A Review: On using ACO Based Hybrid Algorithms for Path Planning of Multi-Mobile Robotics

https://doi.org/10.3991/ijim.v14i18.16371

Ibrahim Ismael Hamad (✉)
Staffordshire University, Staffordshire, UK
Salahaddin University-Erbil, Kurdistan, Iraq
ibrahim.hamarash@su.edu.krd

Mohammad S. Hasan
Staffordshire University, Staffordshire, UK

Abstract—The path planning for Multi Mobile Robotic (MMR) system is a recent combinatorial optimisation problem. In the last decade, many researches have been published to solve this problem. Most of these researches focused on metaheuristic algorithms. This paper reviews articles on Ant Colony Optimisation (ACO) and its hybrid versions to solve the problem. The original Dorigo's ACO algorithm uses exploration and exploitation phases to find the shortest route in a combinatorial optimisation problem in general without touching mapping, localisation and perception. Due to the properties of MMR, adaptations have been made to ACO algorithms. In this review paper, a literature survey of the recent studies on upgrading, modifications and applications of the ACO algorithms has been discussed to evaluate the application of the different versions of ACO in the MMR domain. The evaluation considered the quality, speed of convergence, robustness, scalability, flexibility of MMR and obstacle avoidance, static and dynamic environment characteristics of the tasks.

Keywords—Multi-Robotics, path planning (PP), ACO algorithm, uncertainties

1 Introduction

When multiple mobile robots cooperate to complete a specified task, the group of the robots are referred to as Multi-Mobile Robotics [1]. It is a specific case of Swarm Robotics (SR) where thousands of decentralised mobile robots work together to perform a specific job [2]. These types of robots normally have special properties e.g. member identicality, asynchronism, decentralised control, robustness, scalability, and flexibility. Deterministic methods are too complex for large workspaces and intangible applications. Beni [3] introduced the term SR and emphasised that there are many similar behaviours between SR and insect colonies in biology. This comparison led the robotic community to propose and to use algorithms based on the species societies behaviours to solve problems concurred with SR and MMR. These algorithms referred to as nature-inspired algorithms; they are heuristic or metaheuristics. The ACO algorithm

[4,5] the ABC algorithm [6,7], the Particle Swarm Optimisation (PSO) algorithm [8,9], the Cuckoo Search (CS) algorithm [10,11], the Grey Wolf Optimisation Algorithm (GWO) [12], the Whale Optimisation Algorithm (WOA) [13,14] are some of the mathematically mature optimisation algorithms based on the social behaviour of animal colonies. After three decades and developing tens of heuristic and metaheuristic algorithms, these algorithms did not find their solid way into practical applications [15]. All survey studies of the applications of nature-inspired algorithms addressed ACO as one of the premier optimisation algorithms for Path Planning (PP) of mobile robots [16-18]. This paper reviews the pros and cons of the ACO algorithm and its variants to solve the MMR optimisation problem.

2 The MMR Path Planning Problem

The main goal of PP is to find an optimal, collision-free path from an initial point to an endpoint in the environment with uncertainties to a group of autonomous robots. The general problem of PP for the MMR may be formulated as follows (see figure 1).

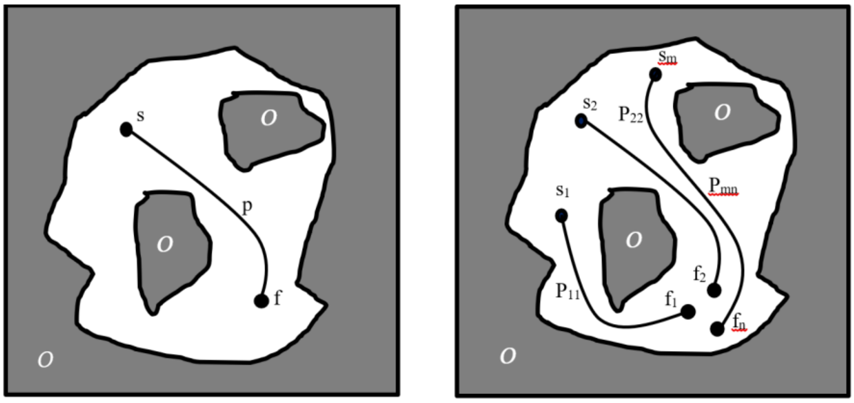

Fig. 1. A visual representation of the path planning (left: single robot, right: multi-robot)

