# *Integrated grasp and motion planning*

**Author:**

Salah Ebrahimpour

**Programme:**

MSc Artificial Intelligence with Business Strategy

**Supervisor:**

Dr. Martin Rudorfer

**Co-Supervisor:**

Dr. Anthony Henry

**Date:**

September 2024

# Abstract

This dissertation explores the integration of grasp and motion planning in robotic systems, focusing on the evaluation of the RRT* algorithm and its performance in comparison to benchmark algorithms such as J+-RRT and IK-RRT. The primary objective is to assess the efficiency of the RRT* algorithm in handling complex and dynamic environments, emphasizing key metrics such as planning time and success rate.

The study also includes a business strategy component, examining the commercialization potential of the integrated system. Through strategic use of tools like SWOT, TOWS, and Porter's Five Forces, this research evaluates market opportunities and competitive positioning, identifying industries where automation and robotics are in high demand, such as manufacturing, healthcare, and logistics.

Experimental results highlight the strengths of the RRT* algorithm in less cluttered environments, while also addressing its limitations in more obstacle-dense scenarios. The dissertation concludes with recommendations for further improvements to the algorithm and outlines strategies for successfully bringing the technology to market, leveraging technical excellence and strategic partnerships.

# Acknowledgement

I would like to express my deepest gratitude to my supervisors, Dr. Martin Rudorfer and Dr. Anthony Henry, for their invaluable guidance, support, and expertise throughout this project. Their insights and encouragement were instrumental in shaping the direction of this dissertation, and I am incredibly grateful for their patience and mentorship.

I would also like to thank my family and friends for their constant support and encouragement, providing me with the motivation to keep going, especially during challenging times. Special thanks to my colleagues and peers at Aston University for their collaboration and feedback, which contributed significantly to the development of my research.

Finally, I extend my appreciation to the faculty and staff of the MSc Artificial Intelligence with Business Strategy program for providing a conducive environment for learning and research. This journey has been an enriching experience, and I am thankful to everyone who has played a part in it.

# Contents

## Figures

## Abbreviations

# Chapter 1: Introduction

## 1.1.  Project background

As robotics continues to evolve, the need for more sophisticated systems capable of autonomous, complex manipulation tasks becomes ever more pressing. Traditionally, the processes of grasp and motion planning have been treated separately in robotic systems. Grasp planning determines how a robot should grip an object based on factors like the object's shape, weight, and texture. Motion planning, on the other hand, focuses on the robot's path and the obstacles it must avoid when executing tasks (Muhayyuddin et al., 2015). However, the separation of these two processes can lead to suboptimal performance, particularly in dynamic environments where grasp configuration can affect motion feasibility and vice versa (Ali & Lee, 2020).

To address these issues, researchers have started exploring integrated approaches, where grasp and motion planning are considered simultaneously. These unified systems aim to enhance efficiency, adaptability, and reliability in robotic tasks. For example, the Grasp-RRT approach represents one such integration, optimizing both grasp and motion planning within the same framework (Ali & Lee, 2020).

## 1.2.  Research Objectives

The main objective of this research is to evaluate the performance of the RRT* algorithm in the context of integrated grasp and motion planning tasks. Specifically, this study aims to assess the algorithm's ability to solve planning tasks within benchmark scenarios introduced by Rudorfer et al., focusing on key performance metrics such as planning time and success rate.

A significant goal of the research is to compare the performance of the RRT* algorithm against the benchmark results provided in Jogramop framework (Rudorfer et al, 2024), particularly in relation to other algorithms such as J+-RRT and IK-RRT. This comparison seeks to identify the strengths and weaknesses of RRT*, highlighting how well it performs in terms of runtime and success rate under the same conditions as the baseline algorithms.

Additionally, the research aims to analyze the adaptability of the RRT* algorithm when applied to different environmental configurations, including variations in object placements and confined spaces. This will provide insights into how RRT* handles dynamic environments and complex obstacle layouts.

Ultimately, the research seeks to contribute to the development of more efficient integrated grasp and motion planning strategies by offering insights into the applicability of the RRT* algorithm for real-world robotic tasks. Based on the comparative analysis, this study will also provide recommendations for improving the algorithm's efficiency and robustness in future applications.
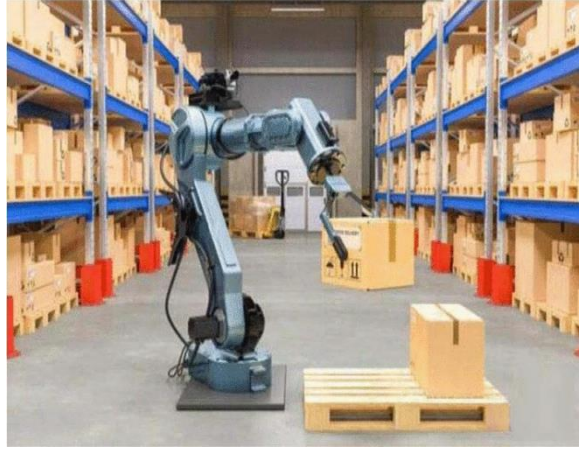
## 1.3. Research Questions/Hypotheses

This research is guided by several key questions and hypotheses:

- How does the performance of RRT* compare to benchmark algorithms such as J+RRT and IK-RRT in terms of planning time and success rate?
- Does RRT* perform better in environments with lower obstacle densities, yielding shorter planning times and higher success rates compared to other algorithms?
- Can RRT* efficiently adapt to complex and confined spaces while maintaining a high success rate?
- Does the success rate of RRT* decline in complex environments due to increased difficulty in navigating through dense obstacles?

The performance of RRT* in motion planning has been widely studied, with previous research demonstrating that RRT* is asymptotically optimal, meaning that as the number of samples increases, the solution converges to the optimal path with high probability (Gammell et al., 2015; Pérez et al., 2012). This characteristic is especially advantageous in environments with lower obstacle densities, where RRT* can utilize its sampling efficiency to yield shorter planning times and higher success rates compared to other algorithms that may not exhibit the same level of optimality, such as J+-RRT and IK-RRT (Qureshi & Ayaz, 2015). Studies have shown that in less cluttered environments, RRT* can outperform its predecessors in both speed and success, thanks to its iterative path refinement process (Tahir et al., 2018). Based on these findings, it is hypothesized that RRT* will perform competitively with these benchmark algorithms in terms of planning time and success rate.

Moreover, the research aims to explore whether the RRT* algorithm can efficiently adapt to complex and confined spaces while maintaining a high success rate. RRT*'s incremental nature allows it to adjust dynamically to environmental changes, making it well-suited for challenging environments (Karaman et al., 2011; Lee & Lee, 2012). However, as the complexity of the space increases, it is hypothesized that the success rate of RRT* may decline due to the increased difficulty in navigating through dense obstacles. This aligns with research indicating that while RRT* performs well in simpler environments, its success rate can be hindered in more complex scenarios, where pathfinding becomes more convoluted (Peng et al., 2023).

In conclusion, the RRT* algorithm shows promising performance when compared to benchmark algorithms, particularly in less cluttered environments. Its ability to adapt to complex spaces is promising, but challenges in dynamic environments highlight the need for optimization. These questions and hypotheses aim to investigate the potential of RRT* while addressing its limitations in real-world applications.

*Figure 2 Robotic arm handling goods in the warehousing industry (Yi et al., 2022)*

## 1.4. Significance of the Study

As robotic systems become more prevalent in sectors such as manufacturing, healthcare, and autonomous services, their ability to perform complex manipulation tasks in dynamic and uncertain environments is increasingly important. This research focuses on assessing the RRT* algorithm's effectiveness in planning both grasp and motion concurrently, particularly in terms of planning time, success rate, and adaptability.

By comparing the RRT* algorithm to established benchmarks, this study offers critical data on its strengths and limitations when applied to real-world scenarios where both grasping and motion must be planned efficiently. These findings have broader implications for improving robotic systems' performance in dynamic environments and will contribute to the development of autonomous systems that are more reliable, adaptable, and capable of handling complex tasks.

Ultimately, the outcomes of this research could enhance the performance of robotic systems in various industries, making them more versatile and effective in executing real-world tasks with greater precision and efficiency.

## 1.5.  Scope and Limitations

This research focuses on the development and evaluation of integrated grasp and motion planning algorithms, specifically in the context of robotic manipulation tasks. The experiments are conducted in simulated environments to ensure repeatability and control over variables. However, while the simulation allows for thorough testing under a variety of conditions, real-world testing of the algorithms may present challenges such as sensor inaccuracies, mechanical limitations of robots, or unexpected environmental factors. These limitations must be addressed in future research to fully realize the practical application of the proposed methods.

## 1.6.  Thesis Structure

This dissertation is structured across several chapters that comprehensively examine the integration of grasp and motion planning in robotics. The organization is as follows: Chapter 1 introduces the project, outlining the background, research objectives, and key questions guiding the study. Chapter 2 presents a literature review, examining previous work on grasp and motion planning, with an emphasis on integrated approaches and existing gaps. Chapter 3 defines the problem and research gap addressed by the study. Chapter 4 describes the methodology, detailing the research design, algorithms used, and experimental setup. Chapter 5 covers project management aspects, while Chapter 6 evaluates the performance of the algorithms, while focusing on metrics such as planning time and success rate. Chapter 7

outlines the business strategy and will provide a commercialization plan for the integrated grasp and motion planning system. In this section we discuss target industries, potential business models, and industry partnerships. Finally, Chapter 8 summarizes the key findings, discusses the implications, and offers recommendations for future research.

# Chapter 2: Literature Review

## 2.1. Introduction

The field of robotics has seen significant advancements in recent years, particularly in the domain of integrated grasp and motion planning. While motion planning is a fundamental aspect of robotics (Elbanhawi & Simić, 2014), the simultaneous planning of grasping an object and the robot's motion remains a notable challenge (Bütepage et al., 2019). This challenge is further intensified by the need to overcome computational obstacles related to sensing, grasp analysis, motion planning, and the execution of the robot arm's movements (Ichnowski et al., 2020).

In this section, we will explore the historical development and theoretical foundations of grasp planning and motion planning, examining the major theories and models that have shaped these fields. We will then explore the integrated approaches that combine these two planning processes and will discuss the challenges and innovations that have emerged from this integration. By critically analysing the existing literature, we aim to identify gaps and areas for future research, setting the stage for the subsequent chapters of this dissertation. This review will provide the necessary context and background for understanding the contributions and implications of our research on integrated grasp and motion planning.

## 2.2. Overview of Grasp Planning

Grasp planning in robotics has advanced significantly, evolving from basic algorithms for simple objects in controlled environments to more sophisticated approaches capable of handling complex and irregularly shaped objects in dynamic settings. Initially, the focus was on convex objects that a single robot could grasp, aiming to optimize stability and grasp force on the robot's contact points (Zafra-Urrea, 2023). As computational power and sensor technology progressed, research expanded to include the manipulation of objects with robotic hands in

dynamic conditions, that lead to the development of various grasp planning algorithms to identify stable grasps (Dharbaneshwer et al., 2020).

## Major Theories and Models

Grasp planning in robotics is supported by several foundational theories, including Force Closure, Form Closure, and Task-Oriented Grasping. Force Closure emphasizes the robot's ability to resist external forces from any direction, ensuring a stable grip through the application of balanced forces. This theory is essential for tasks that require robustness to external disturbances, such as handling in dynamic environments (Bicchi, 1995). On the other hand, Form Closure focuses on geometric stability, where the object's movement is constrained by the positioning of the contact points, providing a secure grasp without relying solely on applied forces (Mishra et al., 1987).

Task-Oriented Grasping shifts the focus from pure stability to optimizing the grasp for the specific task the robot needs to perform. It considers factors like object manipulation and positioning, making it particularly useful in scenarios where the robot needs to perform a sequence of actions with the object, such as in industrial automation or healthcare robotics (Ciocarlie et al., 2009). This approach has been introduced to compute the most suitable grasp for a given task based on a simplified geometrical and structural description of the object and the task requirements (Prats et al., 2007). Recent advancements have also explored the use of over-segmented meshes and relational databases to improve grasp planning, particularly in regrasp planning scenarios (Wan & Harada, 2017). Precision grasp planning for multi-fingered hands involves analysing grasp stability, synthesis, and object representation (Yan et al., 2019).

