A Hierarchically Integrated Framework for Resilient Task Allocation and Planning in Heterogeneous Multi-Robot Systems

Nan Li, Haris Miller, Jiming Ren, Alagappan Swaminathan,
Samuel Coogan, Karen M. Feigh, and Ye Zhao

ABSTRACT

A. Introduction and Problem Statement

Traditional methods for task allocation and planning in multi-robot teams are typically offline, assuming that all tasks are known in advance and assigned before mission execution [1], [2], [3]. However, real-world environments require resilient planning algorithms capable of handling unexpected scenarios [4], [5], such as task failures, unstructured environments, robot failures, and low battery levels.

To address these challenges, we propose a resilient task allocation and planning framework tailored for heterogeneous multi-robot systems, specifically incorporating drones and ground robots such as Turtlebots. The framework combines a high-level task allocation and planning module with a mid-level behavior tree architecture to manage unexpected events during execution.

The task allocation problem involves assigning different types of tasks to a heterogeneous robot team, where each robot has specific capabilities suited to certain tasks. The environment includes both known and unknown elements, such as static and dynamic obstacles (e.g., walls, bushes), adding uncertainty to planning and execution. The goal is to improve resilience to unexpected situations while keeping mission time short.

A video has been prepared to present our algorithm, supplementary technical details, and preliminary experimental results: https://youtu.be/7pSYnkK5tDU.

B. Contribution

This work presents a resilient and modular task allocation and planning framework for heterogeneous multi-robot systems, with the following contributions:

- Real-Time Task Reallocation Algorithm: Enables dynamic task reassignment once all tasks in a cluster are completed, improving adaptability to environmental changes and robot failures, ensuring a more efficient and resilient multi-robot system.
- Obstacle-Aware Clustering: We introduce a clustering method that incorporates obstacle-induced cost maps to improve spatial grouping of tasks. This method enables more realistic and environment-aware task allocation, particularly effective in structured environments with physical barriers such as walls.
- Reactive Behavior Tree: The addition of a mid-level behavior tree aims to supplement the higher-level planner by resolving short-horizon issues that the higher-level planner does not take into account (e.g. task failures, path conflicts, etc.).

C. Overall Framework Summary

The proposed framework follows a hierarchical and modular pipeline as shown in the right side of Figure 1:

- Task Encoding: Tasks are defined using Linear Temporal Logic (LTL), then translated into Büchi automata [6] and combined with environment models to create product automata.
- Task Clustering and Allocation: Tasks are grouped using an obstacle-aware, reward-based auction mechanism. Clustering considers robot heterogeneity and environmental obstacles through a spatial cost map.
- Dynamic Path Planning: Each robot uses a D* Lite-based planner [7], [8] on an adaptive Halton map to generate path plans, which can be updated during execution as the environment changes.
- Coordination Management: A mid-level coordination manager monitors execution and handles real-time events such as failures or path blockages using reactive strategies (e.g., behavior trees).

D. Framework Overview

Our task planner uses Linear Temporal Logic (LTL) to formally define task rules. Tasks are assigned to a heterogeneous robot team through an iterative, obstacle-aware clustered auction. Unlike previous methods [11], [12], our approach incorporates robot-specific rewards and uses a cost map to consider environmental obstacles before clustering.

The cost map assigns penalties to walls regions to discourage clustering across obstructive areas. Penalties are determined by three factors: (1) proximity to the center of a wall—closer regions imply longer detours; (2) proximity to wall junctions—junctions may indicate narrow or blocked passages; and (3) contact with map borders—which represent impassable boundaries. These cost values improve clustering by explicitly accounting for obstacles, leading to more context-aware task groupings.