"Given a space X contains obstacles O, find the paths pi ⊂X\O such that pi connects any two points sn,fm∈X\O and pi∈X\pj i≠j, minimizing the measure J".

Here s and f are the start and final targets, respectively, X is the free searching space, O is the constrained space due to obstacles and J is an objective measure to be minimised, P is the connected path between s and f, n is the initial and m is the final pose.

As the MMR applications are rapidly progressing, different applications have different requirements for path planning. To the knowledge of the authors, there is no universal algorithm for all types of applications. Most algorithms support only a few task requirements. The hybridisation of algorithms is a good alternative to embed non-supported functions to the existing algorithms. Let us call the original algorithm the base and the added methods the embedded algorithms. Although ACO has been the base algorithm for many hybrid algorithms [4,19,20], however only a few of them have been implemented in practical models.

3 Ant Colony Optimisation (ACO)

According to the natural ant characteristics, initially individual ants in a colony come out of the nest and depart in different directions searching for food randomly. Once they find a food source, they take some of the food back home. In their way back, they secret a chemical hormone called Pheromone. The amount of Pheromone may depend on the quality and quantity of the food. As a designated path e.g. shorter than other paths is used more and more by the ants, more Pheromone is deposited on that path. In this way, other ants are stimulated to follow the path/trail with more Pheromone and would probably prefer it to the other paths. Hence, the ants collectively use the shortest path as explained in figure 2 [21].

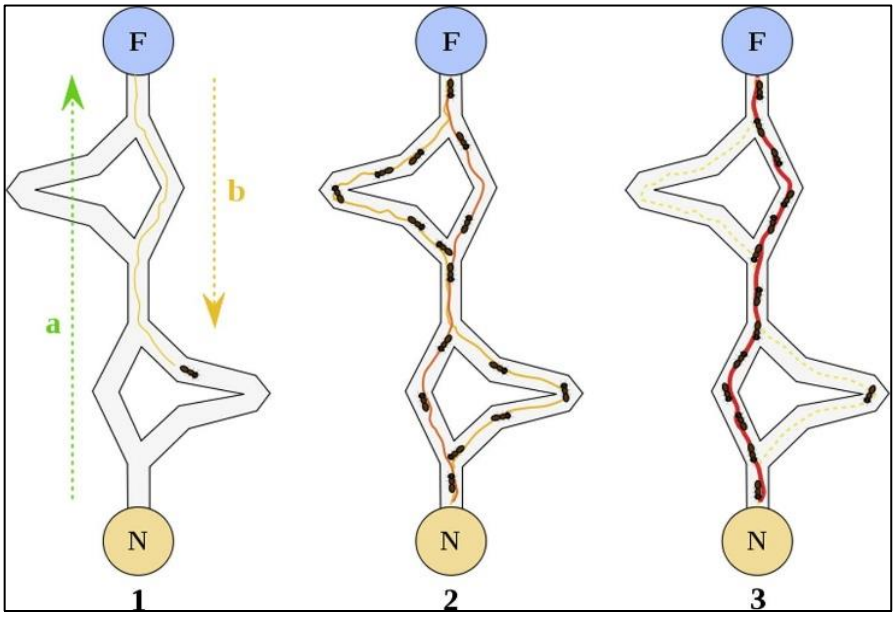

Fig. 2. How Ants find a shortest path [21]

N = Nest, F = Food
1. An ant deposits Pheromone on its way back 2. Some ants choose right path and some the left path 3. Pheromone accumulates at a higher rate on the shorter path

This process is called Stigmergy which is an indirect communication between the ants via Pheromone. It supports efficient collaboration between simple agents who lack any memory or even individual awareness of each other [22].

