UNIVERSITY OF BIRMINGHAM

DOCTORAL THESIS

A CAD-based Integrated Robotic Disassembly Sequence Planning System

Author: Kaiwen JIANG

Supervisor: Professor Duc Truong PHAM

A thesis submitted for the degree of

Doctor of Philosophy

Department of Mechanical Engineering
College of Engineering and Physical Sciences
University of Birmingham

May 2024

For my dearly passed grandpa, Changqing Jiang,
I will always treasure your gift to me in my five,
that was my first metal mechanical pencil set.

I miss you,
infinity times infinity.

Abstract

In a circular economy (CE), products and materials remain in use for as long as possible. Based on the traditional linear economic system, a CE creates a closed loop economic system by reconnecting end-of-use resources, materials, products to the market. As part of a CE, remanufacturing returns the previously sold, leased, used, worn, or non-functional product or component to “like-new” or “better-than-new” condition. Disassembly, cleaning, inspection, repair/replacement, reassembly, and testing are the main steps in remanufacturing. According to the literature, the degree of automation in disassembly is low, with manual disassembly being performed in the majority of cases. Automating disassembly requires automating disassembly planning. Digitalisation is a promising technique for accelerating automation in disassembly planning. Computer-aided design (CAD) is a well-established technology that has been used in digitalising assembly planning. However, few attempts at adopting CAD modelling and simulation can be found in disassembly planning. Starting from collecting disassembly information through a CAD model of the end-of-life (EoL) product,

this work aims to build an integrated system for applying CAD static and dynamic data extraction techniques to remanufacturing disassembly sequence planning. Disassembly plans generated by the system are intended for an industrial robot as the disassembly operator to increase the eventual degree of automation in disassembly. This system further adopts an intelligent metaheuristic optimisation algorithm, the Bees Algorithm (BA), to generate the optimised disassembly sequence for the product. The system places particular emphasis on data mobility and transferability.

The thesis first explains the overall design of the system, then introduces the method of collecting disassembly information from the CAD model. According to a pre-set optimisation objective which is the minimal total disassembly time of the EoL product, disassembly information includes the number of components, names of components, basic disassembly time for each component, etc. An add-in tool is designed to retrieve this information automatically.

The second part of the thesis describes the generation of the product's space interference matrix required for collision-free disassembly planning. In the CAD environment, a collision detection function is applied on each part to capture its interference relationship with the other parts. The results are collected by an algorithm into the space interference matrix of the product.

In the last part of the thesis, the collected product disassembly information and the generated space interference matrix are used as the input of the disassembly sequence optimisation algorithm. Using the enhanced discrete version of the Bees Algorithm, the overall disassembly sequence of the product is optimised, and the best combination of parameter settings is evaluated by a set of optimisation performance measurement standards.

Acknowledgement

I would like to thank my PhD supervisor, Professor Duc Truong Pham. During my Master study time in 2017-2018, Professor Pham made me very interested in his synoptic class discussing about Artificial Intelligence with his professional knowledge and enthusiastic teaching attitude. I am determined to be his PhD student to continue researching this field. Between year of 2018-2024 of my PhD study time, Professor Duc Truong Pham's academic supervision and inspiration directed my research on the topics of remanufacturing robotic disassembly and metaheuristic algorithm optimization. His profound knowledge and rigorous academic attitude pointed me to my ideological confusion and led me to finish my PhD journey.

I would like to thank all the members in AutoReman group for unselfishly sharing their mechanical automation knowledge and robot operation skills. I would like to express my sincere gratitude to Dr Chunqian Ji, who led me stepping into the first PhD research topic about Data Extraction to Computer-aided Design Model;

to Dr Jun Huang, who supervised my Master's final project on the Robotic Unscrewing Search Strategy and kept providing me academic support including presenting work and hosting session on the International Conference; to Dr Yongjing Wang and Dr Mo Qu, who are the project co-supervisor and coordinator, involved me to the Electric Vehicle Battery Disassembly project and provided me with detailed practical training on KUKA and TechMan robot operations.

I would like to thank all the members of the Bees Algorithm group for sharing their expertise in computer programming knowledge and international workshop organization. I would like to express my special gratitude to Dr Natalia Hartono, Dr Asrul Harun Ismail, Dr Sultan Zeybek and Dr Mario Caterino for their company studying and using Combinatorial Optimization algorithms to solve the travelling salesman and vehicle routing problem. Meanwhile, I will always treasure the two-year committee time organizing the BAA workshops.

A big “Thank you!” also goes out to all the lab colleagues, study visitors and friends from other research groups during my past five years, including Dr Mairi Kerin, Dr Joey Lim, Dr Nathinee Theinnoi, Dr Marcel Micheli, Dr Shuihao Xu, Faraj Altumi, Adeyemisi Gbadebo, Feifan Zhao, Chaozhi Liang, Dr Ruiya Li, Dr Yongquan Zhang, Wupeng Deng, Yuqi Wang, Dr Nour Aljafari, Dr Fatma Seyma, Dr Yana Liang, Dr Rongge Guo, and Dr Yang Ma for all their wise advice in both academic and life aspects and for being there to listen when I needed an ear.

Words cannot express the feelings I have for my grandparents - Changqing Jiang and Meishuang Guo, my parents - Tao Jiang and Meili Song, and my sister - Kaixi Jiang for their constant unconditional support, emotionally and financially. I would not be here if it were not for you. Thank you for having faith in me during this challenging journey.

Last, I am also grateful for all the years I spent on reading. Every second of that time built a sanctuary for me, braved my courage, and comforted my pains. There were two sentences that I always put in my profile: “You need to earn whatever you want, that is called DESERVE”, and “What doesn’t kill me makes me a fighter”. I so much love these sentences, which give me an infinite strength to conquer all the pebbles in my past. Although there was more than once that I wanted to give up my PhD life, if time flies back, let me choose again whether to start a PhD, I will say infinite times of the promise that “YES I DO”.

Content

Abstract i
Acknowledgement iv
Content viii
Abbreviations, Acronyms and Symbols x
List of Figures xiv
List of Tables xvi
Chapter 1 Introduction 1
1.1 Background 2
1.2 Hypothesis and research questions 15
1.3 Aim and objectives of the research 16
1.4 Methodology 17
1.5 Thesis outline 19
Chapter 2 Literature review 22
2.1 Literature descriptive analysis 23
2.2 Disassembly sequence planning 27
2.3 Role of CAD in DSP 40
2.4 Metaheuristic optimisation algorithms for DSP 44
2.5 Summary 45
Chapter 3 Disassembly data extraction for the CAD models 47
3.1 Problem preliminaries and system design 48
3.2 Disassembly information collection 51
3.3 CAD model data extraction 56

3.4 Summary 65
Chapter 4 RDSP interference matrix generation from CAD collision detection 67
4.1 Bounding box formation on the CAD model 68
4.2 CAD part movement with collision detection 73
4.3 CAD simulation and space interference matrix generation 75
4.4 Summary 84
Chapter 5 Robotic disassembly sequence planning and optimisation 86
5.1 Generation of feasible robotic disassembly sequences 87
5.2 Robotic disassembly sequence optimisation and the Bees Algorithm 99
5.3 Applying the Enhanced Discrete Bees Algorithm to RDSP 101
5.4 Optimisation algorithm performance evaluation 107
5.5 Results 111
5.6 Summary 115
Chapter 6 Conclusion 117
6.1 Summary of the achievements 118
6.2 Contributions 121
6.3 Future work 123
References 125
Appendix A 1 145

Abbreviations, Acronyms and Symbols

List of Abbreviations and Acronyms

AI	Artificial Intelligence
ALB	Assembly Line Balancing
API	Application Program Interface
APP	Assembly Path Planning
AR	Augmented Reality
ASP	Assembly Sequence Planning
BA	Bees Algorithm
BOM	Bill of Material
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CE	Circular Economy
CEP	Circular Economy Performance

CPS	Cyber-Physical System
DfD	Design for Disassembly
DoE	Design of Experiment
DSG	Disassembly Sequence Generation
DSO	Disassembly Sequence Optimisation
DSP	Disassembly Sequence Planning
DT	Digital Twin
EDBA	Enhanced Discrete Bees Algorithm
EoL	End-of-Life
EVB	Electric Vehicle Battery
I4.0	Industry 4.0
IoT	Internet of Thing
MR	Mixed Reality
NP	Non-deterministic Polynomial
PCB	Printed Circuit Board
RDSP	Robotic Disassembly Sequence Planning
SDSP	Smart Disassembly Sequence Planning
VR	Virtual Reality
WEEE	Waste Electrical and Electronic Equipment
XR	X-Reality

List of Symbols

$\boldsymbol{\theta}$	the rotational angle
$\boldsymbol{B}$	the bounding box
$\boldsymbol{BT}$	the disassembly basic time of the component
$\boldsymbol{c}$	the interference counter
$\boldsymbol{d_j}$	the disassembly direction
$\boldsymbol{D}$	disassembly directions
$\boldsymbol{DetComp}$	the detachable component set
$\boldsymbol{DetDirect}$	the detachable direction set
$\boldsymbol{fit_{DSG}}$	the fitness of initialisation
$\boldsymbol{fit_{DSOiter1}}$	the fitness of first iteration
$\boldsymbol{fit_{DSOconverg}}$	the optimised fitness
$\boldsymbol{F}$	the fixed status of CAD parts
$\boldsymbol{F(Time)}$	the fitness function of total disassembly time
$\boldsymbol{iter}$	the max-iteration
$\boldsymbol{I_t}$	the transform iteration
$\boldsymbol{m}$	number of selected sites
$\boldsymbol{mb}$	number of selected site bees
$\boldsymbol{M}$	the CAD assembly model
$\boldsymbol{M(Component)}$	the component space interference matrix
$\boldsymbol{M(Direction)}$	the directional space interference matrix

$\boldsymbol{M(Direction)_{sum}}$	the directional summation matrix
$\boldsymbol{M(PF)}$	the penalty function matrix
$\boldsymbol{M(Total)_{sum}}$	the total summation matrix
$\boldsymbol{MT}$	the disassembly moving time of the robot end effector
$\boldsymbol{n}$	number of elite sites
$\boldsymbol{nb}$	number of elite site bees
$\boldsymbol{p}$	the translation vector
$\boldsymbol{PT}$	the disassembly penalty time of the product
$\boldsymbol{PT_{tool}}$	the penalty tool change time
$\boldsymbol{PT_{direction}}$	the penalty direction change time
$\boldsymbol{R_{(\theta)}}$	the rotation submatrix
$\boldsymbol{R_t}$	the recovery iteration
$\boldsymbol{s}$	the scaling factor
$\boldsymbol{scoutn}$	number of scout bees
$\boldsymbol{T}$	transformation matrix
$\boldsymbol{x}$	the number of components
$\boldsymbol{x_i}$	the component in the EoL product

List of Figures

FIGURE 1.1 THE RESEARCH BACKGROUND OF THIS THESIS (INTERSECTION C: FOCUS OF THIS THESIS)....14
FIGURE 1.2 THE OVERALL STRUCTURE OF THIS THESIS. 19
FIGURE 2.1 A DISTRIBUTION OF RELEVANT ARTICLES BY YEAR OF PUBLICATION IN THE DSP. 24
FIGURE 2.2 THE CAD TECHNIQUES AREA: A DISTRIBUTION BY YEAR OF PUBLICATION. 25
FIGURE 2.3 THE METAHEURISTIC AREA: A DISTRIBUTION BY YEAR OF PUBLICATIONS. 26
FIGURE 2.4 COMPARISON OF MODELLING METHODS FOR DISASSEMBLED PRODUCTS....32
FIGURE 3.1 THE WORKFLOW OF THIS RESEARCH....49
FIGURE 3.2 DISASSEMBLY TOOLS FOR ADAPTING DIFFERENT TASK REQUIREMENTS. 54
FIGURE 3.3 ONE-SIDED FIXED EOL PRODUCT WITH THE ROBOT END-EFFECTOR APPROACHING. 55
FIGURE 3.4 THE EV BATTERY: (A) LAB PHOTO; (B) CAD MODEL....57
FIGURE 3.5 AN OVERALL VIEW OF THE DEVELOPED SOLIDWORKS ADD-IN TOOL. 58
FIGURE 3.6 A ZOOM-IN VIEW OF THE ADD-IN TOOL'S COVER PAGE. 59
FIGURE 3.7 THE PAGE-CHANGING SCHEME OF THE DEVELOPED ADD-IN TOOL. 60
FIGURE 3.8 AN OVERALL VIEW OF THE ADD-IN TOOL'S ASSEMBLY PAGE. 61
FIGURE 3.9 A ZOOM-IN VIEW OF THE ADD-IN TOOL'S ASSEMBLY PAGE. 62
FIGURE 3.10 AN OVERALL VIEW OF THE ADD-IN TOOL'S PART PAGE: A COMPONENT....63
FIGURE 3.11 A ZOOM-IN VIEW OF THE ADD-IN TOOL'S PART PAGE: A COMPONENT. 64
FIGURE 4.1 THE FLOWCHART OF THE CAD COLLISION DETECTION METHOD TO GENERATE THE SPACE INTERFERENCE MATRIX. 69
FIGURE 4.2 A BOUNDING BOX MODEL: (A) A SPHERE. (B) LOCATING BOUNDING BOX VERTICES. (C) USING EDGE LINES TO CONNECT BOUNDING BOX VERTICES. 70

FIGURE 4.3 A SCHEMATIC DIAGRAM OF THE BOUNDING BOX 71
FIGURE 4.4 A SIMPLE CASE STUDY: (A) CAD MODEL; (B) BOUNDING BOX; (C) EXPLODED VIEW..... 77
FIGURE 4.5 COLLISION DETECTION FOR A CAD BOLT (DIMENSIONAL UNIT: MM). (A) A SUCCESSFUL DISASSEMBLY; (B) DETECTING A COLLISION. 79
FIGURE 5.1 RELATIONSHIPS OF PARAMETERS IN THE BA: (A) BEE ASSIGNMENT; (B) SITE DISTRIBUTION..... 100
FIGURE 5.2 THE LOCAL SEARCH AND GLOBAL SEARCH WORKFLOW IN EDBA ... 103
FIGURE 5.3 THE ORDER OF THE CASES IN THE DESIGN OF THE EXPERIMENT..... 108
FIGURE 5.4 THE BEST FITNESS VALUE AT THREE STAGES AND THE OPTIMISED RATIO BETWEEN THEM 112
FIGURE 5.5 THE AVERAGE NUMBER OF CONVERGENCES ITERATING POINTS IN THE SIX GROUPS. 113
FIGURE 5.6 THE DISTRIBUTION CONVERGENCE ITERATING POINTS IN SIX GROUPS. 114
FIGURE 5.7 THE EXECUTION TIME OF 30 CASES..... 115
FIGURE A.1 AN OVERALL VIEW OF THE ADD-IN TOOL'S PART PAGE: A FASTENER. 147
FIGURE A.2 A ZOOM-IN VIEW OF THE ADD-IN TOOL'S PART PAGE: A FASTENER. 148
FIGURE A.3 GEAR PUMP: (A) CAD MODEL; (B) BOUNDING BOX; (C) EXPLODED VIEW 149
FIGURE A.4 A TEN-PARTS CASE STUDY: (A) CAD MODEL; (B) BOUNDING BOX; (C) EXPLODED VIEW..... 152

List of Tables

TABLE 3.1 A COMPONENT LIST OF THE EVB. 57
TABLE 4.1 THE PLANE CONSTRUCTION OF A BOUNDING BOX 71
TABLE 4.2 THE COORDINATE VALUES OF THE BOUNDING BOX VERTICES 72
TABLE 4.3 THE DIMENSIONS OF A BOUNDING BOX 72
TABLE 4.4 THE EXTRACTED BOUNDING BOX COORDINATE VALUES OF THE SIMPLE CASE STUDY. 83
TABLE 4.5 THE CALCULATED BOUNDING BOX DIMENSIONS OF THE SIMPLE CASE STUDY 83
TABLE 5.1 PARAMETERS REQUIRED IN THE BASIC VERSION OF THE BEES ALGORITHM 100
TABLE 5.2 DISASSEMBLY INFORMATION STORED IN THE SCOUT BEES 101
TABLE A.1 DSP REVIEW PAPERS PUBLISHED SINCE 1987. 145
TABLE A.2 THE EXTRACTED BOUNDING BOX COORDINATE VALUES OF THE GEAR PUMP 150
TABLE A.3 THE CALCULATED BOUNDING BOX DIMENSIONS OF THE GEAR PUMP. 151
TABLE A.4 THE EXTRACTED BOUNDING BOX COORDINATE VALUES OF THE TEN-PARTS CASE STUDY. 153
TABLE A.5 THE CALCULATED BOUNDING BOX DIMENSIONS OF THE TEN-PARTS CASE STUDY. 154

Chapter 1 Introduction

This chapter contains five sections. The first section introduces the research background, laying a foundation for this thesis. The topics include circular economy (CE), remanufacturing disassembly, robotic disassembly automation, and computer-aided design (CAD) simulation. The second section, based on the background investigation, presents a research hypothesis and corresponding research questions. The third section explains research objectives. The fourth section describes the methodologies in this work. The fifth section, using an outline, highlights the key points in the following chapters. An overall structure of the thesis is also provided in the fifth section.

1.1 Background

1.1.1 Circular economy and remanufacturing engineering

The circular economy is an economic system aimed at eliminating waste and the continual use of resources [1], [2], [3], [4], [5]. At the same time, the traditional economy follows a linear model summarised as “take, make, and dispose” [1], [2]. CE contrasts with the conventional linear economy. In the last few years, the concept of CE, which has been defined as a global economic model to minimise the consumption of finite resources [1], [4], [5], has been established, with a focus on the intelligent design of materials, products, and systems [6]. In a CE system, resources are kept in use for as long as possible by extracting the maximum value from them and then recovering and regenerating products at the end of their service life. This model is underpinned by three core principles: designing waste and pollution, keeping products and materials in use, and regenerating natural systems [7]. The CE has gradually received increasing amounts of attention since the last decade. Factories are increasing capital investments by adding circular processes, thus producing sustainable products [6]. Manufacturing enterprises adopt closed-loop life cycle models to support a circular economy strategy [8]. Stakeholders are shifting towards CE-oriented businesses as well. Sustainable performance through business processes is now considered by manufacturing firms [7].

CE principles aim to increase resource use efficiency to achieve a better balance and harmony between the economy, environment and society [7]. Traditionally, CE is based on the so-called 3R principles, i.e., reuse, remanufacturing, and recycling [2], [9], [10], [11], [12], which aim to reuse the product directly at the end of its life cycle; remanufacture an used product to restore it to its original performance, specifications, and warranty; and recycle the product to reuse its materials and reduce resource consumption and pollution generation [7]. Properly recycling and utilising harmful and hazardous substances contained in waste electronic and electric equipment significantly decreases environmental pollution [7]. Recovering materials, especially valuable materials, has become an obligation due to the limited availability of natural resources on Earth [8].

The benefits of adopting a CE are multifaceted, touching on environmental, economic, and social spheres. Environmentally, CE contributes to significant reductions in waste, greenhouse gas emissions, and the depletion of natural resources, fostering a more sustainable interaction with our planet [2], [3]. Economically, it offers businesses opportunities for growth through innovative business models, product longevity, and the potential for reduced material costs [1], [13]. Socially, CE can lead to the creation of jobs and the development of new skills required for a sustainable future, improving societal resilience against resource scarcity and environmental crises [4], [9].

The CE business model is opening new possibilities for organisations in terms of cost and environmental impact optimisation, which will be crucial for the survival of organisations in the future [7]. Industry 4.0, as a technical approach, can support manufacturing organisations in transforming their business model from a linear to a circular economy [1], [14]. An investigation assessed 278 Indian manufacturers with an integration of Industry 4.0 technologies based on their circular economy performance (CEP) in 2023 [1]. The results show that these organisations greatly benefited from green procurement and remanufacturing activities. CE models allow the creation of highly skilled jobs and economic growth, save on the disposal costs that manufacturers must support by law and promote economic savings by recovering the residual value contained within products after their first useful life [7].

CE encompasses a wide array of practices and strategies, including sustainable design, maintenance, repair, reuse, refurbishment, remanufacturing, and recycling [2], [4], [5], [9]. These components work in synergy to keep products, components, and materials at their highest utility and value. The emphasis is on extending the life cycle of materials and products, thereby reducing the need for new resources. Remanufacturing represents a critical endpoint in this sequence before materials are ultimately recycled, signifying a process that brings used products back to like-new or better conditions [9], [11], [12].

Remanufacturing is a comprehensive and rigorous industrial process by which a previously sold, worn, or nonfunctional product is returned to a 'like-new' or 'better-than-new' condition from both a quality and performance perspective [9], [15], [16], [17]. This process involves disassembly, cleaning, inspection, repair, replacement of parts, reassembly, and testing to ensure that the remanufactured product meets or exceeds the original specifications [18], [19], [20], [21]. Remanufacturing differs from other recovery processes by offering a higher value addition, essentially providing second life to products with potentially lower environmental impacts than new manufacturing.

Due to the shortage of global resources and the vision of reducing environmental pollution, remanufacturing has attracted much attention for facilitating sustainable development [22]. Companies dealing in sales and imports were obligated to collect and process used equipment [8]. Remanufacturing is defined as an essential closed loop cycle [4], a central ‘value-retention process’ [10], which plays a vital role in the CE. The design for remanufacturing is identified as an important factor in the CE [10]. The remanufactured products, such as broken or regularly replaced items, are returned to the market in an as-new or better condition. The ecological, social, and economic aspects of remanufactured products are superior to those of newly manufactured products [10].

The achievements in remanufacturing are substantial, offering both tangible and intangible benefits. From an environmental perspective, remanufacturing significantly reduces the demand for raw materials, lowers energy consumption, and minimises greenhouse gas emissions compared to new manufacturing processes. Economically, it allows companies to utilise existing materials and components, reducing production costs and offering consumers high-quality products at a lower price. For example, the automotive remanufacturing industry has a large aftermarket in Europe. Automotive remanufacturing has a long history of development, a large group of employees, and a billion-level turnover [10]. Manufacturing factories, industry companies, and decision makers are encouraged to adopt remanufacturing [5]. Remanufacturing support can smooth the transformation of the business mode from a linear economy to a circular economy, boosting the economy's circularity. Additionally, remanufacturing fosters innovation in product design and production processes, encouraging manufacturers to design products with future remanufacturing in mind. Socially, it contributes to job creation in sectors related to the collection, processing, and remanufacturing of used products, thus playing a vital role in the transition towards a more sustainable and circular economy. Digitalisation can be achieved through remanufacturing to boost CE [3]. Currently, assumptions and secondary data are the main information resources in the CE. The design of circular process theory and the design for remanufacturing are deeply influenced by the lack of empiricism [10]. Digital manufacturing should be implemented within the

framework of industrial ecology and remanufacturing principles to increase CE efficiency, productivity, and traceability.

1.1.2 Product disassembly in remanufacturing

Today, remanufacturing is a pivotal industrial process recognised for its potential to significantly reduce waste and conserve resources. It has gained traction across diverse sectors, including automotive, aerospace, electronics, and heavy machinery [10], [11], [12], [19], [23], [24], owing to its environmental and economic benefits. This process contributes to reducing the extraction of raw materials and energy consumption and presents opportunities for businesses to realise cost savings and foster innovation. However, the adoption and integration of remanufacturing practices vary widely among industries and regions and are influenced by technological advancements, regulatory frameworks, market acceptance, and the availability of returned products. The current state of the remanufacturing industry is different between developed and developing nations. Successful and well-practised remanufacturing is being implemented in countries such as the USA, the UK, and Germany. In contrast, the expected growth in remanufacturing has begun in many sectors among nations, such as India, Brazil, and China [5].

The remanufacturing process is intricate and multifaceted, commencing with disassembling the returned product. This initial step involves the systematic separation of the product into its constituent components and materials, allowing for the identification of reusable parts [17], [19], [25]. Following disassembly, the process includes cleaning, inspecting, and sorting parts based on their condition [10]. Subsequently, necessary repairs or replacements are made, wherein parts may be refurbished or entirely substituted with new parts. The remanufactured product is then reassembled and subjected to rigorous testing to ensure that it meets the original specifications or better. Finally, the product is repackaged and made ready for resale. This sequence underscores the complexity of remanufacturing and its potential to transform waste into valuable products. Disassembly in automotive remanufacturing is identified as an inevitable process after sorting and inspection and before cleaning, reworking, assembly and testing [10]. Products are recovered to obtain raw materials or semifinished products for production through recycling, disassembly, sorting and reengineering [8].

Disassembly plays a crucial role in achieving sustainability in the CE. The mass personalisation of products requires the flexibility of disassembly to accommodate frequent changes [22]. The 3R principles of CE are eased practically by disassembly processes that allow the disassembly of subcomponents for the reuse, remanufacture, and recycling of materials. Disassembly is facilitated by proper product design, which is fundamental for

applying CE principles [7]. The product needs to be designed to ease the disassembly process. Product design can strongly influence disassembly times and processes [7].

Disassembly, the precursor to remanufacturing, entails the systematic breakdown of a product into its components and materials. It is a critical step in determining the efficiency and effectiveness of the remanufacturing process. Disassembly processes are considered the most essential stage for the adoption of specific actions in end-of-life (EoL) management practices [6]. Disassembly requires a deep understanding of the product's design, including how components are interconnected and the best sequence for their removal to maximise the recovery of usable parts. It breaks down EoL products into valuable components to be reused in the future and is thus considered vital in remanufacturing [22]. Until 2024, an investigation using the engineering data of European automotive remanufacturing figures that the degree of automation in the disassembly process is 70% manual and 30% semiautomated [10]. The fully automated ratio is zero. Overall, of all the remanufacturing steps (sorting, inspection, disassembly, cleaning, reworking, assembly, and testing), the fully automated process accounts for only 13%. This task demands precision and expertise, as improper disassembly can damage components, rendering them unusable and thereby defeating the purpose of remanufacturing.

Disassembly sequence planning plays a crucial role in reusing and remanufacturing EoL products [7]. It serves as the linchpin for the entire remanufacturing process, influencing the components' recovery rate, the operation's cost-effectiveness, and the overall environmental footprint. Effective disassembly planning requires the identification of the optimal sequence for part removal, which minimises labour and time while preventing damage to reusable components. Since the efficient utilisation of existing resources is necessary, it is crucial to predict disassembly operation times and the condition of joints for recycling, reusing or remanufacturing [8]. Technological advancements, such as CAD systems and automation, greatly aid this planning, which can simulate disassembly processes and identify bottlenecks. Moreover, an efficient strategy must be adaptable and capable of accommodating products of varying conditions and complexities.

1.1.3 Robotic disassembly and automation

Remanufacturing is expected to have more flexibility and a high degree of automation. As one of the remanufacturing process steps, cleaning has the highest degree of automation, with 77%. According to remanufacturing engineering data investigations, the proportion of disassembly semi-automation is 30%, and disassembly currently has no full automation [10]. Disassembly has excellent

potential for automation in remanufacturing engineering, in addition to other steps, such as cleaning and assembly. Automation in remanufacturing disassembly requires more flexibility regarding the volumes, variants, and diversity of EoL products. In addition to digitalisation in remanufacturing, innovative remanufacturing technologies indicate a potential research-practice gap, which has already been demonstrated in research [12].

Traditionally, manual disassembly has relied heavily on human expertise and experience [26], [27], [28], [29], [30]. Humans excel in adapting to unstructured scenarios and making judgments based on qualitative assessments. Their flexibility and cognitive capabilities enable them to adeptly manage complex disassembly tasks, especially those requiring delicate manipulations or involving products characterised by high variability and low volumes. Despite these advantages, this approach is constrained by various factors, including inconsistency in human performance, the potential for injuries, and the inefficiencies associated with manual labour.

However, using human operators to perform disassembly tasks has drawbacks. First, human performance efficiency in disassembly tasks can significantly fluctuate due to fatigue, expertise deficiency, and subjective decision-making. This inconsistency can adversely impact the disassembly process, affecting the quality and output of remanufactured products. Second, disassembly tasks

frequently involve handling sharp, heavy, or hazardous components. Manual execution of these tasks poses significant safety risks to human operators, including injuries such as cuts, strains, and repetitive stress disorders. Third, manual disassembly is both time-consuming and labour-intensive. The physical limitations of human operators restrict the speed and duration of disassembly activities, thus negatively impacting productivity and cost efficiency.

Therefore, the integration of robots into disassembly introduces numerous benefits over traditional human-centric methodologies [31], [32], [33]. Robots, known for their precision, consistency, and indefatigability, offer a compelling solution to the limitations inherent in human labour. Robots can execute disassembly tasks with remarkable accuracy and uniformity, mitigating the performance variability typically observed in humans. This consistency is crucial for upholding the quality standards of remanufactured products. In addition, assigning hazardous disassembly tasks to robots significantly decreases the risk of injuries among human workers. Robots can safely manage sharp, heavy, and dangerous materials more efficiently than can humans. Robots can operate continuously without succumbing to fatigue, markedly enhancing the throughput of disassembly processes. The efficiency and speed of robotic systems can vastly exceed those of manual disassembly, leading to substantial cost savings and improved productivity.

1.1.4 Gap identification: lack of CAD simulation

In disassembly sequence planning (DSP), a crucial component of remanufacturing and circular economy strategies is the integration of CAD simulations, which represents a promising frontier. However, a discernible research gap persists in fully leveraging CAD simulation capabilities to address DSP challenges effectively. Although CAD tools have been extensively applied in the design and manufacturing phases [10], [34], [35], their potential in simulating and optimising disassembly processes has yet to be fully realised or explored. Considering all EoL management practices, disassembly has been identified as strategic in the circular economy. Only scattered attempts have explored how digital technologies (especially simulations) can support CE adoption by focusing on disassembly processes [6].

Current literature highlights using CAD for static product analysis but often overlooks dynamic simulation features that could detect potential disassembly hurdles, such as accessibility issues and tool requirements. Moreover, studies that integrate digitalisation techniques, such as CAD simulations, and with advanced optimisation algorithms, such as metaheuristic methods, to generate practical and efficient disassembly sequences are scarce [10]. The incorporation of real-time data from the disassembly floor into CAD-based DSP models is another

underexplored area that could enhance the adaptability and accuracy of disassembly plans.

