Collaborative Multi-Robot Systems with Provable Availability, Safety, and Optimality Guarantees

PI: Jingjin Yu - Rutgers University at New Brunswick

Project Synopsis

A **dynamic** multi-robot dispatching problem in which tasks arrive continuously in an unpredictable manner.

Algorithmic issues rising from the logistic automation domain possess unique features that distinguish them from well-studied problems. On one hand, classical pickup and delivery problems (PDP) do not model the non-trivial geometry of physical robots and the possible collisions among multiple robots sharing a limited workspace. On the other, multi-robot path and motion planning research has yet to systematically address the coordination of hundreds to thousands of robots for the **continuous execution of dynamic and stochastic tasks**. The proposed study intends to fill this gap through the modeling and subsequent algorithmic resolution of the problem, which we call the **dynamic multi-robot dispatching problem (DMD)**. Depending on the specific application domain, DMD may be subdivided into unlabeled (e.g., container unloading from ships) and labeled (e.g., order fulfillment) variants, providing rich grounds for structural exploration. Despite the fact that optimal multi-robot coordination is a computationally intractable problem, preliminary efforts indicate that approximately optimal solutions could be computed in polynomial time, through the careful integration of the state-of-the-art multi-robot motion planners and the global coordination of robot flows. Following this route, the proposed research will develop algorithmic solutions for DMD with provable availability and optimality guarantees under stringent safety assurances for human co-workers. Working with collaborators, the research will also seek to maximize its applicability to industrial setups.

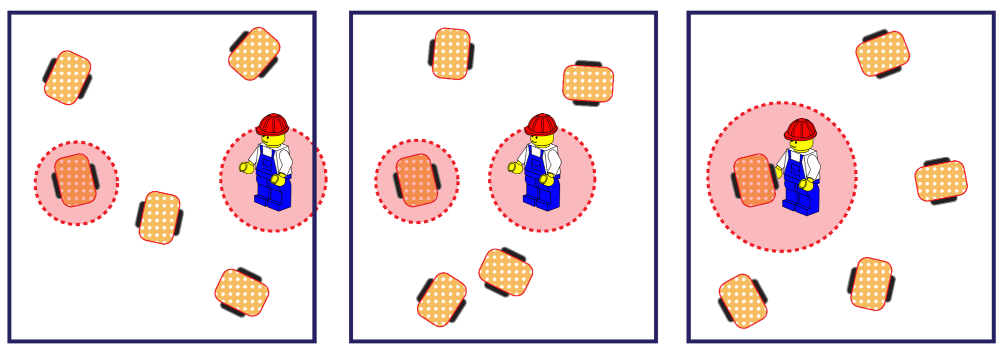
The Dynamic Multi-robot Dispatching Problem

The dynamic multi-robot dispatching (DMD) problem has two variants: **unlabeled** dynamic multi-robot dispatching problem (**UDMD**) and **labeled** dynamic multi-robot dispatching problem **(LDMD).**

A **UDMD** problem is described by a 4-tuple (G, D, X_I, S) in which G is a the graph modeling the workspace of the robots. D describes the arrival process of the tasks for the robots. X_I is the set of vertices of G where the robots initially reside. S is the set of graph vertices modeling the task locations (e.g., a shelf that a mobile robot needs to retrieve) for the robots. A UDMD problem has only one type of robots: robots that are indistinguishable. UDMD can be proven to be computationally "easy" in that it admits many opportunities for efficient optimization due to the robots being unlabeled (see top figure on the right for the intuition).

An **LDMD** problem may be defined similarly as UDMD with the key difference that the robots are labeled, meaning that the robots are not interchangeable. This happens as each robot must go to a specific location, e.g., when a robot is delivering a specific load to a specific target location. LDMD is a much harder problem than LDMD.

Together, UDMD and LDMD combine to yield many variations of full DMD problems. For example, in the container port example, we essentially have a UDMD-LDMD problem where one corresponds to unloading of containers from the ship and the other loading containers onto trucks. In attacking the various DMD formulations, we are interested in both **practical solutions** and **provable guarantees**. In particular, we would like to deliver efficient algorithms for solving the multiple DMD formulations that ensure the resulting system to have **guaranteed availability**, i.e., guaranteeing that the system can handle certain load conditions. At the same time, the algorithm must be adaptive and **safe**, i.e., it should have sufficient "buffer" to absorb unexpected system breakdowns which require human intervention (see figure on the right). Lastly, we will also push the optimality of the system with guaranteed availability and safety.



