Collaborative robotics and its paradigms for industrial manipulators

Weng, Ching-Yen

2021

Weng, C.-Y. (2021). Collaborative robotics and its paradigms for industrial manipulators. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/145983>

<https://doi.org/10.32657/10356/145983>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 17:13:05 SGT

COLLABORATIVE ROBOTICS AND ITS PARADIGMS FOR INDUSTRIAL MANIPULATORS

WENG CHING-YEN

School of Mechanical & Aerospace Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of

Doctor of Philosophy

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

23 June, 2020

Date

A handwritten signature in black ink, appearing to be '翁政堯' (Weng, Ching-Yen), written over a dotted line.

WENG, CHING-YEN

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

23 June, 2020

Date

CHEN, I-MING

Authorship Attribution Statement

This thesis contains material from 8 papers published in the following peer-reviewed journals/from papers accepted at conferences in which I am listed as the first author or co-first author.

Chapter 2 is published as CY Weng, WC Tan and IM Chen, “A Survey of Dual-Arm Robotic Issues on Assembly Tasks,” in ROMANSY 22 – Robot Design, Dynamics and Control, 2018.

The contribution of the co-authors are as follows:

1. Prof. Chen provided the initial project direction.
2. I prepared the manuscript drafts.

Chapter 3 is published as CY Weng and IM Chen, “The task-level evaluation model for a flexible assembly task with an industrial dual-arm robot,” in 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), 2017; CY Weng, WC Tan, Q Yuan and IM Chen, “Qualitative As-

assessment at Task-Level for Performance of Robotic Configurations and Task Plans,” in Journal of Intelligent and Robotic Systems, 2019; CY Weng, Q Yuan, ZJ Lim and IM Chen, “Applications of Light-Weight Wearable Devices to Online Programming of Industrial Dual-Arm Robots,” in Unmanned Systems, 2019; Q Yuan, CY Weng, ZJ Lim and IM Chen, “A Mirrored Motion Remapping Method in Telemanipulation-based Face-to-Face Dual-arm Robot Teaching,” in IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2019; CY Weng, Q Yuan, F Suárez-Ruiz and IM Chen, “A Telemanipulation-Based Human-Robot Collaboration Method to Teach Aerospace Masking Skills,” in IEEE Transactions on Industrial Informatics, 2020; Q Yuan, CY Weng, F Suárez-Ruiz and IM Chen, “Flexible Telemanipulation Based Handy Robot Teaching on Tape Masking with Complex Geometry,” in Robotics and Computer-Integrated Manufacturing, 2020.

The contribution of the co-authors are as follows:

1. Prof. Chen provided the initial project direction.
2. I and Dr. Yuan prepared the manuscript drafts.
3. Dr. Yuan, Dr. Suárez-Ruiz and Mr. Lim assisted in performing the experiments.
4. The manuscript was revised by Dr. Yuan, Dr. Suárez-Ruiz.

Chapter 4 is published as CY Weng, WC Tan, Q Yuan and IM Chen, “Qual-

itative Assessment at Task-Level for Performance of Robotic Configurations and Task Plans,” in Journal of Intelligent and Robotic Systems, 2019; CY Weng and IM Chen, “A vision-based strategy for a cost-effective flexible robotic assembly system without using RCC devices and compliant control,” in 2017 International Symposium on Robotics and Mechatronics, 2017. The contribution of the co-authors are as follows:

1. Prof. Chen provided the initial project direction.
2. I prepared the manuscript drafts.
3. The manuscript was revised by Dr. Yuan.
4. Dr. Yuan assisted in performing the experiments.

Chapter 5 is published as CY Weng, Q Yuan, F Suárez-Ruiz and IM Chen, “A Telemanipulation-Based Human-Robot Collaboration Method to Teach Aerospace Masking Skills,” in IEEE Transactions on Industrial Informatics, 2020; Q Yuan, CY Weng, F Suárez-Ruiz and IM Chen, “Flexible Telemanipulation Based Handy Robot Teaching on Tape Masking with Complex Geometry,” in Robotics and Computer-Integrated Manufacturing, 2020. The contribution of the co-authors are as follows:

1. Prof. Chen provided the initial project direction.
2. I and Dr. Yuan prepared the manuscript drafts.
3. The manuscript was revised by Dr. Suárez-Ruiz.

4. Dr. Yuan and Dr. Suárez-Ruiz assisted in performing the experiments.

23 June, 2020

Date

A handwritten signature in black ink, appearing to be '翁政仁' (Weng, Ching-Yen), written over a dotted line.

WENG, CHING-YEN

Acknowledgements

To my dearest family, my parents, my brother, and my lover, I am grateful for your continuous support and tolerance along the way. Without your steps into my life, it is likely that I have never achieved what it was all about.

To my monitor, Prof. I-Ming Chen, it is truly an unforgettable experience to work in your facilities, and thanks for the unfailing assistance and instructions in not only my academic path but also my life journey.

To my true-hearted friends, the countless screams of joy during this long time are my most memorable scenes. Whenever I feel frustrated, the thought of your company always makes me feel bright and encouraged.

To my colleagues in the Robotics Research Centre, in spite of the vicissitudes, we compose a wonderful story while sharing a laboratory. I will be always missing the good old days about our cooperation in my future time.

WENG, CHING-YEN, June 2020

Contents

Contents	i
Summary	1
1 Introduction	3
1.1 Motivation	3
1.2 Objectives	5
1.3 Organization	6
2 Literature Review	8
2.1 Collaborative Robotics	9
2.1.1 Collaborative Robot - Cobot	9
2.1.2 Robot-Robot Collaboration	12
2.1.3 Human-Robot Collaboration	12
2.2 Manipulation in Industrial Scenarios	14
2.2.1 Teaching Methods	14
2.2.2 Performance Evaluation	15
2.3 Challenges	20
3 A Framework of Collaborative Robotics for Manipulation	22
3.1 Cooperation, Collaboration, and Coordination	23
3.1.1 Cooperation	24
3.1.2 Collaboration	24

3.1.3	Coordination	26
3.2	Representation of a Robotic Task	27
3.2.1	Task	28
3.2.2	Agents	28
3.2.3	Action	30
3.2.4	Strategy	31
3.2.5	Plan and Sub-Plan	31
3.2.6	Coordination in a Task Plan	34
3.2.7	Graphical Models for Collaborative Patterns	36
3.3	Collaborative Manipulation	45
3.3.1	Spatial Configuration	45
3.3.2	Indirect Collaboration	46
3.4	Performance Evaluation	60
3.4.1	Robot-Centered Performance Assessment	62
3.4.2	Connection to Assessment at Economy-Level	67
4	Robot-Robot Collaboration: Peg-in-Hole	69
4.1	Introduction	70
4.2	Methodology	71
4.2.1	Perception	71
4.2.2	Manipulation	74
4.2.3	Task Representation	77
4.3	Experiments	81
4.3.1	Experimental Setup	82
4.3.2	Experimental Results	82
4.4	Analysis	84
4.4.1	Data Validation	85
4.4.2	Influence of Strategies	86
4.4.3	Estimation of a New Strategy	88
4.4.4	Dominance of robotic configuration	89

4.4.5	Discussion	90
4.5	Summary	95
5	Human-Robot Collaboration: Aerospace Masking	96
5.1	Introduction	97
5.2	Methodology	100
5.2.1	Collaborative Pattern	100
5.2.2	Manipulation	102
5.2.3	Skill Recognition	104
5.2.4	Optimization of Taping Skill	106
5.2.5	Optimization	107
5.3	Performance Evaluation	111
5.3.1	User Study	111
5.3.2	System Validation	115
5.3.3	Further Improvement	120
5.3.4	Discussion	124
5.4	Summary	125
6	Conclusion	126
6.1	Contribution	126
6.2	Future Works	129
	Bibliography	143
	List of Figures	144
	List of Tables	149
	Symbols	150
	Acronyms	153
	Lists of Publications	155

Summary

Industrial robots can be applied to a wide range of feasible applications through various levels of programming and integration into manufacturing systems. By the cooperation between various agents, various collaborative patterns can be introduced to achieve their industrial objectives and improve many aspects of the task performance, such as time-efficiency, dexterity, and flexibility. Although this concept has been largely studied, it is too novel to be systematically defined in recent studies, especially for the use cases of industrial applications, leading to the difficulties of mature continuous integration and continuous delivery in manufacturing. To enable the design of industrial manipulators to be collaborative with other agents, such as other robotic systems or humans, a promising method is expected to turn such an abstract concept into a mathematical model so that the existed technologies of performance assessment can be applied to evaluate the deployment of systems. In addition, to consider both the precision and flexibility in human-robot collaboration for the industrial manipulators, to propose a unified framework to cover the contents of collaborative manipulation would support a more effective task execution with human-robot collaboration.

To make up for the gaps in research and industrial requirements, this thesis studies collaborative robotics in the applications of industrial manipulation. Three novel aspects of the problems in the field of collaborative robotics are studied: “the definitions and classifications”, “task representation and performance assessment”, and “collaborative manipulation”. In

the first aspect, commonly used terminologies are clearly defined to avoid confusion and a general classification is proposed to group various collaborative patterns and forms. In the second aspect, a task representation method is developed to conform to the aforementioned classification. Furthermore, this representation is able to model robotic tasks for performance assessment, which can provide a more scientific manner to develop and deploy a robotic system in manufacturing. In the third aspect, a collaborative manipulation framework enables human operators to collaborate with robots in a more flexible, intuitive, and time-efficient manner. To validate the effectiveness, a vision-based peg-in-hole task is conducted in a robot-robot collaborative pattern for the performance assessment. Besides, an aerospace masking task is improved via the telemanipulation with a human-robot collaborative pattern. These two cases effectively demonstrate the applications of the developed theories and prove the feasibility of the proposed methods in this thesis, which not only imply the theorizability of collaborative robotics but also show its potential values in more industrial applications.

Chapter 1

Introduction

1.1 Motivation

With the productivity demands of our human society, the introduction of industrial robots has become an indispensable and irreversible trend of industrial automation for the realization of high productivity. George Devol first advocated the concept of an industrial robot and subsequently applied a patent for it. In less than a decade, George and Joseph Engelberger established the first industrial robot company, “Unimation”, in the United States in 1959. This symbolized the evolution of a brand-new era of robots due to the rapid development of technology from a variety of fields since then.

To integrate industrial robots into the various applications such as picking, handling, assembly, spraying and welding, a complete robot work cell could often costs up to four or five times the price of a single robot. Fortunately, these investments costs could rapidly be recovered through careful planning and proper deployment of the robots in the production line. To augment the deployability of the robots, they can be equipped with various sensors to provide useful feedback in the decision making processes, thereby improving the robots’ abilities to cope with more elaborate tasks that may need higher precision and robustness.

As it is detrimental to the whole society if all human workers are replaced with robots, it would be beneficial for all parties if a middle ground is established, such as an in-depth collaboration between robots and human workers. Furthermore, this collaboration can enhance the overall capabilities of each agent, while alleviating the workload of the human operator and reducing the conflict of interests between laborers and stakeholders. Henceforth, many industrial paradigms are employed for the various industrial tasks with the above considerations in mind.

The diversity of industrial paradigms provides multiple choices on the determination of robotic systems and their corresponding task plans. For instance, for tasks that require precision or efficiency, a purely-robotic environment would be a better choice thanks to the repeatability and speed of the robots. If these tasks occasionally suffer from unmodelled or unpredictable factors, a human operator would be able to provide an efficient and flexible decision making process to close the loop (Human-Robot Collaboration). Furthermore, multiple robots may coordinate (Robot-Robot Collaboration) to achieve better time-efficiency and dexterity in manipulation. Hence, the considerations of different aspects of tasks usually affect the configuration and deployment of robotic systems, which is also an open research topic in the field of industrial robotics.

Despite their diversity, researchers (someone who studies robotics) can categorize the different paradigms under a general classification in order to establish a mathematical representation for a robotic task plan. This mathematical representation plays an important role, especially for different paradigms that involve the introduction of multiple robots or humans because practitioners (researchers, engineers, operators) can adopt various quantitative techniques to assist manufacturers (normally the stakeholders of plants) in decision making. With such tools, the performance of robotic configurations under different tasks can be clearly determined and fine-tuned

in the production lines. In addition, such tools facilitate the manufacturers to close the loops from the selection of robotic configuration to the task plan making, which allows the manufacturers to have reference criteria to optimize the overall solutions under continuous integration and continuous delivery.

As high-mix low-volume tasks are common, it is necessary to increase the flexibility of those robotic systems during deployment to avoid cumbersome and time-consuming calibration processes. Moreover, it is important to achieve efficient collaborative manipulation (especially when human operators are involved) to enable a better robot-teaching process for tasks involving precise contact force and trajectory tracing, in the multi-Degree of Freedom (DoF) manipulation within a constrained environment.

In view of the above, this thesis attempts to represent the robot behaviors within collaborative robotics in this thesis. Various collaborative methods are explored to offer a flexible and efficient robotic manipulation in industrial environments for human operators. Based on these works, the paradigms are applied onto two typical tasks each of which has different collaborative patterns: robotic assembly, and robotic taping. The results of this work aims to further inspire the research study of both collaborative and industrial robotics in time to come.

1.2 Objectives

In this thesis, collaborative robotics on three aspects are studied:

1. This thesis proposes a clear definitions of *cooperation*, *collaboration*, and *coordination* and provide a novel classification for researchers to categorize different collaborative patterns and forms.
2. This thesis develops a mathematical representation that is compatible with different collaborative patterns. It can assist practitioners to

model task plans and connect to traditional system analysis methods for performance assessment.

3. This thesis provides a collaborative manipulation framework for human operators to teach robots with flexibility, intuition, and time-efficiency in different scenarios.

1.3 Organization

The remainder of this thesis is organized as follows:

- Chapter 2 provides the literature review related to this work.
- Chapter 3 discusses the concept of collaborative robotics and proposes a framework to clear the branches of this field. A representation is proposed to model the task plans under a robotic system. With the help of this representation, different collaborative patterns and forms can be modeled. Besides, a manipulation framework is also developed to achieve high-precision and flexible collaborative manipulation in indirect collaboration form. Finally, this representation supports performance assessment at task-level, indicating that the performance at robot-level and economic-level can be bridged thanks to this model.
- Chapter 4 develops a two-stage vision-based dual-arm robotic assembly task under the [Robot-Robot Collaboration \(RRC\)](#) collaborative pattern. Based on this paradigm, the modeling process via the proposed representation is established. Besides, the performance assessment at the task-level is also conducted, which provides practitioners a reference for the use of the proposed method.
- Chapter 5 develops a more efficient and intuitive online programming method for robotic aerospace masking under the [Human-Robot Collaboration \(HRC\)](#) collaborative pattern. Based on this paradigm, the

operators can set up a robot in a more flexible manner and save much time on programming robots, which makes [HRC](#) pace a step forward to real industrial applications.

- Chapter [6](#) highlights the contribution of this work and provides the guideline upon future works.

Chapter 2

Literature Review

The concept of collaborative robotics has been widely used in robotics, facilitating the development of both research and engineering in industrial automation. Although there has been a vast amount of literature on collaborative robotics, to the best knowledge of the author, they have not been well-categorized. Therefore, this thesis discusses the literature based on the following categories, where this taxonomy is also related to the theory that will be illustrated in the later chapters.

1. Collaborative robotics: Collaborative robotics has attracted increasing attention from researchers and engineers on the deployment of various scenarios for improving the efficiency and effectiveness in task execution. The discussion herein focuses on the concept of *cobot* and the paradigms and methodologies that have been employed in the tasks involving multiple robots. Besides, the collaboration between humans and robots on task execution will also be included in this discussion.
2. Manipulation in industrial scenarios: Robot manipulation plays an important role in the success of task execution. In industrial scenarios, there are many programming methods, or so-called *teaching methods*, that provide practitioners flexible ways to transfer the expected skills

to robots. Plus, researchers have proposed many measuring criteria and methodologies for evaluating the performance of the deployment of robots. Hence, the discussion herein will focus on the different teaching methods that have been commonly used in research and manufacturing, and the performance evaluation for robots and automation.

2.1 Collaborative Robotics

2.1.1 Collaborative Robot - Cobot

Cobot is the abbreviation of the term *collaborative robot* which indicates to those robots that are designed to work with human workers friendly. This concept was first invented in a US patent in 1999 [1]. After that, [International Organization for Standards \(ISO\) 10218-1 \[2\]](#) in 2006 provides four types of collaborative operation:

1. Safety-rated monitored stop: When the entrance intention of operators is detected by safety devices, the robot motion is forced to be stopped until the operators leave the workspace. This operation does not require the removal of robot power for stopping the robot.
2. Hand-guiding: When the operation of a safety-rated monitored stop is activated, the robot system allows the operators to manipulate the robot via enabling devices.
3. Speed and separation monitoring: When the entrance intention of operators is detected by safety devices, the speed of robot motion is regulated in accordance with its distance from the operators.
4. Power and force limiting: the design of robot systems is required to have power and force limiting (under 80 Watt/150 N) to avoid human

injuries in either intentional or unintentional case. Note that the criteria of power and force limiting have been removed in the 2011 version of [ISO 10218-1](#) [3].

Recently, [ISO/Technical Specifications \(TS\) 15066](#) [4] rules several new safety requirements for cobots and their workspace. The core idea of [ISO/TS 15066](#) is that it allows contact between robots and humans. Besides, it also provides a guide for power and force limits related to a study of injury based on the threshold to pain, where the study considers quasi-static contact and transient contact on [Physical Human-Robot Interaction \(pHRI\)](#). Under these specifications, the safety issue can be guaranteed by meeting the robot design of force or power limiting while industrial robots and humans are moving in a shared workspace.

To make this possible, the feature of cobots is the capability of forces/torques sensing in their joints, enabling them to timely detect abnormal forces applied to themselves. With this feature, cobots can be programmed to have braking or reverse motion functions. Besides, this feature also supports these cobots to be taught through easily human hand guiding, where the cobots record waypoints while being moved by human operators. Finally, from an economic point of view, cobots can save the costs on setting up an environment that supports safe human-robot collaboration in accordance with [ISO 10218-1](#) [3] (first introduced in [2]) and [ISO 10218-2](#) [5]. Without the use of fencing or other peripheral for safety issues in low-risk applications, implying that the deployment of a human-robot collaboration cell could be built in a more efficient and flexible way thanks to the introduction of cobots. Several well-known cobots have been developed for such low risk applications (Figure 2.1), such as UR-Series from [Universal Robots \(UR\)](#) [6], Yumi from [ABB Robotics](#) [7], PR2 from [Willow Garage](#) [8], etc.

(a) UR-Series [6]

(b) Yumi [7]

(c) PR2 [8]

Figure 2.1: Several models of collaborative robots.

Humanoid robots are also categorized into collaborative robots due to their anthropomorphism design. There are two implied reasons: firstly, the manipulation of the robot involves multiple manipulators, of which they need to collaborate with each other to finish a task [9]; secondly, they are regarded to be much easier to get into human-populated environments for [HRC](#) as they look like humans [10]. Starting from these two points of view, this section will be divided into two parts: [RRC](#) and [HRC](#).

2.1.2 Robot-Robot Collaboration

While the term **RRC** is not well-defined in robotics research and engineering, the study of it is not limited to the interaction of multiple manipulators. Multiple branches of **RRC** have been widely explored, such as control [11] [12] [13], motion planning [14] [15], coordination [16] [17], task allocation [18] [19], exploration [20] [21] etc. In particular, the study of mobile robots dominance this area due to the trend of the distributed system [22]. Nevertheless, the introduce of **RRC** still plays an important role on cost-effectiveness issues, such as energy efficiency in automotive industries [23], time-efficiency in welding tasks [24], dexterity in fixture-less assembly tasks [25], etc.

2.1.3 Human-Robot Collaboration

HRC has been an attractive research field that many researchers and practitioners believe to bring potential benefits for industries. In general, **HRC** is a narrow term originated from **Human-Robot Interaction (HRI)** [26], where Yanco et al. [27] have provided a taxonomy for several **HRI** as shown in Figure 2.2. **HRC**, however, has a deeper meaning which aims to reach a specific goal. In addition to this meaning, **HRC** often focuses on a team composed of several members with complementary skills so that they can finish a given task collaboratively [28]. Therefore, the design of team members and collaboration paradigms are vital for effective and efficient task execution, especially for **HRC** in industrial deployment.

Figure 2.2: Taxonomy of HRI proposed by [27], where human is labeled as “H” and robot is labeled as “R”.

Figure 2.3: Mechanism of collaboration by [10].

The field of HRC has been studied for several decades. In [10], an internal mechanism of collaboration has been summarized from a cognitive robotics point of view, as illustrated in Figure 2.3.

In summary, the study of collaborative robotics covers many aspects from the proposed design of robots to the operation pattern of robots. The border, however, has not been well clarified within the concept, which motivates the study of the classification and mathematical modeling for giving a theory from a new perspective.

2.2 Manipulation in Industrial Scenarios

Robotic systems are expected to perform changing and complex tasks in order to release human operators from tedious and hazardous environments. However, due to their lack of *intelligence*, robots still require human assistance to cope with the difficult parts of such tasks. Robotic systems can benefit from the skills of human operators such as dexterous manipulation, reliable perception and a great sense of kinesthetic coordination. The optimal architecture to get the best out of skill transferring is still an open challenge [29] for existing industrial robotic systems.

2.2.1 Teaching Methods

Traditional Methods

Traditional solutions to cope with robotic manipulation tasks tend to have high demands of offline-programming-based or teach-pendant-based robot teaching. Such approaches are difficult to scale due to the high complexity of robots with multiple [Degree of Freedom](#). Nevertheless, some progress has been made in various tasks such as pick-and-place [30], robotic assembly [31], in-hand manipulation [32]. Challenges in these tasks include controlling the contact forces and collision-free motion planning. Classic teach-pendant-based teaching often relies on recorded poses of the robot's end-effector online, then automatically interpolates those poses in the configuration spaces of robots for trajectory generation. Though more adaptive to industrial applications, this method makes operators tiresome to cope with complex paths or geometries due to the increase of the recorded poses [33]. On the other hand, teleoperation-based teaching enables operators to use their motions to guide robotic systems online, providing great flexibility due to the high cognitive and perceptual capabilities of the human-in-the-loop. With the employment of various light-weight interactive devices that

can capture human motion, such as [Inertial Measurement Unit \(IMU\)](#) [34], vision [35] [36] and [Virtual Reality \(VR\)](#) [37], human motion driven [Programming by Demonstration \(PbD\)](#) [38] provides a more intuitive paradigm for users to deploy robots into tasks without spending much of time on programming. Therefore, people believe that the characteristics of human motion driven teleoperation methods could bring potential values to tasks with the nature of high-mix low-volume [39].

Programming by Demonstration

Recently, as a form of [HRI](#), the concept of [HRC](#) has become a common method to transfer skills from human operators to robotic systems [29]. [Teaching by Demonstration \(TbD\)](#) or [PbD](#) is proposed to be a more intuitive methodology than traditional online programming. Through [TbD](#), operators can physically drag a robot around instead of pressing buttons or touching screens on teach pendants to adjust robots' poses, often referred to as waypoints, and conduct robot teaching. The efficiency of this methodology has been confirmed to be better than traditional teaching on tasks that do not require contact interactions with the environment, welding being a classical example [40]. Moreover, [TbD](#) has improved robot skill learning through different forms of [HRI](#) such as vision [41] and tactile [42]. A more detailed literature review on the methods and algorithms for [TbD](#) can be found in [43]. On the basis of these approaches, [TbD](#) is a promising method to bring robots closer to help humans under multiple scenarios, meeting the public imagination of a future [HRI](#) reality.

2.2.2 Performance Evaluation

Indisputably, the vital problem of employing a robot in manufacturing is to increase net benefits in the contemporary business community. Many researchers have provided methods or guidelines to study the cost-effectiveness

of deploying robotic systems [44] [45] [46].

Economy Considerations

From the commercial point of view, cost and payoff are the most common consideration for practitioners to determine the design of a manufacturing system. In [44], the cost considerations were classified into one-time expenses and on-going costs, as depicted in Table 2.1. Besides, the cost considerations were classified by direct, indirect and non-economic benefits, as depicted in Table 2.2. In [47], the economics in assembly manufacturing was discussed. Several mathematics models were proposed to express the cost functions among manual, fix automation, and flexible automation. J. W. Troxler and L. T. Blank proposed a system value model for justifying the payoff of flexible automation [48]. By means of this justification methodology, the relevant decision factors were identified and the value of the robotic system can be determined, providing the choice of different working modes in manufacturing. In [49], an approach to the evaluation of the [Flexible Manufacturing System \(FMS\)](#) was provided, connecting the cost of strategic performances and the flexibility in the manufacturing system in order to compute the profitability of the [FMS](#). A. M. Dore et al. developed an economic model based on simulation [50]. The evaluation method in this model was presented as a top-level data-flow, including input data, rough plan, detail cell, simulate cell performance, simulate the overall system,

Table 2.1: List of one-time cost and on-going cost.

Cost	Property
Robot price	One-time
Plant modification	One-time
Accessories	One-time
Support	On-going
Safety equipment	One-time
Planning and support activities	On-going
Lost production time during setup	One-time
Training time and expense	On-going
Reprogramming and retooling	On-going
Retraining and early retirement for workers	On-going

Table 2.2: List of direct cost and indirect cost.

Cost	Property
Reduced labor cost	Direct
Higher productivity	Direct
Less scrap and rework	Direct
Lower inspection cost	Direct
Decreased insurance cost	Indirect
Reduced management involvement	Indirect
Fewer returns from the improvement of quality	Indirect
Removal of personnel from hazardous jobs	Non-economic benefits
Compliance with OSHA requirement	Non-economic benefits
Flexibility and growth potential	Non-economic benefits

Technology Considerations

In recent years, much of research about the performance analysis of robotic tasks were carried out [51] [52]. M. Shneier et al. provided a measuring and representing a method for the performance of different kinds of robotic assembly motion [53]. In [47], the equation of cost and flexibility was given to discuss the different kinds of assembly methods among manual, fixed automation and flexible automation. A. S. Kondoleon et al. proposed a comprehensive analysis of the cycle time for robotic assembly systems [54]. In [55], the performance evaluations of multiple 3D space manipulation skills and perception skills for a moving assembly line were discussed. J. Marvel et

al. proposed a set of performance evaluation metrics for force-based assembly, assisting practitioners to measure the quality of the assembly system to accomplish certainly various assembly tasks [56]. In [57], the [Assembly Reliability Evaluation Method \(AREM\)](#) based on the Hitachi [Assembleability Evaluation Method \(AEM\)](#) were applied to various production lines to prevent the fault occurrence. S. Balakirsky et al. developed a performance evaluation method for industrial kitting systems by a set of metrics, such as manipulation robustness, transporting ability, contact errors, failures during planning and execution, recovery ability, etc [58]. In [59], a more efficient Assembly [Failure Modes and Effects Analysis \(FMEA\)](#) was proposed to replace Process [FMEA](#) for the identification and alleviation of the defects caused by assembly. Through the medium of the time penalties of the [Design for Assembly \(DFA\)](#), the level of defects can be estimated from the assembly complexity which can be obtained by the calculation method given in the research.

