A framework to offer high value manufacturing through self-reconfigurable manufacturing systems

Victor Manuel Cedeno-Campos

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Sheffield

Faculty of engineering

Department of Automatic Control and Systems Engineering

August 2016

Abstract

The High Value Manufacturing (HVM) sector is vital for developed countries due to the creation of innovative products with advanced technology that cannot be reproduced at the same cost and time with traditional technology. The main challenge for HVM is to rapidly increase production volume from one-off products to low production volume. This requires highly flexible manufacturing systems that can produce new products at variable production volumes. Current manufacturing systems, classified as dedicated, flexible and reconfigurable systems, are limited to produce one type of product(s), within a production volume range and have fixed layouts of machines. Thus, there is a need for highly flexible systems that can rapidly adjust their production volume according to the production demand (i.e. main HVM challenge).

Therefore, a novel manufacturing framework, called INTelligent REconfiguration for a raPID production change (INTREPID), is presented in this thesis. INTREPID consists of a user interface and communications platform, a job allocation system, a globally distributed network of Reconfigurable Manufacturing Centres (RMCs), consisting of interconnected factories, and Self-Reconfigurable Manufacturing Systems (S-RMSs). The highly flexible S-RMS consists of movable machines and Mobile Manufacturing Robots (MMRs). The novelty of the S-RMS is its capability of forming layouts bespoke to the current production needs. The vision of INTREPID is to offer global HVM services through the network of RMCs. The job allocation system determines the best possible RMCs or factories to perform a job by considering the complexity of the production requirements and the status of the available S-RMSs at each factory.

The planning of the production with S-RMS is challenging due to its high flexibility. The main example of this flexibility is the possibility to create layouts bespoke to current production needs. Yet, this flexibility involves the challenges of determining allocations and schedules of tasks to robots and machines, positions to manufacture, and routes to reach those positions. In manufacturing systems with fixed layouts, production plans are determined by solving a sequence of problems. However, for the S-RMS, it is proposed to determine production plans with a single problem that covers the scheduling, machine layout and vehicle routing problems simultaneously. This novel problem is called the Scheduling, positions Assigning and Routing problem (SAR) problem. In order to determine the best possible production plan(s) for the S-RMS, it is necessary to use optimisation methods.

Dozens of elements, characteristics and assumptions from the constituent problems might be included in the formulation of the SAR problem. Elements, characteristics and assumptions can be considered as decision variables on whether to include or not the elements and characteristics and under which assumptions in the formulation. There are two types of decision variables. Fundamental variables are natural to the SAR problem (e.g. manufacturing resources, factory design and operation), whilst auxiliary variables arise from the aim to simplify the formulation of the optimisation problem (i.e. time formulated as discrete or continuous). Due to the large number of decision variables, there might be millions of possible ways to formulate the SAR problem (i.e. the SAR problem space). Some of these variants are intractable to be solved with optimisation methods. Hence, before formulating the SAR problem, it is necessary to select a problem(s) that is realistic to industrial scenarios but solvable with optimisation methods.

Existing selection methods work with pairwise comparisons of alternatives. However, for a space of millions of SAR problems, pairwise comparisons are intractable. Hence, in this thesis, a novel Decision Making Methodology (DMM) based on the controlled convergence method is presented. The DMM helps down-selecting one or a few SAR problems from millions of possible SAR problems. The DMM is demonstrated with a case study of the SAR problem and the results show a significant reduction of the reviewed SAR problems and the time to select them.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Prof. Tony J. Dodd and Dr. Paul Trodden, for all the guidance and patience through meetings and discussions. Tony motivated me to propose my own research topic and to pursue dreams for a fully automated factory, whilst Paul firmly encouraged me to provide a formal definition of this dreamed factory. Moreover, Tony gave me freedom of action, trusted my judgement to propose and follow my own ideas, guided and challenged me whenever necessary.

I would like to also express my sincere gratitude to my examiners, Dr. Reza Abdi and Prof. Peter Fleming, for their constructive and insightful comments, as well as their valuable time during the examination process.

I dedicate this thesis to my family, specially to my beloved mother and the memory of my beloved grandmother, uncle and aunts. All my gratitude to my friends from Mexico and from around the world for all their support throughout the PhD.

People from around the world, with different opinions, with different religious beliefs, and different societies enrich ourselves, our culture and our societies. We should not be afraid of "the different", neither lock ourselves within walls nor seclude "the others" outside walls.

Also, I would like to thank my sponsor, CONACyT, for giving me the great opportunity to study abroad and contribute to the development of science and technology.

Contents

List of Figures

List of Figures in Appendices

List of Tables

List of Tables in Appendices

Acronyms

AGV Automated Guided Vehicle

AHP Analytic Hierarchy Process

AM Agile Manufacturing

CAD Computer Aided Drawing

CAx Computer Aided Technologies

CBDM Cloud Based Design Manufacturing

CIM Computer Integrated Manufacturing

CPS Cyber Physical Systems

CNC Computer Numerical Control

CVRP Capacitated Vehicle Routing Problem

DFLP Dynamic Facilities Layout Problem

DMM Decision Making Methodology

DMS Dedicated Manufacturing System

DMSS Decision Making Support System

DVRP Dynamic Vehicle Routing Problem

ELECTRE ELimination and (Et) Choice Translating REality

FMS Flexible Manufacturing System

FLP Facilities Layout Problem

GT Group Technology

HFVRP Heterogeneous Fleet Vehicle Routing Problem

HV High Value

HVM High Value Manufacturing

INTREPID INTelligent REconfiguration for a raPID production change

IoT Internet of Things

JIT Just In Time

MC Mass Customisation

MDVRP Multi Depot Vehicle Routing Problem

MFVRP Mixed Fleet Vehicle Routing Problem

MMR Mobile Manufacturing Robot

MRTA Multi Robot Task Allocation

mTSP multiple Travelling Salesman Problem

NM Networked Manufacturing

OVRP Open Vehicle Routing Problem

PLC Programmable Logic Controller

PROMETHEE Preference Ranking Organization METHod for Enrichment Evaluations

PVRP Periodic Vehicle Routing Problem

QFD Quality Function Deployment

QRM Quick Response Manufacturing

RMC Reconfigurable Manufacturing Centre

RMS Reconfigurable Manufacturing System

RP Rapid Prototyping

SAR Scheduling, positions Assigning and Routing problem

SDVRP Split Deliveries Vehicle Routing Problem

SOM Service-Oriented Manufacturing

S-RMS Self-Reconfigurable Manufacturing System

TDVRP Time Dependent Vehicle Routing Problem

TOPSIS Technique for Order Preference by Similarity to Ideal Solution

TPS Toyota Production System

TSP Travelling Salesman Problem

VIKOR Multicriteria Optimisation and Compromise Solution

VRP Vehicle Routing Problem

VRPPD Vehicle Routing Problem with Pickup and Delivery

VRPSD Vehicle Routing Problem with Stochastic Demands

VRPSPD Vehicle Routing Problem with Simultaneous Pickup and Delivery

VRPTW Vehicle Routing Problem with Time Windows

WRM Weighted Rating Method

Chapter 1

Introduction

The manufacturing sector is of paramount importance to the economy. Due to globalisation, the manufacturing sector has become very competitive and challenging. As a result, many companies compete by adding value to their products instead of reducing their costs. Adding value refers to endowing products and services with a unique selling point that is very attractive, if not irresistible, to customers. Examples of added value are customised features and attractive functionalities. Adding value through the use of advanced manufacturing technologies is known as HVM [1].

The distinguishing factor of HVM over other alternatives to add value is the difficulty to reproduce products (i.e. High Value (HV) products) at the same cost, time and quality. The difficulty to reproduce HV products is a consequence of the many years of research and development required to design and develop HV products and their manufacturing processes. Moreover, the main contribution of HVM to society is the creation of intellectual property and patents due to the development of HV products. Important challenges for HVM are [2]:

- Developing approaches for concurrent engineering in order to boost product development
- Developing plug and play and modular systems to produce high-volume production
- Creating and controlling fragmented global supply chains to manufacture near customers
- Developing approaches that facilitate increasing the production volume of new products from one-off products to low production volumes

The last challenge is known as the valley of death [2]. Although, this challenge refers primarily to increase production volume according to the product demand, a broader definition of the valley of death is to increase or decrease (i.e. adjust) production volume according to demand whilst keeping production efficiency (i.e. production cost and time) and quality. For HV products, adjusting production volume is particularly difficult due to the use of advanced technologies for their manufacture. Therefore, it is important to research and develop highly flexible and reconfigurable frameworks and systems that can rapidly adjust their production volume according to a dynamic production demand (i.e. main HVM challenge). The focus on this thesis is to address the valley of death challenge for products manufactured with advanced technology (e.g. aircraft engines) and low production volume (i.e. 1 to 10 products per hour [3]). The valley of death challenge is addressed through the proposal of a manufacturing framework to offer global HVM services and its self-reconfigurable manufacturing system.

The rest of this chapter is organised in the following sections. An overview of relevant research that focuses on manufacturing in a highly dynamic scenario is presented in Section 1.1. The aim and objectives of this thesis, that focus on addressing the main HVM challenge, are presented in Section 1.2. The contributions of this thesis are presented in Section 1.3, and publications are presented in Section 1.4. Lastly, an outline of the rest of this thesis is provided in Section 1.5.

1.1 Motivation

Conventional systems such as flexible, reconfigurable and dedicated systems are capable of manufacturing within a finite range of production volume and product variants, Figure 1.1 [4],[5]. Dedicated Manufacturing Systems (DMSs) are capable of producing high production volumes of a single product. In contrast, Flexible Manufacturing Systems (FMSs) can produce multiple products at low production volumes. Alternatively, Reconfigurable Manufacturing Systems (RMSs) are capable of producing more variants of products than DMSs but less than FMSs at a higher production volume than FMSs but lower than DMSs. These three type of systems are not capable of rapidly changing production capacity for innovative products because they are designed to manufacture specific production volumes and for a limited number of products or product families. Therefore, these type of systems are not capable of addressing the main HVM challenge (i.e. adjusting production volume according to production demand of innovative products).

The production layouts of these conventional manufacturing systems are designed and implemented considering the expected number of product variants and their volume. However, nowadays, companies have to manage unknown and dynamic production requirements that require complete change of the production layout. Hence, dedicated, flexible and reconfigurable manufacturing systems are not flexible enough to address the main HVM challenge. Approaches focused on

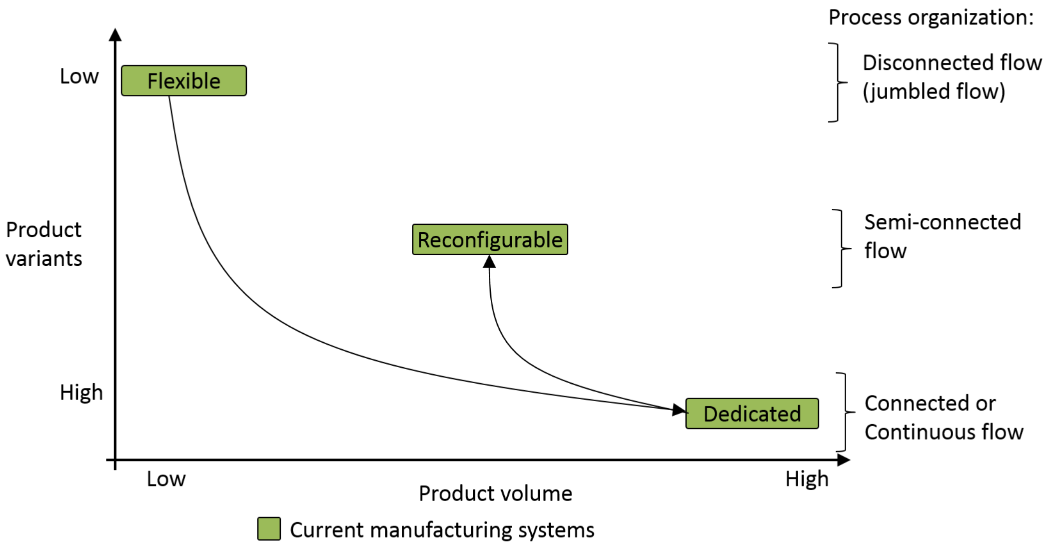


Figure 1.1: Classification of established manufacturing systems according to the production volume, product variants they can manufacture and the organisation of their processes [4],[5]. The flexible, reconfigurable and dedicated manufacturing systems are classified according to the production volume and the number of product variants each system can produce. Flexible Manufacturing Systems (FMSs) have their processes organised in a disconnected flow and the FMSs are able to manufacture a high product variety at low production volumes. Dedicated Manufacturing Systems (DMSs) have their processes organised in a connected flow, and the DMSs are able to produce high production volume but for a single product. Reconfigurable Manufacturing Systems (RMSs) are a hybrid of FMSs and DMSs. RMSs have their processes organised in a semi-connected flow and the FMSs are able to produce a high variety of products at low production volumes.

producing at variable production volumes or for multiple products are:

- Networks of decentralised factories [6],[7]
- Adaptable and cognitive factories [8],[9]
- Adaptable and mobile manufacturing resources such as:
  - Mobile robots for manufacturing [10],[11]
  - Mobile manufacturing cells [12],[13]
  - Manually movable manufacturing equipment (e.g. industrial robots [14] and manufacturing stations [15])

The use of idle manufacturing equipment in factory networks to offer manufacturing services was proposed in the paradigm cloud manufacturing [6]. The research focused on the technologies for product data transfer and management, and makes use of off-the-shelf scheduling algorithms such as first come, first serve. Alternative research on factory networks with a focus on mechanisms for an agile formation and coordination of temporal factories was proposed in the framework NetMan [7]. A transformable factory where production layouts can be manually changed was proposed in [8]. This factory considers traditional equipment such as DMSs and FMSs, which takes considerable time to change the layouts, and setup and calibrate equipment. Also, this factory does not make use of autonomous resources endowed with mobility and high level decision making systems. In contrast to this, providing learning, reasoning and planning abilities to the production resources was proposed to achieve a cognitive factory [9]. The cognitive factory proposed intelligent and autonomous resources which can change production settings automatically depending on well known products.

In order to address the lack of mobility of the cognitive and tranformable factory, research on mobile manufacturing resources has been proposed. Mobile robots such as mobile manipulators were proposed to perform basic assembly and logistics tasks in [10]. This research focused on developing and building systems to facilitate an accurate movement of robots in a well structured environment and the manipulation of parts to transport and assembly. The use of several robots for manufacturing required research on allocation algorithms. This issue was researched in [11], where multiple mobile manipulators were used to perform manufacturing tasks at fixed and known positions. The Hungarian method was implemented to determine robots to tasks allocations. The use of fixed positions to manufacture in a well structured environment reduces the flexibility of the previous two systems and make them prompt to error when the environment is changed. Also, robots require new training for different products and environments.

Mobile manufacturing cells to create layouts depending on the products to manufacture were proposed in [12],[13]. These two approaches focused on developing path planning and optimization algorithms that result in layouts depending on the products. There is an implicit allocation of manufacturing cells to the tasks by the minimisation of travelling distance of manufacturing cells to products. The use of manufacturing cells instead of individual manufacturing processes reduces the flexibility of the system to create novel layouts and manufacture complex products. The use of manually movable resources was proposed in [14],[15]. The resources consist of industrial robots [14] and production stations [15] with plug and play capacity. These two approaches focus on rapid deployment, calibration and change of production settings according to the product to manufacture. The manual movement of resources and the use of fixed locations to manufacture limit the flexibility of these approaches.

These approaches on adaptable factories and mobile manufacturing resources are independent from each other, and they focus on specific problems. These approaches have focus on factories networks, factories and manufacturing resources, but none has proposed a comprehensive framework that covers these three elements all together. There is a lack of comprehensive frameworks and systems that can manufacture innovative products at variable volumes with the use of advanced technologies (i.e. main HVM challenge). In order to address this challenge, highly flexible manufacturing systems that can rapidly and automatically change production layouts to produce multiple products at variable volumes are required. Therefore, a novel framework that makes use of networks of reconfigurable factories and mobile robots as manufacturing resources is presented in this thesis.

Due to the use of reconfigurable factories and mobile robots, it is necessary to solve a very complex problem for the production planning. Production planning with mobile robots requires determining allocations and schedules of tasks to robots, positions to perform the tasks, and routes of movement to reach these positions. Through the rest of this thesis, the production planning problem is called the SAR. Representative problem formulations with the use of mobile robots (i.e. mobile manipulators) for manufacturing focus on path planning for sequences of tasks [16], trajectory planning of the mobile platform and the arm robot [17], trajectory optimisation [18], analysis of dynamic motion of a flexible manipulator with maximum payload [19], multi-robot task allocation [11], large force tasks with multiple robots [20], optimisation of path planning and torque minimisation for multiple robots [21] However, there is a lack of problem formulations that consider the scheduling, positions assigning and path planning to coordinate the production of multiple products under quality, time and cost constraints. Hence, an important contribution is to formulate problems with this type of complexity and detail (e.g. SAR problem).

Prior to solving the SAR problem, it is necessary to formulate an inclusive SAR problem that covers elements and characteristics regarding the factory design and its operation, and mobile robots and their capabilities. Hence, it is necessary to study operations research problems such as scheduling, allocation, machine layout and vehicle routing, and robotics problems such as path planning and multi-robot task allocation. The study of these problems provides a large number of elements to include in the SAR problem formulation, characteristics of the problem and assumptions that can be applied over these elements and characteristics. The combination of these elements, characteristics and their assumptions might result in a vast number of ways to formulate the SAR problem. For example twenty types of assumptions where each assumption has two options can result in approximately one million ways to formulate the SAR problem. Hence, it is necessary to select a SAR problem to formulate that is realistic enough to industrial scenarios but feasible enough to be solved with optimisation methods.

The problem to select a SAR problem to formulate can be addressed with decision making methods focused on selection. The most relevant decision making methods belong to concept generation and selection from the field of product design and development [22],[23]. Relevant decision making methods for selection make use of decision matrix, outranking comparisons, and distance-based comparisons. Decision matrix methods are the Quality Function Deployment (QFD) method, the Weighted Rating Method (WRM), the Pugh's evaluation matrix and the Analytic Hierarchy Process (AHP) [24],[25]. The most relevant outranking methods are ELimination and (Et) Choice Translating REality (ELECTRE) and Preference Ranking Organization METHod for Enrichment Evaluations (PROMETHEE) [26].

Examples of distance-based methods are Multicriteria Optimisation and Compromise Solution (VIKOR) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [27],[28]. These methods are based on pairwise comparison of alternatives against each other at each selection criteria. Although for the SAR problem there is a single selection criteria (i.e. realism versus solvability), there are millions of alternatives. Therefore, it is necessary to propose and develop a novel decision making methodology for selection from a very large number of problems.

1.2 Aim and objectives

The aim of this thesis is to propose, develop and formulate the production planning problem of a novel approach that is capable of autonomously adjusting production capacity according to current production requirements (i.e. surpassing the valley of death for HVM [2]). In order to achieve this aim, the following objectives are proposed:

- Analysis of the literature on the following topics:
  - Highly flexible frameworks and systems that focus on manufacturing in a highly dynamic scenarios.
  - Problems related to production planning with mobile robots. Related problems from operations research are: scheduling, machine layout and vehicle routing, whilst from robotics are: path planning and multi-robot task allocation.
  - Decision making methods focused on selection.
- Propose and develop a novel framework with the aim of offering global HVM services through a user interface and communications platform, a network of reconfigurable factories, a job allocation system, and mobile manufacturing resources (e.g. mobile robots).
- Propose an inclusive production planning problem that considers mobile robots as manufacturing resources (i.e. the Scheduling, positions Assigning and Routing problem (SAR) problem). Propose a notation for the elements, characteristics and assumptions of the SAR problem.
- Propose and develop a decision making methodology to select a SAR problem from a very large SAR problem space. The selection is based on the qualitative criteria of being realistic to industrial scenarios but solvable with optimisation methods.
- Implement the decision making methodology and evaluate a SAR problem case study.

1.3 Contributions

The completion of the aim and objectives leads to the following contributions:

- The first main contribution is a novel framework to offer global HVM services that is described in Chapter 3. The novelty of the framework consists of covering a wide number of technologies in a single comprehensive framework. The framework consists of a user interface and communications platform, a job allocation system, a network of factories and mobile manufacturing resources, Figure 1.2. The framework was proposed and developed through the analysis of highly flexible frameworks and systems. The framework was synthesised from the best working practices and technologies. Challenges and problems related to the framework were identified. The main challenges are designing and developing user interface and communications platform, designing and constructing the factories network and mobile manufacturing resources, and designing and developing the job allocation system. INTREPID was presented in the publication [29].

- The second main contribution is a novel and inclusive production planning problem presented in Chapters 4 and 5. The problem that considers mobile manufacturing resources within a single reconfigurable factory. This production planning problem is the core problem for the job allocation system, which can be later scaled to a cluster of connected factories and multiple distributed factories. The novelty of this production planning is the consideration of mobile robots to perform all the manufacturing tasks. The problem was synthesised from the analysis of the operations research problems such as scheduling, allocation, machine layout (i.e. positions assigning), vehicle routing; and robotics problems such as path planning and multi-robot task allocation. Also, based on this analysis, a notation for the production planning problem was proposed. The novel production planning was called the SAR problem due to the inclusion of the scheduling, positions assigning (i.e. machine layout) and vehicle routing problems. The SAR problem is highlighted within the novel framework in Figure 1.2.

- The third main contribution is a novel Decision Making Methodology (DMM) that is described in Chapter 6. The purpose of the DMM is to select a SAR problem that is realistic but solvable with optimisation methods. A vast number of possible SAR problems (i.e. SAR problem space) can be generated by the combination of all elements, characteristics and assumptions from operations research (i.e. scheduling, allocation, machine layout, vehicle routing) and robotics (i.e. path planning and multi-robot task allocation) problems. The novelty of the methodology is its application to a very large number of alternatives (i.e. SAR problems). The novel methodology introduced in this thesis is motivated by the controlled convergence method. The methodology works by aggregating elements, characteristics and assumptions to a core SAR problem through multiple stages. Metrics to evaluate the performance of the methodology were proposed. The methodology was implemented in LabVIEW and evaluated on an exemplar case study of the SAR problem in Chapter 7. A partial version of the methodology and its implementation was introduced in the publication [30].

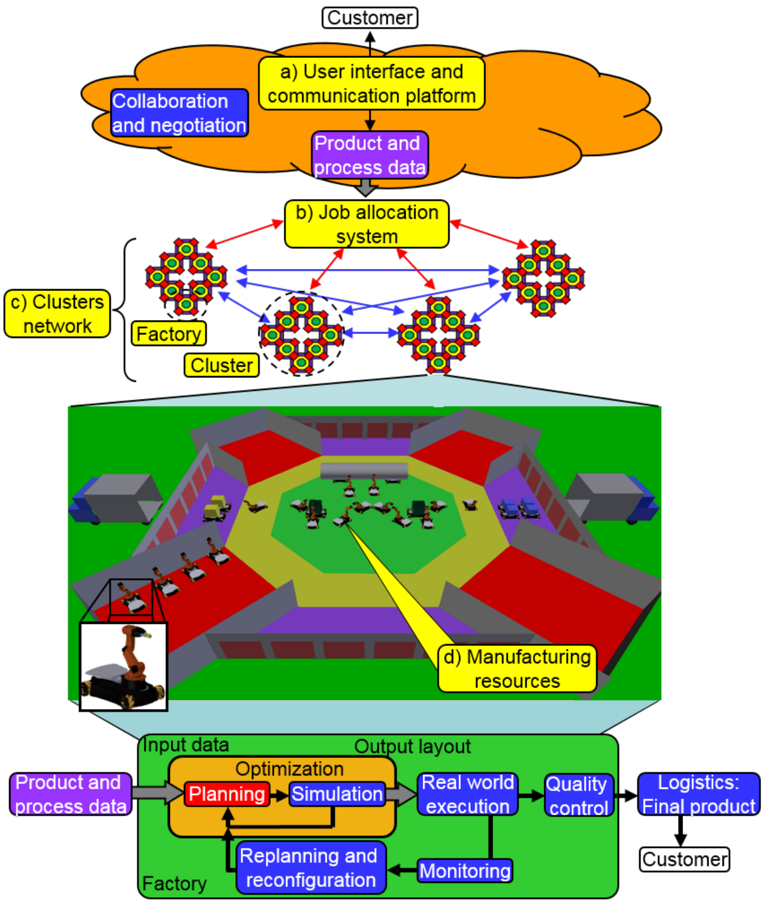


Figure 1.2: First and second contributions of this thesis. First, is the INTelligent REconfiguration for a raPID production change (INTREPID) framework. Second is the production planning problem with mobile manufacturing resources. The parts of INTREPID are shown in yellow rectangles. These are: a) User interface and communications platform; b) Job allocation system; c) Network of Reconfigurable Manufacturing Centres (RMCs); and d) Manufacturing resources (movable machines and Mobile Manufacturing Robots (MMRs)). Bidirectional blue arrows represent communications between RMCs and bidirectional red arrows represent communications between RMCs and the job allocation system. Each RMC is composed of interconnected factories. An example of a factory and production layouts with mobile robots are highlighted. At each factory, the production planning problem with mobile resources is solved. The production planning with mobile resources problem is the second contribution of this thesis. This problem is highlighted with a red rectangle. This problem refers to planning and simulation in an optimisation loop until the best possible production plan is determined. Product and process data from jobs is the input for this optimisation loop, whilst the output are production plans that are executed and monitored in real world. Real-time problems are managed through replanning and reconfiguration.

1.4 Publications

The research developed in this thesis has contributed to the following publications:

- V. M. Cedeno-Campos, P. A. Trodden, T. J. Dodd, and J. Heley, Highly flexible self-reconfigurable systems for rapid layout formation to offer manufacturing services, in 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 4819–4824.

- V. M. Cedeno-Campos, P. A. Trodden, and T. J. Dodd, An interactive methodology to explore optimization scenarios of a reconfigurable factory, in 20th IEEE International Conference on Emerging Technologies and Factory Automation, 2015.

- V. M. Cedeno-Campos, P. A. Trodden and T.J. Dodd, On the fundamentals of the production planning problem selection for mobile manufacturing robots in a self-reconfigurable factory, Journal of Robotics and Computer-Integrated Manufacturing (In preparation).

1.5 Outline

The rest of the thesis is organised in the following chapters:

Chapter 2 presents a literature review of manufacturing paradigms, models, systems and layouts. The paradigms are analysed in order to differentiate high added value from other types of added value. The types of added value by each paradigm are identified. The paradigms are classified according to their industrial age into Industry 1.0 (i.e. use of steam and water energy), Industry 2.0 (i.e. use of electricity and mass labour), Industry 3.0 (i.e. use of computers), and Industry 4.0 (i.e. use of wireless communications). The manufacturing models, systems and layouts are analysed in order to critically appraise their advantages and disadvantages (i.e. adequateness) to address the main HVM challenge (i.e. surpassing the valley of death [2]). The models, systems and layouts are classified according to their stages within the product and process life cycles. This literature review helps to formulate an approach to address the main HVM challenge in Chapter 3.

Chapter 3 presents the novel framework to address the main HVM challenge (i.e. surpassing the valley of death [2]). The name of the framework is INTREPID. INTREPID's vision is to offer global HVM services. The manufacturing paradigms and systems that motivated INTREPID are analysed in this chapter. The paradigms are networked manufacturing, cloud manufacturing, Industry 4.0, cloud robotics and the type of system are the Self-Reconfigurable

Manufacturing Systems (S-RMSs). INTREPID's operation, and its main parts are thoroughly described. These parts are a user interface and communications platform, a network of reconfigurable factories, a job allocation system and mobile manufacturing resources (i.e. S-RMS). The novelty of INTREPID is the inclusion of these four parts in a single comprehensive framework and the proposal of mobile robots and movable machines as manufacturing resources. Literature related to INTREPID's parts is presented, and desired characteristics are highlighted. These desired characteristics are included in the INTREPID' parts and their functionality. Challenges and related problems for INTREPID are identified. These are the design and development of a user interface and communications platform, the design and construction of the factories network and the mobile manufacturing resources, and the design and development of the job allocation system. The rest of the chapters focus on the challenge of developing a job allocation system for mobile manufacturing resources at a single factory.

Chapter 4 presents an analysis of the related problems for the production planning with the use of mobile resources. These problems are scheduling, machine layout or Facilities Layout Problem (FLP), and vehicle routing (i.e. routing). These problems are from the field of operations research, but analogous problems from the field of robotics are also analysed. These are the Multi Robot Task Allocation (MRTA) problem, which corresponds to the scheduling problem, and the path planning problem, that corresponds to the vehicle routing problem. The use of mobile resources lead to the proposal of a novel production planning problem called the Scheduling, positions Assigning and Routing problem (SAR). The novelty of the SAR problem is the inclusion of mobile robots and movable machines as manufacturing resources. The problems of scheduling, machine layout, FLP, routing, MRTA and path planning are analysed in order to identify their defining elements, characteristics and assumptions. The relationship of these elements, characteristics and assumptions to the SAR problem is emphasised. The basic variants of each of the analysed problems (i.e. scheduling, machine layout, FLP, routing, MRTA and path planning) are described and their relevance to the SAR problem is identified. Notations, taxonomies, and basic optimisation objectives of each of the analysed problems are identified whenever there exist any.

Chapter 5 presents definitions and general assumptions that define the core elements of the SAR problem. This chapter also proposes a novel notation for the elements and characteristics of the SAR problem. The novelty of the notation is the wide number of elements and characteristics that are covered. These relevant problems are from operations research (i.e. scheduling, allocation, machine layout, vehicle routing) and robotics (i.e. path planning and multi-robot task allocation). In this chapter, it is proposed to select the SAR problem to formulate with decision making methods in order to select a SAR problem that is realistic to industrial scenarios but solvable with optimisation methods. The realism and solvability of the SAR problem are given by the combination(s) of elements, characteristics and their assumptions in the formulation of the SAR problem. Elements, characteristics and their assumptions are considered as decision variables in the decision making problem. The decision variables are classified in three types: factory design and operation; production requirements; and manufacturing resources variables. These three types occur naturally due to the purpose of describing the production planning problem with mobile manufacturing resources. Hence, variables from these three types are called fundamental, whilst, a type consisting of auxiliary characteristics and assumptions, with the purpose of formulating the SAR problem with optimisation methods, is called system variables. The combination of decision variables (i.e. elements, characteristics and their assumptions) results in a vast number of SAR problem variants called the SAR problem space. An example of one SAR problem variant is presented and represented with the proposed notation. Then, a group of representative assumptions are applied on this SAR problem variant in order to demonstrate the effects of these assumptions on the SAR problem variant. Finally, the simplified SAR problem variant is represented with the proposed notation.

Chapter 6 presents a decision making methodology that helps select the appropriate SAR problem that is realistic but solvable with optimisation methods. The main working principle of the methodology is based on Pugh's controlled convergence method. In brief, the methodology works by aggregating partial groups of the elements, characteristics and assumptions (i.e. decision variables) to a core SAR problem through multiple stages. The novelty of this methodology is its capability of handling a very large number of alternatives by applying the controlled convergence method on subgroups of decision variables through multiple stages. Decision making methods which focus on selection are reviewed in this chapter. Specifically, the reviewed selection methods are from the area of concept generation and selection from the field of product design and development. These concept generation and selection methods have the advantage of promoting the use of abstract representations instead of detailed designs, models, or problem formulations. Working principles of the methodology are presented, and the algorithm to apply the methodology is explained based on these working principles. The algorithm consists of the following steps: (1) Decision variables collection (i.e. brainstorming or literature review); (2) variables grouping by their relationship, variables hierarchisation and weighting of the variables' option;, (3) variables selection to analyse at each stage, (4) variables combination; (5) aggregation into partial SAR problems; and (6) analysis and selection of the partial SAR problems. The steps of variable combination, analysis and selection occur iteratively for all the stages to analyse. Also, metrics to evaluate the performance of the methodology are proposed, and an implementation of the methodology in LabVIEW is described.

Chapter 7 presents the application of the decision making methodology to a case study from the SAR problem. This chapter is divided in three parts, where the first two parts correspond to the application of the DMM on the case study and the third part refers to the representation and the formulation of selected SAR problems with the methodology. The first part of this chapter corresponds to the first three steps of the methodology. The first step is the presentation of decision variables from the case study. The data collection step is performed with a literature review from Chapter 4. Step 2 (i.e. variables hierarchisation and grouping, and weighting of the variables' options) and step 3 (i.e. variables classification in stages) of the DMM are performed in the first part of this chapter. The variables are classified by the type of elements they describe (i.e. factory operation and design, production requirements and manufacturing resources). Steps 4 and 5 (i.e. variables combination and aggregation), and step 6 (i.e. SAR problem analysis and selection) are performed in the second part of this chapter. Steps 4 and 5 consist of combining variables' options and aggregating them to a core SAR problem to generate more detailed SAR problems at each stage. The results of these combinatorial and aggregation processes are graphs that help analyse and select SAR problems at step 6. Representative SAR problems selected at each stage are described to demonstrate the reasons for their selection (i.e. realism, feasibility and interest for exploring at next stages). Further analysis between one stage and its subsequent stage (i.e. intrastage analysis) is performed. Performance metrics of the methodology shows a reduction of the reviewed SAR problem space of 99.986% in comparison with the total SAR problem space. The reviewed SAR problems were 1,664 from a total of 12,582,912 SAR problems. The final selected SAR problems are 4 from the possible of 12,582,912 SAR problems. Therefore, these metrics demonstrate the importance and success of applying the methodology over very large number of alternatives (SAR problems in this case). The last part of this chapter presents the representation, with the proposed notation, and the formulation of selected SAR problems.

Chapter 8 presents a summary of the contributions made in this thesis, and conclusions about these contributions. These contributions are described in relationship to the objectives presented in Chapter 1. This chapter also presents future work for these contributions. Finally, within the future work, a strategy to formulate the four selected SAR problem is presented.

Chapter 2

Literature review

The importance of the manufacturing sector in the UK was identified in [31],[32]. From 2008 to 2013, the UK has decrease from being the 6$^{\text{th}}$ largest manufacturing countries in the world to being the 11$^{\text{th}}$. Moreover, the manufacturing accounted for 75% of research and development of business (R&D) up to 69% in 2013. The R&D in a business is responsible of generating innovative products based on knowledge (i.e. High Value Manufacturing (HVM)). The importance of HVM resides in the creation of novel products, processes and services that generate jobs and wealth to society. The main challenge for HVM enterprises is surpassing the valley of death [2]. The valley of death means increasing production volume from one-off products to low, medium and large scale volume [2].

Two causes for the valley of death have been identified. The first refers to whether or not there is enough demand for novel products, processes or services (i.e. product life cycle), whilst the second is whether or not there is enough production capacity to make these novel products (i.e. process life cycle) [33]. The second cause is in the interest of this thesis, specifically researching approaches that can help HVM enterprises to surpass the valley of death [2]. The two causes are interrelated because, on the one side novel technologies can introduce novel products into the market (i.e. technology-push). On the other side, market demand incentives the development of new technology to produce novel products or to produce them more efficiently (i.e. demand- or market-pull).

From the academic point of view, on the one hand, there are challenges related to create innovative products, processes and associated services. On the other hand, there are challenges related to create systems capable of efficiently producing high value products (HV products) [2]. Further to this, these systems must be flexible enough to produce multiple types of products and increase production volume from one-off products to low, medium and large scale. The last challenge is the topic of this thesis with scope of increasing production volume from one-off products to low production volume (i.e. 1 to 10 products per hour [3]). This implies researching on novel approaches that can provide HVM services for multiple products, where the production volume increases from one-off products to low volume [1].

Recalling a short definition of HVM from Chapter 1; HVM refers to the addition of value to products, processes or services through expertise and advanced technology [1]. However, different paradigms can provide different types of added value (i.e. shorter delivery time, customised products). Therefore, in this chapter, a review of the evolution of manufacturing paradigms up to the conception of HVM is presented in Section 2.1. This review highlights the evolution of paradigms by the type of value they add. This motivates the need for approaches, such as the framework INTelligent REconfiguration for a raPID production change (INTREPID), to address the main HVM challenge, surpassing the valley of death [2]. A specific literature on the state of the art of similar approaches to INTREPID is presented in Chapter 3.

A review of current manufacturing models, systems and layouts classified by their degree of flexibility is presented in Section 2.2. The manufacturing capabilities of these models, systems and layouts are contrasted, evaluated and desired characteristics to meet HVM challenges are emphasised. A gap for researching on highly flexible manufacturing systems is highlighted in this review. Consequently, there is need to research on systems that can adapt in order to produce multiple products and to increase production volume from one-off products to low production volume. Hence, this section introduces the novel highly flexible system (i.e. Self-Reconfigurable Manufacturing Systems (S-RMSs)) that primarily focus producing multiple products from one-off products to low production volume. S-RMSs are explained in detail within context of the framework INTREPID in Chapter 3. Conclusions about HVM and approaches to address their challenges are presented in Section 2.3.

2.1 Manufacturing paradigms evolution

A manufacturing paradigm is understood as a pattern that distinguishes a specific type of manufacturing model or system [34]. Well-known examples are mass production, lean manufacturing, Toyota production system, and mass customisation [35]. These paradigms have evolved from the use of rudimentary tools and hard labour to producing products in workshops (i.e. craft production) up to the use of computer assisted approaches to create and produce customised products [36].

All these paradigms evolved either from the application of technology (i.e. technology-push) or the need or desire from customers (i.e. market-pull) [37]. Technology-push paradigms provide desired features, characteristics, advantages that lure customers into buying them (i.e. added value). The application of one or multiple paradigms results in added value such as higher quality, lower cost, shorter delivery time, better service and more customised products or services [1]. These types of value are indicated in the paradigms description in the following four subsections.

The paradigms have been classified according to their product complexity, demand forecasting and manufacturing system flexibility in [38]; and according to date of introduction, production volume, product variety, and flexibility of the manufacturing system [39]. These classifications focus on enabling technologies or paradigm characteristics (e.g. flexibility), none of them is classified according to the value they can add to the products.

Therefore, a proposed classification is based on the industry stages. These stages are classified by the introduction of four significant technology-push stages. The four stages plus a pre-industry stage along with manufacturing paradigms classified by each stage can be observed in Figure 2.1. These stages and the associated paradigms are reviewed in the next four following subsections. These stages are characterised by the following technologies [40]:

- **Pre-industry**: hand tools for craftsmanship products (Subsection 2.1.1)
- **Industry 1.0**: water- and steam-powered machines start automating some processes (Subsection 2.1.1)
- **Industry 2.0**: electrically-powered machines, the introduction of production lines and the division of labour in specialised processes (Subsection 2.1.2)
- **Industry 3.0**: computer for production processes and more automated manufacturing systems (Subsection 2.1.3)
- **Industry 4.0**: wireless connectivity-assisted production processes and autonomous and decentralised automation (Subsection 2.1.4)

Paradigms of the Pre-industry (i.e. craft production) and Industry 1.0 (i.e. mechanisation) are not in use in industrialised countries. However, paradigms of Industry 2.0 and Industry 3.0 remain in use in industrialised countries. In fact, many of these paradigms coexist or complement each other. Paradigms from Industry 4.0 and forthcoming paradigms are still under development.

2.1.1 Pre-industry and Industry 1.0

The pre-industry stage is commonly known as craft production [36]. It refers to the use of workshops were workers use hand tools to perform most of the tasks by hand. It was necessary to commission jobs before their production. The results were tailored and unique products on the positive side but with large delivery

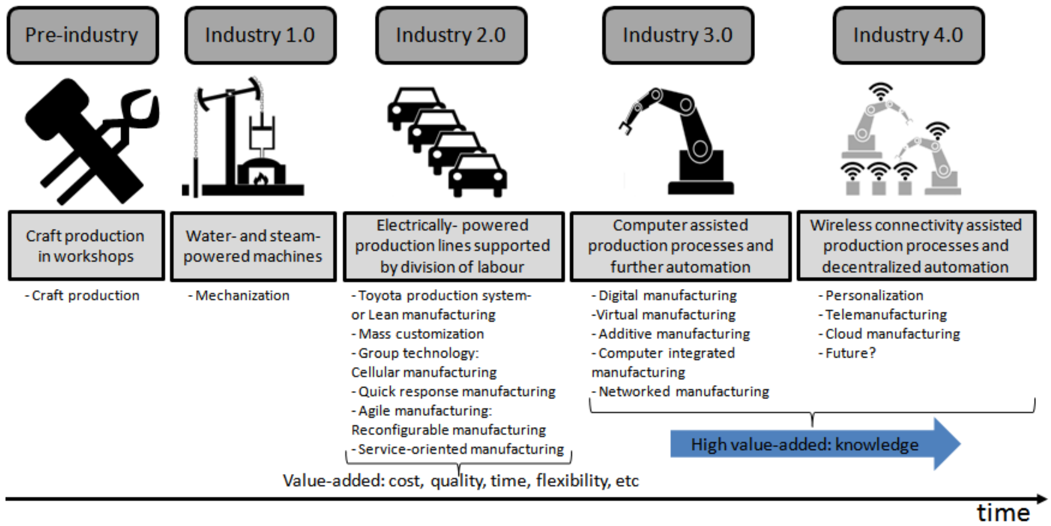
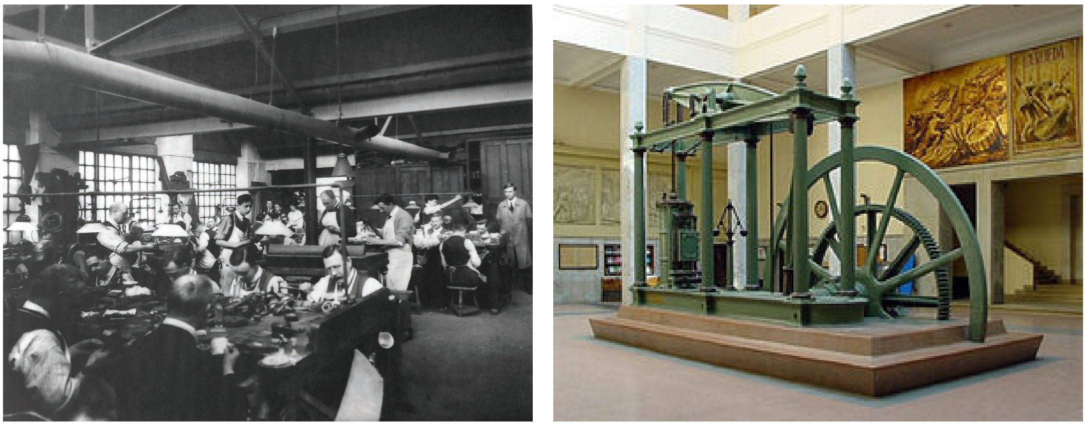


Figure 2.1: Classification of manufacturing paradigms by their industry stage over time and the value added by them. The horizontal axis corresponds to the chronological evolution of industry stages. The paradigms are approximately classified by the industry, not by accurate time they were introduced. The industry stages are divided by the main technological breakthrough in the manufacturing industry. It is shown a pre-industry stage and four industry stages. The stages and their main technological breakthrough are: pre-industry with hand tools, Industry 1.0 with steam and water power, Industry 2.0 with electricity, production line, and division of labor, Industry 3.0 with the use of PCs and Industry 4.0 with the use of wireless communications. The paradigms provide different types of value other than reducing cost, after the introduction of mass production and the production line. Added value based on knowledge (high added value) start its appearance from Industry 3.0 with the use of PCs and is reaching its full potential in Industry 4.0 with the use of wireless communications. Image modified, under Creative Commons Licence, from [42]



(a) (b)

Figure 2.2: Examples of production within the pre-industry and Industry 1.0 stages. (a) English craftsmanship from Deakin and Francis enterprise. This is an example of a craft manufacturing workshop. Image reproduced, with permission, from [43]. (b) Reproduction of James Watts steam engine. This is an example of Industry 1.0 and the use of steam to create machines. Image reproduced, under the GNU Free Documentation license, from [44].

times and high production costs on the negative side. Therefore, there was a need for approaches to automate the production. This need was addressed with the use of steam- and water-powered machines in Industry 1.0. An example of a workshop dedicated to leather products production can be observed in Figure 2.2a.

Industry 1.0 is known as the first industrial revolution or mechanisation [41]. It spanned from 1760 to 1840. This industrial stage started in Great Britain with the use of steam- and water-powered machines. The use of steam and water resulted in bulky and complex machines and mechanisms. These ones had to be carefully designed to take advantage of the flow of steam and water. The most relevant work in Industry 1.0 is the steam engine from James Watt [44]. On the positive side, these machines and mechanisms reduced the time and human labour required to make dull, dirty and dangerous tasks. Products were produced faster and in a cheaper way in comparison to the pre-industry stage. This increased the access to products to a wider but limited sector of the population due to need for expensive and complex machines. Also, the power source of these kind of machines required to have rivers or boilers closely and pipelines to transport water and steam. An example of a machine working with steam can be observed in Figure 2.2b. These challenges were addressed with the introduction of electricity and the development of machines in Industry 2.0.

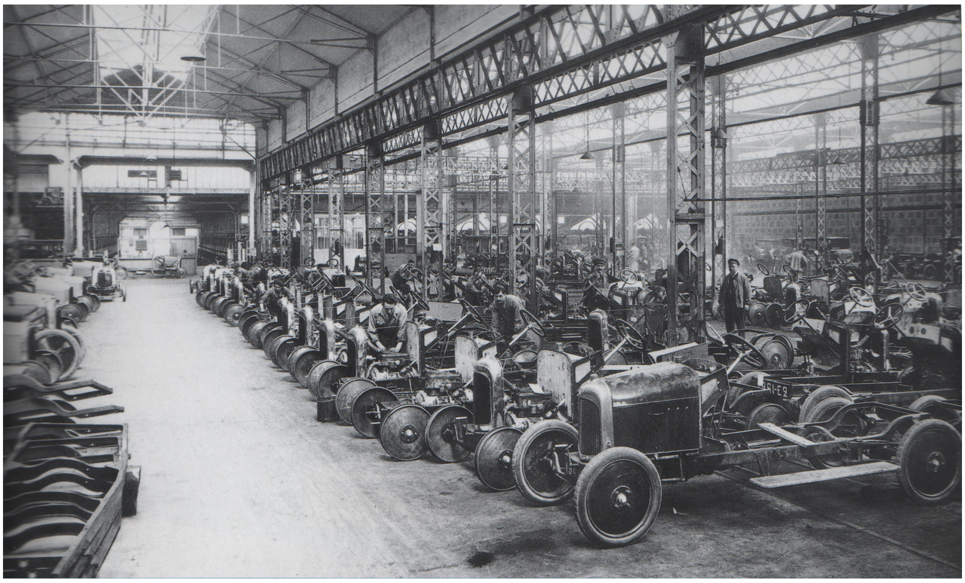


Figure 2.3: Example of a production line to produce the Citroen Type A. This is an example of the Industry 2.0. Image reproduced, under European Union Copyright law, from [46]

2.1.2 Industry 2.0

Industry 2.0 is known as the second industrial revolution [45]. This stage is characterised by the introduction of electricity, the production lines and division of tasks in specialised labour. The use of electricity facilitated the construction of smaller process dedicated machines with internal electric motors (i.e. machine tools, e.g. lathes, milling machines) [47]. The starting point and the most relevant paradigm of this stage is mass production [35]. Mass production was enabled by line layouts that connect the flow of parts and subcomponents. This minimised the time and cost to transport parts, subcomponents and assembly tools to workstations. The introduction of mass production resulted in the following three types of added value: larger volumes, lower costs and faster production times [35]. An example of the first production lines to Citroen Type A is shown in Figure 2.3.

The production line was inspired on the process industry or chemical industry, where the flow of materials is continuous [33]. This was similar to the steam- and water-powered machines of Industry 1.0. The production line evolved to achieve more efficient but inflexible systems known as dedicated manufacturing systems. These systems produce a single type of product but they have the greatest efficiency (i.e. more quantity in less time). Dedicated systems and production line layout along with other manufacturing systems (i.e. flexible, reconfigurable) and layouts (i.e. cell, functional) are analysed in Section 2.2.

The widespread use of production lines increased the competition among enterprises. However, this competition was mainly based on reducing production costs. This practice was unsustainable and eventually led to the surge of paradigms that added value to products instead of reducing their costs [35]. These paradigms including mass production are still in practice. These paradigms are:

1. Toyota Production System (TPS) or Lean Manufacturing
2. Group Technology (GT)
3. Mass Customisation (MC)
4. Quick Response Manufacturing (QRM)
5. Agile Manufacturing (AM)
6. Service-Oriented Manufacturing (SOM) or Servitisation

1) Toyota Production System TPS started in Japan, and it reached its best performance with the automobile enterprise Toyota [48]. Later, it was spread to the U.S. and it became known as lean manufacturing [49]. TPS focuses on reducing or eliminating processes that do not add value to final products. The TPS have a set of enabling approaches that lead Japanese car manufacturing industry to be number one before 1990 [48]. The most important enabling approaches of TPS are [50]:

- **Just In Time (JIT)** and **Kanban**. JIT is an approach to manage the supply chain in order to keep a low inventory. JIT refers to request raw materials or subcomponents only of the nearest scheduled production. Kanban is a signalling system to request only the necessary raw materials or subcomponents to be processed. These approaches facilitate contracting by capability of production services and increases responsiveness to dynamic production orders.

- **Single-Minute Exchange of Die** refers to minimising the time to exchange production setups to produce a different product. Consequently, there is an increment in the product variants that can be manufactured.

These approaches add the following types of value: increased responsiveness with just in time and more product variety with single-minute exchange of die. Other approaches within TPS such as 5s, Poka-yoke and Kaizen result in cost reductions through improving the production efficiency and constant improvement respectively [50].

2) Group Technology GT is a paradigm that focuses on improving manufacturing efficiency by grouping together similar products in product families and their respective manufacturing machines [51]. As a result, the layouts are specific to these product families. This reduces the time of transportation between machines. The layout associated to this paradigm is known as cell or cellular layout, whilst a factory space dedicated to a cell layout is known as a manufacturing cell [52]. A layout can be designed beforehand to produce a specific product family. Consequently, the added value of this paradigm lies in the wider variety of products that can be produced compared to mass production.

3) Mass Customisation MC refers to producing customised products at similar prices than mass production [53]. MC is supported on GT to produce a product family at low costs with similar efficiency than production lines. MC offers customisation at superficial and cosmetics levels in contrast to fully personalised products [35]. MC main strategy is to have core parts or modules in a product and the user selects cosmetic or superficial parts or modules to customise the product (i.e. platform-based product [54]). This also facilitates the development of new products based on previous common platforms. Production strategies based on platform-based products are delayed differentiation (also known as postponement) and assembly to order [55],[56]. Delayed differentiation refers to producing a core product that is fully finished only when a customer makes an order. Similarly, in assembly to order the product is assembled when an order is placed (e.g. Dell computers). The value added by this paradigm is customised products.

4) Quick Response Manufacturing QRM focuses on reducing the lead time (i.e. time from a production order is requested until the time the order is met) of the production processes [57]. This paradigm is a successor of TPS. Whilst TPS focuses on reducing or eliminating non-value-added processes to reduce cost, reduce lead time and improve quality; QRM focuses on reducing lead times in order to reduce cost, eliminate non-value-added processes and improve quality. Consequently, enterprises are capable of producing products on demand. The added value of QRM is short time to manufacture products according to orders with variable demands for each product.

5) Agile Manufacturing AM is a paradigm that aims at producing mass customisation products by rapidly changing production capacities between products [58],[59]. Therefore, it is important to previously have implemented TPS and QRM first. The scope of this paradigm is usually at the organisational level, whilst at the production level is supported by Reconfigurable Manufacturing Systems (RMSs) [60]. These are systems capable of changing production settings and setups to produce different products. The added value of this paradigm is customisation of products at a lower level than MC but rapid response to customer's desires.

6) Service-Oriented Manufacturing SOM refers to developing products and their associated services as a single product (i.e. product-service system -PSS-) [61]. A representative example is the leasing of power-by-the-hour instead of transferring ownership of the gas turbine to airlines [62]. This paradigm is also known as servitisation [61]. Firstly, the servitisation paradigm focused on changing business model from producing products to offer the associated services for their products (e.g. maintenance, updates, overhaul, recycling), but later it focused on the design of PSSs [63]. The added value of SOM is customised service.

In summary, with the paradigms from Industry 2.0, enterprises started competing by adding value rather than by reducing costs. Also, there is a trend towards leaner, more responsive and agile enterprises in order to produce customised products at similar efficiency than mass production (i.e. similar production costs and times). Paradigms from Industry 3.0 further focus on improving the efficiency of manufacturing systems and improve the automation through the use of computers and communications networks.

2.1.3 Industry 3.0

In Industry 3.0, the use of computers is predominant to facilitate most of the production processes [40]. In general, the use of computer assisted tools and platforms has the following advantages: increased collaboration of customers in product design, further automation to produce in more dynamic environments, versions management of product variants (e.g. drawings) and constant innovation of products. Industry 3.0 is characterised with the introduction of added value based on knowledge (high added value) for product development. The main paradigms of this stage are:

1. Digital Manufacturing
2. Virtual Manufacturing
3. Additive Manufacturing
4. Computer Integrated Manufacturing
5. Networked Manufacturing

1) Digital manufacturing refers to the use of PCs to develop (i.e. design and simulation) products [64]. Designs are created through the use of Computer Aided Drawing (CAD) [65]. Digital designs can be simulated and analysed with computer-aided simulation tools such as finite element analysis for static and dynamic analysis, multibody analysis for kinematics and dynamic analysis and computational fluid dynamic for fluids and thermal analysis. If the designs are manufactured with machine tools, the production processes can be simulated through computer-aided manufacturing software. Computer-aided is applied to a broad variety of production processes. The most relevant processes are in the area of production planning are process planning, enterprise resource planning and manufacturing resource planning. In general most of these tools are grouped under the umbrella of Computer Aided Technologies (CAx) [65]. The value added with this paradigm is to support developing customised products and allowing customer to visualise final results before production.

2) Virtual manufacturing is a paradigm based on the use of the tools from digital manufacturing to monitor, simulate and visualise real-time models of production processes, systems or a whole factory [66]. A virtual environment of the physical world allows simulating models and processes prior to their implementation. This speed up not only the development of products but also their manufacturing processes. Virtual manufacturing also allows the visualisation and real-time monitoring of processes and manufactured products leading to the creation of a virtual factory [67]. The added value of this paradigm resides on the rapid development of products and processes; hence, it can be used for customised products.

3) Additive manufacturing refers to the production of products by depositing layers of material instead of removing material [68]. Therefore, this paradigm is also known as layer manufacturing. Other common names for this paradigm are 3D printing and Rapid Prototyping (RP) [69]. Additive manufacturing is supported on digital manufacturing for the creation of 3D digital drawings (CAD files) that later are converted to additive manufacturing format files [68]. The process consists of slicing the 3D parts in layers according to the material and machine characteristics. Nowadays, RP machines are used to produce irregular shapes that are hard or impossible to produce with traditional equipment (i.e. equipment that removes materials). The use of limited materials (special plastics and metals) is an important drawback of this paradigm [70]. The added value of this paradigm is producing customised products with irregular shapes at rapid production rates.

4) Computer integrated manufacturing Computer Integrated Manufacturing (CIM) refers to the integration of isolated areas of manufacturing (e.g. planning, purchasing, inventory control, manufacturing areas, testing) through the use of communication networks [71],[72]. The most common network standards are Process field bus (PROFIBUS), Process field net (PROFINET) and Ethernet [73]. CIM makes use of computer controlled equipment (e.g. robots, Computer Numerical Control (CNC) machines, machine tools with numeric control, Automated Guided Vehicle (AGV)) to control and monitor distributed manufacturing resources [74]. Examples of this kind of computers are Programmable Logic Controllers (PLCs) and programmable automation controllers. The added value of this paradigm is real-time monitoring of the production progress and increase level of automation and production efficiency.

5) Networked manufacturing Networked Manufacturing (NM) focuses on production at a network of distributed factories [75]. The main challenge of this paradigm resides in the coordination of coupled production and supply chains. Therefore, real-time and robust communications and data interchange are vital for production coordination. Additionally, product development within a network requires support systems for collaboration. Web-based manufacturing systems facilitate collaboration [76]. This paradigm is basis for INTREPID due to the need to coordinate factories in a network. The added value with this paradigm resides in the possibility to produce products tailored to countries or regions.

In summary, in paradigms from Industry 3.0, the use of PCs facilitates many of the production processes, increases the automation level, improves product data storage and retrieval and facilitates the creation of new products and processes based on previous ones. Also, there is a trend towards collaborating at different production processes through the use of networking tools. This trend towards collaboration continues in the paradigms from Industry 4.0 but with wireless communications. Further to this, these wireless communications endow manufacturing resources with the ability to make decentralised and autonomous decisions.

2.1.4 Industry 4.0

Industry 4.0 is the newest proposal for an industry stage. In this paradigm, Cyber Physical Systems (CPS) (i.e. mechatronic devices with wireless connectivity) communicate and coordinate with each other through the Internet of Things (IoT) in order to manufacture [77]. This occurs in smart factories where conditions about the factory are shared through the internet [78],[79]. The design principles of Industry 4.0 are [79]:

- **Interoperability**: CPS, persons, and companies are connected through communications networks.
- **Decentralisation**: CPS are capable of taking autonomous decisions.
- **Modularity**: CPS are designed to be modular with plug and play characteristics.
- **Real-time**: monitoring of factory conditions and resources data.
- **Virtualisation**: factory and resources data are updated in a virtual model for monitoring and model simulation purposes.
- **Service-oriented**: CPS, persons and companies offer their services through the Internet of Services -IoS- in a similar way to cloud manufacturing.

Industry 4.0 relies on communications networks for collaboration and for sharing and getting data (IoT). In Industry 4.0, the IoT is used to coordinate tasks between resources, which result in decentralised, autonomous and knowledge-based decisions (automation). Other paradigms that similarly rely on the use of communications networks for manufacturing are:

1. Telemanufacturing
2. Cloud manufacturing or Cloud Based Design Manufacturing (CBDM)

1) Telemanufacturing proposed the use of telecommunications networks to transmit product design data and manufacture near the distribution locations [80]. Telemanufacturing proposes the use of manufacturing dedicated centres with up to date flexible manufacturing systems and the experts to use them. The purpose of telemanufacturing is to offer manufacturing services with the use of these centres. The added values include the selection of provider depending on the product and market, no investment in infrastructure and the use of the newest technology and expertise [29]. In paradigms focused on providing services (i.e. telemanufacturing and cloud manufacturing), the added value makes a difference to both the customer requesting the manufacturing services and the final customer that receives the manufactured products.

2) Cloud Based Design Manufacturing (CBDM) or Cloud Manufacturing is based on networked manufacturing and it makes use of distributed manufacturing facilities [81]. CBDM offers a complete model and computing architecture to access manufacturing services through the internet [82]. CBDM main enabling technologies are derived from cloud computing [6]. These ones are Internet of Things, virtualisation, service-oriented technologies (i.e. service oriented architecture, web service, and semantic web), and advanced computing technologies (i.e. high-performance computing, ubiquitous computing and cloud computing itself). The aim of cloud manufacturing is to offer manufacturing services in a similar way that cloud computing offers computing services. Therefore, INTREPID is based on cloud manufacturing and its well established model and supporting technologies. The values added by this paradigm consist of accessing new technologies and contracting by capability. Also, more collaboration for product development is feasible.

Table 2.1: Manufacturing paradigms are classified by their types of added value. The paradigms are shown in the vertical axis whilst the types of added value in the horizontal one. The symbol plus (+) weights the degree of added value by a paradigm. More symbols indicate a higher weight by a paradigm.

	Types of added value											
Paradigms	More variety (customisation)	Personalisation	Reduced cost	Reduced time	Improved quality	Better service	Attractive features to products	Entrance to new markets	Products tailored to markets	Contract by capability and responsiveness	Production monitoring	Expertise and knowledge
Craft production		+										
Mechanisation			+	+								
Mass production			++	++								
Toyota production system (Lean)	+				++					+		
Group technology	++											
Mass customisation	+++											
Quick response manufacturing										++		
Agile manufacturing	++++											
Service-oriented manufacturing						++						
Digital manufacturing	++											+
Virtual manufacturing	++										+	+
Computer integrated manufacturing											++	
Networked manufacturing								+				
Additive manufacturing	++	++	+	++			++	+	+	+	+	++
Telemanufacturing								++	++			
Cloud manufacturing								+++	++	++	++	++

In summary, the paradigms evolution shows a trend to produce a wider variety of products (e.g. group technology, mass customisation) through a more responsive and agile adaptation to dynamic changes in the product demand (e.g. quick response manufacturing, agile manufacturing). Another trend is about offering manufacturing services supported by communications networks (e.g. networked manufacturing, telemanufacturing cloud manufacturing). In specific, communications networks are used for product development in collaboration or to manufacture in different countries and access new markets and make use of advanced manufacturing technology. A summary of types of added value by each paradigm is shown in Table 2.1.

Considering these trends, one of the contributions of this thesis is the proposal of a comprehensive framework, INTREPID, that brings together these trends to address the main acHVM challenge. INTREPID is based on the paradigms of networked manufacturing and cloud manufacturing, as well as an enhanced version of the RMS (i.e. the S-RMS). It is also based on the design principles of Industry 4.0 and cloud robotics. In the next section, a review and analysis of how current manufacturing models, systems and layouts are integrated within these paradigms is done. Also, the issues and challenges to address HVM are highlighted and the S-RMS are introduced and contrasted against current manufacturing systems.

2.2 Manufacturing models, systems and layouts

The paradigms described in Section 2.1 demonstrate the variety of added value that can be offered to customers. According to the broad definition of HVM given in Chapter 1, HVM refers to achieve any or many of the types of added value through the use of knowledge (i.e. expertise at using advanced designing and manufacturing techniques) [1]. The paradigms make use of different approaches to achieve different types of added value. An example of these approaches is delayed differentiation from the mass customisation paradigm [55]. However, other approaches include the use of manufacturing models, systems and layouts. Examples of these ones are line and cell layouts that are used in mass production and group technology paradigms respectively [35],[52]. This section describes and analyse manufacturing models, systems and layouts.

Recalling that the main challenge for HVM is to survive the valley of death [2]. This requires the production volume and efficiency to increase in order to match the product demand. Production volume and demand commonly increase from one-off products to low, medium and large volumes [33]. This means a change of manufacturing model, from job production to batch production and finally to mass production. Accordingly, changes in the manufacturing models are reflected in changes of the associated manufacturing systems and layouts [33]. Therefore, in this section, the main models, systems and layouts are presented and analysed. In Subsection 2.2.1, manufacturing models are described, whilst manufacturing systems and layouts are described in Subsections 2.2.2 and 2.2.3 respectively. In Subsection 2.2.2, different types of flexibility in manufacturing systems are summarised. Also, the novel Self-Reconfigurable Manufacturing Systems (S-RMSs) are introduced and its types of flexibility are described in Subsection 2.2.2. Models, systems and layouts are associated with each other in Subsection 2.2.4.

The manufacturing capabilities of models, systems and layouts are contrasted and evaluated resulting in the identification of desired characteristics to meet the main HVM challenge (i.e. surpassing the valley of death [2]). Conclusions of this review show a gap for researching on highly flexible systems, which can easily and automatically adapt to produce multiple products and to increase production volume from one-off products up to large volumes. Therefore, as an intermediate step, one objective of this thesis is to propose a manufacturing system that is flexible enough to change from one-off products to low volume. Hence, a highly flexible system (i.e. Self-Reconfigurable Manufacturing System (S-RMS)) is introduced and its types of flexibility described in Subsection 2.2.2. The S-RMS is explained in Chapter 3 together with a complete framework to offer global HVM services (i.e. INTREPID).

2.2.1 Manufacturing models

Manufacturing models refer to the way a product is produced. A model might involve other areas different from manufacturing (e.g. raw materials and products storage, raw materials supply and products distribution management). A manufacturing model determines the production volume that is produced and how products are produced. The models for discrete manufacturing are classified according to production volume and process organisation (i.e. production flow) in Figure 2.4. The most common models are [33]:

- **Job production** refers to the production of a single product. This model is characterised by one-off products or low production volume and a large mixture of products. Products in this model might require repetitive operations with the same machines in complex sequences. Hence, the production flow is irregular and it is common to locate machines separated from each other to increase the flexibility (i.e. disconnected flow). Consequently, it is important to solve the problems of sequencing, schedule and routing products between machines to avoid production delays and job overlapping [83],[84],[85]. Examples of products produced under the job manufacturing model are large and complex products such as bridges, ships, aircrafts and factories, or specialist and tailored products such as clothes, individual computer programs and satellites [86].

- **Batch production** is related to the production of products in low or medium volume groups. Similarly to job production, this type of model can produce a mixture of different products but in a higher production volume. Moreover, the production demand might be a mix of different products. Common examples of products produced under this model are furniture, clothing (e.g. shoes), sports equipment and food (e.g. bread) [86]. The characteristic of this model is the use of the same resources to produce products that require similar operations. However, in this type of model some of the machines are organised in lines that produce a similar sequence of operations for as many products as possible (i.e. semi-connected flow). As a result, there is an increase in the efficiency at cost of losing flexibility. Similarly to job production, challenges for batch production refer to solve the problems of sequencing and scheduling of jobs to machines to minimise production time and lot sizing [87],[88].

- **Mass production** refers to producing a single product in large scale. This model is the more efficient for large volumes but is incapable of producing more than one type of product unless there is reconfiguration of the manufacturing system [35]. Still, this reconfiguration might take weeks or months depending on the type of product and the complexity of the manufacturing system. Examples of products produced under mass production are soap powder, canned drinks, electronics (e.g. cell phones, computers, printers) and automobiles [86]. In this model, the machines are organised in a line that matches the sequence of operations of the single product (i.e. connected flow). Consequently, this model is the most efficient. The main challenge to be addressed is the adequate design of manufacturing systems that minimises the movement of products between machines and allow a continuous and non-stoppable flow of products. In case of the assembly of products, the main challenge is the adequate division of assembly tasks between workers with the objective either to minimise the number of workers or to minimise the average production time [89].

These manufacturing models are mainly characterised by two factors, namely their production flow (process organisation) and the production volume of products [33]. The production volume is defined by the expected product life cycle (i.e. introduction, rapid growth, maturation and commodity or decline). This is, a novel product is introduced to the market, later the product demand grows rapidly until it reaches maturation. Later, there is a point when a product becomes a commodity when is widely demanded or the product declines otherwise. This occurs similarly with the corresponding manufacturing process that changes along

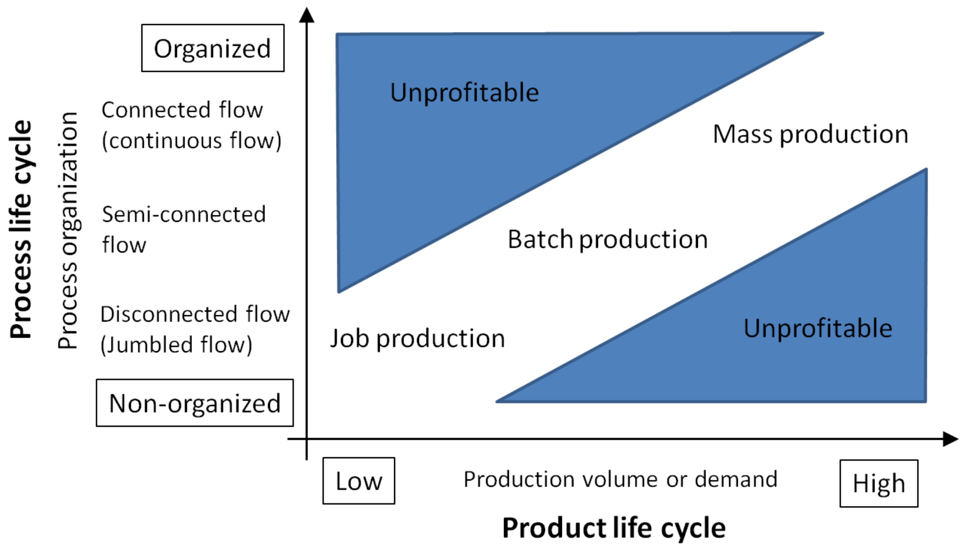


Figure 2.4: Classification of manufacturing models according to their production volume (i.e. product life cycle) and process organisation (i.e. process life cycle). The horizontal axis shows production volume the models can produce whilst the vertical axis shows different types the production flow can be organised. Three main models are feasible, whilst the rest of the combination are within the unprofitable blue area. The three main models are job production, batch production and mass production. Job production is characterised by one-off products or low production volume with a disconnected production flow. Batch production can produce medium production volume with a semi-connected production flow. Mass production can produce large production volume with a connected production flow. The desired evolution of high value products (HV products) should increase from low to medium and large production volume and vice versa for a smooth decline. Adapted from [33]

the process life cycle to meet product demand. The process life cycle has the same stages as the product life cycle. In order for HV products to survive the valley of death, their manufacturing processes have to evolve accordingly with the product demand until product and process reach maturity and become a commodity [2].

There are combinations other than job, batch and mass production. However these ones are not feasible [33]. For example, a high volume can be produced with a process organisation such as semi-connected or disconnected flow but it will not be as efficient as a connected flow. Similarly, a variety of products might be produced with a connected flow, but it requires adjustments to the machines or even the complete reconfiguration of the machine layout. If an enterprise works within any of the unprofitable combinations, its operation is not efficient and a change at the manufacturing systems is required (i.e. reconfiguration). The production flow or process organisation is in agreement with the type of manufacturing systems and layouts that are used. Therefore, the next two subsections describe these ones.

2.2.2 Manufacturing systems

The manufacturing models are related to the manufacturing systems and the arrangement of their machines (i.e. layout). Types of manufacturing systems are a single machine to perform a single task (e.g. machine tools), a single machine doing several tasks (e.g. Computer Numerical Control (CNC) machines), or a group of machines doing several tasks (e.g. a job shop) [90]. Examples of manufacturing systems are: Manually controlled machine tools, Computer Numerical Control (CNC) machine tools, assembly robots, assembly workers or mechanisms and special machines designed to produce a unique operation (e.g. complex braided machines [91]). Manufacturing systems are classified by their flexibility in three main types. These are [4],[5]:

- **Dedicated Manufacturing Systems (DMSs)** are designed to produce a single operation. This type of systems are related to the mass production model. Thus, examples of products produced with DMSs are soap powder, canned drinks, electronics and automobiles [86].

- **Flexible Manufacturing Systems (FMSs)** are able to produce wide range of products. FMS can be designed with one or many types of flexibility. These types are explained in the following paragraphs. FMSs are related to the job production model, and therefore, examples of products produced with FMSs are large and complex products such as ships and aircrafts, or specialist and tailored products such as clothes, individual computer programs and satellites [86].

- **Reconfigurable Manufacturing Systems (RMSs)** are novel systems that aim to extend the types of flexibility of the FMSs. RMSs are systems designed to be easily reconfigurable. Examples of reconfiguration are interchangeable modules, or the addition, removal, replacement or reordering of machines in a any layout [92],[60]. As a result a machine is capable of producing a completely new operation or a new product respectively. Examples of products produced with RMSs are furniture, sports equipment and foods (e.g. bread) [86].

The flexibility and efficiency in these three types of systems depends on factors such as the type of machines used and their layout, as well as the level of automation of the machines, and the level of automation of the material handling systems to transport, load and unload products between machines [93]. The arrangement in line of these machines can facilitate the movement of materials and making them more efficient at producing a single product. In contrast, an arrangement with spaces between machines facilitates the transportation of products that require multiple and repetitive manufacturing operations and when products to be produced are constantly changing. Different types of flexibility have been identified in the literature [94]. These types are focused on FMSs, but can be applied to RMSs. They are summarised in the following paragraphs.

Types of flexibility in FMSs and RMSs

Many authors agree with the definition of the terms flexibility and reconfigurability given by Tolio in [94]. Thus, flexibility is the ability of a system to change its behavior without changing its configuration whilst reconfigurability is the ability to change its behavior of a system by changing its configuration. The use of the terms flexibility and reconfigurability in this thesis are in agreement with Tolio's definitions. The following types of flexibility are relevant to this thesis [94],[95],[96],[97]:

- **Machine**: Number of operations carried out with the same set-up in a machine
- **Process**: Different set of parts that the system can produce with minimal set-up changes
- **Functionality**: Refers to the ability of a single machine or a system of machines to produce a different operation or product respectively
- **Product**: Easiness to change produced products with the same set of parts
- **Material handling**: Easiness to move parts among different machines

- **Operation**: Number of different operations sequences available to manufacture one part
- **Routing**: Number of routes (sequences of machines) to produce the same features in a part or the part itself
- **Volume**: Number of profitable production volumes with the same manufacturing system
- **Control system**: System's potential to autonomously and continuously work for long hours
- **Expansion or scalability**: Easiness to expand the production capacity or capability of the system. This type of flexibility is usually obtained through modularity of resources and systems
- **Production**: Quantity of part types which can be manufactured with the same equipment

These types of flexibility are not enough for the current challenges of a global dynamic demand with challenges for highly customised and high value products [98]. The flexibility is limited due to the design of machines or a system of machines. However, the flexibility can be enhanced through the reconfiguration of machines or the system of machines [60]. For example, the machine flexibility is increased by adding or changing modules of the machine to perform different operations [92]. Current manufacturing systems are preselected and predesigned depending on the expected or estimated production volume and variety of products to produce (i.e. product mix). However, a further step in the evolution of the RMSs are the Self-Reconfigurable Manufacturing Systems (S-RMSs). The aim of these systems is an autonomous and rapid reconfiguration of a machine or a system of machines in contrast to the manual reconfiguration of the RMSs. These are introduced in the following paragraphs.

Introduction to the self-reconfigurable manufacturing systems (S-RMSs)

A highly flexible type of manufacturing systems is necessary due to the challenges of a global and dynamic economy (i.e. dynamic and uncertain demand, aim for highly customised and high value products). Current manufacturing systems and layouts cannot manage these challenges [98]. Therefore, it is necessary systems that can rapidly and automatically change from one-off products to low production volume (i.e. less than 50 products) for a group of different products. A proposed approach to address these challenges and increase manufacturing flexibility is the use of Self-Reconfigurable Manufacturing Systems (S-RMSs).

S-RMSs consist of logistics robots (AGVs), mobile manipulators with interchangeable tools, flexible grippers and fixtures (i.e. fixtureless assembly [99]), and movable machines for advanced processes (i.e. difficult to compact and embed in modular tools) [29]. These novel systems can easily be adjusted to manufacture in different manufacturing models depending on the required production volume and variety. The S-RMSs are capable of producing layouts bespoke to the current products to be produced and their demands. These systems can easily change between layouts to either maximise production efficiency or production variety.

The S-RMS is one of the contributions of this thesis, and it is fully explained in Chapter 3 along with the novel framework INTREPID to address the main HVM challenge. The S-RMS and INTREPID are based on the Industry 4.0 principles [77]. A description of how the S-RMS work under the Industry 4.0 principles is the following:

- **Interoperability**: S-RMSs share the principle for connectivity between manufacturing resources for coordinating an optimal production plan.
- **Decentralisation**: Due to the use of mobile manipulators the decentralisation of decisions is guarantee by their embedded decision systems.
- **Modularity**: S-RMSs works under the same principle to form different layouts or adjust the production volume according to demand.
- **Real-time**: The interconnectivity of the S-RMS facilitates the real time monitoring of the S-RMS.
- **Virtualisation**: Real-time monitoring of positions and performed tasks of the S-RMSs allow easy virtualisation. Hence, this information allows determining a new layout if production requirements change.
- **Service-oriented**: S-RMSs are designed to work under the cloud manufacturing paradigm. Thus, the S-RMSs are embedded within the framework INTREPID to offer global HVM services.

