A Hierarchical Approach to Workstation-based Task Allocation and Motion Planning

Isaac Ngui[1], Scott Lee[2], James Motes[1], Marco Morales[1], and Nancy M. Amato[1]

***Abstract*— This paper introduces a hierarchical approach to workstation-based task allocation and motion planning problems for on-demand and reconfigurable factory environments. This problem is composed of two sub-problems: workstation task planning and payload transportation. This hierarchical approach abstracts away workstation details during payload transportation and payload transportation details away during workstation task planning, enabling scalable planning for large numbers of robots and workstations. This hierarchical approach is expected to offer adaptable solutions for various workstation-based factory scenarios, promoting high throughput while maintaining flexibility.**

I. INTRODUCTION

On-demand manufacturing, or individualism on demand [1], is a growing area where manufacturing is transitioning to producing only what is needed when it is needed reducing bulk products sitting in warehouses. While industry transitions to on-demand manufacturing with current technologies, the full advantages of this new approach requires new technologies and methods motivating research in this area [2], [3]. One research thrust is in reconfigurable factories designed to autonomously adapt themselves to new layouts when new products need to be manufactured.

A standard factory consists of workstations dedicated to specific tasks (e.g., assembly [4], spot welding [5], [6], or inspection [7]) and the transportation of components between workstations. The use of robotic manipulators at workstations and autonomous mobile robots for transportation allows factories to be reconfigured for new tasks and products. This adaptability enables modern on-demand manufacturing as factories can be quickly shifted to make new products without long down times and expensive custom machines.

The physical adaptability of the robots provides one component of an autonomous factory. However, to truly achieve fully autonomous on-demand manufacturing, there is a need for planning algorithms which quickly find plans for robots in these adaptable environments. These planning algorithms must account for the highly coordinated actions that manipulators perform at workstations as well as the transportation of payloads between workstations.

This leads to a natural hierarchical approach where workstations are treated as black boxes in the payload transport problem. Meanwhile, workstation plans can ignore the delivery and removal of additional components. This hierarchical separation of the problem allows for planning algorithms to scale to larger numbers of robots and workstations.

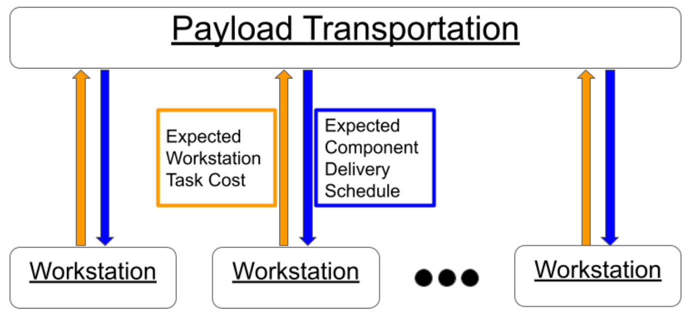

Fig. 1: This figure depicts the hierarchical approach proposed in this extended abstract.

In this work, we consider a factory environment with a set of workstations and mobile robots. The workstations perform tasks using various components. The mobile robots transport the necessary components to and from the workstations. The objective is to find plans for the workstation tasks and payload transport plans to achieve some desired criteria.

In this extended abstract, we discuss a hierarchical approach to this problem and propose some preliminary applications to it. In Section II, explore two real world examples of this problem. In Section III, we define the payload transport and workstation problem and cover existing approaches to them and in Section IV we discuss our proposed approach.

II. MOTIVATING EXAMPLES

This section explores motivating examples of this problem.

1) Assembly Planning: An intuitive example of this problem is assembly planning. Workstations are assigned various stages of product assembly. Planning for the robots at workstations can assume that components needed for their task are present and that completed assemblies will be picked up. This enables planning algorithms to leverage more expensive techniques for the highly coordinated problem of robotic assembly while ignoring the context of the larger factory.

Meanwhile, the payload transport problem can abstract out the details of workstations and model a fleet of mobile robots moving components and sub-assemblies to and from workstations. This black box abstraction includes the time to complete assembly tasks at workstations and the time required by mobile robots to interact with the workstations. By approaching the problem hierarchically, computational resources can be spent either on the intricate details of highly coordinated plans of a few robots at a workstation

[1]Isaac Ngui, James Motes, Marco Morales, and Nancy M. Amato are with Parasol Lab, Department of Computer Science, University of Illinois at Urbana-Champaign (email: ingui2, jmotes2, moralesa, namato@illinois.edu)

[2]Scott Lee is with Parasol Lab, Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign (email: sl148@illinois.edu)

or accounting for a large number of robots in the less complicated payload transport problem.

