2016

Variant-oriented Planning Models for Parts/Products Grouping, Sequencing and Operations

Javad Navaei
University of Windsor

Variant-oriented Planning Models for Parts/Products Grouping, Sequencing and Operations

By

Javad Navaei

A Dissertation
Submitted to the Faculty of Graduate Studies
through the Department of Mechanical, Automotive & Materials Engineering
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy
at the University of Windsor

Windsor, Ontario, Canada

2016

Variant-oriented Planning Models for Parts/Products Grouping, Sequencing and Operations

By

Javad Navaei

APPROVED BY:

____

Dr. Onur Kuzgunkaya
Department of Mechanical and Industrial Engineering, Concordia University

____

Dr. Richard Caron
Mathematics & Statistics

____

Dr. Waguih ElMaraghy
Mechanical, Automotive & Materials Engineering

____

Dr. Zbigniew Pasek
Mechanical, Automotive & Materials Engineering

____

Dr. Hoda ElMaraghy, Advisor
Mechanical, Automotive & Materials Engineering

September 6, 2016

DECLARATION OF CO-AUTHORSHIP / PREVIOUS PUBLICATION

I. Co-Authorship Declaration

I hereby declare that this dissertation incorporates material that is result of joint research of the author and his supervisor Prof. Hoda ElMaraghy. This joint research has been published / submitted to various journals that are listed below.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my dissertation, and have obtained written permission from Prof. Hoda ElMaraghy to include that material(s) in my dissertation.

I certify that, with the above qualification, this dissertation, and the research to which it refers, is the product of my own work.

II. Declaration of Previous Publication

This dissertation includes 6 original papers that have been previously published / submitted for publication in peer reviewed journals and conferences as follows:

Dissertation Chapter	Publication title/full citation
2	Navaei, J. and ElMaraghy, H., 2016. Grouping and sequencing product variants based on setup similarity, *International Journal of Computer Integrated Manufacturing (IJCIM)*, DOI: 10.1080 / 0951192X. 2016. 1190464 (in press).
3	Navaei, J. and ElMaraghy, H., 2016. Grouping part/product variants based on networked operations sequence, *Journal of Manufacturing Systems (JMS)*, 38, 63-76.
3	Navaei, J. and ElMaraghy, H., 2014. Grouping product variants based on alternate machines for each operation. In proceeding of the 47th CIRP Conference on Manufacturing Systems (CMS), Windsor, ON, Canada, Procedia CIRP, 17, 61-66.
4	Navaei, J. and ElMaraghy, H., 2016. Optimal assignments of facilities in backtracking of product variants with networked operations sequence,

	CIRP Journal of Manufacturing Science & Technology (Submitted 6 July 2016, CIRPJ-D-16-00093).
5	Navaei, J. and ElMaraghy, H., 2016. Mathematical models for generating master operations sequence, *Expert Systems with Applications (ESWA)* (Submitted 8 August 2016, ESWA-D-16-03020).
5	Navaei, J. and ElMaraghy, H., 2016. Operations sequence retrieval by construction of master operations sequence using a mathematical model and a novel algorithm, *European Journal of Operational Research* (To be submitted).

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my dissertation. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I certify that, to the best of my knowledge, my dissertation does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my dissertation, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my dissertation.

I declare that this is a true copy of my dissertation, including any final revisions, as approved by my dissertation committee and the Graduate Studies office, and that this dissertation has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

This research aims at developing novel methods for utilizing the commonality between part/product variants to make modern manufacturing systems more flexible, adaptable, and agile for dealing with less volume per variant and minimizing total changes in the setup between variants. Four models are developed for use in four important domains of manufacturing systems: production sequencing, product family formation, production flow, and products operations sequences retrieval. In all these domains, capitalizing on commonality between the part/product variants has a pivotal role.

For production sequencing; a new policy based on setup similarity between product variants is proposed and its results are compared with a developed mathematical model in a permutation flow shop. The results show the proposed algorithm is capable of finding solutions in less than 0.02 seconds with an average error of 1.2%. For product family formation; a novel operation flow based similarity coefficient is developed for variants having networked structures and integrated with two other similarity coefficients, operation and volume similarity, to provide a more comprehensive similarity coefficient. Grouping variants based on the proposed integrated similarity coefficient improves changeover time and utilization of the system. A sequencing method, as a secondary application of this approach, is also developed. For production flow; a new mixed integer programing (MIP) model is developed to assign operations of a family of product variants to candidate machines and also to select the best place for each machine among the candidate locations. The final sequence of performing operations for each variant having networked structures is also determined. The objective is to minimize the total backtracking distance leading to an improvement in total throughput of the system (7.79% in the case study of three engine blocks). For operations sequences retrieval; two mathematical models and an algorithm are developed to construct a master operation sequence from the information of the existing variants belonging to a family of parts/products. This master operation sequence is used to develop the operation sequences for new variants which are sufficiently similar to existing variants. Using the proposed algorithm decreases time of developing the operations sequences of new variants to the seconds.

DEDICATION

To my Mother and Father,

For their affection, love, encouragement and prays

To my Wife,

For dedicating her life to be with me in all my happy and sad moments

ACKNOWLEDGEMENT

The author would like to express his gratitude to his supervisor, Professor Hoda ElMaraghy, for her support, guidance, and inspiration and also for her time and effort dedication. Her significant contribution throughout all phases of this research is greatly appreciated. The author also thanks the committee members for their constructive comments leading to major improvement in his thesis. Special thanks go to Dr. Waguih ElMaraghy for his great discussion during the periodic academic meetings.

The author also acknowledges the help and support by his friend, Mostafa Pazoki, and also his current and former colleagues at the Intelligent Manufacturing Systems (IMS) Centre; many thanks to Dr. Mohamed Kashkoush for his great help and guidance, and also Kourosh Khedri and Mostafa Moussa.

Finally, the author would like to appreciate his family for their endless support and love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
AHP	Analytic Hierarchy Process
ALC	Average Linkage Clustering
ASM	Assembling
BOM	Bill of Materials
DML	Dedicated Manufacturing Lines
FMS	Flexible Manufacturing Systems
GA	Genetic Algorithm
GAMS	General Algebraic Modeling System
GBOM	Generic Bill of Materials
GFL	Generalized Flow Line
GT	Group Technology
IDEF0	Integrated Definition for Function Modeling
MCDA	Multi-Criteria Decision Analysis
MIP	Mixed Integer Programing
MILP	Mixed Integer Linear Programing
PCB	Printed Circuit Boards
PSO	Particle Swarm Optimization
RMS	Reconfigurable Manufacturing Systems
SA	Simulated Annealing
SPT	Shortest Processing Time
Sub-ASSY	Sub-Assembly
TS	Tabu Search

NOMENCLATURE

Production Sequencing Model (Chapter 2):

n Number of product variants.

k Number of work stations.

S_{irp} Setup change time/cost for product i if processed immediately after product r at station p.

W_{ip} is equal to one if product i requires an operation at work station p and zero otherwise.

X_{ij} is equal to one if product i is placed in j^{th} position of the sequence and zero otherwise.

Z_{jmp} is equal to one if, at station p, product in position j is processed immediately after product in position m and zero otherwise.

Q_{ijrmp} is equal to one if product i is placed in position j and is processed immediately after product r which is placed in position m of the sequence and it is zero otherwise.

Product Family Formation Model (Chapter 3):

n Total numbers of positions/variants.

SM_{ij} Integrated similarity coefficient value between variants i and j.

X_{ij} is equal to one if variant j is placed immediately after variant i in the sequence and zero otherwise.

Y_{ik} is equal to one if variant i is placed in k^{th} position of the sequence and zero otherwise.

Z_{ijk} is equal to one if variant i is placed in k^{th} position immediately before variant j in the sequence and zero otherwise.

Production Flow Model (Chapter 4):

S Total number of different operations among all variants.

n_r Total number of operations for part r.

m Total number of machines/locations.

q Total number of part/product variants.

M A large positive number.

V_r Volume required for part/product r.

C_{ki} is equal to one if machine k is capable of performing operation i and zero otherwise.

P_{ri} is equal to one if product r requires operation i and zero otherwise.

O_{ijr} is equal to one if operation j is performed after operation i in product r and zero otherwise.

$d_{hh'}$ backtracking distance from location h to location h'.

L_{hk} is equal to one if machine k is assigned to location h and zero otherwise.

X_{ik} is equal to one if operation i is assigned to machine k and zero otherwise.

Y_{ibr} is equal to one if operation i is performed in b^{th} position of the final operations sequence of product r and zero otherwise.

$Z_{ijbrkhk'h'}$ is equal to one if operations i and j are placed respectively in positions b and b+1 of product r and also operations i and j are performed with machines k and k'

respectively and machines k and k' are assigned to locations h and h' and it is zero otherwise.

Products Operations Sequences Retrieval Model (Chapter 5):

n Total number of operations among all variants.

p Total number of part variants.

A_{ki} is equal to one if operation i exists in k^{th} part variant and zero otherwise.

B_{ijk} is equal to one if operation j of k^{th} part variant is performed immediately after operation i and zero otherwise.

X_{ij} is equal to one if operation j of the master is performed immediately after operation i and zero otherwise.

W_{ig} is equal to one if operation i is placed in g^{th} position of the master and zero otherwise.

S_{ijg} is equal to one if operation i is placed in g^{th} position immediately before operation j in master and zero otherwise.

Chapter 1

Introduction

1.1 Motivation

Today, many modern manufacturing systems are shifting from mass production to mass customization (Fogliatto et al., 2012; Tseng and Hu, 2014; Daie and Li, 2016). Some of the reasons for this shift are changing demands, global competition and new technologies. Customers frequently request new types of products and if a manufacturer is not able to meet these new demands, other competitors would satisfy them leading to potential loss of market share. Therefore, manufacturing systems should be sufficiently flexible and adapt to frequently changing demands and this may result in significant costs for manufacturers if not managed well. “Mass Customization” describes this issue very well. “Customization” refers to the fact that manufacturing systems should be able to produce different variants of products according to customers’ requirements with usually low to medium production volume (Piller, 2004; Yang et al., 2007; Tseng et al., 2010; Fogliatto et al., 2012) and “Mass” means that the system, by using modern techniques such as delayed product differentiation (AlGeddawy and ElMaraghy, 2009; AlGeddawy and ElMaraghy, 2010), should try to combine the cost efficiency of mass production while satisfying the wide scope of variety demanded by customers (ElMaraghy et al., 2013).

One of the best strategies in decreasing production cost in modern manufacturing systems is utilizing the commonality between part/product variants. The commonality between part/product variants refers to different similarity aspects including their setup similarity, operation flow based similarity, operation similarity, production volume similarity and so on. This research is in fact motivated by the vital needs for developing practical models and methodologies using the commonality between part/product variants to help designers and planners to efficiently decrease changeover time and improve productivity, throughput and utilization in manufacturing systems. Despite the fact that many research works in literature have focused on this issue, there are still significant gaps in literature in terms of practical and efficient methods and a need for developing novel methods in this regard.

1.2 Research Scope

In this sub-section, the scope of the research is addressed under different categories as follows:

System Type

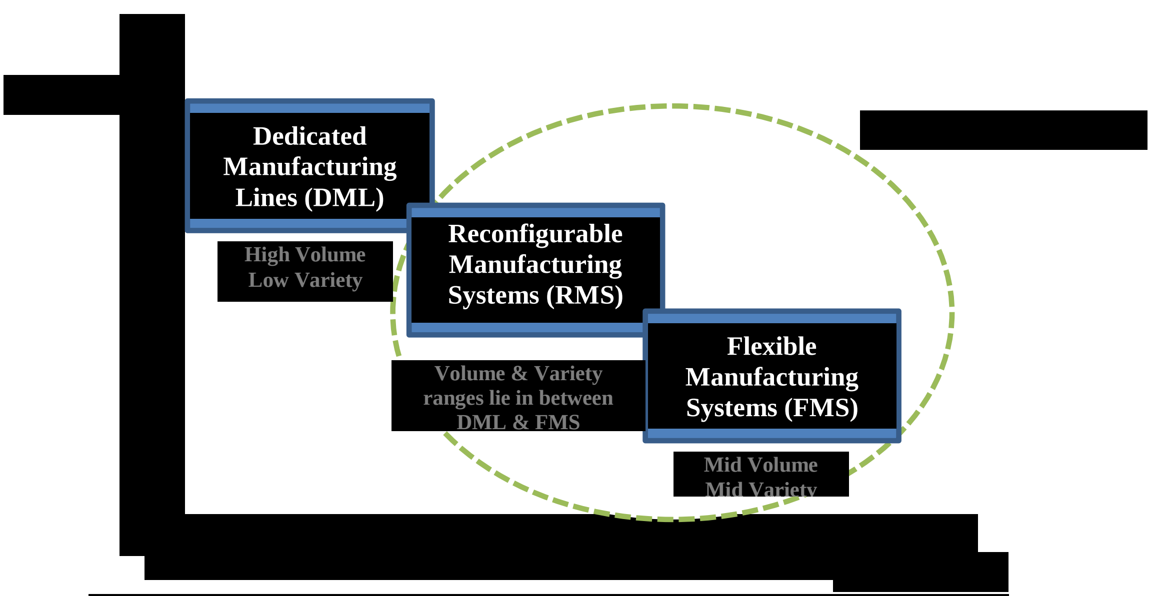
This research encompasses different types and aspects of manufacturing systems. The problems described in this research are applicable in different manufacturing systems such as flow shop, job shop, cellular, flexible and reconfigurable manufacturing systems.

Process Type

The considered part/product variants can have different types of processes from fabrication to assembly or combination of both.

Production Volume Type

This research is mainly targeting mass customization and personalization and, hence, the considered parts/ products usually require low to medium production volumes. Figure 1.1 shows the scope of the research in terms of variety and volume.



[redacted]

Setup Type

The type of the setup changes considered in this research includes tooling, fixtures, and machines CNC programs.

Complexity Type

Manufacturing systems consist of various tools, machines, parts/products and operators. These factors usually contribute to the complexity of a system (Badrous, 2011). In the present work, the complexity of the system corresponds to the variety part/product variants that should be produced. This research aims at managing this type of complexity or in other words, managing part/product variety produced by the system.

Product Type

The potential part/product families for which this research is applicable are quite wide: mechanical parts/products (e.g. engine blocks, valves, chairs, cutting tools, pressing tools, etc.), electronic parts/products (e.g. personal computers, tablets, printed circuit boards (PCBs)), or other part/product families such as label stickers, storage containers, bottle caps and so on.

Model Type

The types of the developed models in this research are within process planning and scheduling domains. Process planning is a link between design and manufacturing and is classified in three different levels: 1) Multi-domain process planning 2) Macro-process planning and 3) Micro-process planning (ElMaraghy, et al., 2013). Scheduling is about assigning resources to some tasks over a period of time to optimize one or more objectives (Pinedo, 2012). In the present work, resources refer to machines and stations and tasks are processes to be performed on the part/product variants.

1.3 Research Gaps and Novelty

This sub-section briefly highlights the research gaps in literature with respect to each of the aforementioned applications and explains the novelty of each section.

This research includes four parts. The first part of the research focuses on permutation flow shop environment. There are huge number of works in literature regarding sequencing product variants in a permutation flow shop (e.g. Toktas et al., 2004; Tseng et al., 2004; Lin and Wu, 2006; Khan and Govindan, 2011; M'Hallah, 2014; Zhao et al., 2015; Wang et al., 2015; He, 2016; Rafai et al., 2016). However, most of these works only focus on mathematical modeling or meta-heuristic methods which are not practical and applicable in real manufacturing systems. There is dearth of literature on using the similarity between variants for sequencing their production steps. Table 1.1 summarizes the research gaps by considering some of the important works in literature.

In this research, a new and practical sequencing policy is proposed which is based on the setup similarity between variants and it aims at minimizing the setup changes required between product variants.

Table 1.1 Research gap in permutation flow shop

No.	Reference	Year	Approach	Setup similarity included?
1	Toktas et al.	2004	branch and bound	N
2	Tseng et al.	2004	MIP	N
3	Lin and Wu	2006	branch and bound	N
4	Khan and Govindan	2011	simulated annealing	N
5	M'Hallah	2014	MIP	N
6	Zhao et al.	2015	evolutionary algorithm	N
7	Wang et al.	2015	branch and bound	N
8	He	2016	branch and bound and heuristic	N
9	Rafai et al.	2016	Adaptive large neighborhood search	N

The second part of the research is grouping part/product variants based on networked operation sequences such as those found in locomotive and automotive industries. Many part/product similarity coefficients have been proposed in literature with respect to different criteria such as operations similarity, operations flow similarity, and volume similarity(e.g. Tam, 1990; Choobineh, 1988; Galan et al., 2007b ; Pattanaik and Kumar, 2010; Goyal et al., 2013; Alhourani, 2013; Wu and Suzuki, 2015; Won and Logendran, 2015; Raja and Anbumalar, 2016). However, there is no work in literature proposing a similarity coefficient for variants having networked operation sequences while there are many practical examples in which the operations of a variant have flexibility to be done before, after or at the same time as other operations which is known as sequencing flexibility. Table 1.2 shows the research gaps regarding operation flow based similarity for product variants having networked structure.

In this research, a novel similarity coefficient for part/product variants is developed based on the networked operations sequence similarity inspired by the analysis used in the field of biology (e.g. enzymes structures comparison). In order to have a more comprehensive similarity coefficient, two other important criteria in grouping part/product families namely operation similarity and volume similarity are also included.

Table 1.2 Research gap in similarity coefficient for variants with networked structure

No.	Reference	Year	Operation similarity	Operation flow based similarity (serial)	Operation flow based similarity (networked)	Volume similarity
1	Choobineh	1988	N	Y	N	N
2	Tam	1990	N	Y	N	N
3	Galan et al.	2007	Y	N	N	Y
4	Pattanaik and Kumar	2010	Y	N	N	Y
5	Goyal et al.	2013	N	Y	N	N
6	Alhourani	2013	N	Y	N	Y
7	Wu and Suzuki	2015	N	Y	N	N
8	Raja and Anbumalar	2016	Y	N	N	N

After grouping product variants based on their similarity, in the third part of the research, processing tasks are assigned to facilities (e.g. machines, etc.) in a way that production flow is optimized. The objective is to minimize the total backtracking distance by considering the production volume of each variant. In literature, backtracking problem has been addressed in two different categories namely generalized flow line (GFL) problems and cellular manufacturing systems. Nevertheless, there is no work in literature on backtracking minimization considering product variants with networked operations sequence and also, machine selection for each operation (Sarker et al., 1991; Sarker et al., 1995; Gong et al., 1999; Chang et al., 2013; Forghani et al., 2015; Golmohammadi et al., 2016). Table 1.3 shows the research gap in literature of backtracking minimization.

In this research, a novel mixed integer programing (MIP) model is developed to optimize three decision variables: (1) to assign various operations of product variants to candidate machines, (2) to assign the machines to candidate locations, and (3) to determine the final sequence of performing operations for variants having networked structures.

Table 1.3 Research gap in backtracking minimization

No.	Reference	Year	Category	Location assignment to machines	Machine assignment to operations	Networked operations sequence
1	Sarker et al.	1991	GFL	Y	N	N
2	Sarker et al.	1995	GFL	Y	N	N
3	Gong et al.	1999	GFL	Y	N	N
4	Chang et al.	2013	Cellular manufacturing	Y	N	N
5	Forghani et al.	2015	Cellular manufacturing	Y	N	N
6	Golmohammadi et al.	2016	Cellular manufacturing	Y	N	N

The last part of the research focuses on data mining and using the information of existing variants for developing new variants. There is no work in literature for constructing a master operations sequence for use in developing the operations sequence of new variants. In fact, Most of the works in the literature focus on generic bills of material (e.g. Hegge and Wortmann, 1991; Romanowski and Nagi, 2004; Shu et al., 2014). There are also some works such as Kashkoush and ElMaraghy (2014, 2015) which consider master assembly sequence but were not applied to manufacturing/fabrication. Table 1.4 shows the research gap regarding the construction of master operation sequence in literature.

In this research, two new mathematical models are developed and a novel algorithm is proposed to generate master operation sequence for variants having serial, networked operation sequences or combinations of both. The operation term in this work refers to both assembly and fabrication processes.

Table 1.4 Research gap in generating master operation sequence

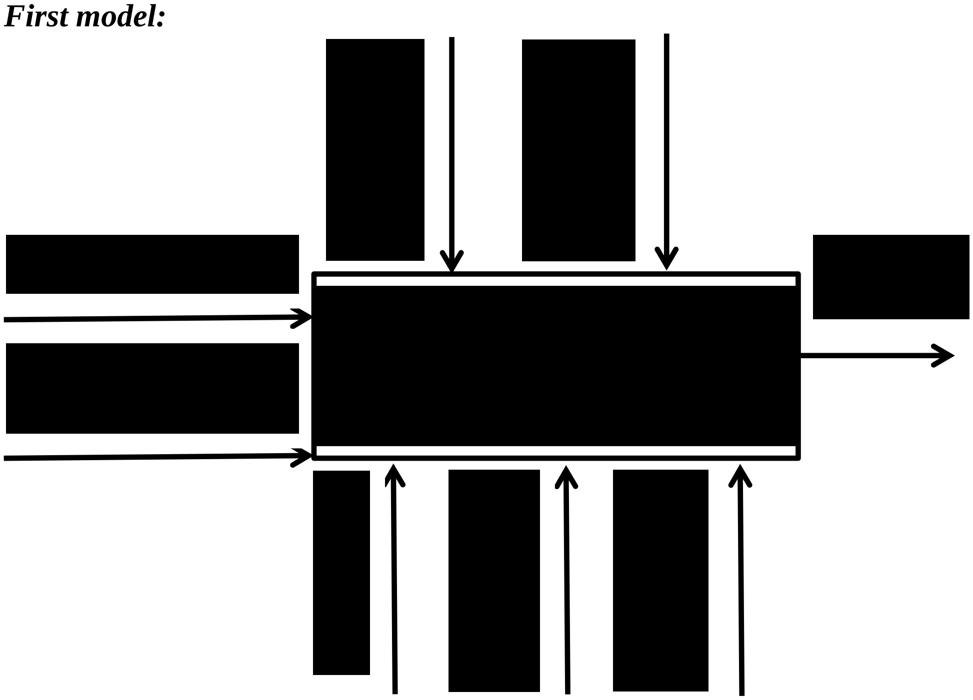
No.	Reference	Year	Generating	Approach	Application
1	Hegge and Wortmann	1991	generic bill of material	code number	assembly
2	Romanowski and Nagi	2004	generic bill of material	data mining	assembly
3	Shu et al.	2014	generic bill of material	simulation and neural network	assembly
4	Kashkoush and ElMaraghy	2014	master assembly sequence	genetic algorithm	assembly
5	Kashkoush and ElMaraghy	2015	master assembly sequence	MIP	assembly

The developed mathematical models and proposed algorithms in this research can be applied in design and operational stages. They can be used for both existing and new product variants.

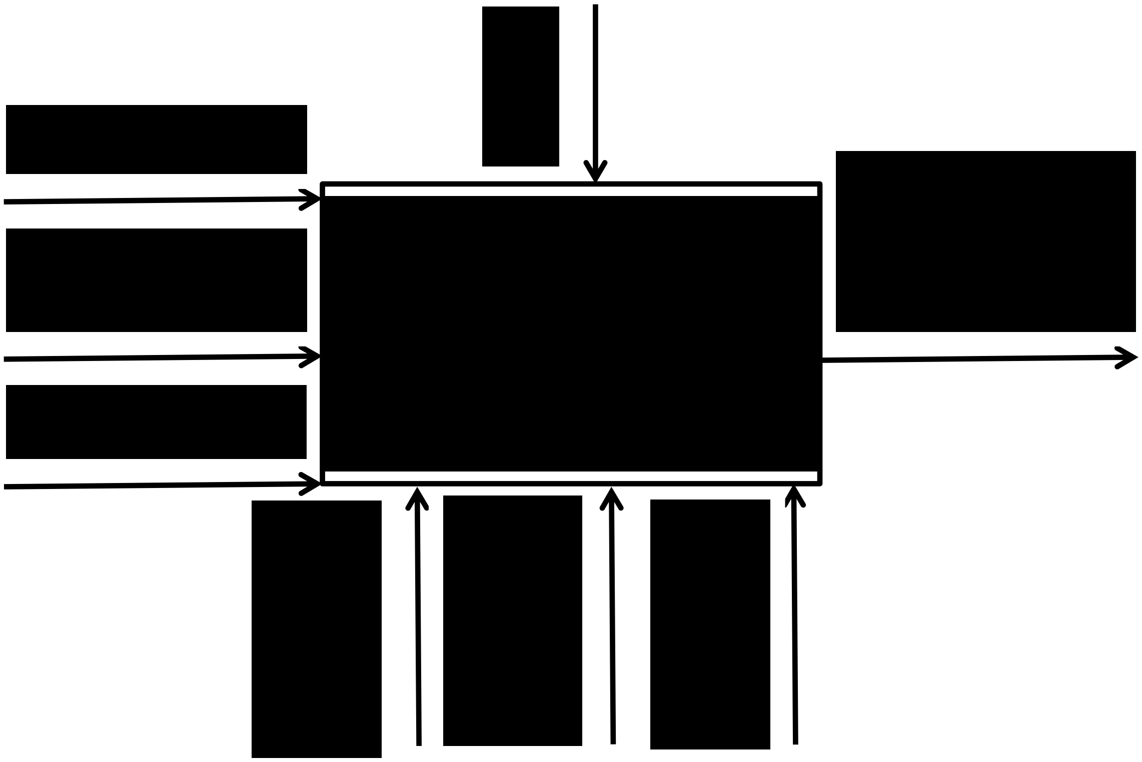
1.4 Developed Models

In this research, four different models are considered and consequently, some novel methods and mathematical models are developed. Figure 1.2 illustrates each model in the form of an $IDEF_0$ diagram showing the main activities along with inputs, outputs, controls and mechanisms (http://www.idef.com/idefo-function_modeling_method/).

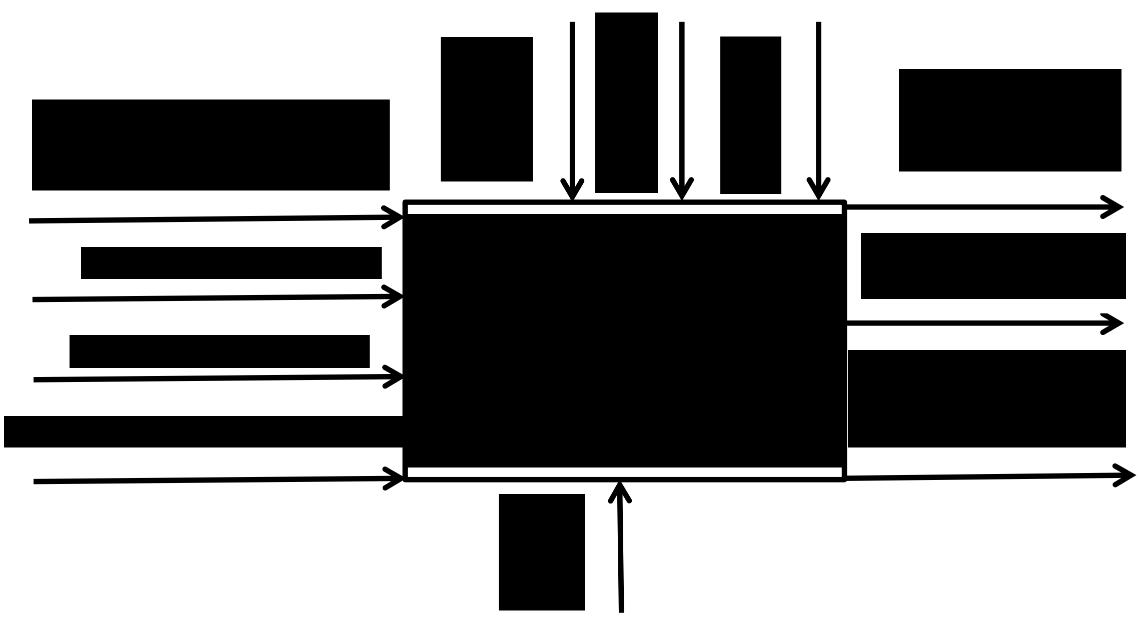
First model:



Second model:



Third model:



Fourth model:

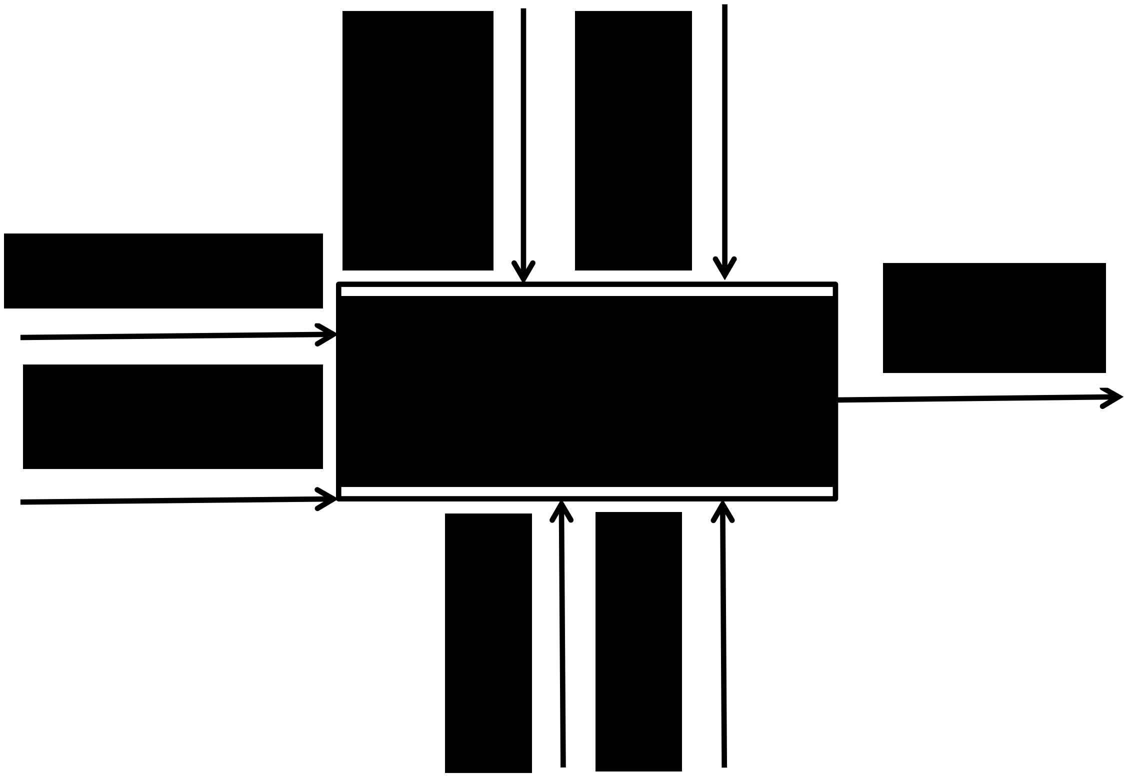


Figure 1.2 IDEF$_0$ for each developed model

1.5 Hypothesis

This research is based on the hypothesis that:

Utilizing the commonality between part/product variants could reduce changeover and operations sequence retrieval time and improve throughput, productivity and utilization of modern manufacturing systems for dealing with less production volume per variant.

