PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
الجـمـهـوريـة الجـزائـريـة الديـمـقـراطيـة الشـعـبيـة

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

HIGHER SCHOOL IN APPLIED SCIENCES
--T L E M C E N--

وزارة التعليـم العالـي والبحـث العلمـي

المـدرسـة العليـا في العلوم التطبيقيـة
-تلمسان-

Final degree project

For obtaining the Engineering Diploma

Field: Industrial engineering
Major: Industrial Management and Logistics

Presented by :
GAOUAR Rim Djazia
TERKI HASSAINE Mohieddine El-Hadi

Subject

Formulation and mathematical programming heuristic for Flexible Layout Design Problem

Publicly defended, on 07/07/2024, in front of the jury composed of :

Mrs. SEKKAL Nourelhouda	MCB	ESSA. Tlemcen	Président
Mr. MALIKI Fouad	MCA	ESSA. Tlemcen	Thesis Director
Mr. BRAHIMI Nadjib	Prof	Rennes Business School	Thesis Co-Directeur
Mr. SOUIER Mehdi	Prof	Univ. Tlemcen	Examiner 1
Mr. HADDOU BENDERBAL Hichem	Associate Professor	Univ. Aix-Marseille	Examiner 2

Academic Year : 2023 / 2024

Gratitude

First and foremost, we would like to thank "Allah," the Almighty and Merciful, who has given us the strength, will, and courage to complete this work.

We wish to express our sincere gratitude and deep appreciation to:

Our supervisor and academic leader, **Mr. Fouad MALIKI**, for nurturing us over these pleasantly spent three years. We are profoundly grateful for his guidance and for enriching our knowledge in this field.

Our co-supervisor, **Mr. Nadjib BRAHIMI**, for affording us this opportunity and for enhancing our understanding of the subject matter. We value immensely the invaluable advice and direction he provided throughout the execution of this modest work.

To the members of our jury **Mrs. Nourelhouda SEKKAL**, **Mr.Mehdi SOUIER**, **Mr.Hichem HADDOU BENDERBAL** for the esteemed honor they bestow upon us by agreeing to evaluate this modest effort.

We would also like to thank our dear professor who has been with us through these past three years, **Mr. Mohammed BENNEKROUF**, for giving us the opportunity to explore plentiful information and for his invaluable guidance and support throughout our journey.

To all our teachers who have contributed significantly to the relevance of our education.

Finally, we extend our thanks to all who have played a part, whether directly or indirectly, in the realization of this thesis. Your interest in our work is greatly appreciated.

Acknowledgement

I am deeply grateful to the people who have supported me during my academic studies.

First and foremost, I am very thankful to my parents, Hayet HADJ SLIMANE and Kamel GAOUAR, for their unwavering love and support. They have provided me with everything I needed to become the person I am today, and I am forever grateful for their sacrifices and encouragement.To my brothers Alaa, Abdessamad.

I also appreciate my extended family, including my aunts, uncles, and cousins, especially Yasmine, Amira, Rihem, and Ibtissem. They have been a constant source of support and inspiration for me since we were young.

To my second family Yasmine, Linda, Hadjer, Ghizlene ,Feriel and the family with whom with whom I studied in the same class last 3 years, we have shared countless memorable moments and created a bond that I will cherish forever. To all my friends Nassim ,Nadir, Azeddine, Wadie, Salah, Jami,Youcef.

I would also like to thank my project partner, Mohieddine El-Hadi Terki Hassaine, for his hard work. Without him, I would not have been able to make it this far in our work.

Finally, I dedicate this work to the younger version of myself, who once thought this day would be so far away. I want to let her know that she has made it, and in the most wonderful circumstances.

Acknowledgement

I am deeply grateful to the people who have supported me during my academic studies.

First and foremost, I am very thankful to my parents, for their unwavering love and support. They have provided me with everything I needed to become the person I am today, and I am forever grateful for their sacrifices and encouragement.To my two sisters, Saliha and Awicha and to my brother Younes. Your presence in my life has been a source of joy, support, and inspiration. Your love and encouragement have pushed me to give my best.

I also appreciate my extended family, including my aunts, uncles, and cousins. They have been a constant source of support and inspiration for me since we were young.

I would like to extend my deepest gratitude to Nassim, Azedine, Nadir, Salah, Wadie, Youcef and Zinedine, who have been like brothers to me. Your presence has been a source of strength and inspiration, and I am truly grateful for all the moments we've shared. To all my friends Amina, Hadjer, Yasmine, Linda, Ghizlene, Feriel and the family with whom I studied in the same class last 3 years, we have shared countless memorable moments and created a bond that I will cherish forever.

To two wonderful individuals, who have always been there for me, even in the toughest times. Your genuine friendship and unwavering support have helped me overcome obstacles and persevere. Thank you for your precious friendship

I would also like to thank my project partner, Rim Djazia GAOUAR, for her perseverance and hard work. Without her, I would not have been able to make it this far in our work.

To my friends, my loved ones, and my 2024 Industrial Engineering class.

Contents

Gratitude i

Acknowledgement ii

Acknowledgement iii

List of Abbreviations vii

List of Figures viii

List of Tables ix

General Introduction 1

1 General Overview of Manufacturing Systems 3
- 1.1 Introduction . . . 3
- 1.2 Definitions . . . 4
  - 1.2.1 Manufacturing Systems . . . 4
  - 1.2.2 Craft Manufacturing . . . 5
  - 1.2.3 DMS(Dedicated Manufacturing System) . . . 6
  - 1.2.4 FMS(Flexible Manufacturing System) . . . 6
  - 1.2.5 RMS(Reconfigurable Manufacturing System) . . . 7
  - 1.2.6 MMS (Matrix-structured Manufacturing System) . . . 11
- 1.3 Assembly Line Balancing . . . 12
- 1.4 Conclusion . . . 13

2 RMS literature review 15
- 2.1 Introduction . . . 15
- 2.2 Literature review . . . 15
  - 2.2.1 DMS . . . 15
  - 2.2.2 FMS . . . 15
  - 2.2.3 RMS . . . 16
  - 2.2.4 MMS . . . 19
  - 2.2.5 ALBP . . . 20
  - 2.2.6 Our reference articles . . . 24
- 2.3 Conclusion . . . 25

3 Considered problem and proposed methods **26**
3.1 Introduction . . . 26
3.2 Model . . . 26
3.2.1 Model Notation . . . 27
3.2.2 Model Formulation . . . 28
3.2.3 Proposed Method . . . 29
3.3 2-Phased Resolution Method . . . 30
3.3.1 Motivation . . . 31
3.3.2 Set-Covering Notation . . . 32
3.3.3 Set-Covering model . . . 32
3.4 Data Sets . . . 32
3.5 Conclusion . . . 34

4 Results and discussions **35**
4.1 Introduction . . . 35
4.2 Results(1200 seconds) . . . 35
4.2.1 A description of the result obtained . . . 36
4.2.2 Bowman C=19, 8 tasks . . . 38
4.2.3 EX C=18, 5 tasks . . . 39
4.2.4 EX C=20, 5 tasks . . . 40
4.2.5 Jackson C=08, 11 tasks . . . 41
4.2.6 Jaeschke C=07, 9 tasks . . . 42
4.2.7 Mansoor C=47, 11 tasks . . . 43
4.2.8 Mertens C=08, 7 tasks . . . 44
4.2.9 Mitchell C=14, 21 tasks . . . 45
4.2.10 Remaining instance (1200s) . . . 46
4.3 Results(300 seconds) . . . 47
4.3.1 Bowman C=19, 8 tasks . . . 47
4.3.2 EX C=18, 5 tasks . . . 48
4.3.3 EX C=20, 5 tasks . . . 49
4.3.4 Jackson C=08, 11 tasks . . . 50
4.3.5 Jaeschke C=07, 9 tasks . . . 51
4.3.6 Mansoor C=47, 11 tasks . . . 52
4.3.7 Mertens C=08, 7 tasks . . . 53
4.3.8 Mitchell C=14, 21 tasks . . . 54
4.3.9 Remaining instances (300s) . . . 55
4.4 Conclusion . . . 56

General Conclusion and Perspectives **57**

Bibliography **58**

Webography **62**

Appendix **63**
4.4.1 Bowman . . . 63
4.4.2 EX C=18 . . . 64
4.4.3 EX C=20 . . . 65
4.4.4 Jackson . . . 66

4.4.5 Jaeschke 67
4.4.6 Mansoor 68
4.4.7 Mertens 69
4.4.8 Mitchell 70

List of Abbreviations

Abbreviation	Definition
AGV	Automated Guided Vehicules
ALB	Assembly-Line Balancing
CEGA	Cell Evaluation and Genetic Algorithms
CNC	Computer Numerical Control
CS	Cuckoo Search
DMS	Dedicated Manufacturing System
FJSP	Flexible Job-Shop Scheduling Problem
FLDP	Flexible Layout Design Problem
FMS	Flexible Manufacturing System
GA	Genetic Algorithms
GRASP	Greedy Randomized Adaptive Search Procedure
LB	Lower Bound
MA	Memetic Algorithm
MALBP	Mixed Assembly Line Balancing Problem
MILP	Mixed-Integer Linear Programming
MMS	Matrix-Structured Manufacturing System
MS	Manufacturing System
NSGA	Nondominated Sorting Genetic Algorithm
OEM	Automotive-Original Equipment Manufacturers
PPS	Production Planning and Scheduling
PSO	Praticle Swarm Optimization
RMS	Reconfigurable Manufacturing System
SA	Simulated Annealing
SALB	Simple Assembly Line Balancing
SH	structure heterogeneity
TTH	Task Time Heterogeneity
UB	Upper Bound

List of Figures

1.1 Manufacturing System source: sketchbubble 5
1.2 Dedicated Manufacturing System 6
1.3 Reconfigurable Manufacturing System 8
1.4 Flow line configuration 9
1.5 Reconfigurable Cellular Manufacturing system 10
1.6 Rotary Machining System 11
1.7 Comparison of a classic MS and a MMS configuration 12
1.8 Diffrent Assembly Lines 13

3.1 An example of a precedence diagram 34

4.1 UB and LB Convergence 36
4.2 GAP From LB 36
4.3 Result example 37
4.4 The objective variation curves example 37

List of Tables

2.1 Production planning & Scheduling . . . 17
2.2 Layout design summarized articles . . . 18
2.3 line balancing/ re-balancing summarized articles . . . 18
2.5 RMS design summarized articles . . . 19

3.1 Index sets . . . 27
3.2 Parameters . . . 27
3.3 Decision variables . . . 28
3.4 Model notation for set covering . . . 32

4.1 Bowman GAP from LB for MILP and 2 phased (1200s) . . . 38
4.2 EX C=18 GAP from LB for MILP and 2 phased (1200s) . . . 39
4.3 EX C=20 GAP from LB for MILP and 2 phased (1200s) . . . 40
4.4 Jackson GAP from LB for MILP and 2 phased (1200s) . . . 41
4.5 Jaeschke GAP from LB for MILP and 2 phased (1200s) . . . 42
4.6 Mansoor GAP from LB for MILP and 2 phased (1200s) . . . 43
4.7 Mertens GAP from LB for MILP and 2 phased (1200s) . . . 44
4.8 Mitchell GAP from LB for MILP and 2 phased (1200s) . . . 45
4.9 Remaining instances: GAP from LP for MILP and 2-phased (1200s) . 46
4.10 Bowman GAP from LB for MILP and 2 phased (300s) . . . 47
4.11 EX C=18 GAP from LB for MILP and 2 phased (300s) . . . 48
4.12 EX C=20 GAP from LB for MILP and 2 phased (300s) . . . 49
4.13 Jackson GAP from LB for MILP and 2 phased (300s) . . . 50
4.14 Jaeschke GAP from LB for MILP and 2 phased (300s) . . . 51
4.15 Mansoor GAP from LB for MILP and 2 phased (300s) . . . 52
4.16 Mertens GAP from LB for MILP and 2 phased (300s) . . . 53
4.17 Mitchell GAP from LB for MILP and 2 phased (300s) . . . 54
4.18 Remaining instances : GAP from LB for MILP and 2-phased (300s) . 55
4.19 MILP and 2-Phased Results within 20 minutes : Bowman . . . 63
4.20 MILP and 2-Phased Results within 20 minutes : EX C=18 . . . 64
4.21 MILP and 2-Phased Results within 20 minutes : EX C=20 . . . 65
4.22 MILP and 2-Phased Results within 20 minutes : Jackson . . . 66
4.23 MILP and 2-Phased Results within 20 minutes : Jaeschke . . . 67
4.24 MILP and 2-Phased Results within 20 minutes : Mansoor . . . 68
4.25 MILP and 2-Phased Results within 20 minutes : Mertens . . . 69
4.26 MILP and 2-Phased Results within 20 minutes : Mitchell . . . 70

General Introduction

Reconfigurable Manufacturing System (RMS) has been the center of research since the 1990s and has been evolving ever since, thanks to its multiple characteristics that positively influence every production line (products) and meet the demands. To further access the reconfigurability characteristics of RMS, a new manufacturing system (MS) called the Matrix-structured Manufacturing System (MMS) has been introduced. The MMS has the flexibility and reconfigurability of RMS manufacturing systems, featuring matrix-structured layouts with multiple stations where flows are often transported by AGVs. Assembly line balancing has also been a focal point of research since the notion of manufacturing systems and stations existed. Researchers have been striving to find the perfect balance for every type of manufacturing system using all existing methodologies or developing their own solutions, known as heuristics or hybrid-heuristics.

In this thesis, we will focus on the line balancing of a flexible layout design problem (FLDP), which involves designing a flexible layout for an assembly segment. This includes the integrated problems of station formation and station location while also anticipating the operational AGV flow,particularly on the part where the MMS layout is introduced to the manufacturing system. Our approach proposes integrating the principle of assigning resources to each station through this matrix-structured layout to reflect the reality of manufacturing systems. We will work on the mathematical model already established by Grunow which takes into consideration a multi objective problem minimzing both the number of opened locations and the transportation flow between the location secured by the AGV's, introducing our own principle of resource allocation. The proposed method involves establishing a mat-heuristic composed of two phases: the first phase compiles Grunow's model without any changes, and the second phase determines the allocation of resources without any constraints on capacity or utilization.

As part of the professional training for obtaining the **state engineer's diploma** in **the Industrial Engineering specialty** at the **Higher School of Applied Sciences of Tlemcen**, a two-month end-of-study internship was carried out within the **IMT Atlantique Nantes**. This internship was an enriching and instructive experience. It allowed us to gain a clearer understanding of the practical application of the theoretical knowledge acquired during our training.

For the structure of our thesis, we will focus on the flexible layout design problem. The work presented will be divided into parts that encompass everything we have learned or accomplished during our two-month internship and since the beginning of our journey at the university. This work will be divided into four chapters, each detailing the methodologies, techniques, and information utilized to achieve our objectives:

In Chapter 1, we will explore the evolution of manufacturing systems, starting from Craft Manufacturing and progressing to the Dedicated Manufacturing System, then to the Flexible Manufacturing System, and finally to the Reconfigurable Manufacturing System and the Matrix-structured Manufacturing System. Additionally, we will define the assembly line balancing problem.

In the next chapter, we will review the existing literature on various manufacturing systems. We will present the reference articles that inspired our work on this problem.

In the third chapter, we will present our mathematical model, including model notation, explanations of constraints, and objective functions. We will then introduce the mat-heuristic approach which we will call 2-phased method, used to resolve the problem, along with its formulation. Finally, we will present the datasets used to test our approach.

In the last chapter, we will test and thoroughly discuss the results of the Mixed Integer Linear Programming (MILP) and our 2-phased method. We will conduct tests on 128 instances with time limits of 20 minutes and 5 minutes.

Chapter 1

General Overview of Manufacturing Systems

1.1 Introduction

The industrial sector has undoubtedly undergone significant transformations over the years, with each revolution bringing about new technologies and systems that have revolutionized the way manufacturing processes are carried out. The first industrial revolution, which took place in the late 18th and early 19th century, saw the introduction of mechanization, which allowed for the mass production of goods. This revolution was followed by the second industrial revolution, which was characterized by the introduction of new energy sources, such as electricity and oil, and the emergence of assembly lines, which made production more efficient.

The third industrial revolution, also known as the digital revolution, was marked by the rise of computer technology and automation, which enabled manufacturers to streamline their operations and reduce costs. This revolution paved the way for the fourth industrial revolution, which is currently underway and is set to transform the industry in ways that were once thought impossible.

The fourth industrial revolution is unique in that it has introduced the Internet of Things (IoT), the Industrial Internet of Things (IIoT), and the Internet of People (IoP) to the mechanisms of an industrial place. This integration of advanced technologies has resulted in the creation of smart factories, which are highly automated and intelligent, with machines that communicate with each other and with humans. The use of AI and machine learning has also made it possible for machines to learn and improve on their own, leading to increased productivity and efficiency. It is also characterized by a focus on sustainability and the environment. Manufacturers are now turning to sustainable practices such as recycling, renewable energy, and eco-friendly materials. This focus on sustainability not only benefits the environment but also helps companies save costs in the long run. As we continue to move forward, it is clear that the industry will continue to evolve, with new innovations and advancements that will shape the future of manufacturing.

