.