1.1.5 A short summary

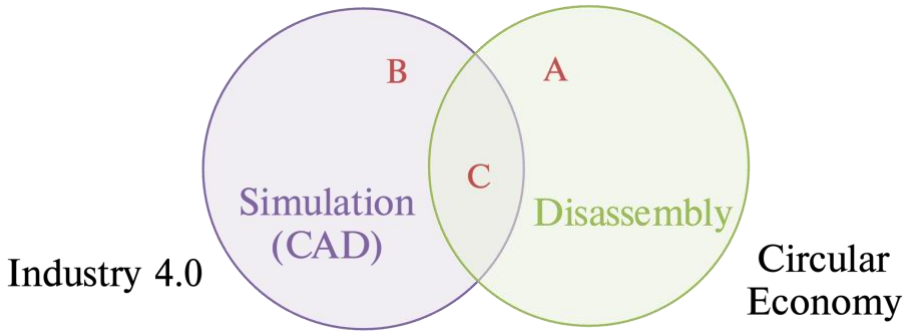


Figure 1.1 The research background of this thesis (intersection C: focus of this thesis)

The circular economy offers numerous benefits for environmental protection, enterprise development, and energy conservation. In addition to reducing and recycling, remanufacturing plays a crucial role in advancing circular economic practices as a core principle of the circular economy. The initial and most critical step in remanufacturing engineering is disassembly, the advancement of which can significantly contribute to the sustainable development of the circular economy. Automating disassembly processes, mainly through the use of robots instead of manual labour, has considerable advantages in terms of efficiency and safety. However, integrating CAD technology into robotic disassembly systems, specifically for the sequence planning of dismantling discarded products, remains

a relatively underexplored area in the literature and forms the primary focus of this thesis. Referring to Figure 1.1, the above section provides a detailed background on disassembly in the CE (section A) and CAD simulation in Industry 4.0 (section B). Intersection C is the research focus of this thesis.

1.2 Hypothesis and research questions

According to the introduced background of the circular economy, the significant steps and status of disassembly in the remanufacturing area, and the gap in CAD model simulations for providing necessary data and information, the research hypothesis is presented below:

"CAD simulations can provide **sufficient** disassembly information and data, which can **be integrated** into a robotic disassembly sequence planning (RDSP) system to generate an **optimised** disassembly sequence."

The following research questions are raised to present the evidence and support the hypothesis:

1) Identification of the disassembly information and data: What information and data can the CAD models provide for disassembly work?

2) Integration of the disassembly information and data into the RDSP: How should the extracted information and data be processed in robotic disassembly sequence planning problems?
3) Optimisation and evaluation of the RDSP: How can the most optimised robotic disassembly sequence be generated and evaluated?

1.3 Aim and objectives of the research

This work aimed to build a CAD-based integrated robotic disassembly sequence planning system to reduce the redundancy of the information transformation process in the integrated disassembly system and increase the degree of automation. The following objectives guided the research:

1. A disassembly sequence planning system for EoL products is designed, including to set the preliminary goal of the disassembly operator and the system optimisation objective.
2. Recognise the necessary disassembly information according to the system optimisation objective and extract data from the CAD models of the EoL products.

3. A CAD model collision detection method is developed to test the collision conditions between the controlled part and others, and then the space interference matrices are generated from the detected collision results.
4. The space interference matrices are input to the disassembly sequence generation algorithm to obtain the feasible disassembly sequences.
5. The feasible disassembly sequences are optimised to find the one that meets the system optimisation objective. An evaluation metric is designed to determine the best parameter settings.

1.4 Methodology

The methodology used in this research is as follows:

1) Review of previous studies: A comprehensive literature review is carried out to demonstrate thorough awareness of the existing work and the necessity for this research.
2) Define a workflow of a RDSP system: A hierarchy model showing the research processes for solving the RDSP, starting from CAD model analysis.
3) Identification of the disassembly information: The constituent factors of the disassembly information are identified and explained of using the minimal total disassembly time as a single objective optimisation function.

4) Extract the CAD data (static): A software add-in tool is developed to present the static CAD data of CAD models.
5) Process the CAD data (dynamic): An application programming interface (API) controlled collision detection algorithm is developed to measure the interference conditions between parts, and the results are presented in a mathematical matrix format.
6) Generate feasible disassembly sequences: An algorithm is given to present the process of generating feasible disassembly sequences, including corresponding disassembly directions of each disassembly step.
7) Optimise for locating the best disassembly sequence: The Bees Algorithm (BA) in its enhanced discrete version is adopted as the RDSP optimisation algorithm for finding the best disassembly sequence.
8) Evaluate the optimisation performance of the enhanced discrete bees algorithms (EDBA): An evaluation criterion for the optimisation performance of EDBA is designed.

1.5 Thesis outline

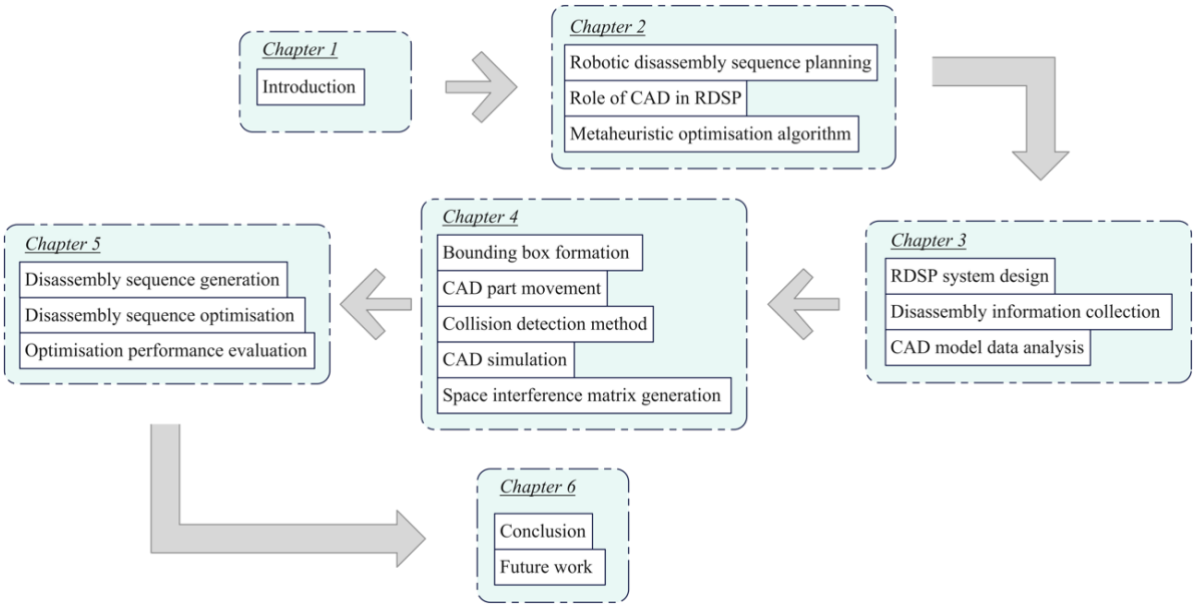


Figure 1.2 The overall structure of this thesis.

The thesis is structured with six chapters. Figure 1.2 presents an overall structure of this thesis, with the achievements stated in each. Apart from the introduction of research background in chapter 1, the rest of this thesis is presented as follows:

- Chapter 2: Literature Review

This chapter provides the research findings in CAD-based robotic disassembly sequence planning and optimisation. A summary of the published disassembly sequence planning review papers is given in the first section. Second, reviews of the matrix-based model representation methods and robot-based operator

selections in DSP are presented. Then, a review of integrating CAD techniques in DSP, which includes CAD models' information extraction and collision detection methods in CAD simulations, is given. Last, different types of metaheuristic optimisation algorithms and their performance evaluations are presented.

- Chapter 3: Disassembly information extraction from the CAD models

This chapter begins the CAD-based RDSP system. First, a macroscopic picture of the designed system shows the overall workflow. The disassembly operator is a single robot, and the system goal is to find the disassembly sequence using the minimised disassembly time. The involved disassembly information is explained in the next step. Third, an add-in tool is developed for presenting the extracted disassembly information of EoL products in CAD environment.

- Chapter 4: Space interference matrix generation using CAD collision detection method

This chapter is the middle part of the whole system. The space interference matrix of an EoL product is generated from its CAD model by detecting collisions between parts in the CAD environment. This process involves several techniques

in the CAD simulation, such as use of the bounding box, measurement of CAD parts, and API program control over the CAD software.

- Chapter 5: Robotic disassembly sequence planning and optimisation

This chapter is the final part of the designed system. The generated space interference matrix from last chapter is a representation of the EoL product, which is taken as an input to the disassembly sequence planning and optimisation processes in this chapter. After finding all the feasible disassembly sequences, the enhanced discrete bees algorithm is applied to find an optimised sequence result. Furthermore, EDBA is evaluated for its optimisation performance to find the best combination of parameter settings.

- Chapter 6: Conclusion

This chapter concludes the contributions and implications of the whole thesis. Future work is also suggested.

Chapter 2 Literature review

This chapter contains five sections. The first section introduces the research approach conducting with a descriptive analysis. The second section reviews the previous work in the DSP problem, including the matrix-based model representation method and types of disassembly operators in the DSP problems. Considering that the DSP problem has grown for decades, a summary of the published review papers of the DSP is given to trace the development progress and shifting viewpoints in the research. The third section examines the usefulness of CAD techniques for representing EoL products in the DSP problems. The fourth section discusses various metaheuristic optimisation algorithms applied to DSP problems. The selection of optimisation algorithms and their performance evaluation are the two highlights of the work. The fifth section summarises the findings from the above three areas in the DSP and provides further motivation to formulate this thesis research.

2.1 Literature descriptive analysis

This section explains the literature review's extraction, screening, and selection processes. This review is transdisciplinary research that falls within the DSP topic and is related to the applications of CAD techniques and metaheuristic optimisation methods. Scopus was used to collect literature because of its well-known robust characteristics. Comprehensive content for the above areas is provided. This review was finalised in December 2023, so the materials published up to 2023, including the in-press publications published in early 2024, were selected for this review.

“Disassembly sequence planning” was the search component within three fields: article title, abstract, and keywords. In total, 663 publications were found in Scopus. After excluding irrelevant and duplicated results, 637 genes were further screened. Figure 2.1 illustrates the distribution of relevant articles by year of publication in the DSP topic. The first DSP paper can be traced back to 1987 [36]. The assembly and disassembly planning operations for space telerobotics were discussed. Disassembly was considered as a reversion to assembly by then. And the AND/OR graph, which gradually became a widely adopted method nowadays, was used to represent feasible sequences. Since then, researchers have steadily increased their attention to disassembly sequence planning.

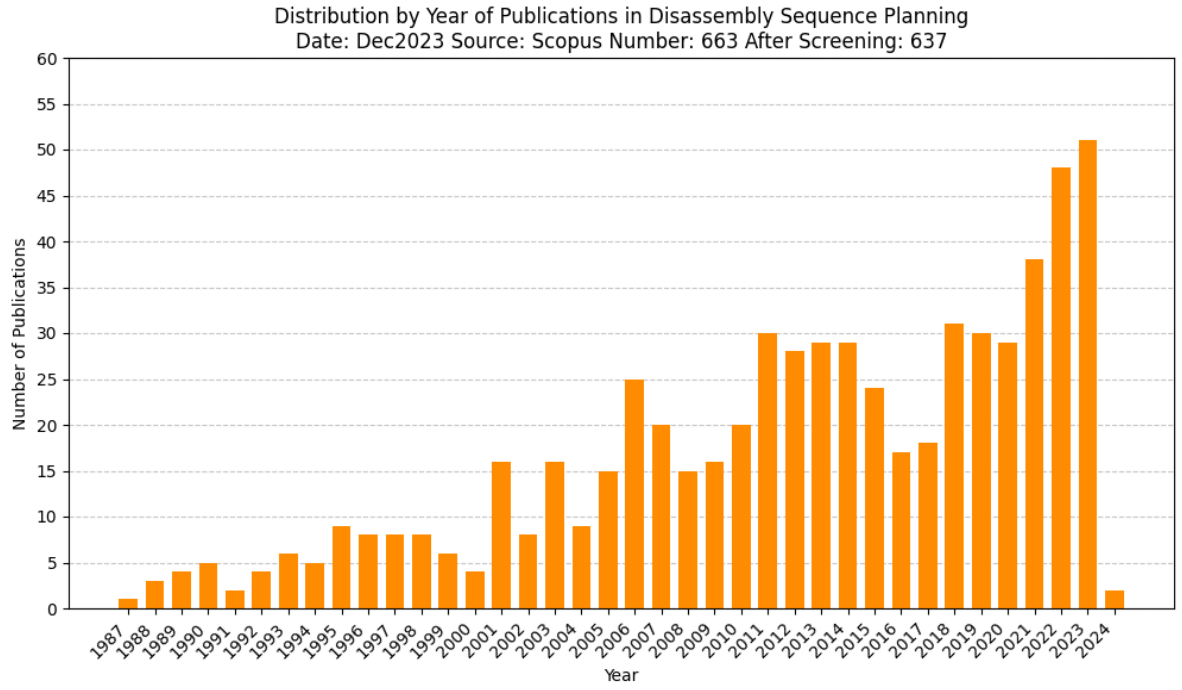


Figure 2.1 A Distribution of relevant articles by year of publication in the DSP.

The feasible disassembly sequence results generated by DSP problems require various input data, which CAD models can provide. Design information, such as bills of materials (BOMs), mating features, dimensions in axes, tools and accessories, can be extracted from a CAD model. To narrow the investigation of the applications of CAD techniques under the DSP topic, the search word “CAD” was added to Scopus, decreasing the number to 80 publications and resulting in 74 publications after screening. Figure 2.2 shows the distribution of publications on the CAD technique under the DSP topic by year. CAD techniques began to support the development of DSP in 1989. Three works were published in 1995 [37], [38], [39], 2001 [40], [41], [42] and 2006 [43], [44], [45]; five, six, and eight

publications were published in 2012 [46], [47], [48], [49], [50], 2017 [51], [52], [53], [54], [55], [56] and 2022 [57], [58], [59], [60], [61], [62], [63], [64], respectively.

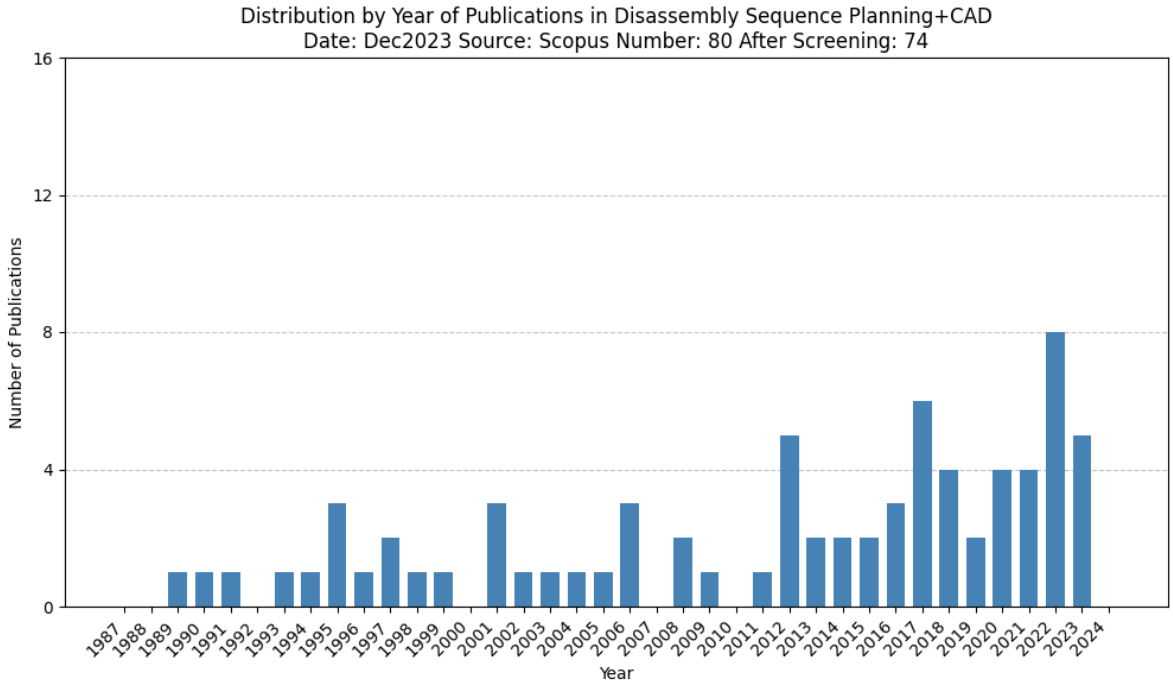


Figure 2.2 The CAD techniques area: a distribution by year of publication.

The disassembly sequence planning problem is a nondeterministic polynomial (NP) problem. Exact methods are difficult to solve NP problems because generating solutions takes a long computational time and requires high resource usage [65], [66]. Metaheuristic methods, which can find an optimum or a near-optimum solution, are more suitable for solving the NP problems [67], [68], [69], [70]. Figure 2.3 shows the search results obtained by adding the search term

“metaheuristics” under the DSP topic in Scopus, with 20 publications in total found after screening.

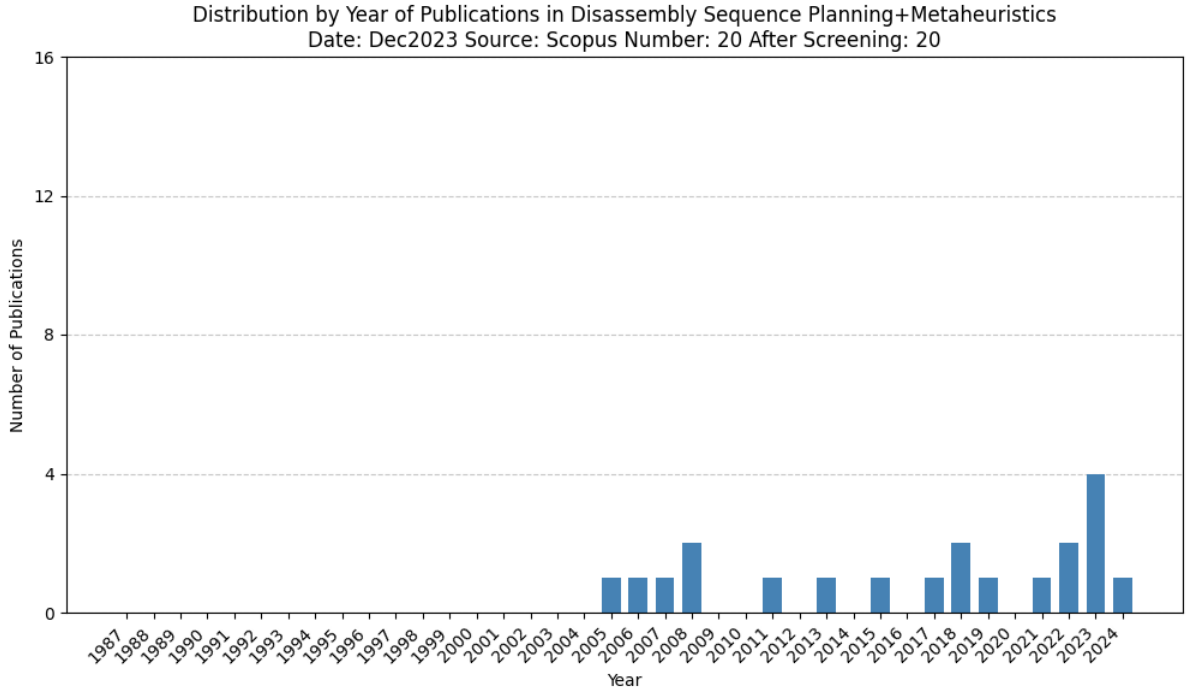


Figure 2.3 The metaheuristic area: a distribution by year of publications.

Additionally, a search utilizing the terms “CAD” and “metaheuristics” was conducted within the context of the DSP topic in Scopus. As of the completion of the literature review for this thesis, no publications have emerged in this interdisciplinary field. This identified research gap underpins the subsequent methodological development presented in this thesis.

2.2 Disassembly sequence planning

2.2.1 DSP review papers

The disassembly sequence planning problem has been studied for decades. Researchers have summarised review papers from various angles since the first paper on the DSP was published in 1987. Up until the work of this thesis, 12 review papers were located based on the Scopus scholar search (see Appendix Table A.1).

Yokota and Brough published the first review paper on assembly automation in 1992 [71]. Beginning with introducing the necessity of cooperative design and production planning in manufacturing activities, an information loop is proposed to send planning information to design engineers. This work discussed the solution in assembly/disassembly sequence planning from five aspects: the planning information of the product, the representation of the assembly structure, the relation types at the part level, the precedence of disconnecting the relations, and the generation of assembly/disassembly sequences. Notably, the sequence planning problem is mixed between assembly and disassembly in this work, while more attention is given to analysing the product's structure. In 1998, O'Shea et al. published a review addressing a state-of-the-art literature survey on disassembly planning [72]. Similar to Lee et al. in 2001, a review and further research

discussion of planning and scheduling issues focused on the disassembly area [73].

In 2003, Dong and Arndt critically reviewed the integration of the CAD technique with disassembly and emphasising the role of CAD in advancing DSP research [74]. They highlighted the importance of assessing product disassemblability to determine disassembly time and costs, utilising CAD data for generating disassembly paths automatically. Their review underlines the significance of disassembly process planning and sequence generation, aiming to optimally deconstruct products into their components. This study outlines four main product representation approaches—graph-based, Petri net-based, geometry-based, and others—noting the use of CAD in Petri net approaches to establish part precedence relationships [75], [76], [77]. This finding underscores the necessity of CAD for effective disassembly planning, aiding in early problem identification and simulation. The analysis concluded that developing CAD techniques is crucial for addressing disassembly process planning challenges, highlighting the value of graphical and geometric models in supporting disassembly modelling within a computer-aided design or computer-aided manufacturing (CAD/CAM) environment.

Path planning is another important element of manufacturing process planning. Three subproblems in assembly planning, namely, assembly sequence planning

(ASP), assembly line balancing (ALB), and assembly path planning (APP), were identified by Ghandi and Masehian in 2015 [78], where disassembly was seen as a reverse version of assembly. The view of considering disassembly to be the opposite of assembly is now partly agreed upon by researchers because of the different characteristics of these two topics, for example, the uncertainty of disassembly problems.

The recent developments and future trends of DSP were reviewed by Zhou et al. in 2018 [79]. The disassembly mode, disassembly modelling and planning methods are identified as the three main steps in DSP. In the disassembly modelling step, preprocessing is defined as the extraction method for obtaining the precedence relationships between parts, which is distinguished from the disassembly model building substep. For example, CAD models are used in this substep because of the known information about the product. However, this method is less commonly used than manual methods, which are the main methods for completing preprocessing because of the strong recognition ability of humans. In addition to the graph-based method, this work introduces the matrix-based method as the main approach to model representation. The disassembly interference matrices derived from the CAD model of the product can be used to generate disassembly sequences. The advantage of using a disassembly interference matrix is that all the relationships of parts can be reflected in the matrices using binary values, whether two parts have interference or not.

Robotic disassembly is emerging as a crucial field in the transition towards strategic recycling and the circular economy, marked by an increasing focus on automating disassembly processes. In 2020, Poschmann, Bruggemann, and Goldmann's systematic review explored how robotic disassembly technologies contribute to this transition, enhancing the digitalisation of the circular economy and streamlining data flow in disassembly processes [26]. They outline a five-step disassembly work chain, emphasising the integration of robotics, especially in disassembly operations. This review identifies sequence planning, robotic applications, vision systems, and path planning as key areas of interest, highlighting the use of CAD models for efficient path planning. This work underlines the importance of robotic disassembly in improving recycling processes and suggests potential research directions for further integration within an information system for the circular economy.

Four major aspects of DSP problems were summarised by Guo et al. in 2021 [80], namely, disassembly modelling, mathematical programming, artificial intelligence, and uncertainty handling. The disassembly modelling aspect is the key to this thesis. Their work revealed that the following disassembly model representation approaches (graph-based, AND/OR graphs, Petri nets, and matrix-based) have been widely used in previous research with proof of their effectiveness. According to this work, product representation is the first basic

element in a disassembly planning system, followed by sequence search and solution optimisation. The product representation builds a foundation for the disassembly sequence search. Selecting a proper representation method to search for disassembly sequences efficiently is important. Among the four modelling methods, the matrix-based approach is specifically focused on in this thesis. Compared to the other three methods, which have the advantage of being intuitively understood by humans, the matrix-based method is less easy to produce manually, especially when facing a complex product with a large number of components. Interpreting and automatically processing the matrix data is recommended and a have-to in the computer system. For example, a product described by graphs always needs to convert component relationship information into a matrix format for further calculation. However, matrix-based model representation requires all relationships of pairwise components, which causes a large amount of matrix data and a long data processing time. This is the disadvantage of using the matrices. Figure 2.4 [80] provides an overall comparison of the modelling methods for disassembled products.

Modeling	Advantages	Disadvantages
Graph-based modeling	1) It represents relationships among components in a product and includes more information of component constraints about a product structure. 2) It is compacted to store and straightforward to analyze. 3) It considers other criteria since other constraints can be added by expressing them logically as precedents. 4) It includes deemed to be quasi-parts, fasteners, and the precedence relations can easily be extended.	1) OR precedence typically cannot be properly represented. Using dummy operations concatenate OR sequences, no single precedence graph encompasses every sequence for most products. 2) Individual operations or logical combinations of operations can be represented by precedence relations themselves in it. It includes all plans, but the set of operations must be fixed, merely their order can change and can only be executed serially. Parallel operations cannot be represented.
AND/OR graph modeling	1) It represents decomposing them into a set of smaller problems to obtain the solution of problems. AND arcs are the decomposition or reduction generates arcs. One AND arc may point to any number of successor nodes. 2) Several arcs may emerge from a single node in an OR graph, indicating a variety of ways in which the original disassembly may be conducted.	1) It is not possible to consider the order of disassembly operations for different subassemblies/components in the same level. 2) The associated disassembly sequence and sequence-dependent operation time cannot be represented. 3) It is not easy to integrate with disassembly resource modeling.
PN modeling	1) PN addresses precedence relations between actions. It is extensively applied in the field of manufacturing/demanufacturing systems. 2) PN is a mathematical and graphical representation of concurrent discrete-event systems. It is to be able to strictly respond to the execution process of a system. It can be analyzed to provide guidance to control a system's execution. 3) It is used to describe asynchronous and concurrent computer system model and has rich system behavior analysis tools and system description means.	1) It is easy to get very large and thus difficult to be analyzed. 2) It needs to be augmented to include time-related information. 3) It lacks linear programming formulation and solution ability.
Matrix-based modeling	1) A matrix-based model contains both disassembly and assembly tasks if all disassembly tasks are reversible. 2) It describes product elements and their relationships. It is used to support the analysis of product functions, interfaces and assembly/disassembly. 3) Depending on applications, various algorithms can be used to manipulate such a model such as clustering, partitioning and coverage.	Decision support can be offered by most matrix-based modeling methods, but not decision automation, and interpretation is difficult.

Figure 2.4 Comparison of modelling methods for disassembled products.

Another product disassembly sequence planning review was presented by Ong et al. in 2021 [81]. The state-of-the-art literature from 2009 to 2021 was summarised in this work, together with the challenges and future directions. Three major research studies on the DSP problem are classified: product representation models, sequencing algorithms, and methodology validation. The disassembly process, which targets different types of operators, has three disassembly mode choices: full manual, full robotic, and human-robot collaboration. This work also classifies disassembly problems by level (complete or partial disassembly) and by type (sequential or parallel disassembly). In addition, this work summarises

the times at which product applications have appeared in publications in the past decade. Most of these studies involved numerical simulations and 3D assemblies, which were performed 14 and 9 times, respectively. Finally, this work discusses the automatic definition of the product representation model. The EoL ontology model can address uncertainty issues in disassembly well. Nevertheless, this model is not practical enough to be widely adopted, as manually labelling the parts is tedious and time-consuming. The CAD model and its BOM files are a better choice for automatically extracting product data within a reasonable time for real applications, including large-size and complex products. This view also coincides with the hypothesis in this thesis.

Sassanelli et al. also published a review in 2021 [6]. This systematic review underscores the crucial role of simulation in advancing both disassembly processes and the adoption of CE principles, particularly within the framework of Industry 4.0. Simulation tools play a pivotal role in optimising disassembly processes, improving product and plant disposal, and strengthening connections between the design and EoL stages. By leveraging Industry 4.0 technologies, such as virtual and augmented reality, simulation enhances human–machine interactions and operational efficiency. This integration of simulation into CE practices not only fosters resource efficiency and sustainability but also empowers manufacturers and researchers with the necessary knowledge and skills to navigate the complexities of modern production systems. As such,

simulation has emerged as a key enabler of CE strategies, driving innovation and progress toward a more circular and sustainable future.

Two more DSP review papers were published in 2023, focusing on the integration of cutting-edge technologies with DSP problems. In 2023, Yang et al. combined X-reality (XR) techniques with DSP problems for EoL aircraft parts, where “X” is an attribute variable. It is a compound word that combines augmented reality (AR), virtual reality (VR), and mixed reality (MR). This study explores the integration of Industry 4.0 (I4.0) technologies, particularly X-reality, and lean management principles into end-of-life aircraft part remanufacturing, with a specific focus on disassembly sequence planning. It addresses the challenges and uncertainties inherent in EoL conditions and proposes "Smart Disassembly Sequence Planning (SDSP)" as a new decision-support agenda. By conducting a comprehensive literature review, this study identified the key concepts of DSP, I4.0, XR, and lean management while reviewing previous efforts in EoL aircraft remanufacturing and XR-assisted DSP. The study concludes that advancing disassembly management presents a significant business opportunity, emphasising the need for abundant evidence for rationalising digital technology adoption. Furthermore, these findings suggest that XR–lean integration has the potential to enhance the digitalisation, quality, and effectiveness of EoL DSP, emphasising the importance of testing, validation, and case studies to verify the method's applicability and usefulness in managerial practices. Future research

directions include the measurement and analysis of efficiency and social sustainability impacts in EoL aircraft parts cases, as well as the ergonomics analysis of human–XR collaboration.

Works using artificial intelligence (AI) techniques to solve DSP problems are summarised and reviewed by Chand and Ravi in 2023 [82]. This study delves into the growing utilisation of AI-based approaches in manufacturing, particularly focusing on DSPs within the context of the remanufacturing and waste management industries. This emphasises the importance of DSP in building a circular economy by optimising disassembly processes to reduce ecological and economic impacts. Through an analysis of state-of-the-art DSP methods, this paper provides insights into disassembly objectives, attributes, and optimisation techniques. This highlights the significance of efficient DSP techniques in minimising raw material usage, exploiting natural resources, and enhancing waste management processes. This paper reviews research articles published between 2012 and 2021 discussing DSP problem-solving techniques, constraints, modelling, and algorithms, with a focus on nature-inspired and hybrid algorithms. It identifies areas for future research, including parallel and incomplete disassembly processes, environmental impact assessment, and the integration of AI, the Internet of Things (IoT), and edge computing in remanufacturing operations. Additionally, it proposes the exploration of human-robot combined

disassembly operations and suggests a consolidated approach based on computational intelligence for optimising the entire disassembly process.

