Analysing and Levelling Manufacturing Complexity in Mixed-Model Assembly Lines

Analyseren en nivelleren van productiecomplexiteit
in mixed-model-assemblagelijnen

Ludimila Luiza De Lima Gabriel Zeltzer

Promotoren: prof. dr. E.-H. Aghezzaf, prof. dr. V. Limère
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Bedrijfskundige Systeemtechnieken
en Operationeel Onderzoek

Vakgroep Industriële Systemen en Productontwerp
Voorzitter: prof. dr. E.-H. Aghezzaf
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2015 - 2016

UNIVERSITEIT
GENT

ISBN 978-90-8578-880-5
NUR 804
Wettelijk depot: D/2016/10.500/12

Promotoren: Prof. dr. El-Houssaine Aghezzaf
Prof. dr. Veronique Limère

Leden van de examencommissie:
Prof. dr. Hendrik Van Landeghem (voorzitter) Universiteit Gent
Prof. dr. Birger Raa (secretaris) Universiteit Gent
Prof. dr. Sidharta Gautama Universiteit Gent
Prof. dr. Ann Nowè Vrije Universiteit Brussel
Prof. dr. Mario Vanhoucke Universiteit Gent

Universiteit Gent
Faculteit Ingenieurswetenschappen
Vakgroep Technische Bedrijfsvoering
Technologiepark 903, B-9050 Zwijnaarde, België
Tel.: +32-9-264.55.02
Fax.: +32-9-264.58.47

Acknowledgements

First of all, I would like to express my sincere and deepest gratitude to my promotor Prof. El-Houssaine Aghezzaf. I would like to thank him for his support, comments and advice. I am grateful for his endless motivation and encouragement. Working with Houssaine is a honour and his guidance as my mentor was essential to the conclusion of this work.

I would like to thank Prof. Veronique Limère for co-advising my thesis. Her helpful insights and critical comments were extremely appreciated, her help was imperative during different phases of my Phd. I am also thankful to Prof. Hendrik Van Landeghem, who gently accepted to chair the examination board, for his prompt feedback whenever needed over the past years.

Many thanks to all the members of the examination board: Prof. Ann Nowé, Prof. Birger Raa, Prof. Mario Vanhoucke and Prof. Sidharta Gautama, whose remark and suggestions definitely helped to improve the quality of this dissertation.

I also would like to thank Veerle Van Beversluys and Kurt De Cock, who always helped with all necessary logistic support. Also many thanks to all colleagues from the Industrial Systems Engineering and Product Design Department for their friendship.

In addition, I would like to thank the industrial collaborators for their input throughout this work: Ford Genk, SAS Automotive, SML, VCST, Van Hool Industrial Vehicles, Volvo Cars Ghent (Belgium) and Torslanda (Sweden) and Volvo Trucks Ghent. I also gratefully acknowledge the financial support of Flanders Drive.

A special thank you to Anita Bernard and Wim De Bruyn for their incredible kindness and support over the past twelve years. Also to Prof. Nuno Silva, who kindly accepted me as a fellowship researcher almost nine years ago and introduced me the researching environment.

Furthermore, I am really grateful to Family Soares Vasco, without their kindness and support during the very beginning of my academic life, I would not be able to reach this moment.

To my close friends far away: André Silva, Hélio Martins, Ivo Pereira, Jorge Santos and Raquel Martins. The distance have not changed our friendship and over the years, I had the privilege to continue sharing great conversations and moments with them. I am also particularly thankful to my precious friend Daisy Kirszenworcel for her wise advice and for pushing me through the difficult times.

Last but not the least, I would like to thank my family. To my siblings for their great sense of humour, their way of looking to the bright side of life always inspire me. To my mom who always believed on my capabilities and countless times encouraged me. To my dad for his efforts and unconditional support during all these years. I also own a special thanks to Nir for being my rock during hard times and celebrating the good moments. Lastly, to my children for all the hope, happiness and light they bring to my days.

Ghent, December 2015
Luiza

Contents

List of Figures

List of Tables

Nederlandse samenvatting –Summary in Dutch–

De laatste jaren kent de auto-industrie een snelle toename op vlak van variaties in modellen en maatwerk. Nieuwe modellen, die vooral worden geïntroduceerd als reactie op de vraag van de klant, hebben uitgebreide keuzemogelijkheden voor verschillende varianten (motortype, comfort, kleurenpalet, enz.) en opties (entertainmentsysteem, start/stop-functie, enz.). Deze grote variabiliteit vergroot de complexiteit van fabrieksprocessen en werkstations, en heeft daardoor een rechtstreekse invloed op de complexiteit van het productiesysteem als geheel. De verschuiving van massaproductie naar massale productie op maat is een trend die zich lijkt voort te zetten in de nabije toekomst, gedreven door de concurrentiestrijd tussen autofabrikanten om in hun traditionele markten marktaandeel te behouden en om marktaandeel te winnen in nieuwe, snelgroeiende markten. Om te voldoen aan de toenemende maatwerkopties, moet er met eenzelfde mixed-model assemblagelijn een grote verscheidenheid aan modellen gebouwd kunnen worden.

Mixed-model assemblagelijnen zijn gestroomlijnde productiesystemen die typisch worden geconfronteerd met het lijnbalanceringsprobleem (ALBP, assembly line balancing problem), een combinatorisch optimaliseringsprobleem voor de optimale verdeling van het assemblagewerk over de werkstations, met een bepaalde doelstelling voor ogen. Het oplossen van mixed-model lijnbalanceringsproblemen

(MMALBPs) is veel complexer dan voor een enkel model, omdat de werkdruk gelijkmatig verdeeld moet worden over alle werkstations voor alle modellen, om overbelasting of inactiviteit te voorkomen.

Ondanks de recente aandacht voor productiecomplexiteit en de uitgebreide studie van het ALBP, is er weinig onderzoek verricht naar hoe complexiteit kan toegepast worden om lijnefficiëntie te verhogen. De complexiteit van het productieproces is de afgelopen jaren een belangrijk aandachtspunt geweest van vele onderzoekers en fabrikanten, maar tot nog toe werd complexiteit niet in overweging genomen bij het balanceren van assemblagelijnen. Het analyseren, meten en toepassen van complexiteit bij het balanceren is nieuw en onbekend terrein, vooral binnen praktische industriële scenario's. Het ontwikkelen van methoden voor het balanceren van mixed-model assemblagelijnen, die van complexiteit gebruik maken, blijft dus één van de belangrijkste uitdagingen voor hedendaagse productiesystemen. In dit proefschrift voeren we een grondige analyse uit van productiecomplexiteit en diverse balanceermethoden voor mixed-model assemblagelijnen. Het uitgevoerde onderzoek is gebaseerd op een studie van complexiteit van echte werkstations en de relatie met hun lijnbalancering. Onze benadering van het MMALBP streeft bijgevolg naar het balanceren van de werkbelasting over de werkstations, met minimalisatie van overbelasting en complexiteit van werkstations. Dit doctoraatsonderzoek stelt de eerste toepassing voor van lijnbalanceringsoplossingen die rekening houden met de empirische complexiteitsanalyse en -metingen.

In hoofdstuk 2 verkennen we de bestaande wetenschappelijke literatuur over mixed-model lijnbalanceringsproblemen en productiecomplexiteit. We bescrhijven hoe complexiteit van werkstations via twee benaderingen: het analyseren van empirische gegevens en het meten van het onzekerheidsniveau middels entropie.

Hoofdstuk 3 richt zich op de definitie en de analyse van complexiteit op basis van empirische gegevens. Eerst wordt de complexiteit van werkstations gedefinieerd, gevolgd door een identificatie van drijfveren van complexiteit. Vervolgens worden drie lineaire en twee statistische modellen ontwikkeld om werkstations te categoriseren als zijnde met hoge of lage complexiteit.

Het doel van de studie in hoofdstuk 4 is om complexiteit te benutten om tot een optimale taakverdeling te komen waarbij geen overbelasting wordt geregistreerd bij de werkstations, en dit voor eender welk model dat samengesteld wordt. Om dit te bereiken wordt eerst een gemengd geheeltallig lineair programmeringsmodel (MILP) ontwikkeld. Daarna worden twee heuristische algoritmes ontworpen, waarbij de eerder ontwikkelde complexiteitsmeting wordt geïntegreerd: de algemene oplossing maakt gebruik van één van deze heuristieken om een eerste oplossing te genereren, terwijl de andere wordt gebruikt als verbeteringsprocedure. Het eerste algoritme bestaat uit een hybride constructie-heuristiek voor het genereren van een initiële lijnbalancering, en combineert verschillende algoritmen voor het in rekening nemen van de volgorde van de taaktoewijzing, de doelfunctie en de variabiliteit van de uitvoeringstijden van de taak. Bij de uitvoering ervan, waarbij mogelijke initiële oplossingen aan het licht komen, wordt de gemeten complexiteit $Complexity_{AJ_k}$ en werkbelasting $Load_k$ van werkstations dynamisch berekend tijdens de interacties van het algoritme. De resulterende gecombineerde meting wordt vervolgens gebruikt in de volgende taaktoewijzing. Het tweede algoritme is een verbeteringsheuristiek waarbij een bepaalde herbalanceringsprocedure gebruikt wordt om de eerste oplossing verder te verbeteren en dus de resterende overbelasting van het werkstation te verminderen. Tijdens dit proces worden de taken van werkstations met de hoogste gecombineerde metingen gepermuteerd naar die met de laagste, en wordt dus de overbelasting voor elk werkstation en model verminderd.

In hoofdstuk 5 worden de rekenkundige resultaten van de praktische toepassing van de ontwikkelde benaderingen binnen een industriële context uiteengezet. Deel 5.1. beschrijft de strategie gebruikt voor gegevensanalyse. Uitgaande van de door de fabrikanten verstrekte gegevens wordt hier een samenvatting gegeven van hoe de gegevens werden opgehaald, gestructureerd en gebruikt. Om de resultaten te genereren, passen we onze modellen toe op de empirische dataset. We beoordelen eerst hoe goed elk van de vijf modellen in staat is om de complexiteit van werkstations als hoog of laag te karakteriseren; de meest accurate classificatie wordt bereikt door het CALC_SAMPLE model. Dit is een lineair model op basis van de vier variabelen van het LOGIT_SAMPLE model, waarbij de gewichten dezelfde zijn als bepaald door het statistisch model. Ten tweede presenteren we de resultaten bekomen door toepassing van het MILP-model. Dit model is in staat om een optimale oplossing voor het probleem te leveren, maar het neemt niet alle beperkingen in overweging. Ten derde bespreken we de resultaten gegenereerd door het gebruik van heuristische oplossingsmethoden. De belangrijkste conclusie van deze studie is dat overbelasting aanzienlijk kan worden verminderd wanneer rekening wordt gehouden met complexiteit van werkstations. Tot slot maken we een algemene vergelijking van alle resultaten, inclusief de karakterisering van de complexiteit van werkstations, entropie gebaseerde complexiteit en overbelasting.

Hoofdstuk 6 presenteert een aantal conclusies en voorstellen in het teken van toekomstig onderzoek.

English Summary

In recent years, the automotive industry has witnessed a rapid increase in model variety and customization. New models, which are mainly being introduced in response to consumers demand, feature long lists of choices in terms of variants (engine model, comfort level, colour palette, etc.) and options (entertainment system, start/stop functionality, etc.). This high variability increases the complexity of factory processes and workstations and thus impacts directly upon the complexity of the manufacturing system as a whole. The shift from mass production to mass customized production is a trend that looks likely to continue in the foreseeable future, driven by automotive manufacturers' struggle to maintain market share in their traditional markets and seize market share in new, fast-growing markets. To cope with this intensified customization, automotive assembly platforms are designed to be capable of assembling a large range of relatively different models. That is they become mixed-model assembly lines. This implies that a high variety of tasks are to be performed at each workstation. As a consequence, the manufacturing complexity at these workstations increases.

Mixed-model assembly lines are flow-line production systems that typically encounter the assembly line balancing problem (ALBP), a combinatorial optimization problem involving the optimal partitioning of assembly work among the workstations with a particular objective in mind. Subsequently, solving mixed-model assembly line balancing problems (MMALBPs) is much more complex than single-

model cases, as workload must be smoothed for all workstations and all models in order to avoid overload or idle time.

Despite the recent focus on manufacturing complexity and the extensive study of the ALBP, little research has explored how complexity can be applied to optimize line efficiency. Manufacturing complexity has been a key concern of many researchers and manufacturers in recent years, however, practical procedures to level complexity have not yet been considered and investigated when balancing the assembly lines. Analysing, measuring and monitoring complexity while creating line balancing solutions is a new and unexplored topic, especially when using real industry scenarios. In this dissertation, we propose an approach that can be used to monitor manufacturing complexity at each workstation while balancing the mixed-model assembly lines.

The research carried out relies on an investigation of real MMAL's aiming to develop a deep analysis of complexity. The goal is to understand what and how complexity is generated, in order to cope and reduce the high complexity and its impacts in the line. During several visits and workshops carried out in collaboration with manufactures, we could observe that work load distribution is directly related with models variety, as tasks' time might differ from model to model. We first explored the existing scientific literature on the mixed-model assembly line balancing problem and manufacturing complexity in Chapter 2. Then, manufacturing complexity is investigated using two approaches: (1) an empirical analysis approach based on data collected in the Field and (2) a quantitative analysis approach measuring the level of uncertainty by means of entropy.

In order to investigate the impact of complexity on the production performance, one must first delineate the concept and then identify as unambiguously as possible highly complex workstations. In chapter 3, first a clear definition of production complexity is proposed and its main drivers and their impacts are determined. Then, causal relationships between drivers of complexity are modelled. Finally, using data from several plants, three linear and two statistical models are derived to empirically classify workstations as having either high or low complexity.

In Chapter 4, first a quantitative complexity measure based on entropy and task times' variations is developed. This measure is used to evaluate the level of manufacturing complexity at each workstation. A mixed-line balancing heuristic procedure is then developed that integrates this monitoring procedure to achieve a workload balance which induces a levelled manufacturing complexity at each workstation. The goal is to level complexity and generate an optimal task assignment where no workstation overload is registered for any of the models being assembled. In order to achieve this goal, a mixed integer linear programming (MILP) model is first constructed. Secondly, two heuristic algorithms are designed, integrating the previously developed complexity measurement: the general solution uses a heuristic to generate an initial solution, while the other is then used as an optimization procedure. The first algorithm consists of a hybrid constructive heuristic for monitoring complexity and generating a balancing solution. During its execution, which results in possible initial solutions, workstations' complexity $Complexity_{AJ_k}$ and workload $Load_k$ measurements are calculated dynamically during the algorithms' iterations. The resulting combined measurement is subsequently taken into account during the next task assignment. The second algorithm is an improvement heuristic involving of a rebalancing procedure that uses the a local search approach to optimize the initial solution and thus reduce remaining workstation work overload. During this process, tasks are permuted from workstations with the highest combined measurements to those with the lowest, thus reducing workstation work overload and levelling complexity for each station and model.

In Chapter 5, we discuss the computational results obtained from the real-life application of the approaches we developed for an industrial case. Section 5.1. describes the data analysis strategy; starting from the information supplied by manufacturers, it summarizes how the data was retrieved, structured and used. In order to generate results, we apply our solutions to the empirical dataset. We first assess how well each of the five models is able to characterize workstation complexity as high or low, finding that the most accurate classification is obtained by the CALC_SAMPLE model. This is a linear model based on the four variables of the LOGIT_SAMPLE model, whose weights are also those determined by the statistical model. Secondly, we present the results obtained by the MILP model. This model is capable of providing an optimal solution to the problem but does not take all constraints into account. Thirdly, we discuss the results generated by the use of heuristic solutions. Finally, we make a general comparison among all results, including workstation complexity characterization, complexity levelling and work overload. This study provides an extensive investigation of manufacturing complexity; and subsequent identification of workstation complexity drivers, workstation complexity characterization and measurement, complexity levelling and mixed-model assembly line balancing.

Chapter 6 presents a number of concluding remarks and provides some suggestions for future research.

Notation

K	is the set of workstations, index k
J	is the set of tasks, index j
O	is the set of operators, index o
M	is the set of models, index m
$KQual_j$	is the set of qualified workstations k for a task j, index $kqual_j$
$OQual_j$	is the set of qualified operators o for a task j, index $oqual_j$
$Pred(j)$	is the set of direct predecessors, index of task $pred(j)$
$maxOp(k)$	is the maximum number of assigned operators assigned to workstation k
AJ_k	is the set of assigned tasks to workstation k
c	represents the cycletime
t_{jm}	is the processing time of task j for model m
b_m	is the demand of model m in the model-mix
$Complexity_{AJ_k}$	is the complexity measurement of workstation k when the set of tasks AJ_k is assigned to k
$Load_k$	is the work load of workstation k
$Overload_{km}$	represents overload at workstation k for model m

List of Acronyms

ALBP	Assembly Line Balancing Problem
ALSP	Assembly Line Sequencing Problem
CPU	Central Processing Unit
EA	Empirical Analysis
EM	Entropic Measurement
GALBP	Generalized Assembly Line Balancing Problem
H	High Complex Workstation
k	Workstation
L	Low Complex Workstation
MILP	Mixed Integer Linear Programming
MMAL	Mixed-Model Assembly Line
MMALBP	Mixed-Model Assembly Line Balancing Problem
R	Rebalancing
SALBP	Simple Assembly Line Balancing Problem

1 Introduction

Over the past three decades, the customization rate in the automotive and vehicle industry has reached its highest level ever. This high level of customization is mainly the result of the industry's efforts to respond to the current highly diversified market demand. As a result, the number of new model introductions per year has grown steadily over time. While these new introductions have helped the industry remain at the forefront of customer satisfaction and new technology, this large variety of models has increased the overall manufacturing complexity of assembly platforms. The shift from mass production to mass customized production inevitably entails a larger number of tools, machines, parts, assembly tasks and processes at workstations. Therefore, mixed-model assembly lines, which are required to manufacture this large variety of different models, have become extremely complex. This is clearly illustrated in Figures 1.1 and 1.2, which offer typical examples of how operators are flooded with information and options in real workstations.

This manufacturing complexity that workers face has received various definitions in the literature. In this dissertation, it is regarded as all the aspects and factors that make an operator's set of tasks mentally difficult, error-prone and stress-inducing because a high degree of alertness is required. With this definition of complexity, we aim to reflect the opinions, judgements and experiences of people who work under different circumstances on the production floor – operators, production engineers, quality controllers and line managers. Needless to say , quantifying manufacturing complexity is a rather difficult process.

Figure 1.1: *Mixed-Model Assembly Line – Border of Line*

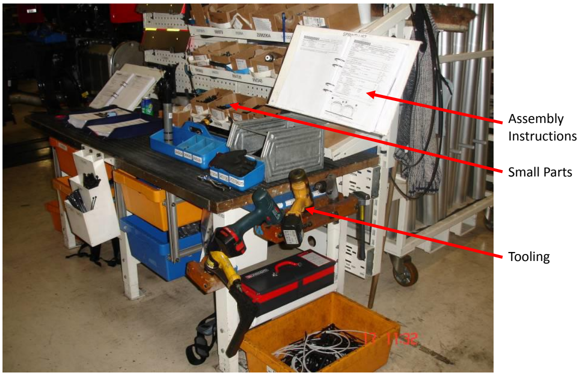
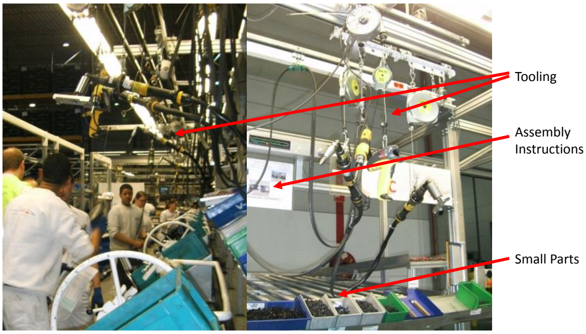


Figure 1.2: *Mixed-Model Assembly Line – Workstation*

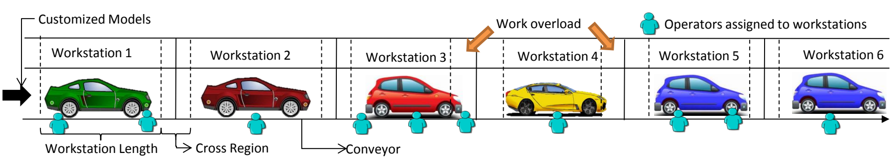


Manufacturing complexity has increased substantially due to a higher degree of uncertainty related to the assembly mix and a larger amount of information that needs to be processed during the assembly process, which in turn results from a greater number of available choices. This high variability and stress make it very difficult for workers to complete all their tasks during each cycle time.

Of course, high model variability also has a direct impact on how the workload is divided across the line, as operating times inevitably differ and workstation loads vary. Spreading workloads across workstations while avoiding operating inefficiencies, such as work overload, is more difficult to achieve in a mixed-model context than in single-model one. Thus, obtaining an efficient line balance in which workstation load stays within the cycle time for all models, and in which complexity is also levelled, is an extremely challenging problem. In this dissertation, we propose a first attempt to solve this challenging issue.

Mixed-model assembly lines consist of a conveyor system moving at a constant speed; different customized models are assembled on the same line (i.e. platform). Along the assembly line, operators at workstations do not move beyond specified boundaries (i.e. cross regions). This configuration is clearly shown in 1.3.

Figure 1.3: *Mixed-Model Assembly Line Overview*



Work overload refers to the amount of assembly work that remains incomplete when an operator reaches a workstation boundary (Matanachai and Yano, 2001). In other words, the operators are not able to finish the work within the cycle time. When work overload occurs, the following counter measures may take place: (i) the operator and their supervisor rush to finish the work, (ii) the remaining work is completed at an intermediate repair station or at the end of the line, (iii) the line is stopped for the operator to finish the work, or (iv) a utility worker is assigned to finish the work. In all four cases, work overload adversely affects cost and quality. A reduction in work overload does not only improve efficiency, but also product quality. This, in turn, has a long-term impact on market share and profitability. If complexity is also levelled, a mixed-model assembly line can become close to ideal. As a consequence, there is a growing interest in the study of complexity in mixed-model assembly lines. In this dissertation, we will focus on manufacturing complexity as induced by assembling a large number of models and model variants on a mixed-model assembly line.

In the following section, we begin by defining workstation complexity and identifying the drivers of complexity. Based on these drivers, we propose statistical models to characterize workstations as 'high' or 'low'. Subsequently, we propose a complexity measurement, with which we may offer solutions for the generation and optimization of the MMALBP (Mixed-Model Assembly Line Balancing Problem). Our approach aims to balance workload while levelling manufacturing complexity.