1.6 Engineering Thesis Questions

In this sub-section, based on the conducted research, the typical engineering thesis questions are answered.

Q.1. What is the engineering problem to be solved?

Despite the fact that similarity notion has a significant impact on decreasing total setup time and improving utilization and productivity of the system in modern manufacturing systems, there is a lack of research in developing and applying novel methods using similarity aspect between part/product variants. Accordingly, four main domains have been addressed: production sequencing, product family formation, production flow, and product operations sequence retrieval.

Q.2. In what sense are previous works to this problem insufficient?

Production sequencing: most of the developed models and algorithms in this area are not practical in industries and some of these methods are computationally time consuming and cannot find solutions in a reasonable time.

Product family formation: in spite of significant works in this area, no research has been carried out for part/product variants having networked structure and hence, no practical method has been developed for grouping variants with such features.

Production flow: most of the works in literature focused on machine assignment to locations and they did not consider machine capability in performing different operations and also product variants having networked structures.

Operations sequence retrieval: Only few methods developed in literature regarding this problem and all of these works focus on assembly application and they lack fabrication aspect.

Q.3. What are the developed solutions in this research/contributions/significance?

Production sequencing: a novel production sequencing policy based on the setup similarity between product variants is proposed.

Product family formation: a novel operation flow based similarity coefficient is proposed which has been inspired by analysis of comparison made on biology field. It is then integrated with other similarity criteria namely operations similarity and production volume similarity to yield a more comprehensive coefficient.

Production flow: A mathematical model is developed to minimize backtracking distance by optimizing three decision variables: (1) assignment of machines to locations, (2) assignments of operations to machines, and (3) final sequence of performing operations for each product variant.

Operations sequence retrieval: two mathematical models are developed and an algorithm is proposed to generate master operations sequence from existing variants which can be used to retrieve operations sequences of new variants. Operations refer to both assembly and fabrication applications.

Considering the economic perspective of the present research, optimizing the objective functions in all the four topics would be equivalent to less cost for manufacturing systems.

1.7 Overview of Case Studies

In the first part of the research, a case study retrieved from the work carried out by Lin and Lio (2003) is provided in which six classes/variants of label stickers visit a calendar machine to glue surface material and liner for producing label stickers. The obtained result from the proposed sequencing policy is compared with the developed mathematical model which demonstrates the accuracy and speed of the proposed sequencing policy.

The second part of the research includes a case study from a local company manufacturing different parts for locomotive and automotive industries. For confidentiality reasons, the name of the company is not revealed. This case study includes seven representative parts with their operation sequences. It clearly illustrates variants having networked operation sequences and how the proposed integrated similarity coefficient is used to group the variants. Sequencing of the variants under some assumption is also provided to show the potential application of the proposed approach and the result is compared with the developed mathematical model to illustrate the efficiency of the proposed approach in terms of time and accuracy.

In the third part of the research, a case study of a family of three engine blocks is considered. The case study shows the real-application of the problem along with the optimum solution found by the developed mathematical model. It also demonstrates how minimizing backtracking improves the total throughput by 7.79%.

The last part of the research includes two case studies, one for assembly application and the other for fabricating application. The first case study considers five variants of pilot control valves while the second one includes nine variants of mostly ejecting and coupling parts/components produced by Rabourdin Industry. Both case studies demonstrate how a master operations sequence is derived from existing variants and used for constructing the operations sequences of new variants in real manufacturing environments.

1.8 Research Map

This research consists of five chapters in which three different models are addressed and the relevant literature review and developed methods/mathematical models are presented in Chapters 2, 3 and 4. Figure 1.3 outlines the research map.

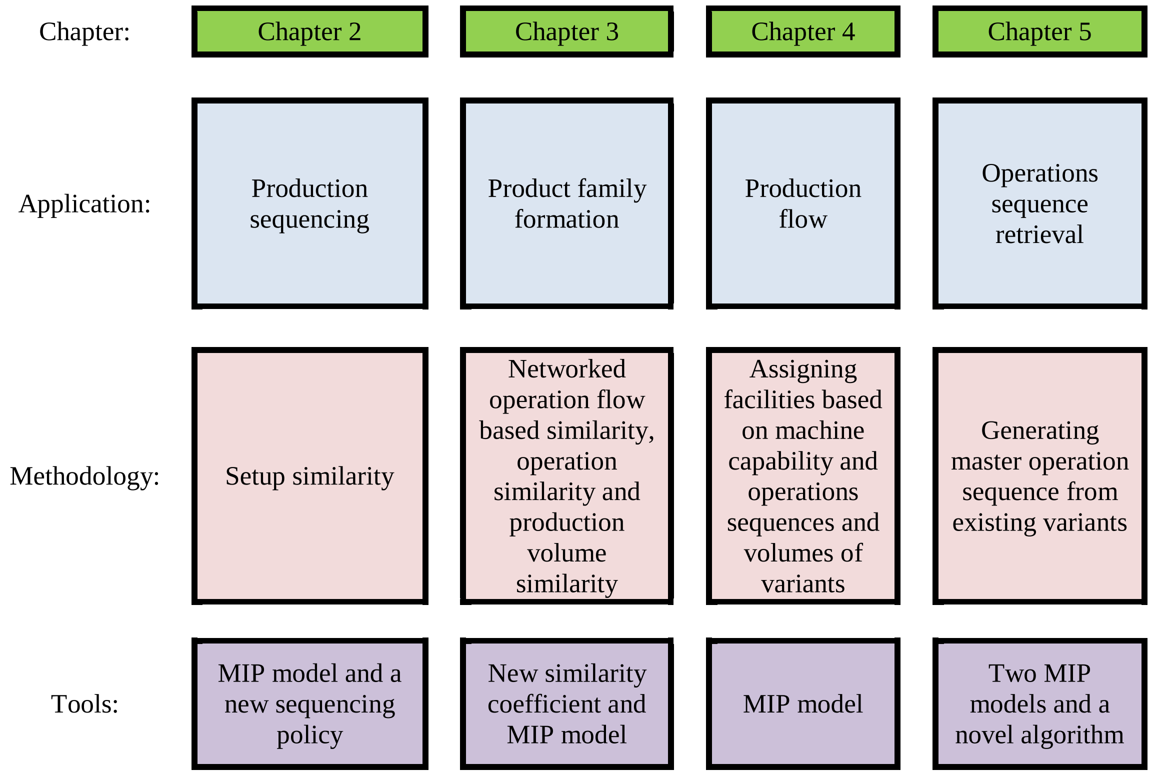


Figure 1.3 Research map

The dissertation consists of six chapters where the literature review of each topic is addressed at the beginning of its corresponding chapter. In Chapter 2, the sequencing policy is proposed along with the developed mathematical model. Chapter 3 considers product family formation with the novel similarity coefficient for variants having networked operation sequences which is then integrated with other similarity criteria. Chapter 4 consists of developing the mathematical model to solve facility assignments and determine the final sequence of performing operations with respect to backtracking minimization. In Chapter 5, the two developed mathematical models and the proposed algorithm are presented for generating a master operation sequence from the information of existing variants which can be used to retrieve the operation sequences of new variants. Finally, in Chapter 6, summary and conclusions with the directions for future works are presented.

Chapter 2

Grouping and Sequencing Product Variants Based on Setup Similarity

2.1 Introduction

Manufacturing systems should be designed to enable physical reconfiguration to shift from producing one product variant to another efficiently as needed. There are generally two levels of reconfiguration which are system level where machines are added/removed and machine level where tools, fixtures and machining directions may be changed (Koren et al., 1999; ElMaraghy, 2005; Koren and Shpitalni, 2010). However, reconfiguration at the system level is not very common and most manufacturing systems try to produce different variants using existing machines by changing tools, fixtures and programs. Therefore, frequent changes of machines setup are inevitable and managing these changes efficiently is important.

Setup time in a work station refers to the time required to change tools, fixtures and configuration of a machine from production/assembly of one product variant to another. Total setup time contributes to the productivity of a system (Luo et al., 2015). Productivity is defined as output over input. Total machine productivity is also defined as total number of finished products over total available time. Therefore, minimizing total setup time increases total finished products and consequently, improves productivity of the system. There are many manufacturing systems in which setup time is more significant compared with processing time (Shtub and Maimon, 1992; Kumar and Narendran, 1997; Rajkumar and Narendran, 1998). Examples of this situation include printed circuit boards (PCBs) and re-sharpening of cutting tools and painting applications where most manufacturing time is spent on changing the setup of machines to perform other operations. In these cases, maximizing throughput can be achieved by minimizing total setup changes and associated cost/time. An effective way to do this is by sequencing products based on their setup similarity.

This chapter proposes a new coefficient for calculating setup time similarity between each pair of product variants. A new sequencing policy is also developed to find the best sequence of product variants to mitigate the effect of frequent changes and setup by processing similar products sequentially in the master production schedule after grouping the product variants based on the proposed setup similarity coefficient and Average Linkage Clustering (ALC) algorithm. Although grouping of parts and products into families based on geometric and/or processing similarity and applying group technology to many downstream manufacturing activities has been researched extensively (Yin and Yasuda, 2006), the concept of grouping product variants based on their machine setup similarity to sequence them accordingly has never been addressed in the literature. The results of applying the proposed grouping method are compared with the developed mathematical model for a flow shop where bypassing is not allowed), to demonstrate the efficacy of the proposed similarity coefficient and sequencing policy.

2.2 Literature Review

The problem considered in this chapter is a flow shop where bypassing is not permitted which is referred to as permutation flow shop (Pinedo, 2012). There are generally two types of methods found in literature dealing with sequencing product variants in permutation flow shop problems: 1) exact methods such as mathematical models for scheduling the product variants for small to medium sized problems, and 2) approximation methods such as heuristic and meta-heuristic algorithms which try to find good, but not necessarily optimal, solutions (Yenisey and Yagmahan, 2014).

In the first category, some exact algorithms and models have been developed based on the features and objective functions. However, most of them are efficient only for small to medium sizes of problems. For instance, Tseng et al. (2004) conducted an empirical analysis to compare the effectiveness of four different integer programming models for the permutation flow shop problem. A set of 60 different problems were solved four times by each of the mathematical models and results were compared based on the computation time. M’Hallah (2014) developed an MIP model for a permutation flow shop with m number of stages to minimize total earliness and tardiness. This model is only suitable for small to medium sized problems and, hence, a meta-heuristic algorithm called VNS was used for large sized problems.

In addition to mathematical models, the branch and bound approach has been used for finding exact solutions (e.g. Toktas et al., 2004; Lin and Wu, 2006; Wang et al., 2015). For instance, Lemesre et al. (2007) developed an exact method based on the branch and bound approach to solve the permutation flow shop, which is in fact a parallel model to improve search speed. The proposed method was able to find optimal solutions for up to 20 machines and jobs. Moslehi et al. (2009) proposed a branch and bound algorithm for a two machine permutation flow shop. The objective function was the minimization of earliness and tardiness. Some lemmas were developed for increasing the computation speed and the performance was examined by considering 380 different problems.

However, the exact solution using mathematical models or other developed exact methods for large size of permutation flow shop problems encompassing many jobs and machines cannot be found (Yenisey and Yagmahan, 2014). For that reason, many researchers have applied or developed some approximation methods, such as metaheuristic methods, to deal with the complexity of the permutation flow shop problems. Zhao et al. (2015) proposed an evolutionary algorithm called shuffled complex evolution algorithm to solve a permutation flow shop problem. They applied OBL strategy to enhance the quality the algorithm. Twenty-nine instances were used to assess the performance of the proposed algorithm. Khan and Govindan (2011) considered a multi-objective permutation flow shop with makespan minimization and tardiness maximization. A simulated annealing (SA) algorithm was proposed and compared with other existing heuristic algorithms to illustrate the efficiency of the method. Lin and Ying (2013) considered an m-machine environment for a permutation flow shop and developed a simulated annealing (SA) algorithm. The objective function minimized makespan and total flow time. The proposed method was compared to six existing algorithms in the literature and proved to outperform them. In addition to SA, Tabu search (TS), genetic algorithm (GA), ant colony optimization (ACO), and particle swarm optimization (PSO) are some of the most common metaheuristic methods applied in literature (e.g. Ishibuchi et al., 2003; Armentano and Arroyo, 2004; Rahimi-Vahed and Mirghorbani, 2007; Yagmahan and Yenisey, 2010; Lin et al., 2015).

Some developed policies such as the shortest processing time (SPT) (Baker and College, 2008) can find good sequencing solutions and are not traditional heuristic approach. These policies are more practical to use in manufacturing systems than the exact solution approaches or traditional meta-heuristic based algorithms.

In order to apply the proposed sequencing policy, product variants are required to be first grouped based on machine setup similarity. Grouping variants based on their similarity has been addressed in literature and in different manufacturing environments. Most methods such as group technology (GT) have been developed for the cellular manufacturing environment (e.g. Alhourani and Seifoddini, 2007; Bhatnagar and Saddikuti, 2010; Paydar et al., 2014; Wu and Suzuki, 2015). Some methods were proposed for reconfigurable manufacturing systems (e.g. Galan et al., 2007a; Galan et al., 2007b; Goyal et al., 2013). In any of the studied systems, there are different criteria based of which the similarity coefficients have been developed; the most common criteria are operation similarity (e.g. Pattanaik and Kumar, 2010), operation sequence similarity (e.g. Choobineh, 1988), volume/demand similarity (e.g. Pattanaik and Kumar, 2010), operation time similarity (Ilic, 2014), and alternative process routings (e.g. Alhourani, 2013).

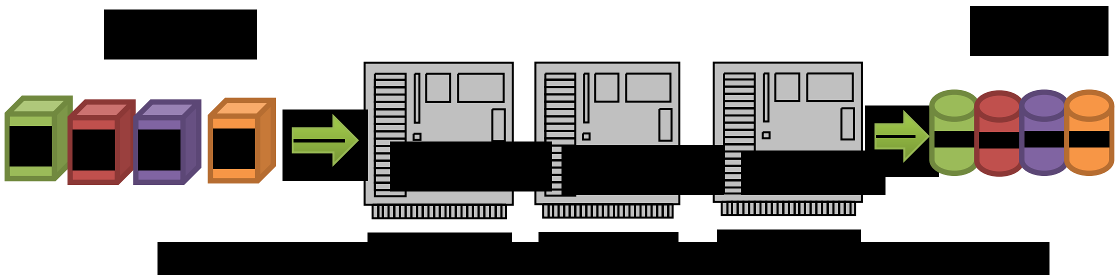
“Grouping” has been referred to using different terminology in literature such as “family formation”, “part/product formation”, “cell formation” and “clustering”. There are several ways in literature for grouping product variants. Mathematical models, descriptive procedures, and hierarchical clustering are some of the most common ones (Galan et al., 2007b). Average Linkage Clustering (ALC) algorithm is one of the hierarchical clustering algorithms which has been used by different researchers and proved to be efficient (Yin and Yasuda, 2006; Galan et al., 2007a). In this chapter, the ALC algorithm is applied for grouping product variants and subsequently a new policy is proposed for sequencing them.

2.3 Problem Description

A manufacturing system consists of several work stations such as milling, turning, drilling tapping, finishing, etc. to fabricate or assemble different product variants belonging to a product family and each of which requires a set of operations. Each station usually has a particular CNC machine which is able to perform different tasks by reprogramming. However, the types of operations performed at each station are not significantly different (i.e. they have some similarities). For instance, in a drilling station, the machine is capable of making holes with different diameters and depths at different location coordinates (x, y, z) or performing boring and tapping according to the programmed instructions.

Product variants have to pass through a similar sequence of stations. A product may not visit some stations according to its process plan. Therefore, product variants are different in two aspects: 1) the number and type of stations a product visits, and 2) the operations they require which may necessitate a machine setup change. Note that consecutive operations carried out at a given station may be identical, hence, requiring no setup changes.

Different product variants are processed based on their sequence and sequence changes between stations are not permitted since a permutation flow shop is considered. Therefore, sequence of the variants does not change from one station to another as bypassing is not allowed. This type of problem is very common in industries. In many manufacturing systems with a single production line, such as those found in auto-manufacturing companies, products are often large which makes it infeasible for products to bypass each other on the same line. In other situations, although the products are small in size, the limited production space does not allow bypassing. At any time, only one operation can be performed at a given station, hence, depending on process plans, it may require setup changes between consecutive variants processed at that station. Figure 2.1 shows a schematic illustration of a permutation flow shop for five variants and three stations. It is noteworthy mentioning that the demand for each variant may vary from single unit to several units. Therefore, product variants might be processed in the form of batches. Based on both the developed mathematical model and the proposed sequencing policy, it would be advisable to finalize one type of variant and then switch to another type in order to minimize setup changes.



Other considered assumptions which are applied are as follows: It is assumed that the time required for changing the setup from product variant A to variant B is equal to the time for changing the setup from variant B to variant A (in a given station). It is also assumed that setup time is more significant than processing time and each product variant is produced in batches of one, typical of customized or personalized products (Hu et al., 2011, AlGeddawy and ElMaraghy, 2012; ElMaraghy et al., 2013). In addition, it is assumed that all products have the same importance and that there is no specific prioritization among the variants. It is assumed that the first product at a given station does not require any setup change, since only the effect of setup changes between two consecutive variants is considered. The objective is to find the best sequence of product variants in order to minimize total setup changes/costs/times. This minimization would lead to maximizing throughput (Kumar and Narendran, 1997).

2.4 Mathematical Model

A mathematical model is developed to clearly demonstrate the problem and assess the efficiency of the proposed sequencing policy. This model is at first a non-linear mathematical model and is subsequently reformulated to a mixed integer linear programming (MILP) to find exact solutions using General Algebraic Modeling System (GAMS) optimization software with CPLEX 12.0.1 MILP solver. The notation used in this model can be structured into two parameter and variable groups. Note that in the following model a dummy product (product zero) has been considered to facilitate the mathematical manipulation.

Parameters:

n Number of product variants.

i,r Indices showing product variants *i,r* $i = 0,1,2,\dots,n$ $r = 0,1,2,\dots,n$.

j, m Indices showing product variant in positions j and m of a given sequence $j = 0,1,2,\dots,n$ $m = 0,1, j\text{-}1$.

k Number of work stations.

p An index showing work station *p* $p=1,2,\dots,k$.

S_{irp} Setup change time/cost for product variant *i* if processed immediately after product variant *r* at station *p*. It is assumed that there is no setup change time/cost for product variants processed first at a given work station (S_{i0p}=0). $i=1,2,\dots,n$, $r=0,1,2,\dots,n$, $p= 1,2,\dots,k$.

W_{ip} is equal to one if product variant *i* has an operation at work station *p* and zero otherwise $i=1,2,\dots,n$, $p= 1,2,\dots,k$.

Variables:

X_{ij} is equal to one if product variant *i* is placed in *j*th position of the sequence and zero otherwise $i=0,1,2,\dots,n$, $j=0,1,2,\dots,n$.

Z_{jmp} is equal to one if, at station p, product variant in position *j* is processed immediately after product variant in position *m* and zero otherwise $j=1,2,\dots,n$, $m=0,1,\dots,j\text{-}1$, $p= 1,2,\dots,k$.

Constants (2.1) to (2.3) are considered ensuring that the dummy product (product zero) is the only product placed in position zero and cannot be placed in other positions of the sequence:

$$X_{0j} = 0 \qquad \forall j = 1,2,\dots,n \quad (2.1)$$

$$X_{i0} = 0 \qquad \forall i = 1,2,\dots,n \quad (2.2)$$

$$X_{00} = 1 \qquad (2.3).$$

The objective function is to minimize the total setup change time/cost which is equal to:

$$Min \sum_{i=1}^{n}\sum_{r=0}^{n}\sum_{m=0}^{j-1}\sum_{j=1}^{n}\sum_{p=1}^{k} Z_{jmp} \cdot X_{ij} \cdot X_{rm} \cdot S_{irp} \qquad (2.4)$$

Subject to:

$$\sum_{i=1}^{n} X_{ij} = 1 \qquad \forall j = 1,2,\dots,n \quad (2.5)$$

$$\sum_{j=1}^{n} X_{ij} = 1 \qquad \forall i = 1,2,\dots,n \quad (2.6)$$

$$\sum_{j=1}^{n} Z_{j0p} = 1 \qquad \forall p = 1,2,\dots,k \quad (2.7)$$

$$\sum_{j=m+1}^{n} Z_{jmp} \le \sum_{i=1}^{n} X_{im}.W_{ip} \qquad \forall m = 0,1,\dots,n-1, \quad \forall p = 1,2,\dots,k \quad (2.8)$$

$$\sum_{m=0}^{j-1} Z_{jmp} = \sum_{i=1}^{n} X_{ij}.W_{ip} \qquad \forall j = 1,2,\dots,n, \quad \forall p = 1,2,\dots,k \quad (2.9)$$

$$X_{ij} \in \{0,1\} \qquad Y_{jp} = 0,1 \quad \forall i = 0,1,2,\dots,n, \qquad \forall j = 0,1,2,\dots,n,$$
$$\forall p = 1,2,\dots,k$$

$$Z_{jmp} \in \{0,1\} \quad \forall j = 1,2,\dots,n, \qquad \forall m = 0,1,\dots,j-1 \quad \forall p = 1,2,\dots,k \quad (2.10)$$

Constraints (2.5) and (2.6) ensure that a sequence position is occupied by only one product and a product is placed in only one position of the sequence. Constraint (2.7) guarantees that there is at least one product to be completed immediately after the dummy product at each station, otherwise the station would always be idle and it would be removed from the system. Constraint (2.8) ensures that a product can be processed immediately after the product in position m at station p if that product is in a position between m+1 and n and also the product in position m has an operation at station p. In other words, if the product in position *m* does not have an operation to be executed at station *p*, then, there is no product (in position between m+1 and n) to be completed immediately after the product in position m at station p. Constraint (2.9) indicates that if a product in position j requires an operation at station p, there must be a product in position between 0 and *j-1* which has been processed immediately before the product at that station, which can also be the dummy product. Finally, Constraint (2.10) indicates that X_{ij}, Y_{jp} and Z_{irp} are binary variables. The developed mathematical model tries to optimize the sequencing of the product variants in all stations. As an alternative approach, the model can be decomposed to each station and the optimal solution for each station is obtained and based on that, the final optimal sequence is found.

In the mathematical model, the objective function is non-linear. The model is reformulated by defining a new binary variable. By this way, it would be possible to have the equivalent linear model and get exact solutions from the equivalent linear model. The new variable is defined as follows:

Q_{ijrmp} is equal to one if product *i* is placed in position *j* and is processed immediately after product *r* which is placed in position *m* at station *p* of the sequence and it is zero otherwise; $\forall i, j = 1,2, \dots, n,\ \forall r = 0,1,2, \dots, n,\ \forall m = 0,1, \dots, j-1 \quad \forall p = 1,2, \dots, k.$

The previous non-linear objective function is reformulated to have the equivalent linear model and three new sets of linear constraints are added as follows. Therefore, there would be thirteen sets of linear constraints with a linear objective function.

$$Min \sum_{i=1}^{n} \sum_{r=0}^{n} \sum_{m=0}^{j-1} \sum_{j=1}^{n} \sum_{p=1}^{k} Q_{ijrmp} \,.\, S_{irp} \qquad (2.11)$$

Subject to:

$$Z_{jmp} + X_{ij} + X_{rm} - 2 \leq Q_{ijrmp} \qquad (2.12)$$

$$Q_{ijrmp} \leq \frac{Z_{jmp} + X_{ij} + X_{rm}}{3} \qquad (2.13)$$

$$Q_{ijrmp} = 0,1 \qquad (2.14)$$

$$\forall i, j = 1,2, \dots, n,\ \forall r = 0,1,2, \dots, n,\ \forall m = 0,1, \dots, j-1 \quad \forall p = 1,2, \dots, k.$$

2.5 Proposed Product Variants Grouping and Sequencing Policy

A new policy for grouping and sequencing variants is where variants are sequenced based on their relative degree of similarity which is discussed in this section.

2.5.1 Grouping Variants

A new method for calculating the setup similarity coefficient between pairs of products is proposed. The product variants share many common components and operations since they all belong to the same family. However, common operations do not guarantee similar setups. For instance, it is clear that the two variants in Figure 2.2 share many common operations. Both variants require milling operations with similar dimensions and, hence, the milling station setup change between these two variants is negligible. On the other hand, both variants require drilling operations but for different hole depth and diameter. This necessitates changing tools and the program used by the drilling machine.

Figure 2.2 Two product variants belong to the same family

The goal of the sequencing policy in this chapter is to find the similarity between product variants to minimize machine setup changes and sequence them accordingly.

Each product requires a set of operations, each of which is performed at one station. The relative degree of setup similarity between variants i and j at a given station can be calculated as follows:

$$\mathrm{DoS}_{\mathrm{ijp}} = 1 - \frac{\text{time required for changing the setup from variant j to variant i in station p}}{\text{total setup time changes for each pair of variants visiting station p}} \qquad (2.15)$$

Where,

$$p \in R_{ij}, \qquad R_{ij} = \{\, stations\ that\ both\ variants\ i\ and\ j\ visit\}$$

It is clear that DoS_{ijp} is meaningful when both variants i and j are processed at station p. If at least one of the variants does not require an operation at station p, then their relative degree of setup similarity at station p does not exist.

The setup similarity of two variants is relative. In other words, the setup similarity of two variants cannot be interpreted by itself without comparing with setup similarity of other pairs of variants. After calculating the relative degree of similarity between two variants at each station, the total setup similarity between them is determined. First, stations that both pair of product variants visit are identified. The weight of each of those stations is then obtained by dividing the total setup time changes in that station over the summation of total setup time changes in all those stations. For each pair of variants, the summation of the station weight multiplied by the relative degree of setup similarity at that station is calculated over the common stations visited by both variants. This value is equal to the setup similarity between two variants and is summarized in Equation (2.16):

$$\mathrm{S}_{\mathrm{ij}} = \sum_{p \in R_{ij}} \frac{\text{total setup time changes for all pair of variants visiting station p}}{\text{total setup time changes for all pair of variants visiting } \mathrm{R}_{ij}} \times \mathrm{DoS}_{\mathrm{ijp}} \qquad (16)$$

$$R_{ij} = \{\, stations\ that\ both\ variants\ i\ and\ j\ visit\}$$

According to Equations (2.15) and (2.16), when calculating the setup similarity of two product variants, only those stations both variants visit are considered. For more clarity, consider Figure 2.3 which demonstrates two variants of a lap chair.