The types of flexibility of the FMSs do not consider the ability to reconfigure a machine or a system of machines to perform a different operation or a different group of operations. Furthermore, an autonomous and rapid reconfiguration can enhance a system manufacturing abilities [93]. Therefore, in the following paragraphs, the types of flexibility that are achieved or surpassed through the S-RMSs are explained.

Types of flexibility achieved through the S-RMSs

S-RMS can achieve and sometimes surpass all types of flexibility achieved through the FMSs. This is, due to the use of mobile manufacturing robots and layout redesign. The use of mobile manipulators (i.e. mobile platforms with arm robots) results in increased machine flexibility because it is possible to achieve different axis and orientations. This is contrast to conventional reconfigurable systems, which require manual reconfiguration [60]. Also, the use of the mobile manipulators facilitates achieving an enhanced degree of material handling flexibility. Reprogrammation or adaptation of the mobile manipulators facilitates achieving process and production flexibilities. This has similarity with adaptable and intelligent CNC machines without a limited space for manufacturing.

The use of flexible grippers facilitates implementing fixtureless assembly [99]. Consequently, flexible grippers facilitate achieving product flexibility. The possibility of changing the layouts and the operations that each robot can execute allows achieving higher levels of routing and operation flexibilities than with traditional manufacturing systems. Traditional systems such as dedicated, flexible and reconfigurable systems make use of Programmable Logic Controllers (PLCs) and machine controllers in order to change operations routes (i.e. routing) and operation type, respectively. The possibility of changing the production layout according to the required production volume guarantee enhanced levels of volume flexibility and scalability. Finally, control flexibility can be achieved through the use of the autonomous robots.

The enhanced types of flexibility allows achieving dynamic layouts where resources are constantly moving and there is no distinguished or fixed layout (i.e. resources are shared between more than one product layout). An important factor of the manufacturing systems is the layout in which its machines are arranged. Hence, current manufacturing layouts, their advantages and disadvantages are analysed in the next subsection.

2.2.3 Manufacturing layouts

The flexibility of the manufacturing systems depends on their machine layout. A layout refers to the physical arrangement of machines. The closer the machines are located, the shorter the transportation time between machines is, but the less flexible the manufacturing system is. Therefore, the layout selection depends on the certainty and the prior knowledge of the products to be manufactured and its frequency. The manufacturing system layouts influence the selection of the type of material handling system and its design. Examples of material handling systems are Automated Guided Vehicles (AGVs), conveying lines and industrial robots (i.e. fixed). The main layouts are [102],[103],[100]:

- **Line**: also known as product oriented layout. Line layout refers to arranging machines in line in the specific sequence a product is produced. The objective is to minimise the required time to transport products between machines. Line layout is relevant for assembling products and it has an important problem known as the assembly line balancing problem [89]. The objective is to solve a trade off between minimising the number of workers and minimising the average working time of each worker. Relevant types of line layouts are: U-type (a U-shape curvature facilitates a worker to rotate and perform two to three operations), double-sided or two-sided (machine at two sides of a line).

- **Functional**: also known as process-oriented layout. In contrast to line layouts, the machines are arranged in groups capable of performing similar operations. This concentrates the necessary auxiliary equipment and expert workers. Functional layouts are commonly known as job shops. This layout facilitates the production of different products at low production volumes. Therefore, it is usually associated with the job production model. The most relevant problem is related to scheduling job to job shop types such as one machine, parallel identical machines, uniformly related machines and unrelated machines [84],[85].

- **Cell**: also known as cellular layout. Cell layouts are hybrids between process-oriented and product-oriented layouts. This layout combines the flexibility of process-oriented layout at a similar efficiency than process-oriented layout. Cell layouts does this by grouping together products in product families and their manufacturing machines in family-oriented cells [52]. Cell layouts belongs to the Group Technology (GT) paradigm [51]. This type of layout is also known as group shop, derived from the term group technology. The cell layout has line layouts for manufacturing sequences with high volume and machines located outside the lines for low volume. The design of this layout has to group correctly products in order to minimise the number of used machines and maximise the number of produced products.

All these types of layouts have their own limitations, advantages and problems to be solved. The limitations and advantages were explained in this subsection. Each type of model, system and layout have their own problems to be solved. Relevant problems to the modelling of the production planning with the use of S-RMS are reviewed in Chapter 4. In the next subsection, relationships between models, systems and layouts are discussed and feasible combinations are highlighted.

2.2.4 Relationships between models, systems and layouts

Layouts, manufacturing systems and models can be combined in different ways, but only a few combinations are feasible (i.e. profitable for an enterprise). These combinations depend on the estimated product demand and variety to be produced. The feasible combinations are classified by their product volume and variants capacities in Figure 2.5. These combinations are [29],[33],[100],[101],[102],[103]:

1. **Job production model with flexible manufacturing systems in a functional layout**: this combination has the greatest flexibility because the FMSs can produce different operations or products. This combinations can produce the widest variety of one-off products or low production volume.

2. **Batch production model with reconfigurable manufacturing systems in a cell layout**: this combination can produce shelves of products that require different operations with the same resources (i.e. reconfigurable machines). The number of product variants is smaller than the previous combination due to the use of a cell layout.

3. **Mass production model with dedicated manufacturing systems in a line layout**: this combination is the most efficient at manufacturing a unique type of product. However, there is not flexibility to produce more than one product. Although there might be reconfiguration to produce a similar product, the reconfiguration might take weeks or months.

4. **Job and batch production models with Self-Reconfigurable Manufacturing Systems (S-RMSs) with non-defined layouts**: this is a proposed new combination that aims at producing between the job and batch production models (higher than job production and equal to batch production, and higher product variety than batch production and equal to job production). Moreover, the S-RMSs can change among manufacturing layouts, create hybrid or novel and non-defined layouts. Though it can form line layouts, it cannot reach the production efficiency of dedicated systems.

Combinations one to three provide specific product volumes and limited ranges of product variants. In summary, cellular layouts can manufacture higher volumes than functional layouts in less time; functional layouts can manufacture higher number of product variants than cellular layouts. In a similar way, reconfigurable systems can manufacture higher number of product variants than flexible systems. The novel S-RMS can produce in a wider range of product volume and product variants than RMS, FMS and DMS. The S-RMS can rapidly increase the production volume from one-off products to low production volume and can cover a wider range of product variants (i.e. from low to high). Although the S-RMS are capable of forming line layouts for a single product (i.e. mass production model), these ones are not capable of reaching the production volume at the similar efficiency than dedicated systems.

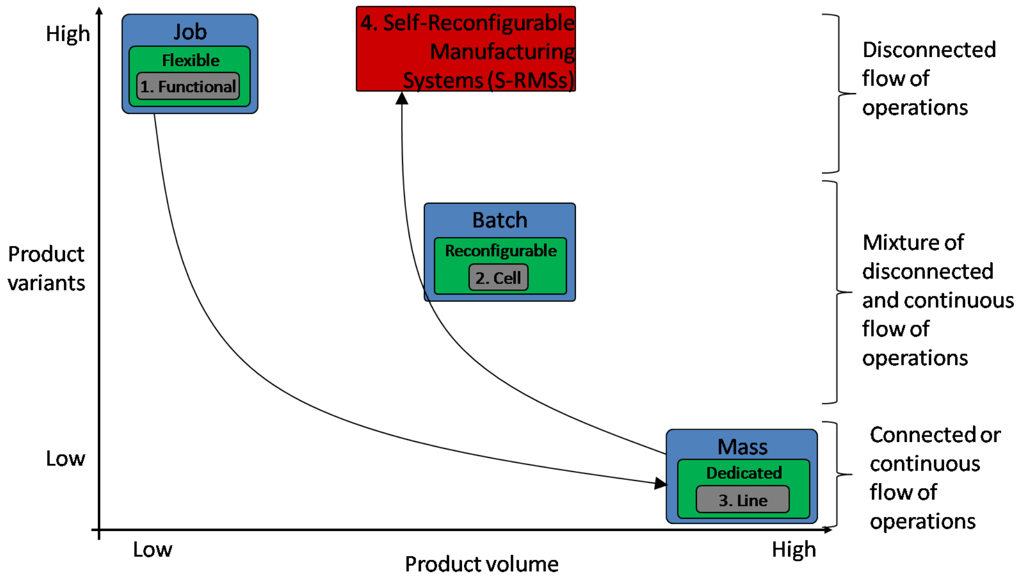


Figure 2.5: Classification of manufacturing models, manufacturing systems and manufacturing layouts according to the production volume and product variants they can generate. The horizontal axis shows the production volume whilst the vertical axis on the left shows the number of product variants that can be produced. The three main manufacturing models (i.e. job, batch and mass production) are highlighted within blue boxes. The three main classes of the manufacturing systems according to their flexibility are highlighted within green boxes. These systems are dedicated, flexible and reconfigurable manufacturing systems. The three main types of layouts (i.e. functional, cell and line) are highlighted in gray boxes. The vertical axis on the right classifies models, systems and layouts according to their process organisation (i.e. process flow). The combinations of models, systems and layouts are numbered from one to four, and a novel type of manufacturing system that is capable of producing the three types of layouts is highlighted in the red box numbered 4. This system is the Self-Reconfigurable Manufacturing System (S-RMS). The combinations are: 1. Job production model with Flexible Manufacturing Systems (FMSs) in a functional layout, 2. Batch production model with Reconfigurable Manufacturing Systems (RMSs) in a cell layout, 3. Mass production model with Dedicated Manufacturing Systems (DMSs) in a line layout, 4. Self-Reconfigurable Manufacturing Systems (S-RMSs) with no defined nor fixed layouts. Adapted from [29],[33],[100],[101],[102],[103].

2.3 Concluding remarks

This chapter presented a thorough review of manufacturing paradigms and the types of added value. This review demonstrate the evolution of manufacturing paradigms by the type of value they add. Also, the review demonstrate the conception of added value based on knowledge (i.e. HVM). Key challenges for HVM were described in Chapter 1. The main challenge for HVM is to survive the valley of death [2]. This requires the rapid automation and scalability of novel manufacturing processes.

Therefore, a review of current manufacturing approaches (i.e. manufacturing models, systems and layouts) was presented in this chapter. This review shows a research gap to propose manufacturing systems that can produce under an HVM environment (i.e. rapid change from one-off products to low production volume, for multiple novel products and the reuse of the same resources to reduce investment).

The review shows that current manufacturing systems and layouts cannot adjust to produce under the HVM environment. Therefore the novel manufacturing system, Self-Reconfigurable Manufacturing Systems (S-RMSs), were introduced. This one is capable of rapidly changing its production layout to produce different products with different production requirements (i.e. demand, deadline). A further step to S-RMS is its adaptation to offer global HVM services in order to enter new markets. This is presented in Chapter 3 with the framework INTelligent REconfiguration for a raPID production change (INTREPID). INTREPID is based on cloud manufacturing, Industry 4.0 principles, networked manufacturing, cloud robotics and the S-RMS. In contrast to cloud manufacturing, the use of advanced job allocation mechanisms is proposed in INTREPID.

In Chapter 3, it is also presented a full description of the S-RMS along with some examples. Later this thesis focuses on the production planning for the S-RMS. In Chapter 4, related problems to model the production planning for the S-RMS are reviewed.

Chapter 3

INTREPID: A comprehensive framework for high value manufacturing

Fierce competition in the manufacturing sector has motivated the change from the mass production paradigm to alternative paradigms such as lean manufacturing, services-oriented manufacturing or servitisation or mass customisation, and recently cloud manufacturing and high value manufacturing. There was a change of pattern, from competing by reducing products cost to competing by adding value to the products. Added value has proven to be critical in deciding which product to purchase (e.g. products from Apple Inc). Examples of added value are higher quality, lower cost, shorter delivery time, better service and more customised products or services [1].

High Value Manufacturing (HVM) focuses on adding value through the application of leading edge technical knowledge and expertise to the creation of products, production processes, or associated services [1]. The importance of adding high value lies in the difficulty for competing enterprises to reach the same novel product, process or service. High value might take years of research and development. However, automating and increasing the production volume from one-off products to low, medium and large scale production for HVM remains an unaddressed challenge [2].

Therefore, this chapter describes the framework INTelligent REconfiguration for a raPID production change (INTREPID), which was introduced in [29]. The big vision of INTREPID is to offer global HVM services built over the following key paradigms: cloud manufacturing, networked manufacturing, Self-Reconfigurable Manufacturing Systems (S-RMSs), industry 4.0 and cloud robotics. It proposes to do this through a globally distributed network of factories, where each factory has HVM machines and mobile manufacturing resources to reconfigure the production layout according to the current production requirements.

In this chapter, HVM challenges and key paradigms to address them are discussed in Section 3.1. A brief summary of INTREPID's parts (Subsection 3.2.1), INTREPID's operation and its main characteristics (Subsection 3.2.2) as well as challenges (Subsection 3.2.3) are described in Section 3.2. The parts of INTREPID are described in detail in Section 3.3. These ones are: a user interface and communications platform (Subsection 3.3.1), a job allocation system (Subsection 3.3.2), a network of reconfigurable facilities (Subsection 3.3.3), and a Self-Reconfigurable Manufacturing System (S-RMS) (Subsection 3.3.4). Concluding remarks are presented in Section 3.4. These remarks draw attention to the key processes that need to be considered at addressing the reconfigurable layout problem.

3.1 Key paradigms for INTREPID

The main HVM challenge (i.e. valley of death [2]) can be addressed with INTREPID. INTREPID is motivated by the following paradigms: networked manufacturing, cloud manufacturing, Self-Reconfigurable Manufacturing Systems (S-RMSs), Industry 4.0 and cloud robotics. These paradigms were briefly introduced in Section 2.1 and Subsection 2.2.2. In this section, the relationship of these paradigms with INTREPID is explained.

Networked manufacturing and cloud manufacturing

In order to enter new markets a distributed network is required. This involves challenges of its own. These challenges are communication and coordination within a globally distributed manufacturing network. The paradigm known as Networked Manufacturing (NM) addresses these challenges [75],[7]. In NM, information from facilities across the network is shared and accessed through the internet. This information is mainly used for collaborative product design and manufacturing. Hence, robust, reliable and real-time communications and mechanisms for collaboration are fundamental. Therefore, communications through internet must address issues such as security and scalability for real-time communications [104].

Web-based manufacturing systems are systems that facilitate collaboration within the networked manufacturing concept. These systems are implemented for different stages of the product lifecycle (i.e. product design, manufacture, service and disposal) [76]. Examples of their applications are collaborative design, collaborative planning and production control, manage product designs, parts design libraries, remote control of manufacturing resources, describe parts and features geometry, dynamic product simulation, model supply chains, manufacturing systems simulation, etc [105].

However, a drawback of these systems is their tailored design to a specific product or industry. Cloud manufacturing or Cloud Based Design Manufacturing (CBDM) is based on networked manufacturing and similarly to web-based manufacturing systems it focuses on distributed manufacturing facilities. In contrast to web-based systems, an unifying computing architecture and model to access manufacturing services through the internet have been proposed by CBDM [82].

CBDM main enabling technologies are derived from cloud computing [6]. These ones are Internet of Things, virtualisation, and service-oriented technologies (i.e. service oriented architecture, web service, and semantic web), advanced computing technologies (i.e. high-performance computing, ubiquitous computing and cloud computing itself). Cloud manufacturing aims to offer manufacturing services in an analogous way to cloud computing offer computing services.

Most of these technologies focus on information and system issues. However, research on the production planning in cloud manufacturing environments remain a challenging research gap. In contrast, studies on web-based Rapid Prototyping (RP) and manufacturing systems have considered production planning [76]. These are process selection, machines remote control and monitoring, and job planning and scheduling.

In RP, process selection refers to sending a job to the most adequate RP machine. If machines are not available there might be necessary to sequence and queue the job. Remote control and monitoring refers to real-time operation of RP machines. Hence, interfacing RP machines to the internet require the use of system independent technologies (e.g. common object request broker architecture) [106]. Nowadays, unloading materials from RP machines is a manual job. Hence, machine and operator schedules have to be considered in order to solve the job planning and scheduling problem.

Moreover, RP machines only perform a single process. This means that complete products might require a series of machining and assembly operations constrained by precedences. Hence, it is necessary to solve a complex planning and scheduling problem to produce the subcomponents before the assembly. Moreover, it is necessary to move parts from output of a process to the input of another one in a precise and timely manner to optimise the production plan. An optimised production plan facilitates producing more products or larger quantities simultaneously at lower cost and time.

Self-reconfigurable manufacturing systems

Different levels of physical changeability (i.e. ability to change) are required in order to provide HVM services within a distributed network of facilities. These levels cover changes in the infrastructure network and their supply chains, and changes in single machines and their setups. Changes in the production network results in the creation of temporal supply chains, and partnerships to design and manufacture products with a shared pool of resources [7],[82]. Different levels of changeability within a distributed network are classified according to level of product change and production level (required degree of production change to achieve the necessary product change) [94]. This classification is observed in Figure 3.1a, where changes in products are achieved through changes in the production level (resources). In Figure 3.1b, the degrees of changeability are achieved through changes in architectural or tangible objects (spaces). These are the levels:

- **Agility**: the product portfolio is changed. This includes changes in the manufacturing capacity and entrance to new markets through production network changes.

- **Transformability**: a product family is changed to a new one. This occurs through changes in production sites(factories), their production and logistics systems. This is done through changes in the location or the master plan.

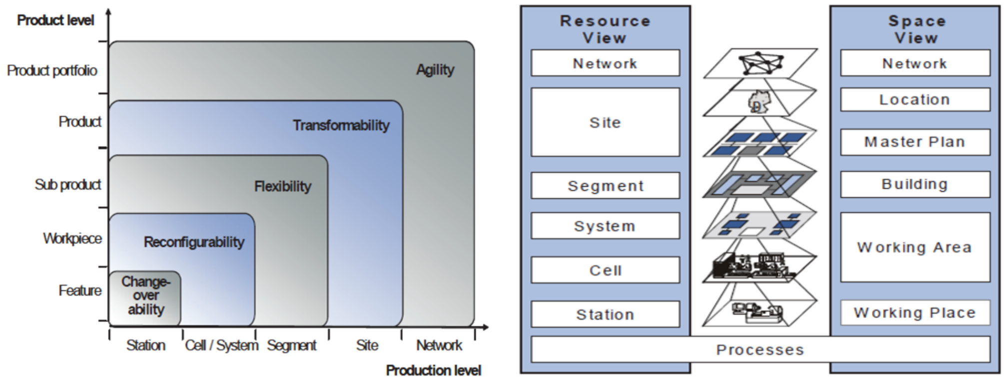
- **Flexibility**: a family of subcomponents or products is changed to a new but similar one. This is done in production segments changes such as production processes and material flow changes.

- **Reconfigurability**: a family of materials or subassemblies is changed in a production cell/system. This occurs through changes in the working area (i.e. addition or removal of functional elements).

- **Changeover ability**: feature of product is changed in a production station through changes in the working place. This is done by doing a different operation with the same machine or workstation.

There is no common consensus for these definitions [107]. Many authors agree with the definition given by Tolio in [94] where flexibility is the ability of a system to change its behaviour without changing its configuration whilst reconfigurability is the ability to change its behaviour of a system by changing its configuration. However, in the classification presented in Figure 3.1a, these terms are synonyms of change but they are distinguished by their application domains.

The levels of flexibility at a factory depend on its type of manufacturing systems and machine layout. These were explained in Chapter 2. Briefly, the main manufacturing systems are dedicated, flexible and reconfigurable, whilst the main layouts are lines, job shop, functional (workshop), and cells. Combinations of systems and layouts result in different levels of flexibility. The S-RMS was introduced in Chapter 2. In summary, S-RMS consist of logistics robots (Automated



(a) (b)

Figure 3.1: Different levels of flexibility and their corresponding architectural object. (a) Degrees of changeability in an enterprise, where the necessary changes in products are referred to changes in production. (b) Production changes are addressed with changes in architectural objects. (a) and b) images are reproduced from [94] with permission of Elsevier and Copyright Clearance Center under the license Number 3897920230512.

Guided Vehicles (AGVs)), mobile manipulators with interchangeable tools, flexible grippers and fixtures (i.e. fixtureless assembly, [99]), and movable machines for advanced processes that cannot be embedded in modular tools.

Industry 4.0 and cloud robotics

S-RMS belongs within the paradigm Industry 4.0. In this paradigm, Cyber Physical Systems (CPS) (i.e. mechatronic devices with wireless connectivity) communicate and coordinate with each other through the Internet of Things (IoT) to manufacture. This occurs in smart factories where conditions about the factory are shared through the internet.

Industry 4.0 and cloud robotics heavily rely on connectivity networks for collaboration and for sharing and getting data (Internet of Things (IoT)). In Industry 4.0, the IoT is used to coordinate tasks between resources whilst in cloud robotics is used for skills and data redistribution (i.e. collective robot learning) [108],[109]. Other advantages of cloud robotics include accessing big data (e.g. libraries of maps), human computation (e.g. human's skills for analysis) and cloud computing (i.e. parallel grid computing for path planning).

In summary, High Value Manufacturing (HVM) challenges are addressed by providing HVM services with a global distributed network of facilities. The network communications challenges are addressed with networked manufacturing and cloud manufacturing. The need for highly flexible manufacturing systems to manufacture multiple products is addressed with the S-RMSs. The communication and coordination challenges to determine an optimal layout according to production requirements are addressed with industry 4.0. Cloud robotics is proposed for self-optimisation of manufacturing processes and sharing or requesting manufacturing skills, procedures, knowledge, etc. INTREPID joins all of these paradigms.

3.2 INTREPID framework

In this section a description of the novel manufacturing framework INTREPID is provided, and the specific parts are described in the four following sections. INTREPID stands for INTelligent REconfiguration for raPID production change. INTREPID is inspired by the following paradigms: cloud manufacturing, networked manufacturing, Self-Reconfigurable Manufacturing Systems (S-RMSs), Industry 4.0 and cloud robotics. INTREPID consists of a globally distributed network of manufacturing facilities to provide cloud based HVM services. INTREPID addresses the issues of providing global HVM services and rapid scalability of the production volume.

3.2.1 INTREPID general description

INTREPID differentiates from other cloud manufacturing approaches on the use of mobile resources (movable machines and mobile manipulators, i.e. S-RMS) that can arrange themselves in production layouts tailored to the current production requirements. This facilitates providing services tailored to the user needs (i.e. contract by capability). This also results in reducing the material handling cost. The use of S-RMS results in a new research gap due to the possibility of determining novel layouts to manufacture several products simultaneously. This is because idle machine times and tool changeover times are reduced by manufacturing products that require similar processes. The main parts of INTREPID and its main processes are shown in Figure 3.2. A detailed explanation of the parts is given in Section 3.3. In brief, the main parts are:

- **User interface and communication platform**: This consists of a web-based manufacturing system for communications. There are two types of communications in this system. These are communication between customer and allocation systems and communications within the infrastructure network (i.e. their allocation systems).

- **Manufacturing infrastructure**: This consists of a globally distributed network of transformable manufacturing facilities. These facilities are called Reconfigurable Manufacturing Centres (RMCs). The clusters consist of physically connected factories. At each factory and each cluster the production layouts are tailored and can change to accommodate the production of several different products. The factories are equipped and designed to facilitate layout change.

- **Job allocation system**: There are three hierarchical levels of allocation systems. Firstly, the network allocation system assigns jobs to the clusters in the network. Secondly, the cluster allocation system operates in factories of a single cluster. Thirdly, at each factory, the factory allocation system assigns jobs to the most adequate layout of manufacturing resources.

- **Manufacturing resources**: These consist of autonomous mobile manipulators (arm robots over mobile platforms) and movable machines capable of physical self-reconfiguration, self-organisation and capable of changing production layout. These resources are modular and their name are Self-Reconfigurable Manufacturing Systems (S-RMSs). There were introduced in Chapter 1 and examples of them are presented in Subsection 3.3.4.

In Figure 3.2, INTREPID is shown at a global scale, where any institution (e.g. entrepreneurs, start-ups, SMEs and global companies) can request and access HVM services through internet from any location to any location in the world [110]. For this purpose, a globally distributed network of reconfigurable manufacturing clusters (i.e. factories joined together) are proposed as infrastructure. Each factory is reconfigurable (i.e. the production layout can change over time to produce different products). Factories can specialise according to the type of products they can produce and their manufacturing resources (i.e. from low precision to very high precision).

A key aspect of this network is bidirectional communications between customers and job allocation systems across the network, as well as communications between factories or clusters to coordinate production of complex products. Referring to the changeability levels listed in Section 3.1 (i.e. Figure 3.1), INTREPID includes the following:

- **Agility** through a global network of factories that supports producing at new markets and changing the product portfolio by changing the production layouts.

- **Transformability, flexibility and reconfigurability** of the complete production layout through the use of the S-RMS.

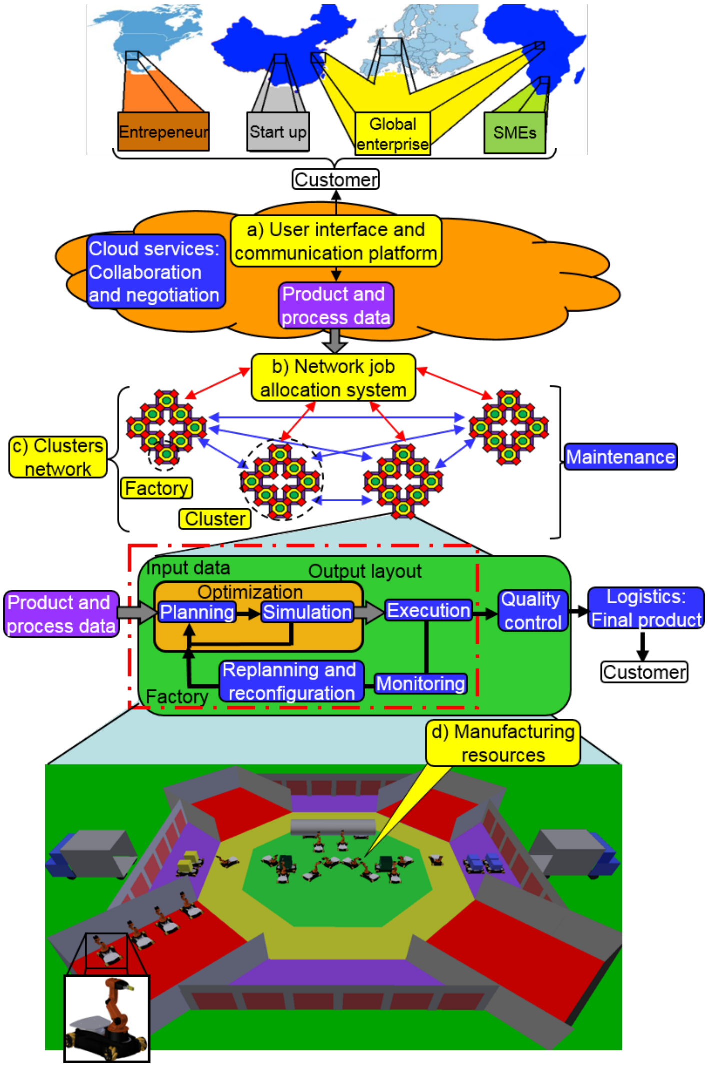


Figure 3.2: INTREPID framework. INTREPID' parts are shown in yellow squares. These are: a) User interface and communications platform; b) Central job allocation system; c) Infrastructure network (network of RMCs); and d) Manufacturing resources (Self-Reconfigurable Manufacturing Systems (S-RMSs)). Blue arrows represent communications among RMCs, whilst red arrows represent communications between the network job allocation system and RMCs. Each RMC is composed of several connected reconfigurable factories. An example of a reconfigurable factory is highlighted and some production layouts with mobile manipulators are observed (i.e. line and cell layouts). The main processes of INTREPID are shown in blue squares. Processes to determine the most adequate layout are planning and simulation in an optimization loop. Product and process data from jobs is the input for this optimisation loop, whilst the output are production plans that are executed and monitored in real world. Real-time problems are managed through replanning and reconfiguration.

- **Changeover ability** is achieved through the variable working range of the S-RMS. Also, the use of the arm robot allows manufacturing on different axes and orientations.

The full scope of the framework is to offer global HVM services, but the production system (i.e. S-RMS) can be implemented in any factory to boost its flexibility and responsiveness. The main advantages of INTREPID are:

- High value manufacturing services
- Additional production capacity
- Entrance to new countries and markets
- Production of multiple novel products simultaneously
- Test of new layouts before investment
- Product design and simulation services

3.2.2 INTREPID operation

INTREPID operates through its four main parts, see Figure 3.2. The main processes are classified in web-based communication processes (e.g. collaboration and negotiation for product design, raw material supply securing and manufacturing processes specification), logistics processes (e.g. raw material supply and final product delivery), and manufacturing process (e.g. planning, simulation and execution). In rough terms, INTREPID's operation sequence consist of four main type of processes. Firstly, processes aimed to collect or develop product and process data and requirements. Secondly, logistics and supply chain management processes to secure raw materials and coordinate materials arrivals and product departures in a Just In Time (JIT) way. Thirdly, processes of job planning and simulating at each factory or group of factories. Fourthly, jobs are executed and monitored at the factory or factories. This sequence is explained in detail in the next paragraphs:

1. **Data collection**: This includes production requirements (e.g. deadlines, demands), raw materials data (e.g. type of material and its properties, initial dimensions), final product data (e.g. bill of materials, drawings, machining features) and data from compulsory or desirable manufacturing processes (e.g. tolerances, quality), see Figure 3.3. Also, there might be negotiation and collaboration processes [80]. For product development, collaboration is important to agree on the product and process data. Negotiation is done if

the manufacturing capacity is not enough to handle production requirements. Through negotiation, the production requirements can be changed based on resources availability [82]. Hence, negotiation and collaboration create a feedback loop with the next step (i.e. job allocation). Data collection, product development collaboration and product requirements negotiation are performed through the user interface and communications platform. These processes are firstly done between users and the network allocation system. Later, when jobs are allocated, the communication is between users and specific factory or factories.

2. **Job allocation**: The second process refers to matching product requirements to production capacity. The product and process data is sent to all the available manufacturing factories. Factories propose bids describing production plans and schedules. The bid are pre-selected and coordinated by cluster and factory allocation systems. This coordination occurs through the user interface and communications platform. These bids are per factory or per group of factories for complex products. Hence, the factories compete for jobs, but cooperate for complex products [111]. The network allocation system selects the most adequate factory or factories based on requirements fulfilment, cost, and logistics.

3. **Planning and simulation**: at each available factory, planning and simulation are done in an optimisation cycle until the best feasible production layout is determined (optimisation), see Figure 3.3. Planning and simulation involve manufacturing resources that are available or are proximate to be available. Planning refers to developing mathematical models that consider available resources to determine if they can meet user's requirements. These models also consider logistics to supply raw materials and deliver products. Simulation refers to test the resulting production layouts from the planning process. This might involve 3D simulation in a virtual factory [40]. The objective is to address robotics challenges such as path planning and collision avoidance for both the mobile platform and the arm robot. Planning is a hierarchical process. This means its scope is increased if resources cannot meet user's requirements. The planning scope ranges from a single factory, many factories, an entire cluster or many clusters. Regarding planning, in order to determine an optimal production layout the following areas are considered: task to robot allocation, process order, available space for layout formation, variety of production processes and layouts, and multi mobile robot path planning, coordination and scheduling (Figure 3.3).

4. **Execution, monitoring and replanning**: The manufacturing resources are capable of locating themselves according to the planned layout, calibrate

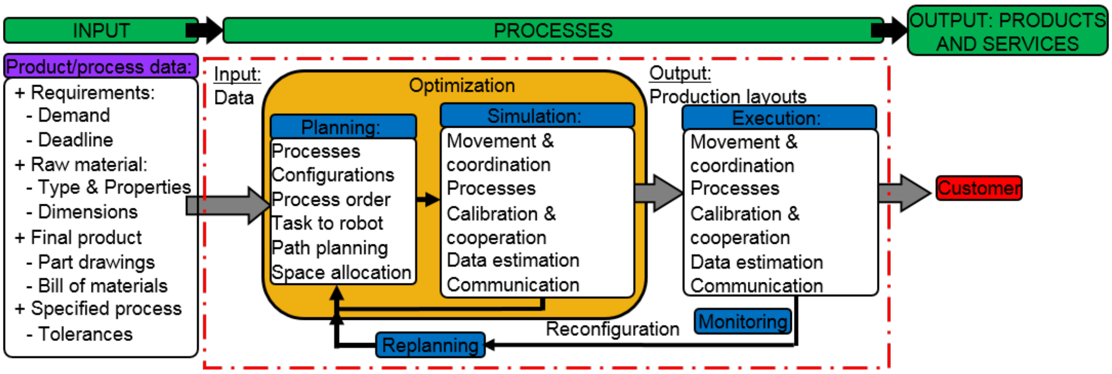


Figure 3.3: Types of data required by INTREPID and production planning processes within INTREPID. The processes are: design (product and process data), planning, simulation, execution, (self-)monitoring and replanning.

and autonomously execute the production plan. Real-time (self-)monitoring of the production schedule and resources performance allows layout replanning. Replanning might range from simple adaptation up to reconfiguring the complete layout depending on the severity of the problem (e.g. broken tools, collisions, increments in demand, additional production orders). Predesigned factories facilitate resources self-organisation and self-reconfiguration with the use of sensing and communications systems [40]. Algorithms and systems related to the calibration, movement, commissioning, coordination and cooperation among robots during process execution are supported by sensing and communication systems. The following areas are simulated and later executed: Movement and coordination of resources, manufacturing processes, system and machines calibration and cooperation, sensors data estimation, and communication signals (Figure 3.3).

Regarding the maintenance process, it is proposed to integrate self-repairing abilities in the manufacturing resources for simple damages [112]. Otherwise, it is proposed to schedule predictive and preventive maintenance, and corrective maintenance if required. The area of quality control is proposed to be done in two steps. These consist of intermediate inspections at important points during the manufacturing process with mobile manipulators, and a quality control inspection before delivery to the customer. Other processes that are important to INTREPID but that will not be discussed are supply chain management and logistics and communication networks.

3.2.3 INTREPID challenges

The main challenges of INTREPID focus on the research, development, implementation and testing of its constituent parts:

- Development and implementation of a user interface and communications platform that provide real-time and robust communications
- Design and construction of factories and RMCs that facilitate the movement of mobile manufacturing resources, including the specification of sensing and communcations systems
- Design and construction of mobile manufacturing resources (S-RMS) that can autonomously reconfigure the production layout
- Research, development and implementation of job allocation mechanisms that consider the design and operation of the infrastructure (RMCs and factories) and S-RMSs.

Specific challenges to facilitate the reconfiguration of the manufacturing layout are:

- Mechanisms for self-organisation and self-assembly for a large number of mobile robots
- Mechanisms for self-calibration of the S-RMS
- Reconfigurable control systems for robots
- Space allocation mechanisms for reconfiguration of resources
- Research and development of automated connection mechanisms of supplying services such as energy, vacuum and cooling liquid
- Design and construction of automated and intelligent warehouses for the manufacturing resources, raw materials and products
- Development of robust and real time communications among manufacturing resources and between central allocation systems and resources
- Negotiation and collaboration mechanisms between user and job allocation systems, and among the different RMCs and factories within a RMC

3.3 INTREPID parts

INTREPID consists of a user interface and communications platform, a network of Reconfigurable Manufacturing Centres (RMCs), a job allocation system and Self-Reconfigurable Manufacturing Systems (S-RMSs). The aim of INTREPID is to offer global HVM services through its parts. Therefore, these parts are described in the following four subsections.

3.3.1 User interface and communication platform

In order to provide global HVM services, the use of global communication networks is of paramount importance. Internet is the most widespread and affordable communication technology in the world. Therefore, the use of internet has been proposed as the technology to access HVM services and interchange manufacturing data. Key research on the use of internet and web-based technologies to manufacture in distant locations will be reviewed in this subsection.

Related literature

Offering manufacturing services through the use of information superhighways was first proposed in telemanufacturing [80]. The infrastructure was dedicated centres with up-to-date technology and expertise. Many researchers have proposed to send design data through internet to manufacture in a different location in respect to the design location (design anywhere, manufacture anywhere) [110]. Examples of equipment to manufacture through internet include, CNC machines [113] (turning [114], milling [115] and drilling [116]), rapid prototyping [117] and manufacturing with robots [118],[119].

The globally dispersed infrastructure network is a very important part is to guarantee secure, reliable and real-time communications [120],[121]. Communications between customers and the central system allows collaboration and negotiation for product and process design and simulation. Furthermore, communications across the manufacturing network allows information sharing for global optimal job allocations and production coordination of complex jobs [80].

A deep survey on technologies to offer manufacturing services is presented in [117],[76]. Some of the most common architectures involve intermediary agents (brokers) that buy and sell manufacturing services between customer and providers. These agents are also capable of negotiating and collaborating with customer during the product design phase. Agents also verify and convert the customer information into valid information for manufacturing providers [82]. These agents or brokers correspond to the job allocation systems proposed in INTREPID.

There are collaboration and negotiation between customers and the job allocation system to develop products or processes. This is done through the user interface and communications platform. This loop consists of design, simulation and redesign of products and processes based on the available type of manufacturing resources. Through negotiation, the production requirements can be adjusted based on resources availability [75].

The data that defines a job was described in Subsection 3.2.1. This data must be either collected from users or developed through collaboration. This data is collected or developed through the user interface and communications platform. This data consist of production requirements (e.g. deadlines, demands, destinations to distribute), raw materials data (e.g. type of material and its properties, initial dimensions), final product data (e.g. bill of materials, drawings, machining features) and compulsory or desirable manufacturing processes (e.g. tolerances, quality), see Figure 3.3.

The user interface and communications platform is a web-based manufacturing system. Web-based manufacturing systems are supporting systems to perform diverse stages of the product lifecycle (i.e. product design, manufacture, service and disposal) through the internet [76]. An important disadvantage of these systems is the numerous instances of them focused on diverse products and industry and product lifecycle stages. Also, there are multiple technologies to develop these systems.

A review on technologies to develop web-based manufacturing systems was done in [76]. The most relevant technologies to address interoperability issues are common object request broker architecture, distributed component object model, remote method invocation, service-oriented architecture protocol and web services [76]. Web-based manufacturing systems focused on simulation were reviewed in [105]. Here, technologies to enable web-based simulation have evolved from Java application, web 2.0 and semantic web (web 3.0)

The main issues that need to be addressed in the development of web-based manufacturing systems are database management (e.g. data storage and retrieval, data classification, data transfer), real-time and secure communications, results visualisation, real-time monitoring of processes progress (e.g. simulation or manufacturing), documentation and model repository. The most efficient technologies to offer web-based manufacturing services have been collected into a single concept called cloud manufacturing or Cloud Based Design Manufacturing (CBDM) [82]. CBDM is based on networked manufacturing and similarly to web-based manufacturing systems it focuses on distributed manufacturing facilities. In contrast to web-based systems, CBDM offers a standard framework for users to access manufacturing services through the internet.

Proposed approach

It is proposed to base INTREPID on the cloud manufacturing paradigm. CBDM refers to the use of shared resources to design and manufacture products where design and production data is agreed through the cloud (i.e. the internet). However, INTREPID is different from CBDM in the following main changes:

1. Job allocation is performed through specialised systems to handle multiple products with multiple operations.
2. Manufacturing infrastructure is designed to facilitate the rapid reconfiguration of the production layout.
3. Manufacturing resources consist of the S-RMS.

CBDM is derived from the cloud computing concept [6]. The main elements of cloud manufacturing are Internet of Things, virtualisation, service-oriented technologies (i.e. service oriented architecture, web service, and semantic web) and advanced computing technologies (i.e. high-performance computing, ubiquitous computing and cloud computing). In contrast to web-based systems, CBDM is characterised by their agility and scalability and the following key characteristics: high performance computing, ubiquitous access, self-service, big data, pay-per-use, resource pooling, virtualisation, multi-tenancy, search engine and crowdsourcing [122]. Besides the difference in characteristics, CBDM outperform web-based manufacturing systems in the following key areas [122]:

- **Computing architecture**: multi-tenancy (i.e. single instance of the application software to serve multiple tenants), and virtualisation of manufacturing resources and their capabilities.
- **Sourcing process**: crowdsourcing and search engine (enables users to find desired service suppliers)
- **Information and communication infrastructure**: Interconnectivity tools (e.g. Internet of things, smart sensors, wireless devices) allow real-time monitoring of design and manufacture data.
- **Programming model**: parallel programming models enable processing large datasets (big data).
- **Design communication**: Use of social media (enables multiple communication channels to reduce distorted information), and real-time communication (enables cross-disciplinary teams to design and redesign products or processes to improve productivity)

These elements provide cloud manufacturing superior characteristics over previous web-based technologies to provide services through the internet [81]. The main characteristics are self-service, rapid scalability, on-demand provision and ubiquitous accessibility.

3.3.2 Infrastructure: Flexible production networks and transformable factories

Highly flexible production networks and logistics are required in order to provide global HVM services. These ones should be able to supply raw materials and commission manufacturing resources for a product mix through rapid changes of the supply chain and production plan respectively. The focus of this thesis is not on the supply chains, but on production plans for a single factory.

The change in the production plan at a single factory can be done in two ways. Firstly, the relayout of the production layout (this is called as layout reconfiguration in the rest of this document) or by moving the materials through different manufacturing machines (this problem is commonly known as routing). For the reconfiguration of the layout, it is require the rapid deployment of services such as electricity, pneumatics, and cooling and lubrication liquids. For the routing challenge, the key problem is to locate the manufacturing resources, so that the travelling distance of the materials is minimised.

Related literature

The problem of locating manufacturing resources to minimise the distance along the production flow constitutes the Facilities Layout Problem (FLP) for manufacturing resources or as the machine layout problem. This combinatorial problem is NP-hard. The problem has variants such as location of factories along the supply chain, location of departments inside a factory, location of machines inside a department, and even the location of machine and robots inside a manufacturing cell [103].

Models and solutions for the FLP are presented in [123]. The most relevant models are mixed integer programming [124], quadratic assignment problem [125] and graph theory model [126]. Whilst, relevant solution methods are Computerised Relative Allocation of Facilities Technique (i.e. heuristic approach) [127], branch and bound (i.e. exact approach), and multi-objective approaches [128]. Metaheuristics are used large problems, relevant metaheuristic are ant colony [129], genetic algorithm [130], tabu search [131] and simulated annealing [132].

The use of transformable factories was proposed in [133]. Transformable factories were contrasted against traditional factories. Key findings are that traditional factories require a slow and complex decision making process because their lifespan is meant to be thirty or more years. Desired characteristics of transformable factories are [134]:

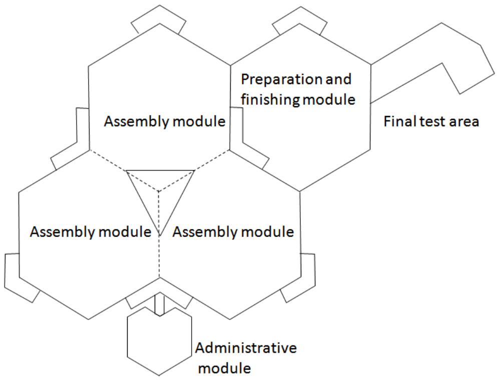
1. Located near the relevant markets to avoid cost of logistics;
2. On-demand work;
3. Plug and play model to create feasible layout to match the production requirements;
4. Highly-automated and integrated;
5. Self-organisation, self-control and self-optimisation of the production systems and networks.

A green field project from BMW proposed a transformable factory in order to achieve modularity [135]. In order to achieve lean operations, the factory was designed in a collaboration between architectural planners and production process engineers. The factory is characterised by a modular structure to achieve scalability (i.e. lightweight steel with as fewer columns as possible that allows extension or reduction of the modular building). Also, plug and produce assembly stations are used to arrange the layout of this factory according to the information and material flow.

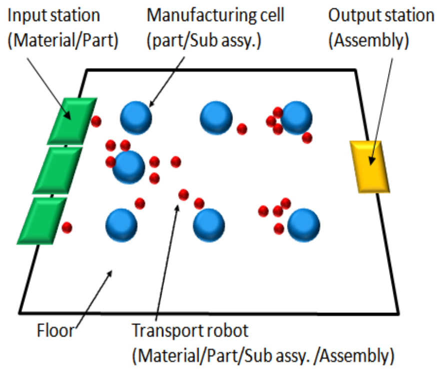
A modular factory was proposed for the Volvo's factory in Kalmar, Sweden [136]. The layout of the factory are four hexagonal areas, see Figure 3.4a. Assembly operations are executed in three of the hexagonal areas, whilst the fourth hexagonal area is used for preparation and finishing operations. In the middle of the three hexagons for assembly, a storage area is located. The purpose of this middle area is to achieve equidistant distances between the hexagonal areas for assembly and the storage area.

The use of manufacturing cells that are manually moved through an empty factory in order to minimise the transportation costs of materials was proposed in [13], see Figure 3.4b. This approach focuses first on machining operations, whilst assembly operations are executed later. In order to determine locations of the manufacturing cells, a particle swarm optimisation algorithm is employed. This proposed factory is limited to a single direction of material flow due to the use of fixed input and output stations (i.e. warehouses). In contrast to the flexible flow of materials and products given by movable warehouses considered in this thesis.

In order to offer global HVM services, there is a need for globally distributed factories that can rapidly change production layouts to manufacture different products. In order to produce complex products that a single factory or RMC is incapable of producing, the coordination of factories in a global network is of



(a)



(b)

Figure 3.4: Examples of factories designed to increase the production flexibility. (a) Volvo's factory in Kalmar, Sweden. A modular layout was implemented in this factory (Redraw from [136]). (b) An empty factory with input stations on the left and output stations are on the right was proposed by Yamada in [13]. This image shows machining cells and transport robots arranged in a layout according to the product that is manufactured. Redraw from [13].

paramount importance. This coordination is also vital in order to schedule the transportation of raw materials, subcomponents and products among factories. The Ericsson Radio Systems factory concept proposed the delegation of tasks from macro-factories to micro-factories [137]. This Ericsson concepts is similar to plants-within-a-plant [138]. In both concepts the added value activities are performed in the micro-factories and functions such as maintenance and quality are performed within macro-factories.

Proposed approach

The specification of algorithms for factories location along the network is not part of this thesis. The focus is on the optimal planning of the location of the manufacturing resources along the production floor and the formation of manufacturing cells. It is proposed to use a globally dispersed network as manufacturing infrastructure. The network consists of Reconfigurable Manufacturing Centres (RMCs), which consist of several connected factories. These factories are reconfigurable because they are designed to facilitate the reconfiguration and relayout of the manufacturing resources. This is done with the use of communications and sensing systems. The RMCs can be divided by their functionality and scope of application in very high precision, high precision, medium precision and low precision. A

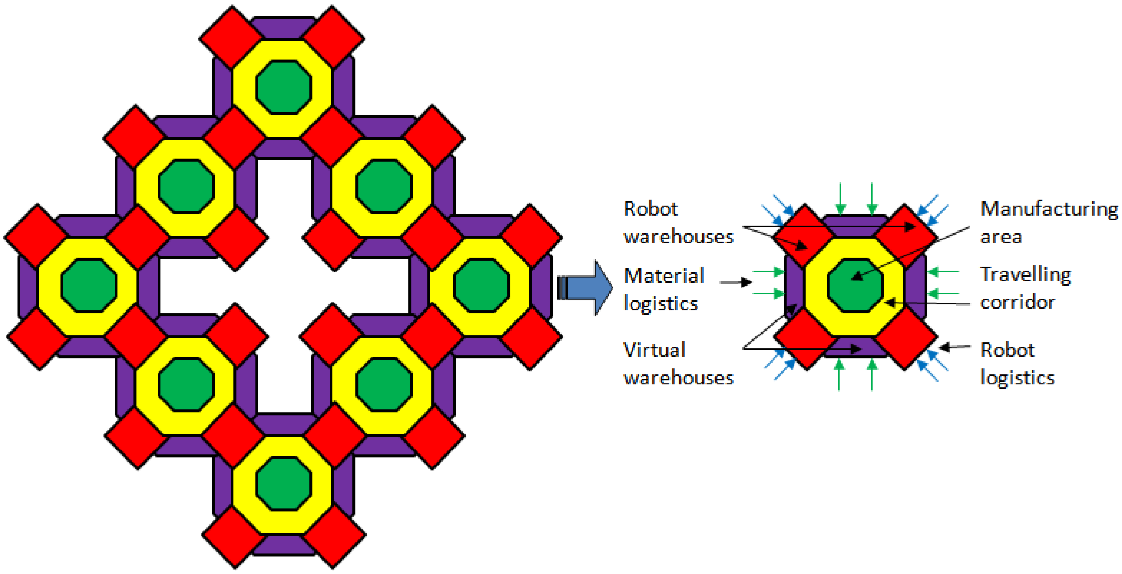


Figure 3.5: Representation of a Reconfigurable Manufacturing Centre (RMC), where the RMC consist of connected reconfigurable factories arranged in a diamond pattern (left) and areas of a single reconfigurable factory (right). The areas of a single factory are: 1. warehouses (i.e. areas to store robots and machines), 2. manufacturing area (i.e. dedicated area (i.e. unmanned) for formation of manufacturing layouts), 3. virtual warehouses (i.e. warehouses to store raw materials and final products), 4. travelling corridor (i.e. dedicated corridor for the movement of robots), 5. robot logistics (i.e. entrances and exits for robots and machines), and 6. material logistics (i.e. entrances for raw materials or exits for final products).

RMC and the areas of a single factory are shown in Figure 3.3.2.

Each factory within each RMC is designed to facilitate the reconfiguration and relayout of the manufacturing resources (i.e. robots and machines). This is because, the factories do not have fixed equipment and structures. Factories mainly consist of an empty area with communication and sensing networks that help robots and machines to self-organise and cooperate. Factories are connected through shared robot warehouses. This allows robots to travel between factories and therefore increase the number of resources that can be considered for the job allocation system (i.e. intra factory resources). A factory consists of the following areas (see Figure):

1. **Warehouses**: are areas to store robots and machines

2. **Manufacturing area**: is a dedicated area (i.e. unmanned) for formation of manufacturing layouts

3. **Virtual warehouses**: refer to warehouses to store raw materials and final products

4. **Travelling corridor**: is a dedicated corridor for the movement of robots

5. **Robot logistics**: refer to entrances and exits for robots and machines

6. **Material logistics**: refer to entrances for raw materials or exits for final products

In order to facilitate the relayout and increase the possible layouts that can be formed, an octagon shape is proposed for each reconfigurable factory. The octagon shape facilitates the connection of multiple factories by its corners and creates warehousing zones to store machines and robots. This shape and the warehousing zones are similar to the hexagonal shape presented in 3.4. Also, the octagon shape does not increase the complexity of determining production layouts from a four-sides factory.

The multiple entrances and exits (i.e. logistics for raw materials and products) of the factory are inspired by distribution centres and raw materials warehouses. The proposed reconfigurable factory has multiple doors to facilitate access of raw materials and products in order to increase the flexibility of manufacturing layouts. The purpose of having multiple doors (i.e. for entrance or exit) is to achieve Just In Time (JIT) characteristics and avoid the use of traditional and fixed warehousing areas. Therefore, the proposed warehouses are virtual (i.e. always in constant change of their content and positions).

The predesigned factories and clusters include communications and sensing systems to facilitate the reconfiguration of layouts. Sensing allows monitoring of resources. This is additional to the self-monitoring of the production performance done by the resources. Communications tools allow resources to be interconnected for movement coordination and cooperation at manufacturing tasks. This is based on the design principles of Industry 4.0, interoperability, real-time sensing and virtual models of the factory.

3.3.3 Job allocation system: Network, cluster and factory levels

A job allocation system is needed to determine the best suitable allocation of resources to accomplish incoming manufacturing jobs within a distributed network of factories. Three hierarchical subsystems can be distinguished for the job allocation systems. These are:

- **Network allocation system**: assigns jobs to the clusters in the network

- **Cluster allocation system**: assigns jobs to factories in a single cluster
- **Factory allocation system**: assigns jobs to the most adequate layout of manufacturing resources

The network allocation system coordinates the production of complex products or products that require distributed factories. The cluster and factory allocation systems have similar solution methods but the amount of available resources is larger in the cluster one. This is because, cluster allocation systems can employ of all the resources within many factories, whilst the factory allocation systems can only employ resources at its respective factory. The cluster allocation system considers the movement of resources across the cluster whilst the factory allocation system only within a factory.

The cluster and factory allocation systems also coordinate production of complex products with other clusters and factories respectively. At the cluster and factory levels, the allocation refers to assigning manufacturing resources to perform manufacturing tasks. At a network level, the allocation refers to assigning jobs to be performed at a factory or factories. In the case of multiple factories, these might be an entire cluster or multiple clusters. The hierarchical allocation is explained in the following paragraph.

The network system transmits production requirements data to all cluster systems. The cluster systems do the same to the factory systems. At each factory, there is planning and simulation to determine the most adequate production layouts. If the resources are not enough at single factories, these ones coordinate with other factories. This coordination is managed by the cluster system. Similarly, if the resources are not enough at a single cluster, clusters coordinate with other clusters. At the network level, the coordination is managed by the network system. The focus of this thesis is on the job allocation system for a single factory. Therefore, relevant literature on the allocation of resources within a single factory is presented in the following paragraphs.

Related literature

Allocation of resources is commonly known as assignment problem and it is applied to agents, robots, workers, and even factories. The most relevant method to solve the assignment problem is the Hungarian method [139]. However, for a decentralised network it is more complicated [11]. Approaches that have addressed the assignment problem in a decentralised network with the Hungarian method require local information of each agent. As a result, the information is collected and redistributed in a central system.

The assignment problem focused on mobile robots is known as Multi Robot Task Allocation (MRTA) in the field of robotics [140]. MRTA considers metrics to measure the requirements of tasks to execute and the cost of executing tasks depending on the robot and the task itself. The MRTA problem consists of determining the most appropiate number of robots to execute each task. The MRTA problem can be solved with MRTA architectures, which are mechanisms to assign robots to tasks.

There are decentralised and centralised architectures for the MRTA problem. In decentralised architectures, costs of performing tasks with a robot or many robots are calculated by each robot. In contrast, these costs are calculated by the central system in centralised architectures. Hence, in decentralised architectures, the time to determine a feasible or optimal allocation is distributed among all the robots. However, in this type of architectures, the results are sub-optimal due to the lack of global information. The most relevant MRTA architectures can be classified by their working principles as follows [141]:

1. **Market-based architectures**: work by broadcasting the requirements and corresponding costs of tasks. Next, robots get this information and plan a bid for the tasks considering cost to perform them. The lowest bid is selected. Whenever there are equal bids, metrics such as the robot scheduled maintenance are used to select a bid. Most of market-based architectures assume that global profit of the allocaation is optimised if local profits are optimised. The best known market-based architectures are MURDOCH and TraderBots [142],[143].

2. **Motivation-based**: The best known motivation-based architecture is ALLIANCE [144]. ALLIANCE is characterised by the use of two behavioural mechanisms to allocate tasks. These mechanisms are:

   - *Impatience*, which produces a robot to undertake a task if no progress is achieved, and
   - *Acquiescence*, which produces a robot to quit over a task if there is no progress

3. **Team consensus**: works with a robot broadcasting help until there are enough robots for the task. This architecture was implemented in order to achieve dynamic role assignment for RoboCup teams [145]. The drawback of this architecture is that the task allocation has to be agreed by the entire team.

In summary, some advantages and disadvantages of these three architectures are the following: Market-based approaches do not require high levels of communications among the robots, but between robots and a central system with information about tasks. The parameters of the behavioural mechanisms (i.e. impatience and acquiescence) have to be tuned in the motivation-based architectures. Lastly, in the team consensus architectures, high levels of communication are required among the robots in order to agreed a task allocation.

A market-based architecture works by decomposing complex tasks in elemental and core tasks to form task trees. Then, robots cooperate and solve elemental subtasks until the complex task is achieved [146]. The multiple levels in the task trees required mechanisms to manage additional tasks. The decomposition of complex tasks results in multiple innovative task allocations.

Most of the decentralised MRTA architectures make use of the Contract Net Protocol, a distributed communication protocol, to broadcast tasks [147]. The decentralised architectures considers negotiation between nodes with tasks and nodes with robots. An advantage of market-based architectures is joint bids for complex tasks [111]. Blackboard systems, where robots can access and interchange data are recommended for ill-defined problems where solutions results from the sum of partial solutions [148],[149].

Proposed approach

In industrial applications, it is desirable to obtain optimal, decidable and reproducible solutions that are based on models rather than random or suboptimal solutions. Hence, for INTREPID, an hybrid approach is proposed. This hybrid approach consist of a centralised allocation for each factory whilst a decentralised allocation for each cluster and the clusters network. The centralised approach consist of the use of exact methods such as mathematical programming. This takes advantage of the resources and the production plan monitoring to keep updating virtual models of factory and its resources. This allows solving real-time problems or plan future reconfigurations with updated information. This approach allow resources to focus on manufacturing tasks, instead of wasting computing and communication resources in coordinating, and determining production layouts with local information.

However, for the network and cluster allocation systems, this approach is infeasible due to the large number of resources. Therefore, a decentralised approach is suggested. The decentralised approach consist of the use market-based methods. The network and cluster systems are based on collecting bids of each factory and selecting the most profitable one. For complex jobs, joint bids of multiple factories or clusters are allowed. Cluster and network systems select bids based on raw material and final product distribution logistics.

At each factory, systems are proposed to monitor and allocate resources to manufacturing tasks in real-time to solve problems. Factory systems have information about the working conditions of the resources and equipment. This allows monitoring the progress of the manufacturing tasks, and it allows adding more efficient resources, or layout reconfiguration. The aim is to achieve the centralised production planning and optimisation with a distributed execution and performance feedback. It is proposed to do this through communication between resources and a factory communication network.

Customers are able to provide information, depending on the requested service, directly to the factory job allocation systems. The information provided is related to final product requirements and specification, as well as desired or specific manufacturing processes and raw materials. The functions of the factory job allocation systems are:

- To transmit necessary global data to every resource,
- To simulate the calculated layouts, and
- To collect real-time data about the working conditions and status of resources in order to schedule their preventive maintenance

3.3.4 Highly flexible manufacturing resources

INTREPID is a novel framework due to the use of the S-RMS. This type of system was introduced in Chapter 2. In brief the systems consist of logistics robots (AGVs), mobile manipulators with interchangeable tools, flexible grippers and fixtures (i.e. fixtureless assembly, [99]), and movable machines for advanced processes (i.e. processes that cannot be embedded in modular tools). The objective of the S-RMS is to form production layouts depending on the production requirements. These robots will perform tasks such as manufacturing (machining and assembly), logistics, quality testing and handling (i.e. machine loading and unloading).

Related literature

Self-adaptability is an important concept to manage uncertainty and disturbances during manufacturing. Therefore, mechatronic modules to provide self-adaptability to traditional manufacturing machines is proposed in the project called Mutable Systems [150]. The drawback of this approach is the production processes are limited by their own design. Also, changes of the process plans require transport of raw materials and subcomponents between machines. Another important concept to improve the current manufacturing systems is cognition [151]. The purpose of providing cognition is to obtain an autonomous reconfiguration of the manufacturing processes.

Manually movable arm robots that can change tasks by changing tool, or the sequence of the manufacturing process by changing positions was described in

[14]. This approach focuses joining and handling tasks in assembly lines. The drawback of this approach is the manual movement of the robots, and the limited locations where robots can manufacture. Manually moved manufacturing cells that can change the production layout depending on the product are proposed in [13]. Although the cells contain auxiliary production equipment, the cells are limited to single processes. Similarly to the manufacturing cells, plug and produce modules require to be moved in order to form different production layouts [152]. The dimensions of the materials that can be manufactured in the modules is limited by its volumetric capacity. In contrast to the use of manufacturing cells and plug and play modules, the use of mobile robots that can perform elemental tasks increases the flexibility and the ability to form novel layouts. An example of this approach is called Biological Manufacturing Systems [153].

In Biological Manufacturing Systems, artificial potential fields were proposed in order to create manufacturing layouts according to the current products. The potential fields work by attracting or repelling manufacturing robots to material transport robot that require specific operations. As a consequence, flows of materials and manufacturing resources emerge based on local interactions between the resources. Examples of the novel layouts are concentric circles, where the most frequently used machines are located in the center of the circles, whilst rarely used machines are located on the outer circles [12].

The limitations of manufacturing cells, movable arm robots and plug and produce modules are surpassed by mobile manipulators because they can manufacture on materials of large or small dimensions due to their arm robots and mobile base. Arm robots are characterised by their redundant degrees of freedom and reachability. The advantages of mobile manipulators over plug and play modules or movable manufacturing cells are their mobility, their reprogrammability, their capability to manufacture material of different dimensions, their reachability achieved with their serial kinematics configuration and the large number of available interchangeable tools. Mobile robots have been used in industrial applications such as transportation of materials and products with Automated Guided Vehicles (AGVs) [154]. The drawback of AGVs is their limitation to move along guidance lines. The best known example of mobile robots in logistics applications is the Kiva system that consist of mobile robots that carry racks of products through warehouses [155].

A review of different industrial applications for mobile manipulators was done in [156]. The results are that mobile manipulators can be used for logistics tasks such as transportation and single- or multi-part feeding into conveying lines, as well as assistive tasks such as machine loading or unloading and pre-assembly. Moreover, mobile manipulators were applied to feed different types of parts to specific conveying lines [157]. Another research on mobile manipulators focus on developing a task allocation methodology based on the Hungarian method [11].

The drawback of this approach is the use of fixed locations to manufacture.

The use of arm robots and mobile manipulators have been proposed for applications such as machining and composites placing, respectively [158],[159]. The use of mobile manipulators to place composites over large components is beneficial due to the serial kinematics of its arm plus the mobility of its base [158]. This is because the arm can reach distant parts of the large component. Similarly, the use of arm robots for machining allows machining parts that require different cutting axis without changing the setup of the material. Key challenges to address machining with arm robots were identified in [159]. These challenges are the rebound force during machining, low pushing and pulling force, and inferior accuracy compared to machining machines. These challenges are addressed with novel mechanisms such as laser tracking system and rebound force control compensator. Moreover, the use arm robots to assembly subcomponents without setups and fixtures was proposed under the term fixtureless assembly [99]. This is known as cooperative robotics through the rest of this thesis.

Proposed approach

Mobile manipulators with mecanum wheels are proposed for INTREPID, see Figure 3.6a for an example. These wheels guarantee forward, backward, diagonal and sideways movements, as well as rotations in x-y plane (i.e. omnidirectional movement). The most common example of mobile robots for transporting materials is the Automated Guided Vehicle (AGV). An example of a robot system for logistics is the Kiva system [155]. An existing conveying robot from FESTO is shown in Figure 3.6b.

Common manufacturing layouts were reviewed in Subsection 2.2.3. In brief, the layouts are: 1. functional (the equipment is divided by the tasks they can perform); 2. lines (the equipment are aligned in a line), 3. cell (hybrid of functional and lines, most repeated processes are in line, but machine not in the line allow for alternative and complementary tasks similar to the performed tasks), and combinations of the last three.

Mobile manipulators are proposed as logistics robots (AGVs) that can carry material in a continuous way (mobile robots moving along the manufacturing line or conveying robots forming a conveying line) or discrete way (mobile robots materials or shelves of materials). Examples of S-RMS layouts with AGVs and mobile manipulators are shown in Figure 3.7. The proposed logistics robots are:

1. **Conveying robots**: are robots that connect with plug and play capabilities to create a conveying line, see Figure 3.7a.

2. **Mobile gripper robots**: move to the manufacturing robots or machines carrying small size materials, see Figure 3.7b.

(a) (b)

Figure 3.6: Examples of mobile robots with application to industry and manufacturing. (a) Omnidirectional Mobile Manipulator: YouBot from Kuka Robotics. Image reproduced, with permission, from [160]). (b) FESTO conveying station (Photo courtesy of Martin Wojtczyk). Image reproduced, with permission, from [161]). Robots in the figures are not scaled.

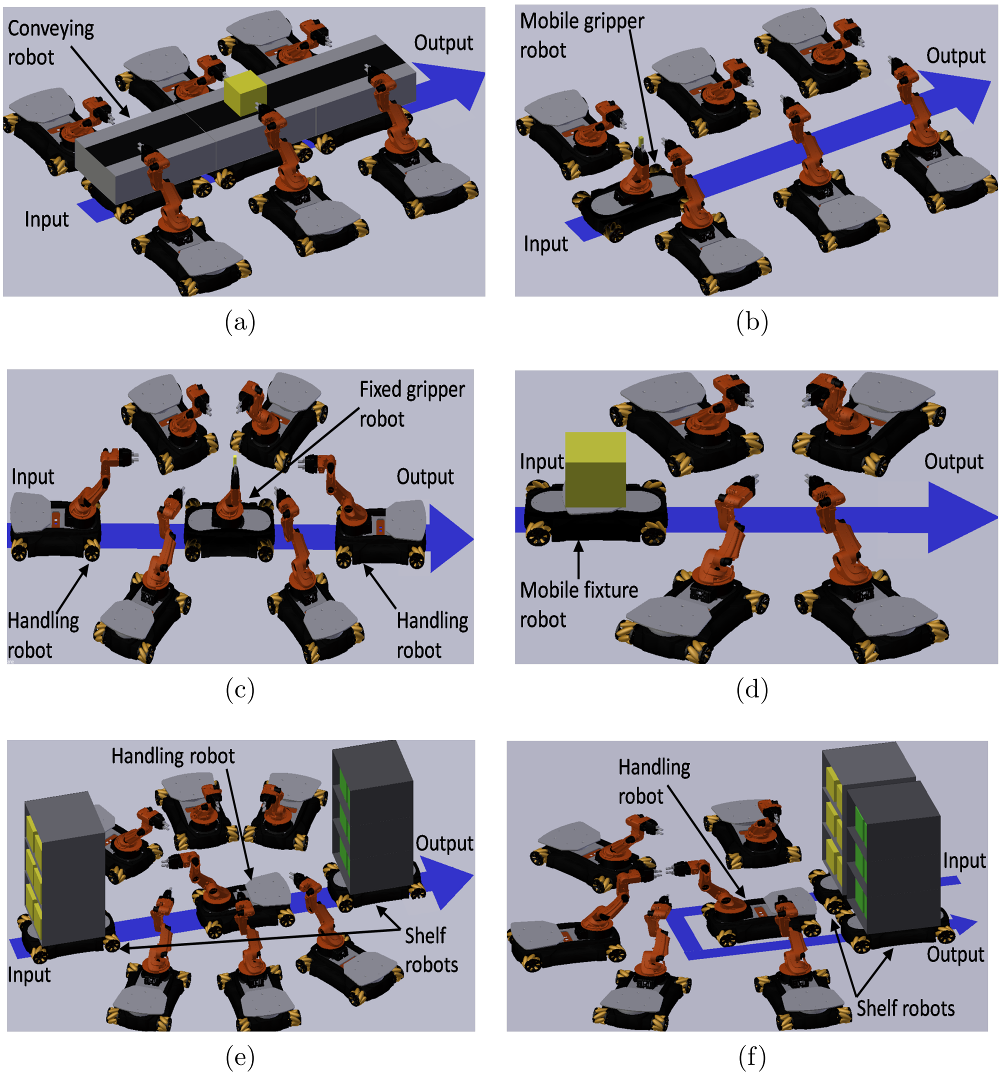
3. **Fixed gripper robots**: are fixed, whilst other robots move and work around them, see Figure 3.7c. These type of robots are able to manage small or medium size materials.

4. **Mobile fixture robots**: move to the manufacturing robots or machines carrying medium or large size materials, see Figure 3.7d.

5. **Shelf-robots**: refer to robots that can carry an entire rack of parts or components. Hence, shelf-robots can transport large quantities of materials and products, see Figures 3.7e and 3.7f.

The robots capabilities can be modified or extended with two methods. The first is to adapt manufacturing processes (e.g. soldering, milling and composites placing) in interchangeable process dedicated tools. The second method is the physical change of the hardware of the robot. Consequently, it is needed to simplify and embed the manufacturing processes related to machining and quality testing either in form of a modular tool or along the arm robot.

More advantages of using robots for manufacturing were reviewed in Subsection 3.1.3 with the description of the Industry 4.0 and cloud robotics paradigms. These are decentralisation of decisions, interoperability and connectivity of resources, real-time monitoring of robots performance, updating virtual models of the factory



(a) (b) (c) (d) (e) (f)

Figure 3.7: Examples of robotic layouts. The material flow is highlighted with blue arrows. Names of robots are specified, otherwise, they are MMRs. Line layouts: (a) With conveying robots. (b) With a mobile gripper robot. Cell layouts: (c) With a fixed gripper robot. (d) With a mobile fixture robot. Cell layouts with shelf-robots: (e) With the same material flow. (f) With different material flow. (a) A line layout formed by conveying robots and manufacturing mobile manipulators by each side. (b) A line layout formed by manufacturing mobile manipulators at each side and a mobile gripper (c) A cell layout with two handling robots, one fixed gripper robot and several MMRs. (d) A cell layout with one mobile fixture robot and MMRs. (e) and (f) Cell layouts with manufacturing mobile manipulators, handling mobile manipulators, one input-shelf and one output-shelf. In (e) the input and output material flow have the same direction, whilst (f) the input and output material flow have opposite directions.

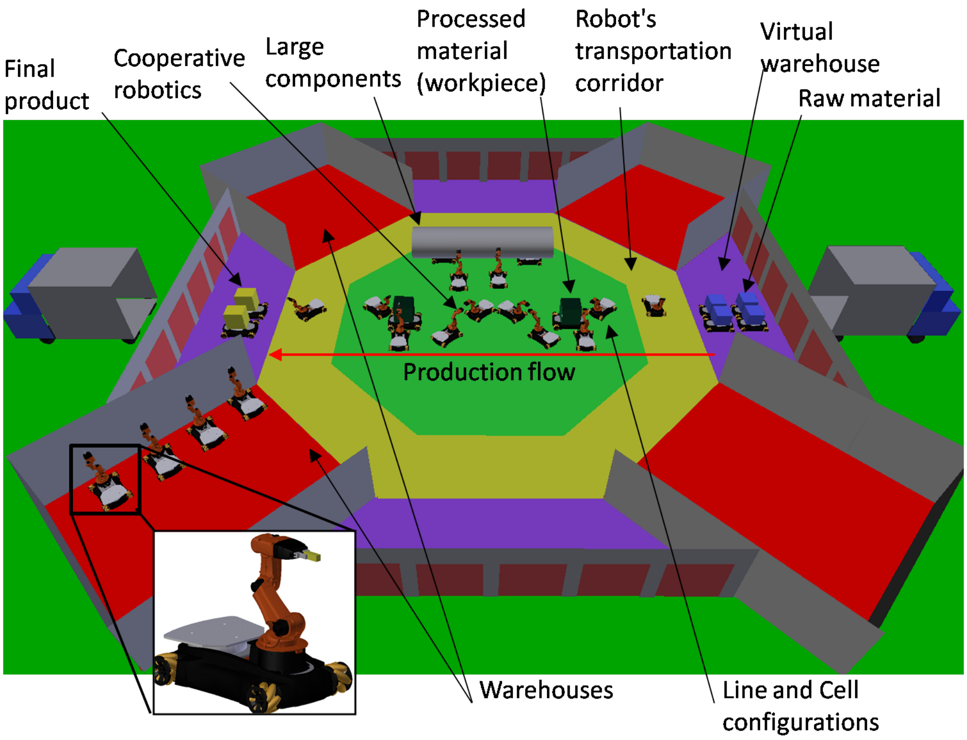


Figure 3.8: Example of a reconfigurable factory with examples of production layouts (i.e. examples of line and cell layouts for small size materials, cooperative tasks between robots and a fixed layout for a large component).

for reconfiguration, and skills sharing and redistribution through the cloud for developing self-optimisation systems. A 3D image of a proposed design for a reconfigurable factory along with examples of expected layouts can be observed in Figure 3.8.

3.4 Concluding remarks

In this chapter, the INTREPID framework, its constituent parts and its operation were described. The enhanced levels of flexibility obtained with INTREPID and its manufacturing system, the S-RMS, were analysed and compared to state of the art approaches. One of the novelties presented in this thesis is the use of S-RMS to create production layouts tailored to the production requirements. Moreover, the S-RMS can change functionality by changing tools and its programming. Due to these types of flexibility, determining production plans for INTREPID is challenging.

Among the challenges for INTREPID, it is of special concern, determining production plans that involve available manufacturing resources to produce the current production mix subject to each product requirements. Production plans for the whole INTREPID network is a challenge that involves millions of elements (e.g. manufacturing resources, and global logistics and supply chains). Even determining production plans for a single cluster involves considering the use of MMRs or machines from other factories, or outsourcing jobs between factories.

Hence, the rest of this thesis focuses on proposing and formulating a novel production planning problem that considers mobile manufacturing resources (S-RMS) within a single factory. This means determining the most adequate positions, routes to reach positions, and schedules (allocation and sequencing) of manufacturing operations to machines. The resulting production plans must be constrained to predefined sequences of operations (process plans).

Consequently, determining production plans implies solving scheduling, positions assigning, and routing problems. Hence a novel problem, the SAR problem, to address these three problems in an inclusive problem is proposed in the next two chapters. The scheduling, positions assigning and routing problems and their constraints are analysed in Chapter 4, whilst, a comprehensive notation for the SAR problem for a single factory is proposed in Chapter 5.

Chapter 4

The SAR problem: analysis of constituent problems

The INTelligent REconfiguration for a raPID production change (INTREPID) framework and its enabling manufacturing system, the Self-Reconfigurable Manufacturing System (S-RMS), were described in the previous chapter. In brief, the S-RMS increases the flexibility of current manufacturing systems with the use of Mobile Manufacturing Robots (MMRs) and movable machines that can form layouts according to the current production requirements. The use of MMRs makes possible to continuously move manufacturing resources towards the most adequate locations to perform manufacturing tasks.

The use of MMRs enhances production plans with features from the robotics field. These features include the decentralisation of decisions, the autonomy to perform tasks, and access to cloud-based computing. These features are highlighted as part of the paradigm Industry 4.0, whose aim is to endow more autonomy to manufacturing resources through wireless communications.

Concluding remarks from last chapter highlight the challenges to determine production plans with the use of S-RMS. Production plans with the use of S-RMS consist of determining the most adequate schedules of tasks to machines, positions of these machines, and routes for the machines to reach these positions. These production plans are constrained to manufacturing sequences (i.e. process plans) specific to each product.