2.2.2 Model representation methods: matrix-based

According to the DSP review papers in the last section, model representation is an important part of solving DSP problems. Graph-based, Petri nets, and matrix-based methods are the mainstream solutions. Among these, matrix-based methods have emerged as particularly advantageous over other common techniques, especially in solving complex DSP problems. Matrix-based methods fundamentally leverage the simplicity and mathematical robustness of matrices to represent the relationships and dependencies among components within an assembly. This representation is instrumental in identifying feasible disassembly sequences, considering constraints such as precedence relations and tool requirements. The primary advantage of matrix-based approaches lies in their computational efficiency. Unlike graph-based or Petri net-based methods, which often require extensive computational resources for graph traversal or state space exploration, matrix operations can be highly optimised for speed. This efficiency is paramount in real-time or near-real-time DSP applications, where decision-making speed is crucial for operational effectiveness.

Another salient feature of matrix-based methods is their inherent suitability for integrating automated systems, particularly those leveraging CAD and CAM technologies. Matrix representations can be directly derived from CAD models, enabling an automated transition from product design to disassembly planning. Some related research has focused on this topic. This integration is less straightforward with graph-based and Petri net-based methods, which may require additional layers of abstraction and conversion to interfaces with CAD/CAM systems. The direct compatibility of matrix-based methods with CAD/CAM tools streamlines the DSP process, enhancing the efficiency of the disassembly planning phase and facilitating a more cohesive workflow from design to disassembly, which builds the basis hypothesis of this work.

Moreover, matrix-based methods offer superior scalability and flexibility in handling complex assemblies. The compact and structured nature of matrix representations allows for the efficient analysis of large-scale systems, where the number of components and disassembly constraints can be significant. This scalability is particularly relevant in industries dealing with complex products, such as aerospace and automotive products, where disassembly processes are critical for maintenance, repair, and remanufacturing activities. In comparison, graph-based and Petri net-based approaches might struggle with scalability issues, as the complexity and size of the graphs or nets increase exponentially with the

number of components, making the analysis and solution derivation more cumbersome.

The adaptability of matrix-based methods to various optimisation criteria and constraints is another merit worth mentioning. By formulating the DSP problem as a matrix optimisation problem, it is possible to apply a wide range of mathematical and heuristic optimisation techniques to find optimal or near-optimal disassembly sequences. This flexibility allows for the consideration of multiple objectives, such as minimising the disassembly time, cost, or environmental impact, thereby supporting the goals of sustainable manufacturing and remanufacturing. While graph-based and Petri net-based methods can also be adapted to different optimisation criteria, the process is often more intricate and less intuitive than that of matrix-based methods.

In conclusion, the exploration of different methodologies for solving DSP problems underscores the pivotal role of method selection in achieving efficient and effective disassembly sequence planning. Matrix-based methods have been increasingly recognised for their potential to address the challenges associated with DSP in a more efficient, flexible, and scalable manner than graph-based and Petri net-based methods. The matrix-based approach, with its computational efficiency, ease of manipulation, direct CAD/CAM integration capabilities,

scalability, and adaptability, presents a compelling case for its preference over graph-based and Petri net-based methods.

2.2.3 The operators in DSP

Disassembly sequence planning (DSP) is a critical component of the remanufacturing process, offering a structured approach to deconstruct products into their constituent parts. The objective of DSP is not only to facilitate recycling and reuse but also to enable efficient remanufacturing, repair, and maintenance activities. Traditional DSPs have predominantly been centred around human operators, given their cognitive flexibility and problem-solving capabilities. However, the advent of robotics has introduced a paradigm shift, with robotic operators offering several distinct advantages over their human counterparts.

Despite the potential merits of HRC, fully robotic systems provide a more streamlined and efficient approach to DSP. These systems are designed for autonomous operation, eliminating the necessity for human intervention and thereby reducing labour costs and safety risks. The automation of DSP through robotics also facilitates the incorporation of advanced optimisation algorithms, further enhancing the efficiency and efficacy of the disassembly process.

In summary, while human operators have historically played a pivotal role in DSP, the limitations of manual disassembly are becoming increasingly evident. Robotic integration into DSP offers significant advantages in terms of precision, safety, efficiency, and adaptability. Fully robotic systems, in particular, represent the future of DSP and are promising for overcoming the challenges posed.

2.3 Role of CAD in DSP

2.3.1 Integration of CAD in the disassembly planning system

The integration of CAD into DSP systems represents a significant leap forward in the field of manufacturing and remanufacturing. This integration not only streamlines the disassembly process but also enhances efficiency, accuracy, and sustainability. In DSP, mathematical programming methods require modelling with a high level of abstraction. The problem description starts from assembly drawing or a CAD file [80]. CAD models can generate structured data in cyberspace for experts to guide disassembly sequence generation and task allocation problems in a cyber-physical system (CPS) paradigm [22]. The pioneering work by Ji and Wang (2023) on selective disassembly sequence optimisation using CAD technologies underscores the evolving nature of CAD integration, offering insights into its potential to optimise disassembly paths while minimising costs and environmental impact [83].

Münker et al. (2023) delved into CAD-based product partitioning for automated disassembly, highlighting the remanufacturing benefits for components such as lithium-ion batteries [84]. This approach exemplifies how CAD integration facilitates the identification of optimal disassembly sequences, significantly reducing the time and resources required for disassembly tasks. Furthermore, a comparative study by Rehal and Sen (2023) on disassembly sequencing schemes emphasizes the superior planning accuracy and efficiency afforded by CAD technologies, marking a departure from traditional, less efficient methods [85].

2.3.2 Collision detection methods

The integration of CAD assembly collision detection methods into disassembly sequence planning (DSP) problems remains a nascent research area. Only one paper was published in 2022 by Prioli et al., who developed a CAD-based collision evaluation method to formulate the disassembly precedence matrix from the geometric collision data in an assembly file [33]. This research area has great potential to lead the way towards optimising disassembly processes. Detecting collisions in a CAD environment is an efficient way to check the interference conditions between parts without causing damage to the components or the disassembly apparatus. The study by Prioli et al. demonstrated the effectiveness

of CAD-based collision detection in DSP [33].. By incorporating collision detection algorithms into DSP, engineers can simulate and evaluate potential disassembly sequences in a virtual environment, identifying and eliminating sequences that would lead to collisions and, consequently, potential damage to parts or inefficiency in the disassembly process.

CAD models can be utilised to simulate the disassembly of assemblies by applying collision detection algorithms to automatically generate disassembly sequences that avoid component interference. This approach not only streamlines the planning process but also significantly reduces the reliance on expert manual analysis of EoL products or trial-and-error methods in the real world, thereby saving time and resources. Murali and Mahapatra (2023) explored selective disassembly sequence generation, emphasising the importance of collision detection algorithms within CAD systems [86]. These algorithms play a pivotal role in enhancing the reliability and safety of disassembly processes, ensuring that parts are disassembled without damage and in an efficient manner. Moreover, the adaptability of CAD software allows for the easy modification of product designs to further facilitate disassembly, promoting a design for disassembly (DfD) philosophy that considers the end-of-life phase of the product right from its design stage. Thus, the benefits of applying CAD assembly collision detection methods in DSPs not only extend beyond operational efficiency but also

contribute to the development of smarter, more adaptable manufacturing systems that can quickly respond to changes in product design or recycling requirements.

2.3.3 CAD simulations and information extraction

CAD simulation is an experimental computer-based approach that can be defined as a way of managing the models from changes, reproduces, or projections in a CAD environment, thereby clarifying the reasons for change [6]. CAD simulations and information extraction have emerged as vital components of comprehensive DSP systems. An AR-based disassembly guiding system automatically extracted the vertexes of CAD feature bounding boxes and then used them as input data in the optimal viewpoint selection algorithm for guiding a virtual scene [87]. Barbu et al. (2022) highlighted the development of a visual assembly planning system that leverages CAD simulations to predict and plan disassembly processes accurately [58]. This capability to simulate disassembly scenarios in advance significantly reduces the risk of errors and inefficiencies, paving the way for more sustainable and cost-effective manufacturing practices. The utilisation of data extracted from CAD models further streamlines the disassembly process. Information on geometric and material properties extracted from CAD models can be directly applied to DSPs, enabling a more nuanced and informed approach to disassembly. This integration not only saves time but also

ensures that disassembly is carried out in the most environmentally friendly manner possible, supporting the goals of sustainable manufacturing.

2.4 Metaheuristic optimisation algorithms for DSP

The development and application of metaheuristic optimisation algorithms have been pivotal in addressing the complexities inherent in DSP. Research efforts, as documented by authors such as Lambert, have focused on crafting sophisticated algorithms capable of navigating the vast solution spaces associated with DSP problems [65], [66]. These algorithms, ranging from genetic algorithms to simulated annealing and beyond, offer robust frameworks for identifying optimal or near-optimal disassembly sequences amid the constraints and variabilities of real-world applications.

2.4.1 Metaheuristic performance evaluations

Performance evaluations of these metaheuristic algorithms reveal their significant potential in enhancing the efficiency and effectiveness of DSP. For instance, Adenso-Díaz et al. provide insights into a two-phase approach that not only

optimises disassembly sequences but also integrates environmental considerations, reflecting an evolving emphasis on sustainability [88]. Furthermore, the work by Adenso-Díaz et al. illustrates the benefits of employing a path-relinking approach for bicriteria DSP problems, showcasing the algorithms' ability to balance multiple objectives effectively [89].

2.5 Summary

This chapter provides a literature analysis of the DSP problem using two approaches: quantitative and qualitative research. Two subtopics covered in this analysis are the applications of CAD models in DSP and the metaheuristic optimisation algorithms for solving DSP problems. An investigation of the descriptive analysis of the published literature from the first DSP paper until 2024 is presented. Three cross-discipline published under the keywords “DSP + CAD”, “DSP + metaheuristic”, and “DSP + CAD + metaheuristic” are classified by their distributions in years. It is found that there is no publication so far have worked within the topic belongs to “DSP”, “CAD”, and “metaheuristic” together, except this thesis. In the qualitative research, this review first detailly focused on the matrix-based model representation method and disassembly operators in DSP problems. Second, the role of CAD models playing in the DSP problems is reviewed. It is identified that a few studies attempt to solve the DSP problem

using CAD collision detection method. Third, the application of metaheuristic algorithms in the DSP problem and the algorithms' performance are reviewed. In summary, the blank research area for integrating the three key terms, "DSP," "CAD," and "metaheuristics," leads this thesis to build a CAD-based integrated disassembly system. The pioneering work of developing the CAD collision detection method to solve DSP problems better adds a novelty to this thesis.

Chapter 3 Disassembly data extraction for the CAD models

This chapter contains five sections. The first section describes preliminaries for the robotic disassembly sequence planning problem and the integration system designed for this thesis. The second section explains the disassembly information collection procedure for end-of-life products. The third section introduces the data extraction work on CAD models. The fourth section presents the case study results. The fifth section summarises this chapter.

3.1 Problem preliminaries and system design

The disassembly information of EoL products can be divided into physical structure information and real-world disassembly information. The physical structure information refers to the outline shape of this EoL product and the composition of each part. The real-world disassembly information refers to the disassembly target of this EoL product. Some disassembly information needs to be collected from the real world for system analysis. This chapter mainly introduces real-world disassembly information about EoL products. According to the target of the disassembly problem, first, the related real-world disassembly information is discussed. Then, using SolidWorks 3D software as the CAD environment, an add-in tool is developed for reading and extracting real-world disassembly information, which is stored in the EoL products' CAD model.

In the developed add-in tool architecture, the CAD model of EoL products plays a significant role, such as human memory during disassembly, which can store and retrieve relevant nonphysical structural information such as basic disassembly time, disassembly tools, and the serial number of components. The enormous amount of information involved in disassembly is proposed for modularisation. With modularised information, it is beneficial to search for, extract and update disassembly information quickly.

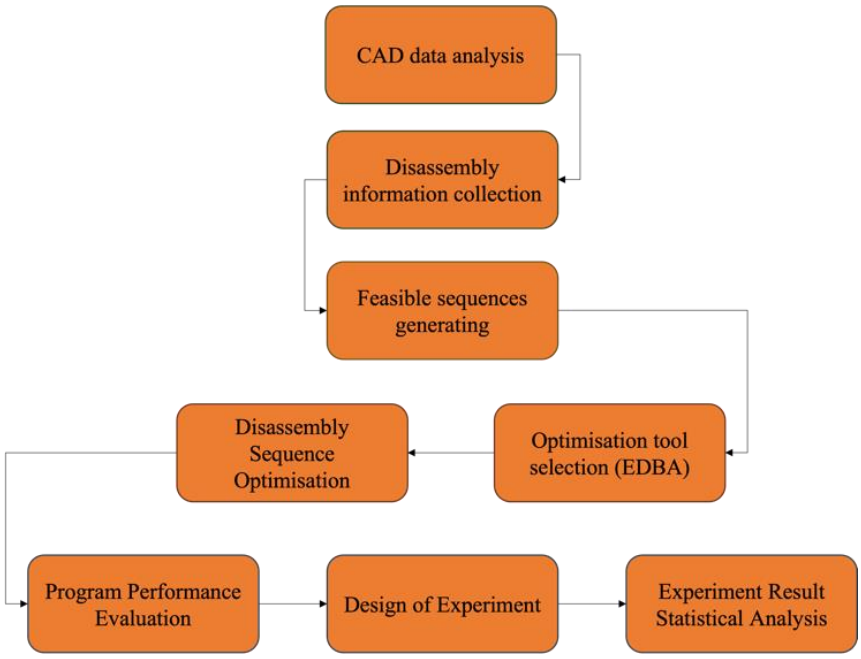


Figure 3.1 The workflow of this research

This study employs a robot as the sole disassembly operator and adopts a disassembly mode characterized by a complete, non-destructive, and sequential process. The disassembly model is built upon the CAD model of the EoL product. Furthermore, the planning method is chosen as a single disassembly cost objective to find the minimum value of the total disassembly time cost. The Bees Algorithm is selected to optimise the generated results based on the planning method. Because DSP is a discrete, combinatorial [90], and nondeterministic polynomial (NP) [79] problem, an enhanced discrete version of the Bees Algorithm [91] is applied in the optimisation process to increase the search efficiency.

This research is divided into five tiers, as depicted in Figure 3.1. The first step, CAD data analysis, initiates by reading a CAD model and identifying the geometry and dimensions of the components. A collision simulation is performed to extract the precedence relationships between the components (i.e., the precedence matrix and interference matrix) and the disassembly location of each component. The second step collects disassembly information from the real world, including basic disassembly time, disassembly tools, disassembly time penalties on direction changes, and tool changes of the robot end-effector. Third, a feasible disassembly sequence generator is built using the precedence relationship data. A group of feasible sequences with directions are generated using the extracted precedence relationship and collected disassembly information as input to the feasible disassembly sequence with the direction generation algorithm. In the fourth tier, the Bees Algorithm in its basic discrete version is introduced for its parameter constitution and the optimisation search scheme. In the Bees Algorithm, the search mechanism for the solution space can be divided into two processes—exploitation and exploration—the two terminologies of which can also be referred to as local search and global search [92]. Compared to the basic discrete version, the EDBA focuses on the local search strategy by adapting three methods—insertion, swapping, and mutation—and is therefore selected as the optimisation tool. Then, the outcome of tier three is used as the initialised population for disassembly sequence optimisation. The relevance between DSP, as a real-world problem, and EDBA, as an optimisation algorithm tool, is

explained and constructed as a program. Finally, in the fifth tier, the performance of the EDBA and its outcome behaviour are evaluated. The performance is tested under a design of experiment (DoE) method, which assesses the performance when modifying selected parameters of the BA by specific algorithm metrics, and the metrics are further statistically evaluated.

3.2 Disassembly information collection

The minimum total disassembly time of the EoL product [91] is set as the single optimisation object. The equation is shown in (1), which is formed by three factors: the disassembly basic time of the component (BT), the disassembly moving time of the robot end effector (MT) and the disassembly penalty time of the product (PT). In addition, two types of product disassembly penalty times are considered here, namely, the penalty tool change time (PT_{tool}) and the penalty direction change time ($PT_{direction}$). This information is collected from the real world as the input data for RDSP.

$$F(Time) = BT + MT + (PT_{tool} + PT_{direction}) \quad (1)$$

The disassembly basic time of a component is the time consumption of a robot with a tool disassembling a component from its EoL product. The time recording starts when the working tool on the robot touches the target component and then ends when this component is confirmed to be fully released from its original posture. Generally, this is also the time point at which the working tool on the robot is ready to leave for the next step. Equation (2) shows the calculation of the disassembly basic time, which accumulates the time consumption for each of the components. x_i refers to the component in the EoL product, and x is the number of components. These representations are also applied in the following equations.

$$BT = bt(x_1) + bt(x_2) + \cdots + bt(x_x) = \sum_{i=1}^{n} bt(x_i) \tag{2}$$

The disassembly moving time of the robot end-effector is presented by equation (3). When a robot has completed its assigned disassembly task and is ready to leave the current component, for example, ready to leave from the component x_1, a signal from the RDSP system will be sent to the robot, indicating that the robot will move to its next disassembly task x_2. The accumulated time spent by the robot moving from its current to the next task is considered the total disassembly moving time.

$$MT = mt(x_1, x_2) + mt(x_2, x_3) + \cdots + mt(x_{i-1}, x_i) = \sum_{i=2}^{n} mt(x_{i-1}, x_i) \quad (3)$$

The disassembly penalty time of a product is used to address the changes occurring during the disassembly process. This study includes two types of penalty time: tool changing of the robot and direction changing of the combined actions by the product and robot. These changes occur between disassembly tasks, for which the time assumption is shown in equation (4). The two types are to be explained individually.

$$PT = pt(x_1, x_2) + pt(x_2, x_3) + \cdots + pt(x_{i-1}, x_i) = \sum_{i=2}^{n} pt(x_{i-1}, x_i) \quad (4)$$

Due to the different component sizes and types, disassembly tool selection is a problem in RDSP. As in a manual disassembly environment, multiple types and sizes of tools are provided to human operators for adapting different task requirements, such as hammers, pliers, grippers (small, medium, and large sizes for tiny, regular, and enormous objects), spanners (M1, M2, M3… for matching sizes of bolts), and screwdrivers (Phillips Head, Slot Head, Hex head, etc.). Figure 3.2 shows the matching relationships between the components and tools. In robotic disassembly, tool design and replacement for robot end-effectors have attracted attention and have been studied for years [93], [94]. As disassembly task

manipulators, robots sometimes require multiple disassembly tools and grippers for disassembling one EoL product. Tool changes are unavoidable, and time is needed for these changes between components. Therefore, equation (4) is applied here, and the result is accumulated in equation (1).

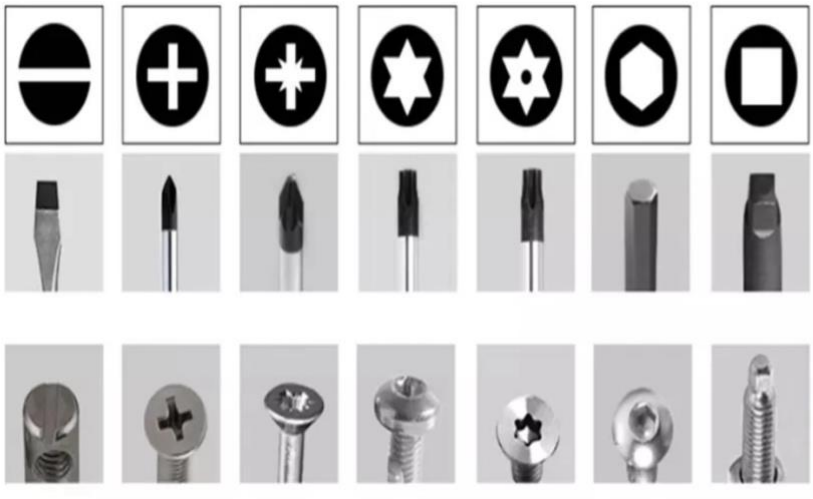


Figure 3.2 Disassembly tools for adapting different task requirements.

The disassembly direction refers to all the degrees of freedom a component can be completely detached from its EoL product. However, the disassembly direction change refers to the change in the relative positional relationship between the robot and the EoL product. In general, EoL products are one-sided fixed, while robots approach the components under disassembly direction instructions; see Figure 3.3. Currently, there is no existing RDSP system that considers moving the EoL product fixture and robot end-effector together,

referencing simultaneous machining in manufacturing. Therefore, the disassembly direction change is currently the robot end-effector direction change.

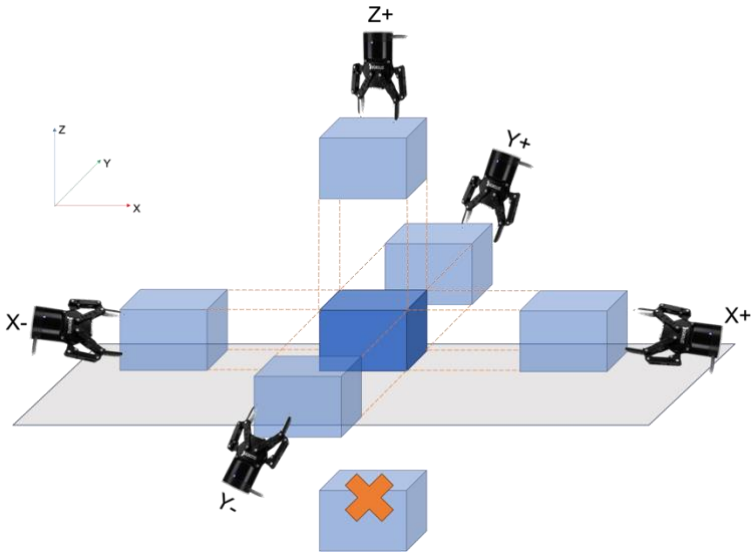


Figure 3.3 One-sided fixed EoL product with the robot end-effector approaching.

Although the proposal of considering multiple disassembly directions in DSP can be found in many works, research in four directions, $Direction(4) = \{X+, X-, Y+, Y-\}$ for 2D models and six directions, $Direction(6) = \{X+, X-, Y+, Y-, Z+, Z-\}$ for 3D models, is still favoured by the majority of researchers [33], [91], [95], [96], [97], [98], [99]. The research object and disassembly direction in this work is a 3D model with six disassembly directions, adapting the method of total penalty for direction change [100]. Equation (5) shows the penalty function matrix $M(PF)$. If no direction change is required between two disassembly components, for example, from $X+$ to $X+$, the

direction change penalty value is 0; if a change in 90° is needed, for example, from $X+$ to $Y+$ the direction change penalty value is 1; if a change in 180° is needed, for example, from $X+$ to $X-$, the direction change penalty value is 2. These values are extracted from the matrix to be used in equation (4), and then the results are accumulated in equation (1).

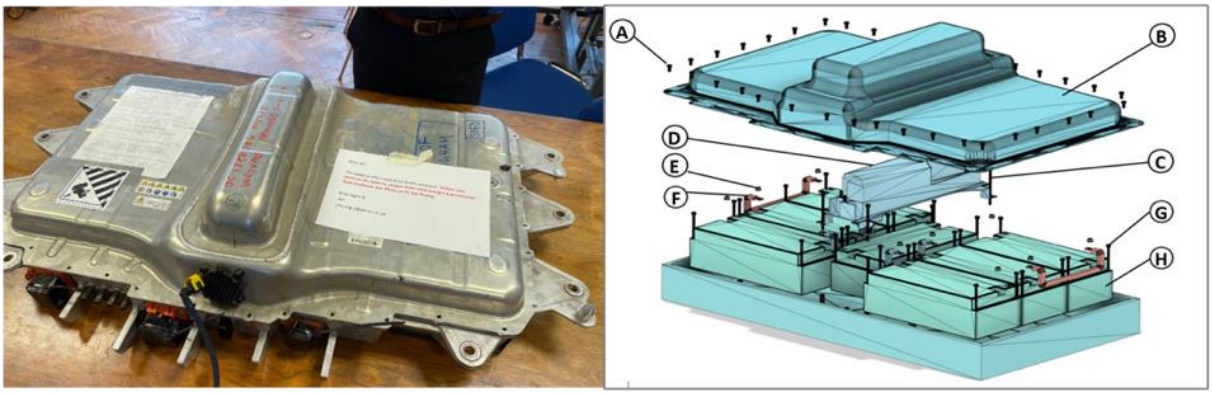
$$M(PF) = \begin{array}{c} \\ X+ \\ X- \\ Y+ \\ Y- \\ Z+ \\ Z- \end{array} \begin{array}{c} \begin{array}{cccccc} X+ & X- & Y+ & Y- & Z+ & Z- \end{array} \\ \begin{bmatrix} 0 & 2 & 1 & 1 & 1 & 1 \\ 2 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 2 & 1 & 1 \\ 1 & 1 & 2 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 2 \\ 1 & 1 & 1 & 1 & 2 & 0 \end{bmatrix} \end{array} \tag{5}$$

3.3 CAD model data extraction

An add-in tool based on the CAD environment is designed and developed for collecting disassembly information from the CAD model of an EoL product. The SolidWorks, a software developed by the Dassault Systems company, is used to present the add-in tool. The SolidWorks is renowned for its powerful capabilities in creating and modelling 3D designs with a primarily usage in engineering, architecture, and industrial design fields. SolidWorks can allow users to create precise and detailed 3D models of parts, assemblies, and drawings, and to facilitate the design and development process of a product. SolidWorks offers a

user-friendly and easy-to-access interface to both beginners and experienced professionals. Some key features of SolidWorks' 3D CAD function include sketching, parametric modelling, part modelling, assembly modelling, simulation, and integration with other software.

A CAD model of an Electric Vehicle battery (EVB) is used as a case study to present the extracted disassembly information and CAD data result. Figure 3.4 presents a lab view and a CAD model of the EVB. The components detail of the EVB is listed in the Table 3.1.



(a) (b)

Figure 3.4 The EV battery: (a) lab photo; (b) CAD model.

Table 3.1 A component list of the EVB.

No.	Component	Quantity	Operation
A	M5 x 10mm Top Screw	35	Unscrew
B	Top Cover	1	Pick-up
C	M6 x 16mm Central Bolt	4	Unscrew

D	Central Junction Box	1	Pick-up
E	Nut	12	Unscrew
F	Busbar	4	Pick-up
G	M6 x 95mm Battery Bolt	28	Unscrew
H	Battery Module	7	Pick-up

The developed add-in tool is programmed in C# language. The whole add-in tool is developed by three sectors, the cover page (Figure 3.5), the assembly page (Figure 3.8), and the part page (Figure 3.10 and Figure A.1). Due to their characteristics in types of the part, components (Figure 3.10) and fasteners (Figure A.1) are separately detailed in the part pages. An overall view is given to each of these pages.

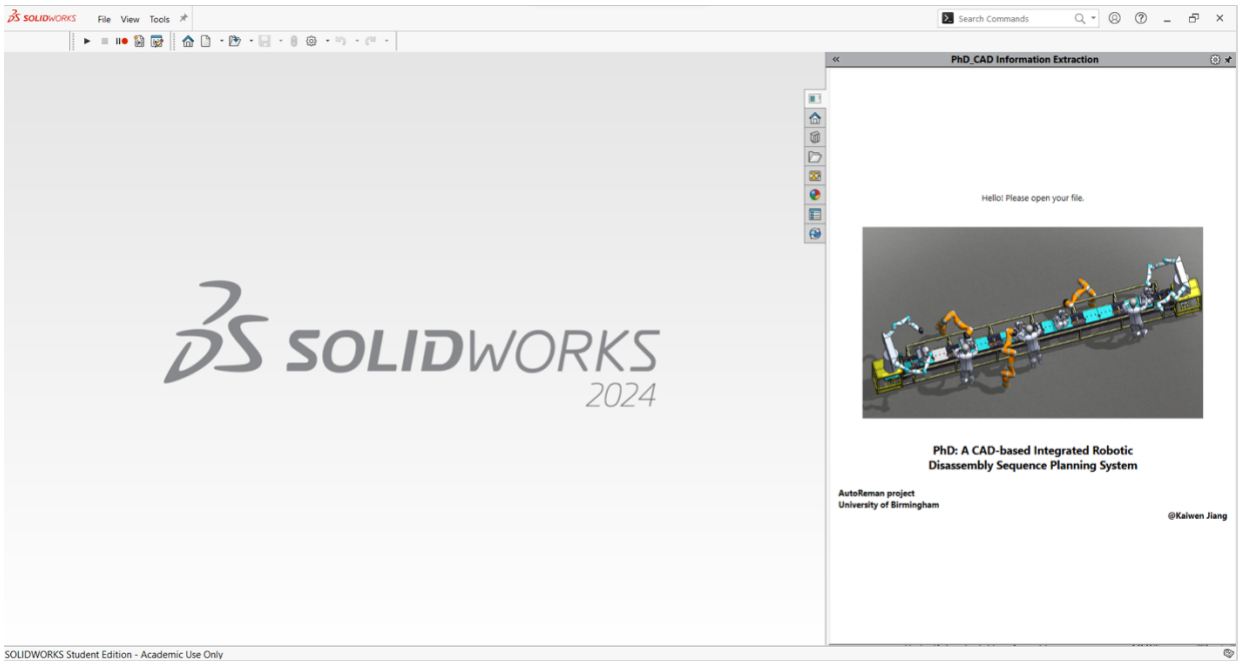


Figure 3.5 An overall view of the developed SolidWorks add-in tool.

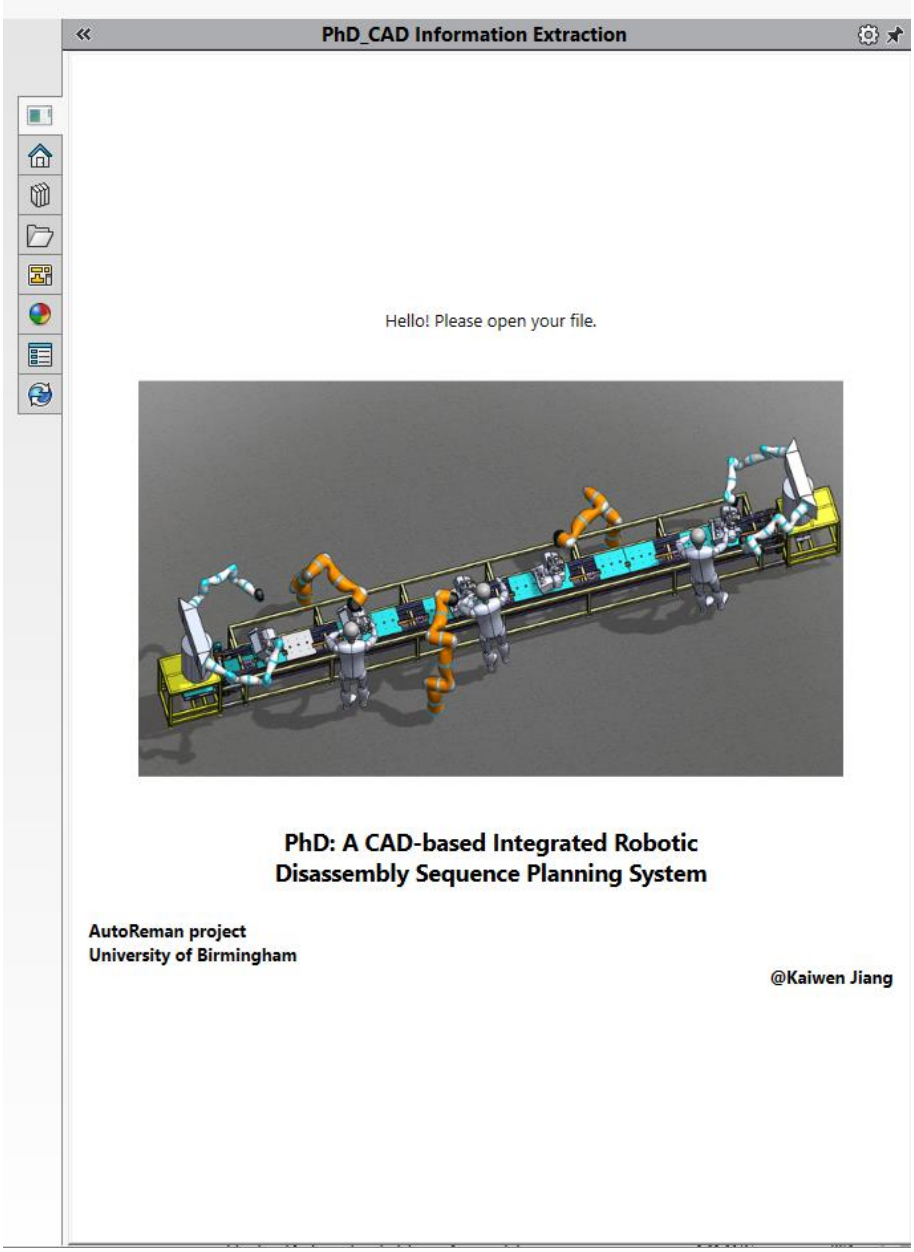


Figure 3.6 A zoom-in view of the add-in tool's cover page.