Research Plan and Result Highlights

The process will be executed in three phase spanning the three years of the project life-span.

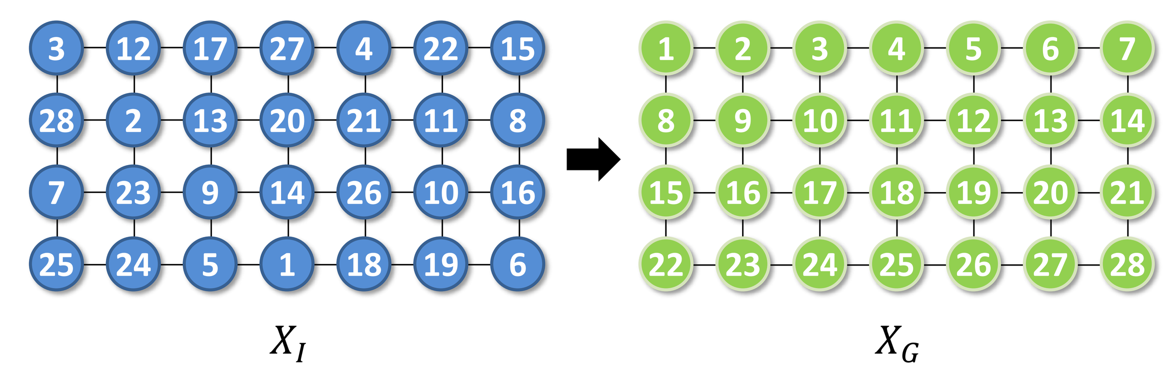
- **Year one (09/2017-08/2018)**: Focusing on the **optimality** aspect of the static multi-robot systems applicable toward logistics applications
- **Year two (09/2018-08/2019)**: Introducing dynamics into the multi-robot collaboration problem and studying the **availability** perspective
- **Year three (09/2019-08/2020)**: Integrating availability, **safety**, and optimality guarantees into a full and complete algorithmic solution

During the past year (first year of the project), significant progress has been made on optimally (and efficiently) solving static cases of multi-robot collaboration for logistic setups. These progress has resulted in the following publications:

- R. Chinta, S. D. Han, and J. Yu. Coordinating the Motion of Labeled Discs with Optimality Guarantees under Extreme Density. WAFR 2018
- J. Yu. Constant Factor Time Optimal Multi-Robot Routing on High-Dimensional Grids. RSS 2018.
- S. D. Han, N. M. Stiffler, A. Krontiris, K. E. Bekris, and J. Yu. Complexity Results and Fast Methods for Optimal Tabletop Rearrangement with Overhand Grasps. IJRR 2018.
- S. D. Han, E. J. Rodriguez, and J. Yu. SEAR: A Polynomial-Time Multi-Robot Path Planning Algorithm with Expected Constant-Factor Optimality Guarantee. IROS 2018.
- S. D. Han, N. M. Stiffler, K. E. Bekris, and J. Yu. Efficient, High-Quality Stack Rearrangement. RA-L 2018.
- J. Yu. Expected constant-factor optimal multi-robot path planning in well-connected environments. MRS 2017.

We highlight on the right a recent breakthrough of ours regarding the optimality-efficiency barrier of multi-robot coordination, appeared in this year's Robotics: Science and Systems conference (2nd paper in the list above).

Multi-Robot Routing in High Dimensions

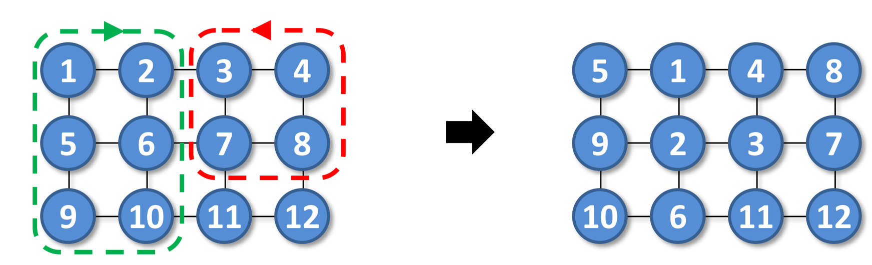


A multi-robot routing problem is specified by a 3-tuple (G, X_I, X_G) in which

- $G = (V, E)$ is an n-vertex connected graph.
- $X_I = \langle X_I(1), \ldots, X_I(n)\rangle$, as a function $X_I: N \to V$, is the initial configuration of the n robots
- $X_G = \langle X_G(1), \ldots, X_G(n)\rangle$, also a function $X_G: N \to V$, is the goal configuration of the n robots.