Evaluation Methods

A manufacturing system always involves massive features that people cannot certainly draw a conclusion to judge its performance. Therefore, a scientific evaluation method possessing the capability of dealing with subjective and objective data is necessary. This kind of evaluation method refers to those methods which can assist us to analyze a system with quantitative or qualitative criteria and then produce a conclusion for our decisions.

[Analytic Hierarchy Process \(AHP\)](#) is a structured approach that is designed to solve complex multi-criteria decision problems [60]. This approach requires the relative importance value (preference) for each criterion from the decision-maker, which is suitable for solving complex problems with quantitative criteria. The principles of this approach include hierarchic structure modeling, priority synthesis, and ranking consistency. In [48], an exam-

ple of the evaluation of an automation system capability was given, where the availability, function, and flexibility, were accounted for, as shown in Table 2.3.

Table 2.3: Structure of AHP analysed in [48].

Goal	Criteria	Alternatives
Availability	Proven operation in similar applications	New machine
	Offline programming	Robot A
	Vendor support	Robot B
	Information Quality	
Function	Speed	
	Accuracy	New machine
	Payload	Robot A
	Repeatability	Robot B
Flexibility	Information quality	
	Part size	
	Part geometry	New machine
	Batch size	Robot A
	Information quality	Robot B

In Design of Experiments (DOE), the fractional factorial designs for Analysis of Variance (ANOVA) can effectively assist practitioners to only run a fraction experiment instead of a full experiment to draw a conclusion [61]. Through this approach, massive efforts of time and resources can be saved while doing a comparative study.

Queuing theory is a mature area which has been successfully employed in many kinds of manufacturing industry [62]. In this theory, a queue is regarded as a system into which customers arrive and wait for the receiving service. The features of a queuing model include arrivals, service mechanism and queuing discipline, which are similar to manufacturing product lines in factories. The queuing model can statistically simulate manufacturing product lines to deal with the stochastic nature, which even can be adaptive to industrial robots. For instance, the evaluation of the quantity of handling robot was modeled based on queuing theory [63]. The loading and unloading time of products, traveling time including loaded and empty, and the

breakdown of the system are considered.

In summary, the traditional teaching method has been widely used in various industrial applications. With the emergence of [HRI](#) technologies, an increasing number of new paradigms will be inspired, facilitating a more efficient and effective manner to transfer human skills to robots. Besides, recall the initial expectation of introducing robots in plants, the business value made by robots is still the most important factor. The lack of performance assessment at the task-level is also necessary to provide more information instead of rough estimation from manufacturers.

2.3 Challenges

This chapter provides a literature review on prior works done in the scope of collaborative robotics and their manipulation in industrial scenarios. From this review, the field of collaborative robotics and collaborative manipulation in the industrial scenario is still an open research and practice issue. For example, the framework of collaborative robotics is not specific, and several terms such as *cooperation*, *collaboration*, and *coordination* are not well-defined. Besides, a methodology for performance assessment at task-level is rare-explored for robot systems and their plans. The way to quantify this performance under a model that is unified-represented does not exist. Given this, the establishment of a bridge of performance quantification between robot-level and economic-level still has a long way to go. Finally, although the concept of [HRC](#) and [PbD](#) has been proposed, the related paradigms on industrial applications with the introduction of robot intelligence are still rare in real deployment due to time-efficiency. Hence, all these issues are worthy of being working on in the future.

The following chapter will provide a theoretical study on the developed framework of collaborative robotics, where the aforementioned terms (coop-

eration, collaboration, and coordination) will be defined clearly. In addition, a novel and unified manipulation framework for human-robot collaboration will be given. Finally, a novel performance quantification method will be presented for practical use in the cases of industrial manipulation.

Chapter 3

A Framework of Collaborative Robotics for Manipulation

In this chapter, a theoretical framework for collaborative robotics is proposed for its deployment in industry. This framework copes with three issues in collaborative robotics before the deployment is feasible in industry: representation of collaboration, collaborative behavior, and performance assessment.

First of all, serving as a vertical connection to all the components with different levels in robotic tasks, a formal representation is developed for clear definitions of them, where this representation can be generalized to various kinds of robotic systems, such as non-collaboration, [RRC](#), and [HRC](#). Besides, this representation models task plans in a graphical form, supporting the extension of additional robotic systems for representing the collaborative behavior of multiple agents.

Secondly, as a necessary part of collaborative behaviors, manipulation plays an important role. Existing manipulation methods and related analysis for [RRC](#) have been widely used in industry. Hence, the proposed framework focuses on the manipulation of indirect [HRC](#) with high precision. Through the proposed kinematics, human operators can work closely with robots in

industry applications.

Thirdly, a robot-centered approach for performance evaluation at task-level, which bridges the gap between the robot-level and economy-level, is of importance before the robotic systems can be successfully deployed in industry scenarios with acceptable cost-effectiveness to manufacturers. Based on our representation models, the developed approach can support the practitioners to quantify the performance of robotic systems at task-level.

The mentioned theoretical contents in this chapter are applied to two different collaborative patterns ([RRC](#) in Chapter 4 and [HRC](#) in Chapter 5) for validation.

3.1 Cooperation, Collaboration, and Coordination

In our lives, cooperation, collaboration, and coordination are terms that with a similar implication in language, understood as several participants engage in a given task together. From the literal point of view, however, there are some differences between these three terms. Based on the Oxford English Dictionary, cooperation refers to

”The process of working together to the same end.”

Collaboration refers to

”The action of working with someone to produce or create something.”

Coordination refers to

”The organization of the different elements of a complex body or activity so as to enable them to work together effectively.”

As so far, these three terms are not well-defined in robotics, making researchers and engineers sometimes confused about their meaning. In this work, this thesis provides a perspective for standardizing their usages.

3.1.1 Cooperation

Putting apart the prefix *co-* which means *together*, the target of the word *operation* often indicates a task. Combining the explanation from the Oxford English Dictionary, the nature of a task is emphasized in this term, namely, the task is executed by some participants together. Therefore, while using this term, the object of study is expected to be the task, and it does not care about the relationships of the participants as well as their behaviors. Given this, this thesis defines the cooperation for robotics as follows:

Definition 3.1.1 (Cooperation). Cooperation in a task means that the task is executed by multiple agents, where these agents directly or indirectly interact with each other for a common and expected goal. A task with cooperation is called a cooperative task.

3.1.2 Collaboration

Collaboration is frequently used in [HRC](#), focusing on the interaction between human(s) and robot(s). In addition, the term *collaborative robot* (or *cobot*) has a definition with the consensus in [HRC](#), that is, a robot designed for direct interaction with humans. Another common usage of collaboration is for multi-robot collaboration, studying the aspects of decisions, perceptions, and actions, from multiple robots. These three usages of collaboration summarize that collaboration in robotics emphasizes the interaction among studied agents, where the agent can be robot or human. For example, robot-robot collaboration emphasizes the interaction among multiple manipulators in a task, while human-robot collaboration emphasizes the interaction be-

tween human(s) and manipulator(s). Thus, this thesis defines a generalized concept for collaboration in robotics as follows:

Definition 3.1.2 (Collaboration). Collaboration indicates the interactive behaviors of an agent/agents in a given task.

The collaboration plays an important role in the modeling of interactions in robotics. A taxonomy for different collaborative forms is necessary for a clear understanding of different collaborative behaviors in robotic tasks. In this section, this thesis focuses on three patterns of collaboration: non-collaboration, robot-robot collaboration, and human-robot collaboration, as depicted in Figure 3.1.

Figure 3.1: Patterns and forms of collaboration.

These three patterns present the different collaborative forms of industrial robots. Firstly, the non-collaboration refers to an agent without collaborative behaviors, which normally indicates a robotic system that has only one manipulator in itself, such as a single-arm robotic system. Secondly, [RRC](#) is more complex than the pattern of non-collaboration. Their collaborative form is based on the location of collaborative behaviors that appears. For a robotic system that needs collaboration with other robotic systems to finish a task, its collaborative form is defined as *external collaboration*, such as a multi-robot system which is composed of multiple robotic systems; For a robotic system that allows collaborative behaviors to appear

within itself, its collaborative form is defined as *internal collaboration*, such as a multi-arm robotic system. Thirdly, with the introduction of humans, the collaboration pattern of HRC can be divided into two forms based on the effects of the human in a task, the *direct collaboration* and the *indirect collaboration*. Direct collaboration means that the human and the robotic system execute different parts of a task, respectively, while indirect collaboration means that the human finishes a whole task only via the robotic system.

3.1.3 Coordination

The prefix *ord-* comes from Latin, where it has the meaning "order." That is, coordination can be regarded as a virtual organizer for making things be done in an orderly way. To achieve this, it must have the capability to establish communication for observing states and delivering commands globally. As a result, this term, as an organizing function, is of importance for not only a successful cooperative task but also an effective collaboration among multiple agents. Nevertheless, coordination is more than a concept for the nature of tasks or the interaction among agents. For example, the coordination of multiple sensors in a system, handling the merging of sensory information. This form of coordination does not involve in the aspect of cooperation and collaboration in robotics. Hence, the term coordination focuses on the aspect of actions, and the coordination for robotics is defined as follows:

Definition 3.1.3 (Coordination). Coordination refers to communication among actions in one or more agents. The existence of coordination is for the effectiveness of the interaction of actions in tasks.

3.2 Representation of a Robotic Task

Before starting to develop a representation of a robotic task, let us investigate a task and its composition from the human's point of view.

In general, a task is defined by a specific goal. For example, assembly is a task with a specific goal. To assembly, we can use a hand to grasp a component or rely on a fixture on the table, depending on the availability of tools in our surroundings. This indicates that the process to achieve a task goal is relevant to the environment we stay in. If there is nothing within our surroundings, then we cannot finish the task. Or if there is a fixture but our motions are constrained from reaching it, then we cannot achieve the goal no matter how we put effort to interact with the environment. In other words, the initial environment we stay directly affects the feasibility of finishing the task. On the other hand, the finishing state of a goal often implies termination of the task, implying that there is a certain environment that proves the success of the goal. Now, assume that we already have a fixture on the table. The next question is how do we assembly two components together. We can use our right hand to guide the component for insertion, or we can use the left hand. These selections are based on the availability of the human body. Different selections also lead to different performances on finishing time, success rate, assembly quality, and human feelings, etc. After determining the guiding manner, we will face almost unlimited sequences of actions that can achieve the goal. All the actions we selected are therefore turned into our plan for this task so that we can know how to finish the given task.

According to this simple example, we can find some interesting common points which can be used for representing a robotic task.

3.2.1 Task

As the origin of the developed representation, a task $\mathfrak{T}$ comes with three necessary elements: a specific goal, an initial state, and a final state.

1. Goal: A goal indicates the expected results of the task given to a robotic system.
2. Initial state: Based on the conditions of the robot workspace, an initial state is pre-determined. In a practical production system in manufacturing, this thesis considers a robotic system to serve as a workstation in a production line. Thus, considering the existence of neighboring workstations (robotic systems or machines), we often need several assumptions to define the initial state of the studied task $\mathfrak{T}$ from the final state of the previous workstation.
3. Final state: Final state indicates the finishing state of the studied task $\mathfrak{T}$, where the goal is achieved. The operation of the robotic system is considered to start a new episode of task $\mathfrak{T}$ after this state.

3.2.2 Agents

From Definition 3.1.2, agents can refer to humans or robots. In this work, this thesis distinguishes robots and humans by using robotic agents and human agents.

A robotic agent set $\mathfrak{R}$ is proposed to defined a set of robotic agent $\mathfrak{R}_i$ that can perform part or full of a task $\mathfrak{T}$, where each robotic agent has following definitions:

Definition 3.2.1 (Robotic Agent). A robotic agent is composed of at least one manipulator and sensors, end-effectors, and other peripherals that are needed for part or full of the executed task.

This definition implies that any unnecessary device for the executed task does not belong to the studied agent, even if it is mounted on the agent. For example, if a task $\mathfrak{T}$ can be accomplished by using only one RGB-Depth (RGB-D) camera, one 6 DoF manipulator, and one gripper in a robotic agent $\mathfrak{R}_1$, then all the mentioned components belongs to $\mathfrak{R}_1$. However, if there is an additional one 6 DoF force-torque sensor mounted on the end of the manipulator, then the force-torque sensor does not belong to $\mathfrak{R}_1$ because it is not effective in task $\mathfrak{T}$. Based on this definition, we can classify the type of robotic system for a robotic agent:

1. Single-arm robotic system: a robotic agent with only one manipulator performs actions in a given task.
2. Multi-arm robotic system: a robotic agent with more than one manipulators performs actions in a given task.

While classifying the type of robotic system for a robotic agent, the number of manipulators that a robotic agent has is not in the priority of the consideration. Instead, the criteria are the number of manipulators that perform actions in a given task. The concept of the type of robotic system is important because it effectively names the robotic agents based on their capabilities, rather than based on their configurations. For example, while deploying a dual-arm robot in a robotic agent, where one of the manipulators is disabled, this robotic system can only be regarded as a single-arm robotic system, where the disable manipulator plays a role of obstacle in the robot workspace.

On the other hand, a human agent is composed of a human and given tool in a task $\mathfrak{T}$. This thesis uses $\mathfrak{H}$ and $\mathfrak{H}_i$ to denote a human agent set and a human agent for the task $\mathfrak{T}$, respectively.

For performing a task, a set of the feasible agent(s) that can finish full of the task is called a feasible agent set $\mathfrak{F}$, defined as follows:

Definition 3.2.2 (Feasible Agent Set). Given a task $\mathfrak{T}$, assume a set of robotic agent set $\mathfrak{R}$ or a set of human agent set $\mathfrak{H}$ exists, of which elements can perform part or full of $\mathfrak{T}$. A feasible agent set $\mathfrak{F} \in \wp(\mathfrak{R} \cup \mathfrak{H}) \setminus \emptyset$ is a set of agents that can finish the task $\mathfrak{T}$, where the $\wp(\bullet)$ refers to the power set of $\bullet$.

3.2.3 Action

From the aforementioned example, we know that robotic actions are like human behaviors, such as perception and movement. In this work, an action set $\mathcal{A}$ indicates a set of behavior that a robotic agent or a human agent can perform. In each action $\mathcal{A}_i \in \mathcal{A}$, an information $I_i = (t_i, r_i)$ is carried, where t_i refers to the execution time and r_i refers to the success rate of the action. Sometimes, it is hard to observe the success rate of every action. Thus, this thesis poses three assumptions:

1. If the success rate of an action $\mathcal{A}_i$ is observable, then this action is called *detection action*, denoted $\mathcal{A}_i \in \mathcal{A}^d$, where $\mathcal{A}^d$ is the detection action set.
2. If the success rate of an action $\mathcal{A}_i \in \mathcal{A} \setminus \mathcal{A}^d$ is unobservable, then its success rate is given to be 100%.
3. The last action in a plan (definition can refer to Chapter 3.2.5) must not be a detection action. This assumption exists because no matter what the result of the last action performs, it will not affect the performance (the execution time of the overall task) in the new episode. Thus, turning this action to a detection action is meaningless.

In each robotic agent, there is always a set of actions that it can perform. Noteworthy, the action sets, from different agents belonging to the same robotic agent set, can be overlapped (fully or partially) or disjoint. In general, the action set can be categorized into three disjointed subsets:

1. Setup $^S\mathcal{A}$: The actions in this set are put in charge of the initialization of a robotic agent, including loading of programs, movement of robot initial pose, etc. This action serves as a necessary preparation before starting manipulation.
2. Manipulation $^M\mathcal{A}$: This set is the most important subset among the three subsets of the action set, where practitioners make more efforts for successful task implementation. The actions in this set include robot perception and robot movement, which is designed to obtain feedback from the environment and interact with the robot workspace.
3. Troubleshooting $^R\mathcal{A}$: The actions in this set are actions appearing after a robotic system fails to achieve expected results of detection actions before the final state. The actions exist to ensure the safety of both the robotic system as well as the workspace while waiting for the resetting of the task to the initial state of another new episode by some mechanisms.

3.2.4 Strategy

For an action $\mathcal{A}_i$, many factors may affect the performance of the action, such as different parameters, algorithms, and additional functions. This thesis calls these factors *strategies*, which are possible to change the carried information I_i of the action. From practitioners' point of view, an ideal result of an adopted strategy should be able to increase the robustness (i.e. improve the success rate r_i) or improve the efficiency (decrease the execution time t_i) of the action it affects.

3.2.5 Plan and Sub-Plan

In general, under a given robotic system, the solutions composed by different sequences of actions in $^M\mathcal{A}$ (as well as the combinations of the adopted

strategies) are various for a given task. Thus, these feasible solutions can constitute a *plan set*. The definition of a plan set is depicted as follows:

Definition 3.2.3 (Plan Set). *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} \subseteq^M \mathcal{A}$ and $\mathcal{E} \subseteq^M \mathcal{A}^2$ with $|\mathcal{E}| \leq C(|^M \mathcal{A}|, 2)$, a plan set is constructed by*

$$\mathcal{P} = \mathbf{p}(\mathcal{G}) = \bigcup_i \mathcal{P}_i, \quad (3.1)$$

where $\mathbf{p} : \mathcal{G} \mapsto \mathcal{P}$ is a function that maps a graph to a plan set. Note that the $|\bullet|$ is the cardinality.

The elements in the plan set are known as *plans*. Each plan is composed of limited number of the actions in $^M \mathcal{A}$ and has the following definition:

Definition 3.2.4 (Plan). *Given a plan set $\mathcal{P}$, a plan $\mathcal{P}_i \in \mathcal{P}$ satisfies:*

$$\mathcal{G}_i = g(\mathcal{P}_i) = (\mathcal{V}_i, \mathcal{E}_i) \subseteq g(\mathcal{P}), \quad (3.2)$$

where $g : \mathcal{P} \mapsto \mathcal{G}$ is a function that constructs a plan to a graph, $\mathcal{V}_i \subseteq \mathcal{V}$, and $\mathcal{E}_i \subseteq \mathcal{E}$.

As an important component for the proposed representation, another concept, sub-plan, is also introduced. A sub-plan is capable to acquire the overall information it carries. Through the functions of detection actions, the success rate of each sub-plan can be acquired. The definition of sub-plan and its graphical sufficiency are provided as follows:

Definition 3.2.5 (Sub-Plan). *Given a plan $\mathcal{P}_i$, each of its sub-plan $\mathcal{SP}_{jk}^i \subseteq \mathcal{P}_i$ constructs a weakly connected subgraph $\mathcal{SG}_{jk}^i = g(\mathcal{SP}_{jk}^i) = (\mathcal{SV}_{jk}^i, \mathcal{SE}_{jk}^i) \subseteq g(\mathcal{P}_i)$, where j and k implies the order and the selection of sub-plan, respectively. In addition, there must be a detection action $\mathcal{A}_l \in \mathcal{A}_i^d \subseteq \mathcal{A}^d$, or an action with zero outdegree $\delta^+(\mathcal{A}_l \in \mathcal{V}_i) = 0$, as a sink vertex in the constituting subgraph $\mathcal{SG}_{jk}^i$.*

Theorem 3.2.1 (Sufficiency of Sub-Plans). *Let a plan $\mathcal{P}_i$ construct a graph $\mathcal{G}_i = g(\mathcal{P}_i)$, the graph constructed by any sub-plan of $\mathcal{P}_i$ must be the subgraph of $\mathcal{G}_i$. However, not every sub-graph of $\mathcal{G}_i$ have their corresponding sub-plans in the plan $\mathcal{P}_i$.*

An illustration of Theorem 3.2.1 can refer to the Chapter 3.2.7.

Although the decomposition manners of a plan are flexible, the upper bounds for the number and the number of selection manners for sub-plans exist:

Theorem 3.2.2 (Number of Sub-Plans). *Let $|\mathcal{A}_i^d| > 0$ in a plan $\mathcal{P}_i$, the number of its sub-plans is no more than $|\mathcal{A}_i^d| + 1$.*

Theorem 3.2.3 (Selection for Sub-Plans). *Let $|\mathcal{A}_i^d| > 0$ in a plan $\mathcal{P}_i$, the number of the selection manners for its sub-plans is no more than $\sum_{n=0}^{|\mathcal{A}_i^d|} C(|\mathcal{A}_i^d|, n)$.*

For convenience and standardization, a manner for proper decomposition is developed:

Theorem 3.2.4 (Proper Decomposition for Sub-Plans). *Given a plan $\mathcal{P}_i$, let its possible sub-plans be $\mathcal{SP}_{jk}^i$. A proper decomposition exists if the selected sub-plans $\mathcal{SP}_{jk}^i$ satisfy two properties:*

1. *Completeness of plan:*

$$g(\mathcal{P}_i) = \bigcup_j g(\mathcal{SP}_{jk}^i). \quad (3.3)$$

2. *Maximization of the number of sub-plans:*

$$\mathcal{SP}_j^i = \operatorname{argmax}_{\mathcal{SP}_{jk}^i} |\mathcal{SP}_{jk}^i|. \quad (3.4)$$

where the $\mathcal{SP}_j^i$ is the expected selection of each sub-plan.

Through Theorem 3.2.4, the number of sub-plans is strictly limited to be $|\mathcal{A}_i^d| + 1$, and these sub-plans $\mathcal{SP}_j^i$ are disjoint with each others in a given plan $\mathcal{P}_i$.

Finally, the information $I_{SP_j^i}$ for a sub-plan $\mathcal{SP}_j^i$ can be obtained:

$$I_{SP_j^i} = \left(\sum_{\mathcal{A}_k \in \mathcal{SV}_j^i} t_k, \prod_{\mathcal{A}_k \in \mathcal{SV}_j^i} r_k \right). \quad (3.5)$$

3.2.6 Coordination in a Task Plan

With communication, the interaction among actions, within an agent or cross multiple agents, can work effectively in a task plan. In this section, this thesis defines the concept of coordination over the sub-plans, called *coordination product* in an n -ary Cartesian product form:

Definition 3.2.6 (Coordination Product). Given a set of sub-plans $\mathcal{SP}$, where $|\mathcal{SP}| > 1$ and every elements of $\mathcal{SP}$ are obtained via proper decomposition. If there are n actions ($n > 1$) that need to be coordinated in a period, then a coordination product $\mathcal{O}_i \in \mathcal{O}$ exists, where $\mathcal{O}$ is a set that satisfies $|\mathcal{O}| = n$ and $\mathcal{O} \in \wp(\mathcal{SP})$.

Next, with the introduction of coordination product(s), the calculation of information for each sub-plan changes. Herein, two operations over $\mathcal{SP}$ are defined for obtaining the equivalent sub-plans. Without losing the generality, both the operations take two sub-plans, $\mathcal{SP}_i$ and $\mathcal{SP}_j$, as examples, and the equivalent sub-plan is denoted as $\mathcal{SP}_{eq}$:

(a) Series operation

(b) Parallel operation

Figure 3.2: Illustration of the operation over $\mathcal{SP}$.

Series Sub-plan

For a series sub-plan (Figure 3.2 (a)), where there is no coordination product. The operation for obtaining the local information $I_{\mathcal{SP}_{eq}}$ can be calculated by:

$$I_{\mathcal{SP}_{eq}} = \left(\sum_{\mathcal{A}_k \in \mathcal{SV}_i} t_k + \sum_{\mathcal{A}_k \in \mathcal{SV}_j} t_k, \prod_{\mathcal{A}_k \in \mathcal{SV}_i} r_k \prod_{\mathcal{A}_k \in \mathcal{SV}_j} r_k \right). \quad (3.6)$$

Parallel Sub-Plan For a parallel sub-plan (Figure 3.2 (b)), where there is a coordination product. The operation for obtaining the local information $I_{\mathcal{SP}_{eq}}$ can be calculated by:

$$I_{\mathcal{SP}_{eq}} = \left(\max \left(\sum_{\mathcal{A}_k \in \mathcal{SV}_i} t_k, \sum_{\mathcal{A}_k \in \mathcal{SV}_j} t_k \right), \prod_{\mathcal{A}_k \in \mathcal{SV}_i} r_k \prod_{\mathcal{A}_k \in \mathcal{SV}_j} r_k \right). \quad (3.7)$$

These two operations can be generalized to more complicated diagrams of task plans, providing a convenient manner for obtaining the equivalent sub-plans.

3.2.7 Graphical Models for Collaborative Patterns

In this section, we start from the pattern of non-collaboration, a single-arm robotic system, through the developed representation.

Non-Collaboration

Given a task $\mathfrak{T}$ and a robotic configuration set $\mathfrak{R} = \{\mathfrak{R}_1, \mathfrak{R}_2, \mathfrak{R}_3\}$ where the single-arm robotic system $\mathfrak{R}_1$ is the object of study. Assume that this robotic system can perform ten actions ${}^M\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_{10}\} \subset \mathcal{A}$, where a detection action $\mathcal{A}_5 \in \mathcal{A}^d$ exists. In addition, two feasible plans are provided $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2\}$, as depicted in Figure 3.3.

Figure 3.3: Topological model of a single-arm robotic system in non-collaboration.

