# 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]

University of Illinois at Urbana-Champaign (Department of Computer Science[1], Department of Mechanical Science and Engineering[2])


## Motivation

On-demand manufacturing is a modern approach which produces only the products that are needed, when they are needed

- Reduces the amount of bulk product sitting in warehouses

Autonomous reconfigurable factories are used to meet on-demand manufacturing needs while utilizing off-the-shelf robots

- Removes the need for expensive custom machines
- Adapts the factory layout for new products with minimal down time

There is a need for efficient planning algorithms for robots operating in reconfigurable factory environments

## Problem Definition

We consider a factory environment with:

- Mobile robot teams which transport components to and from workstations
- Workstations that perform tasks using sets of components

Objective: Find schedules for completing workstation tasks and plan safe trajectories for all robot teams

## Applications

**Assembly Planning**

- Goal: Assemble products from their individual components by following an assembly plan
- Mobile robots transport components and sub-assemblies to and from workstations
- Workstations collectively assemble the product through an assembly process

**Wet Lab Data Collection**

- Mind-in-Vitro Problem (MiV)[1]
- Goal: Collect data from cells in a wet lab using heterogeneous teams of robots
- Mobile robot teams transport cells to different workstations
- Workstations are tasked with collecting data using manipulators equipped with various sensors

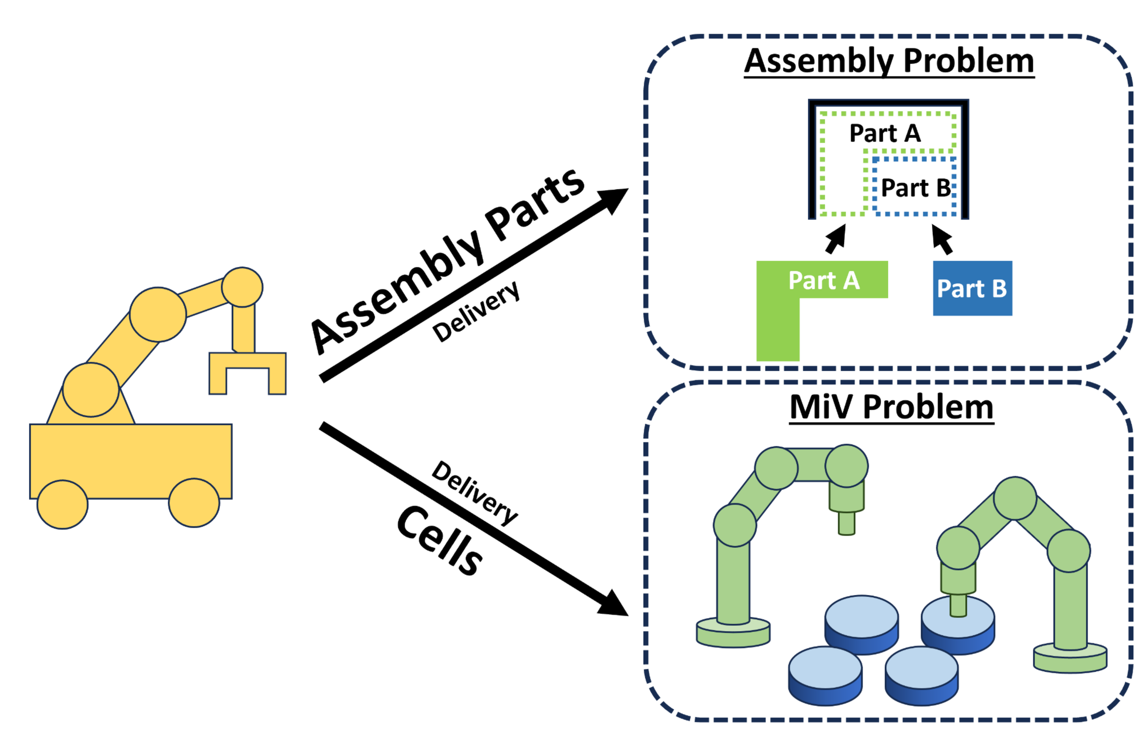


*Figure 1. An illustration of the workstation and payload transportation task for two different example scenarios.*