Figure 2.3 Two variants of a lap chair (Polo's Furniture Blog)

It is evident that most operations needed to finalize these two chair variants are similar and the variants share many common components. Suppose that there is a station dedicated for joining the handles to chairs. There would be no setup change between these two variants at the arms joining station as the black chair visits the station but the white does not. Therefore, the arms joining station is not considered in calculating the setup similarity between these two variants.

ALC algorithm

The Average Linkage Clustering (ALC) algorithm is used to group product variants, after finding the similarity coefficient between each pair of variants. Grouping results are used to find the best sequence of product variants. In this algorithm, products with higher coefficient of similarity are grouped together (Vakharia and Wemmerlöv, 1995, Galan et al., 2007a, Galan et al., 2007b, Eguia et al., 2011, Eguia et al., 2013). A sub-matrix consisting of ungrouped products and a family of products is then constructed. This procedure is repeated until all the products are grouped into a single family and consequently a dendrogram can be drawn. The similarity of product variants at each stage can be obtained using Equation (2.17) (Galan et al., 2007b):

$$S_{ij} = \frac{\sum\sum S_{mn}}{N_i \times N_j} \qquad (2.17)$$

Where *i* and *j* are families' indices; m and n are products of families *i* and *j* respectively; S_{ij} is the similarity coefficient between families *i* and *j*; and N_i and N_j are the number of products at each family.

2.5.2 Sequencing Product Variants

The proposed method is a policy to sequence products based on their setup similarity and it is not a traditional heuristic. This sequencing policy is straightforward and does not need local search as with most traditional heuristic-based sequencing methods.

Following are steps to obtain the final sequence of product variants after the dendrogram is constructed. Any two variants with the highest similarity coefficients, based on the dendrogram, should be placed next to each other in the sequence. If the next level of dendrogram is related to two other variants, they should also be next to each other in the production sequence. If a new variant is further grouped with a pair of variants which have previously been grouped together, then the similarity coefficient between the new variant and each of the two variants is checked. The value of similarity coefficient determines which of the two variants is placed next to the new one in the processing sequence. The same procedure is repeated when two groups, each of which has several variants, are further grouped. In this case, we only check the similarity coefficient with extreme variants at the start or the end of their own groups.

2.6 Illustrative Example

An illustrative example is presented to demonstrate the sequencing policy. Assume that there are five product variants with four stations (machines) and that the setup similarity coefficients, obtained by using the proposed formula, between each pair of variants are as follows:

$$S_{12} = 1 \quad S_{13} = 1 \quad S_{14} = 0.67 \quad S_{15} = 0.75 \quad S_{23} = 0.83 \quad S_{24} = 0.75 \quad S_{25} = 0.83 \quad S_{34} = 0.75 \quad S_{35} = 0.67 \quad S_{45} = 1$$

ALC algorithm is used to further group the variants. A dendrogram is constructed based on the obtained results, after clustering variants as follows:

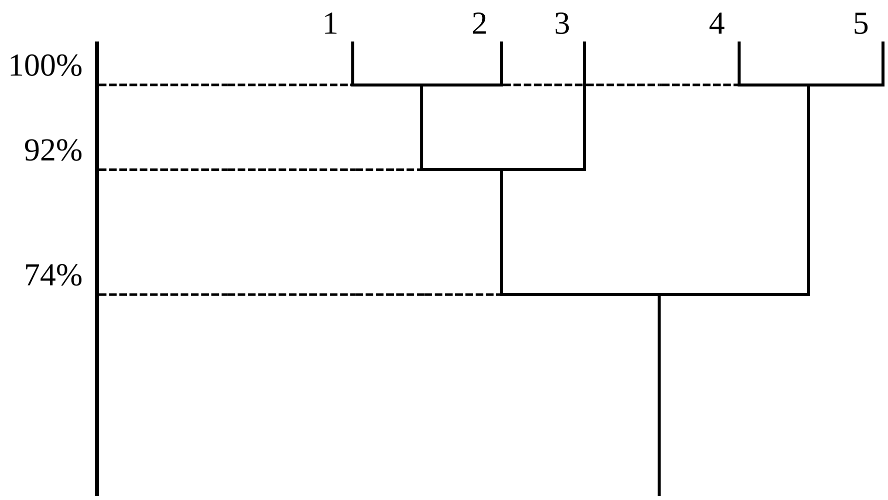


Figure 2.4 Dendrogram for the illustrative example

According to Figure 2.4, Variants 1 and 2, then Variant 3 should be next to each other in the sequence. Similarly, Variants 4 and 5 should be placed next to each other in the production schedule. Therefore, the general structure of setting products in sequence is as illustrated in Figure 2.5:

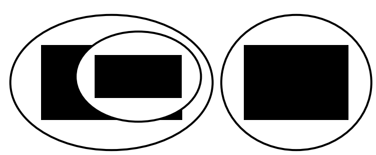


Figure 2.5 Rudimentary variants sequence for the above example

The proper position for Variant 3 should be decided first and that whether it should be next to Variant 1 or Variant 2. The setup similarity value between pairs of 3 and 1 with pairs of 3 and 2 is compared. According to the given information, $S_{13} = 1$ while $S_{23} = 0.83$; therefore, Variant 3 should be next to Variant 1 in the sequence.

The best sequence of Variants 1, 2, and 3 are either 3→1→2 or 2→1→3. Variants 4 and 5 should now be added to the sequence. It is required to compare the similarity for those products in extreme positions, to find the best sequence. Therefore, comparing the similarity coefficients of the following pairs is considered:

$$S_{34} = 0.75,\ \ S_{35} = 0.67,\ \ S_{24} = 0.75,\ \ S_{25} = 0.83$$

Variants 2 and 5 should be placed next to each other in the sequence. Therefore, the best sequences of products would be either 3→1→2→5→4 or 4→5→2→1→3, since the objective is reducing setup cost. The obtained product variants sequence helps production planners in finalizing the production sequence.

2.7 Numerical Results

Several numerical tests were conducted and the results were compared with the exact solutions obtained from the linear mathematical model using GAMS, to demonstrate the efficiency of the proposed grouping and sequencing policy. The computer used for this analysis has the following features: 2.13 GHz CPU processor using Windows 7 operating system with 2 GB of RAM. The mathematical MILP model of the sequencing problem was implemented in GAMS optimization software and solved with CPLEX 12.0.1 MILP solver. The proposed grouping and sequencing policy was implemented in Borland C++ compiler version 5.02.

The problem was solved for small (less than 8 variants), medium (between 8 to 25 variants) and large sized groups (more than 25 variants). However, the GAMS solver was not able to find the exact solutions in reasonable computing time and was terminated after 60 minutes for medium and large problem sizes in most cases and the best obtained answer was reported. The solutions obtained by GAMS in those cases are not necessarily optimal but are sub-optimal as GAMS may not be able to find the optimal solution in 60 minutes.

For this purpose, 3, 4, 6, 8, 10, 15, 25 and 50 jobs *(n)* are selected. The number of stations/machines *(m)* is set to be 1, 3, 5, 10 and 20. Therefore, all the combinations are

40. For each combination, 25 replicates were generated randomly and hence, a total number of 1000 instances were evaluated (8×5×25).

Two measures namely relative average percentage error and run time were considered to assess the performance of the proposed sequencing policy versus GAMS solutions. The relative percentage of error is calculated as follows: 100*(the solution found by proposed methods (or by GAMS) - best solution)/ (best solution). The best solution can be obtained either by the proposed methods or by GAMS. The relative percentage of error is calculated for all 25 replicates of each combination and the average error is obtained for each method accordingly. Therefore, for a given combination of *n* and *m*, the average error obtained for both methods might be higher than zero (For example refer to *n*=8, *m*=20 in Table 2.1). In large problem sizes, since GAMS may not find the optimal solution within 60 minutes, the solution obtained by the proposed method may be better and considered the best solution. This method of error calculation was previously used by other authors such as (Mozdgir et al., 2013).

Table 2.1 summarizes the results.

Table 2.1 Comparison between GAMS method and the proposed sequencing policy

		GAMS		Proposed Sequencing Policy	
n	m	Avg. Error (%)	Avg. time (Sec.)	Avg. Error (%)	Avg. time (Sec.)
3	1	0	3.6	0	
	3	0	3.6	0	
	5	0	4.1	0	
	10	0	3.8	0	
	20	0	3.5	0	
4	1	0	3.4	0	
	3	0	3.6	3.3	< 0.01
	5	0	4	3.7	
	10	0	3.6	4.1	
	20	0	4.2	0.7	
6	1	0	3.3	0	
	3	0	6.7	4.9	
	5	0	8	3.4	

	10	0	52.1	3.8	
	20	0	208.7	2.6	
8	1	0	4.1	0	
	3	0	98.4	4.4	
	5	0	827.4	1.9	
	10	0	2829.6	7.1	
	20	0.47	3600*	2.4	
10	1	0	29.8	0	
	3	0	2480	5.9	
	5	4.27	3600*	0.4	
	10	2.62	3600*	0.8	
	20	6.53	3600*	0	
15	1	0	3060	0	< 0.02
	3	29.5	3600*	0.9	
	5	36.8	3600*	0	
	10	35.9	3600*	0	
	20	37.5	3600*	0	
25	1	88.9	3600*	0	
	3	Out of Memory	N/A	0	
	5	Out of Memory	N/A	0	
	10	Out of Memory	N/A	0	
	20	Out of Memory	N/A	0	
50	1	Out of Memory	N/A	0	
	3	Out of Memory	N/A	0	
	5	Out of Memory	N/A	0	
	10	Out of Memory	N/A	0	
	20	Out of Memory	N/A	0	

* GAMS execution has been terminated after 3600 seconds.

As seen in Table 2.1, the proposed sequencing policy is able to find good solutions (even optimal) within 0.02 seconds which is extraordinarily fast. In contrast, the time used by

GAMS to find the exact solutions exponentially increases with the problem size. Figures 2.6 and 2.7 illustrate this trend for GAMS results in terms of the number of jobs and stations respectively. Since, GAMS could not find the exact solution in 60 minutes for 10 or more jobs (as we set a time limit of 3600 seconds for GAMS), it was terminated. Therefore, the run time of GAMS is illustrated only for 3, 4, 6 and 8 jobs to appropriately demonstrate the trend of run time in terms of number of jobs and stations (Figures 2.6 and 2.7).

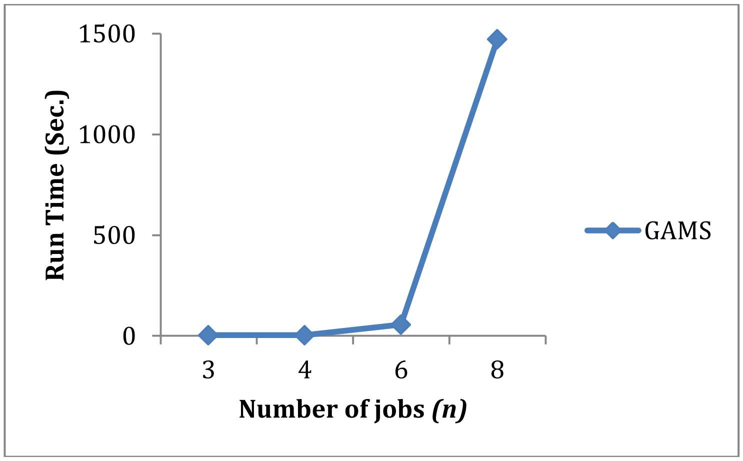


Figure 2.6 GAMS run time in terms of number of jobs *(n)*

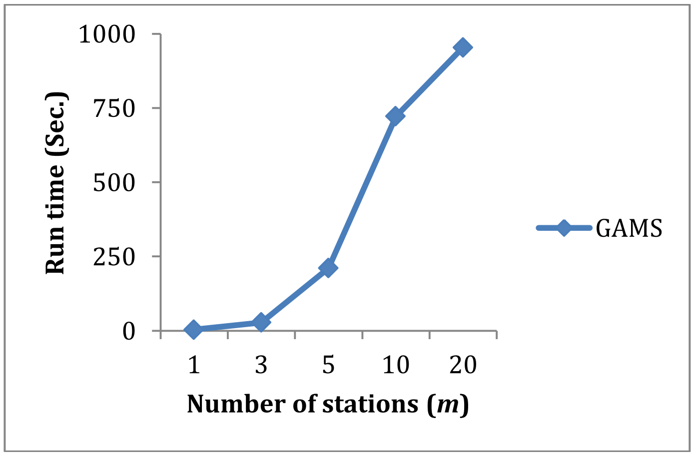


Figure 2.7 GAMS run time in terms of number of stations (*m*)

Figure 2.6 shows that the time spent by GAMS in solving the problem increases exponentially with the number of jobs. Figure 2.7 indicates that although the time increases with the number of stations, it is not as sharp as the trend in Figure 2.6. The proposed sequencing policy is not sensitive to the size of the problem and can quickly find the solutions.

Figures 2.8 and 2.9 compare GAMS to the proposed sequencing policy based on the relative average percentage of error for different numbers of jobs and stations respectively. This comparison is limited to 15 jobs since GAMS cannot find any solution as it runs out of memory for large problem sizes. This also proves the efficiency of the proposed sequencing policy.

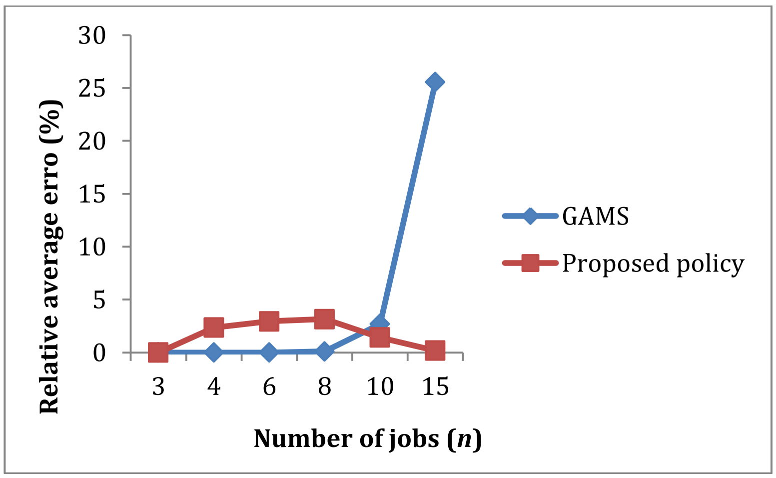


Figure 2.8 Average percentage error comparison in terms of number of jobs *(n)*

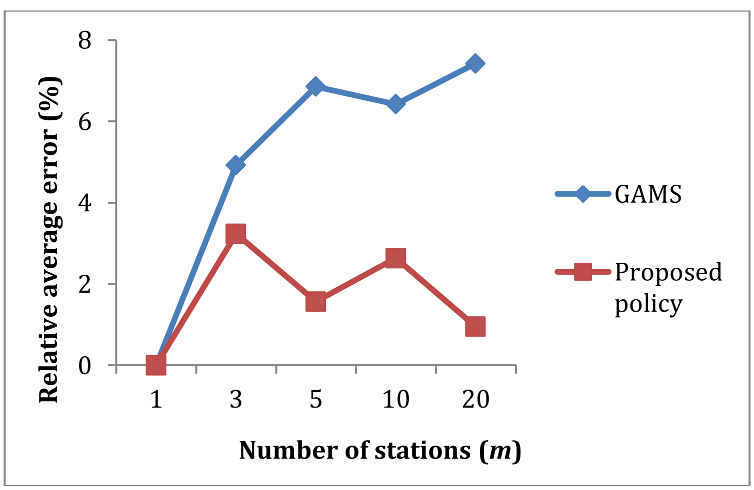


Figure 2.9 Average percentage error comparison in terms of number of stations *(m)*

Figure 2.8 shows that for small number of jobs, GAMS does not have any relative error, and then it increases as the number of jobs increases since it cannot find exact solutions in 60 minutes and the best answer until then was considered. The proposed policy, as shown in Figure 2.8, yields exact solutions for all the studied n=3. Then, the relative error for the proposed sequencing policy increases and while GAMS cannot give the exact solution within 60 minutes, it starts decreasing. In other words, in large problem sizes, the proposed sequencing policy produces more accurate solutions than GAMS. Note that, depending on the situation, the run time for MIP model can change and one might be interested to allow GAMS to solve the problem for a longer time than 60 minutes.

Figure 2.9 compares the relative percentage error in terms of number of stations (m). According to this figure, the proposed sequencing policy yields exact solutions for all the studied m=1. In addition, in spite of lack of any specific trend in error for both methods, the proposed sequencing policy has less relative average error than GAMS for the same aforementioned reason. It is noteworthy mentioning that the initial non-linear model was also solved for some cases and revealed the same solutions. However, it took a longer time to find the optimal solutions.

An efficient way in obtaining optimum solutions is to integrate both mathematical model and the proposed sequencing policy. In this approach, after finding the solution (sequence of jobs) using the proposed policy, it is given to the GAMS software as a “warm start” point (Ferris and Voelker, 2002). In this way, the computing time of GAMS can be reduced. However, since the aim of the present work is to propose a policy to be easily implemented in industry, this integrating approach has not been considered.

2.8 Case study – Labels Stickers Making

A case study of label sticker printing is presented to demonstrate the practicality of the proposed sequencing policy in real manufacturing systems. The required process is to print a specific image on a particular labeling material. The case study is a two-stage hybrid flow shop taken from the work carried out by Lin and Lio (2003) and also considering some information obtained from another source (http://www.a4labels.com/). We only consider the first stage of this industrial application for comparison as the setup time in the second stage is sequence-independent. In stage one, a single high speed calendar machine (Figure 2.10) is used to glue surface material and liner for producing label stickers. There is a total number of six classes (variants) of jobs (label stickers) visiting stage one. The calendar needs setup change when switching from jobs in one class to jobs in another which is dependent on the type of previous and current classes of jobs.

Figure 2.10 An example of a calendar machine (http://www.wotol.com/)

There are six classes/variants of label stickers used for suitcases, shoes, water bottles, anti-theft/shoplifting, windows, and box labels. Moreover, three sticker adhesive bases namely Rubber, Acrylic, and Water-based are used. Each adhesive base consists of two types which are compatible with each other and, hence, there are six adhesive bases in total. The temperature required for each of the six adhesive bases is 120, 130, 110, 130, 120 and 130 ^{o}C respectively based on adhesive specifications. The setup includes two main activities which are: changing the adhesive base and modifying the operating temperature. If the components of the preceding and successor adhesive bases are incongruous with each other when changing the adhesive base, then the preceding one is cleaned out of the container before the successor is installed. If they are compatible then the cleaning/setup procedure is eliminated. Table 2.2 shows this classification. In this case study, the processing time is assumed to be negligible compared to setup time.

Table 2.2 Classification of label stickers

Adhesive base		Temperature (^oC)	Class/variant
Rubber	g1	120	Suitcase label(C1)
	g2	130	Shoe label(C2)
Acrylic	g3	110	Bottled water label(C3)
	g4	130	Anti-theft shoplifting label(C4)
Water-based	g5	120	Window label(C5)
	g6	130	Box label(C6)

Setup time required for cleaning the adhesive base is about 8 minutes on average. In addition, the operating temperature takes about 1.5 minutes to decrease or increase by 1^oC, so it can take up to 30 minutes to reach the correct temperature (adjusting temperature).

The final setup change time from one class of stickers to another can be calculated according to the aforementioned information. Table 2.3 summarizes these values.

Table 2.3 Setup time considering both changing temperature and cleaning the base (minutes)

	To					
From	C1	C2	C3	C4	C5	C6
C1	0	15	23	23	8	23
C2		0	38	8	23	8
C3			0	30	23	38
C4				0	23	8
C5					0	15

The relative degree of similarity is first calculated between each pair of label sticker variants at each station based on the proposed sequencing policy. In this case study, only one station has been considered, hence, p=1. Table 2.4 summarizes the calculated similarities. Equation (2.17) shows that when there is only one station, the relative degree of setup similarity (DoS_{ijp}) is equal to the setup similarity coefficient (S_{ij}).

Table 2.4 Relative degree of setup similarity

	j					
	1	2	3	4	5	6
$DoS_{1\,j1}$	-	0.951	0.925	0.925	0.974	0.925
$DoS_{2\,j1}$		-	0.876	0.974	0.925	0.974
$DoS_{3\,j1}$			-	0.902	0.925	0.876
$DoS_{4\,j1}$				-	0.925	0.974
$DoS_{5\,j1}$					-	0.951

Both the proposed sequencing policy and GAMS yield the same minimum total setup time of 62 minutes. This validates the accuracy of the proposed sequencing policy. The obtained sequence of jobs to be carried out on the Calendar machine is C2→C4→C6→C5→C1→C3. Now consider a situation in which these six variants are sequenced randomly. Using computer, the following random sequence was obtained: C5→C3→C2→C1→C6→C4. This sequence yields total setup time of 107 minutes. This perfectly shows the merit of sequencing based on the proposed policy reducing total setup time by 42%. If it is assumed that the total available time in the system is 8 hours per day, the machine productivity increases from 373/480 to 418/480, equivalent to 12% improvement.

As mentioned before, the aim of proposing the sequencing policy is to have a method which can be easily implemented in industry and it is not designed to serve as a robust approach in order to deal with stochastic setup times. In other words, when there is uncertainty about the setup time of product variants, the obtained sequence from the proposed policy may need to be updated depending on different factors such as magnitude of changes in setup times, initial values of setup times, number of stations with changes in setup times, and so on. However, proposing a robust approach can be a potential direction for future works.

2.9 Summary

In this chapter, sequencing of product variants in a flow shop problem was considered where bypassing is not permitted. It was assumed that setup times/costs are more considerable / important than processing time. Therefore, the effectiveness of the manufacturing system improves by minimizing total setup changes between product variants. The problem was formulated and modeled using MINLP and then converted to MILP model in order to obtain the exact solutions using GAMS optimization software. A new sequencing policy based on machine setup similarity between product variants was developed. This policy is very simple and unlike traditional meta-heuristics can easily be implemented in manufacturing systems. Computational results demonstrated that the proposed sequencing policy is very efficient and capable of finding good and even optimum solutions in less than a second regardless of the number of considered jobs and stations. The validity of the proposed sequencing policy was confirmed numerically and by using a case study in label sticker industry. The novelty of this policy is the utilization of the machine setup similarity in grouping and sequencing product variants. This sequencing policy is very beneficial in cases when jobs' due dates are the same and set up time is more significant than processing time. In other cases, it would still be beneficial to use in order to obtain a good preliminary sequence which may further be modified, if required, by the production planner based on other criteria. This approach reduces the time and effort required to sequence product variants.

Chapter 3

Grouping Part/Product Variants Based on Networked Operations Sequence

3.1 Introduction

Grouping part/product variants based on their similarities is one of the most efficient strategies in modern manufacturing systems to manage the variety. There are different criteria based of which part/product variants can be grouped. One of these criteria is “operations sequence”. A part/product variant (or variant in brief) usually requires a number of operations. Examples of these operations are machining, assembly, painting, finishing, and packaging. The term “operation” here refers to both “fabrication” and “assembly” processes. Operations usually have precedence constraint relationship but it is not necessarily serial. Most of the time, based on the features of a variant, some operations have flexibility to be done before, or after or simultaneously with other. In this case, the operations sequence of the variant would similar to a network of connected operations. This situation usually adds more complexity to the manufacturing system in terms of flow of operations and managing the changeover between different variants. Grouping variants based on operations flow similarity has several advantages including ease of system reconfiguration (Youssef and ElMaraghy, 2006) and reduction of changeover time.

An “operation precedence graph” is a good representation of operations sequence. Figure 3.1 illustrates the difference between operation precedence graphs of two variants with sequential serial operations and networked operations. The numbers in the figure refer to the operations. As a real example, suppose that Operations 1 and 2 at Figure 3.1(a) refer to “Drilling” and “Internal Threading” processes respectively. It is obvious that “Internal Threading” process cannot be carried out unless “Drilling” process is finalized. Now, suppose that Operations 5 and 6 at Figure 3.1(b) represent “Boring” and “Cylindrical Turning” processes. Generally, these two processes do not have any precedence constraints and they can be performed before or after each other or at the same time using a CNC machining center. Those variants that only require the first mentioned group of operations have serial structures while variants requiring the second mentioned group of operations have networked structures. The reason for using the term “networked structure” for variants similar to the one in Figure 3.1(b) is because of the flexibility in performing some operations which makes the structure of a variant precedence graph similar to a network.

Similarity measurement for variants having networked operations sequence has not been considered in literature. There is also a need to develop a novel operations flow based similarity coefficient to measure the similarity between the variants with network structure to be used in grouping them.

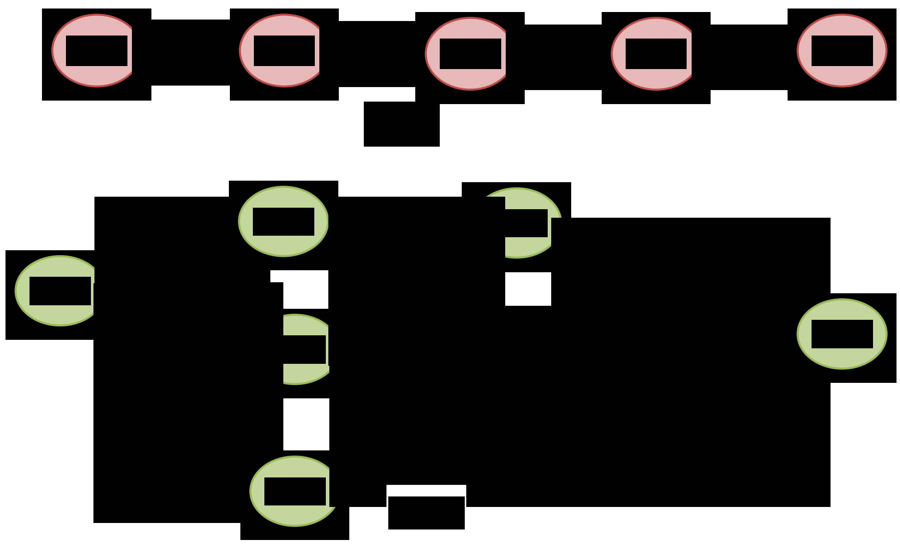


Figure 3.1 An example of comparison between operation precedence graphs of two variants with: (a) sequential serial operations and (b) networked operations

In this chapter, the proposed operations flow based similarity coefficient considers this processing order flexibility which is different from routing flexibility. In other words, it is required to process all the operations of the operation precedence graph; however, some operations have flexibility of being processed in different order.

The proposed operations flow based coefficient is in fact based on analysis and principles used in comparing the similarity of biological structures. The comparison of different metabolic pathways in different systems has enabled biologists to understand the evolutionary and structural relationships between different species (Heymans and Singh, 2003). In this field, there are many biological systems which have network structures.

Consequently, various methods have been proposed in order to compare these metabolic network structures (Heymans and Singh, 2003; Ogata et al., 2000).

In addition to proposing a new operations flow based coefficient which is dominantly based on the operations sequence, two other important similarity criteria used in the literature, namely operations similarity and production volume similarity, are also taken into consideration to yield a more comprehensive similarity measurement which encompasses different similarity aspects.

The most common and well-known similarity coefficient in the literature, regarding operations similarity, is Jaccard's similarity introduced by McAuley (1972). Therefore, this coefficient is used in the present chapter to satisfy the operations similarity aspect.

Production volume similarity identifies and groups the variants with similar volume in order to increase the utilization of machines. One of the most well-known coefficients for production volume similarity was developed by Galan et al. (2007b); but it has a drawback as it does not consider volume ratio. Therefore, in this chapter, a new production volume similarity coefficient is also developed which is in fact a modification on the coefficient developed by Galan et al. (2007b) to obviate its drawback.

After considering all three similarity coefficients (i.e. operations flow based similarity, Jaccard's similarity, and production volume similarity) and obtaining the integrated similarity coefficient, variants are grouped accordingly using the Average Clustering Algorithm (ALC). Grouping variants is useful particularly in a cellular manufacturing system where each cell has its own machines responsible for processing families of similar variants which increases utilization and productivity of machines/stations (Kashkoush and ElMaraghy, 2014a).

A secondary application of grouping variants based on their similarity is finding an appropriate sequence of variants for production/assembly. This application is most suitable when the system processing one variant at a time, then setup is changed for production/assembly of the next variant. The obtained sequence serves as a better starting sequence for production planners and requires modification based on processing time, due dates, etc. The sequencing application is also considered here and a proper method of sequencing variants is provided and validated mathematically.