From Figure 3.3, the plan set $\mathcal{P}$ is modeled by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = {}^M \mathcal{A}$ and $\mathcal{E} = \{(\mathcal{A}_1, \mathcal{A}_3), (\mathcal{A}_2, \mathcal{A}_5), (\mathcal{A}_4, \mathcal{A}_5), (\mathcal{A}_5, \mathcal{A}_3), (\mathcal{A}_3, \mathcal{A}_6), (\mathcal{A}_5, \mathcal{A}_8), (\mathcal{A}_6, \mathcal{A}_9), (\mathcal{A}_8, \mathcal{A}_9), (\mathcal{A}_9, \mathcal{A}_{10})\}$. The graph $\mathcal{G}$ is composed by two sub-graph, $\mathcal{G}_1$ and $\mathcal{G}_2$:

1. For $\mathcal{G}_1$, we have a digraph $\mathcal{G}_1 = g(\mathcal{P}_1) = (\mathcal{V}_1, \mathcal{E}_1)$, where

$$\mathcal{V}_1 = \{\mathcal{A}_2, \mathcal{A}_5, \mathcal{A}_3, \mathcal{A}_6, \mathcal{A}_9, \mathcal{A}_{10}\} \text{ and}$$

$$\mathcal{E}_1 = \{(\mathcal{A}_1, \mathcal{A}_2), (\mathcal{A}_2, \mathcal{A}_5), (\mathcal{A}_5, \mathcal{A}_3), (\mathcal{A}_3, \mathcal{A}_6), (\mathcal{A}_6, \mathcal{A}_9), (\mathcal{A}_9, \mathcal{A}_{10})\}.$$

2. For $\mathcal{G}_2$, we have another digraph $\mathcal{G}_2 = g(\mathcal{P}_2) = (\mathcal{V}_2, \mathcal{E}_2)$, where

$$\mathcal{V}_2 = \{\mathcal{A}_1, \mathcal{A}_4, \mathcal{A}_5, \mathcal{A}_8, \mathcal{A}_9, \mathcal{A}_{10}\} \text{ and}$$

$$\mathcal{E}_2 = \{(\mathcal{A}_1, \mathcal{A}_4), (\mathcal{A}_4, \mathcal{A}_5), (\mathcal{A}_5, \mathcal{A}_8), (\mathcal{A}_8, \mathcal{A}_9), (\mathcal{A}_9, \mathcal{A}_{10})\}$$

These two graphs, $\mathcal{G}_1$ and $\mathcal{G}_2$, also correspond the plans $\mathcal{P}_1$ and $\mathcal{P}_2$ in the plan set $\mathcal{P}$, respectively. For the sub-graph of $\mathcal{G}_1$, assume we have three selections ($k = 1, 2, 3$):

1. Selection A ($k = 1$): $\mathcal{SG}_{11}^1 = (\mathcal{SV}_{11}^1, \mathcal{SE}_{11}^1)$, where $\mathcal{SV}_{11}^1 = \{\mathcal{A}_1, \mathcal{A}_2\}$ and $\mathcal{SE}_{11}^1 = \{(\mathcal{A}_1, \mathcal{A}_2)\}$. The subscript of $\mathcal{SG}_{11}^1$ refer to the first sub-graph of the first selection.
2. Selection B ($k = 2$): $\mathcal{SG}_{12}^1 = (\mathcal{SV}_{12}^1, \mathcal{SE}_{12}^1)$, where $\mathcal{SV}_{12}^1 = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_5\}$ and $\mathcal{SE}_{12}^1 = \{(\mathcal{A}_1, \mathcal{A}_2), (\mathcal{A}_2, \mathcal{A}_5)\}$. The subscript of $\mathcal{SG}_{12}^1$ refer to the first sub-graph of the second selection.
3. Selection C ($k = 3$): $\mathcal{SG}_{13}^1 = (\mathcal{SV}_{13}^1, \mathcal{SE}_{13}^1)$, where $\mathcal{SV}_{13}^1 = \mathcal{V}_1$ and $\mathcal{SE}_{13}^1 = \mathcal{E}_1$. The subscript of $\mathcal{SG}_{13}^1$ refer to the first sub-graph of the third selection.

Based on the Definition 3.2.5, only the selection B and C have corresponding sub-plans in $\mathcal{P}_1$ because there is either a detection action ($\mathcal{A}_5 \in \mathcal{A}_1^d$) or an action satisfying zero outdegree ($\delta^+(\mathcal{A}_{10}) = 0$). For selection A, although it is a subgraph of $\mathcal{G}_1$, it is not a sub-plan of plan $\mathcal{P}_1$ as it does not

satisfy the condition prescribed in the Definition 3.2.5, which also verifies the Theorem 3.2.1. Furthermore, based on Definition 3.2.4, we can know that the selection B ($k = 2$ in this case) meets the requirement of proper decomposition. From (3.2) and (3.3), we have

$$g(\mathcal{P}_1) = \bigcup_{j=1}^{|\mathcal{A}_1^d|+1} g(\mathcal{SP}_{jk}^1) = \bigcup_{j=1}^{|\mathcal{A}_1^d|+1} \mathcal{SG}_{jk}^1. \quad (3.8)$$

Because $|\mathcal{A}_1^d| = 1$, we can obtain another subgraph $\mathcal{SG}_{22}^1$ for the sub-plan $\mathcal{SP}_2^1$ by

$$\mathcal{SG}_{22}^1 = g(\mathcal{P}_1) \setminus \mathcal{SG}_{12}^1 = \mathcal{G}_1 \setminus \mathcal{SG}_{12}^1. \quad (3.9)$$

Thus, $\mathcal{SG}_{22}^1 = (\mathcal{SV}_{22}^1, \mathcal{SE}_{22}^1)$ satisfies $\mathcal{V}_{22}^1 = \mathcal{V}_1 \setminus \mathcal{SV}_{12}^1$ and $\mathcal{E}_{22}^1 = \mathcal{E}_1 \setminus \mathcal{SE}_{12}^1$. Therefore, the proper decomposition for plan $\mathcal{P}_1$ has $\mathcal{SP}_1^1 = \mathfrak{p}(\mathcal{SG}_{12}^1)$ and $\mathcal{SP}_2^1 = \mathfrak{p}(\mathcal{SG}_{22}^1)$. Similarly, with a selection k , the proper decomposition of plan $\mathcal{P}_2$ satisfies (3.3):

$$g(\mathcal{P}_2) = \bigcup_{j=1}^{|\mathcal{A}_2^d|+1} g(\mathcal{SP}_{jk}^2) = \bigcup_{j=1}^{|\mathcal{A}_2^d|+1} \mathcal{SG}_{jk}^2, \quad (3.10)$$

where $\mathcal{SG}_{1k}^2 = (\mathcal{SV}_{1k}^2, \mathcal{SE}_{1k}^2)$ has $\mathcal{SV}_{1k}^2 = \{\mathcal{A}_1, \mathcal{A}_4, \mathcal{A}_5\}$ and $\mathcal{SE}_{1k}^2 = \{(\mathcal{A}_1, \mathcal{A}_4), (\mathcal{A}_4, \mathcal{A}_5)\}$, and since $|\mathcal{A}_2^d| = 1$ in plan $\mathcal{P}_2$, $\mathcal{SG}_2^2 = (\mathcal{V}_2^2, \mathcal{E}_2^2)$ has $\mathcal{SV}_{2k}^2 = \mathcal{V}_2 \setminus \mathcal{V}_{1k}^2$ and $\mathcal{SE}_{2k}^2 = \mathcal{E}_2 \setminus \mathcal{E}_{1k}^2$. Finally, the proper decomposition for plan $\mathcal{P}_2$ has $\mathcal{SP}_1^2 = \mathfrak{p}(\mathcal{SG}_{1k}^2)$ and $\mathcal{SP}_2^2 = \mathfrak{p}(\mathcal{SG}_{2k}^2)$.

As so far, given a task $\mathfrak{T}$ and a single-arm robotic system $\mathfrak{R}_1 \in \mathfrak{R}$, each plan $\mathcal{P}_i$ of the plan set $\mathcal{P}$, as well as each of its sub-plan $\mathcal{SP}_j^i$, can be successfully represented in a graphical form based on the action set $\mathcal{A}$ that $\mathfrak{R}_1$ can achieve.

Robot-Robot Collaboration (RRC)

Normally, the RRC is often designed for a task that needs higher efficiency or dexterity. Extended from the developed representation for the pattern of non-collaboration, this thesis divides the pattern of RRC into two forms: internal (multi-arm robotic system) and external (multi-robot system). From a practical point of view, the coordination of a multi-robot system needs an external setup for the deployed robotic systems, while a multi-arm robotic system does not need to due to the built-in coordination function in its driver. Nevertheless, their kinematics design by their factory makes the multi-robot system more flexible on deployment. For a multi-robot system, a practitioner can determine the deployment of robot bases for every robotic system. For a multi-arm robotic system, however, its manipulators share the same base link in the factory setting, leading to the change of its workspace impossible. Without losing generality, this thesis uses a dual-arm robotic system and two single-arm robotic systems for modeling the multi-arm robotic system and the multi-robot system, respectively. The provided derivations will focus on the information of coordination products. For the further derivation of each sub-plan, readers can refer to the case of non-collaboration.

A dual-arm robot can be regarded as the variance of a single-arm robot, where two single-arm robots sharing the same robot base. Based on the difference of their kinematics, a dual-arm robot can perform single-arm and dual-arm manipulations. Except to consider another manipulator as a static or dynamic obstacle, the single-arm manipulation of a dual-arm robot is the same as a single-arm robot in the execution of the assigned action. While doing dual-arm manipulation, the two manipulators perform synchronized motions in each action, implying that these two actions share the same execution time. As a result, part of detection actions serving the coordination between these two actions is necessary.

Extended from the structure of non-collaboration, assuming that $\mathfrak{R}_2$ is the dual-arm robotic system, and it can perform ten actions ${}^M\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_{10}\}$, where two detection action $\mathcal{A}_3, \mathcal{A}_5 \in \mathcal{A}^d$ exist. As depicted in Figure 3.4, two feasible plans, $\mathcal{P}_1$ and $\mathcal{P}_2$, are executed by the two manipulators, respectively. For example, the detection action $\mathcal{A}_5$, which exist in both $\mathcal{P}_1$ and $\mathcal{P}_2$, constructs a coordination product $\mathcal{O}_1 \in \mathcal{O}$, which describes the coordination of dual-arm manipulation in task execution. Based on the Theorem 3.2.4, we can have the following sub-graphs for each manipulator:

1. For the manipulator executing $\mathcal{P}_1$:

- (a) $\mathcal{SG}_1^1 = (\mathcal{SV}_1^1, \mathcal{SE}_1^1)$: $\mathcal{SV}_1^1 = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$ and $\mathcal{SE}_1^1 = \{(\mathcal{A}_1, \mathcal{A}_2), (\mathcal{A}_2, \mathcal{A}_3)\}$.
- (b) $\mathcal{SG}_2^1 = (\mathcal{SV}_2^1, \mathcal{SE}_2^1)$: $\mathcal{SV}_2^1 = \{\mathcal{A}_3, \mathcal{A}_5\}$ and $\mathcal{SE}_2^1 = \{(\mathcal{A}_3, \mathcal{A}_5)\}$.
- (c) $\mathcal{SG}_3^1 = (\mathcal{SV}_3^1, \mathcal{SE}_3^1)$: $\mathcal{SV}_3^1 = \{\mathcal{A}_5, \mathcal{A}_6, \mathcal{A}_7, \mathcal{A}_{10}\}$ and $\mathcal{SE}_3^1 = \{(\mathcal{A}_5, \mathcal{A}_6), (\mathcal{A}_6, \mathcal{A}_7), (\mathcal{A}_7, \mathcal{A}_{10})\}$.

2. For the manipulator executing $\mathcal{P}_2$:

- (a) $\mathcal{SG}_1^2 = (\mathcal{SV}_1^2, \mathcal{SE}_1^2)$: $\mathcal{SV}_1^2 = \{\mathcal{A}_1, \mathcal{A}_4, \mathcal{A}_5\}$ and $\mathcal{SE}_1^2 = \{(\mathcal{A}_1, \mathcal{A}_4), (\mathcal{A}_4, \mathcal{A}_5)\}$.
- (b) $\mathcal{SG}_2^2 = (\mathcal{SV}_2^2, \mathcal{SE}_2^2)$: $\mathcal{SV}_2^2 = \{\mathcal{A}_5, \mathcal{A}_8, \mathcal{A}_9, \mathcal{A}_{10}\}$ and $\mathcal{SE}_2^2 = \{(\mathcal{A}_5, \mathcal{A}_8), (\mathcal{A}_8, \mathcal{A}_9), (\mathcal{A}_9, \mathcal{A}_{10})\}$.

For obtaining the information $I_{\mathcal{SP}_{eq}}$ at the coordinate product $\mathcal{O}_1$, we can use the (3.6) and (3.7) over $\mathcal{SG}_1^1, \mathcal{SG}_2^1$ and $\mathcal{SG}_1^2$, as depicted in Figure 3.5:

$$I_{\mathcal{SP}_{eq}} = \left(\max \left(\sum_{\mathcal{A}_k \in \mathcal{SV}_1^1} t_k + \sum_{\mathcal{A}_k \in \mathcal{SV}_2^1} t_k, \sum_{\mathcal{A}_k \in \mathcal{SV}_1^2} t_k \right), \prod_{\mathcal{A}_k \in \mathcal{SV}_1^1} r_k \prod_{\mathcal{A}_k \in \mathcal{SV}_2^1} r_k \prod_{\mathcal{A}_k \in \mathcal{SV}_1^2} r_k \right). \quad (3.11)$$

Figure 3.4: Topological model of a dual-arm robotic system in RRC.

Figure 3.5: The operations over $\mathcal{SP}_1^1$, $\mathcal{SP}_2^1$ and $\mathcal{SP}_1^2$.

According to the proposed concepts for RRC, another collaborative form is external collaboration, which needs an external setup for coordination among agents. This fact indicates that the coordination appears among the task plans executed by multiple agents, rather than a single agent like a multi-arm robotic system. Therefore, a coordination product normally

serves as the bond connect multiple detection actions, where these actions are executed by different agents in a task.

For the case of two single-arm robotic systems, their task plans are regarded to be executed by each agent independently. With the addition of coordination among the two agents, some actions are bonded. As an example, assume that there are two single-arm robotic systems, says $\mathcal{R}_1$ and $\mathcal{R}_3$, their task plans and sub-plans after proper decomposition are directly depicted in Figure 3.6. In this example, we can set up one coordination product $\mathcal{O}_1 = (\mathcal{SP}_1, \mathcal{SP}'_1)$ between $\mathcal{A}_3$ and $\mathcal{A}'_4$. The information for this coordination can be easily obtained by (3.7):

$$I_{\mathcal{SP}_{eq}} = \left(\max \left(\sum_{\mathcal{A}_k \in \mathcal{SV}_1} t_k, \sum_{\mathcal{A}_k \in \mathcal{SV}'_1} t_k \right), \prod_{\mathcal{A}_k \in \mathcal{SV}_1} r_k \prod_{\mathcal{A}_k \in \mathcal{SV}'_1} r_k \right), \quad (3.12)$$

where $(\mathcal{SV}_1, \mathcal{SE}_1) = g(\mathcal{SP}_1)$ and $(\mathcal{SV}'_1, \mathcal{SE}'_1) = g(\mathcal{SP}'_1)$.

Figure 3.6: Topological model of two single-arm robotic systems in RRC.

Human-Robot Collaboration (HRC)

Comparing to RRC, the situations become more indeterminable under the introduction of human performance. This is because the behaviors of hu-

mans are more complex to be analyzed than robots. For example, their capabilities, experiences, physical and mental status, etc. These factors have significant effects on their task performance. Besides, human performance in tasks is generally difficult to be measurable and predictable in repetitive experiments. Therefore, this thesis proposed a robot-centered representation for HRC. These graphical representations leave the extendable space for human performance. Once there is an acceptable manner to measure human performance in robotic tasks, then researchers or practitioners can combine the proposed representation for further analysis.

According to the proposed HRC collaborative pattern, it can be further divided into direct collaboration and indirect collaboration. For direct collaboration, both human and robot agents are considered to contribute their manipulation to a task, which is similar to the multi-robot collaboration. Hence, with the measurement of human agents, we can regard humans as another robotic agent. An example of direct collaboration is depicted in Figure 3.7.

Figure 3.7: Topological model of direct collaboration in HRC.

For indirect collaboration, only the robotic agent contributes the manip-

ulation to a task. This contribution, however, is under the interference of a human agent. In other words, without human interference, the robotic agent is not able to finish the task by itself. As a result, the actions executed by the robotic agent are partially or fully bonded to the actions of the human agent. An example of indirect collaboration is depicted in Figure 3.8, where this thesis uses $\mathcal{O}$ to indicate multiple robotic actions that bonded with human actions. Under this collaborative form, the execution time and success rate of the bonded robotic actions depend on the human agent.

Figure 3.8: Topological model of indirect collaboration in HRC.

3.3 Collaborative Manipulation

For effective collaboration in tasks, the spatial configuration of the robot(s) is an essential consideration for practitioners. Due to the reachability of robots, the positions of robot bases have a significant effect on task executions. This section provides techniques for effective collaboration on the manipulation of [RRC](#) and [HRC](#).

3.3.1 Spatial Configuration

The effectiveness of task manipulation considerably depends on the robot configurations and the given workspaces. In general, the setup of a robot, such as the position of the robot base, has a direct impact on the effectiveness of task execution. A bad setup often leads to the infeasibility of planned motion due to singularities and joint limits. With only an intuitive manner to determine the robot setup may cause the infeasibility for the robot to reach a planned configuration. Even though the configuration can be achieved, the efficiency of manipulation is still an unavoidable issue. In the pattern of [RRC](#), how to optimize the setup of robot base(s) to enable all the manipulators to reach the assigned waypoints in task space should be considered before the planning of collaborative manipulation. Researches upon this problem have proposed some methods based on manipulability ellipsoids [64], dexterity [65], reachability map [66], and capability [67]. The latest and most popular method is based on an inverse reachability map [68], which uses the voxelization method to create a workspace and discretize the workspace by octree [69]. This method provides an efficient way for practitioners to optimize the position of robot bases. With this technique, the setup of robot bases in [RRC](#) can be easier to be analyzed. For example, the setup of a dual-arm robot can be regarded as the optimization of the base position of the dual-arm robot and its workspace, of which the planned way-

points of both manipulators should be considered. As for the multi-robot system, the optimization can be achieved through this method by fixing one of the robots.

With frequently direct interaction between humans and robots in direct collaboration, the issue of human safety is always a spotlight in [HRI](#). Especially for the applications in manufacturing, not only labor safety is important, but the efficiency of collaboration is considered by manufacturers. As a result, the regulation of the workspaces of both humans and robots has potential research value for improving both safety and efficiency. For more details on this researches, readers can refer to [\[70\]](#).

3.3.2 Indirect Collaboration

In indirect collaboration, dexterous robot manipulation in the unstructured environment becomes a very important technology to make robots behave smartly and interact with people intelligently. To deal with the challenge in efficient task teaching of a robot with multiple [DoF](#) for dexterous manipulation, this section focuses on two motion retargeting methods, for enabling a human operator to teach robots directly in a normal mapping manner, or face-to-face in a mirrored mapping manner, by direct human motion through motion capture devices. The proposed methods enable an operator to visualize the on-site conditions and understand the robot's actions nicely in telemanipulation. Such methods are very useful to support robot teaching on various kinds of robotic setup and enhance the capabilities of dual-arm robots on more dexterous manipulation skills shortly.

Fast Calibration and Adjustable Motion Mapping (FCAM)

According to Chapter [2.3](#), the limitations of [Offline Programming \(OLP\)](#) includes:

1. The workpiece, the robotic system, and the physical environment need to be modeled in advance.
2. A tedious calibration process is required to determine the relative positions between the robot and the workpiece.
3. The generation of robotic manipulation using off-line motion planning but depending on the workpiece complexity, the algorithm often fails and does not provide a feasible solution.
4. The generated motion of the robot in the off-line environment may fail to work directly due to the differences between the physical and the off-line environment. Therefore, further adjustment of the motions can be troublesome and time-consuming.

Encountering these difficulties in the off-line programming system, this thesis started to work on solutions for dexterous manipulation, trying to arrive at a more efficient and user-friendly solution.

Robot teaching via indirect collaboration with simple motion mapping functions might not work for complex tasks such as following surface curvatures with contact interactions and collision-free motions. Therefore, this thesis develops a [Fast Calibration and Adjustable Mapping \(FCAM\)](#) framework to cope with the aforementioned issues in indirect collaboration. This framework provides an efficient method for an operator to calibrate the robot pose in the teleoperation system, an adjustable motion mapping method for the operator to guide the robot movements using different scale factors, and a clutching mechanism supports motion remapping while the operators reach the limits of their workspace due to the human body structure. [FCAM](#) provides an operator with a more flexible way to set up and telemanipulate the robot. Instead of mapping human motion to the robot motion through a fixed mapping function, the developed adjustable motion mapping method

allows an operator to use two different telemanipulation modes, large-scale motion guidance, and fine motion tuning:

1. The operator can drive the robot efficiently in a large-scale motion while constraint conditions are loose;
2. When the constraints of robot workspace become tougher and require more precise control, the motion remapping can provide a fine motion mode. For example, approaching a contact surface, or inserting the tape into a groove, the fine motion mode is more useful.

The operation flow of **FCAM** is depicted in Figure 3.9.

Figure 3.9: The operation flow of **FCAM**, where $B1$ and $B2$ are the signals automatically/manually sent from the operator.

Kinematics of FCAM

In this work, the theoretical foundation of **FCAM** is based on the Lie theory [71].

Starting from a **Three-Dimensional (3D)** smooth manifold $\mathcal{M}$ (a topological space with a C^∞ structure), where there is an atlas $\{(\mathcal{U}_i, \varphi_i) | i \in \mathcal{I}\}$ defined on $\mathcal{M}$. That is, for each i in a index set $\mathcal{I}$, there is a homeomorphism mapping $\varphi_i : \mathcal{U}_i \rightarrow \mathbb{R}^3$, where $\mathcal{U}_i$ is an open set that covers $\mathcal{M}$, i.e.

$\bigcup_{i \in \mathcal{I}} \mathcal{U}_i = \mathcal{M}$. In addition, at each point $\mathbf{p} \in \mathcal{M}$, there is a tangent space $\mathcal{T}_{\mathbf{p}}\mathcal{M}$ for describing the velocity of $\mathbf{p}$, says $\mathbf{p} \in \mathcal{T}_{\mathbf{p}}\mathcal{M}$. Let the manifold $\mathcal{M}$ be a matrix Lie group and there is an identity element $\mathbf{I} \in \mathcal{M}$, then the tangent space $\mathcal{T}_{\mathbf{I}}\mathcal{M}$ contains the Lie algebra of $\mathcal{M}$. Furthermore, the exponential map $\exp_{\mathbf{I}}(\cdot) : \mathcal{T}_{\mathbf{I}}\mathcal{M} \rightarrow \mathcal{M}$ as well as its inverse, the logarithmic map $\ln_{\mathbf{I}}(\cdot) : \mathcal{M} \rightarrow \mathcal{T}_{\mathbf{I}}\mathcal{M}$, are defined to preserve the distance between $\mathcal{M}$ and $\mathcal{T}_{\mathbf{I}}\mathcal{M}$. For convenience, this thesis sets the point $\mathbf{I}$ as the origin such that $\exp(\cdot) = \exp_{\mathbf{I}}(\cdot)$ and $\ln(\cdot) = \ln_{\mathbf{I}}(\cdot)$.

In this paper, this thesis uses F with a corresponding subscript to represent the *frame*. Given two frames on $\mathcal{M}$, says F_{ε} and F_{ϑ} . The rotation matrix and translation from F_{ε} to F_{ϑ} are denoted ${}^{\varepsilon}\mathbf{R}_{\vartheta} \in SO(3)$ and ${}^{\varepsilon}\mathbf{p}_{\vartheta} \in \mathbb{R}^3$, respectively. Based on ${}^{\varepsilon}\mathbf{R}_{\vartheta}$ and ${}^{\varepsilon}\mathbf{p}_{\vartheta}$, a [Homogeneous Transformation Matrix \(HTM\)](#) (or a pose) is defined as

$${}^{\varepsilon}\mathbf{T}_{\vartheta} = \begin{bmatrix} {}^{\varepsilon}\mathbf{R}_{\vartheta} & {}^{\varepsilon}\mathbf{p}_{\vartheta} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \in SE(3). \quad (3.13)$$

From the Lie algebras point of view, the rotation ${}^{\varepsilon}\mathbf{R}_{\vartheta}$ and the pose ${}^{\varepsilon}\mathbf{T}_{\vartheta}$ can be obtained via the exponential map as follows:

$$\begin{aligned} {}^{\varepsilon}\mathbf{R}_{\vartheta} &= \exp({}^{\varepsilon}\mathbf{\Phi}_{\vartheta} \in \mathfrak{so}(3)) \\ {}^{\varepsilon}\mathbf{T}_{\vartheta} &= \exp({}^{\varepsilon}\mathbf{\Xi}_{\vartheta} \in \mathfrak{se}(3)) \end{aligned} \quad (3.14)$$

The lie algebra, $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$, are calculated based on the skew-symmetric operator $(\cdot)^{\wedge}$:

$$\begin{aligned} {}^{\varepsilon}\mathbf{\Phi}_{\vartheta} &= {}^{\varepsilon}\mathbf{\phi}_{\vartheta}^{\wedge} \\ {}^{\varepsilon}\mathbf{\Xi}_{\vartheta} &= {}^{\varepsilon}\mathbf{\xi}_{\vartheta}^{\wedge} \end{aligned} \quad (3.15)$$

where ${}^\varepsilon\phi_\vartheta \in \mathbb{R}^3$ and ${}^\varepsilon\xi_\vartheta \in \mathbb{R}^6$. Besides, we have

$${}^\varepsilon\xi_\vartheta = \begin{bmatrix} {}^\varepsilon\rho_\vartheta \\ {}^\varepsilon\phi_\vartheta \end{bmatrix}. \quad (3.16)$$

In contrast, the logarithmic map can provide an inverse method to get $SO(3)$ and $SE(3)$ from $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$, respectively. With the inverse of the skew-symmetric operator $(\cdot)^\vee$, defined as

$$\begin{aligned} {}^\varepsilon\phi_\vartheta &= {}^\varepsilon\Phi_\vartheta^\vee \\ {}^\varepsilon\xi_\vartheta &= {}^\varepsilon\Xi_\vartheta^\vee, \end{aligned} \quad (3.17)$$

we have

$$\begin{aligned} {}^\varepsilon\phi_\vartheta &= \ln({}^\varepsilon\mathbf{R}_\vartheta)^\vee \\ {}^\varepsilon\xi_\vartheta &= \ln({}^\varepsilon\mathbf{T}_\vartheta)^\vee. \end{aligned} \quad (3.18)$$

Human Motion Capture

Given a robot in an unstructured environment where the world frame F_W and the robot base frame F_R is commonly not identical, an $\text{HTM} {}^W\mathbf{T}_R$ exists between F_W and F_R . A common robot setup for HRI is a direct installation on the floor. In this case, the parameters that affect ${}^W\mathbf{T}_R$ are the translations on the X, Y and Z direction of F_W , and a yaw angle rotates along the Z-axis of F_W . To have the information of these four parameters, a tedious process for accurate calibration is required. Otherwise, the subsequent implementation of OLP will fail due to the calibration errors. Besides, even though the calibration has been done, unexpected robot motion, such as collision, may cause the robot base to drift from the previous calibration.

Unlike the strict requirement that OLP needs, the introduction of human-in-the-loop can alleviate the traditional calibration issues thanks to the human capability of decision making. In online programming via telemanip-

ulation, human operators can manipulate the robot from any initial robot configuration to a target through the observation. Also, if the manipulation is about to fail, human operators can use their perception to efficiently adjust the robot motion for correction. Given these, the requirement of accurate information on the translations becomes trivial. On the other hand, under the indirect [HRC](#), the used sensor F_S plays an important role in describing the human motion, instead of F_W . Hence, this work proposes a method for human motion capture in two different manners. This thesis considers two cases of human motion: the *model-less motion* and the *model-based motion*. Simply speaking, the model-less motion indicates that the human model is not required, while the model-based motion needs a human model for human motion capture. Therefore, the reference frame for describing the model-less motion is generally in F_R (or F_S), while the model-based motion is in F_B . For the methods of the two cases, this thesis depicted them separately below:

Theorem 3.3.1 (Model-Less Motion Capture). Assume that there is a sensor F_S that can capture human motion ${}^S\mathbf{T}_H$. By recording the ${}^S\mathbf{T}_H$ while aligning the X-axis of F_S with the X-axis of F_R , then ${}^R\mathbf{T}_H$ can be obtained.