As long as the software's CAD environment is initialised, the add-in tool monitors the active file opened in SolidWorks. A cover page of the add-in tool is presented when neither an assembly file nor a part file is opening. Figure 3.7 shows a page-changing scheme of the add-in tool.

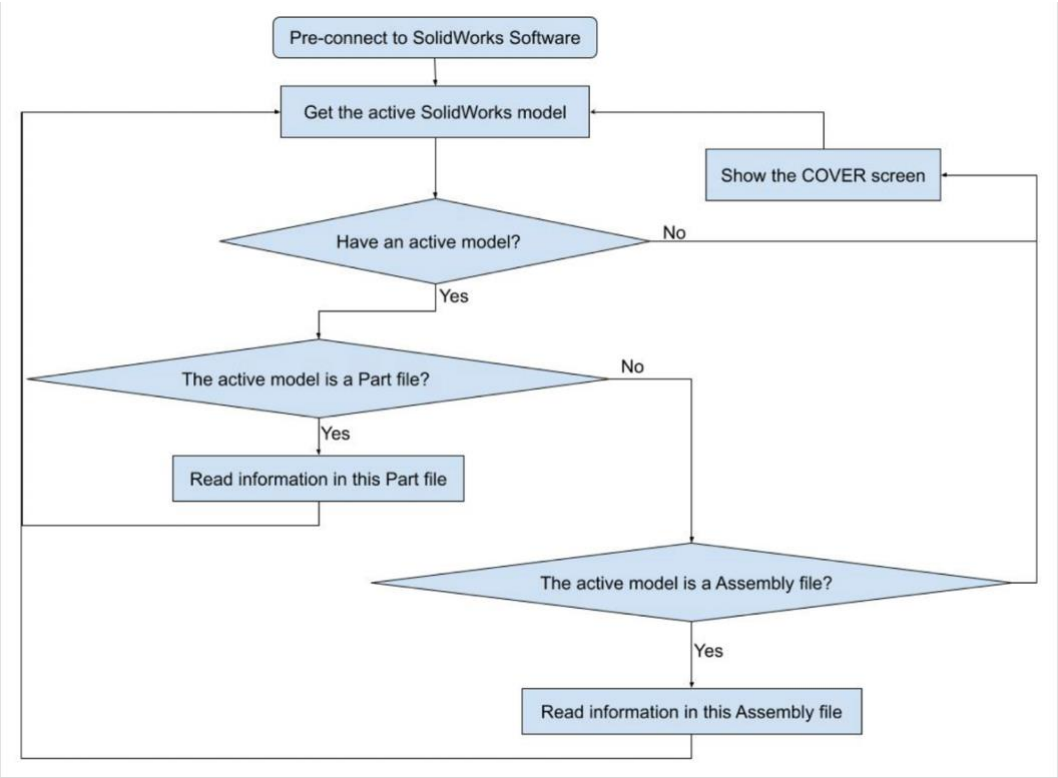


Figure 3.7 The page-changing scheme of the developed add-in tool.

The add-in tool assembly page has five sections: (1) product information, (2) robot information, (3) optimisation algorithm information, (4) disassembly sequence output, and (5) additional information. The product information stores the EoL product's name and its model number. The robot information outlets choices of the disassembly robot operators, with a corresponding robot moving speed. The direction-changing penalty time and the tool-changing penalty time are listed below the robot speed. The "view" button is used to refresh the robot's information. The optimisation algorithm information stores the selected metaheuristic optimisation methods. The algorithm chosen in the presented figure is the enhanced discrete bees algorithm. Other metaheuristic algorithms, such as

the tabu search, the genetic algorithm, etc., are also selectable. The “Optimise” button is used to refresh the optimisation algorithm’s information in the third section. The disassembly sequence output presents the output of the generated optimal disassembly sequence result. This work aims to achieve a high disassembly efficiency in terms of time cost, so the total disassembly time is presented in the fourth section. Further details of the optimal result generated according to the selected optimisation method are presented in the following textbox. The additional information is for taking the extra notes when the CAD designer needs.

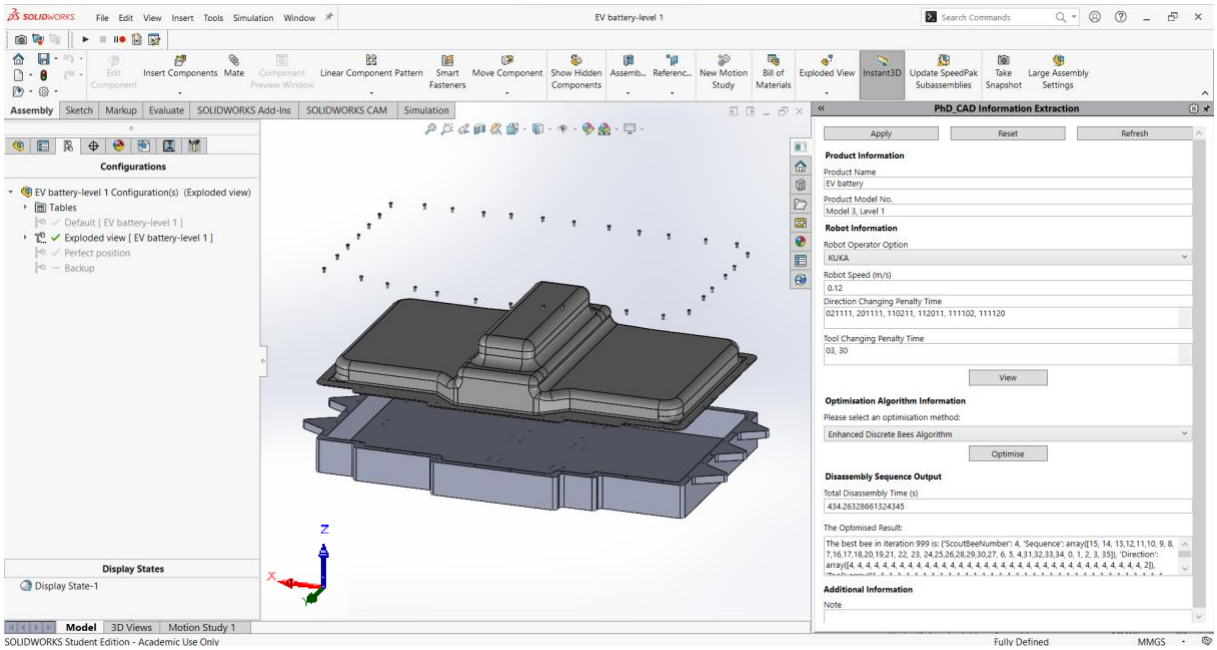


Figure 3.8 An overall view of the add-in tool’s assembly page.

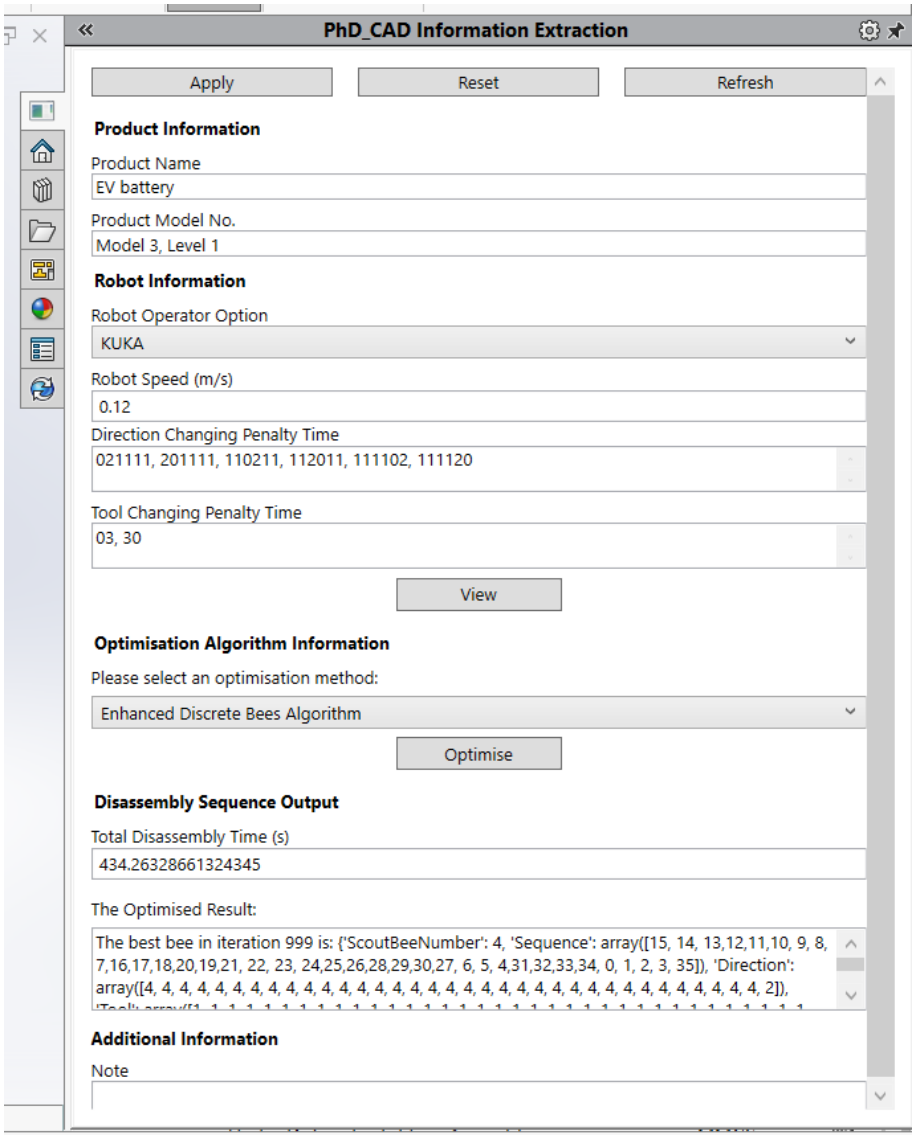


Figure 3.9 A zoom-in view of the add-in tool's assembly page.

The add-in tool part page has three sections: (1) part information, (2) disassembly information, and (3) additional information. The part information stores the name of the product that the part belonged to, the part's name and the part's number. The type of the part is another information stored. Components and fasteners are classified here by a checkbox.

Figure 3.10 and Figure 3.11 are the overall view and the zoom-in view page of a component. Figure A.1 and Figure A.2 are the overall view and the zoom-in view page of a fastener in the Appendix. The disassembly information stores the selected disassembly tool and the basic disassembly time of the part using the selected tool. The dimension of the part can be checked using the dimension function. The value of the selected dimension is shown in the CAD environment and in the add-in tool, too. The additional information is for taking the extra notes when the CAD designer needs.

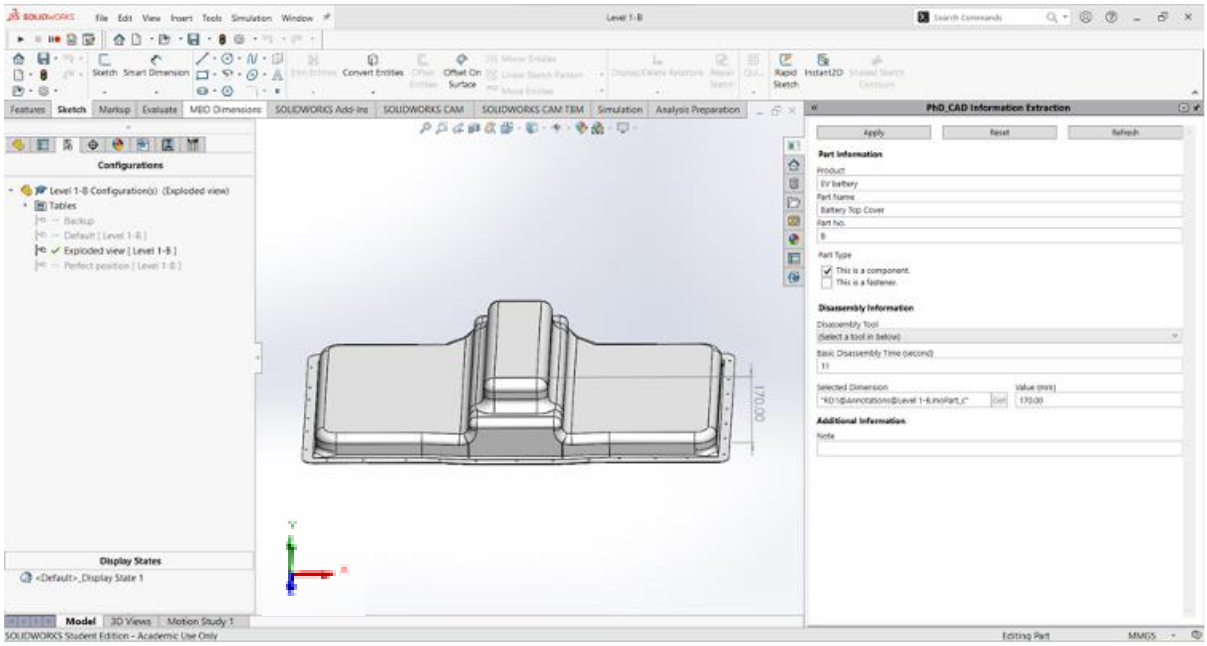


Figure 3.10 An overall view of the add-in tool's part page: a component.

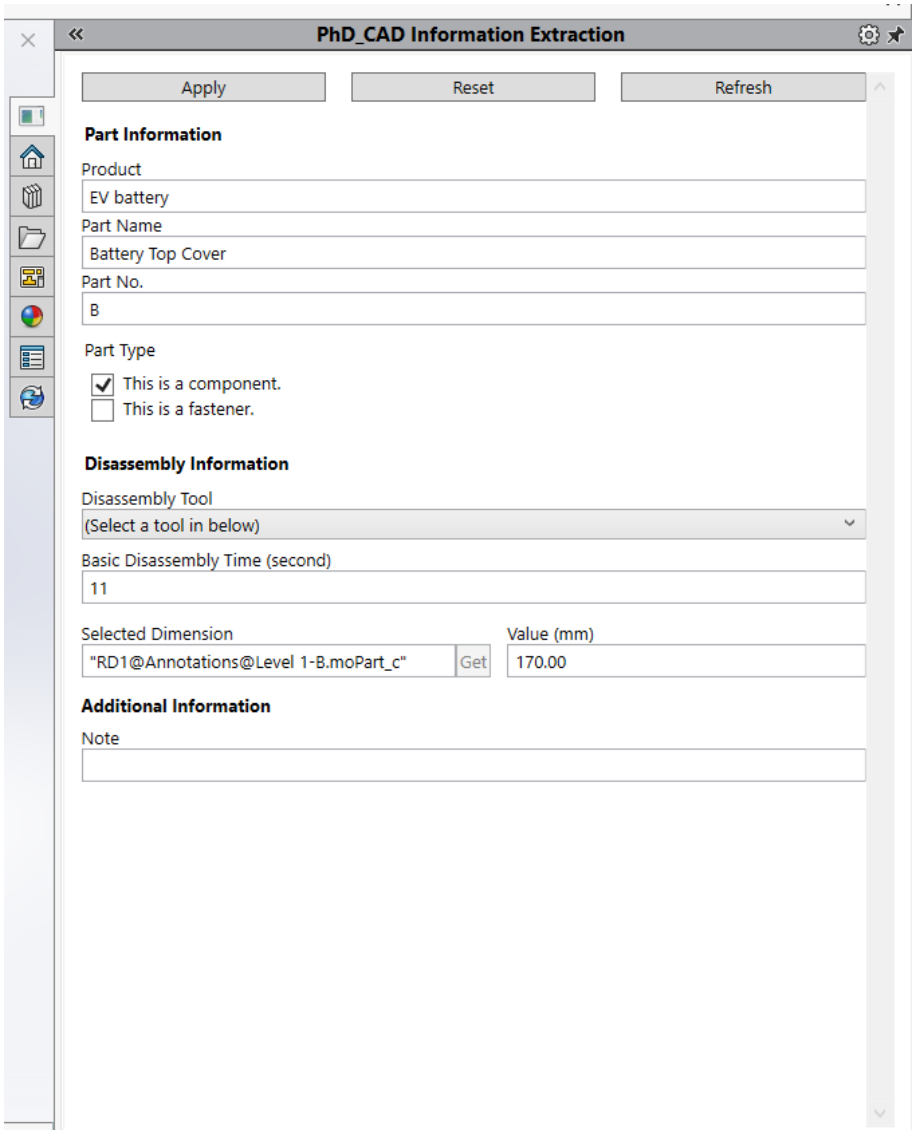


Figure 3.11 A zoom-in view of the add-in tool's part page: a component.

In both the assembly and part pages of the add-in tool, there are three buttons on the top of the page: "Apply", "Reset", and "Refresh". The "Apply" button lets the CAD designer store the input information into the CAD file. The "Reset" button clears all the stored information from the CAD file. The "Refresh" button uses the stored information to cover the information presented on the current page.

3.4 Summary

This chapter analyses the real-world factors of disassembly information using the total disassembly time as the disassembly target. In the CAD environment built by SolidWorks, an add-in tool is developed to extract, find, and present the nonphysical disassembly information that is stored in the CAD models of EoL products. A workflow of the robotic automated disassembly sequence planning process to describe the EoL product disassembly problem is provided in the beginning. Second, a fitness function of the total disassembly time is given with the details describing the factors of each composition in the formula. Moreover, the required disassembly tools and the degree of freedom of the disassembly direction for the EoL product are also analysed. Finally, an add-in tool is developed to display the above nonphysical disassembly information. Three interfaces are included in the add-in tool: the homepage, the page for the EoL product (CAD assemblies), and the page for the EoL components (CAD parts). A flowchart of the page switching rule is also given. The situation in which multiple files are opened at the same time is also considered in the solution. CAD models include EoL product information. In CAD data, the composition of parts in the assembly (such as the optional disassembly direction of parts) and the space relationship between parts (such as interference and interference in the direction) can be applied to the feasibility analysis of disassembly. These issues will be

discussed in the next chapter. From the perspective of researchers and disassembly operators, the modular disassembly information can clarify the real situation of the product in the system where the robot automated disassembly sequence is planned. The use of the CAD model that can be updated will provide sufficient information for the feasibility of the EOL products and optimise the planning of the disassembly sequence.

Chapter 4 RDSP interference matrix generation from CAD collision detection

This chapter contains five sections. The first section introduces the process of generating bounding boxes in a CAD model. The second section explains the mechanism of CAD parts' movements, accompanied by the developed collision detection algorithm. The third section presents the consequences of detecting collisions between parts in the CAD environment and thus generating the corresponding space interference matrices. The fourth section shows the results. The fifth section summarises this chapter.

4.1 Bounding box formation on the CAD model

The CAD models of EoL products store massive amounts of data and information that can help to plan disassembly sequences. The CAD data and information can be divided into geometrical and mating information. EoL product details, for example, the volumes of EoL products and their assembly structures, can be expressed by analysing those data and information. Manually extracting and then processing the EoL products' information was used in past works. Some studies on the disassembly sequence planning problem employ the manual approach to EoL product models but seldom mention how repetitive and error prone this manual work can be, nor does it consume to deal with the enormous processing time. To address these disadvantages, this chapter proposes a method to automatically extract the data and information from CAD models and then studies their space interference relationships using a developed collision detection algorithm designed to determine the space interference relationships between CAD parts.

Figure 4.1 shows a workflow of this chapter. CAD model identification is the first step in the process of the CAD collision detection method. First, a CAD model of the EoL product is loaded. The program then identifies names and quantities of the parts contained in the CAD model before generating bounding boxes. Second,

for each of the part, the API program using the extracted dimension from the bounding box as the maximum moving distance of the part, dragging the CAD part to disassemble from its assembly model. The API program monitors to detect collisions during the movement and to record the interference result. After all directions of one CAD part and all parts in the assembly are checked, the space interference matrix can be output from the API program. A simple case study, with detailed information provided in the following sections, is shown to present the whole process.

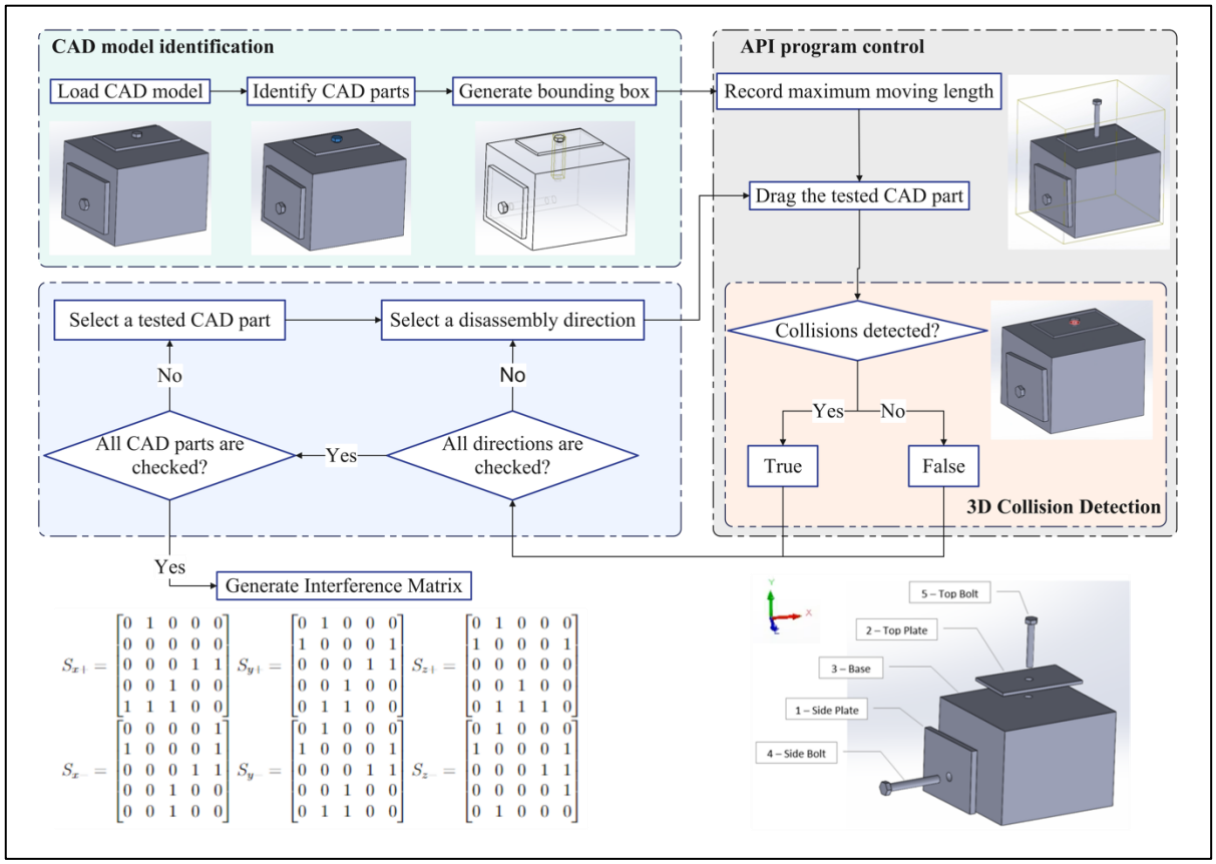


Figure 4.1 The Flowchart of the CAD collision detection method to generate the space interference matrix.

A bounding box is a virtual, cuboid-shaped structure used to surround the target CAD object. A bounding box can be applied to a CAD part or a CAD assembly. Figure 4.2 (a) presents a sphere showing the procedure of forming a bounding box for the CAD object. First, eight vertices of the bounding box are located according to the maximum dimension of the CAD object, as shown in Figure 4.2 (b). Then, in Figure 4.2 (c), the edges of the bounding box are lined up to shape the structure.

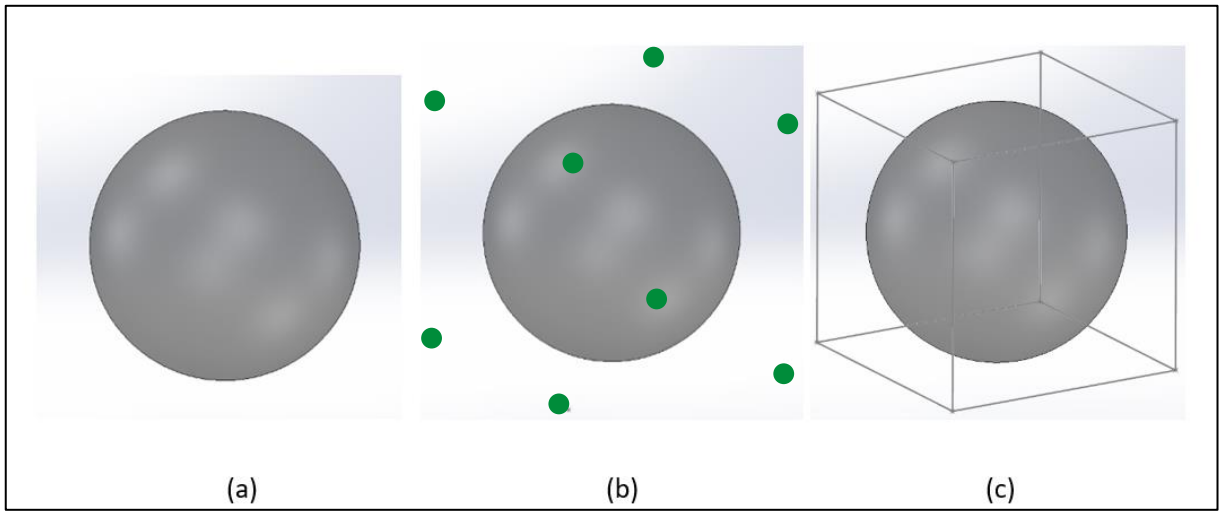


Figure 4.2 A bounding box model: (a) A sphere. (b) Locating bounding box vertices. (c) Using edge lines to connect bounding box vertices.

A schematic diagram of the bounding box is shown in Figure 4.3 to explain the method of obtaining the CAD object's dimensions automated by an API program. A bounding box has eight vertices ($Point$ 0 to $Point$ 7) and twelve edge lines ($Line$ 0 to $Line$ 11). A virtual Cartesian coordinate system can be built based on

these properties. The selection of the coordinate origin of the bounding box within the vertices is random. Aside from the selected point, two perpendicular lines are further taken as the three dimensions. For example, $Plane_{XYZmin}$ is named for the plane constructed by the X-axis and Y-axis at the minimum end of the Z-axis. It is built by $Line\ 0$, $Line\ 4$, $Line\ 6$, $Line\ 8$, $Point\ 0$, $Point\ 2$, $Point\ 4$, and $Point\ 6$. The rest of the planes are listed in Table 4.1.

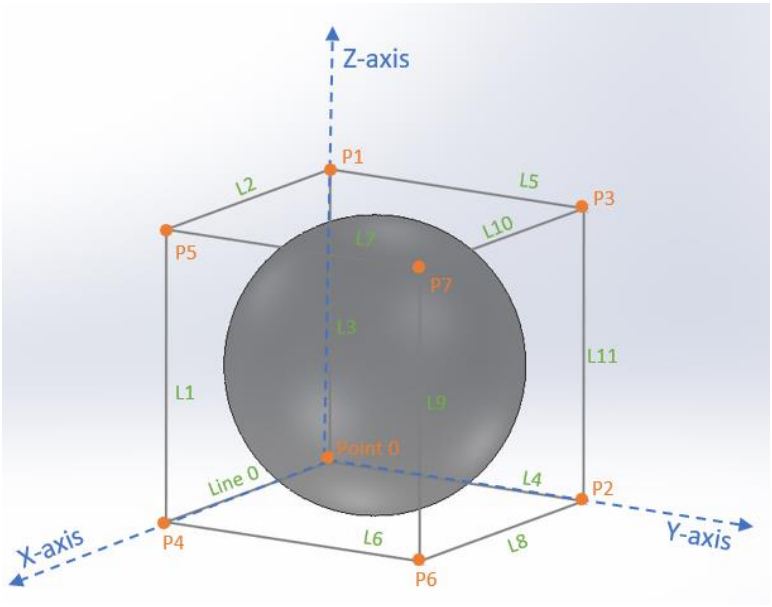


Figure 4.3 A schematic diagram of the bounding box

Table 4.1 The plane construction of a bounding box

Plane	Edge (Line)	Vertex (Point)
$\boldsymbol{Plane_{XYZmin}}$	L0, L4, L6, L8	P0, P2, P4, P6
$\boldsymbol{Plane_{XYZmax}}$	L2, L5, L7, L10	$P1, P3, P5, P7$

$\boldsymbol{Plane_{XZYmin}}$	L0, L1, L2, L3	$P0, P1, P4, P5$
$\boldsymbol{Plane_{XZYmax}}$	L8, L9, L10, L11	$P2, P3, P6, P7$
$\boldsymbol{Plane_{YZXmin}}$	L3, L4, L5, L11	$P0, P1, P2, P3$
$\boldsymbol{Plane_{YZXmax}}$	L1, L6, L7, L9	$P4, P5, P6, P7$

Table 4.2 The coordinate values of the bounding box vertices

Vertex	**Coordinate Value**			**Vertex**	**Coordinate Value**		
	X-axis	Y-axis	Z-axis		X-axis	Y-axis	Z-axis
Point 0	x_{min}	y_{min}	z_{min}	$Point\ 4$	x_{max}	y_{min}	z_{min}
Point 1	x_{min}	y_{min}	z_{max}	$Point\ 5$	x_{max}	y_{min}	z_{max}
Point 2	x_{min}	y_{max}	z_{min}	$Point\ 6$	x_{max}	y_{max}	z_{min}
Point 3	x_{min}	y_{max}	z_{max}	$Point\ 7$	x_{max}	y_{max}	z_{max}

Table 4.3 The dimensions of a bounding box

Bounding Box	**Edge Lines**	**Dimension Value**
$\boldsymbol{B_X}$	$= Line\ 0 = Line\ 2 = Line\ 8 = Line\ 10$	$\lvert x_{max} - x_{min} \rvert$
$\boldsymbol{B_Y}$	$= Line\ 4 = Line\ 5 = Line\ 6 = Line\ 7$	$\lvert y_{max} - y_{min} \rvert$
$\boldsymbol{B_Z}$	$= Line\ 1 = Line\ 3 = Line\ 9 = Line\ 11$	$\lvert z_{max} - z_{min} \rvert$

Three bounding box dimensions of the CAD model can be calculated. In a CAD environment, global coordinates are always available for all spatial points. Therefore, extracting the coordinate values for every eight vertices from the CAD environment is possible. The length of the edges, as the dimension of the CAD model, can be calculated by performing corresponding subtractions. For example, the X-axis length of the sphere model is the length of $Line\ 0$, using $Point\ 4$ minus $Point\ 0$.