In order to determine production plans that consider S-RMS, it is proposed a novel problem that solves the scheduling, machine layout, and vehicle routing problems simultaneously. Therefore, the name of the novel proposed problem is the Scheduling, positions Assigning and Routing problem (SAR) problem. An introduction to the SAR problem is provided in Section 4.1. The problems that constitute the SAR problem are analysed in the following sections:

- The scheduling problem and its analogy in the field of robotics, the Multi Robot Task Allocation (MRTA) problem, are reviewed in Sections 4.2 and Subsection 4.2.3, respectively
- The positions assigning problem is reviewed in Section 4.3, whilst its general versions, the machine layout and the Facilities Layout Problem (FLP) problems are reviewed in Subsection 4.3.1
- The vehible routing problem and its analogy in the field of robotics, the motion planning problem, are reviewed in Section 4.4 and Subsection 4.4.1

Elements, characteristics, assumptions and optimisation objectives that can be include the formulation of these three problems are analysed in this chapter. These are studied in order to understand their relevance to the SAR problem and their integration in a single problem. Also, existing notations to these problems are studied. A summary of the chapter and concluding remarks are presented in Section 4.5.

4.1 Defining the SAR problem

For common manufacturing systems with fixed machines (i.e. dedicated, flexible and reconfigurable) it is common to first design the machine layout and determine paths between machines. This is done considering an estimation of products to produce and their expected demands. Once, the layout and paths among machines are known, the normal operation consists of determining production plans for every arriving group of products to be manufactured. These plans consist of allocation and scheduling of tasks to machines, as well as selecting process plans (i.e. routing) of tasks on the machines. Finally, selection or determination of paths to move materials and products between machines and warehouses.

If the operation under the current layout is infeasible or unprofitable, a redesign and relayout is required. This problem is studied in academia and is known as the dynamic and the reconfigurable layout problems. These problems refer to determining a layout per each planning period. The dynamic layout problem assumes known product requirements data of all planning periods whilst the reconfigurable layout problem only assumes known data of current and next planning period. These problems are reviewed in detail in Subsection 4.3.2.

In case of the SAR problem, the relayout might occur at each task of the current planning period. Consequently, the complexity of determining a feasible or optimal production plan is higher than for the dynamic and reconfigurable layout problems. Moreover, because product requirements of the current period are known, it is possible to design the layout considering the most adequate operation.

As a result, layout design and layout operation problems are addressed together in the SAR problem, see Figure 4.1.

The SAR problem aims to integrate and solve the problems of scheduling, machine layout and vehicle routing in order to determine feasible and optimal production plans tailored to the current production requirements. It is expected that these production plans result in novel machines layouts that take advantage of manufacturing the product mix simultaneously and with mobile manufacturing resources. Hence, this might result in reduced manufacturing costs and times. Elements, characteristics and assumptions from each constituent problem (i.e. scheduling, machine layout and vehicle routing) are reviewed and analysed in the following three sections.

Elements and characteristics are referred as pieces of data in [84]. Elements and characteristics are pieces of data that can be included in a problem or not. The combinations of elements and characteristics define not only different types of problems within a field, but also, different fields. Characteristics can influence (i.e. constraint or relax) elements, and assumptions can influence elements and characteristics. Elements, characteristics and assumptions of each problem influence different aspects of the SAR problem. The ones from the scheduling problem influence the allocation of tasks to resources over time, the ones from machine layout problem influence the design of the machine layouts, and the ones from the vehicle routing influence the paths or routes of materials and products transportation.

4.2 Scheduling

In general terms, the machine scheduling problem refers to the allocation of manufacturing operations in a machine(s) over time [84]. When, there is a single machine, the scheduling problem is reduced to a sequencing problem that can be solved by permutations of the operations. In general, the purpose of the scheduling problems is to minimise the deviation of production time with respect to products due dates [84]. The scheduling problems are classified according to three fields notation proposed by Graham et al. in [85]. The fields are α, β, γ, and they describe the type of problem (i.e. the machine environment), the characteristics of the job or task, and objective to optimise, respectively. These fields are separated by a vertical bar symbol (|). An example of the notation is $1|r_j, prmp, prec|L_{\max}$, where the terms are explained as follows:

- 1: single machine problem (environment).
- r_j: homogeneous or unique release date/time for all tasks or jobs.
- *prmp*: preemptions allowed.

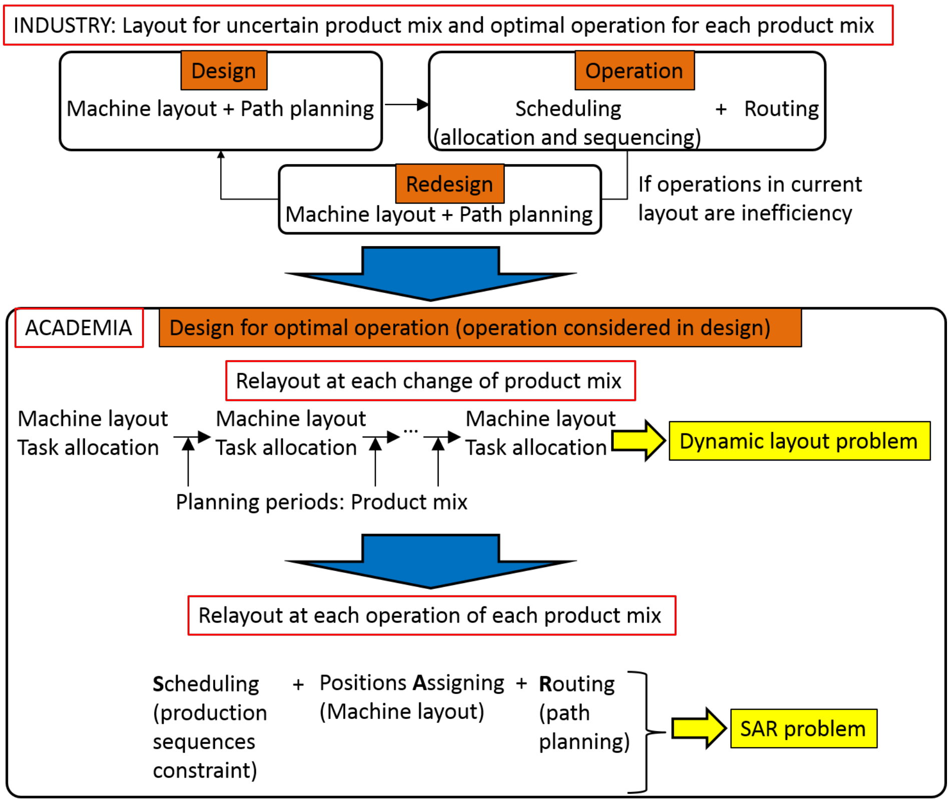


Figure 4.1: Evolution of the production planning problems. Current approaches in industry for addressing the production planning problem with constant production demands are shown in the upper part of this figure. Also, recent approaches from academia to deal with production planning problems in dynamic and reconfigurable layout problems is shown in the lower part of this figure. These problems focus on determining layouts per each period whilst reducing the cost of changing layouts between periods. The proposed problem for dynamic production planning is highlighted as the newest approach from academia [193]. The problem is called the Scheduling, positions Assigning and Routing problem (SAR) problem. The SAR problem refers to addressing the scheduling, positions assigning and vehicle routing simultaneously.

- *prec*: sequences (i.e. precedences) between tasks are considered. Common types for these sequences are a chain or a tree
- $L_{\max}$: optimisation objective of minimising maximum lateness. Lateness is the deviation (negative or positive) of the completion time of operation j against its due date d_j. This is $L_j = d_j - C_j$.

The combination of the elements, characteristics and their assumptions result in a wide variety of scheduling problems [85]. Elements, characteristics and their assumptions from the scheduling problem influence the allocation of tasks to resources over time in the SAR problem. Relevant types of problems (i.e. α field) are described in Subsection 4.2.1, core elements and characteristics (i.e. β field) are described in Subsection 4.2.2, and common optimisation objectives (i.e. γ field) are reviewed in Subsection 4.2.5.

4.2.1 Types of scheduling problems

The type of environment where operations take place define the type of problem that is addressed. The established problems are reviewed and analysed in relation to the SAR problem in the following paragraphs. Relevant scheduling problems to the SAR problem are identified.

The **single machine** problem refers to determining a sequence of tasks to a single machine in order to reduce the difference between due dates with production dates of products. The products in this type of problem require a single type of task. In contrast to the single machine problem, in the **parallel identical machines** problem there are multiple machines working in parallel. However, all machines have the same processing speed. In a similar way to the last problem, the focus is on products that require a single type of task.

Similarly to the last problem, the **parallel uniform machines** problem refers to allocating several jobs to several machines, where each job requires the same type of task. However, in contrast to the last problem, the machines have different processing speeds. These speeds are factors of the job and the machine. The **parallel unrelated machines** problem is identical to the parallel uniform machines problem, but with arbitrary processing speeds for each machine.

In contrast to the previous four problems, in the **job shop** problem, products require more than a single task. This problem refers to complex products that require different sequences of manufacturing tasks (i.e. precedence graphs). The processing speeds for each task depend on each machine. These characteristics are similar to manufacturing within High Value Manufacturing (HVM).

The **flexible job shop** is an enhanced variety of the job shop problem [162]. This problem considers the use of multitask machines. Hence, the flexibility is increased but also the complexity to determine a feasible or optimal schedule. The flexible job shop is addressed in two parts, the tasks to machines allocation. The first is routing (i.e. machine sequence selection) and the second is scheduling (i.e. sequencing). The addition of multi tasking machines makes this problem ideal to partially formulate the SAR problem.

The **flow shop** problem is a special variety of the job shop, where all products have the same sequence of tasks, but the processing times are different for each product. This problem is similar to the single machine problem in the fact that only sequencing is required. Hence, this problem can be solved through permutations.

The **flexible flow shop** problem is also known as hybrid flow shop [163],[164]. This problem is a combination of the flow shop and the parallel machine problem. Similarly to the flow shop, the flexible flow shop processes jobs in the same order (i.e. same manufacturing sequence), but there are more than one machine working in parallel at each stage. The jobs do not have to be processed in all the machines, but they have to follow the same processing order.

The **open shop** problem deals with complex products that require multiple tasks at multiple types of machines. The difference resides in the lack of compulsory sequence of tasks for products. This problem should not be confused with the open shop terminology that refers to whether products are produced to stock or produced to order [165].

The **cyclic scheduling** problem deals with repetitive schedules with the objective to produce demands larger than the unit [166]. The cyclic scheduling problem might apply to the previous scheduling problems that require more than one task (i.e. flow shop, open shop and job shop).

These problems are limited either to a single type of machine (i.e. single machine, parallel identical/uniform/unrelated machines), or limited to a unique or none sequence of tasks (i.e. flow shop, flexible flow shop and open shop respectively). The most relevant problems to produce under the HVM paradigm are the job shop and flexible job shop problem. This is due to its capability of producing several different products simultaneously, where each product requires a different sequence of tasks and machines are multitask. Therefore, the rest of this section focuses on the job shop and flexible job shop problems. The use of different and repeated movable machines in INTREPID, make the job shop and flexible job shop problems the most significant to propose an original problem formulation for the production planning with S-RMS.

4.2.2 Scheduling elements and characteristics

This subsection focuses on the second field from Graham's notation (i.e. job or tasks characteristics). These tasks' characteristics refer to possible assumptions for problem formulation. Elements, characteristics and their assumptions from the scheduling problem influence the allocation of tasks to resources over time in the SAR problem. Scheduling characteristics are divided in deterministic and stochastic. Significant assumptions to the SAR problem that are derived from the job shop and flexible job shop problems are reviewed in the next paragraphs.

Deterministic elements and characteristics

Deterministic elements and characteristics refer to known values (i.e. do not have any uncertainty and the values are known at any instant of time)[84],[85],[167]. Approaches to solve the scheduling problem with deterministic elements and characteristics are grouped under the term predictive scheduling. This refers to offline planning prior to the execution of the manufacturing schedule. The SAR problem formulation might consider relevant elements and characteristics from the scheduling problem. The most important element to describe a problem are called core elements. A core element to describe the scheduling problem is the type of environment where manufacturing operations occur. This environment refers to the types of problem (i.e. α field). These types of problems were described in 4.2.1. However, regarding the second field (i.e. β), core elements and characteristics of the scheduling problem with their respective notation are:

- **Release dates/times** *(r)*: refers to the dates or times when raw materials with which to perform a task are available. This is commonly used as the earliest starting time to perform an operation [168]. In case of manufacturing multiple products, the release times can be homogeneous or heterogeneous for all products. For a long production time the release times can be periodic or non-periodic. This means whether the new products are produced periodically or not.

- **Due dates/times** *(d)*: refers to the dates or times when the execution of jobs (i.e. manufacturing of products) is due. If these dates or times are enforced they are called deadlines. Similarly to the last characteristic, the due dates or times can be homogeneous or heterogeneous for all products and periodic or non-periodic.

- **Weights** *(w)*: denote priorities that are assigned to each task or products. In the SAR problem, weights might be homogeneous or heterogeneous. This is helpful to coordinate the movement of MMRs. At the solution level, weights facilitate assigning resources to the highest priority jobs and assign the rest of jobs considering the previously assigned resources.

- **Machine-dependent processing time** *(p)*: refers to jobs processing times being dependent on the machine that is performing them. Regarding the

S-RMS, the processing time is related to the accuracy or performance of the machine or MMRs or the interchangeable tool. This element might be considered or neglected. Regarding the SAR problem, this characteristic might be different for each task in each machine (heterogeneous) or might be the same (homogeneous) for all tasks in all machines. Consequently, the SAR problem might be simplified to a multi robot task allocation problem (for heterogeneous processing times) or the multi robot path planning problems (for homogeneous processing times.

- **Sequence dependent setup times** *(s)*: this characteristic refers to the necessary time to remove tool and material (i.e. products under manufacturing process) after a manufacturing task or to the time to setup a new tool and material in the same machine. These times can be different from each other or not. Also, any or both of these times can be zero. For the SAR problem, it is considered to have homogeneous or heterogeneous setup and removal times for all machines and products. Also, whether the setup and removal times are equal or unequal for each machine. It is considered that these times depend on the type of product and machine.

The following characteristics might or not be included in the scheduling problem (i.e. considered or neglected), and this results in different variants of the scheduling problem.

- **Preemptions** *(prmp)*: refers to whether it is possible to stop and resume a manufacturing task. The total processing time is accomplished through many manufacturing processes. For the SAR problem this characteristic can be considered or neglected.

- **Precedence constraints** *(prec)*: impose restrictions on the sequences that tasks must follow. A succeeding task cannot precede its preceding task. A group of precedence constraints is known as a process plans. A graphical representation of these constraints is known as precedence graph (i.e. acyclic and directed graph). This can be summarised in an incidence or adjacency matrix, where 1's indicate if a task precedes another task; 0's indicate no precedence. A list can also be used to directly indicate the tasks that precede another task. This graph can have the form of a chain (chains), or trees (tree), namely intree (intree) or outtree (outtree). A chain occurs when tasks have a single predecessor and successor tasks, whilst an intree when tasks have at most one successor task and outtree when tasks have at most one predecessor task. In the SAR problem, this characteristic can be considered or neglected in the case of single task products.

- **Time-dependent processing time or scheduling with deteriorating jobs**: these are a special kind of processing time where their value depends on the time when the task is starting [169],[170]. In machine scheduling, one example is the production of a batch of products has delay penalties. Another example is the maintenance of a machine, where the longer it takes to start the maintenance of a machine, the longer the maintenance last [171]. Regarding the SAR problem, this characteristic might be considered or neglected, and if considered, it might be homogeneous or heterogeneous values of time for all the operations.

- **Resource-constrained** *(res)*: might limit the available periods of time for manufacturing. Machines might require auxiliary resources such as tooling, fixtures, jigs, cooling liquid and cleaning chemicals. The classification of resources is according to resources constraints in [172]:

  - **Renewable**: resource's total usage is constrained
  - **Non-renewable**: resource's total consumption is constrained. This means, that once resources are assigned to a task, they cannot be used in another task
  - **Doubly constrained**: resource's total usage and total consumption are constrained

  Moreover, resources can be classified by their divisibility in:

  - **Discrete**: where resources can only be assigned as positive integer values
  - **Continuous**: where resources can be assigned as real positive values

  Regarding the SAR problem, (auxiliary) resources constrained might be considered or neglected. This is indicated with the notation (i.e. λ, σ, ρ) introduced in [167]. The notation λ, σ, ρ represent number of resources, total amount of each resource, and amount of required resources respectively. Auxiliary resources might be homogeneous for all machines or heterogeneous and dependable on tasks and machines.

- **Restrictions on machine eligibility** *(M)*: denotes a subgroup of machines that can process a task. This means that although machines are identical and working in parallel not all machines can perform the task. In case of the SAR problem this characteristic is represented by the tools that can perform the task, but they might have different accuracy, that results in different degrees of quality. This might restrict the machines that can be used.

- **Buffer or storage capacity**: refers to the capacity of a machine to store a limited or unlimited amount of products after performing an operation. In deterministic scheduling, this characteristic is known as (machine) blocking because machines with limited capacity can become bottlenecks. In case of the SAR problem, this characteristic might be considered (limited) or neglected (unlimited). If is considered, it might be heterogeneous or homogeneous to all machines. Also, this characteristic might be product dependable or not. If buffers are considered, it is necessary to allocate area for movable buffers (shelf robots were explained in Chapter 3).

- **No wait** *(nwt)*: this characteristic refers to products that require continuous tasks. Hence, no buffer is allowed. In the SAR problem, this is addressed with machines in line layout (flow shop).

- **Transportation constraints**: are specific to the flexible job shop, where the transport of work-in-process (i.e. raw materials that are in process of becoming products) must be considered in the scheduling problem [173],[174]. These constraints are usually neglected in the flow shop scheduling problem due to the nearness of the machines in the layout and because these constraints are difficult to formulate. Regarding the SAR problem, this type of constraints are compulsory (i.e. core characteristics) in order to address the path routing problem.

Stochastic characteristics (dynamic scheduling)

Stochastic characteristics influence significantly the formulation of the problem and its solution methods. Solution methods change from predictive scheduling approaches to approaches such as reactive scheduling, robust-proactive scheduling or predictive-reactive scheduling [175],[176]. Although this increases the complexity of formulating a flexible job shop problem, there are examples that integrate these stochastic events in the flexible job shop problem (e.g. machine breakdown and unknown processing, release and due times, machine breakdown and alternative process plans [177],[178]). The stochastic characteristics can be grouped in job-related and resource-related [84],[175]. The following characteristics are self-explanatory, but a short description is provided. In the SAR problem these characteristics (events) might be considered or not. If they are considered, their occurrence might be periodic or non-periodic. These are:

Job-related characteristics:

- **Random processing time** *(X)*, **random release date/time** *(R)* and **random or changing due date/time** *(D)*: are the stochastic versions of their respective deterministic versions. For the SAR problem these characteristics can be considered or neglected. If they are considered, the characteristics have the same options to their deterministic counterparts, but with random (i.e. stochastic) values.

- **Changes in demand**: products might update their requirements during or after the schedule is determined. In the SAR problem, these updates might be periodic or non-periodic.

- **Dynamic priorities (weights)** w: tasks or products might get a higher priority, or rush hour products get higher priority over the current products. Similarly to its deterministic counterpart, priorities might be homogeneous or heterogeneous.

- **Alternative process plans (processing sequences)**: Process plans are precedence constraints among tasks to manufacture a product [177]. A product might have alternative sequences of tasks that result in the same product with the use of different machines. This results in different production times and costs. Therefore, alternative process plans can be considered or neglected. In scheduling, alternative process plans is known as dynamic routing because routing is understood as determining sequences of machines to produce a product. In this case, routing should not be confused with the vehicle routing problem, which refers to selecting the best route or path from a limited set. In the SAR problem, process plan selection is a core part that increases the flexibility of the S-RMS.

Resource-related characteristics:

All the following characteristics can be considered or neglected.

- **Machine or tool breakdowns** *(brkdwn)*: is considered as deterministic in [84] and as stochastic in [179]. These characteristics might happen in the middle of a task. This might result in delays and require a new machine or tool. For the S-RMS, this might require to remove a defective MMR or machine, and to keep redundant resources.

- **Resources (e.g. tools, raw materials) non-availability**: for periods of time, some resources might be unavailable.

- **Machine loading limits**: a product or group of products might exceed the machine manufacturing capacity.

- **Machine blockings** *(block)*: if buffers are full or there is a lack of buffer machines might become bottlenecks.

- **Product quality rejects**: products with low quality might be rejected. This results in the need to remanufacture or manufacture a complete new product if possible.
- **Machine erratic performance**: a non-steady performance of machines might result in quality variations or processing time variations. It might lead to machine replacement and remanufacture of a batch of products.
- **Defective raw materials**: materials that do not meet specifications before their manufacture might be rejected, and cause delay in tasks starting time.
- **Longer machine maintenance time**: delays in maintenance might delay the schedule.

There are many terms that do not have a symbol to describe them, specially the stochastic characteristics. The scheduling problem is a widespread problem across different fields. Examples of scheduling applications are for project scheduling [180], computer processors scheduling [181], and services scheduling (e.g. hospital) [182]. The nature of each problem results in characteristics and different performance. Moreover, the nature of the scheduling problem influences the type of solution methods for each type of problem and the characteristics that can be considered.

Although this section focuses on machine scheduling, scheduling problems share constraints among them. For the SAR problem, the second most relevant problem is the multi modal operation (project scheduling) which corresponds to the alternative process plans from the machine scheduling problem. Process plan selection is of paramount importance to the SAR problem. However, process plan selection can be integrated in the scheduling problem through alternative process plan constraints. The next section focuses on reviewing the Multi Robot Task Allocation (MRTA) problem from the robotics field. This is analogous to the scheduling problem from the operations research field.

4.2.3 Multi robot task allocation, MRTA

The Multi Robot Task Allocation (MRTA) consist of determining task to robot allocations subject to robot capacities and requirements of tasks [183]. The core elements of the problem are a group of tasks, a group of robots (where both can be of one), associated profits to produce each task, and associated costs of producing a certain tasks with a certain robot. The objective is to maximise the profits of producing a group of tasks with a group of robots.

In the field of robotics, tasks are usually homogeneous and performed by homogeneous robots. It does not matter which robot completes which task as long as tasks are completed. However, there is a trend to address more realistic scenarios in robotics. This includes teams of heterogeneous robots that need to complete tasks in a coordinated and specific sequence [184]. Complex scenarios involve robots performing several tasks at different locations such as the treasure hunting problem [185].

MRTA refers to determining the most adequate allocation of tasks to robots constrained to a limited number of robots capable of performing each task [186]. MRTA considers metrics to measure the efficiency of the performed task and the cost of performing the task depending on the robot. MRTA can be solved through a centralised or a decentralised approach. Centralised approaches are capable of providing an optimal solution but it does not perform very well when the problem is scaled and is not robust to failures. Decentralised approaches have the opposite attributes, like scalability, robustness to failures, but it does not provide optimal solutions.

In centralised approaches the cost for performing a task by the robot is calculated by the central system instead of by each individual robot as in decentralised approaches. As a result, the calculation time is distributed, but due to the lack of global information for the individual cost calculation of each robot, the results are sub-optimal.

4.2.4 MRTA taxonomies

The best known taxonomy classifies MRTA problems by the robot capacities, the number of robots performing tasks, and the prior knowledge of the tasks to be performed over time [140]. These are explained in detail as:

- **Robot capacities**: refers to the number of tasks that can be performed by a robot. This characteristic has the following options: single-task robots (ST) and multi-task robots (MT). Single-task robots refer to robots dedicated to perform only one task, whilst multi-task robots refer to robots that are able to perform multiple tasks.

- **Task complexity**: refers to the number of robots required to perform a task. The options of this characteristic are single-robot tasks (SR) and multi-robot tasks (MR). Single-robot tasks can be performed by a single robot whilst multi-robot tasks need to be performed by several robots in collaboration.

- **Data tasks knowledge over time**: refers to the prior knowledge of multiple tasks over a long period of time. This characteristic has the options: instantaneous assignment (IA) and time-extended assignment (TA). Instantaneous assignment considers only the instant or actual data tasks (i.e. similarly to the dynamic scheduling problem where online approaches are required), whilst time-extended assignment considers actual and future data tasks to produce a schedule (i.e. similarly to the deterministic scheduling problem).

The notation for the first taxonomy expresses characteristics separated with dashes (-) e.g. ST-MR-IA means that single-tasks robots perform tasks that require a single robot task for a single period of time. This taxonomy neither considers the dependency between tasks nor its complexity. A taxonomy proposed by Korsah et al. considers the dependency and complexity of the tasks [187]. In this taxonomy, tasks are divided by their complexity in:

- **Complex tasks**: are tasks that can be executed in multiple ways. Complex tasks are decomposed in simple or elemental tasks and each tasks can be executed by multiple robots.
- **Compound tasks**: are tasks similar to the complex ones, but there is only one combination to perform this task. Compound can also be decomposed in simple or elemental tasks.
- **Simple tasks**: are tasks more complex than the elemental tasks, but that still can be performed by a single robot or more. Simple tasks are parts of compound tasks.
- **Elemental tasks**: are single actions executed by only one robot.

Korsah's taxonomy introduced the concept of tasks dependency, which is the level of dependencies between tasks. Tasks are classified according to the task dependency in:

- **Complex dependencies (CD)**: are dependencies in which the profit of a robot for performing a task depends on the performed tasks by all the robots and the way the tasks were performed.
- **Cross-schedule dependencies (XD)**: refer to dependencies in which the profit of one robot not only depends on its own scheduled tasks but also depends on the scheduled tasks of other robots.
- **In-schedule dependencies (ID)**: are dependencies in which the profits of one robot depends only on the tasks that perform. The profit is independent of the sequence to perform the tasks.
- **No dependencies (ND)**: refer to tasks in which the profit of executing task is not affected by past tasks and does not affect future tasks.

The notation for the second taxonomy involves the notation for the first taxonomy (i.e. Robots capacities, tasks complexity and tasks knowledge over time). However, for the second taxonomy the term describing the dependencies is previous to the first taxonomy. For example, the problem XD [ST-MR-IA] describes single-tasks robots perform tasks that require a single-robot tasks for a single period of time. There are cross-schedule dependencies between the single-robot tasks.

Regarding the SAR problem, the use of multi-tasks robots with interchangeable tools and single-task machines are considered. As a result the flexibility of the S-RMS is increased but the complexity of task allocation problem is also increased. In the SAR problem, tasks might require a single robot (i.e. for single task products) or multiple resources (i.e. products with multiple and complex processing sequences). Information about products and their manufacturing tasks considered in advance in the allocation problem (offline planning) but instant information from incoming products requires dynamic allocation of resources (online planning).

According to the second taxonomy, mainly task complexity, the four types of tasks can be considered in the SAR problem. This is as follows: elemental and simple tasks for single task products; compound tasks for products with processing sequences; and complex tasks due to the multiple combinations to decompose tasks to manufacture a product (alternative process plans). The dependency between tasks depends on the similarity of manufacturing tasks that are required simultaneously. For the SAR problem, this is explained as follows:

- Independent robots can produce single-task products (i.e. no dependencies between tasks)

- Independent robots can produce a series of single-tasks products, where the sequence to produce these tasks determine the paths and costs of making the tasks

- Robots in collaboration can produce sequences of tasks for multiple different products (i.e. cross schedule dependencies)

- Robots in collaboration produce sequence of tasks for multiple products, but the efficiency of producing these tasks depend on the processing sequences that are selected for each product (i.e. complex dependencies)

Reviews on approaches to address the MRTA problem are presented in [188],[189]. The MRTA problem is limited in comparison to the scheduling problem. This is because the scheduling problem has been studied since approximately the year 1940 [190], compared to the recently proposed multi-robot task allocation problem (i.e. around the year 1990 [183]). In the next paragraphs, there is a review of optimisation objectives from the scheduling and the MRTA problems relevant to the SAR problem.

4.2.5 Optimisation objectives

This subsection focuses on reviewing optimisation objectives of the scheduling and the MRTA problems. In the scheduling problem, the objectives to optimise focus on minimising production time, whilst in the MRTA problem focus on achieving a higher utility at performing the operations by assigning the most adequate robots to operations. Basic terminology to describe the optimisation objectives in the scheduling problem is the following [84]:

- **Completion time** (C_j): is the time that takes to complete operation j
- **Lateness** (L_j): is the deviation of the completion time of operation j against its due date d_j. This is $L_j = d_j - C_j$. Lateness can be negative or positive
- **Tardiness** (T_j): correspond only to the positive values of the lateness L_j. This is $T_j = \max(L_j, 0)$
- **Earliness** (E_j): correspond only to the negative values of the lateness L_j. This is $E_j = \max(-L_j, 0)$
- **Late job** (U_j): refers to the number of jobs or operations finished after the due date. This is $U_j = 1$ if $C_j > d_j$, or 0 otherwise. Whilst tardiness measures the amount of time that a job is late, late jobs measure the number of jobs that are late.

Specific objectives to optimise depend on the type of manufacturing systems, number of machines of each type, and their machine layout (topology). Other objectives include minimising the idle time of machines, the tardiness (i.e. lateness) to deliver products with respect to the deadlines or due dates, maximising the profits of produced products and minimising the discrepancy between processing times of a group of machines. Relevant objectives to the SAR problem from the scheduling problem are [84],[168]:

- **Maximum completion time or makespan** ($C_{\max}$): refers to reducing the maximal completion time in the entire group of operations. This is $C_{\max} = \max(C_1, ..., C_n)$. Minimising the makespan implies maximising the utilisation of machines
- **Maximum lateness** ($L_{\max}$): minimises the maximal time deviation from all the due dates. This is $L_{\max} = \max(L_1, ..., L_n)$

- **Flow time or total completion time** ($\sum C_j$): refers to minimising the sum of the completion time of all the operations

- **Weighted flow time or total weighted completion time** ($\sum w_j C_j$): refers to minimising the sum of the weighted completion time of all the operations

- **Total tardiness** ($\sum T_j$): is meant to minimise the total tardiness for all the operations

- **Total weighted tardiness** ($\sum w_j T_j$): refers to minimising the total weighted tardiness for all the operations

- **Total earliness** ($\sum E_j$): is meant to minimise the total earliness for all the operations

- **Total number of tardy jobs** ($\sum U_j$): is about minimising the total number of tardy jobs or operations

- **Total weighted number of tardy jobs** ($\sum w_j U_j$): refers to minimising the total weighted number of tardy jobs

- **Total earliness and tardiness** ($\sum E_j + T_j$): is meant to minimise total earliness and tardiness at the same time. This is commonly found in Just In Time (JIT) environments. Hence, for the SAR problem, the formulation should minimise both values simultaneously

- **Total weighted earliness and tardiness** ($\sum w_j E_j + w_j T_j$): refers to minimising the total weighted earliness and tardiness at the same time. Similarly to the previous objective, this is commonly found in Just In Time (JIT) environments

In contrast to the objectives from the scheduling problem, the objectives from the MRTA problem focus on maximising the utility at allocating robots to operations. Each robot determines its own utility to perform an operation. Hence, this decentralised approach scales for large number of operations and robots. The concept of utility U_{rt} to produce a task or operation t with a robot r is defined as the difference between the benefit at certain quality Q_{rt} and the cost C_{rt} of producing the operation t. This is $U_{rt} = Q_{rt} - C_{rt}$. Only positive value of utility are considered in [186] whilst positive and negative values are considered in [140]. The utility U_{RT} for a group of m robots R performing a group of n operations T is given by [186],[140]:

$$U_{RT} = \sum_{r=1}^{m} \sum_{t=1}^{n} U_{rt}$$

This concept of utility does not account for interrelated operations or sequence dependent operations as was described in [140]. It is pointed out that for interrelated operations the utility of all the operations is different to the simple sum of the utilities of each operation t performed by a robot r. There are three cases to measure the total utility of all operations performed by a robot in contrast to the sum of all the utilities of each operation performed by a robot. These are:

- **Independent utilities**:
$$U_{RT} = \sum_{r=1}^{m} \sum_{t=1}^{n} U_{rt}$$
- **Interrelated utilities**:
$$U_{RT} \neq \sum_{r=1}^{m} \sum_{t=1}^{n} U_{rt}$$
- **Interrelated utilities with synergistic relationship**:
$$U_{RT} > \sum_{r=1}^{m} \sum_{t=1}^{n} U_{rt}$$

The MRTA problem is recent in comparison to the scheduling problem. The scheduling problem considers not only the assignation of operations to machines but also the times to perform the operations. The next problem, the machine layout problem (also, known as the Facilities Layout Problem (FLP)) is subsequent to the scheduling problem (i.e. allocation and sequencing of task to resources).

4.3 Positions assigning problem

The Facilities Layout Problem (FLP) is the general problem of locating manufacturing facilities (e.g. factories, departments, machines) to subject to limitations such as the number of facilities and the available area [191],[192]. The main objective for the FLP is to minimise the cost of transporting parts or components between facilities at a known transportation cost depending on the distance and an expected flow of parts or components. Variants of FLP focus on optimisation objectives such as minimising the transport of materials between facilities, or maximising the number of clients that can be served. Industrially relevant FLP variants focus on locating factories or distribution centres to supply as many cities as possible, locating departments within a factory to minimise material handling costs between departments, or locating machines to optimise material handling flow. The last problem (i.e. location of machines to optimise material handling flow) is called the machine layout problem.

In the SAR problem, the positions assigning problem consists of determining the most adequate locations for machines to perform tasks over the total production horizon. These locations can change over time due to the use of mobile manufacturing resources (movable machines and Mobile Manufacturing Robots (MMRs)). Therefore, the machine layout problem is the most relevant to the SAR problem. The adequate location of machines requires optimising two objetives. The first is to locate a machine that can produce as many tasks as possible, whilst the second is to facilitate the movement of machines and MMRs through the complete production horizon. In contrast to current machine layout problems that locate machines within departments and have a limited area and shape, in the SAR problem the focus is on locating machines and MMRs in a single factory. The factory has an unmanned area without obstacles or walls, where machines and MMRs can form layouts according to the current production needs.

Elements, characteristics and their assumptions from machine layout problem influence the design of machine layouts in the SAR problem. Relevant variants of the machine layout and the facility layout (FLP) problems are reviewed in Subsection 4.3.1, whilst variants of the machine layout problem focused on dynamic production requirements are reviewed in Subsection 4.3.2. Elements, characteristics and their assumptions from the machine layout and FLP that are relevant to the SAR problem are reviewed in Subsection 4.3.3. Optimisation objectives related to the positions assigning problem are reviewed in Subsection 4.3.4.

4.3.1 Machine layout problem

The machine layout problem is the most relevant to the SAR problem. Therefore, relevant constraints from the machine layout problem are reviewed in this subsection. Research on the machine layout problem focus on the line layout. Examples include the single-row, double-row, multi-row and loop layouts [102].

In the single-row layout, machines are arranged in a line. Variants of the single-row layout are: a straight line, a semicircular line, or U-shaped line. In the double-row layout, it is possible to locate machines asides a conveying line (or any other material handling system that transport material in a straight line). This facilitates work in parallel of multiple robots and increases the efficiency. The loop layout allows arranging machines in a circle, where the flow of materials is in one direction only. Lastly, the multi-row layout allows the possibility of have multiple rows in parallel, and to transport materials within the same row and among other rows.

Another example of machine layout is the open-field layout, which is not constrained to form line layouts. This type of problem is similar to the type of machine arrangements that occur in a job or open shop environment (i.e. from the scheduling problem). Hence, the open-field layout is of paramount importance to the SAR problem.

4.3.2 Dynamic layout problems

Nowadays, due to a dynamic market economy, the groups of products to manufacture (production mix) are in constant change (i.e. dynamic production mix). Currently, there are three options to manage dynamic production mix, which are robust layout, dynamic layout and reconfigurable layout problems [193]. It is common to refer to each production period for a production mix as a production horizon [194],[195]. The dynamic and reconfigurable depend on how much information about the production mix(es) over production horizon(s) is known in advance. The robust and the dynamic depend on how easy it is to change layouts for each group of production requirements. The layout problems are [193]:

- **Robust layout**: consist of determining a single layout for all the production mix(es) of each production horizon. The objective is to find a layout that in average minimises the material handling cost for all the production mix(es). The layout is not optimal for any production mix, but in average minimised the material handling cost of all the production mix(es).

- **Dynamic layout**: refers to changing the production layout for each production mix [196]. Hence, there is a layout for each production horizon. The objective is to determine an optimal layout for each production mix whilst at the same time minimising the cost of rearranging machines for each production mix.

- **Reconfigurable layout**: similarly to the dynamic layout problem, this problem consists of determining a layout for each production mix. However, only information on the current and next production mix(es) is known. Therefore, the objective focuses on minimising material handling cost for current and next production mix(es) and minimising the cost of rearranging machines between the current and the next layout [193].

The difference between the SAR problem and this type of problems was described in the introduction to the SAR problem (Section 4.1). In brief, the SAR problem increases the flexibility of the dynamic and reconfigurable layout by allowing the relocation of machines for each manufacturing operation, instead of relocation of machines at each change of the production mix. Relevant elements, characteristics and assumptions from the FLP and machine layout problems that can be considered in the formulation of the SAR problem are reviewed in the following subsection.

4.3.3 Machine layout elements and characteristics

The following elements, characteristics and assumptions (i.e. constraints) can be found in the formulation of the FLP and machine layout problems. Elements, characteristics and their assumptions from machine layout problem influence the design of the machine layouts in the SAR problem. Relevant elements, characteristics and assumptions for the SAR problem formulation are [103]:

- **Space representation**: refers whether the problem is represented as discrete or continuous. A discrete representation might be easier, but it might not determine feasible layouts, whilst a continuous representation might found feasible layouts for hard problems but at a higher computational cost. Discrete representations are not adequate to address constraints such as clearance between machines, orientation of machines and points to load or unload raw materials into a machine.

- **Constant facilities shape**: refers to whether the facilities have a fixed shape or not [197]. The FLP problem focuses on assigning areas to the facilities (e.g. factories and departments), whilst the machine layout focuses on facilities with fixed shape (e.g. machines or robots). Regarding the SAR problem, warehouses shape can change over time depending on the shelves that are added to them, whilst machines and robots have constant shapes.

- **Facilities shape**: is about the possible shapes that facilities might have. The options are regular or irregular. Regular refer to rectangles or squares, whilst irregular to polygons with at least an angle of 270°, see Figure 4.2. Within the regular shape option, facilities might have equal or unequal area among them. Departments can be constrained to have fixed shapes. In the SAR problem, warehouses can have irregular shapes, whilst is expected to have regular shape machines, but robots can have regular, irregular or even circular shapes.

- **Non-overlapping**: make sure facilities have unique locations where facilities do not overlap with each other.

- **Clearance**: refers to a free space (i.e. aisles [123]) among machines. This factor is considered to allow servicing and maintenance of machines, but might serve as well as a security factor to avoid unnecessary collisions. Clearance is commonly represented with increased size of the facilities. Clearances in the SAR problem should consider dimensions of MMRs or AGVs to facilitate movement between machines or other MMRs.

- **Orientation**: is about the orientation of machine in a factory. The orientation of department and factories does not matter as long as they have the required area. Regarding the SAR problem, machine orientation is important to know the access points to load and unload of raw materials.

- **Zone constraints**: refer to constraints that promote or avoid the location of machines close to each other. These are also known as adjacency constraints when it is desired to located a machine next to other machine to facilitate the utilisation of similar resources [198]. In the SAR problem, locating machines or MMRs next to each other facilitates the interchange of tools.

- **Repeated machines**: refers to having multiple machines with the same capacities. This was analysed in an extended version of the FLP, where equal facilities can be located separately over a long distance [199]. The conclusions were that repeated facilities improves the accessibility to these facilities and diminishes the material handling effort. In the SAR problem, the use of resources with repeated capacities is assumed.

- **Budget constraints**: are typical to the dynamic FLP, where machines or departments can change locations during the production horizon. The cost of changing facilities is restricted by the budget constraints.

- **Uncertainty of the material flow**: the material flow can be deterministic or stochastic for the production horizon. The static FLP assumes deterministic values of the material flow whilst the dynamic FLP assumes stochastic values.

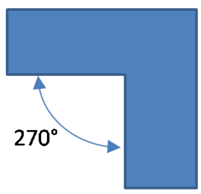


Figure 4.2: Example of an irregular facility. This type of facilities contain angles of 270 degrees.

The FLP or machine layout address problems that consider production requirements that are fixed for the whole time (static). However, there is a trend towards production requirements that continuosly change over time (dynamic). This trend was also observed in the scheduling problem and is addressed with dynamic scheduling approaches. In a similar way, the FLP has a dynamic counterpart, the Dynamic Facilities Layout Problem (DFLP). Approaches to address static and dynamic production requirements are reviewed in the next paragraphs.

4.3.4 Optimisation objectives

The objective(s) to optimise depend on the type of Facilities Layout Problem (FLP) to solve. As was introduced, the FLP have two main variants, the static and the dynamic FLP. In the next paragraphs, relevant optimisation objectives to the SAR problems are reviewed according to the FLP variants.

- Static FLP [123].
  - Distance and flow between facilities are considered. This objective function minimises the distances between facilities with the higher flow costs. This can be applied to discrete and continuous problem formulations.

    $$\min \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \sum_{l=1}^{N} f_{ik} d_{jl} X_{ij} X_{kl}$$

    where:
    N: number of facilities in the layout
    f_{ik}: flow cost from facility i to k
    d_{jl}: distance from location j to l
    X_{ij}: binary variable (0,1) for locating facility i at location j

  - Adjacency between facilities is considered. In contrast to the previous approach, distance between facilities are not know, but the cost between facilities is known. This can be applied only to discrete problem formulations.

    $$\min \sum_{i} \sum_{j} r_{ij} x_{ij}$$

    where:
    r_{ij}: closeness rating between departments i and j
    x_{ij}: binary variable (0,1) equals 1 if departments i and j are adjacent and 0 otherwise

- Dynamic FLP [200],[201]. Minimises the material handling cost (first term) among facilities and the reconfiguration costs (second term) between facilities over periods for the complete planning horizon. This can be applied only to discrete problem formulations.

$$\min \sum_{t=1}^{P}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{m=1}^{N} F_{tijkm}X_{tij}X_{tkm} + \sum_{t=2}^{P}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{m=1}^{N} A_{tijm}Y_{tijm}$$

Where:

i, k: departments in the layout
j, m: location in the layout
Y_{tijm}: binary variable for shifting i from j to m in period t
A_{tijm}: fixed cost of shifting i from j to m in period t (where $A_{tijj} = 0$)
X_{tij}: binary variable for locating deparment i at j and k located at m in period t
F_{tijkm}: cost of material flow between departments i located at j and k located at m in period t
P: number of periods in the planning horizon
N: number of departments in the layout

In contrast to the scheduling problem, there is not a common notation for the FLP. The only distinction is between the static FLP and the Dynamic Facilities Layout Problem (DFLP). The purpose of analysing the FLP and machine layouts problems is to identify their relevant elements, characteristics and assumptions that can be applied to the SAR problem. Therefore, solution approaches are not analysed. However, reviews of solution approaches for the static problem can be found in [202], whilst for the dynamic problem in [203] and [204]. The next section focus on the routing and path planning problem. Similarly to the FLP, in the path planning problem, it is necessary to represent the environment, machines and obstacles. The path planning problem allows more variety at representing the environment, machines and obstacles than the FLP.

4.4 Routing and motion planning problems

These two problems are about determining paths or routes to move vehicles from an initial position to target one in the minimal time, subject to constraints due to the type of robot and the available information from the sorrounding environment. However, whilst the routing problem refers to selecting the most adequate route, the motion planning problem refers to determining or designing a route or path. The routing problem is commonly known as the Vehicle Routing Problem (VRP), but in the rest of this thesis the name might be shortened to routing problem. Elements, characteristics and assumptions of each problem influence different aspects of the SAR problem. Elements, characteristics and their assumptions from the vehicle routing influence the paths or routes of materials and products transportation.

The motion planning problem and the routing problems are reviewed in Subsections 4.4.1 and 4.4.3, respectively. From the motion planning problem point of view, Subsection 4.4.2 covers abstract characteristics such as the representation of objects and the environment as well as practical characteristics and elements about robots. Subsection 4.4.4 presents a taxonomy and classes of problems, whilst 4.4.5 covers elements and characteristics of the routing problem. Optimisation objectives related to the routing and motion planning problems are reviewed in Subsection 4.4.6.

4.4.1 Motion planning

The motion planning problem consists of two problems, namely the path planning and the trajectory planning problems. The path planning consist of determining a path from initial position to target position subject to environment constraints (e.g. obstacles), robot constraints (e.g. type of wheels, on board sensing and communication capabilities) and external disturbances or stochastic information. In addition to determining a path, the trajectory planning problem involves determining times associated with each point along a path. In order to determine times for each point of the path, it is necessary to consider the maximal velocity and acceleration of a robot.

This is important to handle moving obstacles, a changing environment or multiple moving robots. A subproblem of the path planning is the multi-robot coordination problem. This problem refers to coordinating the movement of several robots to reach their final positions without crashing or blocking each other. In case of closed formations, a common strategy is to assign (dynamic) priorities to avoid leaving a robot outside of the formation [205].

There is not formal notation for the motion planning problem. However, a notation proposed by Latombe in [206] is still under use, but each author make minor changes and additions depending on the problem and its context. Taxonomies about the motion planning problem focus on the different type of motion problems or the solution methods. A review of characteristics relevant to the SAR problem is done in the next paragraphs.

4.4.2 Motion planning elements and characteristics

The core elements of the motion planning problem are: initial and target positions, environment and obstacles representation, robot representation and its characteristics (e.g. type of wheels, maximal velocity, maximal acceleration, on board sensing). Elements, characteristics and their assumptions from the vehicle routing influence the paths or routes of materials and products transportation in the SAR problem. Most of the assumptions (i.e. constraints) focus on the characteristics of robots and the environment where robots move. These characteristics are [207],[208]:

- **Dynamic data**: is defined by whether the information about the environment is constant or requires update due to moving obstacles.
- **Environment representation**: similarly to the machine layout problem, the motion planning requires to represent the environment. In a similar way, it can be represented in discrete or continuous way. However, the motion planning problem has more specialised techniques such as occupancy grid, topological maps, and metric maps. These are not reviewed because they are part of the solution and not part of the problem definition.
- **Objects representation**: unlike the machine layout problem, robots can be represented as curvilinear figures in addition to polygonal representations. The motion planning problem has specialised techniques such as polyhedra, cell tree, grid, boundary representation, and constructive solid geometry.
- **Robot's type of wheels**: might be holonomic or non-holonomic types. Non-holonomic wheels type constraints robots to move in a direct line, whilst with holonomic wheels type robots have to maneuver to solve complex paths. Holonomic constraints are common for car-like wheeled robots, whilst non-holonomic are for robots with omnidirectional wheels. Other common type of wheels is differential wheeled, which consist of only two wheels at each side of the robot.
- **Robot maximal velocity and acceleration**: result in dynamic constraints. When the problem involves avoiding static obstacles and controlling the velocity and acceleration, kinodynamic constraints are necessary.

The next characteristics are related to robots but proposed only for the SAR problem. These characteristics further describe the problem. All the following characteristics might be assumed homogeneous or heterogeneous for all robots or tools respectively. These characteristics might be neglected in order to simplify the problem at the cost of realism of the problem. The characteristics are:

- **Robot's type of manipulator**: this might result in different manipulator workspaces.

- **Robot's manipulator payload**: is the maximal amount of weight that the arm in the Mobile Manufacturing Robot (MMR) can carry.

- **Tool weight**: is related with the manipulator payload. Specialised robots might be able to carry very heavy tools.

- **Robot's maximal health span**: refers to the robots condition and performance to execute an operation or move accurately. This characteristic indicates when a robot requires maintenance.

- **Robot's maximal tools**: refers to the maximal number of tools that a robot can carry. For the SAR problem, it is assumed robot with multiple interchangeable tools, so that robots can perform several operations by changing tools. Hence, this characteristic determines the maximal number of operations a robot can perform without visiting a tool depot.

- **Robot's manipulator accuracy**: is the smallest accuracy at which the manipulator of a MMR is capable of performing operations. This is important to match product and its operations quality requirements.

- **Robot's platform accuracy**: refers to the smallest accuracy with which the platform of the MMR can move. The importance of this characteristic lies in the easiness to move within a cluttered environment. This might result in a more effective use of the manufacturing area.

These characteristics are focused on the problem of determining a feasible or optimal path for one or multiple robots. However, characteristics that are specific to the problem of moving objects between positions from and to one or several warehouses are described in the following subsection, the routing problem.

4.4.3 Routing

The routing problem is an abbreviation for the vehicle routing problem. This one should not be confused with the term routing in the scheduling problem, which refers to selecting a proces plan from a variety of plans. The basic (i.e. core) problem is called the Capacitated Vehicle Routing Problem (CVRP) or just Vehicle Routing Problem (VRP) because the vehicles have limited carrying capacity.

The VRP involves a fleet of homogeneous vehicles (i.e. mobile robots) and a single depot (i.e. warehouse), where robots start and finish their routes (i.e. a closed loop). Each route has a known travelling time, distance or cost and the aim is to visit all desired positions with a minimal sum of routes travelling time, distance or cost. The objective is to select a subgroup of routes from a possible network of routes (arcs) that minimises travelled distance, time or cost to deliver objects to requesting positions (nodes) in the network. The solution might involve joining together several short routes in order to obtain a larger route. The final routes must start and finish at depots or warehouses, and for the CVRP the routes must consider the vehicles' carrying capacity.

For the SAR problem, the network of limited routes might connect warehouses to temporally fixed manufacturing positions. However, in order to fully take advantage of the S-RMS and MMRs, it is desired to constantly determine the most adequate manufacturing positions for each operation. Therefore, the routes network is in constant change.

4.4.4 Types of vehicle routing problems

Variants of the VRP consider more realistic scenarios. For the SAR problem, the most relevant VRP variants are described in the following paragraphs. The assumption of using a homogeneous fleet is removed, so that a heterogeneous fleet can be used. This is known as the **Heterogeneous Fleet Vehicle Routing Problem (HFVRP)** or as the **Mixed Fleet Vehicle Routing Problem (MFVRP)** [209],[210]. Another variant results by removing the constraint of using a single warehouse [211]. This is the **Multi Depot Vehicle Routing Problem (MDVRP)**, which considers the use of multiple warehouses that are distributed among the required delivery positions.

The **Vehicle Routing Problem with Time Windows (VRPTW)** considers the delivery of objects to be done in determined time intervals [212]. Depending if these constraints are hard or soft, there are two variants of the problem. If the window constraints are soft, a penalty cost is added. There are two more variants if there are time windows, for unloading or loading, at the required positions or depots.

The **Vehicle Routing Problem with Pickup and Delivery (VRPPD)** do not consider a single warehouse [213]. Instead objects are picked up from certain positions and delivered to other positions with the same vehicle. A subvariant of this variant is the **Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPSPD)** refers to making pickups and deliveries in the same position of the route at the same time [214].

In the **Split Deliveries Vehicle Routing Problem (SDVRP)**, requesting points can be visited several times [215]. This means, that objects can be delivered in multiple trips. Hence, the requesting point can be visited several times. In this problem, the requests can exceed the robots capacity because the demand can be divided (i.e. split) in several trips. A related variant is the **Periodic Vehicle**

Routing Problem (PVRP), which considers repetitive deliveries at a long term planning [216]. In the **PVRP**, deliveries can be done over several days. In contrast to previous problems, robots can visit to the same delivery positions many times, but at a limited frequency.

In the **Open Vehicle Routing Problem (OVRP)**, vehicles are not constrained to return to the warehouse at the end of their route [217]. In practice, this refer to subcontracted vehicles that pick up objects at the warehouse, deliver objects along its route and return to their own storing position. The **OVRP** deals with two objectives, namely minimising the use of vehicles and minimising their travelled distance or time. A subvariant of the **OVRP**, allows returning to the warehouse but vehicle have to use the same route in reverse order [218].

In the **VRP** the data on deliveries and pickups has an important role. The previous problems have assumed deterministic data. However, the data can be stochastic or unknown. This data refers to demands, request times (i.e. time when an order was made), service times (i.e. loading or unloading time), and travel times (i.e. route travelling time).

Stochastic data is data with certain degree of uncertainty represented by probability distributions. Regarding stochastic data, the most relevant problem is the **Vehicle Routing Problem with Stochastic Demands (VRPSD)** [219]. In contrast, unknown data refer to data that is provided or updated in real time. This type of data is addressed through the **Dynamic Vehicle Routing Problem (DVRP)** [220]. The **DVRP** is also known as the on-line VRP or the real-time VRP.

The **Time Dependent Vehicle Routing Problem (TDVRP)** considers travel time or service times as dependent on a function of the current time [221]. In practice, this involves considering the rush hour traffic or multiple deliveries at the same time.

The notation for the VRP is embedded in the names of the problems. This make it complicated to add more than two characteristics. For example, the split delivery VRP with time windows is described by the term SDVRPTW [222],[223]. A suggestion for a more readable notation could include separators between the characteristics and rules about use of prefixes and suffixes such as the scheduling problem (e.g. SD-VRP-TW). The VRP considers a single warehouse, however the SAR problem might consider multiple and distributed warehouses, with homogeneous or heterogeneous content.

4.4.5 Routing elements and characteristics

A vehicle routing taxonomy was proposed in [224],[225],[226]. Elements and characteristics that describe the VRP are analysed from this taxonomy. Elements, characteristics and their assumptions from the vehicle routing influence the paths or routes of materials and products transportation in the SAR problem. Basic elements of the VRP can be divided in elements describing the available resources and elements describing the requirements. These elements along with characteristics that describe the problem are also explained in the following paragraphs.

Elements describing the available resources (assumed deterministic):

- **Number of vehicles or robots**: is considered fixed for the core problem (i.e. the capacitated VRP). Other variants constraint the use of robots to an specific number n, or consider an unlimited number of robots. The SAR problem considers a number greater than one.

- **Robots capacity**: refers to how much payload can the robots carry, or the number of objects independently of their weights. In the SAR problem, the additional constraint of robot battery capacity might be considered or neglected. In order to manage this constraint, distributed battery chargers might be considered, and travelling routes might need to consider passing through charging points after certain working time.

- **Number of warehouses (depots)**: the core problem considers a single depot, where robots must start and end their trip (i.e. a closed loop). However, multiple depots are considered in variants of the problem. The derived problem of collecting the objects within the warehouse are not considered. The SAR problem considers warehouse with either homogeneous or heterogeneous contents, without closed walls, that make possible for robots to collect objects from the outer boundaries of the warehouse. However, if the mix of objects is not uniform robots might be forced to enter inside the warehouse and consider internal networks of shelves within the warehouses.

- **Routes network**: is a predefined network of paths that connect positions to collect or deliver objects. In case of the SAR problem, these objects might be raw materials or products, and the positions are warehouses and manufacturing positions where a machine or a MMR is able to perform an operation. The network might be directed or undirected. The directed refers to single way routes, whilst the undirected to double way routes.

Elements describing requirements (assumed deterministic, stochastic, unknown (i.e. to be known on real time) or time-dependent):

- **Number of stops on route (nodes in the network)**: refer to positions in the network that have requested collections or deliveries. These can be known or partially known. In the SAR problem, these stops refer to machines or MMRs that request raw materials or collection of manufactured products.

- **Demands at the stops**: are the quantity that each position in the network is requesting. These can be for deliveries or collections of objects. Demands can be known or stochastic. In the SAR problem, the delivery of raw materials or collection of products is requested by machines or MMRs.

- **Request times**: are the times at which requests for objects are made. In the core problem, the demands are known in advance. However, in the dynamic VRP, requests can be made on real-time. In the SAR problem, the request times depends on the production planning. For a global planning, request times are known in advance, but for distributed or online planning, the request times are known in real-time.

- **Travelling times, distances or cost of routes**: refer to the times or costs it takes to travel from one point to another within the routes network. This data might be deterministic, stochastic, unknown (known in real-time) or time-dependent (e.g. rush hour). For the SAR problem, this data depends on the location of machines and MMRs in the layout. This data is provided with the facilities layout problem.

Vehicle Routing Problem (VRP) characteristics (most of them deterministic):

- **Load splitting**: refers to whether the collections or deliveries can be divided in several trips, instead of a single trip. In the core problem the assumption is that each position requesting collections or deliveries can be visited once. In a similar way, in the SAR problem, the load can be divided and carried with several robots through several trips.

- **Vehicle homogeneity** is about the type of robots that can be used. In the core problem (i.e. capacitated VRP), a homogeneous fleet is considered. In the SAR problem, robots or machines can be considered homogeneous or heterogeneous.

- **Service time**: refers to the waiting time at the positions in the network to provide on-site services (i.e. collect or deliver objects). This is the only characteristic that can be stochastic. In the SAR problem this is referred as loading and unloading times. Moreover, AGVs can be used as handling robots to collaborate with MMRs in complex manufacturing operations.

- **Time windows**: are a predefined period where collections or deliveries must occur. These constraints can be soft, strict (hard), or a mix. There might be time windows at the positions in the network (machines or MMRs), at the warehouses, or at the vehicles (AGVs). In the SAR problem, time windows can occur only at the warehouses. The rest of time windows are subject to production planning.

- **Type of request**: refer to the type of service that can be provided at positions in the network. The services can be objects delivery, collections or both at the same position. Regarding the SAR problem, the three types can occur, although a mix of collections and deliveries at the same position is highly complicated.

- **Time horizon**: is the number of periods that are addressed in the problem. These can address a single or multiple periods. In the SAR problem, it is expected to address a single period. However, for future work it is expected a multi period version.

Characteristics tailored to the SAR problem (i.e. consider robot limitations:

The first four characteristics can be assumed homogeneous or heterogeneous and the rest can be considered or neglected.

- **Robot's maximal battery span**: determines the maximal number of hours a robot can perform operations or move along the factory. This characteristic might constraint robots to make short operations and keep recharging its battery.

- **Robot's maximal platform payload**: is the maximal weight a robot can carry. Whilst the volume of objects is not considered, the weight is important to known the type and number of objects a robot can carry.

- **Objects content in warehouses**: refer to the type and the number of objects in warehouses. Assuming multiple and distributed warehouses. This characteristic might include raw materials from different products per each warehouse (heterogeneous), a single type of raw material per each warehouse (homogeneous) or a mixture of both.

- **Objects content in shelves**: similarly to the last characteristic, and assuming an heterogeneous content of objects in the warehouses. Shelves migh have also homogeneous or heterogeneous content of raw materials. This might facilitate an AGV to pick up different materials from the same shelf.

- **Objects to shelves loading/unloading time**: this might be dynamic or constant. If it is dynamic, it might depend on factors such as the type of object (raw materials or products), and the type of manipulator robot (kinematic configuration result in different workspaces).

- **Transport time of objects to/from shelves**: similarly to the last characteristic, this might be dynamic or constant. For dynamic instances, it might depend on factors at which the object is transported. These factors are time (e.g. rush hour and heavy traffic), location of shelves within the warehouse, type of object (objects' special handling requirements) and type of robot (type of arm configuration).

4.4.6 Optimisation objectives

The VRP is considered as a generalization of the Travelling Salesman Problem (TSP) [227]. This is because the TSP only considers a single vehicle and the VRP considers multiple vehicles. Hence, the VRP is analogous to the multiple Travelling Salesman Problem (mTSP) [228]. The addition of constraints to force the vehicle to finish at the starting point (i.e. warehouse) was formulated in the truck dispatching problem [227]. In addition, the multicommodity vehicle routing problem was also formulated. However, the most common formulation for the core VRP (i.e. the capacitated VRP) is by Garvin [229],[230]. This is as follows:

$$\min \sum_{(i,j)\in A} \sum_{k\in V} c_{ij} z_{ijk}$$

Where:
$A = N \times N$: is a set of arcs
V: is a set of vehicles
c_{ij}: is the travel time from node i to node j
z_{ijk}: is a binary variable that is set to 1 if the vehicle k traverses the arc (i, j) or 0 otherwise

The purpose of analysing elements, characteristics, assumptions, notations and optimisation objectives from the motion planning and vehicle routing problems is to identify relevant elements, characteristics, assumptions, notations and optimisation objectives for the SAR problem. A review of common solution approaches for the VRP can be found in [231].

4.5 Concluding remarks

In this chapter the analysis of the constituent problems of the SAR problem was made. The notation, core elements, characteristics and optimisation objectives of each problem were analysed. Moreover, additional characteristics that result in variants of the main problems were reviewed. These characteristics were limited to the ones relevant to be included in the SAR problem.

The scheduling and the vehicle routing problems are well studied problems. They both have notations that express their multiple variants. However, the VRP notation is complicated due to the addition of prefixes and sufixes without any dash to distinguish each additional characteristic. The facilities layout problem has an insufficient notation, where additional characteristics are expressed only in the problem formulation. The motion planning notation depends on each author, and the specific characteristics of the problem. There is a lack of a common notation capable of describing the comprehensive and complex problem that is the SAR problem. Hence, in the next chapter, Chapter 5, a notation for the SAR problem is proposed.

Elements and characteristics of each of the problem can be included or not in the SAR problem. However, this result in a decision making problem about how to formulate the SAR problem. Moreover, constraints in the formulation can neglect elements and characteristics or they can be simplified to have the same value. Therefore, a decision making analysis is done in Chapter 7 with a methodology presented in Chapter 6.

Chapter 5

The SAR problem: definitions, assumptions and notation

Constituent problems of the Scheduling, positions Assigning and Routing problem (SAR) problem were analysed in the last chapter. Also, its main elements, characteristics, constraints, optimisation objectives and available notations and taxonomies were analysed. It was concluded that there is a lack of a common notation and a comprehensive group of elements, constraints, optimisation objectives and notation to describe the comprehensive SAR problem.

This chapter focuses on presenting a comprehensive and coherent group of elements and characteristics (i.e. a generic notation) that describe the SAR problem (i.e. production planning with S-RMS within a single factory). Fundamental assumptions that characterise the manufacture with Self-Reconfigurable Manufacturing System (S-RMS) are provided in Section 5.1. The generic notation consist of vectors with variables and fields that describe elements of the SAR problem and characteristics of these elements. This includes symbols referring to set of elements, variables and fields.

The chapter is divided by the type of elements to consider in the SAR problem formulation. These are the following:

- Factory (Section 5.2): includes elements such as the factory and its warehouses (e.g. factory and warehouses physical dimensions), and the way the factory operates (e.g. factory frequency to take new orders, frequency to produce orders)

- Production requirements (Section 5.3): cover the production orders and their production requirements. These production requirements refer to the type of product to manufacture and their production requirements (e.g. deadline, demand, quality, production sequences). The entire set of product to produce in the same period is called product mix.

- Manufacturing resources (Section 5.4): include available elements to produce dynamic production orders. Examples are robots, machines, tools, area to manufacture. It also includes the way resources operate (e.g. robots maximal speed, robots kinematic configurations).

Each of these three sections provide definitions of the elements they cover. Additional elements and characteristics that are unique to the SAR problem are presented through the last three sections. The complete notation for these elements and characteristics is presented in Section 5.5. Also, examples of assumptions and the simplified notation is summarised in this section. The strategy of formulating of the SAR problem as a decision making problem is introduced in Section 5.6. Concluding remarks are provided in Section 5.7.

5.1 Assumptions that characterise the SAR problem for a single factory

The flexibility of the S-RMS is increased with a specially designed factory. The operation of the S-RMS in a single factory and the complexity of determining production plans with the use of S-RMS are described in the following paragraphs. It is considered a factory that can manufacture different products for multiple customers with a S-RMS. Hence, the aim is to produce different products (i.e. product mix) with different requirements (e.g. product demands and deadlines).

In existing factories the manufacturing layout is fixed or considerable time is required to change the current layout. In comparison with traditional factories, the proposed factory is in constant reconfiguration of its machine layout within a short time. This is achieved through the use of the S-RMS, namely the Mobile Manufacturing Robots (MMRs) and movable machines to form layouts tailored to the current production requirements. Moreover, the manufacturing capabilities of the resources can be changed through interchangeable tools and programming different manufacturing operations.

These levels of flexibility result in a factory capable of managing external and internal changes (e.g. processes and parts' variability and uncertainty, random events). Real-time conditions of the factory and its resources are monitored with the goal to continuously adapt to these random events. The resources can autonomously and intelligently adapt to processes and parts' variability and uncertainty. Moreover, it is possible to manufacture one-off products, bespoke products, or batches of low production volume. An abstract factory (i.e. no distinguished shape and boundaries) that highlights the flexibility (i.e. multiple acesses to factory and warehouses) is shown in Figure 5.1.

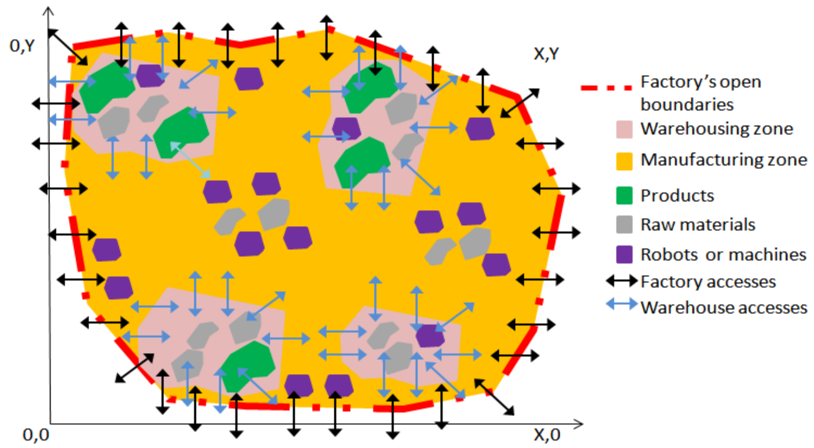


Figure 5.1: Representation of an abstract factory (objects represented as polygons) is located within a coordinate system (x,y), where the factory is represented by a polygon and the polygon points are represented in the cartesian system. The factory has no external or internal walls. This is indicated with the dashed red line. This facilitates the inwards and outwards accesses of raw materials and products respectively. This is indicated with bidirectional black arrows. Warehouses are multiple and distributed along the manufacturing zone. Warehouses are represented by pink polygons. Warehouses do no have exterior or interior walls and this is indicated with the bidirectional blue arrows. Raw materials are represented by grey polygons whilst products are represented by green polygons. Robots and machines are represented by purple polygons. Raw materials, products, robots and machines in warehouses represent stored objects, whilst in the manufacturing zone represented manufacturing layouts. There is no clear manufacturing zone, which means that all the factory can be used except for the warehouses.

In a general way, the SAR problem covers the problems of picking up of raw materials at warehouses, manufacturing products and delivering of products to warehouses. The problems of supplying raw materials to warehouses and picking up products from warehouses are not considered. Assumptions that endow the characteristics of the SAR problem are described in the next paragraphs. These assumptions consider the S-RMS and a single factory at their highest level of flexibility. The assumptions are divided by the main three elements they describe, namely factory environment and operation, available resources and production requirements. These are the following:

General

1. Objects are the factory, warehouses in the factory, robots (MMRs), machines, Automated Guided Vehicles (AGVs), tools, tool depots, raw materials, and

products

2. All the objects' positions are referenced from each of their centroids to an absolute reference coordinate system. Their locations can be dynamic for objects such as robots or machines, but locations can be known at each moment of time with certain degree of uncertainty

Factory environment and operation:

1. The problem considers a single factory. The location of the factory and its boundaries are constant over time.

2. The walls of the factory have as many doors as possible to facilitate movement of raw materials and products from and to the exterior.

3. The factory has two predefined zones, i.e. warehousing zone and manufacturing zone. The warehousing zone can store mix of raw materials, products and manufacturing resources whilst the manufacturing zone, an unmanned area, is only for manufacturing.

4. There are no walls between the warehousing and manufacturing zone. However, they can only be used for their own purpose.

5. The warehousing zone might include multiple and smaller warehouses that might have the same type or a mix of raw materials, products or manufacturing resources (MMRs and machines).

6. The location and boundaries of warehouses can change over time or they can be constrained to static positions and shapes.

7. The factory is equipped to facilitate distributed and continuous manufacturing. The equipment includes but is not limited to distributed sensors, wireless communications, battery-charging zones and tool depots.

8. The factory is equipped with systems to promote CO_2 reduction, as well as efficient resources use (i.e. energy, water).

Available resources

Machines and robots:

1. Robots have known and constant arm kinematic, wheels configuration, values of battery life, accuracy, payload, velocity and acceleration

2. There is a known and constant maximal number of tools that a robot can carry at the same time. The carried tools can change over time

3. Robots can carry multiple types of raw materials at the same time with different quantities and these raw materials can change over time
4. There are limited and known number of robots within the factory. Extra robots can be used from outside the factory at a known cost and delay time per each robot
5. Robots can exchange tools on their own, place them and pick them up from depots and exchange tools with other robots
6. The volume of the transported parts is neglected, but weights are considered. Robots can carry parts of any volume independently of how many parts are assembled together. The only limit is given by each robot payload (both, for mobile platform and arm robot)

Tools:

1. Tools can perform multiple operations (multitasking) or a single operation
2. There is limited and known number of tools with known position and accuracy
3. Depots to store tools are distributed along the warehouses. Depots positions are known but they can change over time. Depots have a group of constant tools with a maximal storing quota and a delay time to resupply tools

Production requirements

Products and manufacturing operations:

1. Products might require multiple or single manufacturing operations.
2. Products that require multiple operations might have a sequence among the operations or not, or both (i.e. some operations might have a sequence among a smaller group and other operations without any sequence)
3. The logistics problem to pick up products from warehouses is not addressed. It is assumed that this activity do not disturb the activity of unloading products from robots to warehouses
4. Each product has a constant and known unloading time to warehouses, raw materials have a known and constant unloading time from warehouses
5. Processing times and required quality of operations are known but might depend on the MMRs or machines that perform the operations

Raw materials:

1. Raw material is treated as a single type of object. Raw materials might require any type of operation (i.e. assembly or machining). Materials might have single operations or multiple operations

2. The logistics problem to (re)supply raw materials into warehouses is not addressed. It is assumed that this activity do not disturb the activity of loading materials from warehouses to robots

3. Each material has a constant and known loading and unloading time

4. It is assumed that the boundaries of the raw materials change over time. However, it is also assumed that robots can handle any volume and shape. It is also assumed that the assembled parts are considered as a single part

In the next three sections, definitions and notation to describe elements of the SAR problem are presented.

5.2 Factory design and operation

In this section, notation to describe the factory design and its operation is presented. This notation consist of vectors for each constituent element of the SAR problem. Variables are distinguished from parameters in that the measure of variables changes over time (e.g. current carried payload by any robot), whilst the measure by parameters is constant (e.g. maximal robot payload). Variables add the term (t). Subindexes denote variable elements, and superindex denote constant elements.

5.2.1 Definitions

General definitions

- *Object* is defined as any physical element that is described by a closed group of points. For this approach objects are: robots, raw materials and products

- *Shape* is defined as a group of points defining the boundaries of an object and is assumed that these are limited to simple polygons. A shape is expressed as $\sigma(t) \in \mathbb{R}^n \times \mathbb{R}^2 = (X(t), Y(t))$ where $(X(t) = \{x_1, x_2, ..., x_{N_{points}}\}$, and $Y(t)) = \{y_1, y_2, ..., y_{N_{points}}\}$

- *Location* refers to the centroid (shortened to cent) of object's shape

Factory and warehouses

- *Zone* is defined as any non-physical element, which is represented by a group of points or the union of groups of points defining their boundaries. Zones are divided into warehousing zone and manufacturing zone, both are contained within the factory zone, see Figure 5.2. A zone is defined as $z(t) \in \mathbb{R}^n \times \mathbb{R}^2 = (X(t), Y(t))$ where $(X(t) = \{x_1, x_2, ..., x_{N_{points}}\}$, and $Y(t)) = \{y_1, y_2, ..., y_{N_{points}}\}$

- *Manufacturing zone* is defined as any zone within the factory to manufacture parts

- *Warehousing zone* is defined as any zone within the factory to store parts, products or robots

- *Shelf* is a vertical container that can store several products or raw materials in a vertical way. They are stored in the warehousing zone. A *depot* is a special type of shelf just for tools. Depots can be in the warehousing or manufacturing zones

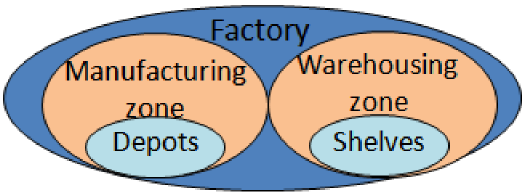


Figure 5.2: Types of zones within a factory. The factory is divided in two main zones, the manufacturing zone and the warehousing zone. None of these areas have walls, but the zones are solely used for their purpose. The warehousing zone can be divided in multiple independent and distributed warehouses that might store a mix of parts or products. Warehouses have shelves to store raw materials or products. Depots (shelves) to store manufacturing tools can be distributed in the warehousing zone. These depots can have constant or dynamic positions.

