**IAES International Journal of Robotics and Automation (IJRA)**
Vol. 15, No. 3, September 2026, pp. 589∼596
ISSN: 2722-2586, DOI: 10.11591/ijra.v15i3.pp589-596


# Cost-aware global frontier matching for ROS 2 multi-robot exploration

**Chu Van Cuong, Tran Tuan Anh**
AViS Lab, Posts and Telecommunications Institute of Technology, Hanoi, Vietnam

**Article Info**

***Article history:***

Received May 2, 2026
Revised Jul 23, 2026
Accepted Aug 6, 2026

***Keywords:***

Cost-aware allocation
Frontier allocation
Global matching
Multi-robot exploration
Path overlap
ROS 2

**ABSTRACT**

Multi-robot frontier exploration supports warehouse mapping and inspection robotics, but geometric assignment can produce overlapping motion and inefficient target pairing. This paper presents a ROS 2 frontier-allocation layer that combines a weighted frontier cost with global one-to-one matching. The study isolates global matching from sequential assignment while keeping the cost formulation and ROS 2 execution stack fixed. Three policies are evaluated on two indoor maps, four team sizes, and three seeds. Across 72 completed main-policy runs, global matching gives the lowest mean completion time, travelled distance, path overlap, and assignment conflict. Relative to sequential cost-based assignment, it reduces completion time by 24.3%, travelled distance by 14.2%, and path overlap by 65.8%, with lower final coverage under the same stopping rule. The results support a coordination-efficiency benefit in the tested ROS 2 simulations; broader claims require larger, heterogeneous, dynamic, and physical deployments.



***Corresponding Author:***

Anh Tran Tuan
Posts and Telecommunications Institute of Technology
Hanoi, Vietnam
Email: anhtt@ptit.edu.vn

## 1. INTRODUCTION

Autonomous exploration lets mobile robots build maps before navigation, inspection, or automation tasks. In industrial robotics, it supports AMR/AGV warehouse mapping, production-floor inspection, and digital-twin preparation. Frontier-based exploration selects goals at the boundary between known free space and unknown space [1], and multi-robot extensions use shared information to coordinate complementary regions [2], [3].

The central problem is assigning useful frontiers without redundant team motion. A nearby target may still be poor if another robot is moving toward the same cluster or if the path overlaps in a corridor. Prior work shows that cost, information gain, overlap reduction, task-allocation mechanisms, and frontier-detection efficiency all affect team behavior [4]-[10].

Recent work has broadened the allocation design space in several directions. Voronoi and partition-based methods distribute robots across different spatial regions, sometimes combined with reinforcement learning to reduce duplicate exploration [11]-[14]. Learning-based and hybrid planning approaches modify frontier selection or path generation to improve exploration behavior in structured environments [15]-[17]. Other studies emphasize trajectory planning, auction-style task allocation, and deadlock behavior in multi-robot systems [18]-[21]. Benchmark-driven system evaluation, utility-value allocation, and multi-resolution frontier scoring further show the need to assess exploration by more than final coverage alone [22]-[25].

These studies provide important components, but many comparisons change the frontier score, assignment mechanism, and execution stack together. This paper therefore studies a narrower gap: whether global one-to-one matching improves team coordination when the frontier score, frontier source, and ROS 2 stack are fixed. The linear assignment solver is standard; the contribution is the evaluated allocation layer. The main contributions are:

– A ROS 2 exploration pipeline in which the allocation layer combines distance, active-target repulsion, robot-specific skip memory, information gain, and spatial dispersion before Nav2 goal execution.
– A controlled comparison of geometric proximity assignment (GPA), sequential cost-based assignment (RSA), and cost-aware global matching (RGM), with distance-only, distance–gain, partition-based, auction-style, and no-dispersion references.
– A simulation evaluation over two indoor maps, four team sizes, and three seeds, using completion time, final coverage, travelled distance, coverage efficiency, assignment conflict, and path overlap.