Research by Bertoni et al. (2021) introduces a generic grasp planning pipeline that enables transparent and generic grasp planning procedures, automating grasping actions irrespective of the end-effector kinematic structure. Tsuji et al. (2010) demonstrate grasp planning for multi-fingered hands with humanoid robots, emphasizing the importance of effective grasp planning through simulation and experimental results. Rubert et al. (2017) focus on characterizing grasp quality metrics, highlighting their significant role in the analytical approach to grasp planning.

## Techniques and Algorithms

Grasp planning techniques and algorithms have significantly evolved, with several prominent methods shaping modern robotic grasping. Analytical approaches rely on mathematical models to evaluate potential grasps by considering the geometry and physical properties of objects. These models allow for precise predictions of grasp stability and performance, often using principles from physics and geometry to calculate the forces and torques involved (Ferrari & Canny, 1992). Sampling-based methods focus on generating many potential grasps, which are then evaluated against predefined criteria such as stability or task suitability. These methods are advantageous when dealing with complex objects, as they provide flexibility in grasp selection through exploration of diverse grasp configurations (Bohg et al., 2014).

In contrast, machine learning approaches have gained prominence by utilizing large datasets to train models capable of predicting successful grasps based on sensory data, including visual and tactile inputs. These approaches enable robots to generalize from past experiences and adapt to new objects and environments with higher accuracy and efficiency. Machine learning techniques such as deep learning have been particularly effective in improving grasp prediction by learning complex relationships between object features and grasp success (Mahler et al., 2017). These advancements have greatly expanded the capabilities of robotic systems in handling diverse tasks and environments.

## Recent Advances

Recent research in grasp planning has concentrated on improving the robustness and adaptability of robotic systems. One of the key advancements is the use of deep learning models, which leverage neural networks to learn complex grasping strategies from extensive datasets. These models have significantly enhanced a robot's ability to generalize across various objects and environments, making grasping more reliable and versatile in real-world applications (Mahler et al., 2019). By training on a diverse range of object geometries and grasp configurations, deep learning approaches enable robots to handle previously unseen objects with greater precision and efficiency.

Another significant advancement is the integration of multiple sensors, including visual and tactile sensors, to provide a comprehensive understanding of both the object and its surrounding environment. This multi-modal sensor integration enhances a robot's perception, allowing for more informed decision-making during grasp planning (Calandra et al., 2018). Additionally, real-time grasp planning algorithms have been developed, enabling robots to dynamically adapt to changes in the environment and object positioning. These algorithms process sensory data rapidly, allowing robots to adjust their grasps in real-time, which is crucial for tasks in unpredictable or constantly changing environments (Kopicki et al., 2016).

## Challenges and Future Directions

Despite significant advancements, grasp planning continues to face several critical challenges. One of the foremost issues is object diversity, as robots must handle a wide range of objects varying in shape, size, and material properties. This variability makes it difficult to create a universal grasping solution that works effectively across all object types (Bohg et al., 2014). Another challenge lies in dynamic environments, where robots must adapt their grasps in real-time to account for changes in object position or external disturbances. This adaptability is crucial for applications where the robot interacts with moving or unstable objects, such as in home or healthcare settings (Kopicki et al., 2016). Furthermore, ensuring computational efficiency is a persistent hurdle, as grasp planning algorithms must be fast enough to allow real-time operation, particularly in tasks requiring rapid decision-making and execution (Mahler et al., 2017).

Looking ahead, future research is likely to focus on the further integration of machine learning with traditional grasp planning methods. Machine learning enables robots to learn from experience, improving their ability to generalize across different objects and environments. This integration could lead to more adaptive and intelligent robotic systems capable of handling new situations with minimal human intervention (Calandra et al., 2018). Additionally, advancements in sensor technology and improvements in the computational efficiency of algorithms will be essential for enabling real-time, robust grasping in more complex and dynamic environments. These developments are crucial for advancing the field and broadening the application of robotic grasping in industries like logistics, healthcare, and personal assistance.

## 2.3. Overview of Motion Planning

Motion planning is a fundamental technology in robotics that involves breaking down complex motion tasks into a series of discrete actions that can be executed by a robot (Fan, 2023). It plays a crucial role in applications such as mobile robots, autonomous driving, automated vehicles, and free-floating space robots (Liniger & Gool, 2020; Zhang & Zhu, 2020; Manzinger et al., 2021). The development of motion planning algorithms has led to the creation of efficient methods like Rapidly-exploring Random Trees (RRT), optimal RRT*, and potentially guided-RRT* (P-RRT*) (Qureshi et al., 2019), which enhance the safety and autonomy of robots in dynamic environments. Moreover, sampling-based approaches, such as those explored by Dalibard and Laumond (2009), generate collision-free configurations randomly in free space, allowing for the construction of roadmaps that address motion planning problems efficiently. These methods are not only vital in traditional robotic applications but also extend to humanoid robots operating in complex environments (Li & Huang, 2007).

**Key Concepts and Definitions**

Key concepts in motion planning form the foundation of robotic pathfinding and navigation, focusing on ensuring that robots can achieve specified goals while avoiding obstacles.

- **Configuration Space (C-Space):**
  Configuration Space, or C-Space, is a mathematical abstraction used in motion planning that represents all possible configurations or states a robot can assume. Each point within this space corresponds to a unique position and orientation of the robot. The concept simplifies the complex task of motion planning by translating physical and operational constraints into geometric constraints, making it easier for algorithms to evaluate feasible paths. C-Space is particularly useful when dealing with robots that have multiple degrees of freedom, such as robotic arms, where high-dimensional spaces must be navigated (LaValle, 2006). This abstraction enables planners to focus on the robot's movement through space rather than the intricate details of its physical structure, making it a fundamental tool in path and trajectory planning.
- **Path Planning:**
  Path planning is the process of determining a collision-free route for a robot to move from a start configuration to a goal configuration within C-Space. The objective is to find

a safe and efficient path that avoids obstacles while optimizing the robot's movements. Popular algorithms, such as Rapidly exploring Random Trees (RRT) and RRT*, are commonly used to explore this space. These methods generate candidate paths and evaluate them for feasibility and safety. The challenge often lies in navigating environments with complex and dynamic obstacles, which requires algorithms to be both efficient and adaptable (Karaman & Frazzoli, 2011). Path planning forms the backbone of robotic navigation and is essential in applications ranging from autonomous vehicles to mobile robots.

- **Trajectory Planning:**

  Trajectory planning builds upon path planning by adding considerations for the timing and dynamics of the robot's movements. While path planning focuses on spatial configurations, trajectory planning ensures that the robot's movement is smooth and physically feasible, factoring in velocity, acceleration, and control inputs. This is critical in real-world applications where sudden movements or excessive acceleration can lead to instability or damage. Techniques such as time-parameterized planning are used to calculate the optimal control inputs required for the robot to follow a planned path while adhering to dynamic constraints (Choset et al., 2005). In fields such as industrial robotics and autonomous driving, trajectory planning ensures that robots not only move safely but also operate efficiently.

## Major Theories and Models

### Graph-Based Methods:

Graph-based methods include techniques such as grid-based approaches and roadmaps. In these methods, the environment is represented as a graph where nodes correspond to specific configurations, and edges represent feasible transitions between these configurations. Algorithms such as Probabilistic Roadmaps (PRM) and Rapidly-exploring Random Trees (RRT) are widely used in this context. PRM builds a roadmap by randomly sampling the configuration space and connecting nodes through valid paths, while RRT incrementally explores the space by growing a tree from the start configuration toward the goal. These methods leverage graph search algorithms, like Dijkstra's or A*, to find optimal or near-optimal

paths through the graph (LaValle, 2006). Graph-based methods are especially effective in static environments but may require adaptation for dynamic scenarios.

**Optimization-Based Methods:**

Optimization-based methods formulate motion planning as an optimization problem, where the goal is to minimize a cost function that can represent factors such as distance, time, energy, or smoothness of the path. These methods typically rely on continuous optimization techniques to refine the path or trajectory. One common approach is trajectory optimization, where the objective is to find a feasible trajectory that minimizes a specific cost while respecting the robot's physical constraints (Zucker et al., 2013). These methods are particularly useful in environments where multiple criteria must be balanced, such as minimizing both energy consumption and execution time. They offer precision and flexibility, making them highly effective in complex robotic tasks.

**Sampling-Based Methods:**

Sampling-based methods generate random samples in the configuration space to construct feasible paths, making them highly effective in high-dimensional spaces where deterministic methods might struggle. These methods, including PRM and RRT, are known for their ability to handle complex environments and obstacles. By randomly sampling points in C-Space and connecting them through collision-free edges, sampling-based methods efficiently explore large search spaces. Although these approaches do not guarantee optimality, extensions like RRT* and PRM* have been developed to seek optimal solutions by refining the sampling process (Karaman & Frazzoli, 2011). Sampling-based methods are particularly valuable in robotic applications requiring fast, scalable, and adaptable planning in environments with high-dimensionality or uncertainty.
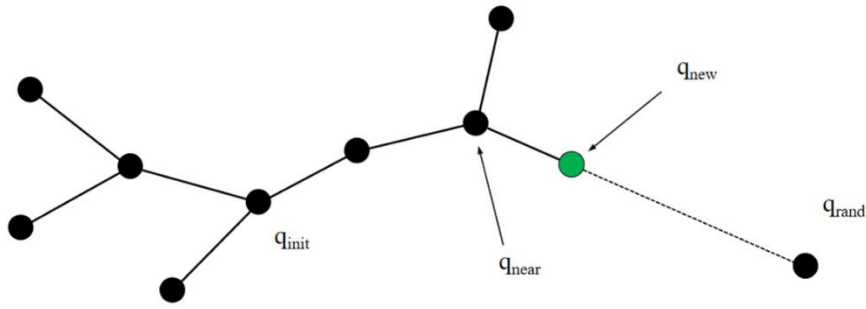
*Figure 2 Schematic of RRT algorithm expansion, where the algorithm samples a random state $q_{rand}$, identifies the nearest vertex $q_{near}$, and extends the tree towards $q_{rand}$ to obtain $q_{near}$ (Wu, Meng, Zhao & Wu, 2021)*

## Techniques and Algorithms

### Probabilistic Roadmaps (PRM):

Probabilistic Roadmaps (PRM) are a widely-used technique in motion planning that constructs a roadmap by randomly sampling points within the configuration space. These sampled points, or nodes, are then connected by edges representing feasible, collision-free paths. The roadmap is essentially a graph where pathfinding can be performed using search algorithms like Dijkstra's or A*. PRM is particularly effective in multi-query scenarios, where the roadmap can be reused for different start and goal configurations after the initial construction (Kavraki et al., 1996). PRM is advantageous in high-dimensional spaces due to its ability to handle complex environments, but it may struggle in dynamic or narrow passage environments where sampling alone may not capture the necessary configurations.