1.2 Definitions

1.2.1 Manufacturing Systems

The modern manufacturing industry (Fig 1.1) thrives on the essential contribution of human workers and their ability to operate a collection of machines and tools to initiate the processes required to produce goods or services that provide the needs of people. However, the success of a manufacturing system is determined by various characteristics such as efficiency, flexibility, quality, safety, and cost-effectiveness.

Efficiency is a vital factor in a manufacturing system as it ensures optimal utilization of resources and minimizes wastage. A manufacturing system that is flexible can quickly adapt to changing consumer needs and market demands, thereby enhancing customer satisfaction. Quality is another essential aspect as it ensures that the products or services produced are of high standards and meet the expectations of the consumers, leading to customer loyalty and repeat purchases. Safety is a crucial factor in a manufacturing system as it ensures the well-being of the workers and prevents accidents or injuries. A manufacturing system that prioritizes safety creates a conducive work environment that fosters productivity and job satisfaction. Moreover, cost-effectiveness is an essential factor as it ensures that the manufacturing system is profitable and sustainable in the long run.

Over the years, the manufacturing industry has undergone several revolutionary changes resulting in the emergence of different manufacturing systems. Each manufacturing system possesses its unique advantages and challenges. For instance, the mass production system introduced during the industrial revolution was highly efficient, but it lacked flexibility and produced standardized products. In contrast, the lean production system, which emerged in the 1990s, is highly flexible and emphasizes quality, but it requires a skilled workforce. The success of a manufacturing system is determined by its ability to meet the needs of the consumers while ensuring efficiency, flexibility, quality, safety, and cost-effectiveness. As the manufacturing industry continues to evolve, it is essential to embrace innovative technologies and best practices to remain competitive and meet the demands of the market.

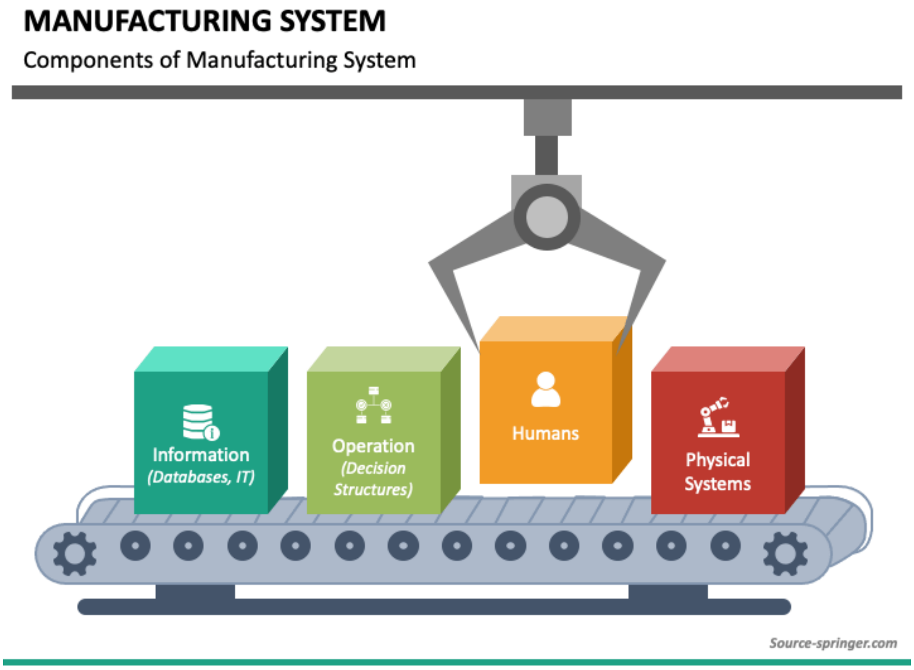

Figure 1.1: Manufacturing System source: sketchbubble

1.2.2 Craft Manufacturing

Craft manufacturing is a process that has been used for centuries, long before the introduction of automated manufacturing lines and systems in the 20th century. This method relies on the expertise of highly skilled workers and the use of simple but flexible tools to produce goods that meet the precise demands of the customer. In this process, the focus is on quality and attention to detail, with the aim of creating a unique product that stands out from the rest.

Craft manufacturing is a time-honored tradition that has been passed down from generation to generation. The skills and techniques used in this process are often learned through apprenticeships, where young workers are taught by experienced craftsmen. This process not only ensures that the craft is preserved, but it also helps to maintain the quality of the product. It is known for its attention to detail and precision. The craftsmen who work in this field take great care to ensure that each product is made to the highest standards. This often involves using specialized tools and techniques that have been refined over many years. The result is a product that is not only functional but also beautiful and unique.

One of the advantages of craft manufacturing is that it allows for a high degree of customization. Because each product is made by hand, the customer can work with the craftsman to create a product that meets their exact needs and specifications. This level of customization is not possible with automated manufacturing, which produces products in large quantities that are designed to meet the needs of the masses.

1.2.3 DMS(Dedicated Manufacturing System)

In the 1990s, there were two main types of manufacturing systems that were common in the industry (Fig 1.2) [GŚ12] . The first one was the Continuous Manufacturing system, which primarily involved producing goods for stock. This system relied on forecasting to estimate the likely demand for the products. The production process was standardized, and the inventory was managed through the first-in, first-out (FIFO) method. Additionally, the work carried out in this system was not diverse, and the workload was balanced. On the other hand, the second type of manufacturing system was called the Intermittent Manufacturing system. This system was designed to satisfy orders placed by customers. The production facilities were flexible enough to handle a wide variety of products and sizes. The storage was done between operations, and the system could accommodate small quantities of products that were flexible in nature. However, the workload in this system was unbalanced, and the production process was not standardized.

The Continuous Manufacturing system was ideal for companies that manufactured products in high volumes and had a stable and predictable demand for their products. This system allowed companies to achieve economies of scale and optimize their production processes. It was also ideal for companies that wanted to keep their inventory levels low and minimize the risk of holding excess stock. On the other hand, the Intermittent Manufacturing system was ideal for companies that produced a wide variety of products or had a constantly changing demand for their products. This system allowed companies to be more agile and responsive to their customers' needs. It also enabled companies to produce small quantities of products efficiently and cost-effectively.

Both the Continuous Manufacturing system and the Intermittent Manufacturing system had their unique strengths and weaknesses. Companies had to choose the system that was best suited to their production requirements and business goals. The decision to adopt a particular manufacturing system had a significant impact on the company's operational efficiency, profitability, and customer satisfaction. Therefore, it was crucial for companies to carefully evaluate their options before making a decision.

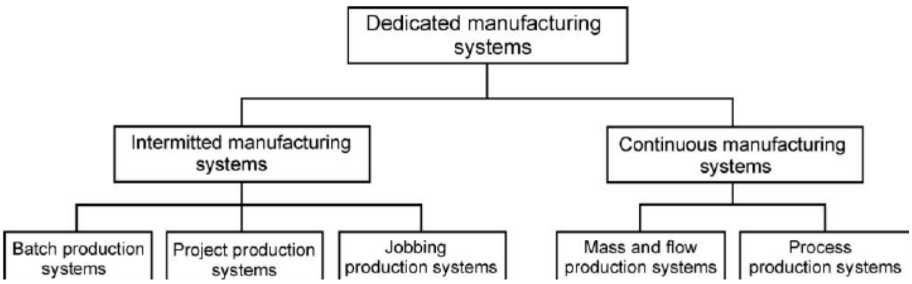

Figure 1.2: Dedicated Manufacturing System

1.2.4 FMS(Flexible Manufacturing System)

In the 1960s, the market competition was spiraling, and companies were struggling to keep up with the ever-changing demands of the consumers. It was during this time that FMS came into existence. It was a revolutionary concept that provided a fast and flexible response to unexpected changes in the market. FMS is a group of numerically controlled machinery that allows for the production of a large variety of small quantities of products. The system is designed to load and unload tools and workpieces automatically, which significantly reduces the need for human intervention. This means that the system can operate virtually unattended for long periods, making it incredibly efficient and cost-effective.

One of the most significant advantages of FMS is its flexibility. The system can quickly adjust to changes in demand, allowing companies to produce a wide variety of products without having to reconfigure their production line. This is particularly beneficial for companies that produce a range of products or have a fluctuating demand for their products.

FMS has become increasingly popular in recent years, and many companies have adopted this technology to improve their productivity and efficiency. The system has also helped companies to reduce their manufacturing costs, as it eliminates the need for manual labor and reduces the risk of errors in production. This MS has revolutionized the manufacturing industry by providing a fast, flexible, and cost-effective solution to the production of small quantities of products. With the growing demand for customized products and the need for quick response times, FMS is quickly becoming an essential tool for companies looking to stay ahead of the competition.

1.2.5 RMS(Reconfigurable Manufacturing System)

An RMS is a production system that is designed to be flexible and adaptable. It allows us to add, modify, delete, and exchange modules and machines, depending on the production needs and changes. This means that RMS can easily accommodate changes in production processes, and it can quickly adjust to new market demands. The primary focus of RMS is to produce part families. Part families are groups of parts that have similar characteristics, such as size, shape, or function. By grouping parts into families, RMS can optimize production processes and reduce the time and cost of manufacturing.

One of the key advantages of RMS is its ability to reconfigure itself quickly. This means that if a company needs to change its production processes, it can do so without having to invest in new equipment or machinery. Instead, it can simply reconfigure its existing RMS to meet the new requirements. Another benefit of RMS is that it can improve the quality of the products produced. By using advanced technology and automation, RMS can reduce the risk of errors and defects, which can lead to higher customer satisfaction and loyalty.

RMS is a flexible and adaptable production system (Fig 1.3) [And17] that can easily accommodate changes in production processes. Its focus on part families allows for optimization of production processes, and its ability to reconfigure quickly can save time and cost. Additionally, it can improve the quality of products produced, leading to higher customer satisfaction and loyalty.

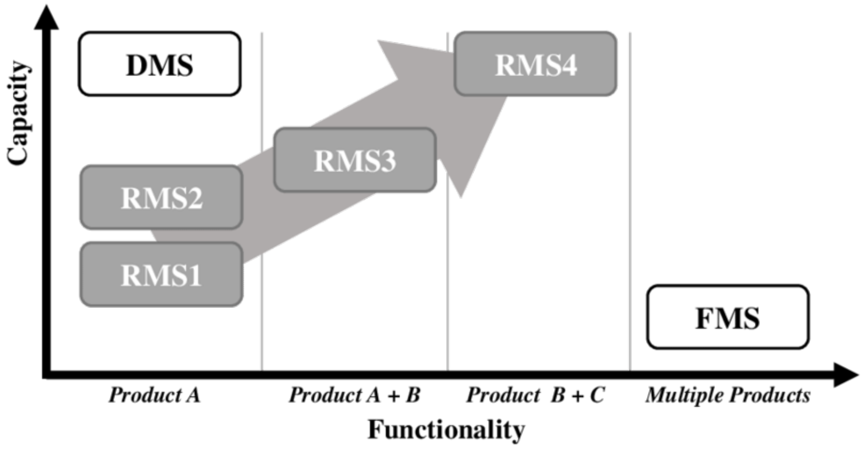

Figure 1.3: Reconfigurable Manufacturing System

In modern manufacturing, the need to remain competitive and adapt to rapidly changing markets has led to the development of advanced production systems. One such system is the RMS. This system has become increasingly popular in recent years due to its many advantages, including scalability, convertibility, customization, modularity, diagnosability, and integrability.

- **Scalability :** it's a key feature of an RMS, as it allows manufacturers to adjust production capacity according to the situation. This can be achieved by adding or removing machines, changing production lines, or reconfiguring existing equipment. By doing so, manufacturers can quickly respond to changes in demand, reduce lead times, and improve overall efficiency.

- **Convertibility:**is another important characteristic of an RMS. This refers to the ability to transform the functionality of the system to satisfy specific requirements. For example, an RMS may be configured to produce one product today and a completely different product tomorrow. This flexibility allows manufacturers to quickly adapt to changes in market demands, without the need to invest in new equipment or systems.

- **Customization:**is a feature of an RMS that is limited to part families. This means that the system can be tailored to produce a variety of products within a specific category, such as automotive parts or medical devices. This flexibility allows manufacturers to produce a wide range of products while maintaining their competitive edge.

- **Modularity:** is another key feature of an RMS. This refers to the ability to change parts of the machinery in order to respond to production changes. For example, an RMS may be designed with interchangeable tooling, allowing manufacturers to quickly switch between different product lines. This modularity also makes it easier to maintain and upgrade the system over time.

- **Diagnosability:**is an important characteristic of an RMS, as it allows for real-time diagnosing of product quality. By monitoring the production process and analyzing data, manufacturers can quickly identify and address any quality issues, reducing waste and improving overall efficiency.

- **Integrability:**is the final characteristic of an RMS, and refers to the ability to rapidly integrate modules by hardware and software interfaces. This allows manufacturers to quickly add new equipment or processes to the system, without the need for extensive reconfiguration or downtime. This flexibility is essential for maintaining competitiveness in today's rapidly changing manufacturing environment.

RMS types

RMS is a flexible and adaptable approach to manufacturing that allows for multiple types and configurations to meet the specific needs of a given production line. When first considering the implementation of an RMS, there are various criteria to consider, such as the type of product being manufactured, the volume of production, and the level of automation required.

- **Reconfigurable Flow Lines (RFL)**: These production lines consist of a series of workstations, each equipped with reconfigurable machines that can perform a variety of tasks (Fig 1.4) [YCGBD21]. This type of RMS is particularly effective in high-volume production environments where efficiency and speed are crucial. The flexibility of the RFL allows for quick changes in production processes, making it an ideal choice for companies that require a high level of adaptability.

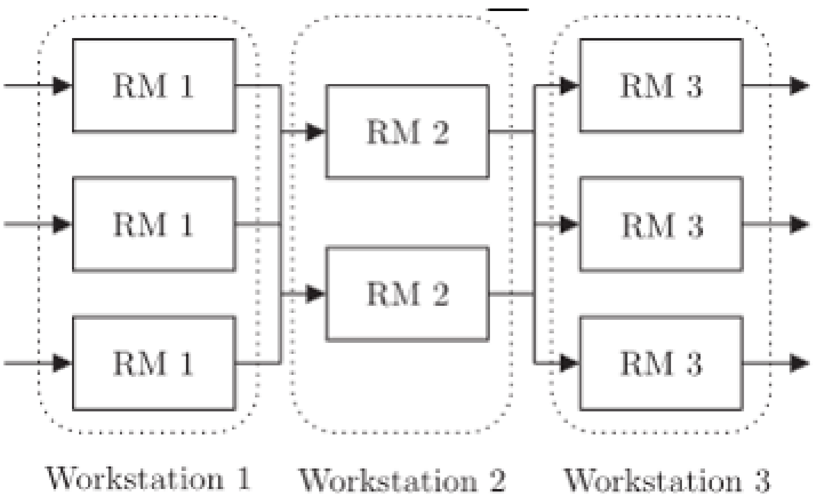

Figure 1.4: Flow line configuration

- **Reconfigurable Cellular Manufacturing System (RCMS)**: This type of production system is based on the concept of group technology, where a group of reconfigurable machines is organized into cells that share similar production tasks (Fig 1.5) [YCGBD21]. The RCMS is particularly effective in low to medium volume production environments where product customization is important. The ability to reconfigure the production line according to product demands makes the RCMS a highly versatile manufacturing system.

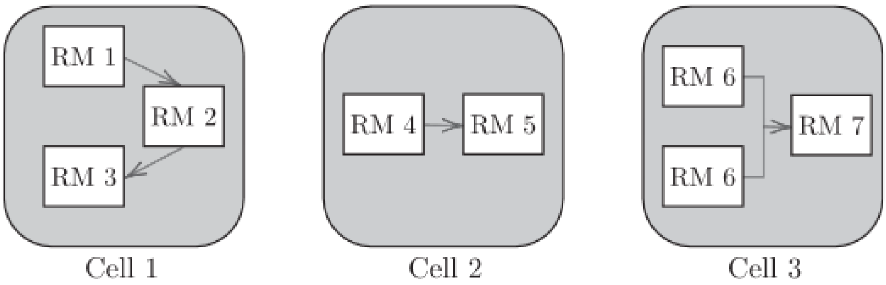

Figure 1.5: Reconfigurable Cellular Manufacturing system

- **The Dynamic Cellular Manufacturing System (DCMS):** is another type of RMS that applies the same principles as the RCMS. The only difference is that the DCMS is composed of movable machines instead of reconfigurable machines. This type of system is particularly effective in environments where space is limited, and production requirements are constantly changing.

- **the Rotary Machining System:** is a type of RMS that utilizes a rotary table to move the product through different modular machines (Fig 1.6) [YCGBD21]. This type of system is particularly effective in high-precision machining applications where accuracy and consistency are crucial. The modular design of the Rotary Machining System allows for easy reconfiguration and modification of the production line, making it an ideal choice for companies that require a high level of flexibility.