3.2 Literature Review

There are many similarity coefficients proposed for grouping variants based on different criteria. Most methods have been developed as group technology (GT) for the cellular manufacturing environment (e.g. Alhourani and Seifoddini, 2007; Bhatnagar and Saddikuti, 2010; Nouri and Hong, 2012; Wu and Suzuki, 2015). Some methods were proposed for reconfigurable manufacturing systems (e.g. Galan et al., 2007a; Galan et al., 2007b; Goyal et al., 2013). There are different criteria for which the similarity coefficients have been developed in any of the studies systems. The most common criteria are operations similarity (e.g. Pattanaik and Kumar, 2010), operations sequence similarity (e.g. Choobineh, 1988; Tam, 1990), volume/demand similarity (e.g. Galan et al., 2007b; Pattanaik and Kumar, 2010), and operations time similarity (Ilic, 2014).

Choobineh (1988) developed a method for measuring similarity coefficient between two parts which uses the common sequences of length 1 through L between two components. The number of cells and assignments of parts to the cells are calculated. The proposed method applies two criteria, parts' operations and operations' sequences, where the later criterion is the significance of the work. However, serial operations sequences were considered in calculating similarity coefficient.

Askin and Zhou (1998) proposed a method to form flow lines in manufacturing cells. Process planning and operations sequence of each part were applied to form the families. The objective was to find a set of flow-line cells at minimum cost of machines and material handling. The cells were assumed to be independent. Two concepts were applied to form the part families: longest common sub-sequence and shortest composite super-sequence. A method was proposed to calculate the similarity coefficient and a modified hierarchical clustering algorithm was proposed to group the parts. The optimum machine sequence for each cell was found using an augmented graph based on the capabilities of machines in performing the operations. The computational experiments showed the efficiency of the method.

Regarding multiple process routing, Gupta (1993) considered a problem in flexible manufacturing systems where alternative routing is possible to address machines failure. A similarity coefficient was designed based on machines, not parts, and it incorporated production volume and number of trips. A dendrogram was used to illustrate the final grouping of machines. A case study was provided using MANUPLAN software and the results show the efficacy of this method.

Alhourani (2013), in a more recent work, addressed the cell formation problem by considering different criteria such as multiple process routings, operations sequence, batch size, production volume and the capacity of machines. He claimed that the mentioned criteria have not been considered together in literature. The proposed method of grouping machines is based on minimization of intercellular movements. Each part has at least one process route when considering multiple process routings and each route may consist of a set of different machines compared to other process routes. This method was compared to other existing methods proposed by other researchers such as Yin and Yasuda (2002), Lei and Wu (2005), and Spiliopoulos and Sofianopoulou (2007) and proved that the proposed method outperforms others in terms of inter-cellular moves.

Alhourani and Seifoddini (2007) developed a new ordinal production data similarity coefficient to form a machine cell in a cellular manufacturing system. The method is based on sequence of operations in which a part visits different machines and the batch size of each part. The results were compared with the current methods available in the literature and a new clustering algorithm to form the machine cells was proposed. The proposed similarity coefficient is generally sensitive in terms of intercellular movements. The machines are first grouped based on their commonality and the cells are formed and finally parts are assigned to the cells based on the volume of traffic between machines. This method of grouping is based more on machine capabilities.

Bhatnagar and Saddikuti (2010) proposed and compared two cellular manufacturing system models, sequential and concurrent. The sequential model utilizes machine-part information while the concurrent model considers both machine-part and machine-operator information to form cells. The results showed the superiority of the concurrent model over the sequential model.

Chang et al. (2013) considered three aspects of cellular manufacturing design simultaneously and claimed that cell formation, cell layout, and intercellular machine sequence are three practical and crucial aspects that are yet to be taken into consideration together. They proposed a mathematical model which encompasses different aspects such as alternative process routings, operations sequences and production volume. However, only the serial operations sequence was considered and the family formation was based on machine selection. A tabu search algorithm was developed to find solutions for large scale problems.

Navaei and ElMaraghy (2014) proposed a new similarity coefficient for grouping product variants considering alternative machines for each operation and based on their similarity in machine usage. It was assumed that for each operation of a given variant more than one candidate machine are available and only one machine should be selected for that operation. Nevertheless, for another product variant having the same operation a different machine might be selected depending on the ability of the new machine to perform other operations of that variant. A non-binary incidence matrix was developed based on this assumption and the average linkage clustering (ALC) algorithm was used to group the variants accordingly. The work (Navaei and ElMaraghy, 2014) was based on probabilistic principles in machine selection and does not consider the structure of the variants operations. The present chapter does not address the choice of alternate machines for operations. It considers the operations structures, represented by the networked operations sequences, and groups product variants based on their operations flow similarity, operations similarity and production volume similarity.

Wu and Suzuki (2015) proposed a new method of cell formation in the cellular manufacturing environment. Their method consists of two phases. Part families are identified in the first phase by using different criteria such as serial operations sequence similarity, and in the second phase, a mathematical model is presented to assign the part families to machines to minimize the total cost, i.e. machine cost, operation cost and inter-cell movement cost. The efficacy of the proposed method was examined using different test problems and sensitivity analysis.

In a different approach, Lin et al. (2010) applied a simulated annealing (SA) algorithm in cell formation. They applied this method to solve the part-machine cell formation problem. They also developed an approach based on Cauchy function to escape from local optimal solutions. They compared the proposed algorithm with thirteen traditional algorithms in the literature by considering four performance measures namely total bond energy, machine utilization, grouping efficiency and percentage of exceptional elements. The results proved the efficiency of the proposed SA.

There are other works in the literature which applied operations sequence in family formation for serial operations sequences and did not consider the network configuration (e.g. Goyal et al., 2012; Goyal et al., 2013; Won and Logendran, 2015, Tambuskar et al., 2015).

It is clear that the literature does not include any work considering a network structure for operations sequences of variants.

Variants should be grouped after finding the integrated similarity coefficient. This "grouping" has been expressed using different terminology in literature. "Clustering", "family formation" and "cell formation" are some of the frequently used titles.

Several methods have been developed for grouping variants. Mathematical programming, hierarchical clustering, descriptive procedures, and array-based clustering are the most popular in literature (Galan et al., 2007b). Average Linkage Clustering (ALC) algorithm is one of the most efficient hierarchical clustering grouping methods. Therefore, in this chapter, ALC algorithm is applied for grouping variants based on their similarities. This method will be described in more detail in Section 3.5.

3.3 Similarity Coefficient Based on Operations Flow

The problem is first described in this section with a brief discussion regarding how biological structure comparison can be utilized in developing the proposed method. The proposed operations flow based similarity coefficient is then presented with an illustrative example. An extension to the proposed coefficient is also provided.

3.3.1 Problem Description

There are a number of part/product variants which belong to a larger family. Each variant has its own sequence of operations with the corresponding operation precedence graph. The operation precedence graph of each variant, unlike previous research in literature, has a network structure meaning that there are some operations that have flexibility to be carried out before, after or at the same time with some other operations (Figure 3.1b). The objective is to find a good similarity coefficient between each pair of such variants based on the similarity of operation flows.

3.3.2 Biological Network Structure

The proposed coefficient, which will be described in Sub-Section 3.3.3, has been inspired by analysis and principles used in biology. There are organisms in biology which have network structures. Moreover, there are biological species whose structures are not only network-based but also have directed edges between nodes. Enzymes are an example of such cases where biological enzymes are represented by nodes and when a given enzyme like e1 catalyzes a reaction the output of which is the substrate (a molecule) for e2, it is represented by an edge from e1 to e2.

In biology, the metabolic pathways of different enzymes are compared to discover their evolutionary distance and organizational relationships. Different notions are considered when comparing two enzyme graphs such as structural similarity, sequence similarity of genes and identity mapping. The basic principle behind the comparison of similarity of two nodes in two graphs is that they are considered similar when the two nodes reference and are referenced by similar nodes (Heymans and Singh, 2003). This principle will be applied in developing the proposed similarity coefficient.

3.3.3 Proposed Operations Flow Based Similarity Coefficient

A new similarity coefficient is proposed to enable comparison of two variants having networked operations sequence. As mentioned before, operation precedence graph can be used as a representation of the operations sequence of a variant. Operations are considered to be nodes and their connections serve as directed edges.

Suppose that there are two Operation Graphs *A* and *B* representing the operations sequences of Variants “a” and “b”. Also, suppose that O_m refers to any common operation between these two variants. The similarity coefficient (SoF) of the two variants based on their operations flow can be obtained using Equations (3.1) to (3.5). Note that the value of this coefficient is between zero and one.

$$SoF_{ab} = \frac{\sum_{for\ all\ O_m \in \{G_A \cap G_B\}} (QC_{in}(O_m) + QC_{out}(O_m))}{\sum_{for\ all\ O_m \in \{G_A \cap G_B\}} (max\{Q_{in}(O_m)\} + max\{Q_{out}(O_m)\})} \tag{3.1}$$

Where:

$$QC_{in}(O_m) = number\ of\ common\ incoming\ edges\ to\ O_m \tag{3.2}$$

$$QC_{out}(O_m) = number\ of\ common\ outgoing\ edges\ from\ O_m \tag{3.3}$$

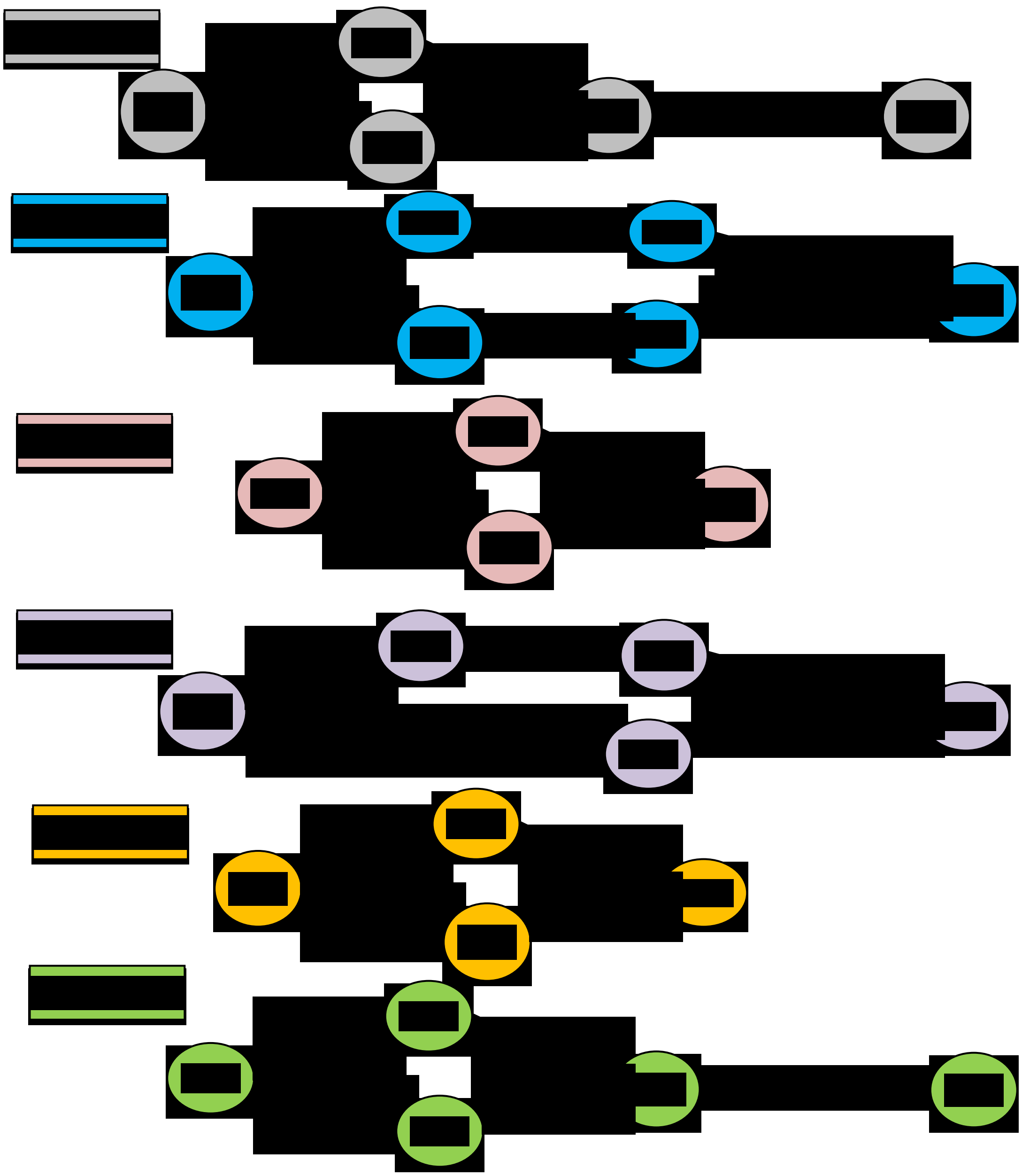
$$Q_{in}(O_m) = number\ of\ incoming\ edges\ to\ O_m \tag{3.4}$$

$$Q_{out}(O_m) = number\ of\ outgoing\ edges\ from\ O_m \tag{3.5}$$

The following steps, according to the above equations, are applied for each common operation like Operation *m*. First, all the nodes from which Operation *m* has incoming edges in Operation Graph *A* are compared with their counterparts in Operation Graph *B* and the total number of common incoming edges is recorded. The same procedure is applied for outgoing nodes from Operation *m* in Operation Graph *A* and its counterparts in Operation Graph *B* and total number of common outgoing nodes is recorded and summed with the recorded incoming edges. The result then appears in the numerator. Meanwhile, the number of incoming and outgoing edges from Operation *m* in Graph *A* is respectively compared with the number of incoming and outgoing edges in Graph *B* and the maximum numbers are recorded and summed together. The obtained value then appears in the denominator.

3.3.4 Illustrative Example

In this sub-section, an example of a family of six variants is considered. This example will also be used in other sections. Figure 3.2 demonstrates the corresponding operation precedence graphs of the variants.



[redacted]

After calculating the incoming and outgoing edges for common operations, the operations flow similarity between each pair of variants would be as follows:

$$SoF_{12} = \frac{(2) + (1 + 0) + (1 + 1) + (1 + 1) + (1)}{(2) + (1 + 1) + (2 + 1) + (1 + 1) + (2)} = \frac{8}{11} = 0.73.$$

$$SoF_{13} = \frac{(2) + (1 + 0) + (1 + 0) + (0)}{(2) + (1 + 1) + (1 + 1) + (2)} = \frac{4}{8} = 0.5,$$

$SoF_{14} = 0.55,$ $SoF_{15} = 0.33,$ $SoF_{16} = 0.18,$ $SoF_{23} = 0.5,$

$SoF_{24} = 0.67,$ $SoF_{25} = 0.5,$ $SoF_{26} = 0.18,$ $SoF_{34} = 0.5,$

$SoF_{35} = 0.67,$ $SoF_{36} = 0.44,$ $SoF_{45} = 0.57,$ $SoF_{46} = 0.18,$ $SoF_{56} = 0.33.$

Calculating the similarity coefficient manually gets difficult as the number of operations and variants increases. However, by developing a code in commercial software such as Excel, MATLAB, or C++ it can be easily calculated. In the present work, Borland C++ software was implemented. Binary matrices were used to decode the operations sequences of variants to the program. Each variant has a binary matrix of $k \times k$ in which k refers to the maximum number of operations among all variants. The element a_{ij} of the matrix for a given variant is equal to one if there is a directed edge from Operation i to Operation j in its operations sequence and is zero otherwise. If all elements of q^{th} column and q^{th} row are equal zero, it means that the variant does not require Operation q. In addition to the variants matrices, the total number of variants and maximum number of operations are the other inputs to the program. The program checks the existence of each node for each variant. If, in the matrix of a given variant, there is at least one element with value of one in the row or column corresponding to a node (operation), it means that the node exists for that variant. Next, for each existing node in a given variant, the total number of incoming edges to that node is calculated by summing up all the elements of the column corresponding to that node (in the matrix of that variant). Similarly, the total number of outgoing edges is calculated by summing up all the elements of the row corresponding to that node. For any common operation of each pair of variants, the number of common incoming and outgoing edges is obtained and the operations flow based similarity between each pair of variants is calculated based on the total incoming and outgoing edges.

3.3.5 Extension of the Proposed Operations Flow Based Similarity- Weighted Edges

In the proposed similarity coefficient, it has been assumed that the flow from one operation to another is equally important for all operations. However, sometimes due to reasons such as route complexity the importance of flows between operations is different. For instance, suppose that two particular stations responsible for two consecutive operations are connected by a gantry crane while other stations are connected by simple conveyors. In this case, the manufacturer/planner may prefer to group the variants requiring the use of the gantry crane together as much as possible.

In such cases, the importance weight of operations flows should be considered in calculating the similarity coefficient. This is referred to as weighted edges (Zager, 2003) in which the edge between two nodes (operations) of *i* and *j* has a weight of w_{ij}. Therefore, Equation (3.1) is modified and the extended operations flow based similarity between Variants *a* and *b* is calculated as follows:

$$SoF'_{ab} = \frac{\sum_{for\ all\ O_m \in \{G_A \cap G_B\}} (\sum_{i=1}^{T} W_{im}.K_{ima}.K_{imb} + \sum_{j=1}^{T} W_{mj}.K_{mja}.K_{mjb}\ )}{\sum_{for\ all\ O_m \in \{G_A \cap G_B\}} (max\{\sum_{i=1}^{T} W_{im}.K_{ima}\ ,\ \sum_{j=1}^{T} W_{jm}.K_{jmb}\ \} +\ max\{\sum_{k=1}^{T} W_{mk}.K_{mka}\ ,\ \sum_{l=1}^{T} W_{ml}.K_{mlb}\ \})} \quad (3.6)$$

Where,

$$K_{ima} = \begin{cases} 1 & if\ Variant\ "a"\ has\ a\ directed\ connection\ from\ Operation\ i\ to\ m \\ 0 & Otherwise \end{cases}$$

(3.7)

and G_A and G_B are the representing operation graphs of Variants a and b, W_{im} is the weight of the edge between Nodes (Operations) i and m and T is the total number of operations among the variants.

Figure 3.3 illustrates this notion for Variants 1 and 2 in the above example.

The extended operation flow based similarity between these two variants using Equation (3.6) can be calculated as follows:

$$SoF_{12} = \frac{(W_{12} + W_{15}) + (W_{12} + 0) + (W_{53} + W_{36}) + (W_{15} + W_{53}) + (W_{36})}{(W_{12} + W_{15}) + (W_{12} + max\{W_{23}, W_{24}\}) + (W_{23} + W_{53} + W_{36}) + (W_{15} + W_{53}) + (W_{46} + W_{36})}.$$

If all the weights are equal to one, then the similarity of operations flow between Variants 1 and 2 would be 8/11 which was obtained previously.

3.4 Other Similarity Criteria Consideration

In this section, two important similarity criteria in the literature, operations similarity and production volume similarity, are considered in combination with the proposed operations flow based similarity to yield a more comprehensive similarity coefficient. The most frequently used coefficient in operations similarity is the Jaccard's similarity

(Yin and Yasuda, 2006). Therefore, it is applied in this chapter as an aspect of operations similarity. In addition, a novel extension to Jaccard's similarity is proposed. The coefficient developed by Galan et al. (2007b) for production volume similarity is modified to avoid its drawback and a new volume based similarity coefficient is proposed and developed.

3.4.1 Operations Similarity

Considering Equations (3.1) to (3.5) and the described example reveals that in the proposed method, if the obtained operations flow based similarity value between two variants is close to one, then, not only do both variants have a similar operations flow, but they also share many common operations. However, the operations flow based similarity value of two variants sharing many common operations may be low as their operations flows are not similar. In addition, depending on the situation of the manufacturing system, operations similarity may have more importance over the operations flow similarity. Therefore, operations similarity criterion is also taken into consideration and for this purpose, Jaccard's similarity coefficient is applied and calculated as follows:

$$C_{ab} = \frac{e}{e+f+g} \qquad 0 \le C_{ab} \le 1 \qquad (3.8)$$

In Equation (3.8), C_{ab} is the Jaccard's similarity coefficient between Variants *a* and *b*; *e* is the number of common operations between Variants *a* and *b*; *f* is the number of operations in Variant *a* but not in Variant *b*; and g is the number of operations in Variant *b* but not in Variant *a*.

Considering the above illustrative example, the operations similarity between variants based on Jaccard's similarity would be as follows:

$$C_{12} = \frac{5}{5+0+1} = 0.83, \quad C_{13} = 0.8, \quad C_{14} = 0.67, \quad C_{15} = 0.5, \quad C_{16} = 1$$

$$C_{23} = 0.67, \quad C_{24} = 0.83, \quad C_{25} = 0.67, \quad C_{26} = 0.83, \quad C_{34} = 0.5$$

$$C_{35} = 0.6, \quad C_{36} = 0.8, \quad C_{45} = 0.5, \quad C_{46} = 0.67, \quad C_{56} = 0.5.$$

The operations similarity value between Variants 1 and 2 (C_{12}=0.83) is not far from the operations flow based similarity for these variants (SoF_{12}=0.73) since Variants 1 and 2 have a high level of similarity in operations flow indicating that they should also have many common operations (0.83 degree of similarity). As a result, the value of C_{12} is close toSoF_{12} . Variants 1 and 6 have identical operations; nonetheless, they follow a different sequence of operations. Therefore, the similarity of operations flows for these variants is low ($SoF_{16} = 0.18$) whereas their Jaccard's similarity has a value of 1.

An extension to Jaccard's similarity coefficient considers the importance weight for each individual operation. In fact, it may be useful to group variants requiring a particular operation like heat treatment together for energy usage considerations. In this case, Equation (3.8) is modified as follows:

$$C'_{ab} = \frac{\sum_{k=1}^{T} W_k . M_{ak} . M_{bk}}{\sum_{k=1}^{T} W_k . N_{abk}} \tag{3.9}$$

Where,

$$M_{ak} = \begin{cases} 1 & if\ Variant\ a\ requires\ Operation\ k \\ 0 & Otherwise \end{cases} \tag{3.10}$$

$$N_{abk} = \begin{cases} 1 & if\ at\ least\ one\ of\ Variants\ a\ or\ b\ requires\ Operation\ k \\ 0 & Otherwise \end{cases}$$

(3.11)

and T is the total number of operations among the variants and W_k is the weight of Operation k. It is clear that $0 \leq C'_{ab} \leq 1$.

3.4.2 Production volume similarity

Another important criterion for grouping variants is volume/demand similarity. Variants with low relative volumes mixed with variants having high relative volumes would cause the system to experience a significant level of under-utilization. On the other hand, grouping products with similar volumes leads to more comparable workloads and allows for better utilization in the system (Kashkoush and ElMaraghy, 2014a).

The production volume similarity coefficient developed by Galan et al. (2007b) is among the most common methods for part/ product family formation (Kashkoush and ElMaraghy, 2014a). This coefficient is formulated as follows:

$$D_{ab} = 1 - \frac{|d_a - d_b|}{d_{max} - d_{min}} \quad , \qquad 0 \le D_{ab} \le 1 \qquad (3.12)$$

Where D_{ab} is the production volume similarity coefficient between Variants a and b, d_a and $d_b$are volumes of Variants a and b respectively, and d_{max} and d_{min} are the maximum and minimum volumes throughout the entire variants.

This coefficient, however, has a drawback as it is based on the volume difference between two variants and does not consider the volume ratio between two variants. For instance, if there are four variants (V1, V2, V3, and V4) with production volumes of 100, 200, 900, and 1000 respectively then according to above formula, $D_{12} = 1 - \frac{100}{900} = 0.888$ and similarly, $D_{34} = 1 - \frac{100}{900} = 0.888$. Despite the fact that the volume difference between Variants 1 and 2, and Variants 3 and 4 is equal to one hundred, Variant 1 requires only 50% of the system capacity required for Variant 2; however, the system capacity increase rate from Variants 3 to Variant 4 is relatively low (only 10%). Therefore, grouping variants based on Equation (3.12), depending on the capacity of manufacturing machines, could reduce system utilization.

For that reason, another factor is taken into consideration which includes volume ratio as follows:

$$DR_{ab} = 1 - \frac{|d_a - d_b|}{\max(d_a, d_b)} \quad , \qquad 0 \le DR_{ab} \le 1 \qquad (3.13)$$

The formula is based on the volume ratio between two variants. However, it does not consider the volume difference between different pairs of variants. For instance, assume that there are four variants with volumes of 1, 2, 100, and 200. According to Equation (3.13), $DR_{12} = 1 - \frac{1}{2} = 0.5$ and similarly, $DR_{34} = 1 - \frac{100}{200} = 0.5$. However, the difference in volume between Variants 1 and 2 is only one while the difference in volume between Variants 3 and 4 is 100.

Therefore, a formula derived from both criteria represents a more effective approach. This can be calculated as follows:

$$DRD_{ab} = 1 - \left(W_1 \times \frac{|d_a - d_b|}{d_{max} - d_{min}} + W_2 \times \frac{|d_a - d_b|}{\max(d_a, d_b)}\right), 0 \le DRD_{ab} \le 1 \;, \; W_1 + W_2 = 1 \quad (3.14)$$

Where, DRD_{ab} is the production volume similarity coefficient between Variants *a* and *b*, and W_1 and W_2 are the weights of each criterion mentioned in Equations (3.12) and (3.13). These weights based on the difference between volumes and machine capacity should be assigned by planners. However, they usually take the same value (0.5-0.5) as they have more or less equal importance.

For above illustrative example, the production volumes for the six variants are assumed according to Table 3.1 as follows:

Table 3.1 Volume for each variant of the illustrative example

Variants	Volume
1	20
2	15
3	40
4	65
5	40
6	50

Also, assume that W_1=W_2=0.5. According to Equation (3.14), the production volume similarity between variants is as follows:

$$DRD_{12} = 1 - (0.5 \times \frac{|20-15|}{75-15} + 0.5 \times \frac{|20-15|}{\max(20,15)} = 0.83$$

$$DRD_{13} = 0.55, \quad DRD_{14} = 0.2, \quad DRD_{15} = 0.55, \quad DRD_{16} = 0.4$$

$$DRD_{23} = 0.44, \quad DRD_{24} = 0.12, \quad DRD_{25} = 0.44,$$
$$DRD_{26} = 0.3, \quad DRD_{34} = 0.56$$

$$DRD_{35} = 1, \quad DRD_{36} = 0.8, \quad DRD_{45} = 0.56, \quad DRD_{46} = 0.73,$$
$$DRD_{56} = 0.8.$$

According to Table 3.1, it is clear that Variants 3 and 5 and then Variants 3, 5 and 6 should have the highest level of production volume similarity. On the other hand, Variants 2 and 4 should have the lowest production volume similarity. These conclusions can be verified by reviewing the results obtained using Equation (3.14).

3.4.3 Integrated Similarity Coefficient

In decision making theory, the most common and simplest multi-criteria decision analysis (MCDA) method for comparing different alternatives based on some defined criteria is the "weighted sum" approach (Triantaphyllou, 2000). Therefore, three addressed similarity coefficients, "operations flow based similarity", "Jaccard's similarity" and "production volume similarity" coefficients, are integrated using the weighted sum to provide a more comprehensive similarity coefficient. The importance of each similarity criterion is determined by the production planners. Thus, the similarity between Variants *a* and *b* can be calculated using Equations (3.15) and (3.16):

$$SM_{ab} = W_{SoF} \times SoF_{ab} + W_C \times C_{ab} + W_{DRD} \times DRD_{ab} \ , \qquad (3.15)$$

Where, $0 \leq SM_{ab} \leq 1 \ , \ W_{SoF} + W_C + W_{DRD} = 1$ (3.16).

In Equations (3.15) and (3.16) $SM_{ab}$is the similarity coefficient of Variants a and b, and W_{SoF}, W_C, and W_{DRD} are the weights of operations flow similarity, Jaccard's similarity, and production volume similarity respectively. These weights are between zero and one, and their sum is one.

