DOCTORAL THESIS

A Hybrid Task Allocation and Motion Planning Framework for Mobile Robots

Jennifer David

Halmstad University | School of Information Technology
DoctoralThesis | Halmstad University Dissertations no. 140

A Hybrid Task Allocation and Motion Planning Framework for Mobile Robots

Jennifer David

Supervisors

Thorsteinn Rögnvaldsson

Co-supervisors

Mattias Ohlsson

Rafael Valencia

DOCTORAL THESIS | Halmstad University Dissertations no. 140

A Hybrid Task Allocation and Motion Planning Framework for Mobile Robots

Halmstad University Dissertation No. 140
ISBN 978-91-90123-06-5 (printed)
ISBN 978-91-90123-07-2 (pdf)
Publisher: Halmstad University Press, 2025 | www.hh.se/hup
Printer: Media-Tryck, Lund

எனக்கா இத்தன கிருபை!
என் மேல் அளவற்ற கிருபை!

Abstract

This thesis investigates the challenges associated with task allocation and motion planning in dynamic and complex environments involving fleets of mobile robots. The primary objective is to coordinate task allocation and trajectory planning in a manner that promotes balanced workload distribution, formulated as the minimization of the maximum operational cost incurred by any individual robot.

A hybrid planning framework is proposed in which centralized task allocation and global path planning are performed under static assumptions, while execution-level feasibility is maintained through local trajectory refinement. Task allocation is formulated using a permutation-matrix representation and explored using multiple solution strategies, including deterministic annealing with Potts neurons, stochastic simulated annealing, and heuristic graph-based methods. The cost matrix, derived from global path planners and representing path length and/or traversal time, enables a systematic comparison of these approaches in terms of solution quality, computational complexity, and scalability. The results indicate that deterministic annealing can produce high-quality, balanced task allocations for small- to medium-scale problem instances, while heuristic methods offer improved robustness and computational efficiency in larger or time-critical scenarios.

At the motion-planning level, the framework adapts the CHOMP (Covariant Hamiltonian Optimization for Motion Planning) algorithm to account for the kinematic constraints of non-holonomic wheeled mobile robots. Rather than serving as a standalone global planner, the modified CHOMP formulation is used as a local trajectory refinement mechanism, enabling collision avoidance and feasibility preservation during execution. Experimental and simulation results demonstrate that this approach improves trajectory smoothness and feasibility in moderately dynamic environments, while also highlighting limitations related to scalability and sensitivity to initialization.

Overall, this thesis presents an integrated task and motion planning framework that emphasizes structured problem formulation and systematic trade-off analysis rather than universal optimality. By explicitly examining the conditions under which different allocation and motion-planning strategies are effective, the work contributes practical insights into multi-robot coordination and supports the informed deployment of autonomous robotic

systems in industrial and logistics automation.

Acknowledgements

This thesis emerged from what can be seen as a multi-robot routing problem. While the work concerns a complete system framework, it was guided and shaped by my supervisors — the global planners — who defined feasible paths through a high-dimensional and complex research space.

I learned immensely during the first six months of my PhD from Dr. Roland, not only as an engineer but also as a human being. I sincerely thank Dr. Rafael for showing me how to carry intelligence with humility and always supporting me. Thanks a tonne to Prof. Karl for demonstrating the value of working efficiently and independently. I am deeply grateful to Prof. Thorsteinn, who exemplified what it means to be a truly dedicated researcher; if I am ever considered a good researcher, I owe much of that to your guidance. I also thank Prof. Mattias for his expertise and for helping me navigate entirely new research areas with such clarity. I am grateful to Dr. Pietro from ABB CRC for trusting in me and for extending my work toward industrial application. I would also like to thank Prof. Bosse from Lund University — it was a privilege to work with someone who could map complex neural network models in his mind and bring them to life on the whiteboard. I thank all my collaborators at ABB CRC, Lund University, Volvo GTT, IRI UPC, MIT Boston, ICT, and TNO, who helped pave the way throughout this journey.

My fellow researchers and lab mates were the true agents in this system. Through coordination and collaboration, you helped resolve conflicts and avoid deadlocks. Prof. Antanas and Prof. Stefan consistently set examples of leadership one could rely on. Prof. Slawomir, Dr. Björn, Dr. Jens, Dr. Eren, and Dr. Martin repeatedly stepped in with a global perspective whenever I found myself trapped in local minima.

The environment included constraints both expected and unforeseen. During these times, my friends provided robustness and fault tolerance, helping maintain system stability through uncertainty. I am grateful for the presence of remarkable women — Anna, Deycy, Sepideh, and Linda — and dear friends such as Saeed and Yuantao, who stood by me throughout. This journey would not have been the same without the companionship of Pascal, Yod, Asif, Gaurav, Kevin, Hassan, Mijin, Mahsa, and Suleyman.

My family formed the core infrastructure of this system — steady, reli-

able, and essential — enabling progress even when re-routing was required. Mom, Dad, João, Heps, Dia, Wills, Rea, Joanan, Niranjani, Indhu, Sanjana, Monisha, Dr. Ken and all my amazing friends all over the world supporting me through different phases of this PhD annealing - this solution would not have converged without any of you. Finally, I thank my Jesus for being always there for me!

Like any realistic multi-robot routing problem, this PhD involved detours, deadlocks, and frequent replanning. I thank my supervisors for helping me re-evaluate cost functions and constraints when progress stalled. This work is the result of collective effort — a coordinated solution achieved through communication, trust, and persistence. I am deeply grateful to all who contributed, directly or indirectly, to reaching this final destination.

Table of Contents

List of Publications

PAPER I: **Multi-Robot Routing Problem with Min–Max Objective**
Jennifer David and Thorsteinn Rögnvaldsson
MDPI Robotics, 10(4), 122, 2021.

PAPER II: **Deterministic annealing with Potts neurons for multi-robot routing**
Jennifer David, Thorsteinn Rögnvaldsson, Bo Söderberg and Mattias Ohlsson
Intelligent Service Robotics 15(3), pp. 321–334, 2022.

PAPER III: **Gradient-based path optimization method for autonomous driving**
Jennifer David, Rafael Valencia, Roland Philippsen and Karl Iagnemma
IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4501-4508, Vancouver, Canada, 2017.

PAPER IV: **Local Path Optimizer for an Autonomous Truck in a Harbor Scenario**
Jennifer David, Rafael Valencia, Roland Philippsen and Karl Iagnemma
Field and Service Robotics (FSR) Conference, Springer Proceedings in Advanced Robotics, vol 5, pp. 499-512, Zurich, Switzerland, 2017.

PAPER V: **Task assignment and trajectory planning in dynamic environments for multiple vehicles**
Jennifer David, Rafael Valencia and Karl Iagnemma
Robotics: Science and System (RSS) Workshop on Task and Motion Planning, Ann Arbor, Michigan, USA, 2016.

PAPER VI: **Simultaneous Path planning and Assignment under Dynamic En-**

vironments
Jennifer David and Rafael Valencia
MDPI Robotics, 14(2), 17, 2025.

Other publications:

1. **J. David**, Valencia, R., Philippsen, R. and Iagnemma, K. Trajectory Optimizer for an Autonomous Truck in Container Terminal. **IEEE International Conference on Robotics and Automation (ICRA), Workshop on Robotics and Vehicular Technologies for Self-driving cars**, Marina Bay Sands, Singapore, 2017.

2. A. Häggström, K.J. Djervbrant., **J. David** and T. Rögnvaldsson. Capacitated Multi Depot Green Vehicle Routing for Transporting End-of-Life electrical waste. **31st European Conference on Operational Research (EURO)**, Athens, 2021.

3. **J. David**, W. Mostowski, M. Aramrattana, Y. Fan, M. Varshosaz, P. Karlsson, M. Roden, A. Bogga, J. Carlsen, E. Johansson and E. Andersson. Design and Development of a Hexacopter for the Search and Rescue of a Lost Drone. **IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) - Workshop on Challenges in Vision-based Drones Navigation**, Macau, China, 2019.

4. Cooney, M., Yang, C., Padi Siva, A., Arunesh, S., **David, J.** Teaching Robotics with Robot Operating System (ROS): A Behavior Model Perspective. **European Robotics Forum (ERF) - Workshop on 'Teaching Robotics with ROS'**, Tampere, Finland, 2018.

5. W. Taha, L. Hedström, F. Xu, A. Duracz, F. Bartha, Y. Zeng, **J. David**, and G. Gunjan. Flipping a first course on cyber-physical systems: an experience report. **Workshop on Embedded and Cyber-Physical Systems Education (WESE)**, New York, 2016.

Abbreviations

ABmTSP Asymmetric Bottleneck multiple Traveling Salesman Problem

AEP Alliance Efficiency Problem

AGV Automated Guided Vehicle

AMR Autonomous Mobile Robots

ASC Automated Stacking Cranes

AT Automated Truck

CHOMP Covariant Hamiltonian Optimization for Motion Planning

CTA Container Terminal Altenwerder

DA Deterministic Annealing

HNN Hopfield Neural Network

ITAPF Integrated Task Assignment and Path Finding

ITAPP Integrated Task Allocation and Path Planning

ITMP Integrated Task and Motion Planning

ITPP Integrated Task and Path Planning

MAPF Multi Agent Path Finding

MFT Mean Field Theory

MILP Mixed Integer Linear Programming

MMMDmATSP Min–Max Multiple Depot multiple Asymmetric Traveling Salesman Problem

MMMDVRP Minimum Maximum Multiple Depot Vehicle Routing Problem

MRC Multi Robot Coordination

MRMP Multi Robot Motion Planning

MRPP Multi Robot Path Planning

MRR	Multi Robot Routing
MRR-MM	Multi Robot Routing with min-max objective
MRS	Multi Robot System
MRTA	Multi Robot Task Allocation
MRTP	Multi Robot Task Planning
PRM	Probabilistic Road Maps
RRT	Rapidly Exploring Random Trees
SPADES	Simultaneous Path planning and task Allocation in Dynamic EnvironmentS
TAMP	Task and Motion Planning
TAPF	Target Assignment and Path Finding
TAPP	Task and Path Planning
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem

List of Figures

List of Tables

1. Introduction

1.1 Motivation

The global cargo industry relies heavily on maritime routes, which account for over 60% of worldwide freight transport. This reliance has fueled significant growth in international container trade, introducing logistical complexities such as managing rising demand, capacity bottlenecks, and complying with increasingly stringent environmental regulations. Concurrently, the rapid expansion of e-commerce over the past decade has reshaped logistics and supply chain operations, necessitating faster, more efficient, and scalable systems. Traditional warehouses and fulfillment centers face substantial challenges in meeting these evolving demands, particularly with respect to order processing speed, inventory accuracy, and reverse logistics, highlighting the limitations of manual operations.

At container terminals, automated systems such as Automated Stacking Cranes (ASC), Automated Trucks (AT), and Automated Guided Vehicles (AGV) facilitate high-volume container handling, reducing turnaround times and congestion significantly. These robotic solutions optimize workflows and align operational practices with sustainability targets by decreasing energy usage and emissions, thus meeting global environmental objectives [1]. Similarly, within warehouse environments, Autonomous Mobile Robots (AMR) and robotic manipulators considerably improve efficiency by automating repetitive tasks such as picking, packing, and sorting, leading to accelerated and more accurate order fulfillment [2]. Real-time inventory management utilizing drones and robotic systems further enhances operational precision and scalability, especially during peak demand periods.

This transformation in logistics is exemplified by highly automated facilities such as Rotterdam's Euromax Terminal and Hamburg's Container Terminal Altenwerder (CTA), as well as by companies including Amazon Robotics and Ocado Robotics. Figure 1.1 illustrates a simplified scenario in a warehouse where two robot types undertake different tasks based on their capabilities. Similarly, Figure 1.2 provides a simplified representation of automation within a container terminal, highlighting the potential for robots to substantially improve operational efficiency and throughput.

1.2 Cargo-ANTs - EU-FP7 Project

Much of the work reported in this thesis was done within the framework of the EU-FP7 project **Cargo-ANTs**[1] and the aim was to enhance autonomous freight transportation using Automated Guided Vehicles (AGVs) and Automated Trucks (ATs), enabling them to operate safely and efficiently in port environments without relying on costly infrastructure like electromagnetic guidance grids. The primary objective is to develop a planning framework that ensures AGVs and ATs can efficiently pick up and transport containers to designated goals without collisions, even in dynamic settings. The Cargo-ANTs project thus forms the foundation for the system design and experimental validation carried out in this thesis. Similar coordination challenges arise in other domains, such as warehouse automation and ride-hailing services, where multiple robots or autonomous agents must efficiently allocate tasks and execute corresponding motion plans. Such systems, generally referred to as Multi-Robot System (MRS), offer distinct advantages over single-robot configurations but also encounter substantial technical and computational challenges arising from the intrinsic complexity of the underlying NP-hard optimization problems. These challenges collectively fall under the broader domain of *motion coordination* for fleets of autonomous mobile robots.

1.3 Problem Statement

Motion coordination in multi-robot systems (MRS) involves several critical challenges that depend on the system's objectives, constraints, and operational context which is detailed in the next chapter. In real-world logistics environments, such as warehouses and container terminals, key challenges include navigation, task allocation, perception, resource management, and path/motion planning. Despite significant progress, efficiently coordinating multiple mobile robots in these complex settings remains a challenging problem due to scalability, collision avoidance, and dynamic task requirements. Motivated by the requirements of the Cargo-ANTs EU project, this thesis addresses *motion planning for multiple mobile robots*. The goal is to develop frameworks that enable safe, efficient, and coordinated robot operations in real-world logistics scenarios.

Traditionally, two main approaches have been employed to address motion planning for multiple robots: centralized and decentralized methods [3]. Centralized approaches combine the configuration spaces of all robots into a single composite space, enabling globally optimal solutions but suffering

[1] https://cordis.europa.eu/project/id/605598/reporting

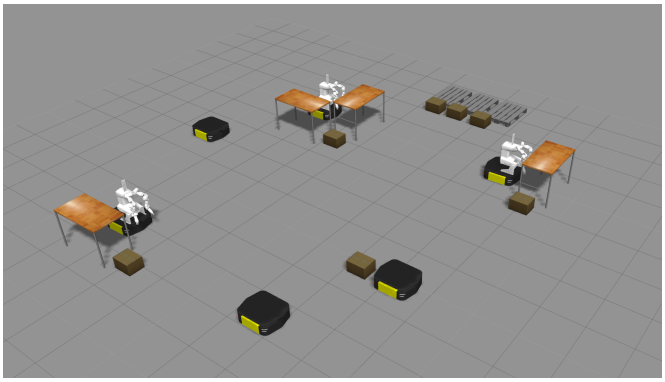


Figure 1.1: Simplified warehouse scenario illustrating two robot types – Ridgeback wheeled mobile robots and Yumi manipulators mounted on Ridgeback platforms – executing tasks according to their capabilities in a Gazebo simulation.

from prohibitive computational complexity and poor scalability [4]. In contrast, decentralized or decoupled approaches compute paths independently for each robot and then resolve conflicts between paths [5]. While computationally efficient, these methods typically produce suboptimal solutions.

More recently, a third class of approaches has emerged that leverages techniques from Operations Research to jointly address task allocation and motion planning [6]. By integrating these traditionally separate problems, such methods strike a balance between solution optimality and computational tractability, often outperforming conventional decoupled approaches in both efficiency and solution quality. This thesis builds upon and adapts these task-and-motion planning approaches, combining the advantages of centralized and decentralized strategies to generate near-optimal solutions with significantly improved computational efficiency [6–8].

1.4 Thesis Objective

This thesis aims to design and develop a robust and computationally efficient hybrid framework for multi-robot motion planning based on an approach that jointly considers task allocation and motion planning. Specifically, the proposed framework integrates the following components in a hybrid way:

- A centralized task allocation method designed to generate near-optimal solutions for practical multi-robot motion planning (MRMP) scenarios.

- A distributed motion planner capable of dynamically refining robot trajectories to preserve task allocation optimality while efficiently resolving conflicts and avoiding collisions.