5.2.2 Notation

t Instant of time

Where:

$t \in \mathbb{R}_0^+$
Factory vector

$\mathbf{f}$ = (factory zone-shape-, location, warehouses, products, raw materials, robots, machines, tools, depots of tools)

$$\mathbf{f} = (z^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{B}, \mathcal{D}, \mathcal{S}^d)$$

Where:

$z^f = (X, Y)$: Complete factory zone

$(x^{cent,f}, y^{cent,f})$ Factory location

$\mathcal{W}$: total group of warehouses, where each warehouse can store raw materials, products, tools and robots

$\mathcal{P}$: total group of products to be manufactured, product mix

$\mathcal{S}^p$: total group of shelves of products in the factory

$\mathcal{M}$: total group of raw materials in the warehouses. The total is equal to the sum of raw materials required by all the products at the current planning horizon

$\mathcal{S}^m$: total group of shelves of materials in the factory

$\mathcal{R}$: total group of robots (Mobile manipulators) available at the factory to manufacture (MMRs) or transporting (AGVs)

$\mathcal{B}$: total group of machines available at the factory

$\mathcal{D}$: total group of tools (manufacturing devices) available at the factory

$\mathcal{S}^d$: total group of depots (shelves) of tools in the factory

Set of warehouses

$$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$$

Warehouse vector

$\mathbf{w}_i$ = (shape, location, set of shelves of products, set of shelves of materials)

$\mathbf{w}_i = (\Sigma^w(t), (X^{cent,w}(t), Y^{cent,w}(t)), \mathcal{S}^p, \mathcal{S}^m)$

where:

$i \in \{1, 2, ..., N_{warehouses}\}$

$N_{warehouses} \in \mathbb{N}^+$: total number of warehouses

$\mathcal{S}^p$: total group of products shelves in the factory

$\mathcal{S}^m$: total group of materials shelves in the factory

Variables

$\Sigma^w(t) = \{\sigma_1^w(t), \sigma_2^w(t), ..., \sigma_{N_{warehouses}}^w(t)\}$ Set of shapes from each warehouse

Where each $\sigma(t) \in \mathbb{R}^n \times \mathbb{R}^2$

$(X^{cent,w}(t), Y^{cent,w}(t)) = \{(x_1^{cent,w}(t), y_1^{cent,w}(t)), ...$
$, (x_{N_{warehouses}}^{cent,w}(t), y_{N_{warehouses}}^{cent,w}(t))\} \in \mathbb{R}^2$ Locations of each warehouse

Storing shelves for raw materials and products

Set of material shelves

$\mathcal{S}^m = \{\mathbf{s}_1^m, \mathbf{s}_2^m, ..., \mathbf{s}_{N_{shelves\_material}}^m\}$

where:

$N_{shelves\_material} \in \mathbb{N}^+$: total number of shelves for materials

Material shelf vector

$\mathbf{s}_i^m =$ (type of stored materials, material supplying time, material loading time, material unloading time, maximal storing quota, current stored quota, location, shape, orientation)

$\mathbf{s}_i^m = (m_i^{s\_m}, \xi_i^{s\_m}(r_j)(m_i), \mu_i^{s\_m}(r_j)(m_i), \eta_i^{s\_m}(r_j)(m_i), q_i^{s\_m,max}, q_i^{s\_m}(t),$
$(x_i^{cent,s\_m}(t), y_i^{cent,s\_m}(t)), \sigma_i^{s\_m}(t), \alpha_i^{s\_m}(t))$

where:

$i \in \{1, 2, ..., N_{shelves\_material}\}$

Parameters

$m_i^{s\_m} \in \mathcal{M}$: Type of stored materials. For materials e.g. 1 = metal [measuring units], 2 = ceramic [measuring units], 3 = wood [measuring units], 4 = composites [measuring units], 5 = plastic [measuring units])

$\xi_i^{s\_m}(r_j)(m_i) \in \mathbb{R}^+$: Material resupplying time depending on the type of robot r_j and the type of material m_i

$\mu_i^{s\_m}(r_j)(m_i) \in \mathbb{R}^+$: Material loading time depending on the type of robot r_j and the type of material m_i

$\eta_i^{s\_m}(r_j)(m_i) \in \mathbb{R}^+$: Material unloading time depending on the type of robot r_j and the type of material m_i

$q_i^{s\_m,max} \in \mathbb{R}^+$: Maximum stored quantity of material (in material' quantity units)

Variables

$q_i^{s\_m}(t) \in \mathbb{R}^+$: Actual stored quantity of material

$(x_i^{cent,s\_m}(t), y_i^{cent,s\_m}(t)) \in \mathbb{R}^2$: Storing shelf actual location

$\sigma_i^{s\_m}(t) \in \mathbb{R}^n \times \mathbb{R}^2$: Variable shape of storing shelf

$\alpha_i^{s\_m}(t) \in [-180, 180]$: Storing shelf actual orientation

Set of product shelves

$\mathcal{S}^p = \{\mathbf{s}_1^p, \mathbf{s}_2^p, ..., \mathbf{s}_{N_{shelves\_product}}^p\}$

where:

$N_{shelves\_product} \in \mathbb{N}^+$: total number of shelves for product

Product shelf vector

$\mathbf{s}_i^p$ = (type of stored product, product picking up time, product loading time, product unloading time, maximal storing quota, current stored quota, location, shape, orientation)

$\mathbf{s}_i^p = (p_i^{s\_p}, \iota_i^{s\_p}(r_j)(p_i), \mu_i^{s\_p}(r_j)(p_i), \eta_i^{s\_p}(r_j)(p_i), q_i^{s\_p,max}, q_i^{s\_p}(t),$
$(x_i^{cent,s\_p}(t), y_i^{cent,s\_p}(t)), \sigma_i^{s\_p}(t), \alpha_i^{s\_p}(t))$

where:

$i \in \{1, 2, ..., N_{shelves\_product}\}$

Parameters

$p_i^{s\_p} \in \mathcal{P}$: Type of stored product. For products e.g. 1 = piston [measuring units], 2 = valves [measuring units], 3 = gears [measuring units], 4 = crankshaft [measuring units]

$\iota_i^{s\_p}(r_j)(p_i) \in \mathbb{R}^+$: Product picking up time depending on the type of robot r_j and the type of product p_i

Rest of variables $\mu_i^{s\_p}(r_j)(p_i), \eta_i^{s\_p}(r_j)(p_i)$, $q_i^{s\_p,max}$, $q_i^{s\_p}(t)$,
$(x_i^{cent,s\_p}(t), y_i^{cent,s\_p}(t)), \sigma_i^{s\_p}(t)$, $\alpha_i^{s\_p}(t)$ were defined previously

5.2.3 Assumptions on factory design and operation

These characteristics describe the physical design of the factory, zones for warehousing and manufacturing. Factory characteristics includes elements such as the factory and warehouses physical dimensions and the way the factory opetates (e.g. factory frequency to take new orders, frequency to produce orders). These assumptions resulted from the analysis of the constituent problems of the SAR problem in Chapter 4. Assumptions related to the factory design and its operation are summarised in Table 5.1.

Table 5.1: Factory design and operation elements and characteristics represent physical elements and characteristics in the factory, such as the warehouses and the factory itself. They also describe the operation of the factory and warehouses. This table contains a comprehensive group of decision variables and their options within the factory design and operation type. Each decision variable has two options.

Factory design and operation		
Decision variables	Options of variables	
1. Product orders acceptance	Non-periodic	Periodic
2. Layout recalculation	Non-periodic	Periodic
3. Warehouses shape change	Variable	Constant
4. Warehouses location change	Variable	Constant
5. Restrictions on resources	Considered	Neglected
6. Parts or products content in warehouses	Heterogeneous	Homogeneous
7. Parts or products redundancy within shelves	Multiple	Single
8. Transport time of Parts or products to/from shelves	Dynamic	Constant
9. a) Time dependable (e.g. peak hour)	Considered	Neglected
10. b) Location dependable	Considered	Neglected
11. c) Part dependable	Considered	Neglected
12. d) Product dependable	Considered	Neglected
13. e) Robot dependable	Considered	Neglected
14. Part/products to shelves loading/unloading time	Dynamic	Constant
15. a) Time dependable (e.g. peak hour)	Considered	Neglected
16. b) Parts/products dependable	Considered	Neglected
17. c) Robot dependable	Considered	Neglected

5.3 Production requirements

This subsection describes products requirements. Production requirements elements include the entire set of products to be produced in the same period (product mix) and each product requirements. These requirements include demands, deadlines, quality, list of raw materials and the required operations over each raw material. It also include list of precedence constraints among operations for each product.

5.3.1 Definitions

- *Product mix* refers to a group of products to manufacture, see Figure 5.3.

- *Product requirements* deadlines, demands, quality, list of raw materials, precedence of operations to manufacture a product.

- *Product* is a group of raw materials that have been assembled together or a raw material that was machined.

- *Raw materials* include subcomponents for assembly and materials to be machined. This term is abreviate to materials.

- *Operation* is defined as an specific assembly requirement or machining requirement associated with specific raw materials.

- *Precedences* between operations arise from the design of the product. The operations that have no precedences can be done at any order. This provides another degree of flexibility to the INTREPID framework.

- *Precedence graph* is a sequence of operations in which the operations must be performed. Each product might have a different graph.

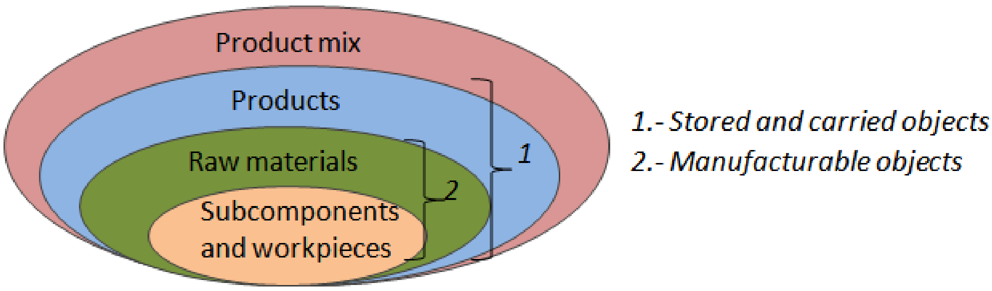


Figure 5.3: Production requirements hierarchy. The hierarchy of the production requirements is product mix, then a single product, which consist of several manufactured raw materials. Raw materials include subcomponents for assembly and materials for machining operations. Raw materials, parts and subcomponents are called manufacturable objects. In contrast, products, raw materials, parts and subcomponents can be stored, carried or transported objects.

5.3.2 Notation

Products set

This subsection states the complete set of products, also known as product mix. This subsection also describes the requirements of a single product (product requirements vector), which refer to required raw materials and their quantities, the group of operations with their respective average processing time and the required quality/accuracy grade (tolerance) and surface quality of each operation

Set of products requirements (i.e. Product mix)

$$\mathcal{P}(t) = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_{N_{products}}\}$$

where:
$N_{products} \in \mathbb{N}^+$ is the total number of products

Product requirements vector
$\mathbf{p}_i$ = (demand, deadline, set of materials and their required quantities by product, set of operations, set of precedences constraints between operations)

$$\mathbf{p}_i = (\omega_i^p, \delta_i^p, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$$

where:
$i \in \{1, 2, ..., N_{products}\}$

Parameters:

$\omega_i^p \in \mathbb{R}^+$: Product demand

$\delta_i^p \in \mathbb{R}^+$: Product deadline

$M_i^p \in \mathcal{M}$: Subset of required types of raw materials for product p_i

$Q_i^p \in \mathbb{R}^+$: Subset of required quantities of each type of raw materials for product p

$O_i^p \in \mathcal{O}$: Subset of required operations for product p. This subset is defined ahead

Π_i^p: Set of precedences for product p_i. This set is defined ahead

Subset of required operations for product p_i

This subset refers to the operations for the previously defined subset of raw materials M_i^p for product p_i

$O^{p_i} = \{\mathbf{o}_1^{p_i}, \mathbf{o}_2^{p_i}, ..., \mathbf{o}_{N_{operations\_product\_p_i}}^{p_i}\} \subset \mathcal{O}$

where:

$N_{operations\_product\_p_i} \in \mathbb{N}^+$: total number of operations required for product p_i

Operation vector for product p_i

$\mathbf{o}_j^{p_i}$ = (subset of possible manufacturing processes for an operation, quality grade -tolerance-, surface quality level, set of parts or subcomponents for the operation -raw materials-, average estimated processing time)

$\mathbf{o}_j^{p_i} = (C_j^{o,p_i}, g_j^{o,p_i}, \varpi_j^{o,p_i}, (M_j^{o,p_i}, Q_j^{o,p_i}), \tau_j^{o,p_i}(r_k))$

Where:

$j = \{1, 2, ..., N_{operations\_product\_p_i}\}$

Parameters:

$C_j^{o,p_i} \in \mathcal{C}$: Subset of possible manufacturing processes. Assembly operations required more than one raw material

$g_j^{o,p_i} \in \mathbb{R}^+$: Operation's required quality/accuracy grade (tolerance)

$\varpi_j^{o,p_i} \in \{1, 2, 3, 4, 5\}$: Surface quality level (1 = very high, 2 = high, 3 = medium, 4 = low, 5 = very low)

M_j^{o,p_i}: subset of types of raw materials for assembly or machining operations o_j from product p_i

$$\sum_{j=1}^{N_{operations}} M_j^{o_j,p_i} = M_i^p \tag{5.1}$$

Q_j^{o,p_i}: Subset of required quantities of raw materials by each operation.
Variables

$\tau_j^{o,p_i}(r_k) \in \mathbb{R}^+$: Processing time of operation o_j depending on robot r_k

Set of precedences between operations of product p_i

The set refers to the precedences of each operation of product p_i

$$\Pi_i^p = \pi_1^{o,p_i}, ..., \pi_{N_{operations\_product\_p_i}}^{o,p_i}$$

where:

$N_{operations\_product\_p_i}$: total number of precedences between operations per each operation o_j. Hence, operations and precedences have the same total number Each precedence defines the set of operations that cannot be done before operation o_j as:

$$\pi_j^{o,p_i} = O_i^p \setminus o_j$$

up to

$$\pi_{N_{operations\_product\_p_i}}^{o,p_i} = O_i^p \setminus o_{N_{operations\_product\_p_i}}$$

Where:

$j \in \{1, 2, ..., N_{operations\_product\_p_i}\}$

5.3.3 Assumptions on production requirements

Production requirements characteristics cover the production orders and their production requirements. production requirements refer to the type of product to manufacture and their production requirements (e.g. deadline, demand, quality, production sequences). The entire set of product to produce in the same period is called product mix. These assumptions resulted from the analysis of the constituent problems of the SAR problem in Chapter 4. Assumptions related to the production requirements are summarised in Table 5.2.

5.4 Manufacturing resources

Manufacturing resources cover elements to produce the current production mix and its requirements. Examples of these resources are machines, robots, tools. These resources are classified as active or inactive depending on their availability to perform any tasks. Also, resources are further classified in unavailable or available depending on whether they are busy performing an operation or not respectively.

Table 5.2: Production requirements elements and characteristics refer to requirements specific to a group of products (i.e. product mix). These also describe constraints over operations. This table contains a comprehensive group of decision variables and their options within the production requirements type. Each decision variable has two options.

Production requirements		
Variable name	Options of variables	
18. Product deadlines	Heterogeneous	Homogeneous
19. Product start time	Heterogeneous	Homogeneous
20. Product deadlines	Non-periodic	Periodic
21. Product start time	Non-periodic	Periodic
22. Product release time	Non-periodic	Periodic
23. Product release time	Non-periodic	Periodic
24. Operations preemptions	Considered	Neglected
25. Delay time for auxiliary operations	Considered	Neglected
26. Processing times between operations	Heterogeneous	Homogeneous
27. a) Robot dependable	Considered	Neglected
28. b) Time dependable	Considered	Neglected
29. Products priorities (weights)	Considered	Neglected
30. Products priorities (weights)	Considered	Neglected
Dynamic products priorities	Considered	Neglected
31. Number of process plan	Multiple	Single
32. Time variation on different process plan	Considered	Neglected
33. Cost variation on different process plan	Considered	Neglected
34. Quality variation on different sequence	Considered	Neglected

5.4.1 Definitions

- *Machines* are specialised equipment that produces a single type of operation. These correspond to the processes that cannot be embedded in a compact and interchangeable tool to be used by the robots. Machines can be manually moved with an overhead crane. The process of moving these machines is one at the time, and do not interfere with the movement of robots.

- *Mobile manipulators* are the generic term for robots that can perform manufacturing and logistics operations. Robots consist of an arm robot mounted over a mobile robot. The mobile robot has a platform that can be used to carry raw materials, parts, subcomponents and products, see Figure 5.4.

- *Carrying robot (AGV)* refers for robots that are currently performing transportation operations.

- *Mobile manufacturing robot MMR* refers to robots that are currently performing manufacturing operations.
- *Available robot* denotes a robot that is ready to perform any task (carrying or manufacturing).
- *Unavailable robot* refers to a robot that cannot be used to perform any task, i.e. robots are charging battery or are in corrective or programmed maintenance.
- *Active robot* applies to robots that are transporting or manufacturing.
- *Inactive robot* is used for robots that are not being used neither for carrying nor for manufacturing.
- *Stationary robot* applies to robots that are waiting to start manufacturing or moving. It can also be applied to raw materials or products that are waiting to be processed or moved. This term is used as a planned state for robots and carried raw materials, parts or products. The position should not interfere with the path of MMRs or AGVs.

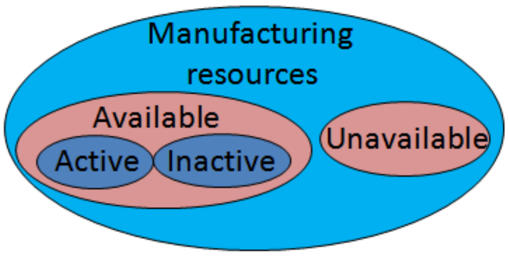


Figure 5.4: Terminology for the manufacturing resources. Resources are machines and mobile robots (i.e. mobile manipulators). However, robots can be employed for manufacturing or transporting operations. When the robots are employed for manufacturing they are called Mobile Manufacturing Robots (MMRs), and for transporting they are called Automated Guided Vehicles (AGVs). Terminology that describe the manufacturing resources are divided in available or unavailable. Available refer to resources that are ready to work (i.e. no scheduled maintenance, their battery is charged, group of tools loaded), whilst unavailable to resources that cannot be used temporally due to maintenance, low battery charge, lack of tools, breakdown. The term available is subdivided in active or inactive. Active refer to resources that are currently in use in the manufacturing zone, whilst inactive to resources that can be used for future plans.

5.4.2 Notation

Robots

A robot is defined by a vector, that groups together parameters and variables.

Set of robots

$\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_{N_{robots}}\}$

where:

$N_{robots} \in \mathbb{N}^+$: total number of robots

Robot vector

$\mathbf{r}_i$ = (wheels, kinematic configuration, shape, mobile positioning accuracy, mobile maximal speed, mobile maximal acceleration, arm positioning accuracy, arm maximal payload, working status, health, battery, carried tools, location, orientation, actual velocity, actual acceleration)

$\mathbf{r}_i = (l_i^r, \kappa_i^r, \sigma_i^r, a_i^r, v_i^{r,max}, \psi_i^{r,max}, a_i^{r,d}, \gamma_i^{r,arm}, \gamma_i^{r,platform}, \beta_i^r, \varsigma_i^r(t), \vartheta_i^r(t), \beta_i^r(t),$
$D_i^r(t), (x_i^{cent,r}(t), y_i^{cent,r}(t)), \alpha_i^r(t), v_i^r(t), \psi_i^r(t))$

where:

$i \in \{1, 2, ..., N_{robots}\}$

Parameters

$l_i^r \in \{1, 2, 3, 4\}$: Wheel configuration (1 = bi-directional (two wheels), 2 = non-holonomic, holonomic -omnidirectional- (3 = mecanum or 4 = Sweden))

$\kappa_i^r \in \{1, 2, 3, 4, 5\}$: Arm kinematic configuration and workspace (1 = cartesian, 2 = cylindrical, 3 = spherical, 4 = scara, 5 = articulated)

$\sigma_i^r \in \mathbb{R}^n \times \mathbb{R}^2$: Constant robot shape

$a_i^r \in \mathbb{R}^+$: Robot positioning accuracy (minimal repeatable accuracy) given by mobile robot

$v_i^{r,max} \in \mathbb{R}^+$: Maximum robot speed

$\psi_i^{r,max} \in \mathbb{R}^+$: Maximum robot acceleration

$a_i^{r,d} \in \mathbb{R}^+$: Tool positioning accuracy (minimal repeatable accuracy) given by arm robot. Assumed constant for the 3 axes

$\gamma_i^{r,arm} \in \mathbb{R}^+$: Maximum arm payload

$\gamma_i^{r,platform} \in \mathbb{R}^+$: Maximum platform payload

$\beta_i^r \in [0, 100]$: Maximal battery duration

Variables

$\varsigma_i^r(t) \in \{1, 2, 3\}$: Working status (1 = working, 2 = available, 3 = unavailable (out of order, charging, scheduled maintenance))

$\vartheta_i^r(t) \in [0, 100]$: Robot health monitoring (lifespan = physical status). This means damage that happens not very often (e.g. wear, tear, robot breakdown)

$\beta_i^r(t) \in [0, 100]$: Actual robot battery

$D_i^r(t) = \{d_1, ..., d_{j_{tools}}\} \subset \mathcal{D}$: Subset of tools carried by each robot. It can change over time. Where: $j_{tools} \in \mathbb{N}^+$ is actual number of tools carried by robot r_i

$(x_i^{cent,r}(t), y_i^{cent,r}(t)) \in \mathbb{R}^2$: Robot actual location

$\alpha_i^r(t) \in [-180, 180]$: Robot actual orientation

$v_i^r(t) \in \mathbb{R}_0^+$: Actual robot speed. Constrained to $v_i^r(t) < v_i^r$

$\psi_i^r(t) \in \mathbb{R}_0^+$: Actual robot acceleration

Machines

A machine is defined by a vector, that groups together parameters and variables.

Set of machines

$$\mathcal{B} = \{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_{N_{machines}}\}$$

where:

$N_{machines} \in \mathbb{N}^+$: Total number of machines

Machine vector

$\mathbf{b}_i$ = (shape, tool positioning accuracy, maximal payload, working status, health, location, orientation)

$$\mathbf{b}_i = (\sigma_i^b, a_i^{b,d}, \gamma_i^b, \varsigma_i^b(t), \vartheta_i^b(t), (x_i^{cent,b}(t), y_i^{cent,b}(t)), \alpha_i^b(t))$$

where:

$i \in \{1, 2, ..., N_{machines}\}$

Tools (Manufacturing devices)

Set of tools, where each tool implies a specific manufacturing process

Set of tools

$$\mathcal{D} = \{\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N_{tools}}\}$$

where:

$N_{tools} \in \mathbb{N}^+$: Total number of tools

Tool vector

$\mathbf{d}_i$ = (manufacturing process, tool (tool bit) accuracy, set-up time, removal time, set of possible tooling materials, surface finishing level, tool condition monitoring)

$$\mathbf{d}_i = (c_i^d, a_i^d, \chi_i^d, \zeta_i^d, M_i^d, \varpi_i^d, \theta_i^d(t))$$

where:

$i \in \{1, 2, ..., N_{tools}\}$

Parameters

$c_i^d \in \mathcal{C}$: Tool type (manufacturing process type) e.g. 1 = Milling, 2 = Polishing, 3 = Cutting, 4 = Grinding, 5 = Boring, 6 = Turning, 7 = Drilling, 8 = Soldering, 9 = Welding, 10 = Brazing, 11 = Fastening (Mechanical assembly) -Threaded fasteners-, 12 = Fastening -Rivets-, 13 = Glueing (adhesive bonding)

$a_i^d \in \mathbb{R}^+$: Accuracy by each tool end effector size e.g. nozzle diameter, drill bit diameter

$\chi_i^d \in \mathbb{R}^+$: Tool set-up time

$\zeta_i^d \in \mathbb{R}^+$: Tool removal time

$M_i^d \in \mathcal{M}$: Feasible tooling' materials (type of materials where the tool can manufacture) e.g. 1 = metal, 2 = ceramic, 3 = wood, 4 = composites, 5 = plastic

$\varpi_i^d \in \mathbb{R}^+$: Surface finishing quality level (1 = very high, 2 = high, 3 = medium, 4 = low, 5 = very low)

Variables

$\theta_i^d(t) \in [0, 100]$: Tool condition monitoring (tool and tool bit lifespan = physical status/conditions). This means damage (e.g. wear, tear, breakdown/breakage). In case of tool bit it may mean the supply of raw material e.g. soldering metal.

Set of tool shelves (depots)

Depots of tools might be located in the warehousing and manufacturing zones.

$\mathcal{B}^d = \{\mathbf{b}_1^d, \mathbf{b}_2^d, ..., \mathbf{b}_{N_{shelves\_tool}}^d\}$

where:

$N_{shelves\_tool} \in \mathbb{N}^+$: Total number of shelves for tool

Tool depot vector

$\mathbf{s}_i^d$ = (current subset of stored tools, raw materials, parts or subcomponents resupplying time, maximal storing quota, current stored quota, location, shape, orientation)

$$\mathbf{s}_i^d = (D^{s\_d}, \xi_i^{s\_d}, q_i^{s\_d,max}, q_i^{s\_d}(t), (x_i^{cent,s\_d}(t), y_i^{cent,s\_d}(t)), \sigma_i^{s\_d}(t), \alpha_i^{s\_d}(t))$$

where:

$i \in \{1, 2, ..., N_{shelves\_tool}\}$

Parameters

$D^{s\_d} \subset \mathcal{D}$: Subset of stored tools

Rest of variables $\xi_i^{s\_d}$, $q_i^{s\_d,max}$, $q_i^{s\_d}(t)$, $(x_i^{cent,s\_d}(t), y_i^{cent,s\_d}(t))$, $\sigma_i^{s\_d}(t)$, $\alpha_i^{s\_d}(t)$ were defined previously

5.4.3 Assumptions on manufacturing resources

The resources characteristics refer to the available elements to manufacture a group of products that is in constant change over time. Examples are robots, machines, tools, area to manufacture. It also includes the way resources operate (e.g. robots maximal speed, robots kinematic configurations). These assumptions resulted from the analysis of the constituent problems of the SAR problem in Chapter 4. Assumptions related to the manufacturing resources are summarised in Table 5.3.

5.5 Complete notation

The last three sections presented notation that represents elements and characteristics that are natural to the SAR problem. Complete notation that represents a generic SAR problem is summarised in this section, as well as a set of assumptions to generate a simplified SAR problem as an example.

SAR problem generic notation

The complete notation consist of the following vectors:

Factory vector

$\mathbf{f} = (z^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{B}, \mathcal{D}, \mathcal{S}^d)$

Table 5.3: Manufacturing resources elements and characteristics describe all the manufacturing resources within a single factory. These are classfied in available, unavailable, active or inactive depending on their actual working condition. Examples of the resources are machines, robots, tools or manufacturing devices. This table contains a comprehensive group of decision variables and their options within the manufacturing resources type. Each decision variable has two options.

Manufacturing resources		
Variable name	Options of variables	
35. Robots health span (maximal)	Heterogeneous	Homogeneous
36. Robots battery span (maximal)	Heterogeneous	Homogeneous
37. Robots kinematic (arm) configuration	Heterogeneous	Homogeneous
38. Maximum carried tools by robots	Heterogeneous	Homogeneous
39. Robots arm payload (maximal)	Heterogeneous	Homogeneous
40. Robots arm positioning accuracy	Heterogeneous	Homogeneous
41. Robots platform payload (maximal)	Heterogeneous	Homogeneous
42. Robots wheels configuration	Heterogeneous	Homogeneous
43. Robots maximal velocity	Heterogeneous	Homogeneous
44. Robots maximal acceleration	Heterogeneous	Homogeneous
45. Machines maximal payload	Heterogeneous	Homogeneous
46. Machines type of operation	Heterogeneous	Homogeneous
47. Machines accuracy	Heterogeneous	Homogeneous
48. Restrictions on machine eligibility	Considered	Neglected
49. Buffers or storage capacity	Considered	Neglected
50. Load splitting	Considered	Neglected
51. Tools life span	Heterogeneous	Homogeneous
52. Tools weight	Heterogeneous	Homogeneous
53. Tools setup time	Heterogeneous	Homogeneous
54. Tools removal time	Heterogeneous	Homogeneous
55. Different tools setup and removel time	Considered	Neglected
56. Tools bit accuracy	Heterogeneous	Homogeneous
57. Tools surface finishing quality level	Heterogeneous	Homogeneous

Warehouse vector

$$\mathbf{w}_i = (\Sigma^w(t), (X^{cent,w}(t), Y^{cent,w}(t)), S^p, S^m)$$

Material shelf vector

$$\mathbf{s}_i^m = (m_i^{s\_m}, \xi_i^{s\_m}(r_j)(m_i), \mu_i^{s\_m}(r_j)(m_i), \eta_i^{s\_m}(r_j)(m_i), q_i^{s\_m,max}, q_i^{s\_m}(t),$$

$(x_i^{cent,s\_m}(t), y_i^{cent,s\_m}(t)), \sigma_i^{s\_m}(t), \alpha_i^{s\_m}(t))$

Product shelf vector

$\mathbf{s}_i^p = (p_i^{s\_p}, \iota_i^{s\_p}(r_j)(p_i), \mu_i^{s\_p}(r_j)(p_i), \eta_i^{s\_p}(r_j)(p_i), q_i^{s\_p,max}, q_i^{s\_p}(t),$
$(x_i^{cent,s\_p}(t), y_i^{cent,s\_p}(t)), \sigma_i^{s\_p}(t), \alpha_i^{s\_p}(t))$

Product requirements vector

$\mathbf{p}_i = (\omega_i^p, \delta_i^p, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$

Operation vector for product p_i

$\mathbf{o}_j^{p_i} = (C_j^{o,p_i}, g_j^{o,p_i}, \varpi_j^{o,p_i}, (M_j^{o,p_i}, Q_j^{o,p_i}), \tau_j^{o,p_i}(r_k))$

Set of precedences between operations of product p_i

$\Pi_i^p = \pi_1^{o,p_i}, ..., \pi_{N_{operations\_product\_p_i}}^{o,p_i}$

Robot vector

$\mathbf{r}_i = (l_i^r, \kappa_i^r, \sigma_i^r, a_i^r, v_i^{r,max}, \psi_i^{r,max}, a_i^{r,d}, \gamma_i^{r,arm}, \gamma_i^{r,platform}, \beta_i^r, \varsigma_i^r(t), \vartheta_i^r(t), \beta_i^r(t),$
$D_i^r(t), (x_i^{cent,r}(t), y_i^{cent,r}(t)), \alpha_i^r(t), v_i^r(t), \psi_i^r(t))$

Machine vector

$\mathbf{b}_i = (\sigma_i^b, a_i^{b,d}, \gamma_i^b, \varsigma_i^b(t), \vartheta_i^b(t), (x_i^{cent,b}(t), y_i^{cent,b}(t)), \alpha_i^b(t))$

Tool vector

$\mathbf{d}_i = (c_i^d, a_i^d, \chi_i^d, \zeta_i^d, M_i^d, \varpi_i^d, \theta_i^d(t))$

Tool depot vector

$\mathbf{s}_i^d = (D^{s\_d}, \xi_i^{s\_d}, q_i^{s\_d,max}, q_i^{s\_d}(t), (x_i^{cent,s\_d}(t), y_i^{cent,s\_d}(t)), \sigma_i^{s\_d}(t), \alpha_i^{s\_d}(t))$

Set of assumptions

These are examples of assumptions from Tables 5.1, 5.2, and 5.3. These assumptions are focused on robots, tools, production requirements and operations.

- Robots:

    1. Robots have the same moving capabilities, this means homogeneous wheel configuration l_i^r, and minimal values of positioning accuracy a_i^r, speed v_i^r and acceleration ψ_i^r. This is represented as $l_i^r = (\mathcal{L}^r - l_i^r)$,

$a_i^r = (a_{\mathcal{R}} - a_i^r)$, $v_i^r = (v_{\mathcal{R}} - v_i^r)$ and $\psi_i^r = (\psi_{\mathcal{R}} - \psi_i^r)$

2. Robots performance capabilities are unlimited, this means battery monitoring ($\vartheta_i^r(t)$) and health monitoring ($\beta_i^r(t)$) are neglected. This is represented as $\cancel{\vartheta_i^r(t)}$, and $\cancel{\beta_i^r(t)}$

3. Robots have the same manufacturing capabilities, this means homogeneous parameters of kinematic configuration κ_i^r, payload γ_i^r, and arm positioning accuracy a_i^{rd}. This is represented as $\kappa_i^r = (\mathcal{K}^r - \kappa_i^r)$, $\gamma_i^r = (\gamma_{\mathcal{R}} - \gamma_i^r)$, and $a_i^{rd} = (a_{\mathcal{R}}^e - a_i^{rd})$

- Tools:

1. Tool setup time and removal are equal for the same tool. This is represented as $\chi_i^d = \zeta_i^d$

2. Tool setup time and removal are equal for all the tools. This is represented as $\chi_{\mathcal{D}} = \zeta_{\mathcal{D}}$

- Production requirements and operations

1. New requests for products arrive at known and constant periods of time. Deadlines of products are at constant and known periods of time. This is represented as $\delta_i^p = (\delta^{\mathcal{P}} - \delta_i^p)$

2. Same type of operation have the same processing time independently of the robot. This is represented as $\tau_i\cancel{(r_k)}$

3. All operations have equal processing times, tolerance and surface finishing quality independent of the robot and tool. This is represented as $\tau_i = \tau^{\mathcal{O}}, g_i = g^{\mathcal{O}}, \varpi_i = \varpi^{\mathcal{O}}$

SAR problem simplification

Simplifications to the notation (i.e. vectors) due to the assumptions are represented by crossing the related variable or parameter as follows:

- Robot vector

  $\mathbf{r}_i = \{\cancel{l_i^r}, \cancel{\kappa_i^r}, \cancel{\sigma_i^r}, \cancel{\alpha_i^r}, \cancel{v_i^{r,max}}, \cancel{\psi_i^{r,max}}, \cancel{a_i^{r,d}}, \cancel{\gamma_i^{r,arm}}, \cancel{\gamma_i^{r,platform}}, \beta_i^r, \varsigma_i^r(t), \cancel{\vartheta_i^r(t)}, \cancel{\beta_i^r(t)}, D_i^r(t), (x_i^{cent,r}(t), y_i^{cent,r}(t)), \alpha_i^r(t), v_i^r(t), \psi_i^r(t))$

- Tool vector

  $\mathbf{d}_i = (c_i^d, a_i^d, \cancel{\chi_i^d}, \cancel{\zeta_i^d}, M_i^d, \varpi_i^d, \theta_i^d(t))$

- Operation vector

  $\mathbf{o}_j^{p_i} = (C_j^{o_j,p_i}, \cancel{g_j^{p_i}}, \cancel{\varpi_j^{p_i}}, (M_j^{o,p_i}, Q_j^{o,p_i}), \cancel{\tau_j^{p_i}})$

Simplified SAR problem

The simplified notation vectors are:

- Factory vector

  $\mathbf{f} = (z^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{B}, \mathcal{D}, \mathcal{S}^d)$

- Warehouse vector

  $\mathbf{w}_i = (\Sigma^w(t), (X^{cent,w}(t), Y^{cent,w}(t)), S^p, S^m)$

- Material shelf vector

  $\mathbf{s}_i^m = (m_i^{s\_m}, \xi_i^{s\_m}(r_j)(m_i), \mu_i^{s\_m}(r_j)(m_i), \eta_i^{s\_m}(r_j)(m_i), q_i^{s\_m,max}, q_i^{s\_m}(t), (x_i^{cent,s\_m}(t), y_i^{cent,s\_m}(t)), \sigma_i^{s\_m}(t), \alpha_i^{s\_m}(t))$

- Product shelf vector

  $\mathbf{s}_i^p = (p_i^{s\_p}, \iota_i^{s\_p}(r_j)(p_i), \mu_i^{s\_p}(r_j)(p_i), \eta_i^{s\_p}(r_j)(p_i), q_i^{s\_p,max}, q_i^{s\_p}(t), (x_i^{cent,s\_p}(t), y_i^{cent,s\_p}(t)), \sigma_i^{s\_p}(t), \alpha_i^{s\_p}(t))$

- Product requirements vector

  $\mathbf{p}_i = (\omega_i^p, \delta_i^p, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$

- Operation vector

  $\mathbf{o}_j^{p_i} = (C_j^{o,p_i}, (M_j^{o,p_i}, Q_j^{o,p_i}))$

- Set of precedences between operations of product p_i

  $\Pi_i^p = \pi_1^{o,p_i}, ..., \pi_{N_{operations\_product\_p_i}}^{o,p_i}$

- Robot vector

  $\mathbf{r}_i = \{\beta_i^r, \varsigma_i^r(t), D_i^r(t), x_i^{cent,r}(t), y_i^{cent,r}(t)), \alpha_i^r(t), v_i^r(t), \psi_i^r(t))$

- Machine vector

  $\mathbf{b}_i = (\sigma_i^b, a_i^{b,d}, \gamma_i^b, \varsigma_i^b(t), \vartheta_i^b(t), (x_i^{cent,b}(t), y_i^{cent,b}(t)), \alpha_i^b(t))$

- Tool vector

  $\mathbf{d}_i = (c_i^d, a_i^d, M_i^d, \varpi_i^d, \theta_i^d(t))$

- Tool depot vector

  $\mathbf{s}_i^d = (D^{s\_d}, \xi_i^{s\_d}, q_i^{s\_d,max}, q_i^{s\_d}(t), (x_i^{cent,s\_d}(t), y_i^{cent,s\_d}(t)), \sigma_i^{s\_d}(t), \alpha_i^{s\_d}(t))$

The assumptions might affect elements and characteristics of the SAR problem. The notation presented in the last three sections represents elements and characteristics that are natural to the SAR problem. This notation helps formulating the SAR problem. However, it is of paramount importance to select a combination of elements, characteristics and their assumptions that result in a tractable problem. This challenge of selecting an appropiate SAR problem is addressed in the next section.

5.6 Problem formulation as a decision making problem

The formulation of the SAR problem is done through the notation presented in the last four subsections. This notation represents elements and characteristics that are natural to the formulation of the SAR problem. The formulation depends on which elements, characteristics are included in the formulation and under which assumptions. Very realistic formulations can be intractable to solve, whilst simple formulations can be solved but do not represent a practical and real problems. Appropiate assumptions can result in simplifications of the SAR problem that are realistic but solvable with optimisation methods. This refers to a balance between fidelity (i.e. realism) of the problem formulation and feasibility to be solved with optimisation methods (i.e. solvability). A simplified (i.e. constrained) problem that can be solved in real-time with well established optimisation methods might be preferable over a highly realistic problem that might be intractable.

Therefore, in order to avoid intractable problem formulations, it is necessary to select a combination of elements, characteristic and their assumptions that result in formulation that is realistic to industrial scenarios but solvable with optimisation methods. In this section, the formulation of the SAR problem as a decision making problem is proposed This refers to the use of decision making methods to select a realistic but solvable SAR problem. Thus, the decision making problem consist of selecting which combination of elements, characteristics and their assumptions to include in the SAR problem formulation. Therefore, the elements, characteristics and assumptions are considered as decision variables on whether to include or not the elements and characteristics and under which assumptions in the formulation. Assumptions, obtained from the analysis in Chapter 4, were presented at the previous sections. These are divided in three types:

- Factory design and its operation, Table 5.1
- Production requirements, Table 5.2
- Manufacturing resources, Table 5.3

These three types of assumptions are natural to the formulation of the SAR problem, and therefore, they are called fundamental variables. Common options for these variables are whether to include specific elements, characteristics and assumptions or not (i.e. neglect them). For example, to consider the time to load and unload parts, subcomponents or products to and from shelves or to neglect it. However, more complex and detailed assumptions might include the specific cost or extra time depending on the specific parts or subcomponents involved in the manufacturing operation. For example, time to load and unload parts, subcomponents or products to and from shelves might depend on the type of parts, subcomponents or products that is loaded and unloaded, the type of robot that loads and unloads the parts, subcomponents or products, or the time when the parts, subcomponents or products is loaded and unloaded (i.e. due to rush hour).

Another type of assumptions, called system variables, facilitate formulating the SAR problem with optimisation methods. Examples of system variables are whether the space and time should be formulated as continuous or discrete, whether the uncertainty of the system should be considered or not, and whether there is a single or multiple objectives to optimise, see Table 5.4. The system variables facilitate the simplication of the SAR problem formulation with optimisation methods. Hence, they are called auxiliary variables.

Table 5.4: System characteristics and assumptions are high level decisions that facilitate formulating the SAR problem as an optimisation problem. Examples of system variables are whether the space and time should be formulated as continuous or discrete, whether the uncertainty of the system should be considered or not, and whether there is a single or multiple objectives to optimise. This table contains a comprehensive group of decision variables and their options within the system type. Each decision variable has two options except for the shape of objects, which has three options.

System variables			
Variable name	Options of variables		
58. Formulation over time	Continuous	Discrete	
59. Formulation over space	Continuous	Discrete	
60. Shape representation	Polygonal	Rectangular	Square
61. Changing shape	Considered	Neglected	
62. Objectives to optimise	Multiple	Single	
63. Uncertainty	Considered	Neglected	
64. Environment knowledge	Unstructured	Structured	
65. General control system	Decentralised	Centralised	
66. Number of periods (time horizon)	Multiple	Single	
67. Events input frequency	Non-periodic	Periodic	
68. Data sampling	Non-uniform	Uniform	

5.6.1 SAR Problem space

There is one ternary variable (three options), and 67 binary variables (two options) presented in Tables 5.1, 5.2, 5.3 and 5.4. When the options of all the variables are exhaustively combined, approximately 4×10^{20} SAR problem variants are generated (see Eq. 5.2). All these variants of the SAR problems are called the SAR problem space. A represention of the combinatorial process can be observed in Fig 5.5.

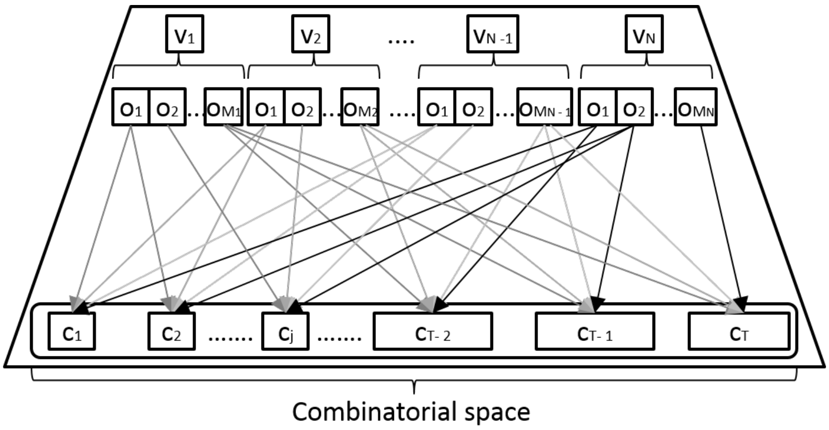


Figure 5.5: Representation of an exhaustive and systematic combination of all the options from all the variables. N variables are represented with the letter V, where each variable has M_i options represented by the letter o options. The combinations (c) of variables' options generate T combinations. The generated combinations with the group of variables and their options is called the combinatorial space. In this case, due to the creation of SAR problems is called the SAR problem space. The number of generated combinations depends on the number of options from each variable. For example, a problem with two binary variables and one ternary can generate twelve problems (i.e. $2*2*3$). When the number of options is equal to all variables, the calculation is reduced to o^N, where N is the total number of variables and o is the total number of options to all variables.

The total number of possible SAR problems (N_{SAR}) depends on the number of variables and their options to combine. An increase in variables or their options results in an increase the in size of the SAR problem space. This number is calculated with the Equation 5.2.

$$N_{SAR} = o_{v_1} * o_{v_2} * ... * o_{v_i} \tag{5.2}$$

where, o_{v_i} are the number of options of variable v_i. If all the variables have the same maximal number of options, Equation 5.2 is simplified to the Equation 5.3.

$$N_{SAR} = o_v^N \tag{5.3}$$

where N is the total number of variables, and o_v is the total number of options to all the variables. The combinations of these assumptions can result in millions of SAR problem variants. Applying Equation 5.2 in the case study, it results in more than 4×10^{20} possible variants of the SAR problem. It is extremely complex to review and analysis the vast number of possible variants. However, it is very important to select a variant SAR problem that is complex enough to be realistic and feasible enough to be solvable. Therefore, it is of paramount importance to effectively explore the problem space in a systematic and analytic way. In the next chapter, Chapter 6, a methodology to deal with this decision making problem is proposed.

5.7 Concluding remarks

A SAR problem notation was presented in this chapter. This one includes features from the scheduling, multi robot task allocation, facilities layout planning (i.e. machine layout), motiong planning and vehicle routing problems. A summary of the elements and characteristics that describe these elements was presented in Tables 5.4, 5.1, 5.2 and 5.3. These ones can be constrained to simplify the SAR problem.

Constraints can vary from neglecting some elements that would make the problem intractable, up to assuming homogeneous values for some of the characteristics. The large number of elements and their characteristics make the problem of formulating a problem very hard. This is because the combination of all these elements and characteristics might results in millions of variants of the SAR problem.

Therefore, in order to formulate the SAR problem, a proposed approach is to review and analyse the relevant problems and their constraints that can be considered in the SAR problem. The decision making problem helps to select a SAR problem that is realistic to industrial scenarios but solvable with optimisation methods.

This approach is a Decision Making Methodology (DMM) that is described in the next chapter, Chapter 6. A software implementation of the DMM is used on the elements and characteristics of Tables 5.4, 5.1, 5.2 and 5.3. The results are variants of the SAR problem that have an adequate balance between realism and feasibility.

Chapter 6

A methodology for problem space exploration and selection

The novel Scheduling, positions Assigning and Routing problem (SAR) problem was proposed in the last two chapters. In brief, the SAR problem refers to the production planning problem with the use of mobile resources (i.e. Self-Reconfigurable Manufacturing System (S-RMS)) within a single factory. The challenge of formulating the SAR problem was introduced in the last chapter (i.e. Section 5.6). Formulating the SAR problem is a complex task due to the large number of elements, characteristics and assumptions that can be included in the formulation. Moreover, solving some of the SAR problem formulations is highly intractable. Therefore, before formulating the SAR problem, it is important to carefully select an appropiate combination(s) of elements, characteristics and their assumptions for formulation. This combination(s) must represent a SAR problem that is realistic to industrial scenarios but solvable with optimisation methods. The problem of selecting an appropiate combination(s) is a decision making problem, where assumptions over elements and characteristics can be considered as decision variables.

The SAR problem formulation involves two types of decision variables. These are fundamental and auxiliary variables. Fundamental variables specify manufacturing resources, production requirements, and factory design and its operation. Auxiliary variables arise from the aim to simplify the formulation of the optimisation problem. The only type of auxiliary variables are called system variables. The types of variables are:

- **Auxiliary**:
  - **System variables**: are variables that help formulating the SAR problem with optimisation methods. These are variables such as whether

uncertainty should be considered during the problem formulation or not, or whether the problem should be formulated in a continuous or discrete space representation.

- **Fundamental**:
  - **Factory design and operation variables**: refer to the design of the factory and the way the factory handles the incoming production requirements. These characteristics describe the physical design of the factory, zones for warehousing and manufacturing and their operation. Also, auxiliary systems such as communications and sensing are included.
  - **Production requirements variables**: are variables that describe general characteristics about production requirements. production requirements refer to the type of products to manufacture and their production requirements (e.g. deadline, demand, quality, production sequences).
  - **Resources variables**: refer to all the resources in the factory, their characteristics and their current status (e.g. available, broken, busy, in maintenance). Examples of these variables are robots, machines and tools, and the way resources operate (e.g. robots maximal speed, robots kinematic configurations).

Current decision making methods for selection rely heavily on pairwise comparisons of alternatives against a reference or against a known ideal and non-ideal alternatives. In case of the SAR problem, there is not a known reference or alternative to compare against and there is a vast number of alternatives. It is challenging and even intractable to apply current methods for selection to a large number of alternatives. Therefore, it is necessary to develop a novel methodology for selection that can be applied to a vast number of alternatives. Thus, a methodology, based on the concept generation and selection methods, is presented in this chapter.

The methodology was implemented and tested with a case study extracted from the elements, characteristics and assumptions proposed in Chapter 5. This chapter is organised in the following subsections. A literature review focused on decision making and problem space exploration is presented in Section 6.1. The proposed methodology, its working principles, and proposed performance metrics are presented in Section 6.2. Conclusions on the methodology and the selected SAR problems are summarised in Section 6.6.

6.1 Literature review on concept generation and selection

The challenge of selecting an appropiate SAR problem can be addressed with conceptual design methods. These methods are widely used in the field of product design and development. Concept generation and selection are steps from the conceptual design process. The aim of the conceptual design is identifying the solution principle [232]. The conceptual design steps are:

1. (Product) requirements list
2. Abstract the requirements to identify the essential problems
3. Establish functional structures (overall function and subfuntions) that address the essential problems
4. Search for working principles that fulfill the subfunctions
5. Combine working principles into working structures
6. Select suitable combinations
7. Firm up into solution principles
8. Evaluate variants against technical and economic criteria
9. Select final solution principles (i.e. concepts)

The concept generation process starts by translating requirements to essential problems through abstraction. Next, functional structures that can address the essential problems are proposed. Next, working principles that can satisfy the functional structures are combined together to generate as many concepts as possible [232].

At the step of concept generation, it is of paramount importance to perform an exhaustive and systematic concept generation process in order to have an exhaustive and comprehensive group of concepts that represent widely all possible solutions [233]. Once, this group is generated, it is necessary to select the best possible combinations of working principles that fullfil the requirements list.

Methods for conceptual design take advantage on the use of abstract representations of concepts instead of detailed designs approaches such as drawings, simulations or mathematical models. Abstract representations are lists of words that describe concepts. The use of abstract representations facilitates comparing dozens of concepts. Abstract representations also facilitate generating new concepts by combining parts or group of parts from concepts of the initial group (i.e. synthesis). The processes of generating, reviewing and selecting concepts are grouped together under the term concept space exploration, whilst for designs is known as design space exploration [234],[235].

Design space exploration approaches make use of software tools, which vary the parameters of designs, drawings and models, to perform an automatic exploration of designs. Design space exploration has been used in the design of circuits, systems and architectures, and concept space exploration for the aircraft concepts [234],[235]. However, modern software tools make use of user-interactive support systems to perform qualitative analysis for designs selection [236],[237]. Similarly, in concept space exploration, experts in product design and development select concepts for the next stages of the design process.

The concept and design spaces belong to the solution space. Concepts and designs focus on satisfying requirements and constraints. Consequently, the solutions (i.e. concepts and designs) for a problem are limited by these requirements and constraints [238]. Requirements are generated from a previous process, the problem formulation process. A correct formulation of an optimisation problem leads to determining feasible or optimal solutions. In contrast, a wrong formulation leads to intractable and ill-defined optimisation problems. The aim of formulating an optimisation problem is to represent this problem as realistic as possible whilst solvable with optimisation methods. This aim must be considered whilst selecting the best possible SAR problem(s) to formulate.

The space of all the possible ways that a problem can be formulated is the problem space. The problem space was distinguished from the solution space through grounded theory development in [239]. A pioneering approach to problem space exploration refers to using different initial conditions for a local search algorithm [240]. The results of this search algorithm were solutions for every problem initial conditions (i.e. problem instance). The problem and solution spaces are part of the process of solving a problem through detailed designs [239]. The problem and solution spaces can be observed in Figure 6.1. Selecting the best possible SAR problem(s) is facilitated with the use of concept space exploration tools such as the abstract representations and the concept generation and selection methods.

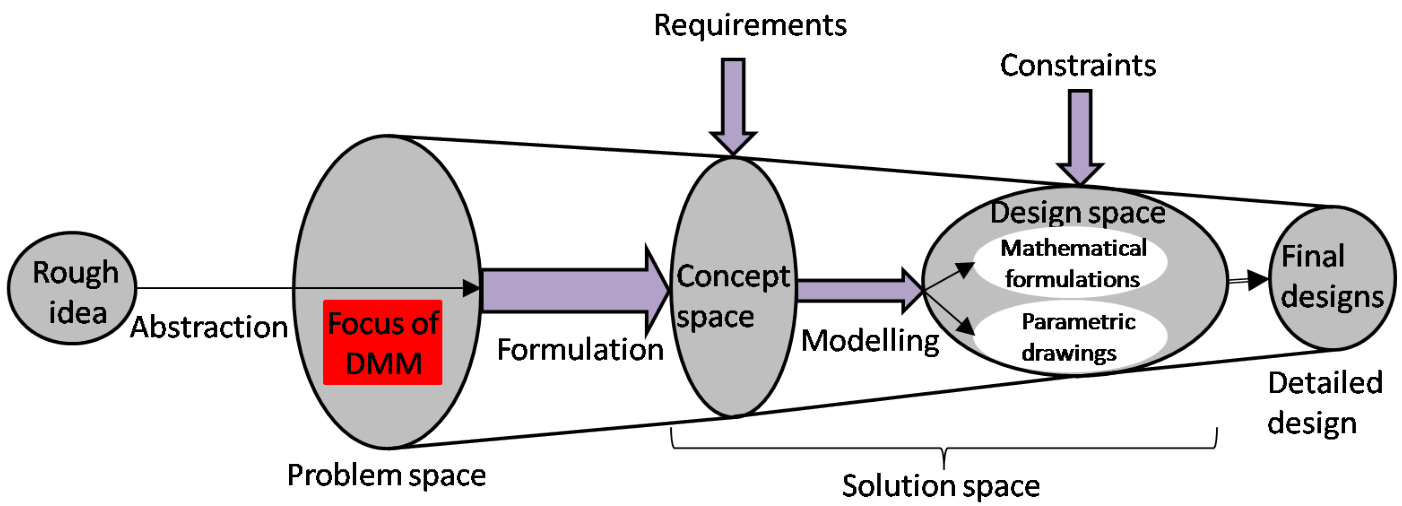


Figure 6.1: Spaces through the product development process. This process start with a rough idea, followed by abstracting the essential problems to solve, generating a problem space of all possible formulations of a problem. Next, generating a concept space by considering requirements of selected problem formulations. Next exploring the design space by considering specific constraints from the selected concepts. The product development process ends with a single or few detailed designs.

Methods for concept generation and selection are divided in decision matrix methods, outranking methods and distance-based methods [22],[23]. Decision matrix methods are the Pugh's evaluation matrix, Quality Function Deployment (QFD) method, the Weighted Rating Method (WRM) and the Analytic Hierarchy Process (AHP) [24],[25]. The most relevant outranking methods are ELimination and (Et) Choice Translating REality (ELECTRE) and Preference Ranking Organization METHod for Enrichment Evaluations (PROMETHEE) [26]. Examples of distance-based methods are Multicriteria Optimisation and Compromise Solution (VIKOR) (for its acronym in Serbian) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [27],[28]. TOPSIS and VIKOR have complex procedures. However, in simple terms, TOPSIS works by minimising the geometric distance between a known ideal concept and maximising the geometric distance between a known non-ideal concept and the alternative concepts [27]. VIKOR works by minimising the Manhattan and Chebyshef distances between a known ideal concept and the alternative concepts [28]. Due to the lack of a known ideal and non-ideal concepts, the distance-based methods are not further studied. The search for a SAR problem that is realistic but solvable requires expert-guided problem space exploration and analysis. Methods that are used for selection and promote concept generation are the Pugh's evaluation matrix, the QFD, and the WRM, whilst methods that only focus on concept selection are AHP, ELECTRE and PROMETHEE.

6.1.1 Decision matrix methods

A decision matrix is a matrix where each alternative is measured against each criterion. Methods based on decision matrix focus on comparing each alternative against a reference alternative referred as datum. The most relevant decision matrix methods are the Quality Function Deployment (QFD) method, the Pugh's evaluation matrix, the Weighted Rating Method (WRM) and the Analytic Hierarchy Process (AHP) [24],[25].

The Pugh's evaluation matrix consists of pairwise comparison of concepts (i.e. alternatives) against a reference concept called datum [22],[24]. The comparison consists of assigning positive or negative signs when the evaluated alternative performs better or worse than the datum respectively. 0's are assigned for an equal performance. The alternative that has more positive and less negative signs than the datum is selected. See Figure 6.2.

An extension of the Pugh's evaluation matrix is to apply this method multiple times through multiple stages [241],[242]. This extended method is called the controlled convergence method. Its working principle consists of applying concept generation and selection processes until there are no new generated concepts. The concept generation process consists of generating new concepts with parts

	Alternatives				
Selection Criteria	A (Datum)	B	C	D	E
Criterion 1	0	+	+	-	-
Criterion 2	0	+	-	-	+
Sum +'s	0	2	1	0	1
Sum 0's	2	0	0	0	0
Sum –'s	0	0	1	2	1
Net score	0	2	0	-2	0
Rank	2	1	2	3	2
Continue?	Combine	Yes	Combine	No	Enhance

Figure 6.2: Example of a Pugh's evaluation matrix inspired from [22]. There are five generic alternatives (i.e. A, B, C, D and E) and two criteria (i.e. criterion 1 and criterion 2). The alternative A serves as a reference called datum. The comparison of alternatives against the datum is with the use of positive signs (+), negative signs (-) and 0's for positive, negative and equal performance respectively. The sum of positive and negative signs, and 0's is shown in the respective rows. The net score results from the difference between the total number of positive signs (+'s) minus the total number of negative signs (-'s). The alternatives with the best net scores are selected, improved or combined. For this example, the best alternative is the B (with a net score of 2, ranking at 1). However, alternatives A,D and E have the same net score (i.e. 0). Hence, these alternatives can be combined or improved. As an example, A and C can be combined, and alternative E can be improved. Alternative D is discarded.

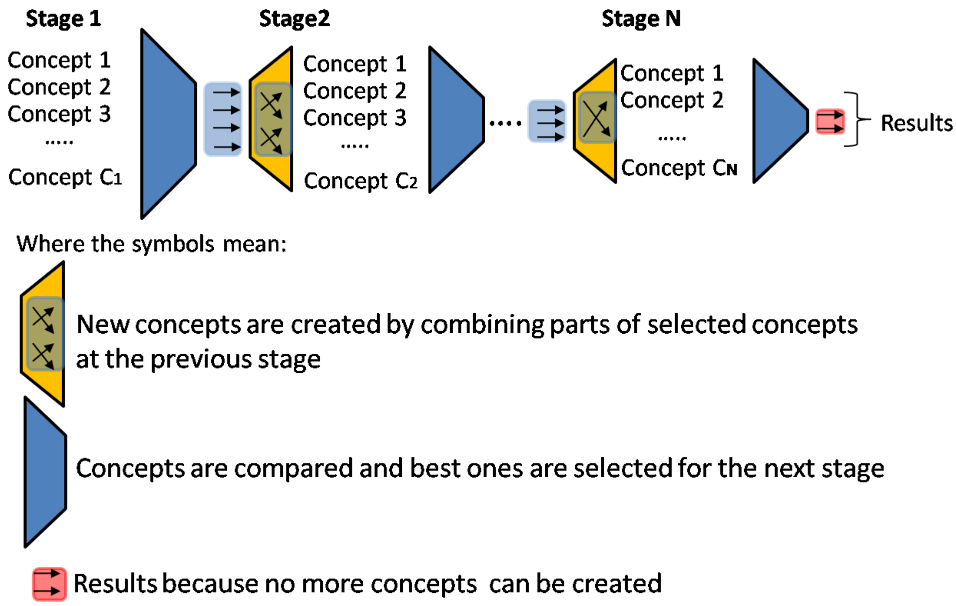


Figure 6.3: Representation of the controlled convergence method applied through N stages for C_1 concepts at the first stage, C_2 concepts for the second stage, and so on, until a final group of C_N concepts at the N stage. The yellow cones represent divergence (i.e. concept generation) processes, whilst the blue cones represent convergence (i.e. concept selection) processes. The arrows within the blue rectangles represent selected concepts at each stage, whilst the arrows within the red rectangles represent final concepts after the controlled convergence is applied.

of selected concepts from the previous stage. A representation of the controlled convergence method can be observed in Figure 6.3.

The process of generating new concepts is called concept generation. The concept selection and concept generation processes are also called concept convergence and concept divergence respectively. The working principle of the controlled convergence method resembles the working principle of the genetic algorithm due to the recombination of successfully selected alternatives through multiple stages [243]. However, the evaluation of new generated concepts or solutions is qualitative (i.e. by expert personnel) rather than quantitative (i.e. by objective function).

The QFD, the WRM and the AHP methods allow assigning priorities to the criteria. As a result of this, alternatives with high performance in very important criteria are selected over alternatives with high performance in less important criteria. The QFD and AHP assign priorities prior to the comparison of alternatives, whilst the WRM assigns the priorities after the comparisons of alternatives. The Pugh' evaluation matrix, QFD and WRM promote the generation of new concepts

		Alternatives					
		B		AC		E+	
Selection criteria	Priority	Weight	Weighted score	Weight	Weighted score	Weight	Weighted score
Criterion 1	60%	2	1.2	3	1.8	1	0.6
Criterion 2	40%	4	1.6	2	0.8	4	1.6
Total score		2.8		2.6		2.2	
Rank		1		2		3	
Develop?		Yes		No		No	

Figure 6.4: Example of the Weighted Rating Method (WRM) inspired from [22]. The WRM incorporates the Pugh's evaluation matrix in the first part of the WRM. For this WRM example, the Pugh's evaluation matrix is used from the Figure 6.2. Representation of the second part of the WRM. The alternatives to evaluate are the combination of alternatives A and C (i.e. AC), the improved version of the alternative E (i.e. E+), and the selected alternative at the first part of the WRM, the alternative B. The criteria to evaluate against consist of criterion 1 and criterion 2. Criteria is assigned a priority value that sum to 100% for all criteria. The alternatives are evaluated against the criteria with weights that range from 1 to 5. This evaluation does not consider the datum. These weights are multiplies by the priority values of the criteria to obtain the weighted score. The weighted scores per each alternative are added to obtain the total score. The alternative with the highest total score is developed in a detailed design. In this example, the successful alternative is the B.

by combining them or improving selected concepts.

The WRM is used in combination with the Pugh's evaluation matrix [22],[24]. The Pugh's evaluation matrix is used first. After this, the most successful alternatives are combined or improved. Later, the criteria are priotised by assigning them a percentage that must sum to 100%. The improved, combined or best alternatives from the last process are compared against the criteria without considering the datum anymore. Weights from 1 to 5 are used in this comparison. These weights are multiplied by the priority of the criteria (in percentanges) and the results of these multiplications are added together for each alternative. The alternative with the highest scored (i.e. multiplied by priority percentage) weight is selected, see Figure 6.4.

In contrast to the WRM, in the QFD method, the criteria is prioritise before the comparison of alternatives [25]. Another key difference with the WRM is that weights can only take values of 1, 3 and 9, where these ones mean a poor, medium and very good, respectively. The absence of weights means that it is not possible to compare alternatives on criteria. Similarly to the Pugh's evaluation matrix, the QFD allows combining alternatives with good performance. For this, the QFD allows measuring how good or bad alternatives can combine with each other. A very strong synergy between two alternatives is expressed with two positive signs (++), a good synergy is expressed with a positive sign (+) and a lack of synergy with a negative sign (-). The lack of signs mean that is not possible to know whether two alternatives have a strong, good or bad synergy. These symbols are introduced at the upper part of the comparison matrix (i.e. in the triangularised matrix). See Figure 6.5.

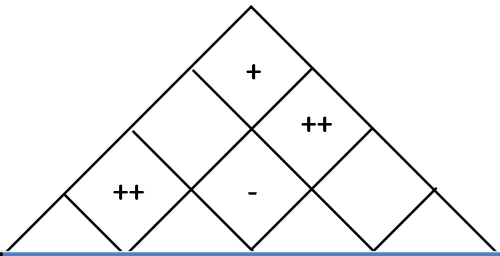
The Pugh's evaluation matrix, the WRM and the QFD promote generating new concepts with the successfully evaluated concepts. For the Pugh's evaluation matrix, the controlled convergence method promotes generating new concepts; for the QFD, a triangularised matrix measures how feasible is to combine concepts; whilst for the WRM, the Pugh's evaluation matrix is used first to evaluate which concepts to combine or improve. In contrast to this, the AHP does not promote generating new concepts.

In the AHP, the prioritization of criteria results from the comparison of each criterion against each other [244]. Once, the priorities for each criteria are known, there are comparisons between each alternative against each other for each criterion. The values for comparison range from 1 to 9. The value of 1 means equal importance between criteria or alternatives. First, pairwise comparisons among criteria are arranged in a square matrix, which eigenvectors are calculated. Next, pairwise comparison among the alternatives is done for each criterion and arranged in a square matrix. The eigenvectors of this matrix are also calculated. Then, the eigenvectors of the alternatives comparison are arranged in a matrix where columns represent criteria and rows alternatives. This matrix is multiplied by the eigenvectors from the criteria matrix. The alternative, resulting from the matrixes multiplication, with the highest total value is selected.

Decision matrix methods work well when it is possible to measure alternatives against criteria with a limited and common range of numerical values [23]. Alternatively to the decision matrix methods, the outranking methods consist of comparing alternatives against a reference in order to obtain a preference ranking of alternatives. The aim is to know the order of the alternatives and which are better, equal or worst than the reference.

6.1.2 Outranking methods for selection

In contrast to the decision matrix methods, outranking methods compare criteria against constituent parts of an alternative (i.e. concept) instead of a complete alternative [26]. These parts are called attributes and each attribute is compared against a single criteria. The most well-known methods are ELimination and



		Alternatives			
		A	B	C	D
Selection criteria	Priority	Weight	Weight	Weight	Weight
Criterion 1	0.2		3	9	1
Criterion 2	0.4	3	9	1	3
Total score		1.2	4.2	2.2	1.4
Continue?		No	Yes	No	No

Figure 6.5: Example of the Quality Function Deployment (QFD). There are four alternatives to compare (i.e. A, B, C and D), whilst two criteria to evaluate against (i.e. criteria 1 and 2). Priority values are assigned to criteria, where these values do not have to sum to 1 (i.e. 100%). Weights evaluate the performance of the alternatives at each criterion. The weights can only take values of 1, 3 and 9. A lack of weight reflects that is not possible to evaluate alternatives againts criteria. The weights of the evaluated alternatives are multiplied per the priority of criteria and added together in order to obtain a total score per each alternative. The alternative with the highest total score is developed. The QFD promotes the combination of alternatives by evaluating which ones have a very strong, good or bad synergy for combination. This evaluation occurs in a triangularised matrix above the alternatives. This matrix relates each alternative with the rest of them. The signs for this evaluation are a double positive sign for a very strong synergy (++), one positive sign for a good synergy (+) and a negative sign for a bad synergy (-).

(Et) Choice Translating REality (ELECTRE) and Preference Ranking Organization METHod for Enrichment Evaluations (PROMETHEE) [245],[246],[247]. In ELECTRE, the comparisons are of the type: an alternative is preferable, incomparable or indifferent to the reference. For an alternative to be as good as the reference the conditions of concordance and discordance must be true. These conditions state that the majority of the evaluated attributes must support that an alternative is as good as the reference (i.e. concordance) and that the rest of the attributes, the minority, are not against that the alternative is as good as the reference in a strong way (i.e. discordance). The discordance condition implies that the minority should be indifferent or incomparable.

Similar to ELECTRE, the PROMETHEE method applies comparisons of the type incomparable, indifferent or preferable [248],[249]. However, PROMETHEE makes use of six types of functions to describe the preference or indifference between alternatives at each criterion. These functions are called preference functions and their values can range from 0 to 1, where 1 is preferable and 0 is indifferent. In PROMETHEE, there are five linear preference functions and one Gaussian preference function. PROMETHEE does not apply the discordance and concordance conditions as ELECTRE does. Instead, each criterion is assigned the most adequate type of function to evaluate all the alternatives to create an evaluation matrix. Later, the evaluation matrix is multiplied by the criteria weights vector to determine the matrix of global preferences. This matrix represents the comparison of each alternative against each alternative for each criterion. In this matrix, the sum of rows represents the good or bad performance of an alternative through all the criteria (i.e. dominance), whilst the sum of the columns represents how much an alternative is dominated by others (i.e. subdominance). A ranking between alternatives is obtained by subtracting the subdominance from the dominance, where the highest positive value is the most preferable value.

6.1.3 Challenges with large number of alternatives

The SAR problem space is of approximately 4×10^{20} problems. Therefore, applying existing selection methods that required pairwise comparisons between problems is intractable. The result, of a comparison of four concept selection methods (i.e. WRM, Pugh's evaluation matrix, AHP and ELECTRE) in eight cases studies, is to use either WRM or Pugh's evaluation matrix due to their simplicity and efficiency [24]. For problems with more concepts or alternatives, a multi-stage controlled convergence is recommended [241]. Therefore, the most relevant method to explore large concept and problem spaces is the controlled convergence method.

Hence, a Decision Making Methodology (DMM) for problem generation and selection was introduced in [30]. The proposed methodology is based on the controlled convergence method. Instead of applying the controlled convergence method over groups of complete concepts, the method is applied over groups of variables that generate partial concepts. This allows generating more complex SAR problems at each stage, and experts select the most adequate based on the qualitative criteria of realism versus feasibility.

This DMM also makes use of abstract representations. The DMM is a hierarchical, multi-stage and interactive. The DMM exploits experts' knowledge to brainstorm decision variables related to the SAR problem. The groups of variables to be analysed are selected by experts at each stage but the most important variables are explored at early stages to create a core SAR problem. The results of the DMM are one or few SAR problems from a very large problem space. The DMM is explained in detail in the next section.

6.2 Methodology

In the last section, the lack of a decision making method that helps selecting concepts from a very large number of concepts was highlighted. The aim is to apply methods for concept generation and selection to select SAR problem(s) for formulation. The selected SAR problem(s) must be realistic to industrial scenarios but solvable with optimisation methods. The proposed Decision Making Methodology (DMM) works by taking advantage of experts' knowledge and experience at many steps of the DMM. These experts are from fields such as factory design, optimisation, robotics and operations research.

In general, the DMM works by collecting decision variables that describe elements, characteristics and their assumptions on how to formulate the SAR problem. Once, a comprehensive group of variables is proposed, the second step of the DMM is to select groups of variables to be analysed through multiple stages (i.e. one group per stage). The process of grouping variables in multiple stages is also performed by experts. In the third step of the DMM, the options of each group of variables are exhaustively and systematically combined together to generate unique SAR problems. In the last step, the SAR problems are analysed and selected by the experts considering the objective of selecting a SAR problem that is realistic but solvable.

The selected SAR problems at each stage become the bases for more complex and more detailed SAR problems at the next stage. This is done through a process called aggregation, which consist of appending the selected SAR problems at the previous stage to the generated SAR problems at the current stage. The SAR problem generation, analysis and selection occurs iteratively until all the groups of variables, from step 2, have been analysed. The most important variables are analysed at the early stages.

The DMM's contribution is to facilitate selecting a few SAR problems from a very large number of SAR problems by dividing the number of SAR problems in multiple stages. See Figure 6.6 for a general representation of the DMM, where the controlled convergence processes (i.e. generation and selection) and the saved number of SAR problems are highlighted.

In this section, the working principles of the DMM are explained in Section 6.3. The DMM is formally described in Section 6.4. An implementation of the DMM is described in Section .

6.3 Working principles

The main working principle of the Decision Making Methodology (DMM) is the controlled convergence method. In brief, the controlled convergence method consists of the application of concept selection (convergence process) and generation (divergence process) steps through multiple stages. The key difference between the classic controlled convergence method and the proposed DMM is its application over partial groups of decision variables through multiple stages. These partial groups of variables are a fraction of the total group of variables. Consequently, these partial groups generate a fraction of the SAR problems compared to the total number that could be generated with the complete group of variables. Hence, the number of analysed SAR problems is the sum of the partial groups of SAR problems generated by the partial groups of variables, instead of the total number of SAR problems that could be generated with the total number of variables In the rest of the document partial groups are called groups.

All options of all variables in each group are combined in systematic and exhaustive way to generate all possible unique combinations of variables' options (i.e. SAR problems). This combination occurs at each stage and adequate combinations of variables' options (i.e. partial SAR problems) are selected at each stage. Later, variables at the current stage are appended or aggregated to the selected partial SAR problems from the previous stage to create more complex and detailed SAR problems at each stage. Variables for the DMM can be collected through brainstorming or a literature review.

In this subsection, the processes of variables collection, combination, aggregation, and analysis are explained. Also, a process that helps reviewing the SAR problem space discarded at previous stages is called SAR problem suggestions. This suggestions process is based on the working principles of recommender systems [250],[251]. Tools that facilitate the DMM processes are also described in this subsection. These tools include a tool that shows the combination and aggregation processes, called combinatorial tree, is described in this subsection. Graphs showing the generated SAR problems are also described.

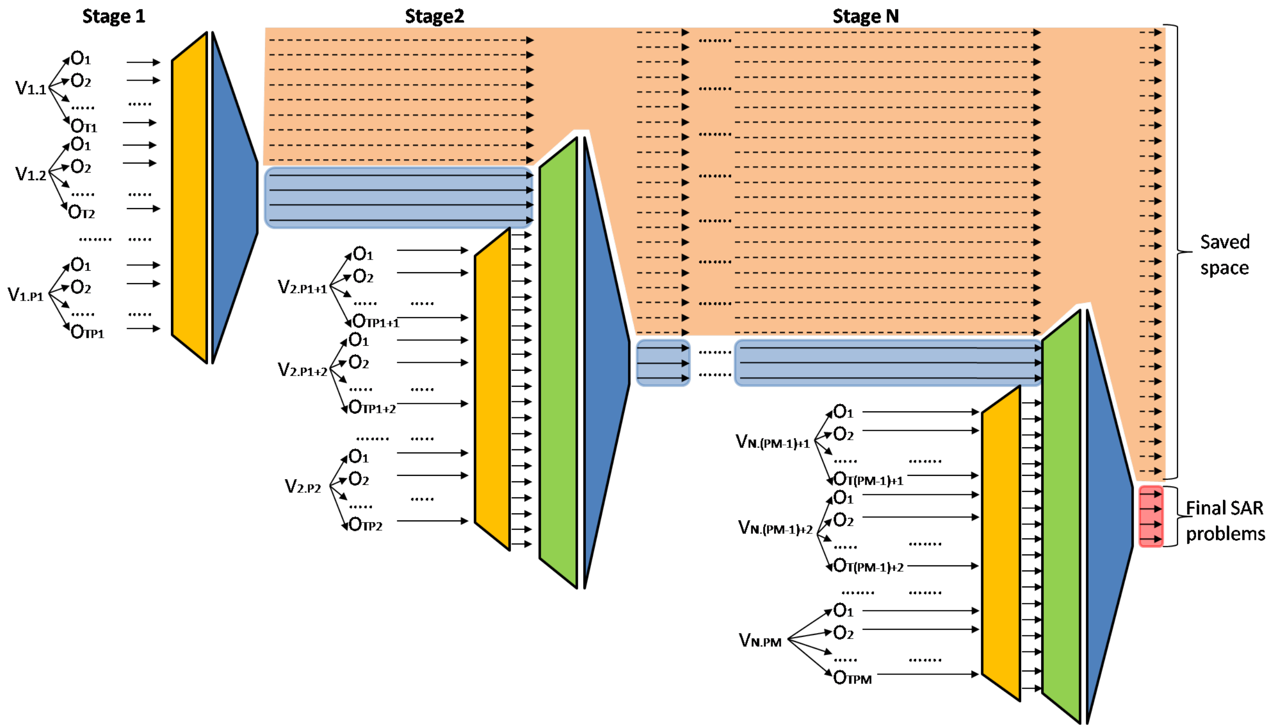


Figure 6.6: Representation of the application of the proposed decision making methodology (DMM) for a number of PM generic variables (V's) through N stages. Each variable can have any number of options (O's). Variables represent decisions on which elements, characteristics and their assumptions should be included in the SAR problem formulation. Yellow cones represent generation steps, where the options of the decision variables are combined together to generate SAR problems. Green cones represent aggregation steps, where the SAR problems selected at the previous stage are aggregated to the generated SAR problems of the current stage. Blue cones represent selection steps, where analysis of the current SAR problem space is performed and a few SAR problems are selected for the next stage. Arrows inside blue rectangles represent selected SAR problems at each stage, whilst arrows inside the red rectangle represent the selected complete SAR problems. Dashed arrows inside the orange region represent the saved SAR problem space.

6.3.1 Variables collection and weighting

Due to the inclusive nature of the SAR problem it is necessary to collect decision variables and their options related to the problems of scheduling, machine and facilities layout and vehicle routing. Two ways are proposed to collect these variables, the first one is through a comprehensive literature review of the related problems, whilst the second is to brainstorm the variables with the help from experts of the fields of optimisation, operations research, robotics and factory design. In the case of this thesis, a comprehensive literature review was presented in Chapter 4. The proposed variables were grouped by their type in system, factory design and operation, production requirements and manufacturing resources variables. The factory design and operation, production requirements and manufacturing resources variables are called fundamental variables because they are natural to the problem they describe (i.e. the SAR problem). The system variables are called auxiliary variables because they facilitate the formulation of the SAR problem with optimisation methods.

After the variables and options collection, options are assigned weights that measures how difficult is to solve problems that include specific options. The range of the weights can be selected according to the user. However, it is recommended to use a scale that allows a high differentiation between the options. The recommended scale for the weights range between the possitive values of 0 and 1000. The use of this scale allows differentiating up to three levels (i.e. units, tenths and hundredths). In order to consider uncertainty of the weighting process, maximal and minimal weights. This weighting process is with the purpose of selecting a SAR problem that is realistic but solvable. In this process the help of experts to assign the weights is of paramount importance.