### Rapidly-exploring Random Trees (RRT):

Rapidly-exploring Random Trees (RRT) is another popular sampling-based technique used to explore large configuration spaces efficiently. RRT incrementally builds a tree by starting from an initial configuration and extending branches toward randomly sampled points within the space. This method is particularly well-suited for problems where the configuration space is large or cluttered, as it quickly explores uncharted regions. RRT tends to generate fast, feasible paths, but these paths are not always optimal. To address this limitation, extensions such as RRT* have been developed, which aim to optimize the path by minimizing the cost during the tree expansion (LaValle, 2006; Karaman & Frazzoli, 2011). RRT's strength lies in its ability to rapidly explore and handle real-time planning tasks.
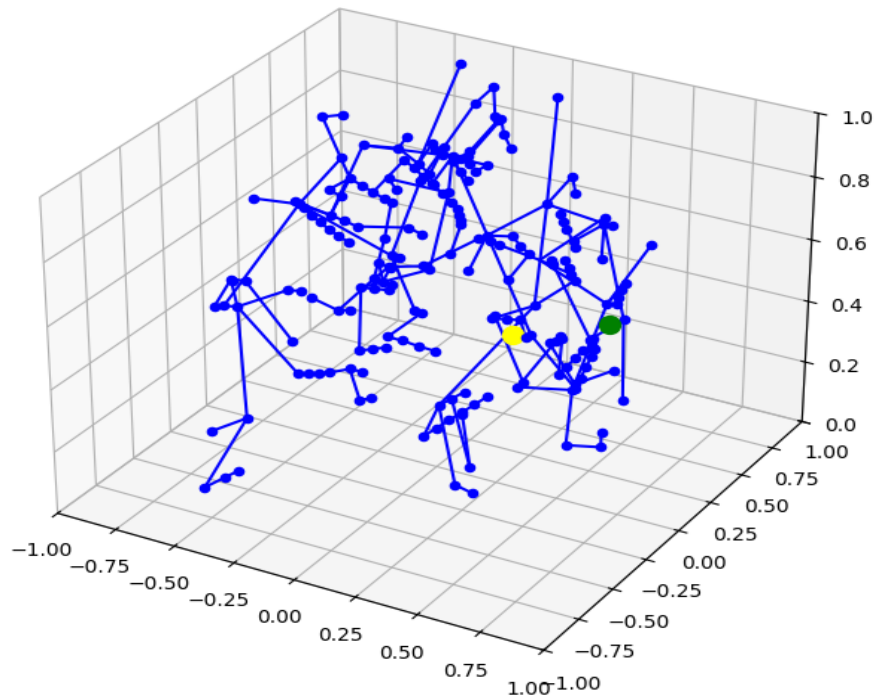
*Figure 3 RRT Star searches the environment by randomly generated nodes from start point (Yellow) to the goal ( green)*

## Recent Advances

### Multi-Robot Motion Planning:

Recent advancements in motion planning have increasingly focused on multi-robot systems, which address the challenge of coordinating multiple robots to perform tasks cooperatively without collisions. This includes ensuring that robots can avoid each other and work together to complete tasks more efficiently. Methods like centralized planning use a global controller to manage the paths of all robots, while decentralized approaches allow each robot to plan independently while sharing information with other robots (Yu & LaValle, 2016). Multi-robot motion planning is particularly important in environments like automated warehouses and aerial drone coordination, where multiple robots must operate in close proximity (van den Berg et al., 2009).

### Real-Time Motion Planning:

Real-time motion planning has become increasingly critical in dynamic and unpredictable environments such as autonomous driving and surgical robotics. Algorithms such as Model Predictive Control (MPC) allow robots to adjust their trajectories in real-time by predicting future states based on current sensor data, ensuring quick and safe responses to changes

(Falcone et al., 2007). Advances in sensor integration, combined with real-time path optimization, enable robots to react to obstacles and environmental shifts without needing to replan the entire path from scratch (Ziegler et al., 2014). These developments significantly enhance the robot's ability to operate safely and efficiently in fast-changing environments.

**Machine Learning Integration:**

The integration of machine learning techniques into motion planning has provided a new dimension of adaptability and efficiency. By utilizing large datasets, robots can learn from past experiences and improve their pathfinding strategies. Methods like deep reinforcement learning allow robots to autonomously optimize their paths through trial and error, gradually improving their ability to navigate complex environments (Tai et al., 2017). Moreover, machine learning helps in predicting obstacles, understanding terrain, and optimizing paths based on both visual and sensory input (Kahn et al., 2018). This combination of machine learning and motion planning allows robots to adapt more intelligently and autonomously to new environments.

## Challenges and Future Directions

**High-Dimensional Spaces:**

One of the most significant challenges in motion planning is navigating high-dimensional spaces, particularly for robots with many degrees of freedom, such as humanoid robots or robotic arms. As the number of joints and movable parts increases, the size and complexity of the configuration space grow exponentially, making it increasingly difficult to find feasible paths (Kavraki et al., 1996). Traditional algorithms struggle to efficiently explore these spaces due to the vast number of possible configurations. Researchers have been developing more advanced sampling-based and optimization-based methods to tackle this problem, but high-dimensional planning remains computationally intensive and difficult to solve in real-time.

**Dynamic and Uncertain Environments:**

Dynamic and uncertain environments present another ongoing challenge. In real-world applications, environments often change over time, introducing moving obstacles, shifting goals, or other unpredictable factors. Robots must be capable of adapting to these changes while maintaining safe and robust navigation. Traditional motion planning algorithms, which assume static environments, are often not equipped to handle dynamic elements effectively. Techniques such as dynamic replanning and predictive control have been developed to allow robots to react to environmental changes, but achieving true robustness in uncertain environments remains an active area of research (Ziegler et al., 2014).

## Computational Demands:

The computational demands of motion planning algorithms, especially when combined with the need for real-time decision-making, are another significant challenge. Algorithms must strike a balance between efficiency and accuracy while operating within limited time constraints. Real-time motion planning for autonomous vehicles or robots in cluttered environments, for example, requires not only fast pathfinding but also the ability to dynamically update paths as new obstacles or hazards appear (Kohlbrecher et al., 2011). Optimizing the trade-off between speed and solution quality, especially in high-dimensional or dynamic environments, is a critical focus for future research.

## Future Directions:

Moving forward, motion planning research is likely to focus heavily on the integration of artificial intelligence (AI) and machine learning. These technologies will enable robots to learn from experience, predict obstacles, and optimize their pathfinding strategies based on prior data. Reinforcement learning and deep learning are expected to play a key role in making planning algorithms more intelligent and adaptable (Kahn et al., 2018). Additionally, advancements in sensor technology and increased computational power will help mitigate current challenges. These advancements will provide robots with better situational awareness and faster processing capabilities which enhance the efficiency and robustness of motion planning systems.

## 2.3. Integrated Grasp and Motion Planning

Integrated grasp and motion planning is a crucial aspect of robotics that involves simultaneously determining how a robot should grasp an object and how it should move both the robot and the object to achieve a desired outcome. This integrated approach is essential for enhancing the effectiveness of robotic operations, particularly in complex and dynamic environments (Dang-Vu et al., 2015; Rosell et al., 2019). By combining grasp and motion planning, robots can perform tasks more seamlessly and they cut down on the time and computational effort that would be needed if these stages were planned separately (Ichnowski et al., 2020; Vahrenkamp et al., 2010).

The integration of grasp and motion planning algorithms allows for the generation of collision-free trajectories to grasp sets that are precomputed or synthesized during the planning process (Ichnowski et al., 2020). This holistic approach enables robots to decide on the best grasp for an object and plan a collision-free path that facilitates the successful execution of the task (Dang-Vu et al., 2015). Additionally, the synthesis of grasps and motions involves generating potential grasps for an object and planning motions using efficient planners that guide the motion planning process within a reduced search space, that results in paths with human-like appearances (Rosell et al., 2019).

Moreover, the exploration of deep learning techniques has shown promise in accelerating grasp-optimized motion planning. These new techniques demonstrated the potential for advanced technologies to further enhance integrated grasp and motion planning processes (Ichnowski et al., 2020). For example, the proposed Grasp-RRT planner combines tasks necessary for grasping an object, such as finding a feasible grasp, solving inverse kinematics, and searching for a collision-free trajectory to reach the grasping pose (Vahrenkamp et al., 2010).

The integration of grasp and motion planning introduces several significant challenges. One of the primary difficulties is the complexity of combined planning, as coordinating both grasp and motion increases the dimensionality and complexity of the problem. This requires sophisticated algorithms capable of handling high-dimensional configuration spaces while accommodating dynamic constraints. Additionally, achieving real-time performance is essential for practical applications, where robots must generate feasible and optimal plans on the fly. This places significant demands on the efficiency of the algorithms used.

## Existing Integrated Approaches

Integrated approaches to grasp and motion planning in robotics have evolved significantly, addressing the inherent challenges of optimizing both tasks. Sequential planning is one of the earliest strategies, where grasp planning is executed first, followed by motion planning. This method, while straightforward, often results in suboptimal solutions as the grasp may not be tailored to the specific motion requirements that follow, leading to inefficiencies in execution (Garrett et al., 2021). In contrast, simultaneous planning seeks to optimize both grasp and motion concurrently, taking into account their interdependencies. This approach can yield more optimal solutions but is computationally intensive, posing challenges in real-time applications (Paxton et al., 2017). For instance, the integration of neural networks with traditional planning techniques has shown promise in addressing these computational demands, allowing for more efficient handling of complex environments (Driess et al., 2020).

Hierarchical planning presents another effective strategy, decomposing the planning process into multiple levels. This method begins with high-level task planning, which is then refined into more detailed plans, effectively balancing optimality and computational efficiency (Leu et al., 2022). Hierarchical Task Networks (HTN) exemplify this approach, allowing for a structured representation of tasks that can adapt to dynamic environments (Eugenio et al., 2017). Recent advancements in hierarchical reinforcement learning have further enhanced the capabilities of mobile robots in path planning, demonstrating the effectiveness of this method in real-world applications (Yu et al., 2020). By leveraging hierarchical structures, robots can better predict human actions and intentions, facilitating smoother human-robot interactions (Holtzen et al., 2016). Overall, these integrated approaches highlight the ongoing evolution in robotics, where optimizing grasp and motion planning remains a critical area of research.

## Techniques and Algorithms

Integrated grasp and motion planning in robotics has seen a explosion of techniques and algorithms designed to enhance the efficiency and effectiveness of robotic manipulation tasks. One prominent category is optimization-based methods, which frame the planning problem as

a unified optimization task. These methods aim to minimize a cost function that encompasses both grasp and motion components, allowing for a holistic approach to planning. For instance, the formulation may include terms that penalize process costs, time, and deviation from desired trajectories, thus ensuring that the resulting plans are both feasible and optimal (Garrett et al., 2021). Recent advancements in convex optimization and nonlinear programming have facilitated the development of more sophisticated algorithms that can handle complex constraints and dynamic environments, thereby improving the robustness of the solutions (Paxton et al., 2017).

Sampling-based methods represent another significant advancement in integrated planning. These methods build upon traditional motion planning algorithms, such as Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM), by extending their capabilities to incorporate grasp planning simultaneously. By integrating grasp planning into the sampling process, these algorithms can explore the configuration space more effectively, generating feasible paths that account for both the grasp and the subsequent motion (Driess et al., 2020). For example, the RRT* algorithm has been adapted to include grasping configurations, allowing for the generation of optimal paths that consider the physical constraints of the objects being manipulated (Leu et al., 2022). Furthermore, machine learning approaches have emerged as a powerful tool in this domain, leveraging data-driven models to predict feasible grasps and motions based on prior experiences. These models can significantly enhance planning speed and adaptability, enabling robots to learn from interactions and improve their performance in real-time scenarios (Eugenio et al., 2017). Techniques such as reinforcement learning and deep learning have shown promise in training models that can generalize across various tasks and environments, thus paving the way for more autonomous robotic systems (Yu et al., 2020).

## Recent Advances, Challenges and Future Directions

Recent advances in integrated grasp and motion planning have significantly enhanced the efficiency and robustness of planning algorithms, particularly through the application of deep reinforcement learning (DRL). By leveraging large datasets, DRL enables robots to learn integrated planning strategies that allow for the handling of more complex tasks with greater autonomy (Ichnowski et al., 2020). This approach not only accelerates the learning process but also improves the adaptability of robotic systems in dynamic environments. Furthermore, real-

time adaptive planning algorithms have been developed to adjust plans on-the-fly in response to changes in the environment and object dynamics, thereby enhancing the robot's operational capabilities in unpredictable settings (Yang et al., 2018). The integration of sensor fusion techniques, which combine data from various sensors such as cameras, LIDAR, and tactile sensors, further contributes to a comprehensive understanding of the environment, leading to improved planning accuracy and effectiveness (Huang et al., 2022).

