Development of Swarm Intelligence Leader-Vicsek-Model for Multi-AGV Path Planning

Shiwei Lin
Faculty of Engineering and Information Technology
University of Technology Sydney
Sydney, Australia

Ang Liu
Faculty of Engineering and Information Technology
University of Technology Sydney
Sydney, Australia

Xiaoying Kong
Faculty of Engineering and Information Technology
University of Technology Sydney
Sydney, Australia

Jack Wang
Faculty of Engineering and Information Technology
University of Technology Sydney
Sydney, Australia

***Abstract*—Automatic guided vehicle (AGV) is the mobile robot widely used in the industry, and multiple AGVs are usually involved for industrial intelligence. Path planning for multi-AGV is essential for automatic transportation, and we propose a swarm-based approach that is inspired by biological subjects. The dynamic virtual leader is assigned for updating the locations and angles for the Leader-Vicsek-Model, and multiple agents are assigned through the centralized approach while the path planning is achieved through the decentralized method.**

Keywords—multi-AGV, path planning, swarm algorithm

I. INTRODUCTION

Mobile robots have many commercial and industrial applications with the development of automatic technologies. They are applied to the industry for the development of intelligent logistics, industrial intelligence, and intelligent factories [1, 2]. The implementation of Automated Guided Vehicles (AGVs) results in increasing significance to improve transportation efficiency with less cost [1, 3, 4]. AGVs for industry comprise towing vehicles, unit load vehicles, pallet trucks, and forklift [5]; for production, AGV can improve safety and decrease labor costs in a production environment for the high demands [6, 7].

AGVs are utilized in flexible production lines of modern manufacturing and integrated into automated intelligent control systems [5, 6, 8]. They can operate at high speeds to transfer products in a chaotic situation in warehouses. AGVs become essential parts of the servo handling system, logistics, warehousing system, and storage industry [6, 9]. They are employed in varying areas, such as distribution and material handing in manufacturing, transportation, and transshipment [10]. Detecting objects in the path and eliminating the problems automatically with sensors' help can increase the adaptability and intelligence of AGVs [6].

In this paper we provide a solution for multi-AGV path planning in the industry to optimize automatic transportation. The paper is organized as follows. Section II reviews the navigation methods of multi-AGV. Section III presents Leader-Vicsek-Model for multi-AGV path planning. Section IV demonstrates the experiment results to validate this solution, and we conclude this paper in Section V.

II. RELATED WORK

More and more AGVs are employed for warehouses to achieve the increasing delivery throughput requirements; therefore, it is typical for operating hundreds of AGVs in a warehouse [4]. Multi-AGV gained more attention in the robotics community in recent years to increase productivity and reduce cost in the industry. Multi-AGV path planning is the most significant factor to ensure the efficient flow of materials for production [11]. Completeness, optimality, and computational complexity are the basis of taxonomizing multi-robot motion planning methods [12]. Vehicle motion planning is difficult to solve for the general motion planning problem, especially when the number of degrees of freedom rises [13].

Furthermore, multi-objective optimization incorporates intelligent optimization methods such as swarm-based algorithms and evolution-based algorithms, and classical optimization algorithms like interactive methods, weighted sum methods, and Pareto-dominated methods [11]. Scheduling, dispatching, and touring of tasks are three issues in the multi-AGV path planning. The existing constraints in multi-AGV path planning are varying from time window constraints, collision-free constraints, and time/distance constraints [11].

Coupled and decoupled are distinct multi-robot motion planning approaches. The coupled methods extend single-robot motion planning algorithms [12]. For multi-AGV routing problems, the algorithms for a single AGV have been modified to avoid conflicts and improve efficiency. An improved A* algorithm and the dynamic rapidly exploring random trees (RRT) algorithm with kinematic constraints are proposed as a bi-level path planning algorithm to decrease the incidence of path conflicts [1]. Dijkstra algorithm is used to compute the paths, then the coordinator algorithm computes the reserved segments periodically, basing on the coordinator diagram [14]. Jump strategy and a distance factor are employed for an optimized artificial potential field (APF) algorithm, treating other agents as dynamic obstacles for collaborative path planning in dynamic space [15].

However, the path search occurs in composite configuration space for the multi-robot case; for relieving the computational load of coupled methods, the decoupled methods were introduced in [12]. Path planning and motion coordination are the compositions of the main motion planning for the multi-robot decoupled motion planning methods [12]. The decoupled multi-robot systems can be classified into two main categories: centralized and decentralized [4, 12].