The ACO algorithm was originally introduced by Marco Dorigo in his PhD in 1992[23] . He converted the ant colony behaviour on how they select a specific path during seeking and collecting food into an artificial optimisation algorithm to solve combinatorial problems [24].

The ACO algorithm is a metaheuristic algorithm which is used to determine an approximate solution to complex combinatorial optimisation problems in a reasonable amount of time [25]. The algorithm works as follows. Initially, a grid search space is defined and n nodes are selected from the search space. Then each ant is positioned at a starting node randomly. At each node, each ant applies a state transition rule to go to an adjacent node and build a complete path. The next node is selected according to the probability function given in **Error! Reference source not found.**) [26]:

$$P_{i,\ j} = \frac{(\tau_{ij})^{\alpha}\ (\eta_{ij})^{\beta}}{\Sigma\,(\tau_{ij})^{\alpha}\ (\eta_{ij})^{\beta}} \tag{1}$$

Here τi, j is the Pheromone trail of the path between node i to node j, $\eta_{i,\ j} = \frac{1}{L_{ij}}$, Li,j is the Euclidean distance between the nodes i to j, α and β are the stimulating of Pheromone concentration and relative importance of visibility, respectively. Roulette wheel method is used to select the best probability [27,28].

After each complete path, the Pheromone concentration of the path is updated by Pheromone amount Δτij which is given by **Error! Reference source not found.** according to [29]:

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{1}{L_k} & \mathit{if\ path\ ij\ is\ used\ \ by\ k^{th}\ ant} \\ 0 & \mathit{Otherwise} \end{cases} \tag{2}$$

The ACO algorithm selects the best solution with the maximum Pheromone laid on the path. If the evaporation is considered, the pheromone amount is modified by inserting the evaporation factor ρ, (0<ρ<1) as given by **Error! Reference source not found.**.

$$\tau_{ij}^{k} = \left(1 - \rho\tau_{ij}\right) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{3}$$

The pseudocode for the ACO algorithm is given in figure 3.

```
  *ACO Algorithm*
  Phase1: * initialization phase*
    For each (r, s), τ(r,s)=τo, End-For *Pheromone
concen-              tration for each path (i,j)*
    For each (r,s), d(r,s)= distance between r and s
                         *search space construction*
    End For
      Set algorithm parameters: Desirability of each
path (η), evaporation rate(ρ), Stimulating factor  of
pheromone concentration (α), relative important be-
tween pheromone and distance(β), number of ants(m),
number of nodes (n)
  Phase2: *Main Loop*
    Set ant k to a node randomly
    For ant k {1,    …. ,    m} *Path construction*
      For S= {1,  ….., n} *searching space scanning*
            Select J=next node according to the maxi-
mum prob bility from equation (1)
      End For
        Calculate the best path using the shortest
length fitness function
    End For
    For all i,   j
      Update Pheromone according to Equation (3)
    End For
  Repeat Phase 2 until a stopping criteria is met
(number      of iteration, best path achievement)
  Output the best path found
```

Fig. 3. ACO algorithm pseudocode

4 Embedded Functions in ACO Base Algorithm in MMR Applications

According to industrial magazines, MMR is rapidly developing in a way that it will change the way human lives [30]. Every day new applications of MMRs are emerging [31-33,2].

As MMR is being deployed increasingly in real life, MMR needs to work in different scenarios. It is well known that a basic requirement for any MMR is that the robots must be able to move safely through the workspace; hence, proper PP must be designed for each application [34,35].

The PP algorithm for MMR is part of a wider interacted engineering problem which contains Simultaneous Localisation Mapping and Perception (SLMP). Various variables like distance, speed, time, task allocation reliability, energy consumption, adaptation to environment, etc. have been considered in most optimisation strategies.

The original ACO algorithm has been the base algorithm for many new and modified nature-inspired algorithms e.g. [36,37,5], etc. that have been developed in recent years. However, the fundamental basic phases i.e. exploitation and exploration of ACO have been repeated.