Proof. While aligning the X-axis of F_S with the X-axis of F_R , the sensor generates a sensory [HTM](#) ${}^S\mathbf{T}_{calib}$ of which the rotation matrix is equal to ${}^S\mathbf{R}_R$. Therefore, we can describe human motion as

$${}^R\mathbf{T}_H = {}^R\mathbf{T}_S {}^S\mathbf{T}_H = {}^S\mathbf{T}_{calib}^{-1} {}^S\mathbf{T}_H. \quad (3.19)$$

However, only the rotation of ${}^S\mathbf{T}_{calib}$ is required, and we want to eliminate the errors along other axes of F_S , i.e. along X- and Y-axis. Hence, this thesis computes the yaw angle ψ of ${}^S\mathbf{T}_{calib}$ through [\[72\]](#), and then replace

the ${}^S\mathbf{T}_{calib}$ with a corrected [HTM](#) ${}^S\mathbf{T}'_{calib}$, where

$${}^S\mathbf{T}'_{calib} = \exp\left(\left[\begin{array}{c} \mathbf{0}_{5 \times 1} \\ \psi \end{array}\right]^\wedge\right) \quad (3.20)$$

.

□

Theorem 3.3.2 (Model-Based Motion Capture). Assume that there is a set of sensors F_{S_i} that can capture part of human's arm motion ${}^S\mathbf{R}_{L_i}$. By recording the ${}^W\mathbf{R}_{S_i}$ while aligning the X-axis of F_B with an orientation in F_W , says ψ , then ${}^B\mathbf{T}_H$ can be obtained when the human model and ψ are available.

Proof. Let each component of a human model be $L_i \in L$. The relative pose of sensor F_{S_i} fixed on a F_{L_i} can be calculated by

$${}^{L_i}\mathbf{R}_{S_i} = {}^{L_i}\mathbf{R}_W {}^W\mathbf{R}_{S_i}, \quad (3.21)$$

where

$${}^{L_i}\mathbf{R}_W = {}^{L_i}\mathbf{R}_B {}^B\mathbf{R}_W. \quad (3.22)$$

With the availability of the human model and a corresponding human pose used in the registration process, we have ${}^{L_i}\mathbf{R}_B = \mathbf{1}_{3 \times 3}$. Besides, having the information of ψ , we can easily know that

$${}^W\mathbf{R}_B = \exp\left(\left[\begin{array}{c} \mathbf{0}_{2 \times 1} \\ \psi \end{array}\right]^\wedge\right). \quad (3.23)$$

For capturing the motion of each part of human motion, we can calculate the motion relative to the registered data ${}^{L_i}\mathbf{R}_{S_i}$:

$${}^B\mathbf{R}_{L_i} = {}^B\mathbf{R}_W {}^W\mathbf{R}_{S_i} {}^{L_i}\mathbf{R}_{S_i}^{-1}. \quad (3.24)$$

Finally, based on the human model again, the hand motion relative to F_B can be obtained:

$${}^B\mathbf{T}_H = \begin{bmatrix} {}^B\mathbf{R}_{L_3} & {}^B\mathbf{R}_{L_1}\mathbf{p}_1 + {}^B\mathbf{R}_{L_2}\mathbf{p}_2 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}, \quad (3.25)$$

where $\mathbf{p}_1 \in \mathbb{R}^3$ and $\mathbf{p}_2 \in \mathbb{R}^3$ are the dimensions of the human's upper arm and forearm. Note that F_{L_1} denotes the human's upper arm, F_{L_2} denotes the human's forearm, and F_{L_3} denotes the human's hand. $\square$

In spite of the slight error in the manual process of the two methods, the operator can use their space cognition to guide the robot and unconsciously compensate for this error in their manipulation.

(a) Model-less motion retargeting

(b) Model-based motion retargeting

Figure 3.10: Illustration of motion mapping.

Normal Motion Retargeting

Due to the introduction of timestamps to differentiate when the motion is generated, this thesis adds the timestamps in the subscript of all the kinematics symbols. In order to establish the mapping from the human motion to the robot motion, as illustrated in Figure 3.10, a mapping function for motion retargeting, represented as $f_M : SE(3) \times SE(3) \times SE(3) \rightarrow SE(3)$,

is required to figure out the desired target pose of the robot end-effector:

$${}^R\mathbf{T}_{E,t} = \begin{cases} f_M({}^R\mathbf{T}_{H,t}, {}^R\mathbf{T}_{H,0}, {}^R\mathbf{T}_{E,0}), & \text{for model-less case} \\ f_M({}^B\mathbf{T}_{H,t}, {}^B\mathbf{T}_{H,0}, {}^R\mathbf{T}_{E,0}), & \text{for model-based case} \end{cases} \quad (3.26)$$

To determine the difference of pose, this thesis discusses a case first. Let an object frame F_O on $\mathcal{M}$ and let its initial pose be defined as ${}^W\mathbf{T}_{O,0}$. The pose that object moves to at time t is denoted ${}^W\mathbf{T}_{O,t}$. To ensure the closure of matrix Lie group, the difference of poses which is represented by

$$\delta^W\mathbf{T}_{O,0:t} = {}^W\mathbf{T}_{O,t} - {}^W\mathbf{T}_{O,0} \quad (3.27)$$

cannot be employed. Nevertheless, the nature of lie group gives the difference of poses as follows:

$$\delta^W\mathbf{T}_{O,0:t} = {}^W\mathbf{T}_{O,t}^{-1} {}^W\mathbf{T}_{O,0}, \quad (3.28)$$

where the distance between the two poses can be calculated by

$$\|\delta^W\boldsymbol{\xi}_O\| = \sqrt{\langle \ln({}^W\mathbf{T}_{O,t}^{-1} {}^W\mathbf{T}_{O,0}), \ln({}^W\mathbf{T}_{O,t}^{-1} {}^W\mathbf{T}_{O,0}) \rangle}. \quad (3.29)$$

Based on this case, the mechanism of f_M in model-less motion can be summarized as

$$\delta^R\boldsymbol{\Xi}_{E,0:t} = k_s \delta^R\boldsymbol{\Xi}_{H,0:t}, \quad k_s \in \mathbb{R}^+. \quad (3.30)$$

Hence, we can get the scheme of f_M in model-less motion mapping:

$${}^R\mathbf{T}_{E,t} = \exp\left({}^R\boldsymbol{\xi}_{E,t}^\wedge\right) = \exp\left((k_s \delta^R\boldsymbol{\xi}_{H,0:t} + {}^R\boldsymbol{\xi}_{E,0})^\wedge\right). \quad (3.31)$$

For f_M in model-based motion, this thesis uses a similar method based on the following mechanism:

$$\delta^R\boldsymbol{\Xi}_{E,0:t} = k_s \delta^B\boldsymbol{\Xi}_{H,0:t}, \quad k_s \in \mathbb{R}^+. \quad (3.32)$$

Although the reference frames for describing the human motion F_B and the robot motion F_R are different, Lie group equipped with a left-invariant Riemannian metric $\langle \cdot, \cdot \rangle$ can ensure that under the distant preserving condition ($k_s = 1$),

$$\langle {}^R\Xi_{E,t}, {}^R\Xi_{E,0} \rangle_R = \langle \lambda_{BR} {}^B\Xi_{H,t}, \lambda_{BR} {}^B\Xi_{H,0} \rangle_R = \langle {}^B\Xi_{H,t}, {}^B\Xi_{H,0} \rangle_S, \quad (3.33)$$

where $\lambda_{BR} : \mathcal{T}_B\mathcal{M} \rightarrow \mathcal{T}_R\mathcal{M}$. Therefore, we can get the scheme of f_M in model-based motion mapping:

$${}^R\mathbf{T}_{E,t} = \exp\left({}^R\boldsymbol{\xi}_{E,t}^\wedge\right) = \exp\left((k_s\delta^S\boldsymbol{\xi}_{H,0:t} + {}^R\boldsymbol{\xi}_{E,0})^\wedge\right). \quad (3.34)$$

Both (3.31) and (3.34) provides a continuous motion mapping from the human motion to the robot motion and ensures the closure of the resultant $SE(3)$ group. However, these methods are based on the interpolation of the Lie group, on which a Riemannian structure is defined. This indicates that the resultant interpolation of the translation of $SE(3)$ follows the geodesic on the Riemannian manifold, which leads to an unexpected trajectory on the Euclidean space where the operators exist in. With the increase of the distance of the relative poses, the increasing curvature of the geodesic makes the interpolation infeasible due to the physical safety issue in the Euclidean space. Hence, the interpolation of translation motion should follow the shortest path on the Euclidean manifold. That is, the traditional interpolation scheme is expected to be recovered:

$$\begin{cases} \delta^R\mathbf{p}_{E,0:t} = k_s\delta^R\mathbf{p}_{H,0:t}, & \text{for model-less case} \\ \delta^R\mathbf{p}_{E,0:t} = k_s\delta^B\mathbf{p}_{H,0:t}, & \text{for model-based case} \end{cases} \quad k_s \in \mathbb{R}^+ \quad (3.35)$$

For increasing the flexibility of f_M , this thesis uses $k_p \in \mathbb{R}^+$ as the scaling factor for translation mapping and $k_r \in \mathbb{R}^+$ as the scaling factor for rotation

mapping, instead of using previous k_s .

Mirrored Motion Retargeting

Under some circumstances, normal motion retargeting may become infeasible due to the limitations of the environment. For example, the telemanipulation of a dual-arm robot without telepresence requires an operator to manipulate the robot face-to-face. If the operating position is on the sides of the robot, the operator will feel uncomfortable with the observation of the robot's workspace. Besides, to avoid the reachable space of the robot manipulator, the operating position may be more distant from the robot base, causing the operator to feel hard to transfer the robot dual-arm manipulation because of the unclear observation.

Given these, this work proposes an on-site teaching method based on mirrored motion retargeting for operators to telemanipulate a dual-arm robot more efficiently. *Mirrored motion* is defined as follows:

Definition 3.3.1 (Mirrored Motion). Given two frames F_O and $F_{O'}$, if an [HTM](#) $\mathbf{T}_{rf}$ with $\det(\mathbf{T}_{rf}) = -1$ exists such that ${}^O\mathbf{T}_{O'} = \mathbf{T}_{rf}$, then $F_{O'}$ generates a mirrored motion corresponding to F_O .

There are many forms of mirrored motion based on the selection of the reference planes for reflection. Without losing generality, this work considers the setup of using the Y-Z plane for generating the mirrored motion so that the X-axis of the robot base frame can be utilized again, where the underlying principle will be further discussed through afterward derivation:

$$\mathbf{T}_{rf} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.36)$$

Nevertheless, the matrix $\mathbf{T}_{rf}$ is not a $SE(3)$ group as its determinant does not equal to 1, making the defined operation on $SE(3)$ cannot be employed under the existence of $\mathbf{T}_{rf}$. Therefore, this work provides another solutions to cope with the kinematics issue of mirrored motion.

Similar to the normal motion retargeting, the two cases of human motion capture are studied again.

Theorem 3.3.3 (Model-Less Mirrored Motion Capture). Assume that there is a sensor F_S that can capture human motion ${}^S\mathbf{T}_H$ while aligning the X-axis of F_S with the reversed X-axis of F_R , then a mirrored motion ${}^R\mathbf{T}_E$ corresponding to the ${}^B\mathbf{T}_H$ can be obtained.

Proof. While aligning the X-axis of F_S with the reversed X-axis of F_R , the sensor generates a sensory HTM ${}^S\mathbf{T}_{calib}$ of which the rotation matrix is equal to

$$\exp\left(\begin{bmatrix} \mathbf{0}_{2 \times 1} \\ \pi \end{bmatrix}^\wedge\right) {}^S\mathbf{R}_R, \quad (3.37)$$

implying that a virtual human body frame F_B is established in front of the robot. Thus, the captured human motion can be described in F_B :

$${}^B\mathbf{T}_H = {}^S\mathbf{T}_{calib} {}^S\mathbf{T}_H. \quad (3.38)$$

To make the rotation movements in a mirrored manner, a mirrored human body frame $F_{B'}$ is created, where the captured rotation motion can be transformed by

$${}^{B'}\mathbf{R}_H = \exp\left(\begin{bmatrix} 0 \\ \pi \\ 0 \end{bmatrix}^\wedge\right) {}^B\mathbf{R}_H {}^B\mathbf{R}_H. \quad (3.39)$$

The relationship between the aforementioned frames is illustrated in Figure 3.11, where we can see the rotations in F_R and $F_{B'}$ are mirrored (corresponding to Definition 3.3.1). In addition, the mirrored translation in F_R

can be achieved by the movement with a reversed direction along each axis of $F_{B'}$. $\square$

Figure 3.11: Illustration of model-less mirror mapping.

Theorem 3.3.4 (Model-Based Mirrored Motion Capture). Assume that there is a set of sensors F_{S_i} that can capture part of human's arm motion ${}^S\mathbf{R}_{L_i}$. By recording the ${}^W\mathbf{R}_{S_i}$ while aligning the X-axis of F_B with an orientation in F_W , says ψ , then a mirrored motion ${}^R\mathbf{T}_E$ corresponding to the ${}^B\mathbf{T}_H$ can be obtained when the human model and ψ are available.

Proof. With a similar manner to Theorem 3.3.3, the model-based mirrored motion capture is achieved through the mirrored frame of each sensor $F_{S'_i}$:

$${}^W\mathbf{R}_{S'_i} = \exp \left(\begin{bmatrix} 0 \\ \pi \\ 0 \end{bmatrix} \wedge \right) {}^W\mathbf{R}_{S_i}. \quad (3.40)$$

To reconstruct the human motion, a mirrored human model is used, where the dimensions of the human's upper arm and forearm are negative. Although the human motion is described in F_B under this manner, it is reconstructed based on the mirrored motion that captured by sensors. In this

manner, the procedure of Theorem 3.3.2 can be followed for obtaining the mirrored motion of ${}^B\mathbf{T}_H$. $\square$

3.4 Performance Evaluation

In manufacturing, the common problem of identifying suitable robotic configuration (the type of robot, actuators, sensors, etc.) for a given task is a difficult issue for manufacturers. Different robotic configurations and corresponding task plans lead to huge differences in performance. Furthermore, these differences are further magnified due to repetition. Performance study of one or multiple robotic configurations in handling a task has been an important but not well-addressed issue in the community. A quantitative assessment allows robot manufacturers or potential operators to make an objective decision based on the results of the assessment, such as deciding on the choice of a robotic system to purchase and possible return on investment. This would lead to improved efficiency and effectiveness in the deployment of robotic systems in the industry.

Research in performance assessment at the economy-level and robot-level has been well explored in recent years, as listed in Table 3.1. In general, performance study at robot-level tends to focus on the design of the robotic mechanism, repeatability and resolution [44] [73] while analysis at economy-level tends to focus on the investment-driven considerations such as cash flow, risk, and capital recovery factor [47]. To connect economy-level and robot-level, assessment at task-level is necessary. Criteria at task-level bridges the gap by looking at factors that cover the perspectives of both sides. For example, improving productivity at task-level through selecting a more suitable type of robot at the robot-level for a given task may bring manufacturers higher capital recovery factors at the economy-level.

Table 3.1: Performance criteria at robot-level and economy-level.

Robot-Level	Economy-Level
Type of structure	Cost
Payload	Value
Joint speed and range	Interest rate
Repeatability	Risk
Resolution	Capital recovery factor
Lifetime	Payback period
etc.	etc.

To evaluate the task-level performance in collaborative robotics, this thesis develops a robot-centered performance assessment approach to quantify the performance of a set of robotic configurations and their corresponding task plans proposed by manufacturers. This approach is not only compatible with the aforementioned three collaborative patterns but also can be utilized in further cross-analysis for providing manufacturers quantitative criteria on system deployment. The proposed steps for performance assessment at task-level are shown in Figure 3.12.

Figure 3.12: Steps of performance assessment at task-level.

3.4.1 Robot-Centered Performance Assessment

Continuous Time Markov Chain The model for performance assessment is established based on Markov Chains [74]. The most remarkable characteristics of Markov Chains is its memoryless property, defined as follows [75]:

Definition 3.4.1 (Markov Chain). Given a temporally homogeneous stochastic process $X = (X_t, t \in T)$ in a probability space $(\Omega, \mathcal{H}, \Pr)$, for a totally ordered index set T , Markov Chains is a stochastic process on a measurable finite state space $\mathcal{S} \in \Omega$ satisfying

$$\Pr(X_t = j \in \mathcal{S} \mid \mathcal{H}_s \in \mathcal{H}) = \Pr(X_t \in \mathcal{S} \mid X_s \in \mathcal{S}), \quad (3.41)$$

where $s, t \in T$ with $s < t$, and $\mathcal{H}_s = \sigma(X_o, \dots X_s)$ is a collection of known events.

Definition 3.4.1 indicates that the future can be predicted by the availability of the present event without past events. The transition probability of a Markov Chains from state i to state j , between index s and index t , can be formulated by [75]:

$$\Pr(X_t = j \in \mathcal{S} \mid X_s = i \in \mathcal{S}) = p_{ij}(s, t). \quad (3.42)$$

Under the circumstance of continuous-time, the probabilistic graph of Markov Chains is generally expressed by a transition rate derived by Kolmogorov differential equations [76]:

$$\begin{cases} \mathbf{P}(s, t) = [p_{ij}(s, t)] = \exp\left(\int_s^t \mathbf{Q}(\zeta) d\zeta\right) \\ \mathbf{Q}(t) = [q_{ij}(t)] = \lim_{h \rightarrow 0} \frac{\mathbf{P}(t, t+h) - \mathbf{I}}{h} \end{cases}, \quad (3.43)$$

where $\mathbf{P}$ is so-called transition probability matrix and $\mathbf{Q}$ is transition rate

matrix. The relationship between the elements of $\mathbf{P}$ and $\mathbf{Q}$ is determined through the infinitesimal property formulated by little-o notation as

$$\begin{cases} q_{ii}(t) = p_{ii}(t, t+h) - 1 + o(h) \\ q_{ij}(t) = p_{ij}(t, t+h) + o(h), \quad i \neq j \end{cases}, \quad (3.44)$$

where h is an infinitesimal time increment. The relationship indicates the following property in a transition probability matrix $\mathbf{P}(t)$ and $\mathbf{Q}(t)$

$$\begin{cases} \sum_j p_{ij}(s, t) = 1, \quad \forall i, s \leq t \\ \sum_j p_{ij}(t) = 0, \quad \forall i \end{cases}. \quad (3.45)$$

Probabilistic Dependencies and Space Mapping Combining information vector and the proposed task representation on the [Probabilistic Graphical Model \(PGM\)](#) which is with characteristics of [Continuous Time Markov Chain \(CTMC\)](#), a linear equation to formulate any existed arc (probabilistic dependency) among its end vertices (sub-plans) is given by

$$q_{ij}(t) \sum_{\mathcal{A}_k \in \mathcal{SV}_i} = \begin{cases} \prod_{\mathcal{A}_k \in \mathcal{SV}_i} r_k, & j = i + 1 \\ 1 - \prod_{\mathcal{A}_k \in \mathcal{SV}_i} r_k, & \text{otherwise} \end{cases}. \quad (3.46)$$

After establishing the probabilistic dependencies between each ordered pair of vertices in [PGM](#), this thesis subsequently constructs a mapping from a task plan space $\mathcal{P}$ to a [CTMC](#) model with a [CTMC](#) state space $\mathcal{S}$. Firstly, an assumption that a studied task plan consists of n manipulation actions assigned ${}^M\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_n\}$ is made. Then, the actions $\mathcal{A}_0$ and $\mathcal{A}_{n+1}$, are set to be the setup ${}^S\mathcal{A}$ and the troubleshooting actions ${}^R\mathcal{A}$, respectively. Hence, there is an action set $\mathcal{A} = \{\mathcal{A}_0, \dots, \mathcal{A}_{n+1}\}$ with $|\mathcal{A}| = n + 2$, which is composed of a manipulation action set ${}^M\mathcal{A}$ and a default action set ${}^D\mathcal{A} = {}^S\mathcal{A} \cup {}^R\mathcal{A} = \{\mathcal{A}_0, \mathcal{A}_{n+1}\}$. Secondly, it is assumed that there are l

detection actions among these actions, namely $|\mathcal{A}^d| = l$. This means that there is a sub-plan set $\mathcal{SP} = \{\mathcal{SP}_1 \cdots \mathcal{SP}_{l+1}\}$ in a plan $\mathcal{P}_i$, where all the elements in $\mathcal{SP}$ are proper sub-plans decomposed by Theorem 3.2.4. The mapping is constructed by the following three functions:

- $f_d : {}^D\mathcal{A} \rightarrow \mathcal{S}$ is injective-only.
- $f_a : {}^M\mathcal{A} \rightarrow \mathcal{SP}$ is surjective-only.
- $f_p : \mathcal{SP} \rightarrow \mathcal{S}$ is injective-only.

Then, we can use above three functions to construct the mapping by:

- $f_d(\mathcal{A}_0) = \mathcal{S}_0$ and $f_d(\mathcal{A}_{n+1}) = \mathcal{S}_{l+2}$.
- $f_a(\mathcal{SP}_j)$, if $\mathcal{A}_i \in \mathcal{SP}_j$.
- $f_p(\mathcal{SP}_i) = \mathcal{S}_i$.

The developed mapping implies that no matter how many actions there are in a task plan, it will eventually lead to $|\mathcal{S}| = |\mathcal{SP}| + 2$. Figure 3.13 shows an example of this mapping with $n = 4$ and $l = 1$. The detection action is $\mathcal{A}_2$ and the state space of this mapping is with $|\mathcal{S}| = |\mathcal{SP}| + 2 = 4$.

Figure 3.13: An example of mapping from a task plan $(n, l) = (4, 1)$ to a state space of CTMC, where $\mathcal{A}_2 \in \mathcal{A}^d$.

Performance Assessment Model Based on this mapping method, a generalized CTMC model with a state space $\mathcal{S} = \{\mathcal{S}_0, \dots, \mathcal{S}_{l+2}\}$ can be established, where each state has a corresponding physical meaning:

- $\mathcal{S}_0$: The robotic system is being set up.
- $\mathcal{S}_{l+2}$: The robotic system is under troubleshooting.
- $\mathcal{S}_i, 1 \leq i \leq l+1$: The robotic system is executing task normally.

Figure 3.14 shows the generalized CTMC model. Note that there is no failure in the last sub-plan because no matter what the result of this sub-plan returns, the manufactured product will be transmitted out of the studied robotic system by the assumptions made previously, thereby nothing will influence the next sub-plan which will be executed in the new episode of the task. Based on (3.46), the transition rate matrix $\mathbf{Q} = [q_{i,j}] \in \mathbb{R}^{(l+3) \times (l+3)}$

can be expressed by

$$\left\{ \begin{array}{ll} q_{1,2} = \frac{1}{t_1}, & q_{1,1} = -\frac{1}{t_1}, \\ q_{(l+3),1} = \frac{1}{t_{l+3}}, \\ q_{(l+3),(l+3)} = -\frac{1}{t_{l+3}}, \\ q_{(l+2),2} = \frac{1}{\sum_{\mathcal{A}_k \in \mathcal{SV}_n} t_k}, \\ q_{(l+2),(l+2)} = -\frac{1}{\sum_{\mathcal{A}_k \in \mathcal{SV}_n} t_k}, \\ q_{i,i+1} = \frac{\prod_{\mathcal{A}_k \in \mathcal{SV}_{i-1}} r_k}{\sum_{\mathcal{A}_k \in \mathcal{SV}_{i-1}} t_k}, & 2 \leq i \leq l+1, \\ q_{i,(l+3)} = \frac{1 - \prod_{\mathcal{A}_k \in \mathcal{SV}_{i-1}} r_k}{\sum_{\mathcal{A}_k \in \mathcal{SV}_{i-1}} t_k}, & 2 \leq i \leq l+1, \\ q_{i,i} = -\frac{1}{\sum_{\mathcal{A}_k \in \mathcal{SV}_{i-1}} t_k}, & 2 \leq i \leq l+1, \\ q_{i,j} = 0, & \text{otherwise} \end{array} \right. \quad (3.47)$$

In order to assess the performance (i.e. the information vector), we need to conduct a steady state analysis based on the proposed **CTMC** model. A vector $\boldsymbol{\pi} = (\pi_0, \dots, \pi_{l+2})^T$, with properties $\pi_i \leq 0, \forall \pi_i \in \boldsymbol{\pi}$ and $\|\boldsymbol{\pi}\|_1 = 1$, is said to be the probability of each state of a **CTMC**. Furthermore, if π is under steady state, then $\boldsymbol{\pi}$ is unique and can be obtained by [77]:

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi} \mathbf{Q} = 0. \quad (3.48)$$

For each element in (3.48), we have:

$$\pi_i q_{i,i} + \sum_{i \neq j} \pi_{ij} q_{i,j} = 0, \forall i. \quad (3.49)$$

By solving (3.49), the probability of each state in the proposed **CTMC** can be calculated. This indicates that we can obtain the **Average Product Rate**

(APR):

$$APR = \pi_{l+1} q_{l+2,2} \prod_{\mathcal{A}_k \in \mathcal{SV}_n} r_k = \pi_{l+1} \frac{\prod_{\mathcal{A}_k \in \mathcal{SV}_n} r_k}{\sum_{\mathcal{A}_k \in \mathcal{SV}_n} t_k}. \quad (3.50)$$

Figure 3.14: Proposed CTMC model for a robotic task plan.

The outputs of the proposed approach can be defined as:

1. APR: The APR indicates the production capability of a robotic system under a certain task plan. Intuitively speaking, the higher the APR is, the better the robotic system and the corresponding plan is.
2. Information vector: The information vector not only provides the necessary parameters for CTMC model, but also works for performance assessment at the economy-level.

3.4.2 Connection to Assessment at Economy-Level

This section presents a discussion on assessment at the economy-level, providing manufacturers with methods to select solutions among available options. Manufacturers can refer to this regarding determining a suitable

robotic configuration, corresponding plan, and strategy for higher cost-effectiveness.

Assuming the estimated profit of an accomplished task is $\bar{M}$ and the estimated cost per second of the studied robotic system (this value depends on the robot, including installation, operation, maintenance, etc.) is C_r . Combining the result of [APR](#) and information vector, the estimated cash flow $\hat{F}$ can be measured by:

$$\hat{F} = APR \cdot \bar{M} - (C_r + \sum_{i=1}^{|SP|} \prod_{\mathcal{A}_k \in \mathcal{SV}_i} (1 - r_k) C_i) \quad (3.51)$$

where C_i is estimated rework cost per second for $\mathcal{SV}_i$.

Once the cash flow is estimated, manufacturers can further estimate the present value PV about this robotic system by [\[47\]](#):

$$PV = \sum_{t=0}^{N_t} \frac{\hat{F}_t}{(1 + \bar{r})^t} \quad (3.52)$$

where $\hat{F}_t$ refers to the estimated cash flow $\hat{F}$ in period t , depending on the estimated unit profit and other costs in [\(3.51\)](#); N_t is the total number of a considered time periods; $\bar{r}$ is fractional interest rate due in each studied period.