More precisely, decentralized approaches solve the complexity of large-scale problems for multi-robot coordinators directly. Each robot determines its paths and resolves conflicts autonomously by collecting information from other robots [4]. A multi-agent system eliminates the demands of a central decision-maker [16]. A Markov decision process (MDP) is involved in distributed deep reinforcement learning to model the environment [16]. Kinematic constraints are proposed with a distributed collaborative method to make a coverage decision for each agent on real-time local communications and sensing [17].

Additionally, auction algorithms are widely used for solving multi-agent task assignments, such as the Consensus-Based Auction algorithm (CBAA) and the Consensus-Based Bundle algorithm (CBBA) [18]. A mathematical model takes realistic features into account, including inaccuracy of the sensors and inertial effects, locality of communication and time delay [19]. Two decentralized control algorithms are proposed as a collective target tracking algorithm and a self-propelled flocking model based on animal collective motion [19]. Another decentralized outdoor flocking and formation algorithm is modelling for a stable group in an error-prone environment [20].

In contrast, centralized strategies involve a single decision-maker to plan the entire path for all robots, and its advantage is to find optimal solutions [4]. The primary task to perform assigned missions is computing dynamic path planning [21]. Single processing collects the information and plans path for each robot in the fleet with collision-free in the centralized method [12]. Serving a set of workstations by multiple AGVs can be achieved by a central management system that schedules the paths and assigns tasks [10]. A global path planning method and the artificial potential fields are exploited for a mobility model for swarms [21].

Dynamic network flow optimization (DNFO) achieves path planning and dynamic task assignment by coming to a velocity synthesis approach and self-organizing map neural network [22]. Potential field method and Mixed-integer linear programming are presented for task assignment and path planning when threats exist [23]. A Mixed-integer optimization problem can also model the time assignment problem as a hybrid metaheuristic optimization algorithm [24]. Path planning and vehicle routing problems are two NP-hard problems. Global optimization is applied the A* algorithm with images, satisfying the capacity constraint, and using a fuzzy-based genetic algorithm to consider the fuzzy workstation demands and travel distances [10].

A centralized motion coordination controller implements the prioritized coordination algorithm to ensure the blockage-free and collision-free motion of multi-AGV [12]. The new genetic algorithms minimize the total path distance for AGVs and each AGV path distance through the heuristic crossover operator approach [11]. Its steps comprise genetic coding, fitness function, population selection, selection action, matrix decoding, and crossover operation [11]. K-Partite graph and a Quad-tree are for optimizing the overall mission goal and individually assign the agents tasks [25].

The path planning search space can be decreased by a multi-layer structure world representation [4]. Prioritized schemes can perform path planning strategies, and priority-based motion planning is a prominent multi-robot motion planning approach, assigning a priority value for each robot [4, 12]. The multi-layer centralized optimized strategy contains the A* algorithm in the first layer, and it is implemented with onboard odometry and processor, WIFI networking capabilities, and an RGB-D sensor [4].

Animal swarms exhibit various typical flocking patterns, and these universal patterns enable them to be reconstructed with agent-based models efficiently [19]. Many-particle systems demonstrate a complex cooperative behavior during phase transition [26]. Viscous flows, aggregation, or biological pattern formation are exhibited to scale the related dynamic and geometrical quantities for nonequilibrium processes [26]. A simple rule and random fluctuations determine the velocity of the particles in nonequilibrium systems, and the Vicsek model investigates transport, clustering, and phase transition for nonequilibrium systems [26]. Vicsek model is expanded to allow for cohesion by an attraction/repulsion pairwise interaction, vary the polarity of the particles, and consider the ambient fluid [27]. A decentralized control framework for outdoor flocking and formation algorithm is with bio-inspiration based on statistical modelling of animal swarms [20].

Formation control algorithms enable multiple vehicles to follow the predefined paths in the particular motivating application for mission planning and execution [28]. A consensus-based cooperative formation strategy is capable of collision avoidance [29]. Flocking, formation control and path-following are implemented in a decentralized architecture while centralized in principle [28]. Stigmergy, flocking, and evolution are three biologically inspired processes of a novel coordination algorithm designed for distributed swarming target localization with environment adaptation and self-coordination [30]. Dynamic priority maps are used for controlling the swarm [31]. Most flocking algorithms rely on information from neighboring agents and assume one virtual leader, while a flocking algorithm is designed to follow multiple virtual leaders [32].