The evidence is limited to the tested indoor simulations; deployment-scale claims require larger, dynamic, heterogeneous, and physical tests.

## 2. THE PROPOSED METHOD

### 2.1. System architecture

The ROS 2 pipeline inputs per-robot poses, local SLAM maps, a merged occupancy grid, candidate frontiers, and active targets. It outputs at most one navigation goal per available robot. Figure 1 summarizes the data flow. LiDAR and odometry update local SLAM maps, local maps are merged, frontier clusters are extracted, a robot-frontier cost matrix is built, and selected goals are sent to Nav2. Nav2 feedback updates active-target and skip-memory states before the next allocation cycle. SLAM, map merging, and local navigation are standard components; the paper focuses on scoring and matching.

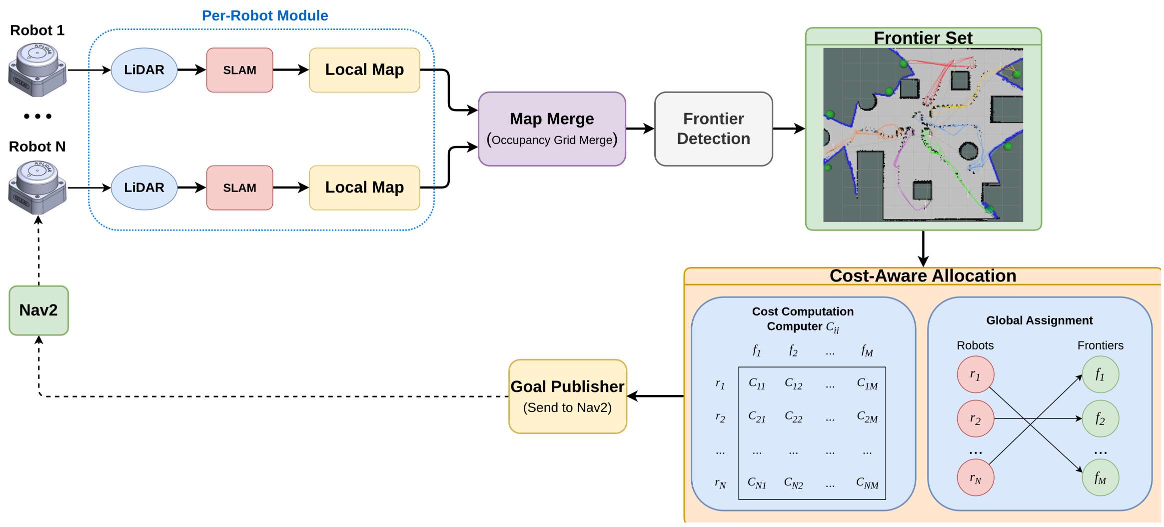


Figure 1. ROS 2 exploration pipeline with cost-aware allocation

### 2.2. Cost-aware frontier score

Let $R = \{r_1, \ldots, r_m\}$ be the set of available robots and $F = \{f_1, \ldots, f_n\}$ be the set of candidate frontiers. For robot $r_i$ and frontier $f_j$, the allocator uses the following combined cost:

$$C_{ij} = w_d \hat{d}_{ij} - w_a A_j - w_s S_{ij} + w_g G_j - w_p P_{ij} \tag{1}$$

where lower values are preferred. The terms encode distance, active-target crowding, robot-specific skip memory, frontier-cluster gain, and spatial dispersion.