During the auction phase, only certain robots are capable of performing a specific type of task (i.e., environment exploration). To balance workloads across different robot types, we introduce a reward mechanism that encourages eligible robots to prioritize these tasks over a common type

The authors are with the Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30332, USA. Email: nan.li@gatech.edu

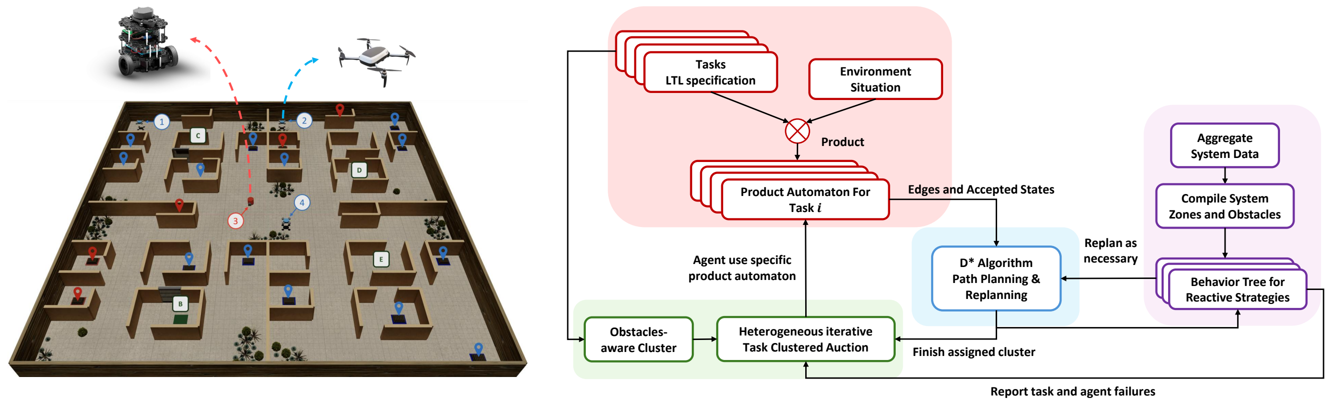

Fig. 1. The **left** side illustrates the simulated environment built in Isaac Sim [9], [10], featuring walls, iron gates, and bushes. The space is divided into four main regions, each containing several small rooms with items (indicated by red and blue markers) that need to be transported to the central green zones (labeled b through e) in a central room. The environment includes a heterogeneous team of robots consisting of three aerial drones (UAVs) and one ground robot (TurtleBot), with agents labeled 1 through 4. This setup is designed to evaluate task allocation, navigation, and coordination strategies in cluttered and semi-structured environments. The **right** side illustrates the overall framework of the hierarchical task planner. The red blocks represent task encoding components, including task specifications and environment information. The green blocks correspond to task allocation modules. The blue block denotes the D* Lite path planning module, while the purple blocks represent the coordination manager, which includes the behavior tree. The entire framework follows a hierarchical structure.

of task (in our case, pick-up and delivery) when applicable. Task allocation is conducted using an iterative, consensus-based auction algorithm [13] at the cluster level. Each robot bids on a task cluster and, once assigned, completes all tasks within that cluster. The auction continues with the remaining clusters until all tasks are allocated.

For path planning, we use the D Lite algorithm [7], [8] to compute optimal paths in partially known environments. Instead of a grid map, we use an adaptive Halton sequence-based map [14], [15], which varies point density based on obstacle distribution—placing more points near obstacles to capture complex geometry, and fewer in open areas to reduce automaton construction time. Its triangular mesh also produces smoother and shorter paths than the axis-aligned grid structure.

Beneath the high-level planner, we implement a mid-level coordination manager responsible for addressing real-time issues and managing execution-level data. A system compiler monitors the system and aggregates data during task execution, and behavior trees handle reactive strategies for critical scenarios such as agent or task failure, path blockages, and other unexpected events that may not be captured by the high-level controller [16], [17]. By reporting and resolving such issues, the coordination manager significantly enhances the planning system's resilience and adaptability in real-world environments.

E. Experimental Results

In our experiments, we evaluate the performance of two task allocation strategies: the Iterative Consensus-Based Auction Algorithm (ICBAA) [13] and the proposed Obstacles-Aware Cluster algorithm. All experiments were conducted in a structured 20×20 map environment (as shown in the left side of Figure 1) with four robots assigned to perform spatially distributed tasks.

TABLE I

PERFORMANCE METRICS FOR ITERATIVE CONSENSUS-BASED AUCTION ALGORITHM (ICBAA).

Metrics	10 tasks	15 tasks	20 tasks
Total Cost (steps)	373	580	764
Total Time (seconds)	52.43	74.45	124.14

TABLE II

PERFORMANCE METRICS FOR OBSTACLES-AWARE CLUSTER ALGORITHM.