Besides, for metaheuristic optimization algorithm, evolutionary algorithms and swarm intelligence algorithms are applied to solve the path planning problem in a feasible time [24, 33]. Angle-encoded particle swarm optimization accelerates the swarm convergence with position and angular velocity [34]. The particle swarm optimization (PSO) and the genetic algorithm consider the dynamic properties of vehicles in a parallel programming paradigm [35]. PSO involves the available computation time to solve the 4D path planning problem by detecting conflicts [36]. Parallelized Ant Colony Optimization (ACO) algorithm accelerates ACO and can deal with complex tasks [33].

From the literature, distinct multi-robot motion planning includes coupled and decoupled approaches. For reducing the computational load for multi-AGV system, the decoupled approaches are proposed, and they can be classified into centralized and decentralized methods. Most swarm-based algorithms and evolution-based algorithms are bio-inspiration, and we consider biological pattern for developing the new algorithm. This paper proposes Leader-Vicsek-Model combine the advantages of centralized and decentralized methods; a decision-maker assigns a virtual leader and groups to the AGVs, while each AGV determines its path and achieves collision-avoidance by collecting data from other robots.

III. Leader-Vicsek-Model

A. Description of Leader-Vicsek-Model

The Vicsek model aims to move the particles in the same direction as their neighbor, following the biological pattern [26]. The Vicsek model generates the random direction while the destination has not been considered, so it is not suitable for path planning in the industry. This paper proposes a new multi-AGV path planning algorithm for commercial or industrial storage to collaborate the AGVs. It improves the Vicsek model, assigning the virtual leader to reach the exit and consider collision avoidance.

The proposed algorithm is described in Figure 1. The storage map is built based on the industrial environment, then generate the collision-free area. Assigning the initial conditions of the Vicsek model and AGVs, and a virtual leader for the current swarm. For updating the angles and positions for multi-AGV, the average angle is calculated for the followers based on the radius. The angle is calculated for the leader until reaching the destination.

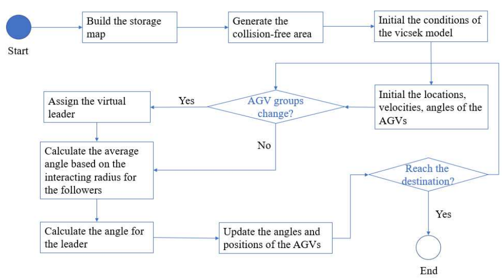

Fig. 1. The multi-AGV path planning process

The model for the AGV system is designed as follows and demonstrated in Figure 2.

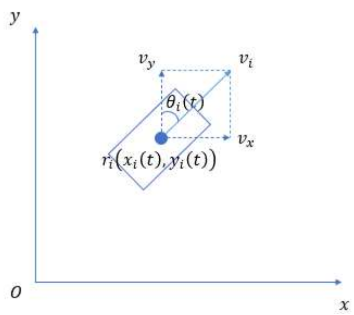

Fig. 2. The model for the AGV

The equations for the kinetic model are listed as (1) – (5):

$$v_{ix}(t) = v_i \sin\theta_i(t) \tag{1}$$

$$v_{iy}(t) = v_i \cos\theta_i(t) \tag{2}$$

$$\theta_i(t) = \theta_i(t-\Delta t) + \int_{t-\Delta t}^{t} \omega \cdot dt \tag{3}$$

$$x_i(t) = x_i(t-\Delta t) + \int_{t-\Delta t}^{t} v_i \sin\theta_i(t) \cdot dt \tag{4}$$

$$y_i(t) = y_i(t-\Delta t) + \int_{t-\Delta t}^{t} v_i \cos\theta_i(t) \cdot dt \tag{5}$$

where $(x_i(t), y_i(t))$ represents the location of the AGV, the angle $\theta_i(t)$ at time t, ω is the angular velocity, v stands for the constant velocity, and the time step is Δt for the AGV i.

The improved Vicsek model applies the dynamic virtual leader for planning the path to achieve faster convergence, shorter path, and more accurate direction for reaching the destination. The virtual leader is dynamically generated when the number of AGVs in the current group changes. The equations for updating the locations and angles for follower-AGV implement the traditional Vicsek model. Biological subjects are intended to move as their neighbourhood and spinning in the same direction for interaction [26]. The Vicsek model is shown in Figure 3, r_i represents the location of the individual i, R is the radius of interaction, and v is the absolute velocity.