6.3.2 Combination process

The combinatorial process consists of appending together a single option of each variable from a group of variables, until each possible combination of options is generated. Each combination of options has only one option from each variable, and each combination of options is different from each other. These combinations of options are partial SAR problems, but they are referred simply as SAR problems. The SAR problems selected at the last stage have options from all the variables and these problems are referred as complete SAR problems.

The total number of SAR problems generated from the combination process depends on the number of options of each variable as explained in Subsection 5.6.1. This number can be calculated with Equation 5.2. However, the application of the adapted controlled convergence method reduces this number. The total number of SAR problems generated at each stage with the DMM depends on the two factors.

These factors are the number of selected SAR problems at the previous stage and the multiplication of all the number of options of each variable from the group of variables at the current stage. The total number of SAR problems generated at each stage (N_{SAR}^{stage}) can be calculated with Equation 6.1.

$$N_{SAR}^{stage} = o_{i.1} * o_{i.2} * ... * o_{i.j} * S_{i-1} \tag{6.1}$$

where, o's denotes the maximal number of options in a variable, the first value of the subscript denotes the number of stage (i.e. i) and the second value of the subscript denotes the number of the variable. j is the total number of variables at the stage i. S_{i-1} is number of selected SAR problems at the previous stage.

The total number of SAR problems through all the stages (N_{SAR}^{Stages}) with the DMM depends results from the summation of total SAR problems generated at each stage. This number can be calculated with Equation 6.2.

$$N_{SAR}^{Stages} = (o_{1.1} * .. * o_{1.j_1}) + (o_{2.j_1+1} * .. * o_{2.j_2} * S_1) + .. + (o_{N.j_{N-1}+1} * .. * o_{N.j_N} * S_{N-1}) \tag{6.2}$$

where, o's denotes the maximal number of options in a variable, the first value of the subscript denotes the number of stage (i.e. i) and the second value of the subscript denotes the number of the variable. Equation 6.2 ranges from stage 1 up to stage N. The total number of variables at each stage are j_1, j_2,..., j_N. The number of selected SAR problems are each stage are S_1 up to S_{N-1}.

The combinatorial process is performed in an exhaustive, systematic and hierarchical way. An exhaustive and systematic combination means that all options of all variables are combined together in order to create all the possible combinations of options. A hierarchical combination means the combinatorial process starts with the most important variable' options until the less important variable' options are combined. Once, variables' options are combined together, the maximal and minimal weights of options from each SAR problem are added together. These weights' sums are called maximal and minimal SAR problem weights.

Combinatorial tree

The combinatorial process resembles a decision tree where the options of each variable are hierarchically appended to the options of the next variable [252]. Therefore, the combinatorial process is represeted with a tool called combinatorial tree, see Figure 6.7a. Due to the systematic and hierarchical nature of the combinatorial process, families of problems can be identified by their most hierarchical options (upper in the combinatorial tree).

It is easy to identify preferred options when SAR problems are being generated in the combinatorial process. These preferred options are highlight with arrows of different colours in Figure 6.7a. Also, preferred options are indicated in the relationship and weighting graphs, see Figures 6.7b and 6.8.

Relationship graph

Weights are assigned to each variables' option that measure the influence of each option in the difficulty of solving specific SAR problems. In the combinatorial process, the options' weights from each combination of options (i.e. SAR problems) are added together to obtain SAR problem weights that measure the solvability of the SAR problems. Whenever, there is uncertainty at weighting, a maximal and minimal SAR problem weights can be used. A graph showing the SAR problem weights is called the relationship graph, see Figure 6.7b. This graph shows the SAR problem weights in the hierarchical order on which the options were appended in the combinatorial tree, fig. 6.7a. Preferred options are indicated in this graph with circles of different colours, where each colour represents a different preferred option.

The SAR problem weights are not unique. This means, that many SAR problems can have the same weight due to the combination and addition of variables' options with similar weights. Hence, in the relationship graph, the SAR problems exploration is based on their constituent options rather than on the SAR problem weights Moreover, the relationship graph allows comparing SAR problems by specific preferred combinations of options. These combinations of options can be from the group of variables currently analysed or from the complete group of variables.

Weighting graph

The SAR problem weights measure how difficult is to solve a SAR problem. Multiple SAR problems can have an equal weight because variables' options can have similar weights when they are combined and added together. It is difficul to identify SAR problems by their weights in the relationship graph. Therefore, a graph where SAR problems weights, from the relationship graph, are ordered increasingly by their weights was proposed. This graph was called the weighting graph, see Figure 6.8. In this graph, the SAR problems are assigned a new position that correspond to new increasing order. The SAR problem positions are different from the SAR problem number of the relationship graph.

The weighting graph facilitates comparing SAR problems by their weight rather than by their constituent options. The SAR problems at the farther left are easier to solve, whilst at the farther right are most difficult to solve. The weighting graph shows the SAR problem weights in an increasing order, from the minimal to the maximal weight. Hence, it is possible to locate SAR problems based on

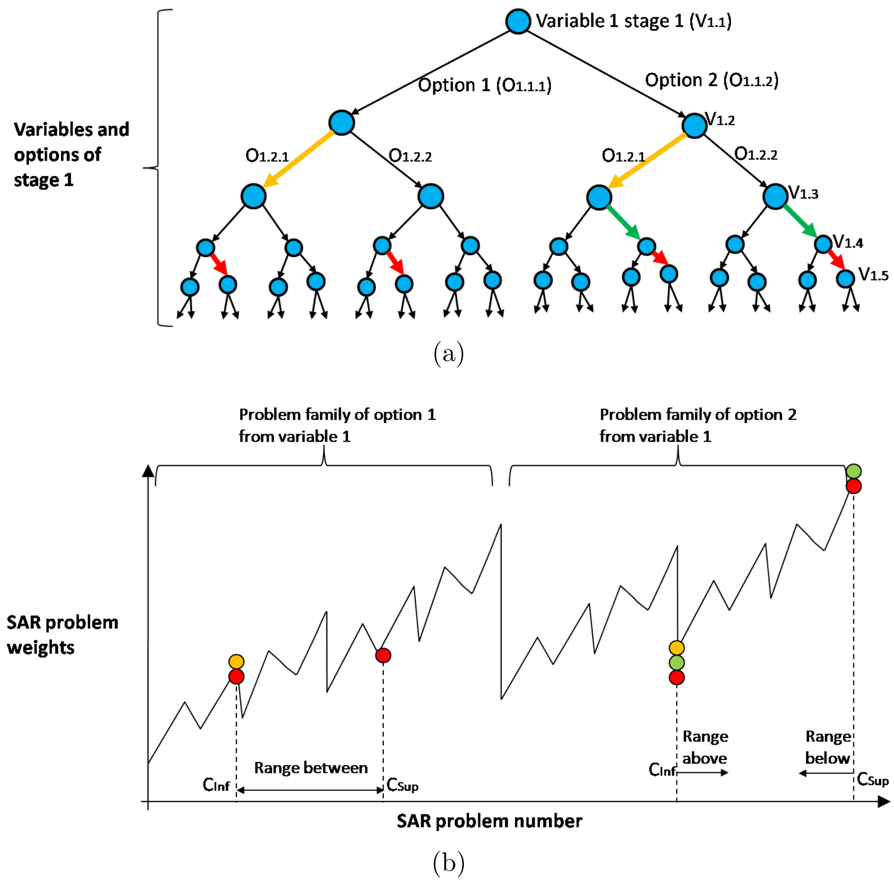


Figure 6.7: Representation of the combinatorial process for one stage with five binary variables. (a) Representation of a combinatorial tree. Blue circles represent variables, arrows represent options and coloured arrows represent preferred options. Variables are denoted with letter V followed by the subscript of the stage number, a dot, and the variable number. Options are denoted with letter O followed by the subscript of the stage number, variable number and option number separated by dots. (b) Representation of a relationship graph. Weights of options are added together to generate SAR problem weights. A single weigth is used in this representation. This graph shows the SAR problems weights as generated during the combinatorial process. The horizontal axis shows the SAR problem number, whilst the vertical axis shows the SAR problems weights. In this graph serves to identify SAR problems that have the same group of upper options (i.e. SAR problem family, e.g. SAR problem families of options 1 and 2 from variable 1). Colored circles highlight SAR problems with preferred options (e.g. orange circles refer to a preferred option for variable 2, green circles for variable 3 and red circles for variable 4). SAR problems can be selected above an inferior indicator (C_{inf}, i.e. inferior combination), below a superior indicator (C_{sup}, i.e. superior combination) or between an inferior and superior indicators (C_{inf} and C_{sup}).

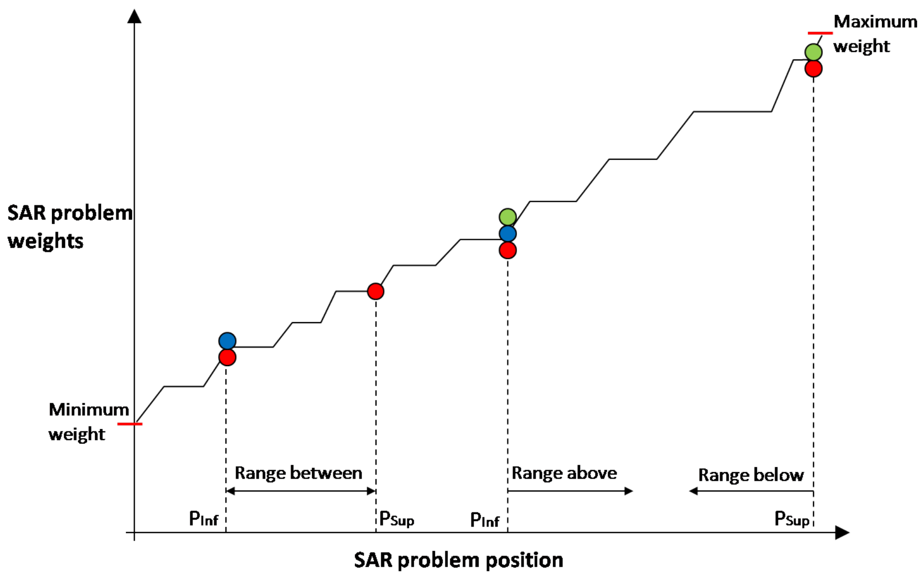


Figure 6.8: Representation of the DMM's weighting graph. The horizontal axis shows the positions of the ordered SAR problem numbers, whilst the vertical axis the weights of the SAR problems. This graph shows the weights of the SAR problems ordered by increasing weights, starting at the farthest left with the minimal weight and ending at the farthest right with the maximal weight. A single weigth is used in this representation. This graph helps comparing SAR problems by their weights instead of the SAR problems constituent options. The easier to solve SAR problems are at the farther left, whilst the most difficult to solve are at the farther right of the graph. SAR problems can be reviewed and compared between two percentages (e.g. an inferior percentage P_{inf} and a superior percentage P_{sup}), within a range above an inferior percentage (P_{inf}) or within a range below a superior percentage (P_{sup}).

specific percentages of SAR problem weights between the minimal and the maximal weights. Preferred options from the relationship graph are indicated in its corresponding SAR problem position in the weighting graph.

6.3.3 Aggregation process

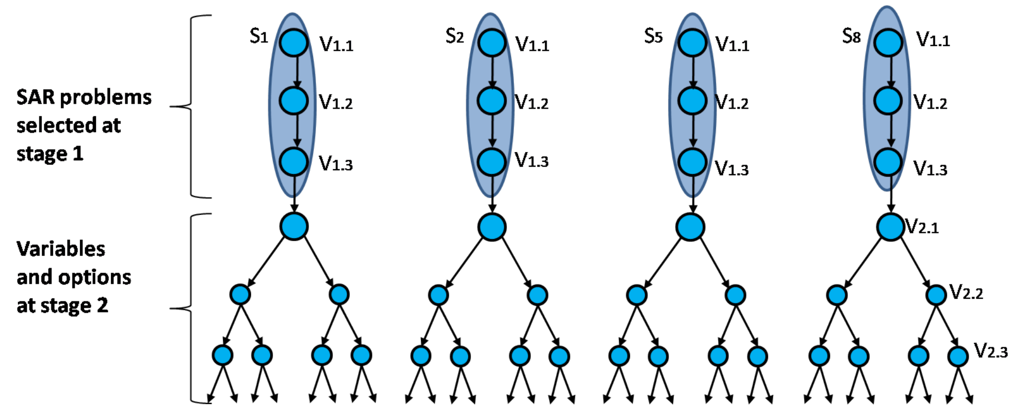
After the first stage, the selected SAR problems at the previous stage are aggregated to the SAR problems generated by combining the group of variables' options from the current stage. This aggregation process results in more complex and detailed SAR problems with options from the current stage and all the previous stages. In this process the variables options are appended to generate more complex and detailed SAR problems. Also, the SAR problem weights from the SAR problems selected at the previous stage are added to the SAR problem weights from the SAR problems generated at the current stage. The aggregation process is represented in Figure 6.9a. The aggregation process also involves adding the weigths of the generated SAR problems in the current stage and the weigths of the selected SAR problems in the previous stage. These added SAR problem weights are represented in a relationship graph, see Figure 6.9b.

6.3.4 Problem space

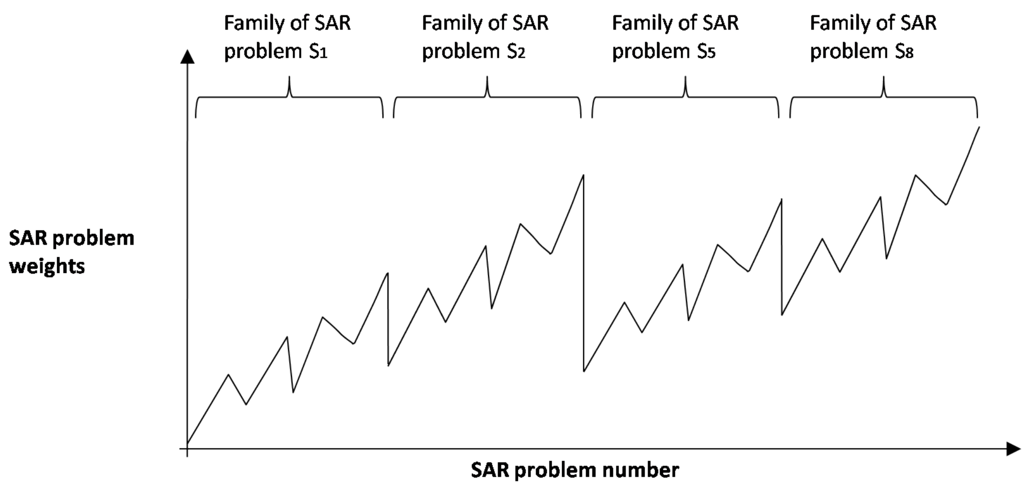
The generated SAR problems at each stage, except for the first stage, are a subtotal of the total number that could be generated. The subtotal SAR problem space corresponds to all the SAR problems generated with the group of variables of the current stage and the selected SAR problems at the previous stage. In the first stage, the subtotal SAR problem space corresponds only to the SAR problems generated with the group of variables of the current stage. The total SAR problem space refers to all SAR problems generated with the accumulated group of variables up to the current stage. The subtotal and the total SAR problem spaces are the same for the first stage, but for any other stage the subtotal and total spaces are different. These spaces can be represented with a graph showing the weights of the SAR problems versus the number of SAR problems with the same weight. This graph is called the problem space graph, see Figure 6.10. This graph is called just space graph through the rest of the document. The number of SAR problems with the same weight is referred as the number of repeated SAR problem weigths, or just a number of SAR problems, within the context of the space graph. Although maximal and minimal weights are allowed, the space graph makes use of the minimal weights.

Problem space evolution

The total and subtotal SAR problem spaces change through the stages. This is due to the addition of options at each stage. As a result, the generated SAR problems have increased weights. For the total problem space, the range of SAR problems weights and the maximal number of SAR problems with the same weight increase at the next stage. However, for the subtotal space there is no clear trend for the changes of the range of SAR problems weights and the maximal number of SAR problems with the same weight. This mean that changes in the range of SAR problems weights and the maximal number of SAR problems with the same weight depend on the number of selected SAR problems at the previous stage and their weights. Total and subtotal space graphs per each stage show the change of the total and subtotal spaces (i.e. problem space evolution). A representation



(a)



(b)

Figure 6.9: Representation of the DMM's aggregation process for two stages. (a) Representation of the combinatorial tree for the aggregation process for stages 1 and 2. The tree shows four selected SAR problems (i.e. S_1, S_2, S_5 and S_8) at the previous stage. The SAR problems from stage 1 has three variables ($V_{1.1}$, $V_{1.2}$ and $V_{1.3}$). Variables are represented by blue circles, options with black arrows and SAR problems are represented by ovals enclosing the variables and options. Combinatorial trees for stage 2 are represented below the selected SAR problems of stage 1. These trees are aggregated to selected SAR problems from stage 1. SAR problem families from the selected SAR problems (i.e. S_1, S_2, S_5 and S_8) are highlighted. In this example, the trees have three variables (i.e. $V_{2.1}$, $V_{2.2}$ and $V_{2.3}$), and two options per each variable. (b) Representation of the relationship graph for the aggregation process of two stages. The vertical axis shows the weight of the SAR problems and the horizontal graph shows the number of the SAR problem. Although maximal and minimal weights are allowed, in this representation a single weigth is used.

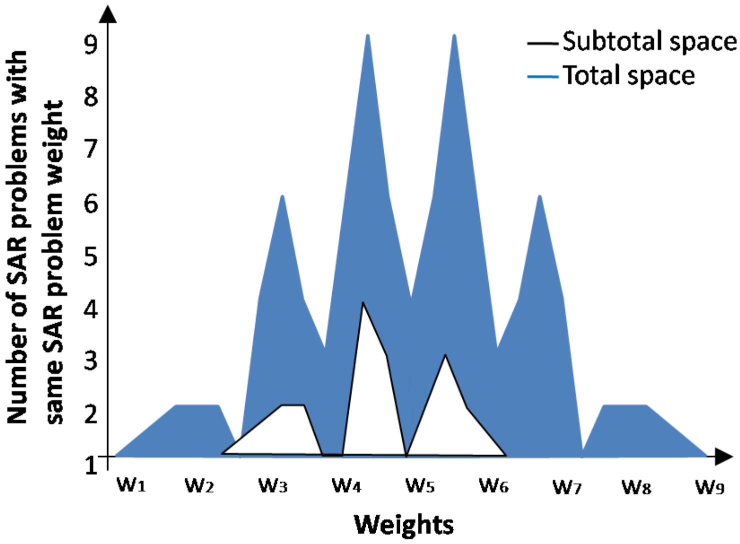


Figure 6.10: Representation of the DMM's problem space graph. The problem space graph shows the total and subtotal SAR problem spaces of one stage. The SAR problems generated with the accumulated group of variables up to the current stage are grouped in the total SAR problem space. Whilst, the SAR problems generated with the group of variables of the current stage and the selected SAR problems at the previous stage are grouped in the subtotal SAR problem space. The subtotal and total SAR problem spaces are equal in the first stage. The blue graph represents the total SAR problem space whilst the white graph the subtotal SAR problem space. The horizontal axis indicates the weights of the SAR problems (i.e. weights w_1, w_2, ..., w_N), whilst the vertical axis indicates the number of combinations (i.e. SAR problems) with the same weight (i.e. 1, 2,..., N). The SAR problems that could be suggested by a recommendation system from the total space are highlighted by the substraction of the blue region minus the white region.

of the space graph showing the evolution of the total space through four stages is shown in Figure 6.11.

Suggested SAR problems

The analysis of SAR problems is enhanced by reviewing and comparing selected SAR problems at the current stage against SAR problems from the total problem space of the current stage. These suggestions are based on recommender systems [250],[251]. The suggestions system is adapted to search for SAR problems from the total SAR problem space that have the same SAR problem weight than the selected SAR problems of the current stage. The aim is to reintegrate SAR problems from the total SAR problem space. A representation of the subtotal and total SAR problem spaces is shown in Figure 6.10. In this figure the difference between the total and subtotal SAR problem spaces is highlighted. The difference between these two spaces is used to search suggestions of SAR problems.

6.3.5 Interactive analysis and selection

The analysis consists of reviewing and comparing SAR problems against each other, i.e. exploring the problem space, in order to determine which SAR problems are selected to the next stage. The analysis is interactive because it is performed by experts with the purpose of selecting SAR problems that are realistic but solvable with optimisation methods. The analysis is performed through the combinatorial tree, and the relationship, weighting and space graphs. In the relationship graph it is possible to review SAR problems by the preferred combination of options, whilst in the weighting graph by a specific percentage or a range of percentages. The SAR problem weights serves as a guide to distinguish how difficult to solve is a problem given the variables' options that contain. The variables' options are the elements, characteristics and their assumptions that can be included in the SAR problem formulation. Whilst the difficulty to solve a SAR problem with optimisation methods is measured by the SAR problem weight, the realism of the SAR problem is identified by experts through the SAR problem space exploration tools. The tools (i.e. combinatorial tree, and the relationship, weighting and space graphs) are used to review the constituent options of each SAR problem and compare them against each other.

The number of generated SAR problems at each stage can be calculated with Equation 6.2. This number depends on two factors. First is the multiplication of all the number of options of each variable from the group of variables at the current stage and con second is the number of selected SAR problems at the previous stage. It would be intractable to explore and analyse a SAR problem space with a large number of SAR problems. Hence, it is important to consider the expected size of

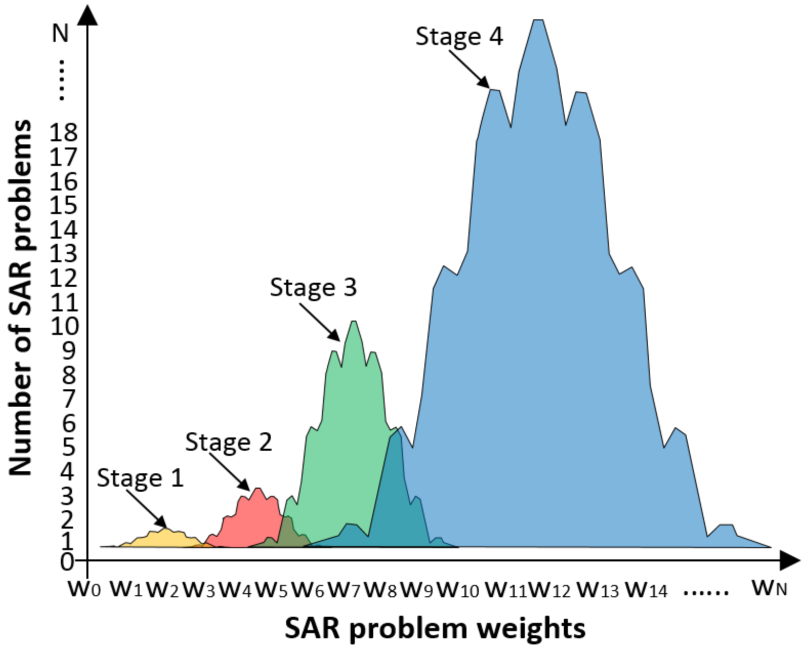


Figure 6.11: Representation of the DMM's problem space evolution. In this example, four total SAR problem spaces corresponding to four stages are represented. The total SAR problem space per stage refers to all the SAR problems that can be generated with the accumulated group of variables up to the current stage. Hence, the total SAR problem space increases in size per each new stage due to the addition of more variables at each new stage. The increments of the total SAR problem space are reflected in the range of SAR problems weights and the maximal number of SAR problems with the same weight. The total space from stage 1 is shown in orange, total space from stage 2 in red, total space from stage 3 in green and total space from stage 4 in blue. The horizontal axis indicates the weights of the SAR problems (i.e. weights W_1, W_2,..., W_N), whilst the vertical axis indicates the number of SAR problems with the same weight (i.e. 1, 2,..., N).

the SAR problem space whilst selecting the variables to analyse at each stage, and the selected SAR problems to succeed at each stage. Based on experience on the DMM application on a SAR problem case study, a feasible space to review is less than 200 SAR problems. However, if the user is willing to spend time exploring a problem space of more than 200 SAR problems, it is possible to select many SAR problems at each stage.

6.4 Methodology algorithm

The working principles described in the last subsection are used in the methodology algorithm. Principles that are executed only once are the data collection (i.e. brainstormed by experts or with literature review). In contrast to this one-off execution, there are principles that are executed in an iterative way. These are the combination, aggregation and analysis principles. In addition, one principle which might or might not be executed in the methodology is the search for suggested SAR problems. Tools that help exploring and analysing the problem space were described also in the subsection. These tools are the combinatorial tree and the relationship, weighting and space graphs. The algorithm of the DMM is shown in Figure 6.12. Formally speaking, the DMM consists of the following steps:

1. **Data collection**. Decision variables and their options related to the SAR problem is brainstormed by experts or collected through a comprehensive literature review. These experts are from the fields of factory design, optimisation, robotics and operations research.

2. **Grouping by their relationship, hierarchisation and weighting**. Variables are ordered by their importance and classified into groups according to the relationship between them. Later, the variables are weighted. Weighting is easier after the hierarchisation because the variables are already arranged in order of importance. The classification in groups facilitates selecting the most important variables of each group to constitute the basic problem at the first stage.

3. **Variable selection for each stage**. Variables are pre-arranged in the stages for analysis. This arrangement can be modified at each stage, prior to the combination of variables options. Changes of the arrangement are influenced by the selected SAR problems at the previous stage. Also, these selected SAR problems might influence the weight of the options. Hence, it is important to consider reweighting after selection. Consequently, the new weights are in agreement with the selected SAR problems.

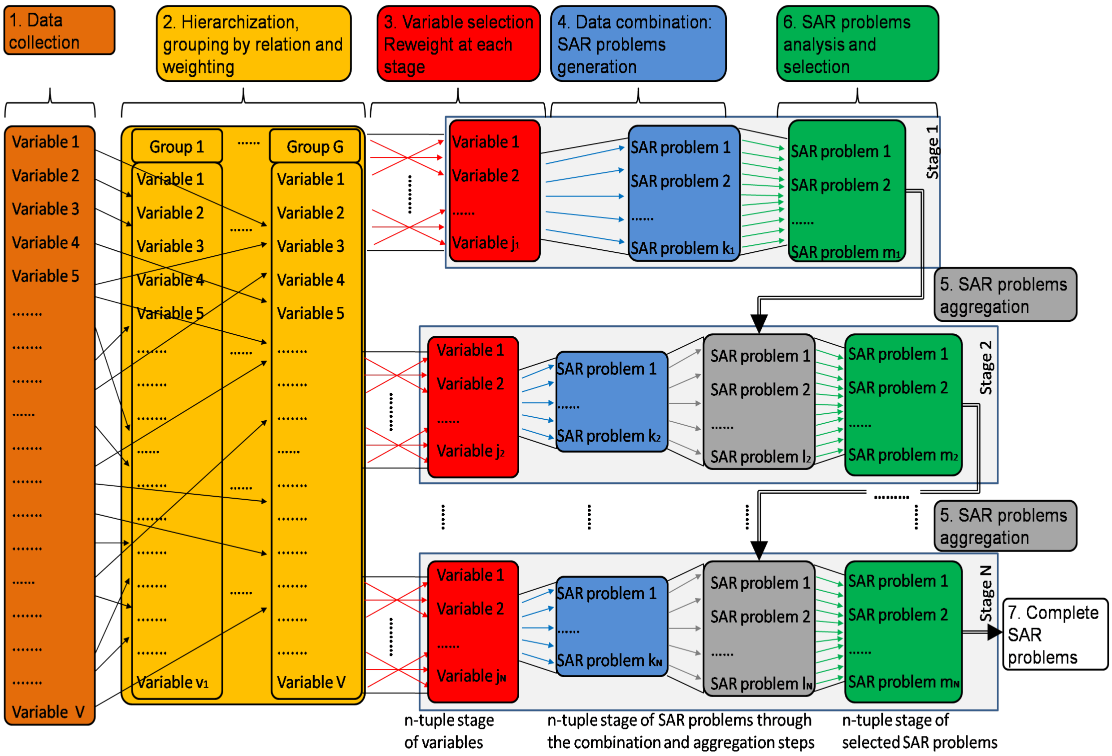


Figure 6.12: Algorithm of the proposed decision making methodology (DMM). The steps of the DMM are: 1. Data collection (in orange rectangles); 2. Hierarchization, grouping by relation and weighting (in yellow rectangles); 3. Variables arrangement in stages (in red rectangles); 4. Data combination (in blue rectangles); 5. Combinations aggregation (in grey rectangles); 6. Analysis and selection (in green rectangles). Steps 3 to 6 are repeated until all the groups of variables have been analysed and the complete SAR problems are selected. The algorithm is represented for V variables in N stages. Variables are arranged in G groups. The variables are rearranged in stages, where stages have at most j_1, j_2 up to j_N variables. The variables' options are combined (i.e. SAR problem generation) in groups that have at most k_1, k_2 up to k_N SAR problems. Selected SAR problems at the previous stage are aggregated to the combined SAR problems of the current stage, after the first stage, on groups that have at most l_1, l_2 up to l_N SAR problems. The groups of combined and aggregated SAR problems are analysed, where these groups have at most m_1, m_2 up to m_N SAR problems. One or few SAR problems are selected at each stage. The selected SAR problems in stage N are the complete SAR problems.

4. **Data combination (combinations generation)**. The options of the current group of variables are combined with each other in a systematic, hierarchical and exhaustive way to generate unique combinations of options. Combinations through the stages are called SAR problems, whilst at the last stage, containing options of all the variables, are called complete SAR problems.

5. **Selected combinations aggregation**. The selected SAR problems at the previous stage are aggregated to the new combinations from step 3. The only omission is at the first stage.

6. **Combinations analysis and selection**. The combinations are reviewed and compared against each other by comparing their specific SAR problem weights in the weighting graph, or the relationship of their options in the relationship graph or both. The selection process should consider the variables from the next stages and the possible generated combinations. Tools for analysis were explained in Section 6.3, but a summary is presented in the next paragraphs.

7. Steps 3 to 6 are repeated until all the variables have been analysed. When all the variables have been analysed, the selected SAR problems at the last stage are called complete SAR problems.

Summary of tools for interactive analysis and selection

The tools for analysis and selection are:

a) **Space graph** (fig. 6.10). Graph showing the total and subtotal SAR problem spaces. The total space is generated with the group of variables at the current stage and the selected SAR problems from previous stages, whilst the subtotal is generated with the group variables at the current stage.

b) **Combinatorial tree** (fig. 6.7). A graph showing the systematic and hierarchical combination process.

c) **Relationship graph** (fig. 6.7). A graph showing the combinations arranged by how they were generated in the combination process. The combinations are not ordered by their added options' weights. Combinations with preferred options are highlighted in this graph.

d) **Weighting graph** (fig. 6.8). A graph showing the combinations weights ordered by the added options' weights. Percentages of the maximal weight and combinations with preferred options are highlighted in this graph.

e) **Suggested SAR problems**. Discarded SAR problems from previous stages are suggested when they match the weights of selected SAR problems.

6.4.1 Metrics of success

In order to assess the performance of the proposed DMM some metrics are proposed. These metrics are mainly based on the relationship between the total and subtotal generated SAR problems, and the selected SAR problems. These metrics are divided in metrics per stage and metrics between the current and next stages or all the stages (i.e. intrastage) as follows:

- **Stage metrics**:

  1. **Saved space per stage**: Percentage between the subtotal generated SAR problems and total number of SAR problems at each stage.
  2. **Most repeated SAR problem weight**: refers to the SAR problem weight that is the most repeated in the space graph of the current stage.
  3. **Range of generated weights**: For the subtotal and total spaces, the range of weights for the generated SAR problems refers to the maximal minus minimal weights of the respective subtotal and total SAR problem spaces.
  4. **Selected SAR problem weights by percentage ranges**: refers to the selected SAR problems in the weighting graph that occur between the following percentage ranges: 0-25%, 25%-50%, 50%-75%, and 75%-100%.

- **Intrastage metrics**:

  1. **Total saved space**: Percentage between the sum of all the generated SAR problems in the stages and the total SAR problems that could be generated.
  2. **Calculation effort waste**: Percentage between the sum of all the generated SAR problems at each stage and the sum of selected SAR problems at each stage.
  3. **Generation effort waste**: Percentage between the selected SAR problems at the last stage (i.e. complete SAR problems) and the sum of generated SAR problems through all the stages.

6.5 Methodology implementation

The DMM can be executed in any platform that can append strings, have basic mathematical tools, and visualisation of numerical graphs. Executing the DMM manually is tedious. Moreover, the fact that other processes (i.e. brainstorming, grouping variables in stages, analysis and selection of SAR problems) require human input makes executing the DMM less ammenable. Therefore, the DMM was implemented as a Decision Making Support System (DMSS) in LabVIEW. Four stages have been implemented in the DMSS, but more can be implemented. However, four stages are enough to evaluate a case study of the SAR problem. The DMSS automates the generation of SAR problems and the creation of visualisation graphs. All the described working principles, graphs and tools for visualisation and the methodology steps were implemented. The DMSS consist of the following windows:

- **Raw data entry window**, Fig. 6.13: is used to provide raw variables and options. The DMM's steps that occur in this window are variables brainstorming, variables hierarchisation and grouping by their relationship and variables' options weighting. This window consist of four subwindows, where variables and options data are provided in sequence. In the first part (i.e. Variables' data), the variables' names, codenames (i.e. short names), number of options of each variable, type of variable and difficulty of the variable are provided. Also, the types of variables are named (e.g. factory design and operation, production requirements). In the second part (i.e. Variables weighting), the options of the variables and their weights are provided. A maximal and a minimal weights are provided. In order to facilitate the weighting process, the variables are shown in decreasing order, from the most difficult type of variables to the less difficult type of variables. In the third part (i.e. variables assignment to stages), the variables are ordered by their type (i.e. the relationship between them, e.g. factory design and operation, production requirements). In this part, the variables are assigned to the stage where they will be analysed. Finally, in the fourth part (i.e. assigned variables), the variables are grouped in the stages they will be analysed and ordered by their type and difficulty.

- **Main window**, Fig. 6.14: is where most of the DMM steps occur, and where most of the graphs for visualisation are included. The main window is divided in two parts, the data selection (i.e. part A) and the SAR problems analysis and selection (i.e. part B). The data selection part is fixed during all the DMSS execution, whilst the SAR problems analysis and selection part changes by each stage through the use of a tab for each stage. The DMM's steps that occur in this window are the variables selection, combination and aggregation (i.e. SAR problems generation), and the analysis and selection processes. Although, the variables are grouped in stages in the raw data entry window, it is possible to change this selection, and it is also possible to change weights of options (if necessary). This window has the following graphs and tools: the combinatorial tree, the relationship, weighting and space graphs. This window also shows the subtotal and total number of SAR problems, and it is possible to visualise the SAR problem that is currently highlighted in either of the relationship and weighting graphs. In the main window, it is possible to select preferred options, and select weights percentages that appear in the relationship and weighting graphs. The combination and aggregation processes occur in the background of the DMSS. However, these processes are activated with the button "RUN" in the main window. Also, these are buttons that activate the raw data entry, comparison and suggestions windows are in the main window. The analysis and selection processes are facilitated with the combinatorial tree, the relationship, weighting and space graphs. The DMSS allows to simultaneously observe the SAR problem that is highlighted in either of the combinatorial tree, the weighting and relationship graphs. Other tools that help the analysis and selection processes are the comparison and suggestions windows and visualising the current SAR problem. Lastly, in the main window, it is possible to save and load variables and options data.

- **Comparison window**, Fig. 6.15: shows the selected SAR problems at the current stage. It also shows the all the subtotal SAR problem space (i.e. generated SAR problems with the group of variables at the current stage).

- **Suggestions window**, Fig. 6.16: is where the suggestions of SAR problems are shown. These suggested SAR problems have the same weight than the selected SAR problems at the current stage.

These four windows facilitate the executing the DMM. Moreover, the windows reduce the number of processes that experts have to do, and allows experts to focus on the processes that require their experience and expertise for evaluation. The operation of the DMSS is as follows: first the raw data entry window is executed only once, then the main window is executed, and this main window is used to activate the comparisons and suggestions windows. The main windows is executed in an iterative process. This process consist of selecting variables and reweighting, if necessary, in the part A and executing the combination, aggregation, analysis and selection processes at the part B. The comparison and suggestions windows can be used as many times as necessary or not used at all. The operation of the DMSS can be observed in Figure 6.17.

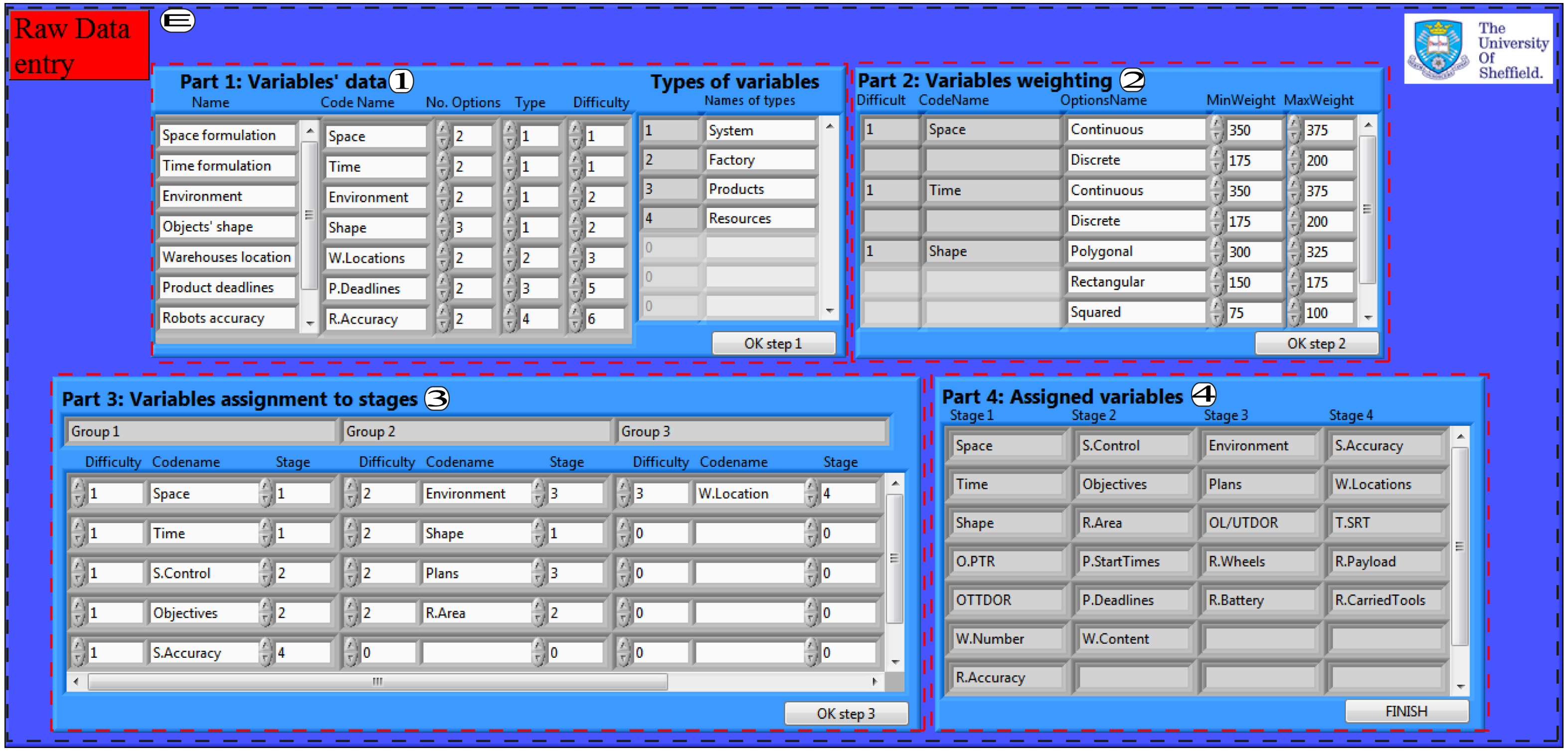


Figure 6.13: Raw data entry window of the DMM implementation. This window is used to provide variables and options data. In this window the variables are brainstormed, hierarchised, grouped by their relationship and weighted. This window has four parts: In part 1 (i.e. Variables' data), the following data about the variables is provided: names, codenames, number of options of each variable, type and difficulty of each variable. Also, the types of variables are named (e.g. production requirements). In part 2 (i.e. Variables weighting), the variables are arranged by their difficulty. Blank spaces allows providing options of variables and weights (maximal and minimal) of the options. In part 3 (i.e. Variables assignment to stages), the variables are grouped by their type. This part allows assigning variables to the stages for analysis. The part 4 (i.e. Assigned variables) shows the variables grouped by the stages where they will be analysed. The parts are executed in sequence, from part 1 up to 4.

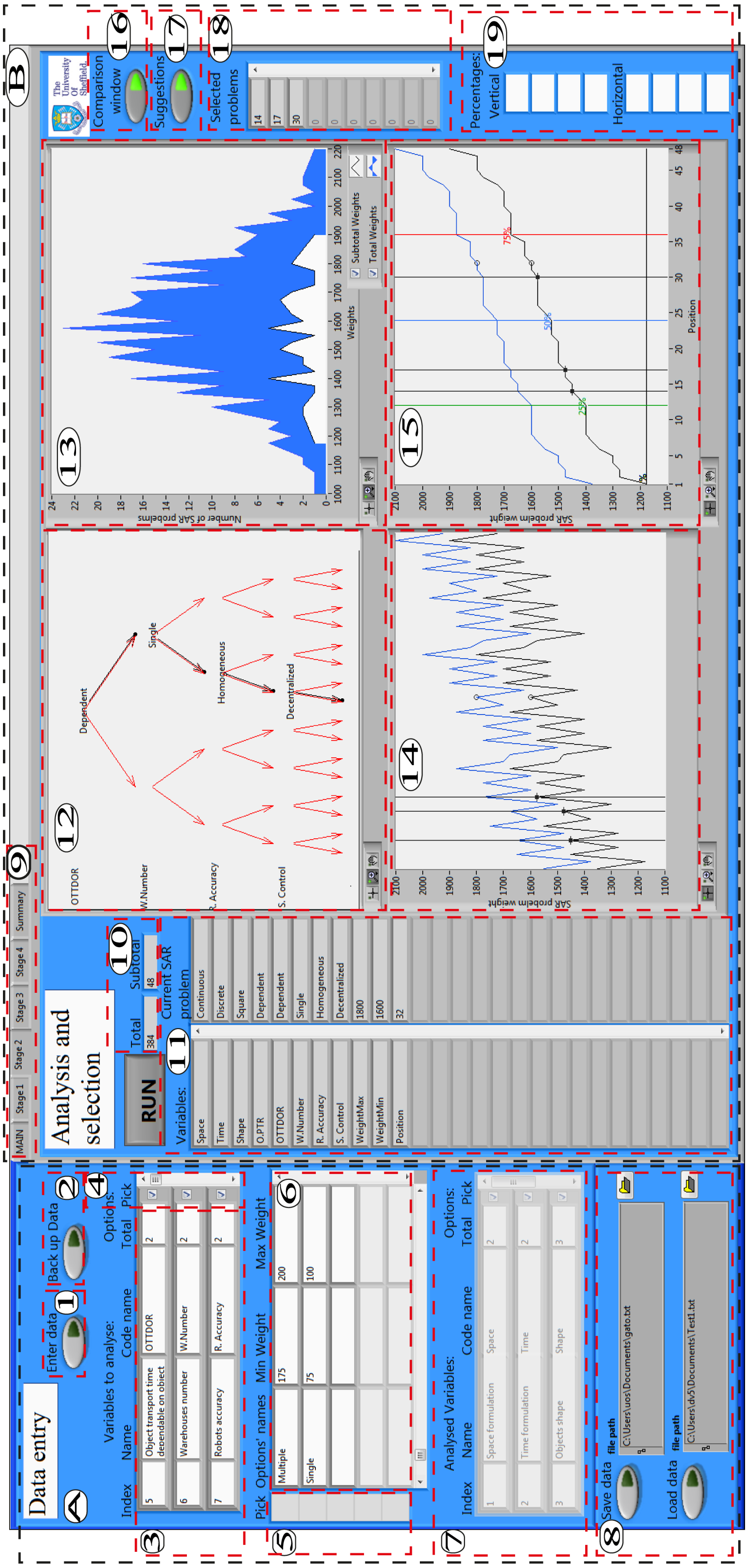


Figure 6.14: Main window of the DMM implementation. There are two main parts, part A for data selection, and part B for variables combination and aggregation, and for SAR problems analysis and selection. The parts are the following. 1: Activate raw data entry window, 2: Back up data, 3: Variables data edition, 4: Variables selection, 5: Preferred options selection, 6: Options edition, 7: Summary of analysed variables, 8: Save and load variables data, 9: Tabs to change through stages, 10: Total and subtotal generated SAR problems, 11: SAR problem highlighted in relationship or weighting graphs, 12: Combinatorial tree, 13: Space graph, 14: Relationship graph, 15: Weithing graph, 16: Activate comparison window, 17: Activate suggestions window, 18: Summary of selected SAR problems at current stage, 19: Weight percentages selection. The RUN button activates the combination and aggregation processes.

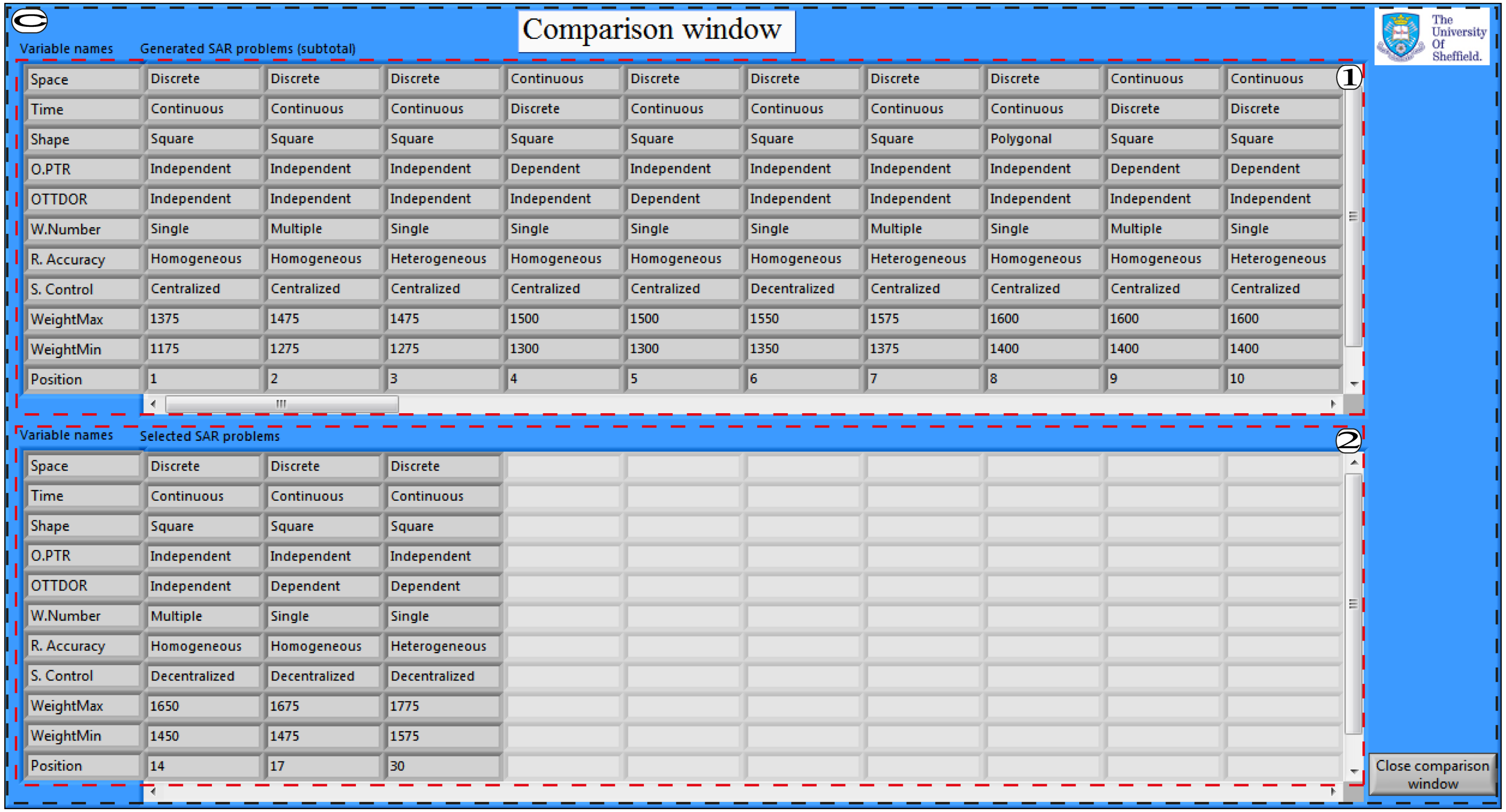


Figure 6.15: Comparison window of the DMM implementation. This window shows the generated SAR problems with the group of variables (i.e. subtotal SAR problem space) and selected SAR problems at the current stage. Part 1 shows the generated SAR problems of the subtotal SAR problem space, whilst Part 2 shows the selected SAR problems of the current stage. The window shows examples of selected and generated SAR problems from the application of the implementation in a SAR problem case study. The SAR problem positions as well as the maximal and minimal SAR problems weights are shown.

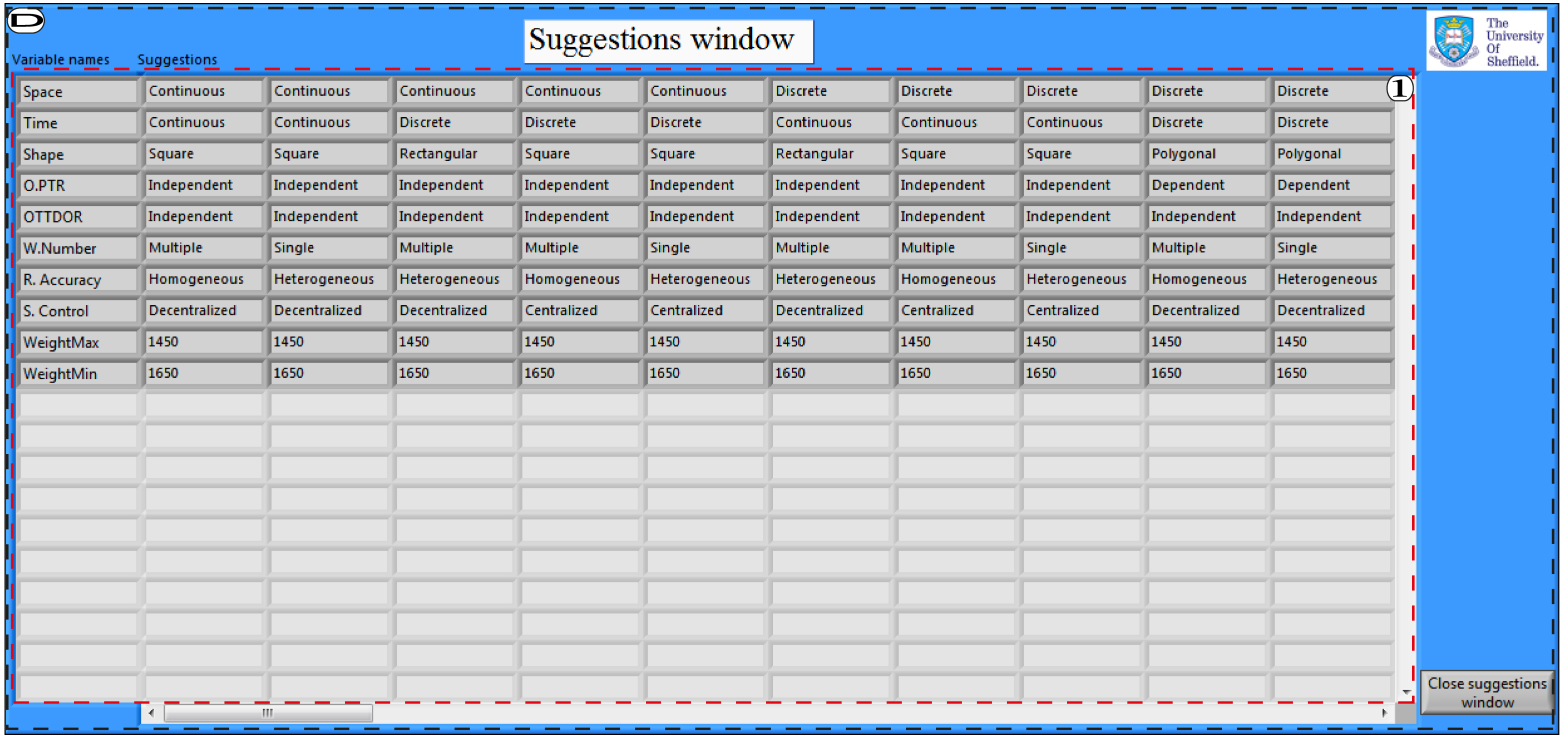


Figure 6.16: Suggestions window of the DMM implementation. This window shows suggestions of SAR problems from the total SAR problem space (Part 1). The suggested SAR problems must have the same weight than the selected SAR problems at the current stage. The window shows examples of suggested SAR problems from the application of the implementation in a SAR problem case study. The maximal and minimal SAR problems weights are shown.

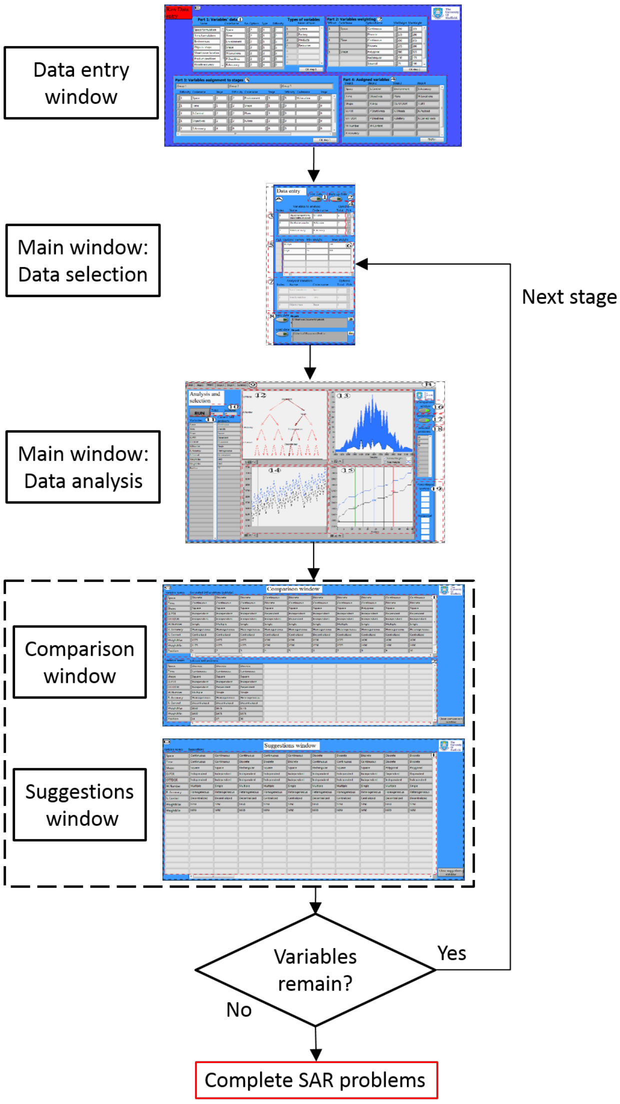


Figure 6.17: Representation of the decision making methodology implementation's operation. The implementation consist of four windows called raw data entry, main, the comparison and suggestions windows. The main window is divided in two parts, the data selection and the data analysis parts. The execution flow of the operation is as follows: first the raw data entry window is executed only once, then the data selection part of the main window is executed, later the data analysis part of the main window is executed. In order to facilitate the analysis, two windows can be activated at any moment or not activated at all. These windows are the comparison and suggestions windows. There windows are sorrounded by a dashed line to represent them as optional. The processes of data selection and analysis are performed in an iterative way until all the variables have been analysed (i.e. multi stage). The SAR problems selected in the final stage are the complete SAR problems that have variables from all the previous stages.

6.6 Concluding remarks

A decision making methodology was presented in this chapter. The aim of the DMM is to select the best possible SAR problem(s) that are realistic but solvable. The problem of selecting one or few SAR problem consist of selecting elements, characteristics and their assumptions that must be included in the SAR problem formulation. The main working principle of the DMM is the controlled convergence, a method from concept generation and selection field. The controlled convergence method was adapted to manage a very large number of variables. Other working principles, graphs and tools that facilitate the selection of one or few SAR problems were explained in detail in this chapter. The main contribution of the proposed DMM is generating and comparing a significant smaller amount of SAR problems (i.e. subtotal space) compared to the total possible number of SAR problems (i.e. total space). The DMM was implemented in LabVIEW and the implementation windows were explained and associated with the corresponding steps of the DMM's algorithm.

Chapter 7

A SAR problem case study

The inclusive nature of the Scheduling, positions Assigning and Routing problem (SAR) problem yields a vast number of elements, characteristics, and their assumptions that can be included in the SAR problem formulation. These elements, characteristics and their assumptions were reviewed and analysed in a comprehensive literature review of the scheduling, machine layout and vehicle routing problems (see Chapter 4). Due to the 68 elements, characteristics, and their assumptions collected through Chapter 4, the SAR problem can be formulated as approximately 4×10^{20} problems (i.e. SAR problem space). As a result is intractable to explore the SAR problem space and select one or few SAR problems to formulate with current selection methods. Therefore, a Decision Making Methodology (DMM), for the purpose of selecting from a large problem space, was presented in the last chapter. The purpose of this DMM is to select a SAR problem that is realistic but solvable with optimisation methods for formulation.

This chapter presents a case study of the SAR problem, from decision variables from Chapter 4, and the DMM is applied on this case study. The SAR problem case study and its assumptions are presented in Section 7.1. The variables hierarchisation and weighting steps of the DMM are presented in Section . The grouping of variables in stages is presented in Section . Results of the case study and their analysis are presented in Section 7.4. The selected SAR problems are represented with the proposed notation and formulated in Section 7.5. Conclusions about the application of the DMM in the SAR problem case study and about the selected SAR problems are summarised in Section 7.6.

7.1 DMM step 1: Data collection

The first step of the Decision Making Methodology (DMM) consist of collecting variables and their options through brainstorming or a literature review. A case study is presented in this section and the DMM is applied on it. The data collection step is the first one in the proposed DMM. In this case, the data collection step is performed through a literature review the problems scheduling, positions assigning (i.e. machine layout) and vehicle routing problems For each of these problems, relevant elements, characteristics, and their assumptions were reviewed and analysed in Chapter 4. These elements, characteristics, and their assumptions are considered as decision variables to include in the SAR problem formulation.

There are four types of decision variables in the SAR problem. These four types are: factory design and operation, manufacturing resources, production requirements and system variables. The factory design and operation, manufacturing resources and production requirements variables are called fundamental variables because their occurrence is natural to the problem they describe. Instead, system variables are called auxiliary variables because they facilitate the formulation of the SAR problem with optimisation methods.

A coherent partial group of variables from Chapter 4 was selected to represent case study of the SAR problem. The variables of the case study are presented in Tables 7.1 and 7.2. Codenames are added to the variables in order to simplify their use through the application of the DMM.

Variables for the case study were selected based on their importance and with the aim of representing the SAR problem. This particular case study focuses on Mobile Manufacturing Robots (MMRs) rather than machines. This means there are variables from the four group of variables described in the last paragraphs, and variables representing the three constituent problems (i.e. scheduling, positions assigning (i.e. machine layout), and routing) of the SAR problem. The following assumptions on the SAR problem case study are applied, whenever possible the assumptions are writen with notation from Chapter 5:

- General: multiple products and robots are considered
- General: products have different demands from each other
- General: tool depots are not considered (i.e. neglected)
- Factory: machines (B) are neglected
- Material shelf: maximal storing quota ($q_i^{s\_m,max}$) and current stored quota ($q_i^{s\_m}$) are neglected
- Product shelf: maximal storing quota ($q_i^{p\_m,max}$) and current stored quota ($q_i^{p\_m}$) are neglected
- Product requirements: products can have different demands (ω_i^p) and products might have multiple operations (O_i^p)

- Operation vector for product: operations are made with a single manufacturing process (c_i^o), operations require a unique quality grade (g_j^o) and a unique surface quality level (ϖ_j^o)

- Robot: Mobile positioning accuracy (a_i^r), arm positioning accuracy ($a_i^{r,d}$), health ($\vartheta_i^r(t)$) are neglected. There is a homogeneous maximal velocity ($\upsilon^{r,max}$), acceleration ($\psi^{r,max}$) and mobile platform payload ($\gamma^{r,platform}$) for all robots.

- Tool: Tool (tool bit) accuracy (a_i^d) and surface finishing level (ϖ_i^d) are neglected. Tool condition monitoring ($\theta_i^d(t)$) is neglected because tools do not wear during operations. The group of tools (D_i^r) that robots carry do not changes over time.

- Objects' shapes: shapes of objects such as warehouses, robots, raw materials, parts and products are constant over time. Thus: $\sigma(t)$ is simplified to σ

Options of all the decision variables are exhaustively and systematically combined to generate unique SAR problems (i.e. each SAR problem has one option of each variable). This results in the SAR problem space (i.e. space of all the SAR problems). Depending on the number of variables and their options, this combinatorial process can result in a huge problem space. The case study has one ternary variable (three options), and twenty two binary variables (two options).

The number of SAR problems increases by increasing the variables or their options. This is calculated with Equation 5.2.

$$o_{v_1} * o_{v_2} * ... * o_{v_i} = 2^{23} * 3^1 = 12,582,912 \tag{7.1}$$

When applying Equation 5.2 in the case study, more than 12×10^6 problems can be generated. It is extremely complex to review and analyse the vast number of possible SAR problems. However, it is very important to select a SAR problem that is complex enough to be realistic and feasible enough to be solvable in order to solve a realistic problem with optimisation methods. Therefore, it is of paramount importance to effectively explore the problems space with the novel DMM. The application of the DMM over the case study is performed in the next sections and the results are presented and discussed in Section 7.4.

7.2 DMM step 2: Variables grouping, hierarchisation and weighting

The second step of the methodology consists of grouping the variables by their relationship, prioritising the variables or group of variables in a hierarchy, and assigning weights to the options of the variables. Variables from the case study were

Table 7.1: System and factory design and operation variables from the SAR problem case study. The selected system variables are: whether space and time can be formulated as discrete or continuous, the control of the system is centralised or decentralised, the number of objectives to optimise is single or multiple, the uncertainty of the system is considered (i.e considered) or neglected (i.e. certain), the factory environment is known (i.e. structured) or not (i.e. unstructured), and whether the objects shape is polygonal, rectangular or square. The selected factory design and operation variables are: whether the parts and products loading and unloading time depends on the type of parts, products and robots or not, the parts and products transport time depends on the type of parts, products and robots or not, the number of warehouses is multiple or single, their content is mixed or unique and whether the location of the warehouses is dynamic or constant.

System variables				
Variable	*Codename*	*Option 1*	*Option 2*	*Option 3*
Space formulation	Space	Continuous	Discrete	
Time formulation	Time	Continuous	Discrete	
System control	S.Control	Decentralised	Centralised	
Objectives to optimise	Objectives	Multiple	Single	
System accuracy	S.Accuracy	Uncertain	Certain	
Environment knowledge	Environment	Unstructured	Structured	
Objects shape	Shape	Polygonal	Rectangular	Square
Factory design and operation variables				
Variable	*Codename*	*Option 1*	*Option 2*	*Option 3*
Parts loading and unloading time dependent on part and robot	PL/UTDPR	Dependent	Independent	
Parts transport time dependable on part and robot	PTTDPR	Dependent	Independent	
Warehouse number	W.Number	Multiple	Single	
Warehouse content	W.Content	Mixed	Unique	
Warehouse location	W.Locations	Dynamic	Constant	

Table 7.2: Production requirements variables and resources variables from the SAR problem case study. The selected production requirements variables are: whether products have multiple or single start times and deadlines, processing time of operations depends on the type of robot, and whether the number of process plans is multiple or single. The selected resources variables focus on Mobile Manufacturing Robots (MMRs) rather than machines. These are: whether robots have an equal or unequal 2D area size, robots manufacturing accuracy is heterogeneous or homogeneous, robots type of wheels is mixed or unique, robots maximal number of carried tools is mixed or unique, robots maximal payload is heterogeneous or homogeneous, robots battery duration is heterogeneous or homogeneous and whether tools setup and removal times are equal or not for all the robots.

Production requirements variables			
Variable	*Codename*	*Option 1*	*Option 2*
Products deadlines	P.Deadlines	Multiple	Single
Products start times	P.StartTimes	Multiple	Single
Operations processing time dependable on robot	O.PTR	Dependent	Independent
Number of process plans	Plans	Multiple	Single
Resources variables			
Variable	*Codename*	*Option 1*	*Option 2*
Robot's 2D area size	R.Area	Unequal	Equal
Robots accuracy	R.Accuracy	Heterogeneous	Homogeneous
Robots type of wheels	R.Wheels	Mixed	Unique
Robots maximal carried tools	R.CarriedTools	Mixed	Unique
Robots maximal payload	R.Payload	Heterogeneous	Homogeneous
Robots battery duration	R.Battery	Heterogeneous	Homogeneous
Tools setup and removal times	T.SRT	Unequal	Equal

grouped by their nature in factory design and operation, manufacturing resources, production requirements and system variables. These variables were classified according to their importance in eight categories. Prior classification of the variables facilitates the hierarchisation and weighting. The weights range from 0 to 1000, where 0 correspond to the easiest to solve and 1000 to the hardest to solve. Moreover, the classification helps managing the different nature of the variables from the scheduling, positions assigning, and routing problems. The eight categories were determined as follows:

1. Variables that affect all the other variables and data.
    - Space formulation
    - Time formulation
    - System control
    - Number of objectives to optimise
    - System accuracy
2. Variables that required special formulations.
    - Environment knowledge
    - Objects shape
    - Number of process plans
    - Robots area
3. Variables that affect production plans given the layout.
    - Warehouses locations
4. Variables where the choice of resources affect the production plan.
    - Parts loading and unloading time dependent on the types of part and robot
    - Parts transport time dependent on the types of part and robot
    - Operations processing time dependent on the type of robot
5. Variables that affect the plan given the product that is manufactured.
    - Products start times
    - Products deadlines

6. Variables that increase the number of options and make it necessary to select a specific resource.
    - Number of warehouses
    - Content products or raw materials at warehouses
    - Robots accuracy
    - Robots type of wheels
7. Variables that differentiate robots and require to solve a trade off between moving a robot or exchanging a tool.
    - Tools setup and removal times
8. Variables that limit the capacity to make an operation.
    - Robots number of carried tools
    - Robots payload
    - Robots maximal battery duration

Variables' options of the eight categories were weighted in the 0-1000 scale with a single value in Tables 7.3 and 7.4. These weights are assigned according to the eight categories. Once the variables are grouped and hierarchised, and the variables' options have weigths, it is necessary to determine on which stage to analyse the variables. This step corresponds to the third step of the DMM and is done in the next section.

7.3 DMM step 3: Variables selection

The third step of the methodology consists of assigning variables to the stages where they can be analysed. The variables were distributed into four stages with the aim to have a division of hard and complex variables, and easy variables (i.e. this is according to their weights). Consequently, there a good mixture of variables with respect to their importance at each stage. Also, there a good mixture of types of variables (i.e. system, factory, requirements and resources).

The variables for stage 1 represent the core problem, whilst the rest of the variables provide more details to the problems. Variables to analyse at each stage are arranged as follows:

- **Stage 1**: space formulation, time formulation, objects' shape, operations processing time dependent on type of robot (O.PTR), parts transportation time depends on type of part and robot (PTTDPR), number of warehouses and the accuracy of robots.

Table 7.3: Part 1 of the variables' options weighting. A single weight is provided, instead of a minimal and maximal (i.e. there is no uncertainty at weigthing). Variables are classified in eight hierarchical categories. The variables' options are weighted according to these eight categories. This table presents the first four categories. These are: variables that affect all the other variables and data with a hard weight of 350 and easy weight of 175; variables that required special formulations with hard weight of 300, medium weight of 150 and easy weight of 75; variables that affect production plans given the layout with hard weight of 275 and easy weight of 125; and variables where the choice of resources affect the production plan with hard weight of 250 and easy weight of 125

Variables that affect all the other variables and data			
	Options and weights		
Variable	*Option 1 (175)*	*Option 2 (350)*	
Space	Discrete	Continuous	
Time	Discrete	Continuous	
S.Control	Centralised	Decentralised	
Objectives	Single	Multiple	
S.Accuracy	Certain	Uncertain	
Variables that required special formulations			
	Options and weights		
Variable	*Option 1 (75)*	*Option 2 (150)*	*Option 3 (300)*
Environment	Structured	Unstructured	
Shape	Square	Rectangular	Polygonal
Plans	Single	Multiple	
R.Area	Equal	Unequal	
Variables that affect production plans given the layout			
	Options and weights		
Variable	*Option 1 (125)*	*Option 2 (275)*	
W.Location	Constant	Dynamic	
Variables where the choice of resources affect the production plan			
	Options and weights		
Variable	*Option 1 (125)*	*Option 2 (250)*	
PL/UTDPR	Independent	Dependent	
PTTDPR	Independent	Dependent	
O.PTR	Independent	Dependent	

Table 7.4: Part 2 of the variables' options weighting. Uncertain at weighting is neglected. Variables are classified in eight hierarchical categories. The variables' options are weighted according to these eight categories. This table presents the remaining four categories. These are: variables that affect the plan given the product that is manufactured with hard weight of 200 and easy weight of 100; variables that increase the number of options and make it necessary to select a specific resource with a hard weight of 175 and easy weight of 75; variables that differentiate robots and require to solve a trade off between moving a robot or exchanging a tool with hard weight of 150 and easy weight of 75; and variables that limit the capacity to make an operation with hard weight of 100 and easy weight of 50.

Variables that affect the plan given the product that is manufactured		
	Options and weights	
Variable	*Option 1 (100)*	*Option 2 (200)*
P.Deadlines	Single	Multiple
P.StartTimes	Single	Multiple
Variables that increase the number of options and make it necessary to select a specific resource		
	Options and weights	
Variable	*Option 1 (75)*	*Option 2 (175)*
W.Number	Single	Multiple
W.Content	Unique	Mixed
R.Accuracy	Homogeneous	Heterogeneous
R.Wheels	Unique	Mixed
Variables that differentiate robots and require to solve a trade off between moving a robot or exchanging a tool		
	Options and weights	
Variable	*Option 1 (75)*	*Option 2 (150)*
T.SRT	Equal	Unequal
Variables that limit the capacity to make an operation		
	Options and weights	
Variable	*Option 1 (50)*	*Option 2 (100)*
R.CarriedTools	Unique	Mixed
R.Payload	Homogeneous	Heterogeneous
R.Battery	Homogeneous	Heterogeneous

- **Stage 2**: system control, number of objectives to optimise, robots' area size, products start times, products deadlines, and content of warehouses.

- **Stage 3**: knowledge about environment, number of process plans, robots' type of wheels, parts loading or unloading time dependent on the type of part and robot (PL/UTDPR) and robots battery duration.

- **Stage 4**: system accuracy, locations of warehouses, equality of tool setup and removal times for all robots, robots' maximal payload, and robots number of carried tools.

Applying Equation 5.2 in the variables' options of each stage, the number of combinations per stage are 192 combinations for stage 1 (i.e. six binary variables and one ternay), 64 combinations for stage 2 (i.e. 6 binary variables), 32 combinations for stage 3 and 4 (i.e. five binary variables). The most important variables are analysed at early stages. This is demonstrated by the sum of the options' weights at each stage. These total sums are for the hardest options: stage 1 with 1,850, stage 2 with 1,575, stage 3 with 1,125, and stage 4 with 975; and for the easiest options: stage 1 with 825, stage 2 with 775, stage 3 with 550, and stage 4 with 475. The average of these sum per the number of variables in each stage is for the hardest options: stage 1 with 264.28, stage 2 with 262.5, stage 3 with 225, and stage 4 with 195; and for the easiest options, stage 1 with 117.85, stage 2 with 129.16, stage 3 with 110, and stage 4 with 95. This data is summarised at Table 7.5.

The total sums and the average sum per the number of variable through the stages show a decreasing rate, demonstrating the importance of the variables at early stages. For the case study, the average weight in the easiest option for stage 2 is bigger than for stage 1. This is because the weight of the option square (i.e. 75) was used in the calculation, instead of the weight of the option rectangular (i.e. 150). Otherwise, the result would have been 128.57, which is closer to 129.16. The assignation of variables at each stage, the sum of variables weights and their average sum per number of variables demonstrate a good balance at assigning the variables in the four stages. Once the variables are assigned to stage for their analysis. The combination and aggregation (i.e. steps 4 and 5 of DMM) of the variables' options generate SAR problems to analyse and select (i.e. step 6 of DMM) at each stage. These three steps are done in the next section.

Table 7.5: The variables selection step results in groups of variables in four stages. These are as follows: stage 1 with variables Space, Time, Shape, O.PTR, PTTDPR, W.Number, and R.Accuracy; stage 2 with variables S.Control, Objectives, R.Area, P.StartTimes, P.Deadlines, and W.Content; stage 3 with variables Environment, Plans, PL/UTDPR, R.Wheels, and R.Battery; and stage 4 with variables S.Accuracy, W.Locations, T.SRT, R.Payload, and R.CarriedTools. The sum of the hard and easy weights of the variables' options and the average sum per the number of variables at each stage shows a decrease indicating the priority given to more important variables at early stages.

	Variable	**Easiest option**	**Hardest option**
Stage 1	Space	175	350
	Time	175	350
	Shape	75	300
	O.PTR	125	250
	PTTDPR	125	250
	W.Number	75	175
	R.Accuracy	75	175
	Sum	**825**	**1,850**
	Average per variable	**117.85**	**264.28**
Stage 2	S.Control	175	350
	Objectives	175	350
	R.Area	150	300
	P.StartTimes	100	200
	P.Deadlines	100	200
	W.Content	75	175
	Sum	**775**	**1,575**
	Average per variable	**129.16**	**262.5**
Stage 3	Environment	150	300
	Plans	150	300
	PL/UTDPR	125	250
	R.Wheels	75	175
	R.Battery	50	100
	Sum	**550**	**1,125**
	Average per variable	**110**	**225**
Stage 4	S.Accuracy	175	350
	W.Locations	125	275
	T.SRT	75	150
	R.Payload	50	100
	R.CarriedTools	50	100
	Sum	**475**	**975**
	Average per variable	**95**	**195**

7.4 DMM steps 4, 5 and 6: Variables' options combination and aggregation, and SAR problems analysis and selection

The fourth and fifth steps of the methodology consist of combining and aggregating variables' options to generate the SAR problems. In the first stage, there is not SAR problems to aggregate, and therefore, only the combination is performed. The sixth step of the methodology consists of analysing and selecting SAR problems to pass to the next stage. The analysis focuses on carefully reviewing SAR problems and comparing them against each other. There are two type of results when applying the DMM. The first type are graphs summarising all the generated SAR problems (from combination and aggregation processes). The second type are a group of selected SAR problems for the next stage (from analysis and selection processes). These selected SAR problems result from the analysis of the summarising graphs.