2) Wet Lab Data Collection: This hierarchical approach extends beyond obvious manufacturing problems. A recent inspiring example is the Mind-in-Vitro (MiV) project [8], which aims to construct a computing system with living neurons. These living neurons are grown in a bio-fab setting requiring constant monitoring. Robots at workstations are configured to record the connection between input stimulation and neuronal response. The living neurons are grown in a separate controlled environment, thus creating the need for mobile robots to transport them between the growing chambers and the workstations. As this is an emerging field, the proper way to grow and measure these neurons is still an area of research, motivating the need for robotic planning algorithms to adapt to new configurations.

III. Background and Preliminaries

This section discusses background information for the payload transport and workstation planning problems.

1) Payload Transportation: The payload transport problem consists of transporting objects to and from locations. In the context of manufacturing, these locations are either workstations or part depots, and the payloads are the components or products being manufactured. This problem has been well studied and is often modeled as a multi-robot task allocation (MRTA) problem [9]. In recent years, the problem has been expanded to the simultaneous task allocation (or target assignment) and pathfinding (TAPF) problem to account for the physical constraints of mobile robots executing the task allocation in a physical environment [10]. These methods rely on highly structured environments where grid world representations of the environment are sufficient for solving the payload transport problem. [11], [12]

One such approach, PC-TAPF [13], uses a grid world representation to model a factory setting for the payload transport problem where mobile robots move partially completed sub-assemblies between workstations according to an assembly sequence. Work done at workstations is abstracted into cost functions which PC-TAPF includes in a mixed-integer linear program (MILP) model of the task allocation problem. This allows the method to ignore the complexity of the workstations while finding plans for the fleet of mobile robots to facilitate the flow of components and subsassemblies between workstations. The simplification of the motion planning to a grid world representation and the abstraction of workstation activity to a simple cost function allows the method to solve problems for up to 40 robots and 20 components/tasks.

When path planning between workstations moves beyond the capacity of grid world representations, either from more complicated environments, or robots which cannot be represented on a grid (e.g., manipulators), many of the underlying multi-robot pathfinding methods used in TAPF methods such as CBS [11], have been generalized to sampling-based motion planning [14]. The same generalizations can often extend the TAPF methods to handle more complex motion requirements [15]. Additionally, many problems will involve more complex interactions with workstations (e.g., picking/placing components). This can require more expensive planning which can hinder the ability of a TAPF approach to plan for larger numbers of robots or payloads. However, like the actions done at workstations, these complex interactions can be abstracted out as cost functions for a more simple representation of the payload transport problem.

2) Workstation Task Planning: Planning for these tasks generally requires task and motion planning (TMP) methods [16], where high-level reasoning is performed to determine actions to complete the task, and low-level motion planning is performed to determine if the action can be executed by the robots.

The Decomposable State Space Hypergraph (DaSH) method [17], a recent multi-robot task and motion planning method, provides a new paradigm for efficient multi-robot planning. It utilizes a hypergraph-based framework to model transitions in the level of coordination required for planning, coupling robots for interactions such as object handoffs, and decoupling robots during independent movements. This approach results in a cheaper representation of the planning space which the authors leverage in heuristics for multi-manipulator rearrangement planning problems. They demonstrate planning times up to three orders of magnitude faster than prior methods and solve problems with up to 20 objects.

IV. Proposed Approach

We propose a hierarchical framework which leverages the hierarchical nature of the problem which lets us ignore unnecessary details at different levels of the planning framework (illustrated in Figure 1). The framework utilizes a task and motion planning (TMP) layer and a task assignment and pathfinding (TAPF) layer. The TMP layer is used to compute robot trajectories to complete tasks at workstations and a cost estimate of completing the task. The workstation only needs to know when the components are arriving and departing, and all other payload transportation details can be ignored.

The TAPF layer solves the payload transportation problem and producing a schedule for the mobile robots to follow when transporting components to and from workstations. This layer uses the cost estimates from the TMP layer in order to produce the schedule for transport. This layer leverages the fact that the payload transportation problem does not need any details about completing the workstation task except for the cost of completing the task.