1.1 The Assembly Line Balancing Problem

The research carried out was developed in collaboration with the automotive industry and their suppliers. The production units under study are responsible for the production of vehicles and original equipment. In this section, we will describe the Mixed-Model Assembly Line (MMAL) model which is used to balance the MMALBP. The following parameters are used:

Notations:

The objective function considered in balancing the line aims to minimize workstation overload and is represented by Eq. 1.1. The main goal is to generate task assignments in which workstation overload and complexity are kept to a minimum.

$$Minimize \sum_{k \in K} \sum_{m \in M} Overload_{km} \tag{1.1}$$

Where:
$Overload_{km}$ represents overload at workstation k for model m and is given by:

$$Overload_{km} = max\ (0, \sum_{j \in AJ_k} t_{jm} - c) \tag{1.2}$$

MMALBP can be formally described as: (i) given a set M of different models, a set K of workstations, a set O of operators, a set J of tasks; (ii) each task j needs to be assigned to a workstation

M	the set of models, index m
K	the set of workstations, index k
O	the set of operators, index o
J	the set of tasks, index j
$KQual_j$	the set of qualified workstations k for a task j, index $kqual_j$
$OQual_j$	the set of qualified operators o for a task j, index $oqual_j$
$maxOp(k)$	the maximum number of operators in the workstation k
j_k	the assigned task j to workstation k
o_k	the assigned operator o to workstation k
c	cycle time
t_{jm}	the processing time of task j for model m
$Pred(j)$	the set of direct predecessors, index of task $pred(j)$
b_m	the demand of model m in the model-mix, with $\sum_{m \in M} b_m = 1$
AJ_k	the set of tasks j assigned to workstation k

k respecting the restrictions of (iii) precedence relationships among tasks, (iv) workstations and (v) operators qualified to execute the task concerned.

Tasks can be assigned and executed only at a set of qualified workstations $KQual$ by a set of qualified operators $OQual$. Each model m requires the execution of a subset of tasks (also called assembly operations). Each task j has a processing time t_{jm} (also called 'task time' or 'operation time'), depending on the model m. Each workstation k also has a maximum number of assigned operators $maxOp(k)$ and a subset of assigned tasks AJ_k.

We consider the MMALBP on the basis of the following assumptions:

- Precedence relationships between tasks are represented by a single diagram in all models. Different models may require distinct subsets of tasks. Each task is executed for at least one model and its duration is known; when a specific task is not executed, its task duration is defined as 0.
- A workstation is considered to be qualified when it is correctly equipped with the required tools, parts, fixtures and instructions

for performing the task concerned.

- The task assignment to workstations is identical for all models, and similar tasks are executed at one particular workstation.
- Models may vary in demand. This demand is expressed as a percentage and represents the product mix.
- Operators are considered qualified when they are able to carry out the assigned task, i.e. when they have the appropriate training and experience and are physically and psychologically able.
- The number of operators per station is fixed and known. Operators' walking distances are limited to specified workstation boundaries; they cannot move beyond these boundaries.
- 'Cycle time' refers to the constant speed of the line and there are no buffers between workstations.

1.2 Complexity in Manufacturing Systems

The introduction of new models has increased the complexity of factory processes, workstations and manufacturing systems. This is especially true in the automotive industry, where customization is rapidly growing. New models are regularly introduced in response to consumer needs. These feature long lists of choices, both in terms of variants (e.g. engine model, comfort level, colour palette) and options (e.g. entertainment system, start/stop functionality). This high variability directly impacts the complexity of manufacturing systems.

This trend is likely to continue in the foreseeable future, driven by automotive manufacturers' struggles to maintain market share in their traditional markets and seize market share in new, fast-growing markets. It will increase the pressure on all automotive assembly plants to boost productivity and lower engineering change-over times.

Currently, relatively little is known about how manufacturing complexity relates to production performance. In this dissertation, we will investigate complexity in relation to workstations on mixed-model lines. First, we will address three main questions: (1) What causes complexity?, (2) How can complexity be defined?, and (3) What is its impact? Subsequently, we will explore possibilities to measure complexity and improve line balance.

1.3 Contribution

Numerous studies have already investigated manufacturing complexity and the ALBP in particular. Various studies have been conducted which associate manufacturing complexity to specific products, process structures and human operators. However, it should be emphasized that while ALBPs have been studied intensively, the relationship between complexity and mixed-model assembly line balancing has been neglected.

The research carried out as part of this dissertation has three main objectives. The first is to investigate manufacturing complexity, aiming to define and characterize workstation complexity within this broader context. The second objective is to propose a quantitative complexity measurement for workstations related to tasks assignment. Finally, as a third objective we intend to propose and implement a novel solution to mixed-model assembly line balancing by taking the complexity measurement into account. Our ultimate goal is to balance workstation workload and level its complexity at the same time.

Having presented the principal objectives of this dissertation, we now define the contributions of this work by summarizing the approach we developed through a set of research questions. The main research questions through which we intend to meet this goal are introduced below.

General Research Question:

How can manufacturing complexity be evaluated and managed in mixed-model assembly lines?

In our research, we have focused on realistic scenarios in which MMALs and manufacturing complexity are common. We offer extensive insights into how complexity impacts automotive manufacturing production and provide new knowledge by introducing innovative approaches to exploring complexity and balancing mixed-model assembly lines.

This general research question can be broken down into three subquestions, which will provide a more detailed overview of the relevance of this PhD thesis.

Research Question 1:

What are the drivers that determine manufacturing complexity in mixed-model assembly lines, and how can these drivers be used to classify workstations as being ‘high’ or ‘low’ in complexity?

In order to improve the efficiency of complex assembly lines, it is first necessary to understand the concept of complexity itself. Therefore, an extensive analysis will be conducted to arrive at a suitable definition that identifies the drivers behind complexity. With these drivers, workstations can then be characterized and divided into different models based on their complexity.

Research Question 2:

How can manufacturing complexity be levelled in mixed-model assembly lines workstations while balancing workload (minimizing work overload)?

Continuing from the previous research question, an objective, quantitative measurement for workstation complexity can be proposed and integrated into optimization approaches. To provide a solution for mixed-model complex assembly line balancing, a set of algorithms can be developed and implemented based on the available knowledge of complexity. This approach should generate a satisfactory solution to MMALBP while minimizing workstation overload and levelling complexity.

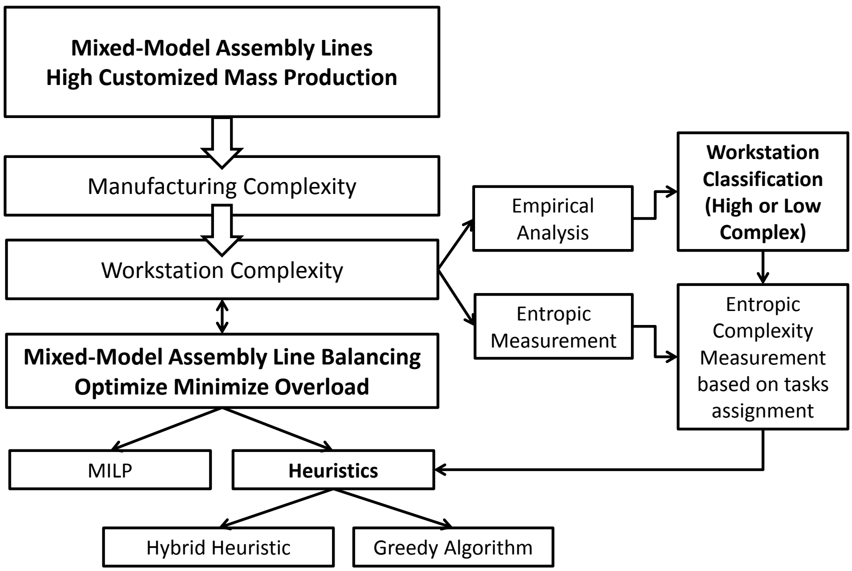
Research Question 3:

What are the results and shortcomings of both approaches when applied to a real world mixed-model assembly line? Analysis of an industry study case.

In order to provide an analysis of the solutions developed, computational tests should be performed on an empirical dataset, thus requiring a case study from the industry.

This research provides an important contribution to the scientific body of knowledge as well as the industry (see Figure 1.4). To level manufacturing complexity and improve line balancing for highly customized MMALs, we initially focused on workstation complexity using two primary methods: (1) empirical analysis based approach and (2) entropic measurement based approach. Empirical analysis was used to identify workstation complexity and provide in-depth knowledge of manufacturing complexity, while entropic measurement quantified workstation complexity based on task assignments. After this initial stage, we concentrated on proposing a solution for the MMALBP by integrating our complexity measurements into heuristic solutions. These heuristics rely on a hybrid heuristic and a greedy algorithm and will be further explained in the course of this dissertation.

Figure 1.4: *Research Global Overview*



1.4 Outline

The remainder of this dissertation is organized as follows. In Chapter 2, we provide a literature review and include an extensive investigation of ALBPs and manufacturing complexity. First, the ALBP will be reviewed and various factors will be explored, such as characterization, related problems and solution methodologies. Subsequently, we will provide an overview of complexity in manufacturing systems. This overview includes diverse aspects of complexity, such as its definition, classification and measurement. Finally, we will also demonstrate how the present study contributes to the scientific body of knowledge.

Chapter 3 presents a definition and analysis of production complexity, first relying on a subjective working definition of complexity. Next, we will detail the causal model of complexity drivers, before describing the methodology developed to distinguish 'high' or 'low' complex workstations. Finally, a number of statistical models will be described which were developed using real production data and can be used to classify complex workstations.

In Chapter 4, evolving from a subjective definition of complexity based on judgement, we propose an approach that can be used to monitor manufacturing complexity at each workstation while balancing the MMALs. First, a quantitative complexity measure based on entropy and task time variations is developed. This measure is used to evaluate the level of manufacturing complexity at each workstation. A mixed-line balancing heuristic procedure integrating this monitoring procedure is then created to achieve a workload balance which induces a levelled manufacturing complexity at each workstation.

Chapter 5 describes the computational results of the solutions developed when applied to an industry study case. In Section 5.1. we explore the data utilized, and in section 5.2. complex workstations are characterized. Section 5.3. demonstrates the performance of the model and heuristics in solving the balancing problem, while section 5.4. presents an overview and comparison of the results obtained.

Chapter 6 concludes this dissertation and offers some recommendations for future research.

2

Literature Review

Mass production customization in the automotive industry is becoming a fast-growing research area. In recent years, many researchers, such as Wiendahl and Scholtissek (1994) and MacDuffie et al. (1996), have begun to study the manufacturing complexity resulting from this mass production customization. Meanwhile, the ALBP, and its variants have been intensively studied for more than seven decades, starting with the work of Salveson (1955) and Jackson (1956), and continue to be the focus of numerous studies today.

In this chapter, we provide an extensive overview of the literature on assembly line balancing and manufacturing complexity in MMALs. In section 2.1 important concepts are first defined. Next, section 2.2 details the existing literature regarding assembly line balancing, while section 2.3 zooms in on manufacturing complexity. Finally, section 2.4 concludes this chapter with a detailed overview of how the present study will contribute to the existing body of knowledge.

2.1 Definitions

This section describes important terms related to MMALs. In order to characterize the system and enable a clear understanding of the problem and the approaches used, we will first provide a set of definitions, which rely on the work of Scholl (1999), Becker and Scholl (2006), and Battaïa and Dolgui (2013).

Assembly is the process of collecting and organizing various elements to create a finished product. It is characterized by the elements used and the work necessary to combine them. Relationships among elements, material flows, and operations are typically visualized using assembly charts.

A **task or operation** is a portion of the total work content in an assembly process. The time needed to complete a task is called task time or processing time. Relationships between tasks are commonly represented by precedence constraints.

A **workstation** is a segment of an assembly line at which a certain number of tasks are performed. It is mainly characterized by the elements necessary to execute the assigned tasks, such as machinery, equipment and operators. Workstation load is the total work content assigned to a station and is represented by the sum of all processing time needed to complete all tasks.

Precedence constraints occur as a result of technological demands on line and product structures. These constraints involve the order in which tasks must be undertaken. This order is often illustrated by precedence diagrams (i.e. graphs) in which nodes represent tasks and arcs represent the order of connected tasks.

Cycle time is the maximum amount of time dedicated to a workpiece per workstation during assembly. It consists of the time available at each station to perform the assigned tasks. Consequently, cycle time cannot be shorter than the longest task time.

Idle time is the positive difference between cycle time and workstation load. It consists of the remaining time available at the workstation after all assigned tasks have been performed (i.e. all the work on a workpiece has been completed) and before the next workpiece arrives at the workstation. When only one model is being produced, idle time is constant. When several models are being assembled, however, idle times differ and depend on the sequence in which the models are assembled.

Overload is the negative difference between cycle time and workstation load. It represents the extra time necessary to execute all tasks. Work overload may be compensated for by the temporary employment of utility workers, by stopping the line or by another sanction. Whichever 'solution' is selected, work overload is inefficient and expensive and should be kept to a minimum.

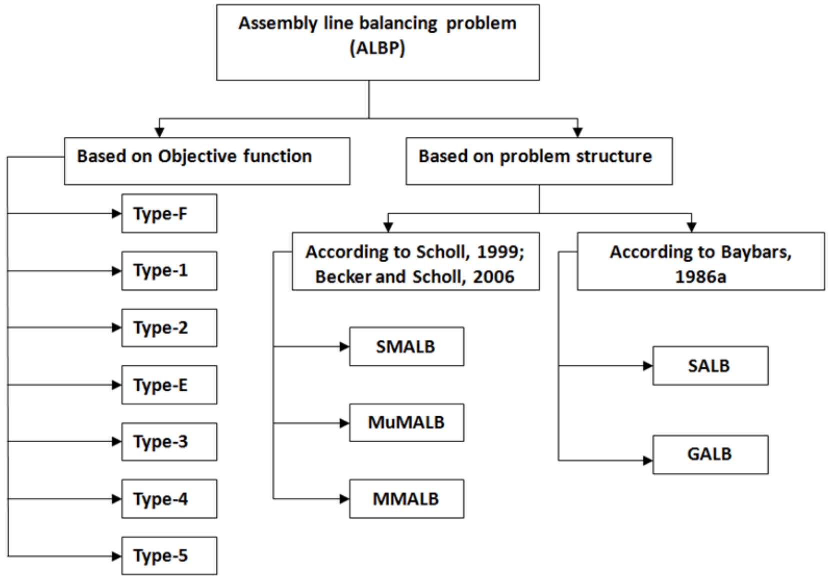
2.2 Assembly line balancing

In this section, we focus on assembly lines and their related problems. Assembly lines are typical flow-oriented production systems and are important building blocks in many manufacturing systems, (Becker and Scholl, 2006). According to Scholl (1999), the ALBP consists of achieving an optimal balanced division of assembly work between workstations with respect to a specific goal. This process involves assigning tasks to workstations as effectively as possible, while satisfying a number of constraints, such as precedence constraints, cycle time, or operator qualifications.

As assembly lines are used for a wide range of production systems, balancing problems can be classified according to various aspects. Uddin and Lastra (2011) extensively study the classification of ALBPs and discuss previous work done by Baybars (1986), Scholl (1999), and Becker and Scholl (2006). ALBPs are either classified by the objective function to be measured and optimized or the problem structure, as summarized in Figure 2.1.

Typical ALBP types are Type F problems, for which the cycle time and number of workstations are given and a feasible line balance needs to be obtained with respect to these two parameters. For Type 1 problems, the number of workstations need to be minimized for a given fixed cycle time, whereas the reverse holds for Type 3 problems (i.e. the cycle time needs to be minimized for a given number of workstations). Type E, on the other hand, requires a reduction in both the number of workstations and cycle time, thus maximizing line efficiency. Types 3, 4 and 5 are described by Kim et al. (1996) and involve a maximization of workload smoothness, a maximization of work relatedness and a combination of both, respectively.

Figure 2.1: *Assembly Line Balancing Problems Classification (Uddin and Lastra, 2011)*



Becker and Scholl (2006) classified ALBPs based on the problem structure and the model variety assembled on the line. Thus, problems can occur on (1) single-model assembly lines (SMALB), which manufacture one homogeneous product, (2) on multi-model assembly lines (MuMALB), which manufacture several products on one or more lines, or (3) on mixed-model assembly lines (MMALB), which manufacture several models of the same basic product.

Traditionally, it was the simple assembly line balancing problem (SALBP) that was most commonly investigated. Recently, however, significant research efforts have been made to model and solve more realistic problems related to generalized assembly line balancing problems (GALBPs). Baybars (1986) details the difference between SALBPs and GALBPs: the former consider a single, straight assembly line used for only one type of product with a limited set of constraints, while the latter involve additional decisions, constraints and optimization objectives.

A complementary classification for MMAL problems was introduced by Merengo et al. (1999), distinguishing between horizontal and vertical balancing concepts. Horizontal balancing considers the workload allocated to each workstation for each model and attempts to smooth out its varying workloads caused by the distinct task times of the model. Vertical balancing, on the other hand, considers the average load of each workstation and aims to align all workstation times for each model separately.

ALBPs are typical combinatorial optimization problems in which an optimal solution is to be found from a finite, and usually very large, set of feasible solutions. Generally, the methods used to solve these problems are classified as either exact or approximate (Battaïa and Dolgui, 2013). Exact methods can find the optimal solution to a problem but are time-consuming and machine-intensive due to the NP -hardness nature of ALBPs. Approximate methods, on the other hand, might not generate optimal solutions; however, they can (in most cases) produce quicker and more feasible solutions within an acceptable computational timeframe. They commonly involve different approaches such as bounded exact methods, simple heuristics and metaheuristics. They are also used to delineate possible solutions that reflect the characteristics of real-world complex assembly lines, as additional constraints can be added. Thus, instead of exact procedures that find optimal solutions to simplified problems, heuristic procedures are used to find solutions to more complex problems (Simaria, 2006).

Many researchers focused on solving balancing problems by using metaheuristic techniques such as genetic algorithms (Simaria, 2006; Haq et al., 2006; Sivasankaran and Shahabudeen, 2014), simulated annealing (Kirkpatrick et al., 1983; McMullen and Frazier, 1998; Özcan et al., 2010; Erel et al., 2001) and ant colony optimization (Simaria and Vilarinho, 2009; Akpınar et al., 2013; McMullen and Tarasewich, 2003). The first metaheuristic technique, genetic algorithms, involves iterative search procedures based on the biological process of natural selection and genetic inheritance. The second technique, simulated annealing, is a typical neighbourhood method analogous to simulating the physical annealing of solids. Finally, the third technique are ant colony algorithms, which are based on the behaviour of insect societies (Simaria, 2006). Applying metaheuristics to general problems is quite common. However, since the mixed-model assembly line balancing problem involves specific characteristics and restrictions, heuristics may be more suitable than metaheuristics because of the possibilities afforded by flexible implementation.

In most cases where heuristics are applied to obtain an assembly line balance, priority rules are used to determine task assignment. The most commonly employed priority rules are based on task attributes such as task time or the number of preceding tasks (Battaïa and Dolgui, 2013). Various solutions have been proposed by researchers, among which Helgeson and Birnie (1961), Hoffmann (1963) and Boctor (1995) are the most notable ones.

Helgeson and Birnie (1961) proposed a well-known heuristic called Ranked Positional Weight (RPW), in which tasks are ranked in descending order according to positional weight. This weight is the result of the sum of the task time and the processing times of all successive tasks. Gonçalves and De Almeida (2002) classified heuristic priority-based procedures for constructing a set of candidate operations as workstation-oriented and operation-oriented. When a task assignment is workstation-oriented (Talbot et al., 1986), it starts with the first workstation and then considers the other workstations successively. When a task assignment is operation-oriented (Hackman et al., 1989), the operation/task with the highest priority is first chosen from all the available operations and is assigned to the first possible workstation.

Local algorithms are also used to generate assembly line balancing solutions (McGovern and Gupta, 2003), especially when combined with priority-based rules or specific problem constraints. Improvement heuristics starts from an existing solution and aims to optimize it using different procedures. It is broadly based on the use of a local search as this method traditionally focuses on combinatorial optimization problems. Reiter and Sherman (1965) proposed local search to solve the travelling salesman problem, which has since been extended and applied to many different areas, such as artificial intelligence, operations research and engineering. T he classical methods of artificial intelligence can be exploited and applied to optimize ALBPs as the assignment of tasks can be characterized as a combinational optimization problem with restrictions.

Thomopoulos (1970) proposed a heuristic procedure based on two objectives. The primary objective is to solve a Type 1 assembly problem complemented by a lower bound on workstation times in the aggregate model. The secondary objective is to level workstation utilization by minimizing the sum of absolute differences between total workstation time and average total workstation time for all models and workstations. Another heuristic procedure involving two objectives was developed by Merengo et al. (1999), which involved weighted differences between the maximum workstation time and the workstation times of all other models.

Later, Bukchin et al. (2002) suggested a three-stage solution approach to Type 1 problems. The authors developed a heuristic that minimized the number of stations by initially solving SMALBP-1 for the average (combined) model and that determined the number of stations and assigning tasks common to all models at particular stations. Then, by reassigning the tasks for each model which are specific to the model concerned, this heuristic preserves the previously-made fixed assignments and optimizes the horizontal balancing objective. Finally, by using a local search procedure, it changes the assignment of common tasks and applies the previous stage to complete the solution by assigning specific tasks as described above. As mentioned before, the high flexibility of heuristic procedures means that they can be used to find solutions to more complex problems. They can also be used to produce an upper bound for an exact method (Battaïa and Dolgui, 2012) or generate intermediate solutions (Essafi et al., 2012).

2.3 Complexity in manufacturing processes and systems

In this section, we will first introduce a definition of complexity and the existing taxonomy before presenting an in-depth discussion of complexity measurement. Two groups of approaches can be identified: (i) approaches based on empirical analysis, and (ii) those that explore uncertainty through entropy.

2.3.1 Complexity definition and taxonomy

A number of researchers have formulated definitions of complexity by attempting to define, model and develop valid and useful complexity measures for manufacturing systems. Frizelle (1996), for example, suggested that a useful complexity measure would need to be separable and additive, as its computation would then be simplified for easy analysis by managers. Later, Deshmukh et al. (1998) provided a clear definition of static and dynamic complexity. He stated that static complexity is related to the structure of the system, variety of components and products and number of processes and machines, while dynamic complexity measures unpredictability or uncertainty in the behaviour of the system over a given time period.

In addition to static and dynamic complexity, two more forms can be distinguished: objective and subjective complexity (Gullander et al., 2011). According to this classification, objective complexity focuses on the measurable parameters of the system, whereas subjective complexity acknowledges that the same production system may be perceived in different ways, depending on certain factors.

This work was followed by many other studies which attempted to understand the impact of complexity on MMALs. Kim (1999), ElMaraghy and Urbanic (2003)(2004) proposed a methodology for systematically assessing product and process complexity and their interrelations. In these studies , a matrix methodology and an objective measure of complexity have been suggested to assess the three levels of manufacturing complexity: product complexity, process complexity and operational complexity.

Schuh et al. (2008) identified the main drivers of complexity (i.e. uncertainty, dynamics, multiplicity, variety, interactions, and interdependencies) and stated that a system's complexity is determined by the combination of these properties. Very recently, ElMaraghy et al. (2012) has published a comprehensive overview of complexity models in design and manufacturing. These authors state that the design of systems with reduced complexity is an important issue for further research. They also present a generic map of how manufacturing complexity cascades down from product design to the individual operator, in terms of both cognitive and physical effort.

2.3.2 Complexity measurement

Measuring manufacturing complexity has been a major challenge for years. Researchers have developed a number of different approaches and methodologies in their attempts to find an efficient means of measurement. These approaches are primarily based on empirical analysis and the use of entropy, which will both be discussed below.

Empirical Analysis

MacDuffie et al. (1996) investigated the effects of product variety manufacturing performance, defined as total labour productivity and consumer-perceived product quality. In their article, they analysed complexity measures that capture different aspects of product mixes in assembly plants: model mix complexity, parts complexity and option content. The first, model mix complexity, is based on the number of different platforms, body styles and models and is scaled by the number of different body shops and assembly lines in each plant. Secondly, parts complexity is partially driven by consumer choice and reflects the impact of a larger variety on product design and the supply system. Finally, option content involves the overall level of installed options and is expressed as the percentage of vehicles built with multiple options, as aggregated across all models in a plant. A statistical analysis carried out on data from 70 assembly plants worldwide (gathered as part of the International Motor Vehicle Programme at MIT) revealed significant negative correlations between these complexity measures and manufacturing performance.