The framework emphasizes solution quality, evaluated primarily in terms of trajectory optimality and computational efficiency, and is validated on small- to medium-scale problems involving non-holonomic mobile robots. It targets realistic operational environments such as warehouses and container terminals, and demonstrates coordinated operation of robot fleets at small to medium experimental scale.

1.5 Research Questions

The overarching research question addressed in this thesis is:

> **RQ1**: *What are the key requirements for integrating task allocation and motion planning into a unified framework for multi-robot systems, and how can such a framework be effectively realized?*

To systematically address RQ1, three sub-questions are formulated and grouped under two primary themes: **Task Allocation** and **Motion Planning**.

> **RQ2**: *What type of problem formulation supports efficient task allocation while balancing workloads across multiple robots?*
>
> **RQ3**: *What methods can be employed to improve the solution quality of such a task allocation formulations?*
>
> **RQ4**: *How can trajectory optimization techniques be adapted to generate smooth, collision-free, and dynamically feasible paths for non-holonomic robots operating in realistic environments?*

Together, these research questions provide a structured foundation for developing a hybrid planning framework that enhances robustness, scalability, and real-world applicability in multi-robot systems.

1.6 Contributions

Based on the four research questions, this thesis explores four key research topics, and its contributions are presented in six appended papers. The following section provides a high-level overview of the contributions of each appended paper, emphasizing their role within the overall thesis rather than their full technical details.

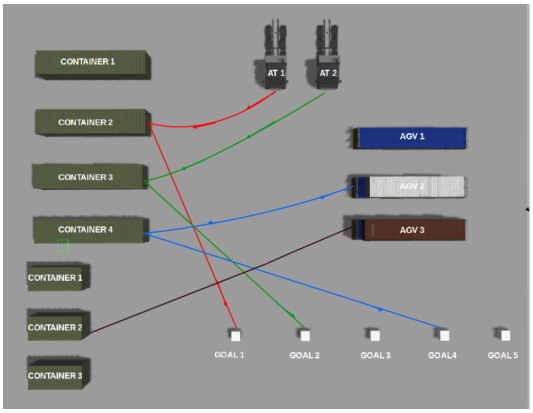


Figure 1.2: Simplified representation of task allocation in a container terminal involving automated trucks (ATs) and automated guided vehicles (AGVs) in a Gazebo simulation.

1.6.1 Paper 1: The Multi-Robot Routing problem with min-max objective (MRR-MM)

This paper presents a comprehensive study of the Multi-Robot Routing with Min-Max objective (MRR-MM) problem, highlighting its relevance and the diverse approaches adopted across different research communities. A central contribution is a novel formulation of the MRR-MM problem using a permutation matrix representation, where the search space consists of all the possible permutation matrices of a given problem size. The optimal solution is defined as the permutation matrix that minimizes the maximum time required by the slowest robot, rather than the total or average travel time. This objective promotes balanced workload distribution and reduces the likelihood of robot overuse or failure.
The key contributions of this paper include:

- Introduction of a structured permutation matrix representation for task allocation in the MRR-MM problem.
- A systematic comparison of three algorithmic approaches, highlighting their respective strengths and limitations.
- Evaluation on real-world datasets to demonstrate practical applicability.
- Identification of trade-offs between computational efficiency and solution optimality to guide method selection under different application constraints.

1.6.2 Paper 2: Deterministic Annealing with Potts Neurons

Building upon Paper 1, this paper introduces a Deterministic Annealing (DA) approach to solve the MRR-MM problem by searching for optimal permutation matrices using mean-field dynamics. Unlike prior DA-based methods that rely on Potts vectors, this work maps the problem onto a Potts matrix, resulting in a more expressive and effective representation. The proposed approach is evaluated in terms of scalability, solution quality, and computational efficiency, and is compared against Simulated Annealing (SA) and a heuristic method [7]. Experimental results demonstrate that the Potts neuron-based DA approach achieves superior performance in both solution quality and runtime, particularly for NP-hard multi-robot routing variants.
The key contributions of this paper include:

- Integration of Potts neurons as a Potts matrix within a DA methodology for solving multi-robot routing problem.
- Detailed algorithmic formulation of the DA optimization process and its convergence behavior.
- Extensive experimental validation focusing on scalability, computational efficiency, and solution quality compared to other generic methods.

1.6.3 Papers 3 and 4: Gradient-based Motion Planners for TAMP method

These two papers focus on developing a motion planner tailored for the MRR-MM problem. Specifically, they explore different strategies for adapting the CHOMP [9] algorithm to handle non-holonomic vehicles, differential-drive robots and multiple mobile robots. The proposed methods are validated in both simulation and real-world experiments, including a truck–trailer system navigating cluttered environments. Results indicate that while standard CHOMP struggles to fully capture complex kinematic constraints, the adapted formulations are effective in moderately dense environments.
The key contributions of these papers include:

- Adapting CHOMP (originally developed for robotic manipulators) for practical use in autonomous driving and mobile robot applications.
- Successfully incorporating *sliding and rolling constraints* into trajectory optimization to achieve smooth curvature tracking for non-holonomic vehicles.

- Highlighting the importance of proper constraint handling (particularly rolling constraints) to manage vehicle-specific kinematic constraints.

- Exploring the effects of incorporating separate objective functionals alongside the smoothness objective for creating a distributed variant to work with multiple robots.

1.6.4 Papers 5 and 6: Development of SPADES and Extended-SPADES Frameworks

These works present SPADES (Simultaneous Path Planning and Task Allocation in Dynamic Environments) and its extension, Extended-SPADES, which together form a lightweight yet effective framework for addressing the MRR-MM problem. SPADES considers scenarios with an equal number of robots and tasks (M = N), performing task allocation via the Hungarian algorithm [10] in a static environment, while employing a distributed motion planner that dynamically refines robot trajectories based on priority. Extended-SPADES broadens this approach to more realistic settings in which the number of robots is smaller than the number of tasks ($M \leq N$), using a distributed multi-CHOMP algorithm. Collectively, these methods enhance adaptability, scalability, and fleet-level cost efficiency for teams of mobile robots.
The key contributions of these papers include:

- Grouping, ordering, and optimally assigning tasks to robots using DA-based or heuristic allocation methods, followed by conversion into executable motion plans via precomputed spatial cost estimates.

- Continuous trajectory refinement in dynamic environments using an extended CHOMP-based distributed motion planner that accounts for temporal and spatial constraints.

By integrating these contributions, this thesis provides a *robust, scalable, and computationally efficient framework* for multi-robot task and motion planning in dynamic, real-world environments.

In this thesis, the term **integrated** refers to the explicit coupling between task allocation and motion planning decisions, while the term **hybrid** describes the architectural combination of centralized task allocation with distributed, local trajectory optimization.

1.7 Outline of this Thesis

The rest of the thesis is organised as follows: **Chapter 2** provides an in-depth review of related work and literature survey on the main topics under study. **Chapter 3** summarizes the appended papers, detailing the key contributions and findings of each. **Chapter 4** discusses the advantages and limitations of the proposed methods. It also highlights potential future research directions and possible extensions of the work. After Chapter 4, the list of references for the complete thesis and all the published papers are appended.

2. Background

2.1 Terminology

2.1.1 Decision-Making in Autonomous Systems

Autonomous mobile robots typically employ hierarchical frameworks for decision-making, with each level addressing different navigation and control objectives. These layers collectively facilitate safe, efficient, and goal-oriented operations, particularly in dynamic environments. Hierarchical decision-making architectures are widely used in autonomous robotics to decompose planning and control across multiple abstraction levels [3]. It typically consists of the following three layers:

- **High-Level Planning (Task and Path Planning)**: High-level planning involves strategic, long-term decisions about what tasks the robot must perform and the optimal routes to achieve these tasks. *Task planning* specifies a sequence of goal-oriented actions, while *path planning* identifies optimal collision-free routes to these goals, usually based on global environmental maps. Such plans are typically computed offline or updated at lower frequencies, assuming relatively static conditions.

- **Mid-Level Planning (Trajectory Planning)**: Mid-level planning dynamically generates feasible trajectories by incorporating the robot's kinematic and dynamic constraints, sensor feedback, and real-time information on environmental changes and obstacles. Unlike global path planning, trajectory planning continuously adapts to environmental uncertainties and ensures safe and realistic robot motions during operation.

- **Low-Level Control (Actuation and Execution)**: Low-level control is responsible for translating planned trajectories into physical actuation commands. This layer accounts for the robot's mechanical limitations, such as maximum velocity and acceleration limits, and precisely controls motors, wheels, and actuators to ensure accurate real-time movement execution.

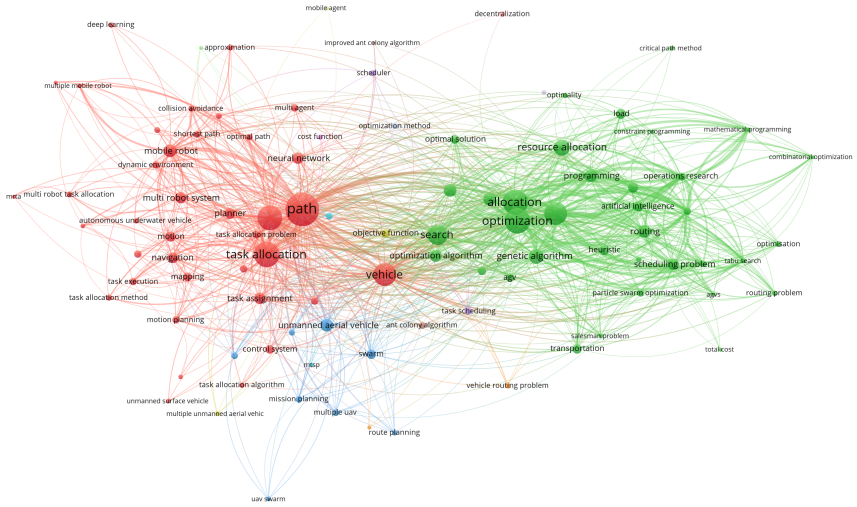


Figure 2.1: Research trends in multi-robot systems (2015–2025), illustrating the increasing integration of task allocation and motion planning approaches.

Advanced integrated planning frameworks, such as Task and Motion Planning (TAMP), Integrated Task and Motion Planning (ITMP), and Integrated Task and Path (ITPP), effectively bridge the gap between high-level planning and low-level execution:

- **Task and Motion Planning (TAMP/ITMP)**: This framework integrates task-level decisions with motion feasibility considerations, ensuring tasks assigned at the strategic level are practically achievable given physical, kinematic, and obstacle constraints.
- **Integrated Task and Path Planning (ITPP)**: ITPP coordinates the assignment of tasks and their corresponding path planning to ensure robots not only receive feasible tasks but also efficiently navigate to task locations while minimizing travel costs and avoiding collisions.

2.1.2 Multi-Robot Systems (MRS)

When multiple robots collaborate in dynamic environments, additional complexities arise—task allocation, coordination, and multi-robot motion planning. **Multi-Robot Coordination (MRC)** is crucial for ensuring system efficiency, scalability, and safety in cooperative robot teams. It involves task allocation, communication, and collision avoidance strategies that allow multiple robots to function collectively in complex environments. A

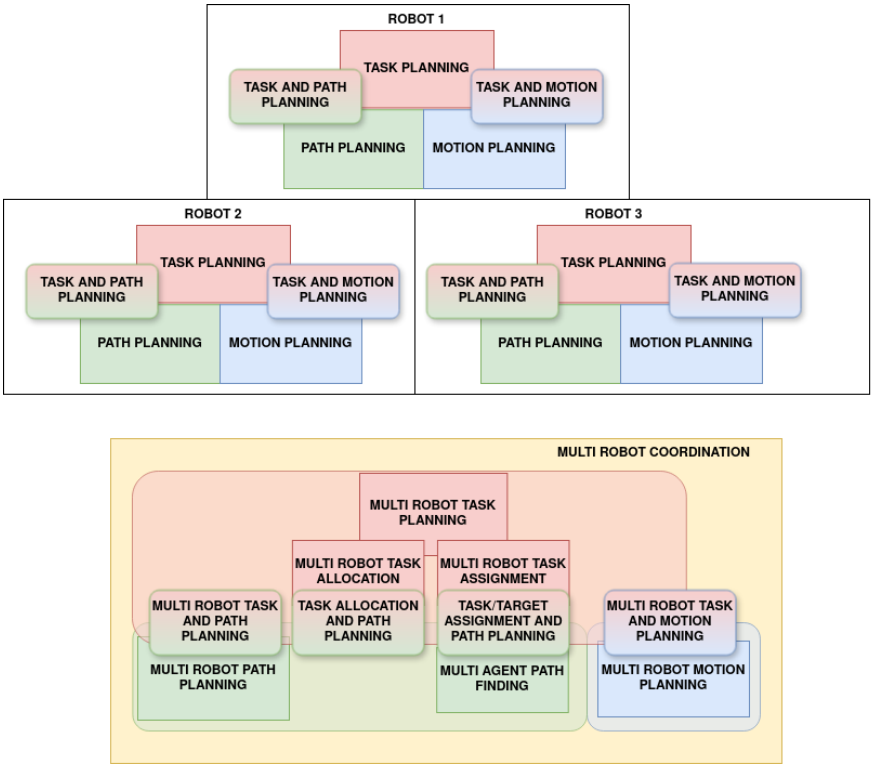


Figure 2.2: Different types of MRS problems in comparison to a single-robot system.

useful analogy is a team sport, where each agent must decide when to move, how to act, and how to coordinate with teammates to achieve a shared goal. Figure 2.1 summarizes some major MRS research trends from 2015 to 2025.

In this thesis, the focus is on *Integrated Task and Motion Planning (ITMP)* but for MRS, where task allocation decisions are tightly coupled with motion feasibility and trajectory optimization in multi-robot systems. These integrated approaches are crucial for real-world autonomous systems. In autonomous warehouse operations, for instance, TAMP frameworks ensure efficient task assignments considering both strategic objectives and operational feasibility, while ITPP ensures seamless, safe navigation to target locations. Similarly, autonomous robots rely on task planning for decisions like route selection based on traffic conditions, global path planning (e.g., using algorithms like A* to navigate the road network), and local trajectory adjustments to dynamically respond to pedestrians or other obstacles. In search-and-rescue scenarios, multi-robot systems apply integrated task planning to distribute coverage tasks efficiently and dynamic motion plan-

ning to navigate complex terrains or debris. By systematically integrating task planning, path planning, and motion control, autonomous robotic systems achieve robust adaptability, enhanced safety, and operational efficiency across diverse, dynamic environments.

The key problem domains within MRS are outlined here as: **Multi-Robot Task Planning (MRTP)** refers to the high-level coordination and sequencing of tasks among multiple robots to achieve a shared objective. Unlike single-robot task planning, MRTP involves distributing and synchronizing tasks across multiple agents in dynamic and uncertain environments. **Multi-Robot Task Allocation (MRTA)** [11–13] is a subproblem of MRTP that focuses on assigning robots to one or more tasks based on task complexity, robot capabilities, and environmental constraints. MRTA aims to optimize workload distribution and ensure efficient task execution. **Multi-Robot Task Assignment** is a specific step within MRTA where tasks are assigned to individual robots or robot teams after deciding how tasks should be distributed. **Multi-Robot Path Planning (MRPP)** [14–16] focuses on computing collision-free paths for multiple robots moving in a shared space while considering static obstacles. **Multi-Robot Motion Planning (MRMP)** extends MRPP by incorporating real-time movement coordination in dynamic environments, accounting for robot kinematics, moving obstacles, and the need for continuous re-planning. Another related domain is **Multi-Agent Path Finding (MAPF)** [17–19], which is more abstract and applies to a broader range of agents beyond robots. MAPF focuses on spatial–temporal coordination but lacks explicit physical motion constraints. This makes it less suitable for systems requiring kinematically feasible trajectory execution.

2.1.3 Integrated Approaches in Multi-Robot Systems

Multi-Robot Task and Motion Planning (MR-TAMP) and **Multi-Robot Task and Path Planning (MR-TAPP)** [20] extend single-robot TAMP and TAPP to multi-robot systems. These problems are inherently more complex due to the need for real-time task reassignment, inter-robot collision avoidance, and inter-robot coordination. A more recent challenge in MRS is the **Integrated Task Assignment and Path Finding (ITAPF)** problem [21; 22], which unifies task allocation and path finding into a single framework. ITAPF is designed for short-term, real-time execution-based coordination, such as UAVs dynamically delivering packages.