A modified version of the ACO called the Ant Colony System (ACS) added three updates to the original ACO algorithm [38]. These modified versions were: the first addition was a local Pheromone update which offered a solution to overcome the local optima problem while were ants were building a solution. The second one introduced a state transition rule which provided a direct way to balance between new edges exploration and exploitation of a priori accumulated knowledge to raise the speed during obstacle avoidance process. The third one applied a global updating rule only to edges that belong to the best ant path. The new ACS produced better results than the original ACO when applied to the Travelling Salesman Problem (TSP) [39].

In recent years, the ACO has been revisited by many researchers as a base algorithm to build new algorithms. For example, the ACO has been used for feature selection in the Support Vector Machine (SVM) classifier and they described the problem of redundant features and falling into local optimum [40]. They modified the fitness function and pheromone updating to achieve better results [41]. The limitations of the stagnation of ACO has been studies in [41]. This paper introduced a hybrid ACO-ABC algorithm to reduce these limitations by using employee bee in the exploration of the ACO algorithm and participated in the exploitation into two parts in the ABC algorithm. The study shows that these two limitations are reduced but not eliminated. The trapping into a local optimum solution of the ACO has been addressed in [42]. The researchers modified the ACO by using the mutation and crossover characteristics of the Genetic Algorithm (GA). They showed successful results for this to the no-wait flow shop scheduling combinatorial problem. The stagnation appears due to Pheromone accumulation on the explored paths which leads to failure in finding new best paths [29].

In the MMR domain, the problem of PP is to find paths for a team of autonomous mobile robots sharing the same workspace to achieve a common task while avoiding interference with each other [43]. There are different classifications for MMR Systems; the two major ones are based on the obstacle behaviour and search strategy. Based on the obstacles and environment, they are classified into static i.e. motionless obstacles and dynamic i.e. moving obstacles. The other classification i.e. search strategy deals with local or offline MMR that searches for obstacles while it is making progress towards the objective and global or online when the environment is forecasted before the robots start their missions [44].

There are specific technical requirements for MMR PP to be addressed, the main ten requirements to be considered are listed below:

- Avoiding collisions with obstacles
- Avoiding interference with each other
- Centralised and decentralised approaches
- Shortest path and computational time including agent memory and energy

- Cooperative behaviour, direct and indirect communication
- Global and local searching strategy including the number of visited waypoints
- Multi-Tasking, multi objectives and targets including search space size
- Instantaneous task allocation
- Avoiding deadlocks
- Flocking and Formation

Mao and Yu have studied Multi robotic PP from its early appearance [45]. They added a new function to update the trial intensity of the pheromone. They presented a solution to the conflict between moving ants. They applied their algorithm to a static environment with dead corners.

A hybrid algorithm from ACO and the Cellular Automata (CA) has been developed in [46]. The study dealt with the collision existence of the MMR and tried to determine collision-free trajectories. They simulated ten e-puck agents in the webots platform. This algorithm is very well formulated for the dynamic environment.

Another hybrid algorithm has been presented by embedding GA into ACO [47]. The study covered the shortest path requirement in a static environment. A global optimum solution is considered with real-time constraints in this study.

In [48], the authors introduced a hybrid algorithm by embedding Fuzzy Logic into ACO to avoid premature convergence when it appears. The study used adaptive control to tune ACO parameters to maintain the diversity in the colony.

A technique which was called marginalisation to select an initial location for the ants has been introduced in [49]. This technique locates the ants on the edge of the nodes to prevent touching in the first phase. The research adjusted α and β in the ACO model to prevent local convergence and determined the optimum values for α=[1,3.5] and β=[1.5,5.5].

Paper [28] embedded A* algorithm into ACO to accelerate the convergence speed of the base algorithm. The study dealt with the deadlock through introducing a subroutine for local diffusion pheromones. It simulated three map types with the new algorithm. This algorithm is suitable for tunnels.