Deshmukh et al. (1998) offered a measure grounded in the part/mix ratio of various models and aimed to measure static complexity regarding the number of parts, machines and operators required to process the part mix. This measurement is based on the information available from production orders and process plans. Later on, Urbanic and ElMaraghy (2006) also considered information flow and concentrated specifically on the quantity, diversity and content of this information. They developed a model to determine process complexity and evaluate alternatives and risk areas regarding product, process and operation tasks in the design stage .

Entirely different approaches were introduced by Meyer and Curley (1995), and Falck et al. (2012). Meyer and Curley (1995) analysed the impact of subjective complexity during the software development process, while Falck et al. (2012) examined the significance of complexity and the relationships between ergonomics, assembly complexity and quality by investigating manual assembly tasks. The aim of these researchers was to support product preparation by increasing productivity and decreasing costs.

As far as perceived complexity is concerned, Mattsson et al. (2014) have recently suggested a method to measure operators' perception of subjective complexity. Their method makes use of a questionnaire to find problem areas at the workstation level.

Entropic Measurement

The variability of mixed-model assembly lines generates a high level of uncertainty, resulting from the large number of available choices. To deal with such uncertainty, researchers have often used the information theory entropy approach to quantify complexity. First defined by Shannon (1948), this approach measures the unpredictability of an event using particular information. The amount of information and degree of choice available in the system are directly related to its complexity level. For example, Frizelle and Woodcock (1995) used an entropy measure for static and dynamic complexity and focused on different states of production obtained by a combination of processes, workstations and parts. Their main goal was to measure the rate of variety to determine the probability of a state's occurrence according to different time measurements.

Fujimoto and Ahmed (2001) put forward a complexity index centred on the ease of assembling a product. The index took the form of entropy to evaluate the 'assemblability' of a product, defined as the uncertainty of gripping, positioning and inserting parts during the assembly process. More recently, researchers have begun to analyse information flows to measure complexity. Sivadasan et al. (2006) presented an entropic measurement which focused on monitoring and mapping information flows. This provides a measure of complexity based on the amount of information required to describe a state, according to flow variations, products, reasons and variation states. Briefly, they proposed an operational complexity of supplier-customer systems that can be interpreted as the uncertainty associated with managing the system, given the level of control and detail of monitoring.

Abad and Jin (2011) defined a set of complexity metrics which used entropy and aimed to quantify a system's capability of handling the complexity induced by model variety. They concentrated on measuring the diversity between demand and complexity in delivered products, considering production processes. Recently, Zhu et al. (2008) and Hu et al. (2008) have proposed a measurement of manufacturing complexity caused by product variety and the modelling of its propagation through the assembly system. This measurement uses entropy to quantify human performance in making choices; that is, the uncertainty faced by the operator when making choices at an assembly station, such as selecting parts, tools, fixtures and assembly procedures.

Table 2.1: *Complexity Measurements - Literature Review*

	To measure complexity of:	Aim	EA	EM
MacDuffie et al. (1996)	Product variety on productivity and quality.	To test the impact of complexity in manufacturing performance (total labour productivity + product quality).	x	-
Deshmukh et al. (1998)	Processing requirements of parts to be produced and machine capabilities.	Present relationships between complexity measure and system performance (waiting time)	x	-
Urbanic and ElMaraghy (2006)	Process.	To evaluate risks and alternatives in a design stage.	x	-
Meyer and Curley (1995)	Knowledge and technology on information systems.	To manage software development	x	-
Falck et al. (2012))	Manual assembly work, ergonomics and assembly quality.	To support increase of production and decrease of costs.	x	-
Mattsson et al. (2014)	Workstation experienced by operators.	To find problem areas at a workstation level.	x	-
Frizelle and Woodcock (1995)	Variety's rate	Measure the probability of a strate to occur based to different times.	-	x
Fujimoto and Ahmed (2001)	Product Assemblability	To define the uncertainty of gripping, positioning and inserting parts during the assembly process.	-	x
Sivadasan et al. (2006)	Supplier-customer systems.	To monitor and manage information and material flows.	-	x
Abad and Jin (2011)	To measure the complexity induced by the input demand mix ratio.	To assess the probability of production output meets the variety of the input.	-	x
Zhu et al. (2008)	Operator choice and assembly line.	To find causes, plan assembly sequences and design mixed-model assembly lines.	-	x

* *EA* refers to *Empirical Analysis* and *EM* to *Entropic Measurement*

Table 2.1 summarizes all the previously mentioned approaches, with complexity measurements focusing on empirical analysis or entropic measurement.

2.4 Conclusion

In this chapter, we explored the scientific literature on the MMALBP and manufacturing complexity. While a large number of studies have dealt with both topics in detail, few researchers have analysed the relationship between complexity and assembly line balancing. Most researchers simply concentrate on different objectives and applications while measuring complexity, and complexity knowledge and measurements are rarely applied to the improvement of production efficiency. To the best of our knowledge, no approaches exist that tackle the complexity of MMALs while quantifying complexity and applying this to the improvement of line balance.

The objective of this dissertation is to put forward new methods to characterize and quantify complexity based on both empirical analysis and entropic approaches. First, we will examine real scenarios to extract knowledge via practical information and interactions with manufacturers, allowing us to develop classification models for workstation complexity. Then, we will explore the concept of entropy to quantify the uncertainty generated by the high variety of MMALs. We propose a complexity measurement, based on entropy and task time variations, to be integrated into a optimization MMALBP solution for Type 3 problems. The outcome of this study is a novel approach that monitors and levels complexity while balancing workload.

3

Analysing Manufacturing Complexity in Mixed-Model Assembly Lines

In an effort to maintain or increase their market share while preventing costs from escalating, manufacturing organizations are increasingly using their current manufacturing systems to produce custom outputs. This greater variety of products significantly increases the complexity of manufacturing systems. This is especially true in the automotive industry, where customization is rapidly growing. To counter the ensuing loss of productivity, manufacturers require a more fundamental approach to dealing with this complexity in their processes. To investigate the impact of complexity on production performance, we must first delineate the core concept and identify highly-complex workstations as unambiguously as possible.

In this chapter, we will define manufacturing complexity and introduce a classification methodology for characterizing mixed-model assembly workstations. In Section 3.2, we will present a definition of complexity and describe the research methodology that was used. Section 3.2 suggests a number of statistical models, based on data from several leading automotive companies, which distinguish between workstations that are low or high in complexity. Finally, we will describe the results that were obtained from these models. This chapter is an adaptation and extension of Zeltzer et al. (2013).

3.1 Production complexity definition and research methodology

The research on complexity analysis presented in this dissertation was carried out in collaboration with the automotive industry. The focus is on driven assembly lines, where manual assembly work is carried out on various models in a mixed-model fashion. Different types of assembly lines were investigated, including two for car models, two for engine models, one for a truck model and several subassembly lines with suppliers. First, Section 3.1.1 offers a working definition of complexity. Then, in section 3.1.2, we describe our collaborations with automotive companies. Finally, Section 3.1.3, presents a causal model of the drivers and impacts of complexity.

3.1.1 Complexity definition

A good definition of complexity should be generic enough to be applicable to various manufacturing systems, while at the same time being specific enough to classify a system as complex or not. Although the literature review provided useful insights into manufacturing complexity (Gullander et al., 2011; Mattsson et al., 2014), most approaches were relatively specific. In our view, there is still a need for a clear, simple and generic definition of complexity. After extensive discussions with manufacturers and a systematic analysis of real systems, the following definition was found (Zeltzer et al., 2013):

The complexity of a workstation is the sum of all technical and ergonomic factors that make the set of tasks to be performed at it cognitively challenging, error-prone and stressful because a high degree of alertness is required of the operator.

This definition recognizes the fact that the inherent complexity of a task is largely determined by the operator who executes it, hence the term 'subjective complexity'. This means, according to our findings, that different operators, production engineers, quality controllers and line managers may perceive a set of tasks differently under different circumstances. This makes the issue of quantifying complexity unambiguously – known as 'objective complexity' – a real challenge. One immediate consequence is that measuring the magnitude of subjective complexity involves behavioural and psychological aspects which are difficult to quantify (Rasmussen, 1983). Secondly, since complexity is a multi-faceted concept, it is almost impossible to measure it directly as no meaningful scale exists. Therefore, we have focused on measuring complexity in an objective, repeatable manner using some of its direct drivers, which are easily and unambiguously observable and quantifiable.

3.1.2 Model building workshops

To gain more insight into manufacturing complexity and identify its drivers, we organized a series of workshops in collaboration with a group of vehicle manufacturers. We identified the components of complexity, classified them as drivers or impacts – consequences – and used them to build a causal model. To gather as much useful information as possible, we selected participants who all encounter complexity in their daily activities, including shop floor employees, production engineers, quality controllers and line managers.

All workshops were organized in a similar way. The objectives of the project were first explained to all participants, after which they were asked to identify two workstations that were low in complexity and two that were high in complexity. They were then asked to use these workstations as a mental reference when describing elements of complexity, drivers or causes of complexity and the impact or consequences of complexity. They were given three sets of questions, each focusing on a relevant aspect of complexity:

Elements of Complexity
How do you experience complexity?
What characteristics make a workstation complex?

Causes/Drivers of Complexity
What elements have changed in recent years, and has complexity increased or decreased?
What elements are under your domain/decision?
What elements can be measured and how/where can this be achieved?

Impact/Consequences of Complexity
How is your work influenced by complexity?
How does it influence your team?
How is your area affected?
Indicate the largest problems.

All participants wrote down their individual answers on separate sticky notes, which were then clustered into three separate sets. Figure 3.1 shows part of a workshop.

Figure 3.1: *Model building workshops*

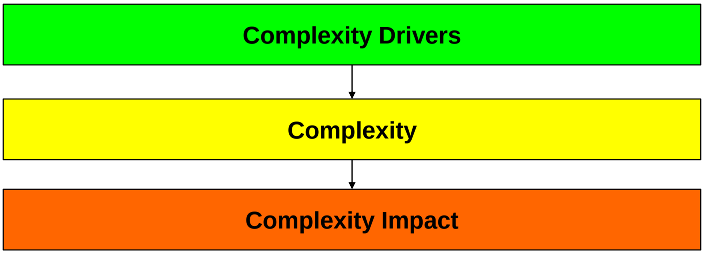
While the first set of questions focused on characterizing complexity, the second concentrated on revealing the consequences of complexity – areas that are affected by complexity and the influence complexity has on manufacturing work and teams. The third set detected the direct drivers of complexity, i.e. the variables that are directly linked to complexity elements as causal factors. In each round, we put relevant notes made by the participants up on a wall and clustered them according to similarity. Finally, we held a brainstorming session during which this list of ideas was discussed and finalized. The results were extensively discussed with our industrial partners; they were found to be both insightful and useful.

We finalized our interactions with different manufactures with a visit to their assembly lines. During these visits, they gave us a general overview of the line and provided us with more information about the two workstations identified during the workshops. As a result, we were able to observe and analyse the complexities explained by our workshop participants.

3.1.3 Causal model of complexity

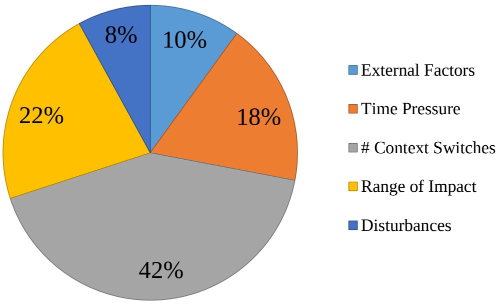
As a next step, the causal links between the elements identified during the workshops were combined into a graphical network structure to obtain a generic complexity model for assembly workstations. This model consisted of three clusters of variables related to complexity drivers, complexity characterization and complexity impacts, as shown in figure 3.2. As the fully detailed model is too large and complex to be summarized in one clear figure, we have divided the network structure across different figures.

Figure 3.2: *Causal model structure overview*



To identify the network elements, we conducted a statistical analysis of the answers given during the workshops in combination with the manufacturers' feedback. This analysis revealed several complexity drivers . Figure 3.3 represents the percentage of each driver based on the workshop participants' feedback. The direct drivers are external factors, time pressure, number of context switches, range of impact, and disturbances.

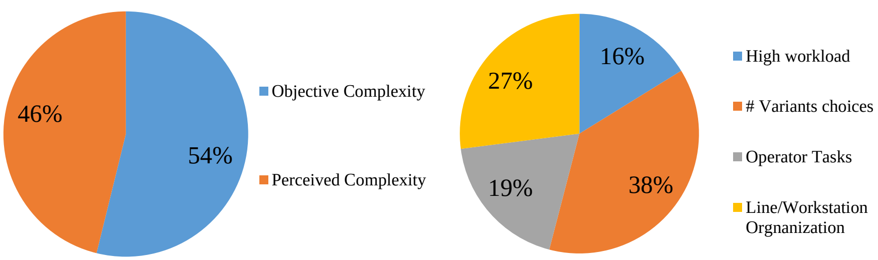
Figure 3.3: *Complexity direct drivers'distribution*



The number of context switches is a cluster of: number of tools, variants, flows, assembly variants, variants of the same model and car models. Evolving from this analysis, eleven direct drivers of complexity were selected for complexity measurement. These drivers are explained in more detail in Table 3.1. The first cluster of variables is represented in 3.6 and 3.7. Complexity drivers are grouped into different layers, with general drivers being identified in top layers and clustered in specific direct drivers. Direct drivers are then linked to complexity.

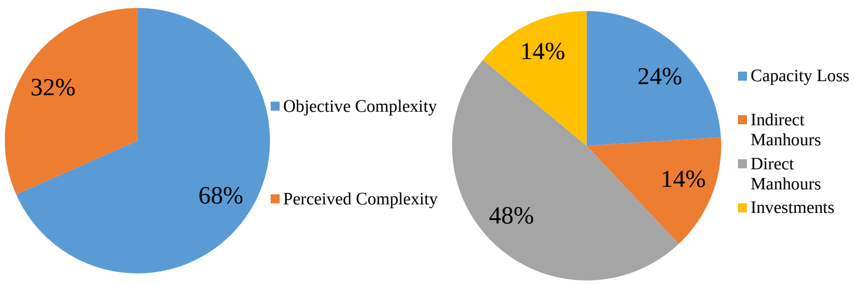
Complexity characterization cluster elements are mainly divided into two main groups: objective complexity and perceived complexity. The main difference, as already cited in Chapter 2, is that objective complexity can be analysed under a quantitative approach, while perceived complexity consists of a subjective approach. Approximately 54% of complexity is objective and 46% is perceived. We identified four important elements: high workload, number of variant choices, operator tasks, and line/workstation organization, as shown in Figure 3.4. Complexity characterization was extremely important for proposing a definition of complexity (Section 3.1.1).

Figure 3.4: *Complexity characterization*



Complexity impacts are derived from objective and perceived complexity. However, the drivers show that approximately 50% (Figure 3.4) of the elements can be categorized in each complexity category; 32% of the consequences are derived from perceived complexity and 68% from objective complexity (Figure 3.5). Figures 3.8 and 3.9 demonstrate the structure of the complexity results. We grouped these elements into different layers: general impacts are identified in top layers and clustered in specific impacts' variables. We identified four direct main elements: capacity loss, indirect man hours, direct man hours and investments. In Figure 3.5, the percentage of each main consequence is shown based on our workshop participants' feedback. It is important to refer that capacity loss variable clusters: numbers of planned balance loss, work instructions, missing parts and operations, quality decrease and errors.

Figure 3.5: *Complexity impacts*



From the complete analysis, we used the first cluster of the causal model, complexity drivers, to build classification models for workstations (Section 3.2). We took the middle cluster, complexity characterization, to create a definition of complexity (Section 3.1.1). The last cluster, complexity impact, provided an overview of possible future actions to cope with complexity.

3.1.4 Empirical Dataset

We set out to test whether the drivers we identified during the workshops could be used to characterize highly complex workstations, and provide an objective means of quantifying complexity. We wanted to determine what the smallest possible subset of drivers was to generate meaningful results (to minimize the effort required for gathering data). It was immediately clear to us that the range of values is very large, which makes modelling quite difficult. To gain more control over scaling, we set up a Likert scale for each variable; we divided the data range across four levels on this scale. For some variables, we added a zero level. These results are shown in Table 3.2.

Table 3.1: *Variables driving complexity (reference period = 1 year)*

Variable name	Values or units	Description
Picking technology	(F)ixed location	(F): operator takes part always on the same location from bulk storage
	Pick to (S)ignal (C)omparing (M)anual	(S): operator picks part from location indicated by a signal (light, display) (C): operator must compare simple information (symbols, colours) (M): operator must read extensive information from manifest
Bulk/sequence kit	(S)equenced kit	(S): every part is in its package in correct assembly sequence
	(K)it (B)ulk	(K): parts are delivered in kits with exact set for one assembly operation (B): parts are by type in their own package
# Packaging types	Integer number	The total number of different packaging types, a type having a specific layout. So, two identical boxes with different inserts are two different types
# Tools per workstation	Integer number	The number of tools that the operator(s) needs to handle to perform all possible assembly variants in this station, excluding automatic tools (servants)
# Machines per workstation	Integer number	Machines that perform automated tasks without operator assistance, with automatic or manual start
# Work methods	Integer number	Each unique set of work methods the operator must master in this station. A method contains several small steps
Distance to parts	Metres	The furthest distance between the normal operator position (or the centre of the station) and the parts at the border of line
# Variants same model	Integer number	The highest number of variants belonging to one model, among all models of which parts are assembled in this station
# Variants in this workstation	Integer number	Total number of variant parts, combined over all models that are assembled in this workstation. So, five types of left hand and right hand mirrors x two models = 20 variants
# Different parts in workstation	Integer number	Total number of unique part references that are assembled in this WS, including all variants and models that typically occur in one year