Similarly, the **Integrated Task Allocation and Path Planning (ITAPP)** problem focuses on long-term planning where task execution involves sequential decision-making (e.g., robots in a production line). Unlike ITAPF, which prioritizes immediate task execution, ITAPP ensures that assigned

tasks are strategically planned over an extended period. A closely related problem is **Target Assignment and Path Finding (TAPF)**, which assigns fixed goal locations to robots while ensuring collision-free navigation. TAPF is useful for structured environments like warehouse robot navigation or parking slot allocation. In short, ITAPF (Short-Term, Reactive Execution) applications are similar to ride-hailing services like Uber assigning rides dynamically. ITAPP (Strategic Planning) is applied in optimizing warehouse robot workflows. TAPF involves Fixed Goal Assignment which is for assigning parking spots to autonomous robots.

Figure 2.2 illustrates these different problem formulations in comparison to single-robot planning frameworks. When extending these frameworks to multi-robot systems, challenges such as scalability and computational complexity significantly increase due to inter-robot dependencies. The problem addressed in this thesis is most closely related to ITAPP, where tasks must be assigned, sequenced, and executed over extended horizons, while explicitly accounting for motion feasibility and inter-robot coordination.

2.2 Multi-robot motion planning with TAMP/ITMP approach

Task and Motion Planning (TAMP), also referred to as Integrated Task and Motion Planning (ITMP), has emerged as a prominent paradigm for multi-robot systems seeking to jointly address high-level task allocation and low-level trajectory generation within a unified framework [20]. At the task level, planning involves logical sequencing, resource management, and scheduling, often expressed through symbolic representations such as the Planning Domain Definition Language (PDDL), Linear Temporal Logic (LTL), or combinatorial optimization models (e.g., vehicle routing and assignment problems) [23; 24]. In contrast, motion planning enforces kinematic and dynamic feasibility, ensuring collision-free and physically realizable trajectories [9; 25].

The fundamental challenges of integrating task planning and motion planning have been analyzed extensively in the literature [26]. A key difficulty lies in reconciling discrete symbolic task representations with continuous geometric and kinodynamic motion constraints. Additional challenges include the propagation of feasibility information across abstraction levels and the combinatorial growth of the joint search space when symbolic decisions and continuous trajectories are tightly coupled. In multi-robot settings, these issues are further exacerbated by inter-robot dependencies, dynamic environments, and the need to reason jointly about task allocation, colli-

sion avoidance, and coordination under partial information. Work by [27] and [28] established foundational principles for addressing these challenges through hierarchical task and motion planning, where symbolic decisions are made incrementally and refined through continuous motion reasoning, enabling planning "in the now" [27]. Subsequent constraint-based formulations formalize this integration by treating task and motion planning as a constraint satisfaction problem, where symbolic plan skeletons guide geometric feasibility checks and motion refinement [28]. These approaches highlight the inherent trade-offs between completeness, scalability, and reactivity that characterize integrated planning systems.

Recent research has explored complementary strategies to manage these trade-offs in multi-robot contexts. Roadmap-based integration methods introduce high-level abstractions that reduce the dimensionality of the joint planning problem. In particular, [29] propose a multi-robot roadmap representation that simultaneously captures goal assignment and abstract motion, explicitly modeling robot interference through cost terms. By decomposing the resulting abstract solution into partially ordered motion planning problems, such approaches achieve improved scalability while retaining coordination awareness. In contrast, trajectory-optimization-based approaches formulate integrated task allocation and motion planning as a single optimal control problem with mixed discrete and continuous decision variables [30]. These methods provide a principled and expressive modeling framework capable of capturing non-holonomic dynamics, collision avoidance, and coordination constraints. However, the resulting mixed-integer or non-convex optimization problems typically require relaxations or multi-stage solution strategies to remain tractable, limiting scalability to small- or medium-sized robot fleets.

[31] present a fully decentralized approach to multi-robot task distribution in which robots autonomously select tasks using bio-inspired mechanisms such as response threshold models, learning automata, and ant colony optimization. Their work demonstrates that robust global coordination can emerge from local decision-making even under noisy and dynamically changing task estimates, without reliance on a central controller. While their focus is on abstract task selection rather than sensor-level data generation or real-time system integration, the underlying principle of decentralized, noise-tolerant autonomy directly motivates the closed-loop, perception-driven system architecture adopted in this thesis.

Together, these works reveal a fundamental spectrum in integrated task and motion planning: abstraction-based methods favor scalability and structured decomposition, while optimization-based formulations offer higher model fidelity at increased computational cost. This thesis builds on these

insights by adopting a hybrid approach that combines structured task allocation with distributed trajectory optimization, aiming to achieve balanced workload distribution, motion feasibility, and practical scalability in multi-robot systems. For multi-robot applications, TAMP/ITMP integration strategies are commonly categorized into three classes [20]: **Hierarchical approaches**, where task planning precedes motion planning and feasibility is validated subsequently; **Feedback-based approaches**, which iteratively revise task assignments based on motion-level feasibility feedback; and **Tightly coupled (hybrid) approaches**, where task and motion decisions are optimized jointly, often yielding improved solution quality and robustness [32].

While TAMP/ITMP approaches offer improved feasibility, robustness, and overall solution quality, they also introduce significant computational challenges due to the exponential growth of the combined symbolic–geometric search space [33]. Nevertheless, successful deployments in domains such as warehouse automation, surveillance, and agricultural robotics demonstrate the practical potential of integrated planning frameworks [34]. Accordingly, this thesis focuses on the multi-robot motion planning problem within a TAMP/ITMP hybrid framework, with particular emphasis on task allocation and routing under motion feasibility constraints.

2.2.1 MRTA Problem types with TAMP/ITMP approach

TAMP/ITMP is an approach that decomposes the overall problem into two key sub-problems—task allocation and motion planning—and then integrates their solutions to achieve higher-quality outcomes. The first step is to carefully understand and analyze the type of multi-robot task allocation (MRTA) problem relevant to the intended application scenario. General MRTA applications include warehouse automation [35], search and rescue [36], exploration [37], and mapping or surveying [38] and each application has a different MRTA variant. MRTA has numerous variants shaped by different assumptions, constraints, and objectives, all aiming to optimize task distribution based on robot capabilities and task demands. Each problem variant is NP-hard and necessitates distinct solution methods. For instance, MRTA assumptions may revolve around tasks and resource availability; objectives might minimize or maximize criteria such as distance, time, or cost; and the objective function can be linear, non-linear, or multi-objective. Constraints also vary, depending on factors like robot carrying capacity [39], navigational modes such as differential robots or drones [40], or task and scheduling constraints such as precedence [41], depot returns [42], time windows [43], robot utilization [44], and human-robot collaboration [45].

Methods for solving these MRTA variants range from brute-force approaches (often infeasible for large-scale problems) to more sophisticated techniques. Examples include Branch and Bound [46], meta-heuristics such as genetic algorithms [47], ant-colony optimization [48], particle swarm optimization [49], and simulated annealing [50]. Many solutions also leverage commercially available ILP/MILP solvers like CPLEX or Gurobi, as well as behavior-based [51], market-based [23], and negotiation-based [43; 52] frameworks. Furthermore, approximation methods from combinatorics can be employed to handle specific classes of NP-hard MRTA problems more efficiently [53].

A widely studied variant of MRTA is the Multi-Robot Routing Problem (MRR), where robots must visit multiple goal locations. This setup arises in numerous practical contexts, including indoor warehouses [21], automated container terminals [54], search and rescue [36], and surveillance [38]. Although the standard objective in MRR is to minimize the makespan for all robots, such an approach often leads to uneven workload distribution. To mitigate this, a "min–max objective" has been introduced, which focuses on minimizing the maximum makespan incurred by the slowest robot. This strategy balances the overall workload and lowers the likelihood of robot failure due to excessive utilization.

The first part of this thesis investigates MRR with a min–max objective (MRR-MM), highlighting its applicability and the inherent challenges it poses in real-world settings. In addition, it outlines a matrix-based formulation developed for MRR-MM, presents a novel Deterministic Annealing method for solving this problem, and compares its performance with established approaches under application-relevant conditions.

2.2.2 Motion Planners for TAMP/ITMP

The second part of this thesis focuses on the low-level motion planner used in conjunction with the task allocation method. In a TAMP/ITMP setting, such a motion planner should satisfy various requirements. First, it must generate trajectories that comply with kinematic and dynamic constraints and also avoid collisions. Second, it should accommodate diverse robot types (e.g., differential-drive, non-holonomic) while optimizing performance metrics such as path length, smoothness, or energy consumption. Mechanisms for callbacks or incremental updates are necessary to enable rapid re-planning if a task-level allocation changes or if feasibility issues arise in dynamic environments. Additionally, the planner should tackle computational complexity efficiently, so that it can manage concurrent trajectories for multiple robots, coordinate collision avoidance and shared resource utilization, and handle

communication delays and sensor uncertainties.

A wide range of methods are used in practice. **Sampling-based approaches**, for example Rapidly-Exploring Random Trees (RRT) [55] and Probabilistic Roadmaps (PRM) [56], effectively handle high-dimensional state spaces and dynamic constraints but may yield suboptimal trajectories needing post-processing. **Search-based methods**, including A* and D* [57], explore discretized spaces systematically and offer theoretical guarantees on optimality. However, they can struggle with continuous states or kinematic constraints inherent to wheeled systems. **Optimization-based planners** such as CHOMP [9], STOMP [58], and TrajOpt [25] integrate factors like smoothness, obstacle avoidance, and dynamics into a single optimization problem, producing collision-free solutions while risking local minima. Finally, Learning-based motion planning approaches, particularly those utilizing reinforcement learning (RL), imitation learning (IL), and deep learning, have shown significant promise in enabling robots to navigate complex and dynamic environments. RL allows robots to learn optimal policies through trial and error, adapting to various scenarios without explicit programming [59]. IL, on the other hand, enables robots to acquire skills by mimicking expert demonstrations, reducing the need for extensive exploration [60]. Deep learning techniques, especially deep reinforcement learning, facilitate the processing of high-dimensional sensory inputs, improving decision-making in real-time applications [61]. Despite their potential, these methods face challenges such as data inefficiency, generalization to unseen environments, and real-time computational demands [62].

In multi-robot contexts, motion planners can be centralized, decentralized, or distributed. For TAMP/ITMP, distributed planners are often favored because each robot computes its trajectory locally and communicates only minimal information to resolve conflicts, offering greater scalability and robustness [24]. Within distributed planning, approaches include distributed potential fields [63], graph-based planners (e.g., distributed A* [57]), distributed optimization (e.g., TrajOpt adaptations) [25], and auction-based coordination [64]. While these methods excel in real-time responsiveness and fault-tolerance, they may risk converging to locally optimal solutions due to incomplete global information and can incur added communication overhead [34]. Nonetheless, their capacity to handle rapidly evolving environments makes them popular for swarm robotics, autonomous logistics, and formation control tasks [33].

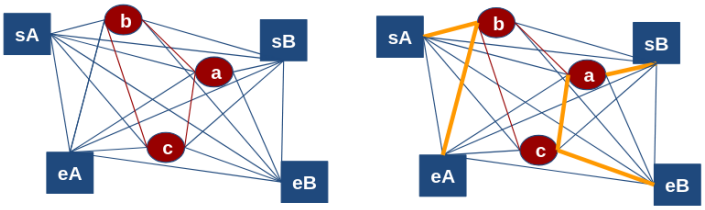


Figure 2.3: Illustration of the MRR-MM problem. Robots A and B start at positions s_A, s_B and end at e_A, e_B, respectively. Tasks $\{a, b, c\}$ are assigned and sequenced to minimize the maximum cost incurred by any robot.

2.3 Multi-Robot Routing Problem with a Min-Max Objective

The Multi-Robot Routing Problem with a Min–Max objective (MRR-MM) concerns the assignment and sequencing of tasks for a set of robots such that the maximum cost incurred by any individual robot is minimized. This objective explicitly targets workload balancing and robustness, making the problem particularly relevant for real-world multi-robot applications where excessive utilization of a single robot may lead to failures or degraded system performance. Despite its practical importance, MRR-MM has been studied independently by different research communities, often under varying assumptions and terminologies. As a result, a wide range of formulations and solution strategies exist in the literature.

The MRR-MM problem appears under different names across disciplines. In robotics, it is related to the *Alliance Efficiency Problem (AEP)* [65] and the *Multi-Robot Routing Problem* [23]. In operations research, closely related formulations include the *Minimum Maximum Multiple Depot Vehicle Routing Problem (MMMDVRP)* [7] and the *Asymmetric Bottleneck Multiple Traveling Salesman Problem (ABmTSP)* [66]. To avoid ambiguity and promote clarity, this thesis adopts the term **Multi-Robot Routing Problem with a Min–Max Objective (MRR-MM)**.

From an **operations research** perspective, MRR-MM is closely related to variants of the *Multiple Depot Multiple Traveling Salesman Problem*, *Multi-Vehicle Routing Problem*, and *Min–Max Vehicle Routing Problem*. These formulations often incorporate specialized constraints such as asymmetric travel costs, open depots, or capacity limits [67–70]. Related problems include the *Orienteering Problem*, the *Nurse Location Problem*, and bottleneck variants of the *Asymmetric Traveling Salesman Problem* [71; 72].

The **mathematics** community has focused on approximation algorithms for NP-hard routing and covering problems related to MRR-MM, such as the *N-Path Cover Problem*, *Set Cover*, and the *Dial-a-Ride Problem* [73; 74]. Notably, Arkin *et al.* [75] proposed an approximation algorithm with a provable constant-factor bound, demonstrating scalability to large problem instances.

In **robotics and artificial intelligence**, MRR-MM appears in variants such as *Multi-Robot Task Assignment*, *Target Assignment and Path Finding (TAPF)*, and *Multi-Vehicle Multi-Goal Planning* [21; 76; 77]. These formulations typically incorporate additional physical constraints, including Dubins curvature constraints, neighborhood visitation, or non-holonomic vehicle models. Auction-based approaches have also been explored, offering improved computational efficiency at the expense of optimality guarantees [23].

2.3.1 Computational complexity and solution approaches

The MRR-MM problem is NP-hard. Parker [78] established this result via a reduction from the Partition problem. Consequently, computing globally optimal solutions becomes intractable for large-scale instances, necessitating alternative solution strategies.

Exact methods. Exact approaches such as Integer Linear Programming (ILP) and Mixed-Integer Linear Programming (MILP) formulations can be solved using commercial solvers (e.g., CPLEX, Gurobi). These methods rely on Branch-and-Bound or Branch-and-Cut techniques to guarantee optimality [79]. While effective for small problem sizes, their computational complexity grows rapidly with the number of robots and tasks.

Approximation algorithms. Certain variants of MRR-MM admit approximation schemes that provide bounded sub-optimality guarantees. Polynomial-Time Approximation Schemes (PTAS) and greedy approximation algorithms have been studied for related routing and covering problems [80; 81]. Arkin *et al.* [75] proposed a practical approximation algorithm with provable bounds, and Turpin *et al.* [7] modeled MRR-MM as a graph problem, yielding near-optimal Hamiltonian path solutions.

Heuristics and metaheuristics. Heuristic and metaheuristic approaches trade optimality for scalability. Local search methods such as Tabu Search, Hill Climbing, and Simulated Annealing [82; 83] iteratively improve candidate solutions. Evolutionary methods, including Genetic Algorithms and

Genetic Programming [84; 85], explore solution populations, while swarm intelligence techniques such as Ant Colony Optimization and Particle Swarm Optimization leverage decentralized agent interactions [86; 87]. Hybrid and hyper-heuristic frameworks combine multiple strategies to exploit complementary strengths [88].

Decentralized and behavior-based methods. Behavior-based and auction-based approaches emphasize robustness and decentralization. Parker's L-ALLIANCE framework prioritizes fault tolerance and adaptability but does not provide optimality guarantees [65]. Auction-based methods enable efficient task allocation through decentralized bidding but typically rely on restrictive assumptions such as symmetric costs or limited task coupling [23].