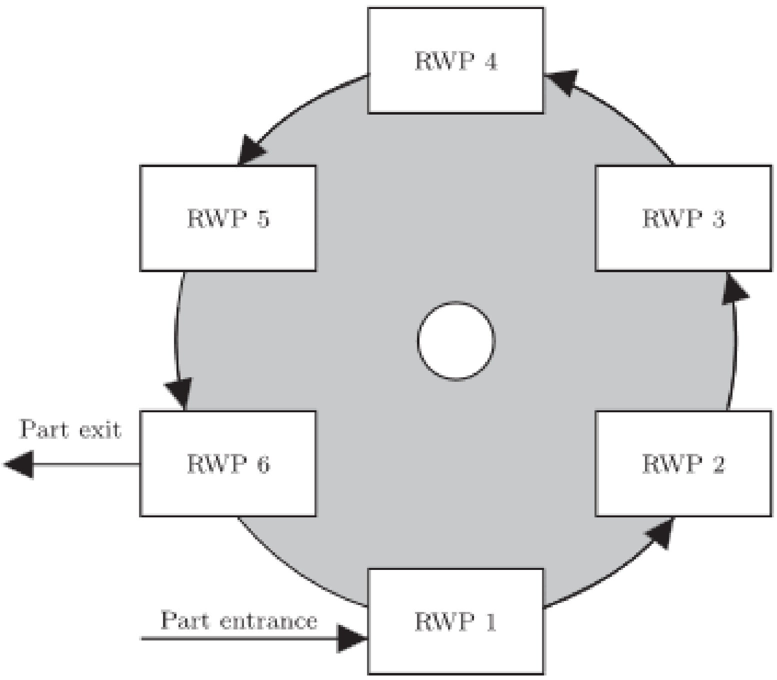

Figure 1.6: Rotary Machining System

1.2.6 MMS (Matrix-structured Manufacturing System)

The matrix manufacturing system, as the name suggests, has a matrix layout where all workstations are interconnected. The two main elements of this system are the products and the workstations. It has some unique principles that distinguish it from traditional manufacturing systems. For instance, each workstation has its own pace and cycle time, which helps prevent starvation and blocking. This system can produce multiple products using routing flexibility with Automated Guided Vehicles (AGVs) to transfer the product flow between workstations. This layout provides a great deal of flexibility for products and task assignment, within certain constraints.

As seen in the figure (Fig 1.7) [SHGT15], the difference between the classic manufacturing system (MS) and the matrix manufacturing system (MMS) is clear. For example, in the classic MS, product 1 must be completed before product 2 to minimize changeover time and maximize workstation utilization. However, in the MMS, a workstation can perform multiple tasks for multiple products, allowing a reduction in the number of workstations or an increase in the number of product types while maintaining high utilization.

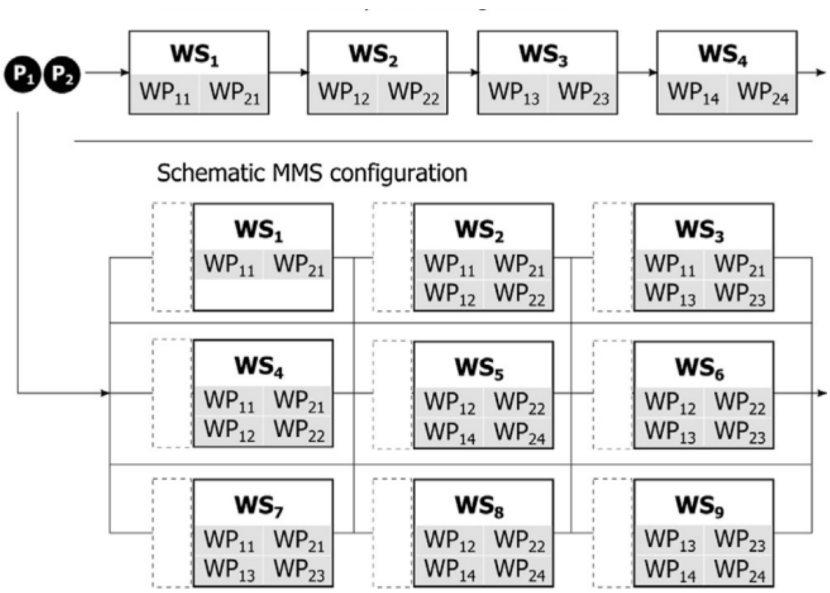

Figure 1.7: Comparison of a classic MS and a MMS configuration

1.3 Assembly Line Balancing

In the manufacturing industry, line balancing (Fig 1.8) [KM13] is an essential process that ensures optimal productivity and efficiency in assembly line operations. The goal of line balancing is to distribute workloads evenly across the production line, minimizing idle time and maximizing throughput. To achieve this, manufacturers must assign the appropriate number of employees or automated machines to each section of the assembly line. This process involves analyzing the cycle time of each station and determining the optimal number of workers or machines required to complete the task within that time frame.

Streamlining workflow is another crucial aspect of line balancing. This involves coordinating workstations and tasks to minimize unnecessary movement and improve the overall flow of the production line. Manufacturers must also continuously evaluate and improve their assembly line processes to identify and eliminate any bottlenecks or inefficiencies that may arise.

There are two primary types of line balancing methods: SALB-1 and SALB-2. SALB-1 focuses on minimizing the number of stations based on cycle time. This involves grouping similar tasks together and eliminating any redundant or unnecessary stations. By reducing the number of stations, manufacturers can minimize setup time and reduce idle time between tasks. On the other hand, SALB-2 aims to reduce cycle time by adjusting the number of stations. This method involves adding or removing stations to balance the workload across the production line. By optimizing the number of stations, manufacturers can achieve faster cycle times and improve overall productivity.

There are two main types of assembly line balancing problems - single-model and multi-model. In both types, there are four subcategories each based on whether the problem is deterministic or probabilistic and whether the assembly line is straight-type or U-type.

The first subcategory, SMDS or Single-Model Deterministic Straight-type, refers to a scenario where there is only one product being manufactured, and the production process is deterministic, meaning that the time required for each task is fixed and known in advance. The assembly line in this case is straight, meaning that the flow of work is linear. SMDU or Single-Model Deterministic U-type is similar to SMDS, but the assembly line is in a U-shape. This is often the case when there are constraints on the floor space available for the production process.

The third subcategory, SMPS or Single-Model Probabilistic Straight-type, is where the production process is not deterministic, and there is some variation in the time required for each task. This could be due to factors such as worker variability or machine breakdowns. The assembly line is still straight in this case. SMPU or Single-Model Probabilistic U-type is the same as SMPS, but the assembly line is in a U-shape.

Moving on to the multi-model subcategories, MMDS or Multi-model Deterministic Straight-type is where there are multiple products being manufactured, but the production process is still deterministic. The assembly line is straight, and each product follows the same sequence of tasks. MMDU or Multi-model Deterministic U-type is similar to MMDS, but the assembly line is in a U-shape.

The seventh subcategory, MMPS or Multi-model Probabilistic Straight-type, is where there are multiple products being manufactured, and the production process is not deterministic. The assembly line is still straight, and there is some variability in the time required for each task. Finally, MMPU or Multi-model Probabilistic Straight-type is the same as MMPS, but the assembly line is in a U-shape.

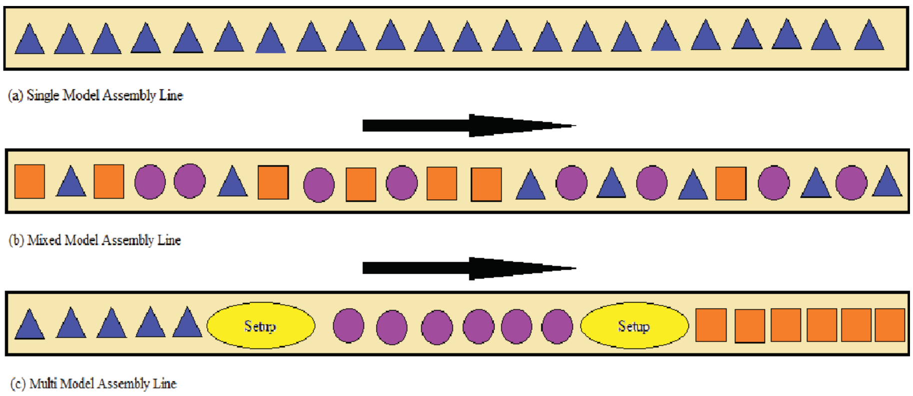

Figure 1.8: Diffrent Assembly Lines

1.4 Conclusion

The manufacturing world has undergone rapid changes over the years, transitioning from the mechanical industry to the energy sector, then to the digital industry, and currently to Industry 4.0 (the integration of artificial intelligence with machines).

There are several types of industrial systems. Craft Manufacturing relies on the expertise of highly skilled workers, allowing for a high degree of customization, but it is not feasible to produce large quantities.

Dedicated Manufacturing Systems, along with Continuous Manufacturing systems and Intermittent Manufacturing systems, are used for large production volumes with less variety. The Flexible Manufacturing System enables the production of a wide variety of products without the need to reconfigure the production line, known for its flexibility, but it is limited in capacity.

Reconfigurable Manufacturing Systems combine the advantages of both Dedicated Manufacturing Systems and Flexible Manufacturing Systems, allowing for the production of high quantities with a wide variety. The Matrix-structured Manufacturing System is a type of RMS where all workstations are interconnected in a matrix layout. This system can produce multiple products using routing flexibility while maintaining high utilization. Assembly Line Balancing is an essential process that ensures optimal productivity and efficiency in assembly line operations.

Chapter 2

RMS literature review

2.1 Introduction

In this chapter, we will discuss the existing literature on the various manufacturing systems we previously mentioned, including Dedicated Manufacturing Systems, Flexible Manufacturing Systems, and Reconfigurable Manufacturing Systems. Our focus will be primarily on Modular Manufacturing Systems (MMS) and Assembly Line Balancing Problems (ALBP).

2.2 Literature review

2.2.1 DMS

This paper [LGZ09] introduces a combined approach of cell evaluation and genetic algorithms (CEGA) to optimize the dedicated remanufacturing system through simulation. The paper begins by highlighting the unique features and challenges of the dedicated remanufacturing process. It then presents a simulation model that incorporates a prioritized stochastic batch arrival mechanism, accounting for factors that influence the total profit. Using this simulation model as a foundation, the CEGA algorithm is developed to optimize the production planning and control strategies for the dedicated remanufacturing facility. The paper further provides a case study based on a remanufacturing facility located in Austin, USA.

2.2.2 FMS

In this study [BD13], a cuckoo search (CS)-based strategy has been developed to optimize the scheduling of a flexible manufacturing system. The goal is to minimize the penalty cost associated with manufacturing delays and maximize machine utilization time. To showcase the application of the CS-based approach, the researchers have modified the Levy flight operator to accommodate the discrete nature of the solution, which was tested on a standard FMS scheduling problem consisting of 43 jobs and 16 machines. The CS scheme was implemented using Matlab, and its performance was compared to other soft computing-based optimization techniques, such as genetic algorithm (GA) and particle swarm optimization, found in the literature. The results demonstrate that the CS-based approach outperforms the existing heuristic algorithms, including GA, for the given problem.

Extensive research has been conducted on the design and operational challenges of automated manufacturing systems, yet many issues remain unresolved. Notably, the scheduling task and the control problem during operation are crucial due to the dynamic nature of flexible manufacturing systems (FMS), such as the flexibility of parts, tools, and automated guided vehicle (AGV) routes. Various traditional optimization techniques have been employed to tackle the FMS scheduling problem. While these methods can provide optimal solutions for small-scale problems, they often prove inefficient when applied to larger-scale issues. In this work [JAPS05], diverse scheduling mechanisms have been developed to generate optimal schedules, including non-traditional approaches like genetic algorithm (GA), simulated annealing (SA) algorithm, memetic algorithm (MA), and particle swarm algorithm (PSA). These methods consider multiple objectives, aiming to minimize both machine idle time and total penalty costs for missed deadlines. The memetic algorithm presented here combines elements of genetic algorithms and simulated annealing. The results of these different optimization algorithms (memetic, genetic, simulated annealing, and particle swarm) are compared, and conclusions are drawn.

Organizing schedules for versatile job workshops is crucial in both production management and combinatorial optimization domains. Yet, attaining the ideal solution through conventional optimization methods remains challenging due to the immense computational complexity involved. Combining various optimization criteria further escalates the intricacy, leading to new challenges. Particle swarm optimization, an evolutionary computational technique, mimics the behavior of flying birds and their information-sharing mechanisms. It harmonizes local search (based on personal experience) and global search (based on neighboring experience), demonstrating high search efficiency. Simulated annealing, a local search algorithm, employs specific probabilities to avoid becoming trapped in local optima, proving effective in diverse situations, including scheduling and sequencing. By thoughtfully integrating these two methodologies in this article [XW05], researchers have developed an easily implemented hybrid approach for the multi-objective flexible job-shop scheduling problem (FJSP). The computational study results have shown that the proposed algorithm is a viable and effective solution, particularly for large-scale FJSP problems.

2.2.3 RMS

This case of study of the LEGO group [WBH12] they tried to make a conceptual model to analyze the most suitable areas for RMS implementations, where they devided their process into four steps: Molding,Decoration,Assembly and Packaging.In each of these steps they narrowed down caracteristics: Responsiveness, Capacity, Functionality and Cost.They put their own scale which goes from 0 to 1 , 0 meaning no existance of the RMS caracteristic in the said step and 1 the full existence. They analyzed the percentage of RMS cacarteristics already found in their process and where they can implement the rest to make their process even more suitable for RMS scale.

RMS optimization problems are grouped into four different categories: RMS design, Production planning and scheduling (PPS), Layout design and Line balancing and re-balancing. This conference paper [BGDJLD21] integrates process planning, scheduling, and layout design problems into a model formulation. The choice of integrating these three optimization problems was made because when treating process planning and scheduling separately, they have conflicting objectives, and layout reconfiguration directly impacts system configuration and material handling distances which affects process planning and scheduling decisions. In the end, the three problems are interconnected by a shared objective: resource allocation to improve system performance.This study aims to minimize total costs: production costs+ machine reconfiguration cost + layout reconfiguration cost + material handling cost and the tardiness cost.

[YCGBD21] The aim of this document is to review research publications focusing on RMS optimization challenges and their resolution techniques. This involves describing the types of RMS and their constituent parts. Furthermore, the relevant objective functions and performance metrics of RMS are discussed. Additionally, a summary of the most commonly employed solution methods and a categorization of optimization problems are provided. We will see a summarizing tables of the search publications in the [YCGBD21]: Production planning & Scheduling (table 2.1), layout design (table 2.2), line balancing/ re-balancing (table 2.3), RMS design (2.5).

- Production planning & Scheduling :

	Objectives	**Count**
scheduling	min makespan	2
	min total cost	2
	min total tardiness	2
	min mean flow time	1
	min mean tardiness	1
planning	max profit	1
	min energy consumption	1
	max throughput	1

Table 2.1: Production planning & Scheduling

- Layout design:

Objectives	Count
min material handling cost	1
min reconfiguration cost	1
min constraint satisfaction penalty	1

Table 2.2: Layout design summarized articles

- Line balancing/ re-balancing

Objectives	Count
min the number of CNC machines	1

Table 2.3: line balancing/ re-balancing summarized articles

- RMS design:

	Objectives	Count
Process planning	min total cost	7
	min completion time	4
	min total cost	3
	min completion time	2
	min makespan	1
	max system flexibility	1
	max system modularity	1
	max throughput	1
	min energy consumption	1
	max machine precision	1
	max smoothness	1
	min machines exploitation time	1
Rotary machining system design	max total cost	2
	min cycle time	1

Flow line configuration selection	min capital cost	5
	min reconfiguration cost	4
	min total cost	2
	max operational capability	2
	max machine reconfigurability	2
	min investment cost	2
	min production cost	1
	min life cycle cost	1
	min inventory cost	1
	min operating cost	1
	min maintenance cost	1
	min capital cost	1
	max machine utilization	1
	max configuration convertibility	1
	min reconfiguration cost	1
	min total tardiness	1
	min reconfiguration cost	1
	min tardiness	1
	max workload balance	1
	min system availability	1
RCMS/DCMS Design	min Inter-cellular movements	2
	min Machine related costs	1
	min total (Re)-Manufacturing costs	1
	min Intra-cellular void	1
	min Total cost	1
	min Total changes in auxiliary modules	1
	min Machine workload balancing	1

Table 2.5: RMS design summarized articles

2.2.4 MMS

In this piece [SWS22], the writer revisits their previous work to examine the effects of flexibility on MMS systems. A cost-focused perspective is applied to assess the long-term planning challenges of MMS design. Specifically, three types of flexibility are explored: Material or Equipment flexibility, Material handling flexibility, and Operation flexibility. The first two relate to the products that need to be produced. Material handling flexibility concerns the loading, unloading, and transportation of materials between stations in the MMS system. Operation flexibility pertains to the ability to produce products using different processes, resulting in the same end product. The author conducts simulations for each flexibility, using the same data and materials as their previous article. The results show that enabling all flexibilities yields the same outcome as disabling operation flexibility.This system remains unaffected by the degree of flexibility in operations.

