UNIVERSITY OF BIRMINGHAM

DOCTORAL THESIS

Intelligent Robotic Disassembly Optimisation for Sustainability using the Bees Algorithm

Author: Natalia HARTONO

Supervisor: Professor Duc Truong PHAM

A thesis submitted in fulfilment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in the

Department of Mechanical Engineering
University of Birmingham
UK, B15 2TT

September 3, 2023

Declaration

I, Natalia Hartono, hereby declare that this Ph.D. thesis entitled “Intelligent Robotic Disassembly Optimisation for Sustainability using the Bees Algorithm” was carried out by my own for the degree of Doctor of Philosophy in the University of Birmingham. I confirm that:

- The presented work has never been previously included in a thesis or dissertation submitted for a degree or other qualifications.
- Where the thesis is based on joint works done by myself with others, a clear statement has been made to illustrate how the contribution was exactly distributed.
- Except where stated otherwise by reference or acknowledgement, the work presented is entirely composed by myself.

Signed Natalia Hartono

Date 3 September 2023

This thesis is dedicated to God Almighty, Lord Jesus Christ, for His miraculous guidance and grace throughout this journey and all my life. Despite setbacks such as failed scholarship applications, paper rejections, and the challenges of the COVID-19 pandemic and lockdowns, His divine presence and support have enabled me to persevere and reach this milestone.

To Tonny Lucky Fung, my husband, and Graciella Adeline Lucky Fung, my daughter, thank you for your unwavering support. To my late father, Budhi Hartono Tanzil, who unfortunately never got to witness my achievements, I will always cherish the memories and the encouragement you gave me to pursue higher education. To my mother, Tjhoa Bun Nio, and my sisters, Pauline, Joanna, and Cindy, thank you for always being there for me.

In a world where doubt held sway,

A little girl dreamed big each day.

"People laughed and said it's vain,

But God whispered, 'Believe in me,' again.

Her father encouraged her, "Work hard and keep the faith,

And one day you'll reach the stars," he said.

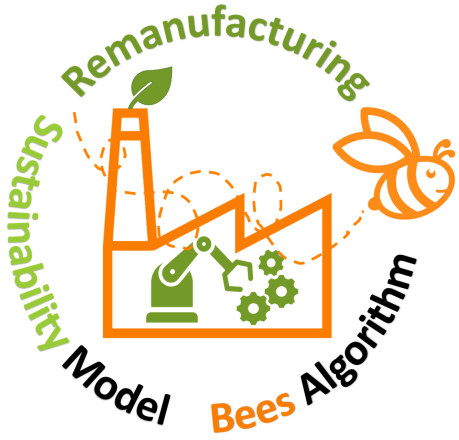
Through doubts and fears, she persevered,

And when the time was right, she shed joyful tears.

It wasn't the end, but the start of a new page,

A reminder that with faith and hope, anything's possible at any age.

"Natalia Hartono - 2023"



Abstract

Robotic disassembly plays a pivotal role in achieving efficient and sustainable product lifecycle management, with a focus on resource conservation and waste reduction. This thesis discusses robotic disassembly sequence planning (RDSP) and robotic disassembly line balancing (RDLB), with a specific emphasis on optimising sustainability models. The overarching goal was to enhance the efficiency and effectiveness of disassembly processes through intelligent robotic disassembly optimisation techniques. At the heart of this research lies the application of the Bees Algorithm (BA), a metaheuristic optimisation algorithm inspired by the foraging behaviour of honeybees. By harnessing the power of the BA, this research aims to address the challenges associated with RDSP and RDLB, ultimately facilitating sustainable disassembly practices. The thesis gives an extensive literature review of RDSP and RDLB to gain deeper insight into the current research landscape. The challenges of the RDSP problem were addressed in this work by introducing a sustainability model and various scenarios to enhance disassembly processes. The sustainability model considers three objectives: profit, energy savings, and environmental impact reduction. The four explored scenarios were recovery (REC), remanufacture (REM), reuse (REU), and an automatic recovery scenario (ARS). Two novel tools were developed for assessing algorithm performance: the statistical performance metric (SPM) and the performance evaluation index (PEI). To validate the proposed approach, a case study involving the disassembly of gear pumps was used. To optimise the RDSP, single-objective (SO), multiobjective (MO) aggregate, and multiobjective nondominated (MO-ND) approaches were adopted. Three optimisation algorithms were employed

— Multiobjective Nondominated Bees Algorithm (MOBA), Nondominated Sorting Genetic Algorithm - II (NSGA-II), and Pareto Envelope-based Selection Algorithm - II (PESA-II), and their results were compared using SPM and PEI. The findings indicate that MO-ND is more suitable for this problem, highlighting the importance of considering conflicting objectives in RDSP. It was shown that recycling should be considered the last-resort recovery option, advocating for the exploration of alternative recovery strategies prior to recycling. Moreover, MOBA outperformed other algorithms, demonstrating its effectiveness in achieving a more efficient and sustainable RDSP. The problem of sequence-dependent robotic disassembly line balancing (RDLBSD) was next investigated by considering the interconnection between disassembly sequence planning and line balancing. Both aspects were optimised simultaneously, leading to a balanced and optimal disassembly process considering profitability, energy savings, environmental impact, and line balance using the MO-ND approach. The findings further support the notion that recycling should be considered the last option for recovery. Again, MOBA outperformed other algorithms, showcasing its capability to handle more complex problems. The final part of the thesis explains the mechanism of a new enhanced BA, named the Fibonacci Bees Algorithm (BA_F). BA_F draws inspiration from the Fibonacci sequence observed in the drone ancestry. This adoption of the Fibonacci-sequence-based pattern reduces the number of algorithm parameters to four, streamlining parameter setting and simplifying the algorithm's steps. The study conducted on the RDSP problem demonstrates BA_F's performance over the basic BA, particularly in handling more complex problems. The thesis concludes by summarising the key contributions of the work, including the enhancements made to the BA and the introduction of novel evaluation tools, and the implications of the research, especially the importance of exploring alternative recovery strategies for end-of-life (EoL) products to align with Circular Economy principles.

Acknowledgements

I am profoundly grateful to the numerous individuals and organisations who have provided invaluable support and assistance throughout my enriching PhD journey. Their contributions have been instrumental in shaping my academic and personal growth. First and foremost, I extend my sincerest appreciation to my exceptional supervisor, Professor Duc Truong Pham OBE, FREng, FLSW, FSME, BE, PhD, DEng, CEng, FIET, FIMechE, for his unwavering patience, boundless encouragement, and expert guidance throughout my research. His belief in my abilities has been a constant source of motivation, and I am truly indebted to him for the countless opportunities he provided me with. I would also like to express my heartfelt thanks to Professor F. Javier Ramírez for his continuous support from the beginning of my research journey. I am deeply thankful to Professor Stefan Dimov, who chaired the viva, and to Professor Abir Jaafar Hussain and Dr. Mozafar Saadat, the examiners, for their rigorous evaluation, insightful feedback and invaluable comments.

Furthermore, I wish to extend my gratitude to the Indonesian Endowment Fund for Education (LPDP) for generously sponsoring my study and to the University of Pelita Harapan and LLDIKTI 3 for granting me the study permit. This research was made possible by the Autonomous Remanufacturing (AUTOREMAN) Project, supported by the UK Engineering and Physical Sciences Research Council (EPSRC), UK, grant no. EP/N018524/1. I am also thankful for the availability of BEAR cloud research computing provided by the University of Birmingham, which facilitated the experiments for this research.

I would like to express my gratitude to the AutoReman group, the Bees Algorithm group, and the Shut-Up-and-Work Network at the University of Birmingham for creating an enjoyable and supportive environment throughout my PhD journey. I express my deep appreciation to Professor Nyoman Pujawan, Dr. Laurence, and Christina, M.T., for their invaluable recommendation letters, which were instrumental in my scholarship and admissions processes.

Finally, I express my sincere appreciation to everyone who has helped me in any way, whether financially, emotionally, or spiritually, especially to my family and friends. Thank you all for your unwavering support, encouragement, and friendship. My journey would not have been the same without each and every one of you.

List of Publications

The findings of this research have been disseminated through various channels, including presentations at workshops and conferences as well as publications. The research has also received awards in recognition of its contributions. The following list presents these contributions in chronological order, categorising them based on published papers, ongoing submissions, and presentations, including the awards received.

Publications:

- N. Hartono, F. J. Ramírez, D.T. Pham. 2022. Optimisation of Robotic Disassembly Plans using the Bees Algorithm. Robotics and Computer-Integrated Manufacturing 78. [1]
- N. Hartono, F. J. Ramírez, D.T. Pham. 2022. A Sustainability-based model for Robotic Disassembly Sequence Planning in Remanufacturing using the Bees Algorithm. IFAC-PapersOnLine 55 [2]
- N. Hartono, F. J. Ramírez, D.T. Pham. 2023. Optimisation of Robotic Disassembly Sequence Plans for Sustainability using the multi-objective Bees Algorithm. In: Intelligent Production and Manufacturing Optimisation—The Bees Algorithm Approach, Springer Series in Advanced Manufacturing. Springer. [3]
- N. Hartono, F. J. Ramírez, D.T. Pham. 2023. A Multiobjective Decision-Making Approach for Modelling and Planning Economically and Environmentally Sustainable Robotic Disassembly for Remanufacturing. Computers and Industrial Engineering [4]

- N. Hartono and D.T. Pham. Robotic Disassembly Sequence Planning and Line Balancing - Research Trends Review and Bibliometric Analysis. Lecture Notes in Mechanical Engineering, Springer. (In Press.)

Manuscripts under review:

- N. Hartono, F. J. Ramírez, D.T. Pham. Using the Bees Algorithm to optimise Robotic Disassembly Sequences and Balance Disassembly Lines for Sustainable Product Recovery. Submitted to Journal of Intelligent Manufacturing on May 15th, 2023.

- N. Hartono and D.T. Pham. A Novel Fibonacci-Inspired Enhancement of the Bees Algorithm: Application to Robotic Disassembly Sequence Planning. Submitted to Cogent Engineering on August 26th, 2023

Presentations:

- The 1st International Workshop on the Bees Algorithm and its Applications (BAA), 29th September 2021, online, title of presentation: A Sustainability Model of Robotic Disassembly using Multiobjective Bees Algorithm.

- University of Birmingham's 3 Minute Thesis, May 2022, title of presentation: Sustaina-bee-lity in Remanufacturing, finalist of 3MT [5]

- The 10th IFAC Conference MIM 2022 on Manufacturing Modelling, Management and Control, Nantes, France, 22nd-24th June 2022, title of presentation: A Sustainability-based Model for Robotic Disassembly Sequence Planning in Remanufacturing using the Bees Algorithm.

- The 5th School of Engineering Postgraduate Researcher Symposium, University of Birmingham, 6th July 2022, title of presentation: A Model for Robotic Disassembly Sequence Plans using the Bees Algorithm.

- The 6th International Workshop on Autonomous Remanufacturing (IWAR), 26th October 2022, online, title of presentation: Optimisation of Robotic Disassembly Sequence Plans for Sustainability using the Multi-Objective Bees Algorithm, Best Presentation Award.

Future presentation:

- The 7th International Workshop on Autonomous Remanufacturing (IWAR), 18-19th October 2023, online, title of presentation: Robotic Disassembly Sequence Planning and Line Balancing - Research Trends Review and Bibliometric Analysis.

Poster presentations:

- University of Birmingham Postgraduate Research Festival, 2021, online, College of Engineering and Physical Sciences's Best Poster and People's Choice Awards. [6]
- BEAR PGR Conference 2021, University of Birmingham, online, Best Poster. [7]
- Circular Revolution 2021, online, Best Design Award. [8]
- Net Zero Futures 21 Conference: Developing Skills and Talent for the Zero-carbon Transition, 2021, Birmingham, 1st place Poster Winner. [9]
- Engineering Professors Council: A Better World EPC Congress, 2022, Bristol, Finalist Poster. [10]
- University of Birmingham Postgraduate Research Festival, 2022, Birmingham, Finalist Poster [11]

Contents

List of Acronyms

A*-MST A* with a Minimum Spanning Tree. 19

A*-NN A* with a Nearest Neighbour. 19

ABC Artificial Bee Colony. 22

ACO Ant Colony Optimisation. 22

AHGA Automated Hybrid Genetic Algorithm. 137, 139

AHP Analytical Hierarchy Process. 17, 19

AI Artificial Intelligence. 15, 30, 97, 127, 165, 174

ARS automated recovery strategy. 6, 56, 57, 65, 72, 88–92, 94, 96, 105, 110, 111, 123, 125, 126

AVNS Adaptive Variable Neighbourhood Search. 138, 139

BA bees algorithm. 7–12, 19, 22, 25, 26, 33, 35, 36, 47, 57, 58, 61, 64, 65, 90, 91, 95, 98, 101, 102, 126, 129–134, 136, 140, 163, 165, 167–175

BA_2 bees algorithm with 2 parameters. 132, 164, 175

BA_F Fibonacci bees algorithm. 12, 130, 140, 141, 144, 145, 147, 148, 154, 155, 160–164, 169–172, 175

BBA basic bees algorithm. 130, 131, 134, 141, 143

BCE Bi-criterion Evolution-based. 22

BLSA Broad Local Search Algorithm. 138, 139

CAD Computer Aided Design. 19, 46

CDDO Child Drawing Development Optimisation. 137, 139

CDG Constraint Decomposition Grid. 22

CE circular economy. 2, 3, 13, 91, 167, 169

D-DQN Double DQN. 22

DLB disassembly line balancing. 5–7, 15, 17, 20

DoE Design of Experiments. 132

DQN Deep Q Network. 22

DS-MOEA Dual-Selection Multi-Objective Evolutionary Algorithm. 19

DSP disassembly sequence planning. 5–7, 15–17, 19

DV Diversity Metric. 22

EDBA Enhanced Discrete Bees Algorithm. 22, 26, 47, 133, 141, 143–148, 154, 155, 160–164, 169–171

EDBA-WMO EDBA without mutation operator. 164

ELS Evolutionary Local Search. 138, 139

EMOGA Extremal Multi-Objective Genetic Algorithm. 22

EoL end-of-life. 2, 3, 5, 10, 13–16, 20, 37, 38, 48, 97–99, 126, 170, 173, 175

FDSPA Fuzzification of Disassembly Sequence Planning. 19

FIA Fibonacci Indicator Algorithm. 137, 139

FSOA Fibonacci Sequence-based Optimisation Algorithm. 138, 139

FSQGA Fibonacci Sequence-based Quantum Genetic Algorithm. 137, 139

FTO Fibonacci Tree Optimisation. 137, 139

GA genetic algorithm. 7, 19, 22, 25, 137, 139

GA-PPX Genetic Algorithm with Precedence Preserving Crossover. 164

GOA Grasshopper Optimisation Algorithm. 19

GROM Golden Ratio Optimisation Method. 137, 139

GWO Grey Wolf Optimiser. 137, 139

HDA Hybrid Driving Algorithm. 22

HI Hypervolume Indicator. 28, 29, 45, 62, 73, 94, 96, 113, 124

IACO Iterative Ant Colony Optimisation. 22

IBEA Indicator-Based Evolutionary Algorithm. 19, 22

IDBA Improved Discrete Bees Algorithm. 22

IGD Inverted Generational Distance. 30

IGSA Improved Genetic Simulated Annealing. 22

IMMO Improved Multi-Objective Multi-verse Optimisation. 22

INSGA-III Improved NSGA-III. 22

ISIACO Interval Search Iterative Ant Colony Optimisation. 22

IWPA Improved Wolf Pack Algorithm. 137, 139

LOA Local Optima Avoidance. 137

MA Memetic Algorithm. 137, 139

MALA Multi-Objective Ant Lion Optimiser. 22

MBGA Many-objective Best-order-sort Genetic Algorithm. 22

MBOHHO Modified Bi-objective Harris Hawks Optimisation. 22

MCDM Multiple-Criteria Decision-Making. 43

MFSG Modified Feasible Solution Generation. 48, 58, 103, 108, 141

MFV maximum fitness value. 88, 90

MILP Mixed Integer Linear Programming. 19, 22

MO multiobjective. 7, 18, 28, 31, 43, 45, 46, 50, 57, 61, 65, 88–92, 106, 133, 134, 159, 168

MO-ND multiobjective nondominated. 7–9, 18, 20, 26, 28, 29, 33, 36, 45, 46, 61, 62, 72, 73, 88, 92, 93, 103, 106, 108, 110, 113, 123, 124, 133, 134, 171

MOABC Multi-Objective Artificial Bee Colony. 22

MOBA multiobjective nondominated bees algorithm. 8, 19, 22, 47, 58, 61, 73, 92–96, 101, 108, 110, 111, 113, 123–126, 168–170

MOCGA Multi-Objective Cellular Genetic Algorithm. 22

MODGWO Multi-Objective Discrete Gray Wolf Optimizer. 22

MOEA Multi-Objective Evolutionary Algorithm. 22

MOEA/D Multi-Objective Evolutionary Algorithm based on decomposition. 19, 22

MOEO Multi-Objective Equilibrium Optimizer. 22

MOFA Multi-Objective Fibonacci Based Algorithm. 138, 139

MOGWO Multi-Objective GWO. 22

MOMVO Multi-Objective Multi-Verse Optimizer. 22

MOPSO Multi-Objective Particle Swarm Optimisation. 22

NFE number of function evaluations. 26, 27, 29, 45, 62, 64, 73, 93, 96, 113, 124, 141, 143, 147, 148, 158, 162–164

NP nondeterministic polynomial. 6, 7, 15, 23, 35, 95, 100, 167

NSGA-II Nondominated Sorting Genetic Algorithm - II. 19, 22, 47, 61, 64, 73, 92–94, 110, 113, 123

NSGA-III Non-dominated Sorting Genetic Algorithm - III. 19, 22

NSHHO Multi-Objective Non-sorted Haris Hawks Optimiser. 22

Opp-PSOGWO Opposition-based Learning PSO GWO. 138, 139

PBEA Problem-specific Bi-criterion Evolutionary Algorithm. 22

PDSA-EA Evolutionary Simulated Annealing Algorithm using Pareto-domination based acceptance criterion. 22

PEI performance evaluation index. 8, 11, 30, 36, 37, 43, 45, 47, 64, 73, 94–96, 103, 110, 113, 124, 141, 158, 162–164, 168–173

PESA-II Pareto Envelope-based Selection Algorithm - II. 19, 22, 47, 61, 64, 73, 92–94, 110, 113, 123

PIMBO Pareto-improved Multi-Objective Brainstorming Optimisation. 22

POSs Pareto optimal solutions. 28, 29, 45, 62, 73, 92, 93, 96, 110, 111, 113, 123, 124

PRDQN Prioritised Experience Replay DQN. 22

PSO Particle Swarm Optimisation. 22, 137, 139

RDLB robotic disassembly line balancing. 6–11, 14, 15, 17, 20, 21, 25, 28, 30, 31, 33, 34, 101–103, 106, 107, 126, 127, 133, 134, 168, 174

RDLBSD sequence-dependent robotic disassembly line balancing. 17, 26, 99, 101–103, 105–108, 110, 123–127, 134, 168–170

RDSP robotic disassembly sequence planning. 6–11, 14–17, 20, 21, 25, 26, 28, 30, 31, 33–37, 46, 47, 49, 58, 61, 65, 95–98, 102, 103, 105, 123, 124, 126, 127, 129, 130, 132–134, 141, 143, 160, 163, 164, 167–169, 171, 174

REC recycling. 56, 72, 88, 89, 91, 92, 94, 96, 105, 125, 126

REM remanufacturing. 56, 72, 88, 89, 92, 94, 96, 105, 125, 126

REU reuse. 56, 72, 88, 89, 91, 92, 94, 96, 105, 125, 126

RS Random Search. 22

RSA Restarted Stimulated Annealing. 22

SA Simulated Annealing. 18, 19, 22

SASSO Self-Adaptive Simplified Swarm Optimisation. 164

SBA standard bees algorithm. 131

SEM standard error of the mean. 147, 148, 160

SO single-objective. 7, 18, 23, 26, 28, 29, 43, 45, 46, 57, 58, 61, 65, 88, 90, 92, 106, 133, 134, 143, 158, 159, 168, 171

SPEA-2 Strength Pareto Evolutionary Algorithm for Multiobjective Optimisation. 22

SPM statistical performance metric. 8, 11, 30, 36, 37, 42, 47, 64, 96, 141, 158, 163, 164, 168–173

SSA Salp Swarm Algorithm. 137, 139

TBA Ternary Bees Algorithm. 132

TS Tabu Search. 137

TSP Travelling Salesman Problem. 134, 164

UK United Kingdom. 1

UN United Nations. 1, 2

VR Virtual Reality. 15

VRP Vehicle Routing Problem. 19, 132, 138, 139, 164

WPA Wolf Pack Algorithm. 137, 139

List of Symbols

α_i	the indicator that takes the value of 1 if component i is to be disassembled and 0 otherwise
γ_i	the indicator taking the value 1 if operation x_{i+1} requires changing the tool used in previous operation x_i
BA_{pop}	Bees Algorithm population
c_T	the cost per unit of time
CD_i	the disposal cost of component i being disposed of
Cy_T	the cycle time
$dp_{i,j}$	the depreciation cost assigned to component i to be disassembled
e	number of elite sites
$ec_{i,j}$	the environmental benefits in the recovering process of component i with mode j
$ed(x_i)$	the environmental benefits in disassembly operation x_i
$ed(x_i, x_{i+1})$	the environmental benefits produced by the movement of the robot between disassembly operations x_i and x_{i+1}, considering that the robot has to change

	the tool in M if operation x_{i+1} requires using a different tool to the one used in the previous operation x_i
$er_{i,j}$	the reclaimed environmental benefits from component i being reused or remanufactured
f	objective
f_1	profit
f_2	energy savings
f_3	environmental impact reductions
f_4	unbalanced line
f_W	conversion factor from kWh to monetary units
$gc_{i,j}$	the energy consumption involved in recovering component i with mode j
$gd_{1,i}(x_i)$	the energy consumption of the robot in the disassembly operation of component i
$gd_{2,i}(x_i, M)$	the energy consumption of the robot in the movement between the position x_i and M
$gd_{3,i}(M)$	the energy consumption of the robot in the tool change
$gd_{4,i}(M, x_{i+1})$	the energy consumption of the robot in the movement between M and x_{i+1}
$gd_{5,i}(x_i, x_{i+1})$	the energy consumption of the robot in the movement between x_i and x_{i+1}
$gr_{i,j}$	the energy reclaimed from component i being reused or remanufactured
i	the index for each component and varies from 0 to N_p

j	the indicator of the recovery mode and equal to 1 if component i is assigned to be reused, 2 if it is to be remanufactured, 3 if it is to be recycled or 4 if it is to be disposed of
m	number of best sites
$m_t(x_i, x_{i+1})$	the moving time between part x_i and x_{i+1}
max_rv	maximum number of re-visits
n	number of scout bee
N_p	number of total parts
nep	recruited bees for elite sites
nr	number of bees recruited for selected sites using ranking based recruitment
nsp	recruited bees for best sites
NWS	the number of workstations
$oh_{i,j}$	the overhead cost assigned to component i to be disassembled
$PD(M, x_{i+1})$	the length between the position of the tool magazine (M) and the point of the disassembly operation x_{i+1}
$PD(x_i, M)$	the distance between the point of the disassembly operation x_i and the position of the tool magazine (M)
$PD(x_i, x_{i+1})$	the distance between the point of the disassembly operation x_i and the point of disassembly operation x_{i+1}
PR_1	the power of the robot used in the disassembly operation

PR_2	the power of the robot used in the movements between the disassembly points
$r_{i,j}$	the indicator of the recovery mode: 1 if mode j is assigned to component i
RC_i	the revenue obtained from component i being recycled
$rc_{i,j}$	the recovery cost of component i being reused or remanufactured
RP_i	the revenue obtained due to the component i to be reused or remanufactured not having been manufactured again for a new product
S_T	the station time
$t_b(x_i)$	the basic time to perform disassembly operation x_i
$t_c(x_i, x_{i+1})$	the tool change time and depends on the tool type
$t_t(x_i, x_{i+1})$	the penalty time for disassembly tool changes between part x_i and x_{i+1}
$t_u(x_i, M)$	the penalty time for process direction changes along the path between x_i and the tool magazine (M)
$t_w(M, x_{i+1})$	the penalty time for process direction changes along the path between the tool magazine (M) and x_{i+1}
$t_z(x_i, x_{i+1})$	the penalty time for disassembly direction changes between part x_i and x_{i+1}
v_e	the line velocity of the industrial robot's end effector
Z	total disassembly time

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Background

The scorching heatwave that engulfed the United Kingdom (UK) in July 2022, with temperatures soaring to a record-breaking 41 degrees Celsius, serves as a vivid reminder of the far-reaching consequences of global climate change. However, the challenges posed by extreme weather events are not exclusive to the UK; regions across the globe are grappling with similar environmental crises. In July 2023, parts of Europe, including Greece, Spain, and Italy, experienced an extreme heatwave with temperatures exceeding 45 degrees Celsius. In light of this urgent situation, taking decisive action becomes paramount. World leaders have established ambitious targets, including a commitment to reduce greenhouse gas emissions by at least 45% by 2030 and achieve net zero emissions by 2050 [12]. These targets underscore the imperative for collaborative efforts in combating climate change and safeguarding the well-being of our planet. This concept aligns with the United Nations (UN)'s comprehensive definition of sustainability, which dates back to 1987. This definition emphasises the significance of satisfying present needs while ensuring the ability of future generations to meet their own needs [13]. Sustainability encompasses economic, social, and environmental dimensions also known as the 'triple bottom line' [14]. By achieving a

harmonious balance between economic prosperity, social equity, and environmental protection, a sustainable future for everyone can be secured.

Sustainable Development Goal 12 of the UN emphasises unsustainable production and consumption patterns as the underlying causes of climate change, biodiversity loss, and pollution [15]. One significant environmental burden arises from the disposal of products in landfills at the end of their life cycle, which pollutes the air, soil, and water. Governments worldwide are recognising this challenge and actively promoting initiatives and solutions to recover products and their components. As part of these efforts, end-of-life (EoL) alternatives are being explored, including reuse, remanufacturing, recycling, and disposal, with disposal being the least preferred option. Instead, the preferred EoL recovery options involve reuse, remanufacturing, and recycling [16, 17].

In response to the urgent need for sustainable practices, the concept of a circular economy (CE) has emerged as a promising approach to address these challenges and extend the utilisation of products and materials [18]. A CE is defined as a system aimed at minimising resource input, waste generation, and energy leakage by slowing, closing, and narrowing material and energy loops [18, 19]. Unlike a traditional linear economy, which follows a ”take-make-dispose” model, a CE promotes recovery options: reuse, repair, recycling, and remanufacturing. To visualise the concept of a CE, the Ellen MacArthur Foundation introduced the butterfly diagram, which effectively portrays the seamless material flow within a CE framework, encompassing both biological and technical cycles [18]. The technical cycle, in particular, focuses on ensuring the continuous circulation of products and materials through various processes, including reuse, repair, remanufacturing, and recycling. This approach aims to extend the effective lifespan of products, enabling them to remain within the circular system for longer periods. Consequently, it fosters the longevity and sustainability of these products even beyond their initial use. Figure 1.1 provides a visual representation of the technical cycle within the butterfly diagram, illustrating the various stages and processes involved in maintaining the circular flow of products and materials. It is

important to note that while recycling is an essential component of a CE, it should be considered the last resort at the end of a product's life due to its higher energy consumption, waste generation, and pollution compared to other options that aim to extend the product's lifespan [20]. Recycling should only be pursued when a product can no longer be used, as it involves transforming the EoL product into its basic material, thus retaining only the value of the materials despite the loss of time and energy invested in making the product [18]. By prioritising options such as reuse, repair, and remanufacturing, a CE strives to minimise environmental impact and maximise resource efficiency.

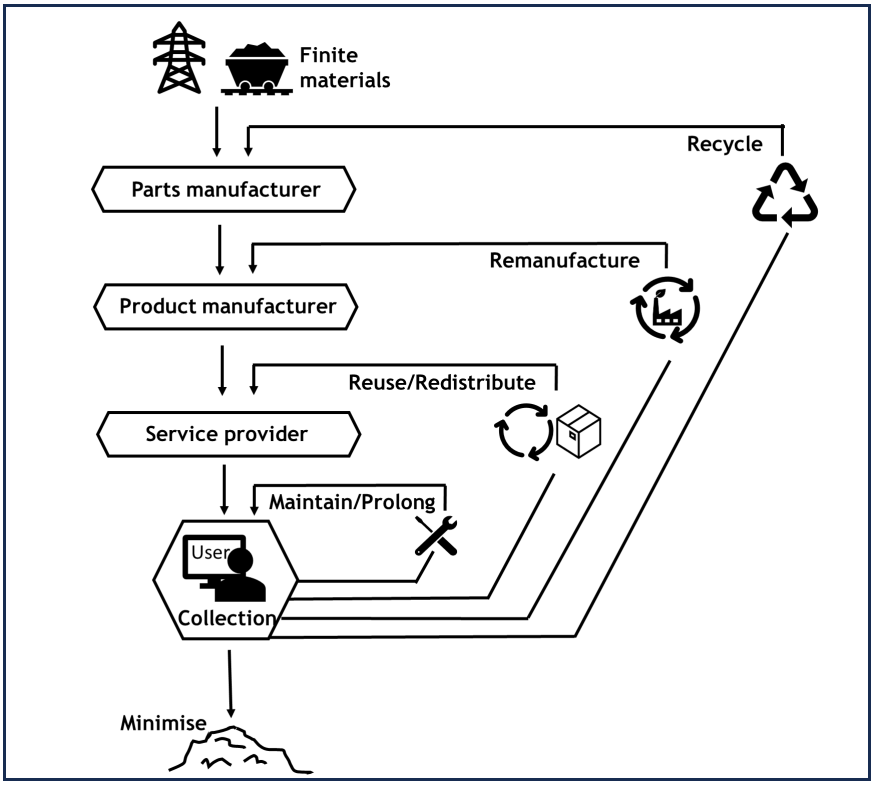


Figure 1.1: Technical cycle (Circular Economy), adapted from [18]

Remanufacturing, considered the backbone of a CE [21], is projected to have a significant impact on the future manufacturing industry, as highlighted by the European Remanufacturing Network. It is anticipated that by 2030, the European Union could witness the emergence of a market worth €90 billion [22]. Remanufacturing plays a pivotal role in transitioning towards a more sustainable CE by restoring EoL products to their original performance, sometimes surpassing

that of new products [23–25]. By reducing waste, raw material consumption, CO_2 emissions, and energy usage, remanufacturing contributes to resource preservation and delivers environmental benefits [26, 27]. In a study conducted by [28], a life cycle model was employed to assess a remanufactured engine, revealing significant reductions in greenhouse gas emissions (73-87%) and energy usage (68-83%). Renault, a pioneer of the circular economy in the automotive industry, reported that their remanufacturing activities generated nearly €120 million in revenues in 2019 while achieving significant savings of 80% in energy, 88% in water, 92% in chemical products, and 70% in waste through the production of remanufactured parts [29].

Remanufacturing hinges upon a pivotal operation: disassembly [17, 30–33], distinguishing it from conventional manufacturing [34]. Disassembly involves the systematic separation of products into parts and subassemblies [35], accompanied by essential inspection and sorting procedures, making it a critical aspect of efficient product reprocessing [36]. Despite sharing similarities with assembly, disassembly is not a mere reversal of the assembly process [37–39]; instead, it possesses its own unique characteristics [40, 41]. For instance, the removal of fasteners and gluing problems are specific challenges encountered in disassembly. Additionally, disassembly follows a divergent flow, where a single product is broken down into its constituent parts, as opposed to the convergent flow observed in assembly [41]. The disassembly process also introduces uncertainties related to the product's condition, such as missing, corroded, or worn-out parts, requiring careful consideration to ensure successful disassembly [41–45].

The significance of disassembly in remanufacturing lies in its ability to enable the recovery and reuse of valuable components, contributing to resource conservation and waste reduction [22, 29]. By systematically separating products into parts and subassemblies, reusable components can be identified and extracted, thereby reducing the demand for new raw materials [22, 29]. These components can also be repaired or rebuilt. In cases where components cannot be reused or remanufactured, they can be recycled or properly disposed of. Additionally, through the process of dismantling components, manufacturers can gain valuable insights into the behaviour of the

product after a certain period of use, allowing for more robust design improvements [46].

Key areas in remanufacturing encompass disassembly operations, specifically disassembly planning and disassembly scheduling [32, 47–49]. Disassembly sequence planning (DSP) involves a meticulous analysis of the product's structure and component geometry to determine highly efficient or nearly optimal disassembly sequences [33, 50]. Conversely, disassembly line balancing (DLB) aims to achieve a well-balanced allocation of tasks during the disassembly process, ensuring smooth inventory flow while considering options for product recovery [41, 50]. Both elements are crucial in achieving efficient disassembly operations. The primary objective of DSP is to develop a comprehensive plan for systematically removing components or subassemblies from complete products [32, 33]. This involves determining the most efficient order for conducting disassembly procedures, considering various criteria such as component preferences and fastener constraints [41, 50]. DLB, on the other hand, focuses on achieving an even distribution of workload among workstations or operators, thereby enhancing the overall efficiency of the disassembly process. The interconnection between DSP and DLB is widely recognised as pivotal in improving efficiency and productivity within remanufacturing plants [47, 48, 51, 52]. This interconnected problem is often referred to as "sequence-dependent line balancing" in the literature. Understanding and addressing the relationship between sequence planning and line balancing is essential for optimising disassembly operations [47–49].

Traditionally, disassembly has been predominantly a manual activity due to the intricacies associated with EoL products [47, 53]. However, with the advent of Industry 4.0, technological advancements have fostered a shift from human labour to automated processes. The increasing number of publications on disassembly automation since 2014 highlights the growing interest in this field. While manual disassembly has been the norm [45, 50, 54, 55], the application of automated disassembly using robots is starting to occur, owing to robots' enhanced efficiency and their ability to handle uncertainties in dynamic disassembly processes [45]. Additionally, robots can effectively and safely manage hazardous disassembly tasks. The transition from manual to

robotic disassembly can result in substantial improvements in the productivity and efficiency of disassembly processes [45]. Robots offers distinctive characteristics, such as diverse kinematics, flexibility, and dynamic capabilities, which set them apart from human operators [48, 54]. As automation will play a pivotal role in the future, this study underscores the significance of robotic disassembly as an indispensable process in remanufacturing. Moreover, the disassembly line is highly suited to an automated system [56]. Chapter 2 will provide further insights into these characteristics and considerations associated with robotic disassembly.

A concise review by the author of the literature in the field of robotic disassembly reveals several key findings. In robotic disassembly sequence planning (RDSP), the majority of articles focus on minimising disassembly time, with only two article addressing sustainability. On the other hand, in robotic disassembly line balancing (RDLB), sustainability is emphasised as one of the primary objectives in half of the reviewed articles. This indicates that RDLB places a greater emphasis on sustainability compared to RDSP. This thesis fills this gap by providing a more comprehensive sustainability model than previous studies on RDSP and RDLB. Additionally, previous research has not provided a comprehensive output of disassembly with recovery options for each component, nor has it utilised an algorithm to find the best recovery option. This thesis addresses these gaps by incorporating the automated recovery strategy (ARS) scenario. In the ARS scenario, the algorithm determines the best recovery option for each disassembled component. Three other scenarios, explained in Chapter 3, are also considered to determine the optimal recovery options for each disassembled component.

Both DSP [50, 57–61] and DLB [41, 62–64] present computational complexity challenges. As the number of disassembled components increases, finding the optimal solution becomes significantly more time-consuming, with computational requirements growing exponentially. The DSP is known to be nondeterministic polynomial (NP)-complete, making mathematical programming methods impractical for solving it [57, 65, 66]. While exact methods have been used for simpler scenarios, they struggle to handle complex situations with numerous components and

intricate product structures [67–69]. As a result, approximate algorithms based on metaheuristics have gained popularity for solving DSP problems within a reasonable computational time. Similarly, DLB is recognised as an NP-complete problem that requires considering multiple criteria [41, 62, 63]. The computational complexity challenge in robotic disassembly is similar to its manual counterpart, as both are NP-complete problems. To address the challenges in robotic disassembly, metaheuristic algorithms are suitable optimisation tools for finding efficient solutions. The literature presents three notable approaches: single-objective (SO), multiobjective (MO) aggregate, and multiobjective nondominated (MO-ND) approaches. The MO aggregate approach assumes linear relationships between objectives and treats the problem as an SO optimisation, while the MO-ND approach considers conflicting objectives and provides a set of nondominated solutions.

In RDSP, most research adopts SO optimisation approaches, with only a few utilising an MO aggregate approach. The genetic algorithm (GA) and the bees algorithm (BA) have been widely employed as metaheuristics in RDSP, offering effective search and exploration capabilities for identifying optimal solutions within a reasonable computational time. Notably, this research makes a significant contribution to the literature by employing an MO-ND approach in RDSP, addressing the limitations of the MO aggregate approach. It is worth mentioning that there is only one other publication by [43] that utilises the MO-ND approach in RDSP. However, this thesis takes a different approach by not assuming linear or conflicting relationships between objectives. Instead, it uses both the SO and MO aggregate approaches before employing the MO-ND approach to determine the most appropriate approach. This allows for a comprehensive evaluation and comparison of different methodologies, providing valuable insights into the suitability of each approach for addressing the research problem. In contrast, in RDLB, researchers commonly employ the MO-ND approach, which is well suited for handling problems with conflicting objectives [52, 70–78]. While the GA is commonly used in research on RDLB, the BA has also demonstrated its effectiveness in this domain. The BA is a robust metaheuristic that efficiently

addresses the complex challenges encountered in both RDSP [44, 51, 55, 79] and RDLB [47–49]. The BA, inspired by the foraging behaviour of honeybees, explores and exploits the search space iteratively, making it a robust metaheuristic for efficiently addressing the complex challenges encountered in robotic disassembly. In this thesis, the BA is chosen as the primary optimisation tool. However, previous research in RDSP and RDLB has been scarce in reducing the parameter settings of the BA. Therefore, Chapter 5 introduced an enhancement to the BA, reducing the number of user-selected parameters to four. This enhancement aims to simplify parameter setting, improve the algorithm's capabilities, expand its applicability in the field of robotic disassembly, and enhance its potential use for solving other optimisation problems. By utilising the MOBA and its parameter reduction enhancement, this research fills a critical gap in the literature and broadens the range of optimisation techniques available for RDSP and RDLB.

In addition to the existing gap in the literature regarding the limited exploration of performance evaluation using statistical methods, previous studies have predominantly relied on descriptive statistics, such as average, median, and standard deviation values, without fully harnessing the potential of statistical tests. Moreover, there is a lack of consistency in the performance metrics employed for MO-ND approaches, with limited reporting on conflicting metric results. To address these limitations, this thesis introduces two novel measures: the statistical performance metric (SPM) and the performance evaluation index (PEI). These measures aim to enhance the performance evaluation process by incorporating robust statistical methods and providing a comprehensive assessment of the results. The SPM facilitates a more rigorous comparison of algorithms by quantifying the statistical significance of performance differences and identifying the optimal parameter settings. Additionally, the PEI is a versatile metric that offers a comprehensive measure of algorithm performance based on multiple metrics. By introducing these new tools, this research not only fills a crucial gap in the literature on performance evaluation in robotic disassembly but also contributes to the broader field of evaluating metaheuristic algorithms. These advancements enable researchers to make more informed decisions and draw meaningful

conclusions from their experimental results.

In summary, the review of the literature in the field of robotic disassembly reveals several significant gaps. The first gap pertains to the lack of consideration for sustainability in RDSP, with only half of the research in RDLB addressing this crucial aspect. Additionally, there is a clear trend of research on sequence-dependent RDLB. Furthermore, comprehensive reporting of disassembly output, particularly regarding recovery options for each component, is lacking, along with the absence of algorithms to determine the optimal recovery choices. Moreover, while MO-ND optimisation approaches are commonly used in RDLB, the application of the MO-ND approach is currently lacking in RDSP studies. Another notable gap in the literature is the lack of specific research focused on reducing the number of parameters of the BA in both RDSP and RDLB. Furthermore, previous studies have underutilised the potential of statistical tests for performance evaluation, relying primarily on descriptive statistics. Finally, the absence of a unified performance evaluation metric across previous studies further emphasises the need for improvement in this area.

1.2 Hypothesis and research questions

Based on the preceding information, the research hypothesis is formulated as follows:

"Sustainable solutions for robotic disassembly sequence planning and line balancing can be developed and optimised using the Bees Algorithm"

The hypothesis will be tested and supported by addressing these research questions:

1. Development of a Sustainability Model: How can a sustainability model be developed for automating the disassembly of end-of-life (EoL) products?

2. Optimisation of Robotic Disassembly Sequence and Line Balancing: What optimisation methods can be applied to find the best solutions for robotic disassembly sequence planning and line balancing?

3. Optimal Parameter Settings and Performance Metrics: What techniques can be developed to determine the optimal parameter settings and performance metrics for optimisation algorithms?

4. Enhancement of the Bees Algorithm: How can the bees algorithm be enhanced to optimise robotic disassembly processes?

1.3 Aims and Objectives of the Research

The aim of this work was to explore and develop efficient and sustainable solutions for RDSP and RDLB, with a particular emphasis on the application of the BA and its novel enhancement. To achieve this aim, the following objectives have guided the research:

1. Develop a sustainability model for the disassembly of EoL products, including the formulation of three predefined recovery scenarios and utilisation of an algorithm to determine the best recovery option for each part.

2. Determine the optimal order for disassembling parts within a robotic cell to optimise efficiency and effectiveness.

3. Balance the disassembly line, considering sequence dependence within a robotic line, to optimise the overall performance of the disassembly line.

4. Validate the proposed approach through a case study on gear pumps, demonstrating its effectiveness in solving the robotic disassembly problem using a real EoL product as an illustrative example.

5. Determine the optimal parameter settings and performance metrics for optimisation algorithms in robotic disassembly, enabling the identification of the most effective parameter configurations and facilitating straightforward comparisons among different algorithms.

6. Enhance the BA to optimise solutions for the robotic disassembly problem.

These objectives collectively provide a clear focus for the research, guiding the investigation and development of efficient and sustainable solutions for RDSP and RDLB. The emphasis is on enhancing the capabilities and effectiveness of the BA, enabling its successful application in addressing the challenges of robotic disassembly. The enhancement of the BA in robotic disassembly enforces the notion of the capabilities of the BA and its variants.

1.4 Thesis Outline

The thesis is structured into six chapters, each contributing to the organisation and content of the research.

- **Chapter 2: Literature Review**

  This chapter provides a solid foundation of knowledge and understanding in the field of robotic disassembly. It covers various aspects, including robotic disassembly sequence planning, robotic disassembly line balancing, optimisation algorithms, performance evaluation, and bibliometric connections. The identified gaps and trends are highlighted in this chapter.

- **Chapters 3 and 4: Robotic Disassembly Sequence Planning and Line Balancing**

  Chapter 3 delves into RDSP, presenting sustainability models, methodologies, four recovery scenarios, optimisation approaches, and performance evaluation. This chapter aligns with objectives 1, 2, 4, and 5. To validate the proposed model and algorithm, a case study using two gear pumps is presented. Additionally, two novel tools for performance evaluation (SPM and PEI) are introduced, enhancing the robustness of the algorithm assessment. Chapter 4 focuses on RDLB, with a particular emphasis on sequence-dependent scenarios. It addresses

objectives 1, 3, 4, and 5.

- **Chapter 5: Enhanced Bees Algorithm for Robotic Disassembly Planning**

  This chapter focuses on the BA and introduces its enhancement, the BA_F. The chapter explains the inspiration and mechanism behind the algorithm, showcasing its development. The same case study as the previous chapters and a single objective of minimising disassembly time were used to validate the proposed enhancement. The chapter addresses objectives 2, 4, 5 and 6.

- **Chapter 6: Conclusion**

  This chapter summarises the key findings, contributions, and implications of the research, emphasising their significance in the broader context of robotic disassembly. It also provides suggestions for future research directions and areas of exploration.

Chapter 2

Literature Review

Remanufacturing is recognised as a pivotal component of a circular economy (CE) due to its numerous benefits [21]. It not only allows products to be restored to a condition equal to or better than new [23–25], benefitting both remanufacturers and consumers, but also generates higher profits compared to other recovery options [3, 29]. Moreover, remanufacturing plays a vital role in promoting environmental sustainability by significantly reducing landfill waste, energy consumption, raw material usage, and greenhouse gas emissions [26, 28, 29]. The inclusion of prolonging the use of end-of-life (EoL) products and keeping them in circulation for an extended period further enhances the environmental sustainability aspect of remanufacturing. Additionally, remanufacturing creates job opportunities, making a positive impact on society as a whole [22]. Disassembly serves as the first step in the remanufacturing process [17, 30–33, 54]. It involves the careful separation of components from EoL products to recover valuable materials for reuse, remanufacturing, or recycling. Effective disassembly enables remanufacturers to obtain high-quality components and materials that can be further processed and utilised in the production of remanufactured products. This highlights the significance of disassembly in facilitating the transition from EoL products to valuable resources for remanufacturing. With the advent of Industry 4.0, automation, particularly robotic disassembly, has gained prominence in hazardous

or challenging disassembly tasks, enhancing worker safety and reducing reliance on manual labour. Robotic systems can improve the economic viability of disassembly operations by increasing speed, accuracy, and productivity [1]. Moreover, adopting robotic disassembly aligns with the growing demand for sustainable and circular manufacturing practises, leading to cost savings and operational efficiency. By integrating robotic systems into the disassembly process, remanufacturers can achieve higher levels of efficiency, accuracy, and productivity. Optimisation of robotic disassembly encompasses various key areas, with a notable emphasis on sequence planning and line balancing [47–49]. Sequence planning involves determining the optimal order of disassembly [33, 41, 54], while line balancing ensures an even distribution of workload among robots [47, 49, 51]. By addressing these factors, the efficiency of robotic disassembly in remanufacturing can be optimised.

This chapter is structured as follows: It begins with a comprehensive analysis of the existing literature on robotic disassembly sequence planning (RDSP) and robotic disassembly line balancing (RDLB). The chapter then explores the utilisation of optimisation algorithms in previous research on RDSP and RDLB, followed by an examination of performance evaluation methodologies. To gain deeper insights into the research landscape, a bibliometric analysis is conducted to identify trends and interconnections within the RDSP and RDLB domains. The chapter concludes with a summary of the key findings.

2.1 Robotic Disassembly

The use of robots to perform disassembly has a number of benefits over the traditional method of disassembly. It has the potential to increase the amount of strategically important materials that can be recovered from EoL products [80] and can also significantly improve industrial processes [81]. Researchers have investigated automated methods for disassembly; however, it is still

understudied [82]. The use of robots has the potential to improve the effectiveness of resource recovery. More research is needed, however, to optimise robotic disassembly and make them economically viable for widespread adoption. The development of a sturdy robotic disassembly sequence design can be achieved through the utilisation of task sequencing algorithms, thereby enhancing efficacy, and distributing the workload evenly across several robotic arms [83]. This highlights the potential benefits of implementing robotic disassembly in industries that deal with large volumes of end-of-life products. The implementation of this approach not only results in a reduction of costs, but also enhances overall productivity, safety and increase of the recovery of the EoL parts.

The integration of Virtual Reality (VR) and vision systems, including cameras, is observed in robotic disassembly operations [84, 85]. These technologies have the potential to enhance disassembly accuracy and efficiency. However, they do not fully address critical issues, such as determining the optimal disassembly sequence and workload distribution, which are crucial factors in achieving maximum efficiency in remanufacturing. Recent advancements using digital twins [86] and Artificial Intelligence (AI), such as deep learning [86–89], have shown promise in addressing these challenges. However, it is important to note that most of these developments emerged after the publication of this thesis, and they primarily utilise a simple objective to demonstrate their potential. Therefore, while they were not incorporated into this research, they serve as valuable areas for further investigation.

The disassembly sequence planning (DSP) and disassembly line balancing (DLB) as well as RDSP and RDLB are recognised as nondeterministic polynomial (NP) complete and intractable problem that is not suitable for treatment by mathematical programming methods when the size of the problem is large [30, 57, 58, 65, 66, 83, 90–96]. The previous research shown that most methods to solve these problems are uses the metaheuristic methods due to their ability to find near-optimal solutions faster than exact method. Metaheuristics can be defined as optimisation algorithms that are capable of solving complex problems in a reasonable amount of time [41, 54, 79, 84, 97].

The process of robotic disassembly poses unique challenges in comparison to manual disassembly, primarily due to the distinct kinematic and dynamic characteristics of robots and humans [43, 44, 48, 54]. In particular, collision avoidance is a critical factor to consider when devising a plan for robotic disassembly [48, 54]. The trajectory of the robot's end effector in order to prevent collision has an impact on the overall disassembly time [49, 54]. Prior studies on robotic disassembly have typically ignored the product's contour when planning the robot's path [54]; however, this should be taken into account. To overcome this obstacle, various methods have been developed by researchers for collision-free robotic disassembly. One of the methods involves considering the geometry of the object being disassembled and calculating the distance between disassembly points to ensure contour-based collision avoidance is respected [48, 49, 54, 98]. This method [49, 54] is used to determine the distance between disassembly points while maintaining a minimum distance of 10 mm between the end effector's moving path and the contours of the EoL product. This robotic collision avoidance trajectory proposed by previous researchers is used in this thesis.

2.2 Robotic Disassembly Sequence Planning

DSP refers to a methodical approach used to identify the optimal sequence of activities in separating an EoL product [99] in detail [32], which involves three main steps, as described by [33]. These steps include determining the disassembly mode (partial or complete), developing a disassembly model (which encompasses disassembly precedence relationships), and selecting disassembly planning methods (objective and optimisation method). Among the different types of disassembly models, graph-based models are the most commonly used, followed by matrix-based models, Petri Net, and others [33]. The RDSP uses robot(s) to dismantle EoL products. The use of robots in disassembly activities has been studied in the literature, and various approaches to optimising

the process have been proposed. Robots can automate the process, saving time, reducing human error, and increasing productivity and efficiency. This thesis focuses solely on robotic disassembly, without human involvement. Thus, research related to human-robot collaboration in disassembly was excluded due to the inherent differences between human and robot capabilities and dynamics. Furthermore, works that integrated DLB and DSP were also omitted, as they will be addressed in the subsequent section. A search through May 2023 in the Scopus database using the keywords ”robot*” AND ”disassembly” AND ”sequenc*” yieled 49 articles on RDSP, and the availability of full articles written in English was verified. Excluding publications from this thesis [1–4], only two addressed sustainability [100, 101], with the majority of the articles focusing on minimising time.

In the publication [1, 2], it is pertinent to note that [52]'s research has been included within the research framework, highlighting their focus on having sustainability objective. However, their research delves into RDSP and RDLB, which are addressed in a separate section of this thesis. Similarly, [102] are also positioned within the research framework, even though they do not explicitly mention RDLB. Nevertheless, upon closer examination, it becomes evident that their study aligns closely with the concept of sequence-dependent robotic disassembly line balancing (RDLBSD). Thus, while both publications are relevant to RDSP within the wider context of the publication, for the purposes of this thesis, they are more aptly categorised under RDLB, considering the incorporation of sequence dependence.

Table 2.1 serves as a valuable starting point for future investigations into sustainable practises in this field. It is evident from the table that this research area has gained significant attention over the past two years. The highlighted articles, including the publication of this thesis, underscore the position of this study in relation to previous work, thereby emphasising the importance of further research on sustainability within the field. Notably, only one previous study [100] addresses the recovery options of reuse, recycling, and disposal for each part, highlighting the necessity of incorporating this aspect into the sustainability model. In [100], the Analytical Hierarchy Process (AHP) was utilised as a decision-making tool to assign weights to the sustainability criteria based

on the decision maker's preferences. These weights were subsequently incorporated into the Simulated Annealing (SA) algorithm to generate optimal solutions for minimising the disassembly time. This thesis differs from [100] by employing a multiobjective nondominated (MO-ND) approach, introducing a different sustainability model, and developing distinct recovery options scenarios.

The majority of prior studies have employed metaheuristics, with a predominant emphasis on single-objective (SO) optimisation. The complexity of multiobjective (MO) optimisation has resulted in a limited number of studies addressing this topic. In many of the early studies, the typical approach has been to begin with more straightforward goals and gradually build upon the foundation laid by those initial objectives. While the MO aggregate approach may be suitable in some cases, where objectives can be added linearly and treated as a SO, it is not appropriate when dealing with conflicting objectives. When there are conflicting objectives, a nondominated approach is more appropriate as it allows researchers to consider multiple objectives without sacrificing the feasibility of any one objective [97].

Previous studies in this field have predominantly employed either an SO or an MO approach. In the limited cases where the MO is utilised, researchers tend to focus on either the aggregate method or the nondominated approach, without adequately considering the potential aggregation or conflict of objectives. In this thesis, the linearity of objectives is explored using those two MO methods: the aggregate method, which assumes that objectives with the same units (monetary value, in this case) can be combined and treated as an SO, and the nondominated approach, which is used to solve conflicting objectives. The maximum value for each objective is also determined using the SO approach in this thesis.

Table 2.1: RDSP Research Position

Author(s)	Year	Approach	RDSP objective(s)	Single/Multi	Output	Performance Measurement
Suzuki et al. [103]	1996	Petri Net	learning control scheme	n.a.	simulation	n.a.
Sundaram et al. [104]	2001	Motion planning	min disassembly steps	n.a.	disassembly tree	n.a
Baeza et al. [105]	2002	Contact surface and unnamed Heuristic	disassembly movement	n.a.	disassembly movement and sequence	n.a.
Puente et al. [106]	2003	Vision system	flexible automatic disassembly	n.a.	simulation	n.a.
Uhlmann et al. [107]	2005	Control system	feasibility of disassembly concept	n.a.	pilot disassembly system	n.a.
Kim et al. [108]	2007	Control system	automatic sequence generation	n.a.	automated disassembly control concept	n.a.
Gil et al. [109]	2007	Visual-force control system	a collaborative robotic system with multiple sensor	n.a.	experiment validation	n.a.
ElSayed et al. [110]	2011	GA**	disassembly sequence generation	n.a.	intelligent automated disassembly cell	n.a.
ElSayed et al. [57]	2012	GA**	min time	SO	disassembly time, sequence, detection time	n.a.
Vongbunyong et al. [111]	2015	Cognitive robotics	skill transfer from human to robot	n.a.	cognitive robotic disassembly experiment	n.a.
Popescu et al. [112]	2016	Software	automatic generation	n.a.	generate sequence from CAD	n.a.
Alshibli et al. [113]	2016	Robot sensory system, Tabu search, GA**	min makespan	MO	run time	run time****
Friedrich et al. [114]	2016	CAD and Vision	automated planning system	n.a.	experiment validation	n.a.
Vongbunyong et al. [115]	2017	Vision system	skill tranfer from human to robot	n.a.	process demonstration platform	n.a.
Friedrich et al. [116]	2017	Djikstra, A*-NN, A*-MST, nearest neighbour	min time	SO	path planning	path (time, length, smoothness), success rate, deviation, detection time***
Parsa and Saadat [59]	2018	GA**	min time	SO	disassembly sequence, tools, destructive/non	n.a.
Wang et al. [117]	2018	Matrix manipulation	detect subassemblies automatically	n.a.	automatic detection of subassembly using matrix	n.a.
Laursen et al. [118]	2018	Programming language	programming model to reverse assembly	n.a.	domain specific language	n.a.
Liu et al. [54]	2018	BA, GA, SA**	min time	SO	disassembly sequence, direction	average fitness value and run time***
Costa et al. [119]	2018	Branch and Bound and CAD automatic generation	min cost	SO	disassembly sequence	n.a.
Alshibli et al. [100]	2018	SA** and AHP (for **environmental**, economic, social criteria)	min time	SO	disassembly sequence, method, recovery option	n.a.
DiFilippo and Jouaneh [120]	2018	Vision and force system	fastest time	n.a.	cognitive system framework	n.a.
Laili et al. [44]	2019	Greedy search, GA, BA**	min time (re-planning)	SO	rapid subassembly detection and sequence	time***
Zhang et al. [98]	2019	Hybrid A* and GA** & obstacle avoidance	min path	SO	experiment on reduction gearbox	convergence speed, run time
Lan et al. [121]	2020	Search for separable pairs & divide and conquer	avoid interlocking	n.a.	disassembly sequence	n.a.
Ramírez et al. [122]	2020	Constructive greedy, hill climbing, GA**	max profit	SO	disassembly sequence	graphical results
Chen et al. [48]	2020	BA**	min time	SO	disassembly sequence	fitness value and run time***
Watanabe and Inada [123]	2020	Reinforcement Learning	min time	SO	experiment to validate concept	n.a.
Wang et al. [53]	2021	Matrix manipulation	representation matrix for complex disassembly	n.a.	mathematical representation	n.a.
Malekkhouyan et al. [101]	2021	VRP and DSP using MILP, GOA**	**min transportation cost, robot and truck carbon footprint, min time**	MO	disassembly sequence	range
Laili et al. [43]	2021	IBEA, MOEA/D, NSGA-II, NSGA-III, DS-MOEA**	min time and max completion rate	MO-ND	time and completion rate result	HI, IGD, (ϵ-indicator)****
Hartono et al. [1]*	2022	BA**	**max profit, energy savings, environmental impact reduction**	SO	disassembly sequence, direction, tools, recovery options	SPM****
Hartono et al. [2]*	2022	BA**	**max profit, energy savings, environmental impact reduction**	SO	disassembly sequence, direction, tools, recovery options	n.a.
Hartono et al. [3]*	2023	MOBA, NSGA-II, PESA-II**	**max profit, energy savings, environmental impact reduction**	MO, MO-ND	disassembly sequence, direction, tools, recovery options	HI,NFE,POSs
Hartono et al. [4]*	2023	BA, MOBA, NSGA-II, PESA-II**	**max profit, energy savings, environmental impact reduction**	SO, MO, MO-ND	disassembly sequence, direction, tools, recovery options	HI,NFE,POSs,SPM****
Laili et al. [79]	2022	Greedy search, GA, BA**	min time	SO	disassembly time	disassembly time****
Laili et al. [124]	2022	Mathematics model formulation	compilation of objectives from previous research	n.a.	n.a.	n.a.
Laili et al. [125]	2022	IBEA, MOEA/D, NSGA-II, NSGA-III, DS-MOEA**	min time	MO-ND	time and completion rate result	HI, IGD, (ϵ-indicator)****
Ye et al. [126]	2022	BA, FDSPA**	min time	SO	disassembly sequence, direction	solution quality and time
Prioli et al. [127]	2022	CAD files to matrix	disassembly matrix	n.a.	disassembly sequence, direction	n.a.
Yang et al. [88]	2022	Deep Learning, BA, GA**	min time	SO	disassembly sequence, direction	disassembly time***
Liu et al. [86]	2023	BA, GA**, Digital Twin and Deep Q-learning	min time	SO	disassembly sequence, direction	run time
Cui et al. [89]	2023	Deep Q-learning, GA, BA**	min time	SO	disassembly sequence, time	disassembly time***

Note: * thesis-derived publications, ** metaheuristic, *** statistic descriptive, **** statistic test, highlighted bold = sustainability-related article

2.3 Robotic Disassembly Line Balancing

As previously noted in Chapter 1, efficient robotic disassembly relies on two fundamental concepts: robotic disassembly line balancing and the disassembly sequence [47, 49, 128]. Previous research has mainly focused on presenting a feasible disassembly sequence to achieve line balancing [129]. While these areas are typically treated as separate entities by most researchers [47, 49], recent studies in manual disassembly have underscored the importance of sequence-dependent disassembly line balancing [130–136], which has also been extended to the field of robotic disassembly [47, 51, 52, 128]. It is worth noting that sequence-dependent disassembly line balancing simultaneously improves both the sequence and the line [47, 49], challenging the conventional viewpoint. An optimised and feasible disassembly sequence significantly enhances the efficiency of line balancing by enabling the allocation of tasks in an optimised manner.

A literature review conducted using the Scopus database identified 70 relevant articles on robot disassembly line balancing and sequence planning using the keywords ”robot*” AND ”disassembly” AND ”line” AND ”balancing” OR ”sequenc*”. After screening for non-English language and survey papers, 51 articles were selected. Further refinement for automated disassembly narrowed down the selection to 37 articles. The formal description of DLB can be traced back to 2002 [41, 137]. Research on RDLB began in 2011 and has gained momentum since 2019. Early assumptions regarding disassembly as the reverse of assembly were challenged [41], particularly in the context of remanufacturing. Disassembly is widely recognised as a complex problem, particularly due to the presence of uncertain conditions associated with EoL products and challenges posed by various connecting mechanisms such as fasteners or glue. Table 2.2 shows the position of this thesis in relation to previous research. In comparison to RDSP (see Table 2.1), the RDLB research places a greater emphasis on sustainability as its objective, with 18 articles focusing on this aspect, excluding the 2 articles from this thesis. The prevalence of the MO-ND

and metaheuristic approach is evident in the literature as the most commonly employed method for solving RDLB. In two articles, the authors self-identified their work as RDSP [102, 138]. However, upon closer examination of the methods and results, it became apparent that the content of these articles aligns more closely with the field of RDLB. Therefore, in the table, these articles are appropriately reclassified as RDLB.

In previous research, 15 articles primarily focused on reducing energy consumption, while three studies specifically aimed to minimise carbon emissions. In contrast, the present work takes a comprehensive approach, considering profit, energy savings, and environmental impact reduction. Notably, this research goes beyond previous studies by incorporating recovery options route for each component, using four sustainable scenarios, making it distinct in its scope and contribution.

In contrast to the RDSP, case studies within the RDLB field predominantly focus on prototypes, simple problems such as ballpoint pens, and benchmark datasets. However, the selection of gear pump as a case study in three articles [47, 49, 128] highlights its significance as a notable medium-sized real-world problem in the research area. This case study of gear pumps will be explained in Chapter 3 and used as validation of the proposed approach in Chapters 3, 4, and 5.

Table 2.2: RDLB Research Position

Author(s)	Year	Approach	RDLB objective(s)	Single/Multi	Output	Performance Measurement
Radaschin et al. [137]	2011	Expert Petri net	concept testing	n.a.	Concept and implementation on prototype	n.a.
Minca et al. [139]	2014	Synchronised Hybrid Petri Nets model	min cycle time	n.a.	A real-time control structure	-
Minca et al. [140]	2015	Mathematical model	min cycle time	n.a.	Mathematical representation	-
Filipescu et al. [141]	2016	Simulation and real-time control	disassembly after assembly	n.a.	Simulation and real-time control	-
Liu et al. [51]	2018”	BA, ABC, GA**	min workstation, workload balance, disassembly priority of high demand parts	MO-ND	Disassembly sequence, direction, robotic assignments	Iterations and population sizes***
Gao et al. [52]	2018”	MOABC**	**min cost, work load, energy consumption**	MO-ND	Disassembly line schedule	n.a.
Alshibli et al. [83]	2019”	SA**	min robot, balanced load, hazard, demand	MO	Disassembly sequence, destructive/not , task allocation	n.a.
Octavian et al. [142]	2019	Concept testing assembly/disassembly	control strategy	n.a.	Concept and implementation on prototype	n.a.
Fang et al. [70]	2019	IBEA, NSGA-II, MOEA/D, PBEA**	**min cycle-time, total energy consumption, peak workstation energy consumption, number of robot**	MO-ND	Objective and performance metric results	HI**** (wilcoxon rank sum)
Fang et al. [71]	2019	MOEA/D, NSGA-II, NSGA-III, INSGA-III**	**min line length and energy consumption**	MO-ND	Performance measurement	HI, IGD
Ming et al. [143]	2019	Illustrative example	**min cycle time, peak and total energy consumption, cost of hazardous tasks**	SO	Using only min cycle time to shows the example task and robot assignment	n.a.
Liu et al. [74]	2019”	MBGA, NSGA-II, SPEA-2, MOEA/D**	min robot, open multi-robotic workstation, load density, cost of hazardous task	MO-ND	Performance measurement	HI, IGD
Çil et al. [144]	2020	RS, GA, IACO, ISIACO**	min cycle time	SO	Task and robot assignment	RPD
Fang and Xu [73]	2020	MOEA/D**	min cycle time, robots	MO-ND	Performance measurement	HI, IGD
Fang et al. [145]	2020	NSGA-II, MOEA/D, PBEA**	min cycle time, robots	MO-ND	Task and robot assignment	HI, IGD
Liu et al. [47]	2020	BA, GA, PSO**	min cycle time, workstation, smoothness index, max working time	MO	Disassembly sequence, direction, robotic workstation assignments	Fitness value and run time
Fang et al. [72]	2020	NSGA-II, RSA, PDSA-EA**	**min cycle time, peak and total energy consumption**	MO-ND	Performance measurement	Execution time, RPF, CP, HI
Fang et al. [146]	2020	NSGA-II, MOEA/D**	**min cycle time, energy consumption**	MO-ND	Performance measurement	HI, IGD
Chen et al. [128]	2020”	NSGA-II, MOEA/D, IBEA**	min workstation, idle time, demand index of disassembly part	MO-ND	Performance measurement	HI, IGD
Dong et al. [75]	2021	MOEA/D, NSGA-II, MALA	**max profit, min energy**	MO-ND	Performance measurement	HI, GD(N), IGD, Epsilon(N)
Zhang et al. [76]	2021	MOMVO, NSGA-II, MOEA/D, MOCGA**	**max profit, min carbon emissions**	MO-ND	Performance measurement	HI, IGD, Epsilon
Lei et al. [147]	2021	CDG, MOEA/D, NSGA-II**	max profit, min idle time	MO-ND	Disassembly sequence, robot, performance measurement	HI, IGD, Epsilon
Wang et al. [102]	2021	MOABC, MOPSO, NSGA-II, SPEA-2**	**min makespan and min energy consumption**	MO-ND	Disassembly scheme (example output of sequence, allocation, time)	HI, IGD, Spread ++
Mei and Fang [87]	2021	DQN, D-DQN, PRDQN	**min idle time, high demand priority, min energy consumption**	MO-ND	Performance measurement	HI, IGD
Tseng et al. [148]	2022	PSO, GA, ACO**	min total make span	SO	Objective results	Objective results***
Zeng et al. [138]	2022	IGSA, NSGA-II, NSGA-III, SPEA-2, EMOGA, MOABC**	**min cycle time, energy consumption, smoothness index, max profit**	MO-ND	Disassembly sequence, robot workstation, performance measurement	HI, Spread, Pure Diversity, DV++
Zhou and Bian [78]	2022”	MBOHHO, NSHHO, MOPSO, MOEO, MOGWO**	**min cycle time, min energy consumption**	MO-ND	Performance measurement	POSs, GD, SS, IGD**** (statistic descriptive and statistical test one way ANOVA for mean value)
Yin et al. [77]	2022	MILP and HDA, NSGA-II, NSGA-III, PDSA-EA**	**min cycle time, peak energy consumption, total energy consumption, improved hazardous index**	MO-ND	Task and robot assignment, performance measurement	HI**** (t-test)
Laili et al. [149]	2022	GA, PSO, BA, MOEA, MOEA/D, IBEA**	n.a.	SO, MO-ND	Description of Evolutionary optimisation to solve RDSP and RDLB	n.a.
Laili et al. [124]	2022	Mathematical model	n.a.	n.a.	Mathematical representation	n.a.
Laili et al. [125]	2022	NSGA-II, IBEA, MOEA/D, PBEA**	**min time, min total energy consumption, min peak workstation energy consumption, the number of robots**	MO-ND	Performance measurement	HI**** (and statistic test wilcoxon rank sum test on HI value)
Zhang et al. [150]	2022	Tabu search**	max profit	SO	Disassembly objective results	n.a.
Zhang et al. [151]	2022	IMMO, NSGA-II, MOEA/D, MOCGA**	**max profit, min carbon emissions**	MO-ND	Disassembly sequence, performance measurement	HI, IGD****
Laili et al. [152]	2022	IBEA, MOEA/D, NSGA-II, NSGA-III, BCE-MOEA/D, BCE-IBEA**	**balance, direction change, cost, number of hazardous task, energy cost, line efficiency, total profit**	MO-ND	Performance measurement	HI, IGD, D-metric**** (chi square and p value friedman test)
Xu et al. [153]	2023	PIMBO, MODGWO, MOABC, NSGA-II, MOEA/D**	**max profit, min energy consumption, max balancing rate**	MO-ND	Disassembly sequence, performance measurement	C-metric, HI, IGD
Qin et al. [154]	2023	IMMO, MOCGA, MOEA/D, NSGA-III**	**max profit, min carbon emissions**	MO-ND	Performance measurement	HI, IGD, (ϵ-indicator)
Liu et al. [49]	2023”	IDBA, EDBA, GA, PSO**	min cycle time, min number of workstation, min smoothness index	MO	Disassembly sequence, direction, robotic workstation assignments, simulation	Iterations and population sizes***
Hartono et al. [129]	2023”*	MOBA, NSGA-II, PESA-II**	**max profit, energy savings, environmental impact reduction, min unbalanced line**	MO-ND	Disassembly sequence, direction, tools, recovery option, robot workstation, line balance, performance evaluation index	HI, NFE, POSs, PEI

Note: * thesis-derived publications (in press), ** metaheuristic, *** statistic descriptive, **** statistic test, ” sequence-dependent, highlighted bold = sustainability-related article, ++ misclassified: RDSP to RDLB.