Additionally, we propose an adaptation of the DaSH method [17] to optimize the cycle times of workstation tasks for the repetitive workstation tasks in this problem. We expect the proposed method to solve the general factory planning problems for the assembly and the MiV [8] problems described in Section II in addition to other problems which can be modeled in this way. This will enable an autonomous factory to autonomously adapt to new products and tasks while maintaining a high throughput.

This method will be developed in the open source Parasol Planning Library [18].

REFERENCES

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & information systems engineering*, vol. 6, pp. 239–242, 2014.

[2] T. Chen, Z. Huang, J. Motes, J. Geng, Q. M. Ta, H. Dinkel, H. Abdul-Rashid, J. Myers, Y.-J. Mun, W. che Lin, Y. yung Huang, S. Liu, M. Morales, N. M. Amato, K. Driggs-Campbell, and T. Bretl, "Insights from an industrial collaborative assembly project: Lessons in research and collaboration," 2022.

[3] M. Saez and P. Spicer, "Robot-to-robot collaboration for fixtureless assembly: Challenges and opportunities in the automotive industry," in *International Manufacturing Science and Engineering Conference*, vol. 84263, p. V002T07A012, American Society of Mechanical Engineers, 2020.

[4] J. Chen, J. Li, Y. Huang, C. Garrett, D. Sun, C. Fan, A. Hofmann, C. Mueller, S. Koenig, and B. C. Williams, "Cooperative task and motion planning for multi-arm assembly systems," *arXiv preprint arXiv:2203.02475*, 2022.

[5] H. Touzani, H. Hadj-Abdelkader, N. Séguy, and S. Bouchafa, "Multi-robot task sequencing automatic path planning for cycle time optimization: Application for car production line," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1335–1342, 2021.

[6] B. Zhou, R. Zhou, Y. Gan, F. Fang, and Y. Mao, "Multi-robot multi-station cooperative spot welding task allocation based on stepwise optimization: An industrial case study," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102197, 2022.

[7] Y. Liu, W. Zhao, T. Lutz, and X. Yue, "Task allocation and coordinated motion planning for autonomous multi-robot optical inspection systems," *Journal of Intelligent Manufacturing*, pp. 1–14, 2021.

[8] X. Zhang, Z. Dou, S.-H. Kim, G. Upadhyay, D. Havert, S. Kang, K. Kazemi, K.-Y. Huang, O. Aydin, R. Huang, *et al.*, "Mind in vitro'platforms: Versatile, scalable, robust and open solutions to interfacing with living neurons," *bioRxiv*, pp. 2023–08, 2023.

[9] A. Khamis, A. Hussein, and A. Elmogy, "Multi-robot task allocation: A review of the state-of-the-art," *Cooperative robots and sensor networks 2015*, pp. 31–51, 2015.

[10] H. Ma and S. Koenig, "Optimal target assignment and path finding for teams of agents," in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pp. 1144–1152, International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[11] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artificial Intelligence*, vol. 219, pp. 40–66, 2015.

[12] W. Hönig, S. Kiesel, A. Tinka, J. W. Durham, and N. Ayanian, "Conflict-based search with optimal task assignment," in *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems*, pp. 757–765, International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[13] K. Brown, O. Peltzer, M. A. Sehr, M. Schwager, and M. J. Kochenderfer, "Optimal sequential task assignment and path finding for multi-agent robotic assembly planning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 441–447, IEEE, 2020.

[14] I. Solis, J. Motes, R. Sandström, and N. M. Amato, "Representation-optimal multi-robot motion planning using conflict-based search," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4608–4615, 2021.

[15] J. Motes, R. Sandström, H. Lee, S. Thomas, and N. M. Amato, "Multi-robot task and motion planning with subtask dependencies," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3338–3345, 2020.

[16] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez, "Integrated task and motion planning," *Annual review of control, robotics, and autonomous systems*, vol. 4, pp. 265–293, 2021.

[17] J. Motes, T. Chen, T. Bretl, M. M. Aguirre, and N. M. Amato, "Hypergraph-based multi-robot task and motion planning," *IEEE Transactions on Robotics*, 2023.

[18] "Parasol planning library." https://gitlab.engr.illinois.edu/parasol-group/parasol/open-ppl.