An online PP algorithm for a typical MMR system with homogeneous, independent and limited communication skills agents has been presented in [50]. This algorithm is efficient to lead the robots in real time from the initial position to their final goals with known initial and target positions only. The algorithm used the Pheromone concept of ACO in combination with Fuzzy logic. The V-Rep simulation of the algorithm shows satisfactory results with limitations in special case waypoints.

Another hybrid algorithm has been introduced by embedding Improved Potential Field into the ACO algorithm [19]. This paper presented a new method for Pheromone updating. The researchers implemented the algorithm to the Traveller II robot.

An improved algorithm based on ACO that uses the roulette wheel algorithm of GA is developed in [20]. In this algorithm, the probability function of ACO has been adjusted to solve the deadlock problem in the MMR environments. The algorithm is verified via simulation analysis.

Based on the discussions presented above, table 1 shows a summary of the embedded algorithms in the ACO algorithm.

Table 1. The summary of the reviewed algorithms for PP of MMR systems

Algorithm, Year	Embedded algorithm to ACO	Addressed Problem of MMR	Verification method and workspace
Chen and Liu 2019 [19]	Improved Potential Field	Convergence speed, collision existence, local optima	Practical test in real environment
Cao et al. 2019[20]	Genetic Algorithm	Deadlock	Simulation in static environment
Dai et al. 2019 [28]	A*	Convergence speed, deadlock	Simulation tunnels
Liu, Mao and Yu 2006 [45]	New function	Conflict between moving ants, dead corners	Simulation in static environment
Ioannidis et al. 2011 [46]	Cellular Automata	Collision existence	Simulation in dynamic environment
Chaari et al. 2014 [47]	Genetic Algorithm	Shortest path, local optima	Simulation in static environment
Castillo et al. 2015 [48]	Fuzzy Logic	Premature convergence, local optima	Simulation in static environment
Cheng et al. 2017 [49]	New function	Preventing touching ants in the first phase	Simulation in static environment
De Almeda et al. 2019 [50]	Fuzzy Logic	Real-time convergence, Collision existence	Simulation in static environment

This review shows that ACO is currently the dominant base algorithm in the field of PP for MMR. This is because of its flexibility for adjustment and adaptation through the Pheromone updating strategy and probability function tuning. These two adjustable parameters help in finding the shortest collision-free path, with a guaranteed global optimum solution. To achieve these two goals, different metaheuristic algorithms have been embedded in the ACO to build new hybrid algorithms. Most of these algorithms have been verified via simulation but rarely have been tested in a real environment with real robots. The results of this review study show that there is no general algorithm exists yet to solve the PP problem of the MMR that meets all nine requirements.

5 Conclusion

A thorough search of the relevant literature has found that ACO is the most used algorithm for PP of MMR systems in recent years. The ACO algorithm has been used as a base algorithm and it has been utilised to solve the PP problem. In this paper, the recent algorithms with major modifications to ACO have been reviewed and their strengths and weaknesses to MMR task applications have been presented. The ACO consists of two phases; the exploration and exploitation and can tune Pheromone updating and probability function selection. These two attributes have been the key parameters in all the reviewed hybrid algorithms. The original ACO performs well in complex combinatorial optimisation problems but it has convergence speed and local optima problems. These two problems have been solved by embedding Genetic, A*, Cellular automata, Improved Potential Field and Fuzzy algorithms.

Based on the survey and results of the reviewed algorithms, industrial applications of MMR systems are increasing, and new applications are arising every day. Therefore, this is still an open and active research field. These new applications need their own algorithms to achieve their objectives. There are needs for studies to introduce new hybrid algorithms in the presence of bottlenecks, deadlocks, dynamic obstacles, environment uncertainties, robustness and hardware applicability.

6 References

[1] Nedjah, N. and Junior, L. S. (2019) 'Review of methodologies and tasks in swarm robotics towards standardization', Swarm and Evolutionary Computation. https://doi.org/10.1016/j.swevo.2019.100565

[2] Innocente, M. S. and Grasso, P. (2019) 'Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems', Journal of Computational Science. https://doi.org/10.1016/j.jocs.2019.04.009.