2.4 Optimisation Algorithms in RDSP and RDLB

Numerous methodologies and algorithms have been developed to address the challenges posed by the disassembly line balancing and sequencing problem [155]. While mathematical and exact methods can provide optimal solutions for small-scale instances, their applicability to larger problems is limited by the NP-hard nature of the problem [41, 62, 63, 155]. A limited number of previous studies have utilised exact methods to address either simple or prototypical problems [77, 143]. It is important to note that exact solutions are currently unable to solve non-linear problems, and they are limited to addressing SO problems [77]. The number of possible subassemblies (N_n), the number of complete disassembly sequences (c_n) and the total number of disassembly sequences (P_n) in a disassembly problem can be determined theoretically using Eqs. (2.1) - (2.3) [156]. For instance, with 4 parts, there are 15 subassemblies, 15 complete disassembly sequences, and a total of 41 disassembly sequences. In the case of 10 parts, the number of subassemblies increases to 1023, the complete disassembly sequences amount to 34,459,425, and the total number of disassembly sequences reaches 314,726,297. These theoretical calculations demonstrate the exponential growth in the number of disassembly sequences as the number of the products increases. Such insights shed light on the combinatorial nature of disassembly problems and underscore the challenges associated with exploring all possible disassembly sequences.

$$N_n = (2.N_{n-1}) + 1 \tag{2.1}$$

$$c_n = (2n - 3).c_{n-1} \tag{2.2}$$

$$
\begin{aligned}
P_1 &= 1 \\
P_2 &= \frac{1}{2}\binom{1}{2} \cdot P_1 + 1 \\
P_3 &= \binom{1}{3} \cdot P_2 + 1 \\
&\vdots \\
P_{10} &= \binom{1}{10} \cdot P_9 + \binom{2}{10} \cdot P_2 \cdot P_8 + \binom{3}{10} \cdot P_3 \cdot P_7 + \binom{4}{10} \cdot P_4 \cdot P_6 + \frac{1}{2}\binom{5}{10} \cdot P_5 \cdot P_5 + 1
\end{aligned}
\tag{2.3}
$$

Another theoretical calculation suggests that in the context of planning the optimal sequence for a product with n parts, the exploration of the search tree involves examining $n!$ nodes, and the collision checking operation requires k x $n!$ computations, where k represents the average number of directions tested for component removal [119]. These theoretical calculations, coupled with the findings from previous research, provide compelling evidence to support the assertion that the disassembly problem is inherently complex and exhibits exponential growth in complexity. Metaheuristic algorithms have garnered significant attention from researchers due to their effectiveness in navigating the expansive search space of disassembly problems and achieving near-optimal solutions [41, 62, 64, 155].

The prevalence of metaheuristic approaches is evident in the research positions presented in Tables 2.1 and 2.2, where many researchers have adopted them. Notably, metaheuristic algorithms offer the advantage of relatively short computation times, meeting practical requirements [155] and facilitating efficient decision-making in real-world scenarios [79]. In a real-world context, the efficient discovery of near-optimal solutions holds greater significance than achieving exact solutions. Moreover, real products exhibit complexity that further amplifies the challenge. During

the initial stages of research, many studies employed toy problems to explore the mathematical and conceptual aspects, which is understandable considering the inherent complexity of the disassembly problem.

In the field of RDSP, the genetic algorithm (GA) and the bees algorithm (BA) are among the commonly used metaheuristics. Similarly, in RDLB, the GA is the most widely adopted approach. While BA is less frequently utilised in RDLB, it has demonstrated success in solving complex real-world problems [157–159]. The BA, developed in 2005 [160, 161], draws inspiration from the foraging behaviour of honeybees. In this analogy, each potential solution corresponds to a food source. A colony of bees, consisting of scout and forager bees, is used to conduct the search. Scout bees perform the initial exploration by randomly exploring the solution space and evaluating solutions based on objective functions, which are subsequently ranked by cost. Forager bees are then deployed to explore the vicinity of the higher-ranking solutions. In the context of the Bees Algorithm, the solution neighbourhood is commonly referred to as a 'flower patch.' The waggle dance observed in honeybees used to allocate a higher number of forager bees to the best solutions and fewer foragers to the other flower patches. Further details on the BA mechanism can be obtained from references [159–161].

The BA has demonstrated its robustness and effectiveness in a wide range of remanufacturing applications [162–165], establishing it as a reliable optimisation approach within this domain. In particular, in the field of robotic disassembly, the BA is recognised as one of the most popular metaheuristic algorithms [44, 47–49, 51, 54, 79, 86, 88, 89, 126, 149]. Consistently outperforming other algorithms, the BA is highly regarded for its exceptional performance and capabilities. Thus, it remains a compelling choice for addressing optimisation problems in robotic disassembly. It is apparent that previous studies utilising the BA in robotic disassembly have not specifically focused on reducing the parameter settings, with only one study reduce the parameter by Laili et al. [44]. This represents a significant gap in the literature, as optimising the parameter settings is crucial for achieving efficient and effective robotic disassembly processes. The parameter settings have

a direct impact on the performance of the algorithms and play a critical role in determining the quality of the solutions obtained. Addressing this gap presents valuable opportunities for enhancing the efficiency and effectiveness of the optimisation tools used in robotic disassembly. In this thesis, the BA is enhanced in two ways. Firstly, it is adapted from Enhanced Discrete Bees Algorithm (EDBA) introduced by [54] to specifically address the challenges of the RDSP and RDLBSD within the context of a sustainability model, considering both SO and MO-ND. Secondly, an improved version of BA is developed, which reduces the number of parameters by drawing inspiration from the life of honeybees. Chapter 5 will also discuss the original Bees Algorithm. This is to enable a comparison between the original version and the new version developed in this study, offering a detailed understanding of the adaptations and enhancements made, to achieve the last objective of this thesis.

2.5 Performance Evaluation

Metaheuristics have been criticised for their parameter settings despite their capability of producing near-optimal solutions faster than exact methods. Furthermore, population-based metaheuristics require establishing suitable population sizes, thereby increasing the problem's complexity. Although some researchers have utilised design experiment techniques, such as Taguchi [83] and Design of Experiments [166–168], to discover optimal parameter settings, this approach may be time-intensive and dependent on the problem. Furthermore, the fair comparison of different metaheuristics is still a topic of debate. The use of the same NFE as a stopping criterion is a commonly adopted method for fair comparison, as supported by existing literature (see e.g. [169]). Additionally, comparing metaheuristics using the same iteration number and population sizes is a frequently employed approach [47, 49, 51, 54]. Another crucial aspect in metaheuristics is the evaluation of performance. Performance evaluation is an important aspect of algorithm

development that aims to assess the performance of different algorithms by comparing them to other algorithms or benchmarks [70, 152].

In the context of metaheuristic algorithms, performance metrics such as solution quality, computational time, and convergence speed are commonly used to evaluate their effectiveness. Typically, the quality of a solution is determined by calculating its percentage deviation from the best-known solution [148]. While determining the best known solutions for simple problems is relatively straightforward, as exact methods can be used to find the optimal solutions, determining the best-known solutions for complex problems, particularly those involving nonlinear functions, becomes increasingly challenging [77]. Computational time is another frequently used performance metric to evaluate the metaheuristic performance [48, 54, 86, 98, 113]. However, it is important to note that this metric is dependent on the specific computer used to run the algorithm. As a result, it is common to see researchers report the specifications of the computer they used for their experiments. This variability in hardware specifications can make it difficult for other researchers to compare results across studies. Moreover, with the increasing use of cloud-based or GPU-based computing, this metric may not be appropriate or sufficient in all cases. Typically, the fitness value is plotted against the number of iterations to determine the convergence speed of a metaheuristic algorithm [148]. This generates a curve that can indicate the rate at which the algorithm converges on the optimal solution. The definition and interpretation of convergence speed can vary based on the different priorities of researchers and the complexities of the problem being solved. Complex problems with numerous variables and nonlinear functions have more challenging convergence definitions than simple problems with few variables. The NFE is another common metric for evaluating metaheuristic algorithms. This metric measures the minimum number of times the algorithm evaluates the objective function to locate a near-optimal solution. A smaller NFE required to identify the best near-optimal solution indicates that the algorithm is more effective at locating the optimal solution. However, NFE alone may not be sufficient as a performance measure since the effectiveness of a metaheuristic algorithm is heavily dependent on its parameter

settings.

SO and MO performance evaluation methods differ due to the nature of the solutions involved. In SO optimisation, the objective is to identify a single optimal solution that maximises or minimises a specific objective function. This approach is suitable when a clear and well-defined objective exists, and the problem can be adequately represented by a single criterion. It is worth noting that in the field of RDSP, the majority of previous research has focused on SO optimisation, as evidenced by the observations in Table 2.1. However, in recent years, there has been a growing trend towards the adoption of MO approaches. In contrast, the research on RDLB places greater emphasis on the utilisation of MO-ND approaches. This is attributed to the inherent complexity of the problems involved, which require the balancing of multiple objectives within the disassembly line. It is important to note that in some articles, the term ”multiobjective approach” refers to a specific type of approach known as the MO aggregate approach. This approach involves combining multiple objectives into an SO using methods such as simple addition or weighting techniques, assuming a linear relationship between the objectives (see [47, 49, 54]). In this thesis, this type of approach is referred to as MO.

On the other hand, the multiobjective nondominated approach, referred to as MO-ND in this thesis, is distinct from both the SO and MO approaches. It explicitly addresses the trade-offs between objectives and aims to generate a set of solutions that accurately represent the characteristics of MO-ND. Instead of aiming for a single solution, a set of solutions is generated that captures the trade-offs between the different objectives. These solutions, known as Pareto optimal solutions (POSs), constitute the Pareto front or Pareto set. Each solution in the Pareto front is considered nondominated, meaning it cannot be improved in any one objective without compromising another. Therefore, evaluation of performance in MO is more complex compared to that of SO optimisation because the output comprises of a set of solutions as opposed to a solitary solution [97, 170].

Prior studies have employed various metrics, such as POSs, Hypervolume Indicator

(HI), number of function evaluations (NFE). The number of POSs is one of the most frequently indicators for assessing convergence speed of MO-ND [171]. The HI serves as an indicator that measures both convergence and solution diversity [97, 172] and has become a standard performance metric [173]. A higher HI is desirable as it indicates a broader range of POSs [170]. The NFE is a speed indicator in optimisation algorithms [170, 174]. A lower value is considered better, as it signifies that the algorithm can reach the optimal solution with fewer steps or function evaluations. The NFE is a metric that can be applied to both SO and MO-ND optimisation problems. It serves as a measure of algorithm efficiency, indicating the ability to obtain satisfactory results with fewer computational steps. A lower NFE value suggests that the algorithm is capable of achieving desired outcomes using a smaller number of function evaluations. It is apparent that in previous studies, researchers have employed a range of different metrics. However, these metrics have been predominantly examined and analysed individually, focusing on their individual characteristics rather than considering their potential interactions or collective impact.

In the context of analysing the output of performance evaluation, previous research has relied on the use of descriptive statistics and the utilisation of visual tools such as box plots and histograms (see [49, 51, 114, 148]). These visual representations serve as effective means to present key findings, including measures such as the mean, standard deviations, and the maximum and minimum values of the objective function achieved by the algorithm. Nonetheless, this statistic descriptive does not provide a complete evaluation of the performance of the algorithm. To address this concern, a number of researchers have employed statistical analyses, primarily focusing on the results derived from objective value measurements e.g. [43, 79, 113]. The statistical test for comparing ultimate results, such as fitness value, run times, and individual performance metrics, is straightforward; however, it only indicates statistical differences in the end results. One study, by [43] introduced an indicator (ε-indicator) without a detailed explanation, but it appears to have used non-parametric statistical techniques to rank the results. The indicator paired each pair of algorithms and ranked them based on the average indicator results. Additionally, the [43] study

reported that there were conflicting results between the HI and the Inverted Generational Distance (IGD).

The identified gaps provide strong motivation to achieve objective 5 of this thesis, which involves determining optimal parameter configurations and facilitating comparisons among different algorithms. To address these gaps, Chapter 3 introduces two important methods: the statistical performance metric (SPM) and the performance evaluation index (PEI). By incorporating the SPM and PEI, this research contributes to the development of robust and efficient solutions for robotic disassembly while enhancing the reliability and validity of the research findings. Furthermore, these methods have broader applicability beyond robotic disassembly and can be utilised in other metaheuristic algorithms. Chapter 3 provides a detailed discussion and application of these methods. The application spans across Chapters 4 and 5, highlighting their significance in achieving the objectives of the thesis.

2.6 Literature Connections: Bibliometric Approach

This section provides a synthesis of the literature reviews on RDSP and RDLB, employing bibliometric analysis to show their interrelations. The relationship among the collected literature is revealed through the utilisation of VOSviewer, a bibliometric analysis tool [175–177]. Various types of analysis can be conducted using this tool. One notable analysis is the keyword analysis, which examines the occurrence of keywords in the dataset. Out of a total of 378 keywords, 50 keywords meet the predefined threshold, as depicted in Figure 2.1. These keywords are classified into four clusters based on the relationships identified by VOSviewer. The clusters include disassembly sequence, disassembly line balancing, robot, and robot system. Additionally, the trend shift in research focus towards energy-related aspects, multiobjective optimisation, and the utilisation of AI techniques such as deep learning and reinforcement learning is evident when

examining the timeline representation in Figure 2.2. Another insightful analysis is the co-citation with unit analysis, which focuses on the cited authors. Among the 1389 authors identified in the literature forty-five meet the minimum citation threshold of 20 for co-citations. Figure 2.3 highlights four prominent clusters of authors, with SM Gupta being the most frequently cited author (with 159 citations), followed by DT Pham (114 citations), Q Liu (78 citations), and MC Zhou (74 citations). In terms of geographical distribution, the research on RDSP and RDLB are primarily led by the United Kingdom (UK) and China, indicating their significant contributions in this field and their connections through joint research. Furthermore, it is noteworthy that MO optimisation has predominantly been employed in the context of RDLB rather than RDSP. The significance of optimisation techniques as useful tools for addressing the complexities involved in robotic disassembly problems is highlighted by this observation, which confirms the findings of the research positions analysis.

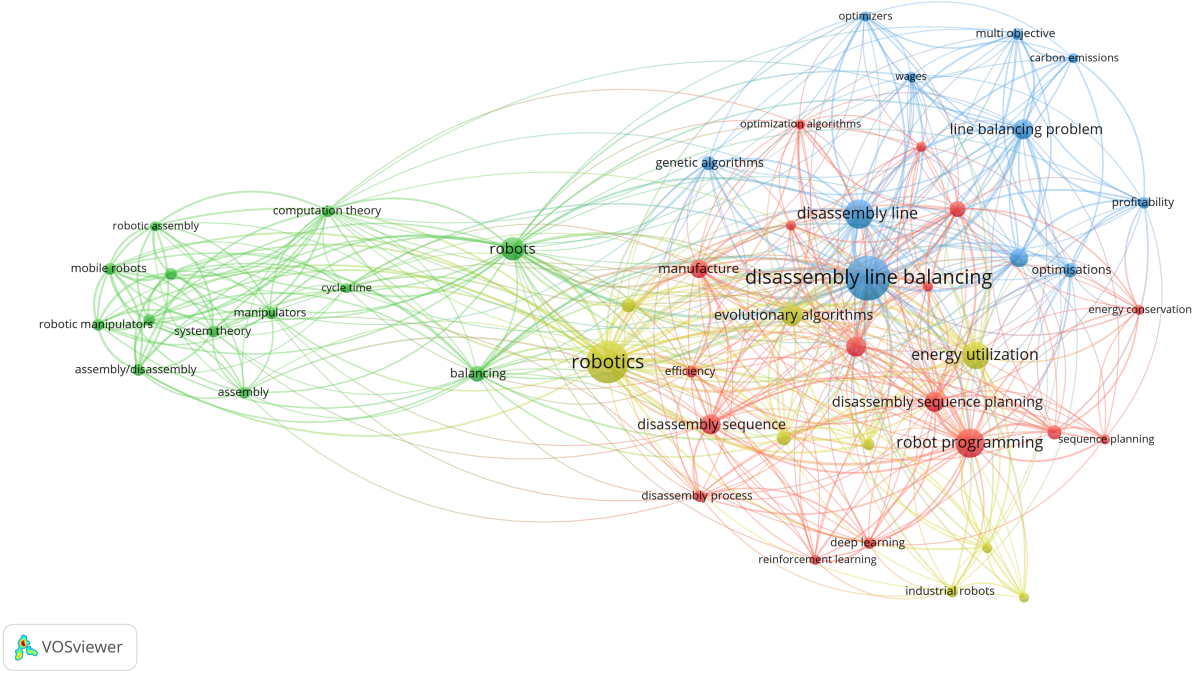


Figure 2.1: Keywords Analysis

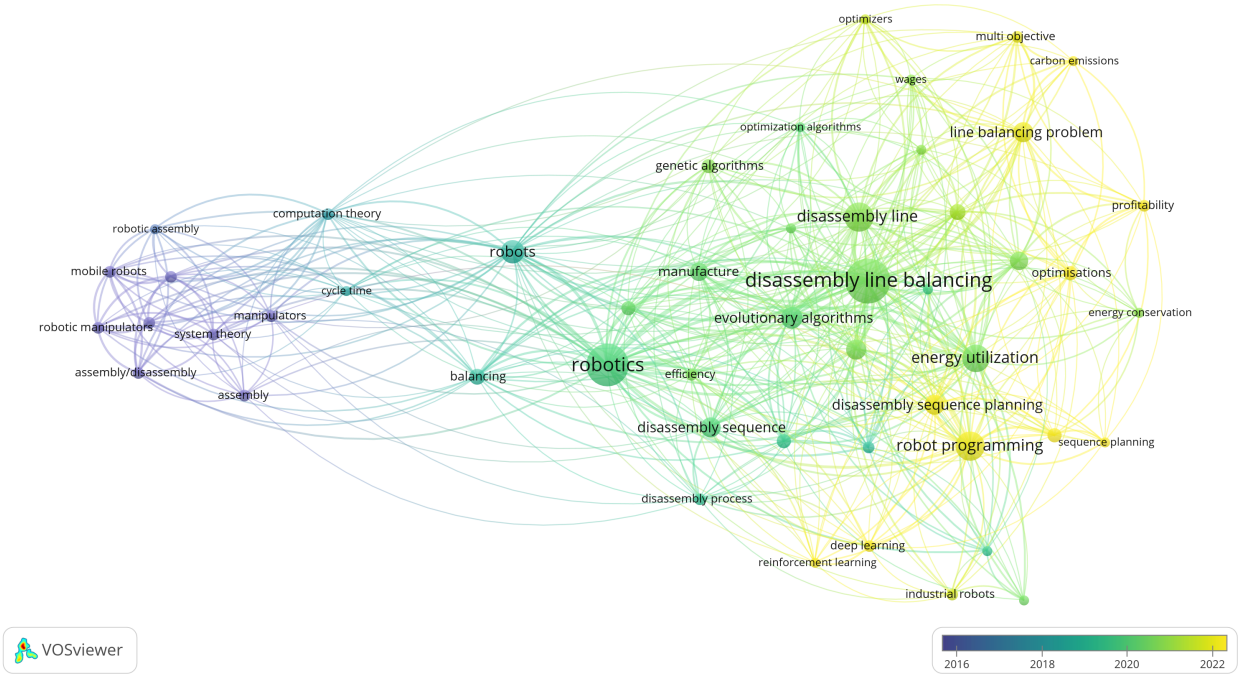


Figure 2.2: Keywords Analysis with timeline

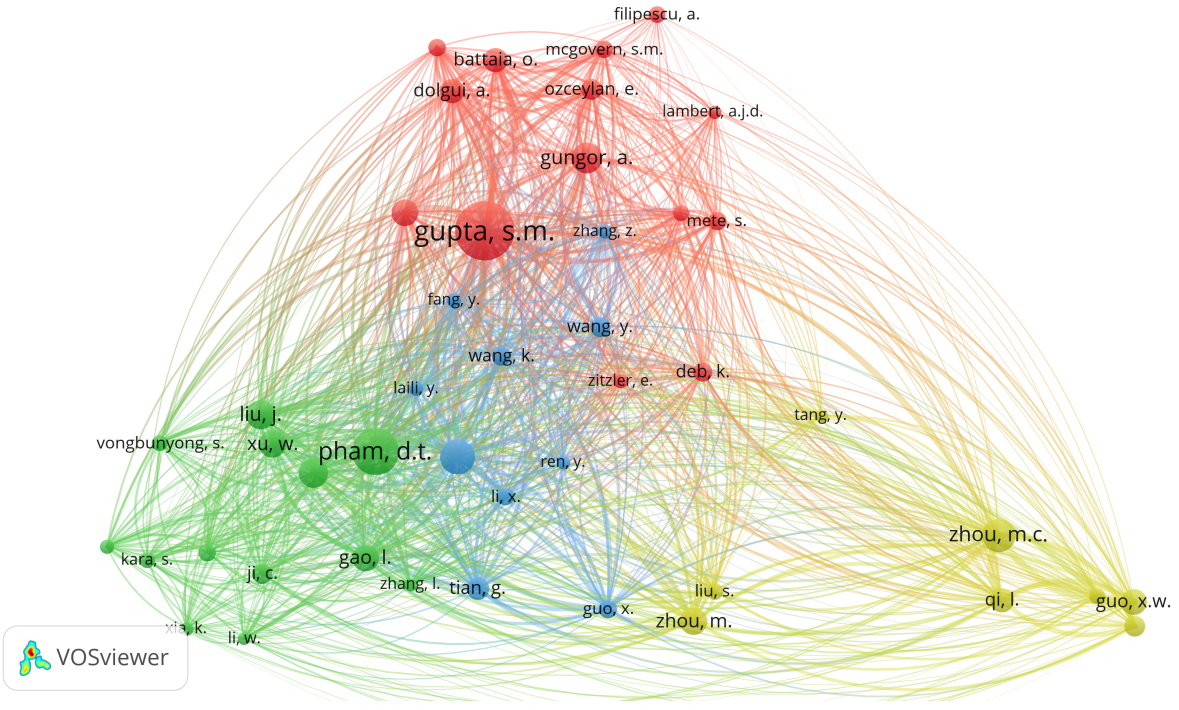


Figure 2.3: Co-citations Analysis

2.7 Summary

This chapter provides a comprehensive analysis of the distinctions and interrelationships between RDSP and RDLB through the use of a research position table and bibliometric analysis. The analysis conducted in this chapter encompasses multiple aspects, including the approach, objectives, methodology, output, performance measurement, and research trend. By outlining the research positions in RDSP and RDLB, this chapter highlights the specific contributions of this thesis to addressing these gaps. Firstly, a notable gap identified in the RDSP is the lack of consideration for sustainability, which aligns with objective 1. Similarly, only half of the research in RDLB addresses this crucial aspect, revealing another gap that aligns with objective 1. Additionally, a clear trend of research on sequence-dependent RDLB is observed, which aligns with objective 3. Furthermore, the literature lacks comprehensive reporting of disassembly output, particularly regarding recovery options for each component and the absence of algorithms to determine the optimal recovery choices, which correspond to objective 1. The case study of gear pumps serves as a validation of the proposed approach and aligns with objective 4. Moreover, the limited application of MO-ND optimisation approaches in RDSP studies reveals a significant gap that aligns with objective 2. Furthermore, previous research has overlooked the potential for improving the bees algorithm (BA) by reducing its parameter settings in robotic disassembly, highlighting a clear gap that corresponds to objective 6. Lastly, the review assesses the current approaches and gaps in the performance evaluation of optimisation algorithms within the context of robotic disassembly. The underutilisation of statistical tests and the absence of a unified performance evaluation metric across previous studies represent gaps that align with objective 5. These comprehensive analyses lay the foundation for addressing these gaps and advancing the field of robotic disassembly. The objectives of this thesis, as outlined in Chapter 1, directly align with the identified gaps and serve as a road-map for the subsequent chapters. As previously mentioned,

Chapters 3 and 4 study RDSP and RDLB, addressing objectives 1, 2, 3, 4, 5. Chapter 5 focuses on the enhancement of the Bees Algorithm, addressing objectives 2, 4, 5, and 6.

Chapter 3

Robotic Disassembly Sequence Planning

The increasing adoption of robotics in disassembly processes aims to enhance their effectiveness and efficiency compared to manual disassembly. Within this context, robotic disassembly sequence planning (RDSP) has emerged as a critical area for improving efficiency and cost-effectiveness in disassembly operations. This chapter specifically addresses the research problem of RDSP, which involves determining the optimal order for disassembling parts and components within a robotic cell, thus addressing objective 2 of this thesis. In addition, this chapter also encompasses objectives 1, 4 and 5.

The challenge of determining the optimal sequence for robotic disassembly lies in its inherent complexity and nondeterministic polynomial (NP) nature. Traditional exact methods often struggle with computational intractability due to the NP-completeness of this problem. To overcome this challenge, the application of metaheuristic algorithms has gained prominence, offering significant advantages over exact methods. Over the past decade, metaheuristic algorithms have demonstrated their dominance in solving complex optimisation problems, including RDSP. As indicated in Table 2.1 in Chapter 2 literature review, metaheuristics have been the predominant approach since 2011. Therefore, the focus of this chapter is to utilise the bees algorithm (BA) as optimisation tools for solving the RDSP.

In addition, this chapter contributes to the existing literature by introducing a sustainability model that has not been previously explored, as depicted in Table 2.1 of the research position, thereby achieving objective 1 of this study. The sustainability model developed in this research is explained in detail, providing a comprehensive understanding of its key components and methodology. Furthermore, four sustainability recovery scenarios are outlined for each component, considering factors such as material reuse, recycling, remanufacturing, and disposal. One of the scenarios involves the use of an autonomous recovery strategy utilising the algorithm. Additionally, three of the scenarios are based on data collection and interviews conducted with remanufacturers in England and Spain. These scenarios aim to maximise resource recovery and minimise environmental impact, thereby contributing to the overall sustainability objectives of the disassembly process.

Objective 4 is addressed in this chapter by conducting a detailed case study on gear pumps to validate the proposed approach. The rationale for selecting this specific case study is thoroughly explained, providing a comprehensive understanding of the reasons behind this choice. Additionally, a comprehensive description of the gear pumps is provided, offering insight into the specific components and characteristics of the system. Furthermore, objective 5 is addressed in this chapter, focusing on determining the optimal parameter settings and performance metrics for optimisation algorithms in the context of robotic disassembly. To achieve this, the chapter introduces a novel statistical performance metric (SPM) and performance evaluation index (PEI), which serve as valuable tools for assessing the performance of optimisation algorithms and determining optimal parameter settings.

This research makes several significant contributions to the field of RDSP. First, the research utilises a multiobjective nondominated bees algorithm to concurrently optimise multiple objectives, considering trade-offs between different objectives. Second, the research introduces a sustainability model that has not been addressed in previous studies. Third, it introduces the novel concept of selecting the best recovery option for each disassembly component, which none of the previous

studies have addressed. This feature provides comprehensive guidance for the disassembly process and enhances the practicality and usefulness of the proposed model. Fourth, the research incorporates an autonomous recovery strategy identification mechanism utilises the algorithm, allowing for dynamic evaluation and selection of optimal recovery strategies based on given constraints and objectives. This autonomous decision-making capability improves the adaptability and effectiveness of the solution. Fifth, the research adopts a realistic simulation approach that closely replicates the real disassembly process, ensuring alignment with real-world scenarios. Reliable data collected from relevant remanufacturers adds credibility and strengthens the applicability of the proposed model. Lastly, two novel tools for performance measurement, statistical performance metric (SPM) and performance evaluation index (PEI), are incorporated to provide a comprehensive evaluation that combines statistical rigour and simplicity for decision-making purposes. These contributions collectively enhance the reliability, robustness, and practicality of the proposed solution for RDSP.

The overview of the contents of this chapter as follows. In the first section, Section 3.1, the chosen case study is presented along with a detailed justification for its selection. The proposed performance evaluation is elaborated upon in Section 3.2, providing a thorough explanation of its methodology. Section 3.3 explains the model and methodology used in this research. Section 3.4 presents the conducted experiments and the corresponding results, followed by a detailed discussion in Section 3.5. Finally, Section 3.6 concludes the chapter by summarising the key findings and outlining recommendations for practitioners and future research directions.

3.1 Case Study Description

The case study chosen for this research is focused on industrial gear pumps, which have been frequently used in previous studies on robotic disassembly of EoL products [47–49, 51, 54, 86,

122, 125, 128, 130]. Furthermore, the broad industrial application and low wear of gear pumps make them an ideal candidate for demonstrating the optimisation results and studying complete disassembly without destruction, highlighting their suitability as a test subject in this research. The results of this study on the robotic disassembly of gear pumps contribute to the broader context of robotic disassembly research, as they demonstrate the potential of using optimisation algorithms to disassemble EoL products efficiently and sustainably.

3.1.1 Case study: industrial gear pumps

A gear pump is a hydraulic pump variant that comprises two gears that are enclosed within a compact housing. The process involves the conversion of motor-generated kinetic energy into hydraulic energy through the flow of oil generated by the pump. External gear pumps are widely utilised in industrial applications due to their compactness, high power output, durability, and cost-effectiveness. The utilisation of pressurised oil flow is a common method for inducing motion in the actuator that is integrated within a given machine or application. The primary part of the pump is the gear pair that is coupled together. The gear pair comprises of two shafts, namely the drive shaft, which is powered by the motor shaft, and the driven shaft. The principle of displacement, which is caused by the contact between the teeth of the shaft gears, results in the rotation of the driven shaft by the driving shaft. Upon activation of the pump, oil is drawn into the inlet (suction) orifice as a result of the pressure differential generated by the disengagement of the teeth of one gear from those of the other. The transportation of oil occurs through the flanks of the gear teeth until it reaches the outlet orifice of the pump. At this point, the oil is propelled towards the outlet orifice, or experiences a pressure, as a result of the interaction between the teeth of the driving and driven shafts. The gear pump is a subject of significant interest for remanufacturers in the context of end-of-life product recovery. This is because certain components of the gear pump exhibit minimal wear and tear after prolonged use, rendering them suitable for reuse or

remanufacturing in new products. Gear pumps are a viable sustainable alternative, as they aid in waste reduction and encourage the implementation of circular economy principles. Moreover, the adaptability of gear pumps enables their utilisation in a diverse array of applications, spanning from hydraulic systems to fuel transfer. The remaining parts may be either recycled or disposed of as a last resort.

The two gear pumps depicted in Figures 3.1 and 3.2 have different flow rates; Gear Pump A has a flow rate of 7.5 l/min, while Gear Pump B has a flow rate of 10 l/min. The data utilised in this study were derived from academic sources, including the research conducted by [54] and [122], in addition to 3D models acquired from Grabcad [178]. Furthermore, the perspectives of expert remanufacturers based in the United Kingdom and Spain were collected for this study. As previously mentioned regarding the disassembly distances, the path distance (PD) matrices for Gear Pumps A and B depict the proximity between adjacent disassembly points while considering collision avoidance computations and all potential routes, including impractical trajectories that limit the choice of prohibited paths during the disassembly procedure. The input data provided in Appendix A.

3.1.2 Key input data and calculation assumptions in this thesis

In order to ensure the appropriate conduct of the case study, it is necessary to make several assumptions. The aforementioned assumptions are relevant to diverse facets of the disassembly procedure, encompassing types of disassembly, task times, the operation of the remanufacturing companies, the expenses incurred, and the robot configurations. The lists are as follows:

- The disassembly procedure is sequential, with operations performed one at a time.
- The disassembly is a complete disassembly, indicating that the entire product is broken down into its component parts.

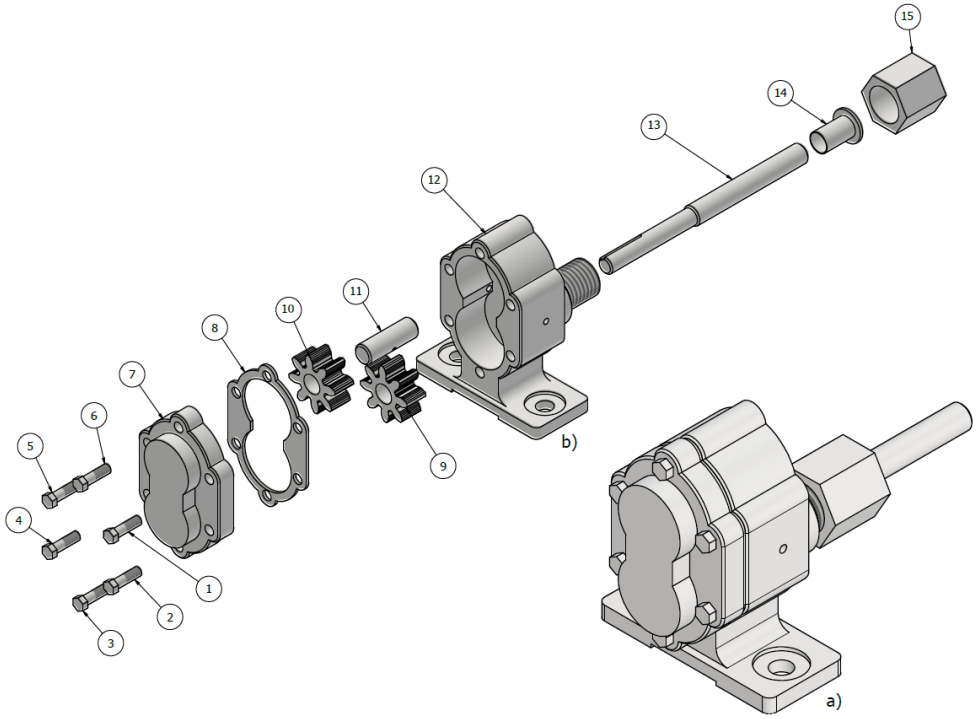


Figure 3.1: Gear Pump A: (a) assembled view; (b) exploded view.

- The disassembly procedure of the components involves non-destructive actions, which presupposes that the robot is capable of executing conventional tasks such as rotation, unscrewing, detachment, gripping, and other similar actions to accomplish all disassembly tasks.

- The task times are known and deterministic. Hence, the robot completes the same operation in the same amount of time for all disassembled components.

- It is presumed that the remanufacturing company operates for a single 8-hour shift each day, for a total of 220 working days per year.

- Given that the robotic cell operates with just one type of gear pump throughout the year, a projection of 70,000 units annually for Gear Pump A and 55,000 units per annually for Gear Pump B is assumed. The data utilised in this study is derived from remanufacturers of gear pumps located in the United Kingdom.

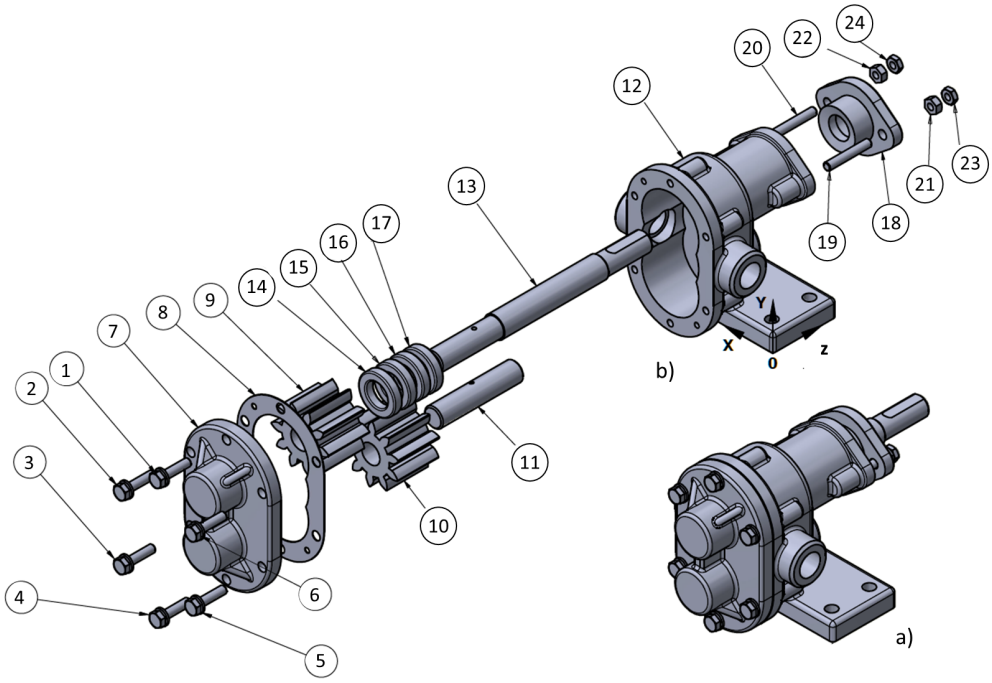


Figure 3.2: Gear Pump B: (a) assembled view; (b) exploded view.

- According to a commercial quotation from a robot manufacturer, the robotic cell's upfront costs (investment) are estimated to be 0.15 M€, and its hourly cost is 120 €/h.

- The study assumes a straight-line depreciation of machinery over 10 years.

- The allocation of overhead expenses is based on the utilisation of resources for individual disassembly procedures. As per [122]'s methodology, a scoring system has been employed to determine the appropriate treatment for various components. The components to be reused are assigned a weight of 2 out of 10, while those to be remanufactured are given a weight of 5 out of 10. Similarly, components designated for recycling and disposal are assigned weights of 2 out of 10 and 1 out of 10, respectively.

- According to the manufacturer's specifications [179], the linear velocity of the robot's end-effector is 12 mm/s, and it will take 10 seconds for the robot to change the tool in the tool magazine (M).

- The penalty times, p_1 and p_2, for changing the direction of a process are assumed to be 1 and 2 seconds, respectively.

3.2 Proposed Performance Evaluation

The literature review reveals that prior research has not fully utilised the potential of statistics, as they were only applied to the final outcomes. In addition, contradictory results have been observed across various performance metrics, see Chapter 2.5. To bridge these identified gaps, two novel approaches are proposed in this thesis. One approach involves the application of statistical tests, while the other utilises a straightforward yet versatile metric. These proposed methods aim to address shortcomings and provide valuable insights into the performance analysis of the algorithm. The specifics of the first approach are elaborated upon in Section 3.2.1, while the details of the second approach are provided in Section 3.2.2.

3.2.1 Proposed Statistical Performance Metric

The proposed method, SPM, employs a statistical test not only for evaluating algorithm performance but also for identifying optimal parameter settings. This method follows a well-defined selection process for the appropriate statistical test. The advantage of this approach lies in its ability to statistically analyse observed differences and determine the optimal parameter settings for the chosen algorithm. The methodology is outlined as follows: Initially, the results are visualised using descriptive statistics. Subsequently, the assumption checklist is performed. If the number of experiments conducted exceeds 30, it is necessary to test whether the data adhere to certain assumptions, such as normality and homogeneity. In the case of any assumption violations, a nonparametric test is carried out. Conversely, if the number of experiments is below 30, a

nonparametric test is employed. Finally, if the results indicated statistical significance, a post hoc test was conducted. The decision process can be observed in the Algorithm 1. The choice of software for conducting the statistical analysis is dependent on the researcher's preference and may involve the use of either commercial or freely available statistical software. In this thesis, the statistical tests were performed using IBM SPSS Statistics 27 and MATLAB 2020b. The SPM introduced in our publication [1].

By employing this method, researchers can rigorously analyse the performance of an algorithm and identify the optimal parameter settings in a statistically sound manner. This systematic approach adds credibility to the experimental results and contributes to the advancement of algorithm optimisation techniques.

3.2.2 Proposed Performance Evaluation Index

As previously mentioned, relying solely on diagrams or figures to illustrate these indicators is insufficient, as it neglects the possibility of conflicting indicators, an aspect that has not received sufficient attention in scholarly investigations. Evaluating the performance of MO optimisation presents greater complexity compared to SO optimisation, as it entails analysing a set of solutions rather than a singular solution as previously discussed. As previously mentioned, there is a possibility of conflicting results of indicators [43]. The second contribution of this research addresses this gap by introducing a performance index that has not been previously employed for evaluating algorithm performance. The concept behind the proposed performance evaluation index (PEI) is derived from formulas commonly used in the Multiple-Criteria Decision-Making (MCDM) literature. The index provides decision makers with a valuable tool for expediently evaluating multiple criteria and facilitating prompt decision-making. To calculate the index, two approaches are discussed in the MCDM literature. The first approach involves assigning weights to different criteria and summing the scores to obtain an overall score. However, this approach

Algorithm 1: Statistical Performance Metric

```
Require: data
Function PerformDescriptiveStatisticsAndBoxplot(data):
    Calculate descriptive statistics for the data;
    Generate a boxplot to visually represent the results;
Function PerformAssumptionChecklist(data):
    Check the data size;
    if the data size is below the threshold (e.g., 30) then
        Perform a nonparametric test;
        Choose an appropriate nonparametric test;
        return;
    end
    Test for normality;
    Test for homogeneity of variances;
Function DecisionStepOne(data):
    assumptionFail ← false;
    PerformAssumptionChecklist(data);
    if any assumption fails (data size, normality, or homogeneity) then
        Perform a nonparametric test based on the failed assumption;
        assumptionFail ← true;
    end
    if all assumptions are met and assumptionFail is false then
        Perform a parametric test;
    end
Function DecisionStepTwo(result):
    if result of the test is statistically significant then
        Conduct post hoc tests to determine specific group differences;
        return;
    end
    return;
Main Algorithm;
PerformDescriptiveStatisticsAndBoxplot(data);
DecisionStepOne(data);
DecisionStepTwo(result);
```

has certain limitations, including the need for additional computations such as normalisation and the inadequate accommodation of conflicting objectives related to specific criteria maximisation or minimisation. To address these limitations, [180] introduced the second approach, known as the multiplicative approach. This approach resolves the limitations of the first approach by

considering the multiplicative combination of scores, resulting in a simplified computation process and improved accommodation of conflicting objectives. In contrast to the additive method, the multiplicative approach does not require the normalisation or re-scaling of criteria, as the final outcome is unaffected by these operations [180].

The multiplicative approach is followed by the proposed PEI methodology, whereby the PEI is derived through the multiplication of indicators that are desired to have higher values and the division of indicators that are preferred to have lower values. Equal weights (ω) were assigned to all the functions, as all the indicators were deemed to be of equal significance. It should be noted that the value of ω is subject to the discretion of decision makers, who may set it based on their individual preferences.

In this example, a common MO-ND performance metric is considered: the Hypervolume Indicator (HI), the Pareto optimal solutions (POSs), and number of function evaluations (NFE). A higher HI is desirable because it signifies a wider range of POSs. Having a higher number of POSs is also desirable. On the other hand, the NFE serves as a reliable measure of computational complexity and is independent of the computer system. In this case, a lower NFE is preferred. The mathematical expression is represented by Equation (3.1) as follows:

$$PEI = [HI^{\omega_1} POSs^{\omega_2}]/NFE^{\omega_3} \tag{3.1}$$

The PEI is a versatile metric that can be tailored by researchers to align with their preferred evaluation criteria through the modification of equations. The addition of supplementary metrics can be incorporated into the equations based on whether higher or lower values are desired. As previously mentioned, metrics with higher desired outcomes are included in the numerator, while those with lower desired outcomes are placed in the denominator. By consolidating multiple performance indices into a single metric, this method streamlines the evaluation process and offers valuable insights for decision-making in complex optimisation scenarios. In addition, the PEI can be used to evaluate the performance of MO as well as SO.

3.3 Model and methodology

The sustainability model for RDSP developed in this research comprises of these steps: model building, model formulation, optimisation methods using single-objective (SO) and multiobjective (MO) aggregate approaches, optimisation using the MO-ND approach, and performance measurement. In the first step, model building, the necessary frameworks and structures for the RDSP decision-making model are constructed. This involves the collection of relevant input data, including product information, component properties, and recovery feasibility, obtained from CAD designs and collaborations with remanufacturers in England and Spain. The gathered data plays a crucial role in evaluating the interference between disassembly parts and establishing precedence relationships. The primary objective is to eliminate infeasible sequences and ensure an optimised disassembly process. The insights and input from remanufacturers are particularly valuable in establishing sustainable recovery strategies and acquiring data for the defined objectives. In the subsequent step, model formulation, a comprehensive explanation is provided regarding the sustainable objectives and recovery strategies. This stage entails the precise definition of specific objectives that contribute to the sustainability of the disassembly process, considering profit, energy savings and environmental impact reduction. The model formulation integrates these objectives, creating a comprehensive framework that serves as the basis for decision-making. The third step focuses on the application of optimisation methods to the RDSP decision-making model, employing both SO and MO aggregate approaches. Initially, SO optimisation is utilised to determine the maximum value of each individual objective. Subsequently, an MO aggregate approach is employed to evaluate the maximum value, assuming linear relationships between the objectives and treating them as an SO.

The subsequent step encompasses the application of the MO-ND approach, which allows for the identification of optimal solutions that are not dominated by others within the objective space. By

considering multiple objectives simultaneously, the model achieves a comprehensive evaluation and generates a set of solutions that strike a balance between competing objectives, thereby fostering a more sustainable and balanced robotic disassembly process. As previously mentioned, this research primarily focuses on utilising the BA as the primary optimisation approach. The MOBA represents a nondominated and modified version of the EDBA, specifically tailored to address the complexities of multiple objectives and recovery options. To facilitate comprehensive comparisons and robust evaluations, two additional comparative algorithms, NSGA-II and PESA-II, are used as well-established algorithms for comparison. These algorithms serve as valuable reference points and benchmarks for assessing the performance and effectiveness of the MOBA in solving RDSP.

Lastly, the final step focuses on performance measurement. The research utilises the SPM and the PEI to effectively assess the performance of the optimisation algorithms under different parameter settings. These performance measurement tools provide robust statistical analysis and enable the systematic comparison and evaluation of the algorithms.

3.3.1 RDSP Model Building

The RDSP model building process begins with the collection of input data regarding the products, their components, properties, and their feasibility for recovery. Some of the data is gathered from remanufacturers located across England and Spain. The CAD design provides valuable information that is extracted and used as input for subsequent steps. The collected data is then utilised to evaluate the interference between disassembly parts and establish precedence relationships among them, with the goal of eliminating infeasible sequences [33]. Robotic disassembly presents unique challenges compared to manual disassembly, as discussed in Chapter 2. Collision avoidance plays a crucial role in robotic disassembly planning, affecting the disassembly time and trajectory of the robot's end effector. Previous studies have commonly neglected the product's contour when designing the robot's path. To address this, researchers have devised methods

to achieve collision-free robotic disassembly. These methods involve considering the object's geometry, calculating distances between disassembly points, and ensuring a minimum distance of 10 mm between the end effector's path and the contours of the end-of-life (EoL) product [49, 54]. Additionally, in contrast to manual disassembly, the model requires additional information regarding the disassembly direction to guide the robot's movements effectively. To address this, feasible disassembly sequences and directions are generated using the modified space interference matrix and interference matrix analyses, known as Modified Feasible Solution Generation (MFSG) [54]. These techniques, originally proposed by Jin et al. [181, 182], provide a comprehensive representation of disassembly precedence between components in six directions (X+, X-, Y+, Y-, Z+, Z-). The interference matrix (C), see Equation (3.2) [181, 182], plays a crucial role in capturing the blocking relationships between components in different directions. Each element in the matrix is a multidimensional vector, indicating whether part j obstructs the movement of part i along the X+, X-, Y+, Y-, Z+, or Z- direction. Specifically, if there is a blockage, the corresponding element (C_{ij}) is assigned a value of 1; otherwise, it is set to 0. It is worth noting that Jin et al.'s method considers the transpose of the positive direction matrix for the negative direction [181, 182]. However, this approach becomes problematic when dealing with fasteners such as bolts. In such cases, disassembling the components before removing the bolts is not feasible. To overcome this limitation, this thesis adopts the technique proposed by [54] as mentioned earlier. This technique involves analysing each matrix individually, ensuring that the disassembly process accounts for the presence of fasteners. By doing so, the model ensures that components are not disassembled before their corresponding fasteners, thus guaranteeing a realistic disassembly process. Additionally, the feasibility of disassembly direction is taken into account, particularly when the product incorporates fasteners, making it a viable approach for practical applications.

$$C = \begin{bmatrix} 0 & C_{12} & \cdots & C_{1n} \\ C_{21} & 0 & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \cdots & 0 \end{bmatrix} \tag{3.2}$$

Robotic Cell

In this thesis, a robotic cell illustration was created to represent the experiments in the RDSP setting. The robotic cell consists of a robot and a tool magazine (M) equipped with a robotic tool changer. Figure 3.3 depicts the layout of the robotic cell, highlighting the positioning of the robot, the tool magazine (M), and the selected gear (pump A or B) depending on the specific case study being examined.

The KUKA LBR iiwa R820 is a 7-axis lightweight robotic system equipped with a jointed arm, specifically designed to accommodate a maximum payload of 14 kg and an 820 mm reach [179]. Its operational efficiency is facilitated by a spacious working volume of 1.8 m^3 and repeatability of 0.15 mm (ISO 9283). The tool magazine (M) serves as an integral component of the robotic system, housing the necessary tools for executing disassembly tasks. It provides a convenient and organised storage solution for the tools required during the disassembly process. In cases where a tool replacement becomes necessary for the subsequent disassembly step, the robot is programmed to navigate to the tool magazine's designated position. In the simulated environment, the tool magazine (M) is located at coordinates x = 300 mm, y = 200 mm, and z = 150 mm. Unfastening and pulling/pushing are the two main categories of disassembly operations for the gear pump. Three types of spanners, spanner 1, spanner 2, and spanner 3, are used to effectively loosen bolts and nuts during the unfastening process. Gripper 1 and gripper 2 are used for the remaining disassembly operations. Appendix A contains detailed information regarding the specific disassembly tools

used for each operation, the corresponding coordinates of the disassembly points relative to the origin coordinates, and the time required to complete each disassembly operation.

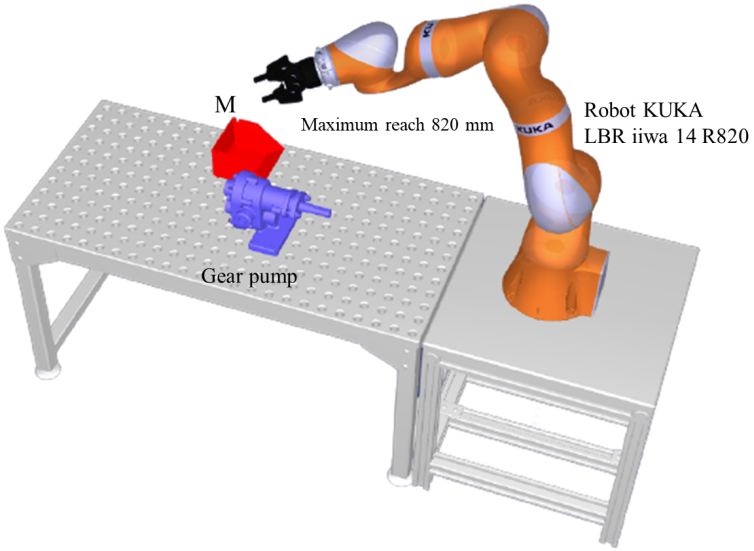


Figure 3.3: Layout of the robotic cell [1]

3.3.2 RDSP Model Formulation

The solution to the robotic disassembly problem within the sustainability model is approached as a MO problem. The primary objective is to find an optimal solution that effectively balances three key goals: maximising profit (f_1), maximising energy savings (f_2), and maximising environmental impact reduction (f_3).

Goal 1. Profit

Equation (3.3) [183] defines the first goal, which is profit, and comprises seven main components. These components include the revenue generated from the reuse or remanufacturing of components, the revenue obtained from recycling components, the costs associated with the disposal of

components, the overall disassembly cost, the expenses related to the recovery of components for reuse or remanufacturing, the company's overhead costs, and the depreciation costs of the machinery (robotic cell) used in the disassembly process.

$$
\begin{aligned}
f_1 = & \sum_{i=1}^{N_p}\sum_{j=1}^{2} RP_i\, r_{i,j}\, \alpha_i \;+\; \sum_{i=1}^{N_p} RC_i\, r_{i,3}\, \alpha_i \;-\; \sum_{i=1}^{N_p} CD_i\, r_{i,4}\,(1-\alpha_i) - \\
& - \Bigg[\sum_{i=1}^{N_p-1} t_b(x_i)\, \alpha_i\, c_T \;+\; \sum_{i=1}^{N_p-1} \Bigg(\frac{PD(x_i, M)}{v_e} \;+\; t_c(x_i, x_{i+1}) \;+\; \frac{PD(M, x_{i+1})}{v_e} \;+\; t_u(x_i, M) + \\
& \quad + \; t_w(M, x_{i+1}) \Bigg) \gamma_i\, \alpha_i\, c_T \;++\; \sum_{i=1}^{N_p-1} \left(\frac{PD(x_i, x_{i+1})}{v_e} \;+\; t_z(x_i, x_{i+1}) \right) (1-\gamma_i)\, \alpha_i\, c_T \Bigg] - \\
& - \sum_{i=1}^{N_p}\sum_{j=1}^{2} rc_{i,j}\, r_{i,j}\, \alpha_i - \sum_{i=1}^{N_p}\sum_{j=1}^{4} oh_{i,j}\, r_{i,j}\, \alpha_i - \sum_{i=1}^{N_p}\sum_{j=1}^{4} dp_{i,j}\, r_{i,j}\, \alpha_i
\end{aligned}
\tag{3.3}
$$

where:

- i is the index for each component and varies from 0 to N_p
- j is the indicator of the recovery mode and equal to 1 if component i is assigned to be reused, 2 if it is to be remanufactured, 3 if it is to be recycled or 4 if it is to be disposed of.
- RP_i is the revenue obtained due to the component i to be reused or remanufactured not having been manufactured again for a new product
- $r_{i,j}$ is an indicator of the recovery mode: 1 if mode j is assigned to component i
- α_i is an indicator that takes the value of 1 if component i is to be disassembled and 0 otherwise.
- RC_i is the revenue obtained from component i being recycled
- CD_i is the disposal cost of component i being disposed of
- $t_b(x_i)$ is the basic time to perform disassembly operation x_i
- c_T is the cost per unit of time
- $PD(x_i, M)$ is the distance between the point of the disassembly operation x_i and the position

of the tool magazine (M)

- v_e is the line velocity of the industrial robot's end effector
- $t_c(x_i, x_{i+1})$ is the tool change time and depends on the tool type
- $PD(M, x_{i+1})$ is the length between the position of the tool magazine (M) and the point of the disassembly operation x_{i+1}
- $t_u(x_i, M)$ is the penalty time for process direction changes along the path between x_i and the tool magazine (M) and formulated as follows:
  - 0 if the direction is not changed
  - p_1 if the direction is changed by 90º
  - p_2 if the direction is changed by 180º
- $t_w(M, x_{i+1})$ is the penalty time for process direction changes along the path between the tool magazine (M) and x_{i+1}, and is formulated as t_u
- γ_i is an indicator taking the value 1 if operation x_{i+1} requires changing the tool used in previous operation x_i
- $PD(x_i, x_{i+1})$ is the distance between the point of the disassembly operation x_i and the point of disassembly operation x_{i+1}
- $t_z(x_i, x_{i+1})$ is the penalty time for process direction changes along the path between x_i and x_{i+1}, and is formulated as t_u
- $rc_{i,j}$ is the recovery cost of component i being reused or remanufactured
- $oh_{i,j}$ is the overhead cost assigned to component i to be disassembled
- $dp_{i,j}$ is the depreciation cost assigned to component i to be disassembled

Goal 2. Energy savings

The energy savings, represented by f_2, are achieved through the disassembly process and subsequent recovery of components, as shown in Equation (3.4) [183]. By reusing or

remanufacturing some of the disassembled components, the model accounts for the energy saved by avoiding the production of new components for new products. This objective, f_2, comprises four key components: the total energy reclaimed from reused or remanufactured components, the energy consumed by the robot during the overall disassembly process, the energy consumed in recovering components for reuse, remanufacturing, or recycling, and the energy consumed in the final treatment of disposed components.

$$
\begin{aligned}
f_2 = \sum_{i=1}^{N_p}\sum_{j=1}^{2} r_{i,j}\, gr_{i,j}\, f_W\, \alpha_i \; - \; \sum_{i=1}^{N_p-1} \Big[& gd_{1,i}(x_i) \; + \; gd_{2,i}(x_i, M)\, \gamma_i + \; gd_{3,i}(M)\, \gamma_i + \\
& + \; gd_{4,i}(M, x_{i+1})\, \gamma_i \; + \; gd_{5,i}(x_i, x_{i+1})\, (1-\gamma_i) \Big]\, f_W\, \alpha_i - \\
& - \sum_{i=1}^{N_p}\sum_{j=1}^{3} r_{i,j}\, gc_{i,j}\, f_W\, \alpha_i \; - \; \sum_{i=1}^{N_p} r_{i,4}\, gc_{i,4}\, f_W\, (1-\alpha_i) \; = \\
= \sum_{i=1}^{N_p}\sum_{j=1}^{2} r_{i,j}\, gr_{i,j}\, f_W\, \alpha_i \; - \; \sum_{i=1}^{N_p-1} \Big[& t_b(x_i)\, PR_1\, \gamma_i \; + \; \frac{PD(M, x_i)\, PR_2\, \gamma_i}{v_e} \; + \; t_c(x_i, x_{i+1})\, PR_2\, \gamma_i + \\
& + \; \frac{PD(M, x_{i+1})\, PR_2\, \gamma_i}{v_e} \; + \; \frac{PD(x_i, x_{i+1})\, PR_2\, (1-\gamma_i)}{v_e} \Big]\, \frac{f_W\, \alpha_i}{3,600} - \\
& - \sum_{i=1}^{N_p}\sum_{j=1}^{3} r_{i,j}\, gc_{i,j}\, f_W\, \alpha_i \; - \; \sum_{i=1}^{N_p} r_{i,4}\, gc_{i,4}\, f_W\, (1-\alpha_i)
\end{aligned}
\tag{3.4}
$$

where:

- $gr_{i,j}$ is the energy reclaimed from component i being reused or remanufactured
- f_W is a conversion factor from kWh to monetary units
- $gd_{1,i}(x_i)$ is the energy consumption of the robot in the disassembly operation of component i
- $gd_{2,i}(x_i, M)$ is the energy consumption of the robot in the movement between the position x_i and M
- $gd_{3,i}(M)$ is the energy consumption of the robot in the tool change
- $gd_{4,i}(M, x_{i+1})$ is the energy consumption of the robot in the movement between M and x_{i+1}

- $gd_{5,i}(x_i, x_{i+1})$ is the energy consumption of the robot in the movement between x_i and x_{i+1}
- $gc_{i,j}$ is the energy consumption involved in recovering component i with mode j
- PR_1 is the power of the robot used in the disassembly operation
- PR_2 is the power of the robot used in the movements between the disassembly points

Goal 3. Environmental impact reduction

Environmental impact reduction, represented by f_3 in Equation (3.5) [183], evaluate the positive environmental outcomes obtained from the disassembly process and subsequent component recovery. This objective takes into account five key components: the total environmental impact reduction reclaimed from components to be reused or remanufactured, the environmental impact reduction resulting from the recovery process of components for reuse, remanufacturing, or recycling, the environmental impact reduction associated with the treatment of disposed components, the environmental impact reduction derived from the disassembly operations, and the environmental impact reduction generated by the movements of the robot between disassembly points.

$$f_3 = \sum_{i=1}^{N_p}\sum_{j=1}^{2} r_{i,j}\, er_{i,j}\, \alpha_i \; - \sum_{i=1}^{N_p}\sum_{j=1}^{3} r_{i,j}\, ec_{i,j}\, \alpha_i \; - \sum_{i=1}^{N_p} r_{i,4}\, ec_{i,4}\, (1-\alpha_i) - \\ - \sum_{i=1}^{N_p-1} ed(x_i)\, \alpha_i \; - \sum_{i=1}^{N_p-1} ed(x_i, x_{i+1})\, \alpha_i \tag{3.5}$$

where:

- $er_{i,j}$ is the reclaimed environmental benefits from component i being reused or remanufactured
- $ec_{i,j}$ is the environmental benefits in the recovering process of component i with mode j
- $ed(x_i)$ represents the environmental benefits in disassembly operation x_i.

- $ed(x_i, x_{i+1})$ represents the environmental benefits produced by the movement of the robot between disassembly operations x_i and x_{i+1}, considering that the robot has to change the tool in M if operation x_{i+1} requires using a different tool to the one used in the previous operation x_i.

Constraints

$$\sum_{j=1}^{4} r_{i,j} \;=\; 1 \qquad \forall i \tag{3.6}$$

$$r_{i,1} \;+\; r_{i,2} \;+\; r_{i,3} \;\leq\; \alpha_i \tag{3.7}$$

$$\alpha_i \;\geq\; \alpha_{i+1} \tag{3.8}$$

$$\sum_{i=1}^{N_p} \alpha_i \;\leq\; N_p - 1 \tag{3.9}$$

where:

- Eq. (3.6) [183] guarantees that each component, i, has only one recovery mode.
- Eq. (3.7) [183] assures that all components to be reused, remanufactured or recycled must be disassembled.
- Eq. (3.8) [183] guarantees that if the disassembly operation of component i is the prerequisite of the disassembly operation of component i+1, component i must be disassembled.
- Eq. (3.9) [183] guarantees the maximum number of total disassembled components.

Sustainability recovery scenarios

The assessment of the three goals is based on the disassembly process outcomes and the subsequent recovery and market sale of the disassembled components. The available recovery options for the components include reuse, remanufacturing, recycling, or disposal. To explore the impact of different recovery choices on outcomes, the model incorporates four distinct scenarios referred to as sustainability recovery scenarios. Tables 3.1 and 3.2 present these scenarios, which were formulated in consultation with remanufacturing industries in Spain and England. The scenarios are named as follows: recycling (REC) scenario, remanufacturing (REM) scenario, reuse (REU) scenario, and automated recovery strategy (ARS) scenario.

Table 3.1: The scenarios for the case study gear pump A

Part	REC scenario	REM scenario	REU scenario	ARS scenario
1	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
2	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
3	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
4	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
5	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
6	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
7	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
8	Disposal	Disposal	Disposal	Disposal
9	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
10	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
11	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
12	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
13	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
14	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
15	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*

Note: *algorithm automates recovery strategy identification.

The REC, REM, and REU scenarios are all predetermined based on the typical recovery scenarios employed by industries for each part. These scenarios aim to maximise component recycling, remanufacturing, and reuse, respectively. In contrast, the ARS scenario utilises an autonomous recovery strategy identification mechanism using an algorithm, allowing for greater adaptability and flexibility in locating optimal solutions. The data presented in Tables 3.1 and 3.2

Table 3.2: The scenarios for the case study gear pump B

Part	REC scenario	REM scenario	REU scenario	ARS scenario
1	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
2	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
3	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
4	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
5	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
6	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
7	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
8	Disposal	Disposal	Disposal	Disposal
9	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
10	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
11	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
12	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
13	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
14	Disposal	Disposal	Disposal	Disposal
15	Disposal	Disposal	Disposal	Disposal
16	Disposal	Disposal	Disposal	Disposal
17	Disposal	Disposal	Disposal	Disposal
18	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
19	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
20	Recycle	Remanufacture	Reuse	Recycle/Remanufacture/Reuse*
21	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
22	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
23	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*
24	Recycle	Recycle	Recycle	Recycle/Remanufacture/Reuse*

Note: *algorithm automates recovery strategy identification.

clearly indicate the allocation of recovery options for each part of the ARS scenario. For instance, components 1, 2, and 3 each offer multiple recovery options, such as recycling, remanufacturing, or reuse. Conversely, component 8 only provides the disposal option as it is the only viable alternative for this particular component.