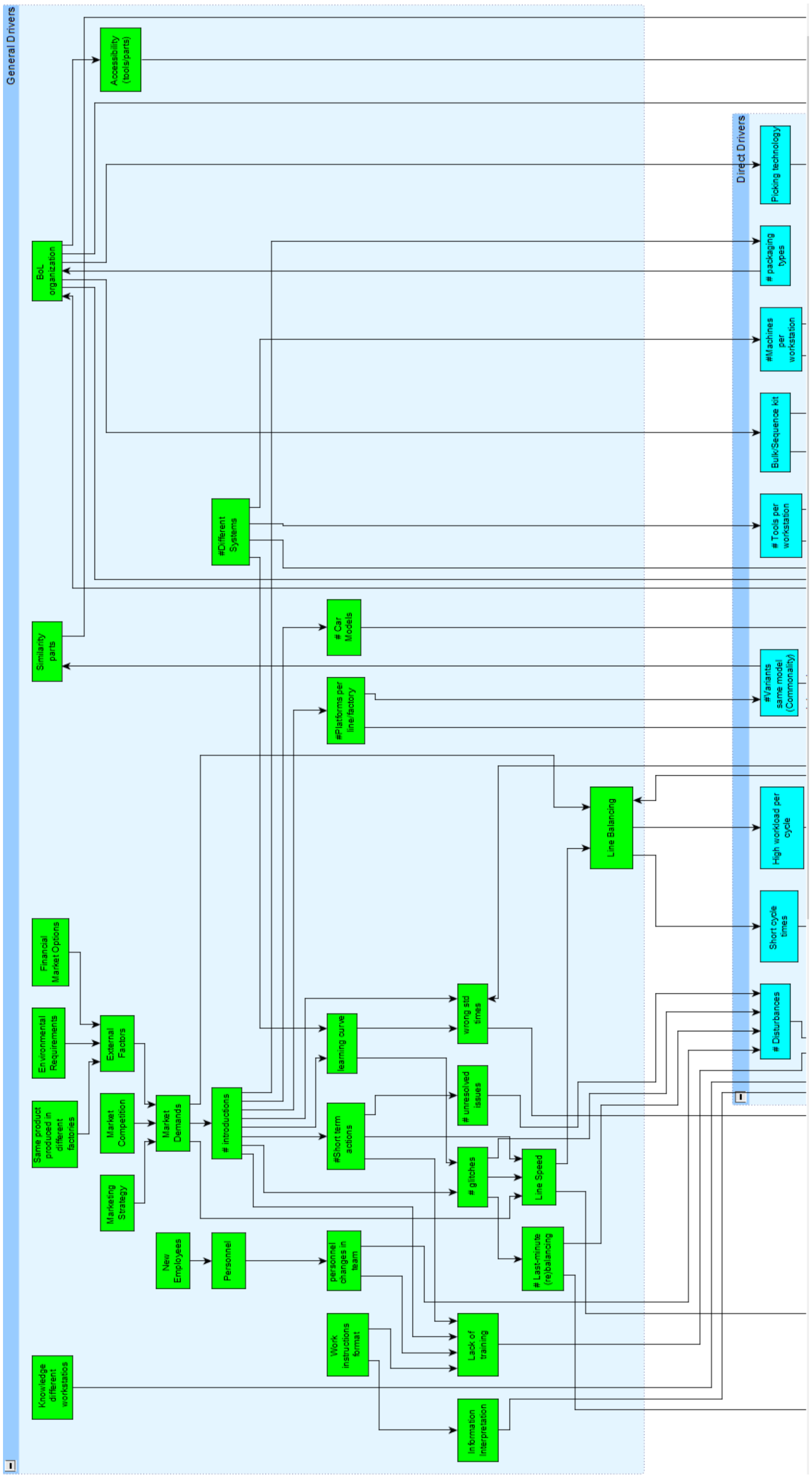


Figure 3.6: *Causal model – Complexity general drivers*

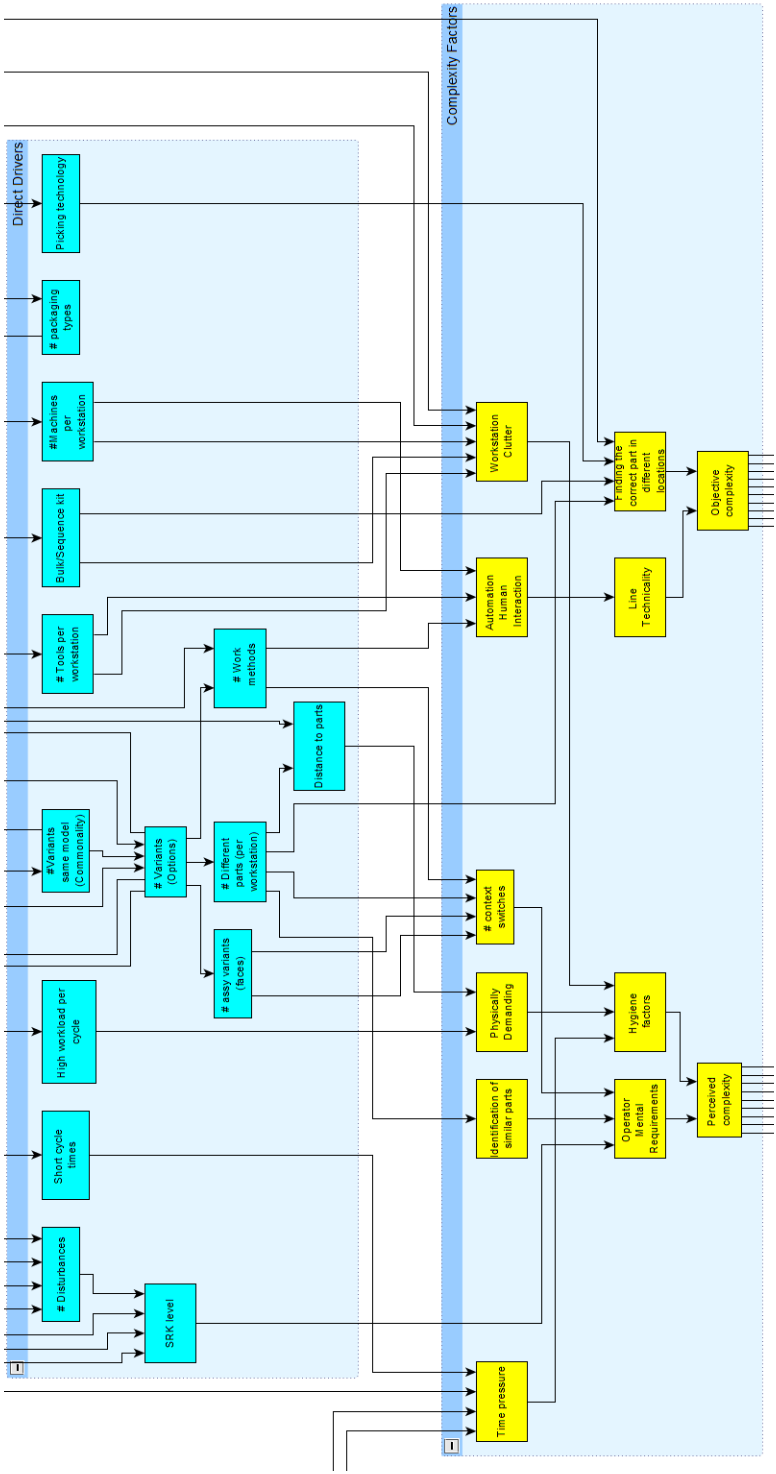


Figure 3.7: *Causal model – Complexity direct drivers and characterization*

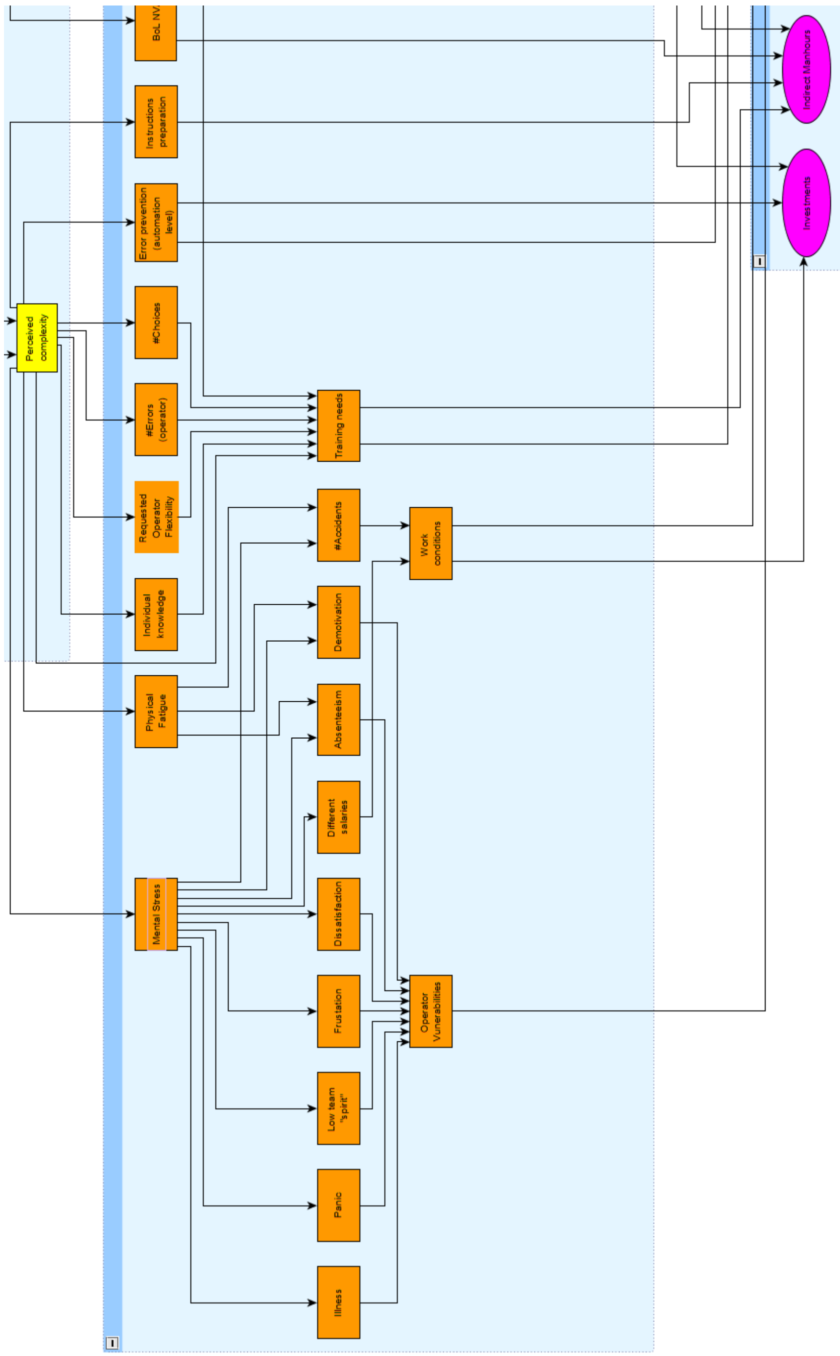


Figure 3.8: *Causal model – Subjective Complexity Impacts*

Figure 3.9: *Causal model – Objective Complexity Impacts*

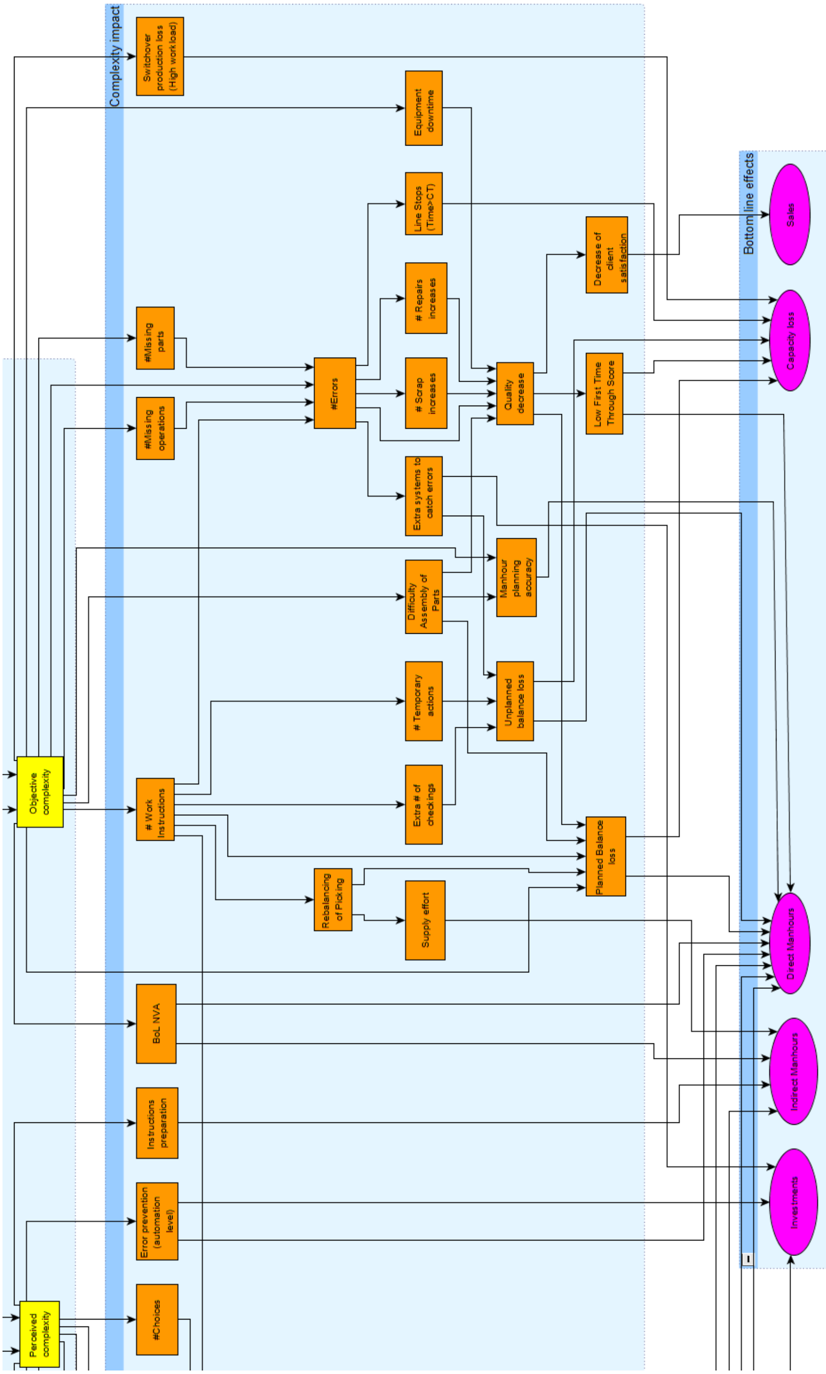


Table 3.2: Likert Scale for the variables data-set.

Complexity-driving variables	Likert scale coding rules				
Picking technology		F	S	C	M
		1	2	3	4
Bulk/sequence kit		F	S	K	B
		1	2	3	
# Packaging types	0	1	2-4	5-8	>8
	0	1	2	3	4
# Tools per workstation	0-1	2-4	5-8	>8	
		1	2	3	4
# Machines per workstation	0	1	2	>2	
	0	1	2	3	
# Work methods	0-2	3-5	, 6-8	>8	
		1	2	3	4
Distance to parts	0-1	1,1-2	2	1-4	>4
		1	2	3	4
# Variants same model	1	2-3	4-5	>5	
		1	2	3	4
# Variants in this workstation	1	2-4	5-10	>10	
		1	2	3	4
# Different parts in workstation	0	1-4	5-10	11-20	>20
		1	2	3	4
# Assembly directions	1	2-3	4-5	>5	
		1	2	3	4

Since no accurate information was available regarding the real inherent complexity of these workstations, we asked operators and supervisors to point out the most and least complex workstations in their areas together. These subjective labels served as benchmarks throughout the study. We provided each of our industrial partners with quantitative information about the driving factors linked to these designated workstations.

Table 3.3: Values of categorical variables as measured over 76 workstations

Values	Fixed	Signal	Comparing	Manual
Picking technology	10	17	4	45
Values	Bulk	Kit	Sequenced kit	
Parts delivery	65	1	10	

In this way, we obtained datasets on 76 workstations in five different manufacturing locations (i.e. four in Belgium and one in Sweden), 41 of which we deemed to be 'LOW' in complexity and 35 'HIGH' in complexity. The variables for this data are listed and explained in

Table 3.1, the categorical variables are shown in Table 3.3, and data on numerical variables can be found in Table 3.4.

Table 3.4: *Range of numerical values as measured over 76 workstations.*

Variable	Min	Max	Average	SD
# Packaging types	0	14	3.76	2.44
# Tools per workstation	0	12	3.99	2.94
# Machines per workstation	0	1	0.25	0.44
# Work methods	1	192	17.39	31.84
Distance to parts	0.8	25	4.78	4.11
# Variants same model	1	192	10.29	23.28
# Variants in this workstation	1	217	22.47	40.44
# Different parts in workstation	0	264	24.07	41.36
# Assembly directions	1	38	4.43	4.75

3.2 Classification models for workstation complexity

Five different approaches were explored in order to obtain useful statistical models for characterizing the complexity of assembly line workstations. The objective of these models is to identify and classify workstations as high or low in complexity. In this section, we will discuss the computational tests used and the results obtained. The five approaches we explored were the following:

1. BASE: a simple linear model which calculated a weighted combination of the Likert scores of all 11 variables to yield a single complexity number. All weights were equal to 1 (Section 3.2.1).

2. LOGIT_ALL: a statistical model based on data from all 76 workstations (Section 3.2.2).

3. CALC_ALL (Calculated - all cases): a simple linear model based on the variables of the LOGIT_ALL model. The weights used were derived from the LOGIT_ALL statistical model (see Section 3.2.3).

4. LOGIT_SAMPLE: a statistical model based on a stratified subset of 53 cases which scored extremely HIGH or LOW in the BASE model (Section 3.2.2).

5. CALC_SAMPLE (Calculated sampled cases): a model similar to the previous one, but based on the LOGIT_SAMPLE model results(Section 3.2.3).

3.2.1 The initial model BASE

We first calculated an overall complexity level score as a weighted sum of all variables. This calculation was performed in two parts: first, we calculated a basic score as an average weighted sum of the Likert scores of the variables for each workstation, and then this score was converted into a normalized number between 0 and 10, based on the maximum and minimum values a workstation could score for each item.

$$Basicscore = \frac{\sum_{Items} Weight(item) \times Score(Item)}{\sum_{Items} Weight(item)} \tag{3.1}$$

$$Adjusted\,score = \frac{(Basic\,score - min\,score)}{(Max\,score - min\,score)} \times 10 \tag{3.2}$$

where:
$Weight(item)$ = weight attributed to the variable;
$Score(item)$ = likert scale score of the workstation on this variable;
$Minscore$ = basic score with all items at their minimum possible Likert value (Table 3.2);
$Maxscore$ = basic score with all items at their maximum possible Likert value.

For workstation 1 (Table 3.5), which had a 'low' subjective score, the Complexity Basic score amounted to 1.73 which was normalized to 3.44 (/10). This example calculation fits both interpretations if we assume LOW scores are below 5/10 and HIGH scores are 5/10 or above. At this stage, we did not have any information regarding the relative impact of each of the variables on complexity. We defined all weights as equal to one; the Basic score then amounted to a simple average of the item scores.

Table 3.5: Example calculation for LOW complexity workstation.

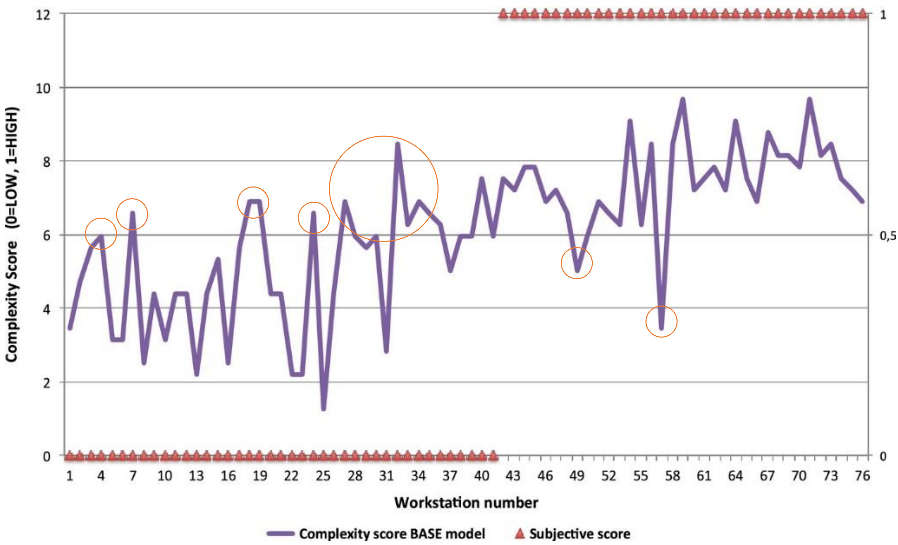
Item	Raw data	Score (item)	Weights
Picking technology	M	4	1
Bulk/sequence kit	B	3	1
# Packaging types	1	1	1
# Tools per workstation	1	1	1
# Machines per workstation	0	0	1
# Work methods	2	1	1
Distance to parts	2	2	1
# Variants same model	2	2	1
# Variants in this workstation	2	2	1
# Different parts in workstation	6	2	1
# Assembly directions	1	1	1
Total of column		19	11

We subsequently plotted out the resulting Adjusted Scores for all 76 workstations (Figure 3.10). We also indicated the subjective complexity labels and sequenced the workstations from LOW to HIGH (and by company). From our results in Figure 3.10, we observed an extensive fluctuation in Adjusted Score values. Based on the standard t-test, we found that the average score for LOW stations (i.e. 4.8 with an SD of 1.7), differs significantly from the average of the HIGH stations (i.e. 7.2 with an SD of 1.2). We concluded that this calculated score can be used to distinguish workstations that are HIGH in complexity from those that are LOW and that the variables it is based on relate to the levels of subjective complexity.

The wide fluctuations in the scores indicates that not all variables have the same explanatory power, or may even contradict each other; regardless of the explanatory power of the Adjusted Score, the BASE model value of some workstations contradicts their subjective classifications. The two workstations that score below 5 in the HIGH section (nos. 50 and 56) are a good illustration of this. The LOW section contains 10 outliers of this kind. If these outliers persist after further tuning of the model, we should check the basis of the subjective score with the workstation operators.

The next logical course of action was to either adjust the weights of the variables or reduce the number of variables (which is an extreme version of the first option, with weights set to zero). To achieve this, we turned to statistics.

Figure 3.10: *BASE model score compared to subjective complexity for all workstations.*



3.2.2 Fitting statistical Logit models LOGIT_ALL and LOGIT_SAMPLE

We needed a statistical model to find the correlation between (a subset of) our variables and their true levels of complexity. At this stage of our research, we had no true measure of this complexity, only the subjective scores provided by the operators. We therefore tried to find a statistically valid model independent of the calculated score which would allow us to compare the two.

Since we were working with only two dependent values (HIGH and LOW), it was not appropriate to perform a linear regression on the 11 independent variables. However, by means of a model called 'Logistic Regression' or 'Logit', we could calculate (from a linear combination of variable values $(A + BX)$ the probability that a resulting workstation score was either HIGH or LOW in complexity. Note that the values 0 and 1 can be assigned either way: HIGH $=$ 1 and LOW$=$ 0, or vice versa. The actual calculations were performed with HIGH $= 0$ and LOW$=$ 1. For clarity's sake, we include the basic formulas here, although they can be found in many statistics textbooks.

The distribution in Figure 3.11 is calculated as a ratio of exponential functions (representing the *odds* of LOW over HIGH) as follows:

$$P[Y = LOW] = \frac{e^{a+\sum_{i=1}^{n} b_i X_i}}{1 + e^{a+\sum_{i=1}^{n} b_i X_i}} \tag{3.3}$$

where $a + \sum_{i=1}^{n} b_i X_i$ is a linear combination of (a subset of) the complexity variables X_i with weights b_i and a scaling constant a.

Figure 3.11: *Cumulative distribution of logistic regression model. Comparing the LP and Logit Models*

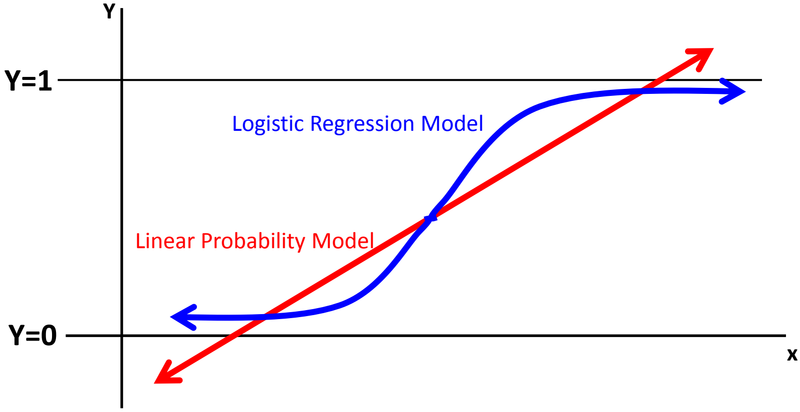


Figure 3.11 shows that this model has a sharply rising transition zone; it therefore appeared suitable for the classification task we envisaged. This approach was inspired by Braaksma et al. (2012), who used it to classify machines based on whether or not they required maintenance. This exponential model was converted into a linear regression model to fit the variables. For input variables, we could use the raw values of the variables as measured, the Likert Scale scores, or a combination of the two. The statistical software SPSS 19 (®from IBM) was used to find the best set of weights *bi* and constant *a* for the subjective scores.

Despite testing various subsets of these variables, we were unable to find a valid Logit model that included all 11 variables and all 76 cases for which we had data. Using all 76 cases, we identified four variables that provided a correct classification in 84.2% of cases (and 88.6% of workstations HIGH in complexity). We called this the LOGIT_ALL model. Table 3.6 displays these numerical results as SPSS outputs.

Table 3.6: *Statistical fit results for LOGIT_ALL model.*

			Predicted		
			HIGHLOW		
	Observed		High	Low	Percentage correct
Classification Table [a]					
Step 1[a]	HIGHLOW	High	31	4	88.6
		Low	8	33	88.5
	Overall percentage				84.2
		B	SE	Wald	df
Variables in the equation					
Step 1[a]	#Packaging types	-1.127	0.592	3.622	1
	Assembly directions	-0.874	0.348	1.591	1
	# Different parts in work stations	-0.243	0.193	6.300	1
	# Work methods	-0.058	0.028	4.491	1
	Constant	6.676	1.837	13.200	1

[a]The cut value is 0.500.

Since we assigned the value 1 to LOW , this model starts with a probability of 1 for LOW through the constant (when all variables equal zero). The positive values of the variables then reduce this probability because the weights b_i are negative. Thus, the higher the number of packaging types, the lower the probability that this workstation is LOW in complexity (and, of course, the higher the probability that it is HIGH in complexity). The higher the absolute value of coefficient b_i, the larger the impact of this variable i.

The LOGIT_ALL model was formulated as follows:

$$P_{LOGIT\_ALL}(LOW) = \frac{e^{6676-1127PTL-0874PWL-0243ADR-0058WMR}}{1+e^{6676-1127PTL-0874PWL-0243ADR-0058WMR}} \tag{3.4}$$

Where PTL is the number of packaging types on Likert scale, PWL are the different parts in the workstation on a Likert scale, ADR is the number of assembly directions as measured directly (raw score), and WMR is the number of work methods as measured directly (raw score). In this model, we used the recoded Likert scale values for two of its four variables and the direct values as measured for the other two.

We also attempted to construct a model that provided a better fit based on a filtered subset of 53 extreme cases (following equation), by removing all workstations with outlier values for Adjusted Score as calculated with the BASE model. This result, 'LOGIT_SAMPLE', is shown in Table 3.7. This model succeeded in correctly classifying cases 98.1% of the time (only one case was classified incorrectly). When we applied this model to all 76 cases and used a cut-off level of 0.8 (i.e. the 0.5 used in the model identification), 62 cases (81.6%) were classified correctly. Thus, while the LOGIT_SAMPLE model is stronger as a classification model, the cases do not fit quite as well (although the difference is marginal).

$$P_{LOGIT\_SAMPLE}(LOW) = \frac{e^{18.164-3173PWL-2326PTL-2182ADL-0.344TEL}}{1+e^{18.164-3173PWL-2326PTL-2182ADL-0.344TEL}} \tag{3.5}$$

where PWL are the different parts of a workstation on the Likert scale, PTL is the number of packaging types on the Likert scale, ADL is the number of assembly directions on the Likert scale, and TWL is the number of tools used in a workstation on the Likert scale.

In this model, we only used Likert scale values. In addition, the factor TW had only a marginal effect, so removing it yielded the same 98% correct classification of the test set. However, it did perform better on the full set of workstations. We can compare the behaviours of these models using the Receiver Operating Characteristics (ROC) theory (Fawcett, 2006), as shown in Table 3.8.