[3] Flori, A., Oulhadj, H. and Siarry, P. (2019) 'A new neighborhood topology for quantum particle swarm optimization (QUAPSO)', in GECCO 2019 Companion - Proceedings of the 2019 Genetic and Evolutionary Computation Conference Companion. https://doi.org/10.1145/3319619.3321949

[4] Dorigo, M. and Stützle, T. (2019) 'Ant colony optimization: Overview and recent advances', in International Series in Operations Research and Management Science. https://doi.org/10.1007/978-3-319-91086-4_10

[5] Kurdi, M. (2019) 'Ant colony system with a novel Non-Daemon Actions procedure for multiprocessor task scheduling in multistage hybrid flow shop', Swarm and Evolutionary Computation. Elsevier Ltd, 44(August 2018), pp. 987–1002. https://doi.org/10.1016/j.swevo.2018.10.012

[6] Karaboga, D. (2005) 'An idea based on Honey Bee Swarm for Numerical Optimization', Technical Report TR06, Erciyes University. doi: citeulike-article-id:6592152.

[7] Aslan, S., Badem, H. and Karaboga, D. (2019) 'Improved quick artificial bee colony (iqABC) algorithm for global optimization', Soft Computing. https://doi.org/10.1007/s00500-019-03858-y

[8] Eberhart, R. and Kennedy, J. (2002) 'A new optimizer using particle swarm theory', in. doi: 10.1109/mhs.1995.494215.

[9] Chopard, B. and Tomassini, M. (2018) 'Particle swarm optimization', in Natural Computing Series. doi: 10.1007/978-3-319-93073-2_6.

[10] Yang, X. S. and Deb, S. (2010) 'Engineering optimisation by cuckoo search', Inter-national Journal of Mathematical Modelling and Numerical Optimisation. https://doi.org/10.1504/ijmmno.2010.035430

[11] Mareli, M. and Twala, B. (2018) 'An adaptive Cuckoo search algorithm for optimisation', Applied Computing and Informatics. https://doi.org/10.1016/j.aci.2017.09.001

[12] Mirjalili, S., Mirjalili, S. M. and Lewis, A. (2014) 'Grey Wolf Optimizer', Advances in Engineering Software. https://doi.org/10.1016/j.advengsoft.2013.12.007

[13] Mirjalili, S. and Lewis, A. (2016) 'The Whale Optimization Algorithm', Advances in Engineering Software. doi: 10.1016/j.advengsoft.2016.01.008.

[14] Gharehchopogh, F. S. and Gholizadeh, H. (2019) 'A comprehensive survey: Whale Optimization Algorithm and its applications', Swarm and Evolutionary Computation. https://doi.org/10.1016/j.swevo.2019.03.004