4.2 CAD part movement with collision detection

The CAD object can be moved by API program control through mathematical matrices. The position and orientation of the CAD object are defined in the CAD environment using a transformation matrix T; see equation (6). T is a 2-by-2 matrix with four elements: a rotation submatrix $R_{(\theta)}$, a translation vector p, a scaling factor s, and an empty value placeholder.

$$T = \begin{bmatrix} R_{(\theta)} & 0 \\ p & s \end{bmatrix} \tag{6}$$

The submatrix $R_{(\theta)}$ in equation (7) guides the motion of rotation in Euclidean space. θ is the rotational angle. Details are given for three axes with three rotating matrices. It is a 3-by-3 orthogonal matrix that produces a pure rotating motion.

The vector p controls the motion of the translation of the CAD object. The moving distances of the X-, Y- and Z-axes are determined by the elements x, y, and z, respectively, as shown in equation (8). The scaling factor s controls the scaling up and down ratio between the input movement values and the actual movement displacements. The value of scaling factor is recommended to be set as 1 for presenting an equal value in the CAD environment from the program input.

$$R_{(\theta)} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$= \begin{cases} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\ \theta & -sin\ \theta \\ 0 & sin\ \theta & cos\ \theta \end{bmatrix}, & rotating\ about\ X-axis \\ \begin{bmatrix} cos\ \theta & 0 & sin\ \theta \\ 0 & 1 & 0 \\ -sin\ \theta & 0 & cos\ \theta \end{bmatrix}, & rotating\ about\ Y-axis \\ \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ sin\ \theta & cos\ \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, & rotating\ about\ Z-axis \end{cases} \quad (7)$$

$$p = [x \quad y \quad z] \quad (8)$$

Therefore, equation (9) shows three translation matrices of directions ($X+, Y+$, and $Z+$) and their reflection direction translation matrices ($X-, Y-$, and $Z-$). The rotational angle θ is 0 degrees. The scaling time factor is 1. The CAD object should be set in a floating status when the API program moves the CAD object.

Otherwise, the CAD object will remain at its position and report a false result to the API program.

$$T_{X+}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&0&1\end{bmatrix}\quad T_{Y+}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix}\quad T_{Z+}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&1&1\end{bmatrix}$$
$$T_{X-}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\-1&0&0&1\end{bmatrix}\quad T_{Y-}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&-1&0&1\end{bmatrix}\quad T_{Z-}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&-1&1\end{bmatrix} \tag{9}$$

4.3 CAD simulation and space interference matrix generation

Product representation is vital in disassembly sequence planning problems. Several methods have been proposed to establish a disassembly model for representing the EoL product, such as graph-based, matrix-based, ontology-based, and CAD-based methods. The geometrical structure of an EoL product and contact information between components can be obtained by studying the interference of the components to find a feasible disassembly solution for the product. Jin et al. used space interference matrices along six directions ($X+, X-, Y+, Y-, Z+$, and $Z-$) to describe the disassembly precedence relationships between different parts [101]. Furthermore, Liu et al. proposed a modified version of space interference matrices by considering the geometric

shape of fasteners [91]. Specifically, a bolt cannot be disassembled in either axial direction but can be disassembled in a single disassembly direction because of the obstacle of its bolt head.

Collision detection is a computational problem of detecting the intersection of two or more objects. This technology is widely used in many applications, such as computer graphics, robotic vision, path planning and computer simulations. 3D collision detection is a built-in function provided by most CAD environments. This function can aid in generating accurate space interference matrices while considering the actual shape of the components. Using the CAD 3D collision detection method is a beneficial approach for replacing the time-consuming, tedious, and repetitive space interference matrices generated in the manual.

A simple case study for explaining the CAD collision detection simulation thus the space interference matrices generation method is presented in Figure 4.4. Figure 4.4 (a) is the full CAD view of the case study with a coordination system attached. Figure 4.4 (b) shows the bounding box results. Figure 4.4 (c) is the exploded view of the case study with the parts labelled.

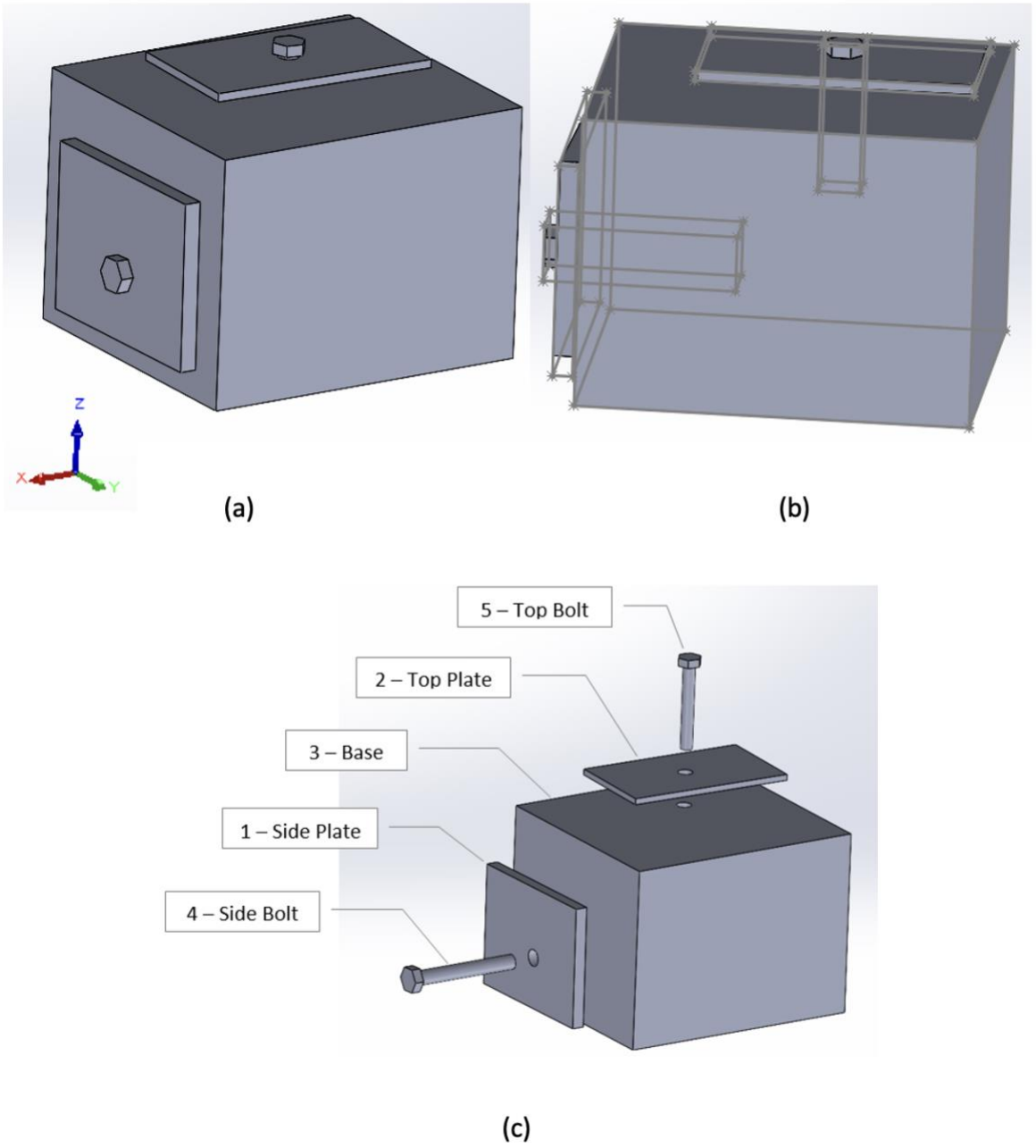


Figure 4.4 A simple case study: (a) CAD model; (b) bounding box; (c) exploded view

Two possibilities when testing for collisions are shown in the Figure 4.5: a) successful disassembly and b) collision detection. First, a component and its disassembly direction are selected to test its collision situation. To disassemble a

component successfully from its previous position, its moving distance must be at least equal to the maximum length of its selected disassembly direction. If a component can be moved clearly, the collision result is false, meaning no collision occurs during the movement; if a component has other components such as obstacles, the collision result is true. Figure 4.5 (a) is an example showing that a bolt is successfully disassembled from the CAD assembly in the $X+$ direction of the assembly. The maximum disassembly length of the bolt is $97.50 \pm 0.50\ mm$. The starting position of the bolt head to the side surface of the base part is $23.36 \pm 0.50\ mm$. After translational movement of the bolt away from its base, the distance between the bolt head and the side surface is $121.36 \pm 0.50\ mm$. The interpolation of the start and end points of this bolt is $98\ mm$, which is at least equal to the length of the bolt. This result means that the bolt can be disassembled from its CAD model without any interference. In Figure 4.5 (b), the red highlighted area indicates that a collision occurs if the bolt is disassembled from the X-direction.

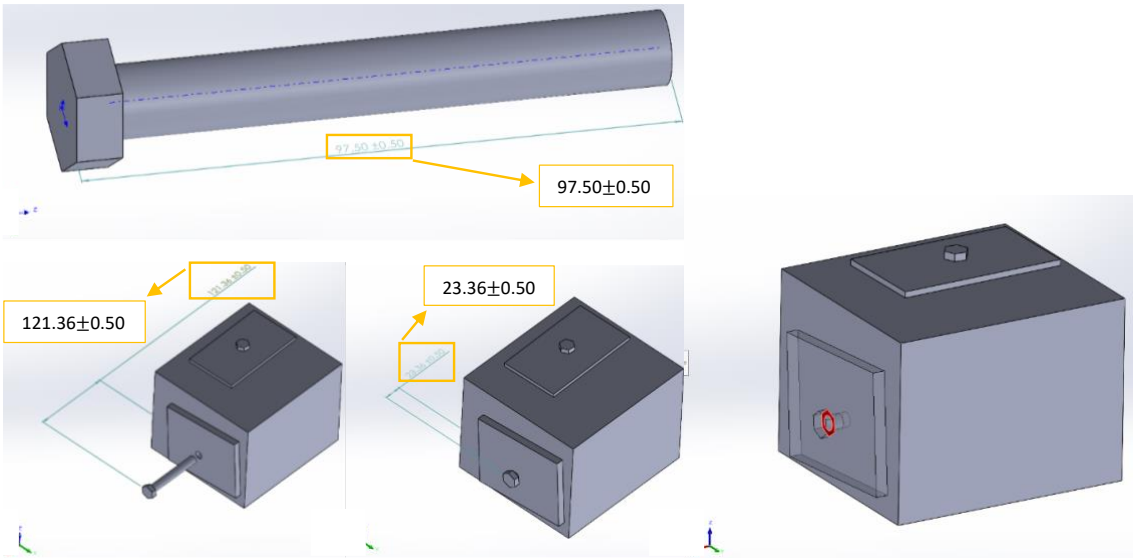


Figure 4.5 Collision detection for a CAD bolt (dimensional unit: mm). (a) A successful disassembly; (b) detecting a collision.

By program-checking all the components contained in the CAD assembly one by one, a complete space interference result of this CAD model can be generated. Algorithm 1 presents a collision-detecting pseudocode to generate the space interference result. The program input includes a CAD assembly model M, dimensions generated from bounding box B, and disassembly directions D. The output of this algorithm includes the fixed status of CAD parts F and space interference results of the assembly model.

The space interference checking program is inspired by mimicking human attempts at disassembly operations. To check the freedom of a component, the human operator would focus on one component and then try to move it from its

assembly in several disassembly directions until one or several directions succeed. Like the program in this work, a CAD part would be picked randomly, and then the program would control it to move away from its CAD assembly model along the disassembly directions. If a collision occurs during the movement, this CAD part cannot be disassembled from the tested direction. The CAD part is restored to its original position and tested in the next disassembly direction. Until all the disassembly directions are tested, the program will stop controlling moving this CAD part and record its interference conditions in the output space interference results in S. Assuming the tested assembly model M has n CAD parts, and the program will test each of the CAD parts from 1 to n. F is the fixed status of the CAD parts. If the program controlling the CAD part is preset as a fixed part, for example, as a base part, the status of its fixture is higher than the moving control instruction in the program; therefore, this CAD part will not be able to move.

The logic of this program is presented in Algorithm 4.1. There are three parameters used as counters: the transform iteration (I_t), the recovery iteration (R_t) and the interference counter (c). The transform iteration is used to record the moving steps of a CAD part to be disassembled along a selected direction. The program checks the interference to the CAD assembly model after each step of a CAD part movement. The maximum moving steps of a CAD part are equal to the bounding box dimension. If, in the designated disassembly direction, no collision occurs during the translational motion of the CAD part, the value of the transform

iteration of this CAD part would be the same as its bounding box dimension. The minimum movement of a CAD part is set as a unit length, depending on the preset value in the program. There is another circumstance while moving the CAD part: interference occurs between the CAD part in the moving and other CAD parts in the assembly model. Because the program checks the interference of the whole assembly model immediately after each step of the CAD part movement, the CAD part stops moving immediately. The achieved moving step of this CAD part is recorded in (It).

The recovery iteration records the movement step number of dragging the CAD part back to its original place. This step prepares the CAD assembly model for detection to the following CAD parts until all the parts have finished to check. Similarly, there are two possibilities for CAD parts during moving: one is that the under-checking CAD part is disassembled completely, and no collision occurs; the other is that the CAD part interferes with one or more other CAD parts, and a highlighted collision area appears. The interference counter c is used to determine whether an interference circumstance exists in the CAD model. If the first condition applies to the CAD model ($c = 1$), the value of Rt would be equal to (I_t) to restore the CAD part back. Otherwise, ($c = 0$), the value of R_t is the same as the bounding box dimension of this CAD part.

Algorithm 1 Detect Collisions While Transforming CAD Parts

Input: CAD Assembly Model($\mathcal{M}$), Bounding Box Dimensions($\mathcal{B}$), Disassembly Directions($\mathcal{D}$).
Output: CAD Parts Fixed Status($\mathcal{F}$), Space Interference result($\mathcal{S}$).

Initialisation: Transform Iteration($\mathcal{I}_t$), Recovery Iteration($\mathcal{I}_r$), Interference Counter($\mathcal{C}$);
Check and clear interference in $\mathcal{M}$;
for each part $\mathcal{P}_i \subset \mathcal{M}$, $i \in [1, n]$; **do**
  if $\mathcal{P}_i$ is a fixed CAD part **then** ▷ ($\mathcal{F} = 1$)
    $\mathcal{F}_i = 1$;
    return;
  else ▷ ($\mathcal{F} = 0$)
    for each disassembly direction $\mathcal{D}_j$, $j \in [1, m]$; **do** ▷ Start dragging component
      for $\mathcal{I}_t \in [0, \mathcal{B}_i j]$ **do**
        $\mathcal{P}_i \rightarrow$ transform one unit of step;
        $\mathcal{C} \leftarrow$ check interference in $\mathcal{M}$;
        if $\mathcal{C} > 0$ **then**
          return $\mathcal{I}_t$;
          break; ▷ Collision happens ($0 < \mathcal{I}_t < \mathcal{B}_i j$)
        else
          return $\mathcal{I}_t$; ▷ No collision, keep transforming until ($\mathcal{I}_t = \mathcal{B}_i j$)
        end if
      end for
      if $\mathcal{I}_t \neq \mathcal{B}_i j$ **then**
        $\mathcal{I}_r = \mathcal{I}_t$;
      else
        $\mathcal{I}_r = \mathcal{B}_i j$;
      end if
      Transform $\mathcal{P}_i$ by $\mathcal{I}_r$ in direction $\mathcal{D}_{-j}$; ▷ Restore $\mathcal{P}_i$ back to its origin position
    end for ▷ End dragging component
  end if
end for

Algorithm 4.1 detects collisions in the assembly model while transforming CAD parts.

The 3D collision detection program is coded using C# language and runs on an 11th Gen Intel(R) Core (TM) i7-11800H @ 2.30 GHz laptop processor. The simple case study, Figure 4.4, verifies the proposed method. The exploded view shows detailed information of the components. The bounding box presents a graphical bounding box generation result with a numerical result listed in the

Table A.5. The generated space interference matrix using the collision detection method is presented in the equation (12).

Table 4.4 The extracted bounding box coordinate values of the simple case study.

No.	Part Name	Extracted coordinate values of bounding box vertices (mm)					
		x_{min}	y_{min}	z_{min}	x_{max}	y_{max}	z_{max}
1	Side plate	202.08	29.46	17.72	212.08	159.95	118.36
2	Top plate	8.67	18.01	151.54	149.86	100.12	156.54
3	Base	0.00	0.00	0.00	200.00	180.00	150.00
4	Side Bolt	125.37	89.96	48.57	223.16	110.03	71.42
5	Top Bolt	68.08	48.94	88.62	91.91	71.05	165.31

Table 4.5 The calculated bounding box dimensions of the simple case study.

No.	Part Name	Calculated bounding box dimensions (mm)		
		$\|B_X\|$	$\|B_Y\|$	$\|B_z\|$
1	Side plate	10.00	130.49	100.64
2	Top plate	141.19	82.11	5.00
3	Base	200.00	180.00	150.00
4	Side Bolt	97.79	20.07	22.85
5	Top Bolt	23.83	22.11	76.69

$$S_{X+}=\begin{bmatrix}0&1&0&0&0\\0&0&0&0&0\\0&0&0&1&1\\0&0&1&0&0\\1&1&1&0&0\end{bmatrix}\quad S_{Y+}=\begin{bmatrix}0&1&0&0&0\\1&0&0&0&1\\0&0&0&1&1\\0&0&1&0&0\\0&1&1&0&0\end{bmatrix}\quad S_{Z+}=\begin{bmatrix}0&1&0&0&0\\1&0&0&0&1\\0&0&0&0&0\\0&0&1&0&0\\0&1&1&1&0\end{bmatrix}$$
$$S_{X-}=\begin{bmatrix}0&0&0&0&1\\1&0&0&0&1\\0&0&0&1&1\\0&0&1&0&0\\0&0&1&0&0\end{bmatrix}\quad S_{Y-}=\begin{bmatrix}0&1&0&0&0\\1&0&0&0&1\\0&0&0&1&1\\0&0&1&0&0\\0&1&1&0&0\end{bmatrix}\quad S_{Z-}=\begin{bmatrix}0&1&0&0&0\\1&0&0&0&1\\0&0&0&1&1\\0&0&0&0&1\\0&1&0&0&0\end{bmatrix} \tag{10}$$

4.4 Summary

This chapter proposes a space interference matrix generation method aided by CAD models. An EoL product can be represented using a CAD assembly model in the CAD environment. Each component in the EoL product can be considered one CAD component in the CAD assembly. A CAD assembly model stores the geometrical structure of the CAD parts and the precedence relationship between CAD parts. In the disassembly sequence planning problem, this information is the same as the space interference matrix. The number of components in the EoL product CAD model is the size of the space interference matrix. The space interference relation between the EoL product CAD parts is the value of the matrix element. Therefore, this chapter proposes a bounding box generation method to measure the size of CAD parts in three dimensions. Then, an API program for controlling CAD parts moving in the CAD environment is introduced. A selected CAD part can perform either a translation or a rotation under the API

program instructions. Their space interference relations can be obtained by iteratively checking the six disassembly directions of every CAD part. Thus, the directional space interference matrix of the EoL product is formed to generate a feasible disassembly sequence.

Chapter 5 Robotic disassembly sequence planning and optimisation

This chapter contains five sections. The first section provides a pseudocode for generating feasible disassembly sequences with a built-up mathematical matrix-based explanation. The second section introduces the Bees Algorithm and its enhanced discrete version as the optimisation algorithm for identifying the best robotic disassembly sequence result. The third section describes a set of indices designed to evaluate the performance of the optimisation algorithm. The fourth section shows the results. The fifth section summarises this chapter.

5.1 Generation of feasible robotic disassembly sequences

This chapter focuses on developing a robotic disassembly sequence planning program. Two algorithms are used: the disassembly sequence generation (DSG) algorithm and the disassembly sequence optimisation (DSO) algorithm. The input of the DSG algorithm includes the extracted CAD model data, the collected disassembly information in Chapter 3, and the generated directional space interference matrix in Chapter 4. The purpose of the DSG algorithm is to create a feasible disassembly sequence with the direction of the EoL product from the input. An n-component structured EoL product has the maximum n! possibility of its disassembly sequence. However, not all of them are feasible because of the precedence relationship between the components. Therefore, the DSG algorithm is designed to formulate a feasible disassembly sequence from the input information of the EoL product. With sufficient information input, such as the directional space interference matrix, a sequence of disassembly directions can also be formulated. Because the DSP problem has been identified as an NP (nondeterministic polynomial) problem, it is difficult and almost impossible to exhaust all the feasible disassembly sequences of an EoL product at once. In addition, for every feasible disassembly sequence, its corresponding disassembly sequence of direction can be a large group because some components have more than one freedom of the disassembly direction. Thus, the output of this DSG

algorithm is designed for one feasible disassembly sequence with a corresponding sequence of directions. By repeatedly running this algorithm, multiple feasible disassembly sequences and their directions can be formulated.

Disassembly sequence optimisation is another area of intense research within the DSP problem. Disassembly sequence optimisation aims to find the best or the near-best solution for the disassembly sequence. The objective in a disassembly sequence optimisation problem can be selected from a single objective to multiple objectives. Several objectives in the disassembly can be chosen as the target in the optimisation, such as the disassembly time, the economic cost, the consumed human labour, the value of remanufacturing, etc. According to the selected objective, the fitness function can be determined. In this work, the disassembly objective is set as the total disassembly time, and the fitness function is given in Chapter 3.1.

Algorithm 1: Disassembly Sequence Generation (DSG) Algorithm

Input:$M(Direction)$, $M(Tool)$

Output: $DisSeq, DisDirect\ and\ DisTool$

1: Sum $M(Direction)$ by row → $M(Direction)_{sum}$

2: Conjoin $M(Direction)_{sum}$ by the order of $Direction(j)$ → $M(Total)_{sum}$

3: $M(Total)_{sum}$ → Generate DetComp,DetDirect

4: $DetComp, DetDirect$ → Random select$detcomp_m, detdirect_{mp}$

5: Assign $detcomp_m, detdirect_{mp}$ → $DisSeq, DisDirect$

6: REPEAT $(loop = 2)$:

7: Generate $M(Component)$ ← $M(Direction)$

8: Update to $M'(Total)_{sum}$: Remove $M(Component)$ ← $M(Total)_{sum}$

9: Steps 4-5

10: $loop \leftarrow loop + 1$

11: UNTIL $(loop = n)$

12: Reorder then assign $M(Tool)$ → $DisTool$

Algorithm 5.1 Pseudocode of the Disassembly Sequence Generation Algorithm

Algorithm 5.1 presents a pseudocode of the disassembly sequence generation (DSG) algorithm, which uses directional space interference matrices as input and demonstrates a process for generating feasible disassembly sequences with directions as the output. A matrix-format mathematical model is built to explain the data processing based on this algorithm.

Directional space interference matrix:

$$M(Direction) = \begin{bmatrix} a_{11}(d_j) & a_{12}(d_j) & \cdots & a_{1n}(d_j) \\ a_{21}(d_j) & a_{22}(d_j) & \cdots & a_{2n}(d_j) \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}(d_j) & a_{n2}(d_j) & \cdots & a_{nn}(d_j) \end{bmatrix} \quad (11)$$

Full optional disassembly direction:

$$Direction(j) = \{d_1, d_2, \ldots, d_j\} \quad (12)$$

Disassembly tool:

$$Tool = \{t_1, \ldots t_{m,} \ldots t_n\} \qquad 1 \leq m \leq n \quad (13)$$

The space interference matrix shown in equation (11), and the disassembly tool list shown in equation (13), are the two input to the DSG algorithm. This space interference matrix describes disassembly precedence relationships between components of an EoL product. This is an $n \times n$-dimensional matrix in which n is the total number of components in the EoL product. The disassembly tool list stores the tool information for each of the component. The number of this matrix is determined by the optional disassembly directions *j* of the EoL product; see equation (12). For example, if an EoL product has four optional disassembly directions, $Direction(j = 4) = \{X+, X-, Y+, Y-\}$, there will be four directional space interference matrices, where $M(X+)$, $\mathrm{M}(X-)$, $\mathrm{M}(Y+)$,

and M($Y-$) is the input to the DSG algorithm. In the following, each process step in DSG is individually explained.

Directional summation matrix:

$$M(Direction)_{sum} = \begin{bmatrix} \sum_{i=1}^{n} a_{1i}(d_j) \\ \sum_{i=1}^{n} a_{2i}(d_j) \\ \vdots \\ \sum_{i=1}^{n} a_{ni}(d_j) \end{bmatrix} \quad (14)$$

Step 1: Check the interference situation of each component.

$M(Direction)_{sum}$ is the directional summation matrix. It represents the overall interference of each component. This matrix has an $n \times 1$ dimension; see equation (14). For each row element, the value is a sum of the values in the directional space interference matrix, $a_{n1}(d_j) + a_{n2}(d_j) + \cdots + a_{nn}(d_j)$.

Total summation matrix:

$$M(Total)_{sum} = \begin{bmatrix} \sum_{i=1}^{n} a_{1i}(d1) & \sum_{i=1}^{n} a_{1i}(d2) & \cdots & \sum_{i=1}^{n} a_{1i}(dj) \\ \sum_{i=1}^{n} a_{2i}(d1) & \sum_{i=1}^{n} a_{2i}(d2) & \cdots & \sum_{i=1}^{n} a_{2i}(dj) \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n} a_{ni}(d1) & \sum_{i=1}^{n} a_{ni}(d2) & & \sum_{i=1}^{n} a_{ni}(dj) \end{bmatrix} \quad (15)$$

Step 2: Obtain a full view of the interference situation of the whole EoL product. The total summation matrix $M(Total)_{sum}$, equation (15), is obtained by combining every directional summation matrix from step 1 in the order of $Direction(j)$. This joint provides a full view of the interference situation of the whole EoL product. The dimension of the total summation matrix is $n \times j$. If the value of its element equals 0, component n is free to detach in the direction dj from the EoL product; otherwise, this component is blocked.

Detachable component with direction set:

$$DetComp = \{detcomp_1, \dots detcomp_m\} \qquad 1 \leq m \leq n \quad (16)$$

$$DetDirect = \{detdirect_{11}, \dots, detdirect_{1p}, \dots, detdirect_{m1}, \dots, detdirect_{mp}\} \; 1 \leq p \leq j \quad (17)$$

Step 3: Construct a selection pool for the detachable components and directions. With the detachable information provided in step 2, the column of the total summation matrix reflects all the detached-available components in direction dj, and the row of the total summation matrix reflects all the detached-available directions of component n. Equations (16) and (17) are constructed as a selection pool for storing the information.

Step 4-5: Select a disassembled component with direction.

The disassembled component with its disassembly direction is randomly selected from the selection pool and then assigned to the set of the disassembly sequence with direction. For example, if picking $detcomp_m$ $(1 \leq m \leq n)$ and $detdirect_{\mathrm{mq}}$ $(1 \leq q \leq p)$ as the disassembled component and the disassembly direction, the disassembly sequence with direction will be updated as equations (20) and (21). Moreover, this information is removed from equations (16) and (17). These two steps generate the first disassembled component with its direction, and the remaining disassembled components with their directions follow the same approach but under an updated total summation matrix each time, which is explained in the next steps.