Table 3.7: *Statistical fit results for LOGIT_SAMPLE model.*

			Predicted		
			HIGHLOW		
	Observed		High	Low	Percentage correct
Classification Table [a]					
Step 1[a]	HIGHLOW	High	33	0	100
		Low	1	19	95
	Overall percentage				98.1
		B	SE	Wald	df
Variables in the equation					
Step 1[a]	#Assembly directions	-2.182	1.657	1.734	1
	#Packaging types	-2.326	2.195	1.123	1
	# Different parts in work stations	-3.173	1.113	8.125	1
	# Work methods	-0.344	0.855	0.162	1
	Constant	18.164	7.328	6.145	1

[a]The cut value is 0.500.

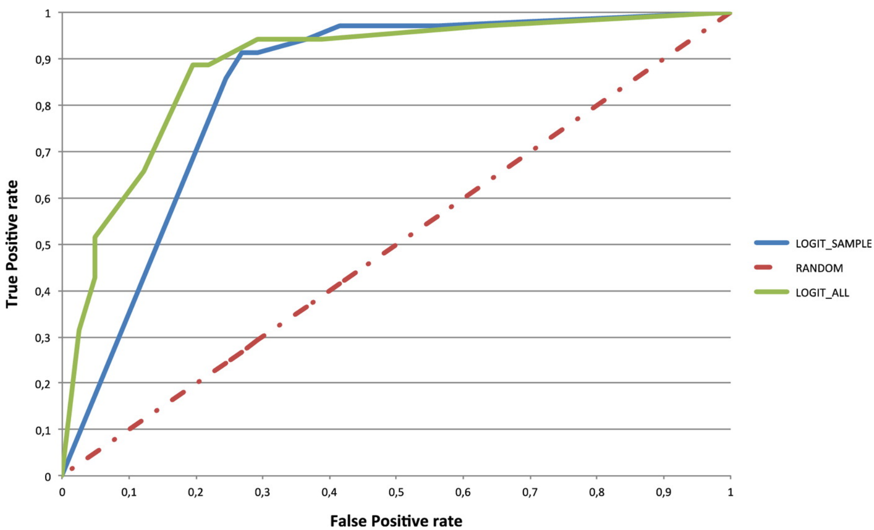
Table 3.8: *Comparison of LOGIT models using ROC metrics.*

ROC Metric	Definition	LOGIT_ALL on all 76 cases(%)	LOGIT SAMPLE on 53 cases(%)	LOGIT SAMPLE on 76 cases(%)
Precision	$TP \setminus (TP + FP)$	88.6	100	74.4
Accuracy	$(TP + TN) \setminus (TP + FP)$	84.2	98.1	81.6
FP rate	$FP \setminus N$	10.8	0	26.8
TP rate	$TP \setminus P$	79.5	97.1	91.4
Specificity	$1 - (FP \setminus N)$	89.2	100	73.2

TP (FP), number of true (false)positives; TN (FN), number of true (false)negatives; N, number of negatives; P, number of positives.

By differentiating the cut-off level between 0 and 1, we can generate ROC curves for both models. Figure 3.12 clearly shows that the two models are of comparable quality and equally strong. The larger the surface between the ROC curve and the 45°line (indicating the expected performance of a random filter), the more discriminating this model is in detecting the condition of the workstation (i.e. its complexity level) in this case.

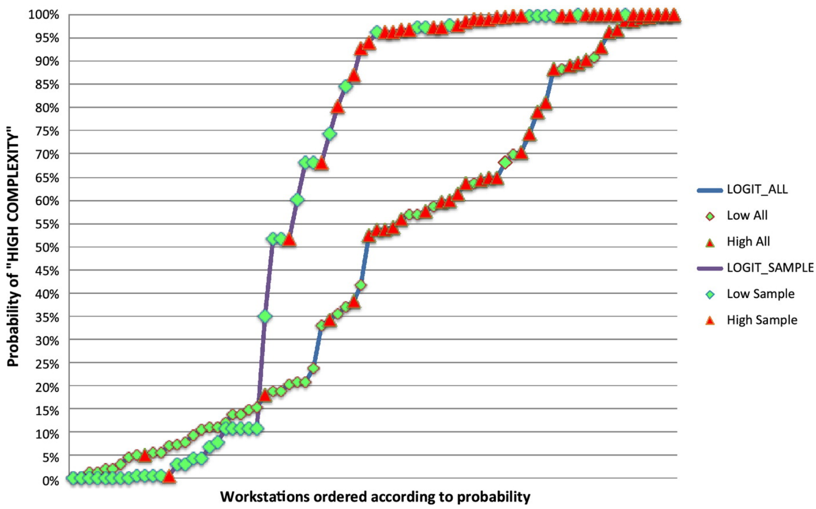
Figure 3.12: *ROC curve of LOGIT_∗ models.*



Next, we wanted to establish which of the two models performed better in practice, and what this could tell us about our linearly-calculated complexity scores. In Figure 3.13, we show the probability of a workstation being HIGH inn complexity (derived as a complement from the above models) for both LOGIT models. We coded each data point a Diamond (green) if its subjective label was LOW and as a Triangle (red) if it was labelled HIGH.

We can make a number of interesting observations based on this graph. The 'LOGIT_SAMPLE' model is steeper (cf. the left curve), and thus more discriminating than the 'LOGIT_ALL' model. With only 12 workstations, it has the smallest intermediate zone. The 40 HIGH workstations (i.e. those with a probability above 85%) receive correct HIGH subjective scores in 30 cases, suggesting a classification that is correct in 75% of the cases. From another point of view, it can also be concluded that there are 10 cases with a questionable subjec-

Figure 3.13: *Distribution of workstation probabilities according to LOGIT_ALL and LOGIT_SAMPLE.*



tive labelling. The LOW workstations (i.e. those with a probability below 15%) are correctly labelled in 23 out of 24 cases, indicating a 96% success rate: only one is labelled incorrectly, which strongly suggests that the subjective label of that case is inaccurate. We chose these cut-off probabilities based on the tails of the cumulative distribution, which are clearly distinguishable in Figure 3.13.

The 'LOGIT_ALL' model rises much more gradually, with a large intermediate zone. While we could assign a HIGH label as soon as the probability exceeds 50%, only 15 extremely HIGH workstations emerge (i.e. with a probability higher than 85%). There are two incorrect subjective labels among these, which means that we have a success rate of 84%. The LOW zone contains 32 workstations, two of which were incorrectly labelled. It can be concluded that this model classifies extreme workstations correctly 94% of the time.

In both models, three of the determining variables are identical, albeit in different numerical formats. This is a strong indication that the numbers of packaging types, assembly directions and different parts in the workstation are good predictors (and sources) of workstation complexity. These three variables belong to the domains of logistics, assembly methods and design for manufacturing, which we also intuitively found to be acceptable.

These two models have both strengths and weaknesses. Overall, the 'all cases' model is the best predictor. It is useful if we are interested in a gradual scale and could be used immediately. In other words, the calculated BASE score is no longer needed since we can obtain a useful measure from the Logit model (the probability of scoring as 'highly complex'). Of course, the formula is much less intuitive than the weighted sum of the Adjusted score, but it is statistically more significant. It could be said that the 'LOGIT_ALL' model curve indicates the likelihood of a workstation being rated 'highly-complex' when multiple people are asked or multiple variables are used.

The 'LOGIT_SAMPLE' model provides a relatively sharp distinction that is ultimately more correct for extreme cases of HIGH and LOW complexity. Thus, this model is more appropriate if we are interested in classifying workstations into only two groups, HIGH and LOW (for whatever reason). It is also more suitable for filtering out extreme workstations from a large set, for example to use them for further research. Since most automotive plants have hundreds of workstations, an automated filtering algorithm to apply to a corporate engineering database is very practical indeed. Its probability value for a workstation is less useful as a direct complexity indicator, however, because it should be combined with the calculated score.

3.2.3 Improved calculated score models CALC_ALL and CALC_SAMPLE

The insights gathered from the Logit models allowed us to redefine the linear score calculation so it included only the significant variables identified above. Since we had two Logit models, we were also able to identify two different calculations. The weights in each CALC_$*$ model were manually adjusted to reflect the relative importance of the variables, as determined by the B weights in the exponential formula of the Logit model (see Table 3.9). Table 3.9 shows the final values of the CALC_$*$ model weights, including the weights from the Logit models from which they were derived.

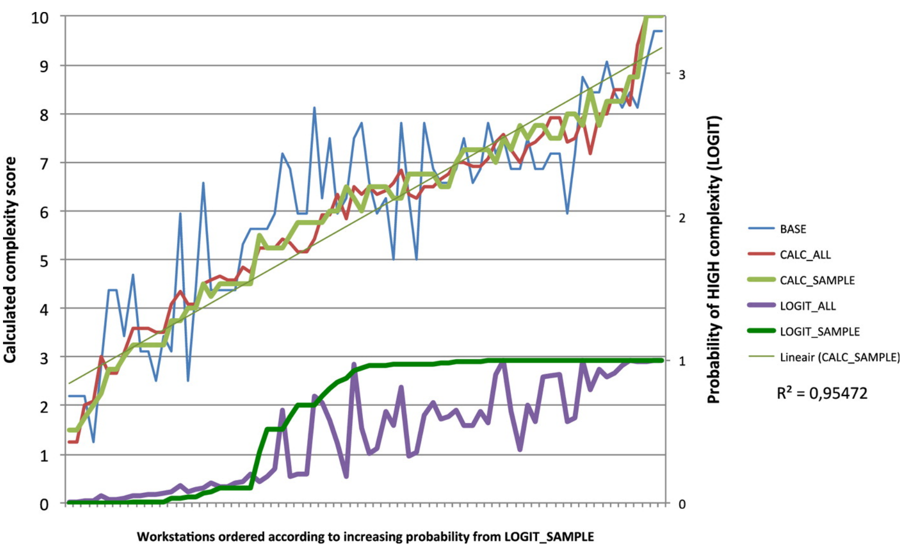
Finally, we compared the complexity scores produced by all of the models. Figure 3.14 contains the values for all workstations, ordered by increasing LOGIT_SAMPLE score. This model was the most discriminating with regard to subjective labels. On the basis of this figure, we have made a number of observations. The values of the

Table 3.9: *Final values of weights used in the linear models.*

Variable	LOGIT ALL	CALC ALL	LOGIT SAMPLE	CALC SAMPLE
#Packaging types	1.27	11	2.182	2
#Assembly directions	0.243	7	2.326	3
#Different parts in workstation	0.874	13	3.173	5
#Work methods	0.058	1		
#Tools per workstation			0.344	1

LOGIT_ALL model exhibit a high variability in the higher regions; however, they align very well with the LOGIT_SAMPLE model in the lower regions. A higher number of cases would probably yield somewhat better results (i.e. smoother scoring curves).

Figure 3.14: *Comparison of complexity scores between all models.*



Based on a linear combination of the variables, the values of the three calculated models do coincide to some extent. The CALC_SAMP LE model yields the smoothest scoring curve. This is clear when we draw a trend line through it: it achieves a $R^2 = 0955$, which is quite high. If we construct the same graph in the LOGIT_ALL sequence of values, we obtain less smooth results, and a lower R^2. The CALC_SAMPLE curve therefore appears to be the best alternative to the Logit models – if simplicity is required.

Although all models seem to serve this purpose in one way or another, we suggest the following quality ratings: LOGIT_SAMPLE has the highest discriminatory power and should be used when workstations are classified as COMPLEX or NOT. CALC_SAMPLE and CALC_ALL both exhibit smooth behaviour across the full range of complexity scores and can be used to attach a numerical measure of complexity (from 1 and 10) to a workstation.

3.3 Conclusion

In this chapter, we started by proposing a definition of production complexity comprehensive enough to characterize various manufacturing systems but also specific enough to define whether a system is of high or low in complexity. To achieve this, a set of direct complexity drivers were extracted from real production data and information provided by manufacturers.

From field research in several automotive companies, we were able to create a structured causal model of the elements and consequences of complexity. With this model, we identified 11 workstation and product line characteristics that determine complexity. We then constructed five different classification models, three of which were calculated as linear combinations of these characteristics and two of which were statistically derived logistic models. All of these models were compared to the subjective labels of HIGH or LOW complexity which we obtained from the operators working at these stations.

We concluded that two models are of particular importance. The first is a logistic model that classifies the complexity of workstations as HIGH or LOW (LOGIT_SAMPLE), as obtained from a sample of 54 stations with extreme scores. This model is suitable for extracting highly complex workstations from engineering databases when only four variables are used as input data. The second is a calculated linear model (CALC_SAMPLE) with weights derived from the Logit model we mentioned above. It yields a good gradual scale for workstation complexity between 0 and 10. These two models provided better results since they were implemented based on real-world situations. By excluding the outliers, we could obtain useful and relevant information.

The second Logit model (LOGIT_ALL), based on all 76 workstation data points, also yields a gradual scale for the probability that a workstation of HIGH in complexity. This model is somewhat less intuitive, however, as it is based on a ratio of exponential functions.

The complexity analysis we carried out provides a clear insight into the ways complexity is experienced in MMALs. In this chapter, we presented an approach to classifying complex workstations. In the next chapter, we will propose new solutions to cope with this complexity we have detected.

4

Workload Balancing and Manufacturing Complexity Levelling in Mixed-Model Assembly Lines

In Chapter 3, we introduced a subjective definition of complexity based on judgement and presented a classification model to characterize workstation complexity as ‘high’ or ‘low’. This classification allows us to identify complex workstations and give an overview of the overall complexity of an entire line. However, it is also necessary to delineate ways to reduce and even out complexity.

Manufacturing complexity is usually high in mixed-model lines and has reached new levels with the recent increase in model variability. Coping with highly complex lines is a key challenge for manufacturing systems, especially for operators who deal with a wide range of choices and are challenged to complete their operations on time. Complexity therefore goes hand in hand with workload distribution. One way to cope with this problem is to design one line for each model;

however, this is not efficient, realistic or optimal. Another approach is to first develop a means to quantify complexity and then level it. Continuing from our previous classification of the complexity of workstations, in this chapter we will first propose a method to objectively measure the overall complexity, on the one hand, and complexity at each workstation in the assembly line, on the other hand. Then, this measurement will be integrated into a procedure that balances workload in MMALs, in order to monitor and level manufacturing complexity.

We begin this chapter in Section 4.1 by introducing a quantitative measurement of manufacturing complexity at workstations. Then, in Section 4.2, we present a mathematical model for MMALs and propose a feasible balancing solution for minimizing workstation overload. Finally, in Section 4.3, we describe our proposed mixed-model assembly line balancing procedure to reduce work overload while levelling manufacturing complexity across assembly lines. In Section 4.4, we demonstrate some computational results for applying this approach to datasets. The parameters used are the same described in Section 1.1.

4.1 Quantitative Manufacturing Complexity Measurement

Considering MMALs, the objective is to level manufacturing complexity while still balancing the line. To achieve these two objectives, quantifying manufacturing complexity is a necessary first step. Shannon (1948) introduced entropy as a measure and a means of quantifying complexity in information systems. Typically, system complexity increases along with growing levels of disorder and uncertainty. Therefore, a higher complexity system requires a larger amount of information to describe its state.

Measurements based on entropy have since been used and developed by researchers to quantify complexity in manufacturing. Some examples involve product assemblability (Fujimoto and Ahmed, 2001), supplier-customer systems (Sivadasan et al., 2006), input demand variety (Abad and Jin, 2011) and operator number of choices (Zhu et al., 2008). This situation is also encountered in MMALs and is especially experienced by operators in different workstations who are flooded by operations information.

The complexity measurement proposed relies on the use of the concept of entropy in combination with assembly task time variation. It is defined as $Complexity_{AJ_k}$ and quantifies workstation complexity. As previously stated, task time may vary from model to model as a result of different model requirements. Consequently, the same task can entail different execution times. $Complexity_{AJ_k}$ can be represented as follows:

$$Complexity_{AJ_k} = H(AJ_k) + \sum_{j \in AJ_k} \sigma j \tag{4.1}$$

Where:

$$H(AJ_k) = - \sum_{j \in AJ_k} p_j \, log \, p_j \tag{4.2}$$

$$\sigma j = \sqrt{\frac{1}{|M| - 1} \sum_{1}^{M} (t_{jm} - \mu_j)^2} \tag{4.3}$$

AJ_k is the subset of tasks assigned to workstation k;

p_j is the occurrence probability of task j;

t_{jm} is the task time of task j for model m;

μ_j is the mean value of all models' task times for task j, and

σj is the standard deviation value of task j.

$H(AJ_k)$ can be used to assess workstation complexity during assembly. Here, p_j is the probability of task j being required given the set of models in M. We added the task time variation values resulting from the various models. This variation is quantified by σj and is added to the classical entropy measurement for complexity.

4.2 Initial MILP Model

To solve the MMALBP, we first developed an optimization Mixed-Integer Linear Programming (MILP) model. This model can provide an optimal solution to the MMALBP but is extremely time-consuming and computationally intensive. Although this MILP model requires high machine performance, it is useful to perform efficiency evaluations of heuristic solutions. The proposed model is represented as follows:

Variables:

$Overload_{km} \geq 0$ represents overload at workstation k for model m,

$X_{kj} \in 0, 1$ is 1 if task j is assigned to workstation k, otherwise 0,

$Y_k \in 0, 1$ is 1 if workstation k is open, otherwise 0,

$Op_k \in Z_+$ is the number of operators assigned to workstation k if open, otherwise 0.

The objective function represented in Equation 4.4 minimizes the overload of each workstation k for each model m. It also takes into account the demand b_m for each model m.

Mathematical Model Formulation:

$$Min \sum_{k \in K} \sum_{m \in M} Overload_{km} \times b_m \tag{4.4}$$

subject to:

$$\sum_{k \in KQual_j} X_{kj} = 1, \forall j \tag{4.5}$$

$$\sum_{k \in K} kX_{kj} \leq \sum_{k \in K} kX_{ki}, \forall i \, and \forall j \in Pred(i) \tag{4.6}$$

$$\begin{aligned} X_{kj} - Y_k \leq 0, \forall k, j \\ Y_k - Y_{k-1} \leq 0, \forall k \geq 1 \end{aligned} \tag{4.7}$$

$$\begin{aligned} \sum_{j \in J} \times X_{kj} \leq Overl_{km} + c \times Op(k), \forall m, k \\ Op_k - maxOp(k)Y_k \leq 0, \forall k \end{aligned} \tag{4.8}$$

Set of Constraints:

Eq. 4.5 ensures that each task j is assigned to one workstation k; Eq. 4.6 means that a task j is only assigned as soon as all tasks in $Pred(j)$ have been assigned. Eq. 4.7 determines the order in which workstations k are defined as 'open'. Finally, the most important restriction is defined in Eq. 4.8, which guarantees that the load of workstation k for model m is as low as possible, i.e. that the sum of all tasks assigned to workstation k is below the cycle time c. It also considers the number of operators Op_k at the workstation k and guarantees that this number does not exceed the maximum number of operators $maxOp(k)$ that can be assigned to that workstation.

While the MILP model presented generates a solution to the MMALBP, it does not take into account the existing complexity of a system. Although the maximum number of operators is one of the parameters of this model, it does not consider an important constraint of the problem regarding qualified operators. The results generated with this model provide a useful first approach to the problem and can also be used in other improvement approaches. Since our aim is to explore workstation complexity, we will next focus on integrating complexity into a solution approach.

4.3 Line Balancing and Complexity Levelling Solution Approach

In this section, we propose our solution for balancing and levelling the MMALBP. First, we introduce the concept of the 'super model' (as described below). Then, we detail a solution that is primarily based on two procedures: a hybrid heuristic algorithm for building a balancing solution and a local search algorithm for optimizing the existing balance – thus providing a rebalanced solution.

4.3.1 Defining a Super Model

Since this study focuses on MMALs, we considered model variation and corresponding task time variability. The assembly time of task j may differ from model to model; this task variability is taken into account with our concept of 'super model' (sm). A sm is an effective representation of all models regarding task time variation when considering task j. As a result, the task time of a sm is calculated using all possible operation times for task j. This variability is characterized by the following equation:

$$t_{jsm} = \sum_{m=1}^{M} t_{jm} \times b_m + \alpha \times (t_j^{max} - \sum_{m=1}^{M} t_{jm} * b_m) \qquad (4.9)$$

Where:
t_{jsm} is the task time of task j for super model sm,
t_{jm} is the task time of task j for model m,
b_m is the demand of each model m,
α is a variant value between 0 and 1, and
t_j^{max} is the maximum task time of task j.

To generate t_{jsm}, we took into account the demand b_m for each model m and the maximum task time t_j^{max} of task j. α is a calibration parameter that assumes five different values from 0 to 1 at intervals of 0.25. By applying this variation, different values of t_{jm} are considered and a set of five solutions can be produced . The definition of the super model (sm) is clearly described by the following small example.

Three models (m1, m2 and m3) are produced simultaneously in the same MMAL. The demands for each model are 25%, 25% and 50%, respectively. The possible values for t_{jsm} and t_{jm} are shown for the task times of three tasks in Table 4.1:

Table 4.1: *Range of numerical values as measured over 76 workstations.*

	m1	m2	m3	sm	sm	sm	sm	sm
demand	0.25	0.25	0.5	α=0	α=0.25	α=0.5	α=0.75	α=1
t1	1	1	3	2	2.25	2.5	2.75	3
t2	3	4	2	2.75	3.06	3.38	3.69	4
t3	2	1	1	1.25	1.44	1.63	1.81	2

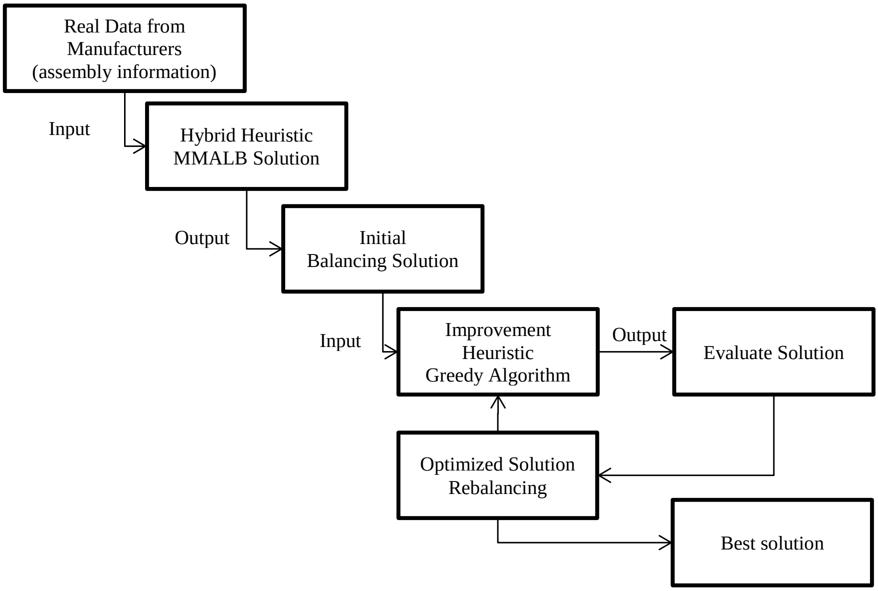
A set of five different t_{jsm} were calculated for each task j, which all vary according to different values of α. These values were used during the execution of the solution developed . It is therefore clear that the use of different task times for super models facilitates the generation of efficient solutions to the problem.

It should be emphasized that when a super model is used to generate solutions for mixed-model problems, different results may be produced for each model. Even when an optimal task assignment is obtained for the super model, the assignment might not be optimal when applied to each individual model. This is a result of the task time variability among models.