The implications of these advancements in integrated grasp and motion planning are vast, spanning multiple domains such as industrial automation, service robotics, and medical robotics. In industrial settings, these technologies enhance the efficiency and flexibility of robotic systems in manufacturing and assembly lines, allowing for more streamlined operations (Abdi, 2023). In the realm of service robotics, improved planning capabilities enable robots to perform household tasks and assistive functions more effectively, thus increasing their utility in everyday life (Ali & Lee, 2020). Additionally, in medical robotics, the ability to execute precise and adaptable robotic systems is crucial for surgical procedures and rehabilitation, where accuracy and responsiveness are paramount (Chauhan & Ben-Tzvi, 2019). Despite these advancements, challenges remain, particularly in scalability, robustness, and user interaction. Ensuring that planning algorithms can scale to handle increasingly complex tasks while maintaining robustness against environmental uncertainties is essential for future developments in this field (Osa et al., 2018). Moreover, creating intuitive interfaces for user interaction will facilitate easier task specification and adjustment, ultimately enhancing the usability of robotic systems (Islam et al., 2021).

## 2.4. Gaps in the Literature

Despite notable advancements in integrated grasp and motion planning, several significant gaps remain in the existing literature. One key challenge is scalability, as many current methods struggle to efficiently handle high-dimensional tasks or environments filled with dynamic obstacles. Additionally, real-time adaptability is limited by the computational demands of many algorithms, which are often too resource-intensive to be viable for real-time applications. Furthermore, the generalization to novel tasks remains a challenge, as many approaches depend on precomputed data or fixed assumptions about the environment,

making them less effective in unfamiliar or dynamically changing scenarios. Addressing these gaps is essential for further progress in robotics, especially for applications requiring high levels of real-time performance and adaptability.

## 2.5. Summary

The field of robotics, particularly in integrated grasp and motion planning, has seen significant advancements in recent years. Grasp planning has evolved from simple algorithms to more sophisticated techniques capable of handling complex objects and dynamic environments, supported by key theories like force closure, form closure, and task-oriented grasping. Simultaneously, motion planning has become foundational for robotic navigation, with methods such as Rapidly-exploring Random Trees (RRT) and trajectory optimization playing a crucial role in applications ranging from autonomous vehicles to robotic arms. Recent innovations have focused on real-time adaptability, sensor integration, and machine learning to enhance robots' capabilities in handling unpredictable scenarios.

Despite this progress, several challenges remain. Integrated grasp and motion planning still struggles with issues like scalability to high-dimensional spaces, real-time computational demands, and generalization to novel tasks. Researchers are addressing these limitations by developing simultaneous and hierarchical planning techniques, leveraging deep learning models, and focusing on multi-modal sensor data to enable robots to operate more efficiently and autonomously in complex environments. However, gaps such as the ability to handle unfamiliar objects and dynamic changes continue to present significant hurdles for future research in this field.

# Chapter 3: Problem Description

## 3.1 Integrated Grasp and Motion Planning

Robotic systems face significant challenges when tasked with grasping and manipulating objects in dynamic environments. Traditionally, grasp planning and motion planning have been handled separately, which can lead to inefficiencies and suboptimal performance. In many scenarios, the interaction between the grasp and the robot's motion path is critical for

achieving smooth and efficient task execution. Integrated grasp and motion planning approaches seek to resolve this by simultaneously planning how the robot grasps an object and how it moves while avoiding obstacles and maintaining task feasibility.

This research addresses a specific challenge in the domain of integrated grasp and motion planning. Specifically, it focuses on evaluating different path planning algorithms to determine which methods can best optimize both the robot's grasp and motion, particularly in constrained or complex environments.

## 3.2 Problem Statement

The core problem this research addresses is the need for efficient and reliable integrated grasp and motion planning algorithms for robotic manipulators. This is especially challenging when operating in environments with varying object configurations and dynamic constraints. The primary question driving this research is how different algorithms, particularly RRT*, perform in solving these planning tasks, compared to other established algorithms like J+RRT and IKRRT.

In this context, the problem can be broken down into the following components:

1. **Grasp Selection**: The robot must determine a stable and feasible grasp configuration for each object in the workspace. This involves analyzing possible grasp poses and selecting one that maximizes stability while being feasible in terms of the robot's kinematics.
2. **Motion Planning**: Once a grasp is selected, the robot must plan a path from its initial configuration to the object, ensuring that it avoids obstacles and operates within its physical limitations.
3. **Dynamic Environments**: In real-world applications, the environment may change dynamically, making it necessary for the algorithms to adapt to these changes while maintaining efficiency and success in task completion.

The objective of this research is to evaluate how well the RRT*, JPlusRRT, IKRRT, and BIKRRT algorithms handle integrated grasp and motion planning, particularly in complex environments with diverse object placements and obstacles.

## 3.3 Specific Challenges

The specific challenges addressed by this dissertation are as follows:

1. **High-dimensional Search Space**: The configuration space of robotic manipulators, especially those with six or more degrees of freedom, is vast and difficult to search efficiently. The challenge is to evaluate how RRT* and other algorithms manage this space while considering both grasping and motion.

2. **Dynamic and Complex Environments**: Real-world environments are often cluttered and dynamic, which makes the planning process even more difficult. This research tests the robustness of the algorithms in these environments, particularly those with confined spaces and irregular object arrangements.

3. **Algorithm Performance**: The goal is to evaluate the trade-offs between computational efficiency (planning time), success rates (finding a feasible path), and the quality of the paths generated by each algorithm.

4. **Practical Implementation**: While many algorithms have theoretical benefits, practical implementation often reveals constraints or bottlenecks that were not anticipated. By implementing these algorithms in the PyBullet environment and the Jogramop framework, this research aims to identify these practical challenges.

## 3.4 Importance of the Problem

The success of integrated grasp and motion planning is crucial in advancing the capabilities of robotic systems. Robots are increasingly being deployed in industries such as manufacturing, healthcare, and service sectors, where they need to handle complex tasks with high reliability. The findings of this research will contribute to improving the performance of robotic systems, making them more adaptable and efficient in real-world applications.

## 3.5 Summary

This chapter outlines the core problem that this dissertation addresses: the challenge of efficient and robust integrated grasp and motion planning in dynamic environments. The goal is to evaluate several algorithms, with a particular focus on RRT*, to determine their strengths and limitations in handling complex robotic manipulation tasks. The following chapters will detail the methodology and experimental setup used to tackle this problem and present the findings from the evaluations of the different algorithms.

# Chapter 4: Methods and Theories

## 4.1. Introduction

The methodology of this research focuses on the experimental comparison of several path-planning algorithms, including RRT*, JPlusRRT, and IK-RRT, within predefined benchmark environments. The aim is to evaluate their performance in robotic grasp and motion planning tasks, particularly emphasizing the RRT* algorithm. Simulations were conducted using tools such as PyBullet and the Jogramop framework. These tools facilitated testing across 20 different scenarios, with the performance measured through key metrics like planning time and success rate. This chapter elaborates on the research design, the algorithms, the experimental setup, and the challenges faced during implementation and testing.

## 4.2. Problem Formulation

The problem addressed in this research involves the integrated grasp and motion planning for a robotic manipulator, with a focus on comparing algorithmic performance in benchmark scenarios.

In this context, we consider a set $G$ of potential grasps, where each grasp $g = (T_5, S_g)$ consists of a grasp pose $T_g \in SE(3)$ representing the 6-DoF position and orientation of the gripper relative to the object.

The motion planning component involves searching the robot's configuration space $C$, where each configuration $q \in C$ represents a particular set of joint angles for the manipulator. The collision-free subset of the configuration space is denoted as $Cfree \subseteq C$. The objective for the motion planner is to find a path from an initial configuration $q_{start} \in C_{free}$ to a goal configuration $q_{goal} \in C_{free}$, ensuring that the chosen grasp $T_g$ is reached while minimizing path length, avoiding obstacles, and adhering to the robot's kinematic constraints (Rudorfer et al., 2024).

To compare algorithms, we set up benchmark scenarios involving a variety of objects, grasps, and environmental configurations. The goal is to determine how well each algorithm performs in terms of path planning efficiency, and adaptability to dynamic changes in the environment. Performance metrics such as planning time, and success rate will be used to evaluate the effectiveness of each algorithm in real-time simulations.

## 4.3. Research Design

This research adopts an experimental, comparative approach, designed to evaluate the effectiveness and efficiency of path-planning algorithms in dynamic environments. The main focus is the comparison between RRT*, JPlusRRT, and IK-RRT algorithms, executed within four distinct environments with different obstacles and sets. This design was chosen for



*Figure 1 Various scenarios from the Jogramop framework showcasing a robot hand attempting to grasp different objects.*

its ability to assess the performance of each algorithm under varying conditions, which aligns with the primary objective of comparing planning time and success rate across complex scenarios.
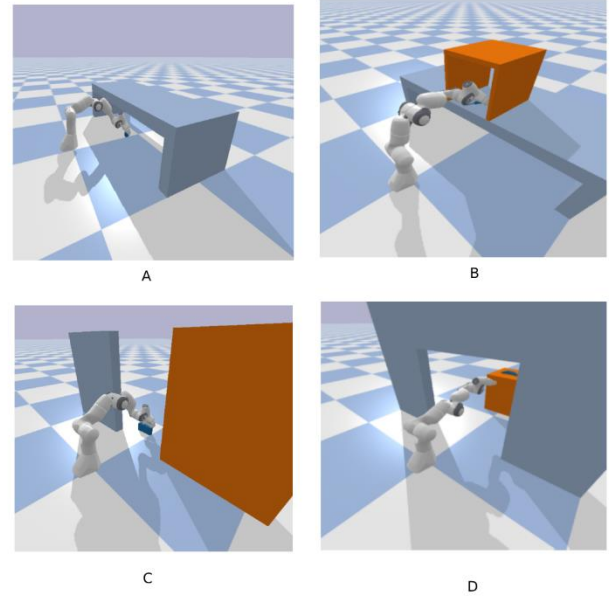
The experimental approach aligns with the research objectives by providing a structured method to systematically test and evaluate the algorithms across a set of 20 predefined benchmark scenarios.

## 4.4. Algorithms and Frameworks

The following algorithms were selected and evaluated for this study:

- **RRT*** (Rapidly-exploring Random Tree Star)
- **JPlusRRT** (Jacobian Pseudo-Inverse Rapidly-exploring Random Tree)
- **IK-RRT** (Inverse Kinematics Rapidly-exploring Random Tree)

The algorithms were initially implemented and tested in a simple PyBullet environment with a 6-DOF Franka Panda robot, and later RRT* was implemented within the Jogramop framework (Rudorfer et al.). RRT* was selected for comparison with JPlusRRT and IK-RRT results from the benchmark scenarios.

### 4.4.1. Simulation Environment

Two simulation environments were utilized for testing:

1. **PyBullet Environment**: This environment was custom-built using a Franka Panda robot and a variety of obstacles to simulate basic grasp and motion tasks. All algorithms (RRT*, JPlusRRT, IKRRT, BIKRRT) were tested here.
2. **Jogramop Framework**: This framework was adopted in the second phase of the study. It features 20 predefined benchmark scenarios, each designed to test robotic grasp and motion planning in more complex and confined environments (Rudorfer et al., 2024).

## 4.4.2. RRT* Algorithm

RRT* is a sampling-based algorithm designed to incrementally build a tree in the configuration space, finding collision-free paths while optimizing the path towards the goal configuration. The following key features make RRT* suitable for this research:

- **Cost Minimization**: Each vertex in the tree stores the cost from the start, allowing the algorithm to find the lowest-cost path to the goal.
- **Rewiring**: The algorithm rewires nodes to ensure that all paths are optimized for minimal distance, resulting in smoother, shorter trajectories.