Overall, existing solution approaches for MRR-MM highlight a fundamental trade-off between computational efficiency and solution optimality. This thesis addresses this challenge by developing a structured formulation and a deterministic annealing-based optimization method that targets balanced workload distribution while maintaining practical scalability.

2.4 Neural Network Based Optimization Methods

Neural networks (NNs) have been widely explored as tools for solving combinatorial optimization problems, particularly those classified as NP-hard, such as the Traveling Salesman Problem (TSP), Vehicle Routing Problem (VRP), scheduling problems, and multi-robot routing. Within this context, neural approaches are typically viewed as heuristic or metaheuristic methods: they do not provide optimal solutions, but often exhibit strong empirical performance and favorable scalability properties.

Early neural approaches to combinatorial optimization formulated problems as energy minimization tasks. A seminal example is the **Hopfield–Tank network** [89], which encodes problem constraints and objectives directly into an energy function. Although such networks can converge rapidly, they are prone to becoming trapped in local minima and are sensitive to parameter tuning. More recent learning-based neural architectures, including sequence-to-sequence models and reinforcement learning policies, learn approximate solvers from data [90]. While these methods can generalize across problem instances and offer fast inference once trained, they typically require substantial training data and do not provide explicit guarantees on solution quality.

An alternative class of neural optimization methods is based on **energy-based and mean-field formulations**, which do not rely on data-driven

learning. Boltzmann machines and related stochastic neural models introduce controlled randomness to escape local minima, closely resembling simulated annealing [91]. Hybrid neural systems further combine neural components with classical optimization or heuristic methods, for example by generating initial solutions or adapting algorithmic parameters during search.

This class of energy-based neural optimization methods is particularly relevant to multi-robot routing problems with a min–max objective, where the solution space is highly combinatorial and structured constraints must be enforced efficiently.

2.4.1 Deterministic Annealing

Hopfield neural networks (HNNs) are recurrent, energy-based models introduced by Hopfield [92] and widely applied to combinatorial optimization problems such as the TSP. If the objective and constraints of a problem can be encoded into an energy function $E(\mathbf{s})$, minimizing this energy corresponds to solving the original optimization problem. A general Hopfield energy function is given by

$$E(\mathbf{s}) = -\sum_{i,j} w_{ij} s_i s_j, \tag{2.1}$$

where w_{ij} denotes synaptic weights and $\mathbf{s}$ is a vector of binary neuron states.

Despite their conceptual simplicity, HNNs suffer from significant limitations, including frequent convergence to poor local minima and poor scalability. To mitigate these issues, stochastic techniques such as *simulated annealing* introduce a temperature parameter kT that controls exploration and enables escape from local minima, ideally converging toward high-quality solutions as $kT \to 0$.

Hopfield and Tank [89] extended binary HNNs by introducing continuous-valued neurons $v_i \in [0,1]$, representing activation probabilities. This formulation, commonly referred to as *Deterministic Annealing (DA)* or *Mean Field Annealing (MFA)*, replaces stochastic sampling with deterministic mean-field dynamics, yielding smoother convergence behavior. Peterson and Söderberg [93] further introduced Z-state *Potts neurons*, which naturally enforce "one-out-of-Z" constraints and improve convergence stability for assignment-type problems.

When combinatorial optimization problems are modeled using Potts neurons, structural constraints are embedded directly into the energy formulation. Through deterministic annealing, continuous-valued solutions are gradually driven toward discrete configurations, often yielding high-quality or near-optimal solutions for large, structured NP-hard problems. This the-

sis leverages deterministic annealing with Potts neurons as an efficient and scalable task allocation method for solving the MRR-MM problem.

2.5 Gradient-Based Trajectory Optimization Methods

Gradient-based trajectory optimization methods are fundamental in the field of motion planning for robotics, providing an efficient way to compute optimal paths for robotic systems. These methods rely on minimizing a cost function using gradient information, where the trajectory is iteratively refined to meet specific objectives while adhering to system constraints. Gradient-based approaches are particularly useful in high-dimensional and dynamic environments, where optimal control and real-time adjustments are crucial for successful robot operation. These methods are widely employed across applications, from autonomous vehicles to robotic manipulators, and can handle complex objectives like energy efficiency, obstacle avoidance, and smoothness. In the context of integrated task and motion planning (ITMP) and multi-robot systems, gradient-based trajectory optimization is particularly attractive due to its ability to refine locally feasible trajectories while preserving high-level task allocations.

CHOMP (Covariant Hamiltonian Optimization for Motion Planning) is an early method that minimizes energy-based cost functions to generate smooth, feasible trajectories, handling both obstacle avoidance and dynamic constraints [9]. STOMP (Stochastic Trajectory Optimization for Motion Planning) builds on CHOMP by introducing stochastic noise into the optimization process, allowing it to escape local minima and explore a wider range of solutions, particularly in high-dimensional spaces [58]. Sequential Quadratic Programming (SQP) is a widely-used technique that solves nonlinear trajectory optimization problems by iterating through quadratic approximations, making it effective for constrained, nonlinear systems [94]. Direct Collocation Methods discretize the trajectory and optimize the state and control inputs directly, offering a straightforward way to handle optimal control problems [95]. Differential Dynamic Programming (DDP) optimizes trajectories for systems with nonlinear dynamics by linearizing the system and cost function iteratively, improving the solution at each step [96]. Model Predictive Control (MPC) is used for real-time trajectory optimization, solving a finite-horizon optimization problem at each time step, making it highly suitable for dynamic environments where the robot must react quickly to changes [97]. Finally, Deep Reinforcement Learning (DRL), when combined with gradient-based optimization, allows robots to learn optimal policies for motion planning in complex environments through interaction, using neural networks to approximate policies and cost functions

[98]. While deep reinforcement learning has shown promise in learning motion policies for complex environments, its reliance on large training datasets and limited guarantees on constraint satisfaction often restrict its use in safety-critical or tightly constrained robotic systems. Each of these methods offers a unique advantage depending on the task and environment, from smooth trajectory generation to handling high-dimensional or uncertain systems, contributing significantly to the field of autonomous robot motion planning. These methods can be broadly categorized into classical optimization-based approaches (e.g., CHOMP, STOMP, SQP, DDP, MPC) and learning-augmented approaches, where neural networks are used to approximate policies or cost functions.

2.5.1 CHOMP Algorithm

CHOMP's advantages include rapid generation of smooth, dynamically compliant trajectories, facilitating real-time execution in many robotic systems. Rolling constraints for non-holonomic vehicles can be introduced via Lagrange multipliers or penalty terms in the objective. Nevertheless, like any gradient-based method, CHOMP is prone to local minima and is sensitive to parameter tuning. Particularly cluttered or dynamic environments may require frequent replanning or integration with a global planner. Complexity also increases for higher-dimensional constraints (e.g., articulated vehicles). Nonetheless, CHOMP remains efficient for generating feasible, high-quality paths, making it a valuable tool for wheeled mobile robots. As long as its sensitivity to local minima and parameter dependence is managed—potentially by leveraging complementary global planning or advanced optimization strategies—it provides a strong foundation for low-level motion planning. These properties make CHOMP particularly suitable as a local refinement layer in multi-robot systems, where global coordination decisions can be preserved while resolving local conflicts through distributed optimization.

In summary, this method and its variants offer an effective solution for wheeled mobile robot motion planning. Incorporating non-holonomic constraints or distributed coordination features can further broaden their applicability, enabling smooth and collision-free paths in a wide range of scenarios.

2.6 Synopses and positioning of this work

This thesis addresses the problem of integrating high-level task allocation with low-level motion planning for multi-robot systems, with a particular focus on wheeled mobile robots operating in structured, dynamic environ-

ments. The work is situated within the TAMP/ITMP paradigm, in which task sequencing decisions (e.g., assignment and ordering of tasks across robots) are tightly coupled with motion feasibility and trajectory execution.

Motivated by the known computational complexity of multi-robot coordination problems and the practical constraints encountered in real-world logistics scenarios, the thesis builds upon and connects multiple research threads. At the task-allocation level, it adopts a min–max formulation of the Multi-Robot Routing Problem (MRR-MM), explicitly targeting balanced workload distribution across robot fleets. This choice distinguishes the work from traditional min-sum formulations and aligns with practical requirements such as fairness, robustness, and avoidance of robot over-utilization.

To address the NP-hard nature of the resulting allocation problem, the thesis investigates deterministic annealing with Potts neurons as a scalable and systematic optimization method. This approach occupies a middle ground between exact optimization techniques, which are often intractable at scale, and purely heuristic methods, which may lack solution quality guarantees. The proposed formulation and solution strategy enable efficient generation of near-optimal task allocations for small- to medium-scale robot fleets.

At the motion-planning level, the work leverages gradient-based trajectory optimization methods, in particular CHOMP and its extensions, to ensure that allocated tasks remain executable under non-holonomic constraints, collision avoidance requirements, and dynamic environmental changes. Rather than proposing a new motion planner, the thesis focuses on adapting and integrating existing optimization-based planners within a distributed execution layer, allowing local trajectory refinement without invalidating global task allocations.

Overall, this thesis positions itself at the intersection of combinatorial task allocation and continuous trajectory optimization. Its primary contribution lies in the structured integration of these components into a hybrid planning framework that balances solution quality, computational efficiency, and practical feasibility. By combining a min–max routing formulation, deterministic annealing-based allocation, and distributed gradient-based motion refinement, the work advances the state of the art in scalable and deployable multi-robot task and motion planning for logistics and warehouse automation scenarios.

3. Summary of Appended Papers

This chapter summarizes the main contributions of the appended papers and explains how they collectively realize the proposed hybrid task and motion planning framework. The individual papers are organized around a common goal: to integrate a centralized task allocation mechanism with a distributed motion planning layer in a computationally efficient and practically deployable manner.

Paper 1 formulates the Multi-Robot Routing Problem with a Min–Max objective (MRR-MM) using a permutation-based representation that jointly captures task assignment and execution order. Building on this formulation, Paper 2 introduces a Deterministic Annealing (DA) solver based on mean-field theory and Potts neurons, enabling scalable and near-optimal centralized task allocation. Papers 3 and 4 address the motion planning layer and propose a distributed, gradient-based trajectory optimization approach capable of resolving inter-robot path conflicts for non-holonomic mobile robots. Finally, Papers 5 and 6 integrate the task allocation and motion planning components into the SPADES and Extended-SPADES frameworks, resulting in a complete TAMP/ITMP solution. This chapter focuses on summarizing the assumptions, formulations, methods, and key findings of these papers, while detailed derivations and experimental analyses are provided in the appended publications.

3.1 Research Questions addressed

Table 3.1 maps the research questions to the corresponding thesis papers.

- **RQ2: Task allocation problem and formulation** is addressed by Paper 1, which explores the MRR-MM problem formulation in detail. This paper demonstrates that a permutation-matrix representation naturally captures balanced task allocation.
- **RQ3: Methods for the MRR-MM problem** is addressed by Paper 2. It shows that the deterministic annealing (DA) method can efficiently

Table 3.1: Mapping of research questions to corresponding thesis papers

Research Question	Papers
Hybrid task and motion planning framework	Papers 5, 6
Task allocation problem and formulation	Paper 1
Methods for MRR-MM problem	Paper 2
Gradient-based trajectory planners	Papers 3, 4, 6

search for optimal permutation matrix solutions and achieve near-optimal allocations in NP-hard multi-robot routing problems.

- **RQ4: Gradient-based trajectory planners** is addressed by Papers 3 and 4. These papers extend CHOMP from robotic manipulators to non-holonomic mobile robots, incorporating kinematic constraints and collision avoidance to enable distributed trajectory planning.

- **RQ1: Hybrid task and motion planning framework** is explored in Papers 5 and 6. Paper 5 combines optimal task assignment (Hungarian algorithm) with lightweight distributed motion refinement to achieve competitive performance in dynamic environments. Paper 6 further demonstrates that task grouping and sequencing, together with distributed motion planning, improve scalability and real-world applicability.

3.2 Assumptions

All appended papers assume environments inspired by automated warehouse and logistics scenarios, where a fleet of mobile robots must execute a large number of spatially distributed tasks. In such settings, the number of tasks N typically exceeds the number of robots M, with $M \leq N$, and the experimental configurations reflect representative warehouse-scale deployments reported in the literature [22].

Robots operate in a multi-depot, open-tour setting: each robot starts from a possibly distinct initial location and is not required to return to its starting point after completing its assigned tasks. All robots are assumed to be functionally identical and capable of executing any task.

The environment is assumed to be known and static at the task allocation level. Communication constraints, sensing uncertainty, and map inaccuracies are neglected, allowing the centralized task planner access to complete global information. Dynamic effects such as collision avoidance, acceleration, and deceleration are addressed at the motion planning level.

Task allocation is based on a graph-defined cost structure consisting of: (i) asymmetric transition costs between tasks and robot start/end locations, and (ii) homogeneous task execution costs, normalized to a constant value. Temporal scheduling is not modeled explicitly; instead, sequencing and timing effects are implicitly encoded in the transition costs.

The output of the task planner is a task assignment and execution order for each robot, which is subsequently refined and executed by the distributed motion planner. In this thesis, the terms robot and vehicle are used interchangeably to denote autonomous mobile robots.

3.3 MRR-MM Problem

Paper 1 studies the *Multi-Robot Routing Problem with a Min–Max objective* (MRR-MM), where the goal is to assign and sequence tasks for a fleet of robots such that the *maximum* route cost over all robots is minimized. In contrast to classical routing formulations that minimize the sum of costs, the min–max criterion explicitly enforces balanced workload distribution, which is essential in warehouse logistics, surveillance, and search-and-rescue applications. The problem is combinatorial and NP-hard, since task assignment and task ordering must be determined jointly. Within standard MRTA taxonomies, MRR-MM corresponds to Single-Task Robots / Single-Robot Tasks with time-extended assignments [11], exhibits in-schedule dependencies in the iTax taxonomy [12], and can be interpreted as a routing problem with soft temporal constraints [99].

3.3.1 Permutation Matrix Formulation

Paper 1 introduces a compact *successor (permutation) matrix* representation that encodes both task assignment and execution order. Inspired by the Hungarian algorithm for linear assignment [10; 100], this formulation generalizes one-to-one matching to the one-to-many, sequential setting required by MRR-MM.

Given M robots and N tasks, a binary successor matrix $\mathbf{S} \in \{0,1\}$, of size $(2M+N) \times (2M+N)$ is defined, where each entry s_{ij} indicates whether node j immediately follows node i (see Chapter 2 for the formal definition). Robot start and end locations are modeled explicitly as additional nodes, allowing each robot route to be represented as a directed path from its start node to its end node. Validity constraints enforce that every task has exactly one predecessor and one successor, while self-loops and invalid transitions are eliminated by fixing corresponding entries of $\mathbf{S}$ to zero. Each valid solution corresponds to a permutation matrix consistent with these

Table 3.2: Solution space (successor permutations): totals and valid (loop-free)

M	N	**Effective Matrix Size**	**# Possible**	**# Valid**
1	2	3×3	4	2
1	3	4×4	18	6
2	2	4×4	4	4
2	3	5×5	36	24
2	4	6×6	288	144
2	5	7×7	2400	960
3	3	6×6	36	36
3	4	7×7	576	432
4	5	9×9	14,400	11,520
5	6	11×11	518,400	432,000

constraints, yielding M disjoint, ordered task sequences that jointly cover all N tasks exactly once.

3.3.2 Solution Space and Complexity

The number of feasible solutions grows factorially with M and N. Assuming that each robot executes at least one task, the number of valid, loop-free successor permutations is

$$\text{valid}(M,N) = N!\binom{N-1}{M-1}M!,$$

while the total number of successor permutations consistent with robot start and end nodes (including invalid loop solutions) is

$$\text{possible}(M,N) = \binom{N}{M}M!\,N!.$$

These expressions, summarized in Table 3.2, illustrate the rapid growth of the solution space and motivate the use of approximate and annealing-based methods rather than exhaustive search.