Each frontier candidate is represented by a cluster centroid $c_j$ on the merged occupancy grid. Distances are measured in the shared map frame. The implementation uses the following cycle-level normalization:

$$\mathrm{norm}(z) = \frac{z - z_{\min}}{z_{\max} - z_{\min} + \varepsilon}, \tag{2}$$

where $z_{\min}$ and $z_{\max}$ are computed over values available in the current allocation cycle. The constant $\varepsilon = 0.1$ avoids division by zero and is kept fixed across all maps, team sizes, and compared policies. The terms are instantiated as

$$\hat{d}_{ij} = \frac{d_{ij}}{\max_{k,l} d_{kl} + \varepsilon} \tag{3}$$

$$A_j = \mathrm{norm}\left( \sum_{a \in T_{\mathrm{act}}} \frac{\mathbb{I}(\|c_j - c_a\|_2 < \rho_a)}{\|c_j - c_a\|_2 + \varepsilon} \right) \tag{4}$$

$$S_{ij} = \mathbb{I}(f_j \in H_i^{\mathrm{skip}}) \tag{5}$$

$$G_j = \frac{|Q_j|}{\max_k |Q_k| + \varepsilon} \tag{6}$$

$$P_{ij} = \mathrm{norm}\left( \sum_{k \neq i} \frac{1}{\|c_j - p_k\|_2 + \varepsilon} \right) \tag{7}$$

where $d_{ij}$ is the Euclidean distance from robot pose $p_i$ to centroid $c_j$, $T_{\mathrm{act}}$ is the set of active targets, $H_i^{\mathrm{skip}}$ is a per-robot skip memory, and $Q_j$ is the set of grid cells in frontier cluster $f_j$. The same parameter set is used for all maps, team sizes, and methods, as summarized in Table 1. Active-target and skip-memory parameters are fixed; sensitivity analysis is left for future work.

Table 1. Allocation parameters

| Parameter | Value | Role |
|---|---|---|
| $\varepsilon$ | 0.1 | Avoids division by zero in normalization |
| $\rho_a$ | 3.0 m | Active-target repulsion radius. |
| $w_d, w_a, w_s, w_g, w_p$ | $w_d = 0.4$, $w_a = 0.2$, $w_s = 0.1$, $w_g = 0.2$, $w_p = 0.3$ | Same cost weighting across all non-ablation cost policies. |
| Frontier representation | cluster centroid | One candidate target per frontier cluster |
| Allocation mode | batch dispatch | A new assignment is issued when available robots can receive goals |
| RGM solver | Hungarian linear sum assignment | One-to-one global matching over the current cost matrix |

### 2.3. Assignment policies and complexity

Three main assignment configurations are evaluated. GPA is the geometric reference policy and assigns frontiers using nearest-distance preference. RSA applies the combined cost in (1) sequentially, assigning robots one by one according to the currently lowest available cost. RGM uses the same cost matrix as RSA but solves all available robot-frontier pairs jointly as a one-to-one assignment.

For RGM, the binary variable $x_{ij}$ equals one when robot $r_i$ is assigned to frontier $f_j$ in the current allocation round:

$$\min_{x_{ij}} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} x_{ij}, \tag{8}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} x_{ij} \leq 1, \quad \forall i, \tag{9}$$

$$\sum_{i=1}^{m} x_{ij} \leq 1, \quad \forall j, \tag{10}$$

$$x_{ij} \in \{0, 1\}. \tag{11}$$

The first constraint allows each available robot to receive at most one frontier, and the second prevents duplicate assignment of the same frontier in one round. Cost-matrix construction requires $O(mn)$ evaluations. The Hungarian solver operates on the rectangular matrix after padding when needed, and the dominant assignment step scales cubically in the larger matrix dimension.

## 3. METHOD

### 3.1. Simulation setup and protocol

Experiments are conducted in a ROS 2 simulation environment using the same SLAM, map-merging, frontier-detection, and Nav2 navigation stack for all compared policies. The allocator is the only component changed. The benchmark contains two indoor maps: Map 1 is a cluttered open-space layout of approximately $12\,m \times 10\,m$, and Map 2 is a structured layout of approximately $15\,m \times 10\,m$ with partial partitions. The configuration can be seen in Table 2.