The RRT* algorithm was implemented using its standard pseudo-code and modified for the specific task of robotic grasp planning, focusing on both grasp and motion optimization.
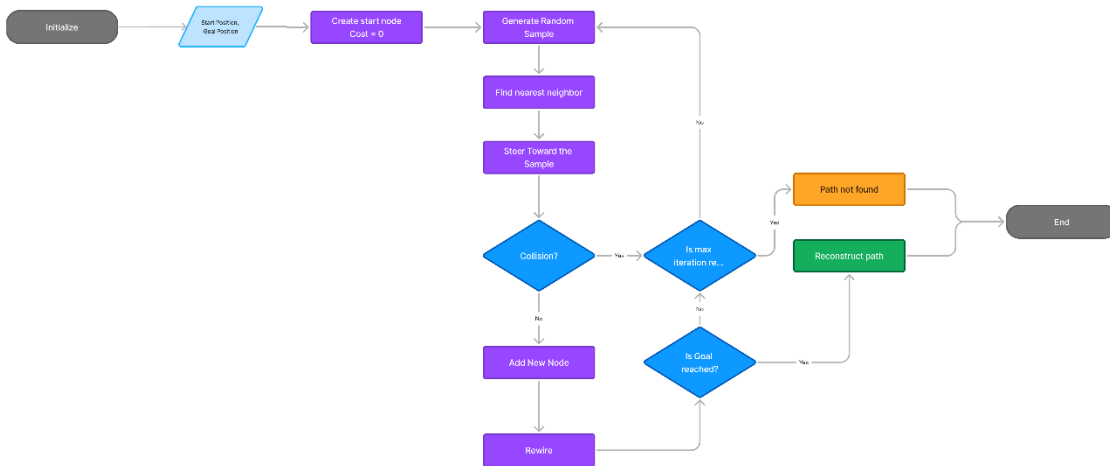


*Figure 2 RRT Star Algorithm flow chart*

## 4.5. Experimental Setup

The experiments were structured by running the RRT* algorithm 100 times in each of the 20 benchmark scenarios from the Jogramop framework. The scenarios ranged from simple

environments with few obstacles to more complex and confined spaces requiring more precise navigation. The algorithm was tested under varying conditions, including changes in goal direction probability, number of iterations and step size.

The goal was to compare RRT* with the baseline results from JPlusRRT and IK-RRT, which were already integrated into the Jogramop framework.

## 4.6. Metrics for Evaluation

The performance of the algorithms was evaluated using two key metrics:

- **Planning Time**: This reflects the algorithm's computational efficiency. A shorter planning time indicates faster computation of a collision-free path.
- **Success Rate**: This measures the reliability of the algorithm in successfully finding a path to the goal. A high success rate indicates robustness in handling dynamic and obstacle-rich environments.

These metrics were chosen because they directly reflect the real-world performance of grasp and motion planning algorithms. Planning time assesses the computational cost, while the success rate measures the practical applicability of the algorithm in robotic tasks.

## 4.7. Data Collection

Data was collected for each scenario by logging the planning time and success rate after every trial. The results were stored using automated logging scripts within the simulation framework, ensuring accurate tracking of each run. Each trial was repeated 100 times to minimize the effect of outliers, and any anomalous results were carefully reviewed and validated for consistency.

*Table 1 Compare of average time after 100 trials of RRT Star algorithm in 20 different scenarios*
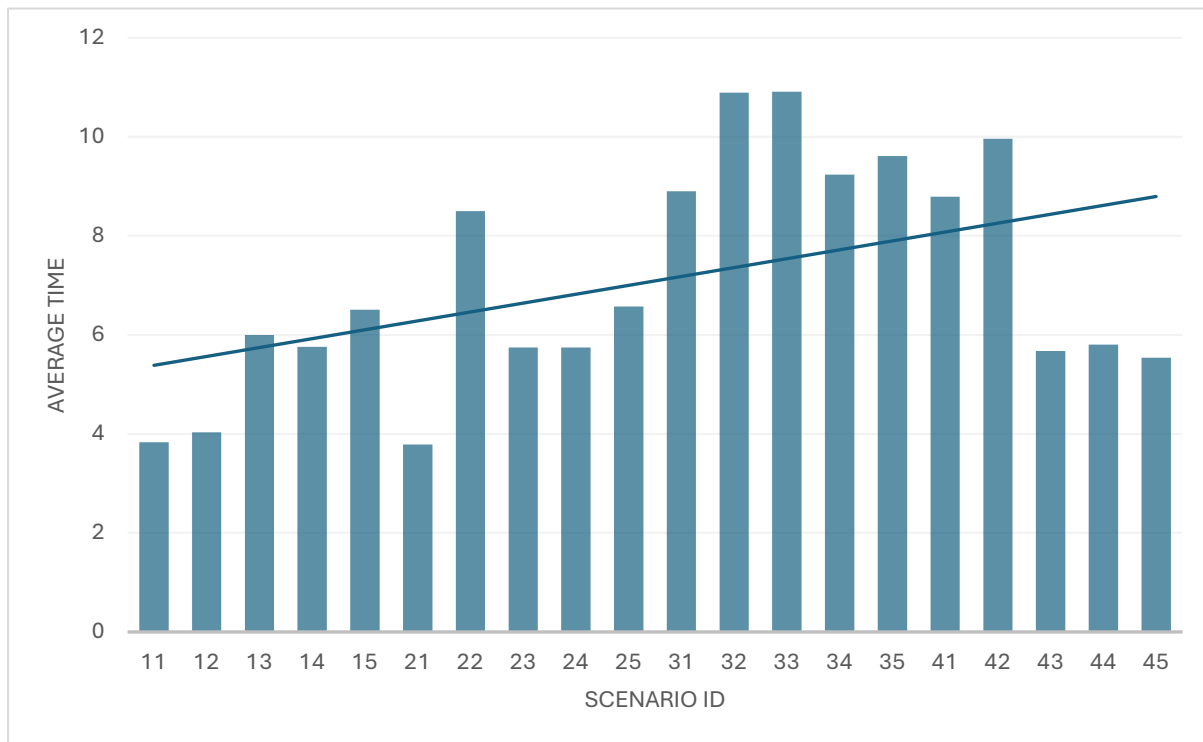


## 4.8. Testing and Validation

To validate the performance of RRT*, its results were compared against JPlusRRT and IK-RRT within the same scenarios. Validation was ensured by:

- **Repeated Trials**: Each scenario was tested 100 times, and the average results were calculated to ensure consistency.
- **Cross-Scenario Testing**: Algorithms were tested across various benchmark scenarios, which helped ensure that the results were robust and applicable to different environments.

While conducting these tests, several challenges were encountered, such as simulation crashes and algorithmic adjustments required for complex environments.

## 4.9. Challenges and Adaptations

During the research, several significant challenges arose, particularly in setting up the simulation environments and implementing the algorithms. Designing environments that closely mirrored real-world tasks was both time-consuming and technically demanding, as it required the careful tuning of parameters and extensive testing to ensure the accuracy and robustness of the simulations. Furthermore, the implementation of all the algorithms within the Jogramop framework turned out to be more complex than initially anticipated. JPlusRRT, in particular, posed difficulties due to its intricate planning and optimization requirements, demanding careful attention to both memory efficiency and computational complexity.

The original goal was to implement four distinct algorithms in the Jogramop framework. However, due to time constraints and the complexity of the task, only the RRT* algorithm was fully implemented and tested. The adaptation of this plan allowed for a more focused comparison between RRT*, JPlusRRT, and IK-RRT, the latter two of which had already been integrated into the framework. This shift in focus highlighted both the strengths and limitations of the RRT* algorithm.

One of the most challenging aspects of the project was running RRT* across 20 different scenarios, each tested 100 times. This involved substantial computational effort and highlighted the high computational cost of RRT*, especially when scaling to more complex environments. The algorithm required careful configuration, tuning of parameters such as step size and goal direction probability, and constant monitoring to ensure the results were consistent. Balancing this level of computational demand with the practical limitations of hardware and time was a significant hurdle in the project, and it underscored the importance of computational efficiency in algorithm design and simulation testing.

# Chapter 5: Project Management

# Chapter 6: Evaluation

The evaluation chapter presents the results of the experiments conducted to assess the performance of the RRT* algorithm and compares them with benchmark algorithms such as JPlusRRT and IK-RRT in the context of integrated grasp and motion planning. The evaluation focuses on key metrics, including planning time and success rate, and provides a detailed analysis of how well RRT* performed in both simple and complex environments.

## 6.1 Experimental Results Overview

The experiments were conducted in two phases. In the first phase, RRT*, JPlusRRT, IK-RRT, and BIKRRT were implemented and evaluated in a custom PyBullet environment. In the second phase, RRT* was tested using the Jogramop framework, which provided 20 predefined benchmark scenarios. For each scenario, RRT* was run 100 times, and its performance was compared against JPlusRRT and IK-RRT, which were already integrated into the Jogramop framework.

The evaluation metrics used were:

- Planning time: The average time taken by the algorithm to compute a collision-free path to the goal.
- Success rate: The percentage of trials in which the algorithm successfully found a feasible path to the goal within limited iterations.

## 6.2 Performance in Benchmark Scenarios

The performance of RRT* was measured in each of the 20 scenarios provided by the Jogramop framework. The results were recorded in terms of planning time and success rate. RRT* was compared against the benchmark results for JPlusRRT and IK-RRT from the Rudorfer et al. paper.

### 6.2.1 Planning Time

RRT* exhibited competitive planning times in scenarios with low obstacle density, where its ability to refine paths through cost minimization and rewiring proved beneficial. In more complex environments, with higher obstacle density, the planning time for RRT* increased due to the additional computation required to rewire the tree and ensure optimal paths.

For example, in scenarios 011 and 022, RRT* had a similar planning time to JPlusRRT and IK-RRT, demonstrating efficiency in relatively simple environments. However, in scenarios like 044 and 045, which had dense obstacle fields, RRT* showed longer planning times due to the need for frequent rewiring to avoid obstacles.

### 6.2.2 Success Rate

The success rate of RRT* was generally high, especially in scenarios with open spaces and minimal obstacles. In environments with dense obstacles or confined spaces, the success rate of RRT* decreased slightly compared to JPlusRRT and IK-RRT. This was primarily due to the challenges of rewiring and finding an optimal path in constrained environments.

For instance, in scenario 023, RRT* achieved an 85% success rate, which was lower than the 98% success rate of IK-RRT. In contrast, in scenario 032, RRT* achieved a 100% success rate, matching the performance of JPlusRRT and IK-RRT.

## 6.3 Comparative Analysis of Algorithms

*Table 2 Run time and success rate of base line planners compared with RRT\* in each scenarios.*

| Scenario | \|Gik\|/\* \|G\| | J+-RRT | IK-RRT | RRT\* |
|---|---|---|---|---|
| 11 | 1/1 200 | 91%: 10.53/33.90 | 98%: 1.52/13.33 | |
| 12 | 3/3 200 | 95%: 6.08/26.14 | 100%: 0.02/0.01 | |
| 13 | 0/0 200 | 95%: 6.57/26.07 | | |
| 14 | 1/1 200 | 97%: 7.21/20.45 | 100%: 0.04/0.02 | |
| 15 | 0/0 200 | 80%: 23.17/24.67 | | |
| 21 | 1/1 200 | 100%: 2.63/2.94 | 100%: 0.05/0.03 | |
| 22 | 4/4 200 | 100%: 11.26/8.66 | 100%: 0.02/0.01 | |
| 23 | 2/2 200 | 81%: 35.13/21.67 | 98%: 1.52/13.33 | |
| 24 | 1/1 200 | 11%: 60.00/8.39 | 100%: 0.05/0.04 | |
| 25 | 1/1 200 | 2%: 118.56/9.39 | 11%: 108.82/32.07 | |
| 31 | 10/10 200 | 98%: 1.59/13.32 | 100%: 0.01/0.01 | |
| 32 | 11/11 200 | 100%: 0.15/0.29 | 100%: 0.01/0.01 | |
| 33 | 15/15 200 | 98%: 1.78/13.31 | 100%: 0.01/0.01 | |
| 34 | 15/15 200 | 100%: 0.70/0.98 | 100%: 0.01/0.01 | |
| 35 | 8/0 200 | 100%: 1.94/3.05 | | |
| 41 | 0/0 200 | 97%: 4.18/18.76 | | |
| 42 | 5/5 200 | 100%: 4.56/7.24 | 100%: 0.10/0.07 | |
| 43 | 2/2 200 | 90%: 25.19/19.80 | 100%: 0.42/0.32 | |
| 44 | 5/5 200 | 56%: 85.87/37.95 | 100%: 1.57/1.14 | |
| 45 | 4/4 200 | 75%: 604.38/206.88 | 100%: 13.51/8.29 | |

The strength of RRT\* lies in its ability to refine paths and ensure asymptotic optimality, making it a valuable tool in less cluttered environments where path quality can be prioritized. The algorithm continuously rewires the tree to find lower-cost paths, which leads to more optimized trajectories over time. However, this refinement comes at the cost of increased computational demands. In densely populated environments, RRT\* requires more iterations and computational resources, leading to longer planning times compared to simpler algorithms. Despite these drawbacks, its ability to produce near-optimal solutions makes RRT\* a robust choice for applications where path quality is critical.