Component space interference matrix:

$$M(Component) = \begin{bmatrix} a_{1n}(d_1) & a_{1n}(d_2) & \cdots & a_{1n}(d_j) \\ a_{2n}(d_1) & a_{2n}(d_2) & \cdots & a_{2n}(d_j) \\ \vdots & \vdots & \ddots & \vdots \\ a_{nn}(d_1) & a_{nn}(d_2) & \cdots & a_{nn}(d_j) \end{bmatrix} \quad (18)$$

The updated total summation matrix is:

$$M'(Total)_{sum}$$

$$= \begin{bmatrix} \sum_{i=1}^{n} a_{1i(d1)} - a_{1n(d1)} & \sum_{i=1}^{n} a_{1i(d2)} - a_{1n(d2)} & & \sum_{i=1}^{n} a_{1i(dj)} - a_{1n(dj)} \\ \sum_{i=1}^{n} a_{2i(d1)} - a_{2n(d1)} & \sum_{i=1}^{n} a_{2i(d2)} - a_{2n(d2)} & \cdots & \sum_{i=1}^{n} a_{2i(dj)} - a_{2n(dj)} \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n} a_{ni(d1)} - a_{nn(d1)} & \sum_{i=1}^{n} a_{ni(d2)} - a_{nn(d2)} & & \sum_{i=1}^{n} a_{ni(dj)} - a_{nn(dj)} \end{bmatrix} \quad (19)$$

Step 6-11: Repeat the selecting actions using the updated total summation matrix. The total summation matrix reflecting the interference situation of the whole EoL product needs to be updated every time the combined information of the disassembled component and direction is selected. $M(Component)$ is a component space interference matrix with $n \times j$ dimensions that shows the

interference situation of a component in full directions. This matrix is generated by extracting the corresponding column from equation (11) and then conjoining by the order of equation (12). Disassembling a component from the EoL product involves removing its component space interference matrix from the total summation matrix. Equation (19) represents the reduction. By repeatedly executing steps 4-5 in $(n-1)$ times, the following disassembled components and their disassembly directions can be generated in the disassembly sequence with direction. The disassembly tool list is then updated with the order of the generated disassembly sequence. Then the program output the results of the DSG algorithm.

$$DisSeq = \{disseq_1(detcomp_m), \dots, disseq_n\} \qquad 1 \le m \le n \quad (20)$$

$$DisDirect = \left\{disdirect_{1\mathrm{p}}(detdirect_{\mathrm{mq}}), \dots, disdirect_{np}\right\} \quad 1 \le q \le p,\ p \in \{1, j\} \quad (21)$$

$$DisTool = \{distool_1(detcomp_m), \dots, distool_n\} \qquad 1 \le m \le n \quad (22)$$

A case study for further explaining the DSG is presented using the central unit dismantling process (components C1-C4 and D) of the EVB CAD model (Figure 3.4). The space interference matrix using the collision detection method as the DSG input is presented in the equation (23). The disassembly tool list is presented

in the equation (24), with 1 as a screwdriver, and 2 as a gripper. The $\mathrm{M(Direction)_{sum}}$ and the $\mathrm{M(Total)_{sum}}$ are presented in the equation (25) and equation (26) separately. If the component C2 is selected to be disassembled from the direction Z+, the $\mathrm{M(C2)}$ is presented in the equation (27), and the $\mathrm{M(Total)_{sum}}$ is updated by $\mathrm{M'(Total)_{sum}}$. By repeating the steps, one feasible disassembly sequence, direction and tool result of the EVB CAD model can be generated as the equations (29), (30) and (31).

$$
M(X+) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}
$$

$$
M(X-) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}
$$

$$
M(Y+) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}
$$

$$
M(Y-) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}
$$

$$
M(Z+) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}
$$

$$
M(Z-) = \begin{array}{c} \\ C1 \\ C2 \\ C3 \\ C4 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} C1 & C2 & C3 & C4 & D \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}
$$

(23)

$$
Tool = \{1, 1, 1, 1, 2\} \tag{24}
$$

$$M(X+)_{sum}=\begin{bmatrix}1\\2\\2\\1\\4\end{bmatrix}\quad M(Y+)_{sum}=\begin{bmatrix}2\\2\\1\\1\\4\end{bmatrix}\quad M(Z+)_{sum}=\begin{bmatrix}0\\0\\0\\0\\4\end{bmatrix}$$
$$M(X-)_{sum}=\begin{bmatrix}2\\1\\1\\2\\4\end{bmatrix}\quad M(Y-)_{sum}=\begin{bmatrix}1\\1\\2\\2\\4\end{bmatrix}\quad M(Z-)_{sum}=\begin{bmatrix}1\\1\\1\\1\\0\end{bmatrix} \tag{25}$$

$$M(Total)_{sum}=\begin{bmatrix}1&2&2&1&0&1\\2&1&2&1&0&1\\2&1&1&2&0&1\\1&2&1&2&0&1\\4&4&4&4&4&0\end{bmatrix} \tag{26}$$

$$M(C2)=\begin{bmatrix}0&1&0&0&0&0\\0&0&0&0&0&0\\0&0&0&1&0&0\\0&0&0&0&0&0\\1&1&1&1&1&0\end{bmatrix} \tag{27}$$

$$M'(Total)_{sum}=\begin{bmatrix}1&1&2&1&0&1\\2&1&2&1&0&1\\2&1&1&1&0&1\\1&2&1&2&0&1\\3&3&3&3&3&0\end{bmatrix} \tag{28}$$

$$DisSeq(EVB)=\{\mathrm{C2},\mathrm{C3},\mathrm{C4},\mathrm{C1},\mathrm{D}\} \tag{29}$$

$$DisDirect(EVB)=\{Z+,Z+,Z+,Z+,Z+,\} \tag{30}$$

$$DisTool=\{1,1,1,1,2\} \tag{31}$$

5.2 Robotic disassembly sequence optimisation and the Bees Algorithm

5.2.1 The Bees Algorithm (BA)

The Bees Algorithm (BA) is a metaheuristic algorithm whose search procedure is designed to find a near-optimal solution to complex and difficult-to-solve problems. This algorithm was proposed by Pham, Ghanbarzadeh, Koc, Otri, Rahim, and Zaidi in 2005 [102], [103]. From the perspective of the foraging behaviour of honeybees in nature, BA is inspired to mimic the procedure of finding the best flower patch. Table 5.1 shows the required parameters in the BA. In the basic version of BA, both local and global searches use a random search strategy. Because DSP is a combinatorial problem, BA, as an optimisation tool, is selected to use its discrete version to optimise the solutions. As a swarm-based optimisation algorithm, the result is improved iteration by iteration. At each iteration in the max-iteration $iter$, the search space is divided into three parts: the Elite Site n, the Selected Site m, and the rest of the sites. According to the distributions of sites, bees are assigned to search within the places. Specifically, elite site bees nb are assigned to search in the elite site, selected site bees mb are assigned to the selected sites, and the rest of the bees are assigned to the remaining sites, which are the non-selected sites in Figure 5.1. The value of all the search sites is equal to that of scout bees.

Table 5.1 Parameters required in the basic version of the Bees Algorithm

Bees			Sites		Iteration
scout bees	selected site bees	elite site bees	selected sites	elite sites	max-iterations
$scoutn$	mb	nb	m	n	$iter$

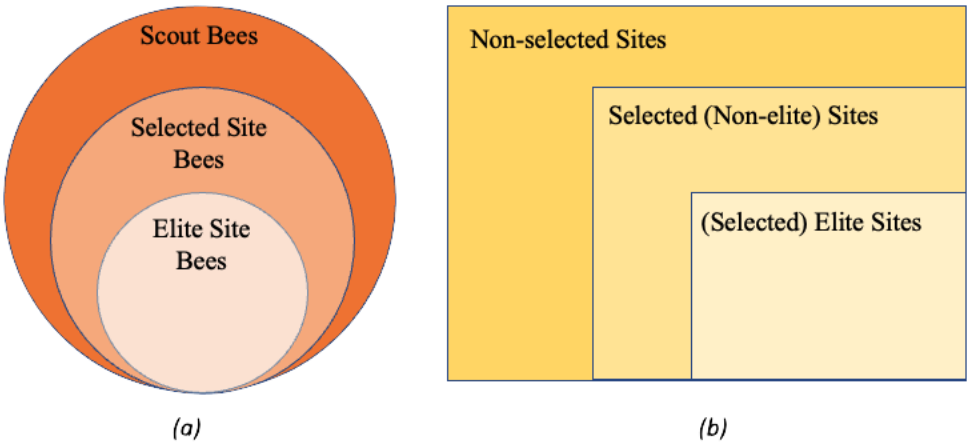


Figure 5.1 Relationships of parameters in the BA: (a) Bee assignment; (b) site distribution.

5.2.2 The information linked from RDSP to BA

The population size in the discrete BA is defined by the number of scout bees. Disassembly information is stored in the scout bees, and the bees are optimised to find the best bee according to the discrete Bees Algorithm. The best solution of the disassembly sequence with its direction is kept in that bee. Table 5.2 gives

a full view of the content of the disassembly information stored in the scout bees. These details can be found in the previous sections.

Table 5.2 Disassembly information stored in the scout bees

The parameter in the BA	The symbol of the parameter
Scout Bee Number	$scoutn_i$
Disassembly Sequence	$\{disseq_1, \dots, disseq_n\}$
Disassembly Direction	$\{disdirect_1, \dots, disdirect_n\}$
Disassembly Tool	$\{distool_1, \dots, distool_n\}$
Direction Change	C_{Direct}
Tool Change	C_{Tool}
Distance	$Distance$
Total Disassembly Time	$F(Time)$

5.3 Applying the Enhanced Discrete Bees Algorithm to RDSP

For single-objective problems, a fitness value can be calculated for the scout bee. For the problem of optimising the minimum disassembly total time, $F(Time)$ is defined and used as the sorting criterion for scout bees in every iteration. Liu, Zhou, Pham, Xu, Ji and Liu proposed an enhanced discrete version of the Bees

Algorithm for solving the robotic disassembly sequence planning problem in 2018 [91]. This study adapts the EDBA and extends the research for analysing the optimising efficiency of the algorithm. First, the search scheme of the EDBA is explained in this section. Then, two works on applying the EDBA to optimise the RDSP problem are carried on extending the research in the next section: (a) design a 30 parameter tunning cases to find the best combination of scout bees and iterations; (b) design four criteria to evaluate the performance of the EDBA.

After initialising the stored information in the scout bees, the fitness values are sorted in the order of elite site local search, selected site local search, and nonelected site global search at every iteration. Figure 5.2 shows the procedure workflow of the EDBA. In both Elite Site and Selected Site Local Search, a waggle dance search of scout bees is performed. Three types of waggled dances are implemented in this work: inserting, swapping and mutating. In the global site search, the random search strategy remains the same. At each time of the search, if new information in the scout bee is generated with a better solution, then this new scout bee successfully competes with the previous scout bee and replaces its place. Otherwise, the information is retained by the old scout bee. Until reaching the max iteration, EDBA stops searching and outputs the best fitness value.

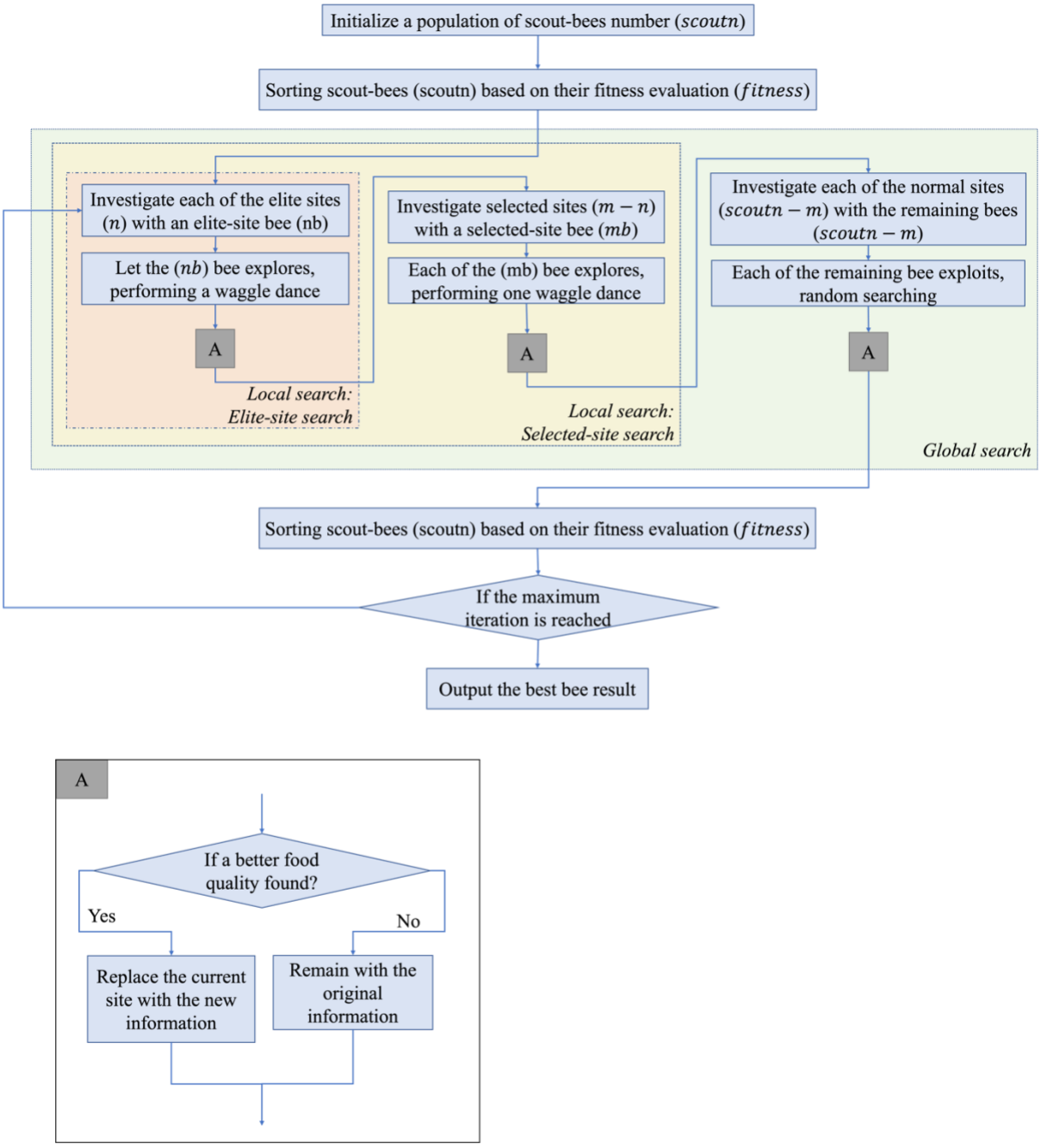


Figure 5.2 The local search and global search workflow in EDBA

5.3.1 Local Search: Inserting Strategy

Algorithm 5.2 shows the pseudocode used to describe the insertion strategy used in the local search. First, two bits a and b are randomly selected from n to locate

two positions in the generated $DisSeq$ and $DisDirect$. Then, the middle subsequence between a and b are extracted from $DisSeq$ and $DisDirect$. Next, an inserting action is performed in step 3 by putting the position of a after the position of b. Finally, both performed subsequence are returned to their extracted positions to generate the new disassembly sequence with direction $DisSeq'$ and $DisDirect$.

Algorithm 2: Inserting Strategy in the Local Search

Input:$DisSeq = \{disseq_1, \dots, disseq_n\}$

$DisDrect = \{disdirect_1, \dots, disdirect_n\}$

Output: $DisSeq'$ and$DisDirect'$

1: Randomly select two bits from n:$a, b \in \{1, n\}$ $(a < b)$

2: In the $DisSeq$ and $DisDirect$, extract $MiddleSeq$ and $MiddleDirect$:

$MiddleSeq = \{disseq_a, \dots, disseq_b\}$

$MiddleDirect = \{disdirect_a, \dots, disdirect_b\}$

3: In the $MiddleSeq$ and $MiddleDirect$, do an insertion:

$$MiddleSeq' = \{\dots, disseq_b, disseq_a\}$$

$MiddleDirect' = \{\dots, disdirect_b, disdirect_a\}$

4: Put $MiddleSeq$ and $MiddleDirect$ back to the $DisSeq$ and $DisDirect$:

$DisSeq' = \{disseq_1, \dots, disseq_b, disseq_a \dots, disseq_n\}$

$DisDirect' = \{disdirect_1, \dots, disdirect_b, disdirect_a \dots, disdirect_n\}$

Algorithm 5.2 An insertion strategy for the local search of the EDBA.

5.3.2 Local Search: Swapping Strategy

The swapping strategy is another local search approach for changing both the disassembly sequence and the disassembly direction. Unlike the inserting strategy, which extracts one bit of information placed in a new location, the swapping strategy involves two bits to form the new disassembly sequence with direction. Algorithm 5.3 shows the pseudocode of the performance of the swapping strategy. Two bits are randomly selected from the total number of components n in the EoL product. Then, the locations of these two bits are found in the generated disassembly sequence with direction. Within the disassembly sequence with direction, by changing the value at two extracted locations, a new disassembly sequence with direction is generated.

Algorithm 3: Swapping Strategy in the Local Search

Input:$DisSeq = \{disseq_1, \dots, disseq_n\}$

$DisDirect = \{disdirect_1, \dots, disdirect_n\}$

Output: $DisSeq'$ and$DisDirect'$

1: Randomly select two bits from n:$a, b \in \{1, n\}$ $(a < b)$

2: Locate $disseq_a, disseq_b$ in $DisSeq$

2: Locate $disdirect_a, disdirect_b$ in$DisDirect$

3: Do a swapping action:

$DisSeq' = \{disseq_1, \dots, disseq_b, \dots, disseq_a \dots, disseq_n\}$

$DisDirect' = \{disdirect_1, \dots, disdirect_b, \dots, disdirect_a \dots, disdirect_n\}$

Algorithm 5.3 A swapping strategy in the local search of EDBA.

5.3.3 Local Search: Mutation Strategy

The mutation strategy focuses on a one-bit change to the disassembly sequence with direction. In this research, the mutating action specifically targets the disassembly direction. Algorithm 5.4 shows the mutating procedure.

Algorithm 4: Mutating Strategy in the Local Search

Input:$DisSeq = \{disseq_1, \dots, disseq_n\}$

$DisDirect = \{disdirect_1, \dots, disdirect_n\}$

Output: $DisSeq'$ and$DisDirect'$

1: Randomly select one bit from n:$a \in \{1, n\}$

2: Locate $disdirect_a$ in$DisDirect$

3: Do a mutating action:

$DisSeq' = DisSeq$

$DisDirect' = \{disdirect_1, \dots, disdirect'_a, \dots, disdirect_n\}$

Algorithm 5.4 A mutation strategy in the local search of EDBA.

5.4 Optimisation algorithm performance evaluation

5.4.1 Design of Experiments

Previous studies have compared the performance of EDBA with that of other optimisation algorithms, such as the Genetic Algorithm with Precedence Preserving Crossover (GA-PPX) and Self-Adaptive Simplified Swarm Optimisation (SASSO), using two gear pump models as case studies [91]. A comparison of the results shows that under different settings of the scout bee population and the max-iterations, EDBA has the ability to achieve a generally fast average algorithm runtime and a good average solution fitness value. This conclusion is obtained under a combination of parameter settings with selected site ($m = 4$), elite site ($n = 1$), selected site bees ($mb = 1$), and elite site bees ($nb = 2$) [91]. Therefore, in this work, these four parameters are set up while the rest of the parameters are adjusted to design in the following experiments. However, the comparison between the EDBA and other algorithms can show an optimising superiority of the EDBA, but it cannot evaluate the performance of the EDBA itself. This work aims to evaluate the optimisation performance of the EDBA and explore how the search schemes affecting results by iterations.

The experimental design is divided into 30 cases, the order of which is shown in Figure 5.3. The cases are divided into six groups according to the value of $iter$.

The range of $iter$ is from 200 to 1000 with 200 as a gap, plus a value of 2000 to show a trend. The range of $scoutn$ in each group is set to 10, 20, 30 and 40, plus a value of 100.

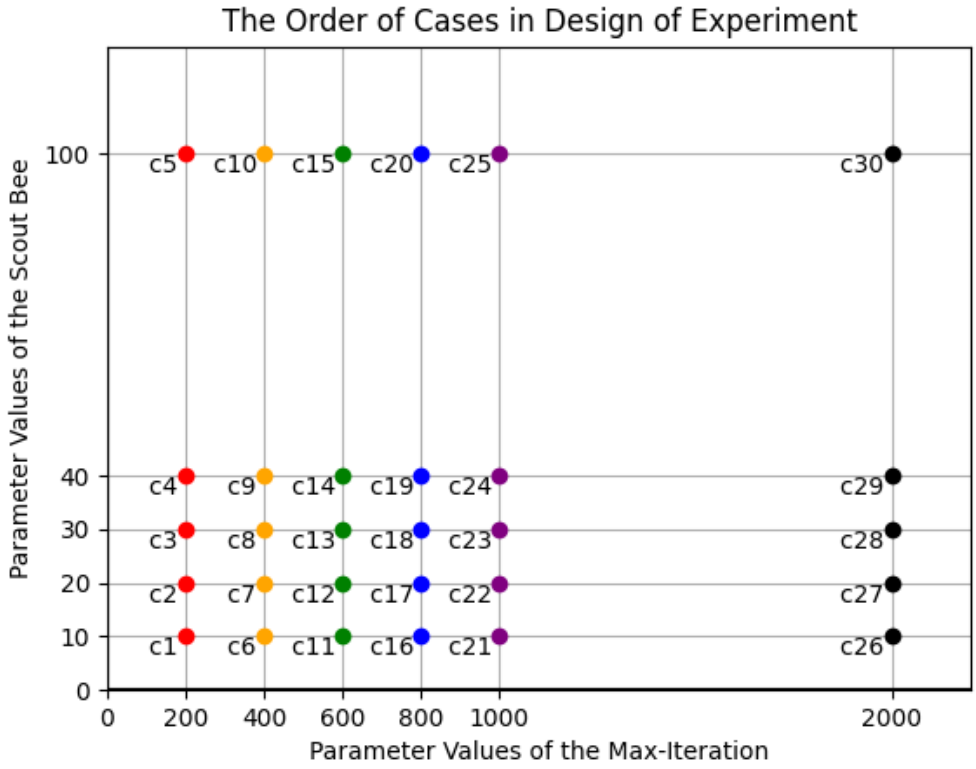


Figure 5.3 The order of the cases in the design of the experiment

5.4.2 Design of Evaluations Criteria

Four aspects are focused on reflecting the RDSP program performance: the best fitness value, the optimised ratio, the convergence iterating point, and the program execution time. These evaluations are recorded at three stages: a) after

the DSG algorithm, b) after the first iteration of the DSO algorithm, and c) at the iteration when the DSO algorithm converges.

The fitness function is repeatedly called and sorted during the RDSP program execution. Thus, the current best fitness can be obtained at any program execution moment. Based on the three stages mentioned above, three values, namely the fitness of initialisation, the fitness of first iteration and the optimised fitness (fit_{DSG}, $fit_{DSOiter1}$ and $fit_{DSOconverg}$), are recorded.

The optimised ratio is used to evaluate how much solution space is optimised upon the converged best fitness in an experiment. Equation (32) is the optimised ratio since the first stage; equation (33) is the optimised ratio since the second stage. The optimised ratio for the rest of the execution time can be obtained via the same method when the optimisation efficiency during the iterations must be investigated in the future.

$$ratio_{DSG} = \frac{fit_{DSG} - fit_{DSOconverg}}{fit_{DSG}} \tag{32}$$

$$ratio_{DSOiter1} = \frac{fit_{DSOiter1} - fit_{DSOConverg}}{fit_{DSOiter1}} \tag{33}$$

Because of the randomness in the metaheuristic algorithms, as well as in the EDBA, the best fitness value and the convergence iterating point are rarely the same when repeating the program. However, from the engineering aspect, to apply the RDSP under an actual remanufacturing process, only one chance of a program running is generally requested for pursuing real-time signal transportation. Therefore, recording the convergence iterating points in repeated experiments under the same parameter combination of EDBA is of interest in this work. The average number of convergence iteration points and their distributions are calculated, and the results are presented in the next section.

Finally, the program execution time should not be neglected when solving a practical problem. For example, a 500-second runtime was set as a stop-ping criterion in the research of surgery scheduling problems [104]; two blockchain platforms with varying numbers of transactions are compared by execution time to reflect the response speeds [105]. From an engineering aspect, program execution time can provide an intuitive understanding of how to transfer the methodology to a similar model or problem size. However, a runtime analysis from the computer science aspect is also encouraged to present an empirical and more precise understanding of the algorithms [106], [107], [108].

5.5 Results

The 30 experimental cases were tested in an 8 GB Memory Apple M1 processor laptop with the Python programming language using the PyCharm environment. Each testing program of the 30 cases consists of three parts: (1) processing the input data from an Excel file; (2) generating the feasible disassembly sequences with directions; and (3) optimising the generated results. The input data processed in the first part include the extracted directional space interference matrices, the directional distance matrices, and the collected disassembly information; the second part uses the DSG algorithm; and the third part adapts the EDBA algorithm. The methods used to obtain these materials were described in the previous sections.

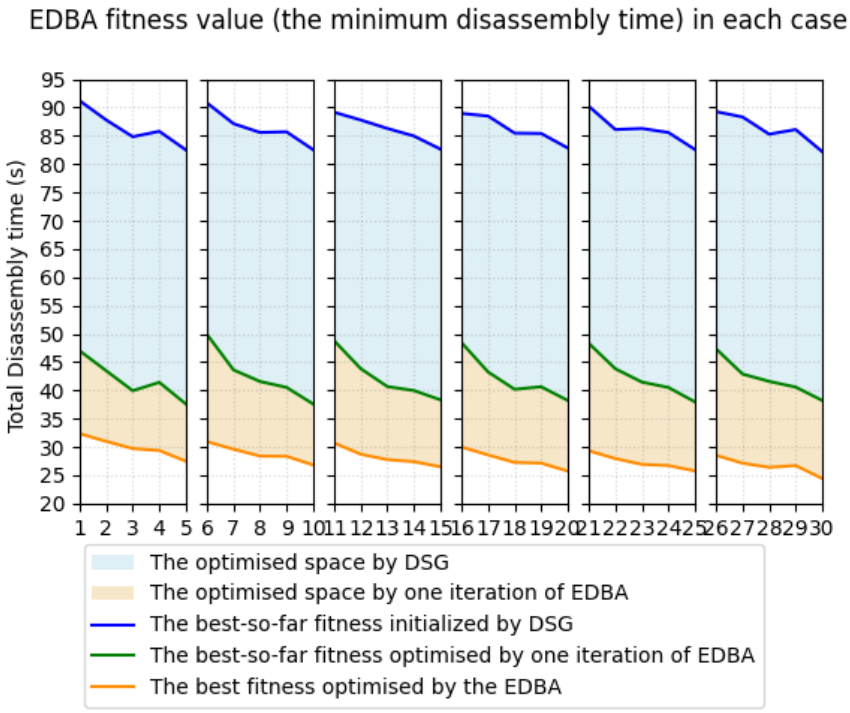


Figure 5.4 The best fitness value at three stages and the optimised ratio between them

Figure 5.4 shows the best fitness values at three stages and the optimised ratio between the stages. The blue line is the best fitness value in the first stage, which shows a similar downwards trend between 80 s and approximately 90 s among the six groups. This is because the number of scout bees has an impact on the generated solutions. The more scout bees that are recruited, the better the value of the solutions. Because this is the first stage after the DSG algorithm, solution optimisation has yet to be performed. This trend is caused only by scout bees, the initialised population. The green line is the best-so-far value in the second stage. It reflects the optimised fitness values after one iteration of the DSO problem. The declining trend is similar to the blue line among all groups, from above 35 s

to less than 50 s. The orange line is the converged fitness value when the algorithm reaches the max-iteration, showing that the EDBA can improve the total disassembly time, in this case, to less than 25 s. It is worth noting that the blue optimised space is approximately three times larger than the orange optimised space, which indicates the importance of the first iteration of the DSO algorithm rather than the following iterations.

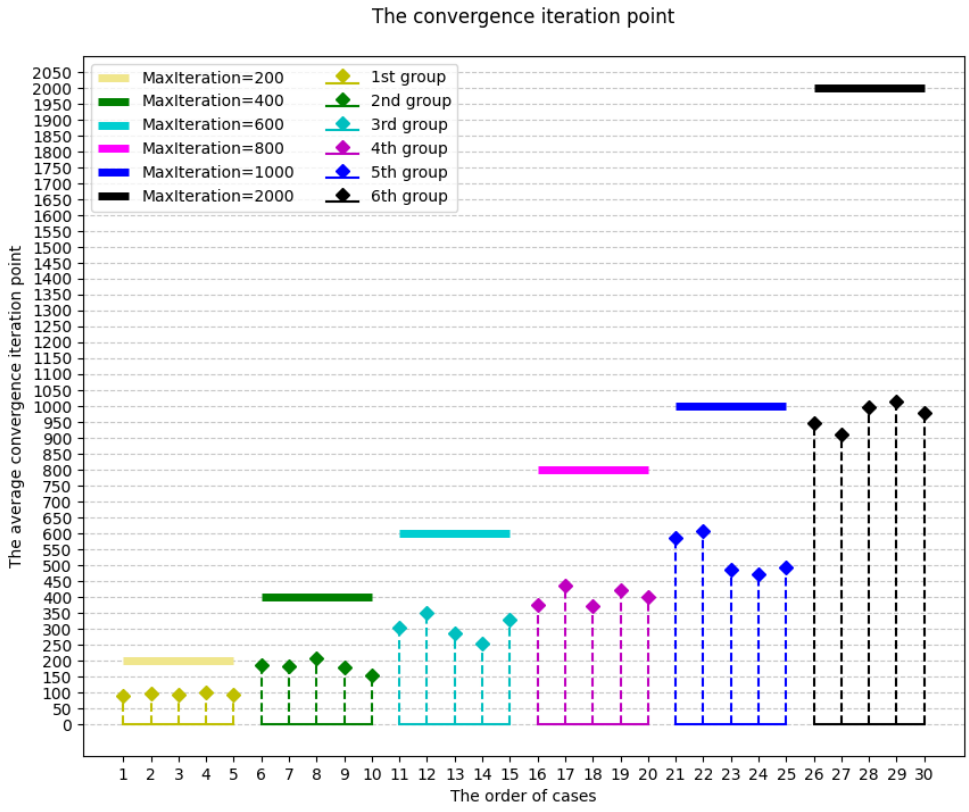


Figure 5.5 The average number of convergences iterating points in the six groups.

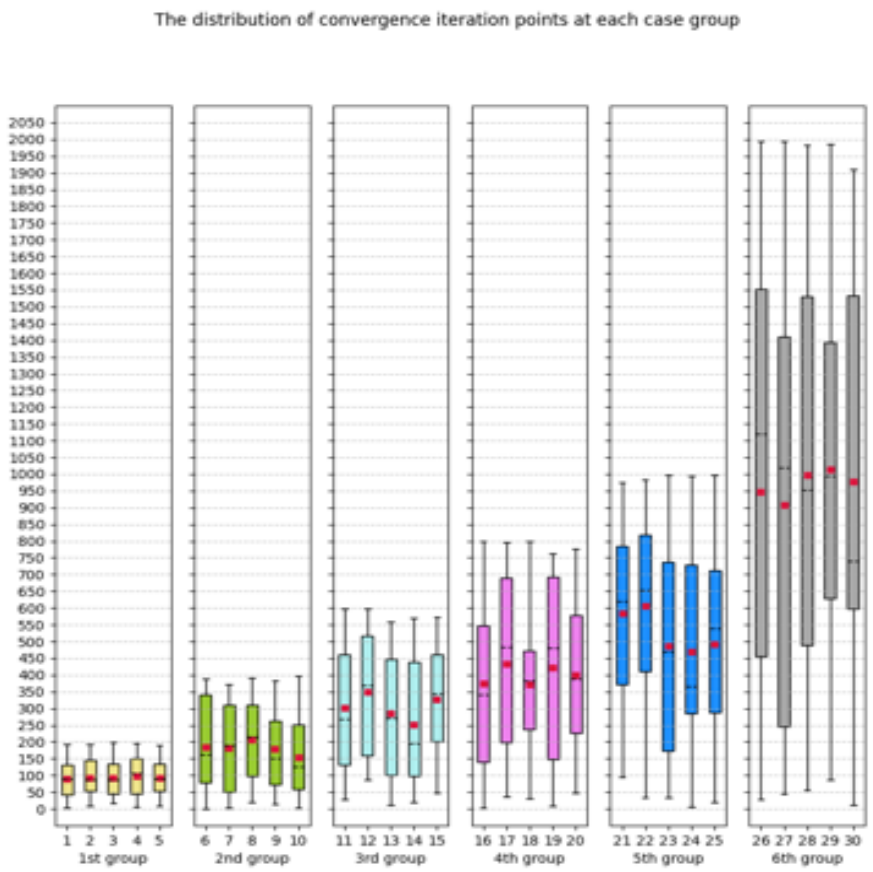


Figure 5.6 The distribution convergence iterating points in six groups.