2.2.5 ALBP

Organizing an assembly line efficiently is a crucial challenge faced by manufacturers. The task involves allocating the total work required to produce a single unit among the various workstations along the line. This basic version of the problem, known as the simple assembly line balancing problem (SALBP), has been extensively studied by researchers and professionals in the field of operations research for nearly 50 years. In this article [BS06], the authors provide a thorough and up-to-date review of the SALBP research, with a particular focus on the latest significant and influential contributions that have shaped the field.

For centuries, from the pioneering days of Henry Ford to the cutting-edge era of Industry 4.0, flow-based assembly processes have been a cornerstone of mass production across diverse industries. At the heart of these systems lies the assembly line balancing problem – a fundamental optimization challenge that determines the efficient division of labor among the various stations along the production line. This paper [BSS22]presents a comprehensive review of the scientific literature on assembly line balancing, covering the advancements published since the last major reviews in 2006 and 2007. It delves into the essential stages of the decision-making process, exploring novel methods for effectively gathering the necessary (precedence graph) data, examining emerging problem variants and models, and highlighting the most significant algorithmic developments. Moreover, it outlines a prospective research agenda for the next fifteen years, charting the future direction of this crucial field.

Mass production through assembly lines has been a longstanding and appealing approach. Since the pioneering work of Henry Ford, the assembly line concept has evolved from rigid, single-model production to more flexible systems. These advancements include parallel workstations, customer-oriented mixed-model and multi-model lines, U-shaped configurations, and unpaced lines with intermediate buffers.Configuring an assembly line requires addressing the critical assembly line balancing problem. This involves distributing the total workload for manufacturing a product across the various workstations along the line. While early research focused on the simple assembly line balancing problem (SALBP) with restrictive assumptions, more recent work has aimed to describe and solve more realistic generalized problems (GALBP). This paper[BS06] provides an overview of the developments in GALBP research.

Assembly lines are specialized manufacturing systems that play a crucial role in the large-scale production of standardized goods. Interestingly, these assembly lines have also become important in the low-volume production of customized products, a process known as mass-customization. Given the significant financial investment required to set up or modify an assembly line, careful planning of its configuration is of paramount importance for businesses. This challenge has attracted the attention of numerous researchers, who have developed optimization models to support real-world configuration planning, known as assembly line balancing problems. Despite the substantial academic effort in this field, there remains a considerable gap between the requirements of actual configuration problems and the current state of research. To bridge this communication gap between researchers and practitioners, this article [BFS07]presents a classification scheme for assembly line balancing, which is a valuable step in identifying the remaining research challenges that can contribute to closing the existing gap.

In this article [KHBB20] The writers set out with minimizing three main goals:total completion time ,total production cost , and total energy consumption. They decided not to directly equate energy consumption with expenses due to the complexities of how energy prices fluctuate over time(TOU:Time of use). After researching different ways to make their processes more efficient and comparing various optimization techniques, they settled on using ϵ-constraint approach. By choosing this method, they aim to strike a good balance between getting things done faster and being more environmentally friendly. They hope that by optimizing their processes in this way, they can not only save time and energy but also contribute to a more sustainable and cost-effective operation in the long run.

This study [BDNS+23] addresses work task and resource allocations to work station with buffer capacity to maximize throughput, minimize total buffer capacity while considering the stochastic behavior of the system. To achieve this, they use a two-step method called sim-opt. In the first step, they employ FACTS Analyzer to simulate different scenarios, giving them a picture of how the system behaves under various conditions. This helps them understand where potential issues might arise and where improvements can be made. In the second step, they turn to NSGA-2, a well-known optimization technique, to find the best solutions. NSGA-2 is particularly useful for handling multiple objectives simultaneously, making it a good fit for this task. By using NSGA-2, they aim to strike a balance between maximizing productivity and minimizing the need for buffer space, considering the uncertainties in the system. Through this combined approach of simulation and optimization, they're able to tackle the complexities of task allocation and resource management in dynamic environments effectively. This not only improves productivity but also ensures that the system remains adaptable and resilient to unforeseen changes.

Many studies on balancing mixed-model assembly lines assume that a task common to multiple models must be performed at a single station. However, in this research[BR06], the authors chose to remove this limitation, enabling a shared task to be assigned to different stations for each model. Their goal was to minimize the combined costs of the stations and any necessary task duplication. An optimal solution method was developed using a backtracking branch-and-bound algorithm, and its effectiveness was evaluated through extensive experimentation. Additionally, a heuristic branch-and-bound-based approach was created to tackle larger-scale problems. The heuristic solutions were compared to a lower bound, and the results demonstrated that this approach provided significantly better outcomes than traditional techniques.

This piece [DK19] examines how mixed-model manufacturing systems address the needs of modern manufacturing, bringing together the benefits of mass production and the ability to create personalized products to meet individual customer requirements. The article explores various metaheuristics that can aid in task assignment, ultimately concluding that simulated annealing is the most effective option. Other methods discussed include Genetic Algorithm, Imperial Competitive Algorithm, and Particle Swarm Optimization. To compare these metaheuristics, the article looks at how they describe problem elements and evaluate potential solutions. The authors conclude that population-based metaheuristics rely on initial population allocation and progression direction, while classic algorithms may not be the best fit for complex multi-objective problems.

[BY21] delves into the type-2 mixed-model assembly line balancing, which involves determining the ideal cycle time for a specific number of workstations. To achieve this, the author recommends the use of a Genetic Algorithm as a greedy randomized adaptive search procedure to minimize cycle time by finding the optimal task assignment. To build the initial population, the author employs the GRASP method and uses RPW to ensure the feasibility of the solution. Local search is also utilized to enhance the constructed solution. Finally, the author compares the results of the executions with the final solutions obtained using the hybrid GRASP-GA. A mathematical formulation is presented, which is akin to the SALB-2 problem. A numerical example is also provided to support the author's proposed method.

The article [JW03] discusses the popularity of the goal chasing method for the mixed-model assembly line balancing problem. However, it has a flaw in that it uses up the "good" parts early on, leaving fewer options for the later stages. The author takes it upon himself to identify these "good" parts and sequences and analyze their relationship with the optimal solution. His solution is the "variance algorithm," which he supports with a numerical example. To begin, he provides the necessary mathematical formulation. He then details the goal chasing method and its shortcomings. He suggests three methods to address the problem: symmetry, horizon, and rate-preserving. The author discusses the issue of faulty units and suggests methods to prevent their occurrence. To support his argument, he presents a mathematical demonstration that utilizing the "variance method" can lead to faster and higher quality results.

The subject of this piece [ESI+11]is the integration of Tabu Search metaheuristic into the MMPAL-2 mixed model. The parallel assembly is utilized to distribute tasks to the various workstations. A heuristic algorithm generates an initial solution which is then passed to the TS to ascertain the minimum cycle time. The author provides a series of test problems to demonstrate the efficacy of this approach. The system in question differs from those previously discussed, as evidenced in the accompanying figure which depicts multiple parallel assembly lines featuring mixed-model products. The author then proceeds to outline the mathematical formulation. It can be concluded that the proposal has had a positive impact on the issue at hand, as evidenced by the fact that out of 87 test problems, 77 showed improvement when the proposed procedure was implemented.

The authors of [SV04] utilized a combination of a mathematical programming model and an iterative genetic algorithm-based approach to tackle the mixed-model assembly line problem (MALBP) with parallel workstations. Their objective was to maximize the production rate of the assembly line with a set number of operators. To accomplish this, they developed a 3-step process. The first stage involved using a constructive heuristic to solve the MALBP-2 and create an initial solution for the genetic algorithm. In the second stage, they applied the genetic algorithm procedure GA-1 to optimize the initial solution. Finally, in the third stage, they introduced genetic algorithm procedure GA-2 to fine-tune the solutions obtained from GA-1 by balancing the workloads within the workstations to ensure each operator performs roughly equal amounts of work for each model being assembled.

This study [YYO20] examines a single-model system, considering the hierarchy of work assignments, positional constraints, and the option to parallelize workstations, with task assignment restrictions. The primary goal is to minimize the costs associated with opening workstations and hiring workers, while finding the optimal solution for assigning tasks to stations and workers to each station and task. To address this challenge, the researchers developed an integer programming model with a single objective and 14 constraints. Recognizing the NP-Hard nature of the problem, they shifted to a more appropriate approach - Simulated Annealing. To generate an initial solution, they utilized a modified version of the Ranked Positional Weight heuristic. Additionally, they implemented a local search procedure to enhance the solution obtained from the previous meta-heuristic.

[RG17] delves into the intricacies of multi-worker assembly lines, where individuals collaborate on the same product within the same workstation. The challenge lies in determining the optimal number of items to produce, utilizing only the existing workstations without adding new ones. To address this, the authors propose a mixed-integer mathematical programming model that aims to minimize the cycle time and the number of workers required. Given the complexity of this problem, classified as NP-Hard, the researchers employ a metaheuristic approach. They explore two distinct strategies based on the simulated annealing algorithm: one that solves the problem directly (DSA) and another that tackles it indirectly (ISA). The study concludes that the DSA approach outperforms the ISA method in terms of both solution quality and computational efficiency.

In this article [BR14], the authors address the challenge of worker assignment and task balancing in an assembly line setting, where individual worker capabilities play a crucial role. To tackle this problem, they present a Mixed-Integer Programming model that aims to maximize production rates by optimizing the allocation of workers to stations and tasks. Given the complexity of the problem, which is classified as NP-Hard, the researchers propose a heuristic approach based on beam search and a task-oriented branch-and-bound procedure. Through extensive numerical analysis and comparison, the authors demonstrate that their proposed heuristic outperforms some existing methods. However, they also acknowledge that the branch-and-bound method has limitations, as it can only efficiently handle instances with a relatively small number of tasks. Recognizing the strengths of both approaches, the authors conclude that the best strategy is to combine their proposed solutions with the method developed by Vila And Pereira (BBR), effectively leveraging the advantages of multiple techniques to address this challenging optimization problem.

In this research [Per18], the scientists tackle a widespread issue faced by various industries - the need to balance an assembly line for multiple products while accounting for resource demands. The goal is to minimize the total cost of the workforce and resources required. The problem is inherently complex, classified as NP-Hard, so the researchers propose a heuristic approach that combines the Hoffman heuristic and Estimation of Distributions Algorithm. This hybrid method delivers excellent results, generating high-quality solutions while providing a clear framework to address the challenges of heterogeneity and multiple resources in assembly line balancing problems.

This research focuses [RE15] on the intricate challenge of managing a mixed-model assembly line, which involves the simultaneous production of various products on the same line. The study addresses the problem of assigning workers to workstations, taking into account their skills and the associated operating costs. The researchers have developed a model that aims to achieve two primary objectives: minimizing the overall cycle time and minimizing the operating costs related to the workforce. Given the high complexity of this problem, the researchers have proposed an alternative approach - the Imperialist Competitive Algorithm (ICA). This algorithm starts with a random initial population and then generates supervised-random solutions in an effort to reach more optimal results. Through a numerical example, the study has demonstrated the efficiency of the proposed ICA method, showcasing its ability to achieve near-optimal outcomes.

In this article [TLZ+16], the focus is on a unique manufacturing system - the mixed-model assembly line with sequence-dependent tasks. The problem is modeled with two key objectives: reducing cycle time and minimizing workload variation. First, the authors simplify the problem by combining the sequence-dependent connections and precedence relations into a single precedence graph, effectively transforming it into a single-model assembly line balancing problem. To address the complexity of this revised problem, the authors propose using a Genetic Algorithm approach. For their numerical example and discussion, they rely on an initial solution provided as input to the GA algorithm, selecting three key factors to base their study on: processing times, the number of immediate successors, and the number of updated tasks for each task.

2.2.6 Our reference articles

Modular manufacturing systems (MMS) aspire to attain exceptional operational agility by incorporating a flexible product flow between stations using automated guided vehicles, and by furnishing redundant resources for each task, thereby eliminating consistent cycle times and the sequential arrangement of stations. This study [SWS21]explores the design of MMS with an economic goal in mind. They construct a mixed-integer program to design MMS. By presenting a numerical case, the authors showcase the efficacy of their approach and identify future research prospects.

This study [HG19] examines the initial configuration of such systems. The flexible layout design problem (FLDP) involves designing a flexible layout for a segment of the assembly of diverse vehicles. It combines station formation and station location issues. Additionally, the FLDP anticipates the operational flow allocation of the automated guided vehicles. The researchers formulate the FLDP as a mixed-integer linear program and develop a decomposition-based solution approach that can optimally solve small to medium-sized instances. Furthermore, they transform this solution approach into a matheuristic that generates high-quality solutions in a reasonable time for large-sized instances. They compare the efficiency of flexible layouts to mixed-model assembly lines and quantify the benefits of flexible layouts, which increase with vehicle heterogeneity.

2.3 Conclusion

In this chapter, we reviewed research papers related to the Flexible Layout Design Problem (FLDP).We examined the state-of-the-art in simple line balancing and discussed the Dedicated Manufacturing Systems (DMS) and Flexible Manufacturing Systems (FMS) problems. We presented the Reconfigurable Manufacturing Systems (RMS) problems and summarized them in tables.

Finally, we reviewed articles on Matrix-structured Manufacturing Systems (MMS) problems and assembly line balancing issues.
These articles led us to study the FLDP problem integrated with resource allocation problems.

Chapter 3

Considered problem and proposed methods

3.1 Introduction

In this chapter, we dive into the adjustments and implementation of the model originally developed by [HG19]. Hottenrott's work solely focused on the task assignment problem and the minimization of cycle time, which is the duration needed to finish a set of tasks or a production cycle. Building on this foundational model, we were inspired by the research conducted by [SWS21] on the economic aspects of resource distribution. We recognized the need to expand the original model to include the dynamic allocation of resources, as this is a crucial element in the real-world implementation of such production systems.

By integrating resource allocation into the problem, we aim to create a more comprehensive and realistic model that can better address the challenges faced by modern manufacturing and service organizations. The distribution of resources, such as labor, equipment, and materials, significantly impacts the overall efficiency, productivity, and cost-effectiveness of the production process.

3.2 Model

In this section, we detail the mixed-integer linear program for the FLDP where they investigate a segment of the final assembly at an automotive OEM in which highly variant, manual tasks are performed. The purpose is to derive layouts that allow for the efficient assembly of a given model mix.At first we have the set of model $m \in M$ that needs to be processed, $t \in T_m$ the set of tasks for each model that has a starting task T^S and an ending task T^E, $V_{m,t}$ the set of successor tasks that shows wich task needs to be done for the next one to begin for each model m, the set of routes R that represents all the possible routes which the AGV can take,the layout is represented by $l \in L$ locations at which stations could be opened as resumed in table 3.1. We decide whether a station is opened at a location (variables X_l) and which tasks are assigned (variables $\mathrm{Y}_{t,l}$). Also, anticipate the models' flow allocations along the positions $i \in I_r$ of the routes (variables $\mathrm{Z}_{m,r,t,i}$).The objectives are to minimize the

number of opened stations as well as to minimize the flow intensity. Common entry and exit points are required, the exact locations of the entry and exit points are determined in the FLDP. The layout is designed for a given model mix and volume that should be produced in a given production time.
We use the notation as summarized in Tables 3.1, 3.2, 3.3.