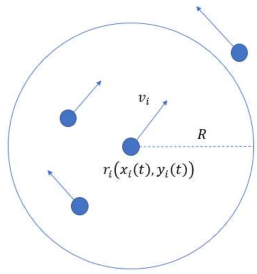

Fig. 3. The Vicsek model

The locations for AGVs are updated by (6), the angles for the virtual leader are calculating by (7) and updating the followers' angles through (8).

$$r_i(t+\Delta t) = r_i(t) + v\Delta t \begin{pmatrix} \cos\theta_i(t) \\ \sin\theta_i(t) \end{pmatrix} \tag{6}$$

$$\theta_i(t+\Delta t) = \theta_i(t) + \eta_i(t) \tag{7}$$

$$\theta_i(t+\Delta t) = \arctan\frac{\langle \sin(\theta_i(t)) \rangle_r}{\langle \cos(\theta_i(t)) \rangle_r} + \eta_i(t) \tag{8}$$

where $\eta_i(t)$ is the noise, the average direction $\arctan\frac{\langle \sin(\theta_i(t)) \rangle_r}{\langle \cos(\theta_i(t)) \rangle_r}$ is calculated based on the interaction range r, and the travel distance is $v\Delta t$.

B. Implementation of Leader-Vicsek-Model

1) Scenario of warehouse

The warehouse map is generated through a real industrial warehouse by MATLAB, and it is shown in Figure 4. The three-dimensional storage system is applied for Area A to Area F, so the multi-AGV system is intended for outbound deliveries in the remaining areas highlighted by blue.

2) Implementation of the model

Implementation of the model divides into the following parts:

Step 1 – Map generation

Step 2 – Initial the model: set the time, timeslot, domain size, particle number, radius, and velocity (5 m/s).

Step 3 – Initial the multi-AGV system: initial the locations and angles for AGVs. The AGVs are turned into the appropriate angles at first. The inappropriate angles will result in unnecessary path segments for AGVs to turn to the direction of the destination, which will increase the costs.

Step 4 – Divide the particles into different areas based on their locations

Step 5 – Get the virtual leader for each area, and the virtual leader is dynamic based on the location of AGVs and the direction of the destination. For example, when AGVs enter the different area, the system will assign them to the group located in the area and get a new virtual leader.

Step 6 – Updating the location and angles for the leader and followers until they reach the destination.

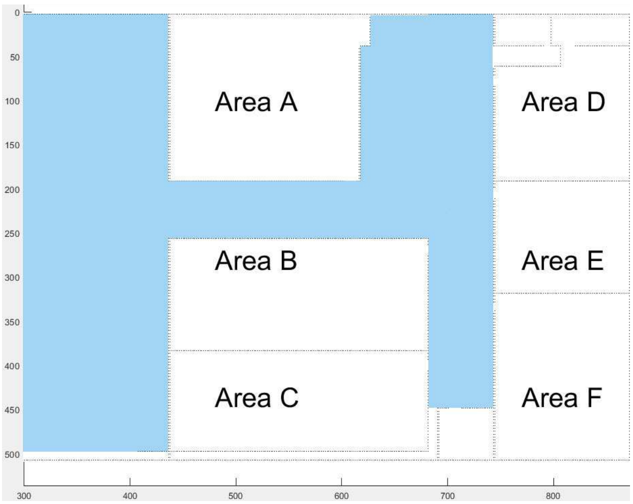

Fig. 4. The scenario of storage

3) *Data*

The locations and angles of the path planning approach are saved as Table 1:

TABLE I. EXAMPLE DATA

Time	Multi-AGV			
	AGV 1	***AGV 2***	**…**	***AGV N***
t_0	$x_1(t_0)$ $y_1(t_0)$ $\theta_1(t_0)$	$x_2(t_0)$ $y_2(t_0)$ $\theta_2(t_0)$	…	$x_N(t_0)$ $y_N(t_0)$ $\theta_N(t_0)$
t_1	$x_1(t_1)$ $y_1(t_1)$ $\theta_1(t_1)$	$x_2(t_1)$ $y_2(t_1)$ $\theta_2(t_1)$	…	$x_N(t_1)$ $y_N(t_1)$ $\theta_N(t_1)$
…	…	…	…	…
t_n	$x_1(t_n)$ $y_1(t_n)$ $\theta_1(t_n)$	$x_2(t_n)$ $y_2(t_n)$ $\theta_2(t_n)$	…	$x_N(t_n)$ $y_N(t_n)$ $\theta_N(t_n)$

IV. SIMULATION EXPERIMENT

We use simulation to validate our Leader-Vicsek-Model through MATLAB. The velocity of AGVs is set to 5 m/s, the total number of particles is 15, and the timeslot is set to 1s.