In the addressed example, suppose that W_{SoF}, W_C, and W_{DRD} are 0.4, 0.3 and 0.3 respectively and that weighted edges and weighted operations are not considered; then the similarity between each pair of variants can be obtained as follows:

$SM_{12} = 0.4 \times 0.73 + 0.3 \times 0.83 + 0.3 \times 0.83 = 0.79,$
$SM_{13} = 0.4 \times 0.5 + 0.3 \times 0.8 + 0.3 \times 0.55 = 0.61,$
$SM_{14} = 0.4 \times 0.55 + 0.3 \times 0.67 + 0.3 \times 0.2 = 0.48,$
$SM_{15} = 0.4 \times 0.33 + 0.3 \times 0.5 + 0.3 \times 0.55 = 0.45,$
$SM_{16} = 0.4 \times 0.18 + 0.3 \times 1 + 0.3 \times 0.4 = 0.49,$
$SM_{23} = 0.4 \times 0.5 + 0.3 \times 0.67 + 0.3 \times 0.44 = 0.53,$
$SM_{24} = 0.4 \times 0.67 + 0.3 \times 0.83 + 0.3 \times 0.12 = 0.55,$
$SM_{25} = 0.4 \times 0.5 + 0.3 \times 0.67 + 0.3 \times 0.44 = 0.53,$
$SM_{26} = 0.4 \times 0.18 + 0.3 \times 0.83 + 0.3 \times 0.3 = 0.41,$
$SM_{34} = 0.4 \times 0.5 + 0.3 \times 0.5 + 0.3 \times 0.56 = 0.52,$
$SM_{35} = 0.4 \times 0.67 + 0.3 \times 0.6 + 0.3 \times 1 = 0.75,$
$SM_{36} = 0.4 \times 0.44 + 0.3 \times 0.8 + 0.3 \times 0.8 = 0.66,$
$SM_{45} = 0.4 \times 0.57 + 0.3 \times 0.5 + 0.3 \times 0.56 = 0.55,$
$SM_{46} = 0.4 \times 0.18 + 0.3 \times 0.67 + 0.3 \times 0.73 = 0.49,$
$SM_{56} = 0.4 \times 0.33 + 0.3 \times 0.5 + 0.3 \times 0.8 = 0.52.$

The three considered similarity criteria namely operations flow based similarity, operations similarity (using Jaccard's similarity coefficient) and production volume similarity are among the most important and commonly used criteria in the literature (Yin and Yasuda, 2006).

There is a relationship between the operations flow based similarity coefficient and the Jaccard's similarity coefficient used for operations similarity criterion. As mentioned before, when two variants do not share similar operations, it means that they probably do not visit similar stations and therefore, their operations flows would also be different. In such case, the values of both Jaccard's similarity and operations flow based similarity coefficients for that pair of variants would be relatively low. However, if two variants share several similar operations (resulting in a high value for Jaccard's similarity coefficient) and, hence, visit similar stations, it still does not guarantee that they will have similar flows of operations; since, depending on features of the two variants, they may visit the stations in different sequence leading to a low value of operations flow based similarity coefficient. Regarding the production volume similarity coefficient, it is not much related to the other two coefficients; however, the production volume similarity criterion is an important factor in grouping variants and many researchers in literature considered it along with other factors (e.g. Galan et al., 2007b; Pattanaik and Kumar, 2010; Kashkoush and ElMaraghy; 2014a).

The weight of each coefficient indicates the relative importance of the corresponding criterion. The weights affect how the variants are grouped and by changing the weights, a variant might be reallocated from one group to another. In the case of cellular manufacturing systems, the number of cells and how the variants are assigned to each cell can be affected by changing the weights of the criteria. In literature, these weights are left for the planner to assign. However, some general guidelines are offered here.

If variants with different flows of operations are assigned to one cell/line, some modifications in switching from one variant to another will be needed requiring investment in time and effort and an increase in changeover time. Therefore, when the connections between stations/machines are very rigid or complex such that modifying them requires considerable time and effort, the weight assigned to the operations flow based similarity coefficient should be relatively high.

When there are several cells/lines each of which is used for a group of variants, it is advised to assign variants having large number of similar operations to the same cell/line to avoid under-utilization within each cell or increased wasteful intercellular movement and hence, increase productivity. In this case, Jaccard's similarity coefficient should get a higher weight.

If variants with different production volumes are assigned to the same cell/line/station/machine, the system will experience under-utilization for those variants with low demand. In this case, the weight assigned to the production volume similarity coefficient should be relatively high.

There are different methods in the literature that can help the planners/decision makers to assign the importance weights to each coefficient. Some of the most common methods include the fixed point scoring, rating, ordinal ranking and paired comparison methods (Galan et al., 2007b). Analytic Hierarchy Process (AHP) proposed by Saaty (1980) is one of the paired comparison methods and is the most popular method for finding the importance weights of qualitative criteria.

3.5 Hierarchical Grouping of Part/Product Variants

Variants are grouped based on the obtained similarity coefficient to achieve the aforementioned advantages of grouping. After grouping variants, a dendrogram is constructed. In order to group the variants, an average linkage clustering (ALC) algorithm (Galan et al., 2007a, Galan et al., 2007b) is used in this chapter. In this hierarchical based algorithm, the values of the obtained similarity coefficient are used to group the most similar pair of variants together. Next, either two other variants are grouped together or another variant is added to the already grouped variants. This procedure continues until all variants are grouped into a single family. At each step, after grouping, the similarity between variants is recalculated by using the following equation (Galan et al., 2007a):

$$SM_{kl} = \frac{\sum_{a\in k}\sum_{b\in l} SM_{ab}}{N_k . N_l} \qquad (3.17),$$

Where k and l are part/product families; a and b are variants of Family k and Family l respectively; SM_{ab} is the coefficient similarity between Variants a and b; SM_{kl} is the similarity coefficient between Families k and l; and N_k and N_l are the number of variants within Families k and l respectively.

Using the current example, first, Variants 1 and 2 are grouped since they have the highest value of similarity among all pairs of variants. The coefficient of similarity between this family and other variants can be obtained using Equation (3.17).

$$SM_{(12),3} = \frac{SM_{13} + SM_{23}}{2 \times 1} = 0.57, \qquad SM_{(12),4} = \frac{SM_{14} + SM_{24}}{2 \times 1} = 0.515,$$

$$SM_{(12),5} = \frac{SM_{15} + SM_{25}}{2 \times 1} = 0.49, \qquad SM_{(12),6} = \frac{SM_{16} + SM_{26}}{2 \times 1} = 0.45.$$

The following table of similarity is constructed accordingly:

Table 3.2 Similarity coefficient of variants after first grouping

	3	4	5	6
1,2	0.57	0.515	0.49	0.45
3		0.52	0.75	0.66
4			0.55	0.49
5				0.52

According to Table 3.2, the highest similarity coefficient is 0.75. Thus, in the next step, Variants 3 and 5 are grouped together. The coefficient similarities are updated as follows:

$$SM_{(1,2),(3,5)} = \frac{SM_{13}+SM_{15}+SM_{23}+SM_{25}}{2\times2} = 0.53, \qquad SM_{(3,5),4} = \frac{SM_{34}+SM_{54}}{2\times1} = 0.535,$$

$$SM_{(3,5),6} = \frac{SM_{36} + SM_{56}}{2 \times 1} = 0.59.$$

Table 3.3 shows the second step in grouping variants.

Table 3.3 Similarity coefficient of variants after second grouping

	3,5	4	6
1,2	0.53	0.515	0.45
3,5		0.535	0.59
4			0.49

This procedure continues until all variants are placed in one group. Having grouped the variants, the dendrogram can be constructed accordingly as illustrated in Figure 3.4.

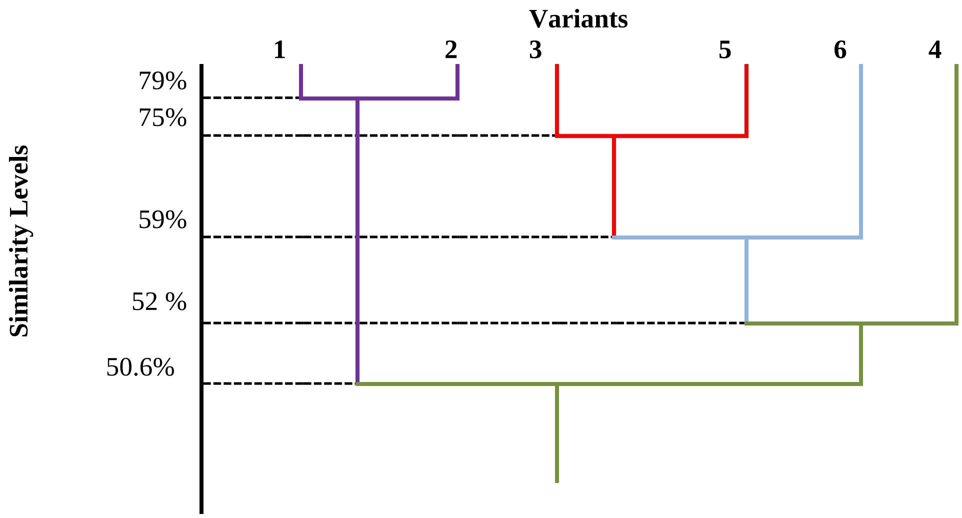


Figure 3.4 The dendrogram illustration of the example

The dendrogram is a good representation of variants grouping and the similarity value at each level.

3.6 Sequencing Variants: An Application of the Proposed Method

As discussed before, one of the applications of grouping variants is to benefit from their similarity in order to sequence them for assembly/production as illustrated by the powertrain part variants shown in Figure 3.5.

Figure 3.5 Axle and transmission part variants of powertrain (http://seissenschmidt.com/en/company/)

When all of these variants are produced in a company, one of the main tasks is scheduling the production in a way that minimizes changeover time. If there is no priority among variants, one of the best ways to schedule for production is to sequence them based on their similarity. As a result, the setup changes between variants will be minimized and throughput will be improved.

It is noteworthy to mention that sequencing is not the main target of proposed similarity coefficient and it is only one of its potential applications. The main application is for those manufacturing systems such as cellular manufacturing system where there are number of cells or production/assembly lines each of which is responsible for processing a group of part/product variants. In such case, by grouping similar part/product variants and assigning each group to a different cell/line, changeover time can be decreased and utilization can be improved.

Despite of not being the main application, the proposed similarity coefficient has potential advantages for sequencing variants. Placing variants with similar operations flow next to each other in the sequence can decrease the changeover time spent in modifying the connections between stations. Placing variants with high number of common operations (second criterion) next to each other can decrease the setup changes (i.e. tools, fixtures, program, or adding/removing machines) in stations. Placing variants in the sequence based on production volume similarity can also help in either improving utilization or decreasing the number of times machines are added or removed.

The obtained sequence serves as a better starting sequence which is not necessarily optimal that can reduce changeover time and hence, improve utilization of the system. Production planners can benefit from the obtained sequence based on similarity coefficients and modify it based on other factors such as due dates, processing times, job priorities and so forth.

Two assumptions are considered here: 1) one type of variants (or a batch of a typical variant) is processed at a time and when it is done, the second type is started. This assumption is logical as many manufacturing systems first produce the total demand of one variant followed by another variant. 2) If two consecutive variants in a production sequence have a common operation, the setup change in the station is insignificant. This is reasonable as it was initially assumed that all variants belong to a family and hence share many similarities.

Quantitative analysis:

At first, it is noteworthy to mention that considering time is out of scope of the present chapter. However, processing time is implicitly improved when grouping product variants according to the presented similarity approach. It is our expectation that the obtained sequence would serve as a better starting point (and not necessarily an optimal sequence) for planners which requires modification by considering other manufacturing aspects. Based on explanations regarding the advantages of the three grouping criteria, it is assumed that the integrated similarity coefficient corresponds to the combination of changeover time and utilization in a way that the higher value of the coefficient represents less changeover and better utilization. This is used as the objective function here. That is, to calculate the objective function for a given sequence, the coefficient value of each two adjacent variants is calculated and added to the coefficient values of other adjacent variants.

In order to find an exact solution based on the aforementioned assumptions, a mixed integer programing (MIP) model is developed. This exact solution is only based on the aforementioned assumptions and it does not guarantee the optimality without considering other manufacturing aspects.

The following parameters and variables have been applied in the model.

Parameters:

n Total numbers of variants (and positions)

SM_{ij} Integrated similarity coefficient value between Variants *i* and *j*

Variables:

$$X_{ij} = \begin{cases} 1 & \text{If Variant } j \text{ is placed immediately after Variant } i \text{ } in\ the\ sequence \\ 0 & \text{Otherwise} \end{cases}$$

(3.18)

$$Y_{ik} = \begin{cases} 1 & \text{If Variant } i \text{ is placed in } k^{th} \text{ position of the sequence} \\ 0 & \text{Otherwise} \end{cases}$$

(3.19)

$$Z_{ijk} = \begin{cases} 1 & \text{If Variant } i \text{ is placed in } k^{th} \text{ position immediately before Variant } j \text{ in the sequence} \\ 0 & \text{Otherwise} \end{cases}$$

(3.20)

The linear mathematical model is to:

$$Maximize \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij} \,.\, SM_{ij} \qquad (3.21)$$

Subject to:

$$Z_{ijk} \leq \frac{Y_{ik} + Y_{j,k+1}}{2} \qquad \forall i, j, k = 1,2, \dots, n \,, \qquad \forall k = 1,2, \dots, n-1 \qquad (3.22)$$

$$Y_{ik} + Y_{j,k+1} - 1 \leq Z_{ijk} \qquad \forall i,j,k = 1,2,\dots,n \qquad (3.23)$$

$$X_{ij} = \sum_{k=1}^{n} Z_{ijk} \qquad \forall i,j = 1,2,\dots,n \qquad (3.24)$$

$$\sum_{i=1}^{n} Y_{ik} = 1 \qquad \forall k = 1,2,\dots,n \quad (3.25)$$

$$\sum_{k=1}^{n} Y_{ik} = 1 \qquad \forall i = 1,2,\dots,n \quad (3.26)$$

$$X_{ij} \in \{0,1\} \qquad \forall i,j = 1,2,\dots,n \qquad (3.27)$$

$$Y_{ik} \in \{0,1\} \qquad \forall i,k = 1,2,\dots,n \qquad (3.28)$$

$$Z_{ijk} \in \{0,1\} \qquad \forall i,j,k = 1,2,\dots,n \qquad (3.29)$$

Equation (3.21), the objective function of the problem, calculates the integrated similarity coefficient of each two adjacent variants in the sequence and tries to maximize it. Equations (3.22) to (3.24) indicate whether or not variant *i* is placed immediately before variant *j* in the sequence. Equation (3.25) guarantees that each position of the sequence is occupied by only one variant. Equation (3.26) ensures that each variant is placed in only on one position of the sequence. Finally, Equations (3.27) to (3.29) indicate that variables X_{ij}, Y_{ik} and Z_{ijk} are binary.

GAMS optimization software was implemented to solve the model using CPLEX 12.0.1 MILP solver. Considering the illustrative example, the sequence obtained by GAMS is: 1→2→4→5→3→6 with the value of 3.30 as the objective function (0.79+0.55+0.55+0.75+0.66=3.30).

Although the presented mathematical model is a good approach to find the sequence, it is able to find the solution only for small to medium size problems. For large size problems the model solution time would not be reasonable. For that reason, the proposed sequencing approach in previous chapter is applied here. This sequencing approach is able to find solutions in a negligible time as it does not involve any constraint or local search. This method is based on the obtained similarity coefficient and using the resulting dendrogram.

The procedure for sequencing variants begins by placing two variants next to each other in the sequence with the highest level of similarity coefficient in the obtained dendrogram. Then, the second level of similarity in the dendrogram is considered. In this level, either two different variants are grouped together or one new variant is added to the pair of variants which have already been grouped in the previous level. In the former case, two new variants, similarly, are placed next to each other in the production sequence. In the latter case, the similarity coefficient between this new variant and each of the two previously grouped variants are compared to each other. The higher value determines where the new variant should be placed in the production schedule. Proceeding to lower levels of dendrogram, two groups of variants may be further grouped together. In this case, only the similarity coefficient between extreme variants is checked. The extreme variants are those which have been placed at the start or end of their own groups. This procedure is repeated until all the variants are placed in the sequence.

According to the dendrogram in Figure 3.4 and the first similarity level (79%), Variants 1 and 2 should be placed next to each other in the sequence. Similarly, based on Level 2 of the dendrogram, Variants 3 and 5 are placed next to each other. In Level 3, Variant 6 is added to the group of Variants 3 and 5. Following the procedure, SM_{36} should be compared with SM_{56} and based on the higher value, the position of Variant 6 with regard to Variants 3 and 5 is determined. Comparing the values leads to placing Variant 6 next to Variant 3 in the sequence. Therefore, the best sequence of these three variants is either 6→3→5 or 5→3→6. Considering Level 4, Variant 4 is added to the pre-grouped Variants 3, 5 and 6. Based on the procedure, the similarity of Variant 4 is compared with respect to the extreme members of these pre-grouped variants which are Variants 6 and 5. Comparing SM_{46} with SM_{54} leads to placing Variant 4 next to Variant 5 in the sequence. Therefore the best sequence of these variants, based on the procedure, is either 6→3→5→4 or 4→5→3→6. At the last level, two groups, Group 1 including Variants 1 and 2 and Group 2 including Variants 3, 4, 5 and 6 are further grouped together. As mentioned before, only similarity coefficients for extreme variants of each group are compared. In Group 1, both Variants 1 and 2 are extreme variants. In Group 2, Variants 4 and 6 are considered extreme variants. Thus, $SM_{14}, SM_{16}, SM_{24}$, and SM_{26} are compared with each other. Since SM_{24} has the highest value, these two groups are placed next to each other in a way that Variants 2 and 4 are adjacent. Therefore, the best sequence of all variants, based on their similarity, would be either 1→2→4→5→3→6 OR 6→3→5→4→2→1. Since the objective function is based on reducing setup costs, both sequences according to this algorithm are considered the best.

As we can see, applying the proposed sequencing approach for the illustrative example yields the same objective value and sequence which indicates the efficiency of the proposed method. It should be noted that the solution obtained from either developed mathematical model or the proposed sequencing approach serve as a good preliminary sequence which can be modified by planners based on other aspects such as processing time, due dates and job priorities.

3.7 Case Study

A company produces different parts for locomotive and automotive industries. The company operates using several shifts and can also use the weekends if needed. The operations are mainly machining operations such as gun drilling, turning, and milling. For confidentiality reasons, the name of the company and variants are not revealed.

Seven part variants are considered. Figure 3.6 illustrates the representative variants. Note that some internal features are not displayed in the images but are embedded in the parts and presented in Figure 3.7.

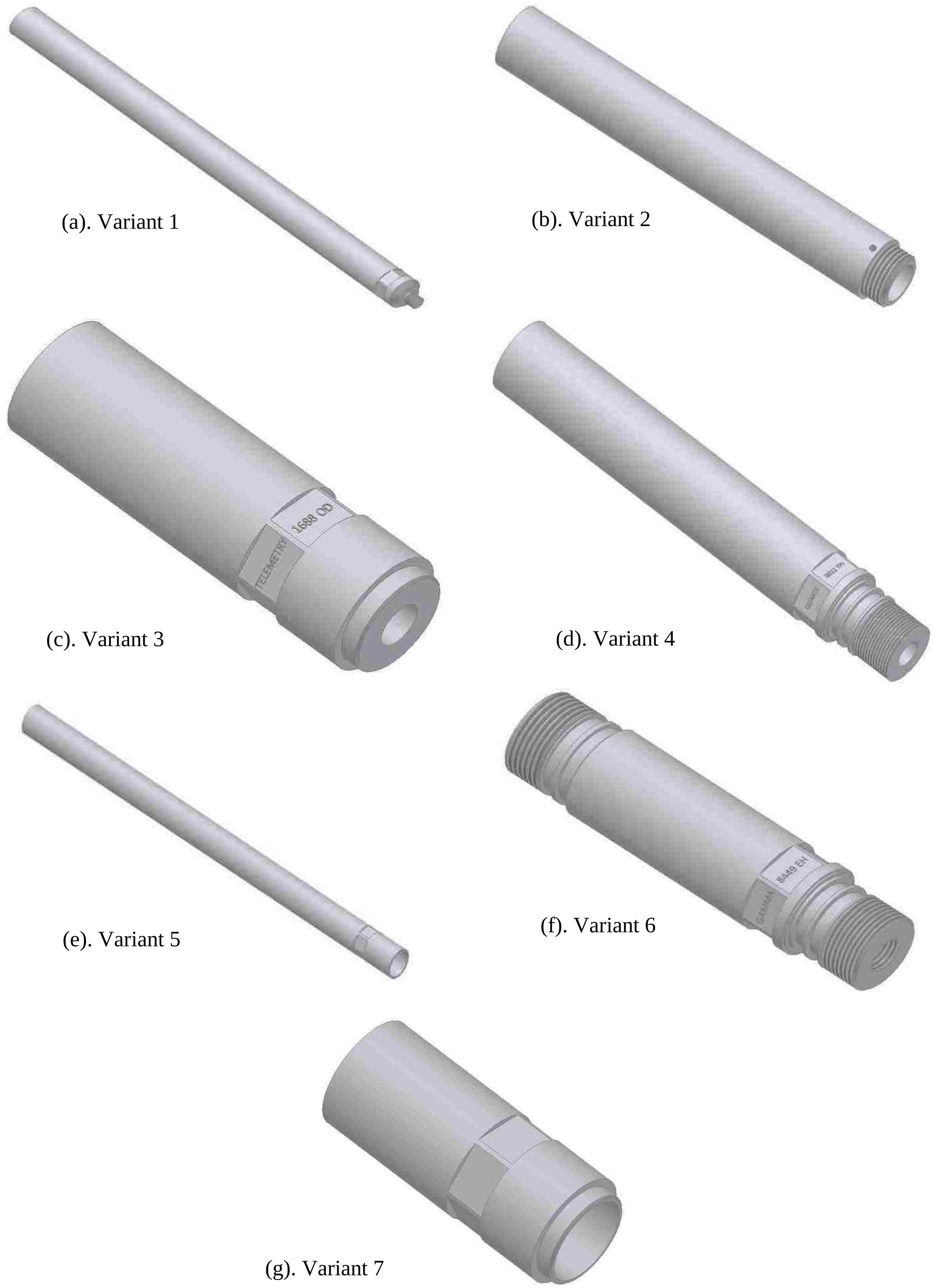


Figure 3.6 Representative part variants (case study)

Table 3.4 shows the demand required for each variant.

Table 3.4 Volume for each variant of the case study

Variants	Volume
1	25
2	200
3	20
4	50
5	40
6	30
7	100

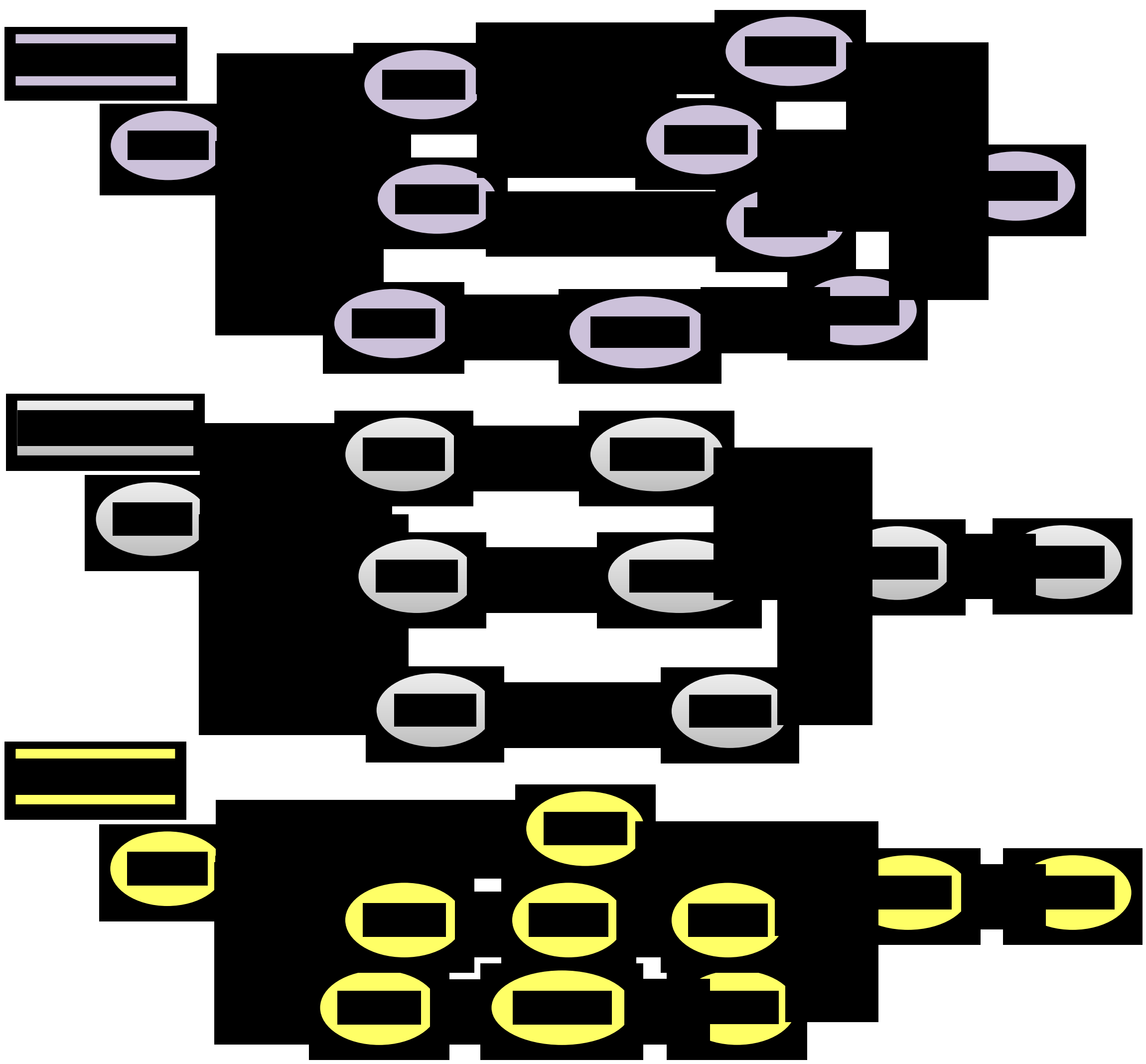
Table 3.5 lists the abbreviations used to demonstrate the corresponding operation precedence graphs of the variants with the complete name of each operation. For clarity purposes, explanation is provided for some operations.

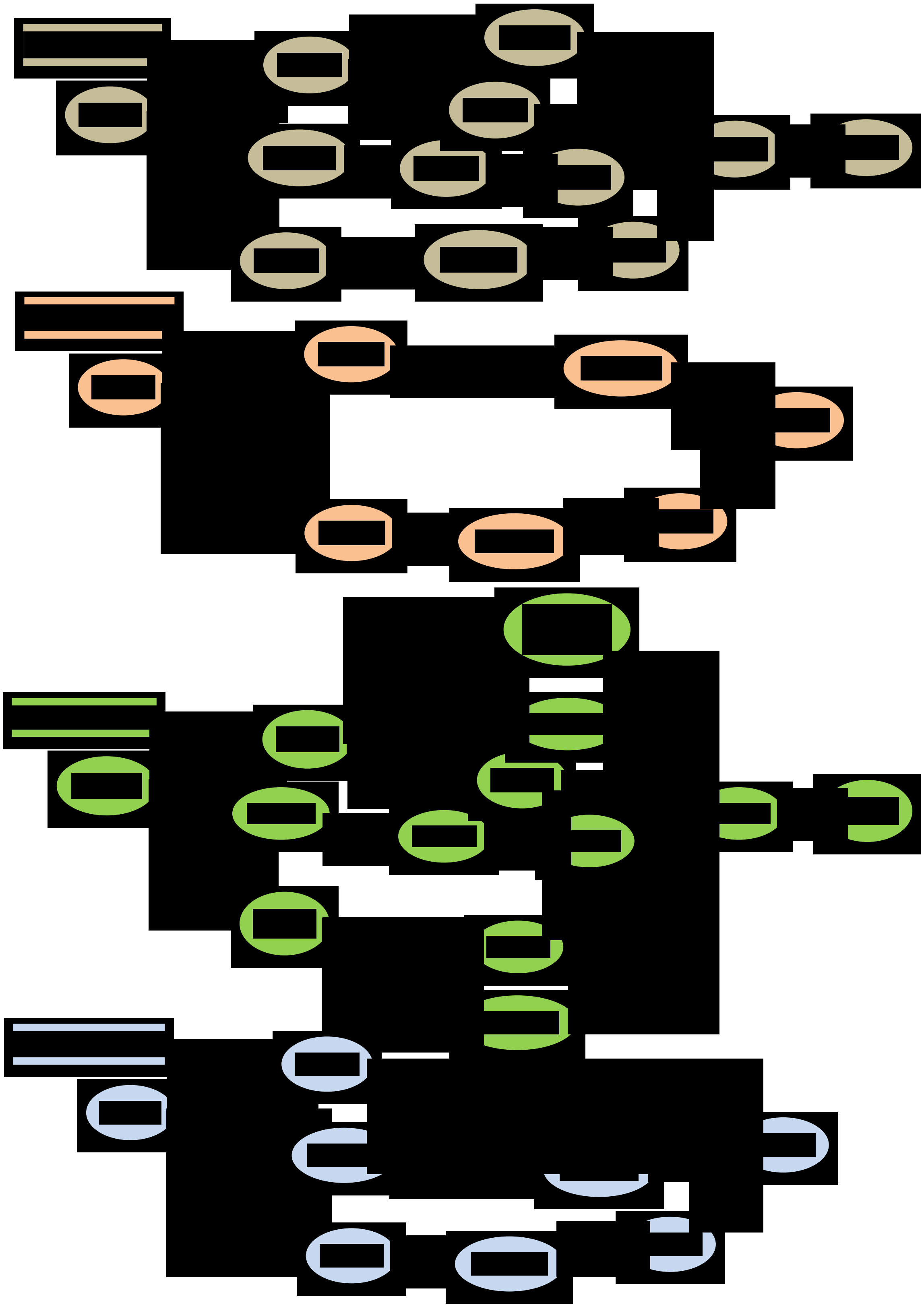
Table 3.5 Abbreviation and complete names of each operation

Abbreviation	Complete name of the operation and description
SW	Saw Cutting
CT	Cylindrical Turning
CTF	Cylindrical Turning-Finishing This operation is for cleaning the outer surface of a part (finishing cuts).
GD	Gun Drilling This operation is applied for deep drilling.
PM	Plain Milling
ExT	External Threading This operation is applied for making thread on the surface of a part.
InT	Internal Threading This operation is applied for making thread into a hole made by gun driller.
InCT	Internal Cylindrical Turning
UC	Under Cutting
InUC	Internal Under Cutting
EDM	Electric Discharge Machining This operation is used to make small and accurate holes on the surface of a part.
LM	Laser Marking Depends on requirements, some parts need to be marked by logo, codes, names, etc.