3.2.1 Model Notation

Index sets	
m $\in M$	Models
t $\in T$	Tasks
$[T^S, T^E] \subset T$	Dummy start and end tasks
T $\backslash$[T$^S, T^E$]	Real tasks
Tm $\subset T$	Tasks for model m
t2 $\in Vm, t$ . . .	Successor tasks of task t for model m(precedence relations): t to be finished before t2 starts
l $\in L$	Locations
r $\in R$.	Routes: potential AGV flow paths
i $\in Ir$	Position index on route r:i=1......$\|r\|$

Table 3.1: Index sets

Parameters	
w_r	Distance on route r
d_m	Demand for model m
$q_{m,t}$.	Task time of task t for model m
τ	Production time
b_t . .	Maximum number of duplicates of task t (b_{T^S}=b_{T^E}=1)
e_l	Level index of location l
f_l	Row index of location l
$p_{r,l}$	Position index of location l on route r

Table 3.2: Parameters

Decision Variables	
X_l	1 if station at location l is opened , else 0
.	
$Y_{t,l}$	1 if task t is assigned to location l,else 0
.	
$Z_{m,r,t,i}$	Units of model m that receive task t at i th location on route r

Table 3.3: Decision variables

3.2.2 Model Formulation

$$Min\ Z_1 \ = \sum_{l \in L} X_l \tag{3.1}$$

$$Min\ Z_2 \ = \sum_{r \in R} \sum_{m \in M} w_r * \ Z_{m,r,T^S,1} \tag{3.2}$$

$$\sum_{l \in L} Y_{t,l} \leq b_t \qquad \forall\ t \in T \tag{3.3}$$

$$\sum_{t \in T} Y_{t,l} \ \leq |T| * \ X_l \qquad \forall \ \ l \in L \tag{3.4}$$

$$\sum_{m \in M} \sum_{t \in T_m} \sum_{r \in R | l \in L} q_{m,t} * \ Z_{m,r,t,p_{r,l}} \ \leq \tau \qquad \forall\ l \in L \tag{3.5}$$

$$\sum_{r \in R} Z_{m,r,0,0} \ = \ d_m \qquad \forall\ m \in M \tag{3.6}$$

$$\sum_{i \in I_r} Z_{m,r,t,i} \ = \ Z_{m,r,0,0} \qquad \forall\ m \in M,\ r \in R,\ t \in T_m \backslash T^S \tag{3.7}$$

$$Z_{m,r,t_2,i} \ \leq \sum_{j \in I_r | j \ \leq \ i} Z_{m,r,t,j} \qquad \forall\ m \in M, r \in R, t \in T_m, t_2 \in V_{m,t}, i \in I_r \tag{3.8}$$

$$\sum_{m \in M | t \in T_m} \sum_{r \in R | l \in r} Z_{m,r,t,p_{r,l}} \ \leq \ Y_{t,l} * \sum_{m \in M} d_m \qquad \forall\ t \in T,\ l \in L \tag{3.9}$$

$$\sum_{l_2 \in L | e_{l_2} \ < \ e_l} Y_{t_2,l_2} \leq |L| * (1 - Y_{t,l}) \qquad \forall\ m \in M,\ t \in T_m,\ t_2 \in V_{m,t},\ l \in L \tag{3.10}$$

$$\sum_{l \in L | f_l = e_2} X_l \leq |L| * \sum_{l \in L | e_l > e_2} X_l \qquad \forall\ e_2 \in 1, \ldots, \max_{l \in L}\ e_l \tag{3.11}$$

$$X_l \in \{0, 1\} \quad \forall\ l \in L \tag{3.12}$$

$$Y_{t,l} \in \{0, 1\} \qquad \forall\ t \in T,\ l \in L \tag{3.13}$$

$$Z_{m,r,t,i} \ \geq \ 0 \qquad \forall\ m \in M,\ r \in R,\ t \in T_m,\ i \in I_r \tag{3.14}$$

We employ a lexicographic multi-objective formulation as shown in Eqs. 3.1 and 3.2.
The objectives are ranked in lexicographic optimization based on their importance.

The optimization process starts with the most important objective, optimizing it without considering the values of the subsequent objectives. Once the first objective is optimized, its optimal value becomes a constraint for the next objective, and so on.

The primary objective (3.1) is to minimize the number of opened stations. Minimizing the number of opened stations is equivalent to maximizing the efficiency of the layout because demand and production time are already set. The efficiency is defined as:

$$\text{Efficiency} = \frac{\text{Total workload}}{\text{Installed capacity}} = \frac{\sum_{m \in M} d_m \cdot \sum_{t \in T_m} q_{m,t}}{\tau \cdot \sum_{l \in L} X_l}$$

As the subordinate objective (3.2), we minimize the flow intensity for the minimum number of stations. Flow intensity represents the transportation effort and is calculated as the sum product of route distance and the number of vehicles assembled along a route.

Constraints (3.3) limit the maximum number of task duplicates, except for the dummy tasks T^S and T^E which exist only once. Constraints (3.4) guarantee that tasks can only be assigned to stations that have been opened. Constraints (3.5) ensure the workload allocated to a station does not exceed the production time. Constraints (3.6) guarantee the fulfillment of demand and constraints (3.7) maintain the balance of flow. Constraints (3.8) satisfy the precedence relations by ensuring that the number of vehicles receiving the successor task at a location cannot be higher than the number of vehicles receiving the predecessor task at all preceding locations along a specific route. Constraints (3.9) link the binary assignment variables $\mathrm{Y}_{t,l}$ to the continuous flow variables $\mathrm{Z}_{m,r,t,i}$ allowing positive flow only when the corresponding task is assigned to the corresponding location. Constraints (3.10) enforce full routing flexibility by forbidding any duplicate of a task t2 that is a successor of task t for any model m to be assigned to the left of any duplicate of task t. Essentially, they guarantee that all duplicates of the successor task t2 are reachable from any duplicate of the predecessor task t by the AGVs, which cannot travel backward. This enhanced flexibility is especially beneficial in the event of disruptions. We therefore refer to constraints (3.10) as robustness constraints. Constraints (3.11) ensure the layout is longer than its width. These constraints permit the use of locations only if their row index is less than the level index of the last used location on the right. Finally, in constraints (3.12)–(3.14), we restrict the domains of the decision variables.

3.2.3 Proposed Method

In our optimization model, we have customized the initial formulation by including the concept of resource allocation to each station and task. This represents a crucial extension to the basic model, as it allows us to capture the practical reality of resource constraints and their impact on the overall system performance.

The key addition to the model is the introduction of a new binary decision variable, denoted as $W_{rs,l}$. This variable takes on a value of 1 if a specific resource rs from the set of available resources R_s is assigned to a particular location l, and 0 otherwise. By incorporating this decision variable, we can now model the available resources, their assignment to various stations, and the execution of tasks at those stations. We also have the set of parameters A_{rs} which represents the set of tasks that the ressource rs can execute.

By including the $W_{rs,l}$ variable, the model can now determine the most efficient allocation of resources to different stations, taking into account factors such as the specific requirements of each task, and the overall system objectives,in our case we consider that the resources are reconfigurable, meaning that we can use them for doing multiple tasks and for multiple stations, if the same resource is needed for two diffrent stations then it is considered bought two times , each resource for each station .This level of detail allows the optimization process to generate more realistic and practical solutions that can be effectively implemented in the actual operational environment.

The incorporation of resource allocation into the model is a significant step forward, as it enables a more comprehensive and realistic representation of the underlying system. This, in turn, leads to improved decision-making capabilities and the development of more robust and effective optimization strategies for the problem at hand.

First we change the second objective (3.2) where we add minimizing the cost of allocated resources (Cr) and the cost of flow intensity (Ct) as shown in Eq (3.15):

$$Min\ Z_2 = \sum_{r \in R} \sum_{m \in M} w_r * \quad Z_{m,r,T^S,1} * Ct + Cr * \sum_{rs \in Rs} \sum_{l \in L} W_{rs,l} \tag{3.15}$$

And we add the constraint needed as well:

$$Y_{t,l} \leq \sum_{rs \in Rs | t \in A_{rs}} W_{rs,l} \quad \forall\, t \in T,\ l \in L \tag{3.16}$$

$$W_{rs,l} \in \{0,1\} \qquad \forall\, rs \in Rs,\ l \in L \tag{3.17}$$

Constraints (3.16) link the binary variables $Y_{t,l}$ to $W_{rs,l}$. This ensures that a task is allocated to a location that has at least one resource to perform the task. Constraints 3.17 define the nature of the variables $W_{rs,l}$.

3.3 2-Phased Resolution Method

In this section, we present a unique resolution approach that effectively addresses the challenges of resource assignment without directly integrating the resource extension into the underlying mathematical model. This method, named the "2-phased resolution method" consists of two distinct yet complementary phases that work together to provide a comprehensive solution.It is also considered as a matheuristic approach which is a hybrid optimization approach that combines elements of mathematical programming with heuristic methods to solve complex optimization problems efficiently.

The first phase involves developing the fundamental mathematical model. This model serves as the foundation, without capturing the core constraints and objectives of the resource assignment problem. By focusing on this essential model, we can ensure that the core problem is addressed in a thorough and structured manner, laying the groundwork for the subsequent phase.

The second phase of our 2-phased resolution method introduces a heuristic approach for assigning resources to the various locations. This phase utilizes the well-established optimization technique of set covering to efficiently allocate resources based on the insights gained from the basic mathematical model. By separating the resource assignment process from the core model, we can explore a wider range of heuristic strategies, potentially leading to more effective and flexible resource allocation solutions.

The mathematical foundation serves as the solid basis, while the intricate resource extension needs are also addressed. This divide between the primary model and supplementary requirements allows for more flexibility in the resource allocation process, enabling the investigation of diverse heuristic techniques without adding undue complexity to the core model.

3.3.1 Motivation

We are going to outline our own proposed approach in two distinct phases, where we allocate resources to the stations and tasks using our own method, rather than integrating it into the MILP. This approach provides us with greater flexibility and control over the resource allocation process, allowing for more customized and optimized solutions.

The initial phase involves running the MILP without any resources, essentially executing the Grunow model as-is. This step serves as a crucial precursor to our second phase, as it provides us with a baseline understanding of the problem and the underlying task structure. By executing the Grunow model without any resources, we can gain valuable insights into the inherent complexities and constraints of the problem, which will inform our subsequent resource allocation strategy.

Once we have the results from the initial phase stored in an Excel sheet, we can move on to the subsequent phase, which entails implementing a set-covering approach to assign the necessary resources to the identified stations and their respective tasks. This approach allows us to carefully analyze the resource requirements for each station and task, and then devise an optimal way to allocate the available resources to meet these requirements. By using a set-covering approach, we can ensure that all the necessary tasks are covered by the assigned resources, while also considering factors such as resource availability, task priorities, and cost-effectiveness.

The set-covering approach involves several steps, including identifying the resource requirements for each station and task, determining the available resources and their capabilities, and then developing an optimization model to assign the resources in the most efficient manner.

3.3.2 Set-Covering Notation

For the set-covering problem, we require mathematical program notation. We define the set of resources $rs \in Rs$,for the location $l \in L$ and for the tasks $t \in T$. As for the parameters used are A_{rs} representing the group of tasks each resource rs can perform, Y_l representing the tasks allocated to each station l which is the result from phase 1. As for this problem it has one decision variable $X_{r,l}$ which equals 1 if resource rs is allocated to station l otherwise 0 as resumed in Table 3.4.

Index set	
rs $\in Rs$	Resources
l $\in L$	Locations
t $\in T$	Tasks

Parameters	
A_{rs}	Set of tasks which resource rs can do
.	
Y_l	Which tasks t are in location l

Decision Variables	
$X_{rs,l}$	1 if resource rs is allocated to station l, 0 else

Table 3.4: Model notation for set covering

3.3.3 Set-Covering model

$$Min\ X = \sum_{rs \in Rs} \sum_{l \in L} X_{rs,l} \tag{3.18}$$

$$\sum_{rs \in Rs | t \in A_{rs}} X_{rs,l} \geq 1 \quad \forall l \in Y | t \in Y_l \tag{3.19}$$

By adopting this two-phase approach, we can leverage the strengths of both the MILP and the set-covering approach, ultimately leading to a more comprehensive and effective resource allocation strategy. The initial phase provides us with a solid foundation for understanding the problem, while the subsequent phase allows us to fine-tune the resource allocation to better suit our specific needs and preferences. The objective function Eq 3.18 aims to minimize the number of stations used. Constraint 3.19 to ensure availability of resource for task t.

3.4 Data Sets

To assess our techniques and determine the more efficient one, we have selected a data set to walk through our numerical example and discussion. We have used the same data set that [HG19] employed for their approach and numerical example for data related to models (task time, successors...). For location-related data (routes, distances...), we relied on [BGK23].

While all our cases share the same number of locations (9), the same set of resources R (20 in total), and the same set of routes (62 possible for the demand to traverse our mms layout), they also have the same number of models (5 per case).In these cases, Grunow has incorporated his own understanding, varying the percentage change from one case to the next. For C_r, we will select the value of 40000 as referenced in [SWS21], and for C_t, we will choose a cost of 1.

To generate instances with diverse vehicles, we consider four levels of structural heterogeneity (sh): 0%, 10%, 25%, and 50%. Structural heterogeneity refers to the degree of dissimilarity in the precedence graph structures of the models. It is defined as the average percentage difference between the minimum supergraph's task count and the precedence graphs of all models.

In the base case, where model-task assignments remain unchanged (sh = 0%), all models must handle every task. Consequently, the precedence graphs of the models are identical and match the data set from Scholl (1993). However, for the case where 25% of the assignments are randomly removed (sh = 25%), we ensure that each task is still required by at least one model and that each model needs a minimum of two real tasks. We also consider four levels of task time heterogeneity (tth), where the task times for each model can deviate from the demand-weighted mean task time by 0%, ±10%, ±25%, and ±50%. In the base case (tth = 0%), all models have the same task times, but for tth = 25%, the task times are chosen to be within the range of 0.75 and 1.25 times the demand-weighted mean task time, maximizing the heterogeneity. Importantly, the demand for each model and the overall workload for each task remain constant across all structure and task time heterogeneity levels to ensure comparable instances.

We have used matplotlib python's library to draw a precedence diagram to visualize the precedence relationships between tasks for a given model, as shown in figure 3.1.

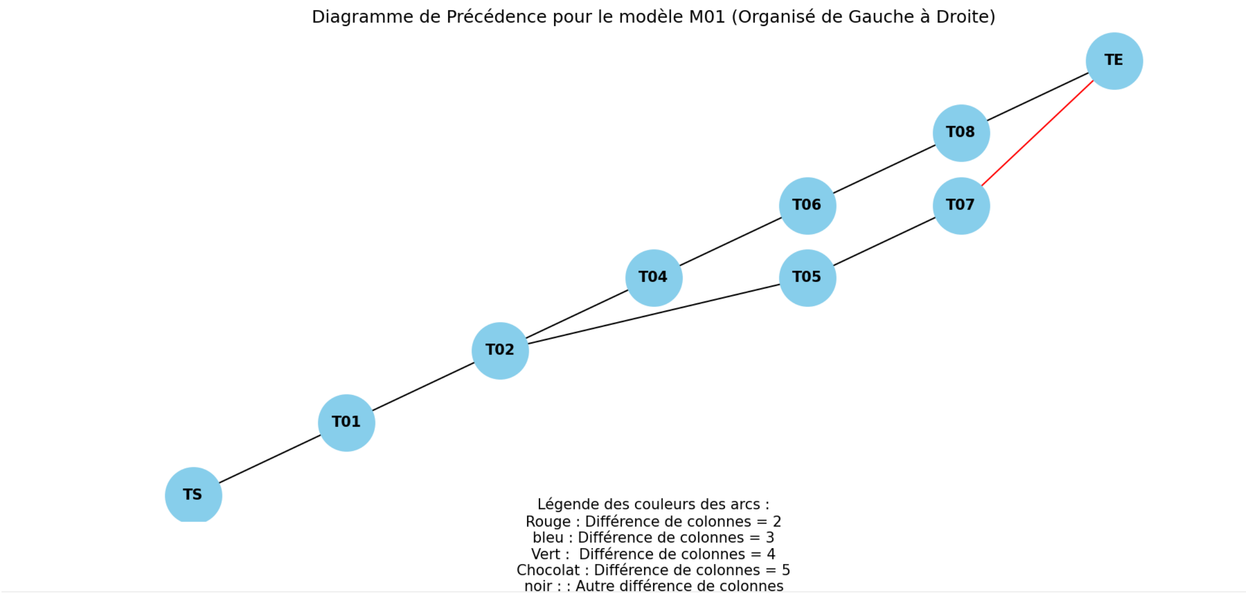

Figure 3.1: An example of a precedence diagram

3.5 Conclusion

Throughout this chapter, we have explored the FLDP with its two objectives: minimizing the number of opened stations and optimizing flow transportation. We have identified that the new FLDP integrated with resources presents an NP-hard problem. To tackle this challenge, we have introduced a 2-phased heuristic aimed at reducing computational time for solving this problem.

In the next chapter, we will discuss the results of the integrated model and the 2-phased approach.
m Repeter ce qu'on a fait dans ce chapitre (une petite intro sur le model, le travail proposé et la discussion sur le resultat assumptions

Chapter 4

Results and discussions

4.1 Introduction

In this chapter, we will discuss the results obtained for all the datasets mentioned in Chapter 3. Each dataset has a specific number of tasks: Bowman has 8 tasks with a cycle time of 19, EX has 5 tasks with cycle times of 18 and 20, Jackson has 11 tasks with a cycle time of 8, Jaeschke has 9 tasks with a cycle time of 7, Mansoor has 11 tasks with a cycle time of 47, Mertens has 7 tasks with a cycle time of 8, and Mitchell has 21 tasks with a cycle time of 14. Each dataset also has its own configurations of ov and tth, we note that in our data set ov=sh and ttv=tth.

Our solution approaches and the instances are implemented in a program application written in Python 3 and interfaced with Gurobi 11. All experiments are run on a computer using an AMD Ryzen 7 5700U with Radeon Graphics processor with 1.8 GHz and 16 GBRAM.

4.2 Results(1200 seconds)

In this section, we are going to talk about the results obtained from simulating our two approaches for 20 minutes. The reason for this choice of time is that we noticed the UB has converged rather quickly for some instances, and so that will be our focus in the next results and discussions.

To substantiate our claim, we selected a random instance "MITCHELL c=14 ov=25 ttv=25 s=00" as depicted in Figure 4.1. The figure illustrates that our primary objective (UB) begins to stabilize near the 20-minute mark, with marginal further improvement that may be insignificant to our analysis. Additionally, it demonstrates that the lower bound (LB) remains stable for a considerable duration well before the 20-minute threshold, after which it steadily progresses closer to the UB, thereby reducing the gap between the two results.