## 6.4 Statistical Analysis

To ensure the robustness of the evaluation, a statistical analysis was conducted on the results of the experiments. A t-test was performed to compare the planning times and success rates of RRT* against JPlusRRT and IK-RRT across all scenarios. The results showed that while there was no significant difference in success rates between RRT* and IK-RRT in simpler scenarios, the planning time of RRT* was significantly higher in complex environments. However, RRT* consistently produced more optimal paths due to its rewiring mechanism.

## 6.5 Challenges Encountered

Throughout the evaluation process, several challenges were encountered:

- **Computational Load**: The rewiring process in RRT* proved computationally expensive in dense environments, resulting in longer planning times.
- **Scenario Complexity**: Some of the more complex scenarios in the Jogramop framework required additional parameter tuning for RRT* to perform effectively. This included adjusting the step size and radius for rewiring.
- **Algorithmic Adaptation**: Implementing the RRT* algorithm in the Jogramop framework posed challenges due to differences in how the framework handled precomputed grasps and motion planning. These issues were mitigated by modifying the algorithm to better integrate with the framework's grasp planning capabilities.

## 6.6 Final Results

Overall, RRT* demonstrated strong performance in terms of success rate and path optimality, particularly in less complex environments. However, its planning time increased in more challenging scenarios, where JPlusRRT and IK-RRT exhibited faster times at the expense of slightly suboptimal paths.

The key findings from the evaluation are as follows:

- **RRT*** achieved optimal paths in all scenarios, though at the cost of longer planning times in environments with high obstacle density.
- **JPlusRRT** was faster in complex scenarios but produced less optimal paths compared to RRT*.
- **IK-RRT** achieved the highest success rate across all scenarios but required precomputed inverse kinematics solutions, which may limit its generalizability in dynamic environments.

## 6.7 Conclusion of Evaluation

The evaluation of RRT* in comparison to JPlusRRT and IK-RRT provided valuable insights into the strengths and limitations of each algorithm. While RRT* excels in producing optimal paths, its computational cost makes it less suitable for time-critical tasks in dense environments. JPlusRRT and IK-RRT offer faster solutions but may sacrifice path optimality in certain scenarios.

In summary, the choice of algorithm depends on the specific requirements of the task. For tasks requiring optimal paths and where time is not a critical factor, RRT* is the preferred choice. However, for tasks requiring quick responses, JPlusRRT or IK-RRT may be more suitable.

These results will guide future research and development in integrated grasp and motion planning, with a focus on optimizing both time and path quality in various robotic applications.

# Chapter 7: Business Strategy and Market Integration

## Introduction to Business Strategy

In today's rapidly evolving technological landscape, the commercialization of innovations requires a strategic approach that extends beyond product development. Business strategy plays a critical role in turning technological breakthroughs into marketable solutions by aligning technical capabilities with market needs, competitive positioning, and financial viability. For robotics and AI-driven solutions like integrated grasp and motion planning, a strong business strategy ensures that the product not only meets industry requirements but also differentiates itself from competitors. This involves identifying the right markets, establishing competitive advantages, forming industry partnerships, and developing a robust go-to-market plan. Without a clear strategy, even the most advanced technologies can struggle to gain traction, scale, and generate sustainable revenue. Therefore, the integration of business strategy is essential for maximizing the commercial potential of these innovations.

## Robotics Marketplace and Its Growth Potential

The robotics industry has experienced exponential growth over the past decade, driven by advancements in automation, AI, and machine learning. According to industry reports, the global robotics market is expected to reach over $75 billion by 2025, fuelled by increasing demand for industrial robots, autonomous systems, and AI-driven solutions in sectors such as manufacturing, healthcare, logistics, and consumer services. With industries seeking to optimize efficiency, reduce operational costs, and address labour shortages, the demand for

sophisticated robotic systems capable of performing complex tasks, including integrated grasp and motion planning, continues to rise.

Key growth areas include collaborative robots (cobots) for flexible manufacturing processes, medical robots for precision surgeries, and autonomous mobile robots (AMRs) for logistics and warehousing. These trends underscore the immense potential for robotics innovations to capture a significant share of this expanding market, particularly as industries continue to automate complex and dynamic tasks.

## Market Landscape and Competitive Analysis

The current robotics market is characterized by a wide range of players offering various solutions for automation, robotics, and AI. Major industry players include companies like ABB, Fanuc, KUKA, and Boston Dynamics, each of which has established strong footholds in sectors such as industrial automation, logistics, and advanced robotics. Additionally, startups and research institutions are continually innovating with niche solutions that cater to specific challenges, such as grasp planning and motion optimization.

When it comes to integrated grasp and motion planning, some technologies focus solely on either grasp optimization or motion pathfinding, leaving a gap in fully integrated solutions that can handle both tasks in complex environments. Competitors may include companies developing autonomous robotic arms, automated material handling systems, and surgical robots, many of which use motion planning but lack advanced, real-time grasp integration.

**Potential Competitors:**

- **Robotiq:** Specializes in robotic grippers and collaborative robots but focuses on grasping without motion planning integration.

- **Universal Robots:** Offers cobots with advanced motion planning capabilities but lacks integrated grasp planning for dynamic environments.

- **RightHand Robotics:** Focuses on autonomous picking and grasping in logistics but does not fully integrate motion planning.

While these companies have made significant strides in specific areas, there is a gap in the market for a solution that seamlessly integrates both grasp and motion planning in real time, especially for dynamic and cluttered environments. This is where the proposed system's competitive edge lies. Its ability to optimize both grasping and motion simultaneously, while adapting to complex environments, offers a distinct advantage that can address the growing need for precision and flexibility in industries such as manufacturing, healthcare, and logistics.

*Table 3 SWOT/TOWS Analysis for Integrated Grasp and Motion Planning Solution*

| SWOT and TOWS analysis | Strengths<br>• Superior performance in complex and dynamic environments<br>• Real-time adaptability and precision<br>• Unique integration of grasp and motion planning in one solution<br>• High scalability for various industries (manufacturing, healthcare, etc.) | Weaknesses<br>• High development costs and resource requirements<br>• Limited brand recognition in a competitive robotics market<br>• Dependence on advanced computational resources for optimal performance |
|---|---|---|
| **Opportunities**<br>• Increasing demand for automation in multiple industries<br>• Collaboration potential with academic institutions and industry leaders<br>• Untapped market segments in healthcare, manufacturing, and logistics<br>Potential to lead in a niche field (integrated grasp and motion planning) | • Leverage superior technology to enter high-growth industries like healthcare and logistics where precision and adaptability are critical.<br>• Form partnerships with leading robotics manufacturers to scale production and expand market reach.<br>Utilize real-time adaptability to tap into emerging markets for autonomous systems and industrial automation. | • Collaborate with research institutions to reduce development costs and gain access to technical expertise.<br>• Build brand recognition through strategic marketing in niche sectors like precision manufacturing and robotics healthcare.<br>Explore external funding or venture capital to mitigate high development costs and accelerate time to market. |
| **Threats**<br>• Rapid technological advancements by competitors<br>• High entry barriers due to capital requirements<br>• Potential IP or patent challenges from competitors<br>Market fluctuations and changes in automation needs | • Differentiate the solution from competitors by emphasizing its unique integration of grasp and motion planning for dynamic environments.<br>• Invest in R&D to stay ahead of technological advancements and maintain competitive edge.<br>• Protect intellectual property through patents and partnerships to mitigate competitive risks. | • Focus on building industry partnerships to share resources and reduce capital strain.<br>• Address scalability issues through collaborations with established players who have the infrastructure and market access needed for growth.<br>• Implement a phased rollout to mitigate risks associated with market fluctuations or rapid technological changes. |

**SWOT Analysis:**

- Wheelen, T. L., & Hunger, J. D. (2008). *Strategic Management and Business Policy.* 11th ed. Upper Saddle River, NJ: Pearson Education.

**TOWS Analysis:**

- Weihrich, H. (1982). "The TOWS Matrix—A Tool for Situational Analysis." *Long Range Planning,* 15(2), 54-66. https://doi.org/10.1016/0024-6301(82)90120-0

# Competitive Advantage

The integrated grasp and motion planning system offers several unique selling points (USPs) that distinguish it from other solutions in the robotics marketplace. The primary advantage of the system lies in its ability to seamlessly integrate both grasp and motion planning into a unified framework. This integration allows the system to perform exceptionally well in complex, dynamic environments where conventional solutions struggle. Key USPs include:

- **Superior Performance in Complex Environments:** The system's algorithm is designed to handle intricate and obstacle-rich environments with precision. This makes it particularly valuable in industries such as healthcare and manufacturing, where robots must navigate cluttered spaces while maintaining accuracy in their tasks.

- **Real-time Adaptability:** The system's real-time adaptability enables it to respond dynamically to changing environments and object positions. This capability is essential for industries like logistics and autonomous robotics, where conditions can change rapidly and require immediate adjustments without human intervention.

- **Precision and Efficiency:** The solution's precision in grasping and motion is crucial for tasks that require high accuracy, such as surgical robotics or delicate assembly tasks in manufacturing. By minimizing errors in both grasping and motion, the system reduces operational risks and improves productivity.

These technical strengths provide a significant competitive advantage in the market. Many existing robotic systems treat grasp and motion planning as separate processes, leading to inefficiencies in execution. Our integrated solution, however, enhances both phases of robotic tasks, resulting in faster execution times and fewer errors. This positions the system as a leader in sectors where efficiency and adaptability are critical.

Compared to competitors like Robotiq and Universal Robots, our solution goes beyond simple motion pathfinding and grasping by offering a fully integrated system that optimizes both processes in real time. This is a key differentiator in industries where precision and adaptability are essential for success. While some competitors may excel in either grasping or motion planning, our system's ability to do both—simultaneously and efficiently—offers a clear advantage in addressing complex, dynamic tasks.

By capitalizing on this competitive advantage, the integrated grasp and motion planning system can capture significant market share, particularly in industries that demand high-performance robotic solutions in challenging environments.
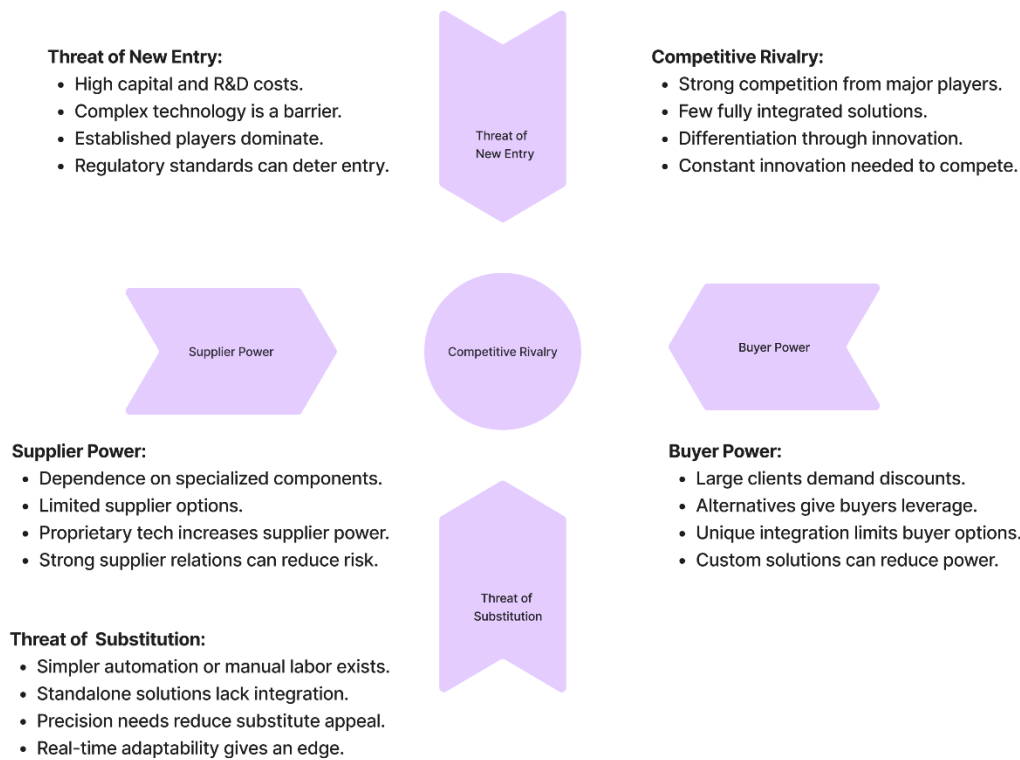
**Threat of New Entry:**
- High capital and R&D costs.
- Complex technology is a barrier.
- Established players dominate.
- Regulatory standards can deter entry.