Table 2. Simulation and measurement configuration

| Item | Configuration used in all compared policies |
|---|---|
| Platform | ROS 2 Humble, Gazebo, homogeneous TurtleBot3 Burger robots, 0.26 m/s nominal maximum speed, simulated 2D LDS-01 LiDAR with 360° view and 0.12–3.5 m range. |
| Mapping/frontiers | SLAM Toolbox 2D LiDAR SLAM; merged occupancy grid at 0.05 m/cell; frontier cells have a free 4-neighbor, clusters use 8-neighbor BFS, $S_{\min} = 0.5$ m, and centroid targets. |
| Protocol/metrics | Shared Nav2 settings for all policies; stops at 0.90 coverage, no reachable frontier with all robots idle, timeout, or fatal failure; $d_c = 1.5$ m and 0.05 m overlap grid. |

The main comparison evaluates GPA, RSA, and RGM with 3, 4, 5, and 6 robots and seeds 0, 2, and 4, yielding 24 completed runs per policy. Paired comparisons use the same map, seed, and robot count. Across seeds, map geometry, robot model, sensor range, software stack, and parameters remain fixed; the seed changes only small start-pose perturbations and tie-breaking among equal or near-equal frontier priorities. Extended analysis uses the same protocol for GGM, DGM, PNA, AUA, and RGM-noD. Two no-dispersion no-data runs are excluded, with no outlier removal. All policies use the same LiDAR SLAM, map merging, frontier clustering, and Nav2 configuration. A run is completed when no feasible frontier remains and all robots return to idle.

Run-level metrics are completion time, final coverage, total distance, coverage efficiency, assignment conflict, and path overlap. Assignment conflict is the fraction of assignment rounds in which any pair of goals is closer than $d_c = 1.5$ m or belongs to the same frontier cluster $\kappa_i^t = \kappa_k^t$. Path overlap rasterizes TF/odometry trajectories onto a 0.05 m grid and computes $|\{q : n(q) \geq 2\}|/|\{q : n(q) \geq 1\}|$, where $n(q)$ is the number of robots visiting cell $q$. Lower conflict and overlap indicate less target crowding and less repeated traversal. We report mean $\pm$ standard deviation and paired Wilcoxon tests over matched map, seed, and robot-count settings.

## 4. RESULTS AND DISCUSSION

### 4.1. Main policy comparison

Table 3 summarizes the main comparison. RGM gives the lowest mean completion time, travelled distance, assignment conflict, and path overlap among the three main policies, while achieving the highest coverage efficiency. Its trade-off is lower final coverage than GPA under the same stopping rule, so the claim is coordination efficiency rather than coverage dominance.

Table 3. Main policy comparison

| Policy | Description | Time (s) | Cov. | Dist. (m) | Eff. | Conflict | Overlap |
|---|---|---|---|---|---|---|---|
| GPA | Geometric proximity assignment | 240.40±51.56 | 0.940±0.011 | 80.34±9.24 | 1.345±0.159 | 0.340±0.162 | 0.024±0.018 |
| RSA | Sequential cost-based assignment | 232.45±47.50 | 0.912±0.005 | 81.24±10.39 | 1.336±0.152 | 0.302±0.127 | 0.027±0.020 |
| RGM | Proposed cost-aware global matching | 176.05±34.75 | 0.896±0.021 | 69.69±7.47 | 1.549±0.114 | 0.250±0.157 | 0.009±0.012 |

Compared with RSA, RGM reduces mean completion time from 232.45 s to 176.05 s (24.3%), total distance by 14.2%, and path overlap by 65.8%. Compared with GPA, it reduces completion time by 26.8%, distance by 13.3%, and path overlap by 62.2%. Coverage efficiency increases to 1.549 $m^2$/m.

The Wilcoxon tests in Table 4 support the efficiency claim. RSA slightly increases overlap relative to GPA (0.027 vs. 0.024), whereas RGM lowers it to 0.009, indicating that the sequential greedy mechanism, not the composite cost itself, causes this degradation. RGM significantly reduces time and distance relative to GPA, RSA, GGM, AUA, and PNA; overlap reduction is significant against GPA, RSA, and AUA.