HT	Heat Treatment This operation is applied for hardening parts. Not all the variants need this operation.
QC	Quality Control This operation is for checking and comparing the actual dimensions of the finalized part with the specifications.

Figure 3.7 demonstrates the representing operation precedence graphs for each of the seven variants.





[redacted]

Three similarity criterion described in previous sections are implemented to find the similarity values between each pair of variants.

The operations flow based similarity between each pair of variants is calculated as follows:

$$SoF_{12} = \frac{3+1+1+1+0+1+0+1+0+0+0+0}{3+1+2+1+1+1+1+1+1+1+1+4} = \frac{8}{18} = 0.444$$

For the other pairs of variants, the following values are obtained:

$SoF_{13} = 0.600,$ $SoF_{14} = 0.667,$ $SoF_{15} = 0.823,$ $SoF_{16} = 0.500,$

$SoF_{17} = 0.800,$ $SoF_{23} = 0.625,$ $SoF_{24} = 0.700,$ $SoF_{25} = 0.363,$

$SoF_{26} = 0.667,$ $SoF_{27} = 0.429,$ $SoF_{34} = 0.833,$ $SoF_{35} = 0.667,$

$SoF_{36} = 0.615,$ $SoF_{37} = 0.667,$ $SoF_{45} = 0.667,$ $SoF_{46} = 0.800,$

$SoF_{47} = 0.632,$ $SoF_{56} = 0.429,$ $SoF_{57} = 0.875,$ $SoF_{67} = 0.421.$

Next, Jaccard's similarity is considered to find the operations similarity between each pair of variants. It is assumed that all operations have the same level of importance.

$C_{12} = 0.583,$ $C_{13} = 0.667,$ $C_{14} = 0.833,$ $C_{15} = 0.600,$

$C_{16} = 0.643,$ $C_{17} = 0.800,$ $C_{23} = 0.583,$ $C_{24} = 0.615,$

$C_{25} = 0.364,$ $C_{26} = 0.571,$ $C_{27} = 0.545,$ $C_{34} = 0.833$,

$C_{35} = 0.600,$ $C_{36} = 0.643,$ $C_{37} = 0.800,$ $C_{45} = 0.500,$

$C_{46} = 0.786,$ $C_{47} = 0.667,$ $C_{56} = 0.357,$ $C_{57} = 0.750,$ $C_{67} = 0.500.$

It is obvious that grouping the variants will change based on the selected criterion. For instance, consider Variants 1, 4 and 5; based on the operations flow similarity, Variants 1 and 5 should be grouped together whereas based on Jaccard's similarity, Variants 1 and 4 should be grouped together. Figures 3.6 and 3.7 show that Variant 5 is basically a sub-set of Variant 1 and for that reason their operations flow similarity is high. On the other hand, although Variants 1 and 4 have some different operations and some different flows, the number of common operations between these variants is high. Therefore, it is the planner's decision to consider which criterion for grouping is the best for the system.

The following results are obtained, considering production volume similarity criterion and using Equation (3.14). It is assumed that in Equation (3.14) W_1=W_2=0.5.

$DRD_{12} = 0.076,\quad DRD_{13} = 0.886,\quad DRD_{14} = 0.681,\quad DRD_{15} = 0.771,$

$DRD_{16} = 0.903,\quad DRD_{17} = 0.417,\quad DRD_{23} = 0.05,\quad DRD_{24} = 0.208,$

$DRD_{25} = 0.156,\quad DRD_{26} = 0.103,\quad DRD_{27} = 0.472,\quad DRD_{34} = 0.617\,,$

$DRD_{35} = 0.694,\quad DRD_{36} = 0.806,\quad DRD_{37} = 0.378,\quad DRD_{45} = 0.872,$

$DRD_{46} = 0.744,\quad DRD_{47} = 0.533,\quad DRD_{56} = 0.847,\quad DRD_{57} = 0.533,\quad DRD_{67} = 0.456.$

W_{SoF}, W_C, and W_{DRD} were assumed to be 0.4, 0.4 and 0.2 respectively. Table 3.6 shows the final similarity coefficient between each pair of variants based on these importance weights.

Table 3.6 Similarity coefficient between each pair of variants (case study)

	2	3	4	5	6	7
1	0.426	0.684	0.736	0.723	0.678	0.723
2		0.493	0.568	0.322	0.516	0.484
3			0.79	0.646	0.664	0.662
4				0.641	0.783	0.626
5					0.484	0.757
6						0.46

Variants are grouped according to ALC algorithm. Figure 3.8 demonstrates the corresponding dendrogram after grouping. This figure provides good insight about how the variants should be grouped.

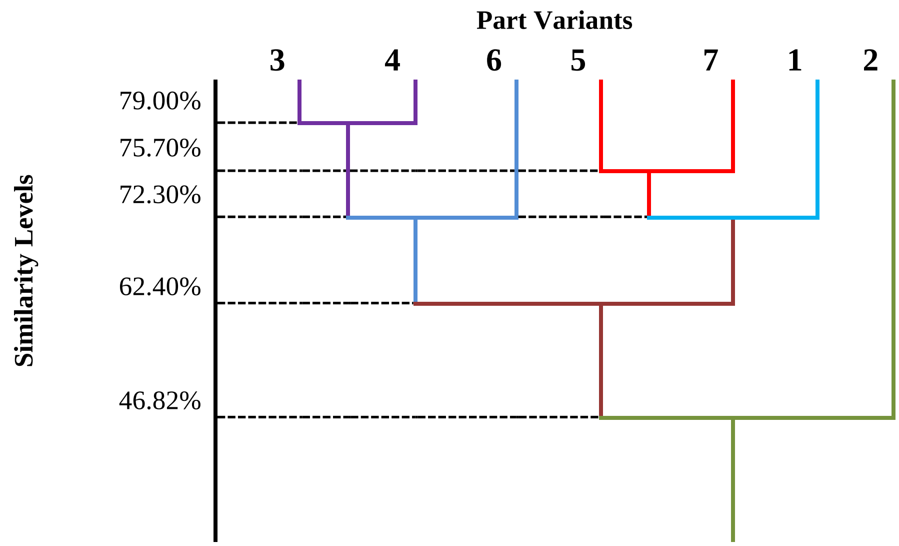


Figure 3.8 The dendrogram illustration of the case study

The assumptions mentioned in previous section are also considered here and the sequence obtained from the proposed sequencing approach is compared with sequences obtained using five other approaches to find out which approach yields higher value with respect to the integrated similarity coefficient.

Based on the proposed sequencing approach, the sequence would be either 2→6→4→3→1→5→7 OR 7→5→1→3→4→6→2. Now, the integrated similarity coefficient values of adjacent pairs are summed together as follows to calculate the objective value: $SM_{26} + SM_{64} + SM_{43} + SM_{31} + SM_{15} + SM_{57} = 0.516 + 0.783 + 0.79 + 0.684 + 0.723 + 0.757 = 4.253$.

Five other sequencing approaches are considered to compare their sequence with the sequence obtained from the proposed sequencing approach.

The first approach is the mathematical model developed in Section 3.6. Using GAMS software, it yields the following sequence: 2→6→4→3→1→7→5 with the objective value of 4.253. The only difference between two sequences is the positions of Variants 5 and 7 and because Variant 1 has the same similarity coefficient with both Variants 7 and 5 (0.723), the values of the objective function for both approaches are equal. This confirms the accuracy of the proposed sequencing approach.

The second approach is to sequence the variants based on their appearance so that the similar variants (in terms of appearance) are placed next to each other in the sequence. The final sequence based on this approach and by considering Figure 3.6 would approximately be: 2→5→1→7→3→4→6 and the value of the objective function would be: 0.322+0.723+0.723+0.662+0.79+0.783=4.003.

The third approach is to sequence the variants based on only Jaccard's similarity coefficient. Considering the value obtained from Jaccard's similarity coefficient, variants with higher values are placed next to each other in the sequence. The obtained sequence using Jaccard's similarity coefficient would be 1→7→5→3→4→6→2 with objective value of 0.723 + 0.757 + 0.646 + 0.79 + 0.783 + 0.516 = 4.215.

The fourth approach is to sequence the variants based on the Shortest Processing Time First (SPTF) policy. In SPTF, variants with less processing times are processed first (Baker and College, 2008). Since the processing times are not considered here, the number of operations for each variant and whether or not they require some time-consuming processes such as heat treatment can be used as a rough indication of processing time. The final sequence obtained from this approach would roughly be 5→7→1→2→3→4→6. The result is: $0.757 + 0.723 + 0.426 + 0.493 + 0.79 + 0.783 = 3.972$.

The last approach is a more comprehensive approach. In this approach, all possible ways of sequencing the seven product variants which is 7!=5040 were considered and the objective function was calculated for all cases. Based on this approach, there are four sequences yielding the highest objective value (4.253): 2→6→4→3→1→7→5, 2→6→4→3→1→5→7, 5→7→1→3→4→6→2 and 7→5→1→3→4→6→2. There are two sequences yielding the lowest objective value (3.002): 4→7→6→5→2→1→3 and 3→1→2→5→6→7→4. The sequence that yields the median value (3.677) is: 2→5→3→4→1→7→6.

Comparing the above results reveals that the proposed sequencing approach yields the highest value in terms of the integrated similarity coefficient. It can also be concluded that on average the sequence obtained from the proposed sequencing approach, in this case study, improves the objective function (combination of changeover time and utilization) by 15.7% ($\frac{4.253-3.677}{3.677}$). In addition, a random sequence was generated and the following sequence was obtained: 5→4→6→3→2→1→7 with the objective value of 3.73. It also shows that the proposed sequencing approach has a better performance over a random generated sequence.

Hence, it can be concluded that if there is only one production line, the sequence of 2→6→4→3→1→5→7 is a good preliminary parts/products sequence with respect to changeover time and utilization.

It is noteworthy to mention that the prosed sequencing approach is based on the values obtained from the integrated similarity coefficient and is not affected by the criteria themselves. In other words, if more criteria are added or some criteria are removed from the integrated similarity coefficient, it does not affect the way the proposed sequencing approach works. For instance, if production volume similarity is disregarded from the integrated similarity coefficient, the procedure of sequencing based on the proposed sequencing approach would be the same as before; however, the final sequence may not be the same as before as the similarity values have changed.

3.8 Summary

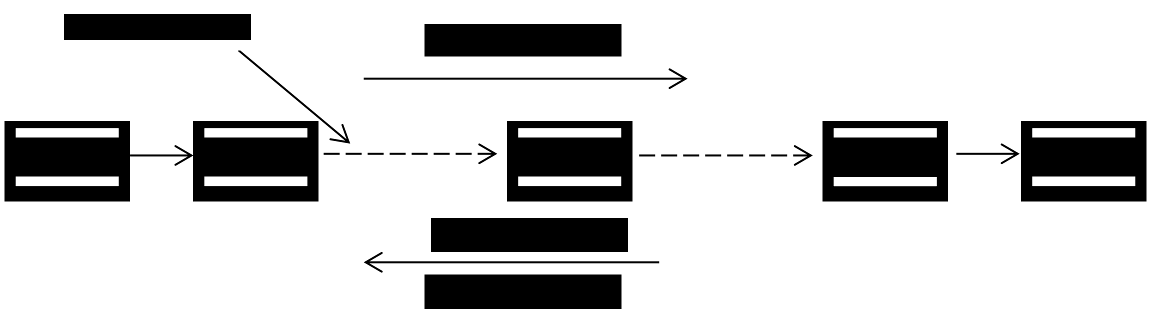
In this chapter, grouping part/product variants based on operations flow similarity, operations similarity and production volume similarity was considered. Such grouping has significant impact on improving productivity and decreasing setup changes. For operations flow based similarity, no method has been developed in literature for grouping variants with network structure of operations sequence. A novel method inspired by analysis used in biology was proposed. In addition, since some flows of operations may be more important than other flows in a given variant, a similarity coefficient taking relative importance was proposed. The most common method in the literature called Jaccard's similarity was applied for developing the operations similarity criterion. Also an extension to this method was proposed for use when operations are not equally important. A new coefficient was presented for production volume similarity, which obviates the drawback of the production volume similarity coefficient developed by Galan et al. (2007b). These three criteria were integrated to yield a more comprehensive similarity coefficient using weighted sum. This provides flexibility to change the importance weight of each mentioned similarity coefficient as needed. Variants were further grouped using the average linkage clustering (ALC) algorithm. Grouping variants based on these three criteria in the manufacturing systems having different cells/lines for different variants can reduce changeover time and ease reconfiguration. As a result, it is also expected to improve productivity and utilization of the system. A secondary application of grouping is sequencing variants. This application was considered with some assumptions and a mathematical model was developed and its results were compared with those obtained from the proposed sequencing method. The results demonstrate the accuracy of the proposed sequencing approach by finding optimum solutions.

Chapter 4

Optimal Assignments of Facilities with backtracking of Product Variants Having Networked Operations Sequence

4.1 Introduction

After grouping product variants based on their similarities and assigning them to different cells, the next step is to optimize flow of product variants within each cell so that the total throughput is maximized. There are generally four types of movements in manufacturing systems: (1) forward flow, (2) repeat operation, (3) by-passing and (4) reverse flow (backtracking) (Aneke and Carrie, 1986). In many industries, various stations are connected using conveyors. Examples of such industries include, but are not limited to, automotive assembly lines, sheet metal production, label stickers production, and printed circuit board (PCB) assembly. One strategy for improving throughput in such industries is to ensure that the products flow downstream and machines/stations are located to maximize forward products flow, hence, minimizing backtracking. Heragu and Kusiak (1988) illustrated the concept of forward and backward flows by considering *m* machines in *m* locations along a straight line as shown in Figure 4.1.



[redacted]

Minimizing backtracking has several advantages; it reduces the workload of conveyors/transporters, improves the availability of jobs in downstream leading to reduction of machine idle time and enhancing utilization of the system and consequently, increasing the total system throughput. Scheduling is also simplified by minimizing backtracking (Sarker et al., 1995).

This chapter also considers a family of product variants with networked operations sequence where some of operations do not have precedence constraints. There are also *m* locations with *m* available machines each of which is capable of performing one or more type of operations. The objectives are to: 1) assign the machines to the available locations, 2) assign operations to available machines, and 3) determine the final sequence of performing operations for each product variant which minimizes the total backtracking distance. The final sequence of operations refers to the optimal sequence of the operations of a given product variant. As mentioned earlier, product variants have networked structures and there is sequence flexibility in performing some operations. Therefore, different ways of processing operations of a product variant may exist.

Minimizing backtracking has been addressed in literature under two different categories: i) for generalized flow line (GFL) problems, and ii) for intracellular machining sequence in cellular manufacturing systems. However, there is no work in backtracking literature considering product variants with networked structures taking into consideration machine selection for each operation based on machine capability.

A novel mathematical model is developed to solve the outlined problem. A non-linear model is first developed then reformulated as a mixed integer programing (MIP) model. A case study of engine blocks is used to demonstrate its application to an industrial setting. The obtained results reveal that backtracking minimization can remarkably increase the total throughput (7.79% improvement in the case study).

4.2 Literature Review

Backtracking of jobs has been considered in: (1) generalized flow line (GFL) problems (Sarker et al., 1991), and (2) cell formation problems. Sarker et al. (1995) considered backtracking of jobs by assigning *M* machines to *M* locations along a linear track. Since the problem is NP-hard especially for large sized problems, a multi-pass heuristic was proposed and improved by applying depth-first insertion heuristic. A simulation model was applied to examine the performance of the system by reducing backtracking.

Sodhi and Sarker (2003) applied the backtracking concept to flexible manufacturing systems. They assumed that all available machines are identical CNCs, however, by adding tool magazines they can perform more operations and consequently less number of stations would be required. A mathematical model was developed to assign tool magazines to different locations in order to minimize three types of costs: total backtracking cost, changeover cost and production cost.

Gong et al. (1999) considered generalized flow line (GFL) problem to minimize total backtracking of jobs by assigning *M* machines to *M* locations along a linear track. A genetic algorithm was proposed and compared with existing methods. It was able to find good quality solutions in acceptable computing time.

More recent works have applied backtracking in cellular manufacturing systems. Chang et al. (2013) developed two mathematical models for cell formation problem. The first model was to solve the cell formation and cell layout problems whereas the second model was developed to determine the sequence of machines in each cell so that forward flow is maximized and backtracking is minimized. Mahdavi et al. (2013) developed an integrated mathematical model to simultaneously solve cell formation and cell layout problems. It was assumed that the cell layouts are not linear; however, there are pre-specified places where cells can be located. The machine layout problem within each cell was also considered to minimize backtracking cost. Two benchmark examples in literature were considered to study the efficiency of the developed model. Forghani et al. (2015) considered the same problem with some new features such as multi-row intra-cell layout, rectangular shape layouts and aisle distance. One of the objectives was to arrange the machines within each cell so that material handling cost is minimized necessitating backtracking minimization.

Golmohammadi et al. (2016) considered a cell formation problem to minimize total moving costs consisting of cost of intra-cell movement, cost of movement between cells and cost of backtracking under dynamic conditions. . A genetic algorithm was proposed to solve the problem and the results were compared with a developed mathematical model. Computational results showed the efficiency of the proposed algorithm.

There are some other works in literature considering backtracking/backward movement (e.g. Lee and Chiang, 2002; Li and Li, 2007; Mahdavi et al., 2008; Paydar et al., 2010; Hogg, 2012; Davies et al., 2013).

Based on the above review, there is no work in the literature dealing with products flow backtracking particularly considering product variants with networked structures and also, regarding machine capabilities in performing different operations and hence, machine selection for each operation.

4.3 Backtracking Problem Description

A number of part/product variants belong to a family; hence, they share some similarities. Each individual part/product variant has different demand and requires a number of operations to be finalized.

There are also *m* machines which should be placed in *m* locations. Each machine has its own capability and is able to perform different types of operations. The operations required for part/product variants follow precedence relationships called “operations sequence”. The term “operation” in the present work refers to both assembling and fabricating/manufacturing processes such as machining, molding, welding, heat treating and quality control. The structure of operations sequence of product variants is generally divided to two categories: serial and networked. In serial operations sequence, the order of performing operations of a product variant is fixed and rigid. However, in networked structure, some operations have sequence flexibility to be carried out before or after other operations. The detailed information about serial and networked operations sequences have been provided in Chapter 3.

Product variants with networked structures are considered in the present chapter. However, to finalize a product variant with a networked structure, only one sequence of performing operations among the alternative sequences should be determined based on the considered objective. There might be more than one sequence yielding the same optimum value for the objective function. In such case, performing the operations based on any of these sequences is acceptable.

It is required to select an appropriate machine for each operation and also to assign each machine to an appropriate location within the layout and lastly, to determine the final sequence of operations for each part/product variant in order to minimize the total backtracking distance. Therefore, three decision variables are used in the model formulation.

Backtracking occurs when a product variant requires going upstream after performing one operation at one station to another operation in another station. Backtracking between two stations is determined based on the distance between them.

4.4 Backtracking Mathematical Model

A mathematical model is developed to assign operations to machines and machines to the locations and also to determine the final sequence of operations for each product variant so that total backtracking distance is minimized. The following is a description of the parameters and variables used in the model.

Parameters:

S: total number of different operations among all variants

n_r: total number of operations for part r

m: total number of machines/locations

q: total number of part/product variants

V_r: volume required for part/product r

$$\mathrm{C_{ki}} = \begin{cases} 1 & \text{If machine k is capable of performing operation i} \\ 0 & \text{Otherwise} \end{cases}$$

$$\mathrm{P_{ri}} = \begin{cases} 1 & \text{If product r requires operation i} \\ 0 & \text{Otherwise} \end{cases}$$

$$\mathrm{O_{ijr}} = \begin{cases} 1 & \text{If Operation j is performed after operation i in product r} \\ 0 & \text{Otherwise} \end{cases}$$

$$\mathrm{d_{hh'}} = \text{backtracking distance based on going downstream from location h to location } h'$$

$$= \begin{cases} \text{total distance between two locations} & \text{upstream flow} \\ 0 & \text{downstream flow} \end{cases}$$

Variables:

$$L_{hk} = \begin{cases} 1 & \text{If machine k is assigned to location h} \\ 0 & \text{Otherwise} \end{cases}$$

$$X_{ik} = \begin{cases} 1 & \text{If operation i is assigned to machine k} \\ 0 & \text{Otherwise} \end{cases}$$

$$Y_{ibr} =$$

$$\begin{cases} 1 & \text{If operation i is performed in } b^{th} \text{ position of the final operations sequence of product r} \\ 0 & \text{Otherwise} \end{cases}$$

Using the listed parameters and variables, a non-linear integer programing model is developed as follows:

$$Minimize \sum_{i=1}^{s} \sum_{j=1}^{s} \sum_{b=1}^{n_r-1} \sum_{r=1}^{q} \sum_{k=1}^{m} \sum_{h=1}^{m} \sum_{k'=1}^{m} \sum_{h'=1}^{m} Y_{ibr} . Y_{j,b+1,r} . X_{ik} . X_{jk'} . L_{hk} . L_{h'k'} . d_{hh'} . V_r \quad (4.1)$$

Subject to:

$$\sum_{k=1}^{m} X_{ik} = 1 \qquad \forall i = 1,2,\dots,s \qquad (4.2)$$

$$\sum_{i=1}^{s} X_{ik} \geq 1 \qquad \forall k = 1,2,\dots,m \qquad (4.3)$$

$$X_{ik} \leq C_{ki} \qquad \forall i = 1,2,\dots,s , \forall k = 1,2,\dots,m \qquad (4.4)$$

$$\sum_{h=1}^{m} L_{hk} = 1 \qquad \forall k = 1,2,\dots,m \qquad (4.5)$$

$$\sum_{k=1}^{m} L_{hk} = 1 \qquad \forall h = 1,2,\dots,m \qquad (4.6)$$

$$\sum_{i=1}^{s} Y_{ibr} = 1 \qquad \forall r = 1,2,\dots,q, \forall b = 1,2,\dots,n_r \qquad (4.7)$$

$$\sum_{b=1}^{n_r} Y_{ibr} = P_{ri} \qquad \forall i = 1,2,\dots,s, \forall r = 1,2,\dots,q \qquad (4.8)$$

$$M(1-\mathrm{Y_{ibr}}) \geq \sum_{j=1}^{s}\sum_{e=b+1}^{n_r}(\mathrm{Y_{jer}} \, . \, \mathrm{O_{jir}}) \qquad \forall i = 1,2,\dots,s, \forall b = 1,2,\dots,n_r - 1, \forall r = 1,2,\dots,q \qquad (4.9)$$

$$\mathrm{Y_{ibr}}, \mathrm{X_{ik}}, \mathrm{L_{hk}} \in \{0,1\} \qquad \forall i = 1,2,\dots,s, \forall b = 1,2,\dots,n_r, \forall r = 1,2,\dots,q, \forall \mathrm{k}, \mathrm{h} = 1,2,\dots,m, \qquad (4.10)$$

Equation (4.1) is the objective function for minimizing the total backtracking distance. Constraint (4.2) indicates that each type of operation is assigned to only one machine. Constraint (4.3) guarantees that each machine will be responsible for at least one type of operation. Constraint (4.4) ensures that an operation is assigned to a machine provided that the machine is capable of performing that operation. Constraints (4.5) and (4.6) indicate that each machine is assigned to only one location and each location can accommodate only one machine. Constraint (4.7) guarantees that only one operation can be placed in b^{th} position of final operations sequence for each product variant. Constraint (4.8) ensures that if a particular operation is required by a given product variant, it is assigned to only one position of the final operations sequence of that variant. Constraint (4.9) guarantees that the final operations sequence of each variant complies with the operations precedence constraint of that variant. Note that parameter "*M*" in Constraint (4.9) refers to a large positive number (*Big M*). Finally, Constraint (4.10) indicates that $\mathrm{Y_{ibr}}, \mathrm{X_{ik}}$, and $\mathrm{L_{hk}}$ are binary variables. Note that Constraint (4.3) can be excluded from the model if there is no need to assign all the machines. In such case, Constraints (4.5) and (4.6) would also be modified. In this model, it is assumed that all available machines are deployed in processing operations.

The developed mathematical model is non-linear. In order to utilize linear solvers it is necessary to reformulate the model to its equivalent linear format (Mozdgir et al., 2013). The only non-linear equation in the above mathematical model is the objective function. Therefore, a new binary variable is defined to reformulate the model. The new variable is as follows:

$Z_{ijbrkhk'h'}$ is equal to one if operations i and j are placed respectively in positions b and b+1 of product r and also operations i and j are performed with machines k and k' respectively and machines k and k' are assigned to locations h and h' and it is zero otherwise; $\forall i, j = 1,2, \dots, s, \forall b = 1,2, \dots, n_r - 1, \forall r = 1,2, \dots, q, \forall \mathrm{k}, \mathrm{h}, k', h' = 1,2, \dots, m.$

The previous non-linear model is reformulated to its equivalent MIP model with a new linear objective function and three new sets of linear constraints as follows:

$$Minimize \sum_{i=1}^{s}\sum_{j=1}^{s}\sum_{b=1}^{n_r-1}\sum_{r=1}^{q}\sum_{k=1}^{m}\sum_{h=1}^{m}\sum_{k'=1}^{m}\sum_{h'=1}^{m}(\mathrm{Z}_{\mathrm{ijbrkh}k'h'} \,.\, \mathrm{d}_{\mathrm{h}h'} . \mathrm{V_r}) \qquad (4.11)$$

$$\mathrm{Z}_{\mathrm{ijbrkh}k'h'} \geq \mathrm{Y_{i,b,r}} + \mathrm{Y_{j,b+1,r}} + \mathrm{X_{ik}} + \mathrm{X}_{\mathrm{j}k'} + \mathrm{L_{hk}} + \mathrm{L}_{h'k'} - 5 \qquad \forall i, j = 1,2, \dots, s, \forall b = 1,2, \dots, n_r - 1, \forall r = 1,2, \dots, q, \qquad \forall \mathrm{k}, \mathrm{h}, k', h' = 1,2, \dots, m, \qquad (4.12)$$

$$\mathrm{Z}_{\mathrm{ijbrkh}k'h'} \leq \frac{\mathrm{Y_{i,b,r}} + \mathrm{Y_{j,b+1,r}} + \mathrm{X_{ik}} + \mathrm{X}_{\mathrm{j}k'} + \mathrm{L_{hk}} + \mathrm{L}_{h'k'}}{6} \qquad \forall i, j = 1,2, \dots, s, \forall b = 1,2, \dots, n_r - 1, \forall r = 1,2, \dots, q, \forall \mathrm{k}, \mathrm{h}, k', h' = 1,2, \dots, m, \qquad (4.13)$$

$$\mathrm{Z}_{\mathrm{ijbrkh}k'h'} \in \{0,1\} \qquad \forall i = 1,2, \dots, s, \forall b = 1,2, \dots, n_r - 1, \forall r = 1,2, \dots, q, \forall \mathrm{k}, \mathrm{h} = 1,2, \dots, m, \qquad (4.14)$$

4.5 Illustrative Example

In this section, an example is used to illustrate how the model works. Assume that there are four product variants each of which requires a set of operations. Also, there are three machines with different capabilities which should be placed in three candidate locations within the production line. Figure 4.2 shows the operations sequence of the four variants. Based on this figure, Variants 1 and 3 have networked operations sequences.