3.3.3 Methods and Evaluation

Paper 1 evaluates three solution strategies: a fast heuristic-based graph search, Simulated Annealing (SA), and Deterministic Annealing (DA). The

heuristic method offers low computational cost but reduced solution quality, while SA is capable of recovering near-optimal solutions at the expense of long runtimes and high variance. DA emerges as a favorable compromise, producing high-quality, well-balanced solutions with significantly lower runtime variability than SA. Simulation results in warehouse-like environments demonstrate that minimizing expected or greedy costs does not necessarily minimize makespan, and that explicit min–max optimization leads to superior workload balancing across robots.

3.4 Deterministic Annealing Based on Potts Neurons

Paper 2 develops a *Deterministic Annealing* (DA) solver for the MRR-MM problem, building directly on the permutation matrix formulation introduced in Paper 1. The key idea is to replace the discrete successor matrix $\mathbf{S}$ with a continuous *Potts assignment matrix* $\mathbf{V}$ of size $(2M+N)\times(2M+N)$, whose entries $v_{ij} \in [0,1]$ represent soft assignments between successor pairs. Each Potts neuron v_{ij} encodes the probability that node j follows node i. Row and column normalization enforce one-of-Z constraints, ensuring that $\mathbf{V}$ converges toward a valid permutation matrix as annealing progresses. Robot start and end nodes are incorporated explicitly, allowing complete task chains to be formed for each robot.

3.4.1 DA algorithm design

The DA method can be broken down into five steps and described in algorithm 1.

1. The Potts matrix $\mathbf{V}$ consists of continuous units $v_{ij} \in [0,1]$ representing thermal average values during annealing, having the same dimension as the $\mathbf{S}$ matrix. The effective sub-matrix of $\mathbf{V}$ remains doubly stochastic throughout the annealing.

2. Defining design variables to detect loop matrices via the $\mathbf{V}$ matrix and to identify the permutation matrix with minimum cost: There are two essential design variables to facilitate loop detection and cost minimization - the propagator matrix ($\mathbf{P}$) and transport cost calculation.

    - The propagator matrix ($\mathbf{P}$), defined as

    $$\mathbf{P} = (\mathbf{I} - \mathbf{V})^{-1} = \mathbf{I} + \mathbf{V} + \mathbf{V}^2 + \mathbf{V}^3 + \ldots, \tag{3.1}$$

    is used to detect loops. Element p_{ij} indicates the total number of possible paths from task i to task j. If $\mathbf{V}$ contains loops, at

Algorithm 1: Deterministic Annealing Algorithm

Data: $M, N, \mathbf{T}, \mathbf{\Delta}$

Result: Permutation matrix $\mathbf{V}$ with no loops and an optimal transport cost

Set kT_{start}, kT_{stop}, $kT_{\text{fac}}^{\min}$, $kT_{\text{fac}}^{\max} = 0.99$, η, and γ ;

Initialize $\mathbf{V}$ matrix ;

Calculate: $\mathbf{P} = (\mathbf{I} - \mathbf{V})^{-1}$;

L_i and R_i ;

$U_{ij}^{\text{loop}} = \frac{p_{ji}}{p_{ii} - p_{ji}}$;

$U_{ij}^{\text{trans}} = \frac{1}{2}\left[p_{j\alpha}(L_i + \Delta_{ij}) + p_{\beta i}(R_j + \Delta_{ij})\right]$;

$U_{ij}^{\text{stable}} = \eta v_{ij}$;

Update: $U_{ij}^{\text{local}} = U_{ij}^{\text{trans}} + \gamma U_{ij}^{\text{loop}} - \eta U_{ij}^{\text{stable}}$;

Set $kT = kT_{\text{start}}$;

while $kT > kT_{\text{stop}}$ **do**

  Update $v_{ij} = \frac{e^{-U_{ij}^{\text{local}}/kT}}{\sum_m e^{-U_{mj}^{\text{local}}/kT}}$ in $\mathbf{V}$ matrix ;

  while $\sum_i v_{ij} < 0.95$ *for some* j, *or* $\sum_j v_{ij} < 0.95$ *for some* i **do**

    Sinkhorn-Knopp normalization ;

  end

  Repeat steps **7** and **8** ;

  Calculate $dV_{\max}$;

  Set the adaptive annealing factor:

$$kT_{\text{factor}} = kT_{\text{fac}}^{\min} + (kT_{\text{fac}}^{\max} - kT_{\text{fac}}^{\min}) \cdot \tanh(dV_{\max})$$

  Update the temperature: $kT = kT \cdot kT_{\text{factor}}$;

end

Output $\mathbf{V}$ matrix ;

least one element of $\mathbf{P}$ approaches infinity, allowing detection and penalization of loops.

- The solution cost is calculated using vectors $\mathbf{L}$ and $\mathbf{R}$. Element L_i denotes the cost to reach task i from any start position, while R_i denotes the cost to finish from task i to any end position. The transport cost (make-span) is:

$$C = \frac{1}{2}(L_\alpha + R_\beta), \tag{3.2}$$

where indices α, β correspond to the indices for the maximum values in the relevant subsets of $\mathbf{L}$ and $\mathbf{R}$.

3. The local field energy equation is derived based on the design variables: For each Potts neuron v_{ij}, the local field energy U_{ij}^{local} combines three components:

- **Transport cost field** (U_{ij}^{trans}), minimizing maximum transport costs:

$$U_{ij}^{\text{trans}} = \frac{1}{2}[p_{j\alpha}(L_i + \Delta_{ij}) + p_{\beta i}(R_j + \Delta_{ij})]. \tag{3.3}$$

- **Loop cost field** (U_{ij}^{loop}), penalizing loops (adapted from [101]):

$$U_{ij}^{\text{loop}} = \frac{p_{ji}}{p_{ii} - p_{ji}}. \tag{3.4}$$

- **Stabilization field** (U_{ij}^{stable}), avoiding oscillations:

$$U_{ij}^{\text{stable}} = -v_{ij}. \tag{3.5}$$

The total local field thus becomes:

$$U_{ij}^{\text{local}} = U_{ij}^{\text{trans}} + \gamma U_{ij}^{\text{loop}} + \eta U_{ij}^{\text{stable}}, \tag{3.6}$$

with weighting factors $\gamma, \eta > 0$.

4. Incorporating this local field energy into the Potts matrix update equation: The Potts matrix $\mathbf{V}$ is iteratively updated based on the calculated local field energy:

$$v_{ij} = \frac{e^{-U_{ij}^{\text{local}}/kT}}{\sum_m e^{-U_{mj}^{\text{local}}/kT}}. \tag{3.7}$$

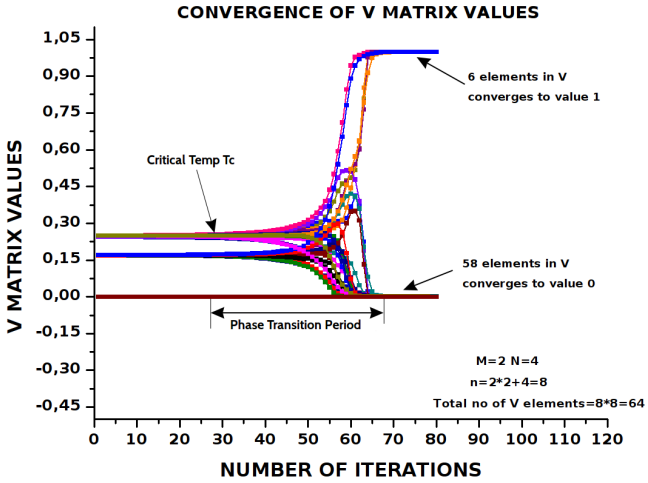


Figure 3.1: Convergence of solutions in the DA method for $M = 2$ and $N = 4$. The initial values of the normalized $\mathbf{V}$ matrix (total of 64 elements) is set to values close to 0.166 and 0.25. The unwanted/dummy variables are set to zero. The system goes through a *phase transition* at the critical temperature T_c when the values of the Potts transfer assignment matrix change rapidly to converge to a solution with the value of 6 elements going to 1 and the remaining 58 elements converging to zero.

5. Designing an annealing schedule to progressively minimize this energy function, converging to the global minimum. At some point the system reaches a critical temperature kT_c, where a *phase transition* takes place. Here the system undergoes rapid changes and converges towards a solution, which is illustrated in figure 3.1. It is important to set the initial temperature $kT_{\text{start}} > kT_c$, but not by much since that leads to unnecessary long convergence time.

6. Since this matrix must remain doubly stochastic, additional normalization using Sinkhorn-Knopp normalization [102] is performed iteratively until convergence. This completes the Deterministic Annealing approach, enabling efficient convergence to an optimal solution without loops and minimum total cost.

3.4.2 Experimental Results

Experimental evaluation demonstrates that DA consistently finds solutions among the lowest-cost regions of the solution space while avoiding the large runtime variability observed in SA. For moderate and large problem sizes,

DA outperforms SA in terms of computation time while achieving comparable solution quality. Scalability studies further show that DA remains effective as the number of tasks per robot increases, making it well suited for warehouse-scale MRR-MM instances. Overall, Paper 2 establishes Deterministic Annealing with Potts neurons as an efficient and robust centralized task allocation method that integrates naturally with the hybrid TAMP/ITMP framework proposed in this thesis.

3.5 CHOMP Extension for Non-Holonomic Robots

This section summarizes the contributions of Papers 3 and 4, which extend the Covariant Hamiltonian Optimization for Motion Planning (CHOMP) [9] framework to non-holonomic wheeled robots, with a particular focus on autonomous trucks and differential-drive platforms. The primary objective of this work is to adapt CHOMP—originally developed for articulated manipulators—to motion planning problems involving rolling constraints, forward-only motion, and realistic vehicle kinematics.

While CHOMP has proven highly effective for generating smooth, collision-free trajectories for robotic arms, its direct application to wheeled vehicles is non-trivial. Non-holonomic robots cannot move laterally, and their motion must respect curvature and directionality constraints. Papers 3 and 4 investigate multiple strategies to address these challenges, culminating in a robust constraint-based formulation suitable for real-world deployment.

3.5.1 CHOMP Formulation for Wheeled Robots

CHOMP optimizes a trajectory $\xi(t) : [0,1] \to \mathbb{R}^d$ by minimizing a composite cost functional

$$U[\xi] = F_{\text{obs}}[\xi] + \lambda F_{\text{smooth}}[\xi], \tag{3.8}$$

where F_{obs} penalizes proximity to obstacles and F_{smooth} encourages smooth motion by minimizing acceleration.

The smoothness term is defined as

$$F_{\text{smooth}}[\xi] = \frac{1}{2(n+1)} \sum_{t=0}^{n} \left\| \frac{q_{t+1} - q_t}{\Delta t} \right\|^2, \tag{3.9}$$

with gradient

$$\nabla F_{\text{smooth}}[\xi](t) = -\frac{d^2\xi(t)}{dt^2}. \tag{3.10}$$

Obstacle avoidance is handled via a cost field $c(\cdot)$ defined over the workspace, yielding the obstacle cost gradient

$$\nabla F_{\text{obs}}[\xi] = \sum_{t=0}^{n} J^\top \|v_t\| \left[(I - \hat{v}_t \hat{v}_t^\top) \nabla c - c\, \kappa_t \right], \tag{3.11}$$

where $\hat{v}_t$ is the normalized velocity vector and κ_t denotes the trajectory curvature.

3.5.2 Curvature-Based Extensions and Limitations

Paper 3 initially explored extending CHOMP by explicitly penalizing trajectory curvature to enforce vehicle-like motion. Two formulations were evaluated: (i) a quadratic curvature penalty and (ii) a penalty on the time derivative of curvature. Although theoretically appealing, these approaches proved ineffective in practice. Discrete curvature estimates were numerically unstable, and the resulting trajectories failed to reliably respect non-holonomic constraints. These results highlighted the limitations of curvature-based penalties for discrete-time trajectory optimization.

3.5.3 Constraint-Based CHOMP for Non-Holonomic Motion

To overcome these limitations, Papers 3 and 4 adopt a constraint-based formulation of CHOMP using Lagrange multipliers. Non-holonomic constraints are encoded as equality constraints and incorporated directly into the optimization update:

$$\begin{aligned} \xi_{i+1} = \xi_i &- \frac{1}{\eta_i} \mathbf{A}^{-1} \nabla U[\xi_i] \\ &+ \frac{1}{\eta_i} \mathbf{A}^{-1} C^\top (C\mathbf{A}^{-1}C^\top)^{-1} C\mathbf{A}^{-1} \nabla U[\xi_i] - \mathbf{A}^{-1} C^\top (C\mathbf{A}^{-1}C^\top)^{-1} b, \end{aligned} \tag{3.12}$$

where $C = \partial \mathcal{H}(\xi_i)/\partial \xi$ is the Jacobian of the constraint functional and $b = \mathcal{H}(\xi_i)$ is the constraint functional.

Two constraints are central to the proposed method:

Rolling Constraint The rolling constraint enforces non-holonomic motion by prohibiting lateral slip:

$$x' \sin\theta - y' \cos\theta = 0. \tag{3.13}$$

Its discrete form is evaluated along the trajectory and aggregated into a constraint functional $\mathcal{H}_{\text{roll}}(\xi)$.

Forward Motion Constraint To prevent backward motion, a forward-only constraint enforces non-negative velocity in the vehicle's local frame:

$$\|x'_R\| - x'_R = 0. \tag{3.14}$$

The combined constraint functional is

$$\mathcal{H}(\xi) = \mathcal{H}_{\text{roll}}(\xi) + \mathcal{H}_{\text{fwd}}(\xi) = 0. \tag{3.15}$$

This formulation enables CHOMP to generate trajectories that are smooth, collision-free, and kinematically feasible for wheeled robots.

3.5.4 Experimental Validation

The proposed constrained CHOMP framework was validated through extensive simulation and real-world experiments. Simulations were conducted using ROS and Gazebo for both differential-drive robots and a high-fidelity model of an autonomous Volvo FH-16 truck. Real-world experiments were performed at the Stora Holm test facility in Gothenburg, Sweden, using a full-scale autonomous truck equipped with GPS, cameras, lidar, radar, and onboard computing systems. The screenshot of the results of constrained CHOMP tested for non-holonomic truck aare shown in figures 3.2 and 3.3 using Gazebo simulator.

Table 3.3 summarizes comparative results between standard CHOMP and constrained CHOMP across multiple platforms. The constraint-based approach significantly improved success rates while maintaining comparable computational performance.

	CHOMP			**Constrained CHOMP**		
	Turtlebot	Sim AT	Sim AT + Trailer	Turtlebot	Sim AT	Sim AT + Trailer
Successful maneuvers	8/10	3/10	1/10	10/10	9/10	7/10
Avg planning time (s)	0.8	1.1	1.2	1.08	0.9	0.97
Avg iteration time (ms)	41	34	25	32	29	38

Table 3.3: Simulation results comparing standard and constrained CHOMP.

3.5.5 Key Findings

The results of Papers 3 and 4 demonstrate that there is a need for explicit rolling and forward-motion constraints essential for adapting CHOMP to non-holonomic robots. Also, curvature-based penalties alone are insufficient for generating realistic wheeled-robot trajectories. Hence, a constrained based CHOMP is required that produces smooth, collision-free, and

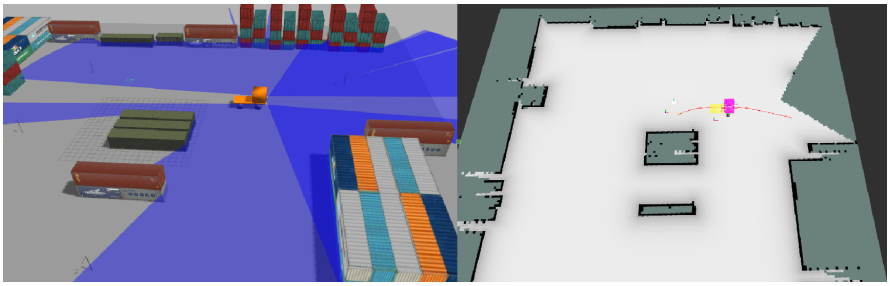
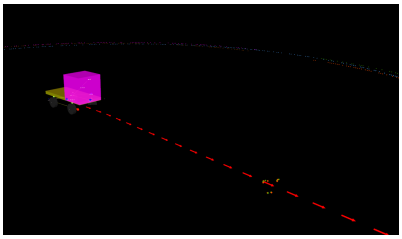
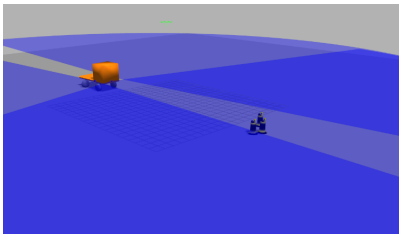


Figure 3.2: Simulation screenshot of the Truck in Gazebo and Rviz



(a) Rviz view of the truck model with obstacles



(b) Gazebo view of the truck model with obstacles

Figure 3.3: Gazebo and RViz view of the constrained straight line trajectory for an initial guess. This trajectory adapts itself as the truck moves closer to the obstacles.

feasible trajectories in real time, with average planning times below one second. The approach scales from small differential-drive robots to full-scale autonomous trucks operating in complex logistics environments. These findings establish constrained CHOMP as a practical and robust local trajectory optimizer for non-holonomic robots, forming a critical component of the hybrid TAMP/ITMP frameworks presented in the subsequent sections.