4.3.2 Solution Approach

Due to the combinatorial nature of balancing problems, which are known as NP-hard (non-deterministic polynomial-time hard) problems, it is difficult to obtain solutions for adopting mathematical and exact methods within adequate computational timeframes. As a result, we developed a set of procedures to address this problem. Figure 4.1 depicts the outline of our solution approach.

Figure 4.1: *Heuristics to Minimize Workstation Work Overload Based on Complexity*



These procedures start with the initialization of all parameters based on real assembly data from manufactures. Then, an initial balancing solution is generated using a hybrid heuristic algorithm; this solution relies on the use of super models. After a feasible solution has been defined, the results are evaluated, if they are not satisfactory, an improvement heuristic is applied for optimization. While the balancing solution considers the super model, the rebalancing solution relies on the work overload of all models. In the following subsections, this outline will be detailed.

4.3.2.1 Hybrid Heuristic Algorithm for Workload Balancing and Complexity Levelling in Mixed-Model Assembly Lines

To generate a feasible assembly line balancing solution while levelling workstation complexity, a procedure consisting of a hybrid heuristic algorithm is developed. This procedure relies on defining an optimal task and an operator assignment by taking into account workstation overload and complexity. This hybrid solution combines a number of different algorithms to address the order of task assignment, objective functions and task time variability. It is described in the following steps:

Step 1. Defining task time

sm task times are populated based on all t_{jm}. Initially α assumes a value of 0.

Step 2. Multi-criterion task prioritization

First, task assignment order is determined. We considered two prioritization criteria:

I . The precedence relationship between tasks: task j can only be assigned if all preceding tasks have already been assigned.

II . Ranked by assignment position weight ($RPW(j)$), as proposed by Helgeson and Birnie (1961). After the precedence relationship diagram has been defined, task weight is calculated based on the longest path between the first and last tasks in the network.

Step 3. Assigning tasks to workstations

After we have established which task j is to be assigned, a workstation k is selected. This selection relies on the following factors:

I . Workstation k load $LoadAJ_k$: tasks are assigned to the workstation with the highest idle time or lowest overload.

II . Workstation k complexity $Complexity_{AJ_k}$: tasks are assigned to the workstation with the lowest complexity level.

The workstation is selected by combining these two values. As both values are relevant, they carry equal weights of 50%. The entropy of each station is taken into account during the assignment of tasks. Tasks are assigned addressing the entropy levels of all workstations, aiming to obtain an equalized distribution of complexity.

In this step, two constraints must be respected: workstation k belongs to $KQual_j$ and operator o belongs $OQual_j$.

Step 4. Addressing task operation time variety

The steps outlined above are repeated for $\alpha = 0.25$, $\alpha = 0.50$, $\alpha = 0.75$ and $\alpha = 1.00$. The solutions obtained are stored in a set.

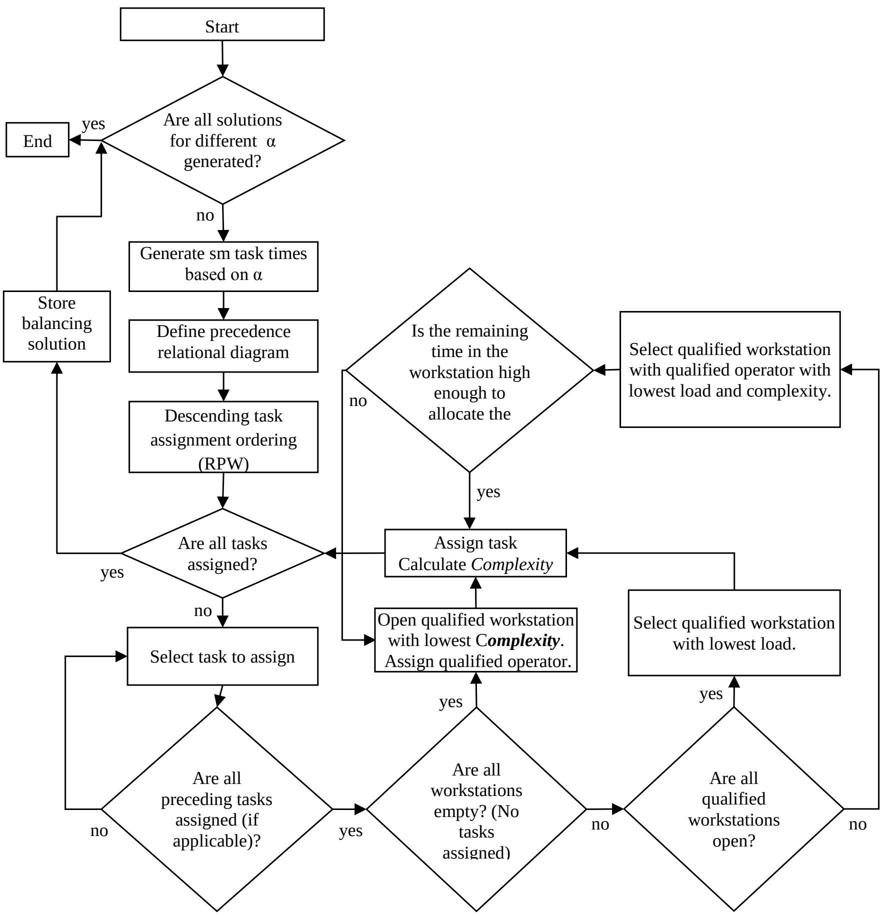
Step 5. Solution evaluation

Based on the workstation's work overload, the best solution is selected from among the previous five interactions. This solution may involve two possible situations : i) no work overload is registered, or ii) some work overload is registered.

If no work overload is registered, this means that an optimal solution has been found in terms of workstation load. Whereas workstation complexity is used during the constructive phase to determine which workstation should be selected for task assignment, it is not used to determine the best balancing solution. In that case, workstation load takes priority. The algorithm proposed is depicted in the following diagram (Figure 4.2).

As a result of this heuristic, a feasible solution can be obtained which may be either optimal or satisfactory. Since our approach relies on real scenarios, a satisfactory solution would be based on a limited parameterization performed by the user and/or an expert. M aximum values for complexity and work overload can thus be defined in advance so acceptable values for workstation complexity and work overload reflect realistic scenarios during production.

Figure 4.2: *Hybrid Heuristic Algorithm to Balance Mixed-Model Assembly Lines to Minimize Work Overload and Complexity*



In real situations, work overload might occur as a result of tasks being assigned to specific workstations, when certain operations need to be executed exclusively by qualified operators at qualified workstations.

4.3.2.2 Local Search Optimization Algorithm for Rebalancing and Reducing Work Overload

To optimize existing solutions, reduce further workstation work overload and equalize complexity workstation levels, we developed a rebalancing procedure. This improvement heuristic relies on a local search algorithm that considers task permutations between workstations (based on workstation load and entropy) to generate the best assignment solution possible. It can be described as follows:

Step 1. Importing an existing solution

The first step consists in specifying an initial solution based on one model of the MMALBP. This solution is delimited as input and may be the result of a manual balance, an exact method or the hybrid heuristics we described in the previous section. The user must also define how many times this procedure should be executed.

Step 2. Calculating total work overload per workstation

In this step, the balancing solution specified in Step 1 is applied to each model. The total work overload is calculated for each workstation based on all models. This value is represented by Equation 4.10:

$$TotalOverload(k) = \sum_{m \in M} Overload_{km} \tag{4.10}$$

Where $TotalOverload(k)$ is the sum of the work overload for each model at workstation k.

Step 3. Task permutation

The tasks assigned to workstations with registered work overload are then permuted to other workstations. This permutation is based on a number of factors:

I . Tasks to be permuted are selected in descending assignment order. As a result of precedence relationship constraints (and to keep permutations to a minimum) the last tasks assigned are the first to be permuted.

II . Task selection is based on a local search. The 'task neighbourhood' search explores possible solutions based on workstation workload.

III . The workstation from which the tasks will be permuted is selected according to the highest overload.

IV . The workstation to which the tasks will be permuted is selected according to the lowest workload and lowest complexity level.

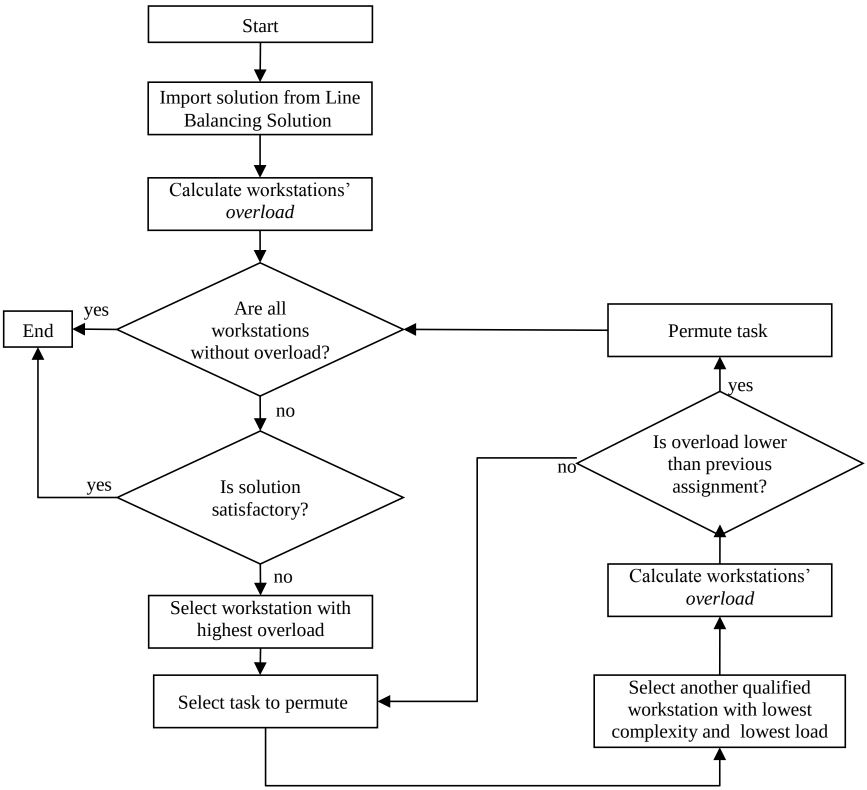
V . The permutation is based first and foremost on the tasks assigned to one workstation. Tasks are permuted only if the entropy level of the workstation is reduced. When no improvement is found, the next workstation with registered overload is considered. Workstations are handled in descending order according to work overload and complexity level.

Step 4. Solution evaluation

After each task permutation, total work overload per workstation is again computed. As a result of the local search procedure, the current solution in each interaction is analysed by taking a set of candidate tasks for permutation into account. The aim is to find a global optimal solution through permutations based on a local search; in other words, tasks are permuted to minimize work overload and level complexity. Each solution is considered only if an improvement is made. While the complexity level of a workstation is used to determine which workstations tasks are permuted, the best solution is ultimately defined by workstation overload.

This rebalancing procedure is repeated until one of the three stop conditions is reached: (i) the optimal solution is found; (ii) the solution meets the workstation load bound desired; or (iii) the number of interactions defined by the user is reached. The algorithm proposed is depicted in the following diagram (Figure 4.3).

Figure 4.3: *Local Algorithm for Rebalancing Mixed-Model Assembly Lines for Solution Optimization*



4.4 Computational Results

This section presents a number of computational results used to verify the proposed complexity measurement and the mixed-model assembly line balancing solutions. We describe and compare the results obtained for the MMABP in the heuristic procedures described in sections 4.3.2.1 and 4.3.2.2.

Our proposed approach has been tested on three datasets of different sizes (Rosenberg and Ziegler, 1993; Wee and Magazine, 1981; Scholl, 1993). The number of tasks varies between a minimum of 25 and a maximum of 297. Moreover, two product mixes were considered:

two models and four models . We also took the five different alphas into account to obtain task time variability and a suitable rebalancing solution.

We developed tests for solutions with and without the use of complexity measurements. Taking all of these aspects into account, we obtained a total number of 72 cases. We extracted the datasets from Tiacci (2015b), in which the precedence diagrams of the problem are also represented. The demand for the two-model datasets was equally distributed, with 50% for each model; the demand for the four-model datasets was distributed as 50%, 20%, 10% and 10%.

To generate these datasets, Tiacci (2015a) used the original datasets for SALPB developed by Scholl (2007). With a single-model problem data, he proposed mixed-model data by varying the task times for both the two- and the four-model instances by means of a variation coefficient of 0.3. During our experiments, we used the proposed cycle time, number of workstations and tasks used by Scholl (2007) for the SALPB. This information is shown in table 4.2.

Table 4.2: *Computational Tests - Cycletime and Number of Workstations*

Dataset	CycleTime	#Workstations	#Tasks
Rosenberg and Ziegler (1993)	21	6	25
Wee and Magazine (1981)	43	50	75
Scholl (1993)	1515	46	297

The implementation of the procedures previously described (in Section 4.3) was split into two parts: the constructive balancing heuristic and the improvement heuristic. The results obtained for each solution were generated based on the various alphas. The algorithms were coded in the programming language C++ and tested on an Intel Core i5-4210U with a 1.70GHz/2.40GHz CPU and 8 GB RAM memory.

During the execution of the hybrid heuristic intended to define a preliminary solution, the calculation of $Complexity_{AJ_k}$ (workstation complexity) and $Load_{AJ_k}$ (workstation workload) measurements were dynamically performed during the interactions of the algorithms . The resulting combined measurement was afterwards used in the next task assignment.

Table 4.3: *Computational Tests - Average Overload (%) per workstation*

Problem	M	Size	UC	$\alpha = 0$	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.75$	$\alpha = 1$	R	CPU time (s)
Roszieg	2	25	n	5.57%	2.38%	3.95%	3.95%	7.95%	2.38%	0.51
			y	5.57%	2.38%	3.95%	3.95%	7.95%	2.38%	1.19
Roszieg	4	25	n	4.38%	3.95%	4.57%	7.52%	8.95%	3.95%	0.74
			y	5.76%	3.38%	4.76%	6.38%	6.38%	3.38%	1.70
Wee-mag	2	75	n	1.14%	1.23%	1.02%	1.16%	1.19%	0.81%	9.05
			y	1.23%	1.44%	1.09%	1.26%	1.42%	1.09%	10.62
Wee-mag	4	75	n	1.19%	1.14%	0.98%	1.05%	0.98%	0.86%	12.78
			y	1.28%	1.33%	1.09%	1.14%	1.21%	1.09%	12.98
Scholl	2	297	n	3.85%	4.55%	4.23%	4.55%	6.32%	2.34%	60.24
			y	4.59%	4.82%	3.84%	3.85%	5.11%	3.84%	68.49
Scholl	4	297	n	4.11%	5.22%	5.15%	6.10%	6.96%	2.57%	65.98
			y	4.55%	4.07%	4.27%	5.96%	5.62%	4.07%	70.96

* M refers to the *number of models*
UC to the *Use of Complexity measurement*
n to the *no* and y to the *yes*
R to *Rebalancing*

Tasks were assigned to workstations with the lowest combined value, on the assumption that both $Complexity_{AJ_k}$ and $Load_{AJ_k}$ carried an equal weight of 50%. During the optimization rebalancing process, tasks are permuted from workstations with the highest to the lowest combined value while reducing workstation work overload for each station and each model, $Overload_{mk}$.

The total complexity of the workstations is the same for all solutions, but there is a difference in the way in which this complexity is spread across workstations. The sum of all workstation complexities remains constant as it is based on tasks times. The implemented framework is used to test the proposed approach in an industrial case. In Table 4.3, we show the results for all datasets and display the average overload per workstation per solution (in %), considering cycle time. CPU times vary from 0.51 of a second to 70.96 seconds. The number of tasks has a high impact on the processing time, because the assignment and permutation of tasks are the core of the problem.

The figures below (Figures 4.4, 4.5 and 4.6) give an overview of the results we obtained for these datasets. Our heuristics were able to generate feasible solutions for the 4 datasets. Workstation overload (in most cases) was below 5% of cycle time. As the values presented consist of average overload per workstation and per solution, cases in which overload is registered for specific models are not displayed but instead shown as an average. Some specific models have a much higher assembling time, generating workload peaks along the line. This is taken into account by our complexity measurement, which combines the uncertainties of the required tasks and their task time variations.

From these tests, we revealed three aspects that needed to be addressed: (i) the number of models, (ii) the number of assembly tasks, and (iii) the use of the complexity measurement. These three aspects all impact on the overload results. On e of the key advantages of using heuristics over exact methods is their computational time. Heuristics can be applied to larger datasets to solve real-world scenarios, their computational time is relatively low, and the approximated solutions they produce are satisfactory. They also offer more flexibility when dealing with very complex problems.

Figure 4.4: *Rosenberg and Ziegler (1993)*

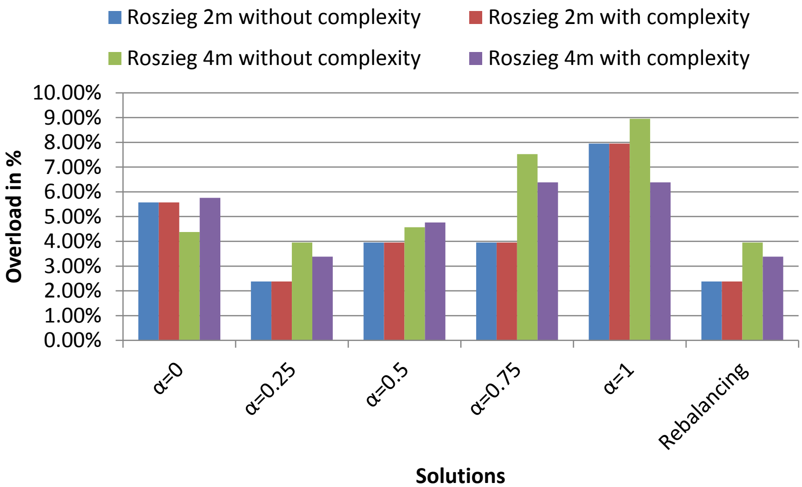


Figure 4.5: *Wee and Magazine (1981)*

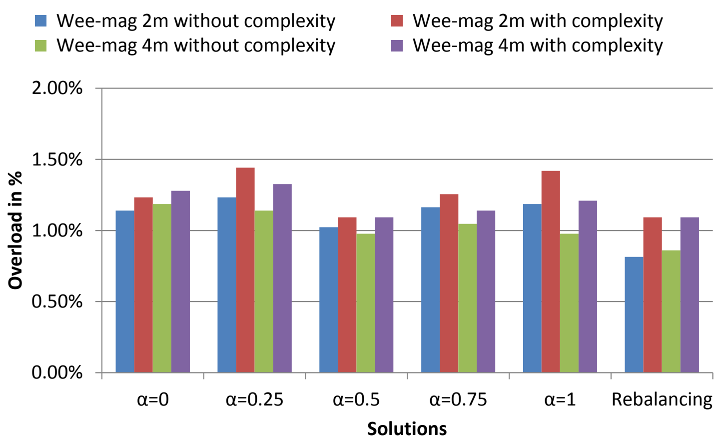


Figure 4.7 demonstrates the complete results of all datasets when complexity measurement is not considered and workstation complexity is levelled. These datasets also take complexity into account.

Figure 4.6: Scholl (1993)

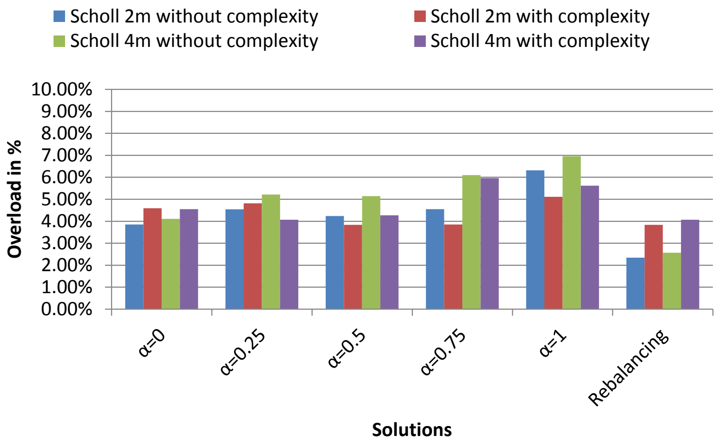
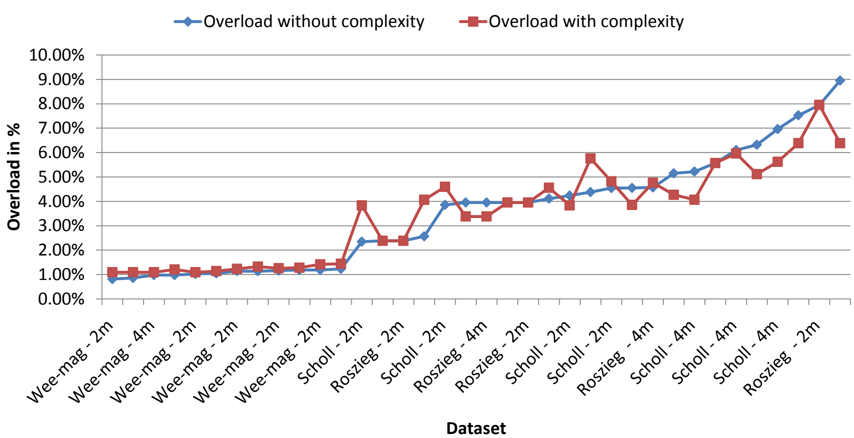


Figure 4.7: Overall Results



We can expect an increase in overload as both workload and complexity levels are considered. However, due to the nature of the heuristics and their flexibility, this may not occur. In some cases, we observed (considering the levelling of complexity) a reduced overload, while in other cases, we did not. This is also a result of the task time variability for the different models.

The datasets and solutions available in the literature do not contain the same parameterization as the problem investigated in this dissertation. Typically, data files only represent information regarding the number of tasks, the number of models, the task times and the precedence relationships. This study reflects real-world assembly lines and therefore also considers other parameters and restrictions, such as demand, qualified operators and workstations. We were able to use these datasets by adapting and generating values randomly. Another crucial aspect is the use of manufacturing complexity. As the problems have a distinct characterization, it is not possible to compare our results with the results available.

4.5 Conclusion

In this chapter, we developed a workstation complexity measurement which correlates the entropy of a system with task assignments. Subsequently, a set of heuristic procedures was implemented to solve MMALBPS. The solutions were mainly based on a hybrid algorithm to generate line balance and a local search algorithm to rebalance the line. Our main goal was to level manufacturing complexity while minimizing workstation work overload. The effectiveness of our approach in balancing and levelling complexity was demonstrated on some datasets available in the literature. It could be concluded that the heuristics implemented were able to generate solutions in which the majority of workstations were well-balanced.

5

Industrial Case Study

In this chapter, we will focus on an industrial case study involving the conceptualizations described in the previous chapters. The computational results that were obtained from this real-life application will be reported in detail. First, we will outline our data-analysis strategy. Second, we will apply our methods to identify complexity and classify complex workstations, as proposed in Chapter 3. Third, by means of the solution presented in Chapter 4, we will exploit complexity to level out the manufacturing complexity of MMALs. Finally, several conclusions will be drawn about the results obtained.