Table 4. Wilcoxon paired tests

| Comparison | Time Δ (%, $p$) | Distance Δ (%, $p$) | Overlap Δ (%, $p$) |
|---|---|---|---|
| RGM vs GPA | -64.358 (26.8%, < 0.001) | -10.650 (13.3%, < 0.001) | -0.015 (62.2%, 0.003) |
| RGM vs RSA | -56.408 (24.3%, < 0.001) | -11.552 (14.2%, < 0.001) | -0.018 (65.8%, 0.001) |
| RGM vs GGM | -64.013 (26.7%, < 0.001) | -6.218 (8.2%, 0.003) | -0.003 (21.8%, 0.363) |
| RGM vs AUA | -60.388 (25.5%, < 0.001) | -7.404 (9.6%, < 0.001) | -0.012 (56.6%, 0.010) |
| RGM vs PNA | -65.287 (27.1%, < 0.001) | -2.604 (3.6%, 0.360) | -0.004 (31.1%, 0.355) |

Figure 2 shows that RGM is faster, travels less, and reduces repeated traversal. RSA can commit early robots to locally attractive targets and leave later robots with poor team-level choices; RGM evaluates assignments jointly. The lower final coverage under RGM is therefore interpreted as an efficiency trade-off under the current stopping rule. Figure 3 illustrates the trajectory-visualization format for GPA, RSA, and RGM runs. Quantitative overlap claims remain based on Tables 3–4 and Figure 2.

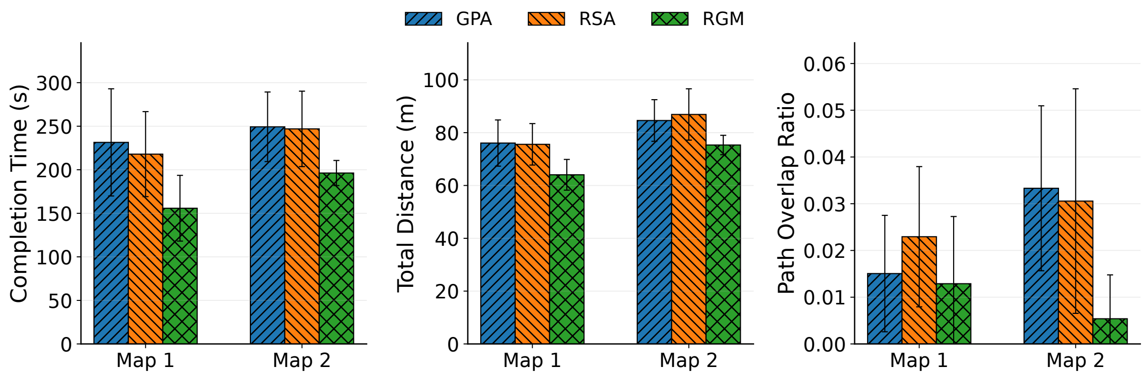


Figure 2. Main quantitative comparison for GPA, RSA, and RGM

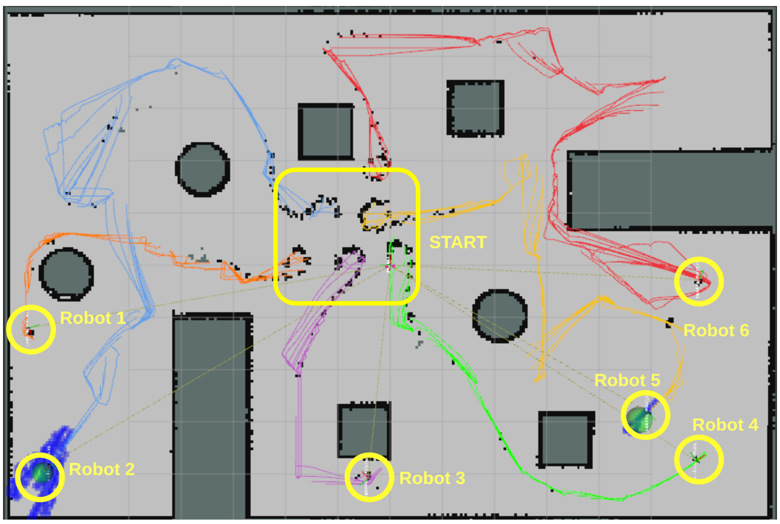


Figure 3. Trajectory visualization used for qualitative inspection of the three main policies