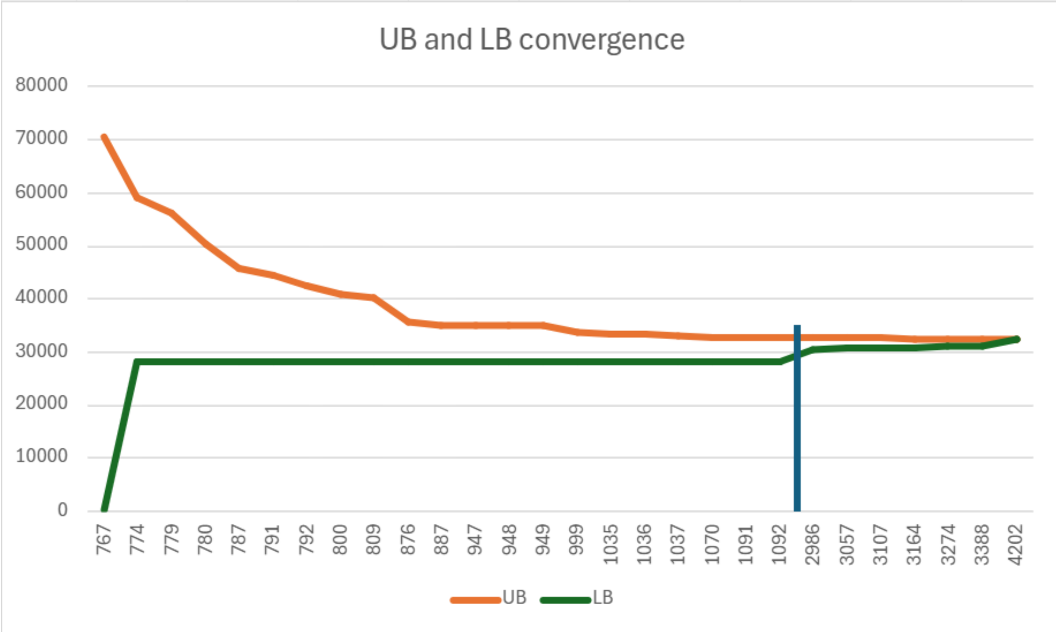

Figure 4.1: UB and LB Convergence

The fact that we also chose to compare the two approaches based on the CPU time and the GAP from LB (figure 4.2) , which is calculated by:

$$GAP\ from\ LB = \frac{UB - LB}{UB} \tag{4.1}$$

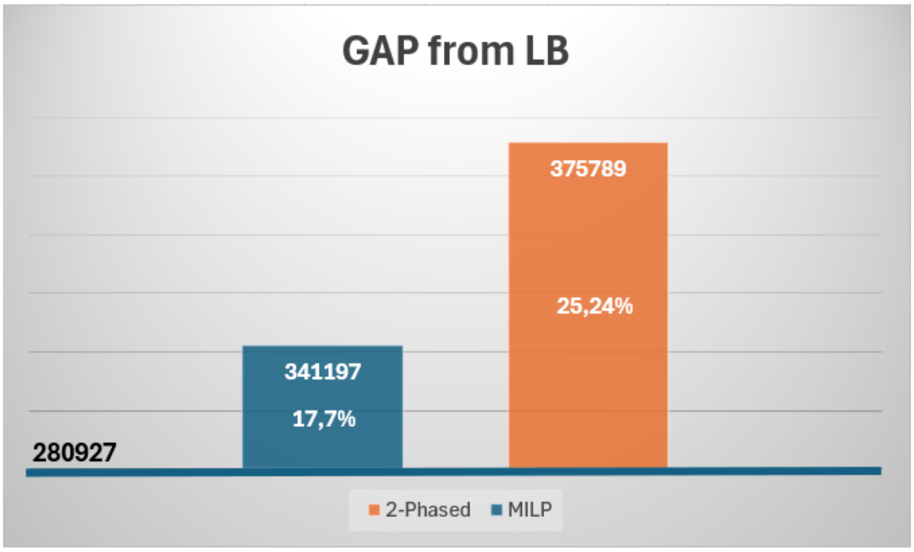

Figure 4.2: GAP From LB

4.2.1 A description of the result obtained

We will use the Matplotlib library in Python to draw and visualize the solutions without taking into account resources, and the variation of the objectives. We take this instance as an example: JACKSON c=08 ov=00 ttv=10 s=00 to explain the form of the results (Figures 4.3, 4.4).

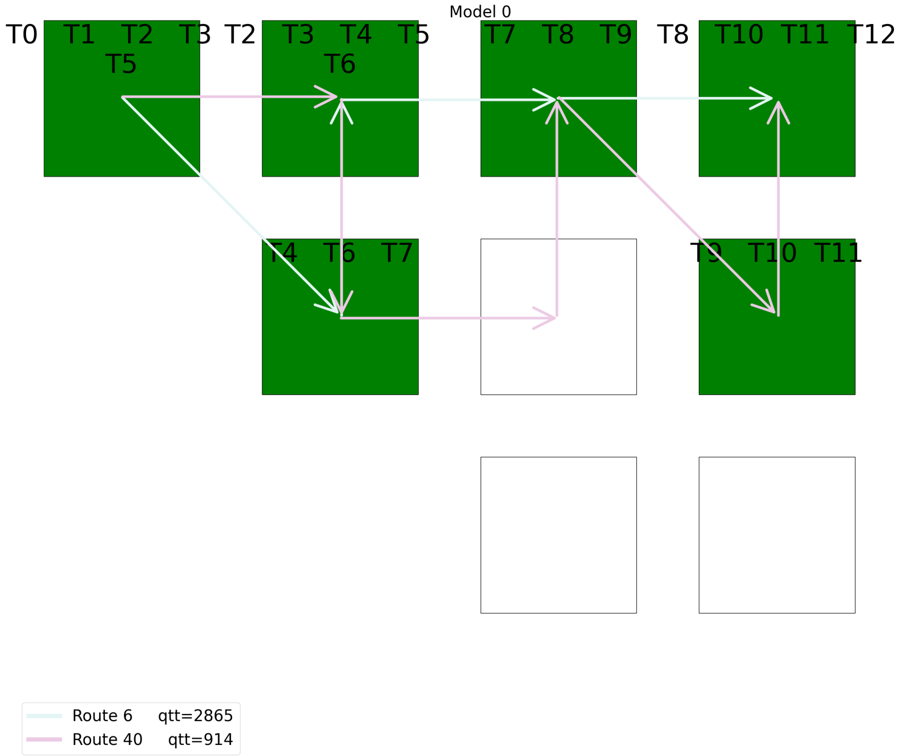

Figure 4.3: Result example

Locations in green are the opened locations. For this example, we have 6 opened locations. In each location, we have the set of tasks for this model (for example, model 0 at location 0 includes tasks: {T0, T1, T2, T3, T5}). The flow of products is represented by arrows, with each color indicating a different route. The quantity allocated to each route is shown in the picture legend.

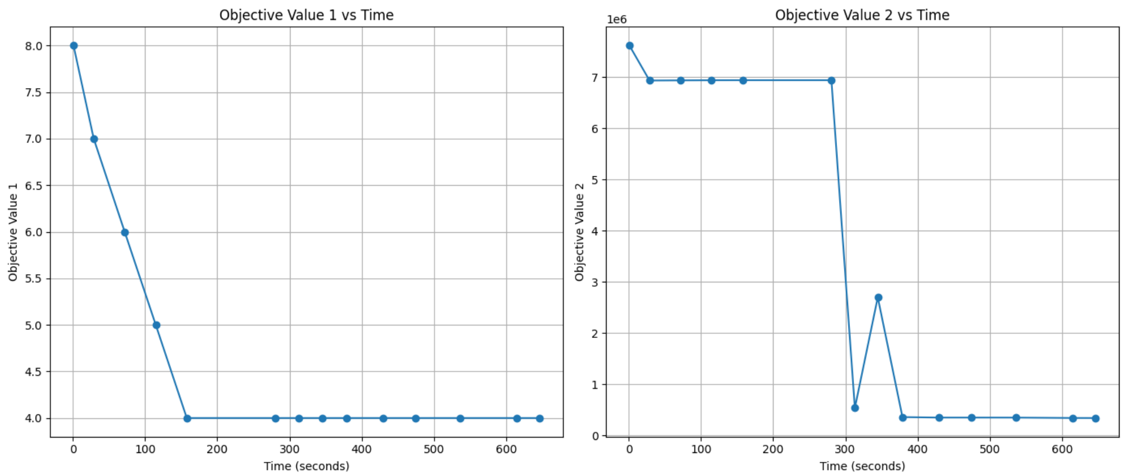

Figure 4.4: The objective variation curves example

As seen in Figure 4.4, the optimization begins with the first objective without considering the value of the second objective. The latter undergoes oscillations before the first objective completes optimization (around 350 seconds), after which it begins its minimization process.

To access all the 128 instances results, check this drive 1.

4.2.2 Bowman C=19, 8 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	34,20	1200	39,42	392
	10	14,70	1200	21,41	203
	25	24,80	1200	30,76	314
	50	24,00	1200	30,57	383
	Average	24,43	1200	30,54	323
10	00	33,00	1200	38,24	95
	10	22,50	1200	28,66	284
	25	33,70	1200	38,89	99
	50	33,90	1200	39,57	173
	Average	30,78	1200	36,34	162,5
25	00	31,20	1200	37,50	113
	10	26,00	1200	32,77	94
	25	29,30	1200	35,80	235
	50	17,70	1200	25,24	206
	Average	26,05	1200	32,83	162
50	00	5,23	1200	30,25	139
	10	4,05	1199	28,28	162
	25	15,50	1200	24,48	182
	50	9,18	1200	18,84	53
	Average	8,49	1199,75	25,47	134

Table 4.1: Bowman GAP from LB for MILP and 2 phased (1200s)

4.2.3 EX C=18, 5 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	00	118	15,87	30
	10	00	142	18,46	68
	25	00	133	13,09	31
	50	00	143	18,46	71
	Average	00	134	16,47	50
10	00	00	122	18,46	51
	10	00	159	00	69
	25	00	122	18,46	51
	50	00	145	15,87	31
	Average	00	137	13,19	50,5
25	00	00	164	18,46	52
	10	00	174	8,41	33
	25	00	150	00	51
	50	00	161	18,46	52
	Average	00	162,25	11,33	47
50	00	00	174	00	71
	10	00	173	18,46	52
	25	00	162	00	52
	50	00	151	18,46	52
	Average	00	165	9,23	56,75

Table 4.2: EX C=18 GAP from LB for MILP and 2 phased (1200s)

4.2.4 EX C=20, 5 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB(%)	CPU(s)	Gap from LB(%)	CPU(s)
00	00	00	99	19,70	40
	10	00	98	19,70	38
	25	00	111	19,70	39
	50	00	100	20,08	25
	Average	00	102	19,79	35,5
10	00	00	121	19,70	41
	10	00	140	19,70	57
	25	00	132	1,38	26
	50	00	140	19,70	41
	Average	00	133,25	15,12	41,25
25	00	00	123	2,60	26
	10	00	125	00	42
	25	00	134	19,70	43
	50	00	125	19,70	42
	Average	00	126,75	10,5	38,25
50	00	00	123	19,70	42
	10	00	123	1,91	26
	25	00	129	19,70	42
	50	00	122	0,95	26
	Average	00	124,25	10,56	34

Table 4.3: EX C=20 GAP from LB for MILP and 2 phased (1200s)

4.2.5 Jackson C=08, 11 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	**CPU (s)**	**Gap from LB (%)**	**CPU (s)**
00	00*	—	1200	—	1200
	10	43,98	1200	53,25	1022
	25	42,51	1200	54,36	1187
	50	39,63	1200	53,08	1094
	Average	42,04	1200	53,56	1101
10	00*	—	1200	—	1168
	10	39,95	1194	54,82	879
	25	40,23	1200	54,39	666
	50	49,33	1200	50,57	925
	823,33	43,17	1198	53,26	1200
25	00	42,04	1200	52,03	578
	10	40,24	1199	55,16	826
	25	43,05	1200	54,29	561
	50	42,03	1200	53,86	470
	Average	41,84	1199,75	53,83	608,75
50	00	41,85	1278	56,83	329
	10	45,90	1200	56,56	415
	25	42,09	1200	57,42	369
	50	46,76	1200	57,02	330
	Average	44,15	1219,5	56,95	360,75

Table 4.4: Jackson GAP from LB for MILP and 2 phased (1200s)

4.2.6 Jaeschke C=07, 9 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00*	—	1200	—	1200
	10	30,77	1200	47,75	506
	25	26,17	1200	49,61	488
	50	20,85	1200	45,61	631
	Average	25,93	1200	47,65	541,66
10	00	26,83	1200	53,12	1035
	10	25,09	1195	47,58	659
	25	19,67	1182	40,63	722
	50	24,82	1200	47,51	583
	Average	24,10	1194,25	47,21	749,75
25	00	26,97	1200	51,11	561
	10	23,32	1199	48,67	319
	25	26,80	1200	49,81	264
	50	23,49	1200	50,84	480
	Average	25,14	1199,75	50,10	406
50	00	7,56	1200	41,15	374
	10	6,21	1200	36,34	342
	25	5,60	1200	35,94	159
	50	28,26	1200	53,42	382
	Average	11,90	1200	41,71	314,25

Table 4.5: Jaeschke GAP from LB for MILP and 2 phased (1200s)

4.2.7 Mansoor C=47, 11 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	26,77	1200	50,86	642
	10	26,80	1200	46,48	405
	25	26,96	1200	50,99	489
	50	26,40	1200	46,19	567
	Average	26,73	1200	48,63	525,75
10	00	30,40	1200	49,11	556
	10	30,25	1200	49	512
	25	30,48	1200	49,17	529
	50	30,27	1200	53,21	569
	Average	30,35	1200	50,12	541,5
25	00	34,29	1200	41,47	212
	10	34,12	1200	51,83	229
	25	34,02	1200	51,76	360
	50	34,06	1200	51,79	187
	Average	34,12	1200	49,21	247
50	00	48,28	1200	62,19	62
	10	38,21	1199	58,54	65
	25	49,10	1200	66,85	57
	50	47,07	1200	61,30	230
	Average	45,66	1200	62,22	103,5

Table 4.6: Mansoor GAP from LB for MILP and 2 phased (1200s)

4.2.8 Mertens C=08, 7 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	00	630	27,43	236
	10	00	586	20,13	166
	25	00	1007	20,13	222
	50	00	925	27,37	209
	Average	00	787	23,76	208,25
10	00	00	575	20,13	175
	10	00	560	27,43	85
	25	00	610	27,43	174
	50	00	658	20,13	191
	Average	00	600,75	23,78	156,25
25	00	00	762	27,43	153
	10	00	595	20,13	136
	25	00	605	11,19	132
	50	00	628	27,43	124
	Average	00	647,50	21,54	136,25
50	00	00	470	27,43	117
	10	00	617	11,19	117
	25	00	470	11,19	121
	50	00	662	20,13	108
	Average	00	554,75	17,48	115,75

Table 4.7: Mertens GAP from LB for MILP and 2 phased (1200s)

4.2.9 Mitchell C=14, 21 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB(%)	CPU(s)	Gap from LB(%)	CPU(s)
00	00*	—	1200	—	1200
	10*	—	1200	—	1200
	25*	—	1200	—	1200
	50*	—	1200	—	1005
10	00*	—	1200	—	1200
	10*	—	1200	—	1069
	25*	—	1200	—	1156
	50	73,10	1200	61,73	1011
	Average	73,10	1200	61,73	1011
25	00*	—	1200	—	1201
	10	56,61	1177	66,74	885
	25	62,91	1200	65,26	810
	50	57,45	1200	64,63	821
	Average	58,99	1192,33	65,54	838,66
50	00	63,05	1196	69,95	625
	10	61,76	1200	70,56	1101
	25	59,59	1200	69,59	443
	50	57,98	1200	69,61	739
	Average	60,59	1199	69,92	652

Table 4.8: Mitchell GAP from LB for MILP and 2 phased (1200s)

Our data set was limited to 128/528 instances due to the restricted number of locations and routes available. We will now dive into a deep discussion between the 2 methods with the 128 instances.