3.3.3 RDSP (SO and MO aggregate approach)

As previously stated, the SO approach utilises the BA to determine the optimal value for each objective across all recovery scenarios. The outcomes were then compared to the MO approaches. The SO approach has similar steps to the MO aggregate approach. The only distinction is that

the SO approach calculates each objective separately, whereas the aggregate approach adds the objectives together and treat it as an SO [97]. The first step of the algorithm involves setting the parameter settings and determining the maximum number of iterations. As mentioned earlier, the input data is a disassembly information matrix based on the MFSG proposed by [54]. This matrix ensures that the disassembly process takes into consideration the presence of fasteners while being feasible by adhering to precedence constraints. Using the matrix thus produces feasible disassembly sequences. In the pseudo-code, the input is the robotic disassembly information matrix, denoted as dis_m. A set of scout bees, denoted as "n", is generated using the Modified Feasible Solution Generation (MFSG) technique to represent all feasible disassembly sequences. These scout bees are then sorted based on their fitness values. Next, the best scout bees from the initial population, referred to as elite site bees (nep), undergo a local search in the elite sites (e). This local search is performed using the swap, insert, and mutation operators, as illustrate in Figure 3.4. The swap and insert operators enable movement of the disassembly sequence, direction, recovery mode, and tools, while the mutation operator only modifies the direction and recovery mode. The mutation operator mutates the best bee of the nep to explore different solutions and find the best fitness value. If the fitness value of the mutated bee is higher than that of the best bee of the nep, it replaces the existing bee. Otherwise, no changes are made. The selection process for the other selected sites (m-e) follows a similar approach to the elite sites (e). The remaining bees (n–m) perform a random search using the MFSG technique to explore the solution space further. The population is then sorted based on fitness values, and the best RDSP information is updated. This process continues until the maximum number of iterations is reached, ensuring that the algorithm continually seeks to improve the RDSP solution by iteratively updating and refining the population of scout bees. The pseudo-code of SO BA and MOBA with aggregate approach is presented in the Algorithm 2.

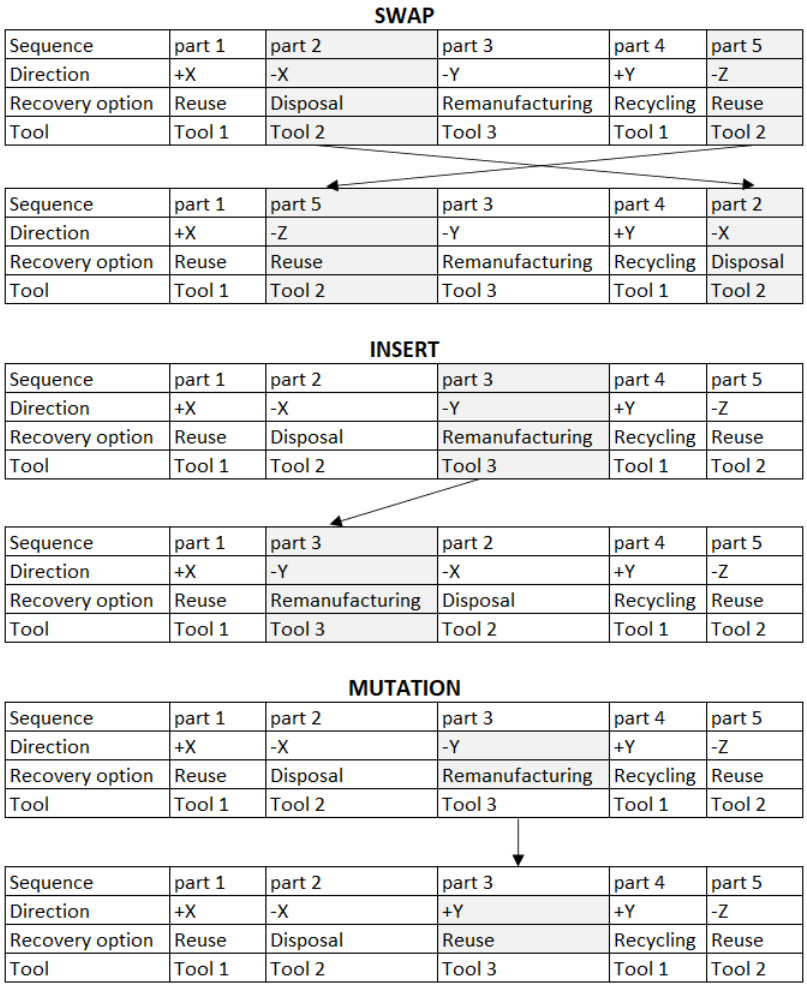


Figure 3.4: Local search illustration in this thesis

Algorithm 2: The pseudo-code SO and MOBA aggregate approach for RDSP

Input : n: number of scout bees, m: number of selected sites, e: number of elite sites, nsp: recruited bees for other selected sites, nep: recruited bees for elite sites, dis_m: robotic disassembly information matrix
Output: RDSP(sequence, direction, mode, tool, (f_1, f_2, f_3 or $*f$ = sum(f_1, f_2, f_3)) *MOBA aggregate approach

Function EDBA(n, m, e, nsp, nep) **:**
  Start
  initialRDSP ← GlobalMFSG($dis\_m : sequence, direction, mode$) `// Generate initial population with feasible disassembly sequences`
  while *stopping criterion not met* **do**
    Evaluate population fitness
    f ← FVALUE(**initialRDSP**)
    Sort population according to f
    Select m sites for local search
    `// Generate local sites with waggle dance`
    for *EliteSite*(1 *to* e) **do**
      `// Assign best elite site bee`
      $BestEliteSiteBee \leftarrow$ *the scout bee that found the elite site*
      for *RecruitedEliteSiteBee*(1 *to* nep) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedEliteSiteBee ← WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction and mode`
        RecruitedEliteSiteBee ← Mutation($dis\_m : direction, mode$)
        Evaluate fitness of **RecruitedEliteSiteBee**
        if ***RecruitedEliteSiteBee*** *is better than* ***BestEliteSiteBee*** **then**
          `// Update BestEliteSiteBee`
          BestLocalBee ← **RecruitedEliteSiteBee**
        end
      end
    end
    for *OtherSelectedSite*(1 *to* $(m - e)$) **do**
      `// Assign best other selected site bee`
      $BestOtherSelectedSiteBee \leftarrow$ *the scout bee that found the other selected site*
      for *RecruitedOtherSelectedSiteBee*(1 *to* (nsp) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedOtherSelectedSiteBee ← WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction and mode`
        RecruitedOtherSelectedSiteBee ← Mutation($dis\_m : direction, mode$)
        Evaluate fitness of **RecruitedOtherSelectedSiteBee**
        if ***RecruitedOtherSelectedSiteBee*** *is better than* ***BestOtherSelectedSiteBee*** **then**
          `// Update BestOtherSelectedSiteBee`
          BestLocalBee ← **RecruitedOtherSelectedSiteBee**
        end
      end
    end
    `// Assign remaining scout bees for global search`
    for *RemainingScoutBee*(1 *to* $(n - m)$ **do**
      $RemianingScoutBee \leftarrow$ GlobalRDSP($dis\_m : sequence, direction, mode$)
    end
    Evaluate fitness of the new population
    Sort population according to f
    `// Store the best RDSP with maximum` f
    Best RDSP = BestBee
  end
  return *Best RDSP (BestBee)*

When there is a clear priority among the objectives and the decision-maker is interested in optimising a specific objective, the SO approach is frequently employed. However, it may not provide a comprehensive perspective on the issue because it does not take into account objective trade-offs, especially if the objectives contradict each other. Therefore, as stated earlier, the thesis emphasises the MO-ND approach, which will be described further in the next section.

3.3.4 RDSP (MO nondominated approach)

To achieve the best balance among the three goals, the MO-ND algorithm is employed. This algorithm generates a set of Pareto-optimal solutions, visualised in the objective space known as the Pareto front [97]. These solutions provide valuable insights for decision-makers, enabling them to choose the most appropriate option according to their preferences.

In this section, the primary focus is on employing the MOBA for the optimisation of the RDSP. Additionally, two benchmark algorithms, namely the Nondominated Sorting Genetic Algorithm - II (NSGA-II) and the Pareto Envelope-based Selection Algorithm - II (PESA-II), are utilised for comparison and evaluation purposes. NSGA-II, in particular, is widely recognised and extensively used in research on RDSP. By employing MOBA, NSGA-II, and PESA-II, this research aims to explore the capabilities and performance of different algorithms in solving the RDSP problem.

The MOBA is an enhanced and adapted version of the BA that was discussed earlier. It has been specifically tailored to handle MO optimisation problems by incorporating nondominated sorting and crowding distance concepts. This adaptation allows MOBA to generate a set of Pareto optimal solutions that represent trade-offs between conflicting objectives. The pseudo-code for MOBA is presented in Algorithms 3.

Visualising multiple objectives can indeed be challenging, especially when dealing with a large number of objectives. In addition, previous researchers have employed various performance metrics to assess the effectiveness of optimisation algorithms for the RDSP problem. This is due

to the fact that no single metric can adequately capture all aspects of performance. Considering these factors, the upcoming section will provide a detailed and comprehensive discussion on the proposed performance evaluation of the MO-ND algorithm. This evaluation aims to shed light on the algorithm's effectiveness and provide valuable insights into its performance across multiple metrics. By doing so, it will contribute to the development of a straightforward yet robust framework for comparing and evaluating the performance of different algorithms.

3.3.5 Performance Evaluation

In assessing the performance of the MO-ND algorithm, several metrics are frequently employed, such as the quantity of nondominated solutions produced, the Hypervolume Indicator (HI), and the number of function evaluations (NFE) as discussed in Chapter 2. This thesis employed these metrics to evaluate the algorithms' performance. The number of nondominated solutions, also known as Pareto optimal solutions (POSs), serves as a measure of convergence speed [171]. However, relying solely on this criterion is inadequate because a higher number of nondominated solutions does not necessarily indicate diverse solutions. Therefore, additional indicators are necessary to comprehensively assess the algorithms' performance. In this thesis, HI is employed to measure both the convergence and diversity of the solution sets obtained from the optimal Pareto front [97, 172]. To ensure a fair contribution from each objective, linear normalisation, as recommended by Knowles et al. [184], is applied using Equation (3.10). Specifically, the normalisation is conducted within the range of [0, 1], with a reference point set at [1.2, 1.2, 1.2].

$$f_{norm} = \frac{f - f_{min}}{f_{max} - f_{min}} \tag{3.10}$$

where: f = objective

A higher value of the HI is considered more preferable as it indicates a wider range of Pareto

Algorithm 3: The pseudo-code of MOBA for RDSP

Input : n: number of scout bees, m: number of selected sites, e: number of elite sites, nsp: recruited bees for other selected sites, nep: recruited bees for elite sites, dis_m: robotic disassembly information matrix
Output: RDSP(sequence, direction, mode, tool, $POSs$)

Function MOBA (n, m, e, nsp, nep) **:**
  Start
  initialRDSP ← GlobalMFSG($dis\_m : sequence, direction, mode$) `// Generate initial population with feasible disassembly sequences and Pareto front set`
  while *stopping criterion not met* **do**
    Evaluate population fitness
    f ← FVALUE(**initialRDSP**)
    Sort population based on nondominated sorting
    Select m sites for local search
    `// Generate local sites with waggle dance`
    for *EliteSite*(1 *to* e **do**
      `// Assign best elite site bee`
      $BestEliteSiteBee \leftarrow$ *the scout bee that* $found$ *the elite site*
      for *RecruitedEliteSiteBee*(1 *to* nep) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedEliteSiteBee ← WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction and mode`
        RecruitedEliteSiteBee ← Mutation($dis\_m : direction, mode$)
        Evaluate fitness of **RecruitedEliteSiteBee** based on nondominated sorting
        if *RecruitedEliteSiteBee* *is better than* ***BestEliteSiteBee*** **then**
          `// Update BestEliteSiteBee`
          BestLocalBee ← **RecrutiedEliteSiteBee**
        end
      end
    end
    for *OtherSelectedSite*(1 *to* $(m - e)$) **do**
      `// Assign best other selected site bee`
      $BestOtherSelectedSiteBee \leftarrow$ *the scout bee that* $found$ *the other selected site*
      for *RecruitedOtherSelectedSiteBee*(1 *to* nsp) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedOtherSelectedSiteBee ← WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction and mode`
        RecruitedOtherSelectedSiteBee ← Mutation($dis\_m : direction, mode$)
        Evaluate fitness of **RecruitedOtherSelectedSiteBee** based on nondominated sorting
        if *RecruitedOtherSelectedSiteBee* *is better than* ***BestOtherSelectedSiteBee*** **then**
          `// Update BestOtherSelectedSiteBee`
          BestLocalBee ← **RecruitedOtherSelectedSiteBee**
        end
      end
    end
    `// Assign remaining scout bees for global search`
    for *RemainingScoutBee*(1 *to* $(n - m)$) **do**
      RemainingScoutBee ← GlobalRDSP($dis\_m : sequence, direction, mode$)
    end
    Evaluate fitness of the new population
    Sort population based on nondominated sorting
    `// Store the Pareto frontier`
    BestRDSP ← BestBee
  end
  return *Best RDSP (BestBee)*

optimal solutions [170]. Additionally, the NFE is utilised to measure the speed at which solutions are found. The novel tools, SPM and PEI, introduced in this chapter were employed to analyse the experimental findings. The main goal of these methods is to determine the best parameter values and offer a simple indicator for evaluating the best-performing algorithms. In this chapter, the PEI is calculated using Equation (3.1), which was introduced in Section 3.2.2.

3.4 Experimental results

The algorithms were implemented and executed using MATLAB 2020b on the University of Birmingham's BEAR cloud service platform. The statistical tests were conducted using IBM SPSS 27. The stopping criteria employed for the algorithms are based on the number of iterations. The iteration size varies from 100 to 500, increasing by 100 in each step. Similarly, the number of populations used ranges from 50 to 80, with increments of 10. In total, there are 20 different parameter settings. To enhance clarity and ease of analysis, each parameter setting is assigned a specific naming convention. For example, "100_50" represents iteration 100 and population size 50. These naming conventions are used to identify and distinguish the groups for statistical testing purposes. Group 1 corresponds to "100_50," Group 2 corresponds to "100_60," up to Group 20, which corresponds to "500_80."

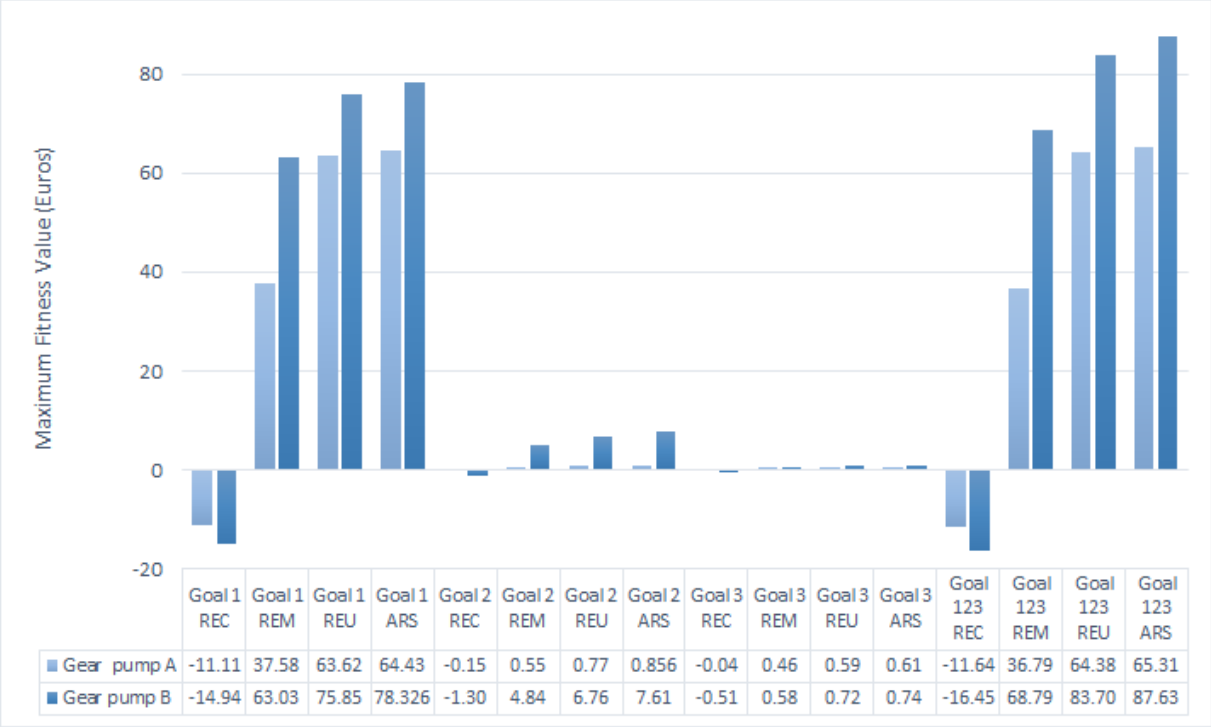
In this research, the parameters of the BA were set as follows: the number of elite sites (e) was set to 1, the number of selected sites (m) was set to 5, the number of recruited bees around elite sites (nep) was set to 10, and the number of recruited bees around selected sites (nsp) was set to 5. The parameter settings for NSGA-II and PESA-II in this study were established by referencing prior research [145]. The crossover probability was set to 0.95, and the mutation probability was set to 0.02. Additionally, for PESA-II, a grid size of 7, an inflation factor of 0.1, and an archive size equal to the number of the population were used. Therefore, if the population size was 50, the

archive size was also set to 50.

3.4.1 SO and MO aggregate results

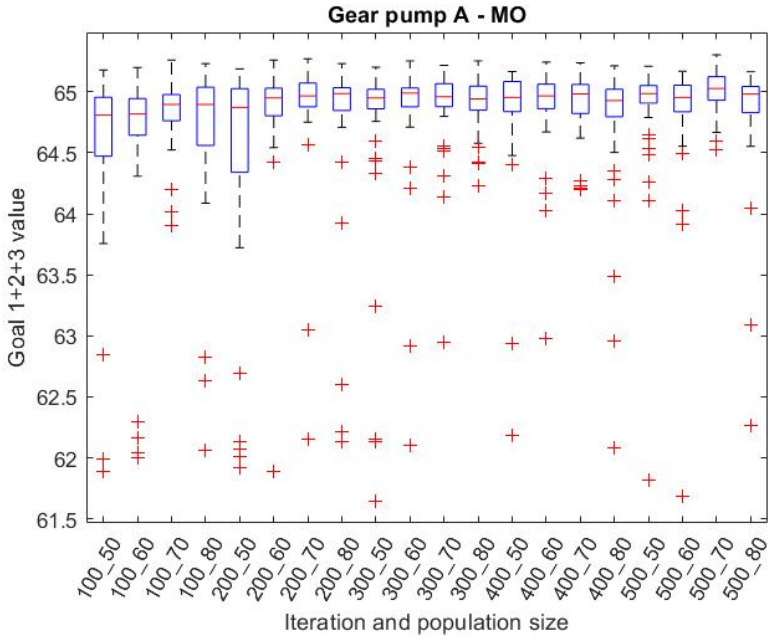
The SO and MO aggregate approach is employed in this thesis to determine the best solutions by evaluating the objectives individually and by aggregating them linearly. Given that previous research has demonstrated the BA as the best-performing algorithm for the RDSP problem [44, 48, 54, 79], no additional comparisons with other algorithms are conducted.

Figure 3.5 displays the maximum fitness value obtained, while Table 3.3 presents the example of disassembly results for Gear Pump A. Figure 3.6 presents the boxplot of the MO aggregate approach for the ARS scenarios. The results of the normality and homogeneity tests can be found in Appendix B. As these tests indicated a violation of the assumptions for parametric tests, non-parametric tests were conducted instead. Specifically, the Kruskal-Wallis test was used for the ARS scenario in both the SO approach and the MO aggregate approach, as shown in Tables 3.4 and 3.5. For the SO approach, the Dunn-Sidak test results for the ARS scenario are presented in Figures 3.7 and 3.8, while Figure 3.9 displays the results for the MO aggregate approach.

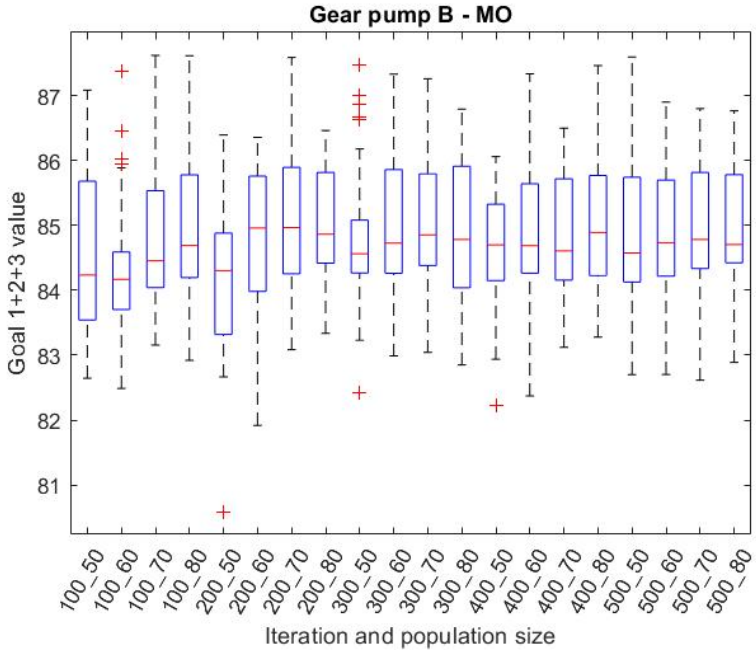


	Goal 1 REC	Goal 1 REM	Goal 1 REU	Goal 1 ARS	Goal 2 REC	Goal 2 REM	Goal 2 REU	Goal 2 ARS	Goal 3 REC	Goal 3 REM	Goal 3 REU	Goal 3 ARS	Goal 123 REC	Goal 123 REM	Goal 123 REU	Goal 123 ARS
Gear pump A	-11.11	37.58	63.62	64.43	-0.15	0.55	0.77	0.856	-0.04	0.46	0.59	0.61	-11.64	36.79	64.38	65.31
Gear pump B	-14.94	63.03	75.85	78.326	-1.30	4.84	6.76	7.61	-0.51	0.58	0.72	0.74	-16.45	68.79	83.70	87.63

Figure 3.5: Maximum Fitness Value of Gear pumps A and B



(a) Gear Pump A



(b) Gear Pump B

Figure 3.6: Boxplot MO aggregate (ARS strategy)

Table 3.3: Example of Disassembly Output of Gear Pump A

	REC scenario	REM scenario	REU scenario	ARS scenario
	Goal 1			
Part sequence	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-14-13-1-2-3-4-5-6-7-12-11-9-10-8
Direction	1-2-2-2-2-2-2-2-2-2-1-1-2-2	1-2-2-2-2-2-2-2-2-2-1-1-2-1	1-2-2-2-2-2-2-2-2-2-1-2-2-1	1-1-1-2-2-2-2-2-2-2-1-2-2-2-2
Recovery option	3-3-3-3-3-3-3-3-3-3-3-3-4-3	2-3-3-3-3-3-3-2-2-2-2-2-2-4-2	1-3-3-3-3-3-3-1-1-1-1-1-1-4-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-4
Tool	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-3-3-1-1-1-1-1-1-4-4-3-3-3-3
MFV (Euros)	-11.11	37.58	63.62	64.43
	Goal 2			
Part sequence	2-1-6-5-4-3-7-9-11-10-8-15-12-13-14	6-1-2-3-4-5-7-11-9-10-8-15-12-13-14	3-4-5-6-1-2-7-11-9-10-8-15-12-13-14	2-1-6-5-4-3-15-14-13-12-7-9-11-10-8
Direction	2-2-2-2-2-2-2-2-2-2-2-1-2-2-2	2-2-2-2-2-2-2-2-2-2-2-1-2-2-1	2-2-2-2-2-2-2-2-2-2-2-1-2-2-1	2-2-2-2-2-2-1-1-1-1-2-1-2-1-1
Recovery option	3-3-3-3-3-3-3-3-3-4-3-3-3-3	3-3-3-3-3-3-2-2-2-2-4-2-2-2-2	3-3-3-3-3-3-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-4
Tool	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	1-1-1-1-1-1-2-3-3-4-4-3-3-3-3
MFV (Euros)	-0.15	0.55	0.77	0.86
	Goal 3			
Part sequence	3-6-4-5-1-2-15-14-13-12-11-10-7-9-8	3-5-4-1-6-2-7-8-10-9-13-11-15-12-14	6-2-1-5-3-4-15-7-9-11-14-10-12-13-8	2-3-4-5-6-1-15-7-14-13-11-12-9-10-8
Direction	2-2-2-2-2-2-1-1-1-1-1-1-2-2-1	2-2-2-2-2-2-2-2-2-2-2-2-1-2-1	2-2-2-2-2-2-1-2-2-2-1-2-1-2-2	2-2-2-2-2-2-1-2-1-1-2-1-2-2-2
Recovery option	3-3-3-3-3-3-3-3-3-3-3-3-3-3-4	3-3-3-3-3-3-2-4-2-2-2-2-2-2-2	3-3-3-3-3-3-1-1-1-1-1-1-1-1-4	1-1-1-1-1-1-1-1-1-1-1-1-1-1-4
Tool	1-1-1-1-1-1-2-3-3-4-3-3-4-3-3	1-1-1-1-1-1-4-3-3-3-3-3-2-4-3	1-1-1-1-1-1-2-4-3-3-3-3-4-3-3	1-1-1-1-1-1-2-4-3-3-3-4-3-3-3
MFV (Euros)	-0.04	0.46	0.59	0.61
	Goal 123			
Part sequence	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	15-14-13-1-2-3-4-5-6-7-12-9-11-10-8
Direction	1-2-2-2-2-2-2-2-2-2-1-2-2-2	1-2-2-2-2-2-2-2-2-2-1-2-2-1	1-2-2-2-2-2-2-2-2-2-1-2-2-2	1-1-1-2-2-2-2-2-2-2-1-1-1-1-1
Recovery option	3-3-3-3-3-3-3-3-3-3-3-3-4-3	2-3-3-3-3-3-3-2-2-2-2-2-2-4-2	1-3-3-3-3-3-3-1-1-1-1-1-1-4-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-4
Tool	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	2-3-3-1-1-1-1-1-1-4-4-3-3-3-3
MFV (Euros)	-11.64	36.79	64.38	65.31

Note:
Direction: 1 = Y+ direction, 2 = Y- direction
Recovery option: 1=reuse, 2=remanufacturing, 3=recycling, 4=disposal
Tool: 1=Spanner-I, 2 = Spanner-II, 3 = Gripper-I, 4 = Gripper-II

Table 3.4: Kruskal-Wallis test results ARS scenario (SO)

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.[a,b]	Decision
1	The distribution of GearA_goal1 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	.026	Reject the null hypothesis.
2	The distribution of GearA_goal2 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	.013	Reject the null hypothesis.
3	The distribution of GearA_goal3 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	.000	Reject the null hypothesis.
4	The distribution of GearB_goal1 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	.000	Reject the null hypothesis.
5	The distribution of GearB_goal2 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	.000	Reject the null hypothesis.
6	The distribution of GearB_goal3 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	<.001	Reject the null hypothesis.

a. The significance level is .050.

b. Asymptotic significance is displayed.

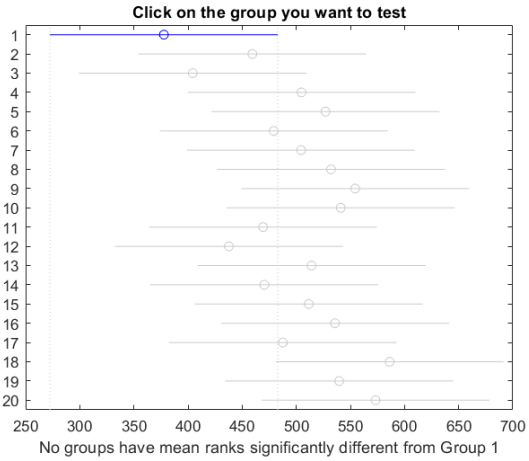
Table 3.5: Kruskal-Wallis test results ARS scenario (MO aggregate)

Hypothesis Test Summary

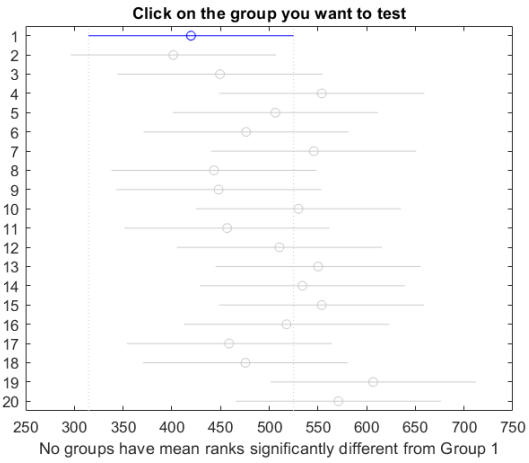
	Null Hypothesis	Test	Sig.[a,b]	Decision
1	The distribution of GearA_goal123 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	<.001	Reject the null hypothesis.
2	The distribution of GearB_goal123 is the same across categories of Iter_pop.	Independent-Samples Kruskal-Wallis Test	<.001	Reject the null hypothesis.

a. The significance level is .050.

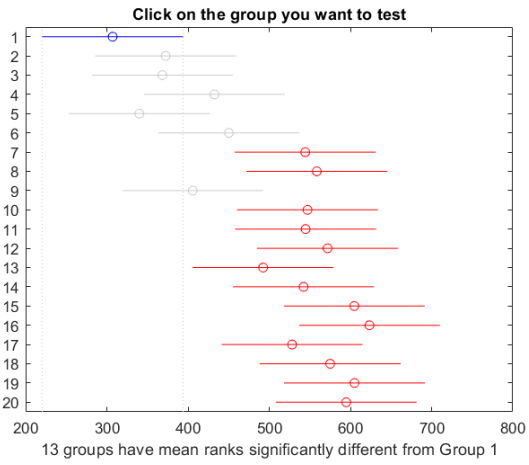
b. Asymptotic significance is displayed.



(a) Goal 1

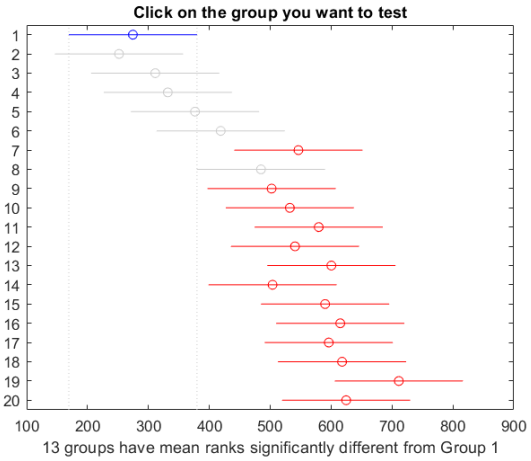


(b) Goal 2

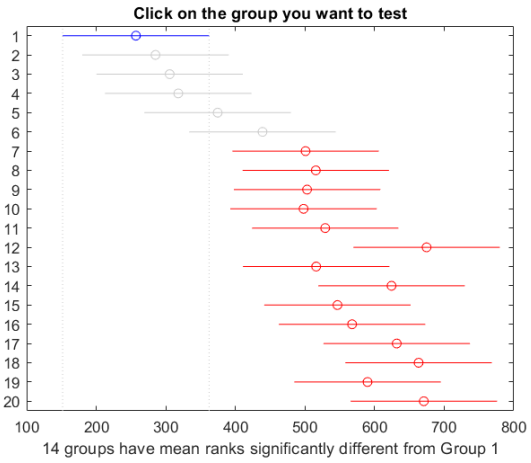


(c) Goal 3

Figure 3.7: Dunn-Sidak test result SO (ARS scenario) Gear Pump A



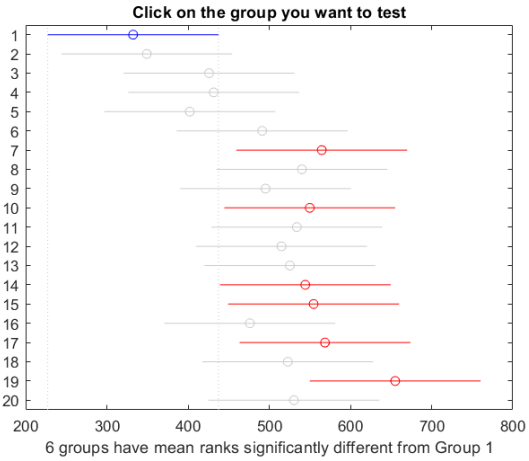
(a) Goal 1



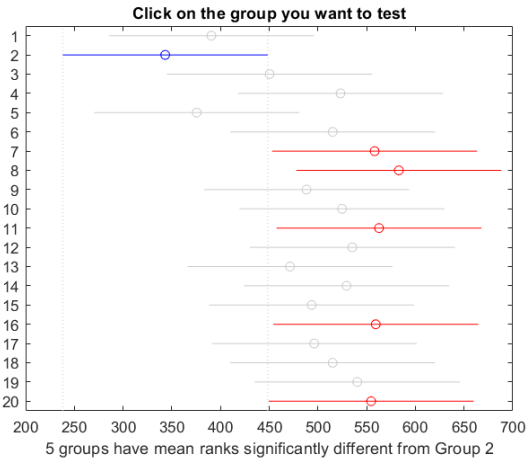
(b) Goal 2

(c) Goal 3

Figure 3.8: Dunn-Sidak test result SO (ARS scenario) Gear Pump B



(a) Gear Pump A



(b) Gear Pump B

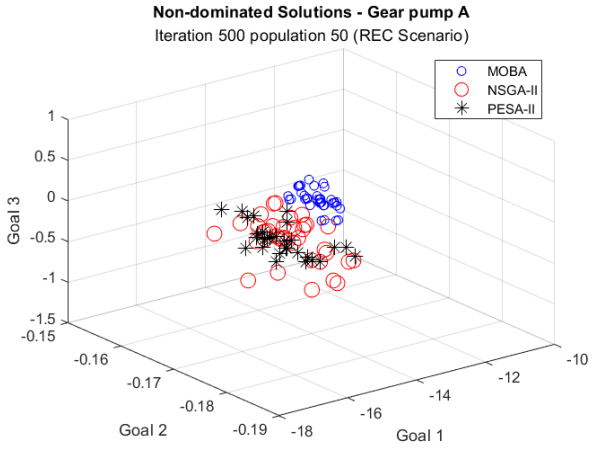
Figure 3.9: Dunn-Sidak test result MO aggregate (ARS scenario)

3.4.2 Multiobjective nondominated results

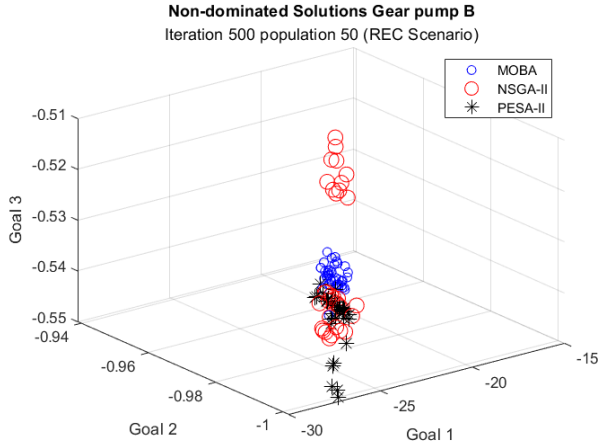
The MO-ND results from these experiments are enormous. To ensure clarity in presenting the results, Figures 3.10 to 3.13 show only one example of the Pareto optimal solutions for iteration 500 with population size 50 for the REC, REM, REU, and ARS scenarios, respectively. These figures represent a small sample of the whole experiment and show how the results of the four algorithms compare in each scenario.

The disassembly output of Gear Pumps A and B using MOBA is presented in Tables 3.6 and 3.7. The number of POSs of Gear Pumps A and B for all scenarios and parameter settings is presented in Figures 3.14 and 3.15. The NFE exhibits a consistent pattern across all scenarios and algorithms, with smaller parameter settings resulting in smaller NFE values, while higher parameter settings correspond to higher NFE values. Due to this clear pattern, the overview of the total NFE for Gear Pumps A and B is shown in Figures 3.16 and 3.17, respectively. The detailed results are presented in Appendix B.

Figures 3.18 and 3.19 show the HI of Gear Pumps A and B for the whole experiment, using the three MO-ND optimisation algorithms MOBA, NSGA-II, and PESA-II for all four scenarios. The higher the HI, the better the set. The PEI, as introduced in Section 3.2.2, is a single index that measures all performance metrics, where a higher index indicates better performance. Figures 3.20 and 3.21 show the PEI for Gear Pumps A and B for all experiments individually, based on the algorithms used in each scenario. Figures 3.22 and 3.23 show the total PEI for each scenario and algorithm for easier interpretation in general.

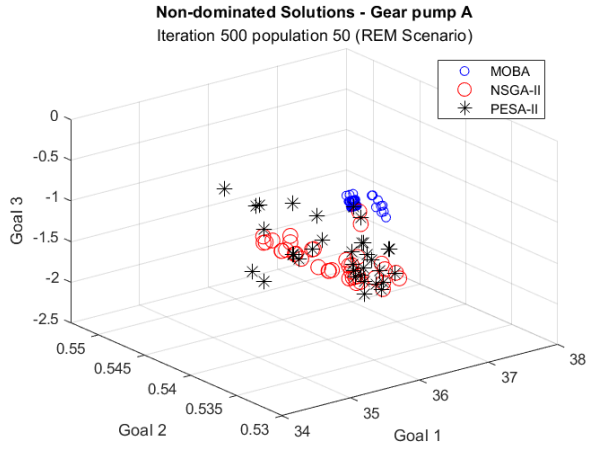


(a) Gear pump A

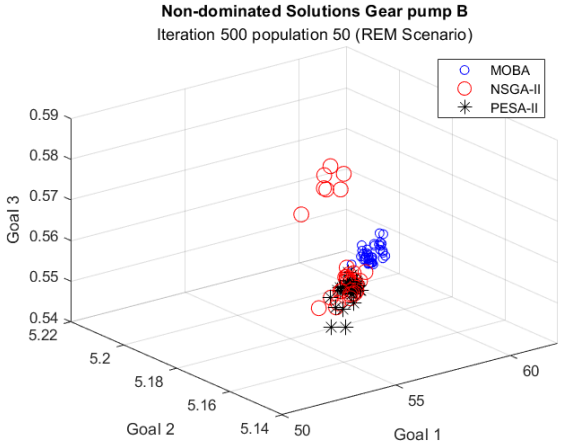


(b) Gear pump B

Figure 3.10: RDSP POSs (REC scenario)

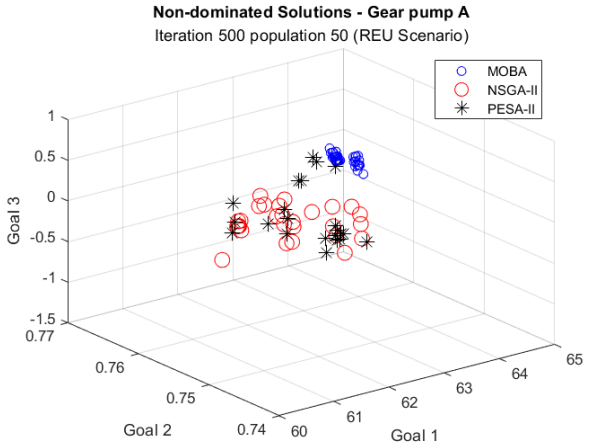


(a) Gear pump A

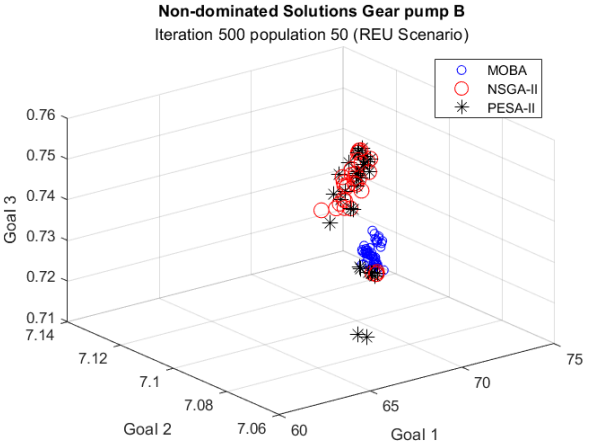


(b) Gear pump B

Figure 3.11: RDSP POSs (REM scenario)

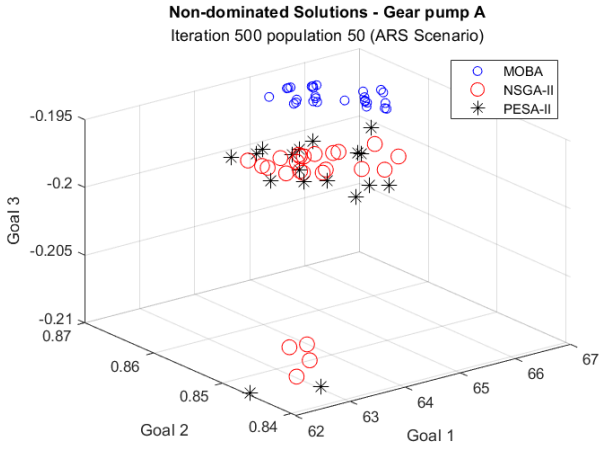


(a) Gear pump A

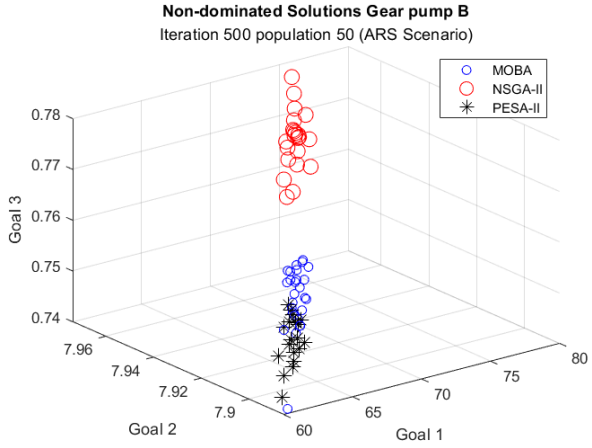


(b) Gear pump B

Figure 3.12: RDSP POSs (REU scenario)



(a) Gear pump A



(b) Gear pump B

Figure 3.13: RDSP POSs (ARS scenario)

Table 3.6: Pareto Optimal Solutions of Gear Pump A (MOBA - Iteration 500, population size 50) ARS scenario

No.	Disassembly Sequence	Disassembly Direction	Disassembly mode	Disassembly Tool	f1	f2	f3
1	1-2-3-6-5-4-15-7-11-9-10-14-13-8-12	2-2-2-2-2-2-1-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.669	0.8583	-0.195
2	1-6-4-5-3-2-15-7-11-9-10-8-12-14-13	2-2-2-2-2-2-1-2-2-2-2-2-2-1-1	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1	1-1-1-1-1-1-2-4-3-3-3-3-4-3-3	64.945	0.8627	-0.195
3	1-6-5-4-3-2-7-11-9-10-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.443	0.8643	-0.195
4	1-6-4-3-2-5-15-7-9-11-10-14-13-8-12	2-2-2-2-2-2-1-2-2-2-2-1-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-4-1-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.696	0.8588	-0.195
5	2-1-3-4-5-6-15-7-9-11-10-8-12-13-14	2-2-2-2-2-2-1-2-2-2-2-2-2-2-2	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1	1-1-1-1-1-1-2-4-3-3-3-3-4-3-3	64.699	0.8647	-0.195
6	2-1-6-5-4-3-7-10-11-9-8-15-12-13-14	2-2-2-2-2-2-2-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.192	0.8656	-0.195
7	2-1-6-5-4-3-15-7-10-11-9-13-14-8-12	2-2-2-2-2-2-1-2-2-2-2-2-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.781	0.8595	-0.195
8	2-1-6-5-4-3-7-11-9-10-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.48	0.8646	-0.195
9	2-4-1-6-3-5-7-9-11-10-8-15-12-13-14	2-2-2-2-2-2-2-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	64.805	0.862	-0.195
10	3-2-5-4-6-1-15-7-10-9-11-8-12-14-13	2-2-2-2-2-2-1-2-2-2-2-2-2-1-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-2-4-3-3-3-3-4-3-3	64.864	0.8622	-0.195
11	3-2-1-5-4-6-7-15-14-13-9-11-10-8-12	2-2-2-2-2-2-2-1-1-1-2-2-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-4-2-3-3-3-3-3-3-4	65.503	0.8602	-0.195
12	3-4-5-6-1-2-7-10-11-9-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.456	0.8646	-0.195
13	3-4-5-6-1-2-7-11-9-10-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.546	0.8646	-0.195
14	3-4-6-1-2-5-15-7-10-11-9-14-13-8-12	2-2-2-2-2-2-1-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.841	0.8601	-0.195
15	5-1-2-3-4-6-7-11-10-9-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.242	0.8624	-0.195
16	5-2-4-3-1-6-7-11-9-10-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.198	0.862	-0.195
17	5-4-3-2-1-6-15-7-10-11-9-14-13-8-12	2-2-2-2-2-2-1-2-2-2-2-1-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.915	0.8608	-0.195
18	5-4-3-2-1-6-7-10-11-9-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.456	0.8646	-0.195
19	5-4-3-2-1-6-7-11-9-10-8-15-12-13-14	2-2-2-2-2-2-2-2-2-2-2-1-2-2-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.149	0.8656	-0.195
20	6-1-2-3-4-5-15-7-9-11-10-13-14-8-12	2-2-2-2-2-2-1-2-2-2-2-2-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.675	0.8586	-0.195
21	6-3-5-1-2-4-15-7-10-9-11-14-13-8-12	2-2-2-2-2-2-1-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-2-4-3-3-3-3-3-3-4	65.561	0.8575	-0.195
22	6-1-2-5-4-3-7-15-14-13-9-11-10-8-12	2-2-2-2-2-2-2-1-1-1-2-2-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	1-1-1-1-1-1-4-2-3-3-3-3-3-3-4	65.472	0.8599	-0.195
23	6-1-2-3-4-5-7-10-11-9-8-15-12-13-14	2-2-2-2-2-2-2-2-2-2-2-1-2-2-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.126	0.8656	-0.195
24	6-1-2-3-4-5-7-11-9-10-8-15-12-14-13	2-2-2-2-2-2-2-2-2-2-2-1-2-1-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.546	0.8646	-0.195
25	6-1-2-3-4-5-7-11-9-10-8-15-12-13-14	2-2-2-2-2-2-2-2-2-2-2-1-2-2-1	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1	1-1-1-1-1-1-4-3-3-3-3-2-4-3-3	65.149	0.8656	-0.195
26	15-6-1-2-3-4-5-7-10-11-9-14-13-8-12	1-2-2-2-2-2-2-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	66.121	0.8595	-0.195
27	15-1-2-3-4-5-6-7-10-11-9-14-13-8-12	1-2-2-2-2-2-2-2-2-2-2-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	66.162	0.8602	-0.195
28	15-1-2-3-6-4-5-7-9-11-10-13-14-8-12	1-2-2-2-2-2-2-2-2-2-2-2-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	65.728	0.8562	-0.195
29	15-6-4-5-2-3-1-7-11-9-10-8-12-14-13	1-2-2-2-2-2-2-2-2-2-2-2-2-1-1	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1	2-1-1-1-1-1-1-4-3-3-3-3-4-3-3	65.117	0.8611	-0.195
30	15-1-5-4-3-2-6-7-11-9-10-13-14-8-12	1-2-2-2-2-2-2-2-2-2-2-2-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	65.869	0.8572	-0.195
31	15-2-5-4-6-1-3-7-11-9-10-8-12-14-13	1-2-2-2-2-2-2-2-2-2-2-2-2-1-2	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1	2-1-1-1-1-1-1-4-3-3-3-3-4-3-3	65.09	0.8615	-0.195
32	15-6-5-4-3-2-1-7-9-10-11-13-14-8-12	1-2-2-2-2-2-2-2-2-2-2-2-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1	2-1-1-1-1-1-1-4-3-3-3-3-3-3-4	65.737	0.856	-0.195

Notes:
Disassembly direction: 1 = Y+ direction and 2 = Y- direction.
Disassembly mode: 1 = reuse, 2 = remanufacturing, 3 = recycle, 4 = disposal.
Disassembly tool: 1 = Spanner-I, 2 = Spanner-II, 3 = Gripper-I, 4 = Gripper-II.

Table 3.7: Pareto Optimal Solutions of Gear Pump B (MOBA - Iteration 500, population size 50) ARS Scenario

No	Disassembly Sequence	Disassembly Direction	Disassembly Recovery Option	Disassembly Tool	f1	f2	f3
1	1-2-23-24-22-21-20-19-6-4-5-3-18-13-7-17-16-11-8-15-9-10-14-12	2-2-1-1-1-1-1-1-2-2-2-2-1-1-2-1-1-2-2-1-2-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-4-4-1-1-4-1	1-1-3-3-3-3-2-2-1-1-1-1-4-4-5-4-4-4-4-4-4-4-4-5	73.408	7.956	0.741
2	1-4-3-5-6-2-23-21-24-22-20-19-18-13-7-8-9-10-17-16-12-15-14-11	2-2-2-2-2-2-1-1-1-1-1-1-1-1-2-2-2-2-1-1-2-1-2-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-4-1-4-4-1	1-1-1-1-1-1-3-3-3-3-3-2-2-4-4-5-4-4-4-4-4-5-4-4-4	74.609	7.967	0.741
3	1-5-6-4-3-2-24-23-21-22-7-19-20-11-9-18-10-17-8-13-16-15-14-12	2-2-2-2-2-2-1-1-1-1-2-1-1-2-2-1-2-1-2-1-1-1-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-4-4-4-1	1-1-1-1-1-1-3-3-3-3-5-2-2-4-4-4-4-4-4-4-4-4-4-5	76.394	7.972	0.741
4	1-5-6-4-24-2-23-21-22-19-20-18-17-13-3-7-8-10-9-11-12-14-15-16	2-2-2-2-1-2-2-1-1-1-1-1-1-1-1-2-2-2-2-2-2-2-2-1	1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-1-1-1-4-4-4	1-1-1-1-3-1-3-3-3-2-2-4-4-4-1-5-4-4-4-4-5-4-4-4	68.272	7.933	0.741
5	1-5-6-4-3-2-24-23-21-22-7-19-20-11-9-18-10-17-8-13-16-15-14-12	2-2-2-2-2-2-1-1-1-1-2-1-1-2-2-1-2-1-2-1-1-1-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-4-4-4-1	1-1-1-1-1-3-3-3-3-3-5-2-2-4-4-4-4-4-4-4-4-4-4-5	76.394	7.972	0.741
6	1-5-3-4-6-24-23-21-22-19-2-20-18-17-7-13-16-10-9-15-14-11-8-12	2-2-2-2-2-1-1-1-1-1-2-1-1-1-2-1-1-2-2-1-1-2-2-1	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-1-1-4-4-1-4-1	1-1-1-1-1-3-3-3-3-2-1-2-4-4-5-4-4-4-4-4-4-4-4-5	73.754	7.960	0.741
7	1-6-5-4-2-24-23-22-21-19-20-3-7-18-10-11-13-9-8-17-16-12-15-14	2-2-2-2-2-1-1-1-1-1-1-2-2-1-2-2-1-2-2-1-1-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-4-1-4-4	1-1-1-1-1-3-3-3-3-2-2-1-5-4-4-4-4-4-4-4-4-5-4-4	73.759	7.962	0.741
8	1-6-4-5-2-23-24-22-21-20-19-3-18-17-16-7-13-9-10-11-15-8-14-12	2-2-2-2-2-1-1-1-1-1-1-2-1-1-1-2-1-2-2-2-1-2-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-1-1-1-4-4-4-1	1-1-1-1-1-3-3-3-3-2-2-1-4-4-4-5-4-4-4-4-4-4-4-5	75.086	7.967	0.741
9	2-1-3-6-4-5-24-23-7-21-22-20-19-18-17-8-10-13-9-16-15-12-14-11	2-2-2-2-2-2-1-1-2-1-1-1-1-1-1-2-2-1-2-1-1-2-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-1-1-4-4-1-4-1	1-1-1-1-1-1-3-3-5-3-3-2-2-4-4-4-4-4-4-4-4-5-4-4	72.489	7.948	0.741
10	2-24-1-3-22-20-5-23-21-19-4-18-6-7-13-8-9-10-12-14-17-16-15-11	2-1-2-2-1-1-2-1-1-1-2-1-2-2-1-2-2-2-2-2-1-1-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-4-4-4-1	1-3-1-1-3-2-1-3-3-2-1-4-1-5-4-4-4-4-5-4-4-4-4-4	60.251	7.885	0.741
11	3-4-1-5-2-6-24-22-23-21-7-20-19-10-11-18-17-9-13-8-12-14-15-16	2-2-2-2-2-2-1-1-1-2-1-1-2-2-1-1-2-1-2-2-2-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-1-4-4-4	1-1-1-1-1-1-3-3-3-3-5-2-2-4-4-4-4-4-4-4-5-4-4-4	74.420	7.960	0.741
12	3-5-4-1-2-6-23-21-24-22-7-20-19-10-9-18-11-13-8-17-16-15-14-12	2-2-2-2-2-2-1-1-1-1-2-1-1-2-2-1-2-1-2-1-1-1-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-4-4-4-1	1-1-1-1-1-1-3-3-3-3-5-2-2-4-4-4-4-4-4-4-4-4-4-5	76.177	7.971	0.741
13	3-5-2-23-21-24-4-1-22-20-19-6-18-17-7-8-16-10-13-9-15-12-14-11	2-2-2-1-1-1-2-2-1-1-1-2-1-1-2-2-1-2-1-2-1-2-2-1	1-1-1-1-1-1-1-1-1-1-1-1-4-1-4-4-1-1-1-4-1-4-1	1-1-1-3-3-3-1-1-3-2-2-1-4-4-5-4-4-4-4-4-4-5-4-4	69.445	7.933	0.741
14	4-5-2-24-23-21-22-1-19-20-3-6-18-17-16-7-13-10-8-9-15-12-14-11	2-2-2-1-1-1-1-2-1-1-2-2-1-1-1-2-1-2-2-2-1-2-1-1	1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-1-1-4-1-4-1-4-1-1	1-1-1-3-3-3-3-1-2-2-1-1-4-4-4-5-4-4-4-4-4-4-5-4-4	71.143	7.942	0.741
15	4-1-3-2-6-24-23-21-22-20-5-19-7-18-8-9-10-13-17-12-14-15-11-16	2-2-2-2-2-1-1-1-1-2-1-2-1-2-2-2-1-1-2-2-2-1-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-4-4-1-4-4	1-1-1-1-1-3-3-3-3-2-1-2-5-4-4-4-4-4-4-5-4-4-4-4	71.506	7.948	0.741
16	4-5-3-1-6-23-21-24-22-2-7-20-19-18-9-13-17-16-8-10-11-15-14-12	2-2-2-2-2-1-1-1-1-2-2-1-1-1-2-1-1-1-2-2-2-1-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-4-1-1-4-4-1	1-1-1-1-1-3-3-3-3-1-5-2-2-4-4-4-4-4-4-4-4-4-4-5	75.579	7.970	0.741
17	5-1-2-6-4-3-24-23-22-21-20-19-7-8-10-11-9-18-13-12-17-16-14-15	2-2-2-2-2-2-1-1-1-1-1-1-2-2-2-2-2-1-1-2-1-1-2-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-4-4-4-4	1-1-1-1-1-1-3-3-3-3-2-2-5-4-4-4-4-4-4-5-4-4-4-4	74.028	7.961	0.741
18	5-1-2-3-4-6-23-21-24-22-20-19-18-17-13-7-8-10-11-9-16-12-15-14	2-2-2-2-2-2-1-1-1-1-1-1-1-1-1-2-2-2-2-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-1-1-1-4-1-4-4	1-1-1-1-1-1-3-3-3-3-2-2-4-4-4-5-4-4-4-4-4-5-4-4	74.595	7.968	0.741
19	5-2-1-4-3-6-24-23-22-21-20-19-7-8-18-11-13-17-9-16-15-10-14-12	2-2-2-2-2-2-1-1-1-1-1-1-2-2-1-2-1-1-2-1-1-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-4-4-1-4-1-1	1-1-1-1-1-1-3-3-3-3-2-2-5-4-4-4-4-4-4-4-4-4-4-5	75.931	7.967	0.741
20	5-3-1-6-24-22-23-21-2-20-19-4-18-7-17-10-13-9-8-12-16-11-14-15	2-2-2-2-1-1-1-1-2-1-1-2-1-2-1-2-1-1-2-2-2-1-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-4-1-4	1-1-1-1-3-3-3-3-1-2-2-1-4-5-4-4-4-4-4-5-4-4-4-4	69.467	7.933	0.741
21	5-4-6-23-24-2-22-21-20-19-18-3-1-7-8-10-9-13-17-12-16-14-11-15	2-2-2-1-1-2-1-1-1-1-1-2-2-2-2-2-2-1-1-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-4-4-1-4	1-1-1-3-3-1-3-3-2-2-4-1-1-5-4-4-4-4-4-5-4-4-4-4	68.455	7.929	0.741
22	6-5-4-1-2-24-23-21-22-19-20-3-18-7-8-13-17-16-10-9-12-15-11-14	2-2-2-2-2-1-1-1-1-1-1-2-1-2-2-1-1-1-2-2-2-1-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-4-4-1-1-1-4-1-4	1-1-1-1-1-3-3-3-3-2-2-1-4-5-4-4-4-4-4-4-5-4-4-4	70.919	7.945	0.741
23	6-5-2-4-23-1-24-21-22-3-20-19-18-17-7-8-10-11-9-13-16-15-12-14	2-2-2-2-1-2-1-1-1-2-1-1-1-1-2-2-2-2-2-2-1-1-1-2-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-4-1-1-1-1-4-4-1-4	1-1-1-1-3-1-3-3-3-1-2-2-4-4-5-4-4-4-4-4-4-4-4-5-4	70.466	7.942	0.741
24	23-5-3-2-1-6-21-24-22-19-20-18-17-4-13-16-7-8-15-10-9-12-14-11	1-2-2-2-2-2-1-1-1-1-1-1-1-2-1-1-2-2-1-2-2-2-1-1	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-1-4-4-1-1-1-4-1	3-1-1-1-1-1-3-3-3-2-2-4-4-1-4-4-5-4-4-4-4-5-4-4	70.230	7.940	0.741
25	23-6-1-5-2-24-21-22-19-20-3-18-17-16-4-7-15-14-13-9-10-11-8-12	1-2-2-2-2-1-1-1-1-1-2-1-1-1-2-2-1-1-1-2-2-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-1-4-4-1-1-1-1-4-1	3-1-1-1-1-3-3-3-2-2-1-4-4-4-1-5-4-4-4-4-4-4-4-5	72.492	7.953	0.741
26	23-6-2-4-5-24-22-21-19-20-3-1-18-17-7-8-13-9-16-11-15-10-14-12	1-2-2-2-2-1-1-1-1-1-2-2-1-1-2-2-1-2-1-2-1-2-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-1-4-1-4-1-4-1	3-1-1-1-1-3-3-3-2-2-1-1-4-4-5-4-4-4-4-4-4-4-4-5	72.569	7.949	0.741
27	24-23-22-5-6-4-2-3-21-19-20-18-1-7-8-13-17-10-9-16-15-12-14-11	1-1-1-2-2-2-2-2-1-1-1-1-2-2-2-1-1-2-2-1-2-1-1-2	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-4-1-4-4-1-4-1	3-3-3-1-1-1-1-1-3-2-2-4-1-5-4-4-4-4-4-4-4-5-4-4	70.029	7.937	0.741
28	24-2-3-5-6-1-23-21-22-20-19-18-17-4-7-16-15-8-10-11-14-13-9-12	1-2-2-2-2-2-1-1-1-1-1-1-2-2-1-1-2-2-2-1-1-2-2-2	1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-4-4-1-1-4-1-1-1	3-1-1-1-1-1-3-3-3-2-2-4-4-1-5-4-4-4-4-4-4-4-4-5	73.340	7.961	0.741

Note:
Disassembly Direction: 1 = Y+ direction, 2 = Y- direction
Disassembly Recovery Option: 1=reuse, 2=remanufacturing, 3=recycling, 4=disposal
Disassembly Tool: 1=Spanner-I, 2 = Spanner-II, 3 = Spanner-III, 4 = Gripper-I, 5 = Gripper-II

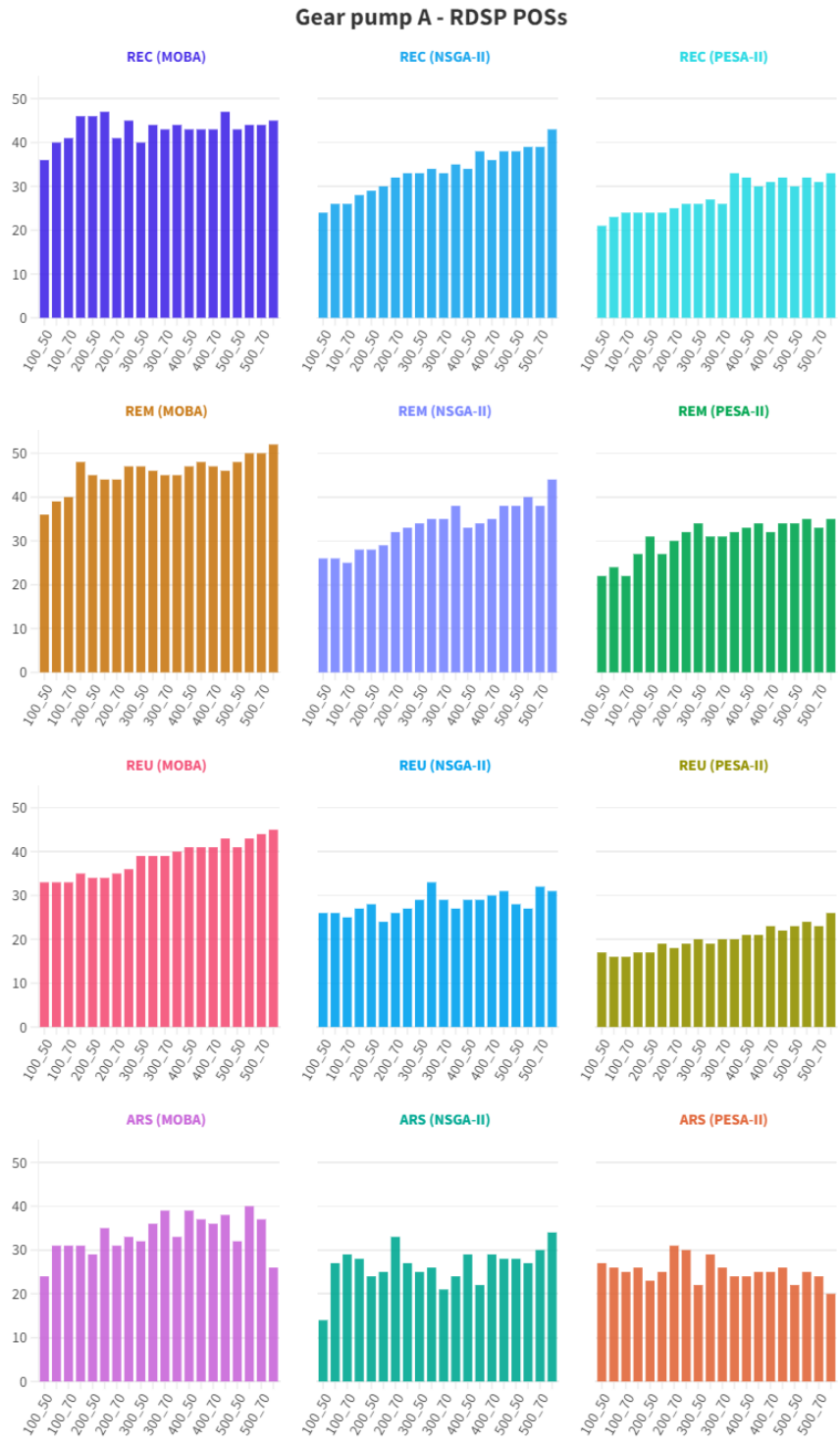


Figure 3.14: Number of Pareto optimal solutions for RDSP of Gear Pump A

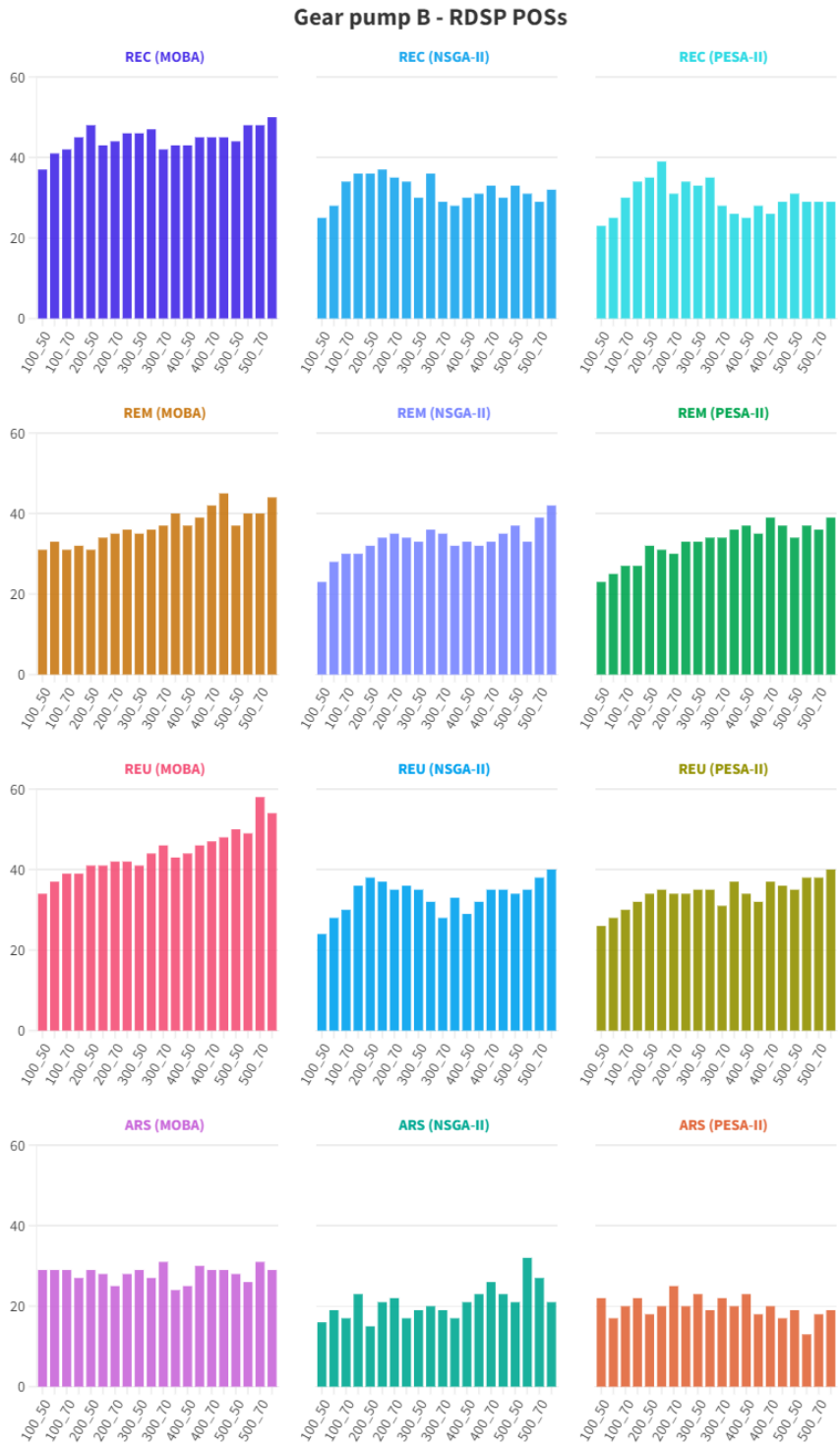


Figure 3.15: Number of Pareto optimal solutions for RDSP of Gear Pump B

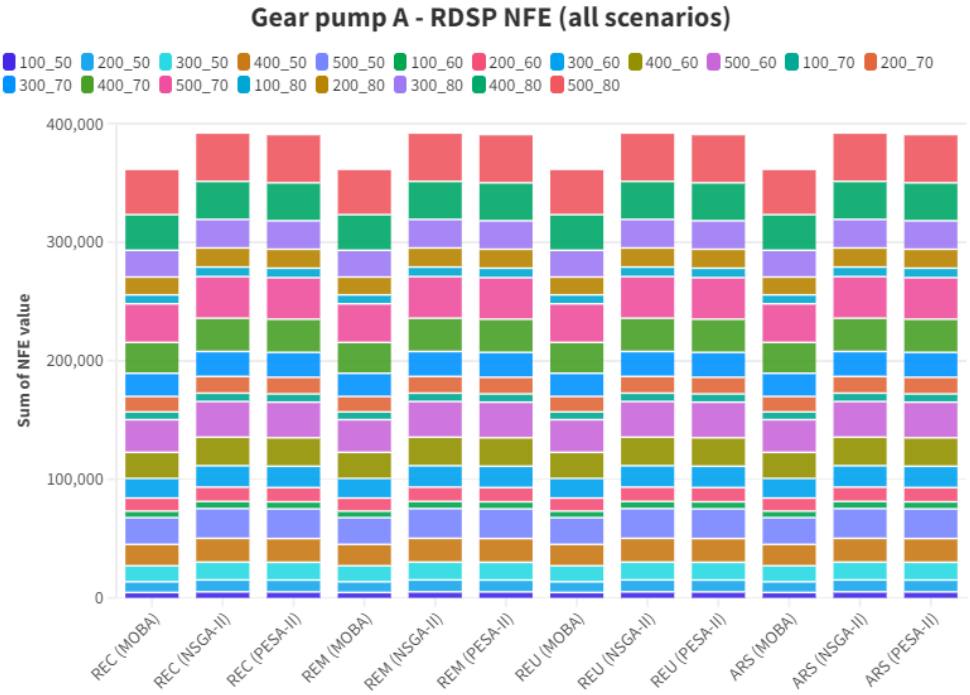


Figure 3.16: Number Function of Evaluation for RDSP of Gear Pump A: The lower the better