The variables selected in stage 1 are the core SAR problem that are the basis for most complex SAR problems at the following stages. Therefore, the importance of selecting SAR problems with a variety of options and tractability (i.e. a balance of hard and easy variable's options). These selections should consider variables from the upcoming stages. The summarising graphs and the selected SAR problems per each stage are presented in the following subsections.

7.4.1 Stage 1

The variables from stage 1 represents the core problem. Variables and variables' options, with variables' codenames and options' weigths in brackets, from stage 1 are:

- Space formulation (Space): Discrete (175), Continuous (350)
- Time formulation (Time): Discrete (175), Continuous (350)
- Objects' shape (Shape): Square (75), Rectangular (150), Polygonal (300)
- Operations processing time dependent on type of robot (O.PTR): Independent (125), Dependent (250)
- Parts transportation time depends on type of part and robot (PTTDPR): Independent (125), Dependent (250)
- Number of warehouses: Single (75), Multiple (150)

- Accuracy of robots: Homogeneous (75), Heterogeneous (150)

DMM's step 4: Variables' options combination

At the first stage, variables' options are only combined among themselves. The aggregation process is not performed because there are not SAR problems selected at previous stage (i.e. there is no previous stage). Hence, the SAR problems generated in stage 1 depends only on variables' options. This means, that one ternary variable and six binary variables result in 192 generated SAR problems (see Eq. 5.2). In stage 1, the subtotal and the total SAR problem space are equal. A representation of the combinatorial process for stage 1 can be observed in Figure 7.1.

Graphs generated from the combinatorial process in stage 1 are shown in Figure 7.2. The space graph is shown in Figure 7.2a. The following metrics can be determined from the space graph.

- The number of generated SAR problems is 192
- The range of generated SAR problems weights is from 825 to 1,850
- The most repeated SAR problem weight is 1,225 with 12 SAR problems

The relationship graph is shown in Figure 7.2b. From this figure, it can be observed that there are four main families of SAR problems resulting from the combinations of the variables space and time with the same options for both (i.e. discrete and continuous). The following families can be distinguished:

- Space: discrete. Time: discrete
- Space: discrete. Time: continuous
- Space: continuous. Time: discrete
- Space: continuous. Time: continuous

DMM's steps 6: SAR problems analysis and selection

In stage 1, fourteen SAR problems were selected. The selected SAR problems are summarised in Tables A.1 and A.2 from the Appendix A. These ones have a good variety and balance of hard and easy variables' options. The reasons to select SAR problems are threefold. Firstly, feasibility of the problems to be solved; secondly, the realism of the problems to industrial scenarios; and thirdly, the interest of combining problems with variables of the next stage. The reasons for selecting these problems are classified as follows (problem positions according to the weighting graph are indicated in brackets):

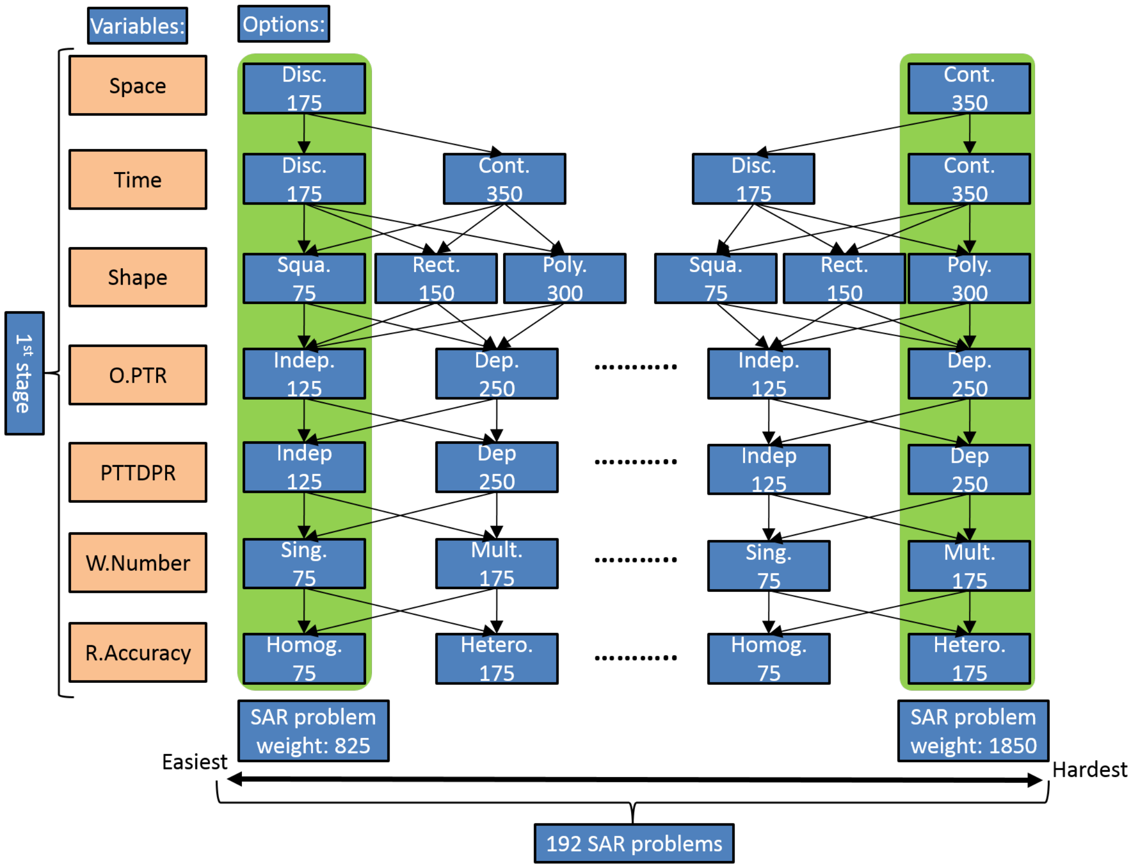
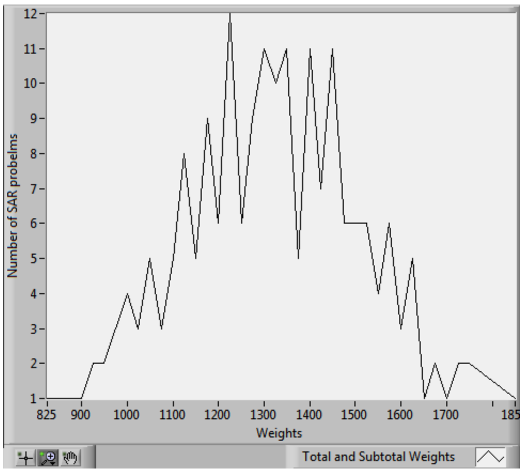
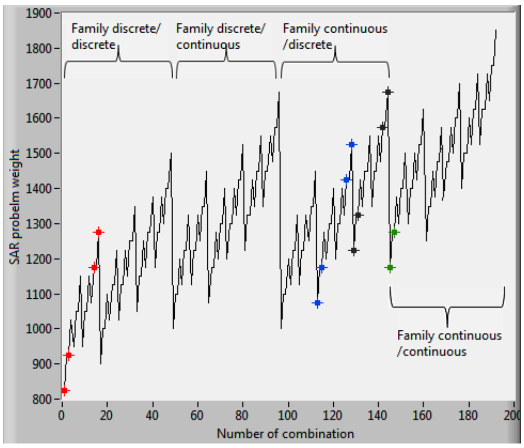


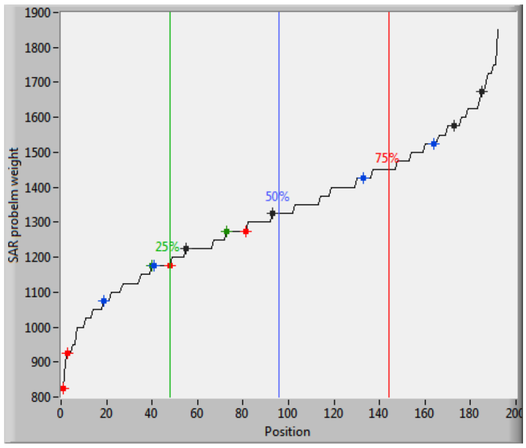
Figure 7.1: Representation of the combinatorial process for stage 1. The variables combined in stage 1 are shown in light orange rectangles, and their options in blue rectangles. The ternary variable and the six binary variables result in a total of 192 SAR problems. The combinatorial process is exhaustive and systematic, starting with the combination of the easiest options at the farthest left and the most difficult options at the farthest right.



(a)



(b)



(c)

Figure 7.2: Graphs showing results from the combinatorial step in stage 1. (a) Total space graph. The weights span from 825 to 1,850. The most repeated weight is 1,225 with 12 SAR problems. (b) Relationship graph. There are four main families classified by the first two variables' options in space: discrete, time: discrete; space: discrete, time: continuous; space: continuous, time: discrete; and space: continuous, time: continuous. Four of the selected problems are in the discrete/discrete family, eight are in the continuous/discrete, whilst two are in the continuous/continuous family. (c) Weighting graph. Six of the selected SAR problems are below 25% (i.e. positions 1, 3, 19, 40, 41, and 48), four between 25% and 50% (i.e. positions 55, 73, 81, 93), one between 50% and 75% (i.e. position 133), and three above 75% (i.e. positions 164, 173, and 185).

- Feasibility to be solved (problem positions 1, 3, 19, 40, 41, 48, 73 and 81)
- Realism to industrial scenarios (problem positions 55, 73, 93, 133, 164, 173 and 185)
- Interest to explore in combination with options of the next stage (problem positions 55, 73, 93, 173 and 185)

The range of the weights of selected partial SAR problems is from 825 to 1,675, whilst the range of weights of the generated SAR problems is from 825 to 1,850. In comparison, the selected SAR problems is representative of the generated SAR problems, with the minimal selected SAR problems at a 100% and the maximal selected SAR problem being at a 89.18% of the maximal generated SAR problem. In the weighting graph (Figure 7.2c), the selected SAR problems are classified by their percentage of difficulty (i.e. feasibility to be solved) within the following ranges:

- Six of the selected SAR problems are below the 25% difficulty (i.e. positions 1, 3, 19, 40, 41, and 48)
- Four of the selected SAR problems are between 25% and 50% difficulty (i.e. positions 55, 73, 81 and 93)
- One of the selected SAR problems is between 50% and 75% difficulty (i.e. position 133)
- Three of the selected SAR problems are above 75% difficulty (i.e. positions 164, 173, and 185)

According to the relationship graph, Figure 7.2b, the fourteen SAR problems are classified in the following families:

- Four from the discrete/discrete/square family
- Eight from the continuous/discrete family, where four have a rectangular representation of objects and four have a polygonal representation
- Two from the continuous/continuous/square family

Representation of selected SAR problems

The selected SAR problems can be represented with the combination of two groups of options. The first group is called roots and the second is called branches. Roots are fixed and are combined with several branches to generate distinct SAR problems. The method to select roots and branches is to find groups of options that represent the most common options at the upper and lower part of selected SAR problems.

The nomenclature for roots and branches is the following: number of stage and type of root or branch. These two values are separated by a dot. The first letters of the alphabet are used to denote roots, whilst the last letters of alphabet are used to denote branches. For example, the root 1.a refers to the first root from stage 1, whilst 1.z refers to the last branch from stage 1.

Roots in this stage are composed by the first three variables' options, whilst branches are composed by the last four variables' options. There are four roots and four branches. Roots and branches are expressed with the variables' codenames and their respective option. Root from stage 1 are:

- **Root 1.a**: space as discrete, time as discrete and shape as square
- **Root 1.b**: space as continuous, time as discrete, and shape as rectangular
- **Root 1.c**: space as continuous, time as discrete, and shape as polygonal
- **Root 1.d**: space as continuous, time as continuous, and shape as square

The branches from stage 1 have the following options:

- **Branch 1.w**: O.PTR as independent, PTTDPR as independent, W.Number as single, and R.Accuracy as homogeneous
- **Branch 1.x**: O.PTR as independent, PTTDPR as independent, W.Number as multiple, and R.Accuracy as homogeneous
- **Branch 1.y**: O.PTR as dependent, PTTDPR as dependent, W.Number as single, and R.Accuracy as heterogeneous
- **Branch 1.z**: O.PTR as dependent, PTTDPR as dependent, W.Number as multiple, and R.Accuracy as heterogeneous

The roots and branches have a mixture of hard and easy options. In stage 1, with the exception of root 1.d that is only combined with branches 1.w and 1.x, the rest of the roots are combined with all the branches. Roots, branches and their combinations can be observed in Figure 7.3. Roots and branches are combined to represent the selected SAR problems. The problems are presented according to their selection on the weighting graph and their positions on this graph are shown in brackets:

- Root 1.a with branches 1.w, 1.x, 1.y and 1.z correspond to the first, second, sixth and ninth selected SAR problems (positions 1, 3, 48 and 81) respectively

- Root 1.b with branches 1.w, 1.x, 1.y and 1.z correspond to the third, fifth, eleventh and twelveth selected SAR problems (positions 19, 41, 133 and 164) respectively
- Root 1.c with branches 1.w, 1.x, 1.y and 1.z correspond to the seventh, tenth, thirteen and fourteenth selected SAR problems (positions 55, 93, 173 and 185) respectively
- Root 1.d with branches 1.w and 1.x correspond to the fourth and eighth selected SAR problems (positions 40 and 73) respectively

Analysis of selected SAR problems

Roots and branches for stage 1 where enumerated in previous paragraphs. An analysis of representative SAR problems selected in stage 1 is provided in the following paragraphs. The **first SAR problem from stage 1 (position 1)** corresponds to the combination of root 1.a and branch 1.w. This SAR problem is the easiest to solve because the root 1.a and branch 1.w are the easiest to solve. The root 1.a considers the formulation of the problem with a discrete grid of squares, where objects can be represented as squares that either fit in a single square (e.g. robots) or cover several squares but are shaped as squares (e.g. warehouses, shelves). The root 1.a considers the time formulation as discrete, where time is represented as integer numbers in a finite set. The branch 1.w considers operations processing times (O.PTR) as independent of the robot that performs the operation. Consequently, any robot can perform any operation. The only consideration for the allocation of robots to operations is the current location of robots and the locations where the operations occur.

In the branch 1.w, parts transport time (PTTDPR) is considered as independent of the robot and the part that is transported. Similarly to the O.PTR assumption, the PTTDPR assumption allows selecting any robot for part transportation, and the only consideration is the current location of the robot, the location of the part to transport and its final location. In the branch 1.w, a single warehouse is considered. As a consequence, the paths of the robots need to consider a unique warehouse. Hence, resulting paths might require to be in circles, where robots collect raw materials perform a series of operations and return to the warehouse to deliver products and collect more materials. The use of a unique warehouse might lead to traffic congestion. The branch 1.w considers a homogeneous accuracy for all robots. As a result any robot can perform any manufacturing operation with the same quality.

The **second SAR problem from stage 1 (position 3)** is similar to the first SAR problem due to the use of the root 1.a. However, the branch is the 1.x, where multiple warehouses are considered. The formulation of a SAR problem

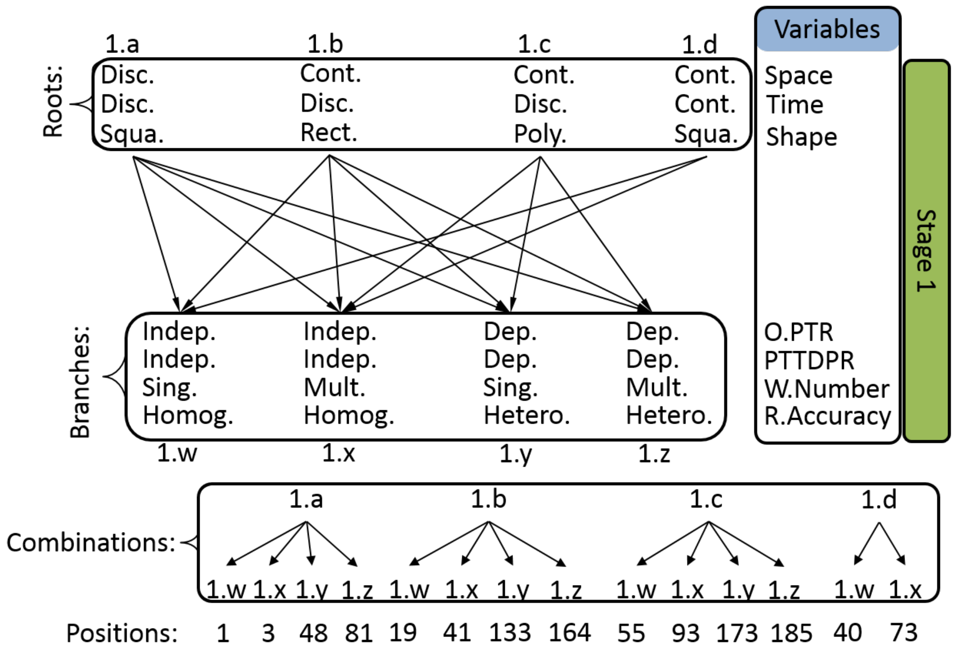


Figure 7.3: Representation of the selected SAR problems in stage 1. The representation is with the use of fixed roots and variable branches, where roots and branches are combinations of variables' options. There are four roots and four branches. The selected SAR problems are classified by their feasibility to be solved, their realism to industrial scenarios and interest of combining problems with variables of the next stage. The classified problems, with positions indicated in brackets, are: feasibility to be solved (1, 3, 19, 40, 41, 48, 73 and 81), realism to industrial scenarios (55, 73, 93, 133, 164, 173 and 185) and interest to explore in combination with options of the next stage (55, 73, 93, 173 and 185). The selected SAR problems from stage 1 can be represented as the combinations of all roots with all branches with the exception of root 1.d, which is only combined with branches 1.w and 1.x. Positions for the SAR problems are indicated below each combination.

that considers multiple warehouses requires to express fixed or variables locations for this warehouses, their dimensions and raw materials and products content. As a result of using multiple warehouses, there might be more feasible solutions for the second SAR problem, due to the multiple options to collect raw materials and deliver products. Also, the trafic congestion at each warehouse might be reduced, if not avoided. The second SAR problem might be more difficult to formulate and solve than the first one, but there is a chance to find more feasible solutions due to the use of multiple warehouses.

The **third SAR problem from stage 1 (position 19)** is similar to the first SAR problem in the use of the branch 1.w. However, the root for the SAR problem is the 1.b, where the space is represented as continuous, and objects are represented as rectangles. Consequently, the problem is more realistic because robots and warehouse can be considered as rectangles. However, a representation of the space as continuous might increase the number of feasible solutions but increase also the difficulty to determine these solutions. This difficulty results from the need to represent objects by their current x,y position and angle of orientation at every discrete time unit. The considerations for the branch 1.w are similar to the ones described for the first SAR problem. However, the warehouse is not limited to have square shape.

The **fourth SAR problem from stage 1 (position 40)** is similar to the first and third SAR problems in the use of the branch 1.w. However, a new root is introduced. This root is the 1.d, where the space is formulated as continuous, but objects are formulated as squares. Moreover, the time is formulated in a continuous way. The considerations of formulating the space in a continuous way were explained previously. However, the use of squares instead of rectangles avoid the need to represent the angle of orientation of the objects per each moment of time. As a consequence, a less realistic problem, but easier to solve is obtained. The representation of time as continuous increases the number of possible solutions and their realism. However, formulating the problem in continuous time require to consider robots positions at each instant of time instead of at discrete instants of time, resulting in a more difficult problem to solve.

The **fifth SAR problem from stage 1 (position 41)** is similar to the third problem (position 19). These two problems have the root 1.b, but the branch is different in the number of warehouses considered. The fifth SAR problem has the branch 1.w, which considers the use of multiple warehouses, whilst the third only considers a single warehouse. The combination of the root 1.b and the branch 1.w is more realistic than the previous described SAR problems due to the consideration of objects as rectangles, the space formulated in a continuous way and multiple number of warehouses. The last assumption might increase the number of feasible solutions over the third problem, but due to the use of rectangles the difficulty to solve is increased in comparison with problems formulated with squares (i.e. fourth).

The **sixth SAR problem from stage 1 (position 48)** makes use of the root 1.a, which is the easiest to solve due to the formulation of time and space as discrete, and the representation of objects as squares. However, the sixth SAR problem introduces a new branch which is the second most difficult to solve among the selected ones. This branch is the 1.y, which considers the operations processing time (O.PTR) dependent on the robot that executes the operation. Therefore, the allocation of robots to operations depends on their time to execute the operations and on the current location of the robot and the operation.

In order to formulate of O.PTR as dependent on the robot requires to know the processing times of each operation per each robot. In the branch 1.y, parts transport time (PTTDPR) depends on the robot and the part that is transported. The PTTDPR assumption requires to consider the robot and part that is transported. Hence, the transportation time does not depends only on the current location of the robot, the location of the part to transport and its final location. The PTTDPR assumption requires to known factors that affect the transportation time depending on the robot and part (e.g. robots' inertia control, parts' weights).

The branch 1.y considers a single warehouse. The consequences of this assumption were explained previously in the first, third and fourth SAR problems. In brief, robots paths might be circles to collect raw materials, perform a series of operations and return to the warehouse to deliver products and collect more materials. This might lead to traffic congestion at the warehouse entrance. In the branch 1.y, the accuracy of robots to perform an operation is considered as heterogeneous. Therefore, the quality of the operation depends on the robot that executes the operation. Similar to the PTTDPR assumption, it is necessary to know factors that affect the quality of executing a determined operation with a given robot (e.g. operation required quality, arm robot minimal precision). As a whole, the sixth SAR problem is as difficult to solve than the first SAR problem. However, the need for additional data regarding the O.PTR, PTTDPR and R.Accuracy increase the complexity of formulating the problem.

The **seventh SAR problem from stage 1 (position 55)** is similar to the first, third and fourth SAR problems due to the use of the branch 1.w. However, the seventh problem considers a more difficult root. This is the 1.c, which considers the space as continuous, the time as discrete and objects as polygons. The root 1.c is similar to the 1.b, with the exception that objects' shape is considered as polygons. Hence, the seventh problem is more realistic than any previous problem at the representation of objects, and the formulation of space as continuous also increases the realism of the problem. However, this also increased the difficulty to formulate the problem due to the need to describe all objects as polygons and to check if their boundaries overlap to avoid collisions. The fact that the time is formulated as discrete and the branch is the easiest one increases the solvability of the problem. Therefore, it is interesting to review this problem with variables' options from more complex stages.

The **eighth SAR problem from stage 1 (position 73)** is very similar to the fourth SAR problem. The eighth problem considers the root 1.d and the branch 1.x, whilst the fourth considers the branch 1.w. The branch of the eighth problem is the easiest to solve, but the root is the most difficult to solve. Both problems consider the space and time formulation as continuous, the shape of objects as squares, the operations processing time (O.PTR) as independent of the robot that executes the operation, the parts transport time (PTTDPR) as independent of the robot and the part that is transported and the accuracy of robots to perform an operation is considered as homogeneous. However, the eighth problem considers multiple warehouses instead of one. As a whole, the eighth SAR problem seems to be more feasible to be solved than the fourth SAR problem due to the consideration of multiple warehouses. The eighth problem is realistic due to the formulation of space and time as continuous and the use of multiple warehouses. However, the problem is limited to representation of objects as squares and it has limitations on the times and accuracy that requires to perform a manufacturing operation and transportation of parts.

The **fourteenth SAR problem from stage 1 (position 185)** is the most difficult of the selected SAR problems. The only option that is easy is the formulation of time as continuous, the rest of the options are formulated as the most difficult ones. This problem considers the root 1.c and the branch 1.z, which is the hardest in stage 1. The root 1.c was analysed for the seventh problem. However, the branch 1.z has not been analysed. This branch is similar to the branch 1.y, which was analysed for the sixth problem. The only difference between the branches 1.y and 1.z is the use of multiple warehouses for the branch 1.z, instead of a single warehouse for branch 1.y. This increases the options for the robots to reach warehouses and avoids circle paths to a single warehouse. As a whole, the fourteenth problem is difficult to formulate and to solve due mainly due to the root, namely, considering objects as polygons and the space formulation as continuous. The realism attained by a polygonal representation is not worthy, and therefore, is better to represents polygons as rectangles and keep the space formulated as continuous. The fourteen SAR problems selected in stage 1 are aggregated to the combinations of variables' options from stage 2 in the following subsection.

7.4.2 Stage 2

Combinations of variables' options from stage 2 are aggregated to the selected SAR problems from stage 1. Variables and variables' options, with variables' codenames and options' weigths in brackets, from stage 2 are:

- System control (S.Control): Decentralised (175), Centralised (350)
- Number of objectives to optimise (Objectives): Single (175), Multiple (350)
- Robots' 2D area (R.Area): Unequal (150), Equal (300)
- Products' start times (P.StartTimes): Single (100), Multiple (200)
- Products' deadlines (P.Deadlines): Single (100), Multiple (200)
- Warehouses content (W.Content): Unique (75), Mixed (150)

DMM's steps 4 and 5: Variables' options combination and aggregation

Combinations of variables' options from stage 2 are aggregated to the selected SAR problems from stage 1. A representation of the combinatorial and aggregation processes can be observed in Figure 7.4. The combinatorial process is identical to the process for stage 1 (see Figure 7.1), but with the post combination of the selected SAR problems in stage 1 (i.e. aggregation process).

Graphs generated from the combinatorial and aggregation processes in stage 2 are shown in Figure 7.5. The SAR problems generated in stage 2 depends of the selected SAR problems in stage 1 (i.e. fourteen) and the total combinations with the variables' options in stage 2 (i.e. 64). These SAR problems correspond to the subtotal SAR problem space. In contrast, the total SAR problem space is the total combination with the variables' options in stage 1 and 2 (i.e. 192 and 64, resulting in 12,288). The subtotal and total SAR problem space graphs are shown in Figures 7.5a and 7.5b respectively. The following metrics can be determined with the total and subtotal space graphs:

- The number of generated SAR problems in the subtotal space is 896, whilst in the total space is 12,288. Hence, the saved space is 92.70%. This percentage is determined from the subtotal and the total number of generated SAR problems.
- The range of generated SAR problems weights from the subtotal space is from 1,600 to 3,250, whilst for the total space is from 1,600 to 3,425. In comparison, the subtotal space is 94.89% of the total space.
- The most repeated SAR problem weight in the subtotal space is 2,425 with 48, whilst for the total space is 2,500 with 485. The maximal number of repeated SAR problems in the subtotal with respect to the total space represents the 9.8%.

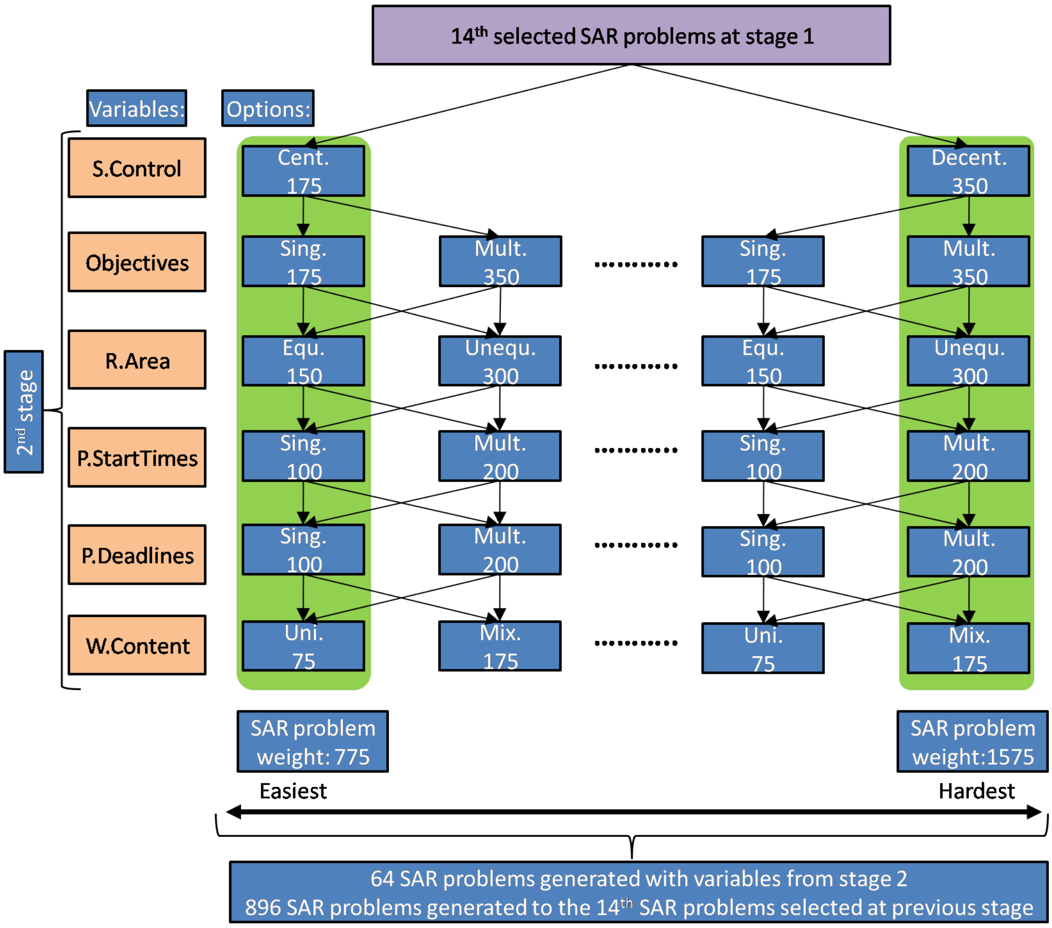


Figure 7.4: Representation of the combinatorial and aggregation processes for stage 2. The variables combined in stage are shown in light orange rectangles, and their options in blue rectangles. The six binary variables result in 64 SAR problems. These 64 SAR problems are aggregated to the fourteen SAR problems selected in stage 1 to generate 896 SAR problems. The combinatorial and aggregation processes are exhaustive and systematic, starting with the combination of the easiest options at the farthest left and the most difficult options at the farthest right.

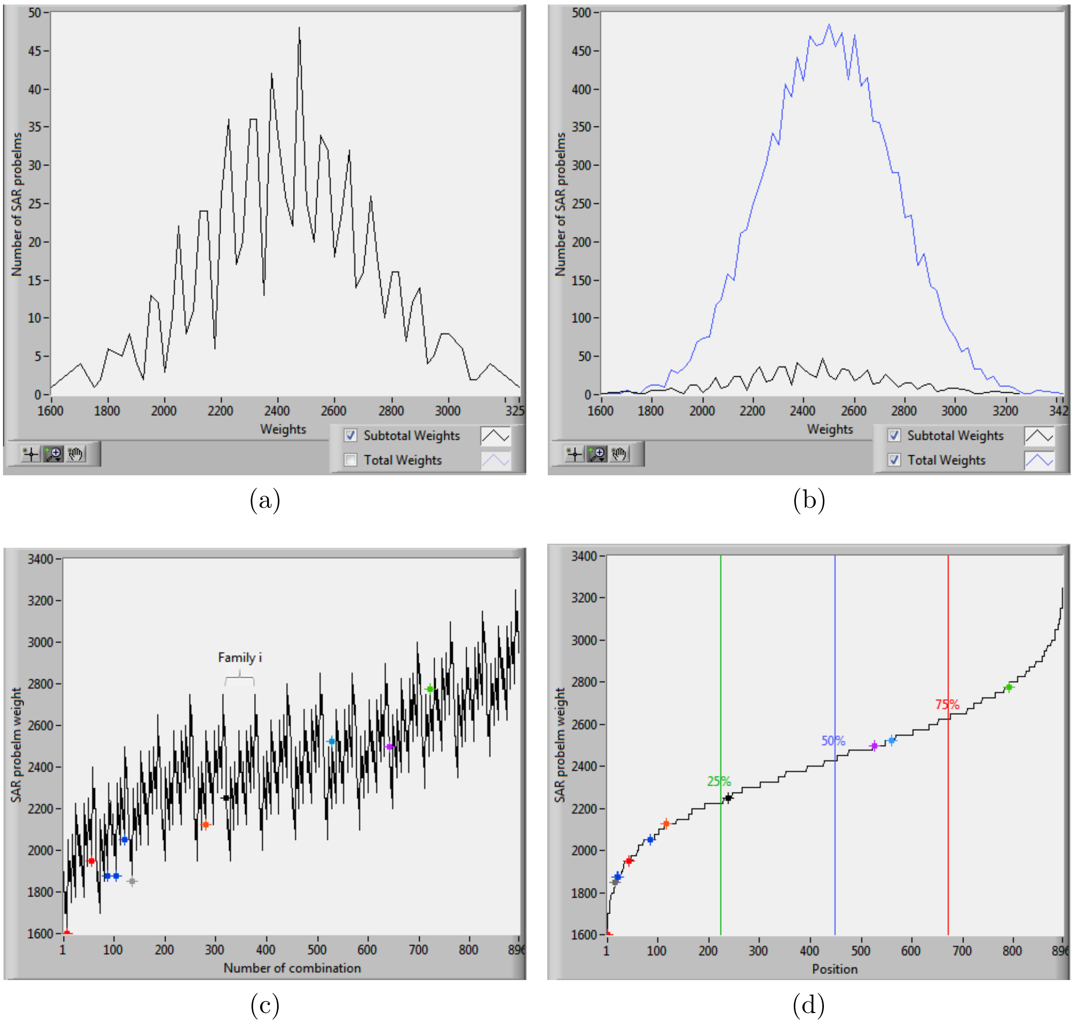


Figure 7.5: Graphs showing results from the aggregation and combination steps in stage 2. (a) Subtotal space graph. The weights from the subtotal graph span from 1,600 to 3,250. The most repeated weight is 2,425 with 48 SAR problems. (b) Total space graph. The weights from the total graph span from 1,600 to 3,425. The most repeated weight is 2,500 with 485 SAR problems. This graph shows the subtotal and total spaces. (c) Relationship graph. There are fourteen families resulting from the selected SAR problems in stage 1. Selected SAR problems at current stage are: two (red dots) from the first family, three (blue dots) from the second family. The rest are individual selections from the third (grey dot), fifth (orange dot), sixth (black dot), ninth (aquamarine dot), eleventh (purple dot), and twelveth (green dot) families. (d) Weighting graph. Seven of the selected SAR problems are below 25% (i.e. positions 1, 15, 20, 21, 43, 86 and 116), one between 25% and 50% (i.e. position 239), two between 50% and 75% (i.e. positions 527 and 559), and one above 75% (i.e. position 791).

DMM's steps 6: SAR problems analysis and selection

In stage 2, eleven SAR problems were selected. The selected SAR problems are summarised in Tables A.3 and A.4 from Appendix A. The reasons to select SAR problems are: feasibility to be solved; the realism to industrial scenarios and interest of combining problems with variables of the next stage. The reasons for selecting these problems are classified as follows (problem positions are indicated in brackets):

- Feasibility to be solved (problem positions 1, 15, 20 and 21)
- Realism to industrial scenarios (problem positions 43, 86, 116, 239, 527, 559 and 791)
- Interest to explore in combination with options of the next stage (problem positions 15, 20, 43, 86, 116, 527 and 791)

The range of the weights of selected partial SAR problems is from 1,600 to 2,775, whilst the range of weights of the generated SAR problems is from 1,600 to 3,250. In comparison, the selected SAR problems is representative of the generated SAR problems, with the minimal selected SAR problems at a 100% and the maximal selected SAR problem being at a 85,38% of the maximal generated SAR problem. The weighting graph can be observed in Figure 7.5d. The selected SAR problems are classified by their percentage of difficulty (i.e. feasibility to be solved) in the following ranges:

- Seven of the selected SAR problems are below the 25% difficulty (i.e. positions 1, 15, 20, 21, 43, 86 and 116)
- One of the selected SAR problems is between 25% and 50% difficulty (i.e. position 239)
- Two of the selected SAR problems are between 50% and 75% difficulty (i.e. positions 527 and 559)
- One of the selected SAR problems is above 75% difficulty (i.e. position 791)

The relationship graph is shown in Figure 7.5c. From this figure, it can be observed that there are fourteen main families of SAR problems resulting from fourteen selected SAR problems in stage 1. According to the relationship graph, the eleven SAR problems are classified in the following families:

- Two from the first family. This is discrete/discrete/square/independent/independing/single/homogeneous.

- Three from the second family. This is discrete/discrete/square/independent/independing/multiple/homogeneous.

The rest of the selections are from individual families. These are at the third, fifth, sixth, ninth, eleventh, and twelveth families. These are highlighted with different colours in Figures 7.5c and 7.5d.

Representation of selected SAR problems

The selected SAR problems are represented with the combination of two groups of options called roots and branches. The terms roots and branches were introduced in Subsection 7.4.1. A method to select roots and branches, and the notation of roots and branches was also described in Subsection 7.4.1. Roots are fixed and can be combined with several branches to represent the SAR problems. Roots in this stage are composed by the first seven variables' options (described in Subsection 7.4.1), whilst branches are composed by the last six variables' options. There are eight roots and six branches in stage 2. Roots and branches are expressed with the variables' codenames and their respective options. Roots from stage 2 can be represented with roots and branches from stage 1. These are:

- **Root 2.a**: 1.b + 1.w. This is space as continuous, time as discrete, shape as rectangular, O.PTR as independent, PTTDPR as independent, W.Number as single, and R.Accuracy as homogeneous

- **Root 2.b**: 1.a + 1.w. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as single, and R.Accuracy as homogeneous

- **Root 2.c**: 1.a + 1.x. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as multiple, and R.Accuracy as homogeneous

- **Root 2.d**: 1.b + 1.x. This is space as continuous, time as discrete, shape as rectangular, O.PTR as independent, PTTDPR as independent, W.Number as multiple, and R.Accuracy as homogeneous

- **Root 2.e**: 1.b + 1.y. This is space as continuous, time as discrete, shape as rectangular, O.PTR as dependent, PTTDPR as dependent, W.Number as single, and R.Accuracy as heterogeneous

- **Root 2.f**: 1.b + 1.z. This is space as continuous, time as discrete, shape as rectangular, O.PTR as dependent, PTTDPR as dependent, W.Number as multiple, and R.Accuracy as heterogeneous

- **Root 2.g**: 1.a + 1.y. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as single, and R.Accuracy as heterogeneous

- **Root 2.h**: 1.a + 1.z. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as multiple, and R.Accuracy as heterogeneous

The branches from stage 2 have the following variables' options:

- **Branch 2.u**: S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Branch 2.v**: S.Control as decentralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Branch 2.w**: S.Control as decentralised, Objectives as single, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Branch 2.x**: S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Branch 2.y**: S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, and W.Content as mixed

- **Branch 2.z**: S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, and W.Content as mixed

The roots and branches have a mixture of hard and easy options. It can be noted that roots 1.c and 1.d from stage 1 were discarded through the analysis in stage 2. Roots, branches and their combinations can be observed in Figure 7.6. Roots and branches are combined to represent the selected SAR problems. The problems are presented according to their selection on the weighting graph and their positions on this graph are shown in brackets:

- Root 2.a with branch 2.u correspond to the second selected SAR problem (position 15)

- Root 2.b with branches 2.u and 2.v correspond to the first and fifth selected SAR problems (positions 1 and 43) respectively

- Root 2.c with branches 2.v, 2.w and 2.x correspond to the sixth, third and fourth selected SAR problems (positions 86, 20 and 21) respectively

- Root 2.d with branch 2.x correspond to the seventh selected SAR problem (position 116)
- Roots 2.e and 2.f with branch 2.y correspond to the ninth and eighth selected SAR problems (positions 527 and 239) respectively
- Roots 2.g and 2.h with branch 2.z correspond to the eleventh and tenth selected SAR problems (positions 791 and 559) respectively

Analysis of selected SAR problems

Roots and branches for stage 2 where enumerated in previous paragraphs. Roots 1.c and 1.d from stage 1 are not selected to stage 2. The reasons are for root 1.c and 1.d due to the aggregation of polygonal objects and the combination of a continuous time formulation with options from stage 2, respectively. The aggregation of roots 1.c and 1.d increases the difficulty to solve SAR problems drastically. It was decided to discard problems containing these roots and focus on the other more solvable roots. An analysis of representative SAR problems selected in stage 2 is provided in the following paragraphs.

The **first selected SAR problem (position 1) of stage 2** considers the root 2.b and branch 2.u, which are the easiest ones for stage 2. The root 2.b corresponds to the first SAR problem of stage 1 (i.e. combination of root 1.a and branch 1.w). Root 2.b was already analysed. Branch 2.u considers a centralised system control, a single objective to optimise, robots with equal 2D area and products with single start times and deadlines. Also, a single warehouse with unique content due to the root 2.b which considers a unique warehouse. The problem as a whole is the easiest to formulate from stage 2. The unique warehouse might result in complicated paths in the solution but it is feasible to find solutions with off-the-shelf methods.

The **second selected SAR problem (position 15) of stage 2** considers the root 2.a, which is the combination of the root 1.b and branch 1.w from stage 1. Therefore, this root was explained as the third SAR problem from stage 1. In brief, the third SAR problem of stage 1 is more realistic because robots and warehouse are considered as rectangles. Also, a representation of the space as continuous might increase the difficulty to determine the solutions, but increase the number of feasible solutions. Objects have to be represented by their current x,y position and angle of orientation at every discrete time unit.

The branch for the second SAR problem of stage 2 is the 2.u, which considers the system control as centralised, a single objective to optimise, the area of the robots as equal, a single start times and deadlines for all products, and the content of the warehouse as unique. A centralised control system increases the possibilities of finding optimal solutions. However, it requires to have current information at each discrete instant of time of every object in the factory, and the current

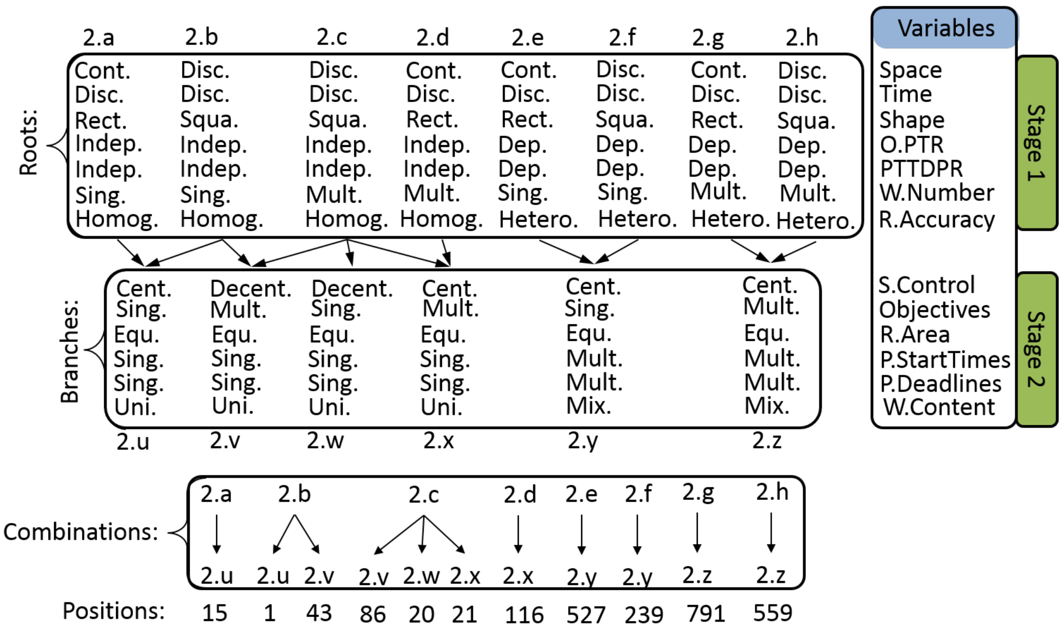


Figure 7.6: Representation of the selected SAR problems in stage 2. The representation is with the use of fixed roots and variable branches, where roots and branches are combinations of variables' options. There are eight roots and six branches. Roots from stage 2 are described through the combination of roots and branches from stage 1. Roots are as follows root 2.a: 1.b + 1.w, root 2.b: 1.a + 1.w, root 2.c: 1.a + 1.x, root 2.d: 1.b + 1.x, root 2.e: 1.b + 1.y, root 2.f: 1.b + 1.z, root 2.g: 1.a + 1.y, and root 2.h: 1.a + 1.z. Roots 1.c and 1.d are discarded in stage 2. The selected SAR problems are classified by their feasibility to be solved, their realism to industrial scenarios and interest of combining problems with variables of the next stage. The classified problems, with positions indicated in brackets, are: feasibility to be solved (1, 15, 20 and 21), realism to industrial scenarios (43, 86, 116, 239, 527, 559 and 791) and interest to explore in combination with options of the next stage (15, 20, 43, 86, 116, 527 and 791) The selected SAR problems from stage 2 can be represented as the combinations of roots with branches as indicated with the arrows. Positions for the SAR problems are indicated below each combination.

status of the manufacturing operations. Branch 2.u considers a single objetive for optimisation, which makes the problem more amenable and solvable. In branch 2.u, the 2D area of the robots is considered as equal. Adding to this, the space formulation is considered as continuous and the shape of objects as rectangles. There might be robots with equal 2D areas but different lateral sizes. However, the total 2D area that a robot can occupy is limited to a single value. Hence, the representation of robots is reduced to the single 2D area value and a lateral side value per each robot.

Branch 2.u considers a single start time and deadline for all the products. These two assumptions facilitate the formulation but constraint the production time for all robots. The second selected SAR problem of stage 2 considers a single warehouse due to the use of root 2.a and branch 2.u considers the content of this warehouse as unique. This is not very realistic to most of the factories, but it is not impossible, for example, to find a factory dedicated to assembly a single subcomponent or part to multiple products. As a whole this branch is the easiest to solve. However, when the branch and the root are analysed as a whole, the problem is still solvable. The only assumptions that might result problematic for solving the problem are the formulation of objects as rectangles, whilst robots have equal 2D areas.

The **third selected SAR problem (position 20) of stage 2** corresponds to the combination of 2.c and branch 2.w. Root 2.c corresponds to the second SAR problem from stage 1, which is the combination of root 1.a and branch 1.x. The root 2.c is more feasible to be solved than the root 2.b. However, the branch 2.w also considers the control system as decentralised, which requires a drastically different type of formulation. This branch considers the system control as decentralised, which requires the robots to determine their own production plans. The rest of the options for branch 2.v are a single objective to optimise, robots with equal 2D area, products have a single start time and deadline, and the content of the warehouse is unique. This means that different products require the same type of raw material to be assembled or machined. Because the only, option which is hard is the decentralised control system, this problem is very interesting to be researched. Moreover, with a decentralised control system it is feasible to research on scalability of the problem.

The **fifth selected SAR problem (position 43) of stage 2** considers the root 2.b and branch 2.v. The root 2.b corresponds to the first SAR problem of stage 1 (i.e. combination of root 1.a and branch 1.w), which is the easiest one. However, the branch is not the easiest one. The branch 2.v is similar to the branch 2.w. The only difference is the consideration of multiple objectives to optimise. Thi branch 2.v considers the system control as decentralised, thus problem formulation is changed drastically in comparison to centralised central system. Also, the branch

2.v considers multiple objectives to optimise, which added to the decentralised control, requires to formulate the problem and use solution methods drastically different to a centralised control. As a whole the problem is difficult to formulate and solve due to the decentralised system control in addition to multiple objectives.

The **tenth selected SAR problem (position 559) of stage 2** considers the root 2.h and the branch 2.z. The root 2.h corresponds to the ninth SAR problem from stage 1 (root 1.a and branch 1.z). The ninth SAR problem from stage 1 considers the easiest root but the most difficult root from stage 1. Moreover, the branch 2.z is one of the most difficult from stage 2. The only two easy options for the branch 2.z is a centralised control system and robots with equal 2D area, but the rest of the options are hard. These hard options are: multiple objectives to optimise, there are multiple start times and deadlines per product and warehouses have mixed content. Hence, the problem as a whole is feasible due to the formulation of time and space as discrete, objects represented as squares, where robots have equal 2D area. The problem is very realistic due to consideration of roots 1.z, where operations processing time depend on the robot that execute the operation, the transportation time of a part depends on the type of part and robot, and robots have heterogeneous minimal values of accuracy. Also, for the root 2.z, where multiple objectives, muliple start times and deadlines for products, multiple warehouses with mixed content are considered. This problem is both, realistic and solvable. The eleven SAR problems selected in stage 2 are aggregated to the combinations of variables' options from stage 3 in the following subsection.

7.4.3 Stage 3

Combinations of variables' options from stage 3 are aggregated to the selected SAR problems from stage 2. Variables and variables' options, with variables' codenames and options' weigths in brackets, from stage 3 are:

- Knowledge of the environment (Environment): Unstructured (150), Structured (300)
- Number of process plans (Plans): Single (150), Multiple (300)
- Parts loading and unloading time depends on the type of part and robot (PL/UTDPR): Independent (125), Dependent (250)
- Robots' types of wheels (R.Wheels): Unique (75), Mixed (175)
- Robots' battery duration (R.Battery): Homogeneous (50), Heterogeneous (100)

DMM's steps 4 and 5: Variables' options combination and aggregation

Combinations of variables' options from stage 3 are aggregated to the selected SAR problems from stage 2. A representation of the combinatorial and aggregation processes can be observed in Figure B.1 from Appendix B. The combination and aggregation processes are identical to the processes for stage 2 (see Figure 7.4), but with the use of variables, options and options' weights from stage 3.

Graphs generated from the combinatorial and aggregation processes in stage 3 shown in Figure 7.7. The SAR problems generated in stage 3 depends of the selected SAR problems in stage 2 (i.e. eleven) and the total combinations with the variables' options in stage 3 (i.e. 32). These SAR problems correspond to the subtotal SAR problem space. In contrast, the total SAR problem space is the total combination with the variables' options in stage 1, 2 and 3 (i.e. 192, 64 and 32 result in 393,216). The subtotal and total SAR problem spaces graph are shown in Figures 7.7a and 7.7b respectively. The following metrics can be determined with the total and subtotal space graphs:

- The number of generated SAR problems in the subtotal space is 352, whilst in the total space is 393,216. Hence, the saved space is 99.91%. This percentage is determined from the subtotal and the total number of generated SAR problems.

- The range of generated SAR problems weights from the subtotal space is from 2,150 to 3,900, whilst for the total space is from 2,150 to 4,550. In comparison, the subtotal space is 85.71% of the total space.

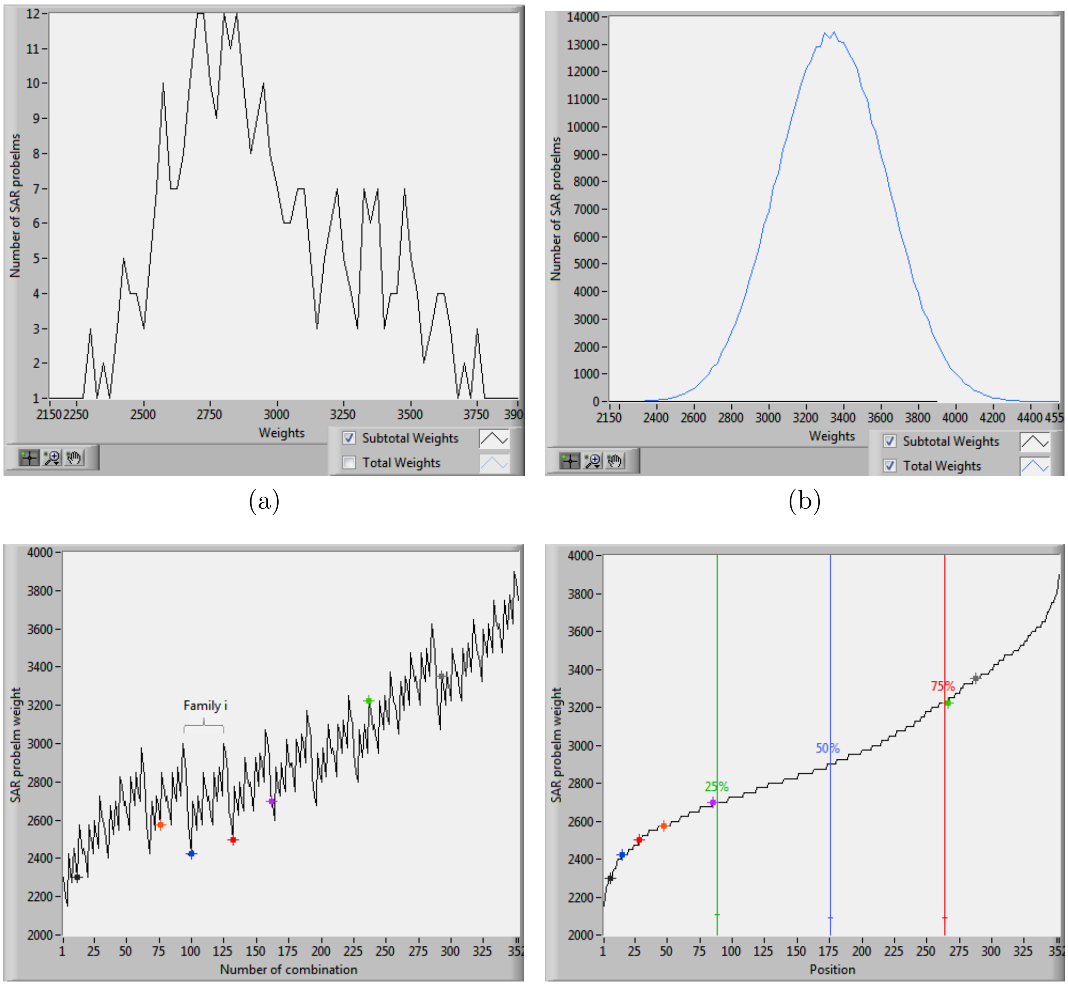
- The most repeated SAR problem weight in the subtotal space is 2,700, 2,725, 2,800 and 2,850 with 12 SAR problems, whilst for the total space is 3,350 with 13,470 SAR problems. The maximal number of repeated SAR problems in the subtotal with respect to the total space represents the 0.089%.

DMM's steps 6: SAR problems analysis and selection

In stage 3, seven SAR problems were selected. The selected SAR problems are summarised in Table A.5 from Appendix A. The reasons to select SAR problems are: feasibility to be solved; the realism to industrial scenarios and interest of combining problems with variables of the next stage. The reasons for selecting these problems are classified as follows (problem positions are indicated in brackets):

- Feasibility to be solved (problem positions 6, 15 and 47)

- Realism to industrial scenarios (problem positions 47, 266 and 288)



(a) (b) (c) (d)

Figure 7.7: Graphs showing results from the aggregation and combination steps in stage 3. (a) Subtotal space graph. The weights from the subtotal graph span from 2,150 to 3,900. The most repeated weight is 2,700 with 12 SAR problems. (b) Total space graph. The weights from the total graph span from 2,150 to 4,550. The most repeated weight is 3,350 with 13,470 SAR problems. This graph shows the subtotal and total spaces. (c) Relationship graph. There are eleven families resulting from the selected SAR problems in stage 2. Selected SAR problems at current stage are individual selections from the following families: first (black dot), third (orange dot), fourth (blue dot), fifth (red dot), sixth (purple dot), eighth (green dot) and tenth (grey dot). (d) Weighting graph. Five of the selected SAR problems are below 25% (i.e. positions 6, 15, 28, 47, and 85), and two above 75% (i.e. positions 266 and 288).

- Interest to explore in combination with options of the next stage (problem positions 28, 47 and 85)

The range of the weights of selected partial SAR problems is from 2,300 to 3,350, whilst the range of weights of the generated SAR problems is from 2,150 to 3,900. In comparison, the selected SAR problems is representative of the generated SAR problems, with the minimal selected SAR problems at a 93.47% of the minimal generated SAR problems and the maximal selected SAR problem being at a 85.89% of the maximal generated SAR problems. According to the weighting graph (Figure 7.7d), the selected SAR problems are classified by their percentage of difficulty (i.e. feasibility to be solved) within the following ranges:

- Five of the selected SAR problems are below the 25% difficulty (i.e. positions 6, 15, 28, 47, and 85)

- Two of the selected SAR problems are above 75% difficulty (i.e. positions 266 and 288)

The relationship graph is shown in Figure 7.7c. From this figure, it can be observed that there are eleven main families of SAR problems resulting from eleven selected SAR problems in stage 2. Seven SAR problems are selected in stage 3. These problems are distributed in individual families. These are at the first, third, fourth, fifth, sixth, eighth, and tenth families. These are highlighted with different colours in Figures 7.7c and 7.7d.

Representation of selected SAR problems

The selected SAR problems are represented with the combination of two groups of options called roots and branches. The terms roots and branches were introduced in Subsection 7.4.1. A method to select roots and branches, and the notation of roots and branches was also described in Subsection 7.4.1. Roots are fixed and can be combined with several branches to represent the SAR problems. Roots in this stage are composed by the first thirteen variables' options (described in Subsection 7.4.2), whilst branches are composed by the last five variables' options. There are seven roots and five branches which are combined together. Roots and branches are expressed with the variables' codenames and their respective options. Roots from stage 3 can be represented with roots and branches from stage 2. These are:

- **Root 3.a**: 2.b + 2.u. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as single, R.Accuracy as homogeneous, S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Root 3.b**: 2.b + 2.v. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as single, R.Accuracy as homogeneous, S.Control as decentralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Root 3.c**: 2.c + 2.w. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as multiple, R.Accuracy as homogeneous, S.Control as decentralised, Objectives as single, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Root 3.d**: 2.c + 2.x. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as multiple, R.Accuracy as homogeneous, S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Root 3.e**: 2.c + 2.v. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as multiple, R.Accuracy as homogeneous, S.Control as decentralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, and W.Content as unique

- **Root 3.f**: 2.f + 2.y. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as single, R.Accuracy as heterogeneous, S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, and W.Content as mixed

- **Root 3.g**: 2.h + 2.z. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as multiple, R.Accuracy as heterogeneous, S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, and W.Content as mixed

The branches from stage 3 have the following variables' options:

- **Branch 3.v**: Environment as structured, Plans as multiple, PL/UTDPR as independent, R.Wheels as unique and R.Battery as homogeneous

- **Branch 3.w**: Environment as structured, Plans as single, PL/UTDPR as independent, R.Wheels as unique and R.Battery as homogeneous

- **Branch 3.x**: Environment as structured, Plans as single, PL/UTDPR as independent, R.Wheels as mixed and R.Battery as homogeneous

- **Branch 3.y**: Environment as structured, Plans as multiple, PL/UTDPR as dependent, R.Wheels as mixed and R.Battery as heterogeneous

- **Branch 3.z**: Environment as structured, Plans as single, PL/UTDPR as dependent, R.Wheels as mixed and R.Battery as heterogeneous

The roots and branches have a mixture of hard and easy options. Roots 2.a, 2.d and 2.e from stage 2 were discarded through the analysis in stage 3. Roots, branches and their combinations can be observed in Figure 7.8. Roots and branches are combined to represent the selected SAR problems. The problems are presented according to their selection on the weighting graph and their positions on this graph are shown in brackets:

- Roots 3.a and 3.c with branch 3.v correspond to the first and third selected SAR problems (positions 6 and 28) respectively

- Roots 3.b and 3.d with branch 3.w correspond to the second and fourth selected SAR problems (positions 47 and 15) respectively

- Root 3.e with branch 3.x correspond to the fifth selected SAR problem (position 85)

- Root 3.f with branch 3.y correspond to the sixth selected SAR problem (position 266)

- Root 3.g with branch 3.z correspond to the seventh selected SAR problem (position 288)

Analysis of selected SAR problems

Roots and branches for stage 3 where enumerated in previous paragraphs. Roots 2.a, 2.d, 2.e and 2.f from stage 2 are not selected to stage 3 because of the formulation of the space as continuous and objects as rectangles. The aggregation of roots 2.a, 2.d, 2.e and 2.f increases the difficulty to solve the SAR problems. Therefore, problems containing these roots were discarded in order to focus on more solvable roots. Representative SAR problems selected in stage 3 are analysed in the following paragraphs.

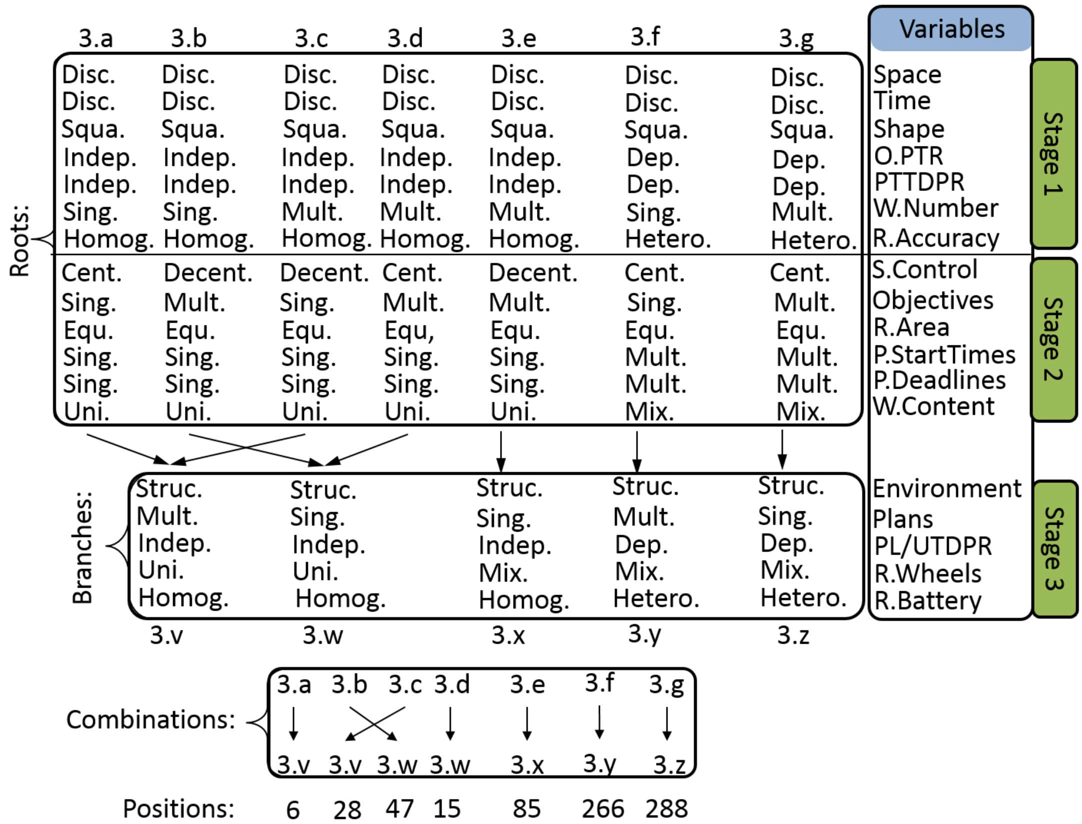


Figure 7.8: Representation of the selected SAR problems in stage 3. The representation uses fixed roots and variable branches, where roots and branches are combinations of variables' options. There are seven roots and five branches. Roots from stage 3 are described through the combination of roots and branches from stage 2. Roots are as follows root 3.a: 2.b + 2.u, root 3.b: 2.b + 2.v, root 3.c: 2.c + 2.w, root 3.d: 2.c + 2.x, root 3.e: 2.c + 2.v, root 3.f: 2.f + 2.y, and root 3.g: 2.h + 2.z Roots 2.a, 2.d and 2.e are discarded in stage 3. The selected SAR problems are classified by their feasibility to be solved, their realism to industrial scenarios and interest of combining problems with variables of the next stage. The classified problems, with positions indicated in brackets, are: feasibility to be solved (6, 15 and 47), realism to industrial scenarios (47, 266 and 288) and interest to explore in combination with options of the next stage (28, 47 and 85). The selected SAR problems from stage 3 can be represented as the combinations of roots with branches as indicated with the arrows. Positions for the SAR problems are indicated below each combination.

The **first selected SAR problem (position 6) of stage 3** considers the root 3.a and branch 3.v, which are the second easiest one for stage 3. The root 3.a corresponds to the first SAR problem of stage 2 (i.e. combination of root 2.a and branch 2.u). The root 3.a was already analysed. The branch 3.v considers the environment as structured, multiple production plans for the products, the loading and unloading time of parts is independent of part and robot (PL/UTDPR), a unique type of wheels for all robots, maximal duration of battery is homogeneous to all robots. The only hard option for this branch is the consideration of multiple production plans for the products. The consideration of multiple production plans and a single start time and deadline for all products, make it necessary to determine the production plans for all products that start and finish at the same. In the first selected SAR problem of stage 3 problem, the assumption PL/UTDPR is considered as independent. In this case, the PL/UTDPR assumption affect the problem in the same way than the PTTDPR assumption. In both, the manufacturing and transportation tasks are independent of the robots that executes them. Also, the consideration of unique type of robot wheels, and homogeneous maximal battery duration allows selecting any robot to perform any task. Although, the first selected SAR problem of stage 3 has the second easiest branch, this problem has a very easy root that has the easiest options from stage 1 and 2. Therefore, the problem as a whole is the easiest to formulate and feasible to solve from stage 3.

The **second selected SAR problem (position 15) of stage 3** considers the root 3.d and branch 3.w. The root 3.d corresponds to the fourth SAR problem of stage 2 (i.e. root 2.c and branch 2.x). This root is feasible to formulate due to the formulation of space and time as discrete, the representation of objects as squares, robot with equal 2D area and a centralised system control. Moreover, the operations processing time is independent of the robot that executes the task, the parts transport time is independent of part and robot type, robots have homogeneous minimal accuracy, and products have a single start time and deadline. However, the root is not realistic, except for the consideration of multiple warehouses with unique content and multiple objectives to optimise. The branch 3.w is similar to the branch 3.v, but instead of multiple production plans, it consider a single plan for each product. Therefore, as a whole the second selected SAR problem of stage 3 is feasible to formulate and solve, but not very realistic.

The **sixth selected SAR problem (position 266) of stage 3** corresponds to the combination of root 3.f and branch 3.y. The root 3.f corresponds to the eleventh SAR problem from stage 1, which is the combination of root 2.f and branch 2.y. The root 3.f is the second most hardest. Although the space and time are formulated as discrete, the objects are represented as squares and there is a single objective to optimise, the rest of the options are the hardest ones. The fact that a single warehouse is considered is easy to formulate but might lead to intractable problem formulations and a lack of feasible solutions. Moreover the branch 3.y is the most difficult from this stage due to the consideration of multiple production plans for each product, parts loading and unloading time dependent on part and robot type, a mix of type of robot wheels, and heterogeneous battery duration for robots. The consideration of multiple production plans and multiple start times and deadlines for the products increases substantially the difficulty of formulating and solving the problem. Also, the consideration of the assumptions O.PTR as dependent, PTTDPR as dependent, R.Accuracy as heterogeneous, PL/UTDPR as dependent, R.Wheels as mixed and R.Battery as heterogeneous makes necessary to determine a specific robot per each task. Hence, the problem is realistic to industry and feasible to formulate, but substantially more difficult to solve. The seven SAR problems selected in stage 3 are aggregated to the combinations of variables' options from stage 4 in the following subsection.

7.4.4 Stage 4

Combinations of variables' options from stage 4 are aggregated to the selected SAR problems from stage 3. Variables and variables' options, with variables' codenames and options' weigths in brackets, from stage 4 are:

- System accuracy (S.Accuracy): Certain (175), Uncertain (350)
- Warehouses locations (W.Locations): Single (125), Multiple (275)
- Tools setup and removal times are equal for all robots (T.SRT): Equal (75), Unequal (150)
- Robots' payload (R.Payload): Homogeneous (50), Dependent (100)
- Robots' number of carried tools (R.CarriedTools): Unique (50), Mixed (100)

DMM's steps 4 and 5: Variables' options combination and aggregation

Combinations of variables' options from stage 4 are aggregated to the selected SAR problems from stage 3. A representation of the combinatorial and aggregation processes can be observed in Figure B.2 from Appendix B. The combination and aggregation processes are identical to the process for stage 2 (see Figure 7.4), but with the use of variables, options and options' weights from stage 4.

Graphs generated from the combinatorial and aggregation processes in stage 4 are shown in Figure 7.9. The SAR problems generated in stage 4 depends of the selected SAR problems in stage 3 (i.e. seven) and the total combinations with the variables' options in stage 4 (i.e. 32). These SAR problems correspond to the subtotal SAR problem space. In contrast, the total SAR problem space is the total combination with the variables' options in stage 1, 2 and 3 (i.e. 192, 64, 32 and 32 result in 12,582,912). The subtotal and total SAR problem spaces graph are shown in Figures 7.9a and 7.9b respectively. The following metrics can be determined with the total and subtotal space graphs:

- The number of generated SAR problems in the subtotal space is 224, whilst in the total space is 12,582,912. Hence, the saved space is 99.998%. This percentage is determined from the subtotal and the total number of generated SAR problems.
- The range of generated SAR problems weights from the subtotal space is from 2,625 to 4,275, whilst for the total space is from 2,625 to 5,500. In comparison, the subtotal space is 77.72% of the total space.
- The most repeated SAR problem weight in the subtotal space is 3,175 and 3,225 with 12 SAR problems, whilst for the total space is 4,050 with 400,000 SAR problems. The maximal number of repeated SAR problems in the subtotal with respect to the total space represents the 0.003%.

DMM's steps 6: SAR problems analysis and selection

In stage 4, four SAR problems were selected. The selected SAR problems are summarised in Table A.6 from Appendix A. The reasons to select SAR problems of stage 4 only two. These are their feasibility to be solved and the realism to industrial scenarios. The reasons for selecting these problems are classified as follows (problem positions are indicated in brackets):

- Feasibility to be solved (problem positions 1 and 8)
- Realism to industrial scenarios (problem positions 194 and 216)

The range of the weights of selected partial SAR problems is from 2,750 to 4,125, whilst the range of weights of the generated SAR problems is from 2,625 to 4,275 In comparison, the selected SAR problems is representative of the generated SAR problems, with the minimal selected SAR problems at a 95.45% of the minimal generated SAR problems and the maximal selected SAR problem being at a 96.49% of the maximal generated SAR problems. According to the weighting graph (Figure 7.9d), the selected SAR problems are classified by their percentage of difficulty (i.e. feasibility to be solved) within the following ranges:

- Two of the selected SAR problems are below the 25% difficulty (i.e. positions 1 and 8)

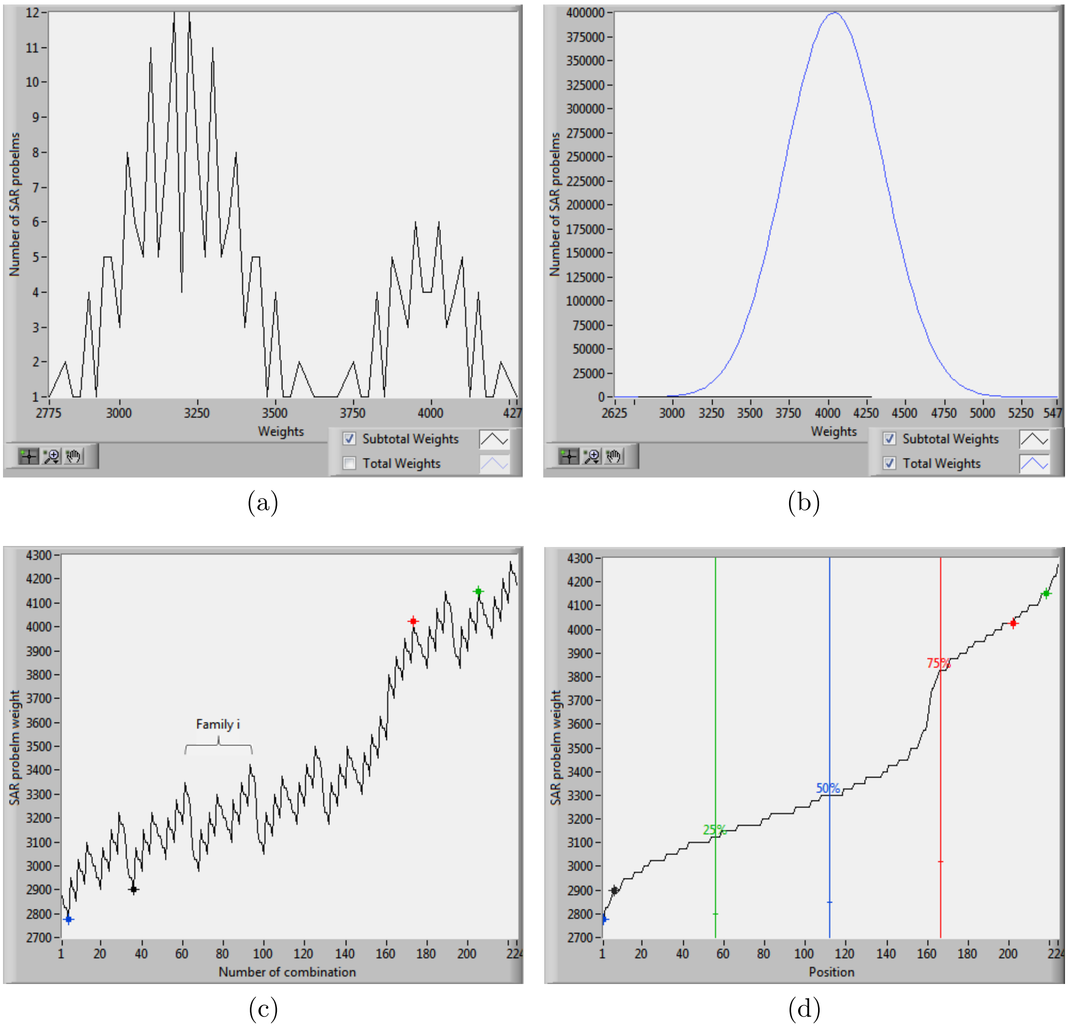


Figure 7.9: Graphs showing results from the aggregation and combination steps in stage 4. (a) Subtotal space graph. The weights from the subtotal graph span from 2,625 to 4,275. The most repeated weights are 3,175 and 3,225 with 12 SAR problems. (b) Total space graph. The weights from the total graph span from 2,625 to 5,500. The most repeated weight is 4,050 with 400,000 SAR problems. This graph shows the subtotal and total spaces. (c) Relationship graph. There are seven main families resulting from the selected SAR problems in stage 3. Selected SAR problems at current stage are individual selections from the following families: first (blue dot), second (black dot), sixth (red dot), and seventh (green dot). (d) Weighting graph. Two of the selected SAR problems are below 25% (i.e. positions 1 and 8), and two above 75% (i.e. positions 194 and 216).

- Two of the selected SAR problems are above 75% difficulty (i.e. positions 194 and 216)

The relationship graph is shown in Figure 7.9c. From this figure, it can be observed that there are seven main families of SAR problems resulting from seven selected SAR problems in stage 3. Four SAR problems are selected in stage 4 and they are distributed in individual families. These are at the first, second, sixth, and seventh families. These are highlighted with different colours in Figures 7.9c and 7.9d.