[15] Senanayake, M. et al. (2016) 'Search and tracking algorithms for swarms of ro-bots: A survey', Robotics and Autonomous Systems. doi: 10.1016/j.robot.2015.08.010.
[16] Oh, H. et al. (2017) 'Bio-inspired self-organising multi-robot pattern formation: A review', Robotics and Autonomous Systems. https://doi.org/10.1016/j.robot.2016.12.006
[17] Zafar, M. N. and Mohanta, J. C. (2018) 'Methodology for Path Planning and Optimization of Mobile Robots: A Review', in Procedia Computer Science. https://doi.org/10.1016/j.procs.2018.07.018
[18] Patle, B. K. et al. (2019) 'A review: On path planning strategies for navigation of mobile robot', Defence Technology. doi: 10.1016/j.dt.2019.04.011.
[19] Chen, G. and Liu, J. (2019) 'Mobile Robot Path Planning Using Ant Colony Algorithm and Improved Potential Field Method', Computational Intelligence and Neuroscience. https://doi.org/10.1155/2019/1932812
[20] Cao, M., Yang, Y. Y. and Wang, L. (2019) 'Application of improved ant colony algorithm in the path planning problem of mobile robot', in ACM International Conference Proceeding Series. doi: 10.1145/3341069.3341073.
[21] Toksari, M. D. (2016) 'A hybrid algorithm of Ant Colony Optimization (ACO) and Iterated Local Search (ILS) for estimating electricity domestic consumption: Case of Turkey', International Journal of Electrical Power and Energy Systems. https://doi.org/10.1016/j.ijepes.2015.12.032
[22] Heylighen, F. (2015) 'Stigmergy as a Universal Coordination Mechanism: components, varieties and applications', Lewis, Ted & Marsh, Leslie (eds), Human Stigmergy: Theoretical Developments and New Applications (Studies in Applied Philosophy, Epistemology and Rational Ethics. doi: 10.1016/j.cogsys.2015.12.002.
[23] Dorigo, M. (1992) Optimization, Learning and Natural Algorithms. Politecnico di Milano.
[24] Dorigo, M. and Stützle, T. (2018) 'The Ant Colony Optimization Metaheuristic', in Ant Colony Optimization. https://doi.org/10.7551/mitpress/1290.001.0001
[25] Handbook of Metaheuristics (2003). doi: 10.1007/b101874.
[26] Gambardella, L. M. and Dorigo, M. (2002) 'Solving symmetric and asymmetric TSPs by ant colonies', in. https://doi.org/10.1109/icec.1996.542672
[27] Zhou, Y. et al. (2018) 'Parallel ant colony optimization on multi-core SIMD CPUs', Future Generation Computer Systems. https://doi.org/10.1016/j.future.2017.09.073
[28] Dai, X., Shuai Long, Z. Z. and D. G. (2019) 'Mobile Robot Path Planning Based on Ant Colony Algorithm With A* Heuristic Method', Front. Neurorobot, 13(15), pp. 1–9. doi: https://doi.org/10.3389/fnbot.2019.00015
[29] Dahan, F. et al. (2019) 'Dynamic flying ant colony optimization (DFACO) for solving the traveling salesman problem', Sensors (Switzerland), 19(8). https://doi.org/10.3390/s19081837
[30] Mark and Westall (no date) 'No Title https://fadmagazine.com/'. Available at: https://fadmagazine.com/2017/12/11/233910/
[31] Nagavalli, S., Chakraborty, N. and Sycara, K. (2017) 'Automated sequencing of swarm behaviors for supervisory control of robotic swarms', in Proceedings - IEEE International Conference on Robotics and Automation. https://doi.org/10.1109/icra.2017.7989312
[32] Soares, P. P. et al. (2018) 'Group of Robots Inspired by Swarm Robotics Exploring Unknown Environments', in IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). Rio de Janeiro: IEEE, pp. 1–7. https://doi.org/10.1109/fuzz-ieee.2018.8491631
[33] Li, J. and Tan, Y. (2019) 'A probabilistic finite state machine-based strategy formulation-target search using swarm robotics', Applied Soft Computing Journal, 77, pp. 467–483: https://doi.org/10.1016/j.asoc.2019.01.023