Figure 5 indicates the initial position of AGVs, which is generated in different areas. Figure 6, Figure 8 and Figure 9 display the locations of AGVs in different time of Leader-Vicsek-Model. In contrast, Figure 7 shows the locations generated by the Vicsek model to compare with our model.

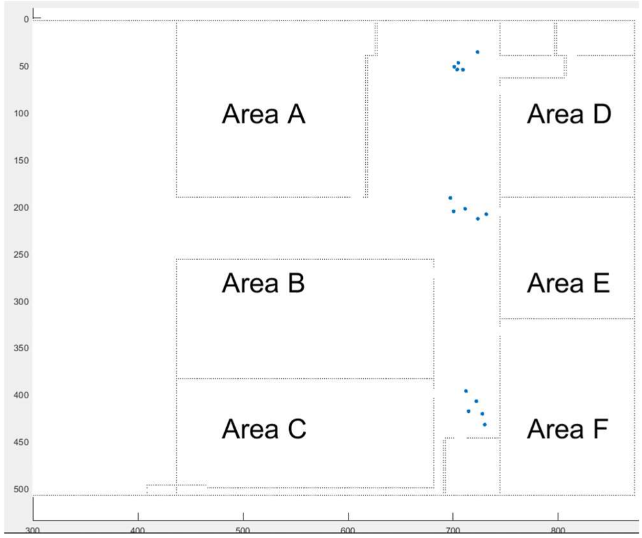

Fig. 5. t = 0

Figure 6 demonstrates the locations of AGVs when $t = 40$. The particles are still in their initial group that is assigned through the interaction radius and the initial locations. Each group has its virtual leader in making AGVs achieve faster path planning.

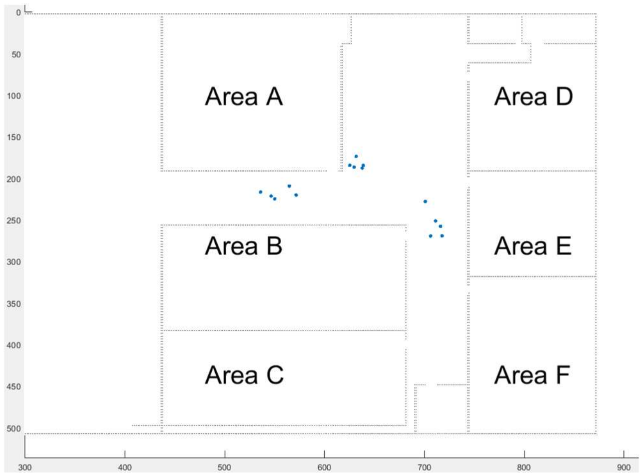

Fig. 6. t = 40

Figure 7 demonstrates the locations of AGVs generated by the Vicsek model when $t = 40$. The initial conditions of the Vicsek model are the same as the Leader-Vicsek-Model. AGVs automatically move in the same direction as their neighbors in the defined radius while cannot consider avoiding the collision or the direction of the outbound deliveries. Some of the AGVs turned more than 360 degree, which results in more energy consumption. Some AGVs are outside the planned area or enter Area A to F, so the Vicsek model is not suitable for path planning in a warehouse.

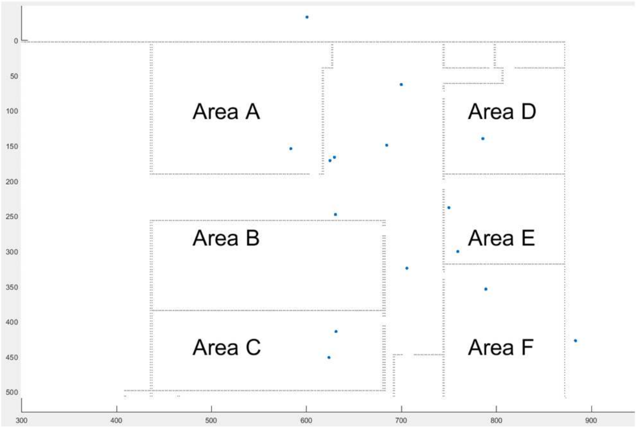

Fig. 7. t = 40 (Vicsek model)

Figure 8 indicates the locations of AGVs when $t = 70$. The particles are in the same group with the same leader. Their updating angles are limited to $[d - 90, d + 90]$ where d is the direction of the virtual leader because their path is tended to have fewer turning in the current area, resulting in lower costs.