3.6 Simultaneous Path–Planning and Task–Allocation in Dynamic Environments

This section summarizes the final contribution of the thesis, presented in Papers 5 and 6, which together realize a hybrid TAMP/ITMP framework for multi-robot systems operating in dynamic environments. The proposed frameworks, *SPADES* and *Extended-SPADES*, integrate centralized task allocation with distributed trajectory optimization, enabling scalable, collision-free execution under changing environmental conditions. SPADES addresses the classical one-robot–one-task setting in static environments,

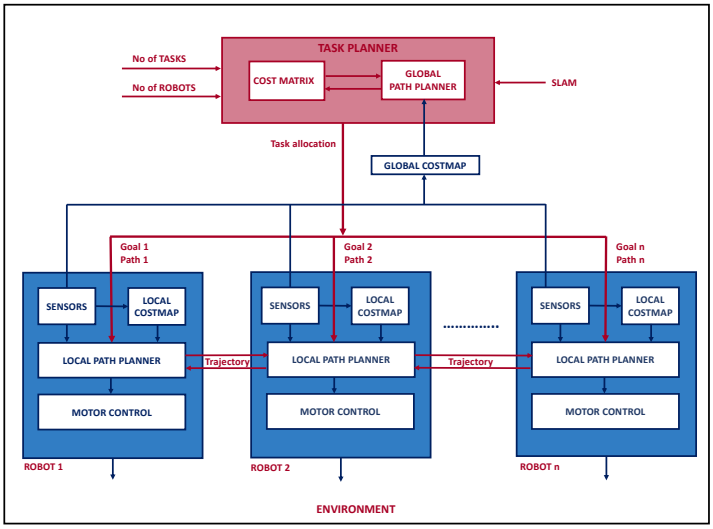


Figure 3.4: Extended-SPADES planning framework illustrating the interaction between centralized task allocation and distributed trajectory refinement for each robot.

whereas Extended-SPADES generalizes the framework to the more challenging case of $M \leq N$, dynamic task sets, and real-time disturbance handling. Figure 3.4 provides an overview of the Extended-SPADES architecture and the interaction between centralized planning and distributed execution.

3.6.1 Baseline Framework: SPADES

SPADES couples a centralized task assignment stage with a local, gradient-based motion planner. Initially, a global cost matrix $\mathbf{\Delta}$ is constructed using geometric paths generated by a global planner (E* in the original implementation [103]). The Hungarian algorithm [10] is then applied to compute an optimal one-to-one assignment between robots and tasks, minimizing the total assignment cost with polynomial complexity $\mathcal{O}(N^3)$. Once tasks are assigned, each robot refines its global reference path using *multi-CHOMP*, a velocity–time reparameterized extension of CHOMP that locally resolves inter-robot collisions by stretching or compressing trajectory segments without altering the global task allocation.

Under the assumptions $M = N$ and a static environment, SPADES is provably optimal: each task is completed exactly once and the total cost is minimized. However, its applicability is limited in large-scale or dynamic settings, as frequent re-computation of the Hungarian assignment becomes

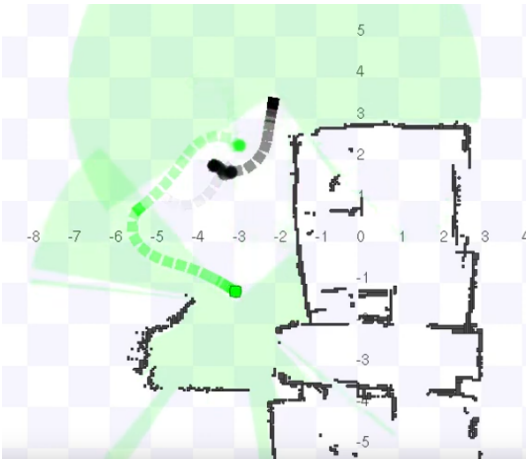


Figure 3.5: Local path conflict resolution using multi-CHOMP in the Stage simulator.

computationally expensive and may lead to oscillatory behavior when tasks or obstacles change.

3.6.2 Extended-SPADES: ITAPP for $M \leq N$ and Dynamic Environments

Extended-SPADES generalizes SPADES to address realistic logistics scenarios characterized by (i) significantly more tasks than robots, (ii) dynamically changing environments, and (iii) the need for balanced workload distribution. Rather than repeatedly solving a one-to-one assignment, Extended-SPADES decouples task allocation from motion execution and explicitly optimizes a *min–max* objective to balance robot makespan. Given the transition cost matrix $\mathbf{\Delta}$ and task execution vector $\mathbf{T}$, Extended-SPADES solves MRR-MM problem. Two task allocation strategies are interchangeable - textbfDeterministic Annealing (DA) solver based on Potts neurons yielding near-optimal, load-balanced solutions, and **graph-based heuristic** that sacrifices optimality in favor of scalability for very large task sets. The output is a set of ordered task chains and corresponding global reference paths. Each robot then independently refines its trajectory using multi-CHOMP, preserving the global allocation while resolving local conflicts.

3.6.3 Distributed Collision Resolution via Multi-CHOMP

To prevent inter-robot collisions during execution, multi-CHOMP augments the standard CHOMP objective with an *interference potential* that generates

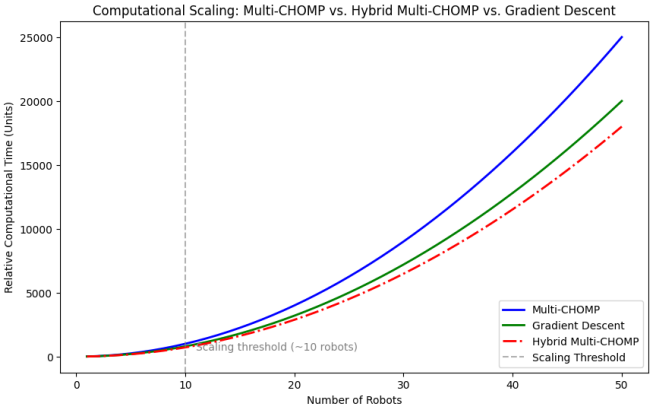


Figure 3.6: Computational scaling of multi-CHOMP, gradient descent, and hybrid multi-CHOMP.

repulsive forces between nearby trajectories:

$$F_{\text{inter}}[\xi] = \sum_{t=0}^{n} c(\|d_t\|)\, n_t, \qquad d_t = q_t^A - q_t^B, \quad n_t = \frac{d_t}{\|d_t\|}, \tag{3.16}$$

where $c(\cdot)$ is a distance-based cost function. Its gradient is combined with the standard smoothness and obstacle gradients and minimized via preconditioned gradient descent. Importantly, this optimization is executed *locally* on each robot, requiring only minimal exchange of time-stamped waypoints. Figure 3.5 illustrates local conflict resolution performed by multi-CHOMP in a Stage simulation **environment**[1]. Multi-CHOMP reparametrizes the time factor in the conflicting trajectories without losing its initial optimal paths.

3.6.4 Scalability and Hybrid Multi-CHOMP

While multi-CHOMP provides robust collision avoidance, its computational cost grows with the number of interacting robots. As shown in Figure 3.6, performance degradation becomes noticeable beyond approximately ten robots. To address this, a *hybrid multi-CHOMP* strategy was introduced, where trajectories are segmented and multi-robot optimization is applied only to conflicting segments, while standard CHOMP is used elsewhere. This hybrid approach preserves solution quality while significantly improving scalability, outperforming standard gradient descent for large robot teams and remaining robust to dynamic disturbances.

[1]https://www.youtube.com/watch?v=8kSTQDGHqTw

3.6.5 Algorithmic Overview

Algorithm 2 summarizes the Extended-SPADES planning loop, highlighting the separation between centralized task allocation and distributed trajectory refinement.

Algorithm 2: Extended-SPADES Framework

Data: Global map with M robots, N tasks, and task costs T and SLAM data
Result: List of obstacle-free optimal trajectory points with task allocation

for *each robot* $m \in \{1, \ldots, M\}$ *and each task* $n \in \{1, \ldots, N\}$ **do**
    Run $A*$ *path planner* on the global map and localization data
    return the global path P_{ij} for each consecutive task
end
Create the *cost matrix* Δ
Run the *Task Planner* on the cost matrix Δ
return the task allocation V and paths P_{ij}
for *each task allocation* $V_m \in \{1, \ldots, M\}$ **do**
    Run *multi−CHOMP* with input: V_m, global path P_{ij}, and local map
end
return Execute the resulting trajectories

3.6.6 Experimental Evaluation

A simple graphical user interface to study **multi-CHOMP**[1] with the interference objective was implemented using C++11 with the Eigen library and the GTK (GIMP ToolKit) library. **Extended-SPADES**[2] was tested on the 2.5-D simulation environment Stage with a number of ROS stacks for mapping and navigation. For calculating the cost matrix, path lengths for all combinations of robot–task, task–task, and task–robot are calculated, producing individual global paths between any two consecutive points which is illustrated in this **video**[3]. The extended-SPADES framework is implemented in this **video**[4]. In figure 3.7 for $M = 6$ and $N = 7$, the robots are shown in blue, the tasks are marked orange, and the final drop off points are green. A "quick and simple" routing solution is to group each tasks to the nearest robot. This is shown in Figure 3.7a and gives a cost of 63. However, in this

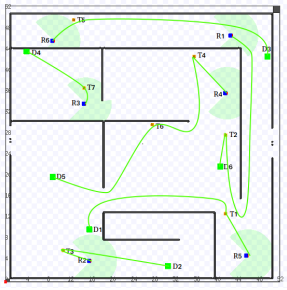
[1] https://github.com/jenniferdavid/multi-CHOMP
[2] https://github.com/jenniferdavid/taskallocation
[3] https://www.youtube.com/watch?v=clFyK9Sv7JI
[4] https://www.youtube.com/watch?v=gXSRuJjHJDM

case, though the heuristic method produces faster solution cost of 80.3 in 5.2 s compared to other methods, the DA gives a better solution of 59.78 in 20.2 s. As expected, SA at 0.999 gives the optimal solution of 55.97 at 1033.57 s. From this experiment, it could also be seen that the expected solution need not be the optimal solution always.

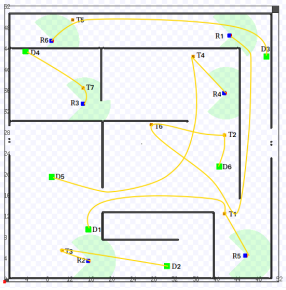
A small part of the experiment was also performed in a 3D simulation environment like a **harbour environment** created for the Cargo-ANTs project. [1]. However, the experiments were limited to smaller environments and fewer robots due to the complexity of the robots as well as the physics engine and dynamics in Gazebo.

Some of the key findings include: **Load balancing:** DA-based allocations reduce worst-case makespan by 15–30% compared to sum-optimal greedy baselines. **Scalability:** End-to-end planning remains below 10 s for up to 50 robots and 250 tasks, with seamless fallback to heuristic allocation beyond this scale. **Robustness:** Over 95% of inter-robot conflicts are resolved locally without triggering global replanning. Overall, Extended-SPADES realizes a complete and scalable framework, combining centralized min–max task allocation with distributed CHOMP-based trajectory refinement to deliver near-optimal, execution-safe plans for large robot fleets in dynamic environments.

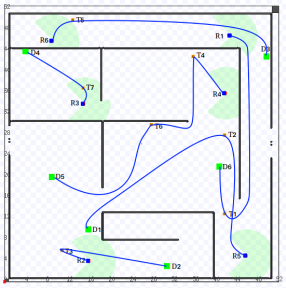
[1] https://github.com/jenniferdavid/Gazebo_Harbour_Models



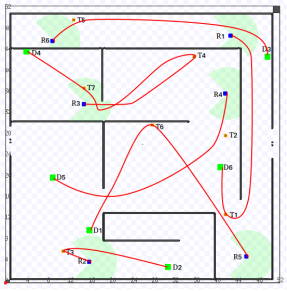
(a) The expected solution is where the robots does the nearest tasks. The solution cost is 63



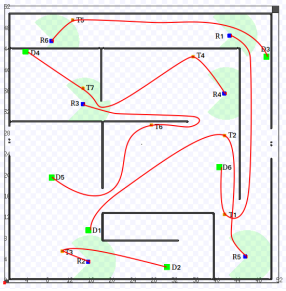
(b) The solution cost of the simple heuristic method is 80.3



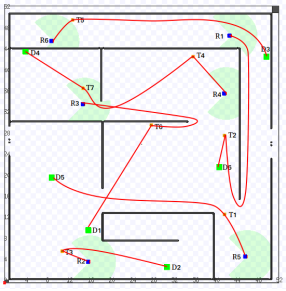
(c) The solution cost of DA at 0.9 annealing rate is 59.78



(d) The solution cost of SA at 0.9 annealing rate is 62.56



(e) The solution cost of SA at 0.99 annealing rate is 58.67



(f) The solution cost of SA at 0.999 annealing rate is 55.97 (The optimal solution)

Figure 3.7: Stage simulator world of a maze map with 6 robots (blue), 7 tasks (orange) and 6 dropoffs (green) which shows the task allocation and the path solution cost for different methods.

4. Conclusions

4.1 Thesis Summary

This thesis contributes to the field of *task and motion planning* for multi-robot systems by studying the Multi-Robot Routing problem with a min–max objective (MRR-MM) in application-driven scenarios inspired by logistics domains such as warehouses and container terminals. The work focuses on the challenge of balancing global coordination, local motion feasibility, and computational tractability in settings where multiple robots must operate concurrently under shared constraints. The contributions of the thesis are organized around four closely related components addressing problem formulation, algorithmic development, and system-level integration.

Integrated Task and Motion Planning Framework. A central contribution of this thesis is the development of a hybrid planning framework, referred to as *Extended-SPADES*, which combines centralized task allocation with distributed motion planning. Rather than fully coupling task allocation and motion planning into a single optimization problem, the framework separates global decision-making from local execution while allowing interaction between the two layers. Task allocation is computed centrally, while execution feasibility is maintained through distributed trajectory refinement using a multi-CHOMP planner. This architecture provides a structured and flexible integration of centralized and decentralized planning paradigms and enables coordinated execution in dynamic environments.

Task Allocation Problem and Formulation. To support systematic analysis of balanced task allocation, this thesis introduces a permutation-matrix formulation of the MRR-MM problem. Each feasible solution corresponds to a loop-free assignment and sequencing of tasks that explicitly minimizes the maximum cost incurred by any robot. The formulation provides a compact and expressive representation of the min–max routing objective and serves as a common basis for evaluating multiple solution strategies. The properties and implications of this formulation are analyzed in detail in Paper 1.

Methods for Solving the MRR-MM Problem. Building on this formulation, the thesis investigates deterministic annealing (DA) with Potts neurons as a solution method for the MRR-MM problem. By relaxing discrete assignment variables into continuous probability distributions and gradually enforcing discreteness through an annealing process, the method offers a deterministic alternative to stochastic and heuristic approaches. Problem-specific stabilization and normalization mechanisms are introduced to improve convergence behavior. Experimental results indicate that this approach can generate high-quality, balanced task allocations for small- to medium-sized problem instances, while highlighting scalability and computational trade-offs relative to heuristic and stochastic methods, as presented in Paper 2.