### 4.2. Extended baselines and ablation

Table 5 compares RGM with solver-isolation and external-style references. GGM isolates distance-only global matching, DGM adds information gain, PNA approximates partition/Voronoi-style nearest-frontier assignment, AUA approximates utility auction allocation, and RGM-noD removes dispersion. These same-stack baselines are not full reimplementations of all auction or Voronoi methods [5], [7], [8], [11]-[14]. RGM is faster than all listed variants and has the lowest mean distance except for the non-significant difference relative to PNA.

Table 5. Extended baselines and no-dispersion ablation

| Policy | Description | Runs | Time (s) | Dist. (m) | Conflict | Overlap |
|---|---|---|---|---|---|---|
| RGM | Proposed cost-aware global matching | 24 | 176.05±34.75 | 69.69±7.47 | 0.250±0.157 | 0.009±0.012 |
| GGM | Distance-only global matching | 24 | 240.06±76.96 | 75.91±7.91 | 0.370±0.177 | 0.012±0.012 |
| DGM | Distance–gain global matching | 24 | 253.31±101.93 | 75.43±12.01 | 0.258±0.154 | 0.014±0.016 |
| PNA | Partition-based nearest-frontier assignment | 24 | 241.33±58.94 | 72.29±7.59 | 0.252±0.167 | 0.013±0.014 |
| AUA | Auction-style utility assignment | 24 | 236.43±92.35 | 77.09±10.24 | 0.320±0.173 | 0.021±0.016 |
| RGM-noD | RGM without spatial dispersion term | 22 | 222.04±48.78 | 77.81±9.82 | 0.297±0.146 | 0.011±0.009 |

Against PNA, RGM has nearly the same conflict rate (0.250 vs. 0.252) and a non-significant distance difference, but reduces completion time by 27.1% ($p < 0.001$). Against AUA, RGM reduces completion time by 25.5%, distance by 9.6%, path overlap by 56.6%, and mean conflict from 0.320 to 0.250. These comparisons support the narrower claim that the tested global matching layer is more time-efficient than the implemented partition- and auction-style baselines under the same simulator, frontier source, and Nav2 stack. Relative to GGM, RGM reduces time by 26.7% and distance by 8.2%, so the gain is not from matching alone. RGM-noD further suggests that dispersion helps reduce time, distance, and conflict, although active-target and skip-memory sensitivity is not evaluated.

### 4.3. Industrial implications and limitations

The allocation layer is relevant to warehouse mapping, smart-factory commissioning, sensor-driven inspection, and digital-twin mapping, where repeated traversal increases time and energy use. The experiments remain simulation-only, with two indoor maps and homogeneous teams of up to six robots; dynamic obstacles, sensor noise, communication delay, heterogeneous platforms, and physical fleets remain future validation targets. A natural next step is learning-based frontier selection upstream of the fixed global matching layer.

## 5. CONCLUSION

This paper presented a ROS 2 frontier-allocation layer that combines weighted cost scoring with global one-to-one matching. Across 72 main-policy runs, RGM achieved the lowest mean completion time, distance, path overlap, and assignment conflict among GPA, RSA, and RGM, while improving coverage efficiency. Relative to RSA, it reduced completion time by 24.3%, distance by 14.2%, and path overlap by 65.8%, with lower final coverage under the same stopping rule. Future work will study larger maps, dynamic obstacles, sensor noise, heterogeneous fleets, communication delays, learning-based frontier selection, and physical deployment.

## ACKNOWLEDGMENTS

The authors thank AViS Lab for its research environment and support.

## FUNDING INFORMATION

Authors state no funding involved.

## AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

| Name of Author | C | M | So | Va | Fo | I | R | D | O | E | Vi | Su | P | Fu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Chu Van Cuong | | ✓ | ✓ | ✓ | ✓ | | | | ✓ | | ✓ | | | |
| Tran Tuan Anh | ✓ | | | ✓ | | | | | | ✓ | | ✓ | | |

| | | |
|---|---|---|
| C : **C**onceptualization | I : **I**nvestigation | Vi : **Vi**sualization |
| M : **M**ethodology | R : **R**esources | Su : **Su**pervision |
| So : **So**ftware | D : **D**ata Curation | P : **P**roject Administration |
| Va : **Va**lidation | O : Writing - **O**riginal Draft | Fu : **Fu**nding Acquisition |
| Fo : **Fo**rmal Analysis | E : Writing - Review & **E**diting | |

## CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

## INFORMED CONSENT

Not applicable because the study uses simulation experiments and does not involve human participants.

## ETHICAL APPROVAL

This study uses simulation experiments and does not involve human participants or animals.

## DATA AVAILABILITY

Derived metrics and configuration details are available from the corresponding author upon reasonable request.

## REFERENCES


[1] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 1997, pp. 146–151, doi: 10.1109/CIRA.1997.613851.

[2] B. Yamauchi, "Frontier-based exploration using multiple robots," in *Proceedings of the Second International Conference on Autonomous Agents*, 1998, pp. 47–53, doi: 10.1145/280765.280773.

[3] B. Yamauchi, "Decentralized coordination for multirobot exploration," *Robotics and Autonomous Systems*, vol. 29, no. 2–3, pp. 111–118, 1999, doi: 10.1016/S0921-8890(99)00046-9.

[4] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. Younes, "Coordination for multi-robot exploration and mapping," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-00)*, 2000, pp. 852–858.

[5] R. Zlot, A. Stentz, M. B. Dias, and S. Thayer, "Multi-robot exploration controlled by a market economy," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002, pp. 3016–3023, doi: 10.1109/ROBOT.2002.1013690.

[6] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005, doi: 10.1109/TRO.2004.839232.

[7] B. P. Gerkey and M. J. Matarić, "Sold!: Auction methods for multirobot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, 2002, doi: 10.1109/TRA.2002.803462.

[8] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multi-robot systems," *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954, 2004, doi: 10.1177/0278364904045564.

[9] M. Juliá, A. Gil, and O. Reinoso, "A comparison of path planning strategies for autonomous exploration and mapping of unknown environments," *Autonomous Robots*, vol. 33, no. 4, pp. 427–444, 2012, doi: 10.1007/s10514-012-9298-8.

[10] M. Keidar and G. A. Kaminka, "Efficient frontier detection for robot exploration," *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 215–236, 2014, doi: 10.1177/0278364913494911.

[11] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14413–14423, 2020, doi: 10.1109/TVT.2020.3034800.

[12] Q. Bi, X. Zhang, J. Wen, Z. Pan, S. Zhang, R. Wang, and J. Yuan, "CURE: A hierarchical framework for multi-robot autonomous exploration inspired by centroids of unknown regions," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 3, pp. 3773–3786, 2024, doi: 10.1109/TASE.2023.3285300.

[13] H. Zhao, Y. Guo, Y. Liu, and J. Jin, "Multirobot unknown environment exploration and obstacle avoidance based on a Voronoi diagram and reinforcement learning," *Expert Systems with Applications*, vol. 264, 2025, Art. no. 125900, doi: 10.1016/j.eswa.2024.125900.

[14] Y. Lei, J. Hou, P. Ma, and M. Ma, "Voronoi-GRU-based multi-robot collaborative exploration in unknown environments," *Applied Sciences*, vol. 15, no. 6, 2025, Art. no. 3313, doi: 10.3390/app15063313.

