Self-Reactive Planning of Multi-Robots with Dynamic Task Assignments

Qin Yang[1], Zhiwei Luo[2], Wenzhan Song[2], and Ramviyas Parasuraman[1]

[1]Department of Computer Science, University of Georgia, Athens, GA 30605, USA
[2]School of Electrical and Computer Engineering, University of Georgia, Athens, GA 30605, USA

***Abstract*— Multi-Robot Systems and Swarms are intelligent systems in which a large number of agents are coordinated in a distributed and decentralized way. Each robot may have homogeneous or heterogeneous capabilities and can be programmed with several fundamental control laws adapting to the environment. Through different kinds of relationships built using the communication protocols, they present various behaviors based on the shared information and current state. To adapt to dynamic environments effectively, and maximize the utility of the group, robots need to cooperate, share their local information, and make a suitable plan according to the specific scenario. In this paper, we formalize the problem of multi-robots fulfilling dynamic tasks using state transitions represented through a Behavior Tree. We design a framework with corresponding distributed algorithms for communications between robots and negotiation and agreement protocols through a novel priority mechanism. Finally, we evaluate our framework through simulation experiments.**

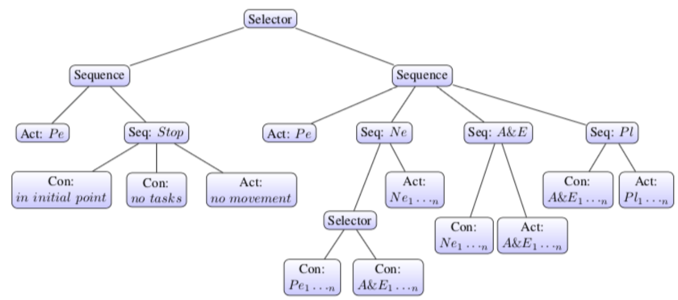

Fig. 1. Behavior Tree acting at every robot. Con - Conditions, Act - Actions, Pe - Perception, Pl - Plan, Ne - Negotiation, $A\&E$ - Agreement and Execution.

I. Introduction

Multi-robot systems (MRS) could potentially share the properties and advantages of swarm intelligence when they are deployed in various tasks such as search, rescue, mining, map construction, and exploration, etc. [1]. Robot swarms that allow task-dependent reconfiguration into a team are among the grand challenges in Robotics [2] necessitating the research at the intersection of communication, control, and perception.

Swarm Robotic System is a kind of Complex Adaptive System(CAS) [3] and has been well studied [4]. Multi-robot modeling and planning algorithms are among those well-studied topics yet possess task-specific or scenario-specific application limitations. Martinoli [5] presents the modeling technique based on rate equations, a promising formal method using temporal logic to specify and possibly prove emergent swarm behavior [6]. Soysal and Sahin [7] apply combinatorics and linear algebra to derive a model for an aggregation behavior of swarms. Some studies also applied control theory to model and analyzed multi-robot and swarm system [8], [9]. Recently, Otte et al. [10] discussed various auction methods for multi-robot task allocation problem in communication-limited scenarios where the rate of message loss between the auctioneer and the bidders are uncertain.

From the multi-agent systems perspective, one of the pioneering works especially in the distributed artificial intelligence includes [11], where the authors defined the Contract Net Protocol (CNP) for decentralized task allocation. Aknine [12] extended this idea to m manager agents and n contractor agents negotiation. A protocol for dynamic task assignment (DynCNET) has been developed by Weyns [13].

However, these methods are a centralized system and based mainly on the contractor/auctioneer and tasks to design the protocol, algorithm, and framework and less considerate of the changes in the agents' status, which restricts the direct applicability in real-world scenarios but only with adaptation. So there is a need for individual robots in an MRS to adapt to the dynamics in environments and task assignments effectively. To address these issues, we first present a framework that combines the stages of perception, communication, and control at an individual robot in an MRS through a Behavior Tree (BT [14], [15]). Then we provide several $Atomic\ Operation$ for the swarm behavior, which allows us to decompose a particular robot's actions such as flocking, pattern formation, and route planning under the same framework. Besides, we propose a priority scheme to avoid plan conflicts inspired by Maslow's human needs hierarchy [16].