Representation of selected SAR problems

The selected SAR problems are represented with the combination of two groups of options called roots and branches. The terms roots and branches were introduced in Subsection 7.4.1. A method to select roots and branches, and the notation of roots and branches was also described in Subsection 7.4.1. Roots are fixed and can be combined with several branches to represent the SAR problems. Roots in this stage are composed by the first eighteen variables' options (described in Subsection 7.4.3), whilst branches are composed by the last five variables' options. There are four roots and two branches which are combined together. Roots and branches are expressed with the variables' codenames and their respective options. Roots from stage 4 can be represented with roots and branches from stage 3. These are:

- **Root 4.a**: 3.a + 3.v. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as single, R.Accuracy as homogeneous, S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as single, P.Deadlines as single, W.Content as unique, Environment as structured, Plans as multiple, R.Wheels as unique, PL/UTDPR as independent and R.Battery as homogeneous

- **Root 4.b**: 3.d + 3.w. This is space as discrete, time as discrete, shape as square, O.PTR as independent, PTTDPR as independent, W.Number as multiple, R.Accuracy as homogeneous, S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as single, P.Deadlines as single, W.Content as unique, Environment as structured, Plans as single, R.Wheels as unique, PL/UTDPR as independent and R.Battery as homogeneous

- **Root 4.c**: 3.f + 3.y. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as single, R.Accuracy as heterogeneous, S.Control as centralised, Objectives as single, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, W.Content as mixed, Environment as structured, Plans as multiple, R.Wheels as mixed, PL/UTDPR as dependent and R.Battery as heterogeneous

- **Root 4.d**: 3.g + 3.z. This is space as discrete, time as discrete, shape as square, O.PTR as dependent, PTTDPR as dependent, W.Number as multiple, R.Accuracy as heterogeneous, S.Control as centralised, Objectives as multiple, R.Area as equal, P.StartTimes as multiple, P.Deadlines as multiple, W.Content as mixed, Environment as structured, Plans as single, R.Wheels as mixed, PL/UTDPR as dependent and R.Battery as heterogeneous

The branches from stage 4 have the following variables' options:

- **Branch 4.y**: S.Accuracy as certain, W.Locations as constant, T.SRT as equal, R.Payload as homogeneous and R.CarriedTools as unique

- **Branch 4.z**: S.Accuracy as certain, W.Locations as dynamic, T.SRT as unequal, R.Payload as heterogeneous and R.CarriedTools as multiple

The roots and branches have a mixture of hard and easy options. It can be noted that roots 3.b, 3.c and 3.e from stage 3 were discarded through the analysis in stage 4. Roots, branches and their combinations can be observed in Figure 7.10. Roots and branches are combined to represent the selected SAR problems. The problems are presented according to their selection on the weighting graph and their positions on this graph are shown in brackets:

- Roots 4.a and 4.b with branch 4.y correspond to the first and second selected SAR problems (positions 1 and 8) respectively

- Roots 4.c and 4.d with branch 4.z correspond to the third and fourth selected SAR problems (positions 194 and 216) respectively

Analysis of selected SAR problems

Roots and branches for stage 4 where enumerated in previous paragraphs. The roots 3.b, 3.c and 3.e from stage 3 are not selected to stage 4 because of the consideration of a decentralised control system. Aggregating roots 3.b, 3.c and 3.e would increase the drastically difficulty to solve the SAR problems with additional variables' options from stage 4. An analysis of representative SAR problems selected in stage 2 is provided in the following paragraphs.

The **first selected SAR problem (position 1) of stage 4** considers the root 4.a and branch 4.y, where both are the easiest ones in this stage. The root 4.a corresponds to the first SAR problem of stage 3 (i.e. combination of root 3.a and branch 3.v). The root 4.a was already analysed in stage 3. The branch 4.y

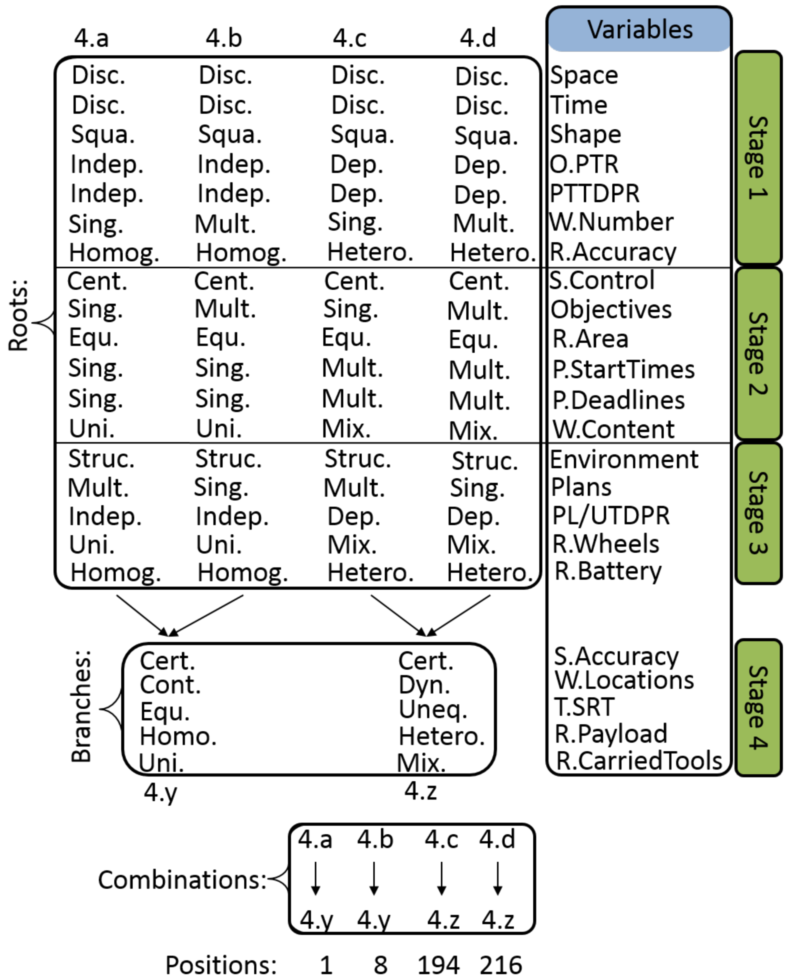


Figure 7.10: Representation of the selected SAR problems in stage 4. The representation of uses fixed roots and variable branches, where roots and branches are combinations of variables' options. There are four roots and two branches. Roots from stage 4 are described through the combination of roots and branches from stage 3. Roots are as follows root 4.a: 3.a + 3.v, root 4.b: 3.d + 3.w, root 4.c: 3.f + 3.y, root 4.d: 3.g + 3.z. Roots 3.b, 3.c and 3.e are discarded in stage 4. The selected SAR problems are classified by their feasibility to be solved, their realism to industrial scenarios and interest of combining problems with variables of the next stage. The classified problems, with positions indicated in brackets, are: feasibility to be solved (1 and 8) and realism to industrial scenarios (194 and 216). The selected SAR problems from stage 3 can be represented as the combinations of roots with branches as indicated with the arrows. Positions for the SAR problems are indicated below each combination.

considers the system accuracy as certain, the locations of warehouses as constant during all the production, the tools setup and removal times as equal for all robots but different between them, the maximal carrying payload as homogeneous for all robots and a unique number of maximal carried tools for all robots. The branch effect on the root are not drastical because the new variables' options are similar to the variables' options from root 4.a.

The assumptions T.SRT as equal, R.Payload as homogeneous, R.CarriedTools as unique have a similar effect to the assumptions PL/UTDPR, O.PTR, and PTTDPR as independent, R.Wheels as unique and R.Battery as homogeneous. Assuming a warehouse that does not change position during production is realistic and facilitates the problem formulation. The consideration of the system accuracy as certain represents that all sensing measures of the whole system do not have any type of uncertainty. The only challenge for the first selected SAR problem of stage 4 is the consideration of multiple production plans with single start times and deadlines for all products. However, because manufacturing and transportation tasks can be executed by any robot, it is very feasible to solve the problem. The problem as a whole is the easiest to formulate from stage 4. However, the large number of considerations (i.e. twenty three variables) increase the difficulty.

The **fourth selected SAR problem (position 216) of stage 4** considers the root 4.d and the branch 4.z, which are the most difficult ones in stage 4. The root 4.d is the combination of the root 3.g and branch 3.z from stage 3. The fourth selected SAR problem of stage 4 is almost completely the opposite to the first selected SAR problem. The only variables' options that are equal are the formulation of time and space as discrete, objects as squares, centralised system control, robot with equal 2D area, structured environment and system accuracy as certain. The branch 4.z considers the system accuracy as certain, the locations of warehouses as dynamic during all the production, the tools setup and removal times as unequal for all robots but different between them, the maximal carrying payload as heterogeneous for all robots and a mixed number of maximal carried tools for all robots.

The effects of branch 4.z on root 4.d are not drastical because the new variables' options are similar to the variables' options from root 4.d. The assumptions T.SRT as unequal, R.Payload as heterogeneous, R.CarriedTools as mixed have a similar effect to the assumptions O.PTR, PTTDPR and PL/UTDPR as dependent, R.Wheels as mixed and R.Battery as heterogeneous. The manufacturing and transportation tasks cannot be executed by any robot. The only assumption that reduced the difficulty of the problem is considering a single production plan per each product. The problem as a whole is the most difficult to formulate and solve from stage 4 due to the large number of considerations (i.e. twenty three variables) and the most difficult assumption for fifteen of them.

The other two selected SAR problems have combinations of options explained in the previous two analysis. Their level of difficulty is between the previous two. The first and second selected SAR problems in stage 4 are highly solvable and they describe realistic scenarios for the production with Self-Reconfigurable Manufacturing Systems (S-RMSs). The third and fourth problems are highly real, and they correspond to relaxations of variables from the first two problems. Therefore, it is recommended to start the problem formulation in the order of appearance in Table A.6. The four selected SAR problems in stage 4 are complete SAR problems that have variables from the four stages.

7.4.5 Intrastage analysis

The analysis between stages refers to the evolution of the subtotal and total SAR problem space through the four stages. The subtotal and total spaces change at each stage due to the increase in the number of generated SAR problems and their weights.

The evolution of the subtotal SAR problem space shows an increase through the stages, see Figure 7.11. The SAR problem weights for stage 1 range from 825 to 1,850, for stage 2 range from 1,600 to 3,250, for stage 3 range from 2,150 to 3,900, and for stage 4 range from 2,625 to 4,275. There is a similar trend with the maximal SAR problem weight at each stage. The maximal SAR problem weights are 1,225 for stage 1, 2,425 for stage 2, 2,700 for stage 3, and 3,175 for stage 4. The number of generated SAR problems in the subtotal space increases from 192 to 896 from stage 1 to stage 2, decreases from 896 to 352 from stage 2 to stage 3, and decreases from 352 to 224 from stage 3 to stage 4.

The evolution of the total SAR problem space also shows an increase through the stages, see Figure 7.12. The SAR problem weights for stage 1 range from 825 to 1,850, for stage 2 range from 1,600 to 3,425, for stage 3 range from 2,150 to 4,550, and for stage 4 range from 2,625 to 5,500. In a similar way, the maximal SAR problem weight per stage shows an increasing trend. This is demonstrated by the following maximal weights per stage: 1,225 in stage 1, 2,500 in stage 2, 3,350 in stage 3, and 4,050 in stage 4. There is also an increment of the number of generated SAR problems from the total space. The number of total SAR problems in stage 1 is 196, in stage 2 is 12,288, in stage 3 is 393,216, and in stage 4 is 12,582,912.

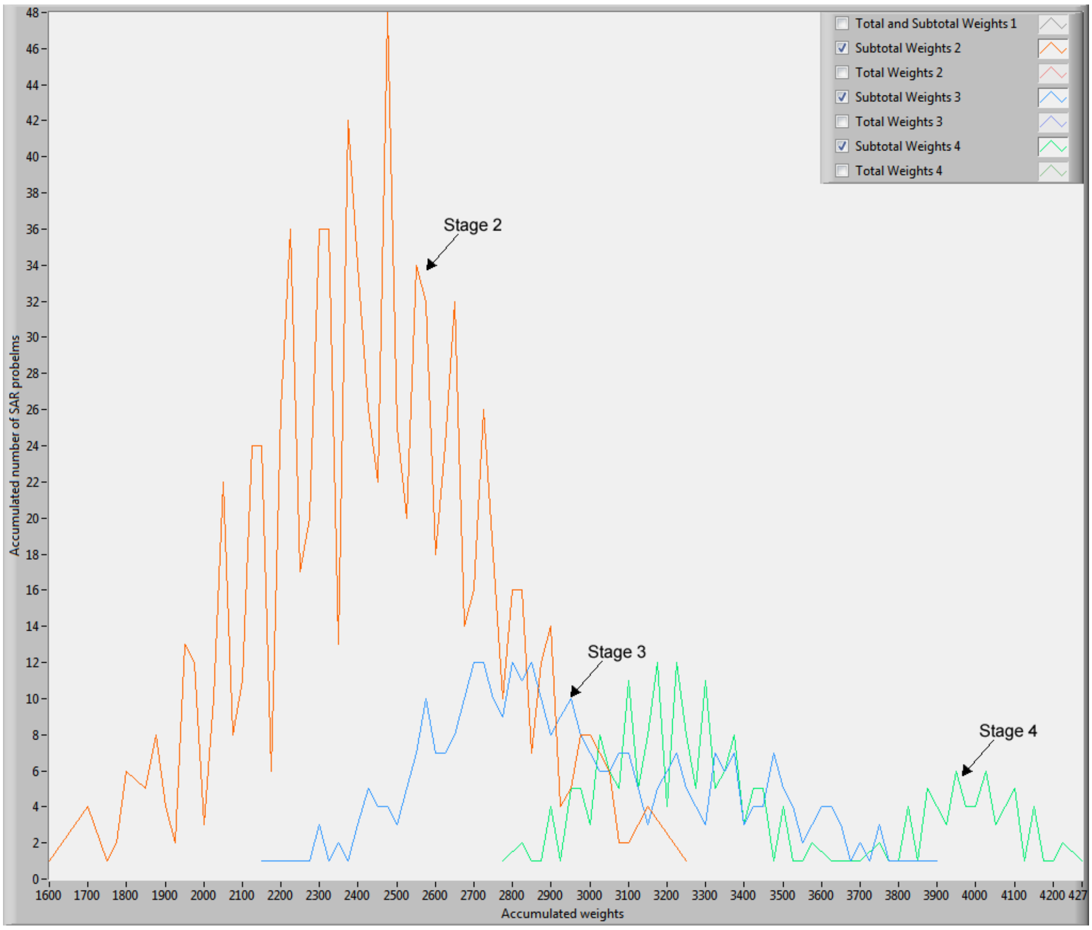


Figure 7.11: Evolution of the subtotal SAR problem space, where there is a decrease of the maximal weight per stage as follows: stage 1 with 1,225, stage 2 with 2,425, stage 3 with 2,700, and stage 4 with 3,175. There is an increase in the number of generated SAR problems from stage 1 to stage 2 but decreases from stages 2 to 3 and 3 to 4. These are as follows: stage 1 with 192, stage 2 with 896, stage 3 with 352, and stage 4 with 224. There are increments in the range of weights generated at each stage. These are: stage 1 from 825 to 1,850, stage 2 from 1,600 to 3,250, stage 3 from 2,150 to 3,900, and stage 4 from 2,625 to 4,275.

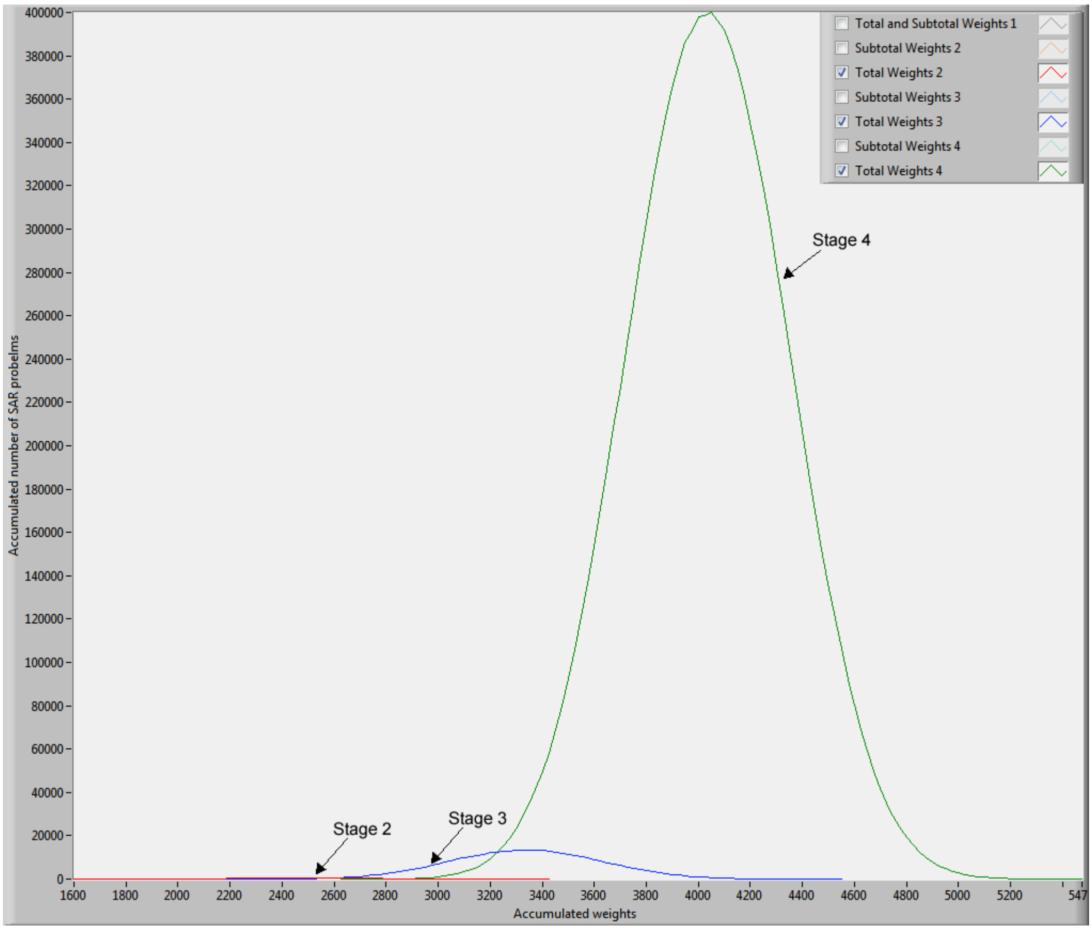
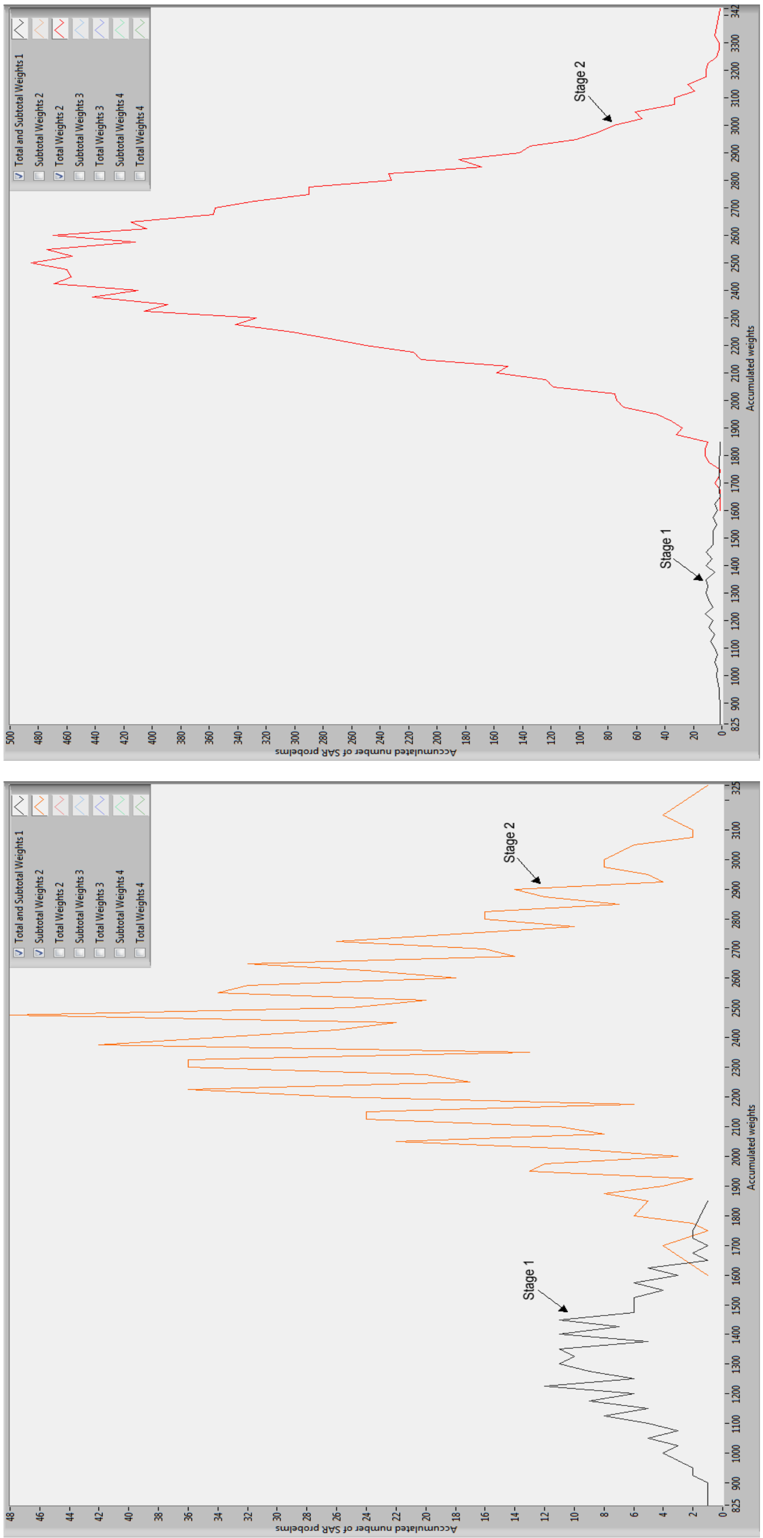


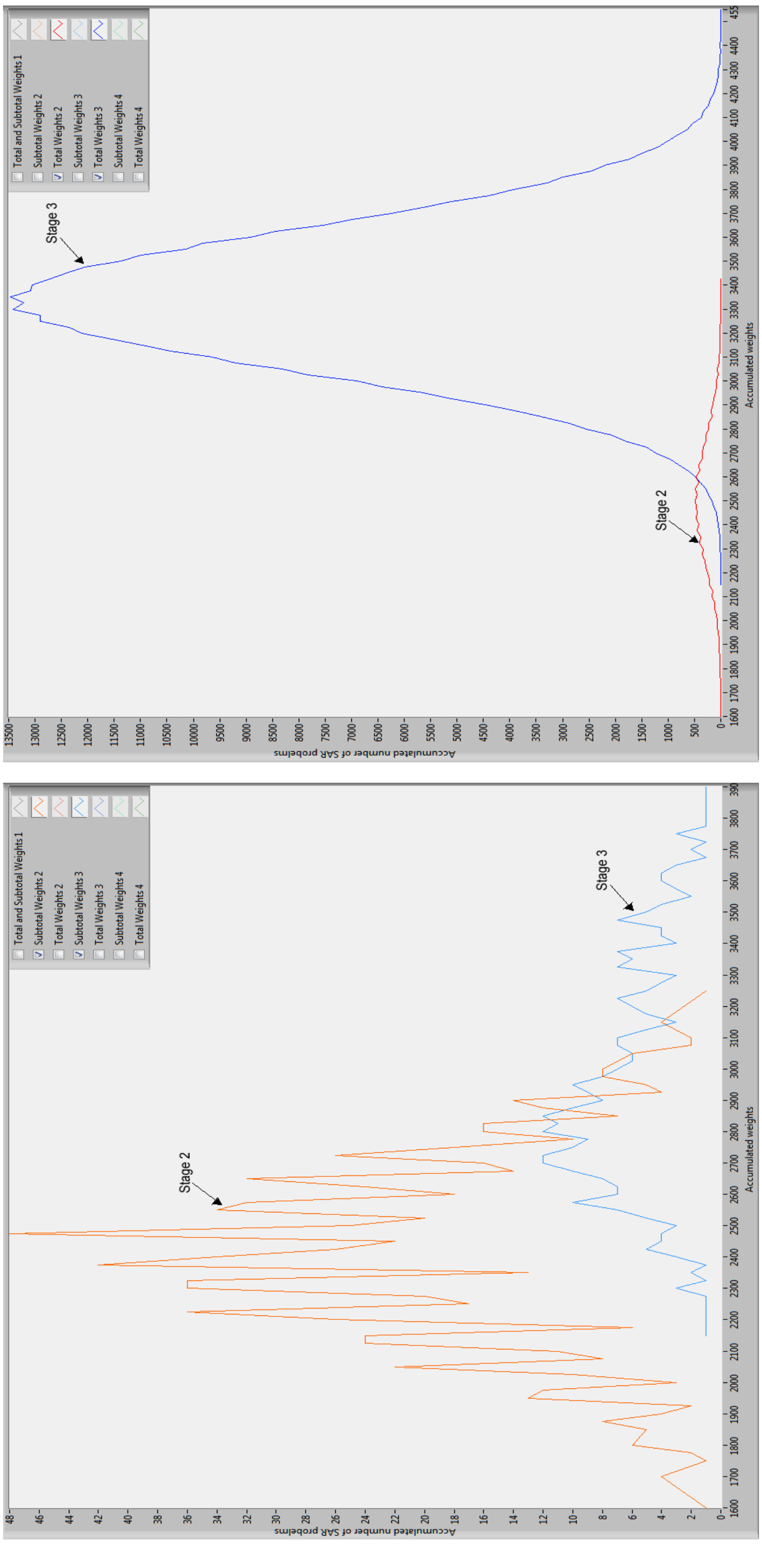
Figure 7.12: Evolution of the total SAR problem space. In contrast to the subtotal space, there is an increase in the maximal weight per stage as follows: stage 1 with 1,225, stage 2 with 2,500, stage 3 with 3,350, and stage 4 with 4,050 SAR problems. The possible generated shows also an increase through the stages as follows: stage 1 with 196, stage 2 with 12,288, stage 3 with 393,216, and stage 4 with 12,582,912 SAR problems. The range of weights shows the same increasing trend as stage 1 from 825 to 1,850, stage 2 from 1,600 to 3,425, stage 3 from 2,150 to 4,550, and stage 4 from 2,625 to 5,500 SAR problem weights.



(a)

(b)

Figure 7.13: Subtotal and total SAR problem space between stages 1 and stage 2. (a) Subtotal space between stage 1 and stage 2. (b) Total space between stage 1 and stage 2.



(a) (b)

Figure 7.14: Subtotal and total SAR problem space between stages 2 and stage 3. (a) Subtotal space between stage 2 and stage 3. (b) Total space between stage 2 and stage 3.

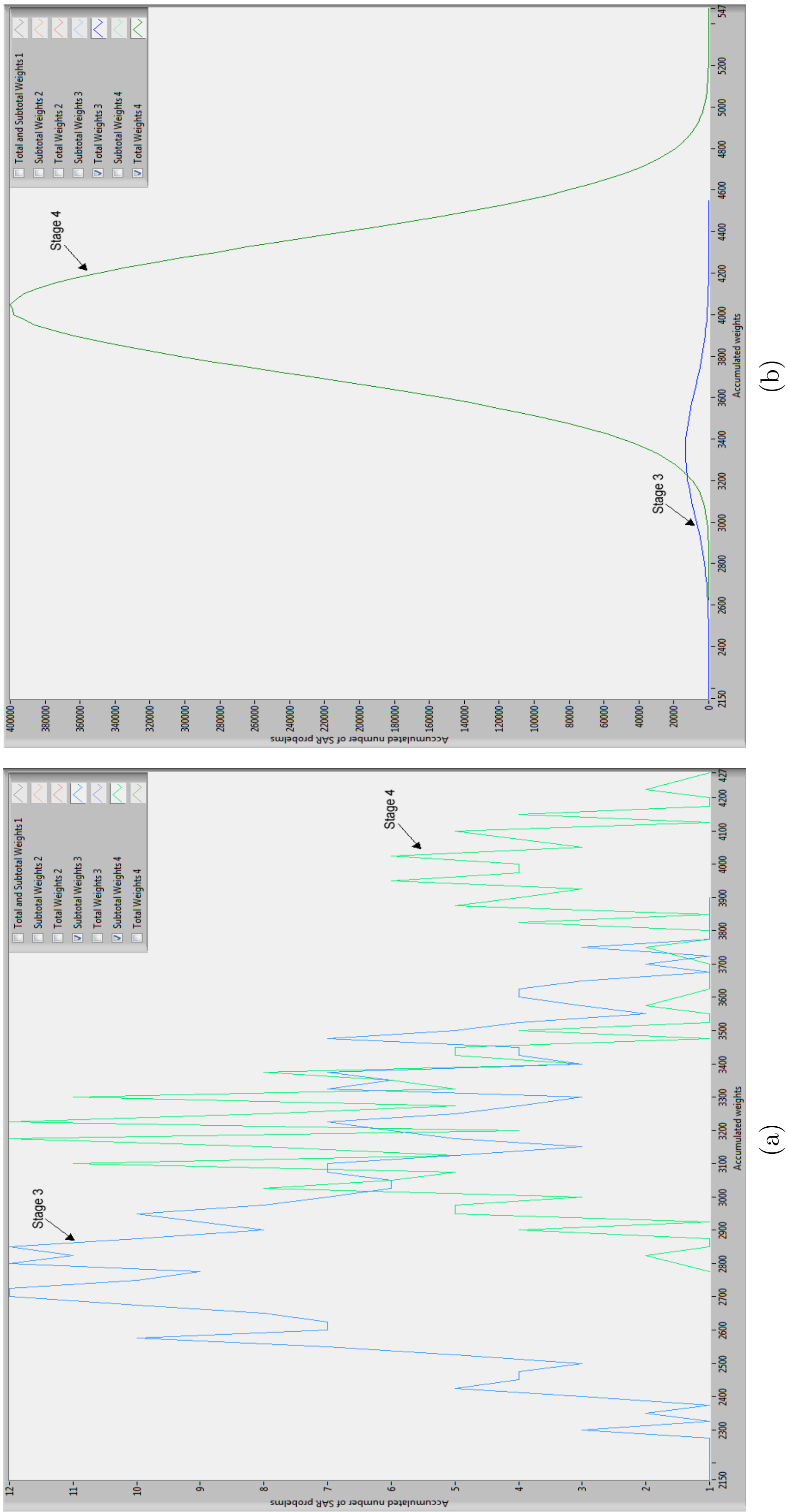


Figure 7.15: Subtotal and total SAR problem space between stages 3 and stage 4. (a) Subtotal space between stage 3 and stage 4. (b) Total space between stage 3 and stage 4.

The increases and decreases of the generated SAR problems in the subtotal space are due to the number of selected SAR problems and variables to analyse at each stage. Seven variables were selected in stage 1. There were six binary and one ternary resulting in 192 generated SAR problems. In stage 2, there are six binary variables to analyse and fourteen SAR problems were selected in stage 1. Hence, the combination and aggregation processes resulted in 896 generated SAR problems. The large number of 896 in stage 2 is due to the large number of SAR problem selected in stage 1 (i.e. 14). However, this resulted in a wide variety of variables' options through the analysis. In stage 3, there are five binary variables to analyse and eleven SAR problems were selected in stage 2. Hence, the combination and aggregation processes resulted in 352 generated SAR problems. Because the number of selected SAR problems and variables to analyse is smaller than in the stage 2, the number is comparatively smaller. In a similar way, the generated SAR problems for stage 4 keep the decreasing trend. In stage 4, there are five binary variables to analyse and eleven SAR problems were selected in stage 3. Hence, the combination and aggregation processes resulted in 224 generated SAR problems.

The total number of generated SAR problems can be calculated by adding together the number of generated SAR problems at each stage. These numbers are determined by the number of selected SAR problems at the previous stage and the multiplication of all the number of options of each variable from the group of variables at the current stage. The total number of generated SAR problems at each stage can be calculated with Equation 6.1 at each stage as follows:

- Stage 1: $o_{1.1} * o_{1.2} * o_{1.3} * o_{1.4} * o_{1.5} * o_{1.6} * o_{1.7} = 2 * 2 * 3 * 2 * 2 * 2 * 2 = 192$
- Stage 2: $o_{2.1} * o_{2.2} * o_{2.3} * o_{2.4} * o_{2.5} * o_{2.6} * S_1 = 2 * 2 * 2 * 2 * 2 * 2 * 14 = 896$
- Stage 3: $o_{3.1} * o_{3.2} * o_{3.3} * o_{3.4} * o_{3.5} * S_2 = 2 * 2 * 2 * 2 * 2 * 11 = 352$
- Stage 4: $o_{4.1} * o_{4.2} * o_{4.3} * o_{4.4} * o_{4.5} * S_3 = 2 * 2 * 2 * 2 * 2 * 7 = 224$

By adding these numbers the total number of generated SAR problems through all the stages result in 1,664. The subtotal and total SAR problem spaces from the stages 1 and 2 can be observed in 7.13a and 7.13b, respectively; from the stages 2 and 3 can be observed in 7.14a and 7.14b, respectively; and from the stages 3 and 4 can be observed in 7.15a and 7.15b, respectively. The evolution of the subtotal SAR problem space between all the stages (i.e. current and next stages) can be observed through Figures 7.13a, 7.14a and 7.15a. Whilst the evolution of the total SAR problem space between all the stages can be observed through Figures 7.13b, 7.14b and 7.15b. The following metrics can be determined from the intrastage analysis:

- **The total saved space**: the total number of generated SAR problems is 1,664 and the total possible SAR problems is 12,582,912. Hence, the total saved space is 99.986%.

- **The calculation effort waste**: is the ratio between the 1,664 generated SAR problems and the 36 selected SAR problems at the four stages. Thus the calculation effort waste is a 2.16% of the generated SAR problems.

- **The generation effort waste**: is the ratio between the 1,664 generated SAR problems and the 4 complete SAR problems. Therefore, the generation effort waste is a 0.024% of the generated SAR problems.

7.5 Formulation of complete SAR problems selected with the DMM

The four selected SAR problems in stage 4 of the DMM are summarised in Section 7.4.4. These four problems are complete SAR problems that contain variables' options from the four stages (i.e. variables from the complete SAR problem case study). The decision variables considered in the SAR problem case study were presented in Tables 7.1 and 7.2. Also, assumptions for this case study were presented in Section 7.1. The SAR problem case study simplified by these assumptions is represented with the proposed notation in Appendix C.

In this section, the four problems selected at the last stage are formulated, which brings this thesis to a sucessful conclusion. The first and second selected SAR problems are very similar, as well as the third and fourth selected SAR problems. Therefore, the formulations are presented in two subsections. First subsection presents formulation for first and second, whilst second subsection presents the formulation for the third and fourth SAR problems. The formulation make use of notation that was presented in Chapter 5. Moreover, the four selected SAR problems are represented with the notation in Appendix C.

7.5.1 SAR problem types 1 and 2

Formulation

The SAR problem type 1 is formulated as follows: It is considered a square factory defined by its shape σ^f with a centroid at the location $(x^{cent,f}, y^{cent,f})$. The problem is formulated in discrete space by squares of the same size. The problem is formulated in time in positive discrete values. Hence, the angle of orientation of all objects (i.e. robots, shelves and warehouses) in the factory are neglected.

The uncertainty of values of the system is neglected. The problem is considered as a central system that determines tasks to robots allocations, scheduling of the tasks, and positions to manufacture and paths for robots to reach these positions. The environment is considered as fully known (i.e. structured). The problem considers a single objective to optimise. This might be the minimisation of production time or the maximisation of profits if products get assigned at profit and production costs are known.

A single warehouse $\mathbf{w}_1$ is considered, which has unique raw material m. There are $\mathcal{S}^m_{N_{shelves}}$ shelves at the warehouse, which contain the same raw material. This means that all products require the same type of raw material, but in different quantities q_i per each product p_i. The warehouse can cover several squares but their final shape must be a square. The warehouse is defined by its shape σ^w. The location of the warehouse within the factory is represented by its center point $(x^{cent,w}, y^{cent,w})$ and this location does not change during production.

Raw materials loading $(\mu_h^{s\text{-}m})$ and unloading times $(\eta_h^{s\text{-}m})$ to any shelf (s_h^m) in the warehouse do not depend on the type of raw material m_i or the robot k that executes the loading and unloading operation. Products loading $(\mu_h^{s\text{-}p})$ and unloading times $(\eta_h^{s\text{-}p})$ to any shelf (s_h^p) in the warehouse do not depend on the type of product p_i or the robot k that executes the loading and unloading operation. Materials supplying time $(\xi_h^{s\text{-}m})$ and products picking up time $(\iota_h^{s\text{-}p})$ to any shelf (h) are independent of the robot r_k and the type of raw material m_i and product p_i that is supplied or picked up respectively. The raw material m_i corresponds to the product p_i. Raw materials and products shelves do not have a limit. Thus, it is not necessary to monitor the current raw materials and products amount.

Multiple products $N_{products}$ that have different demands (ω_i^p) are considered. Also, products have a single start time $(\delta^{start\text{-}p})$ and deadline $(\delta^{end\text{-}p})$ for all the products. Each product can be manufactured through multiple production plans (Π), and the processing time of each operation o_j required by each product are independent of the robot r_k that executes the operation (τ_j^{o,p_i}). The required quality degree (g_j^o) and the surface quality degree (ϖ_j^o) of all operations of all products is homogeneous.

Multiple robots (N_{robots}) are considered to execute all the manufacturing operations. Machines are not considered. Robots have the same area size defined by their shape (σ^r). The current position of any robot r_k is given by its centroid $(x_k^{cent,r}(t), y_k^{cent,r}(t))$. Robots have a unique configuration of wheels (l^r), have a homogeneous value of maximal battery duration (β^r) and a homogeneous value of maximal arm robot payload $(\gamma^{r,arm})$. Robots have a homogeneous maximal velocity $(v^{r,max})$, acceleration $(\psi^{r,max})$ and mobile platform payload $(\gamma^{r,platform})$. Robots have heterogeneous arm robot kinematic configurations (κ_k^r). Current velocity $(v_k^r(t))$, acceleration $(\psi_k^r(t))$, status $(\varsigma_k^r(t))$ and battery charge $(\beta_k^r(t))$ of all robots $\mathcal{R}$ are considered for the central system to solve the SAR problem.

Each robot has a group of tools ($D_k^r = \mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N_{tools}}$), and each group is different. These groups have the same maximal number of interchangeable tools ($d_{N_{tools}}$). Robots can interchange the tools on their own, but they cannot interchange tools among them or within a tool depot. Hence, tools are not wearable (i.e. they can be used as long as required, without requiring replacement). Tool setup times (χ^d) and removal times (ζ^d) are equal for all robots but different between each other.

The formulation for the SAR problem type 2 is identical to SAR problem type 1 except for three assumptions on the number of warehouses, the number of process plans to produce each product and the number of objectives to optimise. For the SAR problem type 2 multiple optimisation objectives are considered. These are the production time, and the profits of all products. A single process plan (Π_1^p) is considered for each product p_i. Multiple warehouses ($\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$) are considered.

7.5.2 SAR problem types 3 and 4

The SAR problems types 3 and 4 are similar to each other, but disimilar with the SAR problems types 1 and 2. These two SAR problems are formulated in the following paragraphs.

Formulation

The SAR problem type 3 is formulated as follows: A square factory is considered that is defined by its shape σ^f with a centroid at the location ($x^{cent,f}, y^{cent,f}$). The problem is formulated in discrete space by squares of the same size. The problem is formulated in time in positive discrete values. Hence, the angle of orientation of all objects (i.e. robots, shelves and warehouses) in the factory are neglected.

The uncertainty of values of the system is neglected. The problem is considered as a central system that determines tasks to robots allocations, scheduling of the tasks, and positions to manufacture and paths for robots to reach these positions. The environment is considered as fully known (i.e. structured). The problem considers a single objective to optimise. This might be the minimisation of production time or the maximisation of production profits.

A single warehouse $\mathbf{w}_1$ is considered. There are $\mathcal{S}_{N_{shelves}}^m$ shelves at the warehouse, which contain mixed raw materials $m_h^{s\text{-}m}$ for any shelf (s_h^m) in the warehouse. Products require different types of raw materials in different quantities (q_i) per each product p_i. The warehouse can cover several squares but their final shape must be a square. The warehouse shape is defined by its shape σ^w. The location of the warehouse within the factory can change during production and is represented by its center point $(x^{cent,w}(t), y^{cent,w}(t))$.

Raw materials loading $(\mu_h^{s\_m}(r_j)(m_i))$ and unloading times $(\eta_h^{s\_m}(r_j)(m_i))$ to any shelf (s_h^m) in the warehouse depend on the type of raw material m_i or the robot r_k that executes the loading and unloading operation. The raw material m_i corresponds to the product p_i. Products loading $(\mu_h^{s\_p}(r_j)(p_i))$ and unloading times $(\eta_h^{s\_p}(r_j)(p_i))$ to any shelf (s_p^m) in the warehouse do not depend on the type of product p_i or the robot r_k that executes the loading and unloading operation. Materials supplying time $(\xi_h^{s\_m}(r_j)(m_i))$ and products picking up time $(\iota_h^{s\_p}(r_j)(p_i))$ to any shelf (h) are independent of the robot k and the type of raw material m_i and product p_i that is supplied or picked up respectively. Raw materials and products shelves do not have a limit. Thus, it is not necessary to monitor the current raw materials and products amount.

Multiple products $N_{products}$ that have different demands (ω_i^p) are considered. Also, there is a start time $(\delta_i^{start\_p})$ and deadline $(\delta_i^{end\_p})$ per each product. Each product can be manufactured through multiple production plans (Π), and the processing times of each operation o_j depend on the robot r_k that executes the operation $(\tau_j^{o,p_i}(r_k))$. The required quality degree (g_j^o) and the surface quality degree (ϖ_j^o) of all operations of all products is homogeneous.

Multiple robots (N_{robots}) are considered to execute all the manufacturing operations. Machines are not considered. Robots have the same area size defined by their shape σ^r. The current position of robots is given by its centroid $(x_k^{cent,r}(t), y_k^{cent,r}(t))$. Robots have mixed configurations of wheels (l_k^r), have a heterogeneous value of maximal battery duration (β_k^r) and a heterogeneous value of maximal arm robot payload $(\gamma_k^{r,arm})$. Robots have a homogeneous maximal velocity $(v^{r,max})$, acceleration $(\psi^{r,max})$ and mobile platform payload $(\gamma^{r,platform})$. Robots have heterogeneous arm robot kinematic configurations (κ_k^r). Current velocity $(v_k^r(t))$, acceleration $(\psi_k^r(t))$, status $(\varsigma_k^r(t))$ and battery level $(\beta_k^r(t))$ of all robots $\mathcal{R}$ are considered for the central system to solve the SAR problem.

Each robot has a group of tools $(D_k^r = \mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N_{tools}})$, and each group is different. These groups have a mixed maximal number of interchangeable tools $(d_{N_{tools}})$. Robots can interchange the tools on their own, but they cannot interchange tools among them or within a tool depot. Hence, tools are not wearable (i.e. they can be used as long as required, without requiring replacement). Tool setup times $(\chi_{k_{tool}}^d)$ and removal times $(\zeta_{k_{tool}}^d)$ for any tool k_{tool} are unequal for any robots k and different between each other.

The formulation for the SAR problem type 4 is identical to SAR problem type 3 except for three assumptions on the number of warehouses, the number of process plans to produce each product and the number of objectives to optimise. For the SAR problem type 4 multiple optimisation objectives are considered. These are the production time, and the profits of all products. A single process plan (Π_1^p) is considered for each product p. Multiple warehouses ($\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$) are considered.

7.6 Concluding remarks

Decision Making Methodology (DMM) was applied over a representative case study of the SAR problem. The DMM proved to be effective at reducing the number of reviewed SAR problems. Four problems were selected at the final stage.

The four selected SAR problems were represented with the notation proposed in Chapter 4 and formulated. The first two problems are highly solvable and they describe realistic scenarios for the production with Self-Reconfigurable Manufacturing Systems (S-RMSs). The last two problems are highly real, and they correspond to relaxations of variables from the first two problems.

Therefore, work for future generations of PhDs should focus on solving the first two problems first, and then start relaxing either individual or groups of variables to increase the realism of the problem. It is noteworthy that this will increase the complexity of the problem. The increase of complexity of the problems will require more detailed input data (e.g. production requirements, available resources, factory design and its operation) for the complex SAR problems.

Chapter 8

Conclusions and future work

8.1 Summary and conclusions

Novel contributions of this thesis were made in the fields of manufacturing, robotics and operations research. This thesis has three main contributions that link the three fields. These contributions are:

- The novel framework INTelligent REconfiguration for a raPID production change (INTREPID), which aim is to offer global High Value Manufacturing (HVM) services, was proposed in Chapter 3. INTREPID consists of a user interface and communications platform, a network of reconfigurable facilities (i.e. Reconfigurable Manufacturing Centres (RMCs)), Self-Reconfigurable Manufacturing Systems (S-RMSs) and a job allocation system. The novelty of INTREPID is the wide number of technologies in a single comprehensive framework. Within INTREPID, the novel contribution is the proposal of highly flexible and autonomous manufacturing system that employs mobile robots such as mobile manipulators to perform manufacturing and transportation tasks (i.e. S-RMSs). This contribution addresses the second objective of this thesis, see Chapter 1. INTREPID was presented in the publication [29].

- The proposal of an inclusive production planning problem that considers mobile manufacturing resources within a single reconfigurable factory, was proposed through Chapters 4 and 5. Due to the inclusion of the scheduling, positions assigning (i.e. machine layout) and vehicle routing problems, the novel production planning was called the Scheduling, positions Assigning and Routing problem (SAR) problem. The SAR problem refers to the job allocation system considering the S-RMS within a single reconfigurable factory. The novelty of the SAR problem is the consideration of mobile robots such as mobile manipulators (i.e. S-RMS) to perform all the manufacturing and transportation tasks. A notation for the SAR problem was proposed. This contribution addresses the third objective of this thesis, see Chapter 1.

- A novel Decision Making Methodology (DMM) to select a SAR problem that is realistic but solvable with optimisation methods was proposed in Chapter 6. The novel DMM helps to select one or a few alternatives from very large number of alternatives (i.e. millions). The DMM was implemented in LabVIEW and evaluated with a case study of the SAR problem in Chapter 7. This contribution addresses the fourth and fifth objectives of this thesis, see Chapter 1. A partial version of the methodology and its implementation was introduced in the publication [30].

A summary and conclusions is presented in the following paragraphs. High Value Manufacturing (HVM) refers to the creation of innovative products with cutting-edge technology (i.e. added value through the use of knowledge). The main challenge for HVM refers to the rapid adjustment of the production capacity to produce the required demand of innovative products (i.e. high value products, HV products). This means, increasing or decreasing production volume from one-off products to low production volume for HV products. The focus of this thesis is the main HVM challenge. An analysis of manufacturing paradigms and their different types of added value was performed in Chapter 2 in order to identify the capacity of these paradigms to address the main HVM challenge. An analysis of current manufacturing systems and layouts identified the lack of approaches to manage rapid changes in the demand of HV products (i.e. main HVM challenge). Traditional manufacturing systems are classified in dedicated, flexible and reconfigurable systems. These type of systems focus on a finite range of production volume, and are only capable of producing a limited number of products or group of products. Also, these manufacturing systems have fixed and static manufacturing layouts, such as product-oriented, process-oriented and manufacturing cells (hybrid of product- and process-oriented). As conclusions of these analyses, traditional manufacturing systems and layouts cannot change rapidly according to the changes in the demand of HV products.

Therefore, in order to manage rapid changes in production demands of HV products (i.e. main HVM challenge), the novel framework INTelligent REconfiguration for a raPID production change (INTREPID) was proposed in Chapter 3. INTREPID consists of a user interface and communications platform, a network of globally distributed Reconfigurable Manufacturing Centres (RMCs) (where each cluster consists of several connected factories), a job allocation system, and Self-Reconfigurable Manufacturing Systems (S-RMSs) in each RMC. INTREPID was motivated by largely accepted and proved paradigms and concepts, such as cloud manufacturing, networked manufacturing, industry 4.0, reconfigurable manufacturing systems, and cloud robotics. These paradigms and concepts make use of cutting-edge technology to manage dynamic demands of HV products. State of the art literature on approaches to manage rapid changes in products demand was analysed, and the best practices and desired characteristics of these approaches were identified. In order to address the HVM challenge, the desired characteristics and best practices were included in INTREPID and its parts. Existing frameworks to manage highly dynamic demands such as cloud manufacturing and NetMan focus on technologies for data transfer and management, and mechanisms for an agile formation and coordination of factories respectively. Despite these efforts, there is a lack of a comprehensive framework to manage highly dynamic demands.

The novelty of INTREPID is in the proposal of a single comprehensive framework and its four parts, which together are capable of managing highly dynamic demands for HV products. A novel contribution made in INTREPID is its highly flexible and autonomous manufacturing system (i.e. S-RMSs), which operate in an unmanned area that facilitates the reconfiguration and movement of resources (i.e. RMCs and reconfigurable factories). In contrast to traditional manufacturing systems with fixed and static machines, S-RMSs make use of mobile manufacturing resources such as mobile robots and movable machines. Consequently, the flexibility of the S-RMS is greater than traditional systems such as dedicated, flexible and reconfigurable manufacturing systems. The aim of INTREPID is to offer global HVM services. For this aim, the user interface and communication platform serves to collect product data, and also to negotiate and to collaborate during the product design stage. Once the product design is known, the job allocation system determines the best possible factories or RMCs to perform a job by considering the complexity of the production requirements and the status of the available S-RMSs at each factory. In summary, a comprehensive framework to offer global HVM services was proposed and its parts and operation were described in Chapter 3. As conclusions from this chapter, rapid changes in the demand of HV products can be managed with Self-Reconfigurable Manufacturing Systems (S-RMSs), reconfigurable factories and job allocation systems that use real-time data and optimisation methods.

Several challenges for INTREPID's implementation were identified. However, the main challenges were: 1. development and implementation of a user interface and communications platform; 2. design and construction of factories and RMCs that facilitate the movement of mobile manufacturing resources; 3. Design and construction of mobile manufacturing resources (S-RMS); 4. Research, development and implementation of job allocation mechanisms that consider the design and operation of RMCs, factories and S-RMSs. The job allocation challenge was the focus of the rest of the thesis. The job allocation system has three levels of scope: 1. Network type, which involves multiples S-RMSs and coordinating supply chains; 2. A single RMC, which involves multiples S-RMSs and the network; and 3. A single factory, which involves a S-RMS. A job allocation system for a single factory refers to the production planning problem. This problem consists of determining production plans with the available resources in order to address the incomming production orders and their requirements. The production planning problem considering a single factory is the minimal problem.

For traditional manufacturing systems (i.e. with static machines), the production planning problem involves three main independent problems that are solved in sequence. These are: 1) the scheduling; 2) the machine layout; and 3) the vehicle routing problems. However, the use of S-RMS (i.e. mobile resources) increases the flexibility by allowing changing of the layout for each manufacturing task of each product from the current production orders. Due to this increased flexibility, a new problem to simultaneously solve these problems was proposed. This novel problem was called the Scheduling, positions Assigning and Routing problem (SAR). In brief, the SAR problem consists of determining allocations and schedules of tasks to machines or robots, the positions in the factory to perform these operations and the routes for machines or robots to reach their positions in the layout. In order to find the best possible solutions to the SAR problem, it is necessary to use optimisation methods. The novelty of the SAR problem is the consideration of mobile resources for manufacturing, in contrast to existing approaches to determine production planning with static and fixed manufacturing resources.

The constituent problems of the SAR problem (i.e. scheduling, machine layout and vehicle routing) and related problems from the field of robotics (i.e. multi robot task allocation and motion planning) were analysed in Chapter 4. Elements, characteristics and assumptions that characterised each of these problems were identified and analysed. Existing optimisation objectives, taxonomies and notations were identified for each of the five problems. Elements, characteristics, assumptions, optimization objectives, taxonomies and notations were analysed in order to understand their application to the SAR problem. The lack of an comprehensive notation for the scheduling, machine layout and vehicle routing was identified in Chapter 4. Therefore, a novel notation for the SAR problem was proposed in Chapter 5. Also in Chapter 5, the core elements and characteristics of the SAR problem were defined, and general assumptions on these elements were provided.

In Chapter 5, the notation for elements, characteristics, and assumptions were classified by the types of aspects of the SAR problem. These types are 1. the factory design and its operation (e.g. warehousing and manufacturing zones); 2. the production requirements (e.g. products, operations); and 3. the manufacturing resources (e.g. robots, machines, tools). These three types are natural to the formulation of the SAR problem, and therefore, they are called fundamental. An additional type of characteristics and assumptions with the purpose of formulating the SAR problem with optimisation methods was called systems variables. Examples of system variables are whether the problem should be formulated in continuous or discrete space and time, or whether the environment should be treated as structured or unstructured. In contrast to the fundamental variables (e.g. production requirements), the system variables are auxiliary to the formulation of the SAR problem with optimisation methods. As conclusions from Chapters 4 and 5, the proposed SAR problem and its notation are able to represent the production planning problem with the use of S-RMS for a single factory.

With the purpose of solving a realistic SAR problem variant, it is necessary to formulate a problem that is realistic to industrial scenarios (i.e. realism) but that can be solved with optimisation methods (i.e. solvability). However, the combination of SAR problem elements, characteristics and assumptions can result in millions of possible SAR problem variants (i.e. SAR problem space). Due to the large number of possible SAR problem variants, it was complex to decide which combinations of elements, characteristics and assumptions to formulate and solve with optimisation methods. Therefore, it was proposed to select a realistic but solvable SAR problem with decision making methods from the field of concept generation and selection. For this purpose, elements, characteristics and assumptions of the SAR problem were considered as decision variables on whether to include or not the elements and characteristics and under which assumptions.

A review of relevant concept generation and selection methods as well as decision making methods was made in Chapter 6. These methods rely heavily on pairwise comparisons between the alternatives and a reference or among the alternatives per each criterion. For a vast number of alternatives, such as in the SAR problem space (i.e. millions of variants), it was intractable to apply existing decision making methods to select an appropiate SAR problem. Therefore, a novel decision making methodology (DMM) was proposed in Chapter 6. The DMM is based on the controlled convergence method. Instead of analysing the complete group of variables, the DMM works by analysing subgroups of decision variables (i.e. elements, characteristics and assumptions) through multiple stages. Partial SAR problems, generated with the subgroups of variables, are analysed and selected by expert personnel from fields such as robotics and optimisation. The selected partial SAR problems are aggregated to the generated partial SAR problems from the next stage and so on. The generation of partial SAR problems and their analysis continues iteratively until all the subgroups of variables are analysed. The first group of partial SAR problem represent the core SAR problem, and groups of variables at the following stages add level of detail and complexity to these core problems. The DMM steps and auxiliary graphs for SAR problem space exploration and selection were described thoroughly in Chapter 6. The novelty of the DMM is its capability of handling millions of alternatives. In the proposed DMM, there is not direct comparison of all the alternatives, in contrast to existing selection methods such as decision matrix methods, outranking methods and distance-based methods [23],[24],[25],[26]. Instead subgroups of partial alternatives are compared and aggregated through multiple stages in the DMM.

The methodology was implemented and tested with an exemplar case study in Chapter 7. The implemented methodology was called a Decision Making Support System (DMSS). The case study was analysed in four stages. In the first stage, fourteen partial SAR problems were selected, in the second stage there were eleven selections, in the third stage, seven selections and finally, four complete SAR problems were selected. These complete SAR problems contain variables from the four stages. The generated partial SAR problems at each stage were analysed and a few SAR problems were selected for the next stage. The selection of the SAR problems was based on their realism to industrial scenarios, their feasibility to be solved with optimisation methods, and the interest to explore the SAR problems in combination with variables of the next stage. Some of the selected SAR problems and their implications to be formulated and solved with optimisation methods were discussed. The application of the methodology result in 1,664 generated and reviewed SAR problems and 36 selected SAR problems selected through the four stages. Four complete SAR problems selected at the last stage from a total SAR problem space of millions (i.e. 12,582,912 SAR problems). Finally, these four complete SAR problems were represented with the notation proposed in 5 and formulated in order to successfully conclude this thesis. As conclusions from Chapter 6 and 7, the DMM proved to yield a good performance and facilitate exploring the SAR problem space and selecting a few SAR problems that are realistic but solvable.

8.2 Future work

The work presented in this thesis consists of: the framework INTREPID and its associated Self-Reconfigurable Manufacturing Systems (S-RMSs), a production planning problem with the use of a S-RMS (SAR problem), and a methodology that helps select a SAR problem. Although, INTREPID is based on a thorough analysis of the most recent literature, new innovations and research proposals can be included in the framework. Also, new challenges for the implementation of INTREPID can be managed with the use of these new innovations and research proposals. The proposed SAR problem is comprehensive and inclusive of variables from problems such as scheduling, machine layout, vehicle routing, multi-robot task allocation and path planning. However, it is possible to extend the literature research to more related problems and therefore, improve the detail of the SAR problem. The methodology is a tool that helps to select a SAR problem that is realistic to industrial scenarios but solvable with optimisation methods. However, because the methodology is novel, additional steps or functionalities might be required to use on applications such as product and system design. The proposed future work is divided into three main works developed in this thesis.

INTREPID

Future work for INTREPID should focus on implementation of INTREPID' parts and the application of INTREPID in industrial case studies. Recommendations for future work are:

- The research and development of the user interface and communications platform must guarantee real-time and robust communications. The use of cloud manufacturing communication technologies was proposed in INTREPID, but other approaches are encouraged to be developed and implemented.
- It is recommended to research on and implement algorithms to locate RMCs based on analyses of RMCs' access to resources, expected types of products to manufacture and expected clients to distribute these products.
- Although an octagon design for the reconfigurable factories and diamond patterns for the RMCs was proposed in INTREPID, other designs should be investigated. However, it is recommended to design any RMC and factory with the objective to facilitate the movement of resources (i.e. robots and machines) across factories and even RMCs. Also, factories' design should be modular to facilitate the addition of more factories (i.e. scalability).
- It is recommended to design and implement auxiliary systems such as communications and sensing in order to facilitate the autonomous movement of mobile robots and movable machines across the factories and RMCs. In addition, it is recommended to design, simulate and implement control systems for the mobile robots and movable machines that take advantage of these communications and sensing systems. Thus, robots, machines and auxiliary systems should be designed concurrently. Also, the design of robots and machines should guarantee an autonomous, rapid and accurate calibration at the arrival to a manufacturing position.
- The job allocation system considers the design and operation of factories and the S-RMS in order to determine feasible and a optimal solutions. Factories

and S-RMS should be designed and operated in order to facilitate obtaining feasible and a optimal solutions for the job allocation problem, whilst still considering a realistic operation of the factories (i.e. dynamic and uncertain production demand).

- Sustainability challenges within a cloud manufacturing and industry 4.0 environment should be considered and addressed. Examples of these challenges are CO_2 emissions generated from transporting raw materials and products around the world, scaling of the production capacity for seasonal and large products, etc.

The SAR problem

INTREPID's job allocation system focused on a single factory correspond to the SAR problem. Recommendations for future work on the SAR problem are provided in the next paragraphs.

- The combination of decentralised and centralised allocation systems for the job allocation system was proposed in INTREPID. Each factory has a centralised system, but the network of RMCs work as a decentralised systems. This approach should be implemented, but other approaches should be investigated and implemented for comparison.

- Once adequate SAR problems were selected and formulated, it is necessary to model and solve these problems with optimisation methods. It is recommended to initially model and solve the simplest two problems (i.e. SAR problems type 1 and 2), and then relax the constraints of the decision variables until the two complex problems (i.e. SAR problems type 3 and 4) are modelled and solved. The relaxation of constraints can occur either one by one or in groups that facilitates the problem modelling and solving.

- For the purpose of solving the selected SAR problems, instances of input data are required. Consequently, it is necessary to propose input data to evaluate the four variants of the selected SAR problems. Also, it is recommended to perform sensitivity analysis of the initial conditions (e.g. initial locations of robots and their carrying tools) of data instances. It is expected a significant influence of the initial conditions on the solutions.

- Practical future work should focus on real world implementation and testing of the four selected SAR problems with mobile robots such as mobile manipulators and movable machines in industrial environments and with realistic production demand. It is recommended to evaluate different types of products (e.g. large components, aircrafts, ships) with different production requirements (i.e. deadlines and demands).

- The SAR problem corresponds to the job allocation problem considering a S-RMS within a single factory. However, the SAR problem should be extended to several factories (i.e. a complete RMC), and several RMCs (i.e. the complete RMCs network). These novel problems should consider different factory and RMC designs and different combinations of communications and sensing devices between factories and S-RMSs.

- The proposed notation is also specific to the SAR problem (i.e. a S-RMS within a single factory). Consequently, the notation should be extended to multiple factories and multiple RMCs.

Methodology and its implementation

Future work for the decision making methodology and its implementation (Decision Making Support System (DMSS)) should focus on:

- Developing and implementing more mechanisms to explore and graphs to visualise the problem space. These mechanisms and graphs should specially focus on reducing user fatigue.

- The methodology should be adapted to manage multiple criteria. For this purpose, additional mechanisms should be investigated and implemented Also, the data visualisation mechanisms for multi criteria should be implemented in the DMSS.

- Research on outranking methods, which result in an order of preferences rather than a single selection should be investigated. Relevant characteristics or functionalities from the outranking methods should be implemented in the DMSS. For example, the current methodology uses an aggregation function (i.e. sum of options' weights), which make it easy to implement and understand, but limits its application to a single criterion. However, more functions such as the six functions proposed in the method Preference Ranking Organization METHod for Enrichment Evaluations (PROMETHEE) [248] should be investigated.

- The methodology should be tested on more applications such as the design of complex products and systems (e.g. aircrafts, nuclear plants, ships) with experts and experienced engineers from relevant industrial sectors. From the application of the methodology, experts and engineers should propose new processes, functionalities or graphs to improve the methodology and facilitate its application on complex problems.

Appendices

Appendix A

Selected SAR problems in each stage

This appendix presents results of the evaluation of the proposed decision making methodology on a SAR problem case study in Chapter 7. In specific, the SAR problem case study was presented in Section 7.1. The selected SAR problems in the four stages are summarised in the following tables:

- Stage 1 in Tables A.1 and A.2
- Stage 2 in Tables A.3 and A.4
- Stage 3 in Table A.5
- Stage 4 in Table A.6

Table A.1: Part 1 of the selected SAR problems in stage 1. This table shows the first seven problems. The variables analysed in stage 1 are space, time, shape, processing time of operations dependent on the type of robot (O.PTR), transport time of parts dependent on the types of part and robot (PTTDPR), number of warehouses (W.Number), and robots accuracy (R.Accuracy). The three reasons to select SAR problems are feasibility of the problems to be solved, realism of the problems to industrial scenarios and interest of combining problems with variables of the next stage. This table shows six SAR problems selected for their feasibility to be solved (positions 1, 3, 19, 40, 41 and 48), one for their realism to industrial scenarios and for the interest to explore in combination with options of the next stage (position 55).

Selected SAR problems in stage 1. Part 1							
Variables	**Options of variables**						
Space	Disc.	Disc.	Cont.	Cont.	Cont.	Disc.	Cont.
Time	Disc.	Disc.	Disc.	Cont.	Disc.	Disc.	Disc.
Shape	Squa.	Squa.	Rect.	Squa.	Rect.	Squa.	Poly.
O.PTR	Indep.	Indep.	Indep.	Indep.	Indep.	Dep.	Indep.
PTTDPR	Indep.	Indep.	Indep.	Indep.	Indep.	Dep.	Indep.
W.Number	Sing.	Mult.	Sing.	Sing.	Mult.	Sing.	Sing.
R.Accuracy	Homog.	Homog.	Homog.	Homog.	Homog.	Hetero.	Homog.
Weight	825	925	1,075	1,175	1,175	1,175	1,225
Position	1	3	19	40	41	48	55

Table A.2: Part 2 of selected SAR problems in stage 1. This table shows the remaining seven problems. The variables analysed in stage 1 are space, time, shape, processing time of operations dependent on the type of robot (O.PTR), transport time of parts dependent on the types of part and robot (PTTDPR), number of warehouses (W.Number), and robots accuracy (R.Accuracy). The three reasons to select SAR problems are feasibility of the problems to be solved, realism of the problems to industrial scenarios and interest of combining problems with variables of the next stage. This table shows two SAR problems selected for their feasibility to be solved (positions 73 and 81), two only for their realism to industrial scenarios (positions) and four for the interest to explore in combination with options of the next stage and their realism to industrial scenarios (position 73, 93, 173 and 185).

Selected SAR problems in stage 1. Part 2							
Variables	**Options of variables**						
Space	Cont.	Disc.	Cont.	Cont.	Cont.	Cont.	Cont.
Time	Cont.	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.
Shape	Squa.	Squa.	Poly.	Rect.	Rect.	Poly.	Poly.
O.PTR	Indep.	Dep.	Indep.	Dep.	Dep.	Dep.	Dep.
PTTDPR	Indep.	Dep.	Indep.	Dep.	Dep.	Dep.	Dep.
W.Number	Mult.	Mult.	Mult.	Sing.	Mult.	Sing.	Mult.
R.Accuracy	Homog.	Hetero.	Homog.	Hetero.	Hetero.	Hetero.	Hetero.
Weight	1,275	1,275	1,325	1,425	1,525	1,575	1,675
Position	73	81	93	133	164	173	185

Table A.3: Part 1 of selected SAR problems in stage 2. This table shows the first six problems. The variables analysed in stage 2 are system control (S.Control), objectives to optimise (Objectives), robots area (R.Area), products' start times (P.StartTimes) and deadlines (P.Deadlines), and warehouses content (W.Content). The reasons for selecting the SAR problems are feasibility to be solved the realism to industrial scenarios and interest of combining problems with variables of the next stage. This table shows four SAR problems selected for their feasibility to be solved (positions 1, 15, 20 and 21), two for their realism to industrial scenarios (positions 43 and 86) and four for the interest to explore in combination with options of the next stage (positions (15, 20, 43 and 86). The problems are repeated in some of the reasons (positions 12, 20 and 43).

	Selected SAR problems in stage 2. Part 1						
	Variables	**Options of variables**					
Stage 1	Space	Disc.	Cont.	Disc.	Disc.	Disc.	Disc.
	Time	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.
	Shape	Squa.	Rect.	Squa.	Squa.	Squa.	Squa.
	O.PTR	Indep.	Indep.	Indep.	Indep.	Indep.	Indep.
	PTTDPR	Indep.	Indep.	Indep.	Indep.	Indep.	Indep.
	W.Number	Sing.	Sing.	Mult.	Mult.	Sing.	Mult.
	R.Accuracy	Homog.	Homog.	Homog.	Homog.	Homog.	Homog.
Stage 2	S.Control	Cent.	Cent.	Decent.	Cent.	Decent.	Decent.
	Objectives	Sing.	Sing.	Sing.	Mult.	Mult.	Mult.
	R.Area	Equ.	Equ.	Equ.	Equ.	Equ.	Equ.
	P.StartTimes	Sing.	Sing.	Sing.	Sing.	Sing.	Sing.
	P.Deadlines	Sing.	Sing.	Sing.	Sing.	Sing.	Sing.
	W.Content	Uni.	Uni.	Uni.	Uni.	Uni.	Uni.
	Weight	1,600	1,850	1,875	1,875	1,950	2,050
	Position	1	15	20	21	43	86

Table A.4: Part 2 of selected SAR problems in stage 2. This table shows the remaining five problems. The variables analysed in stage 2 are system control (S.Control), objectives to optimise (Objectives), robots area (R.Area), products' start times (P.StartTimes) and deadlines (P.Deadlines), and warehouses content (W.Content). The reasons for selecting the SAR problems are feasibility to be solved the realism to industrial scenarios and interest of combining problems with variables of the next stage. This table shows five SAR problems selected for their realism to industrial scenarios (positions 116, 239, 527, 559 and 791) and three for the interest to explore in combination with options of the next stage (positions (116, 527 and 791). The problems are repeated in some of the reasons (positions 527 and 791).

	Selected SAR problems in stage 2. Part 2					
	Variables	**Options of variables**				
Stage 1	Space	Cont.	Disc.	Cont.	Disc.	Cont.
	Time	Disc.	Disc.	Disc.	Disc.	Disc.
	Shape	Rect.	Squa.	Rect.	Squa.	Rect.
	O.PTR	Indep.	Dep.	Dep.	Dep.	Dep.
	PTTDPR	Indep.	Dep.	Dep.	Dep.	Dep.
	W.Number	Mult.	Sing.	Sing.	Mult.	Mult.
	R.Accuracy	Homog.	Hetero.	Hetero.	Hetero.	Hetero.
Stage 2	S.Control	Cent.	Cent.	Cent.	Cent.	Cent.
	Objectives	Mult.	Sing.	Sing.	Mult.	Mult.
	R.Area	Equ.	Equ.	Equ.	Equ.	Equ.
	P.StartTimes	Sing.	Mult.	Mult.	Mult.	Mult.
	P.Deadlines	Sing.	Mult.	Mult.	Mult.	Mult.
	W.Content	Uni.	Mix.	Mix.	Mix.	Mix.
	Weight	2,125	2,250	2,500	2,525	2,775
	Position	116	239	527	559	791

Table A.5: Selected SAR problems in stage 3. The variables analysed in stage 3 are environment knowledge (Environment), number of process plans (Plans), parts loading and unloading time dependent on the type of part and robot (PL/UTDPR), type of robots wheels (R.Wheels), robots battery duration (R.Duration). The reasons for selecting the SAR problems their feasibility to be solved (positions 6, 15 and 47), their realism to industrial scenarios (positions 47, 266 and 288) and the interest to explore in combination with options of the next stage (positions (28, 47 and 85). The problem (position 47) is repeated in some of the reasons.

	SAR problems selected in stage 3							
	Variables	**Options of variables**						
Stage 1	Space	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.
	Time	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.	Disc.
	Shape	Squa.	Squa.	Squa.	Squa.	Squa.	Squa.	Squa.
	O.PTR	Indep.	Indep.	Indep.	Indep.	Indep.	Dep.	Dep.
	PTTDPR	Indep.	Indep.	Indep.	Indep.	Indep.	Dep.	Dep.
	W.Number	Sing.	Mult.	Sing.	Mult.	Mult.	Sing.	Mult.
	R.Accuracy	Homog.	Homog.	Homog.	Homog.	Homog.	Hetero.	Hetero.
Stage 2	S.Control	Cent.	Cent.	Decent.	Decent.	Decent.	Cent.	Cent.
	Objectives	Sing.	Mult.	Mult.	Sing.	Mult.	Sing.	Mult.
	R.Area	Equ.	Equ.	Equ.	Equ.	Equ.	Equ.	Equ.
	P.StartTimes	Sing.	Sing.	Sing.	Sing.	Sing.	Mult.	Mult.
	P.Deadlines	Sing.	Sing.	Sing.	Sing.	Sing.	Mult.	Mult.
	W.Content	Uni.	Uni.	Uni.	Uni.	Uni.	Mix.	Mix.
Stage 3	Environment	Struc.	Struc.	Struc.	Struc.	Struc.	Struc.	Struc.
	Plans	Mult.	Sing.	Sing.	Mult.	Sing.	Mult.	Sing.
	PL/UTDPR	Indep.	Indep.	Indep.	Indep.	Indep.	Dep.	Dep.
	R.Wheels	Uni.	Uni.	Uni.	Uni.	Mix.	Mix.	Mix.
	R.Battery	Homog.	Homog.	Homog.	Homog.	Homog.	Hetero.	Hetero.
	Weight	2,300	2,425	2,500	2,575	2,700	3,225	3,350
	Position	6	15	28	47	85	266	288

Table A.6: Selected SAR problems from stage 4. The variables analysed in stage 4 are system accuracy (S.Accuracy), locations of warehouses (W.Locations), tools setup and removal times (T.SRT), robots maximal payload (R.Payload), maximal number of carried tools by robot (R.CarriedTools). The reasons for selecting the SAR problems are feasibility to be solved the realism to industrial scenarios and interest of combining problems with variables of the next stage. This table shows two SAR problems selected for their feasibility to be solved (positions 1 and 8) and two for their realism to industrial scenarios (positions 194 and 216).

	SAR problems selected in stage 4				
	Variables	**Options of variables**			
Stage 1	Space	Disc.	Disc.	Disc.	Disc.
	Time	Disc.	Disc.	Disc.	Disc.
	Shape	Squa.	Squa.	Squa.	Squa.
	O.PTR	Indep.	Indep.	Dep.	Dep.
	PTTDPR	Indep.	Indep.	Dep.	Dep.
	W.Number	Sing.	Mult.	Sing.	Mult.
	R.Accuracy	Homog.	Homog.	Hetero.	Hetero.
Stage 2	S.Control	Cent.	Cent.	Cent.	Cent.
	Objectives	Sing.	Mult.	Sing.	Mult.
	R.Area	Equ.	Equ.	Equ.	Equ.
	P.StartTimes	Sing.	Sing.	Mult.	Mult.
	P.Deadlines	Sing.	Sing.	Mult.	Mult.
	W.Content	Uniq.	Uniq.	Mix.	Mix.
Stage 3	Environment	Struct.	Struct.	Struct.	Struct.
	Plans	Mult.	Sing.	Mult.	Sing.
	PL/UTDPR	Indep.	Indep.	Dep.	Dep.
	R.Wheels	Uniq.	Uniq.	Mix.	Mix.
	R.Battery	Homog.	Homog.	Hetero.	Hetero.
Stage 4	S.Accuracy	Cert.	Cert.	Cert.	Cert.
	W.Locations	Cons.	Cons.	Dyn.	Dyn.
	T.SRT	Equ.	Equ.	Uneq.	Uneq.
	R.Payload	Homog.	Homog.	Hetero.	Hetero.
	R.CarriedTools	Uniq.	Uniq.	Mix.	Mix.
	Weight	2,750	2,875	4,000	4,125
	Position	1	8	194	216

Appendix B

Representation of combinatorial process

This appendix presents a representation of the combinatorial process from the application of the proposed decision making methodology on a SAR problem case study in Chapter 7. Variables of the SAR problem case study were presented in Section 7.1. The combinatorial process refers to the combination and the aggregation process for the stage 3 and 4. The representation of stage 1 only refers to the combination process, and it was presented in Figure 7.1. The representation of the combination and aggregation processes for stage 2 can be observed in Figure 7.4, for stage 3 in Figure B.1, and for stage 4 in Figure B.2.

Combinations of variables' options from stage 2 are aggregated to the selected SAR problems from stage 1. A representation of the combinatorial and aggregation processes can be observed in Figure 7.4 from Appendix B. The combinatorial process is identical to the process for stage 1 (see Figure 7.1), but with the post combination of the selected SAR problems in stage 1 (i.e. aggregation process).