From Bowman C=19 instances (Table 4.1), we can see that as the structure heterogeneity (sh) increases, the optimization converges more rapidly. The difference between the 2-phased GAP and the MILP GAP values is (6.11%, 5.56%, 6.78%, 16.98%) for sh=00, 10, 25, 50 respectively, with an average difference GAP of 8.86%. We notice that the less similar the structure of the models, the better the MILP performs. The 2-PHASED method gives us satisfactory results for small variations in the structure of the models (sh). Task time heterogeneity (tth) hardly affects the GAP. The solution time for the 2-phased method is almost 1/4 of the MILP time. For instances with 5 tasks, without a doubt, the MILP is better than the 2-phased method, with CPU time less than 130 seconds. As we can see in Tables 4.2 (Tc=18) and 4.3 (Tc=20), the 2-phased method did not give us good results. This is because in phase 1, we focused only on flow transport. From Jackson's Table 4.4 (Tc=8), we can see that changing SH and TTH does not influence the GAP. For TTH=50%, the 2-phased method solved within less than 10 minutes ($\leq$ 600s).In Jaeschke's Table 4.5, the 2-phased method is not suitable for 9 tasks and Tc=7. The difference between the two GAPs is greater than 20%. Mansoor's Table 4.6 (Tc=47)

showed that as SH increases, the GAP also increases. This means with 9 tasks and a high cycle time, the GAP highly depends on SH values. The MILP in Mertens' Table 4.7 is better than the 2-phased method. However, the 2-phased method gives us near-optimal solutions in less time (600s). In Mitchell's Table 4.8 (Tc=14), the GAP was elevated for both methods. For sh=00 and sh=10, the MILP method did not have enough time to optimize the second objective within 1200 seconds. The 2-phased method is better than MILP with low changes in structure heterogeneity. For example, with 5 types of vehicles, if the models are nearly similar for all vehicles, the 2-phased method can solve the problem effectively. The 2-phased method gives us satisfactory results for sh=25 and sh=50, with a difference in GAP of less than 10%.
To summarize, the MILP method remains better to the 2-phased approach for a small number of tasks ($\leq$ 8 tasks). The 2-phased method provides satisfactory results for more than 8 tasks, particularly when SH=00 or SH=10. The 2-phased method has yielded satisfactory results with 21 tasks and outperformed in certain instances. The MILP method did not achieve optimization of the second objective for 11 out of 128 instances.

4.2.10 Remaining instance (1200s)

Now we are going to tackle the instances that gave us no optimal solution for the second objective and so we didn't put them in our first tables, and let them for after to discuss even better the results that we obtained for these particular cases.

Instance	**OV**	**TTV**	**MILP**			**2-Phased**		
			OB1	**OB2**	**CPU (s)**	**OB1**	**OB2**	**CPU (s)**
Jackson	00	00	6	7.10^6	1200	6	$\mathbf{7.10^5}$	1200
	10	00	6	7.10^6	1200	6	$\mathbf{5.10^4}$	1168
Jaeschke	00	00	6	7.10^6	1200	6	$\mathbf{4.10^5}$	1200
Mitchell	00	00	8	7.10^6	1200	8	$\mathbf{9.10^5}$	1200
		10	8	7.10^6	1200	8	$\mathbf{1.10^6}$	1200
		25	8	7.10^6	1200	8	$\mathbf{1.10^6}$	1200
		50	8	7.10^6	1200	8	$\mathbf{9.10^5}$	1005
	10	00	8	7.10^6	1200	8	$\mathbf{9.10^5}$	1200
		10	8	7.10^6	1200	8	$\mathbf{8.10^5}$	1069
		25	8	7.10^6	1200	8	$\mathbf{8.10^5}$	1156
	25	00	8	7.10^6	1200	8	$\mathbf{9.10^5}$	1201

Table 4.9: Remaining instances: GAP from LP for MILP and 2-phased (1200s)

The instances presented in Table 4.9 were stopped during the optimization of the first objective using the MILP method. However, the 2-phased method yielded satisfactory results. This is due to the MILP requiring additional time to initiate efficient optimization for specific instances and parameters in our problem. Consequently, the 2- phased method proves notably superior in this context.

4.3 Results(300 seconds)

We will now test our approach for 5 minutes for several reasons:
First, imagine you are in a meeting and would like to explore different perspectives by adjusting inputs. Having a fast solution means you don't have to wait 20 minutes to see the changes.
Secondly, as observed in the tables of Section 4.2, the 2-phased method found solutions in less than 400 seconds for some instances. Comparing these solutions with those obtained within the same timeframe, we found that the 2-phased method performed better. In this section, we will test our methods with a time limit of 300 seconds.

4.3.1 Bowman C=19, 8 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00*	—	300	—	273
	10	34,26	300	39	215
	25	91,51	300	89,36	229
	50	37,18	300	41,85	215
	Average	54,31	300	56,73	219,66
10	00	44,90	303	44,90	113
	10	38,71	300	43,55	274
	25	53,55	314	54,79	53
	50	43,62	312	43,63	170
	Average	45,19	307,25	46,71	152,5
25	00	41,89	300	45,59	108
	10	34,32	338	40,31	95
	25	41,38	299	45,42	144
	50	34,53	300	40,56	200
	Average	38,03	309,25	42,89	136,75
50	00	25,43	300	45,07	161
	10	21,26	300	41,15	104
	25	49,24	300	54,64	177
	50	30,81	300	38,18	52
	Average	31,68	300	44,76	123,5

Table 4.10: Bowman GAP from LB for MILP and 2 phased (300s)

4.3.2 EX C=18, 5 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	00	71	15,87	29
	10	00	85	18,46	66
	25	00	72	13,09	29
	50	00	85	18,46	66
	Average	00	78,25	16,47	47,5
10	00	00	74	18,46	49
	10	00	87	00	67
	25	00	74	18,46	49
	50	00	87	15,87	30
	Average	00	80,5	13,19	48,75
25	00	00	91	18,46	51
	10	00	104	8,41	30
	25	00	91	00	50
	50	00	90	18,46	50
	Average	00	94	11,33	45,25
50	00	00	104	00	68
	10	00	104	18,46	50
	25	00	90	00	50
	50	00	90	18,46	50
	Average	00	97	9,23	54,5

Table 4.11: EX C=18 GAP from LB for MILP and 2 phased (300s)

4.3.3 EX C=20, 5 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB(%)	**CPU(s)**	**Gap from LB(%)**	**CPU(s)**
00	00	00	58	19,70	39
	10	00	58	19,70	37
	25	00	59	19,70	38
	50	00	59	20,08	24
	Average	00	58,50	19,79	34,5
10	00	00	72	19,70	41
	10	00	85	19,70	57
	25	00	73	1,38	25
	50	00	84	19,70	41
	Average	00	78,50	15,12	41
25	00	00	73	2,60	26
	10	00	73	00	41
	25	00	74	19,70	41
	50	00	73	19,70	40
	Average	00	73,25	10,5	37
50	00	00	72	19,70	40
	10	00	73	1,91	25
	25	00	73	19,70	41
	50	00	73	0,95	25
	Average	00	72,75	10,56	32,75

Table 4.12: EX C=20 GAP from LB for MILP and 2 phased (300s)

4.3.4 Jackson C=08, 11 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	**CPU (s)**	**Gap from LB (%)**	**CPU (s)**
00	00*	—	300	—	300
	10*	—	300	—	300
	25*	—	300	—	300
	50*	—	300	—	300
	Average	—	300	—	300
10	00*	—	300	—	300
	10*	—	300	—	300
	25*	—	300	—	300
	50*	—	300	—	300
	Average	—	300	—	300
25	00*	—	300	—	300
	10	49,83	300	63,08	96
	25	44,12	300	62,39	29
	50	46,02	299	58,27	31
	Average	46,65	299,66	61,24	52
50	00	99,45	301	92,93	26
	10	46,94	300	60,18	28
	25	46,74	300	78	21
	50	50,01	301	60,49	297
	Average	60,78	300,5	72,90	93

Table 4.13: Jackson GAP from LB for MILP and 2 phased (300s)

4.3.5 Jaeschke C=07, 9 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00*	—	300	—	300
	10	34,71	300	53,87	245
	25*	—	300	—	300
	50*	—	300	—	300
	Average	34,71	300	53,87	295
10	00*	—	300	—	300
	10	32,98	301	53,90	300
	25	28,33	300	46,89	319
	50	36,76	300	54,06	300
	Average	32,69	300,33	51,61	306,33
25	00*	—	300	—	300
	10	26,99	300	53,19	215
	25	35,69	300	50,65	271
	50	30,64	300	55,43	275
	Average	31,10	300	53,09	253,66
50	00	21,27	300	49,88	235
	10	22,75	300	51,18	350
	25	21,37	300	46,35	194
	50	47,44	304	60,47	347
	Average	28,20	300	51,97	281,5

Table 4.14: Jaeschke GAP from LB for MILP and 2 phased (300s)

4.3.6 Mansoor C=47, 11 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00*	—	300	—	300
	10	26,96	300	55,82	300
	25	26,96	300	50,99	289
	50	26,96	299	58,89	300
	Average	26,96	299,66	55,23	296,33
10	00	30,40	300	57,83	300
	10	30,42	300	54,62	300
	25	30,60	299	50,78	300
	50	30,48	300	58,07	300
	Average	30,47	299,75	55,32	300
25	00	34,29	300	41,47	208
	10	34,29	300	51,96	231
	25	34,02	300	51,76	249
	50	34,06	300	51,79	176
	Average	34,16	300	49,24	216
50	00	52,77	322	65,42	60
	10	47,36	303	64,65	62
	25	53,60	300	68,87	58
	50	49,36	300	62,98	226
	Average	50,77	306,25	65,48	101,5

Table 4.15: Mansoor GAP from LB for MILP and 2 phased (300s)

4.3.7 Mertens C=08, 7 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB (%)	CPU (s)	Gap from LB (%)	CPU (s)
00	00	16,33	300	39,28	235
	10	00	290	20,13	170
	25	20,24	300	36,30	220
	50	20,98	300	42,61	213
	Average	14,38	297,5	34,58	209,5
10	00	8,63	300	27,37	178
	10	00	296	27,43	83
	25	1,56	300	35,47	146
	50	13,98	300	31,30	148
	Average	6,04	299	30,39	138,5
25	00	8,20	300	33,79	150
	10	4,37	300	24,40	140
	25	00	293	11,19	131
	50	12,94	300	36,82	125
	Average	6,37	298,25	26,55	136,5
50	00	00	286	27,43	115
	10	8,06	300	18,63	117
	25	4,66	300	16,58	120
	50	00	288	20,13	105
	Average	3,18	293,5	20,69	114,25

Table 4.16: Mertens GAP from LB for MILP and 2 phased (300s)

4.3.8 Mitchell C=14, 21 tasks

OV	TTV	MILP		2-Phased	
		Gap from LB(%)	CPU(s)	Gap from LB(%)	CPU(s)
00	00	—	300	—	300
	10	—	300	—	300
	25	—	300	—	300
	50	—	300	—	300
10	00	—	300	—	300
	10	—	300	—	300
	25	—	300	—	300
	50	—	300	—	300
25	00	—	300	—	300
	10	—	300	—	300
	25	—	300	—	300
	50	—	300	—	300
50	00	—	300	—	300
	10	64,95	300	73,47	301
	25	63,01	286	70,25	300
	50	59,70	300	72,03	300
	Average	62,55	295,33	71,91	300,33

Table 4.17: Mitchell GAP from LB for MILP and 2 phased (300s)

For Bowman's instances (Table 4.10), the results are comparable for lower values of SH. No significant changes were observed for EX C=18 and EX C=20 (Tables 4.11, 4.12) as the number of tasks is only 5. Jackson's instances (Table 4.3.4) did not reach the optimization of the second objective for SH=10. The 2-phased method showed a substantial difference in GAP compared to the first method for Jaeschke (Table 4.3.5), Mansoor (Table 4.15), and Mertens (Table 4.16). Mitchell's instances (Table 4.17) did not achieve optimization of the second objective except for SH=50.

4.3.9 Remaining instances (300s)

Instance	**OV**	**TTV**	**MILP**			**2-Phased**		
			OB1	**OB2**	**CPU (s)**	**OB1**	**OB2**	**CPU (s)**
Bowman	00	00	4	6.10^6	300	4	$\mathbf{3.10^5}$	300
Jackson	00	00	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
		10	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	299
		25	8	7.10^6	300	**6**	$\mathbf{7.10^5}$	300
		50	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
	10	00	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
		10	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	299
		25	8	7.10^6	300	**6**	$\mathbf{7.10^5}$	300
		50	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
	25	00	6	7.10^6	300	6	$\mathbf{7.10^5}$	300
Jaeschke	00	00	8	7.10^6	300	**6**	$\mathbf{7.10^5}$	300
		25	8	6.10^6	300	**6**	$\mathbf{6.10^5}$	298
		50	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
	10	00	8	7.10^6	300	**6**	$\mathbf{6.10^5}$	300
	25	00	8	6.10^6	300	**6**	$\mathbf{5.10^5}$	311
Mansoor	00	00	6	7.10^6	300	**4**	$\mathbf{7.10^5}$	300
Mitchell	00	00	**8**	$\mathbf{7.10^6}$	287	9	9.10^5	300
		10	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
		25	8	7.10^6	299	8	$\mathbf{9.10^5}$	300
		50	8	7.10^6	300	8	$\mathbf{1.10^6}$	300
	10	00	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
		10	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
		25	8	7.10^6	300	8	$\mathbf{9.10^5}$	297
		50	8	7.10^6	300	8	$\mathbf{8.10^5}$	300
	25	00	8	7.10^6	284	8	$\mathbf{9.10^5}$	286
		10	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
		25	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
		50	8	7.10^6	300	8	$\mathbf{9.10^5}$	300
	50	00	8	7.10^6	299	8	$\mathbf{1.10^6}$	296

Table 4.18: Remaining instances : GAP from LB for MILP and 2-phased (300s)

For this table, we are going to discuss the instances that didn't give us a solution for the second objective. We can see that the 2-phased method is privileged for these instances. This may be because the MILP needs more time to start optimizing the second objective efficiently, especially for these multiple instances, as opposed to the previous results we got for the majority of our dataset.

As we can see, for the instances with a smaller number of tasks (8-11), the 2-phased method is better than the MILP even in the opened locations. This indicates that the 2-phased method started off with a good solution and then had the time to optimize it further within the time limit. The initial phase provided a solid foundation, allowing the method to refine and improve the solution effectively. This advantage highlights the efficiency of the 2-phased method in handling smaller task sets, where it quickly reaches a viable solution and utilizes the remaining time to enhance it further.In our case, the same thing goes for the instances with a bigger set of tasks. As we can see for Mitchell, which has the greatest number of tasks, it is optimized better with the 2-phased method than MILP.

4.4 Conclusion

Within this chapter, we have explored and observed the distinction between our methods, employing a 20-minute simulation since the majority of the dataset has converged towards a feasible and consistent solution. Subsequently, we assessed the performance of our methods under a more constrained time frame of 5 minutes.

General Conclusion and Perspectives

Within our study, we have addressed the problem using the MILP method and the 2-phased method. We tested the problem with a 20-minute simulation since the majority of the dataset converged towards a feasible and consistent solution. We also assessed the performance of our methods under a more constrained time frame of 5 minutes. Thus, we can draw the following conclusions:

- Our approaches can handle up to 21 tasks due to restricted number of locations
- When the number of tasks is less than 9, the MILP method remains better than 2-phased approach
- 2-phased method performs better when sh=0 or sh=10 (low values of structure heterogeneity)
- 2-phased method provides good results in a short time when the number of tasks is ≥ 11
- 2-phased method approach yields better solutions within a short time restriction.

We suggest some future research directions as follows:

- explore the problem with other configurations of locations and routes
- explore more instances with more tasks
- add that if we purchase multiple units of the same resource, for example 2 or more, the single cost will be reduced.
- impose restrictions on resources such as availability of resources

Bibliography

[And17] Ann-Louise Andersen. Development of changeable and reconfigurable manufacturing systems: supporting context-specific design of changeability. 2017.

[BD13] Shashikant Burnwal and Sankha Deb. Scheduling optimization of flexible manufacturing system using cuckoo search-based approach. *The International Journal of Advanced Manufacturing Technology*, 64:951–959, 2013.

[BDNS+23] Carlos Alberto Barrera-Diaz, Amir Nourmohammadi, Henrik Smedberg, Tehseen Aslam, and Amos H. C. Ng. An Enhanced Simulation-Based Multi-Objective Optimization Approach with Knowledge Discovery for Reconfigurable Manufacturing Systems. *Mathematics*, 11(6):1527, March 2023.

[BFS07] Nils Boysen, Malte Fliedner, and Armin Scholl. A classification of assembly line balancing problems. *European journal of operational research*, 183(2):674–693, 2007.

[BGDJLD21] Isabel Barros Garcia, Joanna Daaboul, Antoine Jouglet, and Julien Le Duigou. An approach to jointly optimize the process plan, scheduling, and layout design in reconfigurable manufacturing systems. In *International Workshop on Service Orientation in Holonic and Multi-agent Manufacturing*, pages 403–415. Springer, 2021.

[BGK23] Osman Baturhan Bayraktar, Martin Grunow, and Rainer Kolisch. Instance set and implementation codes of dynamic reconfigurations of matrix assembly layouts, 2023.

[BR06] Yossi Bukchin and Ithai Rabinowitch. A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs. *European Journal of Operational Research*, 174(1):492–508, 2006.

[BR14] Leonardo Borba and Marcus Ritt. A heuristic and a branch-and-bound algorithm for the assembly line worker assignment and balancing problem. *Computers & Operations Research*, 45:87–96, 2014.