Chapter 4

Robot-Robot Collaboration: Peg-in-Hole

Given a robotic manipulation task, decision on which robotic configuration (robotic system and necessary peripherals, including assistant tools, sensor systems, and so on.) to use and evaluating performance of the solution remains as an open, challenging but a meaningful problem.

Starting from productivity which is one of the main concerns for manufacturers, this chapter introduces the use of the developed methodology for the quantitative assessment of non-collaboration and [RRC](#) paradigms. A vision-based [Peg-in-Hole \(PIH\)](#) case with the implementation of the proposed task representation for performance quantification of different task plans and robotic configurations is studied to validate the effectiveness of the developed methodology for comparing the productivity of different robotic configurations and evaluating the worthiness of the updated solution of a sub-task through observing improvements in productivity. Implemented and tested on the studied tasks with different levels of difficulty through different collaborative forms on a dual-arm robot, the methodology has demonstrated encouraging results. Also, a systematic cross analysis is shown for assisting manufacturers to determine their deployment plans. Finally, the connection

of performance assessment from task-level to economy-level is presented.

4.1 Introduction

Dual-arm industrial robots are increasingly considered as a promising trend of industrial automation [1, 2] due to a range of characteristics: (1) the anthropomorphic manipulators provide operators with more intuitive way to manipulate, (2) the nature of their kinematics make stiffness adjustable when both arms are contacting a common object, (3) high redundancy improves the flexibility and versatility in task execution, (4) synchronized manipulation shortens cycle time. Based on these characteristics, dual-arm robots are therefore widely considered as the mainstream of industrial robots in the next generation and have been employed in many manufacturing and research settings. Unlike single-arm robots, emphasis on dual-arm robots has always been the flexibility of themselves. Current literature on assembly tasks with dual-arm robots has been studied. In [78], a gearbox assembly task is demonstrated with a dual-arm robot by a pre-programmed method. In [79] [80], small part assembly tasks are implemented with vision-based dual-arm robots. In [81], the anthropomorphic feature of dual-arm robots is expressed to implement assembly tasks through PbD paradigm. The literature above shows that dual-arm robots are feasible in realistic assembly applications.

Nevertheless, there is a critical issue that: “should a practitioner replace a single-arm robot with dual-arm robot for higher profits in the industry?” This is an intriguing issue worth deep consideration. Being different from the performance study of various industrial robots and robotic skills [56] [53] or motion planning algorithms for specific manipulation [82], there have been very few studies on the comparison between single-arm robots and dual-arm robots at task-level.

To achieve the quantitative assessment for a robotic system at the task level, this thesis develops a vision-based dual-arm robotic system for a [PIH](#) assembly task. A two-stage vision-based approach is proposed to cope with the uncertainties in the task. With the combination of the database, the robotic system can finish a [PIH](#) assembly task adaptively. Two collaborative patterns are used in this task: Non-Collaboration (single-arm manipulation) and [RRC](#) (dual-arm manipulation), where an internal collaborative form is employed in [RRC](#). Later on, the developed representation is used to model the task plans under different robotic configurations. Besides, this chapter also shows the utilization of the established model to quantify the performance at the task-level. Finally, practitioners can conduct cross-analysis: data validation, the influence of strategies, estimation of a new strategy, and dominance of robotic configuration, to measure their deployment. Furthermore, the criteria at the economy-level can be achieved by the proposed assessment method.

4.2 Methodology

4.2.1 Perception

A two-stage vision-based localization approach is proposed herein. Firstly, rough positions of all objects presented in front of the robot can be obtained through a [RGB-D](#) camera on the robot head. With this information, the robot can move and use its hand-tip camera to closely detect the exact pose of the object that is going to be manipulated. Based on this two-stage approach, the robot can manipulate the objects precisely. The details of this approach will be revealed in the following paragraphs.

Global Detection

The goal of global detection is to search the whole workspace plane and find the rough position of every assembly part presented. In this step, if any of the assembly parts are missing, the next step will not be proceeded by the robotic system. This approach is realized by [Point Cloud Library \(PCL\)](#) [83] with the utilization of point cloud data $\mathbf{P}_S$, where the algorithm is given in Algorithm 1.

Algorithm 1 Global Detection

```

1: procedure GLOBAL DETECTION( $\mathbf{P}_S$ )
2:   Initialize sensor's pose  ${}^W\mathbf{T}_S$  and workspace  $\mathcal{W}$ 
3:   Initialize searching radius  $r_s$  and min. neighborhoods  $n_r$ 
4:   Initialize clustering tolerance  $\epsilon$  and max./min. neighborhoods in a
    cluster  $n_{c,max}/n_{c,min}$ 
5:    $\mathcal{L}_o \leftarrow \emptyset$  ▷ A list of rough positions
6:   for each point  $\mathbf{p}_{S_i} \in \mathbf{P}_S$  do
7:      $\begin{bmatrix} \mathbf{p}_{W_i} \\ 1 \end{bmatrix} = {}^W\mathbf{T}_S \begin{bmatrix} \mathbf{p}_{S_i} \\ 1 \end{bmatrix}$  ▷ Transformation from  $F_S$  to  $F_W$ 
8:      $\mathbf{P}_W \leftarrow PassThrough(\mathbf{P}_W, \mathcal{W})$ 
9:      $\mathbf{P}_W \leftarrow RadiusOutlierRemoval(\mathbf{P}_W, r_s, n_r)$ 
10:     $\mathcal{C} = EuclideanClusterExtraction(\mathbf{P}_W, \epsilon, n_{c,max}, n_{c,min})$ 
11:    for each cluster  $\mathcal{C}_k \in \mathcal{C}$  do
12:       $\mathbf{p}_k \in \mathbb{R}^3 = \frac{1}{|\mathcal{C}_k|} \sum_{\mathbf{p}_{W_i} \in \mathcal{C}_k} \mathbf{p}_{W_i}$ 
13:      Insert  $\mathbf{p}_k$  into  $\mathcal{L}_o$ 
  return  $\mathcal{L}_o$ 

```

(a)

(b)

Figure 4.1: Implementation result of global detection. (a) Received image in F_S . (b) Visualization of Algorithm 1, where "WAIST" is the name of F_R .

In our implementation, four assembly parts are presented in front of the

robot. As the result shown in Figure 4.1, the rough positions of these four assembly parts in the reference frame F_W , which is identical to the robot base frame F_R , are successfully captured.

Local Detection

After the completion of global detection, we can obtain the rough position of each assembly part and move the hand-tip camera mounted to our target. Then local detection can assist the robot to acquire the exact pose of the object so that the robot end-effector of the robot can reach the object precisely with collision avoidance. Noted that the realization of local detection should be guaranteed that only one object is presented in the detection range of the hand-tip camera. This approach is realized by OpenCV [84] with the utilization of RGB image data $\mathbf{M}_{RGB}$, where the algorithm is given in Algorithm 2.

Algorithm 2 Local Detection

```

1: procedure LOCAL DETECTION( $\mathbf{M}_{RGB}$ )
2:   Initialize binary threshold  $k_{bin,th}$ 
3:   Initialize pixel center values  $\mathbf{p}_{c0}$ 
4:   Initialize mapping ratio  $k_m$ 
5:    $\mathbf{M}_{Grey} = ConvertBinary(\mathbf{M}_{RGB})$ 
6:    $\mathbf{M}_{Grey} \leftarrow Threshold(k_{bin,th})$ 
7:    $\mathcal{T} = FindContours(Grey)$ 
8:    $\mathcal{T}_{max} = FindLargestArea(\mathcal{T})$ 
9:    $\lambda_{major}, \lambda_{minor} = PCA(\mathcal{T}_{max})$ 
10:   $\Delta x = -k_m(\frac{1}{|\mathcal{T}_{max}|} \sum_{\mathbf{p}_i \in \mathcal{T}_{max}} p_{yi} - p_{c0y})$ 
11:   $\Delta y = -k_m(\frac{1}{|\mathcal{T}_{max}|} \sum_{\mathbf{p}_i \in \mathcal{T}_{max}} p_{xi} - p_{c0x})$ 
12:   $\Delta \theta = \cos^{-1}(\lambda_{minor}^T [\frac{0}{1}])$ 
13:  return  $\Delta x, \Delta y, \Delta \theta$ 

```

The implementation result is presented in Figure 4.2.

Figure 4.2: Implementation result of local detection. (a) Visualization of Algorithm 2, where the brown rectangle is the captured contour, yellow dot is the center of the contour and the red line is the principal axis of [Principal Component Analysis \(PCA\)](#). (b) Result of compensation value.

Finally, the exact position and orientation of an object can be obtained by the compensation values from Algorithm 2:

$$\begin{aligned}
 {}^W\mathbf{p}_O &= \begin{bmatrix} \Delta x + {}^Wp_{Ex} \\ \Delta y + {}^Wp_{Ey} \\ \frac{1}{2}h_{object} \end{bmatrix}, \\
 {}^W\mathbf{R}_O &= \exp\left(\left[\begin{bmatrix} \mathbf{0}_{2 \times 1} \\ \Delta\theta \end{bmatrix}\right]^\wedge\right)
 \end{aligned} \tag{4.1}$$

where h_{object} is the height of the object, which has been stored in the grasping database in advance. Note that the subscription $\cdot_x$, $\cdot_y$, and $\cdot_z$, refers to the values along X-, Y-, and Z-, Axes.

4.2.2 Manipulation

The manipulation process in the [PIH](#) assembly task is combined with a preset database possessing the grasping pose, placing pose and assembly relationship of each assembly part. A framework for the implementation of how to grasp an object with the aforementioned perception approach is

briefly illustrated in Figure 4.3.

Figure 4.3: Framework of manipulation.

Grasping

The grasping pose for an object, says F_O , can be calculated by:

$$\mathbf{T}_{grasp} = {}^R\mathbf{T}_E = ({}^W\mathbf{T}_R)^{-1} {}^W\mathbf{T}_O {}^O\mathbf{T}_E, \quad (4.2)$$

where ${}^W\mathbf{T}_R$ refers to the pose calibration of the robot in its workspace; ${}^W\mathbf{T}_O$ refers to the exact pose of the object, of which the result is from the proposed perception approach; ${}^O\mathbf{T}_E$ is the grasping pose pre-defined in the grasping database. The frame illustration of (4.2) is given in Figure 4.4.

Figure 4.4: Frame illustration of grasping pose.

Assembly

The assembly pose for an object is related to another object. Here, PIH assembly task only considers the simplest case, namely, two objects. Let the frames of peg part and hole part be F_O and $F_{O'}$, respectively. The assembling pose can be calculated by:

$$\mathbf{T}_{assemble} = {}^R\mathbf{T}_E = ({}^W\mathbf{T}_R)^{-1} {}^W\mathbf{T}_{O'} {}^{O'}\mathbf{T}_O {}^O\mathbf{T}_E, \quad (4.3)$$

where ${}^W\mathbf{T}_R$ refers to the pose calibration of the robot in its workspace; ${}^W\mathbf{T}_{O'}$ refers to the exact pose of the hole part, of which the result is from the proposed perception approach; ${}^{O'}\mathbf{T}_O$ refers to the assembly relationship between the assembly parts in the assembly database; ${}^O\mathbf{T}_E$ is the grasping pose pre-defined in the grasping database. The frame illustration of (4.3) is given in Figure 4.5.

Figure 4.5: Frame illustration of assembly pose.

4.2.3 Task Representation

Tasks

The goal of this task is to pick up around peg and a part with holes on a table and insert the round peg into the round hole within an assigned assembly region. The initial state of this task is pre-determined by some assumptions. First of all, the robotic system is a workstation in a production line, receiving given assembly parts under a certain degree of uncertainty such as various positions (2 DoF on the table) and orientations (1 DoF along the normal vector of the table). This is to simulate the disturbance in transportation. Also, the assembly parts are assumed to be presented by a transportation system without delay, allowing the parts to be presented with a given spatial distribution. The given spatial distribution is to simulate transportation from various directions, meaning that each assembly part is assigned to be presented within a fixed region. The final state utilizes the mechanism of the adopted transportation system, which can transport all assembly parts out of the robotic system without delay to terminate the episode of the task once the goal is achieved.

Figure 4.6 shows the initial state and final state of this task. Based

on manual assembly tasks, workers generally possess knowledge on how to accomplish a given assembly task. Therefore, the information of the assembly parts and corresponding processes is provided to the robot as prior knowledge.

Figure 4.6: State of the task. (a) Initial state. (b) Final state.

Robotic Configurations

The robotic system used in this work consists of a dual-arm industrial robot (KAWADA NEXTAGE Open), a pair of 3D printing two-jaw grippers based on DYNAMIXEL MX-28-T motors, a pair of two-million-pixel industrial cameras mounted on the grippers for local detection, a Microsoft Kinect 2 sensor mounted on the head of the robot for global detection, and a working plane to support the manipulation.

Figure 4.7 shows the robotic system used in this work. The robotic configurations are defined as follows.

1. Dual-arm configuration $\mathfrak{R}_D$:
 - (a) Robot: Two arms of KAWADA NEXTAGE Open (6 DoF + 6 DoF).
 - (b) Sensor: A Microsoft Kinect 2 sensor and two two-million-pixel industrial cameras mounted on the robot.

(c) End-effector: A pair of two-jaw grippers mounted on the robot.

2. Single-arm configuration $\mathfrak{R}_S$:

(a) Robot: The left arm of KAWADA NEXTAGE Open (6 DoF).

(b) Sensor: A Microsoft Kinect 2 sensor and a two-million-pixel industrial camera mounted on the robot.

(c) End-effector: A two-jaw gripper mounted on the robot.

Figure 4.7: Setup of the robotic system.

Task Plans and Actions

Task plans for $\mathfrak{R}_D$ and $\mathfrak{R}_S$ are shown in Figure 4.8. In Figure 4.8, every block refers to a robotic action. The blocks with dash outlines are detection actions. GD refers to “Global detection”; LD refers to “Local detection”; Pic(X) refers to “Pick X”; Pla(X) refers to “Place X”; Ins(X) refers to “Insert X”; Res(X) refers to “Release X and move to initial pose”. P and H inside the brackets refer to “Peg” and “Hole”, respectively. L and R before the underline refers to “using left arm” and “using right arm”, respectively.

Figure 4.8: Task plans for $\mathfrak{R}_D$ and $\mathfrak{R}_S$. The action with dashed border refers to the detection action. In this work, the detection action is not necessary the action with a detection function. Practitioners can select the detection actions based on their needs. In this work, this thesis selects GD, Pic, and Ins, as detection actions due to the possibility of failure of these actions.

For $\mathfrak{R}_D$, there are three detection actions, namely $|\mathcal{SP}| = 4$ and $|\mathcal{S}| = 6$, while for $\mathfrak{R}_S$, there are four detection actions, namely $|\mathcal{SP}| = 5$ and $|\mathcal{S}| = 7$. Note that these results are governed based on Theorem 3.2.4.

Strategies

For an integrated robotic system, sources of uncertainties include but are not limited to variations in the initial pose of assembly parts, modeling errors, sensor noises, and truncation errors. To implement a given task, practitioners often need to think about “what strategies are needed to cope with these uncertainties?” The adopted strategies need to have a higher level of robust-

ness. However, from the manufacturer’s perspective, the chosen strategies need to be cost-effective. This indicates that, sometimes, a strategy set producing the highest success rate may not be the best choice. Strategies considered in the tasks are as follows:

- 1 Strategy $\mathfrak{S}_A$: Assembly detection employed before peg insertion can increase the success rate but also increases the execution time of adding action. However, for different assembly clearances (task difficulties), one cannot estimate whether the effectiveness of this strategy is significant. For example, if the clearance is large enough for more error tolerance, the robot can insert the peg into the hole without detection.
- 2 Strategy $\mathfrak{S}_B$: Assembly point is assigned to be located at the assembly region, but the best location within this region is unknown. In this paper, two different assembly positions were chosen to conduct an experimental study.
- 3 Strategy $\mathfrak{S}_C$: In spite of the lack of force/torque sensor for impedance control, the discretization of fine motion enables the robot to generate vibration during peg insertion and improves the success rate of putting the peg into the hole. However, this strategy causes longer execution time in adding action.

4.3 Experiments

This experimental study validates the effectiveness of the proposed approach on the performance assessment of robotic configurations and plans at task-level from three aspects. The first aspect is the validation of the proposed approach among different tasks. This work considers a simple [PIH](#) task with the different levels of difficulty in this study, where $\mathfrak{T}_1$ is a task with the clearance of 0.25 (easy) and $\mathfrak{T}_2$ is a task with the clearance of

0.1 (hard). Note that the clearance is calculated by: (radius of hole – radius of peg)/radius of hole. The second aspect is to test the generality of the proposed approach in the performance quantification on different robotic configurations. A dual-arm robotic system is used, where two robotic configurations, dual-arm $\mathfrak{R}_D$ and single-arm $\mathfrak{R}_S$, are taken into this study [85]. To simulate the robotic configuration of a single-arm robot, the right arm of the robot is disabled. The third aspect takes a different combination of strategies to verify the usability of the proposed approach.

4.3.1 Experimental Setup

The setup of the experiment study are provided in Table 4.1.

Table 4.1: Descriptions of experiment labels. Here, $\neg\mathfrak{S}_\bullet$ indicates "without the adoption of strategy $\mathfrak{S}_\bullet$ ".

		$\mathfrak{T}_1$					$\mathfrak{T}_2$			
$\mathfrak{R}_D$		$\mathfrak{S}_A$		$\neg\mathfrak{S}_A$			$\mathfrak{S}_A$		$\neg\mathfrak{S}_A$	
		$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$	$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$		$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$	$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$
	$\mathfrak{S}_C$	#1	#3	#5	#7	$\mathfrak{S}_C$	#9	#11	#13	#15
	$\neg\mathfrak{S}_C$	#2	#4	#6	#8	$\neg\mathfrak{S}_C$	#10	#12	#14	#16
$\mathfrak{R}_S$		$\mathfrak{S}_A$		$\neg\mathfrak{S}_A$			$\mathfrak{S}_A$		$\neg\mathfrak{S}_A$	
		$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$	$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$		$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$	$\mathfrak{S}_B$	$\neg\mathfrak{S}_B$
	$\mathfrak{S}_C$	#17	#19	-	-	$\mathfrak{S}_C$	#21	#23	-	-
	$\neg\mathfrak{S}_C$	#18	#20	-	-	$\neg\mathfrak{S}_C$	#22	#24	-	-

4.3.2 Experimental Results

In the experimental study of this work, the execution time of the troubleshooting action was uniformly estimated at 5 seconds. There were 50 replicates for each run, the average of the execution time of each action was taken. Performance quantification of experimental results are provided in Table 4.2 and Table 4.3, where the descriptions of experiment labels are

given in Appendix. Figure 4.9 shows part of the task execution scenes. Note that the APR s are governed by (3.50).

Table 4.2: Experimental results of $\mathfrak{R}_D$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$.

$\mathfrak{T}_1$	#1	#2	#3	#4
$\sum t_i$ (s)	37.39	29.81	36.94	29.49
$\prod r_i$ (%)	100	100	98	98
APR (1/s)	0.0314	0.0413	0.0313	0.0423
$\mathfrak{T}_1$	#5	#6	#7	#8
$\sum t_i$ (s)	35.87	27.77	35.58	27.44
$\prod r_i$ (%)	86	86	94	92
APR (1/s)	0.0287	0.0389	0.0315	0.0422
$\mathfrak{T}_2$	#9	#10	#11	#12
$\sum t_i$ (s)	36.90	29.62	36.90	28.96
$\prod r_i$ (%)	96	88	94	88
APR (1/s)	0.0305	0.0364	0.0302	0.0377
$\mathfrak{T}_2$	#13	#14	#15	#16
$\sum t_i$ (s)	35.87	27.34	35.34	27.40
$\prod r_i$ (%)	86	64	74	74
APR (1/s)	0.0287	0.0294	0.0250	0.0339

Table 4.3: Experimental results of $\mathfrak{R}_S$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$.

$\mathfrak{T}_1$	#17	#18	#19	#20
$\sum t_k$	46.77	37.31	44.69	35.06
$\prod r_k$	100	98	100	96
APR (1/s)	0.0240	0.0305	0.0253	0.0320
$\mathfrak{T}_2$	#21	#22	#23	#24
$\sum t_k$	47.00	37.12	44.90	35.02
$\prod r_k$	96	94	96	96
APR (1/s)	0.0229	0.0294	0.0241	0.0321

Figure 4.9: Part of the task execution scenes.

4.4 Analysis

This section analyzes the experimental results from the previous section. First of all, data validation is achieved through a statistical method, t -test, to prove that the experimental data are reasonable for performance quantification and subsequent analysis. Secondly, quantitative performance, the [APR](#), of each group of the experiment is obtained by the proposed approach, where the differences and explanations among these results are also provided. Finally, a method to the estimation of cost-effectiveness for an additional strategy in a task plan is given, supporting manufacturers with a quick solution to determine the update of a new strategy without repetitive time-consuming experiments.

4.4.1 Data Validation

According to Table 4.4 and Figure 4.10, it can be seen that the strategy affects execution time significantly as the situation between two sets of $\mathcal{SP}_3$ (under $\mathfrak{R}_D$ and $\mathfrak{R}_S$) is different with two sets of $\mathcal{SP}_4$. Note that $\mathcal{SP}_3$ and $\mathcal{SP}_4$ are the subplans described in Figure 4.8. Based on the proposed task representation, experimental results from two sets of identical sub-plans can be considered that they are from one population. Here, two examples are given for the validation, $\mathcal{SP}_3$ and $\mathcal{SP}_4$. Firstly, p -value from calculation of t -test on comparison of two samples of $\mathcal{SP}_3$ is less than 10^{-5} , which is smaller than 0.01, meaning that we have 99% confidence that two samples of $\mathcal{SP}_4$ are not from one population, which also implies that the execution time is influenced by adopted strategy. Another comparison on two samples of $\mathcal{SP}_4$ shows the p -value calculated from t -test is 0.411407, which is greater than 0.01, meaning that we have 99% confidence that two samples of $\mathcal{SP}_4$ are from one population, which implies that the same sub-plans from different groups of experiments are identical.

Table 4.4: Results of $\mathcal{SP}_3$ and $\mathcal{SP}_4$ from experiment #12 and #16.

$\mathcal{SP}_3$	#12	#16	$\mathcal{SP}_4$	#12	#16
Mean (s)	8.211	6.791	Mean (s)	1.801	1.798
Std. Dev. (s)	0.241	0.202	Std. Dev. (s)	0.065	0.074

Figure 4.10: Data visualization of $\mathcal{SP}_3$ and $\mathcal{SP}_4$ between experiment #12 and #16.

4.4.2 Influence of Strategies

Based on the APR calculated, a comparative study to investigate the effect of each strategy can be conducted through ANOVA [86]. As an example, the performance of $\mathfrak{R}_D$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$ with a different combination of adopted strategies are shown in Table 4.5. Note that F_0 is a value for each factor (strategy) in the evaluation. If such a value is greater than $F_{0.05,1,1}$, indicating that this factor (taking this strategy) has a significant impact on the overall performance of the task.

According to Table 4.5, one can know that:

- 1 Strategy $\mathfrak{S}_A$: Obviously, $\mathfrak{S}_A$ can improve the success rate of assembly tasks. In the experiment of $\mathfrak{R}_D$ on $\mathfrak{T}_1$, although $\mathfrak{S}_A$ increases the success rate of peg insertion, incremental improvement does not seem largely due to high error tolerance of assembly parts. For example,

the [APR](#) of experiment #4 and #8 are almost the same, indicating that if rework cost is not considered, the option of $\mathfrak{S}_A$ will not have a significant effect on task performance. Nevertheless, the practical strategy set is still #4 which possesses a higher success rate.

2 Strategy $\mathfrak{S}_B$: The effect of $\mathfrak{S}_B$ is not significant in both groups of experiments.

3 Strategy $\mathfrak{S}_C$: According to experimental results, strategy with the highest [APR](#) is not always the strategy with the highest success rate. This result is caused by different execution time based on options of $\mathfrak{S}_C$.

This analysis provides practitioners some useful information. For example, we know that the strategy $\mathfrak{S}_A$ does not affect $\mathfrak{T}_1$ anymore, then we can assert that it would not affect those tasks easier than $\mathfrak{T}_1$. Therefore, we do not need to consider $\mathfrak{S}_A$ in easier tasks, vice versa.

Table 4.5: ANOVA for $\mathfrak{R}_D$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$.

$\mathfrak{R}_D$ on $\mathfrak{T}_1$ with $\alpha = 0.5$		
Source of Variation	Sum of Squares	F_0
$\mathfrak{S}_A$	1.12×10^{-6}	9
$\mathfrak{S}_B$	6.13×10^{-6}	49
$\mathfrak{S}_C$	2.10×10^{-4}	1681
$\mathfrak{S}_A \mathfrak{S}_B$	3.13×10^{-6}	25
$\mathfrak{S}_B \mathfrak{S}_C$	1.25×10^{-7}	1
$\mathfrak{S}_A \mathfrak{S}_C$	1.25×10^{-7}	1
Error	1.25×10^{-7}	
Total	2.21×10^{-4}	$F_{0.05,1,1} = 161.4$

$\mathfrak{R}_D$ on $\mathfrak{T}_2$ with $\alpha = 0.5$		
Source of Variation	Sum of Squares	F_0
$\mathfrak{S}_A$	4.05×10^{-5}	324
$\mathfrak{S}_B$	2.00×10^{-6}	16
$\mathfrak{S}_C$	7.20×10^{-5}	576
$\mathfrak{S}_A \mathfrak{S}_B$	5.00×10^{-7}	4
$\mathfrak{S}_B \mathfrak{S}_C$	1.80×10^{-5}	144
$\mathfrak{S}_A \mathfrak{S}_C$	4.50×10^{-6}	36
Error	4.50×10^{-6}	1
Total	1.42×10^{-4}	$F_{0.05,1,1} = 161.4$

4.4.3 Estimation of a New Strategy

Sometimes practitioners would like to know whether there is another strategy that could also improve the performance of the robotic system. The experimental data could be reused to provide them with quantification metrics for the addition of a new strategy. Experimental data is taken into consideration without using strategy as a reference. If a practitioner is curious about how performance of a new strategy added into the assembly motion (in the sub-plan $\mathcal{SP}_4$) is, one can replace $\sum_{\mathcal{A}_k \in \mathcal{SP}_4} t_k$ with $\Delta t + \sum_{\mathcal{A}_k \in \mathcal{SP}_4} t_k$ (Δt is the increased execution time of new strategy in an adding action or sub-plan, often obtained through new experiments) to calculate the relationship between APR and Δt , as shown in Figure 4.11. The answer can be obtained that only at $\Delta t \leq 7.576$ (s), 100% success rate achieved by the

new strategy is acceptable (the new APR is no less than original one). If one has an estimation of the increased success rate about the new strategy, the cost-effectiveness (the success rate of $\mathcal{SP}_4$ without strategy is 74%) can be verified by the location of the point in Figure 4.11 (must be within the cost-effective area).