Figure 5.5 and Figure 5.6 are the averages and distributions of the convergence iterating points in the six groups, respectively. The average convergence iterating points result shows that EDBA converges on average approximately half of the maximum iteration value.

Figure 5.7 shows the execution time of the program. The total execution time of the whole program (blue column) includes the input data processing time (grey column) and the sum of the execution time of the DSG algorithm and the EDBA

algorithm (orange column). The number of scout bees has an impact on the execution time. Under the same max-iteration, the more scout bees are recruited in the EDBA, the longer the execution time will be. When comparing the groups, this characteristic is more distinct.

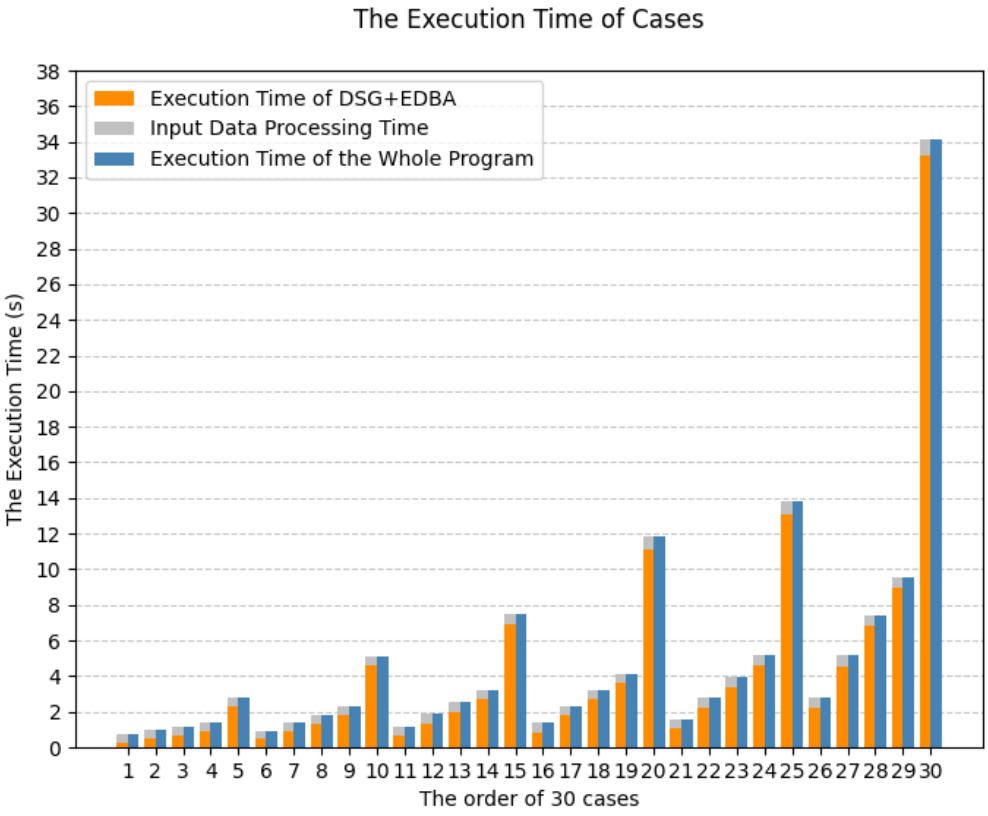


Figure 5.7 The execution time of 30 cases

5.6 Summary

The execution process of RDSP and its program performance are introduced in this chapter. The RDSP is a combination of disassembly sequence generation and disassembly sequence optimisation problems. The disassembly sequence

generation problem focuses on generating feasible disassembly sequences. An initialised population pool is considered to provide individual solutions for later optimisation. However, the disassembly sequence optimisation problem aims to find the best or the near-best solution as an output. The number of its optimisation objectives can be one or more than one. This work selects the minimum total disassembly time as the optimisation target. The metaheuristic algorithm is a popular method for solving this optimisation problem. The Bees Algorithm was created in 2005 and has been extensively applied in solving RDSP problems. An enhanced discrete version of the Bees Algorithm is used in this work as the optimisation method for the RDSP. Three local search strategies are used: inserting, swapping and mutating. The global search strategy remained random. The impacts of two parameters in EDBA, the scout bee and the maximum number of iterations, are explored via a designed experiment. Four criteria are proposed to compare the program performance. The results suggest that although increasing the number of scout bees and max-iterations can help the EDBA algorithm find an improved solution, the efficiency of the improvement decreases. From an engineering aspect, to apply the RDSP method in practical robotic disassembly cases or to develop a robotic disassembly sequence replanning strategy in the future, the choice of parameter values is a trade-off problem.

Chapter 6 Conclusion

This chapter contains three sections. The first section summarises the achievements detailed in the previous three chapters, namely, robotic disassembly information extraction in Chapter 3, CAD model collision detection in Chapter 4, and performance evaluation of the optimisation algorithm in Chapter 5. The second section describes the contributions made in this thesis. The third section recommends future research directions for continually developing a CAD-based integrated robotic disassembly sequence planning system.

6.1 Summary of the achievements

This thesis has built and presented an integrated robotic disassembly sequence planning system using the CAD models as input. The research hypothesis has been answered and verified in Chapter 2, Chapter 3, Chapter 4, and Chapter 5. First, Chapter 2 reviewed the development history of the DSP problem that were proposed since 1987. A further literature investigation was carried on the cross-disciplines of the CAD techniques and the metaheuristics with the DSP problem. A research gap was found during the review process that very little work attempted to build a deep connection between the CAD technique and the DSP problem. However, this thesis found that, instead of manually processing the disassembly information from the CAD model of an EoL product, it is possible to automatically extract the required disassembly information as long as the information is properly defined. So, the methodologies in the following two chapters were dedicated to achieving an automated information collection and transferring the extracted CAD data in the service of an input to the RDSP problem. There are three works done in the Chapter 3: explaining a RDSP problem; showing a work flowchart of the RDSP system; and defining the disassembly information required in the RDSP. An add-in tool was designed for collecting disassembly information extracted from the CAD models. In the Chapter 4, a bounding box generation method was proposed to evaluate the

dimension of CAD parts, and a collision detection method was developed to test the interference conditions between the CAD parts for generating space interference matrices. The collected disassembly information and the extracted space interference matrices were then used as problem input in the Chapter 5. The process of generating feasible disassembly sequences was given in the beginning. Then, the EDBA was used to be an optimisation method for finding the optimal disassembly sequence from all the possible solutions. Because there was no work has been explored on the searching schemes in the EDBA, 30 cases of tuning the EDBA's parameters was done in Chapter 5. The EDBA's fitness values in different stages, the convergence points in cases, and the execution time of the EDBA was recorded. It is suggested that as a metaheuristic optimisation method, the max-iteration can be controlled when the optimising result is good enough, thus saving the execution time.

The following achievements were made in this thesis:

1. Design of the RDSP system: Objective 1 is achieved in Chapter 3, where an overall structure of this system is given to present the order of data and information processes. The disassembly operator is selected using robots and the system optimisation objective is set as the minimal total disassembly time.
2. Disassembly information recognition: Objective 2 is achieved in Chapter 3, where the fitness function of this RDSP system is given. The constituent elements in the fitness function include the basic disassembly time of the

component, moving time of the robot in between the components, and the penalty time covering tool changes and direction changes. The measurement of the dimensions of components is described in Chapter 4, providing the information of the robot's moving distance from point to point.

3. Collision detection algorithm development: Chapter 4 addresses Objective 3. The interference conditions between parts are tested using an API-controlled program. The program gives positive feedback if a collision happens, and negative feedback means no collision. Six disassembly directions are tested one by one. Then, the space interference matrix is generated by getting all the feedback from the program.
4. Processing the space interference matrix: Objective 4 is achieved in Chapter 5. The generated space interference matrices are processed in this chapter for obtaining all the feasible disassembly sequences. A matrix-based mathematical model is built to explain the details of this process.
5. Optimising the feasible disassembly sequences and finding the best combination of parameter settings: Chapter 5 further achieves the objective 5 by finding the best disassembly sequence using EDBA according to the fitness function established in Chapter 3. The fitness of each iteration is recorded and analysed in the 30 designed cases. Three stages during the optimisation are focused on, namely, the fitness of initialisation, the fitness of first iteration and the optimised fitness. The optimisation effect increases with more

considerable computational resources input while the optimisation efficiency improves slightly.

6.2 Contributions

The following key points summarise the contributions of this thesis:

1. Background investigation and research motivation: A thorough investigation of the research background begins with the circular economy, in which the core idea is to extend the usage of products in the market for as long as possible. Remanufacturing is revealed as a solid tool to achieve this concept. Disassembly is a crucial step in remanufacturing and taking the recollected products into parts. Due to the lack of digitalisation and degree of automation in the traditional remanufacturing process, robotic disassembly is an efficient solution. CAD simulation is proposed to be applied in the remanufacturing disassembly research in Chapter 1.
2. Literature review: A descriptive and idiographic literature review is given in Chapter 2. The collection of literature uses Scopus as the search engine to get the relative literature that is newly published. The search terms include “DSP”, “CAD”, and “metaheuristic”, which highly connect to this thesis. Considering

the long development of DSP, review papers on this topic are particularly focused.

3. Disassembly information collection and data extraction from CAD models: Chapter 3 introduces an overall system design, clarifies the disassembly information related to the optimisation goal of this thesis, and presents the extracted CAD data using an add-in tool.
4. Collision detection method in CADs and space interference matrix generation: Space interference matrix is a matrix-based method in DSP problem to describe the components' relationship of an EoL product. In Chapter 4, an algorithm is developed to one-by-one check components from six disassembly directions ($X+, X-, Y+, Y-, Z+, Z-$) using the collision detection function in the CAD simulation.
5. RDSP's feasible sequence generation and optimisation using EDBA: Chapter 5 builds a mathematical matrix calculation by taking the space interference matrix as an input and outputs the feasible disassembly sequences. The disassembly direction for each disassembly step is also calculated. Then, in its enhanced discrete version, the Bees Algorithm is applied to all the feasible disassembly sequences to find the best result under the optimisation objective as the minimal disassembly time. The disassembly tool, change of direction, and change of tool for each disassembly step are included in the optimisation fitness function. The fitness results of each iteration are recorded. Four evaluation indices, the fitness of initialisation, the fitness of the first iteration,

and the optimised fitness, are tracked for further understanding of the optimisation scheme of EDBA.

6.3 Future work

This thesis has presented a CAD-based integrated robotic disassembly sequence planning system. Compared to the traditional DSP solutions by manually building disassembly models for EoL products, CAD models present an approach to reduce the involvement of humans. The future research directions are recommended below:

1. Reversed engineering: The use of reverse engineering technology for the recollected EoL products can genuinely reflect the damage of products in the use process. More accurate disassembly sequence planning results can be obtained using the CAD model generated by reverse engineering as the input of this system.
2. Digital twin: The system in this thesis can be seen as the virtual world of digital twin technology, synchronously connected with the disassembly process of robots in the real world and creating an instant interaction of digital information between the worlds. This interaction can help the system obtain real-time data from the real world so to adjust the planning sequence in time when the robot encounters disassembly obstacles.

3. Disassembly direction: The current disassembly direction only considers six directions ($X+, X-, Y+, Y-, Z+$, and $Z-$). When disassembly work is carried out, one of the six disassembly directions is an obstacle by the fixture. Thus, only five disassembly directions are selectable. Also, some disassembly points of the EoL product are difficult to reach in actual disassembly conditions. These influence the reachability of the robot's end effector. Therefore, a five-axis fixture is suggested in future disassembly work, where a fixture can simultaneously move the EoL product in five different axes.
4. Search strategies in the EDBA: Three local search strategies were adopted in this thesis, namely, inserting, swapping and mutation. They were evaluated using a set of optimization performance indices to understand the optimizing processes of EDBA iteration by iteration. Future research can further investigate the optimizing processes of the local search strategies individually.

References

[1] S. Sahoo and S. K. Jakhar, ‘Industry 4.0 deployment for circular economy performance—Understanding the role of green procurement and remanufacturing activities’, *Bus. Strategy Environ.*, vol. 33, no. 2, pp. 1144–1160, 2024, doi: 10.1002/bse.3542.

[2] A. Sangwongwanich, D.-I. Stroe, C. Mi, and F. Blaabjerg, ‘Sustainability of Power Electronics and Batteries: A Circular Economy Approach’, *IEEE Power Electron. Mag.*, vol. 11, no. 1, pp. 39–46, 2024, doi: 10.1109/MPEL.2024.3356248.

[3] S. Burmaoglu, D. Ozdemir Gungor, A. Kirbac, and O. Saritas, ‘Future research avenues at the nexus of circular economy and digitalization’, *Int. J. Product. Perform. Manage.*, vol. 72, no. 8, pp. 2247–2269, 2023, doi: 10.1108/IJPPM-01-2021-0026.

[4] S. Khan, A. Haleem, and N. Fatma, ‘Effective adoption of remanufacturing practices: a step towards circular economy’, *J. Remanuf.*, vol. 12, no. 2, pp. 167–185, 2022, doi: 10.1007/s13243-021-00109-y.

[5] D. Singhal, S. Tripathy, and S. K. Jena, ‘Remanufacturing for the circular economy: Study and evaluation of critical factors’, *Resour. Conserv. Recycl.*, vol. 156, 2020, doi: 10.1016/j.resconrec.2020.104681.

[6] C. Sassanelli, P. Rosa, and S. Terzi, ‘Supporting disassembly processes through simulation tools: A systematic literature review with a focus on printed

circuit boards', *Journal of Manufacturing Systems*, vol. 60, pp. 429–448, 2021, doi: 10.1016/j.jmsy.2021.07.009.

[7] M. Sergio, C. Franciosi, and R. Iannone, 'An approach to Evaluate the Impact of the Introduction of a Disassembly Line in Traditional Manufacturing Systems', *Journal of Industrial Engineering and Management*, vol. 15, no. 2, pp. 215–232, 2022, doi: 10.3926/jiem.3605.

[8] I. Paprocka and B. Skołud, 'A Predictive Approach for Disassembly Line Balancing Problems', *Sensors*, vol. 22, no. 10, 2022, doi: 10.3390/s22103920.

[9] E. Cesur, M. R. Cesur, Y. Kayikci, and S. K. Mangla, 'Optimal number of remanufacturing in a circular economy platform', *Int. J. Logist. Res. Applic.*, vol. 25, no. 4–5, pp. 454–470, 2022, doi: 10.1080/13675567.2020.1825656.

[10] H. Geist and F. Balle, 'A circularity engineering focused empirical status quo analysis of automotive remanufacturing processes', *Resour. Conserv. Recycl.*, vol. 201, 2024, doi: 10.1016/j.resconrec.2023.107328.

[11] M. Alfaro-Algaba and F. J. Ramirez, 'Techno-economic and environmental disassembly planning of lithium-ion electric vehicle battery packs for remanufacturing', *Resources, Conservation and Recycling*, vol. 154, p. 104461, Mar. 2020, doi: 10.1016/j.resconrec.2019.104461.

[12] M. Kerin and D. T. Pham, 'Smart remanufacturing: a review and research framework', *JMTM*, vol. 31, no. 6, pp. 1205–1235, May 2020, doi: 10.1108/JMTM-06-2019-0205.

[13] B. Sanchez and C. Haas, ‘Capital project planning for a circular economy’, *Construction Management and Economics*, vol. 36, no. 6, pp. 303–312, 2018, doi: 10.1080/01446193.2018.1435895.

[14] A. K. Gulivindala, M. V. A. R. Bahubalendruni, R. Chandrasekar, E. Ahmed, M. H. Abidi, and A. Al-Ahmari, ‘Automated disassembly sequence prediction for industry 4.0 using enhanced genetic algorithm’, *Computers, Materials and Continua*, vol. 69, no. 2, pp. 2531–2548, 2021, doi: 10.32604/cmc.2021.018014.

[15] H. C. Fang, S. K. Ong, and A. Y. C. Nee, ‘An Integrated Approach for Product Remanufacturing Assessment and Planning’, in *Procedia CIRP*, Seliger G., Kohl H., and Mallon J., Eds., Elsevier B.V., 2016, pp. 262–267. doi: 10.1016/j.procir.2016.01.118.

[16] R. Hofmeester and D. R. Eyers, ‘Strategic opportunities for product-agnostic remanufacturing’, *Int. J. Logist. Manage.*, vol. 34, no. 6, pp. 1601–1628, 2023, doi: 10.1108/IJLM-11-2021-0556.

[17] A. J. D. Lambert, ‘Disassembly aimed at product remanufacturing’, presented at the Proceedings of the ASME Design Engineering Technical Conference, 2001, pp. 255–260. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-1342323883&partnerID=40&md5=9f42ec74f44b30c043b99f62f158056e

[18] M. R. Johnson and M. H. Wang, ‘Economical evaluation of disassembly operations for recycling, remanufacturing and reuse’, *International Journal of*

Production Research, vol. 36, no. 12, pp. 3227–3252, 1998, doi: 10.1080/002075498192049.

[19] S. L. Soh, S. K. Ong, and A. Y. C. Nee, 'Design for disassembly for remanufacturing: Methodology and technology', presented at the Procedia CIRP, Elsevier B.V., 2014, pp. 407–412. doi: 10.1016/j.procir.2014.06.053.

[20] F. J. Ramírez, J. A. Aledo, J. A. Gamez, and D. T. Pham, 'Economic modelling of robotic disassembly in end-of-life product recovery for remanufacturing', *Computers and Industrial Engineering*, vol. 142, 2020, doi: 10.1016/j.cie.2020.106339.

[21] A. Priyono, W. Ijomah, and U. Bititci, 'Disassembly for remanufacturing: A systematic literature review, new model development and future research needs', *J. Ind. Eng. Manage.*, vol. 9, no. 4, pp. 899–932, 2016, doi: 10.3926/jiem.2053.

[22] S. Lou, R. Tan, Y. Zhang, and C. Lv, 'Human-robot interactive disassembly planning in Industry 5.0', presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 891–895. doi: 10.1109/AIM46323.2023.10196250.

[23] A. Sharma *et al.*, 'Reshaping Industry: Adoption of Sustainable Techniques providing Remanufacturing Solutions in High-Tech industries', in *E3S Web Conf.*, Saxena K.K., Ed., EDP Sciences, 2023. doi: 10.1051/e3sconf/202345301028.

[24] J. Yao, S. Zhu, and P. Cui, ‘Advanced information-based remanufacturing technology and its application’, in *Proc. - Int. Conf. Digit. Manuf. Autom., ICDMA*, 2010, pp. 164–167. doi: 10.1109/ICDMA.2010.230.

[25] Z. Zheng, W. Xu, Z. Zhou, D. T. Pham, Y. Qu, and J. Zhou, ‘Dynamic modeling of manufacturing capability for robotic disassembly in remanufacturing’, *Procedia Manufacturing*, vol. 10, pp. 15–25, 2017.

[26] H. Poschmann, H. Brüggemann, and D. Goldmann, ‘Disassembly 4.0: A Review on Using Robotics in Disassembly Tasks as a Way of Automation’, *Chemie Ingenieur Technik*, vol. 92, no. 4, pp. 341–359, Apr. 2020, doi: 10.1002/cite.201900107.

[27] W. H. Chen, G. Foo, S. Kara, and M. Pagnucco, ‘Application of a multi-head tool for robotic disassembly’, in *Procedia CIRP*, Brissaud D., Zwolinski P., Paris H., and Riel A., Eds., Elsevier B.V., 2020, pp. 630–635. doi: 10.1016/j.procir.2020.02.047.

[28] J. Li, M. Barwood, and S. Rahimifard, ‘A multi-criteria assessment of robotic disassembly to support recycling and recovery’, *Resources, Conservation and Recycling*, vol. 140, pp. 158–165, Jan. 2019, doi: 10.1016/j.resconrec.2018.09.019.

[29] P. Dario and M. Rucci, ‘Approach to disassembly problems in robotics’, presented at the 1993 International Conference on Intelligent Robots and Systems, Publ by IEEE, 1993, pp. 460–467. [Online]. Available:

https://www.scopus.com/inward/record.uri?eid=2-s2.0-0027809701&partnerID=40&md5=0264b85f8306b20925098be764125727

[30] J. Liu, Q. Liu, Z. Zhou, D. T. Pham, W. Xu, and Y. Fang, 'Collaborative Optimisation of Robotic Disassembly Planning Problems using the Bees Algorithm', in *Springer Series in Advanced Manufacturing*, 2023, pp. 305–335. doi: 10.1007/978-3-031-14537-7_18.

[31] B. Chen, W. Xu, J. Liu, Z. Ji, and Z. Zhou, 'Robotic Disassembly Sequence Planning Considering Robotic Collision Avoidance Trajectory in Remanufacturing', presented at the IEEE International Conference on Industrial Informatics (INDIN), Institute of Electrical and Electronics Engineers Inc., 2020, pp. 494–501. doi: 10.1109/INDIN45582.2020.9442129.

[32] M. Choux, E. Marti Bigorra, and I. Tyapin, 'Task Planner for Robotic Disassembly of Electric Vehicle Battery Pack', *Metals*, vol. 11, no. 3, p. 387, Feb. 2021, doi: 10.3390/met11030387.

[33] J. Liu, Z. Zhou, D. T. Pham, W. Xu, C. Ji, and Q. Liu, 'Collaborative optimization of robotic disassembly sequence planning and robotic disassembly line balancing problem using improved discrete Bees algorithm in remanufacturing', *Robotics and Computer-Integrated Manufacturing*, vol. 61, 2020, doi: 10.1016/j.rcim.2019.101829.

[34] O. Cooke, H. Al-Musaibeli, and R. Ahmad, 'A Feature Extraction Algorithm for Hybrid Manufacturing and Its Application in Robot-Based Additive and Subtractive Processes', in *Int. Conf. Robot. Autom. Ind., ICRAI,*

Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICRAI57502.2023.10089593.

[35] W. Tsimba, G. P. Chirinda, and S. Matope, ‘Machine learning for decision-making in the remanufacturing of worn-out gears and bearings’, *South Afr. J. Ind. Eng.*, vol. 32, no. 3, pp. 135–150, 2021, doi: 10.7166/32-3-2636.

[36] A. C. Sanderson and L. H. De Mello, ‘Planning assembly/disassembly operations for space telerobotics’, presented at the Proceedings of SPIE - The International Society for Optical Engineering, 1987, pp. 109–115. doi: 10.1117/12.942895.

[37] P. Gu and X. Yan, ‘CAD-directed automatic assembly sequence planning’, *International Journal of Production Research*, vol. 33, no. 11, pp. 3069–3100, 1995, doi: 10.1080/00207549508904862.

[38] C. Mascle, ‘Features modeling in assembly sequence and resource planning’, presented at the Proceedings of the IEEE International Symposium on Assembly and Task Planning, 1995, pp. 232–237. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-0029193675&partnerID=40&md5=06c1b13631b50e26c4b2150e408b9459

[39] A. T. Sere, L. Laperriere, and C. Mascle, ‘Automatic identification of available grasping and fixturing surfaces in assembly resources planning’, presented at the IEEE Symposium on Emerging Technologies & Factory Automation, IEEE, 1995, pp. 69–77. [Online]. Available:

https://www.scopus.com/inward/record.uri?eid=2-s2.0-0029505614&partnerID=40&md5=e0b4d9e685a5a393e14ab6fa1065aaf2

[40] G. Dini, F. Failli, and M. Santochi, ‘A disassembly planning software system for the optimization of recycling processes’, *Production Planning and Control*, vol. 12, no. 1, pp. 2–12, 2001, doi: 10.1080/09537280150203924.

[41] A. GÜngÖr and S. M. Gupta, ‘Disassembly sequence plan generation using a branch-and-bound algorithm’, *International Journal of Production Research*, vol. 39, no. 3, pp. 481–509, Jan. 2001, doi: 10.1080/00207540010002838.

[42] N. Uchiyama and S. Takagi, ‘A Study on Assembly/Disassembly Planning: Extraction of Geometrically Separable Subassemblies from Geometric Models of 2-D Assemblies’, *Nihon Kikai Gakkai Ronbunshu, C Hen/Transactions of the Japan Society of Mechanical Engineers, Part C*, vol. 67, no. 658, pp. 2077–2082, 2001, doi: 10.1299/kikaic.67.2077.

[43] X. Pingjun, Y. Yingxue, L. Jiangsheng, and L. Jianguang, ‘Optimising assembly planning based on virtual reality and bionic algorithm’, *International Journal of Manufacturing Technology and Management*, vol. 9, no. 3–4, pp. 265–293, 2006, doi: 10.1504/IJMTM.2006.010058.

[44] H. Wang, D. Xiang, and G. Duan, ‘An intelligent product disassembly planning method’, presented at the IEEE International Symposium on Electronics and the Environment, 2006, p. 357. doi: 10.1109/ISEE.2006.1650092.

[45] G. Zhang, K. Deng, Q. Hou, and S. Lai, ‘Research on assembly modeling and sequence planning based on features’, presented at the IET Conference Publications, 2006, pp. 408–411. doi: 10.1049/cp:20060795.

[46] L. P. Berg, S. Behdad, J. M. Vance, and D. Thurston, ‘Disassembly Sequence Evaluation: A User Study Leveraging Immersive Computing Technologies’, *Journal of Computing and Information Science in Engineering*, vol. 15, no. 1, 2015, doi: 10.1115/1.4028857.

[47] C.-J. Chen, X.-P. Hu, and N. Li, ‘Research on Virtual Disassembly Operation Using Stability Analysis Method’, presented at the Lecture Notes in Electrical Engineering, Springer Science and Business Media Deutschland GmbH, 2012, pp. 1667–1674. doi: 10.1007/978-1-4419-8849-2_216.

[48] Q. Gao, L. Shang, and L. Zhang, ‘An efficient method of assembly path planning in virtual manufacturing system’, presented at the Advanced Materials Research, 2012, pp. 550–558. doi: 10.4028/www.scientific.net/AMR.468-471.550.

[49] Y. Z. Yu, ‘Research on disassembly sequence planning in DFD’, presented at the Advanced Materials Research, 2012, pp. 1865–1869. doi: 10.4028/www.scientific.net/AMR.516-517.1865.

[50] J. Zhong, ‘Selective disassembly strategy based on the CAD model assembly building’, presented at the Applied Mechanics and Materials, 2012, pp. 1095–1098. doi: 10.4028/www.scientific.net/AMM.193-194.1095.

[51] I. Belhadj, M. Trigui, and A. Benamara, ‘Subassembly identification method based on CAD data’, *Lecture Notes in Mechanical Engineering*, vol. 0, pp. 55–62, 2017, doi: 10.1007/978-3-319-45781-9_6.

[52] C. Friedrich, A. Csiszar, A. Lechler, and A. Verl, ‘Fast robot task and path planning based on CAD and vision data’, presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1633–1638. doi: 10.1109/AIM.2017.8014252.

[53] M. Kheder, M. Trigui, and N. Aifaoui, ‘Optimization of disassembly sequence planning for preventive maintenance’, *International Journal of Advanced Manufacturing Technology*, vol. 90, no. 5–8, pp. 1337–1349, 2017, doi: 10.1007/s00170-016-9434-2.

[54] F. Osti, A. Ceruti, A. Liverani, and G. Caligiana, ‘Semi-automatic Design for Disassembly Strategy Planning: An Augmented Reality Approach’, *Procedia Manufacturing*, vol. 11, pp. 1481–1488, 2017, doi: 10.1016/j.promfg.2017.07.279.

[55] S. Tao and M. Hu, ‘A contact relation analysis approach to assembly sequence planning for assembly models’, *Computer-Aided Design and Applications*, vol. 14, no. 6, pp. 720–733, 2017, doi: 10.1080/16864360.2017.1287674.

[56] W. Zhang, M. Ma, H. Li, and J. Yu, ‘Generating interference matrices for automatic assembly sequence planning’, *International Journal of Advanced*

Manufacturing Technology, vol. 90, no. 1–4, pp. 1187–1201, 2017, doi: 10.1007/s00170-016-9410-x.

[57] M. F. Adesso, R. Hegewald, N. Wolpert, E. Schomer, B. Maier, and B. A. Epple, 'Automatic Classification and Disassembly of Fasteners in Industrial 3D CAD-Scenarios', presented at the Proceedings - IEEE International Conference on Robotics and Automation, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 9874–9880. doi: 10.1109/ICRA46639.2022.9811539.

[58] K. Barbu, J. Beck, P. Schäfer, and A. Neb, 'Development of a Visual Assembly Planning System based on Neutral Files', presented at the Procedia CIRP, Elsevier B.V., 2022, pp. 446–451. doi: 10.1016/j.procir.2022.05.006.

[59] S. Dorn, N. Wolpert, and E. Schomer, 'An Assembly Sequence Planning Framework for Complex Data using General Voronoi Diagram', presented at the Proceedings - IEEE International Conference on Robotics and Automation, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 9896–9902. doi: 10.1109/ICRA46639.2022.9811867.

[60] S. Münker and R. H. Schmitt, 'CAD-based AND/OR Graph Generation Algorithms in (Dis)assembly Sequence Planning of Complex Products', presented at the Procedia CIRP, Elsevier B.V., 2022, pp. 144–149. doi: 10.1016/j.procir.2022.02.169.

[61] V. S. S. V. Prasad, M. Hymavathi, C. S. P. Rao, and M. A. R. Bahubalendruni, 'A novel computative strategic planning projections algorithm (CSPPA) to generate oblique directional interference matrix for different

applications in computer-aided design', *Computers in Industry*, vol. 141, 2022, doi: 10.1016/j.compind.2022.103703.

[62] J. P. J. Prioli, H. M. Alrufaifi, and J. L. Rickli, 'Disassembly assessment from CAD-based collision evaluation for sequence planning', *Robotics and Computer-Integrated Manufacturing*, vol. 78, pp. 102416–102416, 2022, doi: 10.1016/j.rcim.2022.102416.

[63] H.-E. Tseng, C.-C. Chang, S.-C. Lee, and C.-C. Chen, 'Connector-link-part-based disassembly sequence planning', *Concurrent Engineering Research and Applications*, vol. 30, no. 1, pp. 67–79, 2022, doi: 10.1177/1063293X211050930.

[64] A. Upadhyay, B. Ladrecha, A. Dubey, S. M. Kuriakose, and P. Goenka, '3D-DSPnet: Product Disassembly Sequence Planning', presented at the ICMEW 2022 - IEEE International Conference on Multimedia and Expo Workshops 2022, Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICMEW56448.2022.9859434.

[65] A. J. D. Lambert, 'Exact methods in optimum disassembly sequence search for problems subject to sequence dependent costs', *Omega*, vol. 34, no. 6, pp. 538–549, 2006, doi: 10.1016/j.omega.2005.01.005.