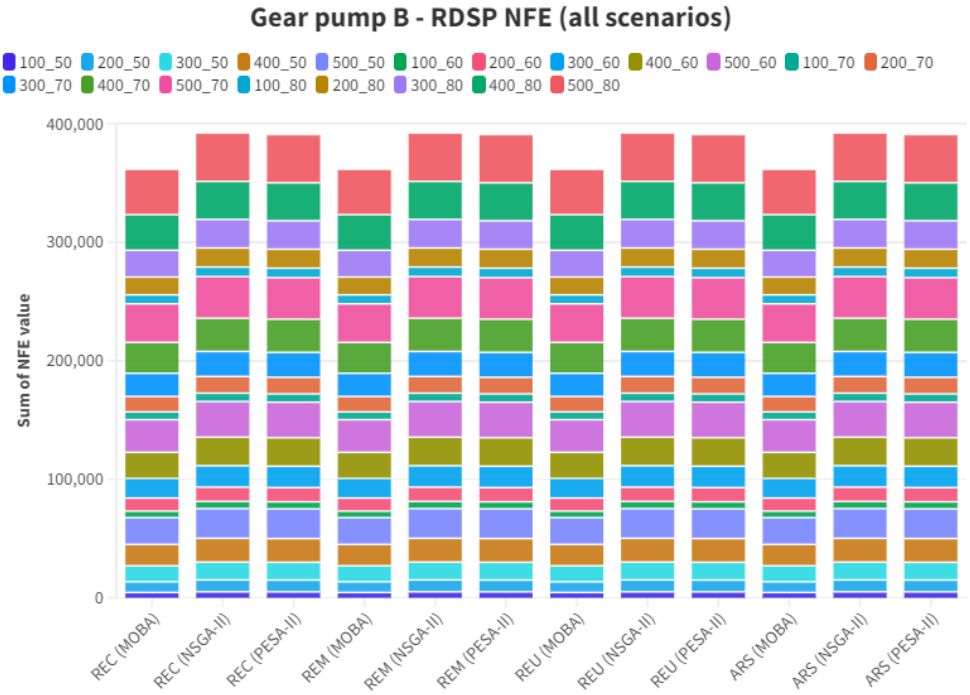


Figure 3.17: Number Function of Evaluation for RDSP of Gear Pump B: The lower the better

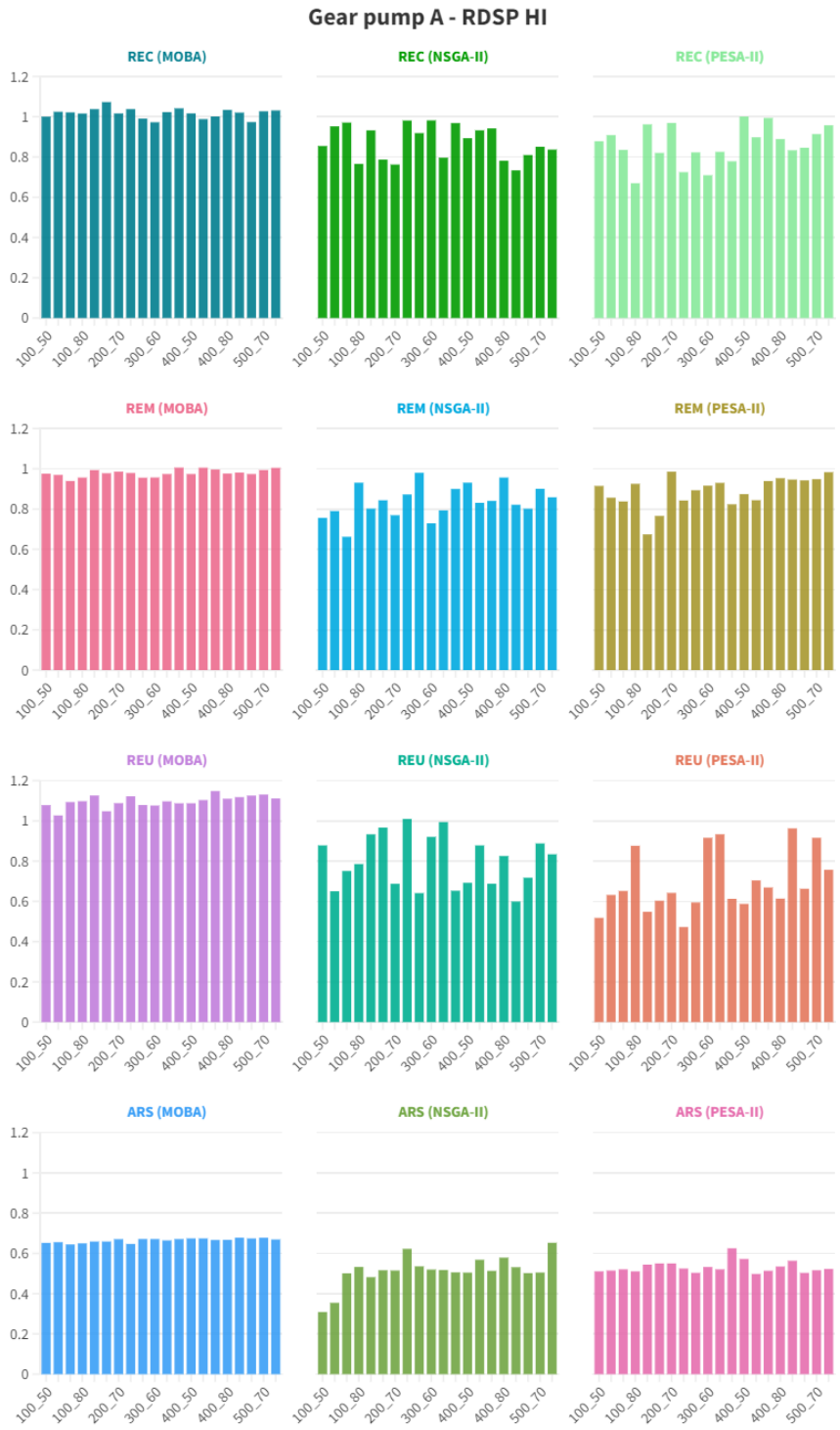


Figure 3.18: Hypervolume Indicator for RDSP of Gear Pump A: The higher the better

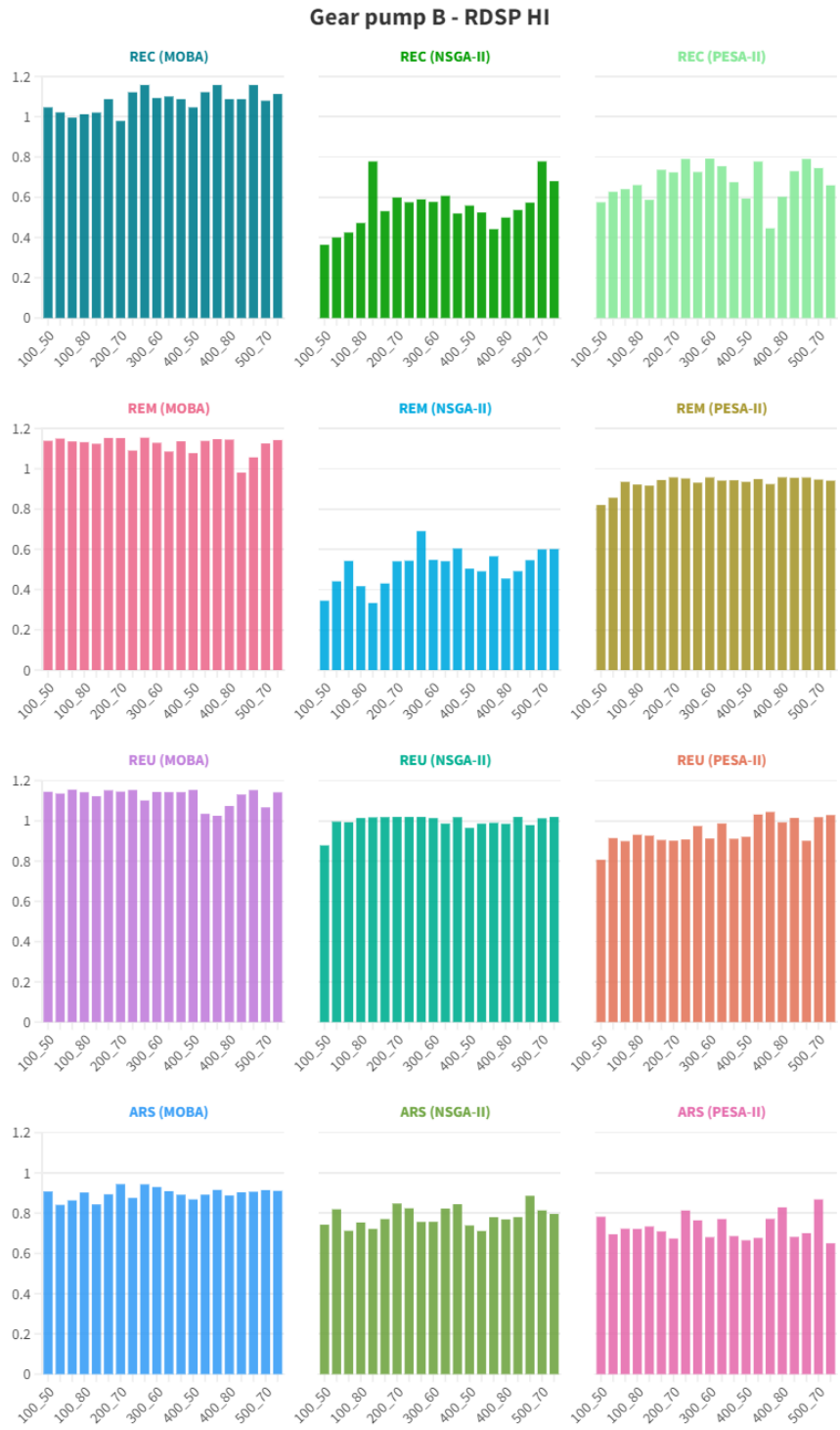


Figure 3.19: Hypervolume Indicator for RDSP of Gear Pump B: The higher the better

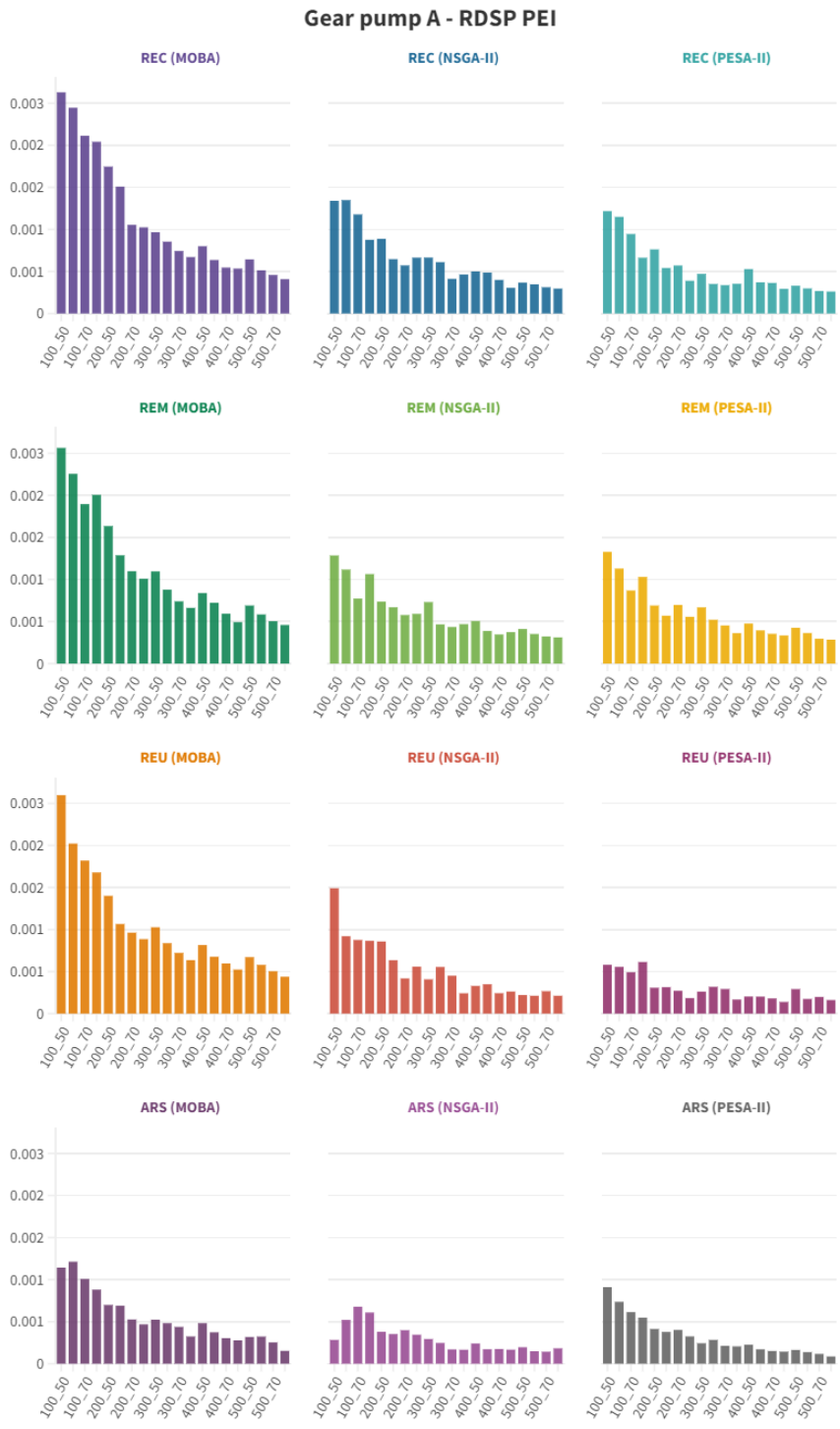


Figure 3.20: PEI for RDSP of Gear Pump A: The higher the better

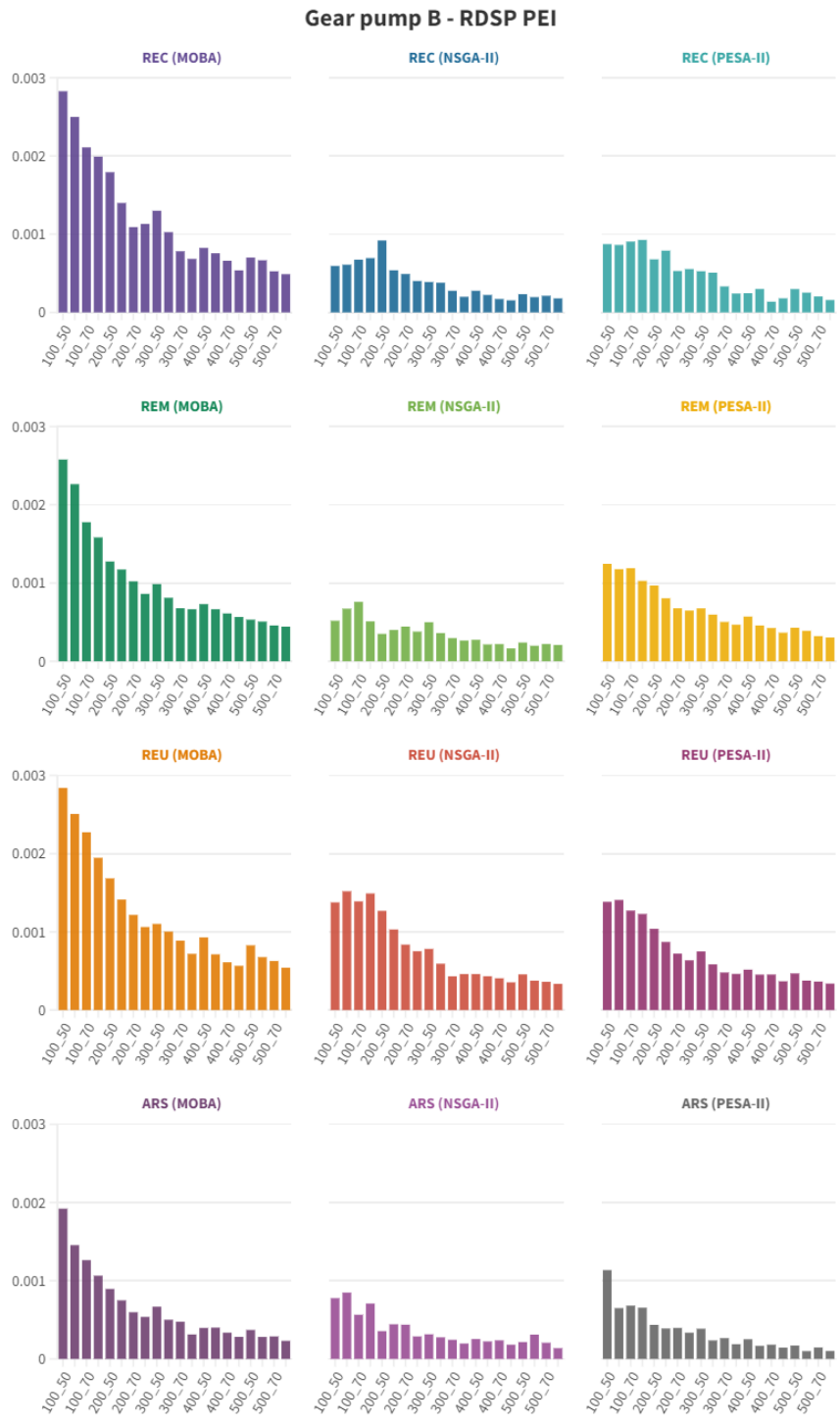


Figure 3.21: PEI for RDSP of Gear Pump B: The higher the better

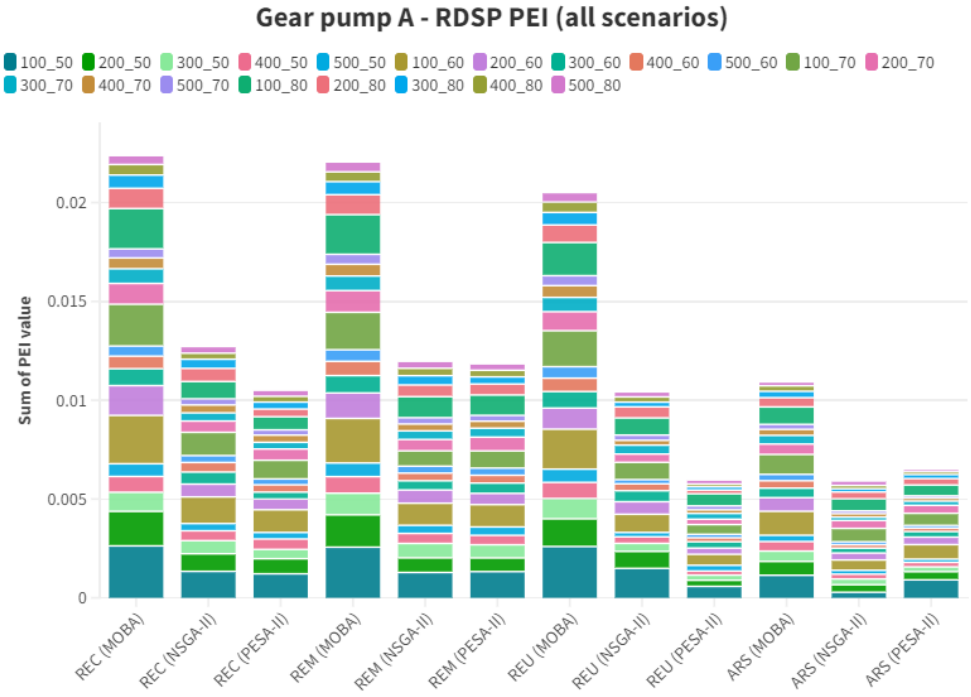


Figure 3.22: Total PEI for RDSP of Gear Pump A: The higher the better

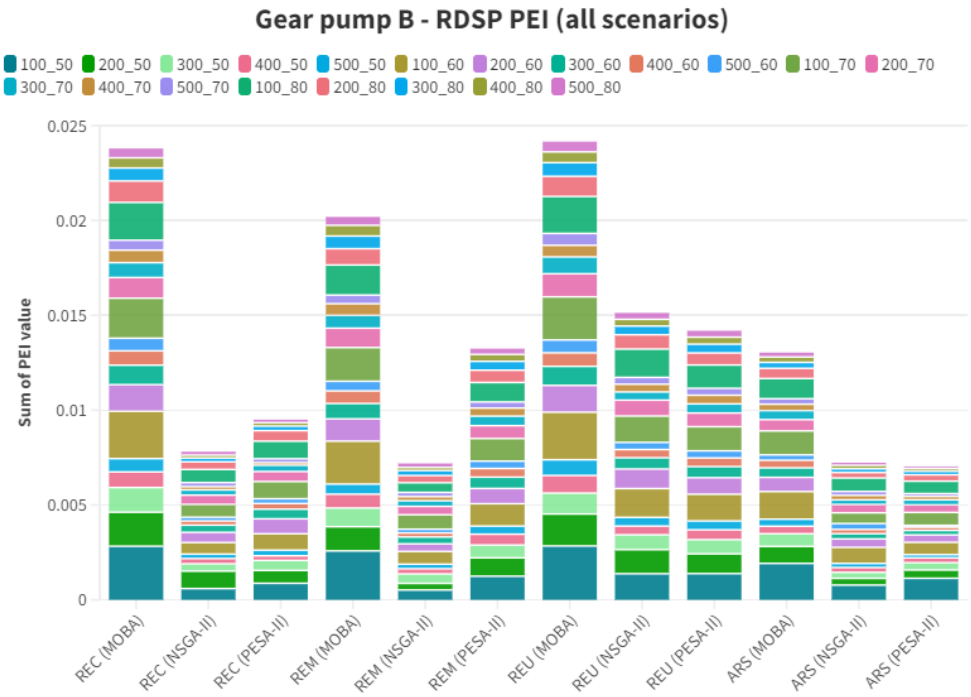


Figure 3.23: Total PEI for RDSP of Gear Pump B: The higher the better

3.5 Discussion

The MO-ND approach differs from the SO and MO aggregate approaches in that it provides a set of solutions rather than a single solution. The discussion begins by presenting the results of the SO and MO aggregate approaches, focusing on identifying the maximum value for each goal and when the goals are aggregated under a linear assumption. On the other hand, the MO-ND approach aims to identify a set of solutions that are not dominated by any other solution in terms of all the objectives simultaneously. This approach offers a more comprehensive analysis of the trade-offs and compromises between different goals. It provides decision-makers with a range of solutions that represent various trade-offs between the objectives, allowing them to select the solution that best aligns with their priorities. Overall, the SO, MO aggregate, and MO-ND approaches all provide valuable insights for the planning of disassembly sequences. They provide distinct perspectives and considerations for optimising the disassembly process based on particular objectives and trade-offs.

3.5.1 SO and MO aggregate Analysis

Figure 3.5 presents the maximum fitness value (MFV) obtained from the experimental results of both the SO and MO aggregate method for all scenarios. The REU and ARS scenarios consistently demonstrate higher monetary value across all individual goals and when aggregated. However, the REC scenario exhibits a negative monetary value, which can be attributed to the additional processing involved. When evaluating the goals individually, Goal 1 (profit) consistently yields the highest value. In the ARS scenario, Gear Pump A achieves the highest value at € 64.43 , closely followed by the REC scenario at € 63.62. The REM scenario yields a value of € 37.58, while the REC scenario results in a negative value of € 11.11. Similarly, for Goal 2 (energy savings), the

ARS scenario achieves the highest value of € 0.856, with the REU scenario following closely at € 0.77. The REM scenario shows a value of € 0.55, while the REC scenario exhibits a negative value of € 0.15. For Goal 3 (environmental impact reduction), the ARS scenario yields the highest value of € 0.61, followed by the REU scenario at € 0.59. The REM scenario yields a value of € 0.46, while the REC scenario results in a negative value of € 0.04.

The MO aggregate methods were utilised to combine Goals 1, 2, and 3, resulting in similar trends to Goal 1. In the ARS scenario, the aggregated value closely mirrors the results of Goal 1, at € 65.31. Likewise, the REU scenario demonstrates an aggregated value of € 64.38, followed by the REM scenario at € 36.79. However, the REC scenario exhibits a lower aggregated value of negative € 11.64. Gear Pump B shows similar results to Gear Pump A. Concerning Goal 1 (profit), the highest value is observed in the ARS scenario at € 78.326, followed by the REU scenario at € 75.85. The REM scenario yields a value of € 63.03, while the REC scenario results in a negative value of € 14.94. For Goal 2 (energy savings), the ARS scenario achieves the highest value of € 7.61, followed by the REU scenario at € 6.76. The REM scenario shows a value of € 4.84, while the REC scenario exhibits a negative value of € 1.3. Similarly, for Goal 3 (environmental impact reduction), the ARS scenario yields a value of € 0.74, the REC scenario yields € 0.72, the REM scenario yields € 0.58, and the REC scenario results in a negative value of € 0.51. The MO aggregate methods were again employed to combine Goals 1, 2, and 3, resulting in similar results to Goal 1. In the ARS scenario, the aggregated value is € 87.63, in the REU scenario, it is € 83.7, in the REM scenario, it is € 68.79, and in the REC scenario, it is negative € 16.45.

Table 3.3 presents example of the disassembly output, including the disassembly sequence, direction, recovery option, tool, and maximum fitness values for Goals 1, 2, 3, and the aggregated goals within the REC, REM, REU, and ARS scenarios. This comprehensive approach provides a more insightful analysis by considering the specific recovery options in each scenario and their relationship to the disassembly sequences and other relevant disassembly information. In contrast to prior studies that mainly concentrate on the end result, this comprehensive analysis provides a

more profound comprehension of the disassembly process and its significance in attaining different objectives within the area of robotic disassembly sequence planning. It is interesting to observe that the ARS scenario predominantly suggests the reuse option for most of the components. This finding highlights the algorithm's ability to identify the optimal recovery option, as it recognises that the knowledge of reusing these components leads to the highest monetary value. The BA's capability to prioritise reuse demonstrates its effectiveness in optimising the goals of the disassembly process.

Figure 3.6 presents the boxplot results for Gear Pump A and Gear Pump B in the ARS strategy for the MO aggregate method. The boxplot displays the maximum, minimum, 25^{th} and 75^{th} percentiles, median, and outliers. The detailed results for other scenarios can be found in Appendix B. Notably, the visual representation of the results does not exhibit a normal distribution for Gear Pump A and Goal 3 in Gear Pump B.

To determine the optimal parameter settings, the methodology introduced in Section 3.2.1, which utilises the novel statistical performance metric, was followed. The study conducted a total of 50 runs, surpassing the minimum requirement of 30 runs for conducting a parametric test. Normality and homogeneity tests were performed to verify if the data followed a normal distribution and if the variance was equal across the groups.

To further investigate the mean differences among the groups, a post hoc test using the Dunn-Sidak method was conducted, as shown in Figures 3.7, 3.8, and 3.9 for ARS scenario. The results revealed significant differences between the groups with 100 iterations and population sizes of 50, 60, 70, 80, and 200 iterations with population sizes of 50 and 60. This implies that, from a statistical standpoint, similar results were obtained starting from 200 iterations with a population size of 60 up to the largest parameter setting of 500 iterations with 80 population. These parameter settings can be utilised to determine the optimal MFV for both the SO and MO aggregate approaches. For example, using 200 iterations with a population size of 60 yields the same results as using 500 iterations with a population size of 80. This means that researchers can choose to use the smaller parameter settings to achieve the same results. It is important to

note that for Gear Pump B, which has a greater number of components compared to Gear Pump A, the optimal parameter settings ranged from 200 iterations with a population size of 70 to 500 iterations with a population size of 80. This observation is understandable, as the increased number of components in Gear Pump B leads to a larger search space. Therefore, researchers conducting future studies with a similar number of components or the same case study can utilise the parameter settings derived from this research. These findings highlight the potential of exploring the statistical performance metric in identifying the optimal parameter settings for the algorithm.

Overall, the experimental results consistently demonstrate that the REU and ARS scenarios generate higher monetary values for both gear pumps. Conversely, the REC scenarios yield negative values. These findings reinforce the existing literature, which emphasises that recycling should be regarded as a last resort within a CE due to its higher energy consumption and processing requirements compared to alternative recovery options such as reuse and remanufacturing. The ARS and REU scenarios exhibit similar outcomes, with the BA identifying the ARS scenario as the optimal recovery option, indicating that reusing each component would yield the highest monetary value across all objectives. Furthermore, the results demonstrate that remanufacturing tends to yield a lower monetary value compared to reuse, likely due to the additional processing required. This aligns with the understanding that reusing products is generally more cost-effective than remanufacturing them. Lastly, it is evident that Goal 1 (profit) consistently yields higher monetary values compared to Goal 2 (energy savings) and Goal 3 (environmental impact reduction) in all scenarios. This finding is consistent with the results obtained from the MO aggregate analysis, where Goal 1 contributes the most to the overall aggregate of the objectives. The higher monetary value associated with Goal 1 suggests that it has a greater impact on the profitability aspect of the disassembly process. This aligns with the understanding that maximising profit is often a primary concern in business and economic contexts. The emphasis on Goal 1 in the MO aggregate results indicates its significant contribution to the overall optimisation of the disassembly sequence. Although the monetary values for energy savings and environmental impact reduction may be

relatively smaller in comparison to profit, it is imperative to consider all three goals within the sustainability model. The next analysis of the nondominated results will provide further insights into the importance of incorporating multiple objectives in the decision-making process and the trade-offs involved in optimising across these objectives.

3.5.2 Multiobjective nondominated Analysis

The MO-ND approach considers trade-offs between objectives, and the results obtained from this approach are presented in Figures 3.10, 3.11, 3.12, and 3.13 for the REC, REM, REU, ARS scenarios, respectively. These figures show the POSs using MOBA, NSGA-II, and PESA-II for iteration 500 and population size 50. In these figures, similarities can be observed between the results of NSGA-II and PESA-II.

To exemplify the value of the detailed output, Tables 3.6 and 3.7 provide a comprehensive detail of the disassembly process for an ARS scenario using the MOBA. These tables include information on the disassembly sequence, direction, recovery mode, tool, and objective values. They enable a thorough evaluation and comparison of objectives, offering valuable insights that have often been overlooked in previous literature. In prior research, there has been a predominant focus on objective results, neglecting the importance of providing a comprehensive detail of disassembly output. However, analysing these details facilitates a deeper understanding of the disassembly process, validation of model accuracy, and more informed decision-making. By employing the MO-ND approach, this study generates a set of solutions that can be compared to results obtained from SO and MO aggregate approaches based on the objectives. Specifically, for Gear Pump A, Figure 3.5 visually represents this comparison by presenting the maximum values of individual goals as € 64.43, € 0.86, and € 0.61, with an aggregated value of € 65.31. However, when considering trade-off solutions, the values shift to approximately € 65, € 0.86, and negative € 0.2, as shown in Table 3.6. Similarly, Table 3.7 showcases the results for Gear Pump B. The MO-ND solutions,

considering trade-offs, demonstrate values of € 76.39, € 7.9, and € 0.74 for Goal 1, Goal 2, and Goal 3, respectively. The individual goal maximums are € 78.33, € 7.61, and € 0.74, with an aggregated value of € 87.63, as depicted in Figure 3.5. The observed discrepancies in both cases indicate that the aggregated goals do not adhere to a linear pattern, suggesting the presence of conflicting objectives. Therefore, the nondominated approach proves more suitable for addressing such scenarios and providing a more comprehensive understanding of the disassembly process for Gear Pump A and Gear Pump B. By incorporating the detailed disassembly output into the analysis, this study fills a gap in the existing literature and contributes to a more holistic understanding of objectives across different approaches. This integration of detailed output in the comparison of objectives represents a novel contribution that has not been explored previously.

The first performance metric in this thesis is the number of POSs. Figures 3.14 and 3.15 present the results for Gear Pumps A and B, respectively. It is evident that there is a consistent trend of increasing POSs as the number of parameter settings increases across all scenarios and algorithms depicted in both figures. MOBA yield higher POSs compared to NSGA-II and PESA-II. Based on these findings, it can be concluded that MOBA is the best-performing algorithms, consistently generating a higher number of POSs across all experiments in the four scenarios.

Figures 3.16 and 3.17 depict the NFE for Gear Pumps A and B, respectively, which serve as the second performance metric for evaluating the effectiveness of the MO-ND result. The NFE is influenced by the parameter settings, with smaller settings resulting in fewer function evaluations. These figures offer an aggregated overview of the total NFE for each scenario and algorithm, facilitating a comprehensive analysis. For a detailed output of the NFE results, please refer to Appendix B, which presents the previously mentioned pattern. Smaller NFE values are desirable, suggesting that the algorithm performs better in terms of computational efficiency. The figures clearly demonstrate that MOBA consistently achieves the lowest total NFE, followed by PESA-II, and NSGA-II for both gear pumps. This compellingly indicates that MOBA outperforms the other algorithms in terms of NFE comparison.

Figures 3.18 and 3.19 show the HI and demonstrate comparable outcomes between NSGA-II and PESA-II. The higher values of HI for MOBA suggest it is performs better than the other algorithms. Moreover, MOBA demonstrate greater consistency in their HI values when compared to the remaining algorithms. The comparison of HI values across different scenarios reveals interesting findings. Notably, scenario ARS demonstrates the lowest HI value among the four scenarios, indicating a smaller hypervolume and a more concentrated or narrowly distributed set of solutions in the objective space. The difference in HI values between scenario ARS and the other three predefined scenarios (REC, REM, and REU) can be attributed to the distinct nature of the ARS scenario. In REC, REM, and REU, the recovery option is predefined, providing the algorithm with a single predetermined recovery option. However, in the ARS scenario, the algorithm has the autonomy to explore and determine the best recovery options independently. Furthermore, the disassembly output indicates that the ARS scenario provides reuse as the most preferred recovery option. This preference for reuse, which is associated with the highest monetary value for the three sustainability goals in the model, may contribute to the reduced diversity observed in the ARS scenario.

Based on the analysis of the three parameter metrics, MOBA demonstrate superior performance compared to the other algorithms. However, since each metric measures different aspects of performance, it is important to consider multiple metrics rather than relying on a single performance metric. While conflicting results among different metrics are less commonly reported in the literature, it is still essential to consider multiple metrics to obtain a comprehensive evaluation of algorithm performance. Many studies often use one or two metrics without explicitly discussing conflicting results. However, by incorporating multiple metrics, the proposed PEI integrates these metrics into a single index to provide a comprehensive evaluation and overcome this issue. The individual PEI results can be observed in Figures 3.20 and 3.21 for both gear pumps. It is worth noting that higher PEI values indicate better performance, as they reflect higher metric values. Across almost all scenarios and algorithms considered in this thesis, it is evident that smaller

parameter settings led to higher PEI values, emphasising the significant impact of parameter configuration on performance. To further reinforce this finding, Figures 3.22 and 3.23 display the PEI values for all scenarios and algorithms across all parameter settings. Each parameter setting is represented by a distinct colour, resulting in a total of 20 colours. Despite the multitude of colours, the figures maintain clarity and readability. By examining the tallest bars in the charts, it becomes apparent that these correspond to the best-performing algorithms across all cases, offering a concise representation of the superior performance achieved by certain algorithms. Notably, MOBA consistently achieves the highest PEI values for both gear pumps, highlighting its strong overall performance across the evaluated scenarios and parameter settings in solving RDSP.

3.6 Summary

The increasing adoption of robotics in disassembly processes aims to enhance their effectiveness and efficiency compared to manual disassembly. RDSP has emerged as an important area for improving efficiency and cost-effectiveness in disassembly operations. RDSP involves determining the optimal order for disassembling parts and components within a robotic cell, addressing objective 2 of this thesis. To handle the inherent complexity and NP nature of determining the optimal disassembly sequence, metaheuristic algorithms have gained prominence, offering significant advantages over traditional exact methods that struggle with computational intractability due to the NP-completeness of the problem. This chapter utilises the BA, to find the optimal solutions for RDSP. Objective 4 is met by conducting a case study on gear pumps, validating the effectiveness of the proposed approach and sustainability model. Objective 1 is realised by introducing a new sustainability model with the objectives of maximising profit, energy savings, and reducing environmental impact. The research introduces the novel concept of selecting the best recovery option (automatic recovery scenario) for each disassembly component, thereby enhancing

the practicality and usefulness of the proposed model. Three predefined scenarios, namely REC, REM, and REU, along with one scenario defined automatically by the algorithm called ARS, exhibit distinct outcomes for the disassembly sequences based on the recovery options of each part. The ARS scenario yields the highest monetary value, followed by REU, REM, and REC. These results consistently demonstrate a similar pattern across all algorithms employed. These findings provide support for the notion that recycling should be regarded as a last resort for recovery, as it entails higher energy consumption and lower monetary value. To assess the effectiveness of the proposed approach, objective 5 involves analysing outcomes and evaluating the algorithm's performance using the proposed novel SPM and PEI.

Overall, MOBA consistently demonstrate superior performance compared to the other algorithms across multiple performance metrics, including POSs, HI, NFE, and PEI. Their effectiveness in finding solutions that balance conflicting objectives and provide better trade-offs is evident. These findings have significant implications for the selection of suitable algorithms in the context of robotic disassembly sequence planning in various scenarios. The superior performance of MOBA in terms of POSs indicates their ability to generate a greater number of POSs, highlighting their efficiency in exploring the solution space. Additionally, MOBA achieve lower total NFE, implying their efficiency in reaching high-quality solutions with fewer function evaluations. Furthermore, the higher HI values indicate a good convergence and diversity in the objective space. The integration of these metrics into the PEI further supports the superiority of MOBA in providing comprehensive and well-balanced solutions. These findings contribute valuable insights into the selection and application of algorithms for disassembly sequence planning in different scenarios. The superior performance of MOBA across multiple metrics showcases their potential for achieving optimal trade-offs and addressing conflicting objectives. Researchers and practitioners can leverage these findings to make informed decisions when choosing algorithms for similar optimisation problems in the domain of RDSP.

Despite the significant results obtained in this thesis, it is crucial to acknowledge the ongoing

challenge posed by the complexity of RDSP. The model assumptions in this research consider an ideal condition for disassembly, assuming that all parts can be completely disassembled and have deterministic times. However, it is important to recognise that real-world disassembly processes may involve various uncertainties and complexities that go beyond these ideal assumptions. Furthermore, it is necessary to consider the generalisability of the proposed model. The research primarily focuses on specific scenarios and components, such as gear pumps, and incorporates data collected from remanufacturers in England and Spain. While these insights are valuable, it is important to conduct further investigation to determine the applicability and generalisability of the proposed model to a wider range of disassembly processes and contexts.

In light of the complexity of RDSP, it is important to consider this model as a foundational framework for understanding the ideal conditions of disassembly, using real gear pumps as a case study. The presence of conflicting objectives emphasises the need for a nondominated approach to effectively address this problem. Future research should aim to incorporate the stochastic nature of incoming EoL products and explore the use of other EoL products with increased complexity to further enhance the practicality and applicability of the model.

Moreover, subsequent to the publication of this chapter, recent advancements in RDSP research have focused on the utilisation of deep learning techniques, and one publication has presented the application of digital twins technology. These emerging technologies offer potential solutions to tackle the complexities and uncertainties inherent in RDSP. By incorporating AI, it becomes possible to achieve more accurate predictions and improve decision-making in disassembly processes. Additionally, digital twins technology enables virtual modelling and simulation of the disassembly operations. Therefore, it is recommended that future research integrates these AI techniques and digital twins approaches to further enhance the practicality and effectiveness of RDSP.

Overall, the research highlights the significance of utilising robotics and metaheuristic algorithms in RDSP to enhance efficiency and effectiveness in disassembly processes. It introduces

a new sustainability model, considers multiple objectives, use the bees algorithm and incorporates a novel performance evaluation index. Future research must consider the complexity of RDSPs, the applicability of the proposed model to various EoL products, and the most recent technological and tool advancements.

Chapter 4

Sequence-Dependent Robotic Disassembly Line Balancing

In the preceding chapter, the interconnection of disassembly line balancing and disassembly sequence planning was highlighted. Disassembly sequence plans aim to determine the optimal sequence for disassembly, while disassembly line balancing focuses on achieving a balanced production line. By carefully designing sequence plans and achieving line balance, the efficiency of the disassembly process can be enhanced [47–49]. This chapter is dedicated to addressing the research problem of sequence-dependent robotic disassembly line balancing (RDLBSD), which will be further explained in the subsequent paragraphs. By focusing on this problem, the chapter achieves objective 3 of the thesis. Moreover, the chapter also encompasses objectives 1, 4, and 5, which are as follows: (1) to develop a sustainability model and recovery scenarios for the end-of-life (EoL) products in the RDLBSD problem, (2) to validate the effectiveness of the proposed approach through a case study involving gear pumps; and (3) to determine the optimal parameter settings and performance metrics for optimisation algorithms employed in the RDLBSD problem.

The requirement to establish a feasible disassembly sequence for the product under investigation poses a significant constraint on disassembly line balancing [41]. The result is

modelled as an n-dimensional integer, deterministic, multi-criteria decision-making problem with an exponentially growing search space, where n represents the number of parts to be removed [41]. This situation gives rise to two key challenges in the field of disassembly line balancing. The first challenge involves finding a feasible disassembly sequence that adheres to the constraints and requirements of the product under investigation. Achieving a feasible disassembly sequence becomes increasingly complex as the number of parts to be disassembled grows exponentially, leading to a combinatorial explosion in the search space. In disassembly line balancing research, the focus has traditionally been on finding a feasible disassembly sequence and addressing the subsequent line balancing problem. However, recent studies have shed light on the importance of simultaneously addressing both sequence feasibility and line balance. This holistic approach recognises the interdependence between sequencing and line balancing and aims to optimise both aspects simultaneously. By considering the order in which parts are disassembled and the assignment of tasks to workstations, researchers aim to minimise idle times, improve efficiency, and achieve a well-balanced disassembly line. The concept of sequence-dependent disassembly line balancing has gained considerable attention in both manual and robotic disassembly research. Manual disassembly line balancing studies have explored the integration of sequencing and line balancing techniques to improve the efficiency of manual disassembly processes [130–135]. Similarly, in robotic disassembly, researchers have investigated sequence-dependent line balancing strategies to optimise the performance of robotic disassembly systems [47–49, 51, 52]. Overall, the integration of sequence feasibility and line balance in disassembly line balancing research reflects a shift towards a more comprehensive and optimised approach. By considering both aspects simultaneously and leveraging advancements in manual and robotic disassembly techniques, researchers aim to achieve efficient and effective disassembly processes while maintaining product integrity and minimising waste.

The disassembly sequence problem and the disassembly line balancing problem are NP-complete problems, implying that computing optimal solutions for large-scale instances is

a challenging and time-consuming process [41]. As previously stated, these issues display exponential growth in the search space, thereby compounding their complexity. Addressing this second challenge requires the adoption of efficient solution methods that can handle the complexity of the problem within reasonable time frames. Researchers have resorted to metaheuristic algorithms as a potential solution to this challenge. Metaheuristic algorithms utilise intelligent search strategies to efficiently explore solution spaces and provide approximate solutions, which are faster than exact methods. These algorithms are suitable for solving complex problems where finding optimal solutions is impractical within time constraints. Table 2.2 in Chapter 2 provides a summary of the metaheuristic algorithms commonly used in robotic disassembly line balancing (RDLB) research, highlighting their applicability and effectiveness in addressing the problem's complexity. The BA is one of the algorithms used to solve optimisation problems in RDLB. Recognised for its ability to handle the complexity of the RDLB problem [47, 49, 51], the BA has established itself as a viable and effective method for addressing this challenge. This facilitates the optimisation of RDLBSD processes, using the MOBA.

Moreover, this chapter contributes a novel sustainability model, as illustrated in Table 2.2 of the research position, thereby accomplishing objective 1 of the study. Although the four sustainability scenarios employed in the previous chapter remain unchanged, the sustainability model has been specifically tailored to address the challenges of the RDLBSD problem. This integration of sustainability considerations within the context of robotic disassembly underscores the area where previous studies were not considered previously. Consequently, this chapter effectively addresses objectives 1 and 4 of the thesis, which entail the practical application of the sustainability model and the thorough examination of specific case studies. This cohesive approach aligns with the overarching theme of the thesis. Furthermore, the continued utilisation of the same novel performance indicator, previously introduced in the preceding chapter, demonstrates a consistent adherence to the central theme of the thesis. This alignment is congruent with objective 5, which centres on the effective utilisation and rigorous evaluation of performance metrics for optimisation

algorithms.

Overall, this research makes several contributions to the field of RDLBSD. First, the enhanced BA is further adapted for the RDLBSD problem. The rest of the following contributions are: the introduction of a sustainability model specifically tailored for the area of RDLB, addressing a gap in previous research. Furthermore, the research incorporates an autonomous recovery strategy identification, similar to the approach adopted in the Chapter 3, enabling dynamic evaluation and selection of optimal recovery strategies based on given constraints and objectives. The research employs a realistic simulation approach based on reliable data collected from relevant sources, enhancing the applicability of the proposed model and its alignment with real-world problems. Lastly, the utilisation of a proposed performance indicator introduced in the previous chapter, which simplifies decision-making processes, is a novel contribution in the research area. Collectively, these contributions present a comprehensive solution to the RDLBSD problem.

The content of this chapter is organised as follows. Section 4.1 presents the research model and methodology, outlining the approach used to address the research objectives. In Section 4.2, the experiments and results are presented, providing an analysis of the collected data. A detailed discussion of the findings is provided in Section 4.3, where the implications and insights derived from the results are thoroughly examined. Finally, Section 4.4 summarises the main findings of the chapter and offers recommendations for practitioners and suggestions for future research. The conclusion of the chapter highlights the key takeaways and emphasises the importance of research in advancing the research area of RDLBSD.

4.1 Model and methodology

In this chapter, a similar four-stage approach to the one used in the robotic disassembly sequence planning (RDSP) framework is adopted. The initial stage entails the construction of a model

using the Modified Feasible Solution Generation (MFSG) technique, which closely aligns with the process described in Section 3.3.1. However, there are notable differences in the second and fourth stages compared to the preceding stages. In the second step, multiobjective nondominated (MO-ND) approaches are employed, consistent with previous research on RDLB literature, where the consideration of multiple objectives necessitates the use of MO-ND approach. Finally, in the last stage, the performance evaluation index (PEI), introduced in Section 3.2.2, is utilised as a novel approach for evaluating performance without relying solely on statistical testing methods. This integration of the PEI provides a valuable contribution to the field by offering a comprehensive and robust performance evaluation framework.

4.1.1 RDLBSD Model Building

The process of constructing the RDLBSD model follows the same approach as the previous RDSP model (see Chapter 3.3.1), as it leverages the same case studies and sustainability model. However, the RDLBSD model introduces a fourth objective, which aims to minimise line imbalances within the disassembly line. Figure 4.1 provides a visual representation of the robotic disassembly line, highlighting the differences compared to the RDSP robotic station. The disassembly line comprises three KUKA robots, a gear pump and a conveyor system that operate in a coordinated sequence to execute the disassembly tasks efficiently. Figure 4.2 shows an example of RDLBSD output.

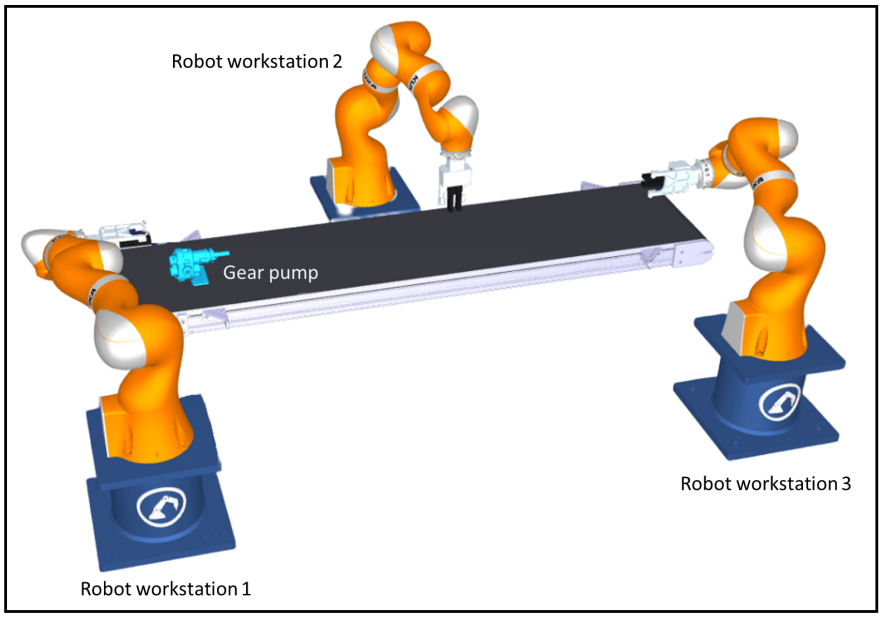


Figure 4.1: Robotic workstations illustration, created using RoboDK

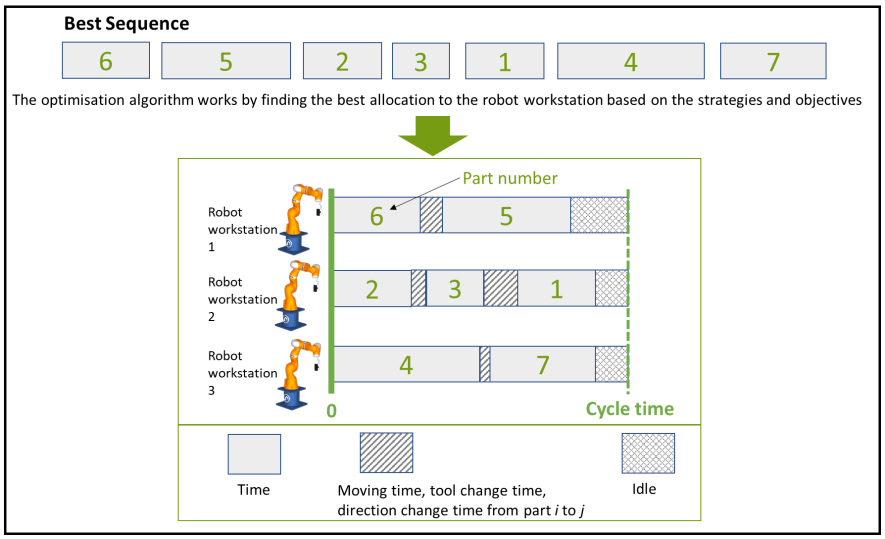


Figure 4.2: Example output of RDLBSD

4.1.2 RDLBSD Model Formulation

The model formulation in this chapter builds upon the sustainability model introduced in the previous chapter, which involves maximising profit (f_1), energy savings (f_2), and environmental impact reduction (f_3). Additionally, a fourth objective is incorporated, aiming to minimise unbalanced lines (f_4) within the disassembly process. By considering all four objectives simultaneously, the model seeks to optimise the disassembly line's performance by balancing these four goals. In line balancing, achieving a state of perfect equilibrium with no idle times is considered the ideal scenario [41]. This practise imposes penalties on solutions that have long idle times, thereby equalising the workload distribution among workstations. The minimum numerical performance value represents the optimal solution with the fewest workstations and balanced idle times. Line balancing methods aim to reduce the number of workstations and equalise idle times, resulting in a nonlinear objective function [41]. This measurement is represented by Equation (4.1) [41], which quantifies the level of balance within the disassembly line. The cumulative operational duration of the robotic workstations (S_T) should not exceed the cycle time (Cy_T) of the disassembly line. The cycle time for gear pumps A and B is set as 210 and 320 seconds, respectively.

$$f_4 = \sum_{i=1}^{NWS} (Cy_T - S_{T_i})^2 \tag{4.1}$$

where:

- f_4 is unbalanced line
- Cy_T is the cycle time
- NWS is the number of workstations
- S_T is the station time

The sustainability recovery scenarios (REC, REM, REU, ARS) that were utilised in the RDSP study are also employed in the RDLBSD study. These scenarios are consistently used throughout

the thesis to evaluate and compare different recovery approaches and strategies.

4.1.3 RDLBSD (MO nondominated approach)

The literature review conducted in Chapter 2 highlights the prevalent use of MO-ND approaches in previous research within the area of RDLB. Out of the total 37 articles reviewed, 24 employed MO-ND methods, indicating their prominence in addressing the complexities in this area. In contrast, a smaller number of studies (3 articles) utilises an MO aggregate approach, which aggregates multiple objectives into an SO function. Furthermore, a limited number of articles (5 articles) focuses on SO, which aims to optimise a single criterion. This analysis of the literature reveals a clear preference for MO-ND approaches, underscoring their effectiveness in handling the multiple conflicting objectives inherent to RDLB. This observation can be attributed to the intrinsic nature of the RDLB problem, which necessitates the optimisation of multiple objectives concurrently. Drawing upon these findings, this section of the study adopts an MO-ND approach to address the intricate complexities associated with the RDLB problem. Furthermore, the findings presented in Chapter 3 clearly indicate that the nondominated approach is more suitable for this model compared to the aggregate approach. This finding underscores the importance of considering the trade-offs between multiple goals.

The RDLBSD problem introduces a distinct approach compared to RDLB. In RDLB, the focus is primarily on finding a feasible disassembly sequence for the line and subsequently addressing the task of line balancing. However, in RDLBSD, the challenge lies in simultaneously considering both the feasibility of the disassembly sequence and achieving line balance. This means that in RDLBSD, the optimisation process takes into account the interplay between the sequencing of disassembly tasks and the distribution of these tasks across workstations to achieve a well-balanced disassembly line. By addressing both sequence feasibility and line balance simultaneously, RDLBSD aims to optimise the efficiency and effectiveness of the disassembly

process. It ensures that the sequences not only satisfies feasibility requirements but also optimally allocates disassembly tasks to minimise unbalanced line and maximise profit, energy savings, and environmental impact savings. This integration of sequence feasibility and line balance distinguishes RDLBSD from RDLB, allowing for a more comprehensive and optimised approach to robotic disassembly line balancing.

In the context of optimising robotic disassembly line balancing within sequence-dependent scenarios, a noteworthy analogy can be drawn to the well-established bin packing problem. In this comparison, the 'robotic workstation' serves as an analogue to the 'bin' in the classical bin packing problem, wherein items are strategically placed within containers. In our context, the 'bin capacity' transmutes into the 'cycle time' constraint, a pivotal limitation that each robotic workstation must adhere to, mandating that the cumulative processing time remains within acceptable bounds. Similarly, the 'robotic disassembly time,' analogous to the 'item sizes' in bin packing, represents the temporal resources required for individual disassembly tasks. This temporal aspect takes on added significance due to its influence on overall system efficiency. Central to this matter is "task allocation", which parallels "item placement" by assigning robotic disassembly tasks to specific robotic workstations while considering the constraints and objectives. Notably, this comparison isn't devoid of nuance. Unlike the unrestricted assignment nature of the traditional bin packing problem, the 'robotic disassembly assignment' requires adherence to 'feasible disassembly sequences.' Consequently, the challenge extends beyond mere allocation; it encompasses constructing sequences that ensure not only the temporal feasibility but also the logical integrity of the disassembly process.

To illustrate, the optimal robotic disassembly sequence emerges as "6-5-2-3-1-4-7" (see Figure 4.2), with respective times of 20s, 25s, 15s, 8s, 10s, 28s, and 10s, within a cycle time of 50s. For the sake of simplicity in this example, the moving time, encompassing disassembly time, tool change time, and direction change time, is considered 1s. The assignment process allocates parts to robotic workstations based on the stipulated cycle time of 50 seconds. As such, the initial assignment

designates parts 6 and 5 to the first robotic workstation. Following this, parts 2, 3, and 1 are allocated to the second workstation. Lastly, the third workstation accommodates parts 4 and 7.

Algorithm 4 presents the pseudo-code for the MOBA, outlining the step-by-step process for solving the RDLBSD problem using an MO-ND approach. Given that this research serves as a further development of the disassembly sequence plans, the MOBA steps explained in the algorithms exhibit a degree of similarity to Chapter 3. Nonetheless, a marked divergence becomes apparent as the focal point transitions towards the disassembly line, introducing a new objective and constraints. The initialisation following similar steps as previously described in Chapter 3. The same parameter settings as in the previous chapter were used. The scout bees were generated using the MFSG techniques, which resulted in feasible disassembly solutions, and then they were used to generate the robotic assignment as described earlier. The scout bees then explore the objective space, and the quality of the sites is sorted by the cost value using nondominated sorting. The selected sites (m) are the best of that sorting, which consists of 'elite sites (e)' and the 'other selected sites ($m - e$)' chosen to do an exploitation of the sites. In the local search, the recruited bees for elite sites (nep) exploit the elite sites, and the recruited bees for other selected sites (nsp) exploit the other selected sites. The swap, insert, and mutation operators are the same as those in Chapter 3. The difference is that in Chapter 3, swap, insert, and mutation are applied to the whole sequence. In this chapter, swap, insert, and mutation operations were executed on sequences within the robotic workstations. The procedure is repeated until robotic workstation disassembly sequences are feasible. The remaining scout bees ($n - m$) are exploring the objective space. The fitness of the newly formed population was assessed, and a sorting process based on nondominated sorting was performed. Subsequently, the best bee was preserved for progression into the next iteration. The iterative process continues until the predetermined stopping criterion, which is the achievement of the maximum iteration count, is met.

Algorithm 4: The pseudo-code of MOBA for RDLBSD

Input : n: number of scout bees, m: number of selected sites, e: number of elite sites, nsp: recruited bees for other selected sites, nep: recruited bees for elite sites, dis_m: robotic disassembly information matrix
Output: RDLBSD(sequence, direction, mode, tool, robotic workstation, $POSs$)

Function MOBA (n, m, e, nsp, nep) **:**
  Start
  initialRDSP ← GlobalMFSG($dis\_m : sequence, direction, mode, tool$) `// Generate initial RDSP with feasible disassembly sequences`
  initialRDLBSD ← RoboticAssignment(**initialRDSP**) `// Generate robotic disassembly line solutions based on initial RDSP and Pareto front set`
  while *stopping criterion not met* **do**
    Evaluate population fitness
    f ← FVALUE(**initialRDLB**)
    Sort population based on nondominated sorting
    Select m sites for local search
    `// Generate local sites with waggle dance`
    for *EliteSite*(1 *to* e) **do**
      `// Assign best elite site bee`
      $BestEliteSiteBee \leftarrow$ *the scout bee that found the elite site*
      for *RecruitedEliteSiteBee*(1 *to* nep) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedEliteSiteBee ← WaggleDance($dis\_m : sequence, direction, mode$)
        end
      end
      `// Mutate the disassembly direction and mode`
      RecruitedEliteSiteBee ← Mutation($dis\_m : direction, mode$)
      Evaluate fitness of **RecruitedEliteSiteBee** based on nondominated sorting
      if *RecruitedEliteSiteBee* *is better than* ***BestEliteSiteBee*** **then**
        `// Update BestEliteSiteBee`
        BestLocalBee ← **RecruitedEliteSiteBee**
      end
    end
    for *OtherSelectedSite*(1 *to* $(m - e)$) **do**
      `// Assign best other selected site bee`
      $BestOtherSelectedSiteBee \leftarrow$ *the scout bee that found the other selected site*
      for *RecruitedOtherSelectedSiteBee*(1 *to* nsp) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedOtherSelectedSiteBee ← WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction and mode`
        RecruitedOtherSelectedSiteBee ← Mutation($dis\_m : direction, mode$)
        Evaluate fitness of **RecruitedOtherSelectedSiteBee** based on nondominated sorting
        if *RecruitedOtherSelectedSiteBee* *is better than* ***BestOtherSelectedSiteBee*** **then**
          `// Update BestOtherSelectedSiteBee`
          BestLocalBee ← **RecruitedOtherSelectedSiteBee**
        end
      end
    end
    `// Assign remaining scout bees for global search`
    for *RemainingScoutBee*(1 *to* $(n - m)$) **do**
      RemainingScoutBeeRDSP ← GlobalMFSG($dis\_m : sequence, direction, mode$)
      GlobalRDLBSD ← RoboticAssignment(**RemainingScoutBeeRDSP**)
    end
    Evaluate fitness of the new population
    Sort population based on nondominated sorting
    `// Store the Pareto frontier`
    Best RDLBSD ← BestBee
  end
  return *Best RDLBSD (BestBee)*

4.1.4 Performance Evaluation

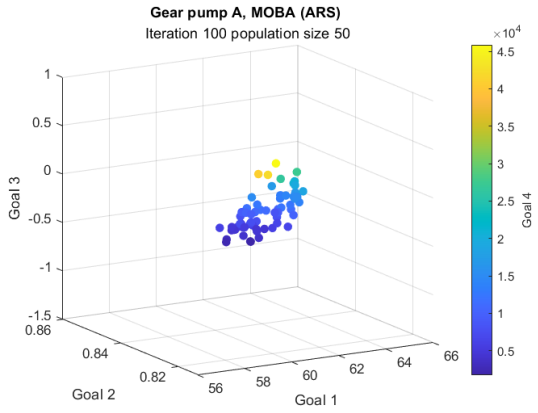
The calculation of the PEI in this chapter is based on Equation (3.1), which was introduced and discussed in Section 3.2.2. The PEI serves as a valuable tool for evaluating and comparing the performance of different optimisation algorithms. As previously explained, the PEI provides a simple yet effective visualisation and resolution for the conflicting performance metrics that are commonly encountered in MO-ND optimisation problems. The PEI simplifies the evaluation of algorithm performance and aids decision-makers in making informed choices by consolidating multiple decision-making criteria into a single index.

4.2 Experimental results

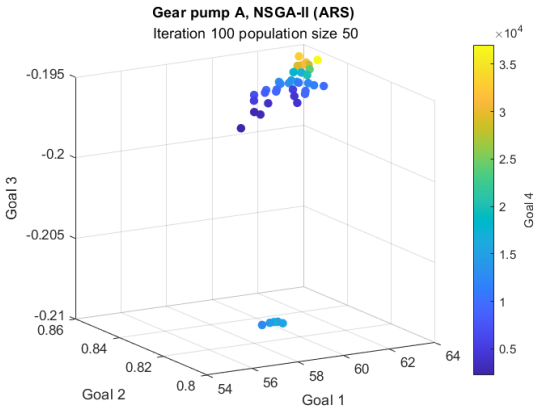
The experiments in this chapter were conducted using the same platform as the previous chapter, MATLAB 2020b on the BEAR cloud service offered by the University of Birmingham. The statistical tests were performed using IBM SPSS version 29. The parameter settings for all algorithms used in this chapter, including the number of iterations and population size, remained consistent with those used in the Chapter 3. The focus of this section is to provide a clear and comprehensive representation of the experimental outcomes, specifically highlighting the results from iteration 100 with a population size of 50 for the ARS scenario. These particular settings were deliberately chosen to emphasise the performance of the algorithms under the smallest parameter settings.

Figures 4.3 and 4.4 display the POSs obtained by the MOBA, NSGA-II, and PESA-II. These figures provide a visual representation of the trade-offs between the different objectives in the RDLBSD. By plotting the solutions in the objective space, the figures demonstrate the diverse range of solutions that exist, allowing decision-makers to select the most suitable solution based

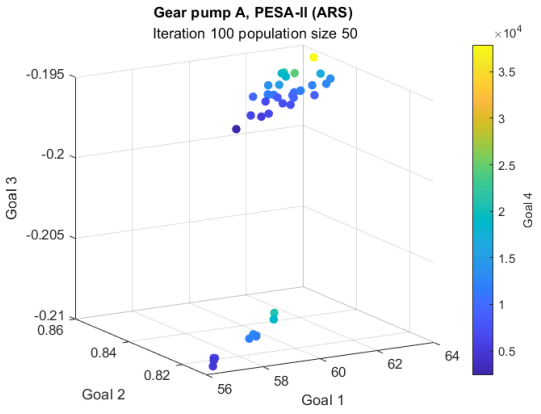
on their preferences and priorities. Due to the large number of POSs from gear pump A (more than 60), the decision is made to present the example output taken from gear pump B in Table 4.1. This table illustrates the detailed disassembly output information generated by the MOBA in the ARS scenario using 100 iterations and 50 populations. The table offer comprehensive insights into the disassembly process, providing specific details about the disassembly sequence, direction, recovery mode, tools, and robotic stations for each solution. This information allows researchers and practitioners to analyse and compare the characteristics and feasibility of different disassembly solutions generated by the algorithms.