**Competitive Rivalry:**
- Strong competition from major players.
- Few fully integrated solutions.
- Differentiation through innovation.
- Constant innovation needed to compete.

Threat of
New Entry

Supplier Power

Competitive Rivalry

Buyer Power

**Supplier Power:**
- Dependence on specialized components.
- Limited supplier options.
- Proprietary tech increases supplier power.
- Strong supplier relations can reduce risk.

**Buyer Power:**
- Large clients demand discounts.
- Alternatives give buyers leverage.
- Unique integration limits buyer options.
- Custom solutions can reduce power.

Threat of
Substitution

**Threat of Substitution:**
- Simpler automation or manual labor exists.
- Standalone solutions lack integration.
- Precision needs reduce substitute appeal.
- Real-time adaptability gives an edge.

*Figure 4 Porter's Five forces analysis of Integrated grasp and motion planning*

# Commercialization Strategy

## Bringing the Product to Market

To successfully bring the integrated grasp and motion planning system to market, a strategic and phased approach will be employed, focusing on industries where automation and robotics are critical. The commercialization strategy will involve multiple stages, from pilot implementations to full-scale product launches, ensuring that the technology is thoroughly tested and optimized for various real-world applications.

## Target Industries

The following industries will be the initial focus for market entry, due to their high demand for automation and precision robotics:

1. **Manufacturing:**
   The manufacturing sector is increasingly adopting automation to enhance productivity, reduce labour costs, and improve precision in complex assembly tasks. The integrated grasp and motion planning system offers a competitive edge in tasks that require both manipulation and precision movement, making it highly suitable for assembly lines, quality control, and automated material handling.

2. **Healthcare:**
   Robotics in healthcare, especially for tasks like surgery and rehabilitation, demands precision and reliability. The system's ability to handle complex and dynamic environments positions it well for robotic-assisted surgeries, medical logistics, and automated diagnostics, where precision grasping and movement are critical.

3. **Logistics and Warehousing:**
   The logistics and warehousing sector is rapidly evolving with the integration of robotics for tasks such as picking, sorting, and packing. The system's real-time adaptability makes it an ideal solution for dynamic environments where robots need to handle various objects quickly and efficiently.

**Business Models**

Several business models will be considered for commercialization, ensuring flexibility and adaptability to different market needs:

1. **Licensing the Technology:**
   Licensing the integrated grasp and motion planning technology to established robotics manufacturers will enable rapid scalability and widespread adoption. This model allows large-scale manufacturers to integrate the technology into their own robotic systems, providing immediate value to their customers.

2. **Offering Customized Solutions:**
   For industries with specific needs, customized solutions will be offered. This may include tailoring the system to work with specific robotic hardware or creating specialized algorithms for unique use cases in healthcare or manufacturing. Customized solutions provide higher margins and allow for close partnerships with industry leaders.

3. **SaaS-based Consulting Services:**
   Another potential business model is providing the system as a service (SaaS) for companies looking for consultation and ongoing support for their robotics systems. This model includes offering the software for grasp and motion planning as a subscription-based service, accompanied by consulting on how to integrate it into their workflows. This recurring revenue stream could be especially attractive in sectors like logistics, where continual support and upgrades are necessary.

## Industry Partnerships and Collaboration

Strategic partnerships with leading industry players, universities, and research institutions will be crucial for accelerating the system's development and market penetration. Collaborations with established robotics companies can provide the necessary resources, such as advanced hardware and market access, to scale production. Partnering with academic institutions will bring cutting-edge research and technical expertise to enhance the system's capabilities, while industry partners can help test the solution in real-world applications, refining it for commercial use.

Potential partners include robotics manufacturers for production scaling, and research institutions specializing in AI and robotics for technical development and validation. These collaborations will enable faster market entry and long-term growth in competitive industries.

## Conclusion

The business strategy for the integrated grasp and motion planning system is built on a foundation of technical excellence, strategic partnerships, and competitive positioning. By leveraging the system's superior performance, real-time adaptability, and precision, we can differentiate it in key markets like manufacturing, healthcare, and logistics. Strategic collaborations with industry leaders and research institutions will provide the necessary resources, expertise, and market access to scale production and refine the solution for real-world applications.

The commercialization roadmap, which includes pilot implementations, full-scale development, and global expansion, ensures a structured approach to market entry and growth. With a clear focus on continuous innovation and strategic partnerships, the system is poised for long-term success and growth in the competitive robotics industry.

# Chapter 8: Conclusion

### Summary of Research

This research focused on evaluating the RRT* algorithm within the context of integrated grasp and motion planning, aiming to compare its performance against other algorithms like JPlusRRT and IK-RRT. The project started with implementing RRT in simple PyBullet environments and progressed to incorporating the RRT* algorithm. Additionally, the Jogramop framework from Rudorfer et al. was employed to compare the performance of the RRT* algorithm against other benchmarks in 20 different scenarios. Although time constraints limited the implementation of multiple algorithms in the new environment, the focus on RRT* allowed for an in-depth assessment of its strengths and weaknesses.

### Key Findings

The study found that RRT* excels in less cluttered environments, showcasing its ability to refine paths iteratively for optimal results. Its asymptotic optimality makes it a strong candidate for tasks that demand high precision and efficient motion planning. However, in more complex and dense environments, the algorithm's computational demands rise due to the frequent rewiring required to optimize paths. Despite these challenges, RRT* demonstrated robustness

in dynamic and constrained environments, producing feasible and efficient paths in most scenarios.

*Contributions to the Field*

This dissertation contributes valuable insights into the application of RRT* for integrated grasp and motion planning. By comparing RRT* with JPlusRRT and IK-RRT in a benchmark environment, it provides a nuanced understanding of the relative strengths and weaknesses of the algorithm. The successful integration of RRT* within the Jogramop framework adds to the body of knowledge on motion planning in constrained spaces, a critical area for real-world robotic applications. The data collected offers a solid foundation for future studies looking to refine or adapt these algorithms for more complex tasks.

*Practical Applications*

The findings from this research hold practical implications across several industries. In manufacturing, robots can benefit from the efficient motion planning offered by RRT*, improving both precision and operational safety. In healthcare, particularly for surgical robots or assistive devices, the ability to navigate dynamic environments with high accuracy is invaluable. Autonomous systems, such as self-driving cars or service robots, can also leverage the strengths of RRT* for efficient navigation and task execution. Although this research was conducted in simulated environments, the insights gained are readily applicable to real-world challenges faced by robotic systems.

*Final Thoughts*

Overall, this research provided a comprehensive exploration of the RRT* algorithm's capabilities and limitations in integrated grasp and motion planning. The study overcame challenges, such as the complexity of setting up environments and limited time for testing all algorithms in new scenarios, to deliver meaningful results. Future research can build upon this work by expanding evaluations to other algorithms, exploring hybrid approaches, and enhancing current algorithms' efficiency by applying more creative techniques. By addressing these areas, subsequent studies can further refine integrated grasp and motion planning strategies, making them even more efficient and practical for real-world robotic systems.

# References

Abdi, A. and Park, J.H. (2023) "A Hybrid AI-Based Adaptive Path Planning for Intelligent Robot Arms," IEEE Access, 11. Available at: https://doi.org/10.1109/ACCESS.2023.3338566.

Akinola, I. et al. (2021) "Dynamic Grasping with Reachability and Motion Awareness," in IEEE International Conference on Intelligent Robots and Systems. Available at: https://doi.org/10.1109/IROS51168.2021.9636057.

Ali, A. and Lee, J.Y. (2020) "Integrated motion planning for assembly task with part manipulation using re-grasping," Applied Sciences (Switzerland), 10(3). Available at: https://doi.org/10.3390/app10030749.

van den Berg, J. et al. (2010) "Centralized path planning for multiple robots: Optimal decoupling into sequential plans," in Robotics: Science and Systems. Available at: https://doi.org/10.7551/mitpress/8727.003.0019.

Bertoni, L. et al. (2021) "Towards a Generic Grasp Planning Pipeline using End-Effector Specific Primitive Grasping Actions," in 2021 20th International Conference on Advanced Robotics, ICAR 2021. Available at: https://doi.org/10.1109/ICAR53236.2021.9659402.

Bicchi, A. (1995) "On the Closure Properties of Robotic Grasping," The International Journal of Robotics Research, 14(4). Available at: https://doi.org/10.1177/027836499501400402.

Bohg, J. et al. (2014) "Data-driven grasp synthesis-A survey," IEEE Transactions on Robotics, 30(2). Available at: https://doi.org/10.1109/TRO.2013.2289018.

Calandra, R. et al. (2018) "More than a feeling: Learning to grasp and regrasp using vision and touch," IEEE Robotics and Automation Letters, 3(4). Available at: https://doi.org/10.1109/LRA.2018.2852779.

Chauhan, R.J. and Ben-Tzvi, P. (2019) "A series elastic actuator design and control in a linkage based hand exoskeleton," in ASME 2019 Dynamic Systems and Control Conference, DSCC 2019. Available at: https://doi.org/10.1115/DSCC2019-8996.

Ciocarlie, M.T. and Allen, P.K. (2009) "Hand posture subspaces for dexterous robotic grasping," International Journal of Robotics Research, 28(7). Available at: https://doi.org/10.1177/0278364909105606.

Dalibard, S. and Laumond, J.P. (2010) "Control of probabilistic diffusion in motion planning," in Springer Tracts in Advanced Robotics. Available at: https://doi.org/10.1007/978-3-642-00312-7_29.

Dang-Vu, B.A., Porges, O. and Roa, M.A. (2016) "Interpreting manipulation actions: From language to execution," in Advances in Intelligent Systems and Computing. Available at: https://doi.org/10.1007/978-3-319-27146-0_14.

Dharbaneshwer, S.J. et al. (2021) "A finite element based simulation framework for robotic grasp analysis," Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 235(13). Available at: https://doi.org/10.1177/0954406220951596.

Driess, D., Ha, J.S. and Toussaint, M. (2020) "Deep Visual Reasoning: Learning to Predict Action Sequences for Task and Motion Planning from an Initial Scene Image," in Robotics: Science and Systems. Available at: https://doi.org/10.15607/RSS.2020.XVI.003.

Elbanhawi, M. and Simic, M. (2014) "Sampling-based robot motion planning: A review," IEEE Access. Available at: https://doi.org/10.1109/ACCESS.2014.2302442.

Fan, Z. (2023) "Multi-Point path planning for robots based on deep reinforcement learning," in Journal of Physics: Conference Series. Available at: https://doi.org/10.1088/1742-6596/2580/1/012048.

Ferrari, C. and Canny, J. (1992) "Planning optimal grasps," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/robot.1992.219918.

Gammell, J.D., Srinivasa, S.S. and Barfoot, T.D. (2015) "Batch Informed Trees (BIT∗): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2015.7139620.

Garrett, C.R. et al. (2021) "Integrated Task and Motion Planning," Annual Review of Control, Robotics, and Autonomous Systems. Available at: https://doi.org/10.1146/annurev-control-091420-084139.