5.1 Data Analysis

The industrial case study in question was a Belgian supplier of assembled automotive components. This supplier produces parts for the main car manufacturer in the region. The relationship between the supplier and the car manufacturer is interactive: the two companies are semi-synchronized because of their proximity to one another. The demand imposed on this mixed-model car line is met by the prompt communication of requests and delivery information between the supplier and the producer. Many options can be selected and assembled, and parts are customized according to various model requirements. This study was conducted in two phases, both of which examining the same part of this assembly line. In the first phase, we focused on analysing complexity and classifying complex workstations; in the second, we worked on levelling complexity. We drew the dataset we used from a real-world assembly line consisting of 109 tasks, nine models, nine workstations and nine operators with a cycle time of 44.2s. This dataset is a subset of the dataset used in Chapter 3.

In the first phase, we gathered data based on real scenarios presented during workshops with the manufacturer (Section 3.1). This data was identified as complexity drivers and was restructured as 11 variables. Our aim was to analyse workstation complexity and identify workstations that are high or low in complexity. After determining the 11 variables, we requested specific information for each variable from the workshop participants. The information that we obtained for our case study can be found in table 5.1 and was later used as input for our methods (Section 3.2) for classifying workstation complexity.

Table 5.1: *Values of Complexity Driver Variables*

Variable	k1	k2	k3	k4	k5	k6	k7	k8	k9
Picking technology	F	C	C	C	C	C	C	C	C
Bulk/Sequence Kit	B	B	B	B	B	B	B	B	B
# Packaging types	2	4	2	3	6	4	2	0	0
# Tools per workstation	2	3	1	2	3	1	1	2	2
# Machines per workstation	1	0	0	0	0	0	0	0	0
# Work methods	3	1	1	6	9	9	7	9	9
Distance to parts	2	2	2	2	2	2	2	2	2
# Variants same model	4	4	4	4	4	4	4	4	4
# Variants in this workstation	4	4	4	4	4	4	4	4	4
# Different parts in workstation	2	8	6	5	9	8	7	0	0
# Assembly directions	5	7	3	13	17	7	5	10	10
Complexity Level (H/L)	0	1	0	0	1	1	1	0	0

k stands for workstation

In the second phase, through regular visits to the plants and close collaboration with the manufacturer, we conducted a thorough investigation of workstation balance. Our aim was to ascertain how complexity affects workload distribution among workstations and how it can be used to improve line balancing. We also gathered information on task assignment, task processing times, precedence relationships between tasks, workstation and operator qualifications, cycle time boundaries, and model variants. Our data could be divided into four parts:

– General information regarding exact cycle times and the numbers of models, tasks, workstations and operators;

– Specific information regarding tasks with preceding tasks, qualified operators and workstations;

– The maximum number of operators per workstation;

– Task time per model.

This information was then collected into an input data file with a fixed format. Our goal was to represent the production data that would be used by the exact and approximate solutions developed.

5.2 Classifying Complex Workstations

We first analysed workstation complexity with the variables and characterizations presented in Chapter 3. The five statistical models that were previously developed were applied to classifying workstations as being low or high in complexity. These five models, named BASE, CALC_ALL, CALC_SAMPLE, LOGIT_ALL and LOGIT_SAMPLE, produced the results provided in Figures 5.1 and 5.2.

According to the subjective complexity classifications provided by the manufacturer, four workstations were classified as being high in complexity and five as low in complexity. This information is shown by the green and red markers in the above figures.

Table 5.2 is based on our initial subjective classification and provides an overview of the results we obtained. All models correctly identified Workstations 1 and 3 as being highly complex. The least accurate classification was generated by the LOGIT_ALL model; only 22% of the workstations were classified correctly.

The CALC_SAMPLE model presented the most accurate classification. This is a linear model based on the four variables of the LOGIT_SAMPLE model, the weights used were also those determined by the statistical model (Section 4.3).

Figure 5.1: *Workstation Complexity Analysis*
Logit Models

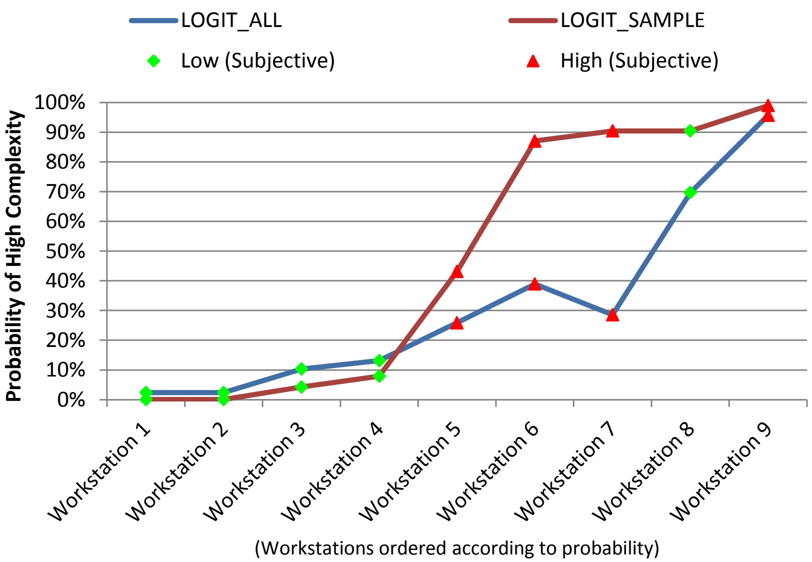


Figure 5.2: *Workstation Complexity Analysis*
Linear Models

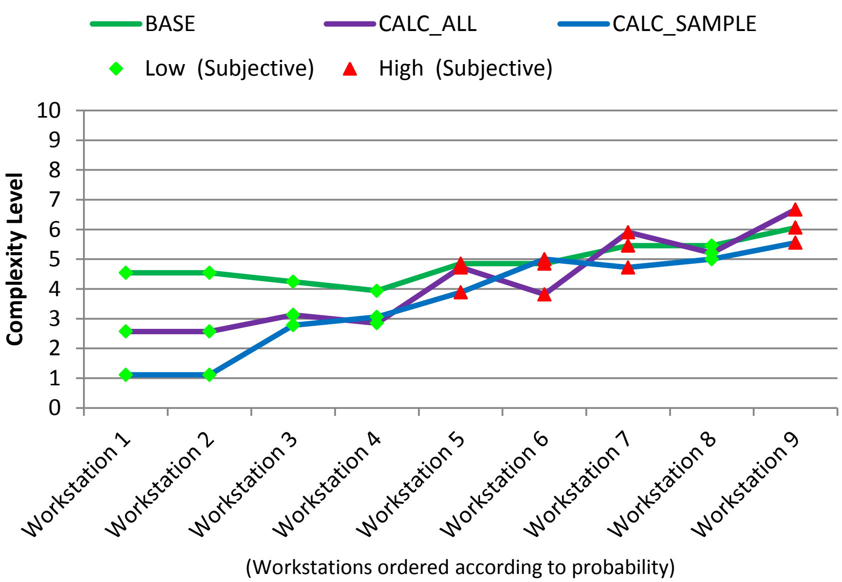


Table 5.2: Workstations' classification

Variable	k1	k2	k3	k4	k5	k6	k7	k8	k9	Total
Subjective	L	H	L	L	H	H	H	L	L	
BASE	L	H	L	H	H	H	H	H	H	67%
CALC_ALL	L	L	L	H	H	H	H	L	L	67%
CALC_SAMPLE	L	H	L	H	H	H	L	L	L	78%
LOGIT_ALL	L	L	L	L	L	L	L	H	H	22%
LOGIT_SAMPLE	L	L	L	L	L	H	H	H	H	67%

k stands for workstation
H stands for high complex workstation
L stands for low complex workstation

Most automotive companies have large engineering databases containing technical and operational data on all tasks (i.e. work elements) executed at their workstations. Currently, assigning tasks to workstations in an effort to balance the line is often the responsibility of operators, team leaders and engineers. Naturally, their experience and personal judgement are determining factors in this process. W e developed the classification models reported in Chapter 3 after interacting with our industry partners and observing real cases. The models may facilitate the extraction of more meaningful information from engineering databases and thus assist in task assignment. They help to pinpoint interesting workstations, and suggest variables that could be causing balance loss by identifying complexity drivers that can generate a correct classification.

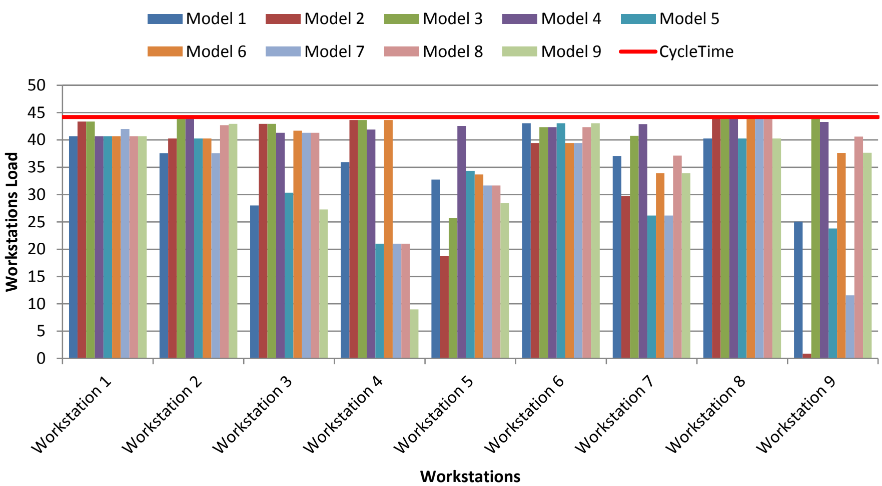
5.3 Workload Balancing and Manufacturing Complexity Levelling

After exploring manufacturing complexity and identifying workstations that are high or low in complexity, we developed tests by means of the modelling and heuristic approaches. Complexity drivers and workstation classifications provide a broader and substantial view of manufacturing complexity (Chapter 3), which is extremely relevant to identify different related aspects of complexity. However , measuring and monitoring complexity attempting to achieve workload balance leads to a levelled manufacturing complexity at each workstation (Chapter 4). The results obtained are presented below.

5.3.1 Modelling Approach

With the MMALBP as our starting point, we first developed an optimization MILP model. With this model, we were able to provide an optimal solution to the problem but we were unable to take an important constraint into account, namely qualified operators. Although the maximum number of operators is one of the parameters of the model, we did not consider which operators were qualified to execute certain tasks. These results are shown in Figure 5.3.

Figure 5.3: *Balancing Mixed-Model to Minimize Work Overload MILP Model – Workstations Load*



Since one of the main constraints was excluded from this model, we expected the results to be better than a complete approach. However, it was interesting to observe how workstation loads still varied from model to model. For example, at Workstation 9, the load was much lower for Model 2 than for the other models. This is a typical means of recognizing the complexity generated by product variability on a line. T he next section presents the complete approach, which considers all constraints and integrates complexity into its solution.

5.3.2 Solution Approach

Using the same dataset from the previous sections, we tested the solution approach we had developed. We evaluated the results obtained by means of various alphas (section 4.3.1, $\alpha = 0.00$, $\alpha = 0.25$, $\alpha = 0.50$, $\alpha = 0.75$ and $\alpha = 1.00$) and rebalancing solutions. We also compared the results before and after levelling workstation complexity during the task assignment process. In addition, we considered the results of our improvement rebalancing heuristic, which are shown below and are grouped by three aspects: the variability of complexity levels, the average work overload per workstation when complexity is not levelled, and the impact of complexity.

Table 5.3 displays the values of workstation complexity levels, $Complexity_{AJ_k}$. The real balance was provided by the manufacturer, and the initial complexity measurement of each workstation is shown in the second column. We can observe a large discrepancy between complexity levels, and notice an especially high level of complexity at Workstation 6 and a particularly low level at Workstations 1, 2 and 3. T he complexity levels produced by our heuristic solution demonstrate a tendency to equalize complexity levels .

Table 5.3: *Workstation Complexity Levels*

	Real Balance	$\alpha = 0$	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.75$	$\alpha = 1$	R
k1	3.99	34.79	10.68	23.53	31.98	26.81	31.98
k2	1.19	13.90	24.49	30.48	20.37	28.03	20.37
k3	1.55	17.15	9.07	23.90	27.41	33.10	27.41
k4	50.52	31.63	16.65	33.71	32.11	19.00	32.11
k5	1.65	37.93	29.46	10.08	24.52	31.34	24.52
k6	102.33	85.61	12.54	30.43	18.78	38.66	18.78
k7	84.48	56.33	33.35	35.37	37.23	21.74	37.23
k8	26.07	0.00	82.48	40.55	46.10	48.26	46.10
k9	5.56	0.00	58.62	49.30	38.83	30.41	38.83

k refers to *workstation*

R to *rebalancing*

Figure 5.4 clearly represents how complexity is levelled through the solutions obtained, starting from very high and low peaks of complexity and evolving toward a more balanced solution. The optimal workstation complexity level is represented by an equal amount of complexity for all workstations. As the total amount of complexity in the system is 277, each workstation needs to have an equal complexity measurement of 31 (red line) for the situation to be ideal.

Figure 5.4: *Workstations Complexity Levels*

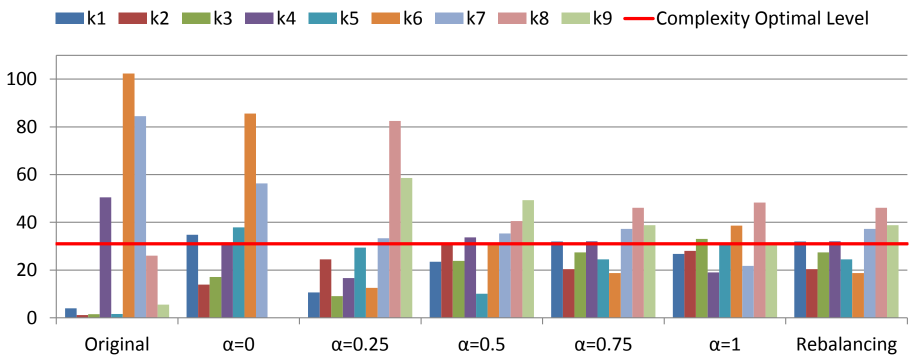


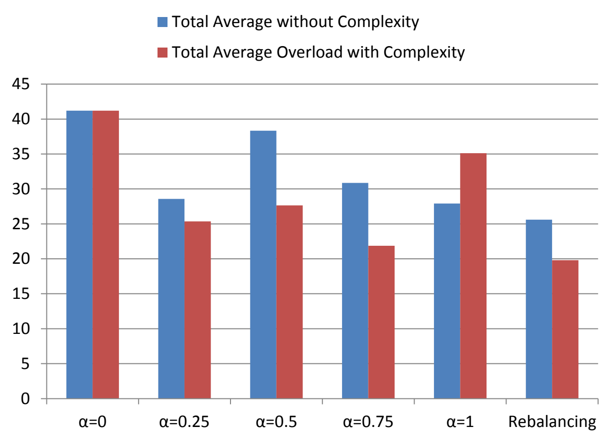
Table 5.4 displays the total work overload at all workstations per alpha after balancing and rebalancing. We obtained these results without levelling complexity. Since rebalancing is an optimization procedure for improving previous balancing solutions, it comes as no surprise that there is an improvement in the total overload. The highest work overload situation occurred when α was 0. In this case, we did not consider the task time variation for a single task in different models.

The results show how model variability influences line balance and the total average overload per workstation when complexity is considered . We have demonstrated that, even though both complexity and workload are taken into account, work overload is occasionally minimized. Figure 5.5 also summarizes these results and displays the total average overload per workstation per solution.

Table 5.4: *Workstations Overload (with/without) Levelling Complexity*

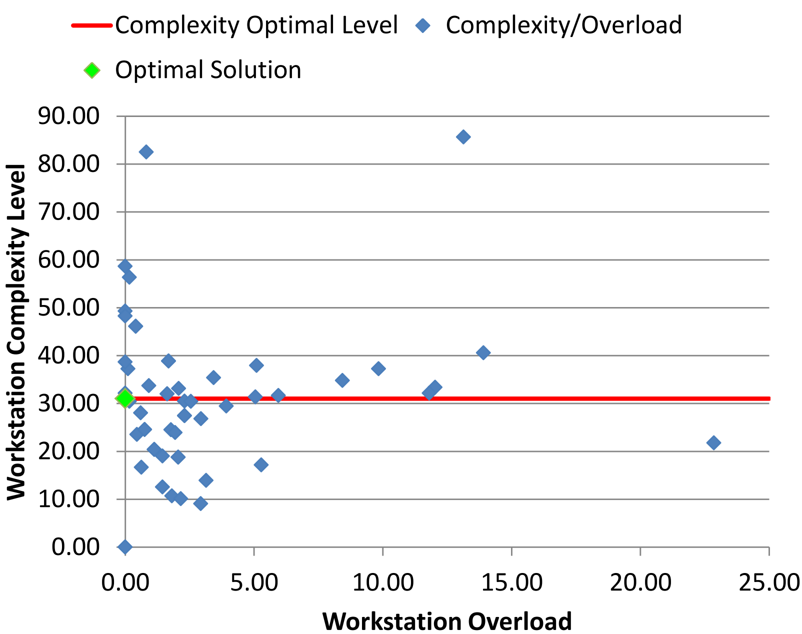
	$\alpha = 0$		$\alpha = 0.25$		$\alpha = 0.5$	
	n	y	n	y	n	y
k1	8.43	8.43	1.81	3.19	0.45	2.36
k2	3.14	3.14	1.77	2.03	2.3	3.28
k3	5.28	5.28	2.93	2.93	1.94	3.94
k4	5.95	5.95	0.63	0.63	0.92	0.33
k5	5.1	5.1	3.92	4.72	2.16	27.62
k6	13.13	13.13	1.45	2.25	2.54	0.8
k7	0.16	0.16	12.03	12.83	3.43	0
k8	0	0	0.82	0	13.9	0
k9	0	0	0	0	0	
Total Average	41.2	41.2	25.36	28.58	27.65	38.33
	$\alpha = 0.75$		$\alpha = 1$		R	
	n	y	n	y	n	y
k1	1.63	0.5	2.94	1.76	1.63	1.76
k2	1.13	1.89	0.6	13.58	1.13	0.45
k3	2.3	0.57	2.07	3.46	2.3	3.46
k4	11.8	0	1.44	2.9	0	2.9
k5	0.76	22.02	5.05	0.59	0.76	0.59
k6	2.06	5.37	0	0.8	2.06	0.8
k7	0.11	0.44	22.85	3.32	9.83	3.32
k8	0.41	0	0	0.54	0.41	11.34
k9	1.68	0.09	0.16	0.97	1.68	0.97
Total Average	21.87	30.87	35.12	27.92	19.8	25.59

Figure 5.5: *Mixed-Model Assembly Line Balancing Solutions Total Work Overload per Workstation*



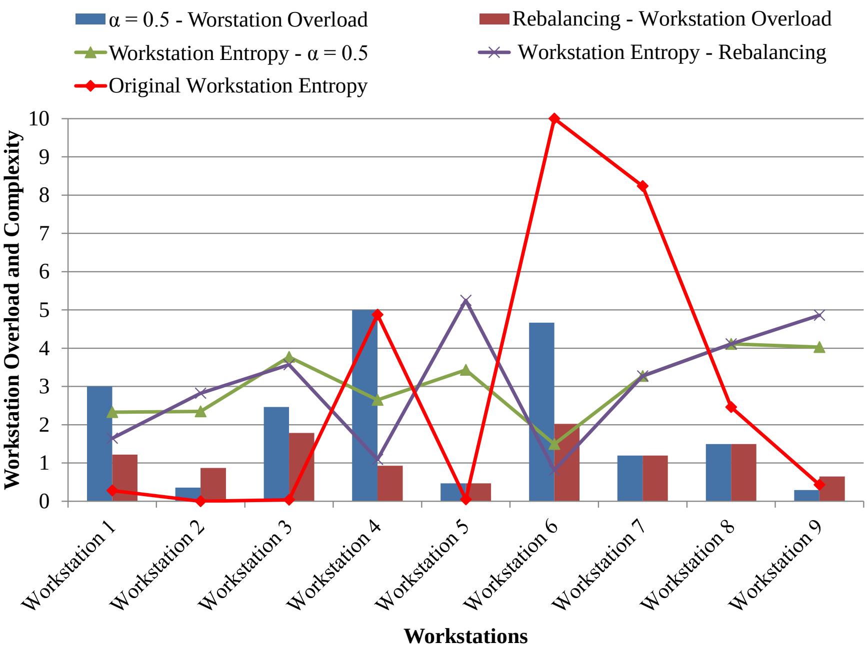
The complexity of workstations and variability of work overload are shown for all solutions in Figure 5.6. Green represents the best solution and is defined by a workstation work overload of zero and a complexity level of 31. It is important to mention that some tasks are assigned to fixed workstations as part of the initial set of constraints and these assignments cannot be changed by the line balancing solutions . Thus, from time to time, work overload is registered but cannot be optimized.

Figure 5.6: *Workstations' complexity variability and work overload*



An overview of both levelling and balancing results is given in Figure 5.7, which shows the complexity levels (i.e. entropy measurement) of the original balance, the result obtained for $\alpha = 0.5$, and the rebalancing procedure. The work overload for each station is also provided.

Figure 5.7: Workstations' complexity variability and work overload



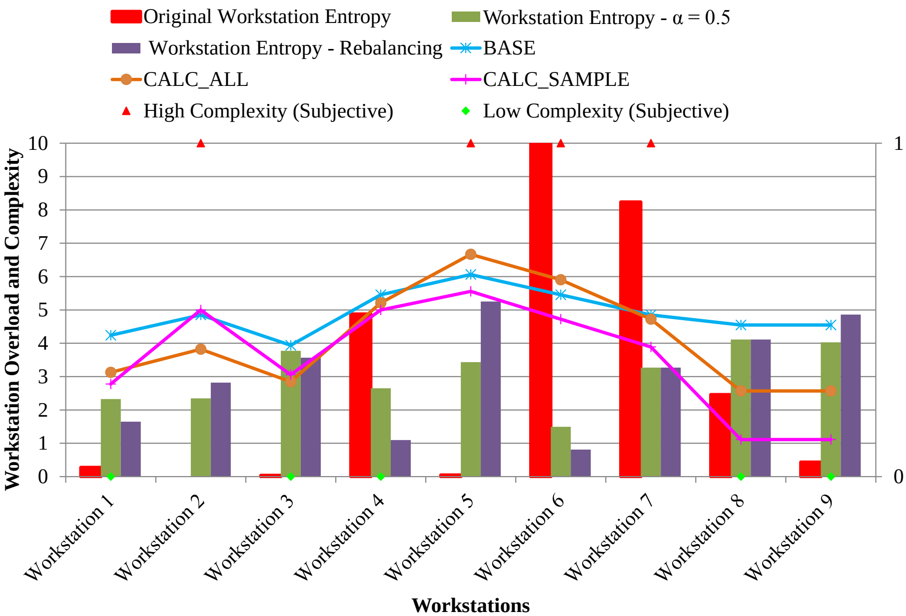
While the initial balance contains workstations that are very low in complexity (such as Workstations 1, 2 and 3) as well as workstations that are very high in complexity (such as Workstations 6 and 7), complexity levels are more levelled after the proposed solution has been applied. It can be concluded that levelling complexity with workstation entropic measurements improves both the complexity smoothness of each workstation and the overall complexity of the manufacturing system.

5.4 Comparison

As we investigated manufacturing complexity in several distinct ways, the results of these different studies will be compared in this section. First, a general overview of the complexity results will be presented, followed by a comparison of the load results of the levelling/balancing heuristics and the MILP model.