(a) MOBA

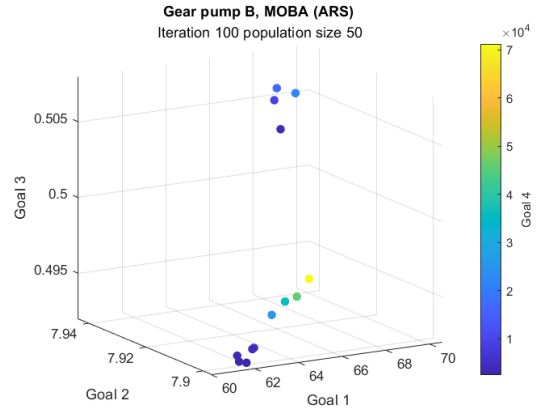


(b) NSGA-II

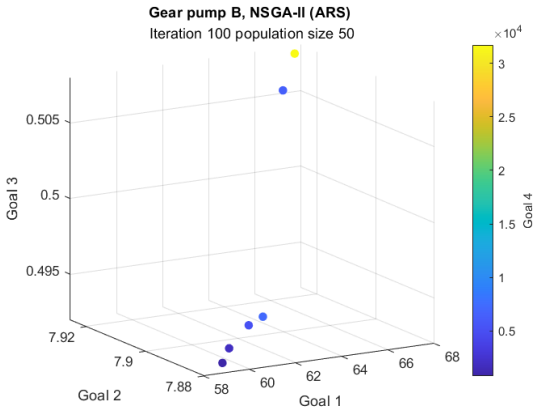


(c) PESA-II

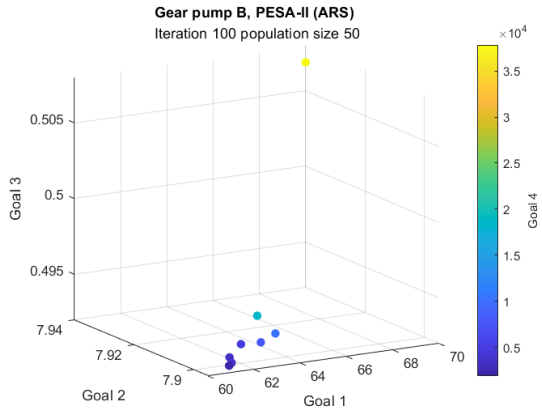
Figure 4.3: RDLBSD POSs (ARS scenario) - Gear pump A



(a) MOBA



(b) NSGA-II



(c) PESA-II

Figure 4.4: RDLBSD POSs (ARS scenario) - Gear pump B

The remaining presentation of the experiment output in this section closely aligns with that of Chapter 3, serving the purpose of enabling a seamless comparison and comprehensive examination of the results. This consistent presentation format ensures ease of interpretation and analysis. Figures 4.5 and 4.6 present the number of POSs for Gear Pumps A and B across all scenarios and parameter settings. A consistent pattern emerges in the relationship between the NFE and the parameter settings, whereby smaller parameter settings correspond to lower NFE values, while higher parameter settings result in higher NFE values. To provide an overview of the total NFE for Gear Pumps A and B, Figures 4.7 and 4.8 are provided, while detailed results can be found in Appendix B. The HI depicted in Figures 4.9 and 4.10 shows its values for Gear Pumps A and B throughout the entire experiment. These figures encompass the use of three MO-ND optimisation algorithms: MOBA, NSGA-II, and PESA-II, across all four scenarios. As previously explained, a higher HI is considered desirable, as it indicates a better performance in terms of convergence and diversity. Furthermore, the PEI, is also desirable to have a higher value, as it signifies better overall performance across multiple metrics. Figures 4.11 and 4.12 present the PEI results for Gear Pumps A and B, respectively, reflecting the experiments conducted with the algorithms employed in each scenario. For a more comprehensive interpretation, Figures 4.13 and 4.14 display the total PEI values for each scenario and algorithm.

Table 4.1: Example RDLBSD output of Gear pump B (ARS Scenario)

Disassembly Sequence	3-23-1-4-2-5-21-6-19-24-7-8-11-22-9-10-20-18-17-13-12-16-15-14	6-23-21-1-19-4-3-5-2-7-8-24-9-22-11-10-20-18-13-12-14-15-16-17	23-1-6-21-2-5-3-4-24-7-9-22-10-20-8-11-19-18-17-16-13-12-14-15
Disassembly Direction	2-1-2-2-2-2-1-2-1-1-2-2-2-1-2-2-1-1-1-1-2-1-1-1	2-1-1-2-1-2-2-2-2-2-2-1-2-1-2-2-1-1-1-2-2-2-2-1	1-2-2-1-2-2-2-2-1-2-2-1-2-1-2-2-1-1-1-1-1-2-2-2
Recovery Option	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-1-4-1-1-4-4-4	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-1-1-1-4-4-4-4-4	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-4-1-1-4-4
Disassembly Tool	1-3-1-1-1-1-3-1-2-3-5-4-4-3-4-4-2-4-4-4-5-4-4-4	1-3-3-1-2-1-1-1-1-5-4-3-4-3-4-4-2-4-4-5-4-4-4-4	3-1-1-3-1-1-1-1-3-5-4-3-4-2-4-4-2-4-4-4-4-5-4-4
Robotic Workstation	1-1-1-1-1-1-1-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3	1-1-1-1-1-2-2-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3	1-1-1-1-1-1-1-1-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3
goal_1	61.428	63.061	61.594
goal_2	7.898	7.905	7.897
goal_3	0.492	0.492	0.492
goal_4	3005.47	5522.20	2451.59
Disassembly Sequence	4-1-24-6-22-20-3-5-23-21-19-18-2-7-17-9-13-16-15-14-8-10-11-12	2-3-4-1-5-23-24-22-6-21-7-8-20-19-11-9-10-18-17-16-15-13-12-14	23-2-4-21-6-3-5-1-19-7-8-10-24-9-22-20-18-11-17-13-12-16-15-14
Disassembly Direction	2-2-1-2-1-1-2-2-1-1-1-1-2-2-1-2-1-1-1-1-2-2-2-2	2-2-2-2-2-1-1-1-2-1-2-2-1-1-2-2-2-1-1-1-1-1-2-1	1-2-2-1-2-2-2-2-2-1-2-2-2-1-2-1-1-1-2-1-1-2-1-1
Recovery Option	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-4-4-4-4-1-1-1	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-1-4-4-4-1-1-4	1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-1-1-4-1-1-4-4-4
Disassembly Tool	1-1-3-1-3-2-1-1-3-3-2-4-1-5-4-4-4-4-4-4-4-4-4-5	1-1-1-1-1-3-3-3-1-3-5-4-2-2-4-4-4-4-4-4-4-4-5-4	3-1-1-3-1-1-1-1-2-5-4-4-3-4-3-2-4-4-4-4-5-4-4-4
Robotic Workstation	1-1-1-1-1-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3-3-3	1-1-1-1-1-1-1-1-1-1-2-2-2-2-2-2-2-2-2-2-2-3-3-3	1-1-1-1-1-1-1-1-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3
goal_1	63.274	70.561	62.899
goal_2	7.898	7.944	7.905
goal_3	0.507	0.492	0.492
goal_4	3936.33	71235.74	4223.05
Disassembly Sequence	24-6-5-1-3-22-2-20-23-4-7-8-10-9-21-11-19-18-13-12-17-16-14-15	5-4-2-24-23-6-22-1-20-21-3-7-19-11-8-10-18-9-13-17-16-15-14-12	23-4-2-5-6-1-24-21-19-22-3-7-20-11-18-17-10-9-13-8-12-16-14-15
Disassembly Direction	1-2-2-2-2-1-2-1-1-2-2-2-2-2-2-1-2-1-1-1-2-1-1-2-2	2-2-2-1-1-2-1-2-1-1-2-2-1-2-2-2-1-2-1-1-1-1-1-1	1-2-2-2-2-2-1-1-1-1-2-2-1-2-1-1-2-2-1-2-2-1-2-2
Recovery Option	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-1-1-1-1-4-4-4-4	1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-4-4-4-4-4-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-4-4-4
Disassembly Tool	3-1-1-1-1-3-1-2-3-1-5-4-4-4-3-4-2-4-4-5-4-4-4-4	1-1-1-3-3-1-3-1-2-3-1-5-2-4-4-4-4-4-4-4-4-4-4-5	3-1-1-1-1-1-3-3-2-3-1-5-2-4-4-4-4-4-4-4-5-4-4-4
Robotic Workstation	1-1-1-1-1-1-1-1-2-2-2-2-2-2-2-2-3-3-3-3-3-3-3-3	1-1-1-1-1-1-1-1-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3	1-1-1-1-1-1-1-1-1-2-2-2-2-2-2-2-3-3-3-3-3-3-3-3
goal_1	61.846	66.718	68.694
goal_2	7.902	7.919	7.934
goal_3	0.492	0.507	0.492
goal_4	3563.04	21580.52	45508.66
Disassembly Sequence	24-4-23-6-3-22-20-2-5-21-19-18-17-1-7-13-8-10-11-16-15-9-14-12	2-5-3-6-4-24-23-1-22-7-21-19-10-11-9-8-20-18-13-12-14-15-17-16	
Disassembly Direction	1-2-1-2-2-1-1-2-2-1-1-1-1-2-2-1-2-2-2-1-1-2-1-1	2-2-2-2-2-1-1-2-1-2-1-1-2-2-2-2-1-1-1-2-2-2-1-2	
Recovery Option	1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-4-1-1-4-4-1-4-4-1	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-4-4-4-4	
Disassembly Tool	3-1-3-1-1-3-2-1-1-3-2-4-4-1-5-4-4-4-4-4-4-4-4-5	1-1-1-1-1-3-3-1-3-5-3-2-4-4-4-4-2-4-4-5-4-4-4-4	
Robotic Workstation	1-1-1-1-1-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3-3-3-3	1-1-1-1-1-1-1-1-1-1-2-2-2-2-2-2-2-2-3-3-3-3-3-3	
goal_1	65.422	67.888	
goal_2	7.916	7.932	
goal_3	0.507	0.492	
goal_4	8765.01	35963.69	
Disassembly Sequence	23-24-2-6-5-4-1-22-20-3-21-19-18-7-13-17-8-9-10-16-15-11-12-14	4-23-1-21-6-5-3-2-24-19-7-22-8-10-11-20-18-17-9-13-16-15-14-12	
Disassembly Direction	1-1-2-2-2-2-2-1-1-2-1-1-1-2-1-1-2-2-2-1-1-2-2-2	2-1-2-1-2-2-2-2-1-1-2-1-2-2-2-1-1-1-2-1-1-1-1-2	
Recovery Option	1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-4-4-1-1-4-4-1-1-4	1-1-1-1-1-1-1-1-1-1-1-1-4-1-1-1-1-4-1-1-4-4-4-1	
Disassembly Tool	3-3-1-1-1-1-1-3-2-1-3-2-4-5-4-4-4-4-4-4-4-4-4-5-4	1-3-1-3-1-1-1-1-3-2-5-3-4-4-4-2-4-4-4-4-4-4-4-5	
Robotic Workstation	1-1-1-1-1-1-1-1-1-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3	1-1-1-1-1-2-2-2-2-2-2-2-3-3-3-3-3-3-3-3-3-3-3-3	
goal_1	66.349	66.484	
goal_2	7.924	7.924	
goal_3	0.492	0.507	
goal_4	25960.74	16956.98	

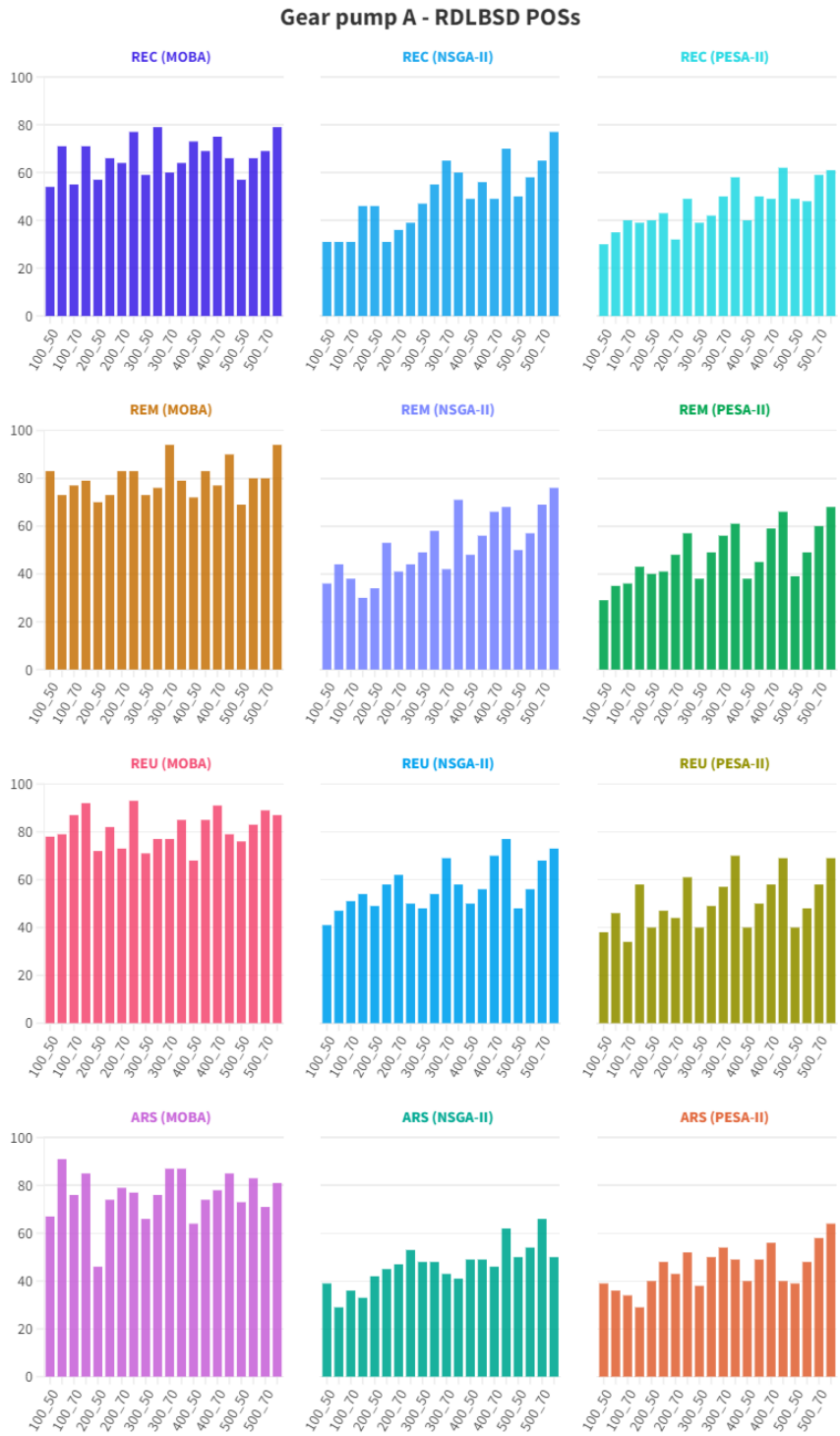


Figure 4.5: Number of Pareto optimal solutions for RDLBSD of Gear Pump A

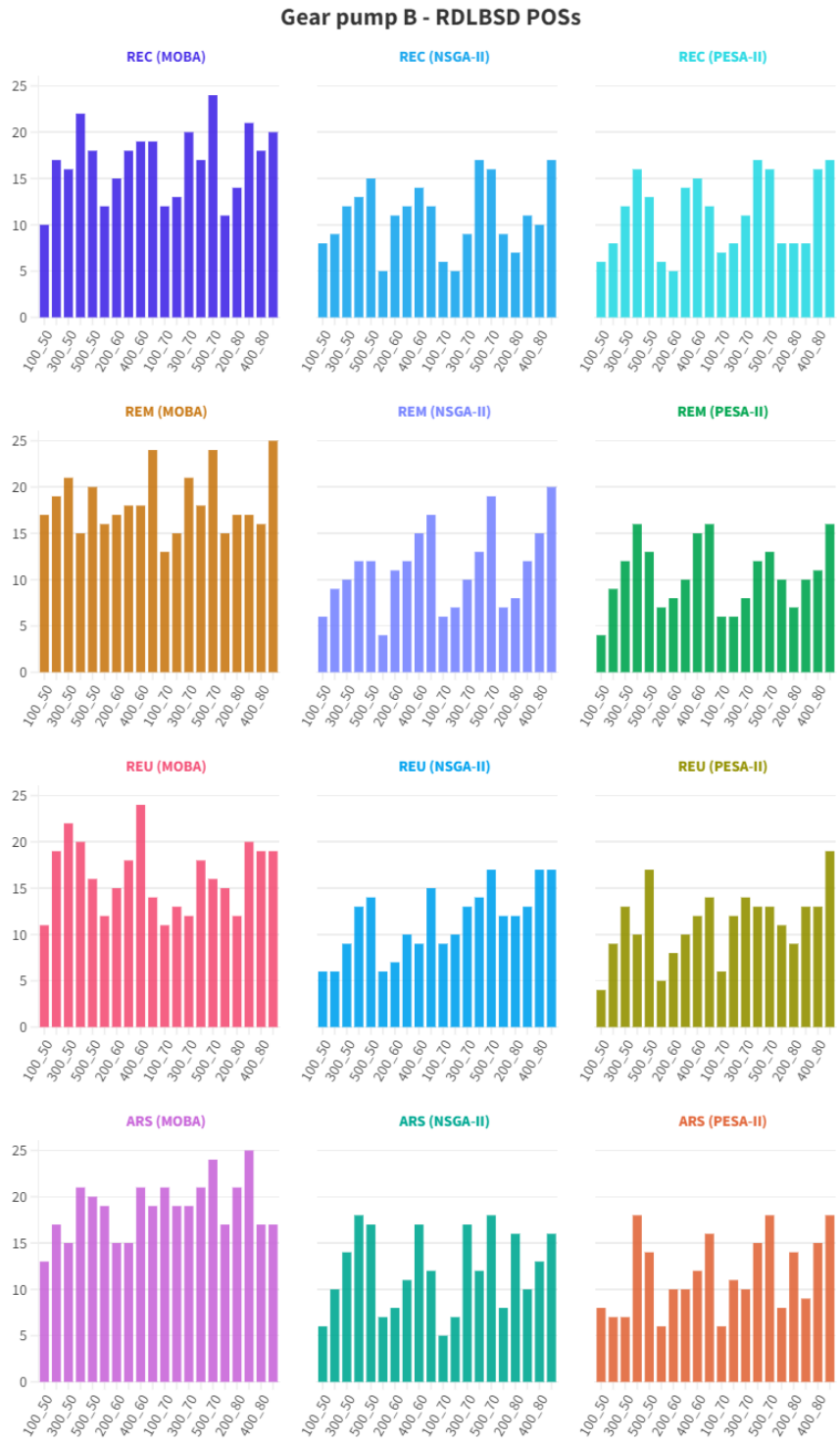


Figure 4.6: Number of Pareto optimal solutions for RDLBSD of Gear Pump B

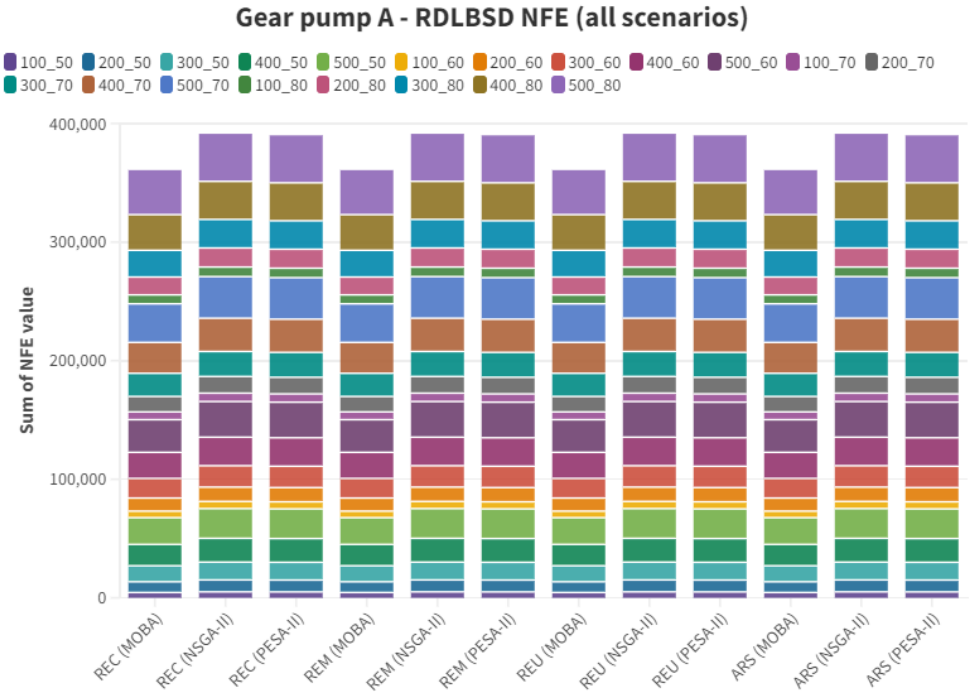


Figure 4.7: Number Function of Evaluation for RDLBSD of Gear Pump A: The lower the better

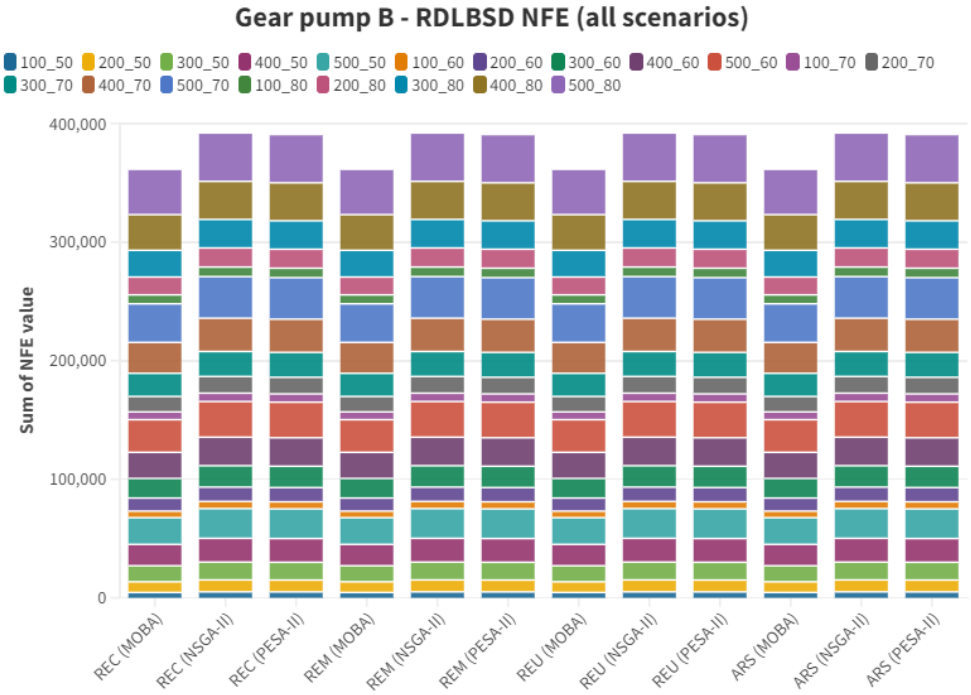


Figure 4.8: Number Function of Evaluation for RDLBSD of Gear Pump B: The lower the better

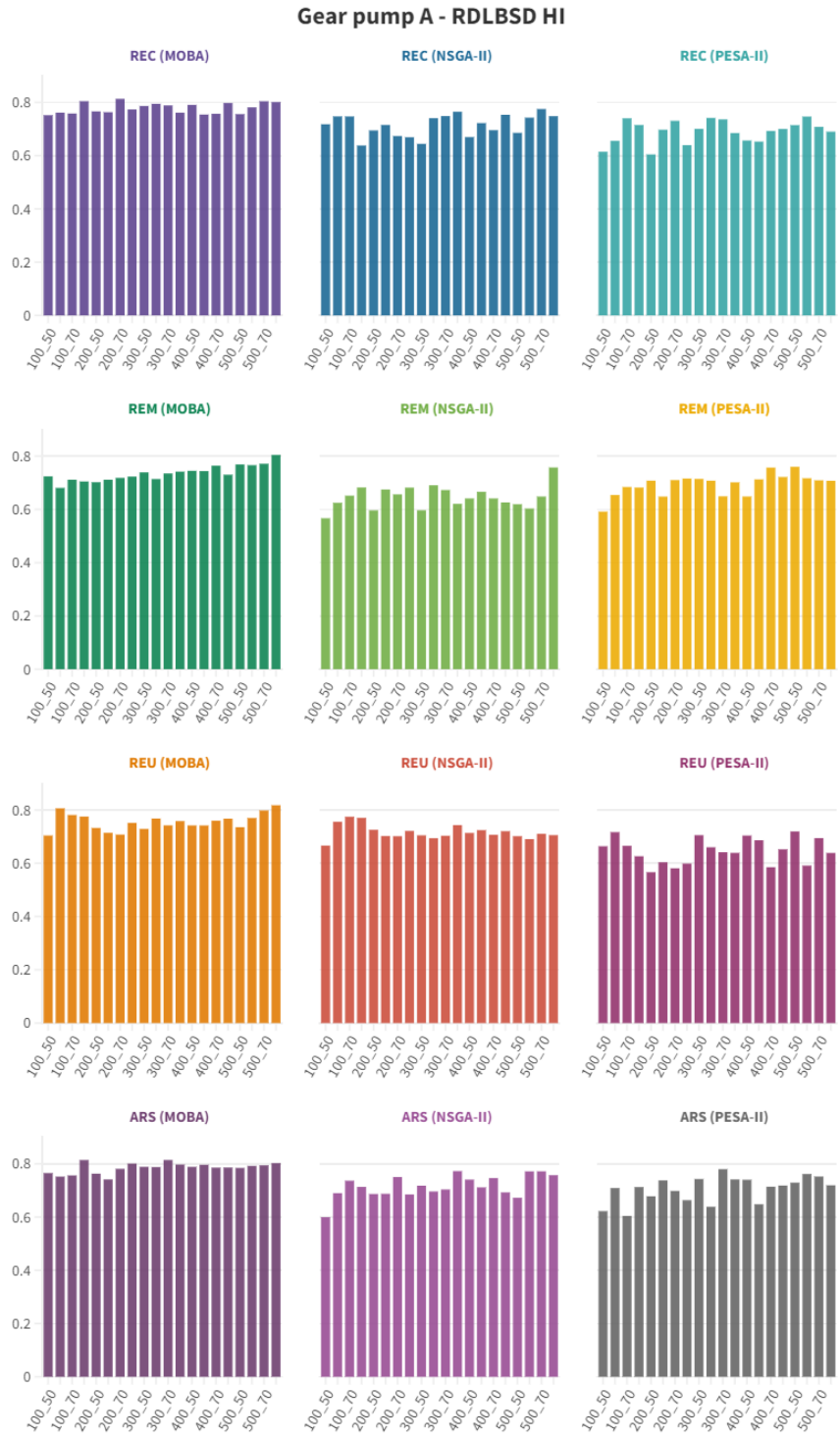


Figure 4.9: Hypervolume Indicator for RDLBSD of Gear Pump A: The higher the better

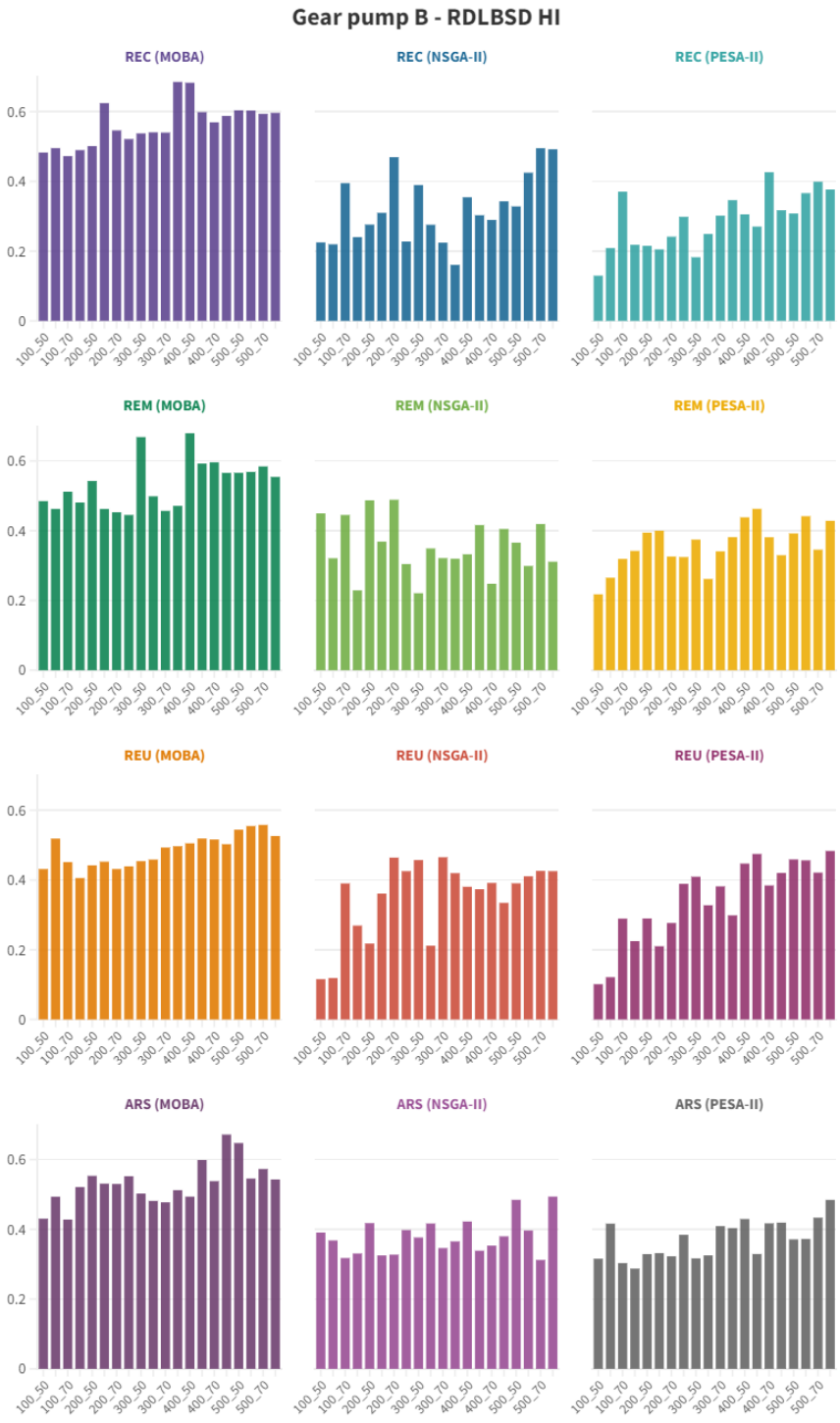


Figure 4.10: Hypervolume Indicator for RDLBSD of Gear Pump B: The higher the better

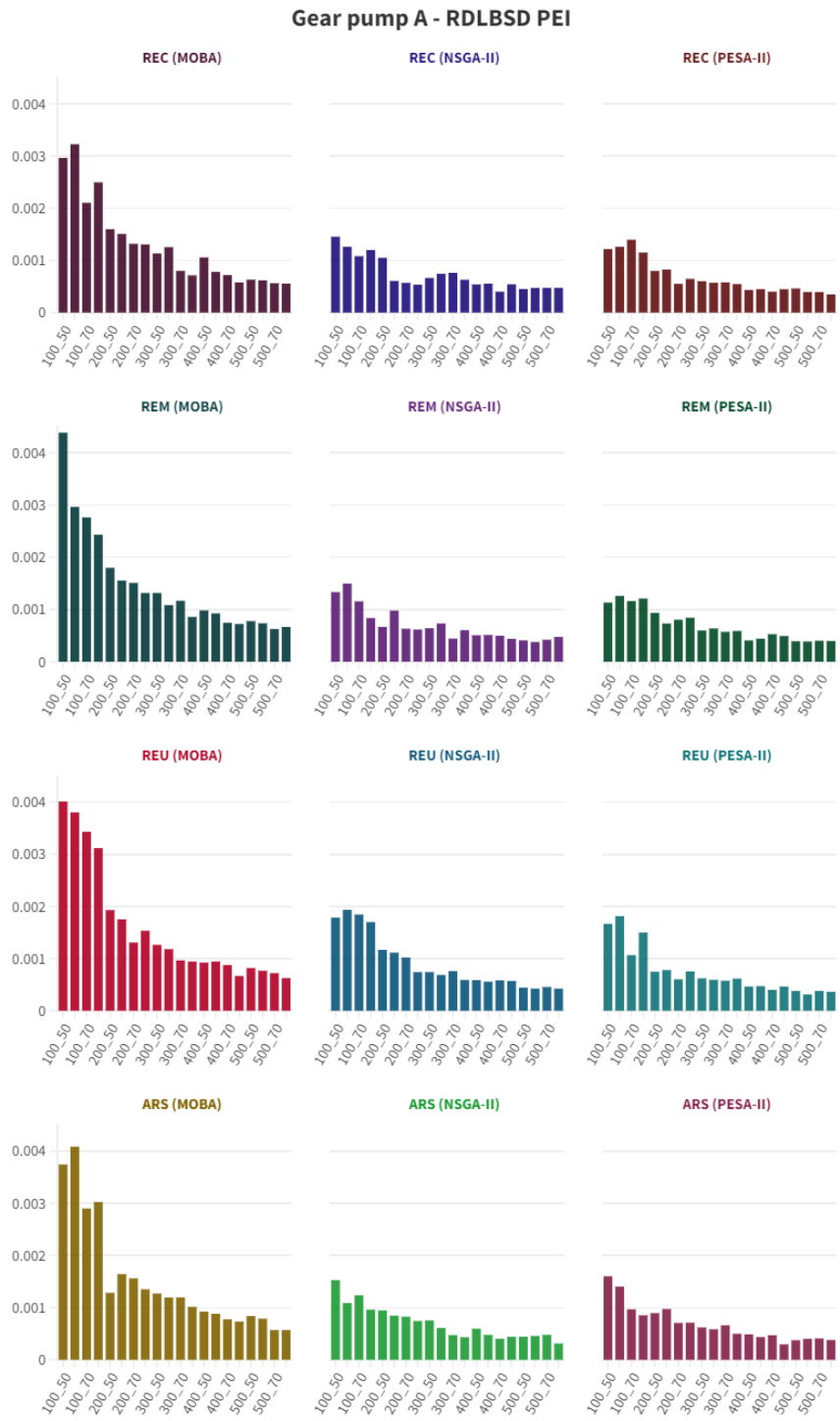


Figure 4.11: PEI for RDLBSD of Gear Pump A: The higher the better

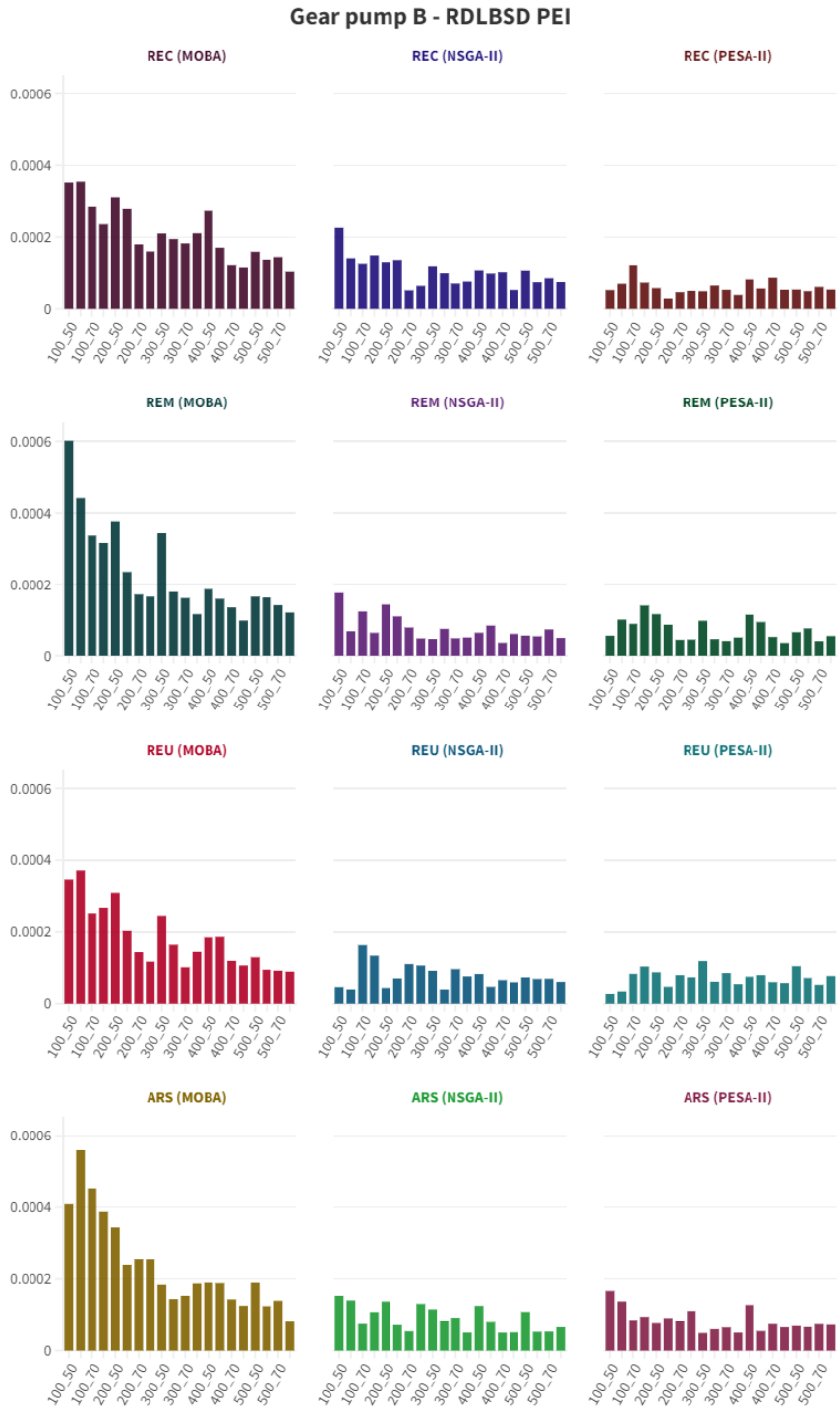


Figure 4.12: PEI for RDLBSD of Gear Pump B: The higher the better

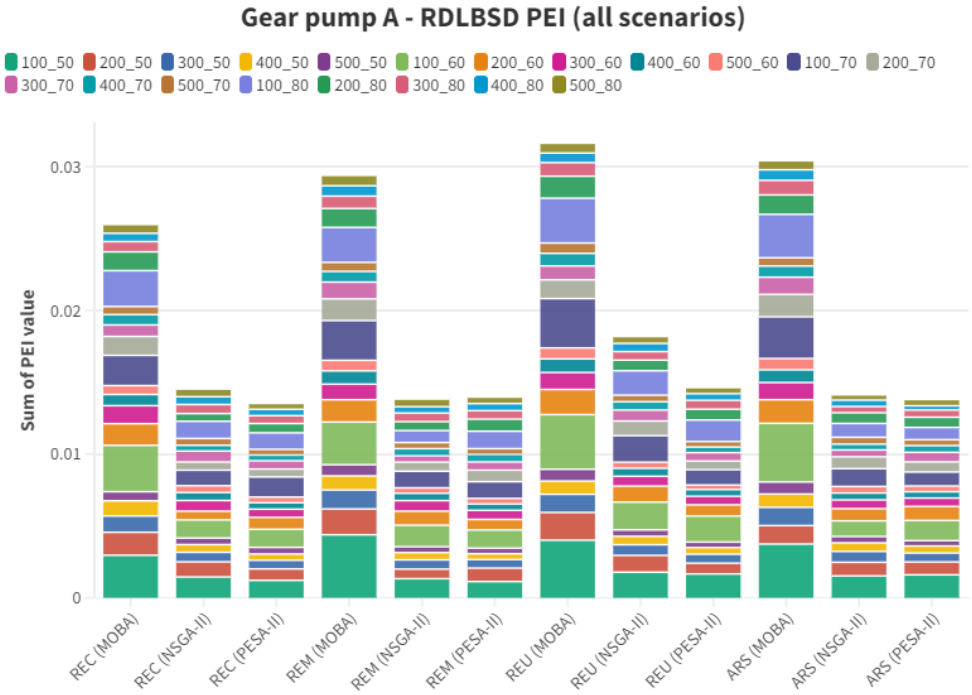


Figure 4.13: Total PEI for RDLBSD of Gear Pump A: The higher the better

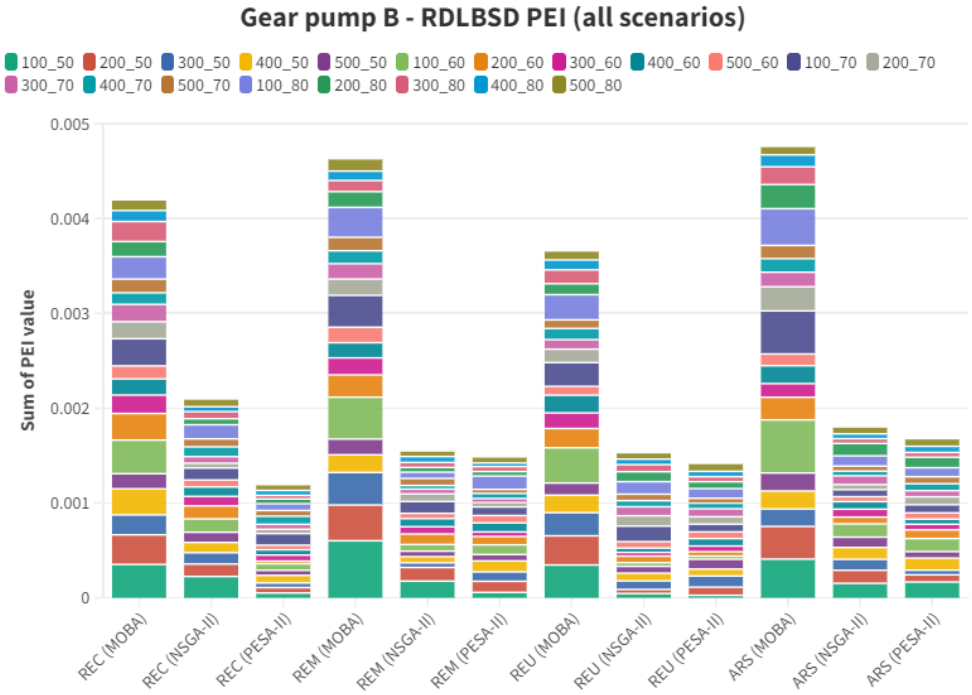


Figure 4.14: Total PEI for RDLBSD of Gear Pump B: The higher the better

4.3 Discussion

The results obtained from the MO-ND approach consist of sets of solutions that represent the various trade-offs between the four defined goals in this thesis: maximising profit, maximising energy savings, maximising reduction in environmental impact, and minimising unbalanced lines. These objectives, being conflicting and distinct, as shown in the Chapter 3, necessitate the use of the MO-ND approach to identify solutions that are not dominated by others and provide a range of feasible options for decision-making. Figures 4.3 and 4.4 visually depict the trade-offs between the different objectives, showcasing the diverse sets of POSs obtained from the MOBA, NSGA-II, and PESA-II in the ARS scenario. These solutions were achieved by applying these algorithms in four dimensions using the smallest parameter setting. By examining the figures, it becomes evident that NSGA-II and PESA-II exhibit similarities in their results, while MOBA yield different outcomes, for both gear pumps.

Visualising outcomes with four dimensions can be challenging, but it is evident that the number of POSs from MOBA is higher. This consistent performance aligns with the findings of the previous chapter, where MOBA demonstrated similar advantages in solving the robotic disassembly problem. While visual presentations in four dimensions may be more difficult to interpret, the next step uses performance measurement, which is valuable for evaluating the algorithms' performance in achieving a balance between profitability, energy efficiency, environmental impact, and line balance in RDLBSD.

Table 4.1 provide examples of the output from the RDLBSD for Gear Pump B. By examining the table, it becomes evident that the output bears similarity to the results obtained from the RDSP. This similarity arises from the fact that in RDLBSD, the optimal disassembly line balancing is achieved simultaneously with the feasible disassembly sequence. This synchronisation between line balancing and sequencing ensures an efficient and effective disassembly process, optimising

both the balance of the assembly line and the disassembly order. Similar to the previous chapter, the performance measurement of the MO-ND approach in this study relies on several indicators. These include the number of POSs, HI, NFE and PEI.

Consistently across all scenarios, the findings demonstrate that the MOBA outperform the comparison algorithms in various aspects. Specifically, the MOBA yield a higher number of POSs, indicating a more diverse and comprehensive range of optimal solutions, as shown in Figures 4.5 and 4.6. Additionally, the MOBA require a lower NFE to achieve these results, highlighting its efficiency in optimising the objectives, as shown in Figures 4.7 and 4.8. Furthermore, the higher HI further signifies their effectiveness in covering a larger area within the Pareto front, indicating a better spread of solutions, as shown in Figures 4.9 and 4.10. Finally, the higher PEI reinforces its suitability for finding optimum solutions for both the RDSP and RDLBSD problems. Figures 4.11 and 4.12 provide valuable insights into the PEI for different scenarios and parametric settings. Notably, the highest PEI varies across scenarios and case studies, suggesting that the optimal parameter settings may differ depending on the specific context. Interestingly, the results indicate that even the smallest parameter settings can yield optimal solutions for both case studies. This finding highlights the importance of carefully selecting parameter settings during the disassembly planning process. Future researchers working on similar case studies are advised to use parameter settings slightly higher than those identified in this study to achieve more effective and efficient results. Furthermore, as the case study involves an increasing number of parts, it is recommended to employ higher parameter settings. This adjustment accounts for the additional complexity introduced by a larger number of parts, ensuring that the disassembly planning process remains robust and capable of handling more intricate scenarios.

Figures 4.13 and 4.14 presents the PEI for all scenarios and algorithms across various parameter settings, providing a comprehensive overview of the results. The figure utilising 20 distinct colours to differentiate the parameter settings. Despite the multitude of colours, the figure provides a holistic view without sacrificing detailed information. By examining the highest total PEI (the

highest bar in the chart), it becomes apparent which algorithms are the top performers, offering a clear indication of their superior performance. The figure reveals that MOBA consistently outperform the comparison algorithms for both Gear Pumps A and B.

These consistent findings across the chapters further strengthen the position of MOBA as robust and effective approaches for solving complex robotic disassembly problems. Similar to the findings in Chapter 3, the analysis of the ARS scenario in terms of objective 1 (profit) reveals that it consistently yields the highest monetary value, with the majority of recovery options focusing on reusing the parts. This finding further reinforces the conclusion that all the algorithms employed in this study can identify optimal and sustainable recovery options for each part, regardless of the specific algorithm used. The difference lies in the performance of the algorithms used. The successful application of these algorithms to finding the best recovery options highlights the suitability and novelty of the proposed approach for further research. Furthermore, the results for the REU, REM, and REC scenarios are in line with the findings presented in Chapter 3. The REU scenario emerges as the second-best recovery option in terms of profit, followed by REM and REC. These consistent findings provide additional support for the literature review, which emphasises that recycling should be considered a last resort in sustainability practises and should only be employed when alternative recovery modes are not feasible.

4.4 Summary

In conclusion, this chapter addresses the research problem of RDLBSD and aims to enhance the efficiency and effectiveness of the disassembly process. By considering the interconnection between disassembly sequence planning and line balancing, a holistic approach is adopted to optimise both aspects simultaneously. The challenges of finding a feasible disassembly sequence and achieving line balance are tackled through the utilisation of metaheuristic algorithms,

specifically the BA, which has proven to be effective in handling the complexity of the RDLBSD problem. The research contributes by developing a sustainability model tailored for RDLBSD and providing recovery scenarios for EoL products, determining optimal parameter settings and performance metrics for optimisation algorithms. The research findings demonstrate the superiority of the MOBA in generating diverse and optimal solutions, as well as its efficiency in covering a larger area within the Pareto front. The integration of line balancing and sequencing in RDLBSD leads to an efficient and effective disassembly process that balances profit, energy savings, environmental impact reduction, and line balance. The consistent findings in the four scenarios (REC, REM, REU, and ARS) further reinforce that recycling should be considered the last option for recovery. The similarity in outcomes between the RDSP and RDLBSD for these four scenarios strengthens the conclusion that recycling should be prioritised after exploring other recovery options.

The insights gained from the study emphasise the importance of selecting appropriate parameter settings for different disassembly planning scenarios, enabling future researchers to enhance their approaches and achieve improved optimisation and performance in robotic disassembly processes. Overall, the research contributes to the advancement of knowledge in the field of robotic disassembly line balancing and provides valuable insights for practitioners using sustainability model and optimisation of robotic disassembly plans and balancing processes. In addition to the aforementioned contributions, this thesis successfully achieves its objectives, which are outlined as follows: Objective 1 is accomplished through the introduction of a new sustainability model tailored specifically for the RDLBSD problem. The model addresses the challenges and requirements of RDLB, bridging the gap in previous research and providing a framework for sustainable disassembly practises. Objective 3 is achieved by focusing on the research problem of RDLBSD. Objective 4 is addressed by validating the effectiveness of the proposed approach through a case study involving gear pumps. The application of the proposed methodology to a real EoL products demonstrates its practicality and potential for optimising robotic disassembly line

balancing processes. Finally, objective 5 is met by determining the optimal parameter settings and performance metrics for the optimisation algorithms employed in the RDLBSD problem. Through rigorous evaluation and comparison, the thesis identifies the most suitable parameter settings for achieving optimal results in different disassembly planning scenarios. By addressing this problem and providing innovative solutions, the thesis contributes to the advancement of knowledge in the area of robotic disassembly line balancing. Overall, the thesis successfully meets its objectives and makes contribution to the RDLBSD research, laying the groundwork for further advancements in the optimisation of robotic disassembly processes. Furthermore, similar to the previous chapter on RDSP, advancements in technology and Artificial Intelligence (AI) offer significant opportunities for the development of the RDLB research area. However, unlike RDSP, which has seen recent publications utilising digital twins and deep learning techniques, there is a noticeable gap in the recent RDLB literature in terms of utilising these technologies and AI. This presents an important opportunity for further research and exploration in order to leverage the potential of digital twins and deep learning approaches in the context of RDLB.

Chapter 5

Enhanced Bees Algorithm for Robotic Disassembly Planning

This chapter provides an overview of the bees algorithm (BA) since its inception in 2005, along with information on its variants and applications. The objective of this chapter is to address objectives 2, 4, 5, and 6 outlined in this thesis. The first objective is to enhance the BA as an effective tool for solving the robotic disassembly problem adressing objective 6. To achieve this objective, the chapter explores the sources of inspiration derived from the remarkable life of bees, uncovering the key concepts that have significantly influenced the algorithm's development. Additionally, the chapter conducts an extensive investigation of other algorithms that share similar names with the inspirations, offering a comparative analysis and deeper insights into the distinct approaches employed in these algorithms. This analysis serves to distinguish the novelty of the proposed enhancement of the BA compared to its variants and other metaheuristic algorithms. Objectives 2 and 4 of this thesis is achieved through the conduct of a case study on gear pumps, as described in Chapter 3.1, in order to validate the effectiveness of the proposed enhancement of the BA in RDSP problem. Furthermore, objective 5 is addressed through the analysis of the outcomes and the evaluation of the algorithm's performance using the proposed performance evaluation.

The chapter is structured as follows: Section 5.1 provides an overview of the BA and its variants, highlighting its application in the context of the robotic disassembly problem. Section 5.2 focuses on the proposed enhancement of the BA, drawing inspiration from the Fibonacci sequence to yield the Fibonacci bees algorithm, or BA_F. A review of other metaheuristics inspired by Fibonacci is provided to highlight the novelty of BA_F. Section 5.3 presents the experiments conducted using both the BA and the enhanced version in RDSP. Section 5.4 discusses the results obtained. Finally, the chapter concludes by summarising the key findings and insights obtained from the research and suggesting areas for further investigation.

5.1 Bees Algorithm

The BA, originally developed for continuous problems, has six user-determined parameters [160] and is commonly referred to as the basic bees algorithm (BBA). The pseudocode of the BBA [161] is given in Algorithm 5.

Algorithm 5: Basic Bees Algorithm Pseudocode

Input : n: number of scout bees, m: number of selected sites, e: number of elite sites, nsp: recruited bees for other selected sites, nep: recruited bees for elite sites, ngh: initial size of neighbourhood

1 **Function** `BBA`(n, m, e, nsp, nep, ngh) **:**
2 **InitialisePopulation** with random solutions
3 **EvaluateFitness** of the population
4 **while** *stopping criterion not met* **do**
5 Forming new population
6 **SelectSites** for neighbourhood search
7 **RecruitBees** for selected sites (more bees for the best e sites) and evaluate fitness
8 **SelectFittestBees** from each patch
9 **AssignRandomBees** to search randomly and evaluate their fitness
10 **end**
11 **return** *BestBee*

The neighbourhood (ngh) for combinatorial problems depends on the design of the local search, which may involve various operators like swap, insert, reverse, mutation, 2-OPT, and 3-OPT. In the combinatorial version of BBA [54, 185–187], where the swap, insert, and reverse operators are used, the neighbourhood size is considered equal to the sequence length. Therefore, for the combinatorial version, only five parameters need to be set: n, m, e, nsp, and nep.

The introduction of site abandonment and neighbourhood shrinking to BBA, resulting in two additional parameters that need to be configured [159, 188, 189], represents a noteworthy enhancement and gives rise to a variant of the algorithm known as the standard bees algorithm (SBA). SBA requires seven to eight parameters to be set by the user. Alternative methods for recruitment, neighbourhood modification, and site abandonment have been extensively explored in the literature [159]. For a comprehensive understanding of the variants of the BA developed

until 2017, interested readers are encouraged to refer to the survey paper by Hussein *et al.*, which provides an in-depth analysis of the various modifications and enhancements in the algorithm [190].

The utilisation of metaheuristics has been the subject of critique owing to their dependence on parameter values. This reliance can result in a lengthy procedure, as the optimal settings for each problem must be determined while considering their unique attributes. Previous researchers opted for the application of Taguchi and Design of Experiments (DoE) methods to discern the optimal parameter setting. However, these techniques are time-consuming [166]. Several efforts have been made to reduce the parameter tuning setting in the BA, such as the application of fuzzy selection for self-tuning [191]. The Ternary Bees Algorithm (TBA), introduced by [44], uses only 3 bees and incorporates a single parameter setting for the site abandonment thresholds. Another reduced-parameter version of BA is BA_2, which has two parameters. BA_2 is inspired by the traplining foraging behaviour of honeybees [192, 193]. The achievement of an adaptive parameter-free version and the potential for further parameter removal remain uncertain in the pursuit of enhanced versions of the BA [159]. However, it is apparent that reducing the number of parameters significantly decreases the effort required for parameter tuning.

In the field of robotic disassembly research, BA has been widely employed and proven to be superior to other algorithms [44, 47–49, 54, 79, 86, 88, 89, 126, 130]. However, only one robotic disassembly study [44] has been conducted on parameter reduction.

BA_2, initially developed for continuous problems and later extended to address combinatorial problems by incorporating specialised local operators tailored for the vehicle routing problem VRP, has been previously investigated by Ismail [192]. However, the primary focus of this thesis centres around robotic disassembly problems. RDSP entails planning the sequence and actions required to disassemble objects, considering factors such as interdependencies, sizes, shapes, connections, accessibility, tools, and multiple dimensions. These inherent complexities differentiate robotic disassembly from the VRP, which typically deals with fewer dimensions. Therefore, direct comparison with both the continuous version of BA_2 or its combinatorial version of VRP falls

outside the scope of this thesis. While earlier research by [192] drew inspiration from the traplining behaviour of bees to reduce the parameter count of the BA, this study takes a different approach by drawing inspiration from the Fibonacci sequence-based family tree pattern of honeybees. Building upon this unique source of inspiration, the proposed enhancement to the algorithm introduces a new configuration with four parameters. By exploring this alternative parameter setting, the research aims to investigate potential benefits and performance improvements. This novel approach contributes to the field by offering a fresh perspective on parameter configuration in the BA for robotic disassembly problems.

To ensure a meaningful and relevant evaluation, this chapter appropriately employs a comparison with Enhanced Discrete Bees Algorithm (EDBA), which is the most commonly adopted BA for RDSP. This choice enables an assessment of the effectiveness of the proposed enhancements in addressing the specific challenges posed by robotic disassembly problems. By aligning with the objectives and scope of this study, this approach facilitates a comprehensive examination of the proposed enhancements and their impact on addressing the complexities associated with robotic disassembly.

5.1.1 Bees Algoritm in RDSP and RLDB

As observed in the preceding chapter, it becomes evident that the predominant approach to address the RDSP involves framing it as a single-objective (SO) problem. This is also reflected in the use of the BA for solving the RDSP, where an SO approach is commonly employed [1, 2, 44, 48, 54, 79, 86, 88, 89, 126]. However, it is worth noting that this thesis is offering a novel perspective to use SO, MO aggregate and multiobjective nondominated (MO-ND) approach and analyse the results for solving the RDSP, that has been elaborated in Chapters 3 and 4. Furthermore, in contrast to RDSP, RDLB problem is predominantly formulated and approached as a multiobjective (MO) problem. Based on the literature review, the MO approach of the BA is

utilised in addressing the RDLB problem. In the studies conducted by [47, 49], the MO aggregate approach was employed, wherein the objectives were aggregated and treated as an SO formulation. Furthermore, the MO-ND approach was adopted by [51] and this thesis, specifically considering the objectives as non-dominated. None of this previous research has reduced the BA parameters in RDLBSD and only one research using reduced-parameter version of BA in RDSP. A notable research gap exists in the specific focus on reducing parameters of BA in robotic disassembly. This study aims to address this gap by investigating parameter reduction tailored specifically for robotic disassembly. By providing a dedicated examination of parameter reduction within the context of robotic disassembly, this research extends the existing literature and contributes to the advancement of knowledge in the research area.

As previously mentioned, the BA was initially developed for the continuous domain. However, when the algorithm is applied to combinatorial problems such as the Travelling Salesman Problem (TSP), the local search component needs to be modified accordingly. In the case of the TSP, the BBA incorporates local search operators such as swap, insert, and reverse [186], which are specifically designed to manipulate the sequence of cities. Similarly, in the context of the RDSP and RDLB problems, the BBA's local search is adapted to include swap, insert, and mutation operators [47, 49, 51, 54] that are tailored to address the specific requirements and constraints of these combinatorial problems. In this research, the BA and the proposed BA utilise the swap, insert, and mutation operators in their local search, as desribed by [54].

5.2 Fibonacci Bees Algorithm

Leonardo Fibonacci, a mathematician from the 13th century, is widely recognised for his significant contributions to the field of mathematics. One of his most notable achievements was the introduction of the Fibonacci sequence [194]. This sequence is a series of numbers, where each

number is the sum of the two preceding numbers. It begins with 0 and 1, and the subsequent numbers are generated by adding the previous two numbers. The sequence progresses as follows: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, and subsequent numbers. The pattern extends indefinitely, revealing a sequence that possesses unique mathematical properties and occurs in diverse natural phenomena. The pattern can be found in a variety of natural phenomena, including leaf arrangements, spirals in seashells, tree branches and flower petals [194, 195]. The bee lineage parallels the Fibonacci sequence, establishing a link between bee ancestry and the numerical pattern seen in the Fibonacci sequence. The family tree of a male bee or drone exhibits a fascinating relationship with the Fibonacci sequence, as the number of ancestors in each successive generation corresponds to the Fibonacci numbers [194, 195]. Drones originate from unfertilised eggs through a reproductive process known as parthenogenesis. Parthenogenesis involves the development of embryos without fertilisation, resulting in male offspring that possess a mother but do not have a father. In contrast, female bees, which include both worker bees and queen bees, originate from fertilised eggs. Figure 5.1 illustrates the family tree of a drone in a simple ancestry model [194, 196]. Interestingly, upon counting the total number of bees in each generation, it is revealed that they conform to the Fibonacci sequence. This correlation emphasises the prevalence of Fibonacci patterns in honeybee population and the drone's mathematical structure in its genealogical lineage.

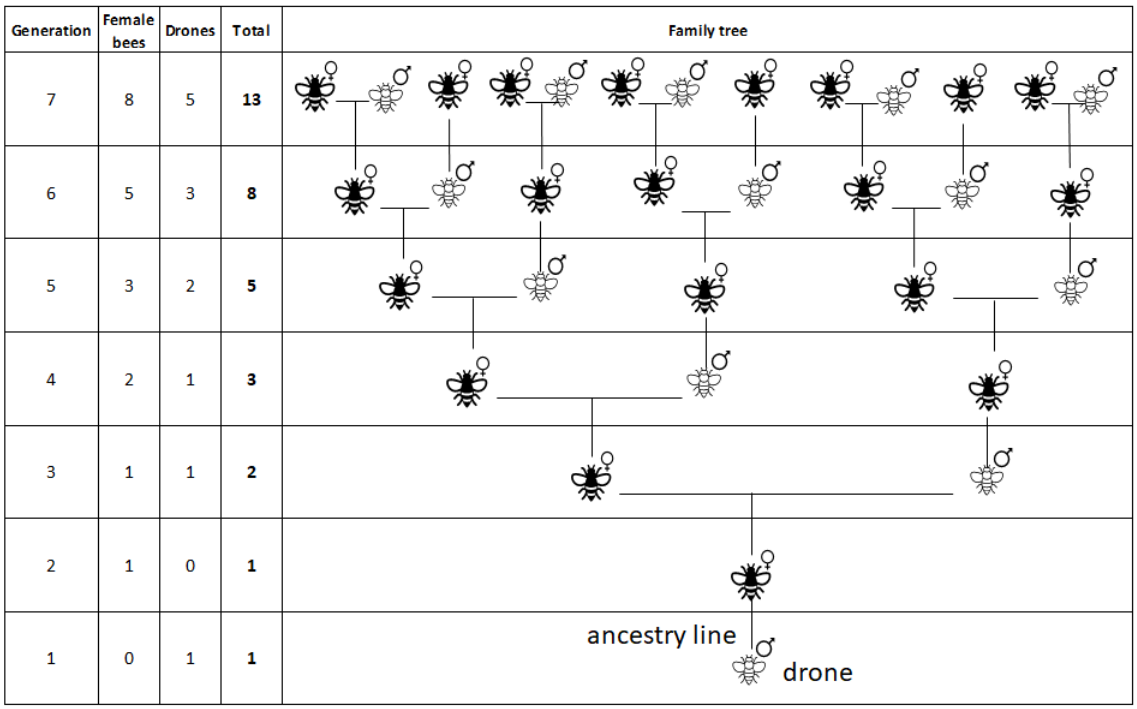


Figure 5.1: Fibonacci sequence in the family tree of a drone (adapted from [196])

To investigate whether previous enhancements of the BA have incorporated the inspiration from the Fibonacci sequence, given it relevance to bees, the following steps were conducted. An extensive search was conducted using the Scopus database, employing the keywords ”bee” and ”Fibonacci.” This search yielded a total of 9 articles; however, none of them were found to be directly relevant to the BA. Additionally, a search was performed using the Scite database, but no pertinent results were found regarding the utilisation of the Fibonacci sequence in the BA. These searches across multiple databases indicate that, as of the present, there is no scholarly evidence to suggest that prior versions of the BA have integrated the Fibonacci sequence as a means of enhancement.

Further exploration was conducted using the Scopus database, employing the keywords ”Fibonacci” and ”heuristic,” with the aim of identifying algorithms that draw inspiration from Fibonacci. The initial search yielded a total of 62 articles. After refining the search based on titles and language criteria (English only), and availability of content, 43 articles were retained for further

analysis. Upon examining the contents, 9 articles were deemed irrelevant and were excluded from the analysis. Additionally, 18 articles that were identified as improvements to the original algorithm were also removed. Of the remaining articles, 15 focused on metaheuristic algorithms, many of which were improved or hybridised with other algorithms such as the genetic algorithm (GA) [197–199], Salp Swarm Algorithm (SSA) [200], Wolf Pack Algorithm (WPA) [201], Tabu Search (TS) [202], Particle Swarm Optimisation (PSO) [203], and Grey Wolf Optimiser (GWO) [203]. These articles covered a range of inspirations related to Fibonacci, including the Fibonacci search, Fibonacci heaps, Fibonacci trees, the golden ratio, and the application of Fibonacci indicators in the stock market.

Table 5.1 summarises the Fibonacci inspirations of these algorithms and their applications. The Automated Hybrid Genetic Algorithm (AHGA) is utilised for benchmark function optimisation, wherein a combination of a GA and a Local Optima Avoidance (LOA) mechanism is employed [197]. Fibonacci numbers are used to select the number of GA banks in AHGA. The Memetic Algorithm (MA) is utilised for solving the multi-stage supply chain network problem, with the option of Fibonacci number generation to select the best offspring regardless of its origin [198]. The Fibonacci Tree Optimisation (FTO) algorithm is designed specifically for power grid optimisation, utilising a data storage structure known as a Fibonacci tree [204]. The Fibonacci Indicator Algorithm (FIA) finds its application in benchmark function optimisation, incorporating Fibonacci retracement and time zone techniques commonly utilised in stock market trading [205]. For wellbore trajectory optimisation in the oil and gas industry, the Fibonacci Sequence-based Quantum Genetic Algorithm (FSQGA) is employed, reflecting the rotation angle step in the quantum GA [199]. The Golden Ratio Optimisation Method (GROM) is applied to benchmark function optimisation, with the movement direction of the algorithm following the formulation of the golden ratio [206]. Similarly, the Improved Wolf Pack Algorithm (IWPA) is employed in benchmark function optimisation, where the step length of scout wolves is adjusted based on the Fibonacci sequence [201]. The Child Drawing Development Optimisation (CDDO) algorithm

uses the golden ratio to calculate two solution points in benchmark function optimisation [207]. The Opposition-based Learning PSO GWO (Opp-PSOGWO) algorithm is applied to benchmark function optimisation, generating an opposite population for searching using the Fibonacci sequence [203]. In the area of bridge structural health monitoring, a Fibonacci Sequence-based Optimisation Algorithm (FSOA) rearranges the population of the algorithm using the golden ratio [200]. Additionally, the table includes algorithms employed for solving the collection centre location problem inspired by the Fibonacci search [202], project scheduling problem using the Multi-Objective Fibonacci Based Algorithm (MOFA) inspired by the Fibonacci sequence [208], and VRP using Broad Local Search Algorithm (BLSA), Adaptive Variable Neighbourhood Search (AVNS), and Evolutionary Local Search (ELS) that inspired by Fibonacci heaps [209–211].