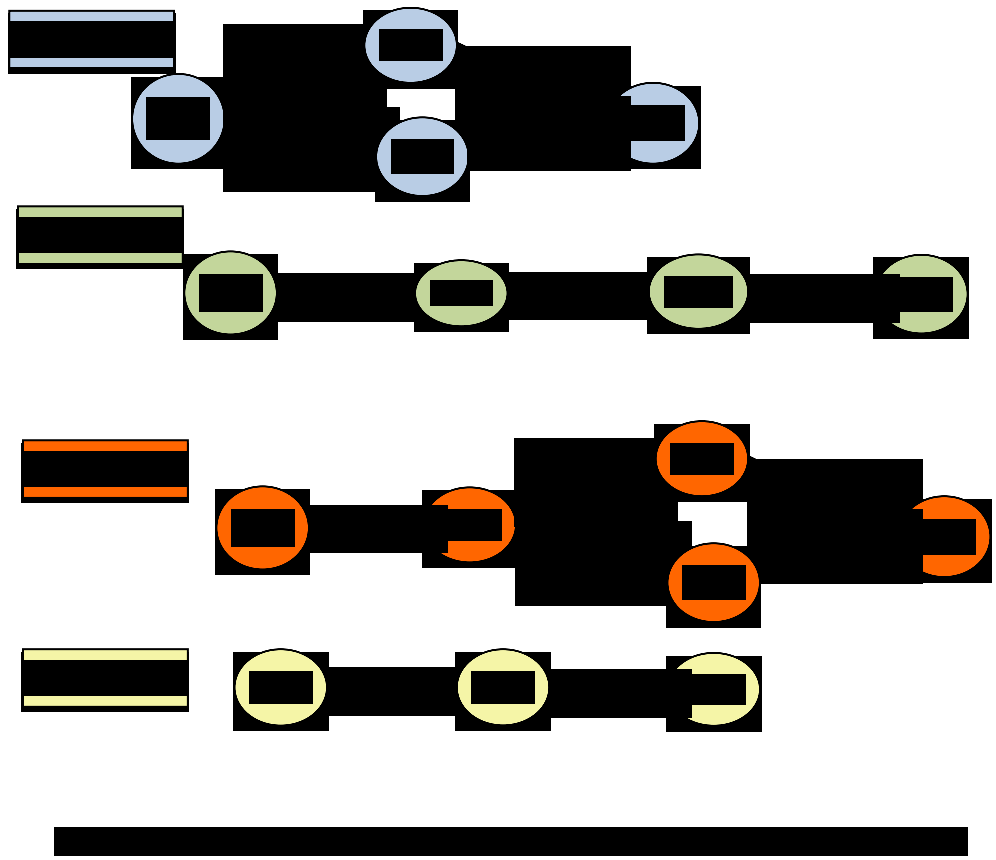


Table 4.1 shows the daily production volume required for each variant.

Table 4.1 Volume for each variant of the illustrative example

Variant	Production Volume (unit)
1	10
2	5
3	20
4	40

Table 4.2 shows the machine capability matrix in performing different operations.

Table 4.2 Machine capability matrix (illustrative example)

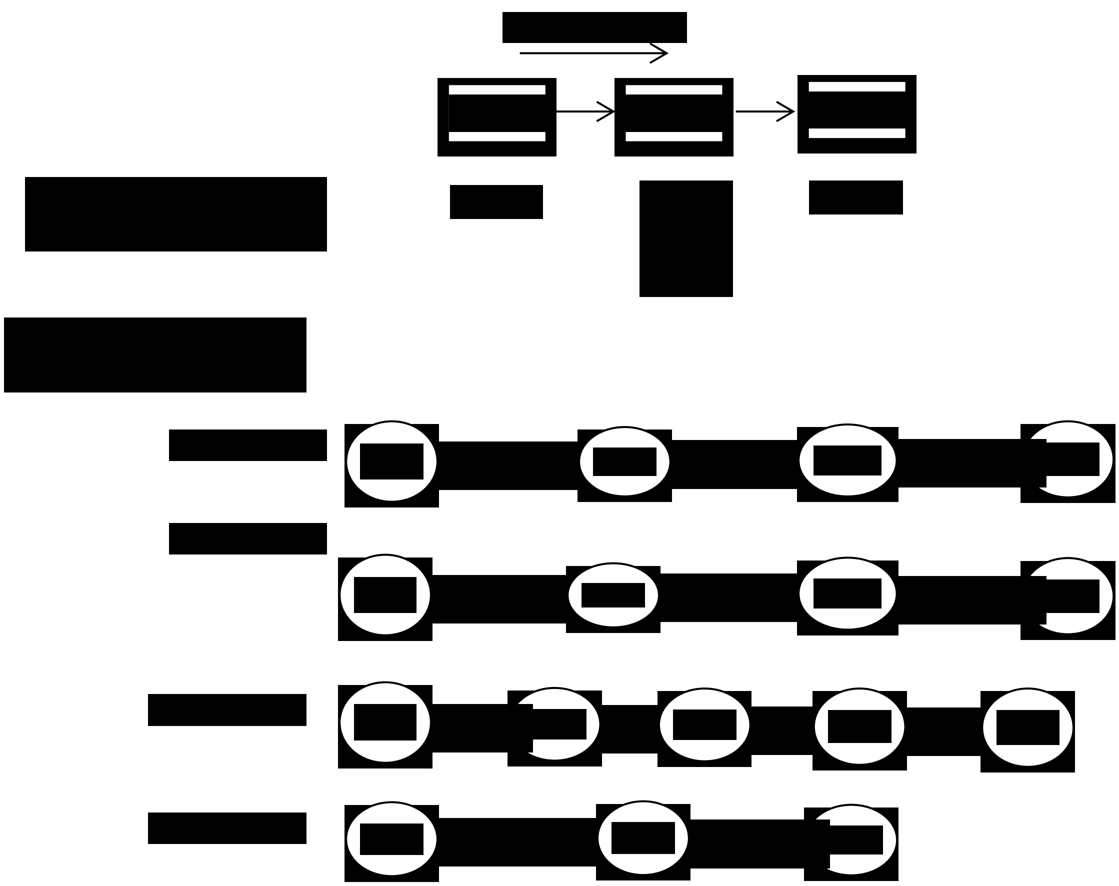
		Operation				
		O1	**O2**	**O3**	**O4**	**O5**
Machine	**M1**	0	1	1	1	0
	M2	1	0	1	0	1
	M3	0	1	0	0	0

Table 4.3 refers to the backtracking distance from one station to another. As mentioned earlier, backtracking occurs when a variant travels in upstream direction.

Table 4.3 Backtracking distance (in meters)

		Station		
		S1	**S2**	**S3**
Station	**S1**	-	-	-
	S2	5	-	-
	S3	12	7	-

GAMS optimization software with CPLEX 12.0.1 solver was used to solve the developed MIP model on a PC with 3.4 GHz Intel Core i7-4770 CPU processor under the Windows 10 operating system with 16 GB of RAM. It took 3.95 seconds to solve this example with the objective value of 325 meters as the minimum total backtracking distance. Figure 4.3 summarizes the results.



[redacted]

4.6 Numerical Experiments

In this section, the performance of the developed mathematical model in terms of solution time is examined. The system used in this experiment is a PC with 3.4 GHz Intel Core i7-4770 CPU processor under the Windows 8 operating system with 16 GB of RAM. Several numerical tests were conducted. The developed mathematical model was implemented in GAMS optimization software and solved with CPLEX 12.0.1 MILP solver.

Different sets of problems were taken into consideration to include different sizes of problems. For the number of operations *(s)*, 5, 7, 10, 15, 20, and 30 were selected while the number of variants *(q)* was set to be 2, 6, 10, 15, and 20. The number of machines/stations *(m)* was also set to be 2, 4, and 6. Since in numerical analysis, it may happen that the solutions for some cases are found much faster or much slower than the normal situation, six different replicates were solved randomly for each combination and the average time was recorded to have a better estimation of the solution time for different problem sizes. Table 4.4 summarizes the results. Since GAMS software is capable of finding small to medium problem sizes, it was terminated after 90 minutes (5400 seconds) if the optimum solution was not found. Moreover, for large sizes of problems, GAMS was not able to find any solution and it yielded the "out of memory" message.

Table 4.4 Performance of the developed mathematical model in terms of solution time using GAMS

s (operations)	m (stations)	q (variants)	Avg. time (Sec.)
5	2	3	2.15
		5	3.63
		10	5.26
		15	6.50
		20	8.82
	4	3	3.48
		5	5.37
		10	11.72
		15	18.93
		20	30.53
	6	3	N/A
		5	

		10	
		15	
		20	
7	2	3	2.48
		5	3.96
		10	4.55
		15	6.03
		20	9.76
	4	3	9.88
		5	12.35
		10	15.69
		15	22.42
		20	37.96
	6	3	48.41
		5	53.26
		10	68.13
		15	77.71
		20	96.33
10	2	3	6.15
		5	8.96
		10	11.23
		15	15.71
		20	22.94
	4	3	30.68
		5	39.11
		10	48.74
		15	68.24
		20	102.13
	6	3	165.01
		5	198.22
		10	236.61
		15	281.85
		20	352.81
20	2	3	218.88
		5	302.19
		10	367.55
		15	547.62
		20	828.46
	4	3	1237.92
		5	1527.92
		10	1975.16
		15	2779.55
		20	4185.63

	6	3	> 5400
		5	> 5400
		10	Out of memory
		15	Out of memory
		20	Out of memory
30	2	3	Out of memory
		5	
		10	
		15	
		20	
	4	3	
		5	
		10	
		15	
		20	
	6	3	
		5	
		10	
		15	
		20	

Figures 4.4, 4.5 and 4.6 illustrate the solution time in terms of different product variants, operations, and machines/stations. Note that Figure 4.6 is limited to 2 and 4 numbers of machines as the information regarding the solution time for 6 numbers of machines is not sufficient to be plotted in the figure.

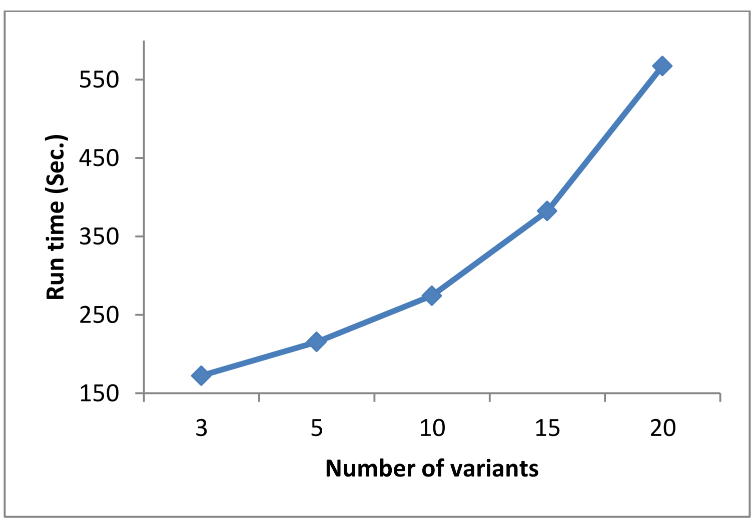


Figure 4.4 GAMS run time in terms of number of product variants

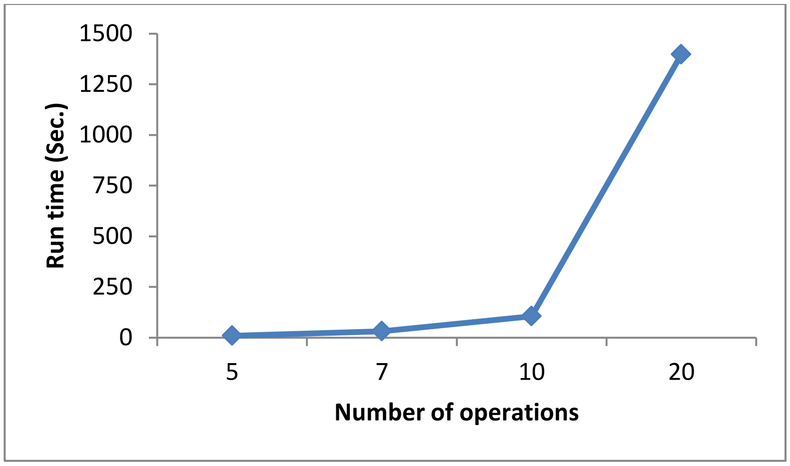


Figure 4.5 GAMS run time in terms of number of operations

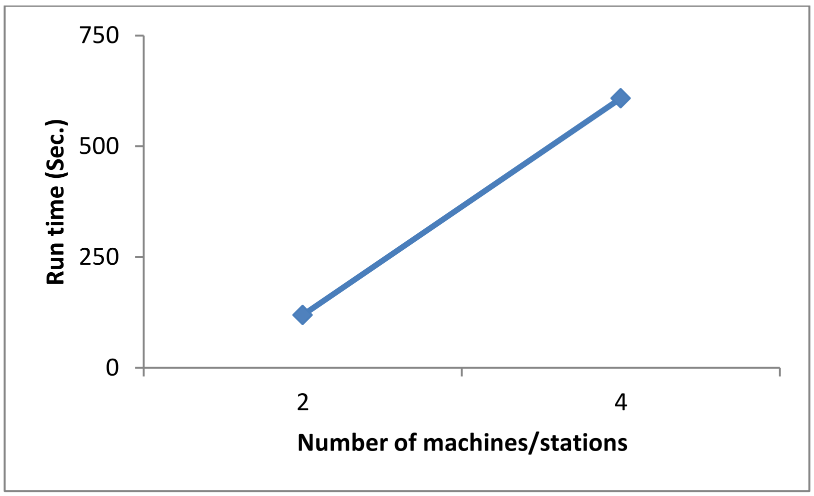


Figure 4.6 GAMS run time in terms of number of machines/stations

Based on the figures, it is observable that the solution time is very sensitive to the number of operations and it grows exponentially as the number of operations increases. The solution time is also sensitive to the number of machines and product variants. However, its sensitivity is less significant for the number of product variants.

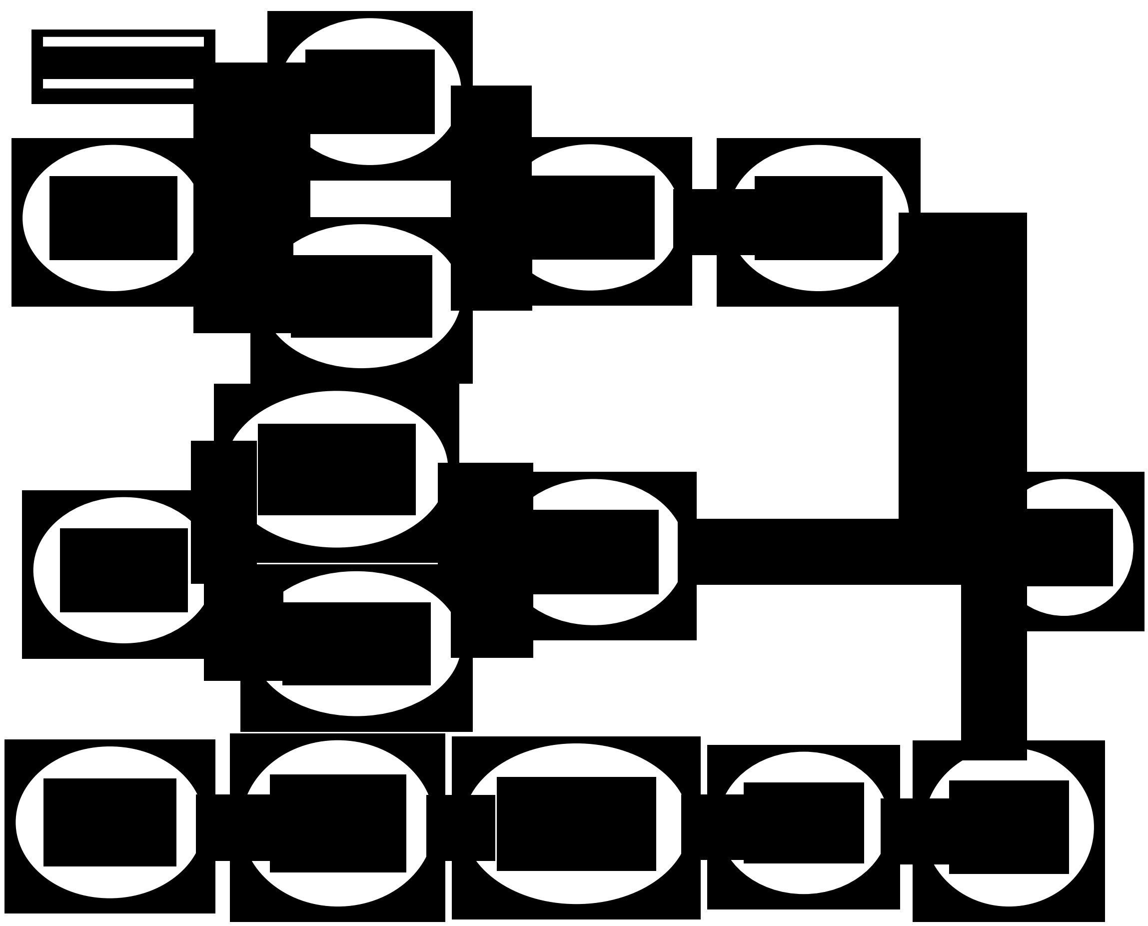
4.7 Case Study

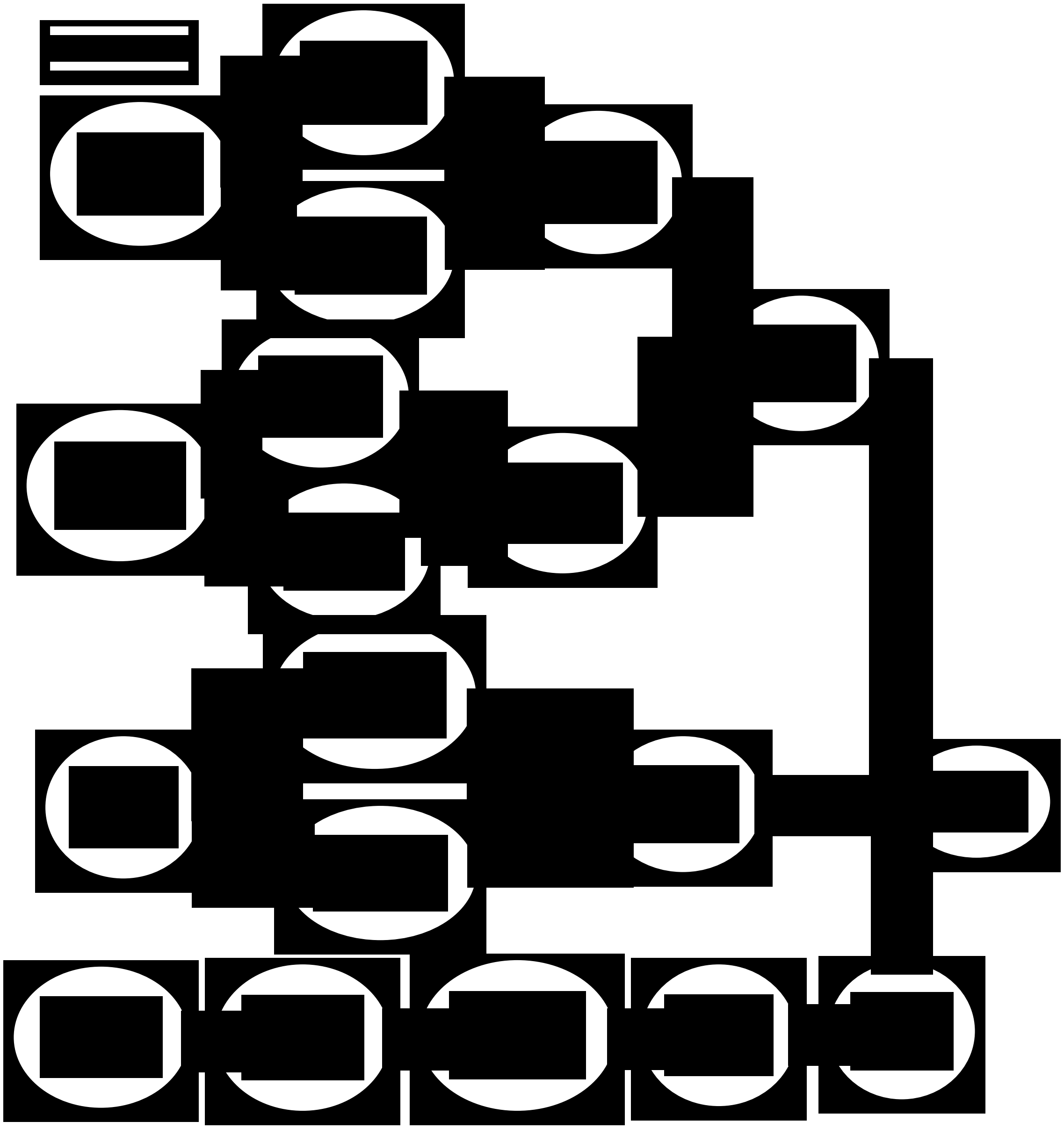
This case is about an automotive company manufacturing different variants of engine cylinder blocks. For confidentiality reasons, the company information is not revealed and some assumptions are made. Three variants of cylinder blocks namely I-4, V-6, and V-8 are considered in this case study with the production volume of 250, 100, and 50 units per day for each variant, respectively. Figure 4.7 illustrates these cylinder blocks.

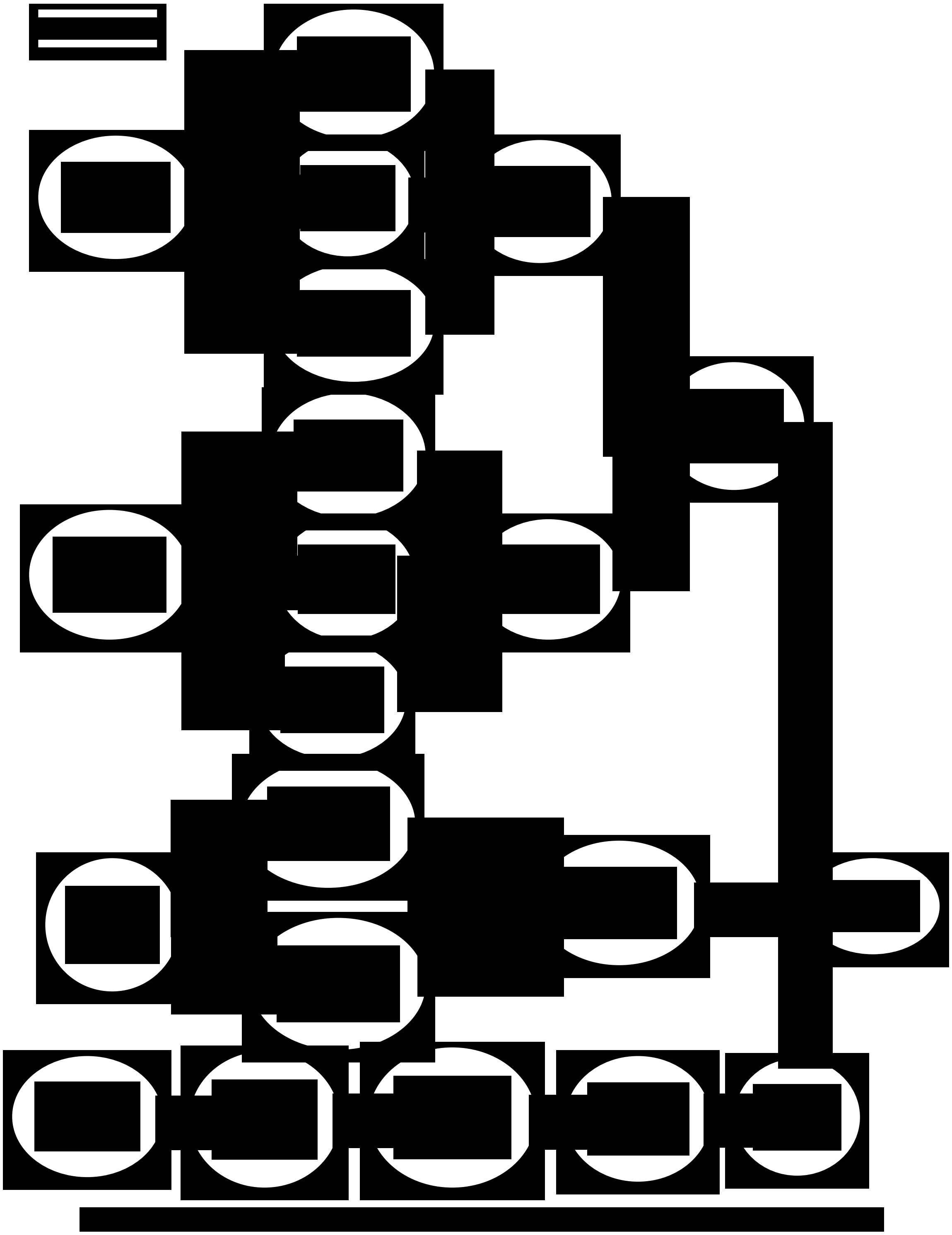
(a) I-4 cylinder block (b) V-6 cylinder block (c) V-8 cylinder block

Figure 4.7 Three variants of engine cylinder block (http://paceperformance.com/)

Figure 4.8 demonstrates operations sequences of the three cylinder blocks.







There are four types of available machines, including two different types of 5-axis CNC machines, one honing machine and one industrial washing machine. There are also some parallel machines of each type at each station to satisfy the demands. Depending on accuracy and cost considerations, the two types of the CNC machines have different capabilities. Table 4.5 shows the capabilities of each machine in performing the various operations.

Table 4.5 Machine capability matrix

	Machine			
	CNC-1	**CNC-2**	**Honing**	**Washing**
Rough milling (deck face, side deck, pan face)	0	1	0	0
Finish milling (deck face, side deck, pan face)	1	0	0	0
Rough cylinder boring	1	1	0	0
Drilling oil holes (deck face, side deck)	1	1	0	0
Honing cylinder bore	0	0	1	0
Camshaft & water pump boring (finish bore)	1	0	0	0
Drilling & reaming & tapping holes on pan face	1	0	0	0
Rough crank bore	1	1	0	0
Honing crank bore	0	0	1	0
Cleaning	0	0	0	1
Contour milling	1	1	0	0

Table 4.6 also shows the backtracking distance between the stations.

Table 4.6 Backtracking distance for case study (in meters)

		Station		
		S1	**S2**	**S3**
Station	**S2**	5	-	-
	S3	9	4	-
	S4	15	10	6

The developed MIP model was solved for this case study using GAMS optimization software and the optimum solution of zero total backtracking distance was obtained. To demonstrate that the variability of the solution time is negligible, the case study was solved five times and GAMS was able to find the optimal solution in 2261.74 seconds on average with 26 seconds standard deviation. Due to capabilities of CNC machines and also sequence flexibility in performing the operations, this case study does not yield a unique solution. One of the obtained solutions suggest locating Machines CNC-2 and

CNC-1 at Stations 1 and 2, and honing and industrial washing machines at Stations 3 and 4 respectively. In addition, all rough milling operations are assigned to CNC-2, all honing operations to the honing machine, the cleaning operation to the washing machine and the rest of the operations are assigned to CNC-1. Consequently, in the final operations sequence of each cylinder block variant, rough milling operations should be carried out first (no restriction on their order), cleaning operation would be the last operation, and immediately before that, honing operations should be performed. The rest of the operations should be carried out after rough milling and before honing processes. In other words, based on the result from the model, the rest of the operations can be carried out in any sequence before honing and after rough milling processes.

Suppose that in the current system instead of assigning “Rough crank bore” operation to CNC-1, it is assigned to CNC-2. In that case, backtracking would occur with the minimum value of 5 meters per product unit. In the current system, there are two CNC-2, six CNC-1, two Honing, and two Washing machines in parallel at Stations 1 to 4 respectively for line balancing and also fulfilling the demands. The cycle time at these stations is 65, 70, 58 and 50 seconds respectively. The cycle time at Stations 1 and 2 includes the wasted time occurred for backtracking. Each cylinder block requires backtracking for traveling from Station 2 to Station 1 and then going back from Station 1 to Station 2 for further processes. It can be assumed that based on the speed of the transporter, every meter of backtracking in this case study corresponds to 2 seconds wasted time. Thus, 10-second wasted time exists for each cylinder block unit. “Rough crank bore” operation takes typically 30 seconds to be processed. If this operation is assigned to CNC-1 at Station 2 to minimize the backtracking, it increases the cycle time at Station 2 by 5 seconds (30s/6 = 5s) but it also decreases the cycle time at Station 2 by 10 seconds for eliminating the backtracking. Therefore, the new cycle time at Station 2 would be 65 seconds (70s+5s-10s = 65s). The total throughput of a system is determined based on the greatest value of the cycle time amongst different stations called “bottleneck”. If only one shift (8 hours) is considered, the first system is capable of manufacturing 411 units while the second system with zero backtracking is capable of manufacturing 443 units. In other words, by minimizing the backtracking, total throughput increases by 32 units per shift. This is equivalent to 7.79% improvement in total throughput. Therefore, by minimizing backtracking, the availability of jobs to machines increases and the total throughput improves accordingly. For further quantitative analysis regarding improvement of throughput by minimizing backtracking, readers are referred to Sarker et al. (1995).