Motion Planning for Non-Holonomic Robots. To ensure that task allocations remain executable, this thesis extends gradient-based trajectory optimization methods to multi-robot, non-holonomic systems. The CHOMP algorithm is adapted to incorporate kinematic constraints, smoothness objectives, and inter-robot interaction costs relevant to differential-drive and car-like robots. A distributed multi-CHOMP formulation enables local trajectory refinement without invalidating global task assignments. These motion planning contributions are presented in Papers 3 and 4 and are integrated into the Extended-SPADES framework in Papers 5 and 6.

Together, these contributions establish a unified framework for studying integrated task allocation and motion planning in multi-robot systems. The formulations, algorithms, and system architecture developed in this thesis provide the foundation for the evaluation, limitations, and future research directions discussed in the following sections.

4.2 Key Results

The primary outcome of this thesis is the demonstration that treating multi-robot task allocation and trajectory planning as interrelated problems enables more balanced and feasible solutions, particularly when using a min–max objective that explicitly limits workload disparities across robots. Across the proposed formulations and frameworks, the results illustrate how different optimization strategies trade solution quality for computational efficiency under varying problem scales and execution constraints.

Within the Extended-SPADES framework and its SPADES predecessor, centralized task allocation combined with local trajectory refinement enables coordinated execution in the presence of dynamic obstacles and inter-robot

interactions. Simulation-based evaluations show that collision-free trajectories can be maintained for small to medium-sized robot fleets without invalidating initial task assignments. Real-world experiments conducted on a full-scale autonomous truck platform demonstrate real-time feasibility, confirming that kinematic constraints, obstacle avoidance, and trajectory smoothness can be satisfied under realistic sensing and control conditions.

For the MRR-MM problem, the permutation-matrix formulation facilitates systematic comparison of heuristic, simulated annealing, and deterministic annealing approaches. The results highlight distinct trade-offs: heuristic methods scale efficiently to large problem instances but often yield higher maximum robot costs; simulated annealing achieves near-optimal solutions at the expense of rapidly increasing computation time; and deterministic annealing provides competitive solution quality for small- to medium-scale instances with reduced runtime variability and faster convergence.

Evaluation metrics throughout the thesis include maximum robot cost, relative solution quality with respect to reference solutions, computational runtime, and scalability with respect to the number of robots and tasks. Across multiple datasets, the min–max objective consistently reduces extreme workload imbalances compared to min-sum formulations. For larger problem instances, task allocation and planning times remain within practical limits, supporting near-real-time deployment under appropriate assumptions.

4.3 Limitations

While the proposed formulations and algorithms demonstrate strong performance across a wide range of scenarios, several limitations remain that bound their applicability and scalability.

- **Scalability limits in very large instances.** Consistent with the NP-hard nature of the MRR-MM problem, computational complexity grows rapidly for very large problem instances. In particular, beyond approximately 500 tasks, the deterministic annealing (DA) solver becomes increasingly sensitive to the annealing schedule and may require multiple restarts to maintain solution quality.

- **Sinkhorn normalization overhead.** Despite optimizations such as coupled row–column updates, the combined Hungarian, balancing, and Sinkhorn normalization steps account for a significant share of the total computation time, especially at low temperature kT. This overhead limits the practical applicability of the approach for extremely large-scale task allocation problems.

Scenario	Extended-SPADES			SPADES			GAP		
	Completeness	Path Smoothness	Obs. Avoid	Completeness	Path Smoothness	Obs. Avoid	Completeness	Path Smoothness	Obs. Avoid
Corridor 1	0.6	0.90	0.75	0.5	0.75	0.78	0.92	0.88	0.90
Corridor 2	0.5	0.85	0.80	0.3	0.78	0.82	0.93	0.89	0.91
Circular Obstacles	0.8	0.95	1.00	0.6	0.77	0.84	0.91	0.86	0.92

Table 4.1: Comparison of the three methods for the given 3 complex scenarios in terms of completeness, path smoothness and obstacle avoidance success rate

- **Local-minima susceptibility of multi-CHOMP.** As a gradient-based method, the multi-CHOMP trajectory optimizer may converge to local minima in highly cluttered environments or dense robot swarms. In such cases, purely local optimization may fail to find feasible trajectories without additional global guidance.

- **Static cost assumption during allocation.** Task allocation is performed under the assumption that the cost matrix $\mathbf{\Delta}$ remains fixed during optimization. Significant dynamic changes in the environment, such as moving obstacles or large disturbances, may temporarily invalidate the optimality of the allocation until the next replanning cycle is executed.

- **Complex challenging scenario** Three commonly studied challenging scenarios, depicted in Figure 4.1, were also considered to evaluate extended-SPADES. As seen from the results in Table 4.1, it can be clearly seen that both SPADES and extended-SPADES did not perform well under the first two corridor scenarios. The algorithm is not able to find a solution and lacks completeness. The framework manages to maintain path smoothness and the obstacle avoidance success rate but fails to find a path. The main reason was due to path oscillations and local minima in trajectory planning. Though adaptive learning models and more efficient heuristic-based optimizations can improve performance, it is not guaranteed that extended-SPADES will work in such complex scenarios.

4.4 Outlook and Future Work

Several avenues for future research emerge from the results and limitations of this thesis, spanning algorithmic extensions, system-level integration, and emerging optimization paradigms.

Constraint-aware task allocation and dynamic feasibility. A natural extension of the proposed deterministic annealing framework is the explicit

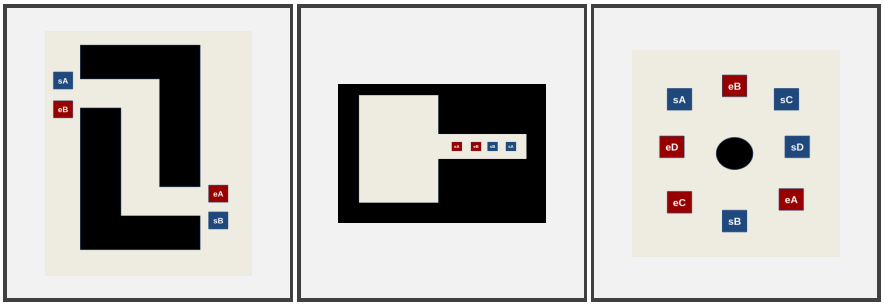


Figure 4.1: Three complex scenarios for multi-robot path planning problem – the corridor bottleneck, narrow passage, and intersection problems, which can explain the completeness of the algorithm.

incorporation of inequality constraints during optimization, rather than handling them indirectly through cost modeling or replanning. Recent work has shown that reformulating combinatorial allocation problems as continuous-time control systems, using Control Lyapunov Functions (CLFs) and Control Barrier Functions (CBFs), enables structured handling of both equality and inequality constraints while preserving convergence guarantees [104]. Integrating such control-theoretic constraint handling with the permutation-matrix formulation and annealing-based optimization developed in this thesis could improve robustness under capacity limits, safety constraints, and dynamically evolving task demands.

Adaptive and online task reallocation. The current framework assumes quasi-static cost matrices during task allocation. Future work could extend Extended-SPADES toward fully online operation, where task allocation and trajectory planning co-evolve in response to dynamic obstacles, task arrivals, or robot failures. This includes incorporating rolling-horizon optimization, incremental annealing updates, and tighter coupling between global task planners and local motion adaptors to reduce the need for global replanning in highly dynamic environments. Such extensions are particularly relevant for long-horizon logistics and warehouse scenarios considered throughout this thesis.

Hybrid optimization and large-scale scalability. While deterministic annealing provides strong performance for small- to medium-scale MRR-MM instances, scaling to very large problem sizes remains challenging. Recent studies on quantum and hybrid quantum–classical optimization methods indicate that such approaches can significantly reduce solution times for

large and dense combinatorial optimization problems [105]. These results motivate future investigation into hybrid architectures where classical task allocation methods, such as deterministic annealing, are combined with quantum or quantum-inspired solvers for subproblems with high combinatorial complexity.

Quantum formulations of multi-robot routing problems. The permutation-matrix formulation introduced in this thesis may be well suited for quantum optimization approaches. Mapping the MRR-MM problem to such formulations opens the possibility of exploiting emerging quantum optimization platforms. Systematic surveys of quantum combinatorial optimization in the NISQ era highlight routing, scheduling, and assignment problems as promising application domains for near-term quantum devices [106]. Future work may explore how min–max objectives, workload balancing, and sequencing constraints can be encoded efficiently within QUBO-based models, enabling empirical evaluation on hybrid quantum solvers.

System-level deployment and heterogeneous robot teams. Finally, future research may extend the proposed framework to heterogeneous robot teams with differing capabilities, dynamic task execution times, and energy constraints. Incorporating richer robot models and execution-level feedback would further strengthen the applicability of the framework to real industrial deployments, building on the simulation and experimental results presented in this work.

4.5 Concluding Remarks

Throughout this thesis, task allocation and motion planning are treated as *integrated decision problems*, in the sense that global task assignments explicitly account for motion-derived costs and execution feasibility is preserved during trajectory refinement. At the same time, the overall system is realized as a *hybrid architecture*, combining centralized task allocation with distributed, local motion planning to balance coordination quality with computational scalability.

This thesis shows that a mean-field deterministic annealing approach based on Potts neurons, when combined with a distributed multi-CHOMP motion planning layer, constitutes a viable integrated task and motion planning strategy for small- to medium-scale multi-robot systems operating in structured and moderately dynamic environments. The proposed Extended-SPADES framework demonstrates how global task allocation and local tra-

jectory refinement can be combined within a coherent architecture while preserving execution feasibility.

By introducing a structured formulation of the MRR-MM problem and evaluating deterministic optimization methods for its solution, this work provides a systematic alternative to purely heuristic or stochastic approaches, while explicitly characterizing their respective trade-offs. The extension of CHOMP to distributed, non-holonomic multi-robot planning further ensures that task allocations produced at the global level can be executed under realistic kinematic and collision-avoidance constraints.

Taken together, the contributions of this thesis illustrate how careful problem formulation, deterministic optimization, and distributed execution can be integrated to address key challenges in multi-robot coordination. Rather than aiming for universal optimality, the work emphasizes practical feasibility and transparent trade-off analysis, thereby contributing both methodological insights and experimentally grounded guidance for the design of autonomous multi-robot systems in logistics and industrial automation.

References

[1] C-I Liu, Hossein Jula, and Petros A Ioannou. **Design, simulation, and evaluation of automated container terminals**. *IEEE Transactions on intelligent transportation systems*, **3**(1):12–26, 2002. 1

[2] Kaveh Azadeh, René De Koster, and Debjit Roy. **Robotized and automated warehouse systems: Review and recent developments**. *Transportation Science*, **53**(4):917–945, 2019. 1

[3] Steven M LaValle. *Planning algorithms, Cambridge university press*. Cambridge university press, 2006. 2, 9

[4] Steven M LaValle and Seth A Hutchinson. **Optimal motion planning for multiple robots having independent goals**. *IEEE Transactions on Robotics and Automation*, **14**(6):912–925, 1998. 3

[5] Jur P Van Den Berg and Mark H Overmars. **Prioritized motion planning for multiple robots**. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 430–435. IEEE, 2005. 3

[6] Matthew Turpin, Kartik Mohta, Nathan Michael, and Vijay Kumar. **Goal assignment and trajectory planning for large teams of interchangeable robots**. *Autonomous Robots*, **37**(4):401–415, 2014. 3

[7] Matthew Turpin, Nathan Michael, and Vijay Kumar. **An approximation algorithm for time optimal multi-robot routing**. In *Algorithmic Foundations of Robotics XI*, pages 627–640. Springer, 2015. 6, 18, 19

[8] Lantao Liu and Dylan A Shell. **An anytime assignment algorithm: From local task swapping to global optimality**. *Autonomous Robots*, **35**(4):271–286, 2013. 3

[9] Matt Zucker, Nathan Ratliff, Anca D. Dragan, Mihail Pivtoraiko, Matthew Klingensmith, Christopher M. Dellin, J. Andrew Bagnell, and Siddhartha S. Srinivasa. **CHOMP: Covariant Hamiltonian optimization for motion planning**. **32**, pages 1164–1193, 2013. 6, 13, 17, 22, 33

[10] H. W. Kuhn. **The Hungarian method for the assignment problem**. *Naval Research Logistics Quarterly*, **2**(1-2):83–97, 1955. 7, 27, 37

[11] Brian P Gerkey and Maja J Matarić. **A formal analysis and taxonomy of task allocation in multi-robot systems**. *The International Journal of Robotics Research*, **23**(9):939–954, 2004. 12, 27

[12] G Ayorkor Korsah, Anthony Stentz, and M Bernardine Dias. **A comprehensive taxonomy for multi-robot task allocation**. *The International Journal of Robotics Research*, **32**(12):1495–1512, 2013. 27

[13] Sameh Sariel and Tucker Balch. **Efficient multi-robot coordination through auction-based task allocation and behavior-based execution**. *Journal of Artificial Intelligence Research*, **35**:725–755, 2009. 12

[14] Steven M LaValle. **Rapidly-exploring random trees: Progress and prospects**. In *Algorithmic and computational robotics: new directions: the fourth workshop on the algorithmic foundations of robotics*, pages 293–308, 2000. 12

[15] Howie Choset. **Coverage for robotics–A survey of recent results**. *Annals of mathematics and artificial intelligence*, **31**:113–126, 2001.

[16] Leping Yang, Dafeng Gu, Long Chen, and Huosheng Hu. **Multi-robot task allocation for robot soccer**. In *2008 IEEE International Conference on Robotics and Biomimetics*, pages 1202–1207. IEEE, 2008. 12

[17] Roni Stern, Ariel Felner, Solomon E Shimony, Eli Boyarski, Liron Cohen, T K Satish Kumar, Han Ma, David Tolpin, Guni Sharon, Glenn Wagner, et al. **Multi-agent pathfinding: Definitions, variants, and benchmarks**. *arXiv preprint arXiv:1906.08291*, 2019. 12

[18] David Silver. **Cooperative pathfinding**. In *Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment Conference*, 2005.