Figure 4.11: Relationship between APR and Δt .

4.4.4 Dominance of robotic configuration

According to the experimental results, the strategy set with the highest APR is considered the best selection of its corresponding configuration. The APR ratio of $\mathfrak{R}_D$ to $\mathfrak{R}_S$ is 1.322 on $\mathfrak{T}_1$ and 1.174 on $\mathfrak{T}_2$, which means the dual-arm configuration is more dominant among the alternatives on easier PIH task (just in this experimental study). This analysis gives practitioners a simpler guide to deploying their robots if he/she possesses the knowledge about the cost ratio of the two configurations. For example, if a practitioner estimates that the cost of dual-arm is double of the single-arm, then the single-arm is more cost-effective because the APR ratio of $\mathfrak{R}_D$ to $\mathfrak{R}_S$ is less than 2.

4.4.5 Discussion

Figure 4.12: The comparison of APR between $\mathfrak{T}_1$ and $\mathfrak{T}_2$.

As shown in Figure 4.12, for different tasks executed by the same robotic configurations and corresponding plan, the APR of $\mathfrak{R}_D$ on $\mathfrak{T}_1$ is obviously higher than on $\mathfrak{T}_2$ (14.22%), while the APR of $\mathfrak{R}_S$ on $\mathfrak{T}_1$ is slightly higher than $\mathfrak{T}_2$ (3.04%). This result shows that the easier task ($\mathfrak{T}_1$) brings higher APR than harder task ($\mathfrak{T}_2$) under the same robotic configurations and corresponding task plans. Besides, the reason that there is merely the difference of APR between $\mathfrak{T}_1$ and $\mathfrak{T}_2$ under $\mathfrak{R}_S$ is that the motion of the hole part is not constrained by another arm, making the hole part movable during the peg insertion action. Therefore, the hole part is endowed with a certain degree of compliance, leading to improvement of the success rate. This improvement leads to the decreased difference of success rate and APR between $\mathfrak{T}_1$ and $\mathfrak{T}_2$.

As shown in Figure 4.13, for different robotic configurations and the corresponding task plans, due to the capability of simultaneous picking of multiple objects by dual-arm robots, an expectation that dual-arm configuration is more efficient in this task than single-arm configuration can be

Figure 4.13: The comparison of APR between $\mathfrak{R}_D$ and $\mathfrak{R}_S$.

validated, improving about 28.62% of APR on $\mathfrak{T}_1$ and 16.03% of APR on $\mathfrak{T}_2$.

The results of different strategy sets of the same robotic configuration are provided in Figure 4.14, Figure 4.15, and Figure 4.16, where the pluses and minuses below the bars refer to the adoption status of another two strategies. The following conclusions can be drawn:

1. The task plans with $\mathfrak{S}_A$ lead to a higher APR value can be drawn, where this strategy averagely improves $\mathfrak{R}_D$ about 3.53% and 15.21% of APR on $\mathfrak{T}_1$ and $\mathfrak{T}_2$, respectively. This shows that $\mathfrak{S}_A$ benefits productivity though it takes more execution time.
2. For strategy $\mathfrak{S}_B$, the average variations of APR performed by both configurations are less than 5%, implying that taking $\mathfrak{S}_B$ or not in task plan is equivalent in performance.
3. Task plans with $\mathfrak{S}_C$ greatly reduce the performance of $\mathfrak{R}_D$ by 25.4% and 16.7% on $\mathfrak{T}_1$ and $\mathfrak{T}_2$, respectively. Experimental results show that $\mathfrak{S}_C$ provides the robotic system higher success rate on peg insertion action, but it considerably extends the execution time for the peg

insertion action, leading to a lower [APR](#). Therefore, $\mathfrak{S}_C$ is not a good strategy for manufacturers.

Figure 4.14: The comparison of [APR](#) on adoption of $\mathfrak{S}_A$.

(a) Under $\mathfrak{R}_D$

(b) Under $\mathfrak{R}_S$

Figure 4.15: The comparison of APR on adoption of $\mathfrak{S}_B$.

(a) Under $\mathcal{R}_D$

(b) Under $\mathcal{R}_S$

Figure 4.16: The comparison of APR on adoption of $\mathcal{S}_C$.

With the requirement of production and related cash flow, further justification at the economy-level (see Chapter 3.4.2) can be made based on these quantified results by manufacturers, if necessary.

Note that to make the experiments more reasonable, this work uses the most intuitive way to design the task plans. Nevertheless, this does not mean that the developed theory and methodologies cannot be applied to other cases. For example, a welding task with multiple trajectories, or a polishing

task with multiple tool changing. Once the studied cases are determined, the developed theory and methodologies can be used by following the manner proposed in this chapter.

4.5 Summary

The chapter demonstrates how to use the proposed approach to quantify the performance of a robotic configuration and the corresponding task plans and strategies on a task. Given a task, all components for the proposed representation should be given first. Next, based on the developed task representation, a task plan can be decomposed for the modeling of [CTMC](#). With the conduction of experiments, the resulting information vectors play an important role in the calculation of [APR](#). Finally, the obtained [APR](#) facilitates the performance quantification, providing essential information for the cross-analysis and further assessment at the economy-level.

Chapter 5

Human-Robot Collaboration: Aerospace Masking

Traditional offline programming or teach pendant based methods limit the collaboration capabilities of users and robots to cope with complex and changing industrial tasks. To address efficient robot manipulation in high-mix low-volume tasks, especially for skillful tasks involving both trajectory and force control requirements, fast robot teaching and skill transferability is critical. Compared to manually dragging a heavy robot, or programming a trajectory by complex calculations, this thesis believes that robots can efficiently learn skills from direct human teaching through telemanipulation, and improve the skills based on optimization with sensory feedback. In aerospace engine [Maintenance, Repair and Operations \(MRO\)](#), the surfaces of aerospace components are required to be masked by tapes. This thesis proposes a fast and intuitive telemanipulation-based method to teach a robot these masking skills and compare the performance of the proposed method with teach pendant based methods among several users. This chapter develops a novel [HRC](#) paradigm based on indirect [HRC](#) form and aims to prove the efficiency and intuitiveness of the aforementioned collaborative manipulation method, the [FCAM](#), for enabling a robot to learn dexterous

manipulation tasks.

5.1 Introduction

In aerospace masking, manual taping of mechanical workpieces is a skillful task where the operator controls the taping direction and uses tools to attach the masking tape with minimal wrinkles. This involves dexterous manipulation to precisely maintain contact with the taping surface under the geometric constraints of the workpieces. Inspired by human skills, this thesis proposes a human-robot communication channel for guiding a robot to learn through a telemanipulation-based [HRC](#) system.

Several methods can be used to teach robots human skills. The most popular method employed in traditional industrial settings is the online programming where the operator *teaches* the robot simple movements by recording robot configurations via a teach-pendant [87]. However, this method is not the best choice for skilled crafts that generally are cumbersome for robot programming. With the increasing complexity of task scenarios, an [OLP](#) methodology is often used. This methodology supports coordination with multiple sensors and peripherals. Besides, more complex algorithms and richer resources can be offline applied to robotic systems. To improve the intuitiveness, [OLP](#) is often combined with graphical simulators, such as V-REP and Gazebo, to support programmers to plan robot motions efficiently. Furthermore, [OLP](#) enables programmers to teach robots without interrupting production processes, which avoids unnecessary downtime of robotic systems [88]. Nevertheless, without a realistic experience, [OLP](#) may require longer programming time than traditional online programming and may not be able to perfectly model some physical properties of the workspace, leading to unfeasible robotic programs.

Recently, as a branch of [HRI](#), the concept of [HRC](#) has become a common

method to transfer skills from human operators to robotic systems [29] [89]. **TbD** or **PbD** is proposed to be a more intuitive methodology than traditional online programming. The efficiency of this methodology has been confirmed to be better than traditional teaching on the tasks that do not require contact interactions with the environment, welding being a classical example [90]. Moreover, **TbD** has improved robot skill learning through different forms of **HRI** such as vision [41], voice [91], and tactile [42]. A more detailed literature review on the methods and algorithms for **TbD** can be found in [43]. Based on these approaches, **TbD** is a promising method to bring robots closer to help humans under multiple scenarios, meeting the public imagination of a future **HRI** reality.

The information extraction from demonstrations to best support robot learning is still an open research challenge [43]. Several types of demonstrations have been used, such as vision-based demonstrations [92] and contact-based demonstrations [93]. Besides, other mechanisms, such as neural network [94] and fuzzy theory [95] are also useful in such scenarios. For more other applications, readers can refer to [96] [97]. From the industrial point of view, to make **TbD** a feasible solution to improve the usability of skill transfer, holistic system design and corresponding method for **HRI** is required to achieve high-quality and efficient robot learning.

In a previous work [98], an automatic taping system based on **OLP** programming was introduced. The automatic taping system uses a scanner for workpiece **3D** model reconstruction. The system can handle the taping tasks for flat, cylindrical, freeform, and grooved surfaces. However, some limitations of the method are encountered.

1. To reconstruct the **3D** model of the workpiece is time-consuming and the quality of the reconstructed model in terms of accuracy and detail depends on the resolution and precision of scanners [99].
2. The actual geometric constraints of the tools need to avoid tool-workpiece

collision and the contact between the workpiece and the stretched tape on the taping tool in manipulation [100].

3. To accurately calibrate the robot pose is of importance, which is also time-consuming in the system setup process.

Figure 5.1: The robotic system and framework used in this chapter. (a) Robotic system used in this chapter. (b) Overview of the setup of this system, the red area is the robot workspace, and the brown area is the teaching area for the human operator. (c) Interaction between human operator, robot, and environment, during the telemanipulation.

Due to such difficulties, this thesis expects that human can perform as a powerful agent for guiding the robot to find its correct taping path efficiently. Figure 5.1 (a) shows the robotic system for aerospace masking used in this work. This system is composed of a rotating platform, a collaborative robot, a taping tool, and a force-torque sensor. For capturing human motions, an

HTC VIVE system for teleoperation is deployed. This system consists of two base stations that can capture the 6-DoF motion of the handheld controllers. The handheld controllers provide haptic feedback, via vibrations, to the operator. The operator teaches the robot from a region that is unreachable to the robot to ensure user safety, as shown in Figure 5.1 (b).

For feasible human teaching in tasks with force interaction, this thesis proposes an HRC mode, as illustrated in Figure 5.1 (c). During the task teaching, the operator uses intuitive hand motions to telemanipulate the robot. The human motion is translated to robot motion by the teleoperation system so that the robot can understand this motion and generate corresponding actions to interact with the environment. Besides, the operator can visually observe the robot's actions and the environment. The robotic system senses and regulates the contact forces between the workpiece and the taping tool. The teleoperation system translates these interaction forces into haptic feedback in the form of vibrations. This approach allows the operator to have an idea of the amount of force being applied to the workpiece and avoids the damages of the robot and the environment.

In this design, the operator can control the robotic system and teach the task with only one demonstration. Then the system improves the learned task using skill recognition algorithms. This approach is convenient and yields high-quality results making it appealing for real-world applications.

5.2 Methodology

5.2.1 Collaborative Pattern

For the limitation of the telemanipulation accuracy and sensing capability of the teleoperation system, purely telemanipulation without robotics assistance could be challenging and sometimes even dangerous. In this task, the robot must apply certain force to the workpiece for task execution. Robot

sensory data provides additional assistance to compensate for the error of human control commands to collaborate with the operator. Therefore, the online assistance of the robot manipulator based on its sensory feedback is expected to benefit the telemanipulation, providing better precision in position control, smoother in force interaction and safer in collision avoidance.

In such a way, the robot's embedded sensors detect the conditions of the applied taping force. An efficient channel that transmits such force information to the operator and gets his/her attention to respond efficiently is helpful. In this work, the vibration actuator in the handheld controller is utilized to transmit the force information to the operator. The vibration magnitude increases linearly with the contact force magnitude to alert the operator in fine adjustment of the taping tool position while taping. Since this channel is almost immediate, the operator can monitor the conditions of interactive contact force on-line based on the haptic feedback, making the telemanipulation a close-loop robot teaching with visual feedback (in precision) and haptic feedback (in contact force). Such functions fit nicely with the taping process teaching requirements.

5.2.2 Manipulation

(a)

(b)

(c)

Figure 5.2: Definitions of frames in the developed system. (a) The reference frame F_W is defined to describe the frame detected by the base stations. The human frame F_B is defined to describe human hand motion. The robot frame F_R is defined to describe the robot motion. (b) The frame of the handheld controller is assumed to be equivalent to the frame of human hand F_H . (c) The frame of tip of the robot taping tool F_E .

Based on the definitions in Chapter 3.3, the frames in the developed system are depicted in Figure 5.2. We can use Theorem 3.3.1 for fast calibration. Next on, Equation 3.31 and Equation 3.35 are employed to achieve human

motion tracking in the model-less case.

This thesis selects the X-axis of the robot base frame F_R as a reference. Aligning the X-axis of F_H with the X-axis of F_R , the sensor generates a sensory [HTM](#) ${}^W\mathbf{T}_{calib}$ of which the rotation matrix is equal to ${}^W\mathbf{R}_R$. Therefore, we can describe the human motion as

$${}^R\mathbf{T}_H = {}^R\mathbf{T}_W {}^W\mathbf{T}_S = {}^W\mathbf{T}_{calib}^{-1} {}^W\mathbf{T}_H. \quad (5.1)$$

However, only the rotation of ${}^W\mathbf{T}_{calib}$ is required, and we want to eliminate the errors along other axes of F_W , i.e. along X- and Y-axis. Hence, we can compute the yaw angle ψ of ${}^W\mathbf{T}_{calib}$, and then replace the ${}^W\mathbf{T}_{calib}$ with a corrected [HTM](#) ${}^W\mathbf{T}'_{calib}$:

$${}^W\mathbf{T}'_{calib} = \exp\left(\begin{bmatrix} ([0 & 0 & \psi]^T)^\wedge & \mathbf{0}_{3 \times 1} \\ & 0 & & 0 \end{bmatrix}\right), \quad (5.2)$$

where the operator $(\cdot)^\wedge$ turns a $\mathbb{R}^3$ vector to a 3×3 skew-symmetric matrix.

For comfortable telemanipulation, this thesis proposes an adjustable motion retargeting method. A motion mapping function $f_M : SE(3) \times SE(3) \times SE(3) \rightarrow SE(3)$ is needed for establishing the relationship between the human pose and the robot pose:

$${}^R\mathbf{T}_{E,t} = f_M({}^B\mathbf{T}_{H,t}, {}^R\mathbf{T}_{C,0}, {}^R\mathbf{T}_{E,0}). \quad (5.3)$$

To ensure the closure of $SO(3)$, the difference of rotations is represented by

$$\delta^R\mathbf{R}_{H,t} = {}^R\mathbf{R}_{H,t}^{-1} {}^R\mathbf{R}_{H,0}. \quad (5.4)$$

Then, we use the following rotation scheme of f_M based on logarithmic map ${}^\varepsilon\phi_\vartheta^\wedge = \ln({}^\varepsilon\mathbf{R}_\vartheta)$:

$$\delta^R\phi_{E,0:t} = k_r \delta^R\phi_{H,0:t}, \quad k_r \in \mathbb{R}^+, \quad (5.5)$$

where ${}^\varepsilon\phi_{\vartheta}^\wedge \in \mathfrak{so}(\mathbf{3})$. Then, we have

$${}^R\mathbf{R}_{E,t} = \exp\left({}^R\phi_{E,t}^\wedge\right) = \exp\left((k_r\delta^R\phi_{H,0:t} + {}^R\phi_{E,0})^\wedge\right). \quad (5.6)$$

To ensure the linearity of translation, the translation scheme of f_M can be obtained by

$${}^R\mathbf{p}_{E,t} - {}^R\mathbf{p}_{E,0} = k_p({}^R\mathbf{p}_{H,t} - {}^B\mathbf{p}_{H,0}), \quad k_p \in \mathbb{R}^+. \quad (5.7)$$

Finally, the pose can be calculated as follows

$${}^R\mathbf{T}_{E,t} = \begin{bmatrix} {}^R\mathbf{R}_{E,t} & {}^R\mathbf{p}_{E,t} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}, \quad (5.8)$$

where k_p and k_r can be adjusted for telemanipulating the robot in different motion scales.

5.2.3 Skill Recognition

The method herein this work is to identify the different demonstrated skills and organize the skills in serial based on the sequence of the actual task and optimize the identified specific skills for better task performance. In this work, the wrench data of the force-torque sensor and the state of the taping tool for the robot is taken to recognize the skills in the demonstrated trajectory. A combination of logical reasoning and machine learning methods is used for skill estimation. Later on, an algorithm for globally analyzing the estimated skill is proposed to correct the wrong estimations due to uncertainties. Finally, the demonstrated trajectory is separated into multiple sub-trajectory, where each sub-trajectory refers to a certain skill recognized by the robot.

Skill Representation

Different symbols are used to represent the different skills in this task. The full set of demonstrated skills are defined as $\mathcal{S} = \{\mathcal{S}_0, \mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3\}$, where $\mathcal{S}_0$ refers to simple moving, $\mathcal{S}_1$ refers to taping skill, $\mathcal{S}_2$ refers to moving with force involved, and $\mathcal{S}_3$ refers to cutting skill.

A demonstrated trajectory $\mathcal{D}$ is considered to be composed of a sequence of robot poses $[{}^R\mathbf{T}_{E,1}, \dots, {}^R\mathbf{T}_{E,|\mathcal{D}|}]$. Each pose is accompanied with the taping force ${}^t f_{e,i}$, wrench data from the force-torque sensor ${}^S\mathbf{w}_{E,i} = [{}^S f_{Ex,i}, {}^S f_{Ey,i}, {}^S f_{Ez,i}, {}^S \tau_{Ex,i}, {}^S \tau_{Ey,i}, {}^S \tau_{Ez,i}]^T$, and the actuation state of the cutter on the tapping tool $C_i \in \{0, 1\}$, where $C_i = 0$ refers to the deactivation of the cutter and $C_i = 1$ refers to the activation of the cutter.

Skill Estimation

Based on logical reasoning, the cutting skill $\mathcal{S}_3$ is a skill starting with the activation of the cutter and terminating with the deactivation of the cutter. Therefore, the cutting skill can be recognized via $C_i = 1$. In some situations, however, the taping force f_i cannot differentiate $\mathcal{S}_0$ and $\mathcal{S}_2$ on simple moving, and $\mathcal{S}_1$ and $\mathcal{S}_2$ on taping skill. Therefore, this work takes the wrench of the force-torque sensor ${}^S\mathbf{w}_{E,i}$ as features and use a machine learning method for classification. A linear [Support Vector Classifier \(SVC\)](#) is employed for training a model with small amount data [101] [102]. This work collects part of a demonstrated trajectory as a training set. For each robot pose in the trajectory, the skills $\{\mathcal{S}_0, \mathcal{S}_1, \mathcal{S}_2\}$ are labeled manually. Table 5.1 depicts the parameters used in the training.

Correction of Skill Estimation

The estimated skills may not be identified correctly in the trajectory due to many uncontrollable factors, such as the error of the trained model, the noise of the force-torque sensor, human mistakes in teaching. Therefore, we

Table 5.1: Parameters Used in the Linear SVC.

Parameters	Contents
Penalty	L2
Loss function	Squared hinge
Algorithm	Dual optimization problem
Tolerance	0.0001
Penalty of the error term	1
Max. iteration	1000

need to analyze these skills in a whole trajectory and correct them based on the actual sequence of the skills in the task as well as other relevant features, based on a pre-defined rule: $\mathcal{S}_0 \rightarrow \mathcal{S}_1 \rightarrow \mathcal{S}_2 \rightarrow \mathcal{S}_3 \rightarrow \mathcal{S}_0$.

With the correction, the robot can divide the demonstrated trajectory $\mathcal{D}$ into multiple sub-trajectories $[\mathcal{D}_1, \dots, \mathcal{D}_m]$ in sequence. Each sub-trajectory is composed of poses with continuous identical skill and is endowed with certain actions related to its skill for execution. The robot will improve these sub-trajectories based on a pre-defined optimization schedule.

5.2.4 Optimization of Taping Skill

Through direct human-robot telemanipulation, the robot can be directly guided to generate a roughly correct path, but it is hardly possible to teach the robot a very precise path directly, especially with both precision and force requirements. Therefore, further correction of the taught path based on the measurements of the force-torque sensor to achieve a functional path for proper task manipulations is very critical.

After recognizing all skills appearing in a demonstration, the robot can improve the learned skills through optimization in practices. This work focuses on the optimization of the trajectories $\mathcal{D}_j \subseteq \mathcal{D}$ corresponding to the taping skill $\mathcal{S}_1$.

Given a taping trajectory $\mathcal{D}_j$ with skill $\mathcal{S}_1$, each robot pose ${}^R\mathbf{T}_{E,l} \in \mathcal{D}_j$ can be decomposed into a translation vector ${}^R\mathbf{p}_{E,l}$ and a rotation matrix

$${}^R\mathbf{R}_{E,l}.$$

Based on the skill recognition approach, the taping results, the corresponding sensory information, and the skill recognition results are provided in Figure 5.3. These results show that the robot can recognize the demonstrated skills in different taping paths via teleoperation in one session.

Figure 5.3: Results for five taping paths, labeled with different color tags, in a single teaching session. (a) Shows the resultant taping paths with different quality levels. (b) Shows the force measurements, ${}^Sf_{Ex}$ (red), ${}^Sf_{Ey}$ (green), ${}^Sf_{Ez}$ (blue), and f (brown). (c) Depicts the torque measurements: ${}^S\tau_{Ex}$ (red), ${}^S\tau_{Ey}$ (green), and ${}^S\tau_{Ez}$ (blue) (d) Red line indicates the skill recognition results of the trained Linear SVC model, and blue line presents the skill recognition results combining the provided prior knowledge for the five taping paths.

5.2.5 Optimization

Force Regulation

During teaching, sudden overshooting of the taping force occurring especially when the taping tool travels between two layers of the rings, during which

direct telemanipulation could cause the taping tool to collide with the next layer. In contrast, an insufficient taping force will cause the tape segment to fail to attach to the workpiece surface. A force regulation approach is to regulate the taping forces by adjusting the approaching distance of the taping tool along the taping direction, as shown in Figure 5.4. Thus, a proportional control law is used to update the position of the taping tool based on the following formula:

$${}^R\mathbf{p}_{E,l} \leftarrow {}^R\mathbf{p}_{E,l} - {}^R\mathbf{R}_{E,l}[k_e(f_l - \bar{f}_l), 0, 0]^T, \quad (5.9)$$

where k_e refers to a proportional parameter determined by the stiffness of the compliance spring; $\bar{f}_l$ refers to the desired taping force. The taping force can be well smoothed after only a few practices.

Figure 5.4: Illustration of the displacement compensation based on the force feedback from the taping tool. The taping force is regulated adjusting the separation distance of the taping tool along the taping direction. The spring mechanism of the taping tool enables contact compliance between the taping tool and the workpiece surface.

Angular Error Correction

The geometric model and physical contact condition during the taping are complex as the taping tool has three angular errors: yaw, pitch, and roll. Meanwhile, the contact force on the workpiece surface includes not only the supporting force $f_{\perp}$ and the supporting torque $\tau_{\perp}$ but also the friction force f_{μ} , as shown in Figure 5.5.

Figure 5.5: Illustration of force analysis in angular error correction. The desired direction for the taping tool to attach tape is along the surface normal $\mathbf{u}$, where the grey plane represents the local tangential plane at the taping point. Given a taping orientation, this work studies the taping tool posture based on two angles. (A) θ is an angle rotating along the roller axis $\mathbf{v}$. (B) ϕ is known as the roller flipping angle rotating along the axis $\mathbf{w}$.

Fortunately, given a correct taping path, the friction forces f_{μ} and the equivalent torque in the middle of the roller ($\tau_{\perp}$) is negligible, and thus the angular correction model can be simplified. Given the taping point and the corresponding taping direction, the expected taping orientation is along the surface normal of the workpiece $\mathbf{u}$. Two angular errors cause the orientation of the taping tool different from the expectation. One is the pitch angle θ , which rotates the taping tool around its roller axis $\mathbf{v}$. As denoted in Figure 5.5 (a), such errors within a small range do not affect the taping quality, the only one drawback is that the tool can not directly apply taping force along the surface normal $\mathbf{u}$. On the other hand, this error can

be easily regulated by the operators' hand motion in the telemanipulation. Therefore, this work does not optimize this angle along the path, but it is necessary that such angle needs to be controlled within a small range during the robot teaching. In this work, for correcting the orientation of the taping tool, this work mainly deals with the roller flipping angle ϕ . As shown in Figure 5.5(b), the rotation axis of ϕ is perpendicular to both the surface normal $\mathbf{u}$ and the roller axis $\mathbf{v}$. When the taping roller is flipped, part of the tape segment will not be pressed onto the surface, and it will also cause wrinkles and incorrect attaching direction on the tape segment.

Based on experimental testing, the friction force f_μ is very sensitive to the roller flipping angle ϕ . To correct such angular errors, the force measurement f of the force-torque sensor ${}^S\mathbf{w}_E$ is applied. For each key point to be optimized, we have

$$f_{\perp,l} = f_l \csc(\phi_l). \quad (5.10)$$

To simplify the model for flipping angle correction, this work approximates $f_\perp$ as a constant value. Base on testing, by applying the following linearized equations to evaluate the roller flipping angle ϕ , which works nicely with good convergence:

$$\phi_j = k_f f_j, \quad (5.11)$$

where k_f is a parameter determined through testing.

With the assumption that the pitch angle θ is trivial, the orientation of the taping tool can be corrected by

$${}^R\mathbf{T}_{E,l} \leftarrow {}^R\mathbf{T}_{E,l} \exp \left(\left(\begin{bmatrix} \mathbf{0}_{4 \times 1} \\ -\phi_l \\ 0 \end{bmatrix} \right)^\wedge \right). \quad (5.12)$$

5.3 Performance Evaluation

5.3.1 User Study

To prove the efficiency and the intuitiveness of the proposed teaching approach, this work conducts a user study to compare the telemanipulation-based teaching approach - *Telemanipulation Method* - against a teach pendant-based teaching method - *Manual Method*. During these experiments, the Telemanipulation Method allows the operator to use hand motions to control the robot via a handheld controller, while the Manual Method requires the operator to manually drag around the robot and record waypoints for task teaching.

Task Descriptions

A given taping task is that the robot needs to move from an initial location, attach tape along a given trajectory on the workpiece surface, cut the tape segment, and finally move back to the initial location, where the whole trajectory must be a collision-free path for the robot and each surface and edge along the given taping trajectory on the workpiece surface must be covered by the tape segment.

Experimental Setup

For the manual method, the participants only need to physically move the robot around and record the robot poses point-by-point via the teach pendant. To ensure that the participants can teach the robot how to cut a tape segment off, the provided interface allows them to control the cutter on the taping tool.