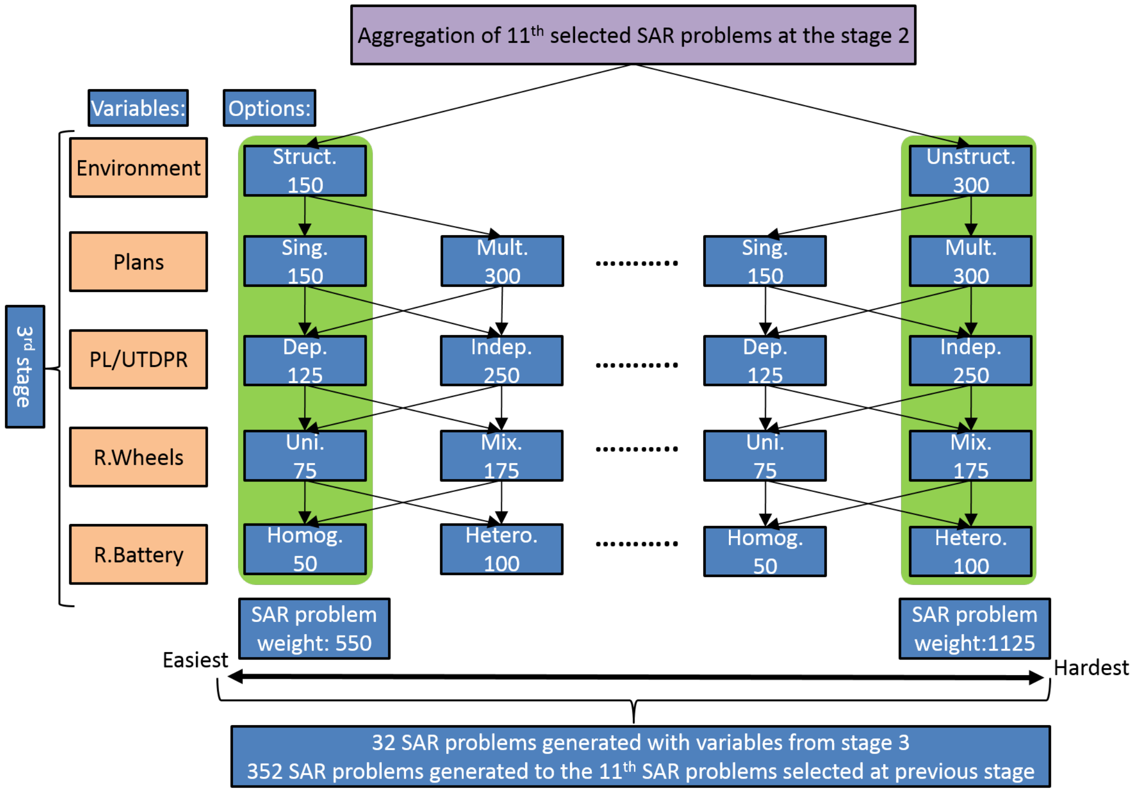


Figure B.1: Representation of the combinatorial and aggregation processes for stage 3. The variables combined in stage are shown in light orange rectangles, and their options in blue rectangles. The five binary variables result in 32 SAR problems. These 32 SAR problems are aggregated to the eleven SAR problems selected in stage 2 to generate 352 SAR problems. The combinatorial and aggregation processes are exhaustive and systematic, starting with the combination of the easiest options at the farthest left and the most difficult options at the farthest right.

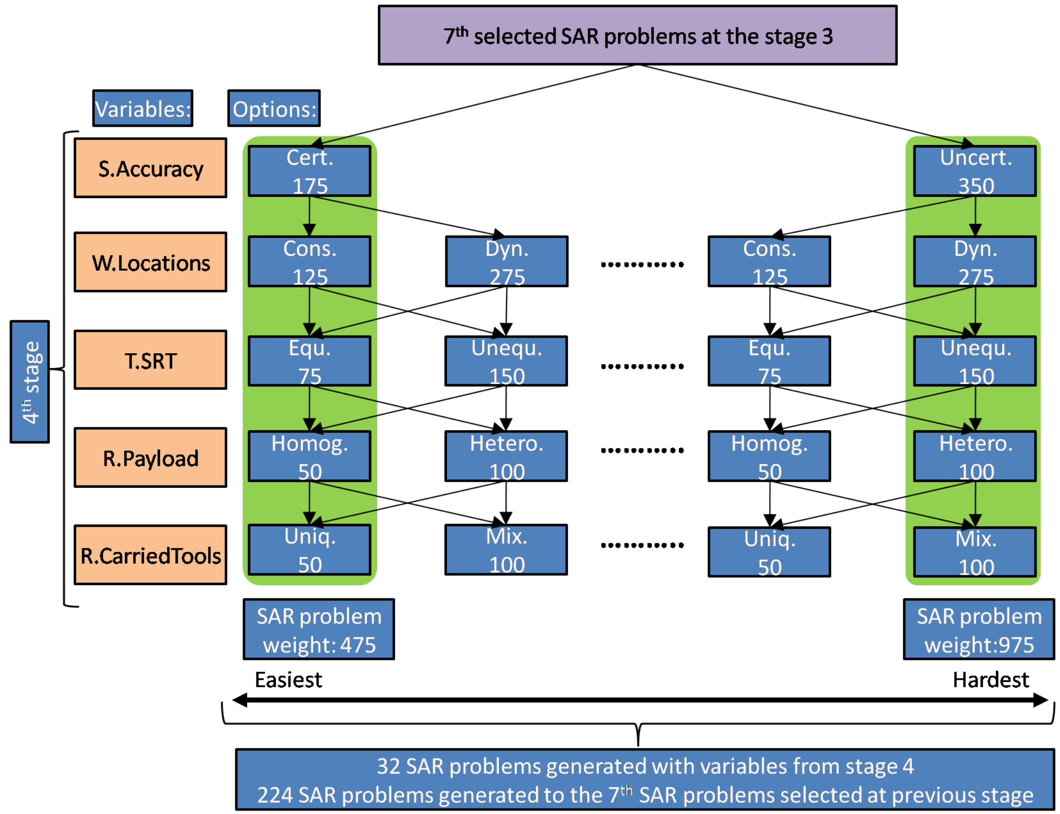


Figure B.2: Representation of the combinatorial and aggregation processes for stage 4. The variables combined in stage are shown in light orange rectangles, and their options in blue rectangles. The five binary variables result in 32 SAR problems. These 32 SAR problems are aggregated to the seven SAR problems selected in stage 3 to generate 224 SAR problems. The combinatorial and aggregation processes are exhaustive and systematic, starting with the combination of the easiest options at the farthest left and the most difficult options at the farthest right.

Appendix C

Representation of selected SAR problems

This appendix presents representations of the SAR problems selected with the proposed decision making methodology on a SAR problem case study in Chapter 7. Four complete SAR problems were selected at the final stage (i.e. stage 4), see Section 7.4.4. These complete SAR problems contain variables from all the stages (stage 1, 2, 3 and 4). The complete SAR problems are represented with the notation presented in Chapter 5.

Variables of the SAR problem case study were presented in Section 7.1. In specific, in Tables 7.1 and 7.2. Also, assumptions for this case study were presented in Section 7.1 The SAR problem case study simplified by these assumptions is represented with the following notation:

Factory vector

$$\mathbf{f} = (z^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{D}, \mathcal{S}^d)$$

Set of warehouses vector

$$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$$

Warehouse vector

$$\mathbf{w}_i = (\sigma^w, (x^{cent,w}(t), y^{cent,w}(t)), \mathcal{S}^p, \mathcal{S}^m)$$

Material shelf vector

$$\mathbf{s}_h^m = (m_h^{s\_m}, \xi_h^{s\_m}(r_k)(m_i), \mu_h^{s\_m}(r_k)(m_i), \eta_h^{s\_m}(r_k)(m_i), \cancel{q_h^{s\_m,max}}, \cancel{q_h^{s\_m}(t)}, (x_h^{cent,s\_m}(t), y_h^{cent,s\_m}(t)), \sigma_h^{s\_m}, \alpha_h^{s\_m}(t))$$

Product shelf vector

$$\mathbf{s}_h^p = (p_h^{s\_p}, \iota_h^{s\_p}(r_k)(p_i), \mu_h^{s\_p}(r_k)(p_i), \eta_h^{s\_p}(r_k)(p_i), \cancel{q_h^{s\_p,max}}, \cancel{q_i^{s\_p}(t)}, (x_h^{cent,s\_p}(t), y_h^{cent,s\_p}(t)), \sigma_h^{s\_p}, \alpha_h^{s\_p}(t))$$

Product requirements vector

$$\mathbf{p}_i = (\omega_i^p, \delta_i^p, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$$

Operation vector for product p_i

$$\mathbf{o}_j^{p_i} = (\cancel{C_j^{o,p_i}}, \cancel{g_j^{o,p_i}}, \cancel{\varpi_j^{o,p_i}}, (M_j^{o,p_i}, Q_j^{o,p_i}), \tau_j^{o,p_i}(r_k))$$

Robot vector

$$\mathbf{r}_k = (l_k^r, \kappa_k^r, \sigma_k^r, \cancel{a_k^r}, \upsilon^{r,max}, \psi^{r,max}, \cancel{a_k^{r,d}}, \gamma_k^{r,arm}, \gamma^{r,platform}, \beta_k^r, \varsigma_k^r(t), \cancel{\vartheta_k^r(t)}, \beta_k^r(t), D_k^r, (x_k^{cent,r}(t), y_k^{cent,r}(t)), \alpha_k^r(t), \upsilon_k^r(t), \psi_k^r(t))$$

Tool vector

$$\mathbf{d}_{k_{tools}} = (c^d_{k_{tools}}, \cancel{a^d_{k_{tools}}}, \chi^d_{k_{tools}}, \zeta^d_{k_{tools}}, M^d_{k_{tools}}, \cancel{\varpi^d_{k_{tools}}}, \cancel{\theta^d_{k_{tools}}(t)})$$

This simplified SAR problem is detailed with assumptions presented in the case study of Section 7.1. In this appendix, this case study is further defined to represent selected SAR problems from the application of the DMM (i.e. Chapter 7). There were four selected SAR problems, and these are represented and formulated in the following two sections.

C.1 SAR Problem types 1 and 2

The SAR problems types 1 and 2 are similar to each other. Hence, they are represented in the notation as a group. There are only three differences between the SAR problems types 1 and 2. These are the number of warehouses in the factory (W.Number), the number of objectives to optimise (Objectives) and the number of production plans per each product (P.Plans).

For the SAR problem type 1 there is a single warehouse, and single objective to optimise, and multiple production plans; whilst for the SAR problem type 2 there are multiple warehouses, multiple objectives to optimise, and single production plans per product. The SAR problem assumptions and notation are distinguished per each type of SAR problem by their type (i.e. **Type 1** and **Type 2**).

Notation

Assumptions for system variables:

- Space formulation as discrete results in shape of objects as squares. Thus, zones (z^{object}) and shapes (σ^{object}) are simplified to a single value measuring the side of square objects e^{object}. Orientation of objects are neglected (α^{object}).
- Time formulation as discrete. Thus: $t \in \mathbb{Z}_0^+$.
- Uncertainty of the system is neglected.
- System control is centralised.
- Environment is fully known (i.e. structured).
- **Type 1** There is a single objective to optimise
- **Type 2** There are multiple objectives to optimise

Assumptions for factory design and operation:

- **Type 1** There is a single warehouse in the factory
  ($\mathcal{W} = \{\mathbf{w}_1, \cancel{\mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}}\}$). Thus, the set of warehouses is simplified to a single warehouse $\mathbf{w}_1$.
- **Type 2** There are multiple warehouse in the factory
  ($\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$). Thus, the set of warehouses remains.
- The warehouse location $((x^{cent,w}\cancel{(t)}, y^{cent,w}\cancel{(t)}))$ is constant over time. Thus, the warehouse location is simplified to $(x^{cent,w}, y^{cent,w})$.
- There is a unique type of material ($m_i^{s\text{-}m}$) in the warehouse(s). Thus, the type of material variable is simplified to $m_i^{s\text{-}m}$ = material X for any shelf h.
- Materials loading ($\mu_h^{s\text{-}m}\cancel{(r_j)}\cancel{(m_i)}$) and unloading times ($\eta_h^{s\text{-}m}\cancel{(r_j)}\cancel{(m_i)}$) and products loading ($\mu_h^{s\text{-}p}\cancel{(r_j)}\cancel{(p_i)}$) and unloading times ($\eta_h^{s\text{-}p}\cancel{(r_j)}\cancel{(p_i)}$) are independent of the type of robot k and the type of product p_i and the type of material m_i related to any product p_i. Thus, the materials loading and unloading times are simplifed to $\mu_h^{s\text{-}m}$ and $\eta_h^{s\text{-}m}$ respectively, and the products loading and unloading times are simplified to $\mu_h^{s\text{-}p}$ and $\eta_h^{s\text{-}p}$ respectively.

- The materials supplying time ($\xi_h^{s\_m}\cancel{(r_j)}\cancel{(m_i)}$) and products picking up time ($\iota_h^{s\_p}\cancel{(r_j)}\cancel{(p_i)}$) are independent of the type of robot k and the type of product p_i and type of material m_i related to any product p_i. Thus, the materials supplying time and products picking up time are simplified to $\xi_h^{s\_m}$ and $\iota_h^{s\_p}$ respectively.

  Factory vector
  $\mathbf{f} = (e^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{D}, \mathcal{S}^d))$

Type 1 Set of warehouses vector
$\mathcal{W} = \{\mathbf{w}_1\}$

Type 2 Set of warehouses vector
$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$

Warehouse vector
$\mathbf{w}_1 = (e^w, (x^{cent,w}, y^{cent,w}), S^p, S^m)$

Material shelf vector
$\mathbf{s}_h^m = (m_h^{s\_m}, \xi_h^{s\_m}, \mu_h^{s\_m}, \eta_h^{s\_m}, (x_h^{cent,s\_m}, y_h^{cent,s\_m}), e_h^{s\_m})$

Product shelf vector
$\mathbf{s}_h^p = (p_h^{s\_p}, \iota_h^{s\_p}, \mu_h^{s\_p}, \eta_h^{s\_p}, (x_h^{cent,s\_p}, y_h^{cent,s\_p}), e_h^{s\_p})$

<u>Assumptions for production requirements:</u>

- Products have a single start time ($\delta_i^{start\_p}$). Thus, there is a start time δ^{start_p} for all products.
- Products have a single deadline ($\delta_i^{end\_p}$). Thus, there is a deadline time δ^{end_p} for all products.
- **Type 1** There are multiple process plans (i.e. precedences between operations, Π) for each product. Thus, the set of process plans is $\Pi^p = \Pi_1^{o,p_i}, ..., \Pi_{N_{operations\_product\_p_i}}^{o,p_i}$.
- **Type 2** There is a single process plan (Π) for each product. Thus, the set of process plans is $\Pi^p = \{\pi_1^{o,p_i}\}$.
- Operations processing times ($\tau_j^{o,p_i}\cancel{(r_k)}$) are independent of the robot k. Thus, the processing time variable is simplified to τ_j^{o,p_i}, where i is the type of product and j is the required operation.

Product requirements vector

$\mathbf{p}_i = (\omega_i^p, \delta^{start\_p}, \delta^{end\_p}, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$

Operation vector for product p_i

$\mathbf{o}_j^{p_i} = (c_j^{o,p_i}, (S_j^{o,p_i}, Q_j^{s_j}, \tau_j^{o,p_i})$

Type 1 Multiple sets of precedences between operations of each product p_i

$\Pi^p = \Pi_1^{o,p_i}, ..., \Pi_{N_{operations\_product\_p_i}}^{o,p_i}$

Type 2 Single set of precedences between operations of product p_i

$\Pi_1^p = \pi_1^{o,p_i}, ..., \pi_{N_{operations\_product\_p_i}}^{o,p_i}$

Assumptions for manufacturing resources:

- Maximal number of carried tools D_k^r is equal for all robots, where $D_k^r = \{d_1, d_2, ..., d_{j_{tools}}\}$. Thus, the maximal number of carried tools $d_{j_{tools}}$ is equal for all sets of tools D_k^r of all robots.
- Type of wheels configuration (l_k^r) is unique for all robots. Thus, the wheels configuration variable is simplified to l^r.
- Maximal battery duration ($\beta_k^r(t)$) is homogeneous for all robots. Thus, the maximal battery duration is simplified to $\beta^r(t)$.
- Maximal arm robot payload ($\gamma_k^{r,arm}$) is homogeneous for all robots. Thus, the maximal arm robot payload is simplified to $\gamma^{r,arm}$.
- Robots have equal area size (e_k^r). Thus, the side size is simplified to e^r.
- Tool setup times ($\chi_{k_{tools}}^d$) and removal times ($\zeta_{k_{tools}}^d$) are equal for all robots. These assumptions change the setup and removal times variables into the parameters χ^d and ζ^d.

Robot vector

$\mathbf{r}_k = (l^r, \kappa_k^r, e^r, v^{r,max}, \psi^{r,max}, \gamma^{r,arm}, \gamma^{r,platform}, \beta^r, \varsigma_k^r(t), \beta_k^r(t), D_k^r,$
$(x_k^{cent,r}(t), y_k^{cent,r}(t)), v_k^r(t), \psi_k^r(t))$

Tool vector

$\mathbf{d}_{k_{tools}} = (c_{k_{tools}}^d, \chi^d, \zeta^d, M_{k_{tools}}^d)$

C.2 SAR problem types 3 and 4

The SAR problems types 3 and 4 are similar to each other, but disimilar with the SAR problems types 1 and 2. Hence, they are represented in the notation as a group. There are only three differences between the SAR problems types 3 and 4. These are the number of warehouses in the factory (W.Number), the number of objectives to optimise (Objectives) and the number of production plans per each product (P.Plans).

For the SAR problem type 3 there is a single warehouse, and single objective to optimise, and multiple production plans; whilst for the SAR problem type 4 there are multiple warehouses, multiple objectives to optimise, and single production plans per product. The rest of the variables for the SAR problem 3 and 4 are the opposite of the SAR problems types 1 and 2. The SAR problem assumptions and notation are distinguished per each type of SAR problem by their type (i.e. **Type 1** and **Type 2**).

Notation

Assumptions for system variables:

- Space formulation as discrete, shape of objects as square. Thus, zones (z^{object}) and shapes (σ^{object}) are simplified to a single value measuring the side of square objects e^{object}. Orientation of objects are neglected (α^{object}).
- Time formulation as discrete. Thus: $t \in \mathbb{Z}_0^+$.
- Uncertainty of the system is neglected.
- System control is centralised.
- Environment is fully known (i.e. structured).
- **Type 3** There is a single objective to optimise.
- **Type 4** There are multiple objectives to optimise.

Assumptions for factory design and operation:

- **Type 3** There is a single warehouse in the factory ($\mathcal{W} = \{\mathbf{w}_1, \cancel{\mathbf{w}_2}, \ldots, \cancel{\mathbf{w}_{N_{warehouses}}}\}$). Thus, the set of warehouses is simplified to a single warehouse $\mathbf{w}_1$.
- **Type 4** There are multiple warehouse in the factory ($\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{N_{warehouses}}\}$). Thus, the set of warehouses remains.

- The warehouse location $((x^{cent,w}(t), y^{cent,w}(t)))$ changes over time. Thus, the warehouse location remains as $(x^{cent,w}(t), y^{cent,w}(t))$.

- There are multiple types of material $(m_i^{s\_m})$ for each shelf h in the warehouses. Thus, the type of material variable for each shelf remains as $m_i^{s\_m} = m_i$ for any shelf h.

- Materials loading $(\mu_h^{s\_m}(r_j)(m_i))$ and unloading times $(\eta_h^{s\_m}(r_j)(m_i))$ and products loading $(\mu_h^{s\_p}(r_k)(p_h))$ and unloading times $(\eta_h^{s\_p}(r_j)(p_i))$ are dependent on the type of robot k and the type of product p_i and any type of material m_i related to any product p_i. Thus, the materials loading and unloading times remain as $\mu_h^{s\_m}(r_k)(m_i)$ and $\eta_h^{s\_m}(r_k)(m_i)$ respectively, and the products loading and unloading times remain as $\mu_h^{s\_p}(r_k)(p_i)$ and $\eta_h^{s\_p}(r_k)(p_i)$ respectively.

- The materials supplying time $(\xi_h^{s\_m}(r_k)(m_i))$ and products picking up time $(\iota_h^{s\_p}(r_k)(p_i))$ depends on the type of robot k and the type of product p_i and type of material m_i related to any product p_i. Thus, the materials supplying time and products picking up time remain as $\xi_h^{s\_m}(r_k)(m_i)$ and $\iota_h^{s\_p}(r_k)(p_i)$ respectively.

Factory vector

$$\mathbf{f} = (e^f, (x^{cent,f}, y^{cent,f}), \mathcal{W}, \mathcal{P}, \mathcal{S}^p, \mathcal{M}, \mathcal{S}^m, \mathcal{R}, \mathcal{D}, \mathcal{S}^d))$$

Type 3 Set of warehouses vector

$$\mathcal{W} = \{\mathbf{w}_1\}$$

Type 4 Set of warehouses vector

$$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_{warehouses}}\}$$

Warehouse vector

$$\mathbf{w}_1 = (e^w, (x^{cent,w}(t), y^{cent,w}(t)), S^p, S^m)$$

Material shelf vector

$$\mathbf{s}_h^m = (m_h^{s\_m}, \xi_h^{s\_m}(r_j)(m_h), \mu_h^{s\_m}(r_j)(m_h), \eta_h^{s\_m}(r_j)(m_h), (x_h^{cent,s\_m}, y_h^{cent,s\_m}), e_h^{s\_m})$$

Product shelf vector

$$\mathbf{s}_h^p = (p_h^{s\_p}, \iota_h^{s\_p}(r_j)(p_h), \mu_h^{s\_p}(r_j)(p_h), \eta_h^{s\_p}(r_j)(p_h), (x_h^{cent,s\_p}, y_h^{cent,s\_p}), e_h^{s\_p})$$

Assumptions for production requirements:

- Products have multiple start times ($\delta_i^{start\_p}$). Thus, there is a start time $\delta_i^{start\_p}$ for each product i.
- Products have multiple deadlines times ($\delta_i^{end\_p}$). Thus, there is a deadline time $\delta_i^{end\_p}$ for each product i.
- **Type 3** There are multiple process plans (i.e. precedences between operations, Π) for each product. Thus, the set of process plans is $\Pi^p = \Pi_1^{o,p_i}, ..., \Pi_{N_{operations\_product\_p_i}}^{o,p_i}$.
- **Type 4** There is a single process plan (Π) for each product. Thus, the set of process plans is $\Pi^p = \{\pi_1^{o,p_i}\}$.
- Operations processing times ($\tau_j^{o,p_i}(r_k)$) depend on the robot k. Thus, the processing time variable remains $\tau_j^{o,p_i}(r_k)$ where i is the type of product and j is the required operation.

Product requirements vector

$$\mathbf{p}_i = (\omega_i^p, \delta_i^{start\_p}, \delta_i^{end\_p}, (M_i^p, Q_i^p), O_i^p, \Pi_i^p)$$

Operation vector for product p_i

$$\mathbf{o}_j^{p_i} = (c_j^{o,p_i}, (S_j^{o,p_i}, Q_j^{o,p_i}), \tau_j^{o,p_i}(r_k)$$

Type 3 Multiple sets of precedences between operations of each product p_i

$$\Pi^p = \Pi_1^{o,p_i}, ..., \Pi_{N_{operations\_product\_p_i}}^{o,p_i}$$

Type 4 Single set of precedences between operations of product p_i

$$\Pi_1^p = \pi_1^{o,p_i}, ..., \pi_{N_{operations\_product\_p_i}}^{o,p_i}$$

Assumptions for manufacturing resources:

- Maximal number of carried tools D_k^r is unequal for all robots, where $D_k^r = \{d_1, d_2, ..., d_{j_{tools}}\}$. Thus, the maximal number of carried tools $d_{j_{tools}}$ is unequal for all set of tools $D_k^r(t)$ of all robots.
- Type of wheels configuration (l_k^r) is mixed for all robots. Thus, the wheels configuration variable remains as l_k^r.

- Maximal battery duration ($\beta_k^r(t)$) is heterogeneous for all robots. Thus, the maximal battery duration remains as $\beta_k^r(t)$.
- Maximal arm robot payload ($\gamma_k^{r,arm}$) is heterogeneous for all robots. Thus, the maximal arm robot payload remains as $\gamma_k^{r,arm}$.
- Robots have equal area size (e_k^r). Thus, the side size is simplified to e^r
- Tool setup times ($\chi_{k_{tools}}^d$) and removal times ($\zeta_{k_{tools}}^d$) are unequal for all robots. Thus, the two variables remain as follows: $\chi_{k_{tools}}^d$ and $\zeta_{k_{tools}}^d$.

Robot vector

$\mathbf{r}_k = (l_k^r, \kappa_k^r, e^r, v^{r,max}, \psi^{r,max}, \gamma_k^{r,arm}, \gamma^{r,platform}, \beta_k^r, \varsigma_k^r(t), \beta_k^r(t), D_k^r,$
$(x_k^{cent,r}(t), y_k^{cent,r}(t)), v_k^r(t), \psi_k^r(t))$

Tool vector

$\mathbf{d}_{k_{tools}} = (c_{k_{tools}}^d, \chi_{k_{tools}}^d, \zeta_{k_{tools}}^d, M_{k_{tools}}^d)$

Bibliography

[1] F. Livesey, Defining High Value Manufacturing, Report for the Confederation of British Industry (CBI) and the Department of Trade and Industry (DTI), 2006.

[2] Technology Strategy Board. High Value Manufacturing Strategy 2012-2015, 2015. [Online]. Available: https://www.gov.uk/government/uploads/ system/uploads/attachment_ data/file/362294/High_ Value_ Manufacturing_ Strategy_ 2012-15.pdf [accessed March 29, 2015].

[3] K. G. Swift, J. D. Booker, and N. F. Edmondson, Strategies and case studies in assembly system selection, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, vol. 218, no. 7, pp. 675-688, 2004.

[4] Y. Koren, General RMS Characteristics. Comparison with Dedicated and Flexible Systems, in Reconfigurable Manufacturing Systems and Transformable Factories, no. 1, A. I. Dashchenko, Ed. Springer Berlin Heidelberg, 2006, pp. 27-46.

[5] G. Zhang, R. Liu, L. Gong, and Q. Huang, An Analytical Comparison on Cost and Performance among DMS, AMS, FMS and RMS, in Reconfigurable Manufacturing Systems and Transformable Factories, A. I. Dashchenko, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 659-673.

[6] X. Xu, From cloud computing to cloud manufacturing, Robotics and Computer-Integrated Manufacturing, vol. 28, no. 1, pp. 75-86, Feb. 2012.

[7] B. Montreuil, J. Frayret, and S. D. Amours, A strategic framework for networked manufacturing, Computers in Industry, vol. 42, pp. 299-317, 2000.

[8] H.-P. Wiendahl and R. Hernández, The Transformable Factory - Strategies, Methods and Examples, in Reconfigurable Manufacturing Systems

and Transformable Factories SE - 20, A. Dashchenko, Ed. Springer Berlin Heidelberg, 2006, pp. 383-393.

[9] M. F. Zaeh, M. Beetz, K. Shea, G. Reinhart, K. Bender, C. Lau, M. Ostgathe, W. Vogl, M. Wiesbeck, M. Engelhard, C. Ertelt, T. Rhr, M. Friedrich, and S. Herle, The Cognitive Factory, in Changeable and Reconfigurable Manufacturing Systems SE - 20, H. A. ElMaraghy, Ed. Springer London, 2009, pp. 355-371.

[10] M. Hvilshøj, S. Bøgh, O. S. Nielsen, and O. Madsen, Multiple part feeding - real-world application for mobile manipulators, Assembly Automation, vol. 32, no. 1, pp. 62-71, 2012.

[11] S. Giordani, M. Lujak, and F. Martinelli, A distributed multi-agent production planning and scheduling framework for mobile robots, Computers & Industrial Engineering, vol. 64, no. 1, pp. 19-30, 2013.

[12] K. Ueda, N. Fujii, I. Hatono, and M. Kobayashi, Facility Layout Planning Using Self-Organization Method, CIRP Annals - Manufacturing Technology, vol. 51, no. 1, pp. 300-402, 2002.

[13] Y. Yamada, Dynamic Reconfiguration of Reconfigurable Manufacturing Systems Using Particle Swarm Optimization, Proceedings of IEEE International Conference on Robotics and Automation, 2006.

[14] S. Makris, G. Michalos, A. Eytan, and G. Chryssolouris, Cooperating Robots for Reconfigurable Assembly Operations: Review and Challenges, in CIRP Conference on Manufacturing Systems 2012, 2012, vol. 3, pp. 346-351.

[15] D. Semere, M. Onori, A. Maffei, and R. Adamietz, Evolvable assembly systems: coping with variations through evolution, Assembly Automation, vol. 28, no. 2, pp. 126-133, Apr. 2008.

[16] W. F. Carriker, P. K. Khosla, and B. H. Krogh, Path planning for mobile manipulators for multiple task execution, IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 403-408, 1991.

[17] A. Mohri, S. Furuno, and M. Yamamoto, Trajectory planning of mobile manipulator with end-effector's specified path, International conference on intelligent robots and systems, 2001, pp. 2264-2269.

[18] M. H. Korayem, M. Nazemizadeh, and V. Azimirad, Optimal trajectory planning of wheeled mobile manipulators in cluttered environments using

potential functions, Scientia Iranica, vol. 18, no. 5, pp. 1138-1147, Oct. 2011.

[19] M. H. Korayem and H. Ghariblu, Analysis of wheeled mobile flexible manipulator dynamic motions with maximum load carrying capacities, Robotics and Autonomous Systems, vol. 48, no. 2-3, pp. 63-76, Sep. 2004.

[20] E. Papadopoulos, A framework for large-force task planning of mobile and redundant manipulators, Journal of Robotic Systems, vol. 16, no. 3, pp. 151-162, 1999.

[21] D. P. Garg and M. Kumar, Optimization techniques applied to multiple manipulators for path planning and torque minimization, Engineering Applications of Artificial Intelligence, vol. 15, pp. 241-252, 2002.

[22] K. T. Ulrich and S. D. Eppinger, Product design and development, 4th ed. Singapore: McGraw-Hill Higher Education, 2008, p. 368.

[23] M. Aruldoss, T. M. Lakshmi, and V. P. Venkatesan, A survey on multi criteria decision making methods and its applications, American Journal of Information Systems, vol. 1, no. 1, pp. 31-43, 2013.

[24] S. Honkala, M. Hämäläinen, and M. Salonen, Comparison of four existing concept selection methods, International conference on engineering design, ICED'07, 2007, no. August, pp. 1-11.

[25] G. E. Okudan and S. Tauhid, Concept selection methods - a literature review from 1980 to 2008, International Journal of Design Engineering, vol. 1, no. 3, p. 243, 2008.

[26] D. Bouyssou, Outranking methods, in Encyclopedia of Optimization, A. C. Floudas and M. P. Pardalos, Eds. Boston, MA: Springer US, 2009, pp. 2887-2893.

[27] S. Vinodh, G. Mulanjur, and A. Thiagarajan, Sustainable concept selection using modified fuzzy TOPSIS: a case study, International Journal of Sustainable Engineering, vol. 6, no. 2, pp. 109-116, Jun. 2013.

[28] S. Vinodh, A. R. Varadharajan, and A. Subramanian, Application of fuzzy VIKOR for concept selection in an agile environment, The International Journal of Advanced Manufacturing Technology, vol. 65, no. 5-8, pp. 825-832, May 2012.

[29] V. M. Cedeno-Campos, P. A. Trodden, T. J. Dodd, and J. Heley, Highly flexible self-reconfigurable systems for rapid layout formation to offer manufacturing services, IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 4819–4824.

[30] V. M. Cedeno-Campos, P. A. Trodden, and T. J. Dodd, An interactive methodology to explore optimization scenarios of a reconfigurable factory, IEEE International Conference on Emerging Technologies and Factory Automation, 2015.

[31] V. Martinez, A. Neely, G. Ren, and A. Smart, High Value Manufacturing: Delivering the promise, 2008.

[32] The manufacturer newsletter, UK Manufacturing Statistics, The Manufacturer Newsletter, 2016. [Online]. Available: http://www.themanufacturer.com/uk-manufacturing-statistics/. [Accessed: 02-May-2016].

[33] R. H. Hayes and S. C. Wheelwright, Restoring Our Competitive Edge: Competing Through Manufacturing, First. Wiley, 1984, p. 440.

[34] Wikipedia, Paradigm — Wikipedia, The Free Encyclopedia, 2016. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Paradigm&oldid=733597253. [Accessed: 01-Feb-2015].

[35] S. J. Hu, Evolving Paradigms of Manufacturing: From Mass Production to Mass Customization and Personalization, Procedia CIRP, vol. 7, pp. 3-8, Jan. 2013.

[36] D. Mourtzis and M. Doukas, The Evolution of Manufacturing Systems: From Craftsmanship to the Era of Customisation, in Handbook of Research on Design and Management of Lean Production Systems, IGI Global, 2014.

[37] G. Di Stefano, A. Gambardella, and G. Verona, Technology push and demand pull perspectives in innovation studies: Current findings and future research directions, Research Policy, vol. 41, no. 8, pp. 1283-1295, Oct. 2012.

[38] D. Mourtzis and M. Doukas, Design and Planning of Manufacturing Networks for Mass Customisation and Personalisation: Challenges and Outlook, Procedia CIRP, vol. 19, no. RoMaC, pp. 1-13, 2014.

[39] Y. Koren, The Global Manufacturing Revolution. Wiley, 2010, p. 424.

[40] K. Zhou, Industry 4.0: Towards Future Industrial Opportunities and Challenges, 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2015, pp. 2147-2152.

[41] Wikipedia, Industrial Revolution — Wikipedia, The Free Encyclopedia, 2016. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Industrial_Revolution&oldid=735411356. [Accessed: 01-Feb-2015].

[42] C. Roser, Industry 4.0, AllAboutLean.com, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Industry_4.0#/media/File:Industry_4.0.png. [Accessed: 15-Mar-2016].

[43] Deakin & Francis England, 230 years of creative craftsmanship, Deakin & Francis England, 2016. [Online]. Available: http://deakinandfrancis.co.uk/2016/01/230-years-of-creative-craftsmanship/. [Accessed: 02-Feb-2015]

[44] N. Perez, Watt steam engine, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Watt_steam_engine#/media/File:Maquina_vapor_Watt_ETSIIM.jpg. [Accessed: 02-Feb-2015].

[45] Wikipedia, Second Industrial Revolution — Wikipedia, The Free Encyclopedia, 2016. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Second_Industrial_Revolution&oldid=733551300. [Accessed: 03-Feb-2016].

[46] Anonymous, Chaine Citroen type A, Le catalogue Citroën 1918-1960, Fabien Sabatés, Editions Massin, 2016. [Online]. Available: https://commons.wikimedia.org/wiki/File:Chaine_Citroen_typea.jpg. [Accessed: 05-Feb-2016]

[47] G. Boothroyd, Fundamentals of machining and machine tools, 3rd ed. Boca Raton Taylor and Francis c2006, 1932, p. 573.

[48] T. Ono, Toyota production system: beyond large-scale production. Portland, OR: CRC Press, 1988, p. 176.

[49] M. Holweg, The genealogy of lean production, Journal of Operations Management, vol. 25, no. 2, pp. 420-437, Mar. 2007.

[50] S. Shingo and A. P. Dillon, A Study of the Toyota Production System: From an Industrial Engineering Viewpoint, Productivity press, 1989, p. 296.

[51] C. Mosier and L. Taube, The Facets of Group Technology and Their Impacts on Implementation A State-of-the-Art Survey, Omega, International Journal of Management Science, vol. 13, no. 5, pp. 381-391, 1985.

[52] H. M. Selim, R. G. Askin, and A. J. Vakharia, Cell formation in group technology: Review, evaluaiton and directions for future research, Computers Industrial Engineering, vol. 34, no. l, pp. 3-20, 1998.

[53] G. Da Silveira, D. Borenstein, and H. S. Fogliatto, Mass customization: Literature review and research directions, vol. 72, no. 49, 2001.

[54] T. W. Simpson, Product platform design and customization: Status and promise, AI EDAM, vol. 18, no. 01, pp. 3-20, 2004.

[55] C. S. T. Hau L. Lee, Modelling the Costs and Benefits of Delayed Product Differentiation, Management Science, vol. 43, no. 1, pp. 40-53, 1997.

[56] O. F. Operations, Assemble-to-Order Manufacturing: Implications For Materials Management, Journal of operations management, vol. 4, no. 4, pp. 347-368, 1984.

[57] N. O. Fernandes and S. do Carmo-Silva, Generic POLCAA production and materials flow control mechanism for quick response manufacturing, International Journal of Production Economics, vol. 104, no. 1, pp. 74-84, Nov. 2006.

[58] A. Gunasekaran, Agile manufacturing: A framework for research and development, International Journal of Production Economics, vol. 62, pp. 87105 1999.

[59] A. Gunasekaran and Y. Y. Yusuf, Agile manufacturing: A taxonomy of strategic and technological imperatives, International Journal of Production Research, vol. 40, no. 6, pp. 1357-1385, Jan. 2002.

[60] Y. Koren, Reconfigurable Manufacturing Systems, Annals of the CIRP, vol. 48, 1999.

[61] J. Gao, Y. Yao, V. C. Y. Zhu, L. Sun, and L. Lin, Service-oriented manufacturing: a new product pattern and manufacturing paradigm, Journal of Intelligent Manufacturing, vol. 22, no. 3, pp. 435-446, 2011.

[62] T. S. Baines, H. W. Lightfoot, S. Evans, a Neely, R. Greenough, J. Peppard, R. Roy, E. Shehab, a Braganza, a Tiwari, J. R. Alcock, J. P. Angus, M. Bastl, a Cousens, P. Irving, M. Johnson, J. Kingston, H. Lockett, V.

Martinez, P. Michele, D. Tranfield, I. M. Walton, and H. Wilson, State-of-the-art in product-service systems, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, vol. 221, no. 10, pp. 1543-1552, Jan. 2007.

[63] G. Q. Huang, T. Qu, R. Y. Zhong, Z. Li, H. D. Yang, Y. F. Zhang, Q. X. Chen, P. Y. Jiang, and X. Chen, Establishing production service system and information collaboration platform for mold and die products, The International Journal of Advanced Manufacturing Technology, vol. 52, no. 9-12, pp. 1149-1160, Jun. 2010.

[64] G. Chryssolouris, D. Mavrikios, N. Papakostas, D. Mourtzis, G. Michalos, and K. Georgoulias, Digital manufacturing: history, perspectives, and outlook, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, vol. 223, no. 5, pp. 451-462, May 2009.

[65] C. Werner Dankwort, R. Weidlich, B. Guenther, and J. E. Blaurock, Engineers' CAx education it's not only CAD, Computer-Aided Design, vol. 36, no. 14, pp. 1439-1450, Dec. 2004.

[66] O. F. Offodile and L. Abdel-Malek, The virtual manufacturing paradigm: The impact of IT / IS outsourcing on manufacturing strategy, International Journal of Production Economics, vol. 75, pp. 147-159, 2002.

[67] M. Iqbal and M. S. J. Hashmi, Design and analysis of a virtual factory layout, Journal of Materials Processing Technology, vol. 118, pp. 403-410, 2001.

[68] N. Guo and M. C. Leu, Additive manufacturing: technology, applications and research needs, Frontiers of Mechanical Engineering, vol. 8, no. 3, pp. 215-243, May 2013.

[69] G. N. Levy and R. Schindel, Rapid manufacturing and rapid tooling with layer manufacturing (LM) technologies, state of the art and future perspectives, CIRP Annals - Manufacturing Technology, vol. 52, no. 2, pp. 589-609, 2003.

[70] D. D. Gu, W. Meiners, K. Wissenbach, and R. Poprawe, Laser additive manufacturing of metallic components: materials, processes and mechanisms, International Materials Reviews, vol. 57, no. 3, pp. 133-164, Nov. 2013.

[71] M. Groover, Automation, Production Systems, and Computer-Integrated Manufacturing, MyiLibrary, 2005. [Online]. Available: http://www.myilibrary.com?ID=734317. [Accessed: 01-Aug-2016].

[72] A.-W. Scheer, Computer Integrated Manufacturing. Towards the factory of the future, 3rd ed. Springer-Verlag Berlin, 1994.

[73] P. Ferrari, A. Flammini, and S. Vitturi, Performance analysis of PROFINET networks, Computer Standards & Interfaces, vol. 28, no. 4, pp. 369-385, Apr. 2006.

[74] A. Gunasekaran, Implementation of computer-integrated manufacturing: A survey of integration and adaptability issues, International Journal of Computer Integrated Manufacturing, vol. 10, no. 1-4, pp. 266-280, Jan. 1997.

[75] S. D. Amours, B. Montreuil, and P. Lefranc, Networked manufacturing: The impact of information sharing, International Journal of Production Economics, vol. 58, pp. 63- 79, 1999.

[76] P. Taylor, H. Yang, and D. Xue, Recent research on developing Web-based manufacturing systems: a review, International Journal of Production, vol. 41, no. 15, pp. 37-41, 2003.

[77] M. Hermann, T. Pentek, and B. Otto, Design Principles for Industrie 4.0 Scenarios, International Conference on System Sciences (HICSS), 2016, pp. 3928-3937.

[78] H. S. Kang, J. Y. Lee, S. Choi, H. Kim, J. H. Park, J. Y. Son, B. H. Kim, and S. Do Noh, Smart manufacturing: Past research, present findings, and future directions, International Journal of Precision Engineering and Manufacturing-Green Technology, vol. 3, no. 1, pp. 111-128, Jan. 2016.

[79] D. Lucke, C. Constantinescu, and E. Westkämper, Smart Factory - A Step towards the Next Generation of Manufacturing, Manufacturing Systems and Technologies for the New Frontier: The 41st CIRP Conference on Manufacturing Systems, 2008, pp. 115-118.

[80] L. Abdel-Malek, C. Wolf, and D. Guyot, Telemanufacturing: A flexible manufacturing solution, International Journal of Production Economics, vol. 5273, no. 96, 1998.

[81] D. Wu, J. L. Thames, D. W. Rosen, and D. Schaefer, Towards a cloud-based design and manufacturing paradigm: Looking backward, looking forward,

Proceedings of the ASME 2012 International Design Engineering Technical Conference & Computers and Information in Engineering Conference IDETC/CIE 2012, no. 1, pp. 1-14, 2012.

[82] D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer, Cloud manufacturing: Strategic vision and state-of-the-art, Journal of Manufacturing Systems, pp. 1-16, May 2013.

[83] P. Chandra, Model for the evaluation of routing and machine flexibility, European Journal of Operational Research, vol. 60, pp. 156-165, 1992.

[84] M. Pinedo, Scheduling: Theory, Algorithms and Systems. Prentice-Hall Englewood Cliffs, NJ, 1995.

[85] R. Graham, E. Lawler, J. Lenstra, and A. Rinnooy Kan, Optimization and approximation in deterministic sequencing and scheduling: A survey, Annals of Discrete Mathematics, vol. 5, pp. 287-326, 1979.

[86] ORC Resources, Methods of Production, Commercial production methods: One-off, batch or mass production?, 2014. [Online]. Available: http://www.ocr.org.uk/Images/174407-unit-r110-commercial-production-methods-one-off-batch-or-mass-production-lesson-element-teacher-instructions.pdf. [Accessed: 06-Nov-2016].

[87] A. Drexl and A. Kimms, Lot sizing and scheduling - Survey and extensions, European Journal of Operational Research, vol. 99, pp. 221-235, 1997.

[88] B. Karimi, S. M. T. Fatemi Ghomi, and J. M. Wilson, The capacitated lot sizing problem: a review of models and algorithms, International Journal of Management Science, vol. 31, no. 5, pp. 365-378, Oct. 2003.

[89] N. Boysen, M. Fliedner, and A. Scholl, A classification of assembly line balancing problems, European Journal of Operational Research, vol. 183, no. 2, pp. 674-693, Dec. 2007.

[90] J. Collins, Integrated manufacturing - the stage of the art, The Production Engineer, no. June, pp. 41-44, 1980.

[91] W. Michaeli, U. Rosenbaum, and M. Jehrke, Processing strategy for braiding of complex-shaped parts based on a mathematical process description, Composites Manufacturing, vol. 1, no. 4, pp. 243-251, 1990.

[92] R. G. Landers, B. K. Min, and Y. Koren, Reconfigurable Machine Tools, CIRP Annals-Manufacturing Technology, vol. 50, no. 1, pp. 269-274, 2001.

[93] H. A. ElMaraghy, Flexible and reconfigurable manufacturing systems paradigms, International Journal of Flexible Manufacturing Systems, vol. 17, no. 4, pp. 261-276, Oct. 2006.

[94] H.-P. Wiendahl, H. A. ElMaraghy, P. Nyhuis, M. F. Zäh, H.-H. Wiendahl, N. Duffie, and M. Brieke, Changeable Manufacturing - Classification, Design and Operation, CIRP Annals - Manufacturing Technology, vol. 56, no. 2, pp. 783-809, Jan. 2007.

[95] A. Sethi and S. Sethi, Flexibility in manufacturing: A survey, International Journal of Flexible Manufacturing Systems, vol. 2, no. 4, pp. 289-328, Jul. 1990.

[96] V. M. Cedeno-Campos, A novel planning methodology for highly flexible self-reconfigurable systems to achieve rapid layout formation, Confirmation review report, Internal report, The University of Sheffield, 2013.

[97] W. Terkaj, T. Tolio, and A. Valente, A Review on Manufacturing Flexibility, Design of Flexible Production Systems, Springer, Tolio T. (Ed.), pp. 41-61, 2009b.

[98] S. H. Minhas, C. Lehmann, J. P. Städter, and U. Berger Reconfigurable strategies for manufacturing setups to confront mass customization challenges, Proceedings from 21st International Conference on Production Research, 2011.

[99] B. H. B. Yeung and J. K. Mills, Design of a Six DOF Reconfigurable Gripper for Flexible Fixtureless Assembly, IEEE Transactions on Systems, Man, and Cybernetics Part C: applications and reviews, vol. 34, no. 2, pp. 226-235, 2004.

[100] S. Benjafaar, S. S. Heragu, and S. A. Irani, Next generation factory layouts: research challenges and recent progress, Interfaces, vol. 32, no. 6, pp. 58-76, 2002.

[101] Management & Development Center, Facility Layout. [Online]. Available: http://www.mdcegypt.com/pages/Operation%20Management/Production%20&%20Operation%20Management/Facility%20Layout/Facility%20Layout%20Introduction/Facility%20Layout%20Introduction.asp [Accessed: 20-Mar-2016].

[102] M. M. D. Hassan, Machine layout problem in modern manufacturing facilities, International Journal of Production Research, vol. 32, no. 11, pp. 2559-2584, Nov. 1994.

[103] A. Drira, H. Pierreval, and S. Hajri-gabouj, Facility layout problems: A survey, Annual Reviews in Control, vol. 31, pp. 255-267, 2007.

[104] S. Subashini and V. Kavitha, A survey on security issues in service delivery models of cloud computing, Journal of Network and Computer Applications, vol. 34, no. 1, pp. 1-11, Jan. 2011.

[105] J. Byrne, C. Heavey, and P. J. Byrne, A review of Web-based simulation and supporting tools, Simulation Modelling Practice and Theory, vol. 18, no. 3, pp. 253-276, Mar. 2010.

[106] R. C. Luo and J. H. Tzou, The development of distributed web-based rapid prototyping manufacturing system, International Conference on Robotics and Automation, 2003, pp. 1717-1722.

[107] G. D. Putnik and A. Sluga, Reconfigurability of manufacturing systems for agility implementation - part I: Requirements and principles, Digital Enterprise Technology, Springer US, 2007, pp. 91-98.

[108] B. Kehoe, S. Member, S. Patil, P. Abbeel, S. Member, and K. Goldberg, A Survey of Research on Cloud Robotics and Automation, IEEE Transactions on automation science and engineering, vol. 12, no. 2, pp. 398-409, 2015.

[109] G. Hu, W. P. Tay, and Y. Wen, Cloud Robotics: Architecture, Challenges and Applications, IEEE Network, no. June, pp. 21-28, 2012.

[110] W. Heinrichs, Design management - Do it anywhere, Electronics Systems and Software, vol. 3, no. September, pp. 30-33, 2005.

[111] D. B. Siassipour, Study of Auction-Based Methods for Multi-Robot Coordination, Internal Report from University of Coimbra, 2010.

[112] J. Lee, M. Ghaffari, and S. Elmeligy, Self-maintenance and engineering immune systems: Towards smarter machines and manufacturing systems, Annual Reviews in Control, vol. 35, no. 1, pp. 111-122, Apr. 2011.

[113] L. Wang, P. Orban, A. Cunningham, and S. Lang, Remote real-time CNC machining for web-based manufacturing, Robotics and Computer-Integrated Manufacturing, vol. 20, pp. 563-571, 2004.

[114] A. J. Alvares and J. C. Espindola-Ferreira, WebTurning: Teleoperation of a CNC turning center through the Internet, Journal of Materials Processing Technology, vol. 179, pp. 251-259, 2006.

[115] H. Hanwu and W. Yueming, Web-based virtual operating of CNC milling machine tools, Computers in Industry, vol. 60, pp. 686-697, 2009.

[116] S. P. Lal and G. C. Onwubolu, Three tiered web-based manufacturing systemPart 1: System development, Journal of Robotics and Computer-Integrated Manufacturing, vol. 23, no. 1, pp. 138-151, Feb. 2007.

[117] H. Lan, Web-based rapid prototyping and manufacturing systems: A review, Computers in Industry, vol. 60, no. 9, pp. 643-656, Dec. 2009.

[118] V. B. Sunil and S. S. Pande, WebROBOT: Internet based robotic assembly planning system, Computers in Industry, vol. 54, no. 2, pp. 191-207, Jun. 2004.

[119] Y. M. Wang, H. L. Yin, N. F. Xiao, and Y. R. Jiang, Internet-based remote manipulation and monitoring of an industry robot in advanced manufacturing systems, International Journal of Advanced Manufacturing Technology, vol. 43, no. 9-10, pp. 907-913, Oct. 2008.

[120] D. Zissis and D. Lekkas, Addressing cloud computing security issues, Future Generation Computer Systems, vol. 28, no. 3, pp. 583-592, Mar. 2012.

[121] S. Subashini and V. Kavitha, A survey on security issues in service delivery models of cloud computing, Journal of Network and Computer Applications, vol. 34, no. 1, pp. 1-11, Jan. 2011.

[122] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, Cloud-based manufacturing: Old wine in new bottles?, Procedia CIRP, vol. 17, pp. 94-99, 2014.

[123] S. P. Singh and R. R. K. Sharma, A review of different approaches to the facility layout problems, The International Journal of Advanced Manufacturing Technology, vol. 30, no. 5-6, pp. 425-433, 2006.

[124] B. Montreuil, A modelling framework for integrating layout design and flow network design, Material Handling '90 SE - 8, vol. 2, Springer Berlin Heidelberg, 1991, pp. 95-115.

[125] T. Koopmans and M. Beckmann, Assignment problems and the location of economic activities, Econometrica, vol. 25, no. 1, pp. 53-76, 1957.

[126] M. M. Hassan and G. L. Hogg, A review of graph theory application to the facilities layout problem, Omega, The International Journal of Management Sciences vol. 15, no. 4, pp. 291-300, Jan. 1987.

[127] G. C. Armour and E. S. Buffa, A heuristic algorithm and simulation approach to relative location of facilities, Journal of Management Sciences, vol. 9, no. 2, pp. 294-309, Jan. 1963.

[128] J. D. Knowles and D. W. Corne, Towards landscape analyses to inform the design of a hybrid local search for the multiobjective quadratic assignment problem, Hybrid Intell Syst, pp. 271-279, 2002.

[129] E.-G. Talbi, O. Rouxb, C. Fonlupt, and D. Robillard, Parallel ant colonies for quadratic assignment problem, Future Generation Computer Systems, vol. 17, pp. 441-449, 2001.

[130] R. Tavakkoli-Moghaddain and E. Shanyan, Facilities layout design by genetic algorithms, Computers and Industrial Engineering, vol. 35, no. 3/4, pp. 527-530, 1998.

[131] S. Abdinnour-Helm and S. W. Hadley, Tabu search based heuristics for multi-floor facility layout, International Journal of Production Research, vol. 38, no. 2, pp. 365-383, 2000.

[132] R. E. Burkard and F. Rendl, A thermodynamically motivated simulation procedure for combinatorial optimization problems, European Journal of Operations Research, vol. 17, pp. 169-174, 1984.

[133] A. Dashchenko, Reconfigurable Manufacturing Systems and Transformable Factories, Springer, 2006.

[134] E. Westkämper, Factory Transformability: Adapting the Structures of Manufacturing, in Reconfigurable Manufacturing Systems and Transformable Factories SE - 19, A. Dashchenko, Ed. Springer Berlin Heidelberg, 2006, pp. 371-381.

[135] J. Reichardt and H.-P. Wiendahl, Changeable Factory Buildings - An Architectural View, in Changeable and Reconfigurable Manufacturing Systems SE - 22, H. A. ElMaraghy, Ed. Springer London, 2009, pp. 389-401.

[136] J. M. A. T. James A. Tompkins, John A. White, Yavuz A. Bozer, Facilities planning, 4th ed. Wiley, 2010, p. 864.

[137] M. Rudberg and B. Martin West, Global operations strategy: Coordinating manufacturing networks, Omega, The International Journal of Management Sciences vol. 36, no. 1, pp. 91-106, Feb. 2008.

[138] F. Sahin, Manufacturing competitiveness: Different systems to achieve the same results, Production and Inventory Management Journal, Vol.41(1), p.56-61, 2000.

[139] H. W. Kuhn, The Hungarian method for the assignment problem, Naval Research Logistics Quarterly, vol. 2, no. 12, pp. 8397, 1955.

[140] B. P. Gerkey, A formal analysis and taxonomy of task allocation in multi-robot systems, International Journal of Robotics Research, vol. 23, no. 9, pp. 939-954, 1998.

[141] A. Gage, Multi-robot task allocation using affect, Graduate Theses and Dissertations, University of South Florida, 2004. [Online]. Available: http://scholarcommons.usf.edu/etd/1042. [Accessed: 17-Apr-2016].

[142] B. P. Gerkey and M. J. Mataric, Sold!: auction methods for multirobot coordination, IEEE Transactions on Robotics and Automation, vol. 18, no. 5, pp. 758-768, Oct. 2002.

[143] M. B. Dias, R. Zlot, M. Zinck, J. P. Gonzalez, and A. Stentz, A versatile implementation of the TraderBots approach to multirobot coordination, Proceedings of the International Conference on Intelligent Autonomous Systems (IAS), 2004

[144] L. E. Parker, ALLIANCE: an architecture for fault tolerant multirobot cooperation, IEEE Transactions on Robotics and Automation, vol. 14, no. 2, pp. 220-240, Apr. 1998.

[145] L. Chaimowicz, M. Campos, and K. Vijay, Dynamic role assignment for cooperative robots, Proceedings of 2002 IEEE International Conference of Robotics and Automation, no. May, pp. 293-298, 2002.

[146] R. Zlot and A. Stentz, Complex task allocation for multiple robots, Proceedings of 2005 IEEE International Conference of Robotics and Automation, pp. 1515-1522.

[147] R. G. Smith, The Contract Net Protocol: High-level communication and control in a distributed problem solver, IEEE Transactions on Computers, vol. C-29, no. 12, pp. 1104-1113, 1980.

[148] H. Penny Nii, The Blackboard model of problem solving and the evolution of Blackboard architectures, AI Magazine, 1986.

[149] B. B. Biswal and B. B. Choudhury, Cooperative task planning of multi-robot systems, in International Symposium on Automation and Robotics in Construction (ISARC 2007), 2007.

[150] Intelligent Manufacturing Systems (IMS), Roadmap on sustainable manufacturing, Energy Efficient Manufacturing and Key Technologies, 2010, [Online]. Available: http://www.ims.org/wp-content/uploads/2011/10/IMS2020_Action-Roadmap_KAT1-2-3.pdf, [Accessed: 10-May-2016].

[151] European Factories of the Future Research Association (EFFRA), Strategic multi-annual roadmap, 2012. [Online]. Available: https://ec.europa.eu/research/industrial_technologies/pdf/ppp-factories-of-the-future-strategic-multiannual-roadmap-info-day_en.pdf, [Accessed: 10-May-2016].

[152] M. Onori, D. Semere, and B. Lindberg, Evolvable systems: an approach to self-X production, International Journal of Computer Integrated Manufacturing., vol. 24:5, pp. 506-516, 2011.

[153] K. Ueda, I. Hatono, N. Fujii, and J. Vaario, Line-less production system using self-organization: A case study for BMS, CIRP Annals - Manufacturing Technology, vol. 50, no. 1, pp. 319-322, 2001.

[154] C. Christo and C. Cardeira, Plug-and-Play Autonomous Mobile Robot (P &PAMR), 2007 IEEE Intenational Symposium of Industrial Electronics, pp. 3215-3220, Jun. 2007.

[155] P. R. Wurman and R. D. Andrea, Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses, AI magazine, vol. 29, no. 1, pp. 9-20, 2008.

[156] S. Bøgh, M. Hvilshøj, and M. Kristiansen, Identifying and evaluating suitable tasks for autonomous industrial mobile manipulators (AIMM), International Journal of Advanced Manufacturing Technology, vol. 61, pp. 713-726, 2012.

[157] M. Hvilshøj, S. Bøgh, O. S. Nielsen, and O. Madsen, Multiple part feeding - real-world application for mobile manipulators, Assembly Automation, vol. 32, no. 1, pp. 62-71, 2012.

[158] F. Dietrich, C. Löchte, S. Jeschke, and A. Raatz, An agent-based concept for planning and control of the production of carbon fibre reinforced plastics

aircraft structures with mobile production units, Production Engineering, vol. 6, pp. 531-539, 2012.

[159] J. Pandremenos and C. Doukas, Machining with robots: a critical review, International Conference on Digital Enterprise Technology (DET 2011), September, 2011.

[160] Kuka, Kuka YouBot, Kuka Youbot Store [Online]. Available: http://www.youbot-store.com/kuka-youbot-omni-directional-mobile-platform-with-arm. [Accessed: 19-Sep-2013].

[161] M. Wojtczyk, A new model to design software architectures for mobile service robots, Technische Universitat Munchen, Munchen, Germany, 2010.

[162] P. Fattahi, M. Saidi Mehrabad, and F. Jolai, Mathematical modeling and heuristic approaches to flexible job shop scheduling problems, Journal of Intelligent Manufacturing, vol. 18, no. 3, pp. 331-342, Jul. 2007.

[163] R. Ruiz and J. A. Vázquez-Rodríguez, The hybrid flow shop scheduling problem, European Journal of Operational Research, vol. 205, no. 1, pp. 1-18, 2010.

[164] X. Wang and Q. Dai, Scheduling for flexible flow-shop problem based on an improved genetic algorithm, in International Conference on Consumer Electronics, 2014, pp. 1-3.

[165] R. Ramasesh, Dynamic job shop scheduling: A survey of simulation research, International Journal of Management Science, vol. 18, no. 1, pp. 43-57, 1990.

[166] E. Levner, V. Kats, D. Alcaide, L. De Pablo, and T. C. E. Cheng, Complexity of cyclic scheduling problems: A state-of-the-art survey, Computers & Industrial Engineering, vol. 59, no. 2, pp. 352-361, 2010.

[167] J. Blazewicz, Scheduling subject to resource constraints: classification and complexity, Discrete Applied Mathematics, vol. 5, pp. 11-24, 1983.

[168] C. Lee, L. Lei, and M. Pinedo, Current trends in deterministic scheduling, Annals of Operations Research, vol. 70, pp. 1-41, 1997.

[169] S. Gawiejnowicz, Time-dependent scheduling, First. Springer Berlin Heidelberg, 2009.

[170] T. C. E. Cheng, Q. Ding, and B. M. T. Lin, A concise survey of scheduling with time-dependent processing times, European Journal of Operational Research, vol. 152, pp. 1-13, 2004.

[171] G. Mosheiov and J. B. Sidney, Scheduling a deteriorating maintenance activity on a single machine, Journal of the Operational Research Society, vol. 61, pp. 882-887, 2010.

[172] J. Blazewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Weglarz, Scheduling under resource constraints, in Handbook on Scheduling: From Theory to Applications, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 425-475.

[173] Q. Zhang, H. Manier, and M. Manier, A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times, Computers and Operation Research, vol. 39, no. 7, pp. 1713-1723, 2012.

[174] J. Liu and B. L. Maccarthy, A global MILP model for FMS scheduling, European Journal of Operational Research, vol. 100, pp. 441-453, 1997.

[175] D. Ouelhadj and S. Petrovic, A survey of dynamic scheduling in manufacturing systems, Journal of Scheduling, vol. 12, no. October 2008, pp. 417-431, 2009.

[176] T. Chaari, S. Chaabane, N. Aissani, and D. Trentesaux, Scheduling under uncertainty: survey and research directions, International conference on advanced logistics and transport, 2014, pp. 229-234.

[177] A. Rajabinasab and S. Mansour, Dynamic flexible job shop scheduling with alternative process plans: an agent-based approach, International journal of advanced manufacturing technology, vol. 54, pp. 1091-1107, 2011.

[178] W. He and D. Sun, Scheduling flexible job shop problem subject to machine breakdown with route changing and right-shift strategies, international journal of advanced manufacturing technology, vol. 66, pp. 501-514, 2013.

[179] V. Suresh, Dynamic scheduling-A survey of research, International journal of production economics, vol. 32, pp. 53-63, 1993.

[180] W. Herroelen, B. De Reyck, and E. Demeulemeester, Resource-constrained project scheduling: a survey of recent developments, Computers operations research, vol. 25, no. 4, pp. 279-302, 1998.

[181] M. Drozdowski, Scheduling multiprocessor tasks - An overview, European Journal of Operational Research, vol. 94, pp. 215-230, 1996.

[182] S. P. Siferd and W. C. Benton, Workforce staffing and scheduling: Hospital nursing specific models, European Journal of Operational Research, vol. 60, pp. 233-246, 1992.

[183] B. Gerkey, Task allocation for heterogeneous robots, Tulane University, 1998.

[184] E. G. Jones, M. B. Dias, and A. Stentz, Time-extended multi-robot coordination for domains with intra-path constraints, Autonomous Robots, vol. 30, pp. 41-56, 2011.

[185] E. G. Jones, B. Browning, M. B. Dias, B. Argall, M. Veloso, and A. Stentz, Dynamically Formed Heterogeneous Robot Teams Performing Tightly-Coordinated Tasks, Robotics and Automation, ICRA 2006, pp. 570-575.

[186] B. P. Gerkey, On multi-robot task allocation, PhD thesis, University of Southern California, Los Angeles, 2003.

[187] G. A. Korsah, A. Stentz, and M. B. Dias, A comprehensive taxonomy for multi-robot task allocation, The International Journal of Robotics Research, vol. 32, no. 12, pp. 1495-1512, Oct. 2013.

[188] B. M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, Market-Based Multirobot Coordination: A Survey and Analysis, Proceedings of the IEEE, vol. 94, no. 7, pp. 1257-1270, 2006.

[189] A. Khamis, A. Hussein, and A. Elmogy, Multi-robot Task Allocation: A Review of the State-of-the-Art, Studies in Computational Intelligence, vol. 604, pp. 31-52, 2015.

[190] J. M. Wilson, Gantt charts: A centenary appreciation, European Journal of Operational Research, vol. 149, pp. 430-437, 2003.

[191] A. Kusiak and S. S. Heragu, The facility layout problem, European Journal of Operational Research, vol. 29, pp. 229-251, 1987.

[192] R. D. Meller and K. Gau, The facility layout problem: Recent and emerging trends and perspectives, Journal of Manufacturing Systems, vol. 15, no. 5, pp. 351-366, 1996.

[193] G. Meng, S. S. Heragu, and H. Zijm, Reconfigurable layout problem, International Journal of Production Research, vol. 42, no. 22, pp. 4709-4729, Nov. 2004.

[194] M. J. Rosenblatt, The dynamics of plant layout, Management Science, vol. 32, no. January, pp. 76-87, 1986.

[195] J. S. Kochhar and S. S. Heragu, Facility layout design in a changing environment, International Journal of Production Research, vol. 37, no. 11, pp. 2429-2446, Jul. 1999.

[196] M. Abedzadeh, M. Mazinani, N. Moradinasab, and E. Roghanian, Parallel variable neighborhood search for solving fuzzy multi-objective dynamic facility layout problem, The International Journal of Advanced Manufacturing Technology, vol. 65, no. 1-4, pp. 197-211, May 2012.

[197] T. A. Lacksonen, Static and Dynamic Layout Problems with Varying Areas, The Journal of the Operational Research Society, vol. 45, no. 1, pp. 59-69, 2013.

[198] G. Y. Chen and K. J. Rogers, Proposition of two multiple criteria models applied to dynamic multi-objective facility layout problem based on ant colony optimization, 2009 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1553-1557, Dec. 2009.

[199] I. Castillo and B. A. Peters, An extended distance-based facility layout problem, International Journal of Production Research, vol. 41, no. 11, pp. 2451-2479, Jan. 2003.

[200] J. Balakrishnan and C. Hung Chen, Dynamic layout algorithms: a state-of-the-art survey, Omega, International Journal of Management Science, vol. 26, no. 4, pp. 507-521, 1998.

[201] T. L. Urban, Solution procedures for the dynamic facility layout problem, Annals of Operations Research, vol. 76, pp. 323-342, 1998.

[202] R. S. Liggett, Automated facilities layout: past, present and future, Automation in Construction, vol. 9, no. 2, pp. 197-215, Mar. 2000.

[203] G. Moslemipour, T. S. Lee, and D. Rilling, A review of intelligent approaches for designing dynamic and robust layouts in flexible manufacturing systems, The International Journal of Advanced Manufacturing Technology, vol. 60, no. 1-4, pp. 11-27, Sep. 2011.

[204] S. Kulturel-Konak, Approaches to uncertainties in facility layout problems: Perspectives at the beginning of the 21st Century, Journal of Intelligent Manufacturing, vol. 18, no. 2, pp. 273-284, Jul. 2007.

[205] S. Liu, D. Sun, C. Zhu, and W. Shang, A dynamic priority strategy in decentralized motion planning for formation forming of multiple mobile robots, IEEE/RSJ International Conference on Intelligent Robots and Systems, no. 1, pp. 37743779, Oct. 2009.

[206] J.C. Latombe, Robot motion planning, Kluwer academic publishers, 1991.

[207] D. Filliat and J. Meyer, Map-based navigation in mobile robots: I . A review of localization strategies, Cognitive Systems Research, vol. 4, pp. 243-282, 2003.

[208] Y. K. Hwang and N. Ahuja, Gross Motion PlanningA Survey, ACM Computing Surveys, vol. 24, no. 3, 1992.

[209] C. Prins, Efficient heuristics for the heterogeneous fleet multitrip VRP with application to a large-scale real case, Journal of Mathematical Modelling and Algorithms, vol. 1, pp. 135-150, 2002.

[210] R. Baldacci, M. Battarra, and D. Vigo, Routing a heterogeneous fleet of vehicles, The Vehicle Routing Problem: Latest Advances and New Challenges, Editors: B. Golden, S. Raghavan, and E. Wasil, Publisher: Springer US, 2008, pp. 3-27.

[211] J. R. Montoya-Torres, J. López Franco, S. Nieto Isaza, H. Felizzola Jimnez, and N. Herazo-Padilla, A literature review on the vehicle routing problem with multiple depots, Computers & Industrial Engineering, vol. 79, pp. 115-129, Jan. 2015.

[212] R. Baldacci, A. Mingozzi, and R. Roberti, Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints, European Journal of Operational Research, vol. 218, no. 1, pp. 1-6, Apr. 2012.

[213] B. Catay, A new saving-based ant algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, Expert Systems with Applications, vol. 37, no. 10, pp. 6809-6817, Oct. 2010.

[214] A. Subramanian, L. M. A. Drummond, C. Bentes, L. S. Ochi, and R. Farias, A parallel heuristic for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, Computers & Operations Research, vol. 37, no. 11, pp. 1899-1911, Nov. 2010.

[215] M. Boudia, C. Prins, and M. Reghioui, An effective memetic algorithm with population management for the split delivery, Hybrid Metaheuristics. Lecture Notes in Computer Science, 2007, pp. 16-30.

[216] T. Vidal, T. G. Crainic, M. Gendreau, N. Lahrichi, W. Rei, T. Vidal, T. G. Crainic, M. Gendreau, and N. Lahrichi, A hybrid genetic algorithm for Multidepot and Periodic Vehicle Routing Problems, Operations Research, vol. 60, no. 3, pp. 611-624, 2012.

[217] F. Li, B. Golden, and E. Wasil, The open vehicle routing problem: Algorithms, large-scale test problems, and computational results, Computers & Operations Research, vol. 34, no. 10, pp. 2918-2930, Oct. 2007.

[218] M. Salari, P. Toth, and A. Tramontani, An ILP improvement procedure for the Open Vehicle Routing Problem, Computers & Operations Research, vol. 37, no. 12, pp. 2106-2120, Dec. 2010.

[219] L. Bianchi, M. Birattari, M. Chiarandini, M. Manfrin, M. Mastrolilli, L. Paquete, O. Rossi-Doria, and T. Schiavinotto, Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands, Journal of Mathematical Modelling and Algorithms, vol. 5, no. 1, pp. 91-110, Dec. 2005.

[220] V. Pillac, C. Gu, V. Pillac, C. Gu, V. P. Illac, and A. M. Edaglia, Dynamic Vehicle Routing Problems: State of the art and prospects, Technical Report 10/4/AUTO. 2011, <hal-00623474>.

[221] M. Andres Figliozzi, The time dependent vehicle routing problem with time windows: Benchmark problems, an efficient solution algorithm, and solution characteristics, Transportation Research Part E: Logistics and Transportation Review, vol. 48, no. 3, pp. 616-636, May 2012.

[222] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, A review of dynamic vehicle routing problems, European Journal of Operational Research, vol. 225, no. 1, pp. 1-11, Feb. 2013.

[223] G. Desaulniers, Branch-and-Price-and-Cut for the Split-Delivery Vehicle Routing Problem with Time Windows, Operations Research, vol. 58, no. 1, pp. 179-192, Feb. 2010.

[224] B. Eksioglu, A. V. Vural, and A. Reisman, The vehicle routing problem: A taxonomic review, Computers & Industrial Engineering, vol. 57, no. 4, pp. 1472-1483, Nov. 2009.

[225] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, The vehicle routing problem: State of the art classification and review, Computers & Industrial Engineering, Dec. 2015, vol.99 pp. 300313.

[226] N. De Jaegere, M. Defraeye, and I. Van Nieuwenhuyse, The vehicle routing problem: State of the art classification and review, KU Leuven, Faculty of Economics and Business, Report KBI_1415, 2014.

[227] G. B. Dantzig, and J. H. Ramser, The Truck Dispatching Problem, Management Science, vol. 6, no. 1, pp. 80-91, 1959.

[228] N. P. Company, N. Christofides, and A. Mingozzi, Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations, Mathematical Programming, vol. 20, pp. 255-282, 1981

[229] G. Laporte and Y. Nobert, Exact Algorithms for the Vehicle Routing Problem, Surveys in Combinatorial Optimization, vol. 132, M. M. Silvano Martello Gilbert Laporte and C. Ribeiro, Eds. North-Holland, 1987, pp. 147-184.

[230] W. W. Garvin, H. W. Crandall, J. B. John, and R. A. Spellman, Applications of linear programming in the oil industry, Journal of Management Science, vol. 3, pp. 407-430, 1957.

[231] P. Toth, and D. Vigo, Models, relaxations and exact approaches for the capacitated vehicle routing problem, Discrete Applied Mathematics, vol. 123, Issues 1-3, pp. 487-512, 2002.

[232] G. Pahl, W. Beitz, J. Feldhusen, and K.-H. Grote, Conceptual design, in Engineering Design SE - 6, Springer London, 2007, pp. 159-225.

[233] B. Motyl, A survey on systematic innovation strategies for product design and development, International conference on innovative methods in product design (IMPROVE), 2011.

[234] M. Gries, Methods for evaluating and covering the design space during early design development, Integration, the VLSI Journal, vol. 38, no. 2, pp. 131-183, Dec. 2004.

[235] M. Buonanno and D. Mavris, A method for aircraft concept selection using multicriteria interactive genetic algorithms, Aerospace Systems Design Laboratory, Georgia Institute of Technology, Report under Grant NGT-1-02006, 2005.

[236] I. C. Parmee, and J. Abraham, User-centric evolutionary design systems - the visualisation of emerging multi-objective design information, International Conference on Computing in Civil and Building Engineering (ICC-CBE), 2004, pp. 1-11.

[237] H. Takagi, Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation, Proceedings of the IEEE, 2001, vol. 89, no. 9, pp. 1275-1296.

[238] T. Lindberg, B. Jobst, and C. Meinel, Is there a need for a design thinking process??, Design of thinking research symposium, 2010, vol. 8, no. October, pp. 243-254.

[239] S. Purao, M. Rossi, and A. Bush, Towards an understanding of the use of problem and design spaces during object- oriented system development, Information and organization, vol. 12, pp. 249-281, 2002.

[240] R. H. Storer, S. D. Wu, and V. Renzo, New search spaces for sequencing problems with application to job shop scheduling, Journal of management science, vol. 38, no. 10, pp. 1495-1510, 1992.

[241] Y.-C. Liu, A. Chakrabarti, and T. Bligh, Towards an ideal approach for concept generation, Design Studies, vol. 24, no. 4, pp. 341-355, Jul. 2003.

[242] S. Pugh, Total design: Integrated methods for successful product engineering, Wokingham, UK: Addison-Wesley Pub, 1991, p. 296.

[243] D. Whitley, A genetic algorithm tutorial, Statistics and Computing, vol. 4, pp. 65-85, 1994.

[244] T. L. Saaty, Decision making with the analytic hierarchy process, International journal of services sciences, vol. 1, no. 1, 2008.

[245] J. Figueira, V. Mousseau, and B. Roy, Electre Methods, in Multiple Criteria Decision Analysis: State of the Art Surveys, New York, NY: Springer New York, 2005, pp. 133-153.

[246] K. Govindan and M. B. Jepsen, ELECTRE: A comprehensive literature review on methodologies and applications, European Journal of Operational Research, vol. 250, no. 1, pp. 1-29, Apr. 2016.

[247] S. Vinodh and R. J. Girubha, Sustainable concept selection using ELECTRE, Clean Technologies and Environmental Policy, vol. 14, no. 4, pp. 651 -656, Nov. 2011.

[248] J. P. Brans and P. Vincke, A Preference Ranking Organisation Method: (The PROMETHEE Method for Multiple Criteria Decision-Making), Management Science, vol. 31, no. 6, pp. 647-656, 2011.

[249] S. Vinodh and R. Jeya Girubha, PROMETHEE based sustainable concept selection, Applied Mathematical Modelling, vol. 36, no. 11, pp. 5301 -5308, Nov. 2012.

[250] P. Resnick and H. R. Varian, Recommender Systems, Communications of the Association for Computing Machinery (ACM), vol. 40, no. 3, pp. 56-58, 1997.

[251] R. Burke, Hybrid Recommender Systems: Survey and, User Modelling and User-Adapted Interaction, vol. 12, pp. 331-370, 2002.

[252] J. R. Quinlan, Decision trees and decision-making, IEEE Transactions on Systems, Man, and Cybernetics, vol. 20, no. 2, pp. 339-346, Mar. 1990.