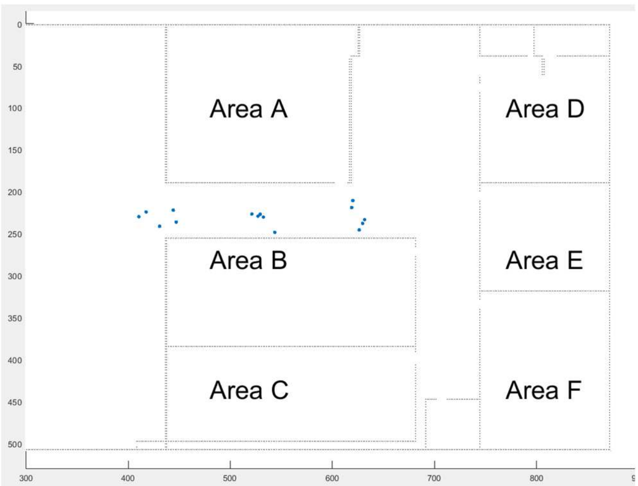

Fig. 8. t = 70

Figure 9 displays the locations of AGVs when $t = 120$ and all AGVs completed outbound deliveries.

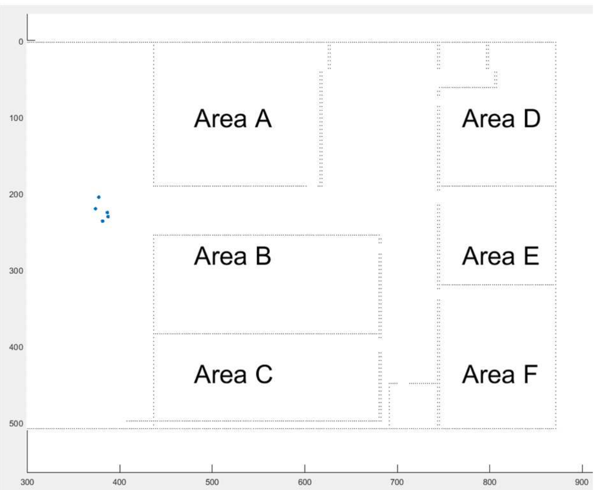

Fig. 9. t = 120

V. CONCLUSION

Leader-Vicsek-Model improves the Vicsek model by assigning a dynamic virtual leader and restricting the updating equations. The AGVs are assigned as different groups based on their initial locations. Each group has one virtual leader to direct the followers to the destination or close to the destination. Every angle is limited to $[0,360]$ to avoid unnecessary turning costs. When generating initial angles, the AGVs are turning into the appropriate angle based on the direction of their virtual leader.

Initial of the model and the AGVs and assigning the AGVs into different groups are completed through a centralized method, while part of the path planning approach is decentralized. Each group only consider their neighbors, and the AGVs in one group are tended to move in the same direction in their interacting radius. When AGVs enter the area of a different group, they become the particles of the group and get a new virtual leader.

REFERENCES

[1] Z. Yuan, Z. Yang, L. Lv, and Y. Shi, "A Bi-Level Path Planning Algorithm for Multi-AGV Routing Problem," *Electronics,* vol. 9, no. 9, 2020, doi: 10.3390/electronics9091351.

[2] X. Guo, Z. Ren, Z. Wu, J. Lai, D. Zeng, and S. Xie, "A Deep Reinforcement Learning Based Approach for AGVs Path Planning," presented at the 2020 Chinese Automation Congress (CAC), 2020.

[3] J. Liu, P. Jayakumar, J. L. Stein, and T. Ersal, "A nonlinear model predictive control formulation for obstacle avoidance in high-speed autonomous ground vehicles in unstructured environments," Vehicle System Dynamics, vol. 56, no. 6, pp. 853-882, 2017, doi: 10.1080/00423114.2017.1399209.

[4] V. Digani, M. A. Hsieh, L. Sabattini, and C. Secchi, "Coordination of multiple AGVs: a quadratic optimization method," Autonomous Robots, vol. 43, no. 3, pp. 539-555, 2018, doi: 10.1007/s10514-018-9730-9.