## Scheduling and Pathfinding

**Payload Transportation Problem**

- Goal: Assign and route mobile manipulators to transport objects between workstations

**Task Allocation Problem**

- Goal: Compute feasible schedules for tasks and allocate them to robots
- Precedence Constrained Multi-Robot Task Assignment and Pathfinding (PC-TAPF) [2]
  - Abstracts work completed by workstations into cost functions
  - Uses a Mixed Integer Linear Program (MILP) to assign tasks to robots while accounting for the cost to complete workstation tasks

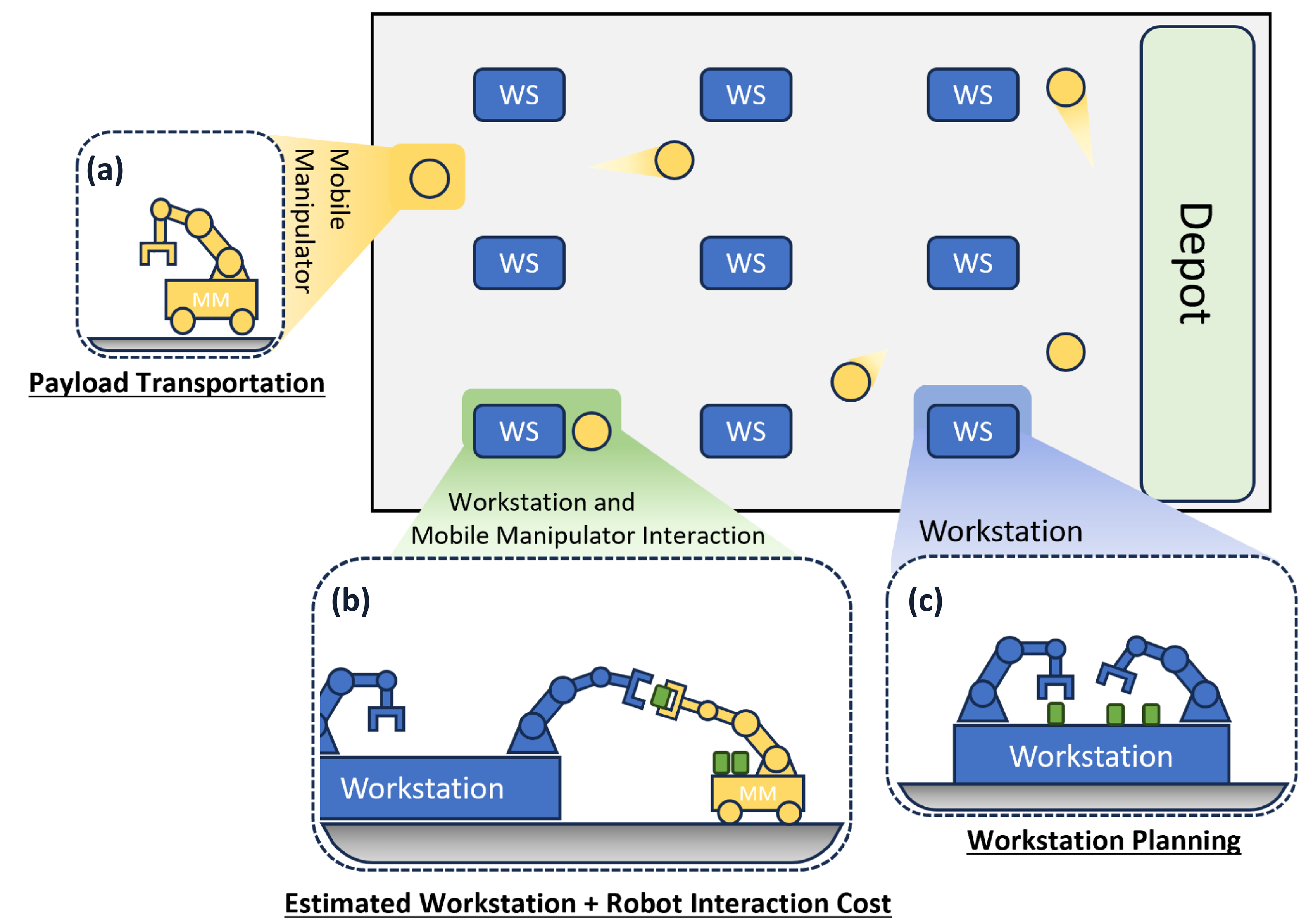


*Figure 2. An illustration of the factory settings with mobile manipulators and workstations.*
*(a) A mobile manipulator tasked to deliver the sub-assemblies to the workstation.*
*(b) Handoff interaction between the workstation and mobile manipulator.*
*(c) A workstation assigned to perform the sub-tasks with the sub-assemblies.*