II. Approach Overview

Assuming that all the robots are connected and they share a global map (through multi-robot SLAM, for example, [17]), we consider a scenario, where a group of robots will cooperate to complete some tasks. Since the tasks assignments change dynamically, the robots need to change their plans and adapt to the new scenario to guarantee the group utility. We decompose the complex tasks into a series of simple sub-tasks through which we can recursively achieve those sub-tasks until we complete the high-level task. Based on this approach, we can divide the whole task plan into three steps: selection, formation, and routing.

The process is represented as a BT in Fig. 1. In the first step, the robots are partitioned into one or more groups to perform multiple tasks such as surveillance and patrolling. Then, they will compute the placement within a formation shape (we assume a circular shape, but this can be relaxed) at each task to circle the area of the task location. Finally, the robots choose the suitable path to get to the goal point in that formation. When the new tasks are assigned, these robots need to split-up to form new groups or merge into an existing group. Accordingly, we divide the proposed self-reactive multi-robot planning into three parts.

- **Perception**
  - **Sensing** the environment through individual robot's sensor (knowing the current state of itself).
  - **Communicating** with other robot (knowing the current state of other robots).
- **Planning**
  - **Computing** based on their current state.
    - **Selection** of a suitable task and forming a group.
    - **Formation** of a desired shape within the group.
    - **Routing** plan to the task and formation placement.
  - **Negotiating** with other robots for planning.
  - **Agreement** with other robots' plan for no conflicts.
- **Execution** of the selection, formation, and routing process after agreement.

Our framework considers the following modules: Perception; Communication; Planning; Negotiation; Agreement and Execution. We will discuss them separately below.

1. Perception: Each robot uses various on-board (local) sensors for localization, mapping, and recognizing objects/obstacles in the environment.

2. Communication: The process of communication between robots includes broadcasting and reception of robot's messages (state) to/from other robots.

3. Planning: In the planning stage, we divide this process into three steps Selection, Formation, and Routing. In each step, we also introduce a priority queue technique, which can help individual robot negotiate with other robots and get an agreement efficiently.

4. Negotiation: Robots will compare their plans with that of the group members until they get an agreement.

5. Agreement and Execution: If all the robots' plans do not have conflict after the negotiation phase, they will have a final agreement per process (Selection/Formation/Routing). After the agreement, each process is executed as and when necessary.

In summary, each robot will first verify the task assignment. If assigned, it will compute an appropriate plan according to its current state and needs. Then, it will communicate with other robots and perform the negotiation and agreement process until there are no conflicts. Finally, the robots will execute their plans. This process is continuously repeated as a loop in the BT which process the flow from left to right.

In addition, to model an individual robot's motivation and needs in the negotiation process, we introduce the priority queue technique inspired by Maslow's hierarchy of human psychological needs, in which we define the robot' need hierarchy at five levels: *safety needs, basic needs (energy, time constraints, etc.), capability (heterogeneity, hardware differences, etc.), team cooperation (team performance, coordination, and global behaviors), and self-actualization (learning and self-upgrade).*

TABLE I

ENERGY LEVEL COMPARISON IN DYNAMIC TASK ASSIGNMENTS

Tasks Style	Priority	Collision	Max	Min	Mean
1+1+1	high energy	√	69.42	55.52	64.18
1+1+1	low energy	√	63.09	45.70	56.00
1+1+1	task + high energy	√	70.04	56.75	63.24
1+1+1	task + low energy	√	63.25	49.18	56.62
1+2	task + low energy	√	45.97	31.78	39.60
2+1	task + low energy	√	44.69	31.66	39.07
1+1+1	task + low energy	-	49.88	29.48	41.20
1+2	task + low energy	-	32.70	23.14	28.50
2+1	task + low energy	-	38.53	24.72	30.36

III. EVALUATION THROUGH SIMULATION STUDIES

For our experimental verification, we consider 20 fully-connected robots in an obstacle-free environment simulated in the CORE [18] network simulation framework with dynamic task assignments. A task can be added anytime during the process. For example, *1+1+1* means the formation tasks are assigned in sequential order, whereas *2+1* means two tasks are first assigned simultaneously followed by one task assignment at a later time. Suppose each robot has different battery levels in the initial state and every moving step, communication round and non-moving status will cost 0.1%, 0.01% and 0.04% energy, respectively.