[34] Karpenko, A. P. and I.A., L. (2018) 'Advanced Cat Swarm Optimization Algorithm in Group Robotics Problem A', in INTELS'18. Elsevier B.V., pp. 95–101. https://doi.org/10.1016/j.procs.2019.02.020
[35] Liua, J. et al. (2020) 'Collaborative optimization of robotic disassembly sequence planning and robotic disassembly line balancing problem using improved discrete Bees algorithm in remanufacturing', Robotics and Computer Integrated Manufacturing, 61(101829), pp. 1–18. https://doi.org/10.1016/j.rcim.2019.101829
[36] Li, X. et al. (2017) 'Hybrid artificial bee colony algorithm with a rescheduling strategy for solving flexible job shop scheduling problems', Computers and Industrial Engineering. Elsevier Ltd, 113, pp. 10–26. https://doi.org/10.1016/j.cie.2017.09.005
[37] Jia, Z. hong et al. (2019) 'Multi-objective energy-aware batch scheduling using ant colony optimization algorithm', Computers and Industrial Engineering. Elsevier, 131(November 2018), pp. 41–56. https://doi.org/10.1016/j.cie.2019.03.033
[38] Dorigo, M. and Gambardella, L. M. (1997) 'Ant colony system: A cooperative learning approach to the traveling salesman problem', IEEE Transactions on Evolutionary Computation. https://doi.org/10.1109/4235.585892
[39] Dorigo, M. and Stützle, T. (2003) 'The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances', in. doi: 10.1007/0-306-48056-5_9
[40] 40Peng, H. et al. (2018) 'An Improved Feature Selection Algorithm Based on Ant Colony Optimization', IEEE Access, 6, pp. 69203–69209. https://doi.org/10.1109/access.2018.2879583
[41] Janakiraman, S. (2018) 'A hybrid ant colony and artificial bee colony optimization algorithm-based cluster head selection for IoT', in Procedia Computer Science. https://doi.org/10.1016/j.procs.2018.10.407
[42] Engin, O. and Güçlü, A. (2018) 'A new hybrid ant colony optimization algorithm for solving the no-wait flow shop scheduling problems', Applied Soft Computing Journal, 72, pp. 166–176. https://doi.org/10.1016/j.asoc.2018.08.002
[43] Parker, L. E. (2009) 'Multiple Mobile Robot Teams, Path Planning and Motion Coordination in', in Encyclopedia of Complexity and Systems Science. https://doi.org/10.1007/978-0-387-30440-3_344
[44] Das, P. K., Behera, H. S. and Panigrahi, B. K. (2016) 'A hybridization of an im-proved particle swarm optimization and gravitational search algorithm for multi-robot path planning', Swarm and Evolutionary Computation. https://doi.org/10.1016/j.swevo.2015.10.011
[45] Liu, S., Mao, L. and Yu, J. (2006) 'Path planning based on ant colony algorithm and distributed local navigation for multi-robot systems', in 2006 IEEE International Conference on Mechatronics and Automation, ICMA 2006. https://doi.org/10.1109/icma.2006.257476
[46] Ioannidis, K., Sirakoulis, G. C. and Andreadis, I. (2011) 'Cellular ants: A method to create collision free trajectories for a cooperative robot team', Robotics and Autonomous Systems. https://doi.org/10.1016/j.robot.2010.10.004
[47] Châari, I. et al. (2014) 'SmartPATH: An efficient hybrid ACO-GA algorithm for solving the global path planning problem of mobile robots', International Journal of Advanced Robotic Systems. https://doi.org/10.5772/58543
[48] Castillo, O. et al. (2015) 'A new approach for dynamic fuzzy logic parameter tuning in Ant Colony Optimization and its application in fuzzy control of a mobile robot', Applied Soft Computing Journal. https://doi.org/10.1016/j.asoc.2014.12.002
[49] Cheng, J. et al. (2017) 'An improved ACO algorithm for mobile robot path planning', in 2016 IEEE International Conference on Information and Automation, IEEE ICIA 2016. https://doi.org/10.1109/icinfa.2016.7831958

[50] De Almeida, J. P. L. S. et al. (2019) 'Bio-inspired on-line path planner for cooperative exploration of unknown environment by a Multi-Robot System', Robotics and Autonomous Systems. https://doi.org/10.1016/j.robot.2018.11.005

7 Authors

Ibrahim Ismael Hamad is a Professor at Salahaddin University-Erbil, Kurdistan, Iraq and is an MRes student in the School of Computing and Digital Technologies, Staffordshire University, Stoke-on-Trent, UK.

Mohammad S. Hasan is a senior lecturer in the School of Computing and Digital Technologies, Staffordshire University, Stoke-on-Trent, UK.

Article submitted 2020-06-18. Resubmitted 2020-07-07. Final acceptance 2020-07-09. Final version published as submitted by the authors.