[BS06] Christian Becker and Armin Scholl. A survey on problems and methods in generalized assembly line balancing. *European journal of operational research*, 168(3):694–715, 2006.

[BSS22] Nils Boysen, Philipp Schulze, and Armin Scholl. Assembly line balancing: What happened in the last fifteen years? *European Journal of Operational Research*, 301(3):797–814, 2022.

[BY21] LAKHDAR Belkharroubi and KHADIDJA Yahyaoui. A hybrid grasp-genetic algorithm for mixed-model assembly line balancing problem type 2. *International Journal of Computing*, 20(3):424–432, 2021.

[DK19] K Dziki and D Krenczyk. Mixed-model assembly line balancing problem with tasks assignment. In *IOP Conference Series: Materials Science and Engineering*, volume 591, page 012013. IOP Publishing, 2019.

[ESI+11] Gholam Reza Esmaeilian, Shamsuddin Sulaiman, Napsiah Ismail, Maryam Hamedi, and Megat Mohammad Hamdan Megat Ahmad. A tabu search approach for mixed-model parallel assembly line balancing problem (type ii). *International Journal of Industrial and Systems Engineering*, 8(4):407–431, 2011.

[GŚ12] Arkadiusz Gola and Antoni Świć. Directions of manufacturing systems' evolution from the flexibility level point of view. *Innovations in Management and Production Engineering, Oficyna Wyd. Polskiego Towarzystwa Zarzadzania Produkcja, Opole*, pages 226–238, 2012.

[HG19] Andreas Hottenrott and Martin Grunow. Flexible layouts for the mixed-model assembly of heterogeneous vehicles. *OR Spectrum*, 41(4):943–979, 2019.

[JAPS05] J Jerald, Prabaharan Asokan, G Prabaharan, and R Saravanan. Scheduling optimisation of flexible manufacturing systems using particle swarm optimisation algorithm. *The international journal of advanced manufacturing technology*, 25:964–971, 2005.

[JW03] Mingzhou Jin and S David Wu. A new heuristic method for mixed model assembly line balancing problem. *Computers & Industrial Engineering*, 44(1):159–169, 2003.

[KHBB20] A. Khezri, H. Haddou Benderbal, and Lyes Benyoucef. Sustainable Multi-objective Process Plan Generation in RMS Through Modelling Energy Consumption. In Lyes Benyoucef, editor, *Reconfigurable Manufacturing Systems: From Design to Implementation*, pages 161–177. Springer International Publishing, Cham, 2020. Series Title: Springer Series in Advanced Manufacturing.

[KM13] Naveen Kumar and Dalgobind Mahto. Assembly line balancing: a review of developments and trends in approach to industrial application. *Global Journal of Researches in Engineering Industrial Engineering*, 13(2):29–50, 2013.

[LGZ09] Jianzhi Li, Miguel González, and Yun Zhu. A hybrid simulation optimization method for production planning of dedicated remanufacturing. *International Journal of Production Economics*, 117(2):286–301, 2009.

[Per18] Jordi Pereira. Modelling and solving a cost-oriented resource-constrained multi-model assembly line balancing problem. *International Journal of Production Research*, 56(11):3994–4016, 2018.

[RE15] Reza Ramezanian and Abdullah Ezzatpanah. Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem. *Computers & Industrial Engineering*, 87:74–80, 2015.

[RG17] Abdolreza Roshani and Davide Giglio. Simulated annealing algorithms for the multi-manned assembly line balancing problem: minimising cycle time. *International Journal of Production Research*, 55(10):2731–2751, 2017.

[SHGT15] Malte Schönemann, Christoph Herrmann, Peter Greschke, and Sebastian Thiede. Simulation of matrix-structured manufacturing systems. *Journal of Manufacturing Systems*, 37:104–112, 2015.

[SV04] Ana Sofia Simaria and Pedro M Vilarinho. A genetic algorithm based approach to the mixed-model assembly line balancing problem of type ii. *Computers & Industrial Engineering*, 47(4):391–407, 2004.

[SWS21] Patrick Schumacher, Christian Weckenborg, and Thomas S Spengler. Economic design of matrix-structured manufacturing systems. In *Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems: IFIP WG 5.7 International Conference, APMS 2021, Nantes, France, September 5–9, 2021, Proceedings, Part II*, pages 516–524. Springer, 2021.

[SWS22] Patrick Schumacher, Christian Weckenborg, and Thomas S Spengler. The impact of operation, equipment, and material handling flexibility on the design of matrix-structured manufacturing systems. *IFAC-PapersOnLine*, 55(2):481–486, 2022.

[TLZ+16] Qiuhua Tang, Yanli Liang, Liping Zhang, Christodoulos A Floudas, and Xiaojun Cao. Balancing mixed-model assembly lines with sequence-dependent tasks via hybrid genetic algorithm. *Journal of Global Optimization*, 65(1):83–107, 2016.

[WBH12] Ying Wang, Arne Bilberg, and Ronen Hadar. Implementation of reconfigurable manufacturing systems, the case of the lego group. In *4th World Conference on Production & Operations Management/19th International Annual EurOMA Conference Amsterdam, Amsterdam, Netherlands,(Submitted manuscript)*, pages 1–9, 2012.

[XW05] Weijun Xia and Zhiming Wu. An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems. *Computers & industrial engineering*, 48(2):409–425, 2005.

[YCGBD21] Abdelkrim R Yelles-Chaouche, Evgeny Gurevsky, Nadjib Brahimi, and Alexandre Dolgui. Reconfigurable manufacturing systems from an optimisation perspective: a focused review of literature. *International Journal of Production Research*, 59(21):6400–6418, 2021.

[YYO20] Seyda Topaloglu Yildiz, Gokalp Yildiz, and Raziye Okyay. Assembly line balancing problem with hierarchical worker assignment, positional constraints, task assignment restrictions and parallel workstations. *International Journal of Industrial Engineering*, 27(3), 2020.

Webography

[1] All results of the MILP and 2-phased methods for 20 minutes and 5 minutes : https://drive.google.com/results

Appendix

4.4.1 Bowman

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	4	346165	34.23
		10	4	346139	14.67
		25	4	346107	24.81
		50	4	343395	24
	10	00	4	346161	32,49
		10	4	346135	22,54
		25	4	346105	33,65
		50	4	343395	33,86
	25	00	4	341609	31,24
		10	4	341531	25,87
		25	4	341401	29,32
		50	4	341197	17,66
	50	00	4	335789	5,23
		10	4	335789	4,05
		25	4	335789	15,5
		50	4	335789	9,18
2-Phased	00	00	4	375789	39,42
		10	4	375789	21,41
		25	4	375789	30,76
		50	4	375789	30,76
	10	00	4	375789	38,24
		10	4	375789	28,66
		25	4	375789	38,89
		50	4	375789	39,57
	25	00	4	375789	37,50
		10	4	375789	32,77
		25	4	375789	35,80
		50	4	375789	25,24
	50	00	4	456251	30,25
		10	4	449253	28,28
		25	4	375789	24,48
		50	4	375789	18,84

Table 4.19: MILP and 2-Phased Results within 20 minutes : Bowman

4.4.2 EX C=18

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	2	176665	0
		10	2	176665	0
		25	2	176665	0
		50	2	176665	0
	10	00	2	176665	0
		10	2	176665	0
		25	2	176665	0
		50	2	176665	0
	25	00	2	176665	0
		10	2	176665	0
		25	2	176665	0
		50	2	176665	0
	50	00	2	176665	0
		10	2	176665	0
		25	2	176665	0
		50	2	176665	0
2-Phased	00	00	2	209995	15,87
		10	2	216665	18,46
		25	2	203275	13,09
		50	2	216665	18,46
	10	00	2	216665	18,46
		10	2	176665	00
		25	2	216665	18,46
		50	2	209995	15,87
	25	00	2	216665	18,46
		10	2	192881	8,41
		25	2	176665	00
		50	2	216665	18,46
	50	00	2	176665	00
		10	2	216665	18,46
		25	2	176665	00
		50	2	216665	18,46

Table 4.20: MILP and 2-Phased Results within 20 minutes : EX C=18

4.4.3 EX C=20

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	2	163000	0
		10	2	163000	0
		25	2	163000	0
		50	2	163000	0
	10	00	2	163000	0
		10	2	163000	0
		25	2	163000	0
		50	2	163000	0
	25	00	2	163000	0
		10	2	163000	0
		25	2	163000	0
		50	2	163000	0
	50	00	2	163000	0
		10	2	163000	0
		25	2	163000	0
		50	2	163000	0
2-Phased	00	00	2	203000	19,70
		10	2	203000	19,70
		25	2	203000	19,70
		50	2	203952	20,08
	10	00	2	203000	19,70
		10	2	203000	19,70
		25	2	165280	1,38
		50	2	203000	19,70
	25	00	2	167350	2,60
		10	2	163000	00
		25	2	203000	19,70
		50	2	203000	19,70
	50	00	2	203000	19,70
		10	2	166166	1,91
		25	2	203000	19,70
		50	2	164566	0,95

Table 4.21: MILP and 2-Phased Results within 20 minutes : EX C=20

4.4.4 Jackson

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	6	7364830	33,33
		10	6	505960	43,98
		25	6	488364	42,51
		50	6	471438	39,63
	10	00	6	7347684	42,9
		10	6	455594	39,95
		25	6	462500	40,23
		50	6	551868	49,33
	25	00	6	450586	42,04
		10	6	424548	40,24
		25	6	455606	43,05
		50	6	450076	42,03
	50	00	6	420178	41,85
		10	6	454158	45,90
		25	6	416016	42,09
		50	6	456418	46,76
2-Phased	00	00	6	686892	100
		10	6	606266	53,25
		25	6	615104	54,36
		50	6	606568	53,08
	10	00	6	610736	100
		10	6	605605	54,82
		25	6	606080	54,39
		50	6	5656688	50,57
	25	00	6	544328	52,03
		10	6	565812	55,16
		25	6	567548	54,29
		50	6	565430	53,86
	50	00	6	565934	56,83
		10	6	565528	56,56
		25	6	565732	57,42
		50	6	565430	57,02

Table 4.22: MILP and 2-Phased Results within 20 minutes : Jackson

4.4.5 Jaeschke

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	6	7018433	33,33
		10	6	431425	30,77
		25	6	391425	26,17
		50	6	391425	20,85
	10	00	6	391425	26,83
		10	6	427123	25,09
		25	6	391425	19,67
		50	6	425321	24,82
	25	00	6	383673	26,97
		10	6	383673	23,32
		25	6	391425	26,80
		50	6	391425	23,49
	50	00	6	388777	7.56
		10	6	388777	6.21
		25	6	388777	5,60
		50	6	396367	28.26
2-Phased	00	00	6	706735	100
		10	6	571523	47,75
		25	6	573413	49,61
		50	6	569559	45,61
	10	00	6	610935	53,12
		10	6	610327	47,58
		25	6	529567	40,63
		50	6	609149	47,51
	25	00	6	573083	51,11
		10	6	573167	48,67
		25	6	570823	49,81
		50	6	609155	50,84
	50	00	6	510629	41,15
		10	6	572043	36,34
		25	6	569711	35,94
		50	6	610467	53,42

Table 4.23: MILP and 2-Phased Results within 20 minutes : Jaeschke

4.4.6 Mansoor

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	4	326381	26,77
		10	4	326381	26,80
		25	4	326381	27
		50	4	326381	26,40
	10	00	4	326381	30,40
		10	4	326381	30,25
		25	4	326381	30,48
		50	4	326381	30,27
	25	00	4	326381	34,29
		10	4	326381	34,12
		25	4	326381	34,02
		50	4	326381	34,06
	50	00	4	326381	48.28
		10	4	326381	38.20
		25	4	326381	49.10
		50	4	326381	47.10
2-Phased	00	00	4	486381	50,86
		10	4	446381	46,48
		25	4	486381	50,99
		50	4	446381	46,19
	10	00	4	446381	49,11
		10	4	446381	49
		25	4	446381	49,17
		50	4	486381	53,21
	25	00	4	366381	41,47
		10	4	446381	51,83
		25	4	446381	51,76
		50	4	446381	51,79
	50	00	4	446381	62,19
		10	4	486381	58,54
		25	4	486381	65,85
		50	4	446381	61,30

Table 4.24: MILP and 2-Phased Results within 20 minutes : Mansoor

4.4.7 Mertens

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	4	317500	3,88
		10	4	317500	2,93
		25	4	317500	2,98
		50	4	317500	2,39
	10	00	4	317500	4,61
		10	4	317500	4,90
		25	4	317500	1,77
		50	4	317500	2,69
	25	00	4	317500	4,10
		10	4	317500	3,70
		25	4	317500	3,62
		50	4	317500	3,78
	50	00	4	317500	9.05
		10	4	317500	4.61
		25	4	317500	2.70
		50	4	317500	2.89
2-Phased	00	00	4	437500	27,43
		10	4	397500	20,13
		25	4	397500	20,13
		50	4	437124	27,37
	10	00	4	397500	20,13
		10	4	437500	27,43
		25	4	437500	27,43
		50	4	397500	20,13
	25	00	4	437500	27,43
		10	4	397500	20,13
		25	4	357500	11,19
		50	4	437500	27,43
	50	00	4	437500	27,43
		10	4	357500	11,19
		25	4	357500	11,19
		50	4	397500	20,13

Table 4.25: MILP and 2-Phased Results within 20 minutes : Mertens

4.4.8 Mitchell

Method	OV	TTV	OB 1	OB 2	GAP
MILP	00	00	8	7639386	62,50
		10	8	7643066	37,50
		25	8	7622974	62,50
		50	8	7624702	25
	10	00	8	7625222	62,50
		10	8	7625626	25
		25	8	7632854	62,50
		50	8	1192536	73,10
	25	00	8	7626212	37,50
		10	8	699408	56,61
		25	8	819136	62,91
		50	8	694334	57,45
	50	00	8	708588	63,05
		10	8	701094	61,76
		25	8	656916	59,59
		50	8	629732	57,98
2-Phased	00	00	8	911648	100
		10	8	1033862	100
		25	8	1036854	100
		50	8	807140	100
	10	00	8	954686	100
		10	8	839586	100
		25	8	875820	100
		50	8	838280	61,73
	25	00	8	918578	100
		10	8	912440	66,74
		25	8	874378	65,26
		50	8	835206	64,63
	50	00	8	871260	69,95
		10	8	910476	70,56
		25	8	872922	69,59
		50	8	870650	69,61

Table 4.26: MILP and 2-Phased Results within 20 minutes : Mitchell

Abstract

This thesis investigates the Flexible Layout Design Problem (FLDP), which involves a set of stations arranged in a matrix-structured manufacturing system (MMS). Initially, the main objectives are to minimize the number of stations used and to minimize the flow of transport between stations. Subsequently, we implement a Mixed-Integer Linear Programming (MILP) model which integrate resource allocations into the FLDP and propose a matheuristic method to address the problem. We conduct tests using a chosen dataset for 20 minutes and then for 5 minutes. Finally, we discuss the results obtained and conclude with perspectives for future research.

Key words: Reconfigurable Manufacturing Systems, Matrix-structured manufacturing system, Assembly line balancing, Flexible layout design

---

Résumé

Ce projet de fin d'études étudie le Problème de Conception de Layout Flexible (FLDP), qui implique un ensemble de stations disposées dans un système de fabrication à structure matricielle (MMS). Initialement, les principaux objectifs sont de minimiser le nombre de stations utilisées et de réduire le flux de transport entre les stations. Par la suite, nous mettons en œuvre un modèle de Programmation Linéaire Mixte en Nombre Entier (MILP) qui intègre les allocations de ressources dans le FLDP, et proposons une méthode matheuristique pour résoudre le problème. Nous réalisons des tests sur un jeu de données choisi pendant 20 minutes puis pour 5 minutes. Enfin, nous discutons des résultats obtenus et concluons en évoquant les perspectives pour les recherches futures.

Mots clés: Systèmes de Fabrication Reconfigurables, Systèmes de Fabrication à Structure Matricielle, Équilibrage de Lignes d'Assemblage, Conception de Layout Flexible

---

ملخص

يدور موضوع مشروع التخرج حول مشكلة تصميم الهياكل المرنة (FLDP) ، التي تتضمن مجموعة من المحطات المنظمة في نظام تصنيع ذو هيكل مصفوفة (MMS) . في البداية، الأهداف الرئيسية هي تقليل عدد المحطات المستخدمة وتقليل تدفق النقل بين المحطات. بعد ذلك، نقوم بتنفيذ نموذج للبرمجة الخطية المختلطة والعدد الصحيح (MILP) الذي يدمج تخصيصات الموارد في (FLDP) ، ونقترح أسلوباً رياضياً لحل المشكلة. نجري اختبارات على مجموعة بيانات مختارة لمدة 20 دقيقة ثم لمدة 5 دقائق. في الختام، نناقش النتائج المحصلة و نختتم بمناقشة آفاق البحث المستقبلية

الكلمات المفتاحية: أنظمة التصنيع قابلة لإعادة التكوين، نظام التصنيع ذو الهيكل المصفوفي، توازن خط الإنتاج، تصميم الهياكل المرنة