Figure 5.8 demonstrates the results obtained regarding complexity. The red and green markers represent the initial subjective workstation classifications provided by the manufacturers; workstations high in complexity are represented as 1 and those low in complexity as 0. This figure also shows the characterizations generated by the linear models (BASE, CALC_ALL and CALC_SAMPLE). The workstation complexity measurements (i.e. entropy) are provided for the original balance, for the best balancing (when $\alpha = 0.5$) and for the rebalancing solutions. To facilitate our analysis, we normalized the scores for each workstation on a scale from 0 to 10.

Figure 5.8: *Mixed-Model Assembly Line Balancing Solutions Complexity Measurement and Classification per Workstation*



We are able to observe the relation between the subjective classification of the workstations and the original entropy levels. For example, workstations 1 and 3 are both classified as low in complexity, and both originally contain a low level of complexity. The opposite occurs in workstations 6 and 7, which are simultaneously classified as high in complexity and with a high level of complexity. When analysing the classification models, we can observe the same tendency with the probability of workstation classification and the levels of complexity.

As mentioned previously (Section 5.3.1), an optimal solution is obtained by means of the MILP model. This approach is only valid for the MMALBP as it does not consider the constraints of numerous problems and in particular disregards manufacturing complexity. However, it might be used as lower bound for the optimization problem.

Table 5.5 shows the gap between the heuristics and the MILP results for the load of each workstation and each model. Workstations 5, 6 and 7 have a higher load for almost all models. This is the result of the complexity of these workstations.

Table 5.5: *Heuristics and MILP – Results: Gap difference Workstation's load*

	m1	m2	m3	m4	m5	m6	m7	m8	m9	**Average**
k1	-17.37	-5.21	**15.49**	-0.27	-10.35	-11.07	-22.32	-9.99	-20.97	-9.12
k2	-4.16	-6.86	**0.16**	-4.16	**11.6**	-6.86	**6.82**	**3.85**	-5.96	-0.62
k3	**0.60**	-3.00	**8.7**	**8.88**	-1.74	**9.78**	-11.37	-3.72	1.32	**1.05**
k4	-2.94	-1.82	-1.82	-18.70	-4.84	-5.56	-4.84	-4.84	**27.54**	-1.98
k5	**11.96**	**7.66**	**0.64**	**1.22**	**4.04**	**16.84**	**4.04**	**5.39**	**9.98**	**6.86**
k6	**1.80**	-1.80	**1.98**	**5.58**	**3.24**	**16.66**	-1.80	**1.98**	-5.40	**2.47**
k7	**14.03**	**26.37**	**23.79**	**1.67**	**10.01**	**31.77**	**10.01**	**7.43**	**37.43**	**18.06**
k8	-9.39	-36.67	-27.19	**4.03**	-10.41	-36.13	-20.31	-10.83	-17.87	-18.31
k9	**5.47**	**21.33**	-17.07	**1.75**	-1.55	-15.39	**39.77**	**10.73**	-26.07	**2.11**

k refers to *workstation*

m refers to *model*

5.5 Conclusion

In this chapter, we discussed an industry study case and used various approaches to exploit complexity and generate results. We completed our experiments with a set of nine workstations, nine models, and 109 tasks. Initially, we characterized workstation complexity using the models we described in Chapter 3. We obtained our best results on subjective classification and automatic complexity classification with the CALC_SAMPLE model.

Next, we generated results by means of the MILP model to obtain a solution to the MMALBP and minimize work overload. With this approach, we produced an optimal solution; however, we did not consider all of the constraints of the problem.

Finally, we focused on the heuristics developed. We discovered an evolution in our results generated by our mixed-line balancing heuristics when we concentrated on workstation overload and complexity levelling. We also provided a general overview of the initial complexity of the workstations and the results we obtained by our approaches.

6

Conclusion and Further Research

Producing a large number of models (and model variants) on an MMAL involves a high level of manufacturing complexity. We observed that this complexity is experienced differently in different phases of the production process. In the literature on manufacturing complexity and MMALBPs, we found that researchers have developed various approaches to both define and measure complexity and solve balancing problems. Nevertheless, the analysis, measurement and application of complexity in MMALs remains a new and challenging problem – especially when real datasets from the industry are taken into account.

This dissertation presents the first attempt to provide an empirical complexity analysis and classify workstations according to their complexity . It also proposes an entropic complexity measurement for balancing workloads and levelling complexity. However, manufacturing complexity is an extremely broad and complex problem. Although we have investigated complexity and delineated many important factors, it is still a subject that should be vigorously studied by future research teams.

6.1 Some concluding remarks

This section reviews the research questions presented in Chapter 1 and how they have been answered.

Research Question 1:

What are the drivers that determine manufacturing complexity in mixed-model assembly lines, and how can these drivers be used to classify workstations as being ‘high’ or ‘low’ in complexity?

To answer the first research question, we first conducted a literature review of manufacturing complexity. In Chapter 2, we presented a summary of the existing research on the definition and measurement of complexity. This led us to identify two key approaches to complexity characterization: one based on empirical analysis and another based on entropic measurement. To engage with and research real-life scenarios, we visited automotive plants and investigated real assembly lines. During these visits, information was gathered via our own observations, as well as through workshops with operators, team leaders and engineers. Chapter 3 describes how this information was retrieved and structured.

To classify workstations as high or low in complexity, an empirical analysis was conducted to extract a set of direct complexity drivers from real production lines. Five different models were constructed, as detailed in Chapter 3. Three of these were calculated as linear combinations of these variables and two as statistically-derived logistic models. We concluded that two models are of particular importance. The first is a logistic model that classifies workstations as high or low (LOGIT_SAMPLE), obtained from a sample of 54 stations with scores at the extremes of classification. This model is suitable to identify workstations that are high in complexity in engineering databases with only four variables as input data. The second interesting model is a calculated linear model (CALC_SAMPLE), with weights derived from the logit model mentioned above, which yields a good gradual scale of workstation complexity (between 0 and 10). In Chapter 4, these models are applied to a case study from the industry. The empirical analysis of complexity was extremely useful in our definition of a correlated entropic complexity measurement, which identified what makes workstations complex.

Research Question 2:

How can manufacturing complexity be levelled in mixed-model assembly lines workstations while balancing workload (minimizing work overload)?

A mixed integer linear programming model was developed to assign tasks to workstations. The objective was to optimize mixed-model assembly line balancing by minimizing workstation work overload. The mathematical model and its constraints are described in Chapter 4. While the MILP model generates a solution to the MMALBP, it does not take into account the existing complexity of the system. The results generated by our model provided a useful first step to solving this problem.

Drawing on this empirical complexity analysis, we developed a workstation complexity measurement that correlates the uncertainty of the system with the assignment of tasks. Subsequently, we implemented a set of heuristic procedures (cf. Chapter 4) to solve MMALBPs. These solutions were mainly based on a hybrid algorithm for generating line balance and a local search algorithm for rebalancing the line. Our main goal was to level manufacturing complexity while balancing workload.

Research Question 3:

What are the results and shortcomings of both approaches when applied to a real world mixed-model assembly line? Analysis of an industry study case.

In Chapter 5, we reported on our application of our proposed solutions to an industry case study. The mixed-line balancing heuristics generated a solution in which the majority of the workstations were well balanced. A general overview was provided of the initial complexity measurements of the workstations and the results obtained with our approaches. A number of different solutions were effective but our rebalancing solution ultimately produced the best output.

General Research Question:

How can manufacturing complexity be evaluated and managed in mixed-model assembly lines?

The research provided deep insight into manufacturing complexity, especially with regard to real-world workstations on MM- ALs. We identified various aspects of complexity and developed an empirical analysis and an entropic measurement. Drawing on these characterizations, we exploited system complexity and introduced statistical models that were able to classify workstations as high or low in complexity. We also proposed a solution to level manufacturing complexity. Using typical artificial intelligence methodologies, we developed heuristics that combined algorithms to generate an initial solution to the MMALBP – and a local algorithm that provided a rebalancing solution.

6.2 Future Research

Our research has contributed to the relevant literature in this field by exploiting manufacturing complexity. However, the optimization of manufacturing complexity is a very comprehensive topic, so a number of issues remain unexplored and require further research. In this section, an overview will be given of potential research avenues.

With regard to the causal model that characterizes manufacturing complexity, future researchers should look into the impact of existing complexity on direct and indirect costs, the subjective interpretations of complexity by workstation operators, the quality and number of errors generated by complexity, and overall sales. Since complexity is driven by a number of different variables, it can impact on multiple areas of manufacturing.

MMALBPs should be further analysed in different ways. Whereas our objective was to minimize workstation overload, other possible objectives include minimizing workstation complexity level, minimizing simultaneous work overload and complexity, and/or limiting one of these variables. In this respect, other datasets should also be used in the computational tests.

Another issue closely related to MMALBPs is the sequencing of models on a line. According to Boysen et al. (2009), Assembly Line Sequencing Problems (ALSPs) involve the sequencing of different models to avoid work workstation overload. There are two main approaches to ALSPs: (i) minimizing inefficient line stoppage, and (ii) developing procedures to avoid overload. Various factors can be investigated and optimized, such as workstation length, task time, cycle time, human factor policies Celano et al. (2004) and many others.

Since assembly line sequencing problems are directly related to MMALBPs, future researchers should focus on exploiting workstation complexity by analysing the differences between models to efficiently sequence products. Because model sequencing requires a detailed manufacturing schedule, it is of vital importance that any overloads are minimized that might occur after balance has been achieved. Measurements of manufacturing complexity can therefore be taken into account not only to improve mixed-model assembly line balancing but also to facilitate sequencing.

Future researchers should also focus on designing and developing a robust Decision Support System (DSS) for assembly line balancing. A DSS is an information system that supports the decision-making (Keen (1980)) process. As the MMALBP involves both large amounts of manufacturing data and practical subjective knowledge, the DSS could be both data- and knowledge-driven. Techniques such as data mining and predictive analytics could be used to extend the solutions presented in this dissertation and extract more useful information from the data retrieved from our manufacturer collaborators. In the next phase of research, new heuristics and known metaheuristics could be added and combined to generate accurate solutions to real problems.

In short, the application of operations research to information systems is a wide, challenging topic in need of much further exploration.

References

Abad, A. G. and Jin, J. (2011). Complexity metrics for mixed model manufacturing systems based on information entropy. *International Journal of Information and Decision Sciences*, 3(4):313–334.

Akpınar, S., Bayhan, G. M., and Baykasoglu, A. (2013). Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks. *Applied Soft Computing*, 13(1):574–589.

Battaïa, O. and Dolgui, A. (2012). Reduction approaches for a generalized line balancing problem. *Computers & Operations Research*, 39(10):2337–2345.

Battaïa, O. and Dolgui, A. (2013). A taxonomy of line balancing problems and their solutionapproaches. *International Journal of Production Economics*, 142(2):259–277.

Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management science*, 32(8):909–932.

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

Boctor, F. F. (1995). A multiple-rule heuristic for assembly line balancing. *Journal of the Operational Research Society*, pages 62–69.

Boysen, N., Fliedner, M., and Scholl, A. (2009). Production planning of mixed-model assembly lines: overview and extensions. *Production Planning and Control*, 20(5):455–471.

Braaksma, A., Meesters, A., Klingenberg, W., and Hicks, C. (2012). A quantitative method for failure mode and effects analysis. *International journal of production research*, 50(23):6904–6917.

Bukchin, J., Dar-El, E. M., and Rubinovitz, J. (2002). Mixed model assembly line design in a make-to-order environment. *Computers & Industrial Engineering*, 41(4):405–421.

Celano, G., Costa, A., Fichera, S., and Perrone, G. (2004). Human factor policy testing in the sequencing of manual mixed model assembly lines. *Computers & Operations Research*, 31(1):39–59.

Deshmukh, A. V., Talavage, J. J., and Barash, M. M. (1998). Complexity in manufacturing systems, part 1: Analysis of static complexity. *IIE transactions*, 30(7):645–655.

ElMaraghy, W., ElMaraghy, H., Tomiyama, T., and Monostori, L. (2012). Complexity in engineering design and manufacturing. *CIRP Annals-Manufacturing Technology*, 61(2):793–814.

ElMaraghy, W. and Urbanic, R. (2003). Modelling of manufacturing systems complexity. *CIRP Annals-Manufacturing Technology*, 52(1):363–366.

Erel, E., Sabuncuoglu, I., and Aksu, B. (2001). Balancing of u-type assembly systems using simulated annealing. *International Journal of Production Research*, 39(13):3003–3015.

Essafi, M., Delorme, X., and Dolgui, A. (2012). A reactive grasp and path relinking for balancing reconfigurable transfer lines. *International Journal of Production Research*, 50(18):5213–5238.

Falck, A.-C., Örtengren, R., and Rosenqvist, M. (2012). Relationship between complexity in manual assembly work, ergonomics and assembly quality. In *Ergonomics for Sustainability and Growth, NES 2012 (Nordiska Ergonomisällskapet) konferens, Saltsjöbaden, Stockholm, 19-22 augusti, 2012*, number Book of abstracts, Abstract B4: 2.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Frizelle, G. (1996). Getting the measure of complexity. *Manufacturing Engineer*, 75(6):268–270.

Frizelle, G. and Woodcock, E. (1995). Measuring complexity as an aid to developing operational strategy. *International Journal of Operations & Production Management*, 15(5):26–39.

Fujimoto, H. and Ahmed, A. (2001). Entropic evaluation of assemblability in concurrent approach to assembly planning. In *Assembly and Task Planning, 2001, Proceedings of the IEEE International Symposium on*, pages 306–311. IEEE.

Gonçalves, J. F. and De Almeida, J. R. (2002). A hybrid genetic algorithm for assembly line balancing. *Journal of Heuristics*, 8(6):629–642.

Gullander, P., Davidsson, A., Dencker, K., Fasth, Å., Fässberg, T., Harlin, U., and Stahre, J. (2011). Towards a production complexity model that supports operation, re-balancing and man-hour planning. In *Proceedings of the 4th Swedish Production Symposium (SPS): Lund, Sweden*.

Hackman, S. T., Magazine, M. J., and Wee, T. (1989). Fast, effective algorithms for simple assembly line balancing problems. *Operations Research*, 37(6):916–924.

Haq, A. N., Rengarajan, K., and Jayaprakash, J. (2006). A hybrid genetic algorithm approach to mixed-model assembly line balancing. *The International Journal of Advanced Manufacturing Technology*, 28(3-4):337–341.

Helgeson, W. B. and Birnie, D. P. (1961). Assembly line balancing using the ranked positional weight technique. *Journal of Industrial Enginering*, 12:394–398.

Hoffmann, T. R. (1963). Assembly line balancing with a precedence matrix. *Management Science*, 9(4):551–562.

Hu, S., Zhu, X., Wang, H., and Koren, Y. (2008). Product variety and manufacturing complexity in assembly systems and supply chains. *CIRP Annals-Manufacturing Technology*, 57(1):45–48.

Jackson, J. R. (1956). A computing procedure for a line balancing problem. *Management Science*, 2(3):261–271.

Keen, P. G. (1980). Decision support systems: a research perspective. *Decision Support Systems: Issues and Challenges (New York: Pergamon Press, 1980)*, pages 23–44.

Kim, Y. K., Kim, Y. J., and Kim, Y. (1996). Genetic algorithms for assembly line balancing with various objectives. *Computers & Industrial Engineering*, 30(3):397–409.

Kim, Y.-S. (1999). *A system complexity approach for the integration of product development and production system design*. PhD thesis, Massachusetts Institute of Technology.

Kirkpatrick, S., Vecchi, M., et al. (1983). Optimization by simmulated annealing. *science*, 220(4598):671–680.

MacDuffie, J. P., Sethuraman, K., and Fisher, M. L. (1996). Product variety and manufacturing performance: evidence from the international automotive assembly plant study. *Management Science*, 42(3):350–369.

Matanachai, S. and Yano, C. A. (2001). Balancing mixed-model assembly lines to reduce work overload. *IIE transactions*, 33(1):29–42.

Mattsson, S., Karlsson, M., Gullander, P., Van Landeghem, H., Zeltzer, L., Limère, V., Aghezzaf, E.-H., Fasth, Å., and Stahre, J. (2014). Comparing quantifiable methods to measure complexity in assembly. *International Journal of Manufacturing Research*, 9(1):112–130.

McGovern, S. M. and Gupta, S. M. (2003). Greedy algorithm for disassembly line scheduling. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 2, pages 1737–1744. IEEE.

McMullen, P. R. and Frazier, G. (1998). Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations. *International Journal of Production Research*, 36(10):2717–2741.

McMullen, P. R. and Tarasewich, P. (2003). Using ant techniques to solve the assembly line balancing problem. *IIE transactions*, 35(7):605–617.

Merengo, C., Nava, F., and Pozzetti, A. (1999). Balancing and sequencing manual mixed-model assembly lines. *International Journal of Production Research*, 37(12):2835–2860.

Meyer, M. H. and Curley, K. F. (1995). The impact of knowledge and technology complexity on information systems development. *Expert Systems with Applications*, 8(1):111–134.

Özcan, U., Çerçioğlu, H., Gökçen, H., and Toklu, B. (2010). Balancing and sequencing of parallel mixed-model assembly lines. *International Journal of Production Research*, 48(17):5089–5113.

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *Systems, Man and Cybernetics, IEEE Transactions on*, (3):257–266.

Reiter, S. and Sherman, G. (1965). Discrete optimizing. *Journal of the Society for Industrial & Applied Mathematics*, 13(3):864–889.

Rosenberg, O. and Ziegler, H. (1993). A comparison of heuristic algorithms for cost-oriented assembly line balancing. *Zeitschrift für Operations Research*, 36(6):477–495.

Salveson, M. E. (1955). The assembly line balancing problem. *Journal of Industrial Engineering*, 6(3):18–25.

Scholl, A. (1993). Data of assembly line balancing problems. schriften zur quantitativen betriebswirtschaftslehre. Technical report.

Scholl, A. (1999). *Balancing and Sequencing of Assembly lines*. Contributions to Management Science. Physica-Verlag Heidelberg, second edition.

Scholl, A. (2007). Data sets for salbp. `/http://www.assembly-line-balancing.de`. Accessed: 2015-08-07.

Schuh, G., Monostori, L., Csáji, B. C., and Döring, S. (2008). Complexity-based modeling of reconfigurable collaborations in production industry. *CIRP Annals-Manufacturing Technology*, 57(1):445–450.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423.

Simaria, A. S. (2006). *Assembly line balancing - new perspectives and procedures*. PhD thesis, Universidade de Aveiro.

Simaria, A. S. and Vilarinho, P. M. (2009). 2-antbal: An ant colony optimisation algorithm for balancing two-sided assembly lines. *Computers & Industrial Engineering*, 56(2):489–506.

Sivadasan, S., Efstathiou, J., Calinescu, A., and Huatuco, L. H. (2006). Advances on measuring the operational complexity of supplier–customer systems. *European Journal of Operational Research*, 171(1):208–226.

Sivasankaran, P. and Shahabudeen, P. (2014). Literature review of assembly line balancing problems. *The International Journal of Advanced Manufacturing Technology*, 73(9-12):1665–1694.

Talbot, F. B., Patterson, J. H., and Gehrlein, W. V. (1986). A comparative evaluation of heuristic line balancing techniques. *Management science*, 32(4):430–454.

Thomopoulos, N. T. (1970). Mixed model line balancing with smoothed station assignments. *Management Science*, 16(9):593–603.

Tiacci, L. (2015a). Coupling a genetic algorithm approach and a discrete event simulator to design mixed-model un-paced assembly lines with parallel workstations and stochastic task times. *International Journal of Production Economics*, 159:319–333.

Tiacci, L. (2015b). Data sets for mmalbp. `/http://www.assembly-line-balancing.de`. Accessed: 2015-08-07.

Uddin, M. K. and Lastra, J. L. M. (2011). *Assembly line balancing and sequencing*. INTECH Open Access Publisher.

Urbanic, R. J. and ElMaraghy, W. H. (2006). Modeling of manufacturing process complexity. In *Advances in Design*, pages 425–436. Springer.

Wee, T. and Magazine, M. (1981). An efficient branch and bound algorithm for an assembly line balancing problem. part ii: maximize the production rate. In *Working paper no. 151*. Dept. of Management Sciences, University of Waterloo.

Wiendahl, H.-P. and Scholtissek, P. (1994). Management and control of complexity in manufacturing. *CIRP Annals-Manufacturing Technology*, 43(2):533–540.

Zeltzer, L., Limère, V., Van Landeghem, H., Aghezzaf, E.-H., and Stahre, J. (2013). Measuring complexity in mixed-model assembly workstations. *International Journal of Production Research*, 51(15):4630–4643.

Zhu, X., Hu, S. J., Koren, Y., and Marin, S. P. (2008). Modeling of manufacturing complexity in mixed-model assembly lines. *Journal of Manufacturing Science and Engineering*, 130(5):051013.

List of Publications

Journal Articles

Zeltzer, L., Aghezzaf, E-H., Limère, V., 2015. Workload Balancing and Manufacturing Complexity Levelling in Mixed-Model Assembly Lines. International Journal of Production Research.

Mattsson, S., Karlsson, M., Gullander,P., Van Landeghem,H., **Zeltzer, L.**, Limère, V., Aghezzaf, E-H., Fasth, A., Stahre, J., 2014. Comparing quantifiable methods to measure complexity in assembly. International Journal of Manufacturing Research
DOI: 10.1504/IJMR.2014.059602

Zeltzer, L., Limère, V., Van Landeghem,H., Aghezzaf, E.-H, Stahre, J., 2013. Measuring Complexity in Mixed-Model Assembly Workstations. International Journal of Production Research.
DOI: 10.1080/00207543.2013.783246

Papers at International Conferences

Zeltzer, L., Limère, V., Aghezzaf, E-H., 2013. Balancing and Sequencing Mixed-Model Assembly Lines to Minimize Work Overload. Multi-disciplinary International Scheduling Conference 2013 (MISTA). Ghent, Belgium.

Zeltzer, L., Limère, V., Aghezzaf, E-H., Van Landeghem,H., 2013. Balancing Mixed-Model Assembly Lines in Real World Complex Workstations.7th IFAC Conference on Manufacturing Modelling, Management and Control (MIM). St.Petersburg, Russia.

Gullander, P., Mattsson, S., Fässberg, T., Van Landeghem, H., **Zeltzer, L.**, Limère, V., Aghezzaf, E-H., Stahre, J., 2012. Comparing Two Methods to Measure Assembly Complexity from an Operator Perspective. 5th Swedish Production Symposium. Linköping, Sweden

Zeltzer, L., Limère, V., Aghezzaf, E-H., Van Landeghem, H., 2012. 7th International Multi-Conference on Computing in the Global Information Technology. Venice, Italy.

De Bruyn, W., Borodin, D., **Zeltzer, L.,**, Van Vreckem, B., 2010. Key performance indicators : linking with ISA-95 and moving toward KPI-driven factory. 11th International scientific and practical conference : innovations, ICT technologies and their application in education. 2010, Borisoglebsk, Russia.

Meeting abstracts, presented at national conference

Zeltzer, L., Limère, V., Aghezzaf, E-H., Van Landeghem,H., 2013. Measuring complexity in mixed-model assembly workstations. 27th annual conference of the Belgian Operations Research Society (Orbel 27). Kortrijk, Belgium.

Poster

Shorten New Product Development and Introduction in Bio (pharmaceutical) Industries
Zeltzer, L., Caluwaerts, P., Borodin, D., Van Vreckem, B., De Bruyn, W., 2010. Shorten New Product Development and Introduction in Bio (pharmaceutical) Industries. FlandersBio, Knowledge for Growth Convention. Ghent, Belgium.