[5] L. Lynch, T. Newe, J. Clifford, J. Coleman, J. Walsh, and D. Toal, "Automated Ground Vehicle (AGV) and Sensor Technologies- A Review," in Twelfth International Conference on Sensing Technology (ICST), 2018, pp. 347-352.

[6] M. Kostov, V. Kostova, and R. Markoska, "AGV guidance system simulations with a programmable robotics kit," International journal of reasoning-based intelligent systems, vol. 7, no. 1-2, pp. 42-46, 2015.

[7] M. Cramer, J. Cramer, D. De Schepper, P. Aerts, K. Kellens, and E. Demeester, "Benchmarking low-cost inertial measurement units for indoor localization and navigation of AGVs," Procedia CIRP, vol. 86, pp. 204-209, 2019.

[8] P. Li, Y. Xu, T. Shen, and S. Bi, "INS/UWB integrated AGV localization employing Kalman filter for indoor LOS/NLOS mixed environment," in International Conference on Advanced Mechatronic Systems (ICAMechS), 2019, pp. 294-298.

[9] Z. Cui, H. Xu, Z. Chen, H. Yang, S. Huang, and M. Gong, "Design of a novel AGV with automatic pick-and-place system based on scissor lifting platform," presented at the 2020 Chinese Automation Congress (CAC), 2020.

[10] P. T. Zacharia and E. K. Xidias, "AGV routing and motion planning in a flexible manufacturing system using a fuzzy-based genetic algorithm," The International Journal of Advanced Manufacturing Technology, vol. 109, no. 7-8, pp. 1801-1813, 2020, doi: 10.1007/s00170-020-05755-3.

[11] Z. Han, D. Wang, F. Liu, and Z. Zhao, "Multi-AGV path planning with double-path constraints by using an improved genetic algorithm," PLoS One, vol. 12, no. 7, p. e0181747, 2017, doi: 10.1371/journal.pone.0181747.

[12] M. A. Guney and I. A. Raptis, "Dynamic prioritized motion coordination of multi-AGV systems," Robotics and Autonomous Systems, vol. 139, 2021, doi: 10.1016/j.robot.2020.103534.

[13] C. Goerzen, Z. Kong, and B. Mettler, "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance," Journal of Intelligent and Robotic Systems, vol. 57, no. 1-4, pp. 65-100, 2009, doi: 10.1007/s10846-009-9383-1.

[14] R. Olmi, C. Secchi, and C. Fantuzzi, "Coordination of industrial AGVs," International journal of vehicle autonomous systems, vol. 9, no. 1-2, pp. 5-25, 2011.

[15] J. Sun, J. Tang, and S. Lao, "Collision Avoidance for Cooperative UAVs With Optimized Artificial Potential Field Algorithm," IEEE Access, vol. 5, pp. 18382-18390, 2017, doi: 10.1109/access.2017.2746752.

[16] R. N. Haksar and M. Schwager, "Distributed Deep Reinforcement Learning for Fighting Forest Fires with a Network of Aerial Robots," in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018: IEEE, pp. 1067-1074, doi: 10.1109/IROS.2018.8593539.

[17] Y. Jin, Y. Wu, and N. Fan, "Research on distributed cooperative control of swarm UAVs for persistent coverage," in the 33rd Chinese Control Conference, 2014: TCCT, CAA, pp. 1162-1167, doi: 10.1109/ChiCC.2014.6896792.

[18] S. Hunt, Q. Meng, and C. J. Hinde, "An Extension of the Consensus-Based Bundle Algorithm for Multi-agent Tasks with Task Based Requirements," in 2012 11th International Conference on Machine Learning and Applications, 2012, vol. 2: IEEE, pp. 451-456, doi: 10.1109/ICMLA.2012.163.

[19] C. Virágh et al., "Flocking algorithm for autonomous flying robots," Bioinspiration & biomimetics, vol. 9, no. 2, pp. 025012-025012, 2014, doi: 10.1088/1748-3182/9/2/025012.

[20] G. Vasarhelyi et al., "Outdoor flocking and formation flight with autonomous aerial robots," in 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014: IEEE, pp. 3866-3873, doi: 10.1109/IROS.2014.6943105.

[21] E. Falomir, S. Chaumette, and G. Guerrini, "A 3D Mobility Model for Autonomous Swarms of Collaborative UAVs," in 2019 International Conference on Unmanned Aircraft Systems (ICUAS), 2019: IEEE, pp. 196-204, doi: 10.1109/ICUAS.2019.8798199.