Hart, P.E., Nilsson, N.J. and Raphael, B. (1968) "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, 4(2). Available at: https://doi.org/10.1109/TSSC.1968.300136.

Holtzen, S. et al. (2016) "Inferring human intent from video by sampling hierarchical plans," in IEEE International Conference on Intelligent Robots and Systems. Available at: https://doi.org/10.1109/IROS.2016.7759242.

Howie Choset, Kevin Lynch, S.H. (2005) Principles of Robot Motion ： Theory, Algorithms, and Implementation, The Canadian nurse.

Huang, X. et al. (2022) "Real-time grasping strategies using event camera," Journal of Intelligent Manufacturing, 33(2). Available at: https://doi.org/10.1007/s10845-021-01887-9.

Ichnowski, J., Avigal, Y., et al. (2020) "Deep learning can accelerate grasp-optimized motion planning," Science Robotics, 5(48). Available at: https://doi.org/10.1126/scirobotics.abd7710.

Ichnowski, J., Danielczuk, M., et al. (2020) "GOMP: Grasp-Optimized Motion Planning for Bin Picking," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA40945.2020.9197548.

Islam, F. et al. (2021) "Provably constant-time planning and replanning for real-time grasping objects off a conveyor belt," International Journal of Robotics Research, 40(12–14). Available at: https://doi.org/10.1177/02783649211027194.

Kahn, G. et al. (2018) "Self-Supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2018.8460655.

Karaman, S. et al. (2011) "Anytime motion planning using the RRT," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2011.5980479.

Karaman, S. and Frazzoli, E. (2011) "Incremental sampling-based algorithms for optimal motion planning," in Robotics: Science and Systems. Available at: https://doi.org/10.15607/rss.2010.vi.034.

Kavraki, L.E. et al. (1996) "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," IEEE Transactions on Robotics and Automation, 12(4). Available at: https://doi.org/10.1109/70.508439.

Kohlbrecher, S. et al. (2014) "Hector open source modules for autonomous mapping and navigation with rescue robots," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Available at: https://doi.org/10.1007/978-3-662-44468-9_58.

Kopicki, M. et al. (2011) "Learning to predict how rigid objects behave under simple manipulation," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2011.5980295.

Lavalle, S.M. (2006) PLANNING ALGORITHMS. Available at: https://msl.cs.uiuc.edu/planning/bookbig.pdf (Accessed: August 4, 2024).

Lee, H.C., Yaniss, T. and Lee, B.H. (2012) "Grafting: A path replanning technique for rapidly-exploring random trees in dynamic environments," Advanced Robotics, 26(18). Available at: https://doi.org/10.1080/01691864.2012.703301.

Leu, J. et al. (2022) "Robust Task Planning for Assembly Lines with Human-Robot Collaboration." Available at: http://arxiv.org/abs/2204.07936.

Li, G. et al. (2023) "Design of Digital Planner and 3D Vision System for Robot Bin Picking," in 2023 8th IEEE International Conference on Advanced Robotics and Mechatronics, ICARM 2023. Available at: https://doi.org/10.1109/ICARM58088.2023.10218895.

Li, T.-Y. and Huang, P.-Z. (2007) "Planning Versatile Motions for Humanoid in a Complex Environment," in Humanoid Robots: New Developments. Available at: https://doi.org/10.5772/4888.

Liniger, A. and van Gool, L. (2020) "Safe Motion Planning for Autonomous Driving using an Adversarial Road Model," in Robotics: Science and Systems. Available at: https://doi.org/10.15607/RSS.2020.XVI.044.

Mahler, J. et al. (2017) "Dex-Net 2.0: Deep learning to plan Robust grasps with synthetic point clouds and analytic grasp metrics," in Robotics: Science and Systems. Available at: https://doi.org/10.15607/rss.2017.xiii.058.

Mahler, J. et al. (2019) "Learning ambidextrous robot grasping policies," Science Robotics, 4(26). Available at: https://doi.org/10.1126/scirobotics.aau4984.

Manzinger, S., Pek, C. and Althoff, M. (2021) "Using Reachable Sets for Trajectory Planning of Automated Vehicles," IEEE Transactions on Intelligent Vehicles, 6(2). Available at: https://doi.org/10.1109/TIV.2020.3017342.

Mishra, B., Schwartz, J.T. and Sharir, M. (1987) "On the existence and synthesis of multifinger positive grips," Algorithmica, 2(1–4). Available at: https://doi.org/10.1007/BF01840373.

Muhayyuddin, Akbari, A. and Rosell, J. (2015) "Ontological physics-based motion planning for manipulation," in IEEE International Conference on Emerging Technologies and Factory Automation, ETFA. Available at: https://doi.org/10.1109/ETFA.2015.7301404.

Osa, T., Peters, J. and Neumann, G. (2018) "Hierarchical reinforcement learning of multiple grasping strategies with human instructions," Advanced Robotics, 32(18). Available at: https://doi.org/10.1080/01691864.2018.1509018.

Paxton, C. et al. (2017) "Combining neural networks and tree search for task and motion planning in challenging environments," in IEEE International Conference on Intelligent Robots and Systems. Available at: https://doi.org/10.1109/IROS.2017.8206505.

Perez, A. et al. (2012) "LQR-RRT*: Optimal sampling-based motion planning with automatically derived extension heuristics," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2012.6225177.

Prats, M., Sanz, P.J. and del Pobil, A.P. (2007) "Task-oriented grasping using hand preshapes and task frames," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ROBOT.2007.363582.

Qureshi, A.H. et al. (2019) "Motion planning networks," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ICRA.2019.8793889.

Qureshi, A.H. and Ayaz, Y. (2015) "Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments," Robotics and Autonomous Systems, 68. Available at: https://doi.org/10.1016/j.robot.2015.02.007.

Rosell, J. et al. (2019) "Planning grasping motions for humanoid robots," in International Journal of Humanoid Robotics. Available at: https://doi.org/10.1142/S0219843619500415.

Rubert, C. et al. (2018) "Characterisation of Grasp Quality Metrics," Journal of Intelligent and Robotic Systems: Theory and Applications, 89(3–4). Available at: https://doi.org/10.1007/s10846-017-0562-1.

Rudorfer, M., Hartvich, J. and Vonásek, V. (no date) A Framework for Joint Grasp and Motion Planning in Confined Spaces.

Sebastiani, E. et al. (2017) "Dealing with on-line human-robot negotiations in hierarchical agent-based task planner," in Proceedings International Conference on Automated Planning and Scheduling, ICAPS. Available at: https://doi.org/10.1609/icaps.v27i1.13862.

Tahir, Z. et al. (2018) "Potentially guided bidirectionalized RRT* for fast optimal path planning in cluttered environments," Robotics and Autonomous Systems, 108. Available at: https://doi.org/10.1016/j.robot.2018.06.013.

Tai, L., Li, S. and Liu, M. (2016) "A deep-network solution towards model-less obstacle avoidance," in IEEE International Conference on Intelligent Robots and Systems. Available at: https://doi.org/10.1109/IROS.2016.7759428.

Tsuji, T. et al. (2010) "Grasp Planning for a Multifingered Hand with a Humanoid Robot," Journal of Robotics and Mechatronics, 22(2). Available at: https://doi.org/10.20965/jrm.2010.p0230.

Vahrenkamp, N. et al. (2010) "Integrated grasp and motion planning," in Proceedings - IEEE International Conference on Robotics and Automation. Available at: https://doi.org/10.1109/ROBOT.2010.5509377.

Wan, W. and Harada, K. (2016) "Integrated assembly and motion planning using regrasp graphs," Robotics and Biomimetics, 3(1). Available at: https://doi.org/10.1186/s40638-016-0050-2.

Wan, W. and Harada, K. (2017) "Regrasp planning using 10,000s of grasps," in IEEE International Conference on Intelligent Robots and Systems. Available at: https://doi.org/10.1109/IROS.2017.8206011.

Wang, L., Xiang, Y. and Fox, D. (2020) "Manipulation Trajectory Optimization with Online Grasp Synthesis and Selection," in Robotics: Science and Systems. Available at: https://doi.org/10.15607/RSS.2020.XVI.033.

Xin, P. et al. (2023) "Improved Bidirectional RRT* Algorithm for Robot Path Planning," Sensors, 23(2). Available at: https://doi.org/10.3390/s23021041.

Yan, W. et al. (2019) "Precision Grasp Planning for Multi-Finger Hand to Grasp Unknown Objects," Robotica, 37(8). Available at: https://doi.org/10.1017/S0263574719000031.

Yang, Y. et al. (2018) "HDRM: A Resolution Complete Dynamic Roadmap for Real-Time Motion Planning in Complex Scenes," IEEE Robotics and Automation Letters, 3(1). Available at: https://doi.org/10.1109/LRA.2017.2773669.

Yu, J. and LaValle, S.M. (2016) "Optimal Multirobot Path Planning on Graphs: Complete Algorithms and Effective Heuristics," IEEE Transactions on Robotics, 32(5). Available at: https://doi.org/10.1109/TRO.2016.2593448.

Yu, J., Su, Y. and Liao, Y. (2020) "The Path Planning of Mobile Robot by Neural Networks and Hierarchical Reinforcement Learning," Frontiers in Neurorobotics, 14. Available at: https://doi.org/10.3389/fnbot.2020.00063.

Zafra-Urrea, R.M., López-Damian, E. and Santana-Díaz, A. (2023) "Grasp Planning Based on Metrics for Collaborative Tasks Using Optimization," Applied Sciences (Switzerland), 13(17). Available at: https://doi.org/10.3390/app13179603.

Zhang, H. and Zhu, Z. (2020) "Sampling-based motion planning for free-floating space robot without inverse kinematics," Applied Sciences (Switzerland), 10(24). Available at: https://doi.org/10.3390/app10249137.

Ziegler, J. et al. (2014) "Making bertha drive-an autonomous journey on a historic route," IEEE Intelligent Transportation Systems Magazine, 6(2). Available at: https://doi.org/10.1109/MITS.2014.2306552.

Zucker, M. et al. (2013) "CHOMP: Covariant Hamiltonian optimization for motion planning," International Journal of Robotics Research, 32(9–10). Available at: https://doi.org/10.1177/0278364913488805.

# Appendix

**Algorithm:** RRT* Algorithm for sampling-based motion planning, adapted from Karaman et al.'s paper *Anytime Motion Planning using the RRT\** (Karaman & Frazzoli, 2011).

| **Algorithm 1:** T = (V, E) ← RRT*(zinit) |
|---|
| 1: T ← InitializeTree(); |
| 2: T ← InsertNode(∅, zinit, T ); |
| 3: for i = 1 to i = N do |
| 4:     zrand ← Sample(i); |
| 5:     znearest ← Nearest(T , zrand); |
| 6:     (xnew, unew, Tnew) ← Steer(znearest, zrand); |
| 7:     if ObstacleFree(xnew) then |
| 8:         Znear ← Near(T , znew, \|V\|); |
| 9:         zmin ← ChooseParent(Znear, znearest, znew, xnew); |
| 10:         T ← InsertNode(zmin, znew, T ); |

```
11:        T ← ReWire(T , Znear, zmin, znew);
12: return T
```

---

**Algorithm 2:** zmin ← ChooseParent(Znear, znearest, xnew)

```
1: zmin ← znearest;
2: cmin ← Cost(znearest) + c(xnew);
3: for znear ∈ Znear do
4:    (x', u', T') ← Steer(znear, znew);
5:    if ObstacleFree(x') and x'(T') = znew then
6:       c' = Cost(znear) + c(x');
7:       if c' < Cost(znew) and c' < cmin then
8:          zmin ← znear;
9:          cmin ← c';
10: return zmin
```

---

**Algorithm 3:** T ← ReWire(T , Znear, zmin, znew)

```
1: for znear ∈ Znear \ {zmin} do
2:    (x', u', T') ← Steer(znew, znear);
3:    if ObstacleFree(x') and x'(T') = znear and Cost(znew) + c(x') < Cost(znear) then
4:       T ← ReConnect(znew, znear, T);
5: return T
```