Table 5.1: Metaheuristics inspired by Fibonacci

Author(s)	Year	Name	Fibonacci	Application	Hybrid
Gudla and Ganguli [197]	2005	AHGA	sequence	Benchmark function	GA
Yeh [198]	2006	MA	sequence	Multistage supply chain network	GA, greedy
Aras and Aksen [202]	2008	not named	search	Collection Centre Location problem	Tabu search
Zachariadis and Kiranoudis [209]	2010	BLSA	heaps	VRP	
Wei *et al.* [210]	2014	AVNS	heaps	VRP	
Zhang *et al.* [211]	2015	ELS	heaps	VRP	
Da *et al.* [204]	2018	FTO	tree	Power Grid	
Etminaniesfahani *et al.* [205]	2018	FIA	retracement and time zone	Benchmark function	
Sha and Pan [199]	2018	FSQGA	sequence	Oil and Gas (wellbore trajectory)	GA
Nematollahi *et al.* [206]	2020	GROM	goden ratio	Benchmark function	
Zhao *et al.* [201]	2020	IWPA	sequence	Benchmark function	WPA
Hosseinian and Baradaran [208]	2020	MOFA	sequence	Project Scheduling problem	
Abdulhameed and Rashid [207]	2022	CDDO	goden ratio	Benchmark function	
Khosla and Verma [203]	2022	Opp-PSOGWO	sequence	Benchmark function	PSO, GWO
Tran-Ngoc *et al.* [200]	2023	FSOA	golden ratio	Bridge Structural Health Monitoring	SSA

The findings presented in Table 5.1 provide additional support for the distinctiveness of the proposed Fibonacci bees algorithm (BA_F) compared to previous research. While previous algorithms utilised Fibonacci numbers in various ways, such as for selecting the number of banks, reflecting rotation angles, adjusting step lengths, or generating populations, the BA_F introduces a novel framework specifically inspired by the Fibonacci sequence. In the process of improving BA, various ideas were experimented with and tested to identify improvements. The chosen and most effective idea is outlined below.

BA_F is based on the observed Fibonacci sequence-based family tree pattern in drones, where the number of drone ancestors follows the Fibonacci sequence, as mentioned earlier. However, the BA_F introduces a different approach. Instead of employing the Fibonacci sequence to count ancestors, BA_F uses it to determine the number of bees sent to the flower patches. This decision is motivated by the inherent growth pattern represented by the Fibonacci sequence, which aligns with the objective of maximising the foraging efficiency of the algorithm. By employing a ranking-based approach that recruit bees according to the Fibonacci sequence for targeting the most promising patches,BA_F aims to exploit the patches more effectively, potentially yielding improved results. This departure from the conventional interpretation of the Fibonacci sequence within the context of drone ancestry showcases an innovative adaptation of the concept to address the specific requirements of the proposed algorithm.

In addition, unlike the original BA, which employs 'elite' and 'other selected' sites, BA_F focuses solely on the selected sites, eliminating the need to distinguish between 'elite' and 'other selected' categories in the best sites. Within these selected sites (sites with the best fitness values), bees conduct their search, and the allocation of bees to exploit the selected sites (nr) is determined by a ranking system that follows the Fibonacci sequence. The top-ranked bee receives the maximum number from the specified Fibonacci sequence, while lower-ranking bees are assigned decreasing numbers from the sequence. For example, with three patches, m = 3 and $maxnr$ = 8. The 1^{st} patch attracts 8 recruited bees ($nr1$ = $maxnr$ = 8), the 2^{nd} patch attracts 5 recruited bees ($nr2$ =

5), and the 3^{rd} patch attracts 3 recruited bees ($nr3 = 3$). Although it is not certain if the allocation of foragers in nature follows the Fibonacci sequence, this differential allocation does reflect the fact that more foragers are allocated to higher-quality flower patches [212, 213]. After a specified number of tries, if the fittest bee in a patch remains unchanged, nr for that patch becomes zero and a fresh set of patches is initialised.

Both BBA and EDBA share the same parameters and structure, with the only distinction being the incorporation of a check for disassembly sequence feasibility after random initialisation. By disregarding that minor point, comparing EDBA and BA_F becomes equivalent to comparing BBA and BA_F. This chapter will compare the proposed BA_F and EDBA in the context of RDSP. The problem is the minimisation of disassembly time. The comparison will utilise the statistical performance metric (SPM) and the performance evaluation index (PEI) presented in Chapter 3. The objective function and number of function evaluations (NFE) will be recorded for the comparison purposes. As outlined in Chapter 3, the input data consists of a disassembly information matrix based on MFSG introduced by [54]. This matrix guarantees that the disassembly procedure considers fasteners while following to precedence constraints. Thus, the matrix yields feasible disassembly sequences. The input in the pseudo-code is represented by the robotic disassembly information matrix, denoted as dis_m. The pseudo-code for EDBA in the RDSP has been provided in Algorithm 2 in Chapter 3, while the pseudo-code for BA_F is presented in Algorithm 6.

Algorithm 6: The pseudo-code of BA_F for RDSP

Input: n: number of scout bees, m: number of selected sites, nr: number of bees recruited for selected sites using ranking based recruitment, max_rv: maximum number of re-visits before the nr is set to zero, dis_m: robotic disassembly information matrix
Output: RDSP(sequence, direction, tool, f:cost)

Function BA_F (n, m, nr, max_rv) **:**
  Start
  initialRDSP $\leftarrow GlobalMFSG(dis\_m : sequence, direction)$ `// Generate initial population of feasible disassembly sequences`
  revisit = 0
  while *stopping criterion not met* **do**
    Evaluate population fitness
    $f \leftarrow$ FVALUE($initialRDSP$)
    Sort population according to f
    Select m sites for local search
    `// Generate local sites with waggle dance`
    for *Selected Site (1 to m)* **do**
      `// Assign best local bee`
      $BestLocalBee \leftarrow$ *the scout bee that found the site*
      `// Allocation of bees to the selected sites`
      $nsp \leftarrow$ *number from Fibonacci sequence*
      for *Recruited Bee*(1 *to* nr) **do**
        `// Do feasibility check`
        while *feasibility not met* **do**
          RecruitedBee $\leftarrow$ WaggleDance($dis\_m : sequence, direction$)
        end
        `// Mutate the disassembly direction`
        RecruitedBee $\leftarrow Mutation(dis\_m : direction)$
        Evaluate fitness of $RecruitedBee$
        if $RecruitedBee$ *is better than* $BestLocalBee$ **then**
          `// Update BestLocalBee`
          $BestLocalBee = RecruitedBee$
        end
        else
          revisit $\leftarrow$ revisit + 1
          if $revisit \geq max\_rv$ **then**
            `// Abandon site and randomly generate a new site`
            NewSite $\leftarrow$ MFSG($dis\_m : sequence, direction$)
          end
        end
      end
    end
    `// Assign remaining scout bees for global search`
    for *RemainingScoutBee* (1 *to* $(n - m)$) **do**
      $RemainingScoutBee \leftarrow$ GlobalMFSG($dis\_m : sequence, direction$)
    end
    Evaluate fitness of the new population
    Sort population according to f
    `// Store the best RDSP with minimum cost` f
    Best RDSP = Bee with minimum cost f
  end
  return *Best RDSP (Bee with minimum cost f)*

5.3 Experiments

The selected test problem for investigation is the RDSP problem with a single-objective (SO) approach. As previously explained, the EDBA proposed by Liu et al. [54] serves as the basic bees algorithm (BBA) employed to address the challenges of the robotic disassembly problem. The objective in this RDSP is to minimise the disassembly time, as expressed in Equation (5.1) [54].

$$Z = \sum_{i=0}^{N_p-1} t_b(x_i) + \sum_{i=0}^{N_p-2} t_z(x_i, x_{i+1}) + \sum_{i=0}^{N_p-2} t_t(x_i, x_{i+1}) + \sum_{i=0}^{N_p-2} m_t(x_i, x_{i+1}) \tag{5.1}$$

where

- Z is the total disassembly time
- N_p is the number of total parts
- $t_b(x_i)$ is the basic time for disassembling part x_i
- $t_z(x_i, x_{i+1})$ is the penalty time for disassembly direction changes between part x_i and x_{i+1}
- $t_t(x_i, x_{i+1})$ is the penalty time for disassembly tool changes between part x_i and x_{i+1}
- $m_t(x_i, x_{i+1})$ is the moving time between part x_i and x_{i+1}

5.3.1 Experimental Setup and Metrics

There are two commonly employed stopping criteria in optimisation algorithms: iterations or NFE. In this thesis, the stopping criteria for the algorithm are based on the number of iterations. The parameter settings used for the EDBA are based on the recommended settings from [54]. In their work, the authors applied population numbers of 10, 20, 30, 40, and 50 for Gear Pump A at iterations 100, 200, 300, and 400. Similarly, for Gear Pump B, the same population numbers

were employed between iterations 100 and 600. The values for the numbers of selected sites (m), elite sites (e), recruited bees for other selected sites (nsp), and recruited bees for elite sites (nep) were assigned as follows: 4, 1, 1, and 2, respectively. According to their findings, the best results were obtained for Gear Pump A at iteration 300 using a population of 20 and for Gear Pump B at iteration 500 using a population of 40 [54]. In this thesis, to ensure a fair comparison, the iteration and population sizes for all algorithms have been set to the same values.

To ensure a meaningful comparison with the best results obtained from EDBA, it was decided to use 100, 200, 300, 400, and 500 iterations for both gear pumps, along with population sizes of 21, 31, 41 and 51. It is important to note that in the context of the EDBA article, the term "populations" refers to the number of scout bees (n) . Therefore, when calculating the total populations using the EDBA parameter settings and applying Equation (5.2) [188], the corresponding values for EDBA are 21, 31, 41, and 51 for the population sizes.

$$BA_{pop} = (e * nep) + ((m - e) * nsp) + (n - m) \tag{5.2}$$

5.3.2 Experimental Parameter Setting

In this section, the determination of the experimental parameters employed in the study is presented. The objective was to find the optimal parameter settings for BA_F and assess its performance in comparison to EDBA. To steps taken to determine the optimal parameter settings for BA_F are as follows:

1. Experimental Design: Define the experimental parameters. The numbers of selected sites (m), the maximum number of recruited bees around the selected sites ($maxnr$), and the maximum number of re-visits (max_rv) are configured as depicted in Table 5.2.

2. Initial Runs: Perform 10 runs for each scenario, E1 to E100.

3. Top Results Identification: Identify the best results from the initial runs.

4. Iterations and Performance Validation: Evaluate the performance of the selected best results from the initial runs across 50 runs using 100 iterations.

5. Descriptive Statistics: Analyse the descriptive statistics of the results.

6. Best Result Selection: Identify and select the best result.

Table 5.2: Experimental design for BA_F

m	*maxnr*	*max_rv*				
		0	1	5	10	15
2	3	E1	E2	E3	E4	E5
	5	E6	E7	E8	E9	E10
	8	E11	E12	E13	E14	E15
	13	E16	E17	E18	E19	E20
	21	E21	E22	E23	E24	E25
3	3	E26	E27	E28	E29	E30
	5	E31	E32	E33	E34	E35
	8	E36	E37	E38	E39	E40
	13	E41	E42	E43	E44	E45
	21	E46	E47	E48	E49	E50
4	3	E51	E52	E53	E54	E55
	5	E56	E57	E58	E59	E60
	8	E61	E62	E63	E64	E65
	13	E66	E67	E68	E69	E70
	21	E71	E72	E73	E74	E75
5	3	E76	E77	E78	E79	E80
	5	E81	E82	E83	E84	E85
	8	E86	E87	E88	E89	E90
	13	E91	E92	E93	E94	E95
	21	E96	E97	E98	E99	E100

Since BA_F uses a ranking-based mechanism to give the number of bees sent to the selected sites, for comparison purposes, it is necessary to calculate the population, as shown in the Equation (5.3), to obtain comparable population sizes for BA_F and EDBA. Table 5.3 presents values calculated using m = 5 and various $maxnr$ with a target maximum population of 51 bees. The maximum value of nr is $nr1$.

$$BA_F population = (n - m) + \sum_{i=1}^{m}(nr_i) \tag{5.3}$$

Table 5.3: Example of BA_F for a population of 51

Experiment	*n*	*maxnr=nr1*	*nr2*	*nr3*	*nr4*	*nr5*	*BA_Fpopulation*
E76	48	3	2	1	1	1	51
E81	44	5	3	2	1	1	51
E86	37	8	5	3	2	1	51
E91	25	13	8	5	3	2	51
E96	6	21	13	8	5	3	51

The experiment was then conducted ten times on gear pump B for a maximum of 100 iterations each. Gear pump B was selected as it was more complex than gear pump A. The parameter settings and average results of BA_F are presented in Table 5.4. Using the best parameter settings from [54], the EDBA achieved a disassembly time of 150.113 with 5550 function evaluations.

Table 5.4: BA_F results of the initial runs (10 independent runs)

Experiments	*m*	*maxnr*	*n*	NFE	Disassembly time				
					max_rv=0	*max_rv=1*	*max_rv=5*	*max_rv=10*	*max_rv=15*
E1 to E5	2	3	48	5348	173.8317	171.5567	160.185	145.065	151.3192
E6 to E10	2	5	45	5345	164.3175	165.0625	151.9225	145.7392	151.4817
E11 to E15	2	8	40	5340	160.9383	154.3392	147.3933	148.6567	143.74
E16 to E20	2	13	32	5332	150.0267	151.0992	148.7533	142.4158	142.9308
E21 to E25	2	21	19	5319	148.09	146.1292	146.4492	144.4858	144.3008
E26 to E30	3	3	48	5448	172.5958	168.9592	155.4217	146.0192	155.5033
E31 to E35	3	5	44	5444	160.6658	167.5758	149.49	143.3275	146.0117
E36 to E40	3	8	38	5438	161.3008	156.2958	146.8142	146.0458	144.9183
E41 to E45	3	13	28	5428	153.625	148.8067	143.9108	144.1133	143.245
E46 to E50	3	21	12	5412	151.1475	145.6175	**141.955**	**142.8275**	**143.6292**
E51 to E55	4	3	48	5548	173.1675	173.7192	153.4317	149.655	147.2775
E56 to E60	4	5	44	5544	165.3075	169.1067	148.6542	145.0658	145.8825
E61 to E65	4	8	37	5537	152.553	160.473	146.28	145.689	144.548
E66 to E70	4	13	26	5526	149.4717	150.3658	**143.9317**	**142.383**	**142.0917**
E71 to E75	4	21	8	5508	148.7025	147.88	144.2342	143.1633	143.0742
E76 to E80	5	3	48	5648	173.423	177.248	154.938	144.268	147.38
E81 to E85	5	5	44	5644	162.987	165.988	149.543	144.179	148.528
E86 to E90	5	8	37	5637	156.544	157.107	144.883	143.512	143.217
E91 to E95	5	13	25	5625	149.994	152.092	144.622	143.033	141.015
E96 to E100	5	21	6	5606	151.319	150.675	143.292	142.158	147.656

The preliminary findings indicate that all scenarios exhibit fewer NFEs compared to the benchmark algorithm, EDBA. Furthermore, the majority of results across all scenarios demonstrate lower average disassembly times compared to EDBA. These results suggest that BA_F offers higher accuracy and efficiency. However, further experiments were necessary before definitive conclusions could be drawn. Based on the average results and NFE, experiments E48, E49, E50, E68, E69, and E70 were selected for further investigations through 50 independent runs. The statistical results are shown in Table 5.5.

Table 5.5: Descriptive Statistics of E48, E49, E50, E68, E69, and E70 (50 runs)

Descriptive Statistics

	N	Minimum	Maximum	Mean		Std. Deviation
	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic
E48	50	135.32	154.15	142.3295	.54402	3.84683
E49	50	136.68	158.32	142.1598	.61608	4.35638
E50	50	135.32	154.04	142.1537	.54545	3.85691
E68	50	137.25	151.38	143.0255	.51073	3.61138
E69	50	135.32	148.13	141.6850	.39180	2.77047
E70	50	135.32	153.49	141.6220	.52304	3.69846
Valid N (listwise)	50					

Table 5.5 shows that scenario E69 yielded a lower standard error of the mean (SEM). This lower SEM signifies a more accurate estimate of the population mean. Moreover, the standard deviation is also smaller, indicating reduced data variability. This characteristic is desirable regardless of whether the objective is to minimise or maximise a certain parameter. The preference is for results to cluster closely around the means, as it signifies greater precision in achieving the intended outcome. Therefore, scenario E69 with m = 4, $maxnr$ = 13, n = 26 emerges as the optimal choice. Fifty independent runs were subsequently conducted with the above parameter settings to compare BA_F and EDBA. The best parameter settings for BA_F specified a minimum population size of 31. Consequently, the comparative assessment with a population size 21 was eliminated.

5.3.3 Experimental Results

The experiments were run 50 times with stopping criteria set at iterations 100, 200, 300, 400, and 500, while the population sizes used were 31, 41 and 51. For ease of reference, a grouping code was employed, where "100_31" indicates the experiment with a stopping criterion of 100 and a population size of 31. This coding scheme helps to conveniently identify the specific parameter settings used in each experiment. The best disassembly results using the EDBA and BA_F algorithms are presented in Tables 5.6 and 5.7 for Gear Pump A and Gear Pump B, respectively.These tables display the sequence, direction, tool, and total time of disassembly. The best result aligns with the best-known values for the case study, as reported in [54], who conducted experiments using 1000 runs and obtained values of 87.5731 and 135.3167 for gear pumps A and B, respectively. Tables 5.8 and 5.9 display the average values of the disassembly time, SEM, average NFE, and delta. Delta is the percentage difference between the average objective value (disassembly time) to the best-known value. The boxplots for gear pumps A and B are presented in Figures 5.2 and 5.3, respectively.

Table 5.6: Gear pump A best results (EDBA and BA_F)

Best Disassembly Output	
Sequence	2-1-6-5-4-3-7-10-9-11-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	2-1-6-5-4-3-7-10-11-9-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	3-4-5-6-1-2-7-10-11-9-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	3-4-5-6-1-2-7-10-9-11-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	5-4-3-2-1-6-7-10-11-9-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	5-4-3-2-1-6-7-10-9-11-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	6-1-2-3-4-5-7-10-11-9-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731
Sequence	6-1-2-3-4-5-7-10-9-11-8-13-15-14-12
Direction	2-2-2-2-2-2-2-2-2-2-2-2-1-1-1
Tool	1-1-1-1-1-1-4-3-3-3-3-3-2-3-4
Time	89.5731

Note:
Direction: 1 = Y+ direction, 2 = Y- direction
Tool: 1=Spanner-I, 2 = Spanner-II, 3 = Gripper-I, 4 = Gripper-II

Table 5.7: Gear pump B best results (EDBA and BA_F)

Best Disassembly Output	
Sequence	24-22-20-23-21-19-1-2-3-4-5-6-7-13-10-9-8-12-14-15-16-17-18-11
Direction	1-1-1-1-1-1-2-2-2-2-2-2-2-2-2-2-2-2-2-2-2-2-1-1
Tool	3-3-2-3-3-2-1-1-1-1-1-1-5-4-4-4-4-5-4-4-4-4-4-4
Time	135.3167
Sequence	24-22-20-23-21-19-1-6-5-4-3-2-7-13-10-9-8-12-14-15-16-17-18-11
Direction	1-1-1-1-1-1-2-2-2-2-2-2-2-2-2-2-2-2-2-2-2-2-1-1
Tool	3-3-2-3-3-2-1-1-1-1-1-1-5-4-4-4-4-5-4-4-4-4-4-4
Time	135.3167
Note:	
Direction: 1 = Y+ direction, 2 = Y- direction	
Tool: 1 = Spanner-I, 2 = Spanner-II, 3 = Spanner-III, 4 = Gripper-I, 5 = Gripper-II	

Table 5.8: EDBA and BA_F average results for gear pump A

Iteration_population	EDBA				BA_F			
	Mean	SEM	Delta	Ave_NFE	Mean	SEM	Delta	Ave_NFE
100_31	90.5237	0.33	0.95%	3530	90.0960	0.213	0.52%	3506
200_31	90.0357	0.27	0.46%	7030	89.7208	0.064	0.15%	7006
300_31	89.8065	0.23	0.23%	10530	**89.5731**	**0.000**	**0.00%**	10506
400_31	89.6047	0.03	0.03%	14030	**89.5731**	**0.000**	**0.00%**	14006
500_31	89.7387	0.14	0.17%	17530	**89.5731**	**0.000**	**0.00%**	17506
100_41	90.6014	0.32	1.03%	4540	89.7404	0.137	0.17%	4516
200_41	89.8637	0.21	0.29%	9040	**89.5731**	**0.000**	**0.00%**	9016
300_41	89.6364	0.04	0.06%	13540	**89.5731**	**0.000**	**0.00%**	13516
400_41	89.5747	0.00	0.00%	18040	**89.5731**	**0.000**	**0.00%**	18016
500_41	**89.5731**	**0.00**	**0.00%**	22540	**89.5731**	**0.000**	**0.00%**	22516
100_51	89.7875	0.08	0.21%	5550	**89.5731**	**0.000**	**0.00%**	5526
200_51	89.7066	0.08	0.13%	11050	**89.5731**	**0.000**	**0.00%**	11026
300_51	89.6047	0.03	0.03%	16550	**89.5731**	**0.000**	**0.00%**	16526
400_51	89.6047	0.03	0.03%	22050	**89.5731**	**0.000**	**0.00%**	22026
500_51	89.6047	0.03	0.03%	27550	**89.5731**	**0.000**	**0.00%**	27526

Table 5.9: EDBA and BA_F average results for gear pump B

Iteration_population	EDBA				BAF			
	Mean	SEM	Delta	Ave_NFE	Mean	SEM	Delta	Ave_NFE
100_31	150.9088	1.36	15.59%	3530	146.6653	1.08	11.35%	3506
200_31	146.918	0.90	11.60%	7030	140.9998	0.36	5.68%	7006
300_31	145.5225	0.91	10.21%	10530	140.5303	0.42	5.21%	10506
400_31	142.217	0.70	6.90%	14030	139.7332	0.35	4.42%	14006
500_31	142.0165	0.45	6.70%	17530	139.1772	0.27	3.86%	17506
100_41	148.5027	0.97	13.19%	4540	142.6655	0.63	7.35%	4516
200_41	143.5897	0.56	8.27%	9040	140.5633	0.34	5.25%	9016
300_41	142.4695	0.63	7.15%	13540	139.8480	0.19	4.53%	13516
400_41	142.0715	0.81	6.75%	18040	139.0933	0.23	3.78%	18016
500_41	141.9507	0.54	6.63%	22540	138.6493	0.21	3.33%	22516
100_51	148.5675	1.08	13.25%	5550	141.6850	0.39	6.37%	5526
200_51	143.7058	0.59	8.39%	11050	139.9833	0.29	4.67%	11026
300_51	142.9958	0.69	7.68%	16550	139.2005	0.15	3.88%	16526
400_51	142.0362	0.60	6.72%	22050	139.2242	0.15	3.91%	22026
500_51	142.0932	0.64	6.78%	27550	138.8473	0.14	3.53%	27526

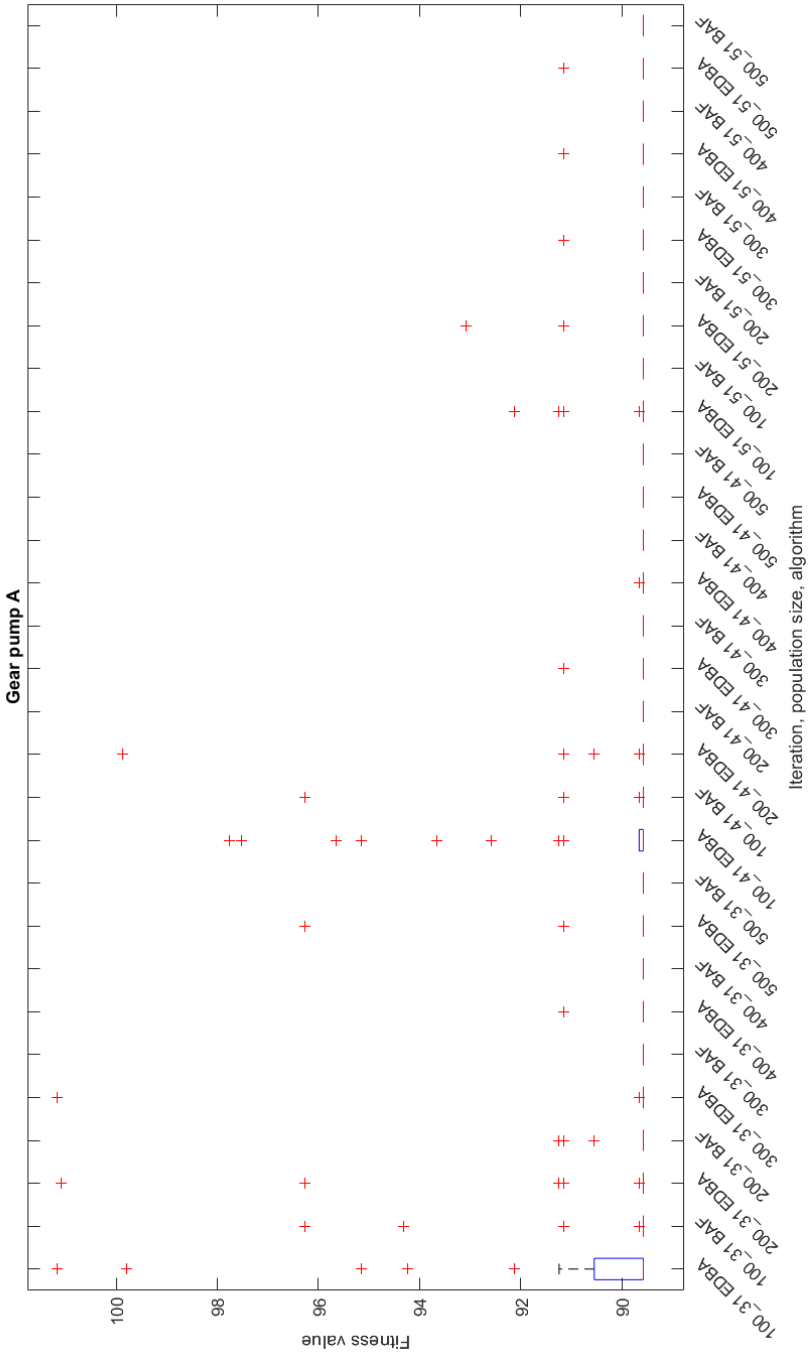


Figure 5.2: EDBA and BA_F results (Gear Pump A)

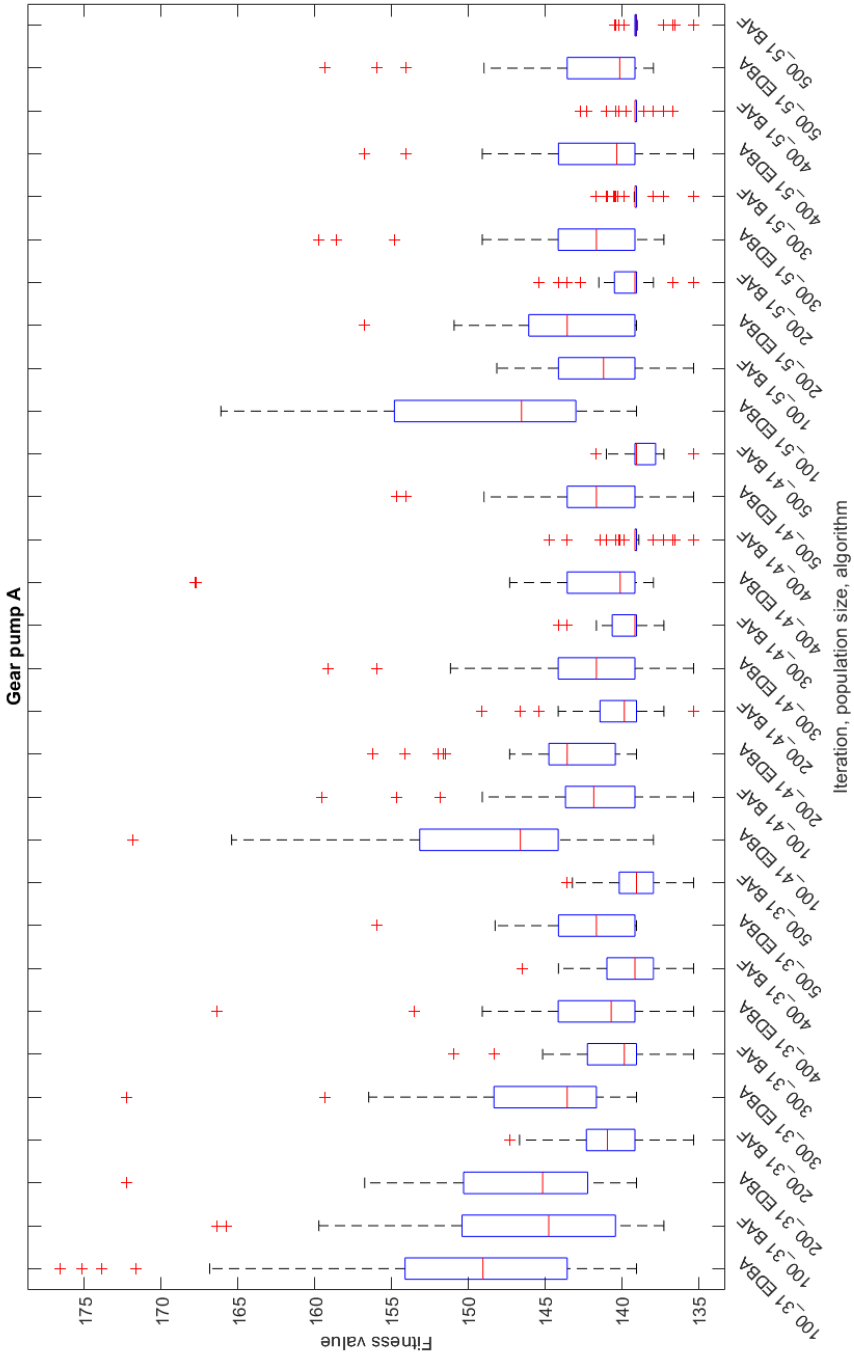


Figure 5.3: EDBA and BA_F results (Gear Pump B)

5.3.4 Statistical Performance Metric Results

The statistical assumptions, as discussed in Section 3.2.1, were checked prior to conducting the subsequent statistical test. From the boxplots, it is evident that the distribution of the data do not follow a normal distribution. The statistical tests for normality and homogeneity, as presented in the Appendix B, confirm that the results deviate from both normality and homogeneity assumptions. Therefore, since the data violate the statistical assumptions required for conducting parametric tests, nonparametric tests are employed. The significance of differences between the mean ranks of various iterations and population sizes on EDBA and BA_F results is examined using the Kruskal-Wallis ANOVA test, as depicted in Tables 5.10 and 5.11. The test results indicate the rejection of the null hypothesis ($p < 0.05$), providing strong evidence of a difference between at least one pair of iterations and population sizes. Consequently, a post hoc test, namely, the Dunn-Sidak test, is conducted. The results of the Dunn-Sidak test are presented in Figures 5.4 and 5.5.

Table 5.10: Kruskal-Wallis test results (Gear Pump A)

Kruskal-Wallis ANOVA Table

Source	SS	df	MS	Chi-sq	Prob>Chi-sq
Columns	8.18798e+06	29	282344	263.11	1.4359e-39
Error	3.84605e+07	1470	26163.6		
Total	4.66485e+07	1499			

Table 5.11: Kruskal-Wallis test results (Gear Pump B)

Kruskal-Wallis ANOVA Table

Source	SS	df	MS	Chi-sq	Prob>Chi-sq
Columns	9.89515e+07	29	3412119.9	533.75	3.24619e-94
Error	1.78946e+08	1470	121731.9		
Total	2.77897e+08	1499			

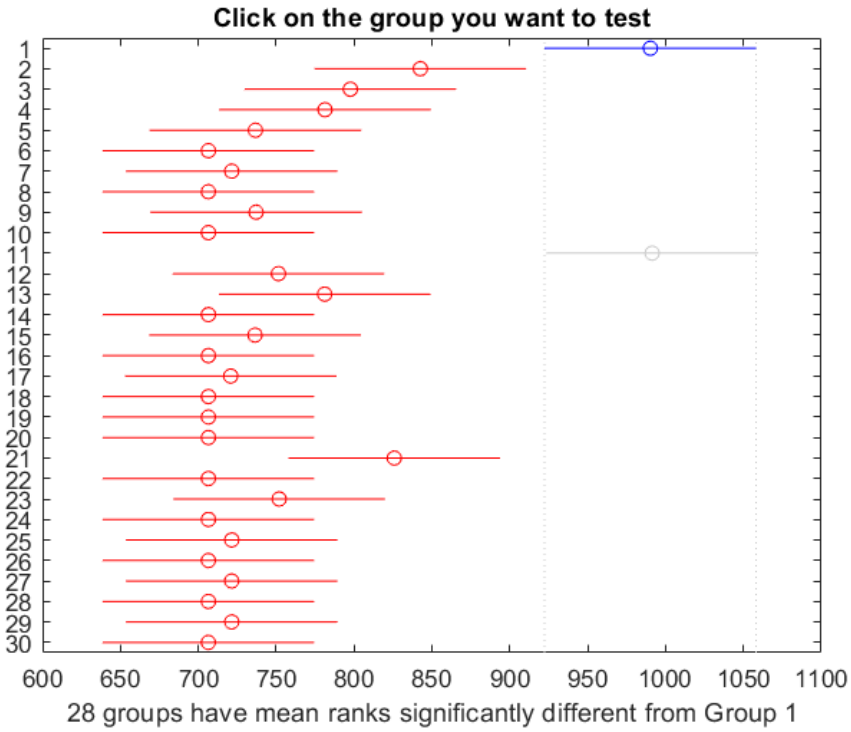


Figure 5.4: Dunn-Sidak test results (Gear Pump A)
Notes: Group 1 =100_31 EDBA, Group 2 = 100_31 BA_F, Group 3 = 200_31 EDBA, Group 4 = 200_31 BA_F,..., Group 29 = 500_51 EDBA, Group 30 = 500_51 BA_F

The Dunn-Sidak results offer detailed insights into the groups that display variation in performance based on iteration within each group size. Specifically, for Gear Pump A, the results show significant differences of groups 1 and 11 (100_31 EDBA and 100_EDBA) to the other groups. It is noteworthy that the BA_F yielded consistent outcomes across all parameters, as evidenced by the groups with an even-numbered labels. However, in the case of Gear Pump B, which as a larger dataset in comparison to Gear Pump A, both EDBA and BA_F displayed more pronounced disparities across all groups.

These findings emphasise the importance of selecting optimal parameter settings for disassembly time optimisation, taking into account both iteration size and population size. The observed variability in performance highlights the need for careful parameter selection to achieve optimal results.

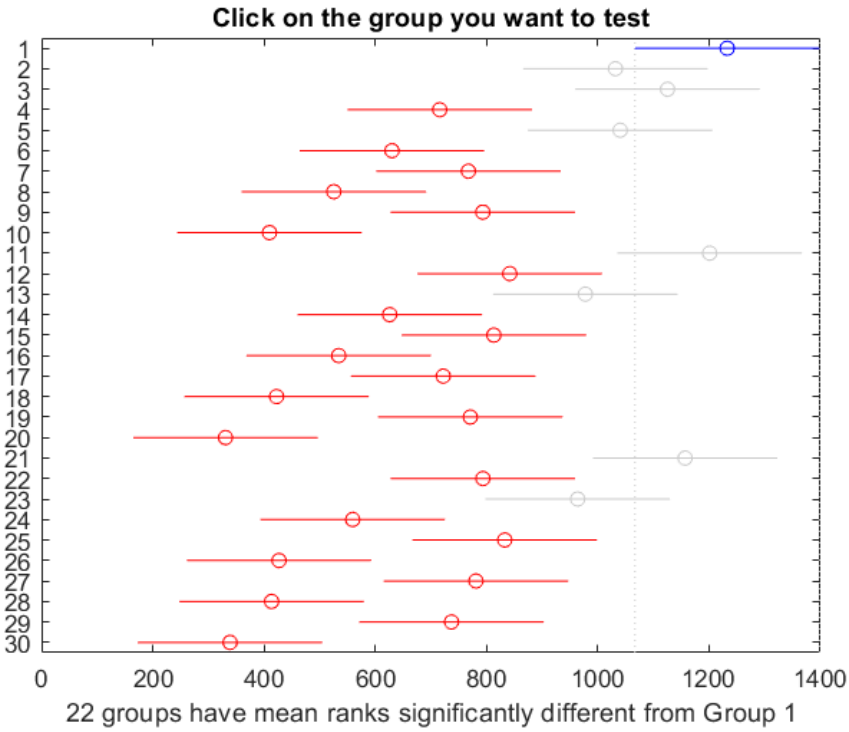


Figure 5.5: Dunn-Sidak test results (Gear Pump B)
Notes: Group 1 =100_31 EDBA, Group 2 = 100_31 BA_F, Group 3 = 200_31 EDBA, Group 4 = 200_31 BA_F,..., Group 29 = 500_51 EDBA, Group 30 = 500_51 BA_F

In the subsequent step, groups with significant differences from the previous statistical test are eliminated, and all iterations and population sizes are combined to focus on the groups that demonstrate similar performance with the lowest mean objective values. This step ensures that the final test results only include the groups that yield the best results. The outcomes of the final statistical tests can be found in Figure 5.6, Tables 5.12 and 5.13.

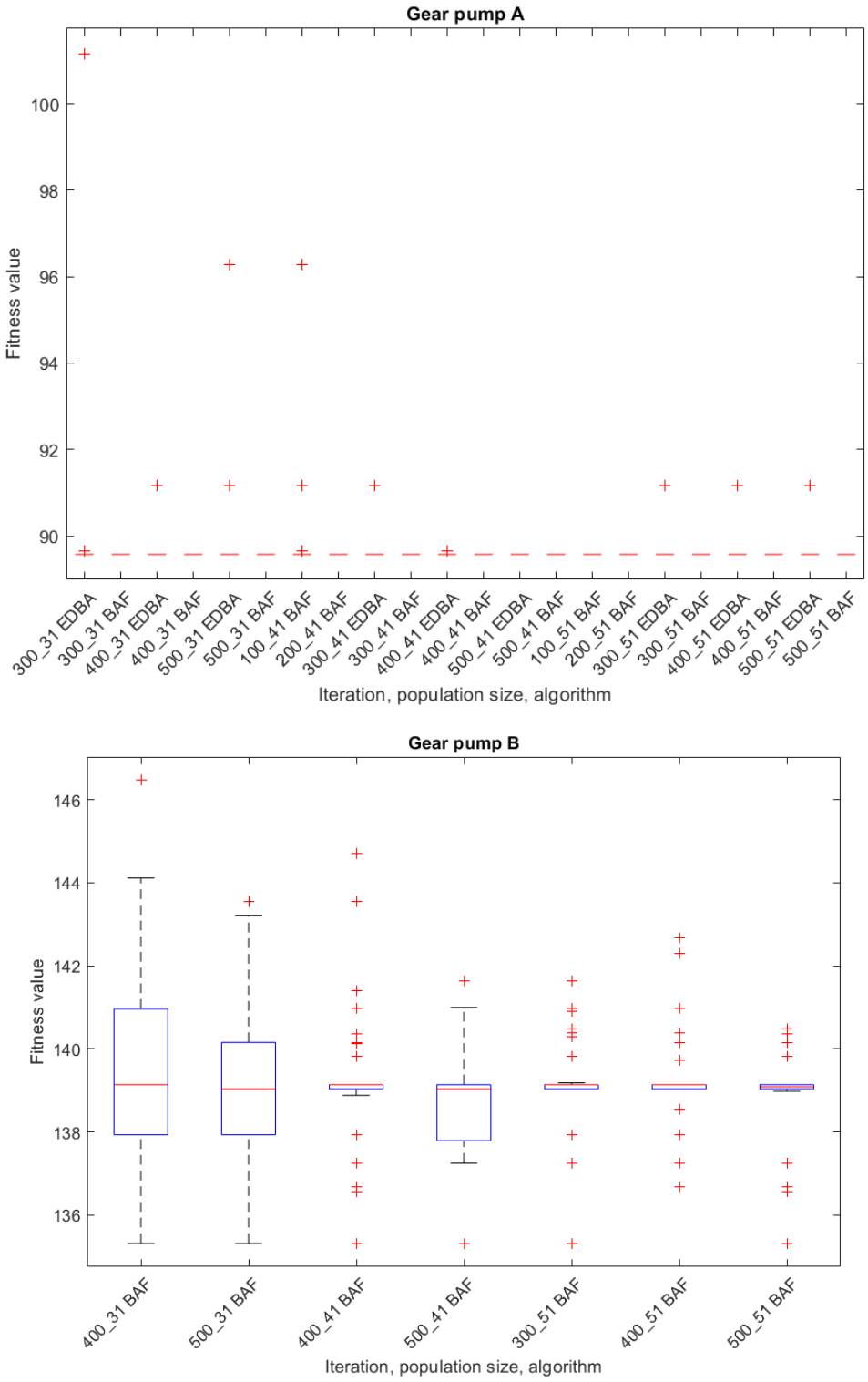


Figure 5.6: EDBA and BA_F boxplot final results

Table 5.12: Kruskal-Wallis final results (Gear Pump A)

Kruskal-Wallis ANOVA Table

Source	SS	df	MS	Chi-sq	Prob>Chi-sq
Columns	103511.1	21	4929.1	27.21	0.1641
Error	4077771.9	1078	3782.72		
Total	4181283	1099			

Table 5.13: Kruskal-Wallis final results (Gear Pump B)

Kruskal-Wallis ANOVA Table

Source	SS	df	MS	Chi-sq	Prob>Chi-sq
Columns	111867.9	6	18644.7	11.44	0.0756
Error	3299830.1	343	9620.5		
Total	3411698	349			

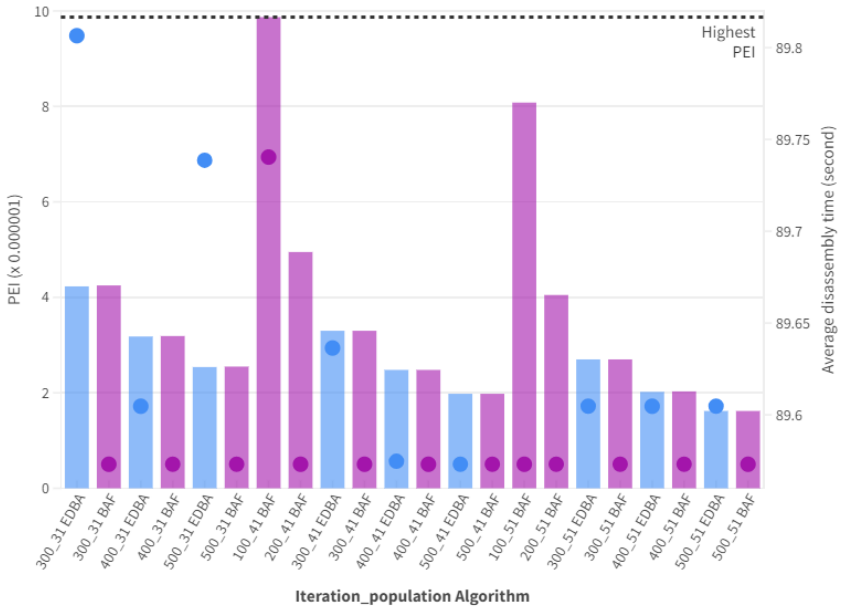
5.3.5 PEI results

As previously highlighted, PEI is designed to serve as a versatile index. In this experiment, the objective is to minimise the disassembly time while also aiming for the lowest value of the NFE. To achieve this, both metrics are placed in the denominator during the PEI calculation, as shown in Equation (5.4), with equal weight assigned to all the metrics. It is important to note that since none of the metrics are desirable to have the highest value, a value of 1 is assigned to the numerator.

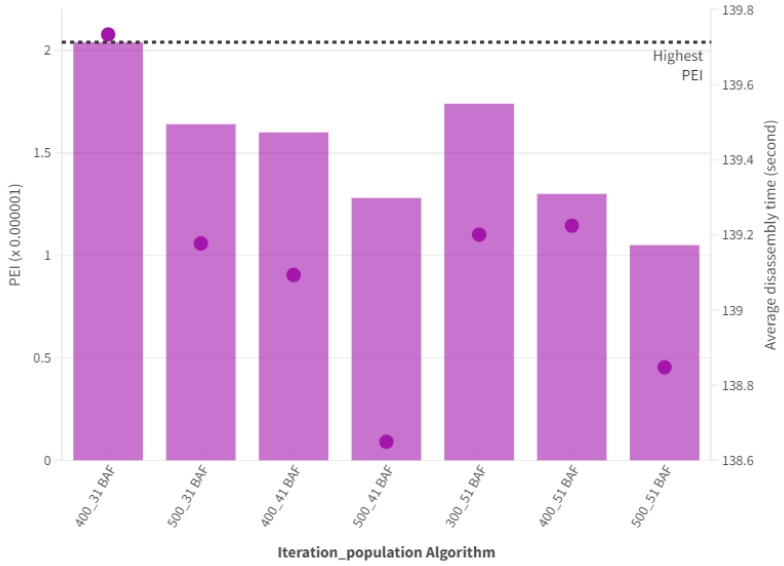
$$PEI = 1/[Obj^{\omega_1} NFE^{\omega_2}] \tag{5.4}$$

To ensure a meaningful comparison, only the groups that exhibited no significant differences in means were considered for the calculation. Figures 5.7a and 5.7b and Appendix B show the PEI calculation results, scaled to units of 10^{-6}. The figures provide a visual representation of the PEI values for the selected groups, allowing for an easy comparison of their performance. Note that, although this work employs both SPM and PEI, they can be used on their own as they are independent performance measures. SPM is more suitable for an SO problem, while PEI can be

utilised for both SO and MO optimisation.



(a) Gear Pump A



(b) Gear Pump B

Figure 5.7: PEI (histogram) and Average Disassembly Time (dot): Higher PEI and Lower Disassembly Time are Better

Note: The line shows the highest PEI; EDBA represented by blue and BA_F by purple colour.

5.4 Discussion

The experimentation process for determining the optimal parameters setting for BA_F was conducted systematically in six steps, including statistical tests for result analysis. This approach was implemented to ensure that the best parameters are obtained for the subsequent step of comparing with EDBA. The best parameters for the RDSP problem were found to be m=4, $maxnr$ = 13, and max_rv = 10. The study conducted by [54] used a population range of 21 to 51. The number of scout bees (n) in BA_F was chosen in the range 6 to 26 to yield a similar population size of 31 to 51. As previously mentioned, the population size 21 was eliminated from the final experiments. After determining the optimal parameters settings, 50 independent runs were conducted for both EDBA and BA_F.

The minimum disassembly time can be achieved using both the EDBA and BA_F, as presented in Tables 5.6 and 5.7 for both gear pumps. These tables give the sequence, direction, tools, and total time required to disassemble the gear pumps. However, the average results reveal differences between the EDBA and BA_F. The average results of EDBA and BA_F, presented in Tables 5.8 and 5.9, indicate that BA_F outperforms EDBA. Examination of the outcomes for Gear Pump A shows that BA_F consistently exhibits superior performance in comparison to EDBA across all 15 parameter configurations. BA_F achieves a perfect accuracy rate in 12 instances, as evidenced by a delta value of 0%, indicating that the accuracy is 100%. In contrast, EDBA only achieves 100% accuracy once with the same settings. The standard error of the mean (SEM) and delta exhibit a decreasing trend as larger parameter values are used. EDBA performed best in 500 iterations and with a population size of 41, giving an average NFE of 22,540. In contrast, BA_F demonstrated the best performance in almost all cases. The average NFE was 5526 was the lowest NFE required to achieve optimal outcomes in 100 iterations and with a population size of 51. Similarly, on Gear Pump B, it is evident that BA_F outperforms EDBA for all population sizes and iterations. Figure

2 displays the boxplot results for Gear Pump A, indicating that BA_F achieves better results. The algorithm demonstrates the ability to find the minimum disassembly time with minimal data spread, particularly at higher iterations for all tested population sizes. In Figure 3, the boxplot results for Gear Pump B exhibit improved data visualisation, allowing for better observation of data spread. In this boxplot, it becomes even clearer that BA_F outperforms EDBA in terms of mean, minimum disassembly time, and data spread across all tested iterations and population sizes.

As mentioned earlier, research by [54] indicates that EDBA performs best on Gear Pump A in 300 iterations and with a population of 21 (NFE = 7520), while Gear Pump B achieves optimal performance in 500 iterations with a population of 41 (NFE = 22,540). However, it is important to emphasise that the conclusions drawn in this thesis are not solely based on statistically descriptive results. To ensure the reliability and validity of the findings, additional statistical tests are imperative to validate the parameter settings that yield the best performance. Relying solely on the statistically descriptive results may not provide the conclusive evidence required to draw robust conclusions. Since the experiments involved 50 independent runs, it is necessary to assess whether parametric statistical tests can be employed, as discussed in Section 3.2.1. The statistical assumptions for parametric tests include having a sample size of more than 30, data following a normal distribution, and data belonging to the same populations. However, the boxplot results indicate that the data do not follow a normal distribution, and thus normality and homogeneity tests were conducted. The results confirm that the data violate the assumptions of normality and homogeneity, necessitating the use of nonparametric tests despite the sample size exceeding 30. As previously mentioned, the Kruskal-Wallis test indicated a significant difference among the iterations within the same population size groups (see Tables 5.10 and 5.11. To further investigate these differences, the Dunn-Sidak post hoc test was employed to identify specific groups that exhibited statistically significant variations in mean values.

Figures 5.4 and 5.5 clearly demonstrate noticeable differences, particularly in the outcome obtained from EDBA. Subsequently, the groups with significant differences are removed, and

the statistical tests are repeated. The boxplot results, as shown in Figure 5.6, visually indicate that the data distributions in each group are similar. The Kruskal-Wallis test (see Tables 5.12 and 5.13) confirmed that there was no statistically significant difference among the groups ($p > 0.05$). Therefore, post hoc tests are not needed, as the Kruskal-Wallis test has already confirmed the similarity among the groups.

The final step involves the utilisation of the proposed index, as detailed in Section 3.2.2. Within this chapter, the index is calculated based on the disassembly time and NFE. The calculation for the index has been previously explained in the previous section, and the obtained results, depicted in Figure 5.7, exhibit the PEI values in conjunction with the average disassembly time. By examining Figure 5.7, the parameter settings that yield the highest PEI can be identified, while the average disassembly time has been graphically presented for analytical purposes. As mentioned previously, the PEI is designed to be interpreted such that a higher index indicates better performance. The PEI results reveal that the best PEI of 9.87 x 10^{-6} for Gear pumps A was obtained in 100 iterations and with a population size of 41 for BA_F while for EDBA the optimum PEI of 4.23 x 10^{-6} was found in 300 iterations and with a population size of 31. The results indicate that BA_F with smaller NFE can produce statistically similar outcomes to those with a higher NFE. Figure 5.7a displays the average disassembly time as a dot, indicating that BA_F consistently outperforms EDBA in all instances. Directly comparing similar parameter settings for the two algorithms reveals that BA_F consistently yields higher PEI values – a highly desirable outcome for both gear pumps. In term of average disassembly time, BA_F yields lower values, aligning with the desired objective of minimising the disassembly time. The results obtained from the PEI analysis of Gear Pump B (Figure 5.7b) clearly indicate that the BA_F outperforms the EDBA. The optimal results were achieved after 400 iterations and with a population size of 31, yielding a PEI value of 2.04 x 10^{-6}.

These findings highlight the robustness of the chosen parameter settings, which cannot be easily discerned based on statistical description alone. It is evident from the results that the average disassembly time exhibits a pattern of decreasing values with higher population size and iteration.

However, when considering the average disassembly time and the PEI together, it becomes apparent that the parameter settings yielding the lowest average disassembly time may not always be the optimal ones. This is because the NFE also plays a crucial role, as a higher NFE indicates that the algorithm required more evaluations to achieve the best results. Therefore, a balance between the average disassembly time and the NFE needs to be considered to determine the best parameter settings.

By utilising the SPM and PEI, it becomes possible to identify the optimal parameter settings and evaluate the performance of the algorithms. From a statistical standpoint, it is evident that BA_F outperform EDBA, with BA_F demonstrating a faster convergence (smaller NFE). This observation is further supported by the average results presented in Tables 5.8 and 5.9, where it can be seen that BA_F consistently achieves lower disassembly times across various iterations and population sizes. BA_F consistenly outperforms EDBA for both gear pumps, particularly in larger datasets, such as Gear Pump B, demonstrating that it is capable of handling more complex problems. Moreover, the statistical tests and PEI calculations consistently indicate that BA_F outperforms EDBA. These findings provide valuable insights into the comparative performance of the algorithms and highlight the advantages of employing BA_F in terms of achieving lower disassembly times and higher PEI values.

5.5 Summary

The enhancement of the BA was achieved by reducing the number of parameters to 4 and by simplifying the algorithm steps, removing elite sites and incorporating the Fibonacci sequence to guide the recruitment of bees for exploring the local search space. The SPM and PEI analyses conducted in this study indicate that BA_F exhibits better performance than the benchmark EDBA in this RDSP problem. EDBA has been compared against other algorithms, including EDBA

variants (EDBA without mutation operator (EDBA-WMO)), Genetic Algorithm with Precedence Preserving Crossover (GA-PPX), and Self-Adaptive Simplified Swarm Optimisation (SASSO), and has demonstrated better performance. Therefore, the fact that BA_F outperforms EDBA in the current study means that BA_F also outperforms these other algorithms. This further strengthens the evidence supporting the effectiveness of BA_F as a robust and efficient algorithm in the context of RDSP. This conclusion is supported by several key factors, including the lower NFE values and the ability to achieve a minimum disassembly time. BA_F consistently outperforms EDBA on both gear pumps. These findings highlight the effectiveness of BA_F as an optimal choice for achieving efficient and effective RDSP. The SPM and PEI are both valuable tools in the field of algorithm analysis. These tools serve as effective means to identify optimal parameter settings and evaluate the performance of algorithms. SPM allows for statistical comparison and analysis of algorithm performance based on specific metrics, providing insights into variations and differences among different parameter settings. The PEI is a versatile metric that combines multiple metrics, such as disassembly time and NFE, into a single index to provide a comprehensive evaluation of algorithm performance. By combining SPM and PEI, researchers and practitioners can make informed decisions regarding parameter settings and algorithm selection to optimise their outcomes in various domains. Moreover, the validation of the proposed approach through a case study provides substantial evidence of the algorithm's robustness and effectiveness in successfully solving the robotic disassembly problem, achieving objective 4. Although this chapter specifically focuses on addressing the challenges of the RDSP problem, it is important to consider the wider applicability of the proposed algorithm in future research. Future studies will explore and compare the proposed enhancements with the BA_2 approach in a broader context, encompassing both continuous and combinatorial problems. The parameters used in this study are smaller compared to those in TSP and VRP problems, which typically employ population sizes of 200 to 400 or even larger. This suggests the need for further research on the potential applications of BA_F in other contexts. This would enable a comprehensive evaluation and understanding of the algorithm's

performance across different problem domains. Furthermore, investigating analogies with other features of honeybees, such as their learning process, olfactory system, and hive organisation, could lead to more improvements to the BA. The algorithm can also be utilised for parameter tuning in machine learning including deep learning, thereby increasing its usefulness as an advanced AI tool. This extension and exploration of the BA in different contexts will contribute to its versatility and applicability in various optimisation domains.

Chapter 6

Conclusion

Remanufacturing, as the backbone of acircular economy (CE), plays a vital role in prolonging the lifespan of products and providing benefits to both customers and manufacturers. The initial step in the remanufacturing process is disassembly, and the integration of robots in disassembly operations enhances efficiency and effectiveness. The research in this field primarily focuses on two key aspects: disassembly sequence planning and line balancing, both of which involve NP-complete problems. To address these challenges, approximate optimisation methods, such as metaheuristics, are commonly employed. In this thesis, the overarching aim is to explore and develop efficient and sustainable solutions for robotic disassembly sequence planning (RDSP) and robotic disassembly sequence planning (RDSP), with a specific emphasis on the application of the bees algorithm (BA) and its novel enhancement as an intelligent nature-inspired algorithm. This thesis presents a comprehensive investigation into these solutions, examining their effectiveness and highlighting their contributions to the field.

The thesis can be summarised by the following key components:

1. Problem Definition and Research Overview: In Chapter 1, the research problem was clearly defined, emphasising the increasing adoption of robotics in disassembly processes and the associated challenges of optimising disassembly sequences and balancing workloads. An

overview of the subsequent chapters was provided to establish the structure and direction of the research.

2. Literature Review and Gap Identification: Chapter 2 gives a thorough literature review, analysing existing research in the areas of RDSP and RDLB. Through the research position table and bibliometric analysis, critical gaps in the fields of RDSP and RDLB were identified, emphasising the need to bridge these gaps to achieve more efficient disassembly processes.

3. RDSP using the sustainability model, sustainability scenarios and case study validation: Chapter 3 addresses the challenges of the RDSP problem by introducing a sustainability model and sustainability scenarios. A case study on gear pumps was conducted to validate the proposed approach and demonstrate its effectiveness in achieving efficient and effective disassembly processes. The optimisation process focused on the utilisation of the BA. Both single-objective (SO) and multiobjective (MO) approaches, including the aggregate approach and the nondominated approach, were employed. The findings indicated that nondominated approaches were more suitable for addressing the RDSP problem. The analysis of the sustainable recovery option scenarios using the algorithm revealed that it yielded the highest objectives in terms of monetary value compared to the three predefined scenarios, thereby highlighting the viability of using algorithms to identify the best recovery options. The multiobjective nondominated bees algorithm (MOBA) outperformed other benchmark algorithms, as evidenced by the SPM and PEI results.

4. RDLBSD using the sustainability model, sustainability scenarios and case study validation: Chapter 4 focuses on addressing the problem of sequence-dependent robotic disassembly line balancing (RDLBSD) by adopting a holistic approach that considers the interconnection between disassembly sequence planning and line balancing. The utilisation of the sustainability model, scenarios, and case study validation was consistent with the approach employed in the previous chapter. Building on the findings from Chapter 3, which highlighted

the suitability of the nondominated approach, the MOBA was employed to optimise both aspects simultaneously. This approach resulted in an efficient and effective disassembly process that balanced profitability, energy savings, environmental impact reduction, and line balance. The performance of MOBA algorithm was demonstrated through the PEI, and it outperformed other algorithms.

5. Enhancement of the BA: Chapter 5 was dedicated to enhancing the BA specifically for the robotic disassembly problem, with validation conducted on the RDSP problem. The proposed Fibonacci bees algorithm (BA_F), a novel enhancement that reduces parameter settings from six to four and simplifies algorithm steps, was introduced. The effectiveness of BA_F was rigorously evaluated using two novel tools introduced in the Chapter 3: the SPM and the PEI. A comparison was made between BA_F and the EDBA, demonstrating that BA_F outperforms the EDBA in solving the RDSP problem.

6.1 Contributions

This work has made the following key contributions:

1. Sustainability model and sustainability scenarios: Objective 1 in this thesis has been successfully achieved through the development of a sustainability model and scenarios, as demonstrated in Chapters 3 and 4. The sustainability model addresses three objectives in RDSP and four objectives in RDLBSD. The findings demonstrate that these problems are well suited to be solved using a nondominated approach, which takes into account the inherent conflicts among the objectives and the need for trade-offs in decision-making. Notably, the profit objective yields the highest monetary value, highlighting the profitability of remanufacturing and its significant role in promoting a CE. Additionally, remanufacturing contributes to reducing the environmental burden by extending the product lifespan and

enabling multiple reuse cycles before recycling or disposal. To explore various recovery paths for each part, predefined scenarios of recycling (REC), remanufacturing (REM), and reuse (REU) were employed. A novel scenario, ARS, was also introduced, where the algorithm identifies the optimal recovery option for each part. The findings illustrate that the ARS scenario can identify the optimal recovery option under ideal circumstances where all parts can be disassembled and are in suitable condition for reuse. This sustainability model serves as a foundation for future research endeavours, enabling the incorporation of complexities such as unexpected disassembly scenarios.

2. Optimal Sequence Order in Robotic Disassembly: Chapters 3 and 5 were dedicated to achieving the second objective of this thesis, which focused on determining the optimal sequence order in robotic disassembly. The BA was applied in chapter 3, while the BA_F was applied in chapter 5. The findings of chapter 3 demonstrated that the MOBA outperforms other algorithms, as proven by the SPM and PEI. The integration of the sustainability model and scenarios provided valuable insights into incorporating sustainability considerations into the decision-making process. In chapter 5, the enhancement of BA, BA_F, is introduced, and the findings demonstrate that the BA_F outperforms the basic bees algorithm in the robotic disassembly problem, EDBA. In both chapters, the optimal parameter settings were successfully identified.

3. Sequence-Dependent Robotic Disassembly Line Balancing: Chapter 4 addressed the third objective of this thesis by optimising RDLBSD. The interconnection between disassembly sequence planning and line balancing was taken into account. To achieve this, the MOBA was employed, utilising the sustainability model and scenarios developed earlier in the thesis. The work presented in this chapter successfully optimised RDLBSD while considering sustainability objectives.

4. Case Study Validation: The selection of real end-of-life (EoL) gear pumps as the case

study not only highlights their practical applicability but also their suitability for validating the proposed approach. These gear pumps have been extensively studied in the literature of robotic disassembly sequence planning and line balancing, further emphasising their relevance in achieving objective 4 of this thesis. By using these gear pumps as the case study, the research successfully validates the proposed approach in Chapters 3, 4, and 5.

5. Novel Methods for Optimal Parameter Settings and Performance Evaluation. Novel methods were proposed in this thesis to address the challenges of finding optimal parameter settings and evaluating algorithm performance, effectively achieving the fifth objective. These methods fill the gaps in previous research where the potential of statistical tests was underutilised and conflicting results from multiple performance metrics needed to be simplified. Chapter 3 introduces the use of SPM to identify optimal parameter settings through rigorous statistical tests specifically suited for SO approaches. Additionally, the PEI was introduced as a simple yet versatile index to evaluate algorithm performance, applicable to both SO and MO-ND approaches. Both SPM and PEI were utilised in Chapters 3, 4, and 5, providing valuable insights and streamlining the evaluation process. Notably, these tools are not limited to robotic disassembly but can be generalised to compare optimisation algorithms in various domains, further contributing to the field.

6. Enhancement of the BA for Robotic Disassembly Planning: Inspired by the Fibonacci sequence inherited from drone ancestry, the BA was augmented by incorporating the concept of the Fibonacci sequence into the local search process and reducing the number of parameter settings to four. This enhancement, BA_F, was introduced in chapter 5, successfully achieving objective 6 of this study. The BA_F was utilised in the SO version, applied in the RDSP problem. The findings demonstrate that the BA_F outperforms EDBA, especially on larger problems. This highlights the BA_F's effectiveness in addressing complex real-world disassembly challenges. Furthermore, the reduction of parameter settings not only improves

the algorithm's performance but also reduces the time required for parameter tuning.

6.2 Implications of Findings

The findings of this thesis carry significant implications that extend beyond the field of robotic disassembly.

1. Theoretical implications:

   (a) The development and application of the sustainability model contribute to the theoretical understanding of incorporating sustainability principles into robotic disassembly. This expands the theoretical knowledge base and promotes a broader understanding of sustainable practices.

   (b) The investigation and optimisation of sequence order in robotic disassembly expand existing theories and models. The nondominated approach proves suitable for achieving optimal sequence orders, advancing theoretical understanding in the field.

   (c) The use of sequence-dependent robotic disassembly line balancing aligns with current theories on sequence-dependent line balancing methodologies.

   (d) The application and extension of multiple criteria decision-making concepts to robotic disassembly through the PEI offer new insights into evaluating algorithm performance, advancing theoretical understanding in complex decision-making scenarios.

   (e) The use of statistical methodologies for parameter settings, SPM, presents a novel perspective on optimising the efficacy of evaluation tools for assessing algorithm performance.

   (f) The introduction of the BA_F represents a theoretical contribution by incorporating the Fibonacci sequence concept into the BA. This advancement enhances the theoretical

understanding of algorithmic improvements inspired by natural phenomena, drawing analogies from the drone's ancestry.

(g) Another significant theoretical contribution is the reduction of parameter settings in the BA, a novel approach not fully explored in the context of robotic disassembly before. This reduction saves time in finding optimal parameter settings without compromising algorithm performance.

2. Methodological implications:

(a) SPM: The introduction of SPM addresses a methodological gap by leveraging statistical tests to assess performance in optimisation. This extends beyond robotic disassembly and enhances the methodology of performance evaluation.

(b) PEI: The PEI provides a simplified and versatile approach to evaluating algorithm performance that is applicable to various optimisation problems. This methodological improvement facilitates effective algorithm selection and enables fair comparisons among different approaches.

3. Practical implications: The practical implications highlight the application and benefits of the developed sustainability model in real-world scenarios, particularly for EoL products. By incorporating sustainability objectives and automatic recovery scenarios into decision-making, the model empowers industry practitioners to make informed decisions that align with sustainability goals. One significant practical implication is the potential to encourage industry focus on remanufacturing by showcasing its profitability and aligning it with sustainability goals. The sustainability model guides industry practitioners towards more sustainable practices, including remanufacturing, as a viable business strategy.

6.3 Future work

While this thesis has made contributions to the area of robotic disassembly and the enhancement of the BA, there are several areas for future research that can further enhance the understanding and application of these findings. Potential directions for future research include the following:

1. Stochastic and Uncertainty Analysis: Incorporating stochastic elements and addressing uncertainties in disassembly processes can provide a more realistic representation of real-world scenarios. Future research should explore the use of probabilistic models and optimisation algorithms capable of handling uncertainties in disassembly sequences and line balancing.

2. Hazardous and Selective Disassembly: Expanding the scope of the robotic disassembly problem to include hazardous parts and selective disassembly, such as prioritising the retrieval of expensive components, presents opportunities for future research to address more complex scenarios and considerations.