[19] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R Sturtevant. **Conflict-based search for optimal multi-agent pathfinding**. *Artificial Intelligence*, **219**:40–66, 2015. 12

[20] Luke Antonyshyn, Jefferson Silveira, Sidney Givigi, and Joshua Marshall. **Multiple mobile robot task and motion planning: A survey**. *ACM Computing Surveys*, **55**(10):1–35, 2023. 12, 13, 15

[21] Han Ma, Sven Koenig, and Nora Ayanian. **Multi-agent path finding with deadline constraints using conflict-based search**. *IJCAI*, 2016. 12, 16, 19

[22] Minghua Liu, Hang Ma, Jiaoyang Li, and Sven Koenig. **Task and path planning for multi-agent pickup and delivery**. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2019. 12, 26

[23] Michail G Lagoudakis, Evangelos Markakis, David Kempe, Pinar Keskinocak, Anton J Kleywegt, Sven Koenig, Craig A Tovey, Adam Meyerson, and Sonal Jain. **Auction-Based Multi-Robot Routing.** 13, 16, 18, 19, 20

[24] Howie Choset, Kevin Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E Kavraki, and Sebastian Thrun. *Principles of robot motion: theory, algorithms, and implementation*. MIT press, 2005. 13, 17

[25] John Schulman, Jonathan Ho, Alex X Lee, Ibrahim Awwal, Henry Bradlow, and Pieter Abbeel. **Motion planning with sequential convex optimization and convex collision checking**. *International Journal of Robotics Research*, **33**(9):1251–1270, 2014. 13, 17

[26] Masoumeh Mansouri, Federico Pecora, and Peter Schüller. **Combining task and motion planning: Challenges and guidelines**. *Frontiers in Robotics and AI*, **8**:637888, 2021. 13

[27] Leslie Pack Kaelbling and Tomás Lozano-Pérez. **Hierarchical task and motion planning in the now**. In *2011 IEEE International Conference on Robotics and Automation*, pages 1470–1477, 2011. 14

[28] Tomás Lozano-Pérez and Leslie Pack Kaelbling. **A constraint-based method for solving sequential manipulation planning problems**. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3684–3691, 2014. 14

[29] João Salvado, Robert Krug, Masoumeh Mansouri, and Federico Pecora. **Motion Planning and Goal Assignment for Robot Fleets Using Trajectory Optimization**. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7939–7946, 2018. 14

[30] João Salvado, Masoumeh Mansouri, and Federico Pecora. **Combining Multi-Robot Motion Planning and Goal Allocation using Roadmaps**. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10016–10022, 2021. 14

[31] Javier de Lope, Darío Maravall, and Yadira Quiñonez. **Self-organizing techniques to improve the decentralized multi-task distribution in multi-robot systems**. *Neurocomputing*, **163**:47–55, 2015. Recent Advancements in Hybrid Artificial Intelligence Systems and its Application to Real-World Problems Progress in Intelligent Systems Mining Humanistic Data. 14

[32] Mustafa Mukadam, Jing Dong, Xinyan Yan, Frank Dellaert, and Byron Boots. **Continuous-time Gaussian process motion planning via probabilistic inference**. *The International Journal of Robotics Research*, **37**(11):1319–1340, 2018. 15

[33] Alaa Khamis, Ahmed Hussein, and Ahmed Elmogy. **Multi-robot task allocation: A review of the state-of-the-art**. *Cooperative Robots and Sensor Networks*, pages 31–51, 2015. 15, 17

[34] Zhenyu Yan, Nicolas Jouandeau, and Arab Ali Cherif. **A survey on coordination of multiple unmanned vehicles: taxonomy and challenges**. *International Journal of Advanced Robotic Systems*, **10**(12):399, 2013. 15, 17

[35] Daniel Claes, Frans Oliehoek, Hendrik Baier, and Karl Tuyls. **Decentralised online planning for multi-robot warehouse commissioning**. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 492–500. International Foundation for Autonomous Agents and Multiagent Systems, 2017. 15

[36] Joseph L Baxter, EK Burke, Jonathan M Garibaldi, and Mark Norman. **Multi-robot search and rescue: A potential field based approach**. In *Autonomous robots and agents*, pages 9–16. Springer, 2007. 15, 16

[37] Wolfram Burgard, Mark Moors, Cyrill Stachniss, and Frank E Schneider. **Coordinated multi-robot exploration**. *IEEE Transactions on robotics*, **21**(3):376–386, 2005. 15

[38] Dieter Fox, Jonathan Ko, Kurt Konolige, Benson Limketkai, Dirk Schulz, and Benjamin Stewart. **Distributed multirobot exploration and mapping**. *Proceedings of the IEEE*, **94**(7):1325–1339, 2006. 15, 16

[39] Christian Groth and Dominik Henrich. **Single-shot learning and scheduled execution of behaviors for a robotic manipulator**. In *ISR/Robotik 2014; 41st International Symposium on Robotics*, pages 1–6. VDE, 2014. 15

[40] Thomas Lemaire, Rachid Alami, and Simon Lacroix. **A distributed tasks allocation scheme in multi-UAV context**. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, **4**, pages 3622–3627. IEEE, 2004. 15

[41] Maria Gini. **Multi-robot allocation of tasks with temporal and ordering constraints**. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 15

[42] Tolga Bektas. **The multiple traveling salesman problem: an overview of formulations and solution procedures**. *Omega*, **34**(3):209–219, 2006. 15

[43] Mitchell McIntire, Ernesto Nunes, and Maria Gini. **Iterated multi-robot auctions for precedence-constrained task scheduling**. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 1078–1086. International Foundation for Autonomous Agents and Multiagent Systems, 2016. 15, 16

[44] Chayan Sarkar, Himadri Sekhar Paul, and Arindam Pal. **A scalable multi-robot task allocation algorithm**. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–9. IEEE, 2018. 15

[45] Matthew Gombolay, Ronald Wilcox, and Julie Shah. **Fast scheduling of multi-robot teams with temporospatial constraints**. 2013. 15

[46] Jens Clausen. **Branch and bound algorithms-principles and examples**. 1999. 16

[47] Chun Liu and Andreas Kroll. **A centralized multi-robot task allocation for industrial plant inspection by using a* and genetic algorithms**. In *International Conference on Artificial Intelligence and Soft Computing*, pages 466–474. Springer, 2012. 16

[48] Jianping Wang, Yuesheng Gu, and Xiaomin Li. **Multi-robot task allocation based on ant colony algorithm**. *Journal of Computers*, **7**(9):2160–2167, 2012. 16

[49] Xun Li and Hong-xu Ma. **Particle swarm optimization based multi-robot task allocation using wireless sensor network**. In *2008 International Conference on Information and Automation*, pages 1300–1303. IEEE, 2008. 16

[50] Alejandro R Mosteo and Luis Montano. **Simulated annealing for multi-robot hierarchical task allocation with flexible constraints and objective functions**. In *Workshop on Network Robot Systems: Toward Intelligent Robotic Systems Integrated with Environments". IROS*, 2006. 16

[51] James Parker, Ernesto Nunes, Julio Godoy, and Maria Gini. **Exploiting spatial locality and heterogeneity of agents for search and rescue teamwork**. *Journal of Field Robotics*, **33**(7):877–900, 2016. 16

[52] Lantao Liu, Dylan A Shell, and Nathan Michael. **From selfish auctioning to incentivized marketing**. *Autonomous Robots*, **37**(4):417–430, 2014. 16

[53] Rajiv T Maheswaran, Milind Tambe, Emma Bowring, Jonathan P Pearce, and Pradeep Varakantham. **Taking DCOP to the real world: Efficient complete solutions for distributed multi-event scheduling**. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 310–317. IEEE Computer Society, 2004. 16

[54] WC Ng, KL Mak, and YX Zhang. **Scheduling trucks in container terminals using a genetic algorithm**. *Engineering Optimization*, **39**(1):33–47, 2007. 16

[55] Steven M LaValle and James J Kuffner Jr. **Randomized kinodynamic planning**. *The International Journal of Robotics Research*, **20**(5):378–400, 2001. 17

[56] Lydia E Kavraki, Petr Svestka, Jean-Claude Latombe, and Mark H Overmars. **Probabilistic roadmaps for path planning in high-dimensional configuration spaces**. *IEEE Transactions on Robotics and Automation*, **12**(4):566–580, 1996. 17

[57] Maxim Likhachev, Dave Ferguson, Geoff Gordon, Anthony Stentz, and Sebastian Thrun. **Planning in large dynamic environments using anytime heuristic search**. *Journal of Artificial Intelligence Research*, **32**:267–306, 2009. 17

[58] Mrinal Kalakrishnan, Sachin Chitta, Evangelos Theodorou, Peter Pastor, and Stefan Schaal. **STOMP: Stochastic trajectory optimization for motion planning**. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4569–4574, 2011. 17, 22

[59] R. Matoušek R. Parák, J. Kůdela and M. Juříček. **Deep-Reinforcement-Learning-Based Motion Planning for a Wide Range of Robotic Structures**. *Computation*, **12**(6):116, 2024. 17

[60] C. R. Garrett A. Handa R. Salakhutdinov M. Dalal, A. Mandlekar and D. Fox. **Imitating Task and Motion Planning with Visuomotor Transformers**. *Proceedings of The 7th Conference on Robot Learning*, **229**:2565–2593, 2023. 17

[61] Z. He Z. Dong and C. Song. **A review of mobile robot motion planning methods: from classical motion planning workflows to reinforcement learning-based architectures**. *Journal of Systems Engineering and Electronics*, **34**(2):439–459, 2023. 17

[62] A. Asgharpoor Golroudbari and M. H. Sabour. **Recent Advancements in Deep Learning Applications and Methods for Autonomous Navigation: A Comprehensive Review**. *arXiv preprint arXiv:2302.11089*, 2023. 17

[63] M Ani Hsieh and Vijay Kumar. **Decentralized controllers for shape generation with robotic swarms**. *Robotica*, **26**(5):691–701, 2008. 17

[64] M Bernardine Dias, Robert Zlot, Nidhi Kalra, and Anthony Stentz. **Market-based multirobot coordination: A survey and analysis**. *Proceedings of the IEEE*, **94**(7):1257–1270, 2006. 17

[65] Lynne E Parker. **L-ALLIANCE: A mechanism for adaptive action selection in heterogeneous multi-robot teams**. Technical report, Oak Ridge National Lab., TN (United States), 1995. 18, 20

[66] Michel Gendreau Gilbert Laporte Franca, Paulo M. and Felipe M. Muller. **The m-Traveling Salesman Problem with Minmax Objective**. *Transportation Science, JSTOR*, **29**:267–275, 1995. 18

[67] Moustapha Diaby. **Linear programming formulation of the multi-depot multiple traveling salesman problem with differentiated travel costs**. *Traveling Salesman Problem, Theory and Applications*, pages 257–282, 2010. 18

[68] Paul Oberlin, Sivakumar Rathinam, and Swaroop Darbha. **A transformation for a heterogeneous, multiple depot, multiple traveling salesman problem**. In *2009 American Control Conference*, pages 1292–1297. IEEE, 2009.

[69] Salma Naccache, Jean-François Côté, and Leandro C Coelho. **The multi-pickup and delivery problem with time windows**. *European Journal of Operational Research*, **269**(1):353–362, 2018.

[70] Paulo M França, Michel Gendreau, Gilbert Laporte, and Felipe M Müller. **The m-traveling salesman problem with minmax objective**. *Transportation Science*, **29**(3):267–275, 1995. 18

[71] Zhou Xu, Dongsheng Xu, and Wenbin Zhu. **Approximation results for a min–max location-routing problem**. *Discrete Applied Mathematics*, **160**(3):306–320, 2012. 18

[72] Robert Pěnička, Jan Faigl, Petr Váňa, and Martin Saska. **Dubins orienteering problem**. *IEEE Robotics and Automation Letters*, **2**(2):1210–1217, 2017. 18

[73] Nicos Christofides and S Korman. **Note—a computational survey of methods for the set covering problem**. *Management Science*, **21**(5):591–599, 1975. 19

[74] Jean-François Cordeau and Gilbert Laporte. **The dial-a-ride problem: models and algorithms**. *Annals of operations research*, **153**(1):29–46, 2007. 19

[75] Esther M. Arkin, Refael Hassin, and Asaf Levin. **Approximations for Minimum and Min-Max Vehicle Routing Problems**. *J. Algorithms*, **59**(1):1–18, April 2006. 19

[76] Lingzhi Luo, Nilanjan Chakraborty, and Katia Sycara. **A distributed algorithm for constrained multi-robot task assignment for grouped tasks**. 19

[77] Jan Faigl, Petr Váňa, Robert Pěnička, and Martin Saska. **Unsupervised learning-based flexible framework for surveillance planning with aerial vehicles**. *Journal of Field Robotics*, **36**(1):270–301, 2019. 19

[78] Lynne E Parker. **ALLIANCE: An architecture for fault tolerant multirobot cooperation**. *IEEE transactions on robotics and automation*, **14**(2):220–240, 1998. 19

[79] Laurence A. Wolsey. *Integer Programming*. John Wiley & Sons, 1999. 19

[80] Vijay V. Vazirani. *Approximation Algorithms*. Springer, 2001. 19

[81] Vašek Chvátal. **A greedy heuristic for the set-covering problem**. *Mathematics of Operations Research*, **4**(3):233–235, 1979. 19

[82] Fred Glover. **Tabu search—Part I**. *ORSA Journal on Computing*, **1**(3):190–206, 1989. 19

[83] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. **Optimization by simulated annealing**. *science*, **220**(4598):671–680, 1983. 19

[84] John H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, 1975. 20

[85] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992. 20

[86] Marco Dorigo and Luca Maria Gambardella. **Ant colony system: a cooperative learning approach to the traveling salesman problem**. *IEEE Transactions on Evolutionary Computation*, **1**(1):53–66, 1997. 20

[87] James Kennedy and Russell Eberhart. **Particle swarm optimization**. In *Proceedings of the IEEE International Conference on Neural Networks*, **4**, pages 1942–1948, 1995. 20

[88] Edmund K. Burke, Michel Gendreau, Matthew Hyde, Gabriela Ochoa, Tony Prescott, and Susan Varley. **Hyper-heuristics: A survey of the state of the art**. *Journal of the Operational Research Society*, **64**(12):1695–1724, 2013. 20

[89] John J Hopfield and David W Tank. **Neural computation of decisions in optimization problems**. *Biological cybernetics*, **52**(3):141–152, 1985. 20, 21

[90] Irwan Bello, Hieu Pham, Quoc V. Le, Mohammad Norouzi, and Samy Bengio. **Neural Combinatorial Optimization with Reinforcement Learning**. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 20

[91] David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. **A learning algorithm for Boltzmann machines**. *Cognitive Science*, **9**(1):147–169, 1985. 21

[92] John J Hopfield. **Neural networks and physical systems with emergent collective computational abilities**. *Proceedings of the national academy of sciences*, **79**(8):2554–2558, 1982. 21

[93] Carsten Peterson and Bo Söderberg. **A new method for mapping optimization problems onto neural networks**. *International Journal of Neural Systems*, **1**(01):3–22, 1989. 21

[94] John T. Betts. **Optimal Methods for Trajectory Planning and Control**. *Advances in the Astronautical Sciences*, **108**:331–358, 2001. 22

[95] Srinivas Akella and Shankar Sastry. **Path-Constrained Trajectory Optimization Using Sparse Sequential Quadratic Programming**. *Journal of Guidance, Control, and Dynamics*, **22**(6):948–954, 1999. 22

[96] Yohav Tassa and Emanuel Todorov. **Control-Limited Differential Dynamic Programming**. *Proceedings of Robotics: Science and Systems (RSS)*, 2008. 22

[97] James B. Rawlings and David Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009. 22

[98] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. **Proximal Policy Optimization Algorithms**. *arXiv preprint arXiv:1707.06347*, 2017. 23

[99] Ernesto Nunes, Marie Manner, Hakim Mitiche, and Maria Gini. **A taxonomy for task allocation problems with temporal and ordering constraints**. *Robotics and Autonomous Systems*, **90**:55–70, 2017. 27

[100] Rainer E Burkard and Eranda Cela. **Linear assignment problems and extensions**. In *Handbook of combinatorial optimization*, pages 75–149. Springer, 1999. 27

[101] Jari Hakkinen, Martin Lagerholm, Carsten Peterson, and Bo Soderberg. **Local routing algorithms based on Potts neural networks**. *IEEE transactions on neural networks*, **11**(4):970–977, 2000. 31

[102] Richard Sinkhorn and Paul Knopp. **Concerning nonnegative matrices and doubly stochastic matrices**. *Pacific Journal of Mathematics*, **21**(2):343–348, 1967. 32

[103] Roland Philippsen and Roland Siegwart. **Smooth and efficient obstacle avoidance for a tour guide robot**. In *None*, number LSA-CONF-2003-018, 2003. 37

[104] Alisina Bayati, Dhananjay Tiwari, and Srinivasa Salapaka. **A Control Barrier Function Approach to Constrained Resource Allocation Problems in a Maximum Entropy Principle Framework**. *arXiv preprint arXiv:2504.01378*, 2025. 47

[105] Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Benjamin Weder, and Frank Leymann. **Quantum combinatorial optimization in the nisq era: A systematic mapping study**. *ACM Computing Surveys*, **56**(3):1–36, 2023. 48

[106] Seongmin Kim, Sang-Woo Ahn, In-Saeng Suh, Alexander W Dowling, Eungkyu Lee, and Tengfei Luo. **Quantum annealing for combinatorial optimization: a benchmarking study**. *npj Quantum Information*, **11**(1):77, 2025. 48

Jennifer David

Jennifer David is a Mechatronics Engineer who received her MS (by Research) in Robotics from the Indian Institute of Technology Madras in 2014. Her research interests include perception, planning, and control of wheeled robotic systems. She has extensive industry experience in autonomous driving for cars and heavy-duty vehicles in the automotive sector, as well as in perception and planning for mobile robots in nuclear robotics applications. Alongside her research, she also contributes to STEM outreach activities aimed at promoting science and engineering education among school students.

School of Information Technology

ISBN: 978-91-90123-06-5 (printed)
Halmstad University Dissertations, 2026

HALMSTAD
UNIVERSITY