For the telemanipulation method, participants have to calibrate their pose before teaching, and then provide taping demonstration via telemanipulation. By using the handheld controller, the participants move the robot

and control the cutter with their hand motion and the trigger during telemanipulation. To ensure physical safety, every participant is required to stand outside the reachable workspace of the robot during teaching, where the experiment executor ensures user security through an emergency stop button. Additionally, the speed of the robot is limited to be less than 0.2 (m/s).

With informed consent, 10 participants (all male, with ages between 23 and 40 years old) are provided with instructions to operate the system using both teaching methods. All participants are experienced in the use of IT devices and have a robotic background. Except for the system developer, none of the participants has experience in the telemanipulation methods, while three of them have experience in the manual methods. For each method, participants are given 10 minutes to familiarize with the system and ask questions to clearly understand the requirements for the taping task. Afterward, the user is required to finish the task three times without assistance. The task is regarded as completed once the robot can autonomously reproduce it successfully.

Criteria

According to [ISO/International Electrotechnical Commission \(IEC\) 25022](#) (the measurement of quality in use), the user study evaluates the usability from three aspects defined in [ISO 9241-11](#) are studied: effectiveness, efficiency, and satisfaction. The descriptions of the used criteria are provided as follows:

- Effectiveness: To evaluate the completeness with which users achieve specified goals, the average amount of practice in the familiarization process is used.
- Efficiency: To evaluate efficiency, teaching time is used to assess participants' teaching performance. In the manual method, the teaching

time starts from the loading of the teaching program and ends at finishing the recording of the robot waypoints; in the telemanipulation method, the teaching time starts from the grasping of the handheld controller and ends at pressing the menu button for finishing demonstration. After participants finish teaching, experiment executor is responsible for examining the results executed by the robot. If the quality of taping is not achieved, the participant is required to teach the robot again and the teaching time will be accumulated.

- Satisfaction: To evaluate the freedom from discomfort and positive attitudes towards the use of the system, a questionnaire is used to quantify participants' feelings. With a seven-point rating scale [103] where 1 refers to strongly disagree and 7 refers to strongly agree, two subjective metrics are measured: ease of use [104] and enjoyment [105]. The statement for ease of use is "This method is easy to teach the robot how to complete the taping task," and for enjoyment is "I would choose this method to teach the robot in the future."

Table 5.2: Descriptive and Inferential Statistics for User Performance.

Descriptive Statistics				
	Manual Method		Telemanipulation Method	
Measurement	Mean	SD	Mean	SD
Practice (times)	1.1	0.3	1.1	0.3
Teaching Time (s)	294.9	101.0	77.9	33.2
Ease of Use	3.2	1.48	6.4	0.52
Enjoyment	2.8	1.69	6.4	1.07
Inferential Statistics				
Measurement	t -value		p -value	
Practices (times)	0		> 0.01	
Teaching Time (s)	-7.19		< 0.01	
Ease of Use	6.25		< 0.01	
Enjoyment	6.65		< 0.01	

Analysis

This work uses a paired t -test to validate that the differences between the manual method and the telemanipulation method are significant. Two-Sided testing is used and the significant level α is selected as 0.01, indicating that we need confidence of 99% to reject the type I error (a statement that the measurement between these two methods is indifferent).

Figure 5.6: Comparison between the manual and the telemanipulation methods. (a) Boxplots for optimal teaching time, ease of use, and enjoyment between the two teaching methods. (b) Learning curve for the two methods across three different trials.

The descriptive statistics and the inferential statistics for the measurement are provided in Table 5.2, the trial with the shortest (optimal) teaching time of participants are selected as their teaching time. The comparisons of these measurements and the learning curve are presented in Figure 5.6.

According to the experimental results, the average amount of practice in the familiarization processes is the same, implying that participants can grasp the skills of the two teaching methods under the provided instructions of the two methods. Despite the differences in teaching time, the participants can finish the tasks three times without assistance. These results indicate that the telemanipulation method is effective after participants' practices, at least the same as the traditional manual method. Besides, the system is proved to be more efficient and intuitive than commonly used teach pendant-based teaching method. For the performance comparison between the manual method and the telemanipulation method on the taping task, the user study shows that our method has advantages on efficiency (nearly four times faster), ease of use (6.4 v.s. 3.2) and enjoyment (6.4 v.s. 2.8). With the p -value of each measurement less than 0.01, indicating that we have the confidence of 99% to confirm that the difference in each measurement between these two methods is significant. The proposed method enables operators with on-site sensing, telemanipulation, and coordination with the robot. Even though the telemanipulation method involves a higher cognitive burden on the operator as compared against the manual method, our experimental results indicate that operators find the proposed method easier to use and prefer it over the manual method.

5.3.2 System Validation

With an available demonstration, the developed skill recognition approach enables the robot to pick up skills from the demonstration and the proposed optimization approach supports the robot to improve its learned skills con-

tinuously.

An experiment for testing the performance of robot learning is conducted. The goal of these experiments is to prove that:

1. Skills in the demonstrations with different teaching qualities can be recognized by the robot.
2. The robot can finish the taping task successfully in the first practice after only one demonstration provided by human telemanipulation.
3. The robot can improve its skills with the increase of practice times.

Skill Recognition

Given an imperfect demonstration where some parts of the tape segment are detached from the workpiece surface, the result of skill recognition in the teaching episode is shown in Figure 5.7. As we can see, the skill estimated by f cannot differentiate $\mathcal{S}_1$ and $\mathcal{S}_2$, leading to the possibility of failure to finish the cutting skill $\mathcal{S}_3$ due to the wrong optimization of $\mathcal{S}_2$ in the future. On the other hand, the correction of estimation successfully conveys the operator's intention to the robot, enabling the robot to recognize the trajectory it needs to optimize. The weight $\mathbf{W} \in \mathbb{R}^{6 \times 3}$ and the bias $\mathbf{b} \in \mathbb{R}^3$ are:

$$\mathbf{W} = \begin{bmatrix} 0.408 & -1.081 & 0.633 \\ -0.283 & 0.385 & 0.184 \\ -0.15 & 0.022 & -0.074 \\ 0.206 & -0.246 & 0.173 \\ 0.751 & -0.157 & -0.167 \\ -0.367 & 0.175 & 0.341 \end{bmatrix}$$

$$\mathbf{b} = [0.998, -1.291, -0.918]^T$$

Figure 5.7: Results of skill estimation, where the blue line tells the robotic system which waypoints are needed to be optimized.

Optimization

With a desired taping force $\bar{f} = 7.1$ (N) and two proportional parameters, $k_e = 0.0544$ and $k_f = 0.5$ at the first episode; $k_e = 0.0136$ and $k_f = 0$ for the following episodes. The imperfect demonstration and the results of robot learning are shown in Figure 5.8, where some parts of the tape are detached from the workpiece surface in human teaching. After optimization, the tape segment can be attached to the workpiece surface firmly.

The optimization results of each iteration are given in Figure 5.9. As we can see, after five episodes of optimization, the taping force of each waypoint gradually approaches to the desired taping force. To assess the quality and the uniformity of the taping force during taping, this work calculates an [Root-Mean-Square \(RMS\)](#) error:

$$e_{RMS} = \sqrt{\frac{\sum_{i=1}^{N_c} (f_i - \bar{f})^2}{N_c}} \quad (5.13)$$

where N_c is the number of waypoints where there is a contact between the

Table 5.3: The RMS Error of Taping Force of Each Episode.

Episode	Teaching	#1	#2	#3	#4	#5
e_{RMS}	5.449	1.850	2.818	2.076	1.487	1.044

robot and the workpiece. Table 5.3 depicts the results of this metric, where this error is obviously reduced about 80.9% in five episodes, proving the effectiveness of the force optimization.

Figure 5.8: Result of an imperfect human demonstration and robot learning. The human demonstrations are provided via the telemanipulation method. The different episodes are the sequential results of robot learning. Gaps between tape segments are set to differentiate the taping quality for each episode.

Figure 5.9: Optimization of taping force in each episode, where the setpoint is the desired taping force for the taping skill $\mathcal{S}_1$.

Demonstration

In the demonstration, this work shows the feasibility of a [HRC](#) paradigm on the teaching of a taping task. Figure 5.10 shows two tape segments, the first one is applied during the teaching session, where the operator uses the handheld device to guide the robot, and the second segment corresponds to the replayed task by the robot. In this demonstration, a skillful operator can complete the teaching session in 1 minute, after that, the robotic system learns the task, performs the skill recognition and replays the shortcut and smoothed task.

Figure 5.10: Demonstration of the developed system in the taping task. The first row is human demonstration, and the second row is the reproduction from robot. Each column denotes the stage of task and its categorized skill.

Taping Quality

Figure 5.11 presents the taping quality under each method, selected from the previous paper [98] and demonstrations in this work, showing that the proposed [HRC](#) paradigm can achieve similar taping quality to other methods.

(a) Manual method [98]

(b) OLP method [98]

(c) Teach-pendant method

(d) The proposed HRC paradigm

Figure 5.11: The taping quality under each method.

5.3.3 Further Improvement

A complex workpiece surface with multiple layers is very common in engine parts maintenance. In such a condition, the tip of the taping tool would be easily stuck in the grooves between layers, which damages the tool due to the lacking of human consciousness in teleoperation. To make the best use of taping tool to not only apply tapes on the layers of the workpiece surface but also reach the grooves between these layers, this work uses a [High-Density Polyethylene \(HDPE\)](#) material for providing compliance while supporting tape attaching on the designed tool, as shown in Figure 5.12 (a). With the soft covering material attached on the surface of the tip of the taping tool, the taping tool can move into the grooves for taping, providing a higher

taping coverage for better taping quality, as illustrated in Figure 5.12 (b).

Figure 5.12: Design of the taping tool. (a) Components in the designed taping tool. (b) Illustration of the adaption capability of the design of taping tool on workpiece with complex surface.

Similarly, with the employment of the aforementioned [FCAM](#), a skillful operator can guide the taping tool to follow the workpiece surfaces and attach the tape onto the surface as well as the edges between layers. When the tip of the taping tool is approaching the narrow grooves on the workpiece, the operator can change to fine-motion tuning mode to enable the taping tool to move into the groove while applying the tape attachment.

To validate the feasibility of the used methods in the developed system and evaluate its performances, the same workpiece is used for experiments. Due to such complex surface constraints, the robot needs to be manipulated

in a fine-motion tuning mode. It takes about 5 minutes for a skillful operator to finish the taping task, as demonstrated in Figure 5.13.

Figure 5.13: Implementation of teleoperation methods in the complex surface masking task. (a) Fast registration for determining the robot's heading orientation. (b) Recording the initial pose of the operator's hand and the robot taping tool tip for motion mapping. (c) The operator telemanipulates the robot to a desired pose in the large-scale guidance mode. (e) The operator changes to fine-motion tuning mode for taping. (f-h) The operator repetitively uses the clutching mechanism for finishing the taping task in the fine tuning mode. (i) The operator finishes the task. (j-l) The robot is doing the taping.

Comparing the taping quality of the teaching performance against man-

ual method, as shown in Figure 5.14, it is obvious that with the use of FCAM, the taping quality and its corresponding reproduction results are generally equivalent to the manual taping using the new taping tool, and they are much better than the previous taping tool which cannot handle edge and groove taping. Besides, the number of the recorded waypoints for the presented quality through the teleoperation methods is 3327, which is considerably time-consuming for practitioners to record such a great amount of waypoints through teach-pendent-based methods.

Figure 5.14: The taping results of human manual taping with the previous taping tool, human manual taping with the new taping tool, human teleoperation, and the robot reproduction.

Finally, the clutching mechanism of FCAM also allows the operator to telemanipulate the robot more flexibly. In the actual taping process, for example, the horizontal taping may be required. Another demonstration is conducted where the operator applies the clutching mechanism to guide the robot from a pose for vertical taping to a pose for horizontal taping. The results are shown in Figure 5.15. The clutching mechanism supports the operator to change the motion mapping frequently, and manipulate the robot to work more flexibly in its workspace.

Figure 5.15: The scenario of telemanipulating the robot to another desired pose for a different taping path.

5.3.4 Discussion

According to these experimental results, our skill recognition approach shows its capability to recognize the skills in demonstrations with force interactions, providing useful information for the robot to improve the learned skills. Our experimental results of optimization prove that the developed system with the proposed optimization approaches supports the robot to successfully apply its learned skills from a human demonstration in its first attempt, after only one demonstration. The following optimization supports the robot to improve its taping skills based on the taping force, enabling the robot to correct manual errors during the teaching phase, such as an uneven taping force or detached tape segment. This validates an effective solution enabling robots to learn force-involved dexterous human skills and adaptively apply them to aerospace masking.

5.4 Summary

In this chapter, this thesis studies a taping task with application in aerospace masking. Herein, this thesis presents the use of an [HRC](#) paradigm to provide human demonstrations for a robotic taping system, which provides a fast and efficient approach to deploy robots into aerospace masking. Comparing to the results using [OLP](#) method in our previous work [98], our results show a more efficient way to set up the whole system, without the modeling of workpiece, the pose calibration of the robot and workpiece, and transferring the simulated offline path to the actual robot and compensate for the errors between the physical and the simulated model of the system. This provides a promising solution for flexible production in factories, especially for high-mix low-volume tasks.

Chapter 6

Conclusion

6.1 Contribution

The aims of this thesis are to present a study on the concept of collaborative robotics and a research of some paradigms for industrial manipulation. This thesis studies three novel aspects in collaborative robotics:

1. This thesis proposes a clear definitions of *cooperation*, *collaboration*, and *coordination* and provide a novel classification for researchers to categorize different collaborative patterns and forms.
2. This thesis develops a mathematical representation that is compatible with different collaborative patterns. It can assist practitioners to model task plans and connect to traditional system analysis methods for performance assessment.
3. This thesis provides a collaborative manipulation framework for human operators to teach robots with flexibility, intuition, and time-efficiency in different scenarios.

The theoretical contents are provided in Chapter [3](#). Herein, this thesis clarifies and redefine the difference and meaning of three common terms:

cooperation, collaboration, and coordination. Next on, this thesis proposes a classification of various collaborative patterns and forms, including non-collaboration, [RRC](#), and [HRC](#). Based on the location of collaborative behaviors, [RRC](#) can be further categorized into external collaboration and internal collaboration. In [HRC](#), considering the effects of the human in a task, two collaborative forms exist: direct collaboration and indirect collaboration. Later, this thesis develops a task representation based on graph theory to model different task plans under the proposed collaborative patterns and forms. With this representation, we can easily integrate system analysis techniques to quantify the performance at task-level so that practitioners can determine the cost-effectiveness in a data-driven method instead of experience-based decision making based on [CTMC](#) and statistics methods. Besides, this thesis provides a collaborative manipulation framework, [FCAM](#), which is based on differential geometry and group theory and can offer human operators to teach robots in a more flexible, intuitive, and time-efficiently manner in indirect collaboration. This framework is compatible with model-based and model-less motion retargeting. Based on the teaching direction, the framework supports direct teaching and mirrored teaching. To make this framework ease-to-use, handheld devices can be combined to realize manipulation state changing. Heading registration can be finished in one button press, which greatly shortens the tedious calibration setup in the robot deployment process. Besides, human operators can switch manipulation mode between large-scale guidance and fine-motion tuning based on the needs of teaching. If a motion is hard to reach, human operators can also use the clutching mechanism to re-orient the robot’s end-effector pose.

The experimental contents and implementation are given in Chapter [4](#) and Chapter [5](#). In Chapter [4](#), this thesis studies a typical robotic [PIH](#) task under non-collaboration and [RRC](#) patterns, where an industrial dual-arm robot is used to perform both single-arm and dual-arm manipulation in this

task. In addition, vision-based perception is integrated into the robotic system for providing useful information about the positions and orientations of objects. This thesis also provides the implementation and demonstrate how the developed task presentation can be employed on modeling the different task plans under different agents. With the representation, this thesis validates that the proposed [CTMC](#) technique can be applied to quantify the productivity, [APR](#), at the task-level by the conducted experiments. Besides, the four suggested aspects (data validation, the influence of strategies, the estimation of a new strategy, and the dominance of robotic configuration) in cross-analysis can be employed for practitioners' initial decisions, and the resultant information vectors are later used for further assessment at the economy-level.

In Chapter 5, this thesis studies a aerospace masking task under [HRC](#) pattern. In this case, a robot is required to perform multiple skills to finish a taping task. Based on the developed [FCAM](#) framework, this thesis uses a [VR](#) device for telemanipulation and take advantage of the human operator's visual sensing and the robot's tactile sensing capabilities to achieve the collaboration. Besides, during the teaching phase, the robot plays as an active learner, who can adapt to the human's unconscious errors in the force profile, to slightly adjust the taping force to prevent the excessive force from damaging the taping tool and workspace. After the human demonstration, the robot recognizes the demonstrated skills via the developed skill learning algorithms which can estimate and correct the demonstrated skills. Finally, the robot can learn the demonstrated skills in the task sequence and apply optimization algorithms for each skill continuously in the phase of task execution. The experiments show that the developed paradigm is more time-efficient, flexible, and intuitive for human operators. Without the tedious setup and programming processes, human operators can teach the robot much easier than traditional teach-pendant based or [OLP](#) meth-

ods. Besides, the further improvement of the end-effector also shows the precision of the [FCAM](#) framework and offers a promising solution on the aerospace masking tasks that have various workpieces with different shapes.

The two paradigms demonstrate that the developed framework of collaborative robotics for manipulation can successfully represent different task plans under different agents and make the performance of robotic tasks at task-level measurable. In addition, through the application of various collaborative patterns and forms in the robotic tasks, the execution of robotic tasks can become more time-efficient and intuitive, providing the practitioners more flexibility and convenience on the deployment of their robotic systems.

6.2 Future Works

Although the theoretical contents in this work are validated in the given two cases, there are still many open research issues:

1. The modeling of task plans has the assumption that the scenario is structured, as usual as situations in manufacturing nowadays. Hence, those industries that have unstructured environments, such as unpredictable and cluttered environments, are not considered.
2. The connection of human factors cannot be integrated into the developed task representation as the part of human behavior research is also an quite open issue. Besides, this thesis pays more attention to robot-centered research.
3. The robot's learning capability under the proposed [FCAM](#) framework depends on the quality of the human demonstration. If a low-quality demonstration is provided, the robot will fail to learn. For example, the operator provides a low-precision trajectory or a force profile that

cannot meet the task requirements.

Despite the existence of these limitations, this thesis believes that the presented methods and paradigms of this work can be further improved to provide manufacturers a feasible solution to deploy and teach their robot efficiently and intuitively in various industrial scenarios. Finally, this thesis expects that this work can inspire more researchers and make robotics be a more cost-effective solution in automation for releasing human workers from tedious and unhealthy workloads.

Bibliography

- [1] J. E. Colgate and M. A. Peshkin, “Cobots,” Sept. 14 1999, uS Patent 5,952,796.
- [2] *Robots and robotic devices — Safety requirements for industrial robots (Part 1: Robots)*. ISO/TS 10218-1, 2006.
- [3] *Robots and robotic devices — Safety requirements for industrial robots (Part 1: Robots)*. ISO/TS 10218-1, 2011.
- [4] *Robots and robotic devices — Collaborative robots*. ISO/TS 15066, 2016.
- [5] *Robots and robotic devices — Safety requirements for industrial robots (Part 2: Robot systems and integration)*. ISO/TS 10218-2, 2011.
- [6] C. Bernier, “Collaborative robot series: Universal robots,” URL: <https://blog.robotiq.com/bid/64942/Collaborative-Robot-Series-Universal-Robots> (visited on 2019-08-27), 2013.
- [7] D. Kirschner, R. Velik, S. Yahyanejad, M. Brandstötter, and M. Hofbaur, “Yumi, come and play with me! a collaborative robot for piecing together a tangram puzzle,” in *International Conference on Interactive Collaborative Robotics*. Springer, 2016, pp. 243–251.
- [8] W. Garage, “Pr2,” *Personal Robot*, vol. 2, 2013.

- [9] N. Yagiz, Y. Hacıoglu, and Y. Z. Arslan, “Load transportation by dual arm robot using sliding mode control,” *Journal of Mechanical science and Technology*, vol. 24, no. 5, pp. 1177–1184, 2010.
- [10] A. Bauer, D. Wollherr, and M. Buss, “Human–robot collaboration: a survey,” *International Journal of Humanoid Robotics*, vol. 5, no. 01, pp. 47–66, 2008.
- [11] S. Arimoto, F. Miyazaki, and S. Kawamura, “Cooperative motion control of multiple robot arms or fingers,” in *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, vol. 4. IEEE, 1987, pp. 1407–1412.
- [12] J. Yamaguchi, E. Soga, S. Inoue, and A. Takanishi, “Development of a bipedal humanoid robot-control method of whole body cooperative dynamic biped walking,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 1. IEEE, 1999, pp. 368–374.
- [13] S. Hayati, “Hybrid position/force control of multi-arm cooperating robots,” in *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, vol. 3. IEEE, 1986, pp. 82–89.
- [14] Y. Koga and J.-C. Latombe, “On multi-arm manipulation planning,” in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*. IEEE, 1994, pp. 945–952.
- [15] F. Suárez-Ruiz, X. Zhou, and Q.-C. Pham, “Can robots assemble an ikea chair?” *Science Robotics*, vol. 3, no. 17, p. eaat6385, 2018.
- [16] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, “Coordination and decentralized cooperation of multiple mobile manipulators,” *Journal of Robotic Systems*, vol. 13, no. 11, pp. 755–764, 1996.

- [17] S. Ahmad and H. Guo, “Dynamic coordination of dual-arm robotic systems with joint flexibility,” in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*. IEEE, 1988, pp. 332–337.
- [18] B. P. Gerkey and M. J. Mataric, “A formal analysis and taxonomy of task allocation in multi-robot systems,” *The International Journal of Robotics Research*, vol. 23, no. 9, pp. 939–954, 2004.
- [19] G. A. Korsah, A. Stentz, and M. B. Dias, “A comprehensive taxonomy for multi-robot task allocation,” *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512, 2013.
- [20] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative multi-robot exploration,” in *ICRA*, 2000, pp. 476–481.
- [21] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, “Coordinated multi-robot exploration,” *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.
- [22] L. E. Parker, “Current state of the art in distributed autonomous mobile robotics,” in *Distributed Autonomous Robotic Systems 4*. Springer, 2000, pp. 3–12.
- [23] D. Meike, M. Pellicciari, G. Berselli, A. Vergnano, and L. Ribickis, “Increasing the energy efficiency of multi-robot production lines in the automotive industry,” in *2012 IEEE International Conference on Automation Science and Engineering (CASE)*. IEEE, 2012, pp. 700–705.
- [24] G. Sanchez and J.-C. Latombe, “Using a prm planner to compare centralized and decoupled planning for multi-robot systems,” in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 2. IEEE, 2002, pp. 2112–2119.

- [25] W. Nguyen and J. K. Mills, “Multi-robot control for flexible fixtureless assembly of flexible sheet metal auto body parts,” in *proceedings of IEEE International Conference on Robotics and Automation*, vol. 3. IEEE, 1996, pp. 2340–2345.
- [26] B. J. Grosz, “Collaborative systems (aaai-94 presidential address),” *AI magazine*, vol. 17, no. 2, pp. 67–67, 1996.
- [27] H. A. Yanco and J. Drury, “Classifying human-robot interaction: an updated taxonomy,” in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, vol. 3. IEEE, 2004, pp. 2841–2846.
- [28] J. R. Katzenbach and D. K. Smith, *The wisdom of teams: Creating the high-performance organization*. Harvard Business Review Press, 2015.
- [29] B. Chandrasekaran and J. M. Conrad, “Human-robot collaboration: A survey,” in *SoutheastCon 2015*, 2015, pp. 1–8.
- [30] K. M. Nalini and R. R. Gondkar, “Robotic recognition for unstructured 2-D parts to pick and place objects,” in *Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on*. IEEE, 2017, pp. 1478–1482.
- [31] F. Suárez-Ruiz, X. Zhou, and Q.-C. Pham, “Can robots assemble an ikea chair?” *Science Robotics*, vol. 3, no. 17, 2018.
- [32] M. Liarokapis and A. M. Dollar, “Deriving dexterous, in-hand manipulation primitives for adaptive robot hands,” in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 1951–1958.

- [33] H. Fukui, S. Yonejima, M. Yamano, M. Dohi, M. Yamada, and T. Nishiki, “Development of teaching pendant optimized for robot application,” in *Advanced Robotics and its Social Impacts (ARSO), 2009 IEEE Workshop on*. IEEE, 2009, pp. 72–77.
- [34] Q. Yuan, T. Y. Seng, Q. Lu, and I.-M. Chen, “Task-orientated robot teleoperation using wearable imus,” in *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*. IEEE, 2017, pp. 111–115.
- [35] G. Vasiljevic, N. Jagodin, and Z. Kovacic, “Kinect-based robot teleoperation by velocities control in the joint/cartesian frames,” *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 805–810, 2012.
- [36] D. Kruse, R. J. Radke, and J. T. Wen, “A sensor-based dual-arm tele-robotic manipulation platform,” in *2013 IEEE International Conference on Automation Science and Engineering (CASE)*. IEEE, 2013, pp. 350–355.
- [37] D. Rakita, B. Mutlu, and M. Gleicher, “An autonomous dynamic camera method for effective remote teleoperation,” in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2018, pp. 325–333.
- [38] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A survey of robot learning from demonstration,” *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.
- [39] T. Zhang, Z. McCarthy, O. Jowl, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, “Deep imitation learning for complex manipulation tasks from virtual reality teleoperation,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.

- [40] K. teaching. [Online]. Available: <https://blog.robotiq.com/bid/71737/Teaching-Welding-Robots-by-Demonstration-Kinetiq-Teaching>
- [41] M. Kawato, F. Gandolfo, H. Gomi, and Y. Wada, “Teaching by Showing in Kendama Based on Optimization Principle,” in *ICANN '94*. M. Marinaro, P. G. Morasso: Springer, 1994, pp. 601–606.
- [42] K. Kuklinski, K. Fischer, I. Marhenke, F. Kirstein, M. V. Aus Der Wieschen, D. Sølvason, N. Kruger, and T. R. Savarimuthu, “Teleoperation for learning by demonstration: Data glove versus object manipulation for intuitive robot control,” in *International Congress on Ultra Modern Telecommunications and Control Systems and Workshops*, vol. 2015-January, 2015, pp. 346–351.
- [43] A. Billard, S. Calinon, and Dillmann, “Learning from Humans,” in *Springer Handbook of Robotics*. Springer, 2016, p. 1995.
- [44] H. H. Poole, *Fundamentals of robotics engineering*. Springer Science & Business Media, 2012.
- [45] A. J. Critchlow, “Introduction to robotics,” 1985.
- [46] K. Sambasiva Rao and S. Deshmukh, “Strategic framework for implementing flexible manufacturing systems in india,” *International Journal of Operations & Production Management*, vol. 14, no. 4, pp. 50–63, 1994.
- [47] D. E. Whitney, *Mechanical assemblies: their design, manufacture, and role in product development*. Oxford Series on advanced manufacturing, 2004, vol. 1.
- [48] J. W. Troxler and L. T. Blank, *Justifying flexible automation: A system value model*. Society of Manufacturing Engineers, 1987.