In our case, G is a k dimensional grid with $n = m_1 \times \cdots \times m_k$, in which $m_i, 1 \le i \le k$ is the length or size of a dimension i. In the above example, we have $k = 2, m_1 = 7, m_2 = 4, n = m_1 \times m_2 = 28$.

Given the problem setup, which implies maximum robot density, the only allowed moves are synchronous rotations of robots along disjoint cycles on G. E.g., the following figure shows ten robots rotate along two disjoint cycles which can be completed in a single (time) step, or **makespan**.



Two synchronous rotations that take 1 (time) step

Main Results

Given an instance $p = (G, X_I, X_G)$, define its **distance gap** $d_g(p)$ as

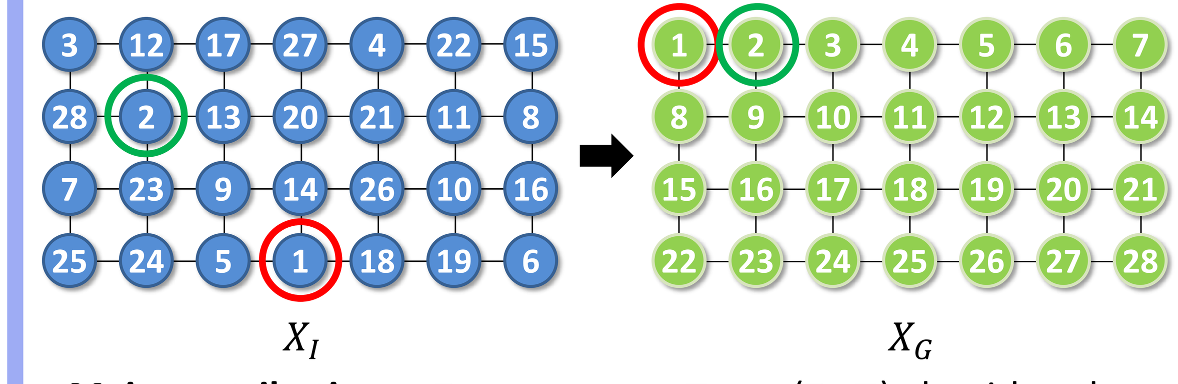
$$d_g = \max_{1 \le i \le n} dist(X_I(i), X_G(i))$$

e.g., for the example problem above (repeated here), we may compute the individual $dist(X_I(i), X_G(i))$ and then d_g as:

$dist(X_I(1), X_G(1)) = 6$, $dist(X_I(2), X_G(2)) = 1$, ... , and

$$d_g(p) = \max_{1 \le i \le 28} dist(X_I(i), X_G(i)) = 8.$$

The max distance of 8 is achieve by robot $i = 7$, among others.



Main contribution: a PartitionAndFlow (PaF) algorithm that

- Computes an $O(d_g(p))$ makespan time-optimal plan,
- Runs in $O(|V|^2)$, i.e., quadratic time in the worst case, and
- Works for arbitrary fixed dimension k and supports maximum robot density.

More formally,

Theorem (PartitionAndFlow in k Dimensions). Let $G = (V, E)$ be an $m_1 \times \cdots \times m_k$ grid with $m_1 \ge \cdots \ge m_k \ge 2$ and $m_{k-1} \ge 3$. Let p be an arbitrary instance of the multi-robot routing problem on G. Then, PaF computes a solution path set with $O(d_g(p))$ makespan in $O(d_g^k(p)|V|)$ and also $O(|V|^2)$ running time.

That is, the PaF algorithm computes a **constant-factor time-optimal** solution in **low polynomial time** for an arbitrary instance of the multi-robot routing problem on k-dimensional grids. For large k and when G is not degenerate, PaF runs in strictly sub-quadratic time. Moreover, when $m_1, \ldots, m_k$ are of similar magnitudes, the running time of PaF is almost linear, i.e., approaching $O(|V|)$.

IIS-1734419 NRI: FND: Collaborative Multi-Robot Systems with Provable Availability, Safety, and Optimality Guarantees. PM: Ralph Wachter