[15] R. Wang, J. Zhang, M. Lyu, C. Yan, and Y. Chen, "An improved frontier-based robot exploration strategy combined with deep reinforcement learning," *Robotics and Autonomous Systems*, vol. 181, 2024, Art. no. 104783, doi: 10.1016/j.robot.2024.104783.
[16] Y. Ning, T. Li, C. Yao, W. Du, and Y. Zhang, "HMS-RRT: A novel hybrid multi-strategy rapidly-exploring random tree algorithm for multi-robot collaborative exploration in unknown environments," *Expert Systems with Applications*, vol. 247, 2024, Art. no. 123238, doi: 10.1016/j.eswa.2024.123238.
[17] F. Baghyari, T. Parsons, J. Seo, B. Kim, M. Kim, and H. Lee, "Adaptive multi-robot exploration for unknown environments using edge-weighted path planning," *IEEE Access*, vol. 13, pp. 108127–108140, 2025, doi: 10.1109/ACCESS.2025.3581807.
[18] A. Madridano, A. Al-Kaff, D. Martin, and A. de la Escalera, "Trajectory planning for multi-robot systems: Methods and applications," *Expert Systems with Applications*, vol. 173, 2021, Art. no. 114660, doi: 10.1016/j.eswa.2021.114660.
[19] M. Zhao, H. Lu, S. Cheng, S. Yang, and Y. Shi, "A multi-robot cooperative exploration algorithm considering working efficiency and working load," *Applied Soft Computing*, vol. 128, Art. no. 109482, 2022, doi: 10.1016/j.asoc.2022.109482.
[20] X. Yan, Z. Zeng, K. He, and H. Hong, "Multi-robot cooperative autonomous exploration via task allocation in terrestrial environments," *Frontiers in Neurorobotics*, vol. 17, Art. no. 1179033, 2023, doi: 10.3389/fnbot.2023.1179033.
[21] J. Grover, C. Liu, and K. Sycara, "The before, during, and after of multi-robot deadlock," *The International Journal of Robotics Research*, vol. 42, no. 6, pp. 317–336, 2023, doi: 10.1177/02783649221074718.
[22] K. Pongsirijinda, Z. Cao, K. Bhowmik, M. Shalihan, B. P. L. Lau, R. Liu, C. Yuen, and U.-X. Tan, "Distributed multi-robot potential-field-based exploration with submap-based mapping and noise-augmented strategy," *Robotics and Autonomous Systems*, vol. 179, Art. no. 104752, 2024, doi: 10.1016/j.robot.2024.104752.
[23] D. Brugali, L. Muratore, and A. De Luca, "Mobile robots exploration strategies and requirements: A systematic mapping study," *The International Journal of Robotics Research*, vol. 44, no. 9, pp. 1461–1506, 2025, doi: 10.1177/02783649241313471.
[24] M. Zhu, Z. Liu, W. Li, S. Wang, and Q. Zhang, "Improved wavefront frontier detection-utility value task allocation for multi-robot collaborative environmental exploration," *Automation in Construction*, vol. 182, 2026, Art. no. 106740, doi: 10.1016/j.autcon.2025.106740.
[25] Z. Zhai, L. Xu, Y. Zhang, G. Zhang, and Y. Chen, "Multi-resolution field-based algorithm for autonomous robot exploration," *Scientific Reports*, vol. 16, no. 16509, 2026, doi: 10.1038/s41598-026-46119-3.

## BIOGRAPHIES OF AUTHORS

**Chu Van Cuong** is a lecturer at Posts and Telecommunications Institute of Technology, Hanoi, Vietnam. His interests include IoT, embedded systems, ROS 2 robotics, and autonomous multi-robot exploration. Email: cuongcv@ptit.edu.vn.

**Tran Tuan Anh** is a lecturer at Posts and Telecommunications Institute of Technology, Hanoi, Vietnam. He received the Ph.D. degree from Hanoi University of Science and Technology in 2021. His interests include cryptography, blockchain, artificial intelligence, photonic devices, and embedded systems. Email: anhtt@ptit.edu.vn.