## Workstation Planning

**Workstation Interactions**

- Compute coordinated trajectories for mobile manipulators to deliver components to workstations

**Task and Motion Planning (TMP)**

- Multi-robot task planning determines which subtasks should be performed and which robots should complete each subtask
- Motion planning generates safe and efficient robot trajectories for the subtasks
- Decomposable State-Space Hypergraph (DaSH) [3]
  - Utilizes a hypergraph-based representation to model transitions
    - Couples robots for collaborative interactions
    - Decouples robots for individual actions
  - Results in a cheaper representation of the planning space making planning simpler and generally faster

## Proposed Approach

**Hierarchical Framework**

- Each planning layer focuses on a specific TMP component and focuses only on the relevant details necessary to generate its plans
- Information in the form of solution costs are passed between layers to create a cohesive planning framework
- TMP Layer
  - Used to find robot trajectories to complete tasks assigned to workstations
  - Cost of completing the task is passed to the TAPF layer
- TAPF Layer
  - Used to solve the payload transportation problem
  - Utilizes task completion times provided by the TMP layer to schedule mobile robots to transport objects between workstations

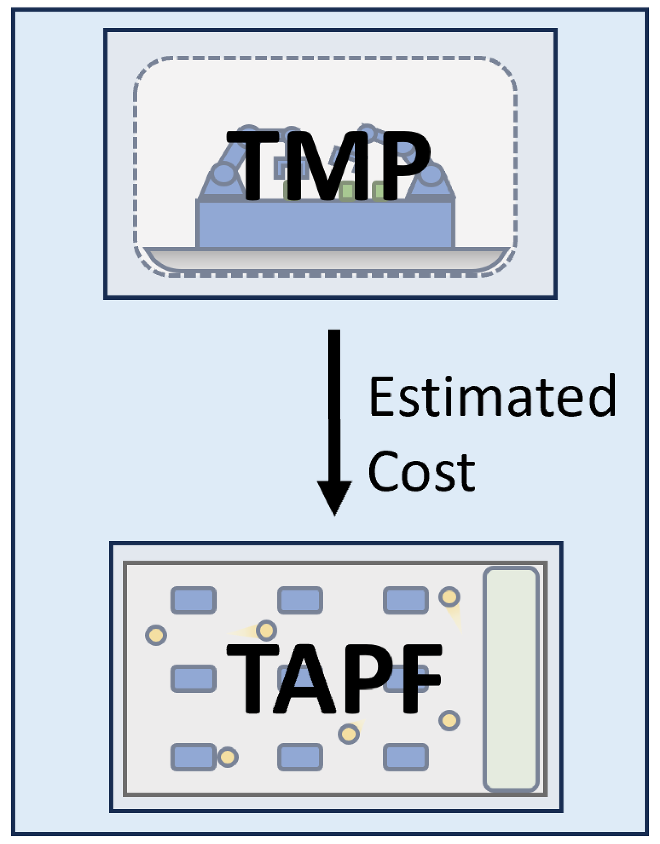


*Figure 3. Framework for TAPF and TMP*

**Cost Estimates**

- Cost estimates are calculated in each individual layer to provide information in other planning layers
- Allocation in the TAPF layer depends on the cost of completing tasks at workstations, which is calculated in the TMP layer
  - For repetitive tasks, we propose a novel DaSH [3] variant which computes cycle-time optimal plans for the workstations

## References


[1] Zhang, Xiaotian, et al. "Mind in Vitro'platforms: Versatile, scalable, robust and open solutions to interfacing with living neurons." bioRxiv (2023): 2023-08.

[2] 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.

[3] 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.