3. Leveraging Advanced Artificial Intelligence (AI) Techniques and Technologies: Integrating advanced AI techniques and technologies, such as deep learning and digital twin technology, holds promise for improving the accuracy and decision-making capabilities of disassembly processes. While the most recent research in RDSP has explored the use of digital twins and deep learning, these techniques are still in their infancy and require further exploration. It is noteworthy that their application in RDLB is currently limited in the literature, presenting an opportunity for future research to investigate their potential impact on the line balancing efficiency and effectiveness. By utilising deep learning algorithms and leveraging digital twin technology, researchers can optimise and refine the disassembly processes, enabling more precise decision-making and improved overall performance.

4. Generalisability and scalability: While this thesis has focused on specific scenarios and components, future research should aim to assess the generalisability and scalability of the proposed models. Conducting studies on a wider range of real EoL products and more complex items will help validate the effectiveness and applicability of the proposed approaches.

5. Industry Collaboration and Implementation: Collaborating with industry stakeholders and implementing the proposed models and algorithms in real-world settings can provide valuable insights and practical feedback. Future research should aim to establish partnerships with remanufacturers and manufacturers to validate and implement the proposed approaches in industrial settings.

6. Extension of BA_F: Future research should explore extending the BA_F to other optimisation problems and domains to assess its applicability. Additionally, conducting a fair comparison between BA_F and the recently introduced bees algorithm with 2 parameters (BA_2) can provide further insights into their relative performance and effectiveness. Furthermore, investigating analogies with other features of honeybees, such as their learning process, olfactory system, and hive organisation, could lead to more improvements to the BA. The BA can also be utilised for parameter tuning in machine learning and deep learning, thereby enhancing its potential to address advanced artificial intelligence techniques, as mentioned in point 3. This extension and exploration of the BA in different contexts will contribute to its versatility and applicability in various optimisation domains.

By addressing these future research directions, the field of robotic disassembly can continue to advance and develop more efficient and sustainable disassembly processes. Moreover, these research directions can contribute to the ongoing development of the BA and its applications in various domains.

References

[1] N. Hartono, F. J. Ramírez, D. T. Pham, Optimisation of Robotic Disassembly Plans using the Bees Algorithm, Robotics and Computer-Integrated Manufacturing 78 (2022) 102411.

[2] N. Hartono, F. J. Ramírez, D. T. Pham, A Sustainability-based Model for Robotic Disassembly Sequence Planning in Remanufacturing using the Bees Algorithm, IFAC-PapersOnLine 55 (2022) 1013–1018.

[3] N. Hartono, F. J. Ramírez, D. T. Pham, Optimisation of Robotic Disassembly Sequence Plans for Sustainability using the multi-objective Bees Algorithm, in: Intelligent Production and Manufacturing Optimisation—The Bees Algorithm Approach, Springer Series in Advanced Manufacturing, 1 ed., Springer, 2023, pp. 337–363.

[4] N. Hartono, F. J. Ramírez, D. T. Pham, A Multiobjective Decision-Making Approach for Modelling and Planning Economically and Environmentally Sustainable Robotic Disassembly for Remanufacturing, Computers and Industrial Engineering (2023).

[5] N. Hartono, 'Sustaina-bee-lity in Remanufacturing' Live @ UoB 3MT Finals 2022, https://www.youtube.com/watch?v=MrSFcP3BVfg, 2022.

[6] N. Hartono, Sustaina-BEE-lity in remanfacturing, https://api.ltb.io/show/ABDQC, 2021.

[7] N. Hartono, UoB Advanced Research Computing (BEAR) PGR Conference, https://twitter.com/uob_bear/status/1461651063141027845?t=jsPYGkTWYRuZQnMxY2ZPmA&s=19, 2021. Accessed: 2021-11-19.

[8] N. Hartono, Sustaina-bee-lity in Remanufacturing, in: Circular Revolution 2021, 2021.

[9] N. Hartono, Sustaina-bee-lity in Remanufacturing, in: Net-Zero Futures 21 Conference: Developing Skills and Talent for the Zero-Carbon Transition, 2021.

[10] N. Hartono, Remanufacturing: Pathway to Sustaina-bee-lity, in: Engineering Professors Council: A Better World EPC Congress 2022, 2022.

[11] N. Hartono, Route to Sustaina-bee-lity, https://api.ltb.io/show/ABMAZ, 2022.

[12] United Nations, Net-zero Coalition, n.d. URL: https://www.un.org/en/climatechange/net-zero-coalition.

[13] United Nations, Sustainability, n.d. URL: https:///www.un.org/en/academic-impact/sustainability.

[14] B. Purvis, Y. Mao, D. Robinson, Three Pillars of Sustainability: In Search of Conceptual Origins, Sustainability science 14 (2019) 681–695.

[15] United Nations, United Nations Sustainable Development, n.d. URL: https://www.un.org/sustainabledevelopment/.

[16] H. B. Lee, N. W. Cho, Y. S. Hong, A Hierarchical End-of-Life Decision Model for Determining the Economic Levels of Remanufacturing and Disassembly under Environmental Regulations, Journal of Cleaner Production 18 (2010) 1276–1283.

[17] K. Xia, L. Gao, W. Li, K. M. Chao, Disassembly Sequence Planning using a Simplified Teaching-Learning-Based Optimization Algorithm, Advanced Engineering Informatics 28 (2014) 518–527.

[18] Ellen MacArthur Foundation, Towards a Circular Economy: Business Rationale for an Accelerated Transition, 2015.

[19] M. Geissdoerfer, P. Savaget, N. M. Bocken, E. J. Hultink, The Circular Economy - A New Sustainability Paradigm?, Journal of Cleaner Production 143 (2017) 757–768.

[20] Q. Chen, X. Lai, Y. Hou, H. Gu, L. Lu, X. Liu, D. Ren, Y. Guo, Y. Zheng, Investigating the Environmental Impacts of Different Direct Material Recycling and Battery Remanufacturing Technologies on Two Types of Retired Lithium-Ion Batteries from Electric Vehicles in China, Separation and Purification Technology 308 (2023) 122966.

[21] European Remanufacturing Council, Supporting Remanufacturing – The Backbone of the Circular Economy, https://www.remancouncil.eu/, n.a.

[22] European Remanufacturing Network, The European Remanufacturing Network, https://www.remanufacturing.eu/, n.a.

[23] BSI, British Standards Document BS 8887-220 Design for manufacture, assembly, disassembly and end-of-life processing (MADE). The process of remanufacture. Specification, 2010. URL: https://doi.org/10.3403/30205839U.

[24] M. Matsumoto, W. Ijomah, Remanufacturing, in: Handbook of Sustainable Engineering, Springer, 2013, pp. 389–408.

[25] RIC, What is Remanufacturing?, http://www.remancouncil.org/educate/remanufacturing-information/what-is-remanufacturing, 2017.

[26] J. Chiodo, W. L. Ijomah, Use of Active Disassembly Technology to Improve Remanufacturing Productivity: Automotive Application, International Journal of Computer Integrated Manufacturing 27 (2014) 361–371.

[27] X. Zhang, M. Zhang, H. Zhang, Z. Jiang, C. Liu, W. Cai, A Review on Energy, Environment and Economic Assessment in Remanufacturing Based on Life Cycle Assessment Method, Journal of Cleaner Production 255 (2020) 120160.

[28] V. M. Smith, G. A. Keoleian, The Value of Remanufactured Engines: Life-Cycle Environmental and Economic Perspectives, Journal of Industrial Ecology 8 (2004) 193–221.

[29] Ellen MacArthur Foundation, Circular Example: Europe's First Circular Economy

Factory for Vehicles: Renault, https://ellenmacarthurfoundation.org/circular-examples/groupe-renault, n.a.

[30] H. Wang, D. Xiang, Y. Rong, L. Zhang, Intelligent Disassembly Planning: A Review on Its Fundamental Methodology, Assembly Automation (2013).

[31] L. Wang, X. V. Wang, L. Gao, J. Váncza, A Cloud-Based Approach for WEEE Remanufacturing, CIRP annals 63 (2014) 409–412.

[32] A. J. Lambert, Disassembly Sequencing: A Survey, International Journal of Production Research 41 (2003) 3721–3759.

[33] Z. Zhou, J. Liu, D. T. Pham, W. Xu, F. J. Ramirez, C. Ji, Q. Liu, Disassembly Sequence Planning: Recent Developments and Future Trends, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture (2018) 0954405418789975.

[34] A. Priyono, W. Ijomah, U. S. Bititci, Disassembly for Remanufacturing: A Systematic Literature Review, New Model Development and Future Research Needs, Journal of Industrial Engineering and Management (JIEM) 9 (2016) 899–932.

[35] BSI, British Standards Publication BS ISO 8887-1:2017 Technical product documentation-Design for manufacturing, assembling, disassembling and end-of-life processing - Part 1: General concepts and requirements, 2017.

[36] H. J. Han, J. M. Yu, D. H. Lee, Mathematical Model and Solution Algorithms for Selective Disassembly Sequencing with Multiple Target Components and Sequence-Dependent Setups, International Journal of Production Research 51 (2013) 4997–5010.

[37] L. Brennan, S. M. Gupta, K. N. Taleb, Operations planning issues in an assembly/disassembly environment, International Journal of Operations & Production Management (1994).

[38] F. Touzanne, J. Henrioud, C. Perrard, Method of Disassembly Sequence Generation for Recycling System Design, in: Proceedings of the 2001 IEEE International Symposium on Assembly and Task Planning (ISATP2001). Assembly and Disassembly in the Twenty-first Century.(Cat. No. 01TH8560), IEEE, 2001, pp. 458–463.

[39] S. K. S. Fan, C. Fan, J. H. Yang, K. F. R. Liu, Disassembly and Recycling Cost Analysis of Waste Notebook and the Efficiency Improvement by Re-design Process, Journal of Cleaner Production 39 (2013) 209–219.

[40] F. Jovane, Q. Semeraro, A. Armillotta, On the Use of the Profit Rate Function in Disassembly Process Planning, The Engineering Economist 43 (1998) 309–330.

[41] S. McGovern, S. M. Gupta, The Disassembly Line: Balancing and Modeling, McGraw-Hill, 2011.

[42] S. Parsa, M. Saadat, Human-Robot Collaboration Disassembly Planning for End-of-Life Product Disassembly Process, Robotics and Computer-Integrated Manufacturing 71 (2021) 102170.

[43] Y. Laili, X. Li, Y. Wang, L. Ren, X. Wang, Robotic Disassembly Sequence Planning With Backup Actions, IEEE Transactions on Automation Science and Engineering (2021).

[44] Y. Laili, F. Tao, D. T. Pham, Y. Wang, L. Zhang, Robotic Disassembly Re-Planning using a Two-Pointer Detection Strategy and a Super-Fast Bees Algorithm, Robotics and Computer-Integrated Manufacturing 59 (2019) 130–142.

[45] S. Vongbunyong, S. Kara, M. Pagnucco, Application of Cognitive Robotics in Disassembly of Products, CIRP Annals - Manufacturing Technology 62 (2013) 31–34.

[46] L. Lindkvist Haziri, E. Sundin, Supporting Design for Remanufacturing-A Framework for Implementing Information Feedback from Remanufacturing to Product Design, Journal of Remanufacturing 10 (2020) 57–76.

[47] J. Liu, Z. Zhou, D. T. Pham, W. Xu, C. Ji, 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 61 (2020) 101829.

[48] B. Chen, W. Xu, J. Liu, Z. Ji, Z. Zhou, Robotic Disassembly Sequence Planning Considering Robotic Collision Avoidance Trajectory in Remanufacturing, in: 2020 IEEE 18th International Conference on Industrial Informatics (INDIN), volume 1, IEEE, 2020, pp. 494–501.

[49] J. Liu, Q. Liu, Z. Zhou, D. T. Pham, W. Xu, Y. Fang, Collaborative Optimisation of Robotic Disassembly Planning Problems using the Bees Algorithm, in: Intelligent Production and Manufacturing Optimisation—The Bees Algorithm Approach, Springer, 2023, pp. 305–335.

[50] S. Ong, M. Chang, A. Nee, Product Disassembly Sequence Planning: State-Of-The-Art, Challenges, Opportunities and Future Directions, International Journal of Production Research 59 (2021) 3493–3508.

[51] J. Liu, Z. Zhou, D. T. Pham, W. Xu, J. Yan, A. Liu, C. Ji, Q. Liu, An Improved Multi-Objective Discrete Bees Algorithm for Robotic Disassembly Line Balancing Problem in Remanufacturing, The International Journal of Advanced Manufacturing Technology 97 (2018) 3937–3962.

[52] Y. Gao, Q. Wang, Y. Feng, H. Zheng, B. Zheng, J. Tan, An Energy-Saving Optimization Method of Dynamic Scheduling for Disassembly Line, Energies 11 (2018) 1261.

[53] Y. Wang, F. Lan, J. Liu, J. Huang, S. Su, C. Ji, D. T. Pham, W. Xu, Q. Liu, Z. Zhou, Interlocking Problems in Disassembly Sequence Planning, International Journal of Production Research 59 (2021) 4723–4735.

[54] J. Liu, Z. Zhou, D. T. Pham, W. Xu, C. Ji, Q. Liu, Robotic Disassembly Sequence Planning using Enhanced Discrete Bees Algorithm in Remanufacturing, International Journal of Production Research 56 (2018) 3134–3151.

[55] J. Liu, Z. Zhou, D. T. Pham, W. Xu, J. Cui, C. Yang, Service Platform for Robotic Disassembly Planning in Remanufacturing, Journal of Manufacturing Systems 57 (2020) 338–356.

[56] M. A. Ilgin, H. Akçay, C. Araz, Disassembly Line Balancing using Linear Physical Programming, International Journal of Production Research 55 (2017) 6108–6119.

[57] A. ElSayed, E. Kongar, S. M. Gupta, T. Sobh, A Robotic-driven Disassembly Sequence Generator for End-of-life Electronic Products, Journal of Intelligent & Robotic Systems 68 (2012) 43–52.

[58] K. Meng, P. Lou, X. Peng, V. Prybutok, Multi-Objective Optimization Decision-Making of Quality Dependent Product Recovery for Sustainability, International Journal of Production Economics 188 (2017) 72–85.

[59] S. Parsa, M. Saadat, Intelligent Planning using Genetic Algorithm for Automated Disassembly, in: Advances in Manufacturing Technology XXXII, IOS Press, 2018, pp. 189–194.

[60] Y. Xing, D. Wu, L. Qu, Parallel Disassembly Sequence Planning using Improved Ant Colony Algorithm, The International Journal of Advanced Manufacturing Technology 113 (2021) 2327–2342.

[61] Q. Lu, Y. Ren, H. Jin, L. Meng, L. Li, C. Zhang, J. W. Sutherland, A Hybrid Metaheuristic Algorithm for a Profit-Oriented and Energy-Efficient Disassembly Sequencing Problem, Robotics and Computer-Integrated Manufacturing 61 (2020) 101828.

[62] E. Cevikcan, D. Aslan, F. B. Yeni, Disassembly Line Design with Multi-manned Workstations: A Novel Heuristic Optimisation Approach, International Journal of Production Research 58 (2020) 649–670.

[63] J. Guo, Z. Pu, B. Du, Y. Li, Multi-Objective Optimisation of Stochastic Hybrid Production Line Balancing Including Assembly and Disassembly Tasks, International Journal of Production Research 60 (2022) 2884–2900.

[64] Y.-S. Ma, H.-B. Jun, H.-W. Kim, D.-H. Lee, Disassembly Process Planning Algorithms for End-of-Life Product Recovery and Environmentally Conscious Disposal, International Journal of Production Research 49 (2011) 7007–7027.

[65] J. F. Gonçalves, J. J. de Magalhães Mendes, M. G. Resende, A Hybrid Genetic Algorithm for the Job Shop Scheduling Problem, European Journal of Operational Research 167 (2005) 77–95.

[66] A. Elsayed, E. Kongar, S. M. Gupta, A Genetic Algorithm Approach to End-of-life Disassembly Sequencing for Robotic Disassembly, Proceedings of the 2010 Northeast Decision Sciences Institute Conference 1 (2010) 402–408.

[67] M. R. Johnson, M. H. Wang, Economical Evaluation of Disassembly Operations for Recycling, Remanufacturing and Reuse, International Journal of Production Research 36 (1998) 3227–3252.

[68] A. Lambert, Linear Programming in Disassembly/Clustering Sequence Generation, Computers & Industrial Engineering 36 (1999) 723–738.

[69] A. Lambert, Exact Methods in Optimum Disassembly Sequence Search for Problems

Subject to Sequence Dependent Costs, Omega 34 (2006) 538–549.

[70] Y. Fang, Q. Liu, M. Li, Y. Laili, D. T. Pham, Evolutionary Many-Objective Optimization for Mixed-Model Disassembly Line Balancing with Multi-Robotic Workstations, European Journal of Operational Research 276 (2019) 160–174.

[71] Y. Fang, H. Wei, Q. Liu, Y. Li, Z. Zhou, D. T. Pham, Minimizing Energy Consumption and Line Length of Mixed-model Multirobotic Disassembly Line Systems using Multi-Objective Evolutionary Optimization, volume 1, American Society of Mechanical Engineers (ASME), 2019.

[72] Y. Fang, H. Ming, M. Li, Q. Liu, D. T. Pham, Multi-Objective Evolutionary Simulated Annealing Optimisation for Mixed-Model Multi-Robotic Disassembly Line Balancing with Interval Processing Time, International Journal of Production Research 58 (2020) 846–862.

[73] Y. Fang, H. Xu, Constraint Handling Methods for Resource-Constrained Robotic Disassembly Line Balancing Problem, volume 1576, Institute of Physics Publishing, 2020.

[74] Q. Liu, Y. Li, Y. Fang, Y. Laili, P. Lou, D. T. Pham, Many-Objective Best-Order-Sort Genetic Algorithm for Mixed-Model Multi-Robotic Disassembly Line Balancing, volume 83, Elsevier B.V., 2019, pp. 14–21.

[75] C. Dong, P. Liu, X. W. Guo, L. Qi, S. Qin, G. Xu, Multi-objective Ant Lion Optimizer for Stochastic Robotic Disassembly Line Balancing Problem Subject to Resource Constraints, volume 2024, IOP Publishing Ltd, 2021.

[76] S. Zhang, L. Guo, X. Guo, S. Liu, L. Qi, S. Qin, Y. Tang, Z. Zhao, Multi-Objective Multi-Verse Optimizer for Multi-Product Partial U-Shaped Disassembly Line Balancing Problem, Institute of Electrical and Electronics Engineers Inc., 2021.

[77] T. Yin, Z. Zhang, Y. Zhang, T. Wu, W. Liang, Mixed-Integer Programming Model and Hybrid Driving Algorithm for Multi-Product Partial Disassembly Line Balancing Problem with Multi-Robot Workstations, Robotics and Computer-Integrated Manufacturing 73 (2022).

[78] B. Zhou, J. Bian, Multi-Mechanism-Based Modified Bi-Objective Harris Hawks Optimization for Sustainable Robotic Disassembly Line Balancing Problems, Engineering Applications of Artificial Intelligence 116 (2022) 105479.

[79] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Robotic Disassembly Sequence Re-planning, Springer International Publishing, Cham, 2022, pp. 131–142.

[80] J. Li, M. Barwood, S. Rahimifard, Robotic Disassembly for Increased Recovery of Strategically Important Materials from Electrical Vehicles, Robotics and Computer-Integrated Manufacturing 50 (2018) 203–212.

[81] H. Poschmann, H. Brueggemann, D. Goldmann, Disassembly 4.0: A Review on using Robotics in Disassembly Tasks as a Way of Automation, Chemie Ingenieur Technik 92 (2020) 341–359.

[82] M. Daneshmand, F. Noroozi, C. Corneanu, F. Mafakheri, P. Fiorini, Industry 4.0 and

Prospects of Circular Economy: A Survey of Robotic Assembly and Disassembly, The International Journal of Advanced Manufacturing Technology (2022) 1–28.

[83] M. Alshibli, A. ElSayed, E. Kongar, T. Sobh, S. M. Gupta, A Robust Robotic Disassembly Sequence Design using Orthogonal arrays and Task Allocation, Robotics 8 (2019) 20.

[84] S. Vongbunyong, W. H. Chen, Disassembly Automation Automated Systems with Cognitive Abilities, Springer International Publishing, 2015.

[85] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Robotic Disassembly for Remanufacturing, Springer International Publishing, Cham, 2022, pp. 7–25.

[86] J. Liu, Z. Xu, H. Xiong, Q. Lin, W. Xu, Z. Zhou, Digital Twin-driven Robotic Disassembly Sequence Dynamic Planning under Uncertain Missing Condition, IEEE Transactions on Industrial Informatics (2023).

[87] K. Mei, Y. Fang, Multi-Robotic Disassembly Line Balancing Using Deep Reinforcement Learning, volume 2, American Society of Mechanical Engineers, 2021.

[88] C. Yang, W. Xu, J. Liu, B. Yao, Y. Hu, Robotic Disassembly Sequence Planning Considering Robotic Movement State Based on Deep Reinforcement Learning, in: 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), IEEE, 2022, pp. 183–189.

[89] J. Cui, C. Yang, J. Zhang, S. Tian, J. Liu, W. Xu, Robotic Disassembly Sequence Planning Considering Parts Failure Features, IET Collaborative Intelligent Manufacturing 5 (2023) e12074.

[90] T. F. Go, D. A. Wahab, M. N. A. Rahman, R. Ramli, A. Hussain, Genetically Optimised Disassembly Sequence for Automotive Component Reuse, Expert Systems with Applications 39 (2012) 5409–5417.

[91] H. H. T. Huang, M. H. Wang, M. R. Johnson, Disassembly Sequence Generation using a Neural Network Approach, Journal of Manufacturing Systems 19 (2000) 73–82.

[92] J. G. Kang, P. Xirouchakis, Disassembly Sequencing for Maintenance: A Survey, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 220 (2006) 1697–1716.

[93] H. E. Tseng, C. C. Chang, S. C. Lee, Y. M. Huang, A Block-based Genetic Algorithm for Disassembly Sequence Planning, Expert Systems with Applications 96 (2018) 492–505.

[94] L. Zhong, S. Youchao, O. E. Gabriel, W. Haiqiao, Disassembly Sequence Planning for Maintenance Based on Metaheuristic Method, Aircraft Engineering and Aerospace Technology (2011).

[95] T. C. Kuo, Waste Electronics and Electrical Equipment Disassembly and Recycling using Petri Net Analysis: Considering the Economic Value and Environmental Impacts, Computers and Industrial Engineering 65 (2013) 54–64.

[96] C. A. Tovey, Tutorial on Computational Complexity, Interfaces 32 (2002) 30–61.

[97] E.-G. Talbi, Metaheuristics: from Design to Implementation, Wiley, 2009.

[98] W. Zhang, H. Lu, M. Ma, C. Kong, H. Huang, S. Wang, The Manipulator Path Planning of Bolt Disassembly Based on Improved Genetic Algorithm and A* Algorithm, in: 2019 6th International Conference on Systems and Informatics (ICSAI), IEEE, 2019, pp. 176–182.

[99] J. Dong, 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 217 (2003) 299–312.

[100] M. Alshibli, A. El Sayed, O. Tozanli, E. Kongar, T. M. Sobh, S. M. Gupta, A Decision Maker-centered End-of-Life Product Recovery System for Robot Task Sequencing, Journal of Intelligent & Robotic Systems 91 (2018) 603–616.

[101] S. Malekkhouyan, A. Aghsami, M. Rabbani, An Integrated Multi-Stage Vehicle Routing and Mixed-Model Job-Shop-Type Robotic Disassembly Sequence Scheduling Problem for E-Waste Management System, International Journal of Computer Integrated Manufacturing 34 (2021) 1237–1262.

[102] K. Wang, L. Gao, X. Li, P. Li, Energy-Efficient Robotic Parallel Disassembly Sequence Planning for End-of-Life Products, IEEE Transactions on Automation Science and Engineering (2021).

[103] T. Suzuki, T. Zanma, A. Inaba, S. Okuma, Learning Control of Disassembly Petri Net - An Approach with Discrete Event System Theory, in: Proceedings of IEEE International Conference on Robotics and Automation, volume 1, IEEE, 1996, pp. 184–191.

[104] S. Sundaram, I. Remmler, N. M. Amato, Disassembly Sequencing using a Motion Planning Approach, in: Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), volume 2, IEEE, 2001, pp. 1475–1480.

[105] J. P. Baeza, F. T. Medina, S. P. Mendez, Disassembly movements for geometrical objects through heuristic methods, in: Environmentally Conscious Manufacturing II, volume 4569, SPIE, 2002, pp. 71–80.

[106] S. Puente, F. Torres, R. Aracil, Non-Destructive Disassembly Robot Cell for Demanufacturing Automation, IFAC Proceedings Volumes 36 (2003) 97–102.

[107] E. Uhlmann, T. Friedrich, G. Seliger, R. Harms, Realization of an Adaptive Modular Control for a Disassembly System, in: (ISATP 2005). The 6th IEEE International Symposium on Assembly and Task Planning: From Nano to Macro Assembly and Manufacturing, 2005., IEEE, 2005, pp. 32–35.

[108] H. J. Kim, R. Harms, G. Seliger, Automatic Control Sequence Generation for a Hybrid Disassembly System, IEEE transactions on automation science and engineering 4 (2007) 194–205.

[109] P. Gil, J. Pomares, S. v. P. C. Diaz, F. Candelas, F. Torres, Flexible Multi-Sensorial System for Automatic Disassembly using Cooperative Robots, International Journal of Computer Integrated Manufacturing 20 (2007) 757–772.

[110] A. ElSayed, E. Kongar, S. Gupta, T. Sobh, An Online Genetic Algorithm for Automated Disassembly Sequence Generation, in: Proceedings of the ASME 2011 International

Design Engineering Technical Conferences & Computers and Information in Engineering Conference, IDETC/CIE, 2011, pp. 657–664.

[111] S. Vongbunyong, S. Kara, M. Pagnucco, Learning and Revision in Cognitive Robotics Disassembly Automation, Robotics and Computer-Integrated Manufacturing 34 (2015) 79–94.

[112] D. Popescu, R. Iacob, R. Parpala, T. Dobrescu, Virtual to Real in Robotic Assembly/disassembly Tasks, UPB Sci. Bull., Series D 78 (2016).

[113] M. Alshibli, A. El Sayed, E. Kongar, T. M. Sobh, S. M. Gupta, Disassembly Sequencing using Tabu search, Journal of Intelligent & Robotic Systems 82 (2016) 69–79.

[114] C. Friedrich, A. Lechler, A. Verl, A Planning System for Generating Manipulation Sequences for the Automation of Maintenance Tasks, in: 2016 IEEE International Conference on Automation Science and Engineering (CASE), IEEE, 2016, pp. 843–848.

[115] S. Vongbunyong, P. Vongseela, J. Sreerattana-Aporn, A Process Demonstration Platform for Product Disassembly Skills Transfer, Procedia CIRP 61 (2017) 281–286.

[116] C. Friedrich, A. Csiszar, A. Lechler, A. Verl, Fast Robot Task and Path Planning Based on CAD and Vision Data, in: 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), IEEE, 2017, pp. 1633–1638.

[117] Y. Wang, F. Lan, D. T. Pham, J. Liu, J. Huang, C. Ji, S. Su, W. Xu, Q. Liu, Z. Zhou, Automatic Detection of Subassemblies for Disassembly Sequence Planning., in: ICINCO (1), 2018, pp. 104–110.

[118] J. S. Laursen, L.-P. Ellekilde, U. P. Schultz, Modelling Reversible Execution of Robotic Assembly, Robotica 36 (2018) 625–654.

[119] C. M. Costa, G. Veiga, A. Sousa, L. Rocha, E. Oliveira, H. L. Cardoso, U. Thomas, Automatic Generation of Disassembly Sequences And Exploded Views From SOLIDWORKS Symbolic Geometric Relationships, in: 2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE, 2018, pp. 211–218.

[120] N. M. DiFilippo, M. K. Jouaneh, Using the Soar Cognitive Architecture to Remove Screws from Different Laptop Models, IEEE Transactions on Automation Science and Engineering 16 (2018) 767–780.

[121] F. Lan, Y. Wang, D. T. Pham, J. Liu, J. Huang, C. Ji, S. Su, W. Xu, Q. Liu, Z. Zhou, Interlocking Problem in Automatic Disassembly Planning and Two Solutions, in: Informatics in Control, Automation and Robotics: 15th International Conference, ICINCO 2018, Porto, Portugal, July 29-31, 2018, Revised Selected Papers 15, Springer, 2020, pp. 193–213.

[122] F. J. Ramírez, J. A. Aledo, J. A. Gamez, D. T. Pham, Economic Modelling of Robotic Disassembly in End-of-Life Product Recovery for Remanufacturing, Computers & Industrial Engineering 142 (2020) 106339.

[123] K. Watanabe, S. Inada, Search Algorithm of the Assembly Sequence of Products by using Past Learning Results, International Journal of Production Economics 226 (2020) 107615.

[124] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Modelling of Robotic Disassembly Sequence Planning, Optimisation of Robotic Disassembly for Remanufacturing (2022) 59–69.

[125] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Solutions for Robotic Disassembly Sequence Planning with Backup Actions, Optimisation of Robotic Disassembly for Remanufacturing (2022) 111–130.

[126] F. Ye, J. Perrett, L. Zhang, Y. Laili, Y. Wang, A Self-Evolving System for Robotic Disassembly Sequence Planning Under Uncertain Interference Conditions, Robotics and Computer-Integrated Manufacturing 78 (2022) 102392.

[127] J. P. J. Prioli, H. M. Alrufaifi, J. L. Rickli, Disassembly Assessment From CAD-Based Collision Evaluation for Sequence Planning, Robotics and Computer-Integrated Manufacturing 78 (2022) 102416.

[128] Q. Chen, B. Yao, D. T. Pham, Sequence-Dependent Robotic Disassembly Line Balancing Problem Considering Disassembly Path, in: International Manufacturing Science and Engineering Conference, volume 84263, American Society of Mechanical Engineers, 2020, p. V002T07A019.

[129] N. Hartono, F. Javier Ramírez, D. T. Pham, Using The Bees Algorithm to Optimise Robotic Disassembly Sequences and Balance Disassembly Line for Sustainable Product Recovery, 2023.

[130] J. Liu, S. Wang, Balancing Disassembly Line in Product Recovery to Promote the Coordinated Development of Economy and Environment, Sustainability 9 (2017) 309.

[131] C. B. Kalayci, S. M. Gupta, Artificial Bee Colony Algorithm for Solving Sequence-Dependent Disassembly Line Balancing Problem, Expert Systems with Applications 40 (2013) 7231–7241.

[132] C. B. Kalayci, S. M. Gupta, A Particle Swarm Optimization Algorithm with Neighborhood-Based Mutation for Sequence-Dependent Disassembly Line Balancing Problem, The International Journal of Advanced Manufacturing Technology 69 (2013) 197–209.

[133] C. B. Kalayci, S. M. Gupta, A Tabu Search Algorithm for Balancing a Sequence-Dependent Disassembly Line, Production Planning & Control 25 (2014) 149–160.

[134] C. B. Kalayci, O. Polat, S. M. Gupta, A Hybrid Genetic Algorithm for Sequence-Dependent Disassembly Line Balancing Problem, Annals of Operations Research 242 (2016) 321–354.

[135] S. Wang, X. Guo, J. Liu, An Efficient Hybrid Artificial Bee Colony Algorithm for Disassembly Line Balancing Problem with Sequence-Dependent Part Removal Times, Engineering Optimization 51 (2019) 1920–1937.

[136] Y. Ren, L. Meng, C. Zhang, F. Zhao, U. Saif, A. Huang, G. P. Mendis, J. W. Sutherland, An Efficient Metaheuristics for a Sequence-Dependent Disassembly Planning, Journal of

Cleaner Production 245 (2020) 118644.

[137] A. Radaschin, A. Filipescu, V. Minzu, E. Minca, Adaptive Disassembly Sequence Control by using Mobile Robots and System Information, in: 15th International Conference on System Theory, Control and Computing, IEEE, 2011, pp. 1–6.

[138] Y. Zeng, Z. Zhang, T. Yin, H. Zheng, Robotic disassembly line balancing and sequencing problem considering energy-saving and high-profit for waste household appliances, Journal of Cleaner Production 381 (2022) 135209.

[139] E. Minca, A. Filipescu, A. Voda, Modelling and Control of an Assembly/Disassembly Mechatronics Line Served by Mobile Robot with Manipulator, Control Engineering Practice 31 (2014) 50–62.

[140] E. Minca, H. G. Coanda, F. Dragomir, O. Dragomir, A. Filipescu, Cycle Time Optimization of a Reversible A/DML Served by a Mobile Robotic System, in: 2015 19th International Conference on System Theory, Control and Computing (ICSTCC), IEEE, 2015, pp. 99–104.

[141] A. Filipescu, A. Filipescu, A. Voda, E. Minca, Hybrid Modeling, Balancing and Control of a Mechatronics Line Served by Two Mobile Robots, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 234–239.

[142] D. Octavian, V. Gurgu, E. Minca, A. Filipescu, F. Dragomir, O. Dragomir, Optimal Control of The Complete Assembly/Disassembly Cycle for a Mechatronics Line Prototype, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 620–625.

[143] H. Ming, Q. Liu, D. T. Pham, Multi-Robotic Disassembly Line Balancing with Uncertain Processing Time, volume 83, Elsevier B.V., 2019, pp. 71–76.

[144] Z. A. Çil, S. Mete, F. Serin, Robotic Disassembly Line Balancing Problem: A Mathematical model and Ant Colony Optimization Approach, Applied Mathematical Modelling 86 (2020) 335–348.

[145] Y. Fang, H. Xu, Q. Liu, D. T. Pham, Evolutionary Optimization using Epsilon Method for Resource-Constrained Multi-Robotic Disassembly Line Balancing, Journal of Manufacturing Systems 56 (2020) 392–413.

[146] Y. Fang, H. Zhang, Q. Liu, Z. Zhou, B. Yao, D. T. Pham, Interval Multi-Objective Evolutionary Optimization for Disassembly Line Balancing with Uncertain Task Time, volume 2, American Society of Mechanical Engineers, 2020.

[147] S. Lei, X. Guo, M. Zhou, J. Wang, L. Qi, S. Qin, A constrained decomposition grid approach to disassembly line balancing problems, volume 2021-December, IEEE Computer Society, 2021, pp. 162–167.

[148] H. E. Tseng, C. C. Chang, T. W. Chung, Applying Improved Particle Swarm Optimization to Asynchronous Parallel Disassembly Planning, IEEE Access 10 (2022) 80555–80564.

[149] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Evolutionary Optimisation for Robotic Disassembly Sequence Planning and Line Balancing, Optimisation of Robotic Disassembly for Remanufacturing (2022) 85–110.

[150] S. Zhang, P. Liu, X. Guo, J. Wang, S. Qin, Y. Tang, An Improved Tabu Search Algorithm for Multi-robot Hybrid Disassembly Line Balancing Problems, in: 2022 International Conference on Cyber-Physical Social Intelligence (ICCSI), IEEE, 2022, pp. 315–320.

[151] S. Zhang, X. Guo, J. Wang, S. Liu, S. Qin, Z. Zhao, An Improved Multi-Objective Multi-Verse Optimization Algorithm for Multifunctional Robotic Parallel Disassembly Line Balancing Problems, in: 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2022, pp. 562–567.

[152] Y. Laili, Y. Wang, Y. Fang, D. T. Pham, Solutions for Robotic Disassembly Line Balancing, Optimisation of Robotic Disassembly for Remanufacturing (2022) 143–151.

[153] G. Xu, Z. Zhang, Z. Li, X. Guo, L. Qi, X. Liu, Multi-Objective Discrete Brainstorming Optimizer to Solve the Stochastic Multiple-Product Robotic Disassembly Line Balancing Problem Subject to Disassembly Failures, Mathematics 11 (2023) 1557.

[154] S. Qin, S. Zhang, J. Wang, S. Liu, X. Guo, L. Qi, Multi-objective Multi-verse Optimizer for Multi-robotic U-shaped Disassembly Line Balancing Problems, IEEE Transactions on Artificial Intelligence (2023).

[155] K. Z. Gao, Z. He, Y. Huang, P. Y. Duan, P. N. Suganthan, A Survey on Meta-Heuristics for Solving Disassembly Line Balancing, Planning and Scheduling Problems in Remanufacturing, Swarm and Evolutionary Computation 57 (2020) 100719.

[156] A. Lambert, Optimal Disassembly of Complex Products, International Journal of Production Research 35 (1997) 2509–2524.

[157] B. Yuce, M. S. Packianather, E. Mastrocinque, D. T. Pham, A. Lambiase, Honey Bees Inspired Optimization Method: The Bees Algorithm, Insects 4 (2013) 646–662.

[158] H. De la Torre Gutiérrez, D. T. Pham, Identification of Patterns in Control Charts for Processes with Statistically Correlated Noise, International Journal of Production Research 56 (2018) 1504–1520.

[159] M. Castellani, D. T. Pham, The Bees Algorithm—A Gentle Introduction, Springer International Publishing, Cham, 2023, pp. 3–21.

[160] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi, The Bees Algorithm, Technical Note, Manufacturing Engineering Centre, Cardiff University, UK (2005).

[161] D. T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri, S. Rahim, M. Zaidi, The Bees Algorithm—A Novel Tool for Complex Optimisation Problems, in: Intelligent Production Machines and Systems, Elsevier, 2006, pp. 454–459.

[162] M. Caterino, M. Fera, R. Macchiaroli, D. T. Pham, Task Optimisation for a Modern Cloud Remanufacturing System Using the Bees Algorithm, in: Intelligent Production and Manufacturing Optimisation—The Bees Algorithm Approach, Springer, 2023, pp. 365–382.

[163] M. Caterino, M. Fera, R. Macchiaroli, D. T. Pham, Cloud Remanufacturing: Remanufacturing enhanced through Cloud Technologies, Journal of Manufacturing Systems 64 (2022) 133–148.

[164] S. Zeybek, Prediction of the Remaining Useful Life of Engines for Remanufacturing Using a Semi-supervised Deep Learning Model Trained by the Bees Algorithm, in: Intelligent Production and Manufacturing Optimisation—The Bees Algorithm Approach, Springer, 2023, pp. 383–397.

[165] M. Kerin, N. Hartono, D. T. Pham, Optimising Remanufacturing Decision-Making using the Bees Algorithm in Product Digital Twins, Scientific Reports 13 (2023) 701.

[166] N. Hartono, A. H. Ismail, S. Zeybek, M. Caterino, K. Jiang, M. Sahin, Parameter Tuning for Combinatorial Bees Algorithm in Travelling Salesman Problems, in: AIP Conference Proceedings, volume 2485, AIP Publishing, 2023.

[167] G. A. Shah, A. Polette, J.-P. Pernot, F. Giannini, M. Monti, Case-based Tuning of a Metaheuristic Algorithm Exploiting Sensitivity Analysis and Design of Experiments for Reverse Engineering Applications, Engineering with Computers (2022) 1–17.

[168] Q. T. Pham, D. T. Pham, M. Castellani, A Modified Bees Algorithm and a Statistics-Based Method for Tuning its Parameters, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 226 (2012) 287–301.

[169] M. A. Eirgash, V. Toğan, A Novel Oppositional Teaching Learning Strategy Based on the Golden Ratio to Solve the Time-Cost-Environmental Impact Trade-Off Optimization Problems, Expert Systems with Applications 224 (2023) 119995.

[170] A. H. Halim, I. Ismail, S. Das, Performance Assessment of the Metaheuristic Optimization Algorithms: An Exhaustive Review, Artificial Intelligence Review (2020) 1–87.

[171] J. Durillo, A. Nebro, F. Luna, C. Coello Coello, E. Alba, Convergence Speed in Multi-Objective Metaheuristics: Efficiency Criteria and Empirical Study, International Journal for Numerical Methods in Engineering 84 (2010) 1344–1375.

[172] Y. Cao, B. J. Smucker, T. J. Robinson, On using the Hypervolume Indicator to Compare Pareto fronts: Applications to Multi-criteria Optimal Experimental Design, Journal of Statistical Planning and Inference 160 (2015) 60–74.

[173] E. Zitzler, J. Knowles, L. Thiele, Quality Assessment of Pareto set Approximations, in: Multiobjective Optimization, Springer, 2008, pp. 373–404.

[174] Y. Sun, G. G. Yen, Z. Yi, IGD Indicator-based Evolutionary Algorithm for Many-Objective Optimization Problems, IEEE Transactions on Evolutionary Computation 23 (2018) 173–187.

[175] N. J. Van Eck, L. Waltman, VOS: A New Method for Visualizing Similarities Between Objects, in: Advances in data analysis, Springer, 2007, pp. 299–306.

[176] N. J. Van Eck, L. Waltman, Visualizing Bibliometric Networks, in: Measuring Scholarly Impact: Methods and Practice, Springer, 2014, pp. 285–320.

[177] N. J. Van Eck, L. Waltman, Citation-Based Clustering of Publications using Citnetexplorer and Vosviewer, Scientometrics 111 (2017) 1053–1070.

[178] Grabcad Community, Gear pump 10 l/min., 2020. URL: `https://grabcad.com/`

library/gear-pump-10l-min-1(Accessed:Jul22,2020).

[179] KUKA, Technical Specification, LBR iiwa 7 R800, LBR iiwa 14 R820, Technical Report, KUKA Deutschland GmbH, 2020.

[180] C. Tofallis, Add or Multiply? A Tutorial on Ranking and Choosing with Multiple Criteria, INFORMS Transactions on education 14 (2014) 109–119.

[181] G. Jin, W. Li, K. Xia, Disassembly Matrix for Liquid Crystal Displays Televisions, Procedia CIRP 11 (2013) 357–362.

[182] G. Jin, W. Li, S. Wang, S. Gao, A Systematic Selective Disassembly Approach for Waste Electrical and Electronic Equipment with Case Study on Liquid Crystal Display Televisions, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture (2015) 0954405415575476.

[183] F. J. Ramírez, D. T. Pham, Internal Communication Universidad de Castilla-La Mancha and University of Birmingham, 2019.

[184] J. D. Knowles, L. Thiele, E. Zitzler, A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers, TIK-Report 214 (2006).

[185] S. Zeybek, E. Koç, The Vantage Point Bees Algorithm, in: 2015 7th International Joint Conference on Computational Intelligence (IJCCI), volume 1, IEEE, 2015, pp. 340–345.

[186] A. H. Ismail, N. Hartono, S. Zeybek, D. T. Pham, Using the Bees Algorithm to Solve Combinatorial Optimisation Problems for TSPLIB, IOP Conference Series: Materials Science and Engineering 847 (2020) 012027.

[187] S. Zeybek, A. H. Ismail, N. Hartono, M. Caterino, K. Jiang, An Improved Vantage Point Bees Algorithm to Solve Combinatorial Optimization Problems from TSPLIB, in: Macromolecular symposia, volume 396, Wiley Online Library, 2021, p. 2000299.

[188] D. T. Pham, M. Castellani, The Bees Algorithm: Modelling Foraging Behaviour to Solve Continuous Optimization Problems, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 223 (2009) 2919–2938.

[189] D. T. Pham, M. Castellani, A Comparative Study of The Bees Algorithm as a Tool for Function Optimisation, Cogent Engineering 2 (2015) 1091540.

[190] W. A. Hussein, S. Sahran, S. N. H. Sheikh Abdullah, The Variants of the Bees Algorithm (BA): A Survey, Artificial Intelligence Review 47 (2017) 67–121.

[191] D. T. Pham, A. H. Darwish, Fuzzy Selection of Local Search Sites in the Bees Algorithm, in: Proceedings of the 4th international virtual conference on intelligent production machines and systems (IPROMS 2008), 2008, pp. 1–14.

[192] A. H. Ismail, Enhancing the Bees Algorithm using the Traplining Metaphor, Ph.D. thesis, University of Birmingham, 2021.

[193] A. H. Ismail, D. T. Pham, Bees Traplining Metaphors for the Vehicle Routing Problem Using a Decomposition Approach, Springer International Publishing, Cham, 2023, pp. 261–287.

[194] T. C. Scott, P. Marketos, On the Origin of the Fibonacci Sequence, MacTutor History of Mathematics 23 (2014).

[195] T. Omotehinwa, S. Ramon, Fibonacci Numbers and Golden Ratio in Mathematics and Science, International Journal of Computer and Information Technology (ISSN 2279-0764) (2013).

[196] T. Koshy, Fibonacci and Lucas Numbers with Applications, One, Wiley Online Library, 2017.

[197] P. K. Gudla, R. Ganguli, An Automated Hybrid Genetic-Conjugate Gradient Algorithm for Multimodal Optimization Problems, Applied Mathematics and Computation 167 (2005) 1457–1474.

[198] W.-C. Yeh, An Efficient Memetic Algorithm for the Multi-Stage Supply Chain Network Problem, The International Journal of Advanced Manufacturing Technology 29 (2006) 803–813.

[199] L. Sha, Z. Pan, FSQGA based 3D Complexity Wellbore Trajectory Optimization, Oil & Gas Sciences and Technology–Revue d'IFP Energies nouvelles 73 (2018) 79.

[200] H. Tran-Ngoc, T. Le-Xuan, S. Khatir, G. De Roeck, T. Bui-Tien, M. Abdel Wahab, A Promising Approach using Fibonacci Sequence-based Optimization Algorithms and Advanced Computing, Scientific Reports 13 (2023) 3405.

[201] Q. Zhao, R. Tao, J. Li, Y. Mu, An Improved Wolf Pack Algorithm, in: 2020 Chinese Control and Decision Conference (CCDC), IEEE, 2020, pp. 626–633.

[202] N. Aras, D. Aksen, Locating Collection Centers for Distance-and Incentive-Dependent Returns, International Journal of Production Economics 111 (2008) 316–333.

[203] T. Khosla, O. P. Verma, An Efficient Particle Swarm Optimization with Fusion of Figurate Opposition-based Learning and Grey Wolf Optimizer, in: 2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), IEEE, 2022, pp. 208–213.

[204] C. Da, Z. Wenchao, S. Siqing, L. Yiqun, L. Yazhou, K. Xianbo, A Parameter Identification Algorithm for Turbine-governor System in Regional Power Grid, in: 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), IEEE, 2018, pp. 1–5.

[205] A. Etminaniesfahani, A. Ghanbarzadeh, Z. Marashi, Fibonacci Indicator Algorithm: A Novel Tool for Complex Optimization Problems, Engineering Applications of Artificial Intelligence 74 (2018) 1–9.

[206] A. F. Nematollahi, A. Rahiminejad, B. Vahidi, A Novel Meta-Heuristic Optimization Method Based on Golden Ratio in Nature, Soft Computing 24 (2020) 1117–1151.

[207] S. Abdulhameed, T. A. Rashid, Child Drawing Development Optimization Algorithm Based on Child's Cognitive Development, Arabian Journal for Science and Engineering 47 (2022) 1337–1351.

[208] A. H. Hosseinian, V. Baradaran, P-GWO and MOFA: Two New Algorithms for the MSRCPSP with the Deterioration Effect and Financial Constraints (Case Study of a Gas

Treating Company), Applied Intelligence 50 (2020) 2151–2176.

[209] E. E. Zachariadis, C. T. Kiranoudis, An Open Vehicle Routing Problem Metaheuristic for Examining Wide Solution Neighborhoods, Computers & Operations Research 37 (2010) 712–723.

[210] L. Wei, Z. Zhang, A. Lim, An Adaptive Variable Neighborhood Search for a Heterogeneous Fleet Vehicle Routing Problem with Three-Dimensional Loading Constraints, IEEE Computational Intelligence Magazine 9 (2014) 18–30.

[211] Z. Zhang, L. Wei, A. Lim, An Evolutionary Local Search for the Capacitated Vehicle Routing Problem Minimizing Fuel Consumption under Three-Dimensional Loading Constraints, Transportation Research Part B: Methodological 82 (2015) 20–35.

[212] P. K. Visscher, T. D. Seeley, Foraging Strategy of Honeybee Colonies in a Temperate Deciduous Forest, Ecology 63 (1982) 1790–1801.

[213] K. Shackleton, N. J. Balfour, H. Al Toufailia, E. James, F. L. Ratnieks, Honey bee Waggle Dances Facilitate Shorter Foraging Distances and Increased Foraging Aggregation, Animal Behaviour 198 (2023) 11–19.

Appendix A

Input Data

Table A.1: Gear pump A. Properties and disassembly requirements for all components.

Item	**Name**	**Material**	**Volume** (mm^3)	**Weight** (g.)	**Disassembly point** **X**	**Y**	**Z**	**Disassembly tool**	$t_b(x_i)$ (s)
1	Bolt A	Steel	1.006,5	7,9	49,4	105,5	-12,6	Spanner-I	3
2	Bolt B	Steel	1.006,5	7,9	74,4	81	-12,6	Spanner-I	3
3	Bolt C	Steel	1.006,5	7,9	74,4	45	-12,6	Spanner-I	3
4	Bolt D	Steel	1.006,5	7,9	49,4	20,5	-12,6	Spanner-I	3
5	Bolt E	Steel	1.006,5	7,9	24,4	45	-12,6	Spanner-I	3
6	Bolt F	Steel	1.006,5	7,9	24,4	81	-12,6	Spanner-I	3
7	Cover	Steel	68.552,5	538,1	49,4	63	-20,6	Gripper-II	4
8	Gasket	Rubber	4.450,4	4,2	49,4	105,5	1,4	Gripper-I	3
9	Gear A	Steel	15.215,5	119,4	49,4	81	3,4	Gripper-I	6
10	Gear B	Steel	15.215,5	119,4	49,4	45	3,4	Gripper-I	6
11	Driven Shaft A	Steel	5.207,0	40,9	49,4	81	-7,6	Gripper-I	4
12	Base	Steel	195.539,3	1535,0	49,4	81	49,4	Gripper-II	8
13	Driven Shaft B	Steel	18.267,2	143,4	49,4	45	152,4	Gripper-I	4
14	Packing Gland	Steel	2.709,0	21,3	49,4	45	91,4	Gripper-I	2
15	Gland Nut	Steel	12.046,9	94,6	49,4	45	96,4	Spanner-II	3

Table A.2: Gear pump B. Properties and disassembly requirements for all components.

Item	**Name**	**Material**	**Volume** (mm^3)	**Weight** (g.)	**Disassembly point**			**Disassembly tool**	$t_b(x_i)$ (s)
					X	**Y**	**Z**		
1	Bolt A	Steel	1,243.1	9.8	59.1	114	-48.4	Spanner-I	4
2	Bolt B	Steel	1,243.1	9.8	90.3	89	-48.4	Spanner-I	4
3	Bolt C	Steel	1,243.1	9.8	90.3	33	-48.4	Spanner-I	4
4	Bolt D	Steel	1,243.1	9.8	59.1	8	-48.4	Spanner-I	4
5	Bolt E	Steel	1,243.1	9.8	27.9	33	-48.4	Spanner-I	4
6	Bolt F	Steel	1,243.1	9.8	27.9	89	-48.4	Spanner-I	4
7	Cover	Steel	95,973.5	753.4	59.1	82	-64.6	Gripper-II	5
8	Gasket	Rubber	5,496.3	5.2	59.1	114	-31.4	Gripper-I	4
9	Gear A	Steel	21,301.7	167.2	59.1	82	-30.9	Gripper-I	6
10	Gear B	Steel	21,301.7	167.2	59.1	40	-30.9	Gripper-I	6
11	Shaft A	Steel	6,430.7	50.5	59.1	40	-48.9	Gripper-I	4
12	Base	Steel	273,755.0	2149.0	59.1	114	7.1	Gripper-II	4
13	Shaft B	Steel	22,560.0	177.1	59.1	82	136.1	Gripper-I	8
14	Gland A	PTFE	3,243.6	7.1	59.1	94.8	34.1	Gripper-I	3
15	Gland B	PTFE	3,243.6	7.1	59.1	94.8	41.1	Gripper-I	3
16	Gland C	PTFE	3,243.6	7.1	59.1	94.8	48.1	Gripper-I	3
17	Gland D	PTFE	3,243.6	7.1	59.1	94.8	55.1	Gripper-I	3
18	Gland E	Steel	14,456.3	113.5	59.1	82	79.1	Gripper-I	3
19	Bolt stud A	Steel	998.1	7.8	35.1	82	89.1	Spanner-II	3
20	Bolt stud B	Steel	998.1	7.8	83.1	82	89.1	Spanner-II	3
21	Nut A	Steel	289.5	2.3	35.1	82	84.1	Spanner-III	4
22	Nut B	Steel	289.5	2.3	83.1	82	84.1	Spanner-III	4
23	Nut C	Steel	289.5	2.3	35.1	82	87.1	Spanner-III	4
24	Nut D	Steel	289.5	2.3	83.1	82	87.1	Spanner-III	4

Table A.3: Gear pump A. (PD Matrix)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	M
1	0.0	55.0	85.5	121.0	85.5	55.0	70.5	Inf	Inf	Inf	Inf	164.4	258.6	209.9	213.3	202.6
2	55.0	0.0	56.0	85.5	97.6	86.0	59.8	Inf	Inf	Inf	Inf	195.5	285.5	238.3	241.5	191.9
3	85.5	56.0	0.0	55.0	86.0	97.6	59.8	Inf	Inf	Inf	Inf	231.5	321.5	274.3	277.5	189.3
4	121.0	85.5	55.0	0.0	55.0	85.5	70.5	Inf	Inf	Inf	Inf	249.4	343.6	294.9	298.3	196.8
5	85.5	97.6	86.0	55.0	0.0	56.0	59.8	Inf	Inf	Inf	Inf	231.5	321.5	274.3	277.5	189.3
6	55.0	86.0	97.6	85.5	56.0	0.0	59.8	Inf	Inf	Inf	Inf	195.5	285.5	238.3	241.5	191.9
7	70.5	59.8	59.8	70.5	59.8	59.8	0.0	72.5	62.0	62.0	51.0	209.9	311.5	260.4	273.8	187.3
8	Inf	Inf	Inf	Inf	Inf	Inf	72.5	0.0	64.5	100.5	53.5	145.3	244.6	195.9	199.3	215.6
9	Inf	Inf	Inf	Inf	Inf	Inf	62.0	64.5	0.0	78.0	11.0	171.7	271.0	222.3	225.7	212.8
10	Inf	Inf	Inf	Inf	Inf	Inf	62.0	100.5	78.0	0.0	67.0	207.7	307.0	258.3	261.7	210.4
11	Inf	Inf	Inf	Inf	Inf	Inf	51.0	53.5	11.0	67.0	0.0	178.9	278.1	229.4	232.8	201.8
12	164.4	195.5	231.5	249.4	231.5	195.5	209.9	145.3	171.7	207.7	178.9	0.0	144.2	84.9	96.1	171.9
13	258.6	285.5	321.5	343.6	321.5	285.5	311.5	244.6	271.0	307.0	278.1	144.2	0.0	61.0	56.0	423.1
14	209.9	238.3	274.3	294.9	274.3	238.3	260.4	195.9	222.3	258.3	229.4	84.9	61.0	0.0	Inf	362.1
15	213.3	241.5	277.5	298.3	277.5	241.5	273.8	199.3	225.7	261.7	232.8	96.1	56.0	Inf	0.0	316.5
M	202.6	191.9	189.3	196.8	189.3	191.9	187.3	215.6	212.8	210.4	201.8	171.9	423.1	362.1	316.5	0.0

Table A.4: Gear pump B (PD Matrix)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	M
1	0.0	76.2	132.2	126.0	132.2	76.2	68.2	Inf	Inf	Inf	Inf	75.5	236.5	121.7	128.7	135.7	142.7	179.5	213.5	213.5	208.5	208.5	211.5	211.5	244.1
2	76.2	0.0	76.0	132.2	138.4	82.4	74.4	Inf	Inf	Inf	Inf	131.7	242.7	139.5	146.5	153.5	160.5	185.7	219.7	171.7	214.7	166.7	217.7	169.7	219.0
3	132.2	76.0	0.0	76.2	82.4	138.4	116.4	Inf	Inf	Inf	Inf	187.7	284.7	195.5	202.5	209.5	216.5	227.7	261.7	213.7	256.7	208.7	259.7	211.7	240.7
4	126.0	132.2	76.2	0.0	76.2	132.2	110.2	Inf	Inf	Inf	Inf	181.5	278.5	189.3	196.3	203.3	210.3	221.5	255.5	255.5	250.5	250.5	253.5	253.5	280.1
5	132.2	138.4	82.4	76.2	0.0	76.0	116.4	Inf	Inf	Inf	Inf	187.7	284.7	195.5	202.5	209.5	216.5	227.7	213.7	261.7	208.7	256.7	211.7	259.7	296.6
6	76.2	82.4	138.4	132.2	76.0	0.0	74.4	Inf	Inf	Inf	Inf	131.7	242.7	139.5	146.5	153.5	160.5	185.7	171.7	219.7	166.7	214.7	169.7	217.7	279.3
7	Inf	Inf	Inf	Inf	Inf	Inf	0.0	85.2	Inf	Inf	Inf	123.7	220.7	131.5	138.5	145.5	152.5	163.7	197.7	197.7	192.7	192.7	195.7	195.7	250.3
8	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0	52.5	94.5	Inf	58.5	219.5	104.7	111.7	118.7	125.7	162.5	196.5	196.5	191.5	191.5	194.5	194.5	245.8
9	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0	62.0	Inf	90.0	187.0	97.8	104.8	111.8	118.8	130.0	164.0	164.0	159.0	159.0	162	162	252.6
10	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	62.0	0.0	38.0	132.0	229.0	139.8	146.8	153.8	160.8	172.0	206.0	206.0	201.0	201.0	204	204	267.0
11	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	80.0	Inf	0.0	150.0	247.0	157.8	164.8	171.8	178.8	190.0	224.0	224.0	219.0	219.0	222	222	265.3
12	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	253.9
13	236.5	242.7	284.7	278.5	284.7	242.7	220.7	219.5	187.0	229.0	247.0	181.0	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	320.8
14	121.7	139.5	195.5	189.3	195.5	139.5	131.5	104.7	97.8	139.8	157.8	66.2	134.8	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	266.1
15	128.7	146.5	202.5	196.3	202.5	146.5	138.5	111.7	104.8	146.8	164.8	73.2	Inf	27.0	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	268.8
16	135.7	153.5	209.5	203.3	209.5	153.5	145.5	118.7	111.8	153.8	171.8	80.2	Inf	Inf	27.0	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	271.6
17	142.7	160.5	216.5	210.3	216.5	160.5	152.5	125.7	118.8	160.8	178.8	87.2	Inf	Inf	Inf	27.0	0.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	274.6
18	179.5	185.7	227.7	221.5	227.7	185.7	163.7	162.5	130.0	172.0	190.0	124.0	Inf	Inf	Inf	Inf	56.8	0.0	Inf	Inf	Inf	Inf	Inf	Inf	288.6
19	213.5	219.7	261.7	255.5	213.7	171.7	197.7	196.5	164.0	206.0	224.0	158.0	Inf	Inf	Inf	Inf	Inf	54.0	0.0	68.0	Inf	73.0	Inf	70	313.7
20	213.5	171.7	213.7	255.5	261.7	219.7	197.7	196.5	164.0	206.0	224.0	158.0	Inf	Inf	Inf	Inf	Inf	54.0	68.0	0.0	73.0	Inf	70	Inf	274.4
21	208.5	214.7	256.7	250.5	208.7	166.7	192.7	191.5	159.0	201.0	219.0	153.0	Inf	Inf	Inf	Inf	Inf	Inf	25.0	73.0	0.0	68.0	Inf	71	311.3
22	208.5	166.7	208.7	250.5	256.7	214.7	192.7	191.5	159.0	201.0	219.0	153.0	Inf	Inf	Inf	Inf	Inf	Inf	73.0	25.0	68.0	0.0	71	Inf	271.6
23	211.5	217.7	259.7	253.5	211.7	169.7	195.7	194.5	162.0	204.0	222.0	156.0	Inf	Inf	Inf	Inf	Inf	Inf	Inf	70.0	23.0	71.0	0	68	312.7
24	211.5	169.7	211.7	253.5	259.7	217.7	195.7	194.5	162.0	204.0	222.0	156.0	Inf	Inf	Inf	Inf	Inf	Inf	70.0	Inf	71.0	23.0	68	0	273.3
M	244.1	219.0	240.7	280.1	296.6	279.3	250.3	245.8	252.6	267.0	265.3	253.9	320.8	266.1	268.8	271.6	274.6	288.6	313.7	274.4	311.3	271.6	312.7	273.3	0

Table A.5: Input data for f_1. Gear pump A.

Item	RP_i	RC_i	CD_i	$rc_{i,1}$	$rc_{i,2}$	$oh_{i,1}$	$oh_{i,2}$	$oh_{i,3}$	$oh_{i,4}$	$dp_{i,1}$	$dp_{i,2}$	$dp_{i,3}$	$dp_{i,4}$
1	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
2	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
3	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
4	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
5	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
6	0.35	0.0024	0	0.1	0.3	0.0022	0.0056	0.0022	0.0011	0.09	0.13	0.10	0.07
7	8.50	0.1614	0	1.2	4	0.1530	0.3826	0.1530	0.0765	0.12	0.17	0.14	0.09
8	0.00	0.0000	0.3	0	0	0.0012	0.0030	0.0012	0.0006	0.09	0.13	0.10	0.07
9	12.70	0.0358	0	1	7.5	0.0340	0.0849	0.0340	0.0170	0.17	0.26	0.21	0.14
10	12.70	0.0358	0	1	7.5	0.0340	0.0849	0.0340	0.0170	0.17	0.26	0.21	0.14
11	3.50	0.0123	0	0.3	2	0.0116	0.0291	0.0116	0.0058	0.12	0.17	0.14	0.09
12	37.00	0.4605	0	4.5	8.3	0.4366	1.0914	0.4366	0.2183	0.23	0.35	0.28	0.18
13	6.30	0.0430	0	0.7	3.2	0.0408	0.1020	0.0408	0.0204	0.12	0.17	0.14	0.09
14	2.50	0.0064	0	0.5	1.2	0.0060	0.0151	0.0060	0.0030	0.06	0.09	0.07	0.05
15	3.00	0.0284	0	0.7	1.6	0.0269	0.0672	0.0269	0.0134	0.09	0.13	0.10	0.07

Table A.6: Input data for f_1. Gear pump B.

Item	RP_i	RC_i	CD_i	$rc_{i,1}$	$rc_{i,2}$	$oh_{i,1}$	$oh_{i,2}$	$oh_{i,3}$	$oh_{i,4}$	$dp_{i,1}$	$dp_{i,2}$	$dp_{i,3}$	$dp_{i,4}$
1	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
2	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
3	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
4	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
5	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
6	0.43	0.0049	0	0.1	0.3	0.0024	0.0060	0.0024	0.0012	0.12	0.17	0.14	0.09
7	11.90	0.3767	0	1.2	2.5	0.1854	0.4634	0.1854	0.0927	0.14	0.22	0.17	0.12
8	0.00	0.0026	0.2	0	0	0.0013	0.0032	0.0013	0.0006	0.12	0.17	0.14	0.09
9	15.78	0.0836	0	0.6	3	0.0411	0.1029	0.0411	0.0206	0.17	0.26	0.21	0.14
10	15.78	0.0836	0	0.6	3	0.0411	0.1029	0.0411	0.0206	0.17	0.26	0.21	0.14
11	4.32	0.0252	0	0.3	0.9	0.0124	0.0311	0.0124	0.0062	0.12	0.17	0.14	0.09
12	37.80	1.0745	0	1.5	3.5	0.5287	1.3218	0.5287	0.2644	0.12	0.17	0.14	0.09
13	7.78	0.0885	0	0.7	2	0.0436	0.1089	0.0436	0.0218	0.23	0.35	0.28	0.18
14	0.00	0.0036	0.15	0	0	0.0018	0.0044	0.0018	0.0009	0.09	0.13	0.10	0.07
15	0.00	0.0036	0.15	0	0	0.0018	0.0044	0.0018	0.0009	0.09	0.13	0.10	0.07
16	0.00	0.0036	0.15	0	0	0.0018	0.0044	0.0018	0.0009	0.09	0.13	0.10	0.07
17	0.00	0.0036	0.15	0	0	0.0018	0.0044	0.0018	0.0009	0.09	0.13	0.10	0.07
18	3.60	0.0567	0	0.2	0.7	0.0279	0.0698	0.0279	0.0140	0.09	0.13	0.10	0.07
19	0.35	0.0039	0	0.1	0.3	0.0019	0.0048	0.0019	0.0010	0.09	0.13	0.10	0.07
20	0.35	0.0039	0	0.1	0.3	0.0019	0.0048	0.0019	0.0010	0.09	0.13	0.10	0.07
21	0.20	0.0011	0	0.1	0.3	0.0006	0.0014	0.0006	0.0003	0.12	0.17	0.14	0.09
22	0.20	0.0011	0	0.1	0.3	0.0006	0.0014	0.0006	0.0003	0.12	0.17	0.14	0.09
23	0.20	0.0011	0	0.1	0.3	0.0006	0.0014	0.0006	0.0003	0.12	0.17	0.14	0.09
24	0.20	0.0011	0	0.1	0.3	0.0006	0.0014	0.0006	0.0003	0.12	0.17	0.14	0.09

Table A.7: Input data for f_2. Gear pump A.

Item	$gr_{i,1}$	$gr_{i,2}$	$gd_{1,i}$	$gd_{3,i}$	$gc_{i,1}$	$gc_{i,2}$	$gc_{i,3}$	$gc_{i,4}$
1	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
2	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
3	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
4	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
5	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
6	0.115	0.115	0.042	0.069	0.006	0.035	0.012	0.023
7	1.302	1.302	0.056	0.069	0.065	0.391	0.130	0.260
8	0.014	0.014	0.042	0.069	0.001	0.004	0.001	0.003
9	0.850	0.850	0.083	0.069	0.043	0.255	0.085	0.170
10	0.850	0.850	0.083	0.069	0.043	0.255	0.085	0.170
11	0.138	0.138	0.056	0.069	0.007	0.041	0.014	0.028
12	2.480	2.480	0.111	0.069	0.124	0.744	0.248	0.496
13	0.592	0.592	0.056	0.069	0.030	0.178	0.059	0.118
14	0.234	0.234	0.028	0.069	0.012	0.070	0.023	0.047
15	0.284	0.284	0.042	0.069	0.014	0.085	0.028	0.057

Note: $gd_{2,i}$, $gd_{4,i}$ and $gd_{5,i}$ values are obtained from the *GD* matrix

Table A.8: GD matrix for f_2. Gear pump A.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	M
1	0.000	0.032	0.050	0.071	0.050	0.032	0.263	Inf	Inf	Inf	Inf	0.254	0.401	0.366	0.339	0.017
2	0.032	0.000	0.033	0.050	0.057	0.050	0.257	Inf	Inf	Inf	Inf	0.248	0.395	0.359	0.333	0.016
3	0.050	0.033	0.000	0.032	0.050	0.057	0.256	Inf	Inf	Inf	Inf	0.247	0.394	0.358	0.331	0.016
4	0.071	0.050	0.032	0.000	0.032	0.050	0.260	Inf	Inf	Inf	Inf	0.251	0.398	0.362	0.336	0.016
5	0.050	0.057	0.050	0.032	0.000	0.033	0.256	Inf	Inf	Inf	Inf	0.247	0.394	0.358	0.331	0.016
6	0.032	0.050	0.057	0.050	0.033	0.000	0.257	Inf	Inf	Inf	Inf	0.248	0.395	0.359	0.333	0.016
7	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.271	Inf	Inf	Inf	0.123	0.392	0.357	0.330	0.016
8	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.038	0.059	Inf	0.262	0.143	0.115	0.347	0.018
9	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.046	0.006	0.260	0.159	0.130	0.345	0.018
10	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.046	0.000	0.039	0.259	0.180	0.151	0.343	0.017
11	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.039	0.000	0.254	0.163	0.134	0.338	0.017
12	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	Inf	Inf	Inf	0.014
13	0.401	0.395	0.394	0.398	0.394	0.395	0.392	0.143	0.159	0.180	0.163	0.383	0.000	Inf	Inf	0.035
14	0.366	0.359	0.358	0.362	0.358	0.359	0.357	0.115	0.130	0.151	0.134	0.348	0.036	0.000	Inf	0.030
15	0.339	0.333	0.331	0.336	0.331	0.333	0.330	0.347	0.345	0.343	0.338	0.321	Inf	0.432	0.000	0.026
M	0.017	0.016	0.016	0.016	0.016	0.016	0.016	0.018	0.018	0.017	0.017	0.014	0.035	0.030	0.026	0.000

Table A.9: Input data for f_2. Gear pump B.