[22] D. Zhu, H. Huang, and S. X. Yang, "Dynamic Task Assignment and Path Planning of Multi-AUV System Based on an Improved Self-Organizing Map and Velocity Synthesis Method in Three-Dimensional Underwater Workspace," IEEE Trans Cybern, vol. 43, no. 2, pp. 504-14, Apr 2013, doi: 10.1109/TSMCB.2012.2210212.

[23] A. Seil and H. J. Kim, "Simultaneous task assignment and path planning using mixed-integer linear programming and potential field method," in 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013), Kimdaejung Convention Center, Gwangju, Korea, 2013, pp. 1845-1848, doi: 10.1109/ICCAS.2013.6704241.

[24] S. Chaimatanan, D. Delahaye, and M. Mongeau, "A Hybrid Metaheuristic Optimization Algorithm for Strategic Planning of 4D Aircraft Trajectories at the Continental Scale," IEEE computational intelligence magazine, vol. 9, no. 4, pp. 46-61, 2014, doi: 10.1109/MCI.2014.2350951.

[25] S. Alqahtani, I. Riley, S. Taylor, R. Gamble, and R. Mailler, "Task Allocation in Uncertain Environments using a QuadTree and Flow network," in 2018 International Conference on Unmanned Aircraft Systems (ICUAS), 2018: IEEE, pp. 74-83, doi: 10.1109/ICUAS.2018.8453382.

[26] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles," Physical Review Letters, vol. 75, no. 6, pp. 1226-1229, 1995, doi: 10.1103/PhysRevLett.75.1226.

[27] H. Chaté, F. Ginelli, G. Grégoire, F. Peruani, and F. Raynaud, "Modeling collective motion: variations on the Vicsek model," The European Physical Journal B, vol. 64, no. 3-4, pp. 451-456, 2008, doi: 10.1140/epjb/e2008-00275-9.

[28] L. A. Valbuena Reyes and H. G. Tanner, "Flocking, Formation Control, and Path Following for a Group of Mobile Robots," IEEE Transactions on Control Systems Technology, vol. 23, no. 4, pp. 1268-1282, 2015, doi: 10.1109/tcst.2014.2363132.

[29] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-UAV system," in 2014 American Control Conference, 2014: American Automatic Control Council, pp. 2077-2082, doi: 10.1109/ACC.2014.6858777.

[30] A. L. Alfeo, M. G. C. A. Cimino, N. De Francesco, M. Lega, and G. Vaglini, "Design and simulation of the emergent behavior of small drones swarming for distributed target localization," Journal of Computational Science, vol. 29, pp. 19-33, 2018, doi: 10.1016/j.jocs.2018.09.014.

[31] D. J. Howden, "Fire tracking with collective intelligence using dynamic priority maps," in 2013 IEEE Congress on Evolutionary Computation, Cancún, México, 2013: IEEE, pp. 2610-2617, doi: 10.1109/CEC.2013.6557884.

[32] H.-S. Su, "Flocking in Multi-Agent Systems with Multiple Virtual Leaders Based Only on Position Measurements," Communications in theoretical physics, vol. 57, no. 5, pp. 801-807, 2012, doi: 10.1088/0253-6102/57/5/10.

[33] U. Cekmez, M. Ozsiginan, and O. K. Sahingoz, "A UAV path planning with parallel ACO algorithm on CUDA platform," in 2014 International Conference on Unmanned Aircraft Systems (ICUAS), 2014: IEEE, pp. 347-354, doi: 10.1109/ICUAS.2014.6842273.

[34] V. T. Hoang, M. D. Phung, T. H. Dinh, and Q. P. Ha, "Angle-Encoded Swarm Optimization for UAV Formation Path Planning," in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018: IEEE, pp. 5239-5244, doi: 10.1109/IROS.2018.8593930.

[35] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time UAV Path Planning," IEEE Transactions on Industrial Informatics, vol. 9, no. 1, pp. 132-141, 2013, doi: 10.1109/tii.2012.2198665.

[36] J. Cobano, D. Alejo, G. Heredia, and A. Ollero, "4D trajectory planning in ATM with an anytime stochastic approach," in ACM International Conference Proceeding Series, 2013: ACM, in ATACCS '13, pp. 1-8, doi: 10.1145/2494493.2494494.