Metrics	10 tasks	15 tasks	20 tasks	25 tasks
Total Cost (steps)	199	497	687	868
Total Time (seconds)	36.26	36.62	52.70	79.02

The ICBAA method adopts a dynamic task allocation scheme where a consensus-based auction is triggered every time a robot completes its assigned task. While ICBAA demonstrates flexibility in dynamic environments, it incurs higher total cost and longer execution time, especially as the number of tasks increases as shown in Table I.

In contrast, the Obstacles-Aware Cluster algorithm—used throughout this paper—achieves better overall performance as shown in Table II. Since the proposed algorithm performs auction at the cluster level rather than individual tasks, it significantly reduces the time required for convergence during the auction process. This makes it suitable for handling a larger number of tasks efficiently.

This framework enables resilient task allocation, planning, and operator interaction in complex and dynamic real-world scenarios. Simulation results in Isaac Sim as shown in the left size of figure 1 have been demonstrated to validate the efficacy of our proposed planning algorithm.

REFERENCES

[1] H. Chakraa, F. Guérin, E. Leclercq, and D. Lefebvre, "Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art," *Robotics and Autonomous Systems*, vol. 168, p. 104492, 2023.

[2] J. G. Martin, F. J. Muros, J. M. Maestre, and E. F. Camacho, "Multi-robot task allocation clustering based on game theory," *Robotics and Autonomous Systems*, vol. 161, p. 104314, 2023.

[3] C. Sarkar, H. S. Paul, and A. Pal, "A scalable multi-robot task allocation algorithm," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5022–5027.

[4] S. Kalluraya, G. J. Pappas, and Y. Kantaros, "Multi-robot mission planning in dynamic semantic environments," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1630–1637.

[5] S. Mayya, D. S. D'antonio, D. Saldaña, and V. Kumar, "Resilient task allocation in heterogeneous multi-robot systems," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1327–1334, 2021.

[6] M. Kloetzer and C. Belta, "A fully automated framework for control of linear systems from temporal logic specifications," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 287–297, 2008.

[7] S. Koenig and M. Likhachev, "Fast replanning for navigation in unknown terrain," *IEEE transactions on robotics*, vol. 21, no. 3, pp. 354–363, 2005.

[8] J. Ren, H. Miller, K. M. Feigh, S. Coogan, and Y. Zhao, "Ltl-d*: Incrementally optimal replanning for feasible and infeasible tasks in linear temporal logic specifications," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 4495–4502.

[9] NVIDIA. (2024) Isaac sim. [Online]. Available: https://developer.nvidia.com/isaac/sim

[10] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance gpu-based physics simulation for robot learning," *CoRR*, vol. abs/2108.10470, 2021. [Online]. Available: https://arxiv.org/abs/2108.10470

[11] M. Elango, S. Nachiappan, and M. K. Tiwari, "Balancing task allocation in multi-robot systems using k-means clustering and auction based mechanisms," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6486–6491, 2011.

[12] F. Janati, F. Abdollahi, S. S. Ghidary, M. Jannatifar, J. Baltes, and S. Sadeghnejad, "Multi-robot task allocation using clustering method," in *Robot Intelligence Technology and Applications 4: Results from the 4th International Conference on Robot Intelligence Technology and Applications*. Springer, 2017, pp. 233–247.

[13] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE transactions on robotics*, vol. 25, no. 4, pp. 912–926, 2009.

[14] H. Faure and C. Lemieux, "Generalized halton sequences in 2008: A comparative study," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 19, no. 4, pp. 1–31, 2009.

[15] X. Wang and F. J. Hickernell, "Randomized halton sequences," *Mathematical and Computer Modelling*, vol. 32, no. 7-8, pp. 887–899, 2000.

[16] M. Colledanchise, A. Marzinotto, D. V. Dimarogonas, and P. Oegren, "The advantages of using behavior trees in mult-robot systems," in *Proceedings of ISR 2016: 47st International Symposium on Robotics*. VDE, 2016, pp. 1–8.

[17] Z. Zhou, D. J. Lee, Y. Yoshinaga, S. Balakirsky, D. Guo, and Y. Zhao, "Reactive task allocation and planning for quadrupedal and wheeled robot teaming," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 2110–2117.