We consider combinations of priorities at two levels: *high energy* as a basic need meaning that the robots with higher energy levels with get higher priority; *low energy* as a basic need; *task priority* in addition to high/low energy priorities. For example, if we adopt a priority queue with *task + low energy* combination, the application scenario would be to address the emergency task first and then maintain robots in the field as long as possible. We intend to compare the utility and behaviors of the individual robot and the system with different priority combinations. Table I provides a summary of the final energy levels in various experiments [1].

IV. CONCLUSION

Our work introduces a novel framework combining the parts of perception, communication, and control in a multi-robot system through a Behavior Tree, which not only considers individual robot's needs and action plans but also emphasizes on the complex relationships created through communications between the robots. In our framework, we propose a *negotiation* and *agreement* mechanism, which can avoid plan conflicts effectively through a prioritization scheme to manage the conflicts and the negotiation process. The experimental results show the significance of the proposed framework, aiding our future works in this direction.

[1]More details about the project are available at https://hero.uga.edu/selfreactivemrs

REFERENCES

[1] R. Parasuraman, J. Kim, S. Luo, and B.-C. Min, "Multipoint rendezvous in multirobot systems," *IEEE transactions on cybernetics*, 2018.

[2] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield *et al.*, "The grand challenges of science robotics," *Science Robotics*, vol. 3, no. 14, p. eaar7650, 2018.

[3] J. H. Holland, "Studying complex adaptive systems," *Journal of systems science and complexity*, vol. 19, no. 1, pp. 1–8, 2006.

[4] H. Hamann and H. Wörn, "A framework of space–time continuous models for algorithm design in swarm robotics," *Swarm Intelligence*, vol. 2, no. 2-4, pp. 209–239, 2008.

[5] A. Martinoli, "Swarm intelligence in autonomous collective robotics: From tools to the analysis and synthesis of distributed control strategies," Ph.D. dissertation, Citeseer, 1999.

[6] A. F. Winfield, J. Sa, M.-C. Fernández-Gago, C. Dixon, and M. Fisher, "On formal specification of emergent behaviours in swarm robotic systems," *International journal of advanced robotic systems*, vol. 2, no. 4, p. 39, 2005.

[7] O. Soysal and E. Şahin, "A macroscopic model for self-organized aggregation in swarm robotic systems," in *International Workshop on Swarm Robotics*. Springer, 2006, pp. 27–42.

[8] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 539–557, 2004.

[9] J. T. Feddema, C. Lewis, and D. A. Schoenwald, "Decentralized control of cooperative robotic vehicles: theory and application," *IEEE Transactions on robotics and automation*, vol. 18, no. 5, pp. 852–864, 2002.

[10] M. Otte, M. J. Kuhlman, and D. Sofge, "Auctions for multi-robot task allocation in communication limited environments," *Autonomous Robots*, pp. 1–38, 2019.

[11] R. G. Smith and R. Davis, "Frameworks for cooperation in distributed problem solving," *IEEE Transactions on systems, man, and cybernetics*, vol. 11, no. 1, pp. 61–70, 1981.

[12] S. Aknine, S. Pinson, and M. F. Shakun, "An extended multi-agent negotiation protocol," *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 1, pp. 5–45, 2004.

[13] D. Weyns, N. Boucké, T. Holvoet, and B. Demarsin, "Dyncnet: A protocol for dynamic task assignment in multiagent systems," in *First International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2007)*. IEEE, 2007, pp. 281–284.

[14] M. Colledanchise and P. Ögren, *Behavior Trees in Robotics and AI: An Introduction*. CRC Press, 2018.

[15] M. Colledanchise, R. N. Parasuraman, and P. Ogren, "Learning of behavior trees for autonomous agents," *IEEE Transactions on Games*, 2018.

[16] A. H. Maslow, "A theory of human motivation." *Psychological review*, vol. 50, no. 4, p. 370, 1943.

[17] R. Valencia and J. Andrade-Cetto, "Active pose slam," in *Mapping, Planning and Exploration with Pose SLAM*. Springer, 2018, pp. 89–108.

[18] J. Ahrenholz, "Comparison of core network emulation platforms," in *2010-Milcom 2010 Military Communications Conference*. IEEE, 2010, pp. 166–171.