Item	$gr_{i,1}$	$gr_{i,2}$	$gd_{1,i}$	$gd_{3,i}$	$gc_{i,1}$	$gc_{i,2}$	$gc_{i,3}$	$gc_{i,4}$
1	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
2	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
3	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
4	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
5	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
6	0.896	0.896	0.056	0.056	0.045	0.269	0.090	0.179
7	11.152	11.152	0.069	0.056	0.558	3.346	1.115	2.230
8	0.112	0.112	0.056	0.056	0.006	0.034	0.011	0.022
9	6.791	6.791	0.083	0.056	0.340	2.037	0.679	1.358
10	6.791	6.791	0.083	0.056	0.340	2.037	0.679	1.358
11	1.111	1.111	0.056	0.056	0.056	0.333	0.111	0.222
12	21.897	21.897	0.056	0.056	1.095	6.569	2.190	4.379
13	4.733	4.733	0.111	0.056	0.237	1.420	0.473	0.947
14	0.122	0.122	0.042	0.056	0.006	0.037	0.012	0.024
15	0.122	0.122	0.042	0.056	0.006	0.037	0.012	0.024
16	0.122	0.122	0.042	0.056	0.006	0.037	0.012	0.024
17	0.122	0.122	0.042	0.056	0.006	0.037	0.012	0.024
18	4.819	4.819	0.042	0.056	0.241	1.446	0.482	0.964
19	0.885	0.885	0.042	0.056	0.044	0.266	0.089	0.177
20	0.885	0.885	0.042	0.056	0.044	0.266	0.089	0.177
21	0.223	0.223	0.056	0.056	0.011	0.067	0.022	0.045
22	0.223	0.223	0.056	0.056	0.011	0.067	0.022	0.045
23	0.223	0.223	0.056	0.056	0.011	0.067	0.022	0.045
24	0.223	0.223	0.056	0.056	0.011	0.067	0.022	0.045

Note: $gd_{2,i}$, $gd_{4,i}$ and $gd_{5,i}$ values are obtained from the *GD* matrix

Table A.10: GD matrix for f_2. Gear pump B.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	M
1	0.000	0.044	0.077	0.073	0.077	0.044	0.039	Inf	Inf	Inf	Inf	0.044	0.137	0.070	0.074	0.079	0.083	0.104	0.124	0.124	0.121	0.121	0.122	0.122	0.141
2	0.044	0.000	0.044	0.077	0.080	0.048	0.043	Inf	Inf	Inf	Inf	0.076	0.140	0.081	0.085	0.089	0.093	0.107	0.127	0.099	0.124	0.096	0.126	0.098	0.127
3	0.077	0.044	0.000	0.044	0.048	0.080	0.067	Inf	Inf	Inf	Inf	0.109	0.165	0.113	0.117	0.121	0.125	0.132	0.151	0.124	0.149	0.121	0.150	0.123	0.139
4	0.073	0.077	0.044	0.000	0.044	0.077	0.064	Inf	Inf	Inf	Inf	0.105	0.161	0.110	0.114	0.118	0.122	0.128	0.148	0.148	0.145	0.145	0.147	0.147	0.162
5	0.077	0.080	0.048	0.044	0.000	0.044	0.067	Inf	Inf	Inf	Inf	0.109	0.165	0.113	0.117	0.121	0.125	0.132	0.124	0.151	0.121	0.149	0.123	0.150	0.172
6	0.044	0.048	0.080	0.077	0.044	0.000	0.043	Inf	Inf	Inf	Inf	0.076	0.140	0.081	0.085	0.089	0.093	0.107	0.099	0.127	0.096	0.124	0.098	0.126	0.162
7	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.049	Inf	Inf	Inf	0.072	0.128	0.076	0.080	0.084	0.088	0.095	0.114	0.114	0.112	0.112	0.113	0.113	0.145
8	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.030	0.055	Inf	0.034	0.127	0.061	0.065	0.069	0.073	0.094	0.114	0.114	0.111	0.111	0.113	0.113	0.142
9	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	0.036	Inf	0.052	0.108	0.057	0.061	0.065	0.069	0.075	0.095	0.095	0.092	0.092	0.094	0.094	0.146
10	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.036	0.000	0.022	0.076	0.133	0.081	0.085	0.089	0.093	0.100	0.119	0.119	0.116	0.116	0.118	0.118	0.154
11	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.046	Inf	0.000	0.087	0.143	0.091	0.095	0.099	0.103	0.110	0.130	0.130	0.127	0.127	0.128	0.128	0.154
12	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.147
13	0.137	0.140	0.165	0.161	0.165	0.140	0.128	0.127	0.108	0.133	0.143	0.105	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.186
14	0.070	0.081	0.113	0.110	0.113	0.081	0.076	0.061	0.057	0.081	0.091	0.038	0.078	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.154
15	0.074	0.085	0.117	0.114	0.117	0.085	0.080	0.065	0.061	0.085	0.095	0.042	Inf	0.016	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.156
16	0.079	0.089	0.121	0.118	0.121	0.089	0.084	0.069	0.065	0.089	0.099	0.046	Inf	Inf	0.016	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.157
17	0.083	0.093	0.125	0.122	0.125	0.093	0.088	0.073	0.069	0.093	0.103	0.050	Inf	Inf	Inf	0.016	0.000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.159
18	0.104	0.107	0.132	0.128	0.132	0.107	0.095	0.094	0.075	0.100	0.110	0.072	Inf	Inf	Inf	Inf	0.033	0.000	Inf	Inf	Inf	Inf	Inf	Inf	0.167
19	0.124	0.127	0.151	0.148	0.124	0.099	0.114	0.114	0.095	0.119	0.130	0.091	Inf	Inf	Inf	Inf	Inf	0.031	0.000	0.039	Inf	0.042	Inf	0.041	0.182
20	0.124	0.099	0.124	0.148	0.151	0.127	0.114	0.114	0.095	0.119	0.130	0.091	Inf	Inf	Inf	Inf	Inf	0.031	0.039	0.000	0.042	Inf	0.041	Inf	0.159
21	0.121	0.124	0.149	0.145	0.121	0.096	0.112	0.111	0.092	0.116	0.127	0.089	Inf	Inf	Inf	Inf	Inf	Inf	0.014	0.042	0.000	0.039	Inf	0.041	0.180
22	0.121	0.096	0.121	0.145	0.149	0.124	0.112	0.111	0.092	0.116	0.127	0.089	Inf	Inf	Inf	Inf	Inf	Inf	0.042	0.014	0.039	0.000	0.041	Inf	0.157
23	0.122	0.126	0.150	0.147	0.123	0.098	0.113	0.113	0.094	0.118	0.128	0.090	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.041	0.013	0.041	0.000	0.039	0.181
24	0.122	0.098	0.123	0.147	0.150	0.126	0.113	0.113	0.094	0.118	0.128	0.090	Inf	Inf	Inf	Inf	Inf	Inf	0.041	Inf	0.041	0.013	0.039	0.000	0.158
M	0.141	0.127	0.139	0.162	0.172	0.162	0.145	0.142	0.146	0.154	0.154	0.147	0.186	0.154	0.156	0.157	0.159	0.167	0.182	0.159	0.180	0.157	0.181	0.158	0.000

Table A.11: Input data for f_3. Gear pump A.

Item	$er_{i,1}$	$er_{i,2}$	$ec_{i,1}$	$ec_{i,2}$	$ec_{i,3}$	$ec_{i,4}$	$ed(x_i)$
1	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
2	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
3	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
4	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
5	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
6	0,00213	0,00213	0,00021	0,00064	0,00111	0,00221	0,00262
7	0,11637	0,11637	0,01164	0,03491	0,07534	0,15068	0,00349
8	0,00030	0,00030	0,00003	0,00009	0,00001	0,00355	0,00262
9	0,02490	0,02490	0,00249	0,00747	0,01672	0,03344	0,00524
10	0,02490	0,02490	0,00249	0,00747	0,01672	0,03344	0,00524
11	0,00779	0,00779	0,00078	0,00234	0,00572	0,01145	0,00349
12	0,43960	0,43960	0,04396	0,13188	0,21490	0,42980	0,00000
13	0,02717	0,02717	0,00272	0,00815	0,02008	0,04015	0,00349
14	0,00508	0,00508	0,00051	0,00152	0,00298	0,00595	0,00175
15	0,02109	0,02109	0,00211	0,00633	0,01324	0,02648	0,00262

Note: $ed(x_i, x_{i+1})$ values are obtained from the *ED* matrix

Table A.12: ED matrix for f_3. Gear pump A.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	M
1	0.00000	0.00184	0.00286	0.00405	0.00286	0.00184	0.01505	Inf	Inf	Inf	Inf	0.01453	0.02294	0.02090	0.01937	0.00096
2	0.00184	0.00000	0.00187	0.00286	0.00326	0.00288	0.01469	Inf	Inf	Inf	Inf	0.01417	0.02258	0.02054	0.01901	0.00091
3	0.00286	0.00187	0.00000	0.00184	0.00288	0.00326	0.01460	Inf	Inf	Inf	Inf	0.01409	0.02249	0.02045	0.01892	0.00090
4	0.00405	0.00286	0.00184	0.00000	0.00184	0.00286	0.01485	Inf	Inf	Inf	Inf	0.01434	0.02274	0.02070	0.01918	0.00093
5	0.00286	0.00326	0.00288	0.00184	0.00000	0.00187	0.01460	Inf	Inf	Inf	Inf	0.01409	0.02249	0.02045	0.01892	0.00090
6	0.00184	0.00288	0.00326	0.00286	0.00187	0.00000	0.01469	Inf	Inf	Inf	Inf	0.01417	0.02258	0.02054	0.01901	0.00091
7	Inf	Inf	Inf	Inf	Inf	Inf	0.00000	0.01548	Inf	Inf	Inf	0.00702	0.02242	0.02038	0.01886	0.00089
8	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.00000	0.00216	0.00336	Inf	0.01497	0.00818	0.00655	0.01980	0.00102
9	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.00000	0.00261	0.00037	0.01487	0.00906	0.00744	0.01971	0.00101
10	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.00261	0.00000	0.00224	0.01479	0.01027	0.00864	0.01963	0.00100
11	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.00224	0.00000	0.01451	0.00930	0.00767	0.01934	0.00095
12	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.00000	Inf	Inf	Inf	0.00081
13	0.02294	0.02258	0.02249	0.02274	0.02249	0.02258	0.02242	0.00818	0.00906	0.01027	0.00930	0.02191	0.00000	Inf	Inf	0.00200
14	0.02090	0.02054	0.02045	0.02070	0.02045	0.02054	0.02038	0.00655	0.00744	0.00864	0.00767	0.01987	0.00204	0.00000	Inf	0.00171
15	0.01937	0.01901	0.01892	0.01918	0.01892	0.01901	0.01886	0.01980	0.01971	0.01963	0.01934	0.01834	Inf	0.02470	0.00000	0.00150
M	0.00096	0.00091	0.00090	0.00093	0.00090	0.00091	0.00089	0.00102	0.00101	0.00100	0.00095	0.00081	0.00200	0.00171	0.00150	0.00000

Table A.13: Input data for f_3 objective. Gear pump B.

Item	$er_{i,1}$	$er_{i,2}$	$ec_{i,1}$	$ec_{i,2}$	$ec_{i,3}$	$ec_{i,4}$	$ed(x_i)$
1	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
2	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
3	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
4	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
5	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
6	0,00223	0,00223	0,00022	0,00067	0,00137	0,00273	0,00349
7	0,16291	0,16291	0,01629	0,04887	0,10547	0,21095	0,00437
8	0,00183	0,00183	0,00018	0,00055	0,00001	0,00439	0,00349
9	0,03483	0,03483	0,00348	0,01045	0,02341	0,04682	0,00524
10	0,03483	0,03483	0,00348	0,01045	0,02341	0,04682	0,00524
11	0,00962	0,00962	0,00096	0,00289	0,00707	0,01413	0,00349
12	0,54442	0,54442	0,05444	0,16332	0,30086	0,60171	0,00349
13	0,03375	0,03375	0,00337	0,01012	0,02479	0,04959	0,00699
14	0,00155	0,00155	0,00016	0,00047	0,00002	0,00599	0,00262
15	0,00155	0,00155	0,00016	0,00047	0,00002	0,00599	0,00262
16	0,00155	0,00155	0,00016	0,00047	0,00002	0,00599	0,00262
17	0,00155	0,00155	0,00016	0,00047	0,00002	0,00599	0,00262
18	0,02528	0,02528	0,00253	0,00758	0,01589	0,03177	0,00262
19	0,00162	0,00162	0,00016	0,00049	0,00110	0,00219	0,00262
20	0,00162	0,00162	0,00016	0,00049	0,00110	0,00219	0,00262
21	0,00047	0,00047	0,00005	0,00014	0,00032	0,00064	0,00349
22	0,00047	0,00047	0,00005	0,00014	0,00032	0,00064	0,00349
23	0,00047	0,00047	0,00005	0,00014	0,00032	0,00064	0,00349
24	0,00047	0,00047	0,00005	0,00014	0,00032	0,00064	0,00349

Note: $ed(x_i, x_{i+1})$ values are obtained from the *ED* matrix

Table A.14: ED matrix for f_3. Gear pump B.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	M
1	0.0000	0.0025	0.0044	0.0042	0.0044	0.0025	0.0206	Inf	Inf	Inf	Inf	0.0207	0.0229	0.0211	0.0212	0.0213	0.0214	0.0218	0.0227	0.0214	0.0226	0.0213	0.0226	0.0213	0.0082
2	0.0025	0.0000	0.0025	0.0044	0.0046	0.0028	0.0197	Inf	Inf	Inf	Inf	0.0198	0.0221	0.0202	0.0203	0.0204	0.0205	0.0210	0.0218	0.0205	0.0218	0.0204	0.0218	0.0205	0.0073
3	0.0044	0.0025	0.0000	0.0025	0.0028	0.0046	0.0204	Inf	Inf	Inf	Inf	0.0206	0.0228	0.0210	0.0211	0.0211	0.0212	0.0217	0.0226	0.0212	0.0225	0.0211	0.0225	0.0212	0.0081
4	0.0042	0.0044	0.0025	0.0000	0.0025	0.0044	0.0218	Inf	Inf	Inf	Inf	0.0219	0.0241	0.0223	0.0224	0.0225	0.0226	0.0230	0.0239	0.0226	0.0238	0.0225	0.0238	0.0225	0.0094
5	0.0044	0.0046	0.0028	0.0025	0.0000	0.0025	0.0223	Inf	Inf	Inf	Inf	0.0224	0.0247	0.0228	0.0229	0.0230	0.0231	0.0236	0.0244	0.0231	0.0243	0.0230	0.0244	0.0231	0.0099
6	0.0025	0.0028	0.0046	0.0044	0.0025	0.0000	0.0217	Inf	Inf	Inf	Inf	0.0219	0.0241	0.0223	0.0223	0.0224	0.0225	0.0230	0.0239	0.0225	0.0238	0.0224	0.0238	0.0225	0.0093
7	Inf	Inf	Inf	Inf	Inf	Inf	0.0000	0.0206	Inf	Inf	Inf	0.0041	0.0231	0.0213	0.0214	0.0215	0.0216	0.0220	0.0229	0.0216	0.0228	0.0215	0.0228	0.0215	0.0084
8	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0000	0.0018	0.0032	Inf	0.0207	0.0073	0.0035	0.0037	0.0040	0.0042	0.0054	0.0066	0.0214	0.0226	0.0213	0.0227	0.0214	0.0082
9	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0000	0.0021	Inf	0.0210	0.0063	0.0033	0.0035	0.0037	0.0040	0.0043	0.0055	0.0216	0.0229	0.0215	0.0229	0.0216	0.0084
10	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0021	0.0000	0.0013	0.0214	0.0077	0.0047	0.0049	0.0051	0.0054	0.0058	0.0069	0.0221	0.0234	0.0220	0.0234	0.0221	0.0089
11	0.0211	0.0202	0.0209	0.0223	0.0228	0.0222	0.0213	0.0037	0.0027	0.0013	0.0000	0.0214	0.0083	0.0053	0.0055	0.0057	0.0060	0.0064	0.0075	0.0221	0.0233	0.0220	0.0233	0.0220	0.0089
12	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0085
13	0.0229	0.0221	0.0228	0.0241	0.0247	0.0241	0.0231	0.0073	0.0063	0.0077	0.0083	0.0232	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0107
14	0.0211	0.0202	0.0210	0.0223	0.0228	0.0223	0.0213	0.0035	0.0033	0.0047	0.0053	0.0214	0.0045	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0089
15	0.0212	0.0203	0.0211	0.0224	0.0229	0.0223	0.0214	0.0037	0.0035	0.0049	0.0055	0.0215	Inf	0.0009	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0090
16	0.0213	0.0204	0.0211	0.0225	0.0230	0.0224	0.0215	0.0040	0.0037	0.0051	0.0057	0.0216	Inf	Inf	0.0009	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0091
17	0.0214	0.0205	0.0212	0.0226	0.0231	0.0225	0.0216	0.0042	0.0040	0.0054	0.0060	0.0217	Inf	Inf	Inf	0.0009	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0092
18	0.0218	0.0210	0.0217	0.0230	0.0236	0.0230	0.0220	0.0054	0.0043	0.0058	0.0064	0.0222	Inf	Inf	Inf	Inf	0.0019	0.0000	Inf	Inf	Inf	Inf	Inf	Inf	0.0097
19	0.0227	0.0218	0.0226	0.0239	0.0244	0.0239	0.0229	0.0227	0.0230	0.0234	0.0234	0.0230	Inf	Inf	Inf	Inf	Inf	0.0242	0.0000	0.0023	Inf	0.0236	Inf	0.0236	0.0105
20	0.0214	0.0205	0.0212	0.0226	0.0231	0.0225	0.0216	0.0214	0.0216	0.0221	0.0221	0.0217	Inf	Inf	Inf	Inf	Inf	0.0228	0.0023	0.0000	0.0236	Inf	0.0237	Inf	0.0092
21	0.0226	0.0218	0.0225	0.0238	0.0243	0.0238	0.0228	0.0226	0.0229	0.0234	0.0233	0.0229	Inf	Inf	Inf	Inf	Inf	Inf	0.0249	0.0236	0.0000	0.0023	Inf	0.0024	0.0104
22	0.0213	0.0204	0.0211	0.0225	0.0230	0.0224	0.0215	0.0213	0.0215	0.0220	0.0220	0.0216	Inf	Inf	Inf	Inf	Inf	Inf	0.0236	0.0223	0.0023	0.0000	0.0024	Inf	0.0091
23	0.0226	0.0218	0.0225	0.0238	0.0244	0.0238	0.0228	0.0227	0.0229	0.0234	0.0233	0.0230	Inf	Inf	Inf	Inf	Inf	Inf	Inf	0.0237	0.0008	0.0024	0.0000	0.0023	0.0105
24	0.0213	0.0205	0.0212	0.0225	0.0231	0.0225	0.0215	0.0214	0.0216	0.0221	0.0220	0.0216	Inf	Inf	Inf	Inf	Inf	Inf	0.0236	Inf	0.0024	0.0008	0.0023	0.0000	0.0091
M	0.0082	0.0073	0.0081	0.0094	0.0099	0.0093	0.0084	0.0082	0.0084	0.0089	0.0089	0.0085	0.0107	0.0089	0.0090	0.0091	0.0092	0.0097	0.0105	0.0092	0.0104	0.0091	0.0105	0.0091	0.0000

Appendix B

Statistical Results and Experiments

Table B.1: Chapter 3 - Normality Test Results – Goal 1 for Gear pump A and B (ARS scenario)

Tests of Normality

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Iter_pop	Statistic	df	Sig.	Statistic	df	Sig.
GearA_goal1	100_50	.103	50	.200*	.951	50	.038
	100_60	.167	50	.001	.845	50	<.001
	100_70	.073	50	.200*	.956	50	.063
	100_80	.348	50	<.001	.435	50	<.001
	200_50	.114	50	.123	.921	50	.003
	200_60	.314	50	<.001	.435	50	<.001
	200_70	.365	50	<.001	.400	50	<.001
	200_80	.298	50	<.001	.470	50	<.001
	300_50	.286	50	<.001	.638	50	<.001
	300_60	.340	50	<.001	.539	50	<.001
	300_70	.148	50	.008	.720	50	<.001
	300_80	.286	50	<.001	.552	50	<.001
	400_50	.339	50	<.001	.514	50	<.001
	400_60	.280	50	<.001	.537	50	<.001
	400_70	.319	50	<.001	.523	50	<.001
	400_80	.373	50	<.001	.423	50	<.001
	500_50	.312	50	<.001	.512	50	<.001
	500_60	.189	50	<.001	.695	50	<.001
	500_70	.093	50	.200*	.959	50	.082
	500_80	.209	50	<.001	.744	50	<.001
GearB_goal1	100_50	.156	50	.004	.892	50	<.001
	100_60	.084	50	.200*	.921	50	.003
	100_70	.136	50	.022	.919	50	.002
	100_80	.072	50	.200*	.975	50	.352
	200_50	.173	50	<.001	.870	50	<.001
	200_60	.108	50	.200*	.897	50	<.001
	200_70	.181	50	<.001	.730	50	<.001
	200_80	.217	50	<.001	.727	50	<.001
	300_50	.146	50	.010	.857	50	<.001
	300_60	.215	50	<.001	.765	50	<.001
	300_70	.071	50	.200*	.978	50	.479
	300_80	.122	50	.062	.945	50	.022
	400_50	.196	50	<.001	.854	50	<.001
	400_60	.167	50	.001	.872	50	<.001
	400_70	.145	50	.011	.912	50	.001
	400_80	.125	50	.051	.933	50	.007
	500_50	.264	50	<.001	.722	50	<.001
	500_60	.126	50	.044	.933	50	.007
	500_70	.139	50	.017	.828	50	<.001
	500_80	.135	50	.023	.876	50	<.001

*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Table B.2: Chapter 3 - Normality Test Results – Goal 2 for Gear pump A and B (ARS scenario)

Tests of Normality

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Iter_pop	Statistic	df	Sig.	Statistic	df	Sig.
GearA_goal2	100_50	.205	50	<.001	.857	50	<.001
	100_60	.178	50	<.001	.831	50	<.001
	100_70	.135	50	.023	.831	50	<.001
	100_80	.188	50	<.001	.715	50	<.001
	200_50	.236	50	<.001	.799	50	<.001
	200_60	.171	50	<.001	.795	50	<.001
	200_70	.135	50	.024	.898	50	<.001
	200_80	.155	50	.004	.927	50	.004
	300_50	.168	50	.001	.794	50	<.001
	300_60	.201	50	<.001	.830	50	<.001
	300_70	.167	50	.001	.932	50	.006
	300_80	.188	50	<.001	.792	50	<.001
	400_50	.183	50	<.001	.747	50	<.001
	400_60	.192	50	<.001	.761	50	<.001
	400_70	.215	50	<.001	.755	50	<.001
	400_80	.179	50	<.001	.849	50	<.001
	500_50	.168	50	.001	.910	50	.001
	500_60	.165	50	.002	.772	50	<.001
	500_70	.275	50	<.001	.759	50	<.001
	500_80	.230	50	<.001	.662	50	<.001
GearB_goal2	100_50	.093	50	.200*	.956	50	.063
	100_60	.117	50	.082	.969	50	.203
	100_70	.086	50	.200*	.968	50	.192
	100_80	.090	50	.200*	.975	50	.353
	200_50	.076	50	.200*	.986	50	.824
	200_60	.126	50	.045	.959	50	.081
	200_70	.078	50	.200*	.974	50	.339
	200_80	.067	50	.200*	.984	50	.714
	300_50	.088	50	.200*	.968	50	.195
	300_60	.095	50	.200*	.971	50	.259
	300_70	.108	50	.197	.956	50	.058
	300_80	.085	50	.200*	.953	50	.047
	400_50	.055	50	.200*	.988	50	.873
	400_60	.091	50	.200*	.974	50	.322
	400_70	.104	50	.200*	.927	50	.004
	400_80	.092	50	.200*	.979	50	.499
	500_50	.110	50	.181	.952	50	.041
	500_60	.108	50	.200*	.954	50	.051
	500_70	.102	50	.200*	.957	50	.065
	500_80	.112	50	.158	.953	50	.044

*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Table B.3: Chapter 3 - Normality Test Results – Goal 3 for Gear pump A and B (ARS scenario)

Tests of Normality

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Iter_pop	Statistic	df	Sig.	Statistic	df	Sig.
GearA_goal3	100_50	.364	50	<.001	.305	50	<.001
	100_60	.362	50	<.001	.679	50	<.001
	100_70	.371	50	<.001	.647	50	<.001
	100_80	.340	50	<.001	.554	50	<.001
	200_50	.381	50	<.001	.687	50	<.001
	200_60	.327	50	<.001	.553	50	<.001
	200_70	.418	50	<.001	.448	50	<.001
	200_80	.438	50	<.001	.438	50	<.001
	300_50	.350	50	<.001	.653	50	<.001
	300_60	.424	50	<.001	.475	50	<.001
	300_70	.409	50	<.001	.467	50	<.001
	300_80	.437	50	<.001	.387	50	<.001
	400_50	.356	50	<.001	.554	50	<.001
	400_60	.411	50	<.001	.472	50	<.001
	400_70	.523	50	<.001	.380	50	<.001
	400_80	.506	50	<.001	.230	50	<.001
	500_50	.403	50	<.001	.494	50	<.001
	500_60	.414	50	<.001	.233	50	<.001
	500_70	.498	50	<.001	.346	50	<.001
	500_80	.478	50	<.001	.330	50	<.001
GearB_goal3	100_50	.388	50	<.001	.617	50	<.001
	100_60	.282	50	<.001	.649	50	<.001
	100_70	.415	50	<.001	.471	50	<.001
	100_80	.409	50	<.001	.557	50	<.001
	200_50	.421	50	<.001	.609	50	<.001
	200_60	.222	50	<.001	.868	50	<.001
	200_70	.268	50	<.001	.786	50	<.001
	200_80	.269	50	<.001	.839	50	<.001
	300_50	.279	50	<.001	.823	50	<.001
	300_60	.297	50	<.001	.732	50	<.001
	300_70	.373	50	<.001	.630	50	<.001
	300_80	.324	50	<.001	.744	50	<.001
	400_50	.228	50	<.001	.853	50	<.001
	400_60	.195	50	<.001	.850	50	<.001
	400_70	.239	50	<.001	.813	50	<.001
	400_80	.398	50	<.001	.623	50	<.001
	500_50	.336	50	<.001	.779	50	<.001
	500_60	.329	50	<.001	.746	50	<.001
	500_70	.303	50	<.001	.667	50	<.001
	500_80	.352	50	<.001	.684	50	<.001

a. Lilliefors Significance Correction

Table B.4: Chapter 3 - Homogeneity Test Results for Gear pump A and B (ARS scenario)

Test of Homogeneity of Variance

		Levene Statistic	df1	df2	Sig.
GearA_goal1	Based on Mean	2.789	19	980	<.001
	Based on Median	1.228	19	980	.226
	Based on Median and with adjusted df	1.228	19	540.824	.228
	Based on trimmed mean	1.458	19	980	.093
GearA_goal2	Based on Mean	3.127	19	980	<.001
	Based on Median	2.136	19	980	.003
	Based on Median and with adjusted df	2.136	19	828.592	.003
	Based on trimmed mean	2.777	19	980	<.001
GearA_goal3	Based on Mean	6.979	19	980	<.001
	Based on Median	4.195	19	980	<.001
	Based on Median and with adjusted df	4.195	19	86.095	<.001
	Based on trimmed mean	5.060	19	980	<.001
GearB_goal1	Based on Mean	1.848	19	980	.015
	Based on Median	1.610	19	980	.047
	Based on Median and with adjusted df	1.610	19	759.820	.048
	Based on trimmed mean	1.682	19	980	.034
GearB_goal2	Based on Mean	3.045	19	980	<.001
	Based on Median	2.837	19	980	<.001
	Based on Median and with adjusted df	2.837	19	877.316	<.001
	Based on trimmed mean	3.008	19	980	<.001
GearB_goal3	Based on Mean	27.225	19	980	<.001
	Based on Median	7.233	19	980	<.001
	Based on Median and with adjusted df	7.233	19	316.830	<.001
	Based on trimmed mean	21.041	19	980	<.001

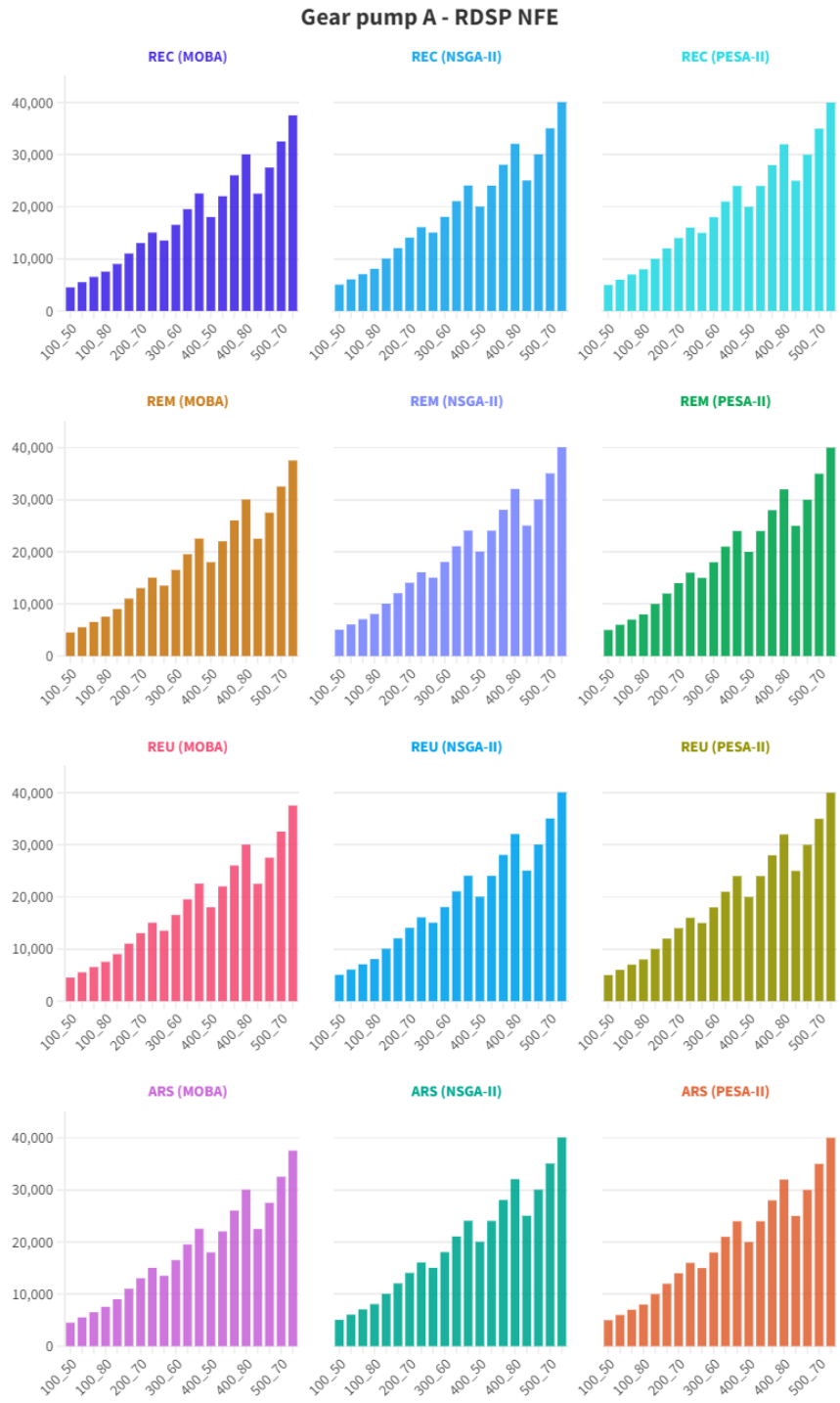


Figure B.1: Chapter 3 - NFE Gear Pump A: The lower the better (all scenario)

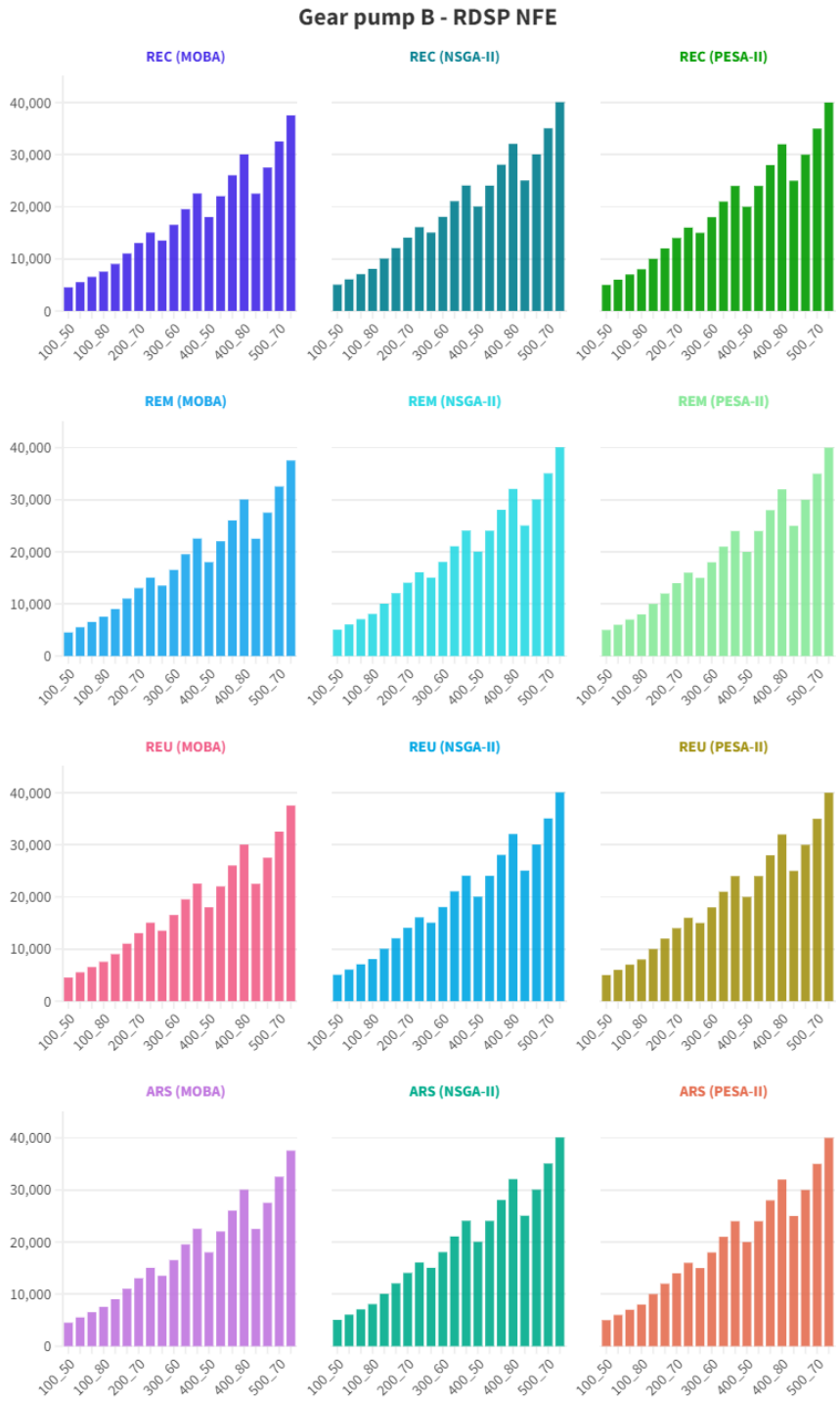


Figure B.2: Chapter 3 - NFE Gear Pump B: The lower the better (all scenario)

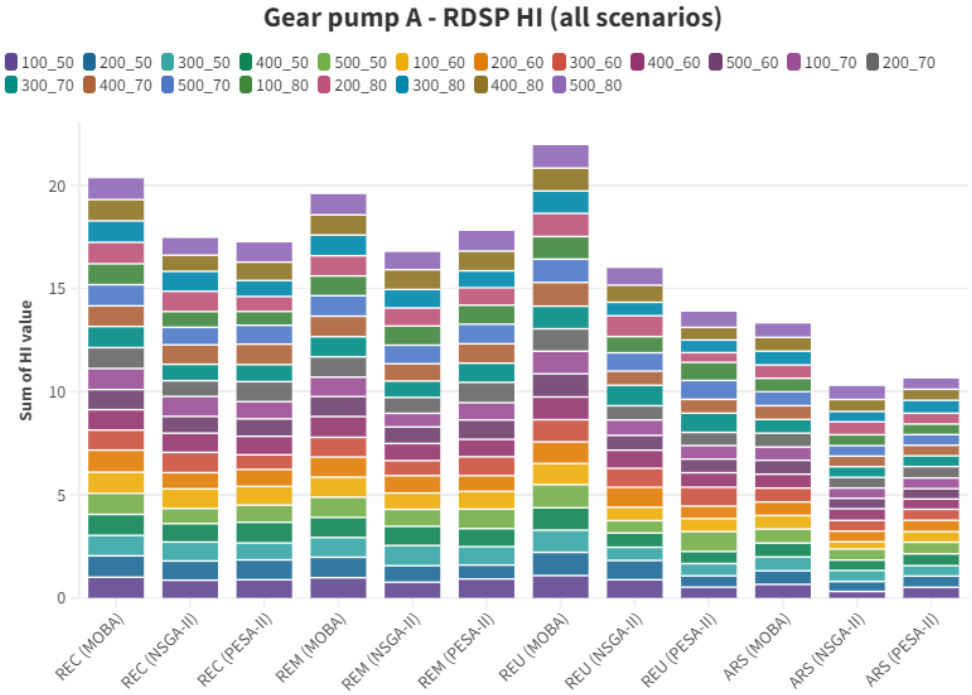


Figure B.3: Chapter 3 - Total HI Gear Pump A

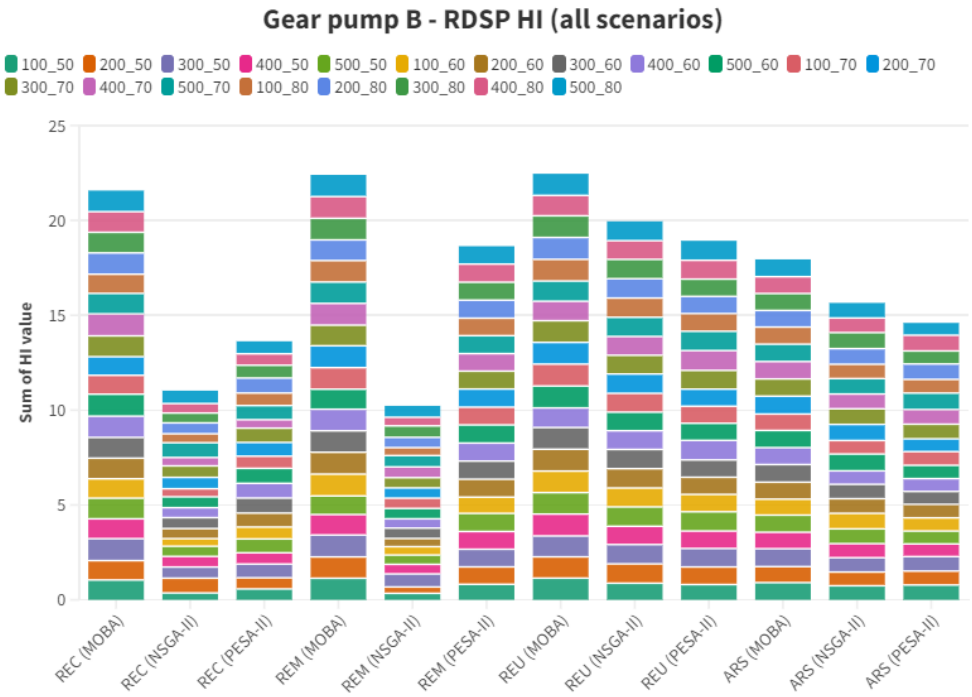


Figure B.4: Chapter 3 - Total HI Gear Pump B

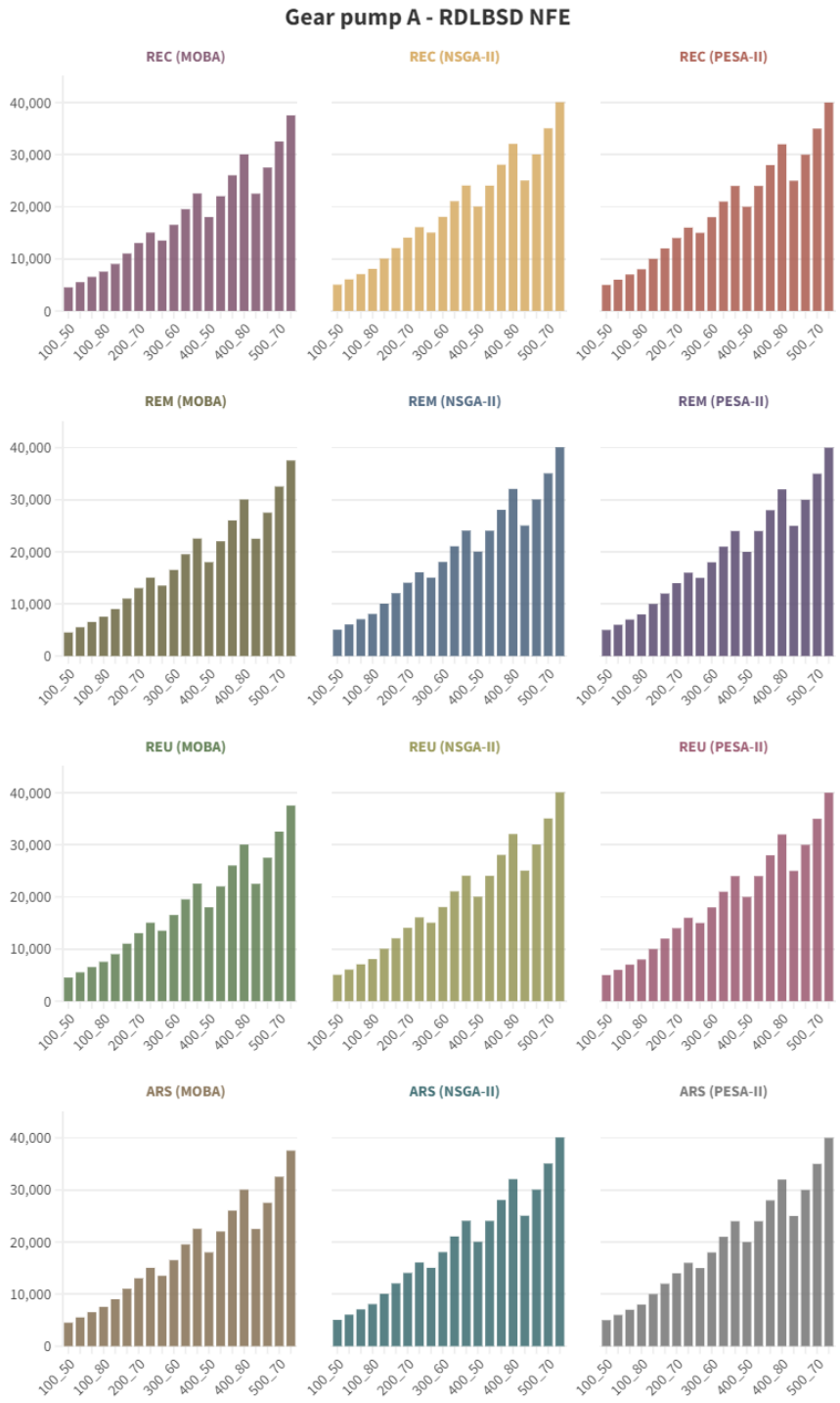


Figure B.5: Chapter 4 - NFE Gear Pump A: The lower the better (all scenario)

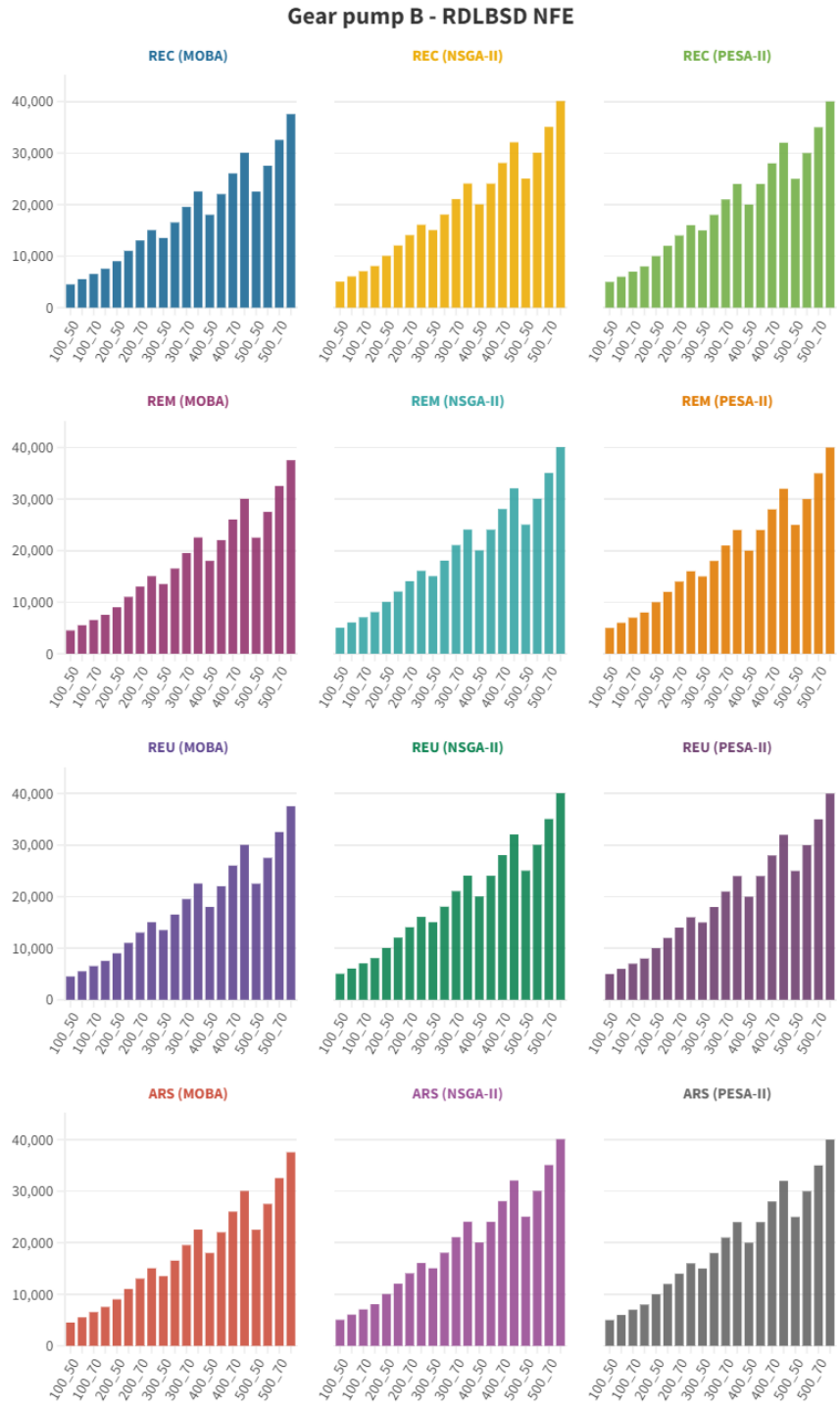


Figure B.6: Chapter 4 - NFE Gear Pump B: The lower the better (all scenario)

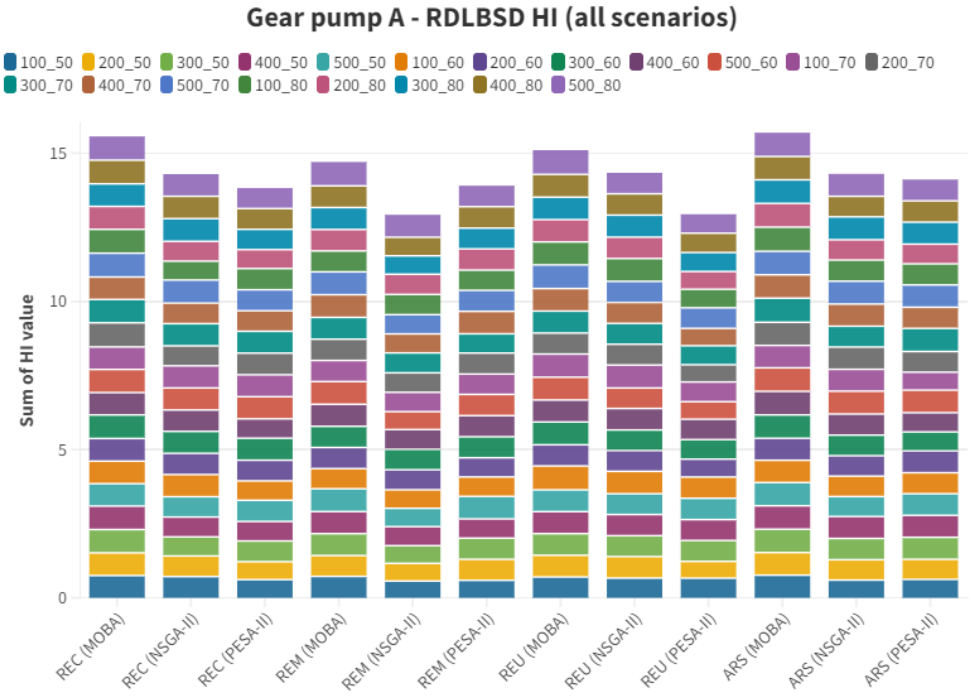


Figure B.7: Chapter 4 - Total HI Gear Pump A

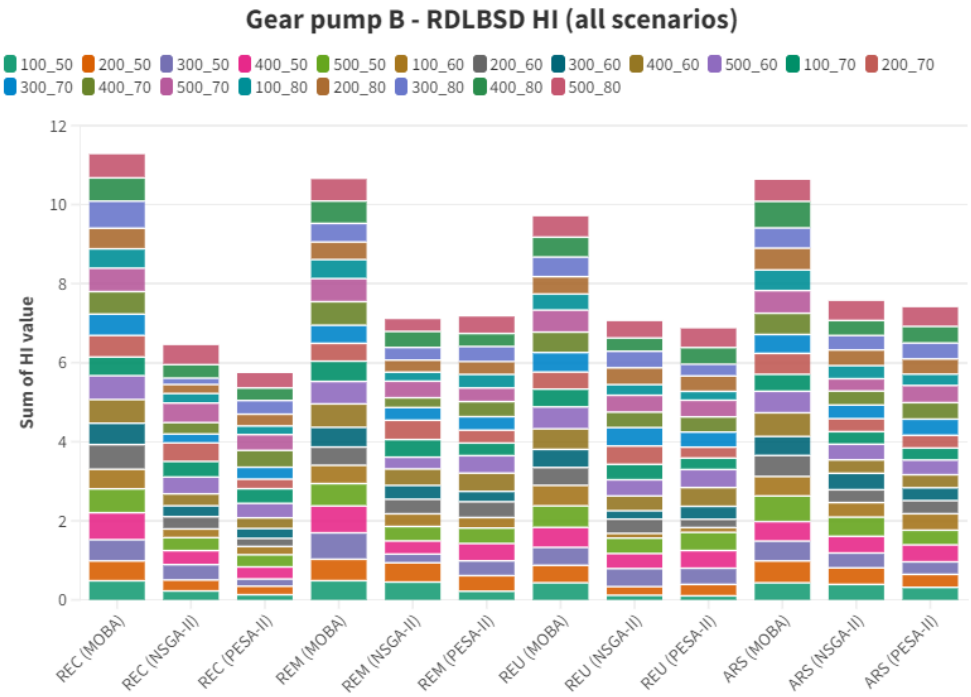


Figure B.8: Chapter 4 - Total HI Gear Pump B

Table B.5: Chapter 5 - Statistic Descriptive Gear pump A (EDBA)

Descriptive Statistics

	N	Minimum	Maximum	Mean		Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Statistic	Std. Error	Statistic	Std. Error
100_31EDBA	50	89.573075	101.162892	90.52371217	.332783501	2.353134702	3.473	.337	12.528	.662
200_31EDBA	50	89.573075	101.089717	90.03571100	.266969206	1.887757356	5.036	.337	26.691	.662
300_31EDBA	50	89.573075	101.162892	89.80653200	.231768390	1.638850002	7.071	.337	49.995	.662
400_31EDBA	50	89.573075	91.156050	89.60473450	.031659500	.223866471	7.071	.337	50.000	.662
500_31EDBA	50	89.573075	96.273242	89.73873783	.137062230	.969176323	6.577	.337	44.584	.662
100_41EDBA	50	89.573075	97.766233	90.60144183	.317767542	2.246955838	2.348	.337	4.467	.662
200_41EDBA	50	89.573075	99.880575	89.86365250	.209943483	1.484524605	6.554	.337	44.654	.662
300_41EDBA	50	89.573075	91.156050	89.63639400	.044314069	.313347786	4.841	.337	22.331	.662
400_41EDBA	50	89.573075	89.656108	89.57473567	.001660667	.011742686	7.071	.337	50.000	.662
500_41EDBA	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
100_51EDBA	50	89.573075	92.128442	89.78746183	.083408083	.589784210	2.665	.337	6.072	.662
200_51EDBA	50	89.573075	93.083650	89.70660550	.081926390	.579307060	4.865	.337	25.257	.662
300_51EDBA	50	89.573075	91.156050	89.60473450	.031659500	.223866471	7.071	.337	50.000	.662
400_51EDBA	50	89.573075	91.156050	89.60473450	.031659500	.223866471	7.071	.337	50.000	.662
500_51EDBA	50	89.573075	91.156050	89.60473450	.031659500	.223866471	7.071	.337	50.000	.662
Valid N (listwise)	50									

Table B.6: Chapter 5 - Statistic Descriptive Gear pump A (BA_F)

Descriptive Statistics

	N	Minimum	Maximum	Mean		Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Statistic	Std. Error	Statistic	Std. Error
100_31BAF	50	89.573075	96.273242	90.09603383	.213049543	1.506487763	3.401	.337	11.321	.662
200_31BAF	50	89.573075	91.239083	89.72082150	.064355858	.455064639	2.884	.337	6.773	.662
300_31BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
400_31BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
500_31BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
100_41BAF	50	89.573075	96.273242	89.74039850	.137031324	.968957783	6.576	.337	44.578	.662
200_41BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
300_41BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
400_41BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
500_41BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
100_51BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
200_51BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
300_51BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
400_51BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
500_51BAF	50	89.573075	89.573075	89.57307500	.000000000	.000000000	.	.	.	.
Valid N (listwise)	50									

Table B.7: Chapter 5 - Statistic Descriptive Gear pump B (EDBA)

Descriptive Statistics

	N	Minimum	Maximum	Mean		Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Statistic	Std. Error	Statistic	Std. Error
100_31EDBA	50	139.033333	176.575000	150.90883333	1.355282004	9.583290953	1.169	.337	.923	.662
200_31EDBA	50	139.033333	172.216667	146.91800000	.897536638	6.346542432	1.365	.337	3.629	.662
300_31EDBA	50	139.033333	172.216667	145.52250001	.905623939	6.403728287	1.905	.337	5.125	.662
400_31EDBA	50	135.316667	166.341667	142.21700001	.695753314	4.919718863	2.760	.337	11.216	.662
500_31EDBA	50	139.033333	155.975000	142.01650000	.446996939	3.160745668	1.802	.337	6.354	.662
100_41EDBA	50	137.933333	171.858333	148.50266666	.970620387	6.863322574	1.110	.337	1.830	.662
200_41EDBA	50	139.033333	156.233333	143.58966667	.559459598	3.955976759	1.370	.337	2.006	.662
300_41EDBA	50	135.316667	159.141667	142.46950000	.626484478	4.429914229	1.782	.337	4.347	.662
400_41EDBA	50	137.933333	167.808333	142.07150000	.811744506	5.739900446	3.761	.337	15.363	.662
500_41EDBA	50	135.316667	154.658333	141.95066667	.540956512	3.825140176	1.503	.337	2.922	.662
100_51EDBA	50	139.033333	166.100000	148.56750000	1.084145360	7.666065359	.743	.337	-.472	.662
200_51EDBA	50	139.033333	156.741667	143.70583334	.587708917	4.155729603	.813	.337	.436	.662
300_51EDBA	50	137.250000	159.708333	142.99583332	.689773543	4.877435500	1.760	.337	3.617	.662
400_51EDBA	50	135.316667	156.741667	142.03616668	.601938789	4.256349995	1.459	.337	2.645	.662
500_51EDBA	50	137.933333	159.325000	142.09316667	.638956006	4.518101248	2.179	.337	5.207	.662
Valid N (listwise)	50									

Table B.8: Chapter 5 - Statistic Descriptive Gear pump B (BA_F)

Descriptive Statistics

	N	Minimum	Maximum	Mean		Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Statistic	Statistic	Std. Error	Statistic	Statistic	Std. Error	Statistic	Std. Error
100_31BAF	50	137.250000	166.341667	146.66533333	1.084143166	7.666049842	.897	.337	-.044	.662
200_31BAF	50	135.316667	147.300000	140.99983334	.355423280	2.513222113	.413	.337	.190	.662
300_31BAF	50	135.316667	150.908333	140.53033333	.417165671	2.949806745	1.188	.337	2.579	.662
400_31BAF	50	135.316667	146.475000	139.73316667	.353007526	2.496140156	.360	.337	.165	.662
500_31BAF	50	135.316667	143.550000	139.17716666	.269367468	1.904715632	.419	.337	-.048	.662
100_41BAF	50	135.316667	159.525000	142.66550001	.626549960	4.430377257	1.643	.337	3.917	.662
200_41BAF	50	135.316667	149.083333	140.56333333	.344897451	2.438793261	1.347	.337	2.576	.662
300_41BAF	50	137.250000	144.133333	139.84800000	.185892247	1.314456685	1.066	.337	2.067	.662
400_41BAF	50	135.316667	144.700000	139.09333334	.229905457	1.625677076	.627	.337	3.363	.662
500_41BAF	50	135.316667	141.650000	138.64933334	.213043334	1.506443860	-.728	.337	.697	.662
100_51BAF	50	135.316667	148.133333	141.68500000	.391803927	2.770472139	.213	.337	-.353	.662
200_51BAF	50	135.316667	145.408333	139.98333334	.288195181	2.037847668	.544	.337	.999	.662
300_51BAF	50	135.316667	141.650000	139.20050001	.145088606	1.025931372	-.843	.337	4.073	.662
400_51BAF	50	136.683333	142.666667	139.22416667	.152822140	1.080615714	.646	.337	3.198	.662
500_51BAF	50	135.316667	140.475000	138.84733334	.139047709	.983215778	-1.593	.337	3.192	.662
Valid N (listwise)	50									

Table B.9: Chapter 5 - Gear pump A normality test

Tests of Normality

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Iter_pop	Statistic	df	Sig.	Statistic	df	Sig.
GearA_EDBA	100_31	.364	50	<.001	.461	50	<.001
	100_41	.423	50	<.001	.522	50	<.001
	100_51	.482	50	<.001	.406	50	<.001
	200_31	.480	50	<.001	.269	50	<.001
	200_41	.478	50	<.001	.197	50	<.001
	200_51	.531	50	<.001	.248	50	<.001
	300_31	.517	50	<.001	.127	50	<.001
	300_41	.540	50	<.001	.198	50	<.001
	300_51	.536	50	<.001	.125	50	<.001
	400_31	.536	50	<.001	.125	50	<.001
	400_41	.536	50	<.001	.125	50	<.001
	400_51	.536	50	<.001	.125	50	<.001
	500_31	.528	50	<.001	.167	50	<.001
	500_41	.	50	.	.	50	.
	500_51	.536	50	<.001	.125	50	<.001
GearA_BAF	100_31	.456	50	<.001	.395	50	<.001
	100_41	.509	50	<.001	.169	50	<.001
	100_51	.	50	.	.	50	.
	200_31	.527	50	<.001	.349	50	<.001
	200_41	.	50	.	.	50	.
	200_51	.	50	.	.	50	.
	300_31	.	50	.	.	50	.
	300_41	.	50	.	.	50	.
	300_51	.	50	.	.	50	.
	400_31	.	50	.	.	50	.
	400_41	.	50	.	.	50	.
	400_51	.	50	.	.	50	.
	500_31	.	50	.	.	50	.
	500_41	.	50	.	.	50	.
	500_51	.	50	.	.	50	.

a. Lilliefors Significance Correction

Table B.10: Chapter 5 - Gear pump B normality test

Tests of Normality

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Iter_pop	Statistic	df	Sig.	Statistic	df	Sig.
GearB_EDBA	100_31	.129	50	.036	.889	50	<.001
	100_41	.123	50	.058	.928	50	.005
	100_51	.144	50	.011	.915	50	.002
	200_31	.130	50	.035	.898	50	<.001
	200_41	.165	50	.002	.866	50	<.001
	200_51	.130	50	.033	.906	50	<.001
	300_31	.191	50	<.001	.825	50	<.001
	300_41	.159	50	.003	.841	50	<.001
	300_51	.183	50	<.001	.814	50	<.001
	400_31	.219	50	<.001	.735	50	<.001
	400_41	.278	50	<.001	.521	50	<.001
	400_51	.200	50	<.001	.838	50	<.001
	500_31	.178	50	<.001	.777	50	<.001
	500_41	.183	50	<.001	.850	50	<.001
	500_51	.229	50	<.001	.717	50	<.001
GearB_BAF	100_31	.146	50	.009	.900	50	<.001
	100_41	.171	50	<.001	.866	50	<.001
	100_51	.145	50	.010	.936	50	.010
	200_31	.110	50	.179	.971	50	.255
	200_41	.225	50	<.001	.838	50	<.001
	200_51	.240	50	<.001	.870	50	<.001
	300_31	.175	50	<.001	.911	50	.001
	300_41	.264	50	<.001	.865	50	<.001
	300_51	.335	50	<.001	.777	50	<.001
	400_31	.114	50	.123	.963	50	.116
	400_41	.268	50	<.001	.842	50	<.001
	400_51	.350	50	<.001	.770	50	<.001
	500_31	.167	50	.001	.948	50	.029
	500_41	.282	50	<.001	.858	50	<.001
	500_51	.395	50	<.001	.708	50	<.001

a. Lilliefors Significance Correction

Table B.11: Chapter 5 - Gear pump A and B homogeneity test

Test of Homogeneity of Variance

		Levene Statistic	df1	df2	Sig.
GearA_EDBA	Based on Mean	11.653	14	735	<.001
	Based on Median	3.872	14	735	<.001
	Based on Median and with adjusted df	3.872	14	262.360	<.001
	Based on trimmed mean	7.551	14	735	<.001
GearA_BAF	Based on Mean	17.344	14	735	<.001
	Based on Median	4.337	14	735	<.001
	Based on Median and with adjusted df	4.337	14	94.091	<.001
	Based on trimmed mean	8.133	14	735	<.001
GearB_EDBA	Based on Mean	7.254	14	735	<.001
	Based on Median	5.442	14	735	<.001
	Based on Median and with adjusted df	5.442	14	539.180	<.001
	Based on trimmed mean	6.670	14	735	<.001
GearB_BAF	Based on Mean	29.396	14	735	<.001
	Based on Median	22.333	14	735	<.001
	Based on Median and with adjusted df	22.333	14	232.077	<.001
	Based on trimmed mean	26.876	14	735	<.001

Table B.12: PEI values - Gear pump A

Iteration_population Algorithm	Average disassembly time	Average NFE	PEI (x 0.000001)
300_31 EDBA	89.8065	10530	4.23
300_31 BA_F	89.5731	10506	4.25
400_31 EDBA	89.6047	14030	3.18
400_31 BA_F	89.5731	14006	3.19
500_31 EDBA	89.7387	17530	2.54
500_31 BA_F	89.5731	17506	2.55
100_41 BA_F	89.7404	4516	9.87
200_41 BA_F	89.5731	9016	4.95
300_41 EDBA	89.6364	13540	3.30
300_41 BA_F	89.5731	13516	3.30
400_41 EDBA	89.5747	18040	2.48
400_41 BA_F	89.5731	18016	2.48
500_41 EDBA	89.5731	22540	1.98
500_41 BA_F	89.5731	22516	1.98
100_51 BA_F	89.5731	5526	8.08
200_51 BA_F	89.5731	11026	4.05
300_51 EDBA	89.6047	16550	2.70
300_51 BA_F	89.5731	16526	2.70
400_51 EDBA	89.6047	22050	2.02
400_51 BA_F	89.5731	22026	2.03
500_51 EDBA	89.6047	27550	1.62
500_51 BA_F	89.5731	27526	1.62

Table B.13: PEI values - Gear pump B

Iteration_population Algorithm	Average disassembly time	Average NFE	PEI (x 0.000001)
400_31 BA_F	139.7332	14006	2.04
500_31 BA_F	139.1772	17506	1.64
400_41 BA_F	139.0933	18016	1.60
500_41 BA_F	138.6493	22516	1.28
300_51 BA_F	139.2005	16526	1.74
400_51 BA_F	139.2242	22026	1.30
500_51 BA_F	138.8473	27526	1.05