- [49] G. Azzone and U. Bertele, “Measuring the economic effectiveness of flexible automation: a new approach,” *The International Journal of Production Research*, vol. 27, no. 5, pp. 735–746, 1989.
- [50] A. Dore and E. Lo, “Economic evaluation of robot-based assembly systems,” *The International Journal of Production Research*, vol. 29, no. 2, pp. 267–276, 1991.
- [51] E. Messina and A. Jacoff, “Performance standards for urban search and rescue robots,” in *Unmanned Systems Technology VIII*, vol. 6230. International Society for Optics and Photonics, 2006, p. 62301V.
- [52] B. A. Weiss and C. Schlenoff, “Evolution of the score framework to enhance field-based performance evaluations of emerging technologies,” in *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*. ACM, 2008, pp. 1–8.
- [53] M. O. Shneier, E. R. Messina, C. I. Schlenoff, F. M. Proctor, T. R. Kramer, and J. A. Falco, “Measuring and representing the performance of manufacturing assembly robots,” NIST, Tech. Rep., 2015.
- [54] A. S. Kondoleon, *Cycle time analysis of robot assembly systems*. Society of Manufacturing Engineers, 1979.
- [55] J. Shi and R. Menassa, “Flexible robotic assembly in dynamic environments,” in *Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop*. ACM, 2010, pp. 271–276.
- [56] J. Marvel, J. Falco, and J. Marvel, *Best practices and performance metrics using force control for robotic assembly*. US Department of Commerce, National Institute of Standards and Technology, 2012.
- [57] T. Suzuki, T. Ohashi, and M. Asano, “Assembly reliability evaluation method (arem),” in *ASME 2002 International Mechanical Engineering*

- Congress and Exposition.* American Society of Mechanical Engineers Digital Collection, 2008, pp. 657–662.
- [58] S. Balakirsky, T. Kramer, Z. Kootbally, and A. Pietromartire, “Metrics and test methods for industrial kit building,” *Gaithersburg, MD2013*, 2012.
 - [59] S. Kmenta, B. Cheldelin, and K. Ishii, “Assembly finea: a simplified method for identifying assembly errors,” in *ASME 2003 International Mechanical Engineering Congress and Exposition.* American Society of Mechanical Engineers Digital Collection, 2008, pp. 315–323.
 - [60] I. I. Karayalcin, “The analytic hierarchy process: Planning, priority setting, resource allocation: Thomas l. saaty mcgraw-hill, new york, 1980, xiii+ 287 pages,£ 15.65,” 1982.
 - [61] G. E. Box, J. S. Hunter, and W. G. Hunter, “Statistics for experimenters,” in *Wiley Series in Probability and Statistics.* Wiley Hoboken, NJ, USA, 2005.
 - [62] N. Viswanadham and Y. Narahari, *Performance Modeling of Automated Systems.* PHI Learning Pvt. Ltd., 2015.
 - [63] D. Huang and F. Wu, “Study on quantity of handling robot base on queuing theory,” in *2010 International Conference on Measuring Technology and Mechatronics Automation*, vol. 3. IEEE, 2010, pp. 624–627.
 - [64] T. Yoshikawa, *Foundations of robotics: analysis and control.* MIT press, 1990.
 - [65] J. K. Salisbury and J. J. Craig, “Articulated hands: Force control and kinematic issues,” *The International journal of Robotics research*, vol. 1, no. 1, pp. 4–17, 1982.

- [66] R. Diankov, “Automated construction of robotic manipulation programs,” Ph.D. dissertation, University of Tokyo, 2010.
- [67] F. Zacharias, C. Borst, and G. Hirzinger, “Capturing robot workspace structure: representing robot capabilities,” in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Ieee, 2007, pp. 3229–3236.
- [68] F. Burget and M. Bennewitz, “Stance selection for humanoid grasping tasks by inverse reachability maps,” in *2015 IEEE International conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 5669–5674.
- [69] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “Octomap: An efficient probabilistic 3d mapping framework based on octrees,” *Autonomous robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [70] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, “A depth space approach to human-robot collision avoidance,” in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 338–345.
- [71] V. Kac and I.-d. L. algebra algebra Lie, *Lie Algebras*. Springer, 1990.
- [72] G. G. Slabaugh, “Computing euler angles from a rotation matrix,” *Retrieved on August*, vol. 6, no. 2000, pp. 39–63, 1999.
- [73] K.-T. Park, C.-H. Park, and Y.-J. Shin, “Performance evaluation of industrial dual-arm robot,” in *2008 International Conference on Smart Manufacturing Application*. IEEE, 2008, pp. 437–440.
- [74] J. G. Kemeny and J. L. Snell, *Markov Chains*. Springer-Verlag, New York, 1976.

- [75] R. Durrett, *Probability: theory and examples*. Cambridge university press, 2019, vol. 49.
- [76] W. Feller, “On the theory of stochastic processes, with particular reference to applications,” in *Selected Papers I*. Springer, 2015, pp. 769–798.
- [77] W.-K. Ching and M. K. Ng, “Markov chains,” *Models, algorithms and applications*, 2006.
- [78] Y. Yamada, S. Nagamatsu, and Y. Sato, “Development of multi-arm robots for automobile assembly,” in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3. IEEE, 1995, pp. 2224–2229.
- [79] J. Ge, Z. Chi, and Q. Li, “Small part assembly with dual arm robot and smart camera,” in *ISR/Robotik 2014; 41st International Symposium on Robotics*. VDE, 2014, pp. 1–6.
- [80] S. Fang, X. Huang, H. Chen, and N. Xi, “Dual-arm robot assembly system for 3c product based on vision guidance,” in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2016, pp. 807–812.
- [81] J. Takamatsu, K. Ogawara, H. Kimura, and K. Ikeuchi, “Recognizing assembly tasks through human demonstration,” *The International Journal of Robotics Research*, vol. 26, no. 7, pp. 641–659, 2007.
- [82] W. Wan and K. Harada, “Developing and comparing single-arm and dual-arm regrasp,” *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 243–250, 2016.
- [83] R. B. Rusu and S. Cousins, “Point cloud library (pcl),” in *2011 IEEE international conference on robotics and automation*, 2011, pp. 1–4.

- [84] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
- [85] C.-Y. Weng, W. C. Tan, and I.-M. Chen, "A survey of dual-arm robotic issues on assembly tasks," in *ROMANSY 22-Robot Design, Dynamics and Control*. Springer, 2019, pp. 474–480.
- [86] D. C. Montgomery, *Design and analysis of experiments*. John wiley & sons, 2017.
- [87] Z. Pan, J. Polden, N. Larkin, S. Van Duin, and J. Norrish, "Recent progress on programming methods for industrial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 87–94, 2012.
- [88] G. Wittenberg, "Developments in offline programming: an overview," *Industrial Robot: An International Journal*, vol. 22, no. 3, pp. 21–23, 1995.
- [89] L. Rozo, S. Calinon, D. G. Caldwell, P. Jiménez, and C. Torras, "Learning physical collaborative robot behaviors from human demonstrations," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 513–527, June 2016.
- [90] J. Lambrecht, M. Kleinsorge, M. Rosenstrauch, and J. Krüger, "Spatial programming for industrial robots through task demonstration," *International Journal of Advanced Robotic Systems*, vol. 10, no. 5, p. 254, 2013.
- [91] M. Ehrenmann, O. Rogalla, R. Zollner, and R. Dillmann, "Teaching Service Robots complex Tasks: Programming By Demonstration for Workshop and Household Environments," *2001 International Conference on Field and Service Robots (FSR)*, pp. 397–402, 2001.

- [92] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine, “One-Shot Imitation from Observing Humans via Domain-Adaptive Meta-Learning,” *arXiv preprint arXiv:1802.01557*, 2018.
- [93] T. R. Savarimuthu, A. G. Buch, C. Schlette, N. Wantia, J. Roßmann, D. Martínez, G. Alenyà, C. Torras, A. Ude, B. Nemec, *et al.*, “Teaching a Robot the Semantics of Assembly Tasks,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 5, pp. 670–692, 2018.
- [94] T. Gao, Y. Liu, L. Liu, and D. Li, “Adaptive neural network-based control for a class of nonlinear pure-feedback systems with time-varying full state constraints,” *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 5, pp. 923–933, Sep. 2018.
- [95] L. Liu, Y. Liu, and S. Tong, “Fuzzy based multi-error constraint control for switched nonlinear systems and its applications,” *IEEE Transactions on Fuzzy Systems*, pp. 1–1, 2018.
- [96] D. Li and D. Li, “Adaptive neural tracking control for an uncertain state constrained robotic manipulator with unknown time-varying delays,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 12, pp. 2219–2228, Dec 2018.
- [97] L. Liu, Y. Liu, and S. Tong, “Neural networks-based adaptive finite-time fault-tolerant control for a class of strict-feedback switched nonlinear systems,” *IEEE Transactions on Cybernetics*, pp. 1–10, 2018.
- [98] Q. Yuan, T. S. Lembono, I.-M. Chen, S. N. Landen, and V. Malmgren, “Automatic robot taping with force feedback,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2017, pp. 1821–1826.

- [99] Q. Yuan, I.-M. Chen, and T. S. Lembono, “An agile robot taping system – modeling, tool design, planning and execution,” *Industrial Robot: An International Journal*, vol. 43, no. 5, pp. 503–512, 2016.
- [100] Q. Yuan, I.-M. Chen, T. S. Lembono, S. N. Landen, and V. Malmgren, “Strategy for robot motion and path planning in robot taping,” *Frontiers of Mechanical Engineering*, vol. 11, no. 2, pp. 195–203, 2016.
- [101] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “LIBLINEAR: A Library for Large Linear Classification,” *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.
- [102] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and Others, “Scikit-learn: Machine learning in Python,” *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.
- [103] R. Likert, “A technique for the measurement of attitudes,” *Archives of Psychology*, vol. 22 140, p. 55, 1932.
- [104] F. D. Davis, “Perceived Ease of Use, and User Acceptance of Information Technology,” *MIS Quarterly*, vol. 13, no. 3, pp. 319–340, 1989.
- [105] A. Lin, S. Gregor, and M. Ewing, “Developing a scale to measure the enjoyment of web experiences,” *Journal of Interactive Marketing*, vol. 22, no. 4, pp. 40–57, 2008.

List of Figures

2.1	Several models of collaborative robots.	11
2.2	Taxonomy of HRI proposed by [27], where human is labeled as “H” and robot is labeled as “R”.	13
2.3	Mechanism of collaboration by [10].	13
3.1	Patterns and forms of collaboration.	25
3.2	Illustration of the operation over $\mathcal{SP}$	35
3.3	Topological model of a single-arm robotic system in non- collaboration.	36
3.4	Topological model of a dual-arm robotic system in RRC. . .	41
3.5	The operations over $\mathcal{SP}_1^1, \mathcal{SP}_2^1$ and $\mathcal{SP}_1^2$	41
3.6	Topological model of two single-arm robotic systems in RRC. .	42
3.7	Topological model of direct collaboration in HRC.	43
3.8	Topological model of indirect collaboration in HRC.	44
3.9	The operation flow of FCAM, where $B1$ and $B2$ are the signals automatically/manually sent from the operator.	48
3.10	Illustration of motion mapping.	54
3.11	Illustration of model-less mirror mapping.	59
3.12	Steps of performance assessment at task-level.	61
3.13	An example of mapping from a task plan $(n, l) = (4, 1)$ to a state space of CTMC, where $\mathcal{A}_2 \in \mathcal{A}^d$	65
3.14	Proposed CTMC model for a robotic task plan.	67

4.1	Implementation result of global detection. (a) Received image in F_S . (b) Visualization of Algorithm 1, where "WAIST" is the name of F_R	72
4.2	Implementation result of local detection. (a) Visualization of Algorithm 2, where the brown rectangle is the captured contour, yellow dot is the center of the contour and the red line is the principal axis of PCA. (b) Result of compensation value.	74
4.3	Framework of manipulation.	75
4.4	Frame illustration of grasping pose.	76
4.5	Frame illustration of assembly pose.	77
4.6	State of the task. (a) Initial state. (b) Final state.	78
4.7	Setup of the robotic system.	79
4.8	Task plans for $\mathfrak{R}_D$ and $\mathfrak{R}_S$. The action with dashed border refers to the detection action. In this work, the detection action is not necessary the action with a detection function. Practitioners can select the detection actions based on their needs. In this work, this thesis selects GD, Pic, and Ins, as detection actions due to the possibility of failure of these actions.	80
4.9	Part of the task execution scenes.	84
4.10	Data visualization of $\mathcal{SP}_3$ and $\mathcal{SP}_4$ between experiment #12 and #16.	86
4.11	Relationship between APR and Δt	89
4.12	The comparison of APR between $\mathfrak{T}_1$ and $\mathfrak{T}_2$	90
4.13	The comparison of APR between $\mathfrak{R}_D$ and $\mathfrak{R}_S$	91
4.14	The comparison of APR on adoption of $\mathfrak{S}_A$	92
4.15	The comparison of APR on adoption of $\mathfrak{S}_B$	93
4.16	The comparison of APR on adoption of $\mathfrak{S}_C$	94

5.1	The robotic system and framework used in this chapter. (a) Robotic system used in this chapter. (b) Overview of the setup of this system, the red area is the robot workspace, and the brown area is the teaching area for the human operator. (c) Interaction between human operator, robot, and environment, during the telemanipulation.	99
5.2	Definitions of frames in the developed system. (a) The reference frame F_W is defined to describe the frame detected by the base stations. The human frame F_B is defined to describe human hand motion. The robot frame F_R is defined to describe the robot motion. (b) The frame of the handheld controller is assumed to be equivalent to the frame of human hand F_H . (c) The frame of tip of the robot taping tool F_E . .	102
5.3	Results for five taping paths, labeled with different color tags, in a single teaching session. (a) Shows the resultant taping paths with different quality levels. (b) Shows the force measurements, ${}^S f_{Ex}$ (red), ${}^S f_{Ey}$ (green), ${}^S f_{Ez}$ (blue), and f (brown). (c) Depicts the torque measurements: ${}^S \tau_{Ex}$ (red), ${}^S \tau_{Ey}$ (green), and ${}^S \tau_{Ez}$ (blue) (d) Red line indicates the skill recognition results of the trained Linear SVC model, and blue line presents the skill recognition results combining the provided prior knowledge for the five taping paths.	107
5.4	Illustration of the displacement compensation based on the force feedback from the taping tool. The taping force is regulated adjusting the separation distance of the taping tool along the taping direction. The spring mechanism of the taping tool enables contact compliance between the taping tool and the workpiece surface.	108

5.5	Illustration of force analysis in angular error correction. The desired direction for the taping tool to attach tape is along the surface normal $\mathbf{u}$, where the grey plane represents the local tangential plane at the taping point. Given a taping orientation, this work studies the taping tool posture based on two angles. (A) θ is an angle rotating along the roller axis $\mathbf{v}$. (B) ϕ is known as the roller flipping angle rotating along the axis $\mathbf{w}$	109
5.6	Comparison between the manual and the telemanipulation methods. (a) Boxplots for optimal teaching time, ease of use, and enjoyment between the two teaching methods. (b) Learning curve for the two methods across three different trials.	114
5.7	Results of skill estimation, where the blue line tells the robotic system which waypoints are needed to be optimized.	117
5.8	Result of an imperfect human demonstration and robot learning. The human demonstrations are provided via the telemanipulation method. The different episodes are the sequential results of robot learning. Gaps between tape segments are set to differentiate the taping quality for each episode.	118
5.9	Optimization of taping force in each episode, where the set-point is the desired taping force for the taping skill $\mathcal{S}_1$	118
5.10	Demonstration of the developed system in the taping task. The first row is human demonstration, and the second row is the reproduction from robot. Each column denotes the stage of task and its categorized skill.	119
5.11	The taping quality under each method.	120
5.12	Design of the taping tool. (a) Components in the designed taping tool. (b) Illustration of the adaption capability of the design of taping tool on workpiece with complex surface.	121

5.13	Implementation of teleoperation methods in the complex surface masking task. (a) Fast registration for determining the robot's heading orientation. (b) Recording the initial pose of the operator's hand and the robot taping tool tip for motion mapping. (c) The operator telemanipulates the robot to a desired pose in the large-scale guidance mode. (e) The operator changes to fine-motion tuning mode for taping. (f-h) The operator repetitively uses the clutching mechanism for finishing the taping task in the fine tuning mode. (i) The operator finishes the task. (j-l) The robot is doing the taping.	122
5.14	The taping results of human manual taping with the previous taping tool, human manual taping with the new taping tool, human teleoperation, and the robot reproduction.	123
5.15	The scenario of telemanipulating the robot to another desired pose for a different taping path.	124

List of Tables

2.1	List of one-time cost and on-going cost.	17
2.2	List of direct cost and indirect cost.	17
2.3	Structure of AHP analysed in [48].	19
3.1	Performance criteria at robot-level and economy-level.	61
4.1	Descriptions of experiment labels. Here, $\neg\mathfrak{S}_\bullet$ indicates "without the adoption of strategy $\mathfrak{S}_\bullet$ ".	82
4.2	Experimental results of $\mathfrak{R}_D$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$	83
4.3	Experimental results of $\mathfrak{R}_S$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$	83
4.4	Results of $\mathcal{SP}_3$ and $\mathcal{SP}_4$ from experiment #12 and #16.	85
4.5	ANOVA for $\mathfrak{R}_D$ on $\mathfrak{T}_1$ and $\mathfrak{T}_2$	88
5.1	Parameters Used in the Linear SVC.	106
5.2	Descriptive and Inferential Statistics for User Performance.	113
5.3	The RMS Error of Taping Force of Each Episode.	118

Symbols

Abstract Concepts

$\emptyset$	An empty set
F	A frame
$\mathfrak{F}$	A feasible agent set
$\mathfrak{H}$	A human agent set
$\mathcal{L}_o$	A list of objects
$\mathfrak{R}$	A robotic agent set
$\mathfrak{S}$	A strategy set
$\mathfrak{T}$	A task

Mathematics-Related Concepts

APR	An average product rate
$\mathcal{A}$	An action set
$\mathcal{A}^d$	A detection action set
$^S\mathcal{A}$	A setup action set
$^M\mathcal{A}$	A manipulation action set
$^R\mathcal{A}$	A troubleshooting action set
C_i	An estimated rework cost per second for $\mathcal{SV}_i$
C^∞	A infinite topological space that has infinite continuous derivatives
$\mathcal{E}$	An edge
$\hat{F}_t$	An estimated cash flow in period t
$\mathcal{G}$	A graph

I	An information set
$\mathbf{I}$	An identity element
$\mathcal{I}$	An index set
k_p	motion scaling factor for translation mapping
k_r	motion scaling factor for rotation mapping
k_s	motion scaling factor
$\mathcal{M}$	A manifold
N_t	The total number of a considered time periods
$\mathcal{O}$	A coordination product set
PV	A present value
$\mathbf{P}$	A transition probability matrix
$\mathbf{p}$	A point
${}^\varepsilon \mathbf{p}_\vartheta$	The translation from F_ε to F_ϑ
$\mathcal{P}$	A plan set
$\mathbf{Q}$	A transition rate matrix
r_i	The success rate of the information I_i carried by action $\mathcal{A}_i$
$\bar{r}$	A fractional interest rate due in each studied period
${}^\varepsilon \mathbf{R}_\vartheta$	The rotation matrix from F_ε to F_ϑ
$\mathcal{S}$	A state space
$SE(3)$	A Special Euclidean Group
$\mathcal{SE}_i$	An edge of a sub-graph $\mathcal{SG}$
$\mathfrak{se}(3)$	The lie algebra of $SE(3)$
$\mathcal{SG}$	A sub-graph
$SO(3)$	A Special Orthogonal Group
$\mathfrak{so}(3)$	The lie algebra of $SO(3)$
$\mathcal{SP}^i$	A sub-plan in plan $\mathcal{P}_i$
$\mathcal{SP}_{eq}$	An equivalent sub-plan
$\mathcal{SV}_i$	A vertex of a sub-graph $\mathcal{SG}$
t_i	The execution time of the information I_i carried by action $\mathcal{A}_i$

${}^\varepsilon \mathbf{T}_\vartheta$	The homogeneous transformation matrix from F_ε to F_ϑ
$\mathbf{T}_{rf}$	The reflection matrix
$\mathcal{T}_p \mathcal{M}$	A tangent space that describes the velocity of $\mathbf{p}$
$\mathcal{U}_i$	An open set that covers a manifold
$\mathcal{V}$	A vertex
X	A stochastic process
${}^\varepsilon \Phi_\vartheta$	The $\mathfrak{so}(3)$ from F_ε to F_ϑ
${}^\varepsilon \Xi_\vartheta$	The $\mathfrak{se}(3)$ from F_ε to F_ϑ
$(\Omega, \mathcal{H}, \text{Pr})$	A probability space

Operators

$C(i, j)$	The number of j -combinations from a given set of i elements
$\exp_\bullet : \mathcal{T}_\bullet \mathcal{M} \mapsto \mathcal{M}$	A exponential map at point $\bullet$
f_M	The motion retargeting function
$g : \mathcal{P} \mapsto \mathcal{G}$	A function that constructs a plan to a graph
$\ln_\bullet : \mathcal{M} \rightarrow \mathcal{T}_\bullet \mathcal{M}$	The logarithmic map at point $\bullet$
$\mathfrak{p} : \mathcal{G} \mapsto \mathcal{P}$	A function that maps a graph to a plan set
$\delta^+(\bullet)$	The outdegree of a vertex $\bullet$
$\delta_{\bullet a:b}$	The difference of $\bullet$ between timestamp a to timestamp b
$\phi_i : \mathcal{U}_i \mapsto \mathbb{R}^3$	A homeomorphism mapping in an atlas
$\wp(\bullet)$	The power set of $\bullet$
$(\bullet)^\wedge$	The skew symmetric of $\bullet$
$(\bullet)^\vee$	The inverse skew symmetric of $\bullet$
$ \bullet $	The cordinality of $\bullet$

Acronyms

3D	Three-Dimensional.
AEM	Assembleability Evaluation Method.
AHP	Analytic Hierarchy Process.
ANOVA	Analysis of Variance.
APR	Average Product Rate.
AREM	Assembly Reliability Evaluation Method.
CTMC	Continuous Time Markov Chain.
DFA	Design for Assembly.
DOE	Design of Experiments.
DoF	Degree of Freedom.
FCAM	Fast Calibration and Adjustable Mapping.
FMEA	Failure Modes and Effects Analysis.
FMS	Flexible Manufacturing System.
HDPE	High-Density Polyethylene.
HRC	Human-Robot Collaboration.
HRI	Human-Robot Interaction.
HTM	Homogeneous Transformation Matrix.

IEC	International Electrotechnical Commission.
IMU	Inertial Measurement Unit.
ISO	International Organization for Standards.
MRO	Maintenance, Repair and Operations.
OLP	Offline Programming.
PbD	Programming by Demonstration.
PCA	Principal Component Analysis.
PCL	Point Cloud Library.
PGM	Probabilistic Graphical Model.
pHRI	Physical Human-Robot Interaction.
PIH	Peg-in-Hole.
RGB-D	RGB-Depth.
RMS	Root-Mean-Square.
RRC	Robot-Robot Collaboration.
SVC	Support Vector Classifier.
TbD	Teaching by Demonstration.
TS	Technical Specifications.
UR	Universal Robots.
VR	Virtual Reality.

List of Publications

Journals

- H Dong, CY Weng, C Guo, H Yu and IM Chen, “Real-time Avoidance Strategy of Dynamic Obstacles via Half Model-free Detection and Tracking with 2D Lidar for Mobile Robots,” in *IEEE/ASME Transactions on Mechatronics*, 2020. **[In Press]**
- C Xia, **CY Weng***, Y Zhang and IM Chen, “Vision-based Measurement and Prediction of Object Trajectory for Robotic Manipulation in Dynamic and Uncertain Scenarios,” in *IEEE Transactions on Instrumentation and Measurement*, 69(11), 2020, pp. 8939-8952.
- Q Yuan, **CY Weng***, F Suárez-Ruiz and IM Chen, “Flexible Telemanipulation Based Handy Robot Teaching on Tape Masking with Complex Geometry,” in *Robotics and Computer-Integrated Manufacturing*, 66, 2020, pp. 101990.
- **CY Weng**, Q Yuan, ZJ Lim and IM Chen, “Applications of Light-Weight Wearable Devices to Online Programming of Industrial Dual-Arm Robots,” in *Unmanned Systems*, 8(3), 2020, pp. 211-219. **[Selected Paper in IEEE CIS-RAM 2019]**
- **CY Weng**, Q Yuan, F Suárez-Ruiz and IM Chen, “A Telemanipulation-Based Human-Robot Collaboration Method to Teach Aerospace Mask-

ing Skills,” in *IEEE Transactions on Industrial Informatics*, 16(5), 2020, pp. 3076-3084.

- **CY Weng**, WC Tan, Q Yuan and IM Chen, “Qualitative Assessment at Task-Level for Performance of Robotic Configurations and Task Plans,” in *Journal of Intelligent and Robotic Systems*, 96, 2019, pp. 439-456.

Conference Proceedings

- **CY Weng**, Q Yuan, ZJ Lim and IM Chen, “Applications of Light-Weight Wearable Devices to Online Programming of Industrial Dual-Arm Robots,” in *2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2019. [**Selected Paper**]
- Q Yuan, **CY Weng***, ZJ Lim and IM Chen, “A Mirrored Motion Remapping Method in Telemanipulation-based Face-to-Face Dual-arm Robot Teaching,” in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2019.
- **CY Weng**, W Yin, ZJ Lim and IM Chen, “A Framework for Robotic Bin Packing with a Dual-Arm Configuration,” in *The 15th IFToMM World Congress*, 2019. [**Finalist: Best Paper Award**]
- **CY Weng**, WC Tan and IM Chen, “A Survey of Dual-Arm Robotic Issues on Assembly Tasks,” in *ROMANSY 22 – Robot Design, Dynamics and Control*, 2018.
- WC Tan, **CY Weng**, Y Zhou, KH Chua and IM Chen, “Historical Data is Useful for Navigation Planning: Data Driven Route Generation for Autonomous Ship,” in *2018 IEEE International Conference*

on Robotics and Automation (ICRA), 2018.

- **CY Weng** and IM Chen, “A vision-based strategy for a cost-effective flexible robotic assembly system without using RCC devices and compliant control,” in *2017 International Symposium on Robotics and Mechatronics, 2017.*
- **CY Weng** and IM Chen, “The task-level evaluation model for a flexible assembly task with an industrial dual-arm robot,” in *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), 2017.*

* Co-first author