[66] A. J. D. Lambert, 'Exact method for disassembly sequence optimization subjected to sequence dependent costs', *International Journal of Operations and Quantitative Management*, vol. 11, no. 2, pp. 75–89, 2005, [Online]. Available:

https://www.scopus.com/inward/record.uri?eid=2-s2.0-33644623166&partnerID=40&md5=b6f46c23913e6eb0c4028a7043bc4e21

[67] C. Blum and A. Roli, 'Metaheuristics in combinatorial optimization: Overview and conceptual comparison', *ACM Comput. Surv.*, vol. 35, no. 3, pp. 268–308, Sep. 2003, doi: 10.1145/937503.937505.

[68] Y. Ren *et al.*, 'An efficient metaheuristics for a sequence-dependent disassembly planning', *Journal of Cleaner Production*, vol. 245, 2020, doi: 10.1016/j.jclepro.2019.118644.

[69] L. Zhong, S. Youchao, O. E. Gabriel, and W. Haiqiao, 'Disassembly sequence planning for maintenance based on metaheuristic method', *Aircraft Engineering and Aerospace Technology*, vol. 83, no. 3, pp. 138–145, 2011, doi: 10.1108/00022661111131221.

[70] E.-G. Talbi, 'Machine learning into metaheuristics: A survey and taxonomy of data-driven metaheuristics', *ACM Comput. Surv.*, vol. 00, no. 00, 2021.

[71] K. Yokota and D. R. Brough, 'Assembly/Disassembly Sequence Planning', *Assembly Automation*, vol. 12, no. 3, pp. 31–38, 1992, doi: 10.1108/eb004372.

[72] B. O'Shea, S. S. Grewal, and H. Kaebernick, 'State of the art literature survey on disassembly planning', *Concurrent Engineering Research and Applications*, vol. 6, no. 4, pp. 345–357, 1998, doi: 10.1177/1063293X9800600407.

[73] D.-H. Lee, J.-G. Rang, and P. Xirouchakis, ‘Disassembly planning and scheduling: Review and further research’, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 215, no. 5, pp. 695–709, 2001, doi: 10.1243/0954405011518629.

[74] J. Dong and G. Arndt, ‘A review of current research on disassembly sequence generation and computer aided design for disassembly’, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 217, no. 3, pp. 299–312, 2003, doi: 10.1243/095440503321590479.

[75] K. E. Moore, A. Gungor, and S. M. Gupta, ‘A Petri net approach to disassembly process planning’, *Computers & Industrial Engineering*, vol. 35, no. 1–2, pp. 165–168, Oct. 1998, doi: 10.1016/S0360-8352(98)00051-5.

[76] K. E. Moore, A. Gungor, and S. M. Gupta, ‘Disassembly Petri net generation in the presence of XOR precedence relationships’, presented at the Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, IEEE, 1998, pp. 13–18. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-0032267397&partnerID=40&md5=0a220376dfda2cdb5d76f6097dbdbc52

[77] K. E. Moore, A. Güngör, and S. M. Gupta, ‘Petri net approach to disassembly process planning for products with complex AND/OR precedence relationships’, *European Journal of Operational Research*, vol. 135, no. 2, pp. 428–449, Dec. 2001, doi: 10.1016/S0377-2217(00)00321-0.

[78] S. Ghandi and E. Masehian, ‘Review and taxonomies of assembly and disassembly path planning problems and approaches’, *CAD Computer Aided Design*, vol. 67–68, pp. 58–86, 2015, doi: 10.1016/j.cad.2015.05.001.

[79] Z. Zhou *et al.*, ‘Disassembly sequence planning: Recent developments and future trends’, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 233, no. 5, pp. 1450–1471, 2019, doi: 10.1177/0954405418789975.

[80] X. Guo, M. Zhou, A. Abusorrah, F. Alsokhiry, and K. Sedraoui, ‘Disassembly Sequence Planning: A Survey’, *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 7, pp. 1308–1324, 2021, doi: 10.1109/JAS.2020.1003515.

[81] S. K. Ong, M. M. L. Chang, and A. Y. C. Nee, ‘Product disassembly sequence planning: state-of-the-art, challenges, opportunities and future directions’, *International Journal of Production Research*, vol. 59, no. 11, pp. 3493–3508, 2021, doi: 10.1080/00207543.2020.1868598.

[82] M. Chand and C. Ravi, ‘A state-of-the-art literature survey on artificial intelligence techniques for disassembly sequence planning’, *CIRP Journal of Manufacturing Science and Technology*, vol. 41, pp. 292–310, 2023, doi: 10.1016/j.cirpj.2022.11.017.

[83] J. Ji and Y. Wang, ‘Selective disassembly sequence optimization based on the improved immune algorithm’, *Robotic Intelligence and Automation*, vol. 43, no. 2, pp. 96–108, 2023, doi: 10.1108/RIA-06-2022-0156.

[84] S. Münker *et al.*, ‘CAD-Based Product Partitioning for Automated Disassembly Sequence Planning with Community Detection’, presented at the Lecture Notes in Mechanical Engineering, Springer Science and Business Media Deutschland GmbH, 2023, pp. 570–577. doi: 10.1007/978-3-031-34821-1_62.

[85] A. Rehal and D. Sen, ‘An Efficient Disassembly Sequencing Scheme Using the Shell Structure’, *CAD Computer Aided Design*, vol. 154, 2023, doi: 10.1016/j.cad.2022.103423.

[86] G. B. Murali and G. S. Mahapatra, ‘Selective Disassembly Sequence Generation Using Secondary Part Information Matrix’, presented at the Smart Innovation, Systems and Technologies, Springer Science and Business Media Deutschland GmbH, 2023, pp. 659–669. doi: 10.1007/978-981-99-0264-4_55.

[87] C. J. Chen, J. Hong, and S. F. Wang, ‘Automated positioning of 3D virtual scene in AR-based assembly and disassembly guiding system’, *Int J Adv Manuf Technol*, vol. 76, no. 5–8, pp. 753–764, Feb. 2015, doi: 10.1007/s00170-014-6321-6.

[88] B. Adenso-Díaz, S. García-Carbajal, and S. Lozano, ‘An efficient GRASP algorithm for disassembly sequence planning’, *OR Spectrum*, vol. 29, no. 3, pp. 535–549, 2007, doi: 10.1007/s00291-005-0028-x.

[89] B. Adenso-Díaz, S. García-Carbajal, and S. M. Gupta, ‘A path-relinking approach for a bi-criteria disassembly sequencing problem’, *Computers and Operations Research*, vol. 35, no. 12, pp. 3989–3997, 2008, doi: 10.1016/j.cor.2007.06.002.

[90] P. Wu, H. Wang, B. Li, W. Fu, J. Ren, and Q. He, ‘Disassembly sequence planning and application using simplified discrete gravitational search algorithm for equipment maintenance in hydropower station’, *Expert Systems with Applications*, vol. 208, 2022, doi: 10.1016/j.eswa.2022.118046.

[91] J. Liu, Z. Zhou, D. T. Pham, W. Xu, C. Ji, and Q. Liu, ‘Robotic disassembly sequence planning using enhanced discrete bees algorithm in remanufacturing’, *International Journal of Production Research*, vol. 56, no. 9, pp. 3134–3151, May 2018, doi: 10.1080/00207543.2017.1412527.

[92] Asrul Harun Ismail, ‘Enhancing the Bees Algorithm using the Traplining Metaphor’, Doctor of Philosophy, University of Birmingham, 2021. [Online]. Available: https://etheses.bham.ac.uk/id/eprint/12128/

[93] G. Seliger, T. Keil, U. Rebafka, and A. Stenzel, ‘Flexible disassembly tools’, in *Proceedings of the 2001 IEEE International Symposium on Electronics and the Environment. 2001 IEEE ISEE (Cat. No.01CH37190)*, Denver, CO, USA: IEEE, 2001, pp. 30–35. doi: 10.1109/ISEE.2001.924498.

[94] C. Favi, M. Marconi, M. Germani, and M. Mandolini, ‘A design for disassembly tool oriented to mechatronic product de-manufacturing and recycling’, *Advanced Engineering Informatics*, vol. 39, pp. 62–79, 2019, doi: 10.1016/j.aei.2018.11.008.

[95] S. Smith, L.-Y. Hsu, and G. C. Smith, ‘Partial disassembly sequence planning based on cost-benefit analysis’, *J. Clean. Prod.*, vol. 139, pp. 729–739, 2016, doi: 10.1016/j.jclepro.2016.08.095.

[96] Y. M. Huang and C.-T. Huang, ‘Disassembly matrix for disassembly processes of products’, *International Journal of Production Research*, vol. 40, no. 2, pp. 255–273, Jan. 2002, doi: 10.1080/00207540110079770.

[97] Y. Wang *et al.*, ‘Interlocking problems in disassembly sequence planning’, *International Journal of Production Research*, vol. 59, no. 15, pp. 4723–4735, 2021, doi: 10.1080/00207543.2020.1770892.

[98] S. Smith, G. Smith, and W.-H. Chen, ‘Disassembly sequence structure graphs: An optimal approach for multiple-target selective disassembly sequence planning’, *Adv. Eng. Inf.*, vol. 26, no. 2, pp. 306–316, 2012, doi: 10.1016/j.aei.2011.11.003.

[99] H.-E. Tseng, C.-C. Chang, S.-C. Lee, and Y.-M. Huang, ‘Hybrid bidirectional ant colony optimization (hybrid BACO): An algorithm for disassembly sequence planning’, *Engineering Applications of Artificial Intelligence*, vol. 83, pp. 45–56, 2019, doi: 10.1016/j.engappai.2019.04.015.

[100] E. Kongar and S. M. Gupta, ‘Disassembly sequencing using genetic algorithm’, *The International Journal of Advanced Manufacturing Technology*, vol. 30, pp. 497–506, 2006.

[101] G. Q. Jin, W. D. Li, and K. Xia, ‘Disassembly matrix for liquid crystal displays televisions’, presented at the Procedia CIRP, Elsevier B.V., 2013, pp. 357–362. doi: 10.1016/j.procir.2013.07.015.

[102] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, ‘Bee algorithm a novel approach to function optimisation’, *Technical Note: MEC*, vol. 501, 2005.

[103] D. T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri, S. Rahim, and M. Zaidi, ‘The bees algorithm—a novel tool for complex optimisation problems’, in *Intelligent production machines and systems*, Elsevier, 2006, pp. 454–459.

[104] A. Farsi, S. A. Torabi, and M. Mokhtarzadeh, ‘Integrated surgery scheduling by constraint programming and meta-heuristics’, *International Journal of Management Science and Engineering Management*, pp. 1–13, 2022, doi: 10.1080/17509653.2022.2093289.

[105] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, ‘Performance Analysis of Private Blockchain Platforms in Varying Workloads’, in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, Aug. 2017, pp. 1–6. doi: 10.1109/ICCCN.2017.8038517.

[106] P. K. Lehre and X. Qin, ‘More Precise Runtime Analyses of Non-elitist Evolutionary Algorithms in Uncertain Environments’, *Algorithmica*, Oct. 2022, doi: 10.1007/s00453-022-01044-5.

[107] P. K. Lehre and X. Qin, ‘Self-Adaptation via Multi-Objectivisation: A Theoretical Study’, in *Proceedings of the Genetic and Evolutionary Computation Conference*, in GECCO ’22. New York, NY, USA: Association for Computing Machinery, 2022, pp. 1417–1425. doi: 10.1145/3512290.3528836.

[108] X. Qin and P. K. Lehre, ‘Self-adaptation via multi-objectivisation: an empirical study’, in *International Conference on Parallel Problem Solving from Nature*, Springer, 2022, pp. 308–323.

Appendix A 1

Table A.1 DSP review papers published since 1987.

Title	Year	Author	Publication
Assembly/disassembly sequence planning	1992	Yokota and Brough	Assembly Automation
State of the art literature survey on disassembly planning	1998	O'shea et al.	Concurrent Engineering Research and Applications
Disassembly planning and scheduling: Review and further research	2001	Lee et al.	Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture
A review of current research on disassembly sequence generation and computer aided design for disassembly	2003	Dong and Arndt	Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture
Disassembly sequencing for maintenance: A survey	2006	Kang and Xirouchakis	Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture
Review and taxonomies of assembly and disassembly path planning problems and approaches	2015	Ghandi and Masehian	CAD Computer Aided Design
Disassembly sequence planning: Recent developments and future trends	2019	Zhou et al.	Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture
Disassembly sequence planning: A survey	2021	Guo et al.	IEEE/CAA Journal of Automatica Sinica

Product disassembly sequence planning: state-of-the-art, challenges, opportunities and future directions	2021	Ong et al.	International Journal of Production Research
Supporting disassembly processes through simulation tools: A systematic literature review with a focus on printed circuit boards	2021	Sassanelli et al.	Journal of Manufacturing Systems
A state-of-the-art literature survey on artificial intelligence techniques for disassembly sequence planning	2023	Chand and Ravi	CIRP Journal of Manufacturing Science and Technology
Integrating X-reality ad lean into end-of-life aircraft parts disassembly sequence planning: a critical review and research	2023	Yang et al.	International Journal of Advanced Manufacturing Technology

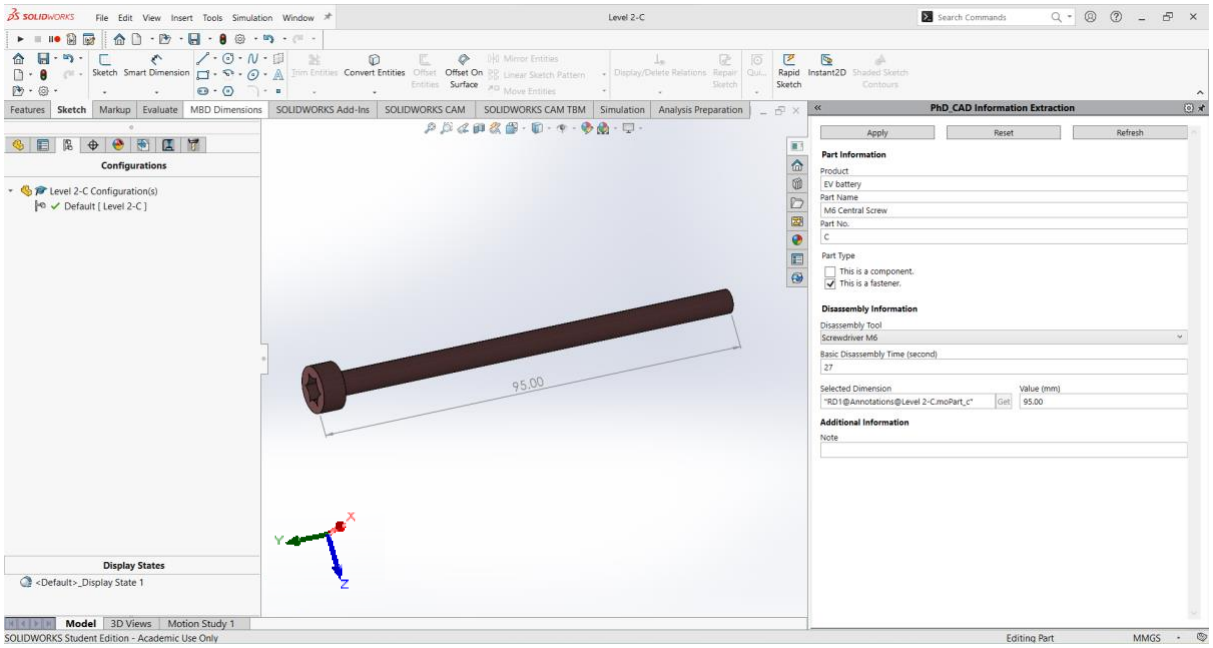


Figure A.1 An overall view of the add-in tool's part page: a fastener.

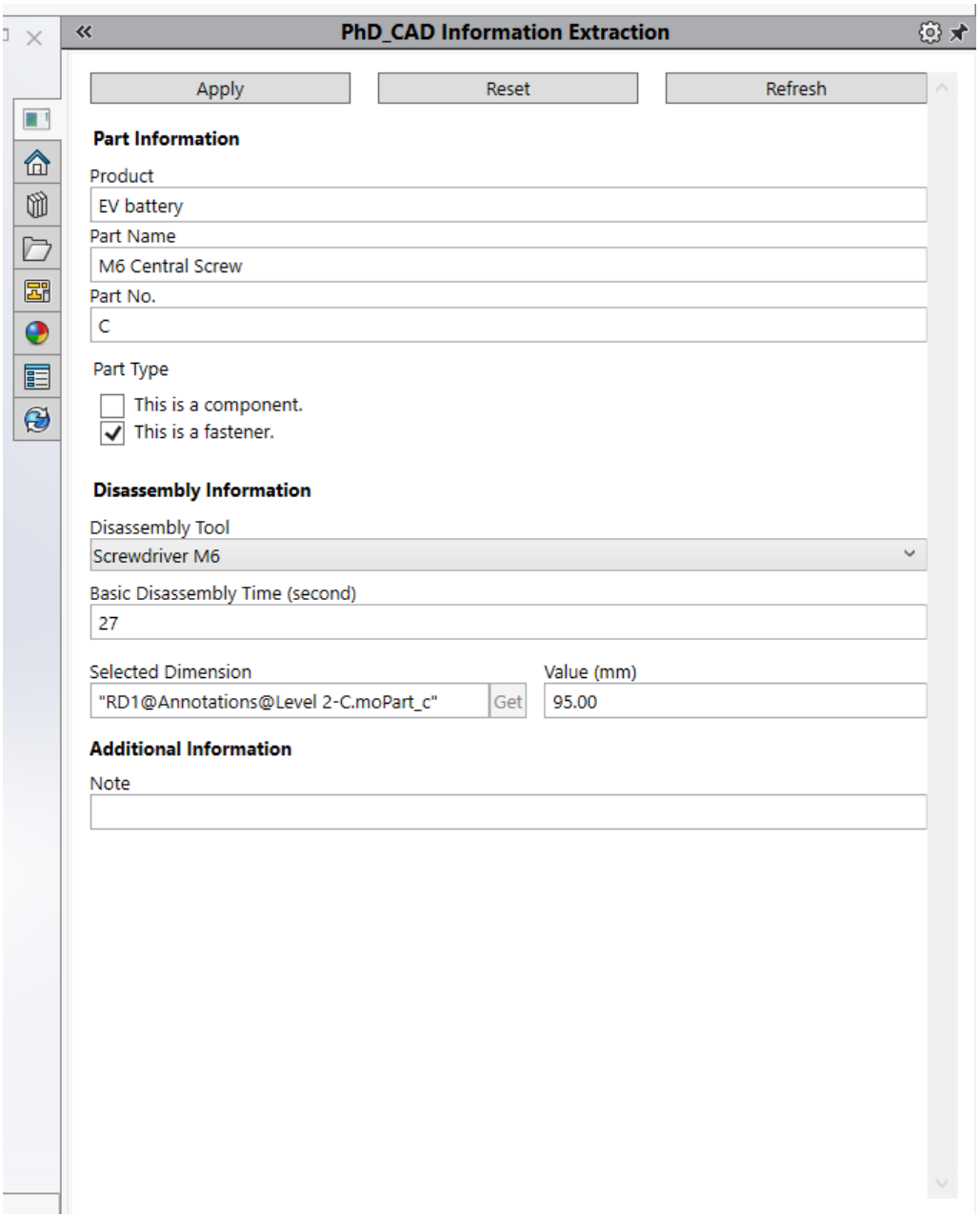


Figure A.2 A zoom-in view of the add-in tool's part page: a fastener.

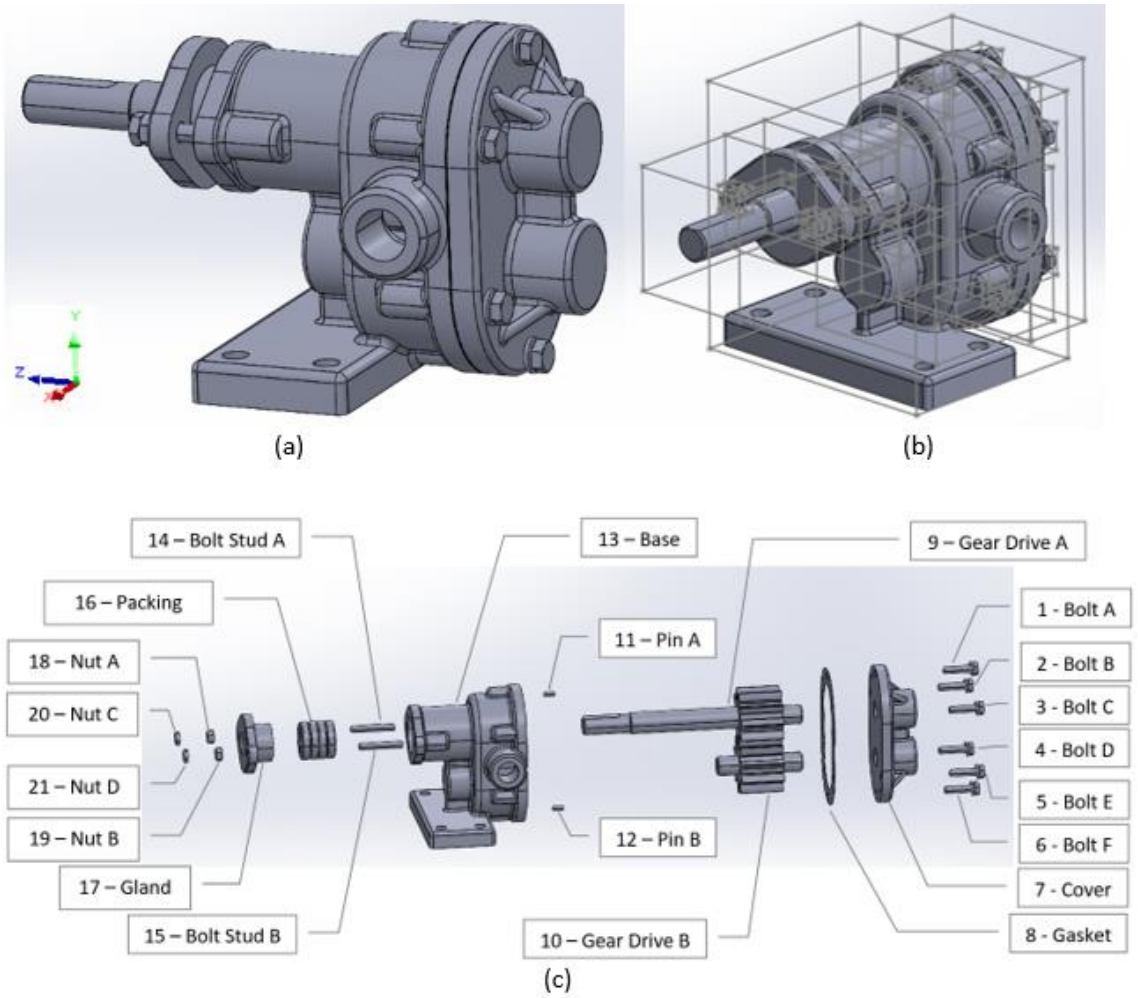


Figure A.3 Gear Pump: (a) CAD model; (b) bounding box; (c) exploded view

Table A.2 The extracted bounding box coordinate values of the gear pump.

No.	Part Name	The extracted coordinate values of bounding box vertices (mm)					
		x_{min}	y_{min}	z_{min}	x_{max}	y_{max}	z_{max}
1	Bolt A	-6.05	46.94	-17.51	6.05	59.05	11.49
2	Bolt B	-37.27	21.94	-17.51	-25.17	34.05	11.49
3	Bolt C	25.17	21.94	-17.51	37.27	34.05	11.49
4	Bolt D	-37.27	-34.05	-17.51	-25.17	-21.94	11.49
5	Bolt E	25.17	-34.05	-17.51	37.27	-21.94	11.49
6	Bolt F	-6.05	-59.05	-17.51	6.05	-46.94	11.49
7	Cover	-38.00	-59.00	-33.50	38.00	59.00	-0.50
8	Gasket	-38.00	-59.00	-0.50	38.00	59.00	0.00
9	Gear Drive A	-28.04	-6.95	-18.00	28.04	48.95	167.00
10	Gear Drive B	-34.52	-55.46	-18.00	34.52	13.46	54.00
11	Pin A	-25.23	41.00	-3.00	-21.23	45.00	7.00
12	Pin B	21.23	-45.00	-3.00	25.23	-41.00	7.00
13	Base	-60.00	-61.00	-2.50	60.00	59.00	93.00
14	Bolt Stud A	-27.00	18.00	83.00	-21.00	24.00	120.00
15	Bolt Stud B	21.00	18.00	83.00	27.00	24.00	120.00
16	Packing	-16.00	5.00	58.00	16.00	37.00	86.00
17	Gland	-34.00	-4.00	86.00	34.00	46.00	110.00
18	Nut A	-29.77	15.21	109.99	-18.22	26.78	115.01
19	Nut B	18.22	15.21	109.99	29.77	26.78	115.01
20	Nut C	-29.78	15.22	114.98	-18.21	26.77	118.61
21	Nut D	18.21	15.22	114.98	29.78	26.77	118.61

Table A.3 The calculated bounding box dimensions of the gear pump.

No.	Part Name	The calculated bounding box dimensions (mm)		
		$\|B_X\|$	$\|B_Y\|$	$\|B_z\|$
1	Bolt A	12.10	12.11	29.00
2	Bolt B	12.10	12.11	29.00
3	Bolt C	12.10	12.11	29.00
4	Bolt D	12.10	12.11	29.00
5	Bolt E	12.10	12.11	29.00
6	Bolt F	12.10	12.11	29.00
7	Cover	76.00	118.00	33.00
8	Gasket	76.00	118.00	0.50
9	Gear Drive A	56.08	55.90	185.00
10	Gear Drive B	69.04	68.92	72.00
11	Pin A	4.00	4.00	10.00
12	Pin B	4.00	4.00	10.00
13	Base	120.00	120.00	95.50
14	Bolt Stud A	6.00	6.00	37.00
15	Bolt Stud B	6.00	6.00	37.00
16	Packing	32.00	32.00	28.00
17	Gland	68.00	50.00	24.00
18	Nut A	11.55	11.57	5.02
19	Nut B	11.55	11.57	5.02
20	Nut C	11.57	11.55	3.63
21	Nut D	11.57	11.55	3.63

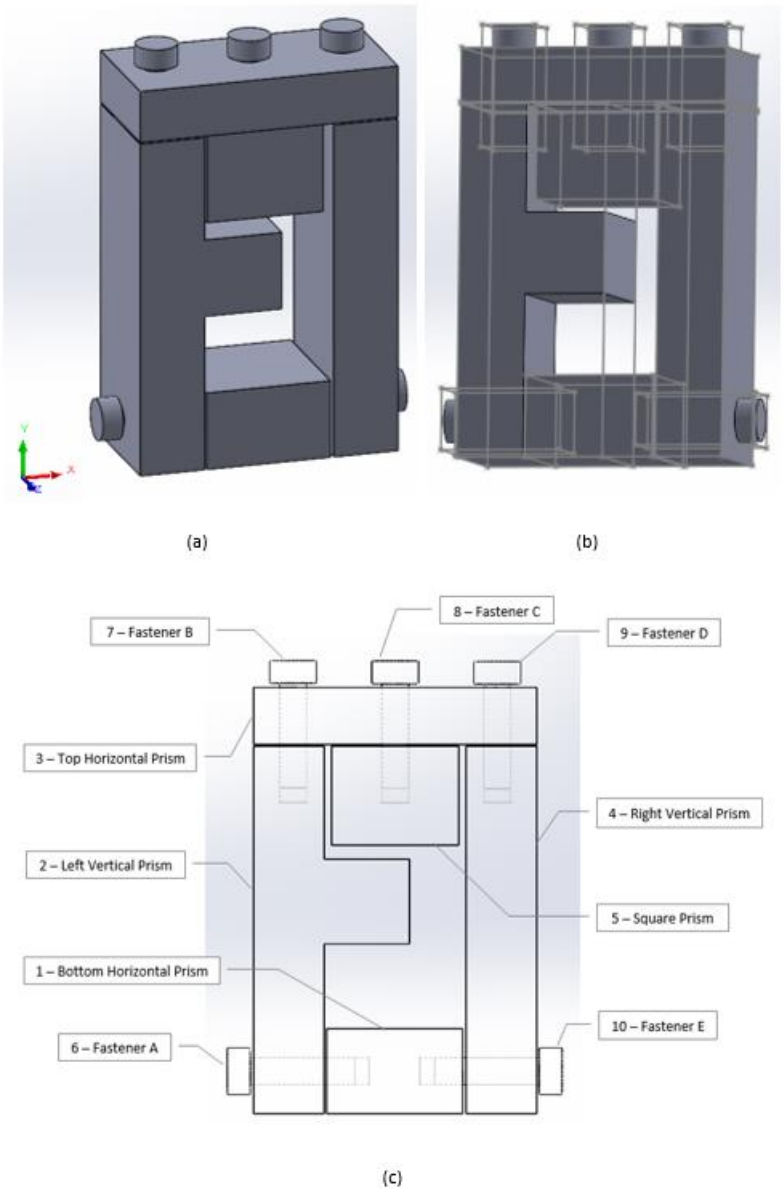


Figure A.4 A ten-parts case study: (a) CAD model; (b) bounding box; (c) exploded view

Table A.4 The extracted bounding box coordinate values of the ten-parts case study.

No.	Part Name	The extracted coordinate values of bounding box vertices (mm)					
		x_{min}	y_{min}	z_{min}	x_{max}	y_{max}	z_{max}
1	Bottom Horizontal Prism	-24.00	-15.00	0.00	24.00	15.00	50.00
2	Left Vertical Prism	-49.74	-15.00	0.00	5.25	115.00	50.00
3	Top Horizontal Prism	-49.74	115.61	1.73	50.25	135.61	50.00
4	Right Vertical Prism	25.25	-15.00	1.38	50.25	115.00	50.00
5	Square Prism	-22.24	79.89	5.00	22.75	114.89	45.00
6	Fastener A	-58.94	-11.28	13.71	-13.94	11.28	36.28
7	Fastener B	-45.61	100.15	15.13	-25.88	145.15	34.86
8	Fastener C	-10.36	99.96	14.38	10.86	144.96	35.61
9	Fastener D	25.22	99.84	13.97	47.27	144.84	36.02
10	Fastener E	14.09	-9.60	15.39	59.09	9.60	34.60

Table A.5 The calculated bounding box dimensions of the ten-parts case study.

No.	Part Name	The calculated bounding box dimensions (mm)		
		$\|B_X\|$	$\|B_Y\|$	$\|B_z\|$
1	Bottom Horizontal Prism	48.00	30.00	50.00
2	Left Vertical Prism	54.99	130.00	50.00
3	Top Horizontal Prism	99.99	20.00	48.27
4	Right Vertical Prism	25.00	130.00	48.62
5	Square Prism	44.99	35.00	40.00
6	Fastener A	45.00	22.56	22.57
7	Fastener B	19.73	45.00	19.73
8	Fastener C	21.22	45.00	21.23
9	Fastener D	22.05	45.00	22.05
10	Fastener E	45.00	19.20	19.21