4.8 Summary

Backtracking minimization of a family of product variants was considered in this chapter. Minimizing backtracking can result in significant improvement in total throughput. Product variants were considered to have networked operations sequence. A mathematical model was developed to optimize three decision variables: (1) location of each machine at each station, (2) assignment of operations to the machines considering machine capability and (3) final sequence of performing operations for each product variant. No work in the backtracking literature has considered all the addressed features. Particularly, there is no work considering networked product variants and machine selection for performing each operation.

Since the developed mathematical model is non-linear, it was reformulated to its equivalent MIP model to be able to use LP solvers. A case study demonstrated real-life application of the model and highlighted the importance of minimizing backtracking.

Chapter 5

Generating Serial and Networked Master Operations Sequence

5.1 Introduction

Another strategy of managing variety in the modern manufacturing systems is to use the available information of current variants for design and production of new variants. Operations sequences of existing variants are one of the available useful information that should benefit from. The term "operation" in this chapter also refers to both "fabrication/manufacturing" and "assembly" processes.

The sequence by which operations are performed for a given variant has a significant impact on many system operation aspects such as operation cost, operation difficulty, need for reworking, and need for changing tools and fixtures. There are quite few works in the literature regarding this problem. Most of these works focus on generating a master bill of material called generic bill of material (GBOM) and are for assembly purposes only (e.g. Hegge and Wortmann 1991, Romanowski and Nagi 2004). In other words, there is no work in the literature considering construction of master operations sequence for processes other than assembly. In addition, the methodologies and approaches applied in the addressed works are based on assembly trees and are not usually applicable to fabrication. Assembly trees cannot efficiently represent the precedence relationships in fabricating applications. Moreover, most of the developed algorithms for master assembly sequence or generic bill of material (GBOM) in literature are not retrieval-based and do not benefit from available legacy assembly sequence information. In fact, most of the works in literature aim at generating feasible assembly sequences using a pre-defined set of feasibility constraints (e.g. ElMaraghy and Laperrière 1992) and then searching the best assembly sequence from the generated ones based on some criteria. Therefore, there is a need to develop a model which not only can be applied for both assembly and fabrication, but it also deploys the available information of the existing variants.

Generating a master operations sequence derived from operations sequences of a family of existing variants has several advantages. Frequent changes in demands necessitate design and production of new variants. Operations sequence planning/process planning of these new variants from scratch requires time, cost and effort. However, the generated master operations sequence can be used to construct the operations sequence for new variants that fall within or significantly overlap with the boundary of the considered family and would require minimal updates. This will lead to reducing the required time and cost for developing operations sequences of the new variants and consequently, enhancing the productivity. In addition, since major modification in a manufacturing system is not desired, most manufacturers prefer to use as much as possible their existing way of performing operations for the next generations of parts/products unless new machines with new capabilities or different technologies have been introduced which would affect the existing/legacy operations sequences. By using the obtained master operations sequence, the new variant can be processed in line with the existing variants. This will also result in less changeover time when changing the system setups for processing new variants.

In this chapter, two new mathematical models are developed to generate master operations sequence for a family of part/product variants. As mentioned in previous chapters, depending on the features of a given variant, its operations sequence can be serial or networked. The first mixed integer programing (MIP) model is developed for variants with serial operations sequences. The second model, a generalized form of the first model, is developed for variants with various process sequence structures (e.g. serial, networked or combination of both).

As the number of operations increases, the ability of the models to find the optimum master operations sequence in a reasonable time decreases. Therefore, a novel algorithm is proposed which is not only capable of dealing with variants having both serial and networked operations sequences but also is capable of finding solutions for the large size problems. The proposed algorithm is quite fast and comparing its solutions with the solutions obtained from the two developed mathematical models indicates that it is able to find the optimum solutions for all the studied cases. Results illustrate the efficiency of the algorithm in terms of run time and quality of the obtained solutions. Two case studies are demonstrated for assembly and fabricating applications.

5.2 Literature Review

As mentioned before, there is a dearth of literature on this topic. In particular, there is no work considering generating a master operations sequence other than assembly processes. The "Operation" term in the present chapter, like previous chapter, refers to both fabrication/manufacturing (including machining, heat treatment, quality control, etc.) and assembly processes.

Most of the works in the literature focus on generic bill of material (e.g. Hegge and Wortmann 1991, Jiao et al. 2000, Romanowski and Nagi 2004). A bill of material (BOM) of a specific product is a list of the raw materials, components, sub-assemblies, parts and also the number of each item required to manufacture that product. BOM usually involves three aspects: (1) items: how a product is constructed from semi-finished product and purchased parts; (2) goes-into relationships: how a particular parent and component are related together; (3) employment: in which application, a particular BOM is used (Jiao, et al. 2000). Generic Bill Of Material (GBOM), which was first developed by Hegge and Wortmann (1991), represents a general structure of different variants belonging to a product family and performs as a tool for designing new variants and facilitates search of similar parts. Romanowski and Nagi (2004) developed a data mining-based methodology to generate generic bill of material (GBOM). In their method, they tried to unify similar BOM trees into a single GBOM by using text mining. They also deployed some data from an industrial BOM database to demonstrate their proposed method. In another application, Shu et al. (2014) applied generic bill of material (GBOM) concept in supply chain construction. By using generic bill of material (GBOM), they assessed the control of production disruption risk and examine the effect of production uncertainty in supply chain companies. They developed a function based on random simulation and neural network to approximate the level of uncertainty and used genetic algorithm and simulated annealing to find near optimal scheme of supply chain construction.

The limitation of the proposed generic bill of material (GBOM) is that they are based on assembly processes with some restrictions on the BOMs' structures which make them inappropriate for use with processes other than assembly.

There are few works in the literature which tried to generate master assembly sequence from existing assembly sequence of variants. Martinez et al. (2000) generated a master assembly sequence and called it "parent". Master assembly sequence is a comprehensive assembly sequence for a set of products which have a significant number of similar components and share a similar structure. This parent plan was developed for a hypothetical product called "meta-product" which includes all the components of a family of products, which is the same as generating master part for a family using group technology. They considered three types of constraints in assembly space to ensure that only feasible master assembly sequences are generated. The difference in the path length was considered as a measure for selecting the best master assembly sequence. The path length is a measure of the deviation of the generated assembly sequence from the existing ones. Kashkoush and ElMaraghy (2014) generated master assembly sequence by using the assembly trees of members of products families. Each assembly tree was converted to an equivalent matrix. Feasible master assembly sequence was found by minimizing Robinson-Foulds distance from the assembly sequence of each product variant in the family. Kashkoush and ElMaraghy (2015) also developed a mathematical model to generate a master assembly sequence from the assembly information of individual family members. They first developed a non-linear model and then in order to find the exact solution, the model was linearized. In both works, the authors indicated that the generated master assembly sequence can reduce the assembly planning cost and enhance productivity in assembly industries. Lai and Huang (2003) proposed a method called ASP-LMPR to find the optimal assembly sequence of a product family. Their method uses Boolean operators to generate feasible assembly sequences at each loop. They also considered all evaluation factors available in the literature to determine the feasible and optimal assembly sequences and applied an illustrative example to show the efficiency of their proposed method.

In all above works, the proposed methods have been applied to assembly processes and used non-directed trees which are considered as limitations. The non-directed trees, as mentioned before, cannot efficiently illustrate the precedence relationships in fabricating processes and are limited to assembly applications.

Azab and ElMaraghy (2007a) proposed a new process planning approach to reconfigure the process plan of a composite master part instead of defining a new process plan for new product variants. This reconfiguration can be achieved by adding/removing some features according to a novel binary integer programming model. The configured process plan may have some inserted/deleted features in order to meet the requirements of new product variants. This reconfigured Process Plan method minimizes the extent of plan change hence reducing the cost of set-up changes on the shop floor. It was assumed that the process plan for the composite part of a family of products is available in advance. Azab and ElMaraghy (2007b) also proposed a sequential hybrid approach for macro level of process planning. They compared their model with the re-planning approach and showed that their proposed model is more practical, faster, less costly and easy to implement. Similarly, the authors assumed that the process plan for the composite part is available beforehand.

According to the literature review, there is no work which considers generating master operations sequence from the available operations sequence of the existing variants. This is what both developed models and the proposed algorithm in this chapter aim to achieve.

5.3 Problem Scope and Definition

There are a number of part/product variants belonging to a family. Each individual variant requires a number of operations to be finalized. As aforementioned, “operation” in this chapter refers to both “fabrication/manufacturing” and “assembly” processes. Examples of fabrication/manufacturing processes include machining, welding, molding, heat treatment, quality control and so on. Operations of a variant follow a precedence relationship which is called “operations sequence”. A generic or master operations sequence is derived by considering the operations sequences of the existing variants. The derived master operations sequence is then deployed to extract the operations sequences of new variants which belong to the same family of the variants or significantly overlap with its scope.

The developed mathematical models and the proposed algorithm which will be described in the following section try to employ the knowledge embedded in available operations sequences of the existing variants. Therefore, the problem is to construct a generic or master operations sequence that best represents each individual operations sequence (i.e. minimum distance from each of them) even if there are some conflicts between the master operations sequence and some individual one(s).

The following assumptions are considered:

- All part/product variants belong to a family and, hence, share many similarities.
- The new variants belong to or significantly overlap with the scope of the considered family. Hence, they also share many similarities with the existing variants.
- Operations sequence information of the existing variants is available.
- The same operation name or number is used for all different variants requiring a particular operation.

As mentioned in Chapter 3, an “operation precedence graph” is a good representation of operations sequence of a given variant. This graph indicates how operations of variants relate to each other and their sequence. There are typically two types of operations sequences for variants: serial and networked. Figure 5.1 illustrates the representing operation precedence graphs of two variants with serial and networked operations sequences.

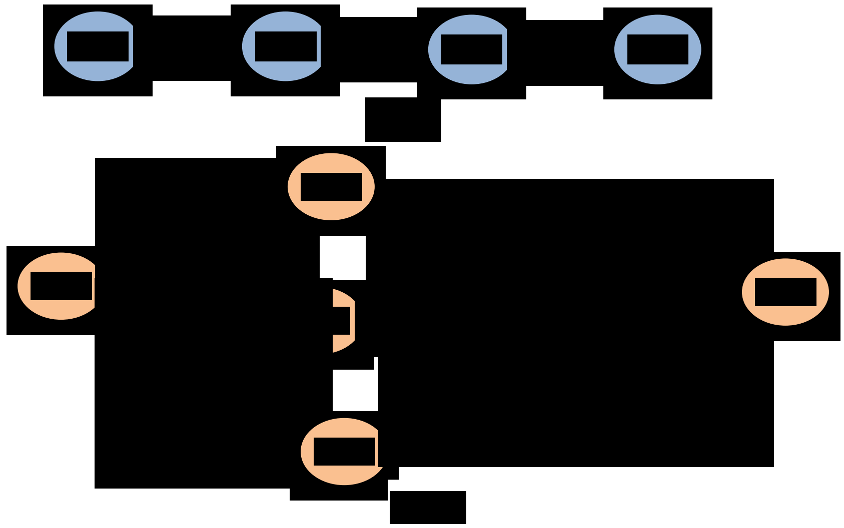


Figure 5.1 An example of comparison between operation precedence graphs of two variants with: (a) sequential serial operations and (b) networked operations

According to Figure 5.1, Operation 3 of the first variant must be processed exactly after Operation 2 and before Operation 4. In contrast, Operation 3 of the second variant has the flexibility to be carried out before, after or simultaneously with Operations 2 and 4.

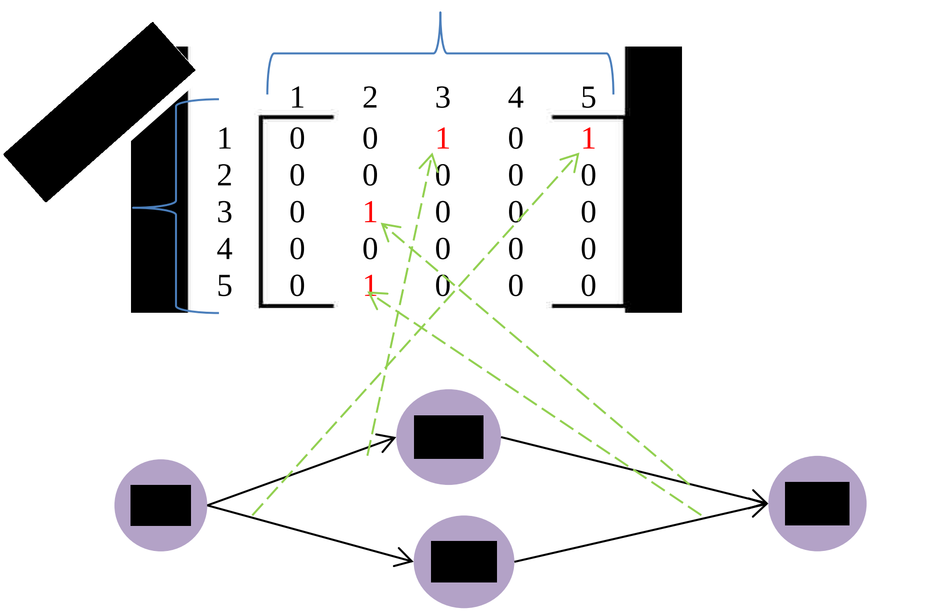
Two mathematical models are developed and an algorithm is proposed to find the master operations sequence from existing variants with both serial and networked structures in order to be used for generating the operations sequences of new variants.

5.4 Master Generating Mathematical Models

In this section, two mathematical models are developed. The objective function is to minimize the sum of dissimilarity distances between the generated master operations sequence with each individual operations sequence of the existing variants. Before developing the models, the next sub-section describes how a given operations sequence is encoded and how the dissimilarity distance between any two operations sequences is calculated.

5.4.1 Encoding Operations Sequences and Calculating the Objective Function

A matrix structure is used to encode operations sequences of variants. Each matrix has a size of $m \times m$ where m refers to the total number of operations among all the variants. For the corresponding matrix of a given variant, a_{ij} is equal to one if Operation j is performed immediately after Operation i and zero otherwise. In other words, if there is a directed edge from Operation i to Operation j in the representing operation precedence graph of the variant, the value of element a_{ij} in the matrix would be one. Therefore, the encoded matrix is binary. It is clear that if a variant does not require a particular operation, in the encoded matrix of the variant, the corresponding row and column of that operation will be zeros. This is to keep the matrices' sizes of all variants the same $(m \times m)$ which facilitates the numerical computation of the objective function and construction of the master operations sequence. Figure 5.2 shows the operations sequence-to-matrix encoding scheme for the operation precedence graph of a variant. Note that while this variant does not require Operation 4, the row and column of this operation appears in the encoded matrix. The main advantage of the proposed encoding scheme is that only one matrix can be encoded from a given operations sequence (or its corresponding operation precedence graph) and also only one operations sequence can be extracted from the encoded matrix. As shown in Figure 5.2, the number of elements with value of one in the encoded matrix of a variant is equal to the number of edges appearing in the corresponding operation precedence graph of that variant.

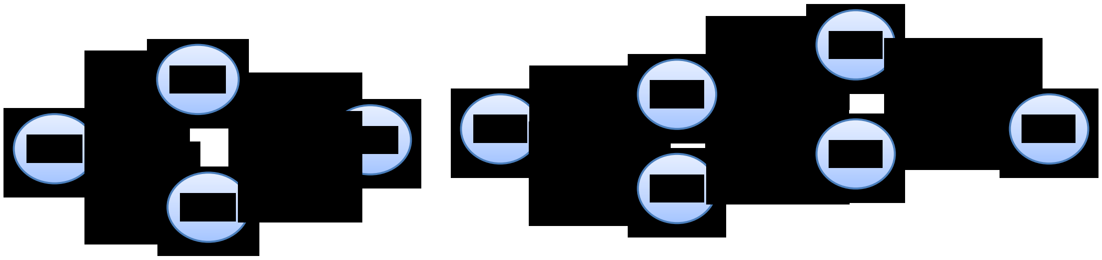


[redacted]

The output of the mathematical models is also a matrix called “master matrix” which has the same size *($m \times m$)* of existing variants matrices. The master operations sequence can easily be extracted from this master matrix. It is noteworthy to mention that the k^{th} row or column ($k=1,2,\ldots,m$) of this matrix has at least one element with the value of one indicating that the master operations sequence has all the operations required by the considered variants.

The objective is to find a master operations sequence which has the minimum dissimilarity with the operations sequences of existing variants. Hence, the master matrix should have the minimum dissimilarity distances from the encoded matrices of the existing variants. To find the dissimilarity distance, the difference between elements of the master matrix with each individual encoded matrix is calculated and the result is summed over all the number of existing variants. For calculating the difference between the master matrix and a given encoded matrix, first those rows and columns of the master matrix corresponding to the operations which do not exist in the considered variant are set as zeros. In fact, to compare the master operations sequence with a given operations sequence of an existing variant, it is required to exclude those operations (with their incoming and outgoing edges) from the master operations sequence that did not appear in that variant. Then the absolute difference between each element of the master matrix and its counterpart in the encoded matrix is calculated and summed with other difference values. This procedure is repeated for all remaining variants and the summation of all the obtained values yields the dissimilarity distance between master operations sequence and operations sequences of the existing variants.

To better explain the objective function, consider the two operation precedence graphs illustrated in Figure 5.3. The first graph is the representation of the operations sequence for one of the existing variants while the second graph is the representation of the potential master operations sequence. The corresponding encoded matrices have also been illustrated.



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

[redacted]

To find the dissimilarity distance between these two matrices, first, the rows and columns of 3 and 4 are excluded from the comparison as Operations 3 and 4 do not exist in the existing variant (see the red lines in Figure 5.3). Then, the absolute differences between remaining elements of two matrices are calculated and summed. Therefore, the dissimilarity distance between the operations sequence of the existing variant and the potential master operations sequence is: |0-1|+|1-1|+|1-1|+|0-1|=2.

5.4.2 Serial Master Operations Sequence Generating Model

In this sub-section the first mathematical model, serial master operations sequence generating model, is developed for constructing a master operations sequence using the available information of the existing variants with serial operations sequences. This model is based on the assumption that new variants also have serial operations sequences. Therefore, the developed model derives strictly serial master operations sequence to guarantee that the extracted operations sequence for a new variant will also be serial.

It should be noted that for this model, it might be the case that the optimum master operations sequence should be networked to minimize the total dissimilarity distance from the existing variants. However, based on the assumption made in this model, the generated master operations sequence would always be serial; hence, the optimality of the obtained solution is not guaranteed.

Nevertheless, if the model yields a single solution with zero dissimilarity distance (zero objective function), then it is assured that the optimum master operations sequence is in fact serial and the obtained solution from the serial master operations sequence generating model is optimal; otherwise, there is no guarantee for optimality and it is advisable to use the general master generating model (second mathematical model) or the proposed master generating algorithm.

The following notations are used in the serial master operations sequence generating model.

Parameters:

n total numbers of operations among all variants

p total numbers of part variants

$$A_{ki} = \begin{cases} 1 & \text{If Operation } i \text{ exists in } k^{th} \text{ part variant} \\ 0 & \text{Otherwise} \end{cases} \quad (5.1)$$

$$B_{ijk} = \begin{cases} 1 & \text{If Operation } j \text{ of } k^{th} \text{ part variant is performed immediately after Operation } i \\ 0 & \text{Otherwise} \end{cases}$$
(5.2)

Variables:

$$X_{ij} = \begin{cases} 1 & \text{If Operation } j \text{ of the master is performed immediately after Operation } i \\ 0 & \text{Otherwise} \end{cases}$$
(5.3)

$$W_{ig} = \begin{cases} 1 & \text{If Operation } i \text{ is placed in } g^{th} \text{ position of the master} \\ 0 & \text{Otherwise} \end{cases}$$
(5.4)

$$S_{ijg} = \begin{cases} 1 & \text{If Operation } i \text{ is placed in } g^{th} \text{ position immediately before Operation } j \text{ in master} \\ 0 & \text{Otherwise} \end{cases}$$
(5.5)

Using the listed parameters and variables, a non-linear integer programing model is developed as follows. Note that to find the total dissimilarity distance, square of distance has been used for its simplicity over the absolute distance while both would yield the same result.

$$Minimize \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{p}((X_{ij} - B_{ijk})^2 \cdot A_{ki} \cdot A_{kj}) \qquad (5.6)$$

Subject to:

$$S_{ijg} \leq \frac{W_{ig} + W_{j,g+1}}{2} \qquad \forall i,j,g = 1,2,\dots,n \qquad (5.7)$$

$$W_{ig} + W_{j,g+1} - 1 \leq S_{ijg} \qquad \forall i,j,g = 1,2,\dots,n \qquad (5.8)$$

$$X_{ij} = \sum_{g=1}^{n} S_{ijg} \qquad \forall i,j = 1,2,\dots,n \qquad (5.9)$$

$$W_{i,n+1} = 0 \qquad \forall i = 1,2,\dots,n \qquad (5.10)$$

$$\sum_{i=1}^{n} W_{ig} = 1 \qquad \forall g = 1,2,\dots,n \qquad (5.11)$$

$$\sum_{g=1}^{n} W_{ig} = 1 \qquad \forall i = 1,2,\dots,n \qquad (5.12)$$

$$X_{ij} \in \{0,1\} \qquad \forall i,j = 1,2,\dots,n \qquad (5.13)$$

$$W_{ig} \in \{0,1\} \qquad \forall i = 1,2,\dots,n \;, \qquad \forall g = 1,2,\dots,n+1 \qquad (5.14)$$

$$S_{ijg} \in \{0,1\} \qquad \forall i,j,g = 1,2,\dots,n \qquad (5.15)$$

Equation (5.6) is the objective function of the problem which minimizes total dissimilarity distance between the master operations sequence and operations sequences of the existing variants. In fact master operations sequence, by definition, must yield a graph similar to the graph of an individual variant when excluding the operations appearing in the master graph but not in the graph of that variant. Therefore, by defining parameters A_{ki} and A_{kj}, only common operations between master graph and individual graphs are considered.

Equations (5.7) to (5.9) determine whether or not Operation *j* is processed immediately after Operation *i* in the master operations sequence. Equation (5.10) guarantees that no operation can be placed in position *n+1*. This position is, in fact, a dummy position and is applied in Equations (5.7) to (5.9) for finding the value of X_{ij}. Equation (5.11) ensures that each position is occupied by only one operation. Similarly, Equation (5.12) guarantees that each operation is placed only on one position of the master operations sequence. Finally, Equations (5.13) to (5.15) indicate that X_{ij}, W_{ig} and S_{ijg} are binary variables.

The developed mathematical model is non-linear. GAMS optimization software with CPLEX Solver is used. It is reformulated to its equivalent linear model before implementing it in GAMS (Mozdgir et al. 2013, Navaei et al. 2013). In the serial master operations sequence generating model, all constraints are linear and the only non-linear formulation is the objective function which consists of a square term. To reformulate a square function to its equivalent linear format, it is first extended as follows: $\left(X_{ij} - B_{ijk}\right)^2 = X_{ij}^{\ 2} - 2X_{ij}B_{ijk} + B_{ijk}^{\ 2}$. Then, since X_{ij} is a binary variable, the value of $X_{ij}^{\ 2}$ is equal to X_{ij} itself. This rule can also be applied for a binary parameter such as B_{ijk}. Therefore the objective function is linearized as follows:

$$Minimize \ \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{p} X_{ij}.A_{ki}.A_{kj} - 2X_{ij}.B_{ijk}A_{ki}.A_{kj} + B_{ijk}\,A_{ki}.A_{kj} \qquad (5.16).$$

Using Equation (5.16) as the linear objective function and initially linear Constraints (5.7) to (5.15), the elements of the master matrix can be determined and based on the obtained matrix, the master operations sequence can be constructed.

5.4.3 Generalized Master Operations Sequence Generating Model

A model is developed for generating master operations sequence from existing variants with serial or networked operations sequences or combination of both structures. It is a generalized form of the serial master operations sequence generating model and hence, can deal with variants having various operations sequence structures. Therefore, it is more complicated than the serial model and contains greater number of constraints.
Both the serial and generalized models share some similar notations and hence, they are not repeated here. The objective function of the generalized model is also the same as the serial model and hence, the linearized form is presented here.

$$Minimize \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{p} X_{ij}.A_{ki}.A_{kj} - 2X_{ij}.B_{ijk}A_{ki}.A_{kj} + B_{ijk}\,A_{ki}.A_{kj} \qquad (5.16)$$

Subject to:

$$\sum_{j=1}^{n} X_{ij} + \sum_{j=1}^{n} X_{ji} \;\; \geq \;\; 1 \qquad \forall i = 1,2,\dots,n \qquad (5.17)$$

$$\sum_{u=1}^{\tau-1} X_{m_{(u,l)},m_{(u+1,l)}} + X_{m_{(\tau,l)},m_{(1,l)}} \;\; \leq \;\; \tau - 1 \qquad \forall T \subseteq N, N = \{1,2,\dots,n\}, \tau = |T|, |T| \geq 2, \forall l = 1,2,\dots,(\tau-1)! \qquad (5.18)$$

$$X_{ii} = 0 \qquad \forall i = 1,2,\dots,n \qquad (5.19)$$

$$X_{ij} \in \{0,1\} \qquad \forall i,j = 1,2,\dots,n \qquad (5.20)$$

Where in Equation (5.18):

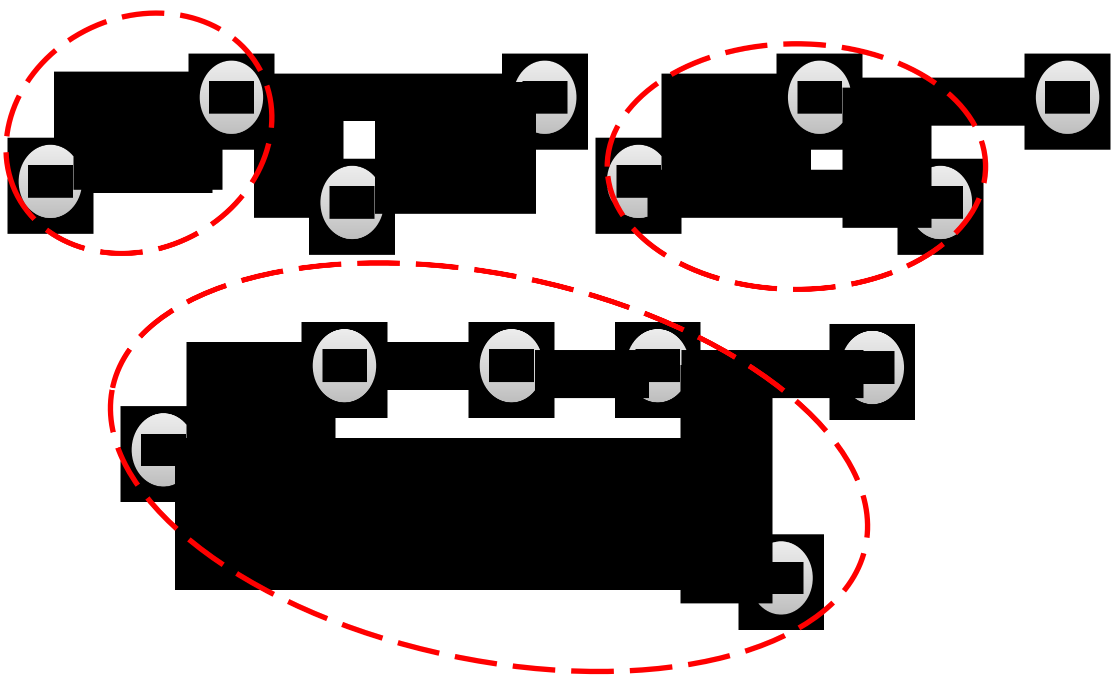
$N = \{1,2,\dots,n\}$, $T \subseteq N$, $\tau = |T|$

$m_{(u,l)}$= u^{th} member of l^{th}member of Set S.

S is a set whose members are actually sets themselves. Members of *S* are all the circular permutations of all members of *T*. In other words, the total members (sets) of *S* are equal to the total distinctive ways of creating directed loops using all members of *T* which are equal to$(\tau - 1)!$. Thus, Set *S* can be shown as follows:

$$S = \left\{\{m_{(1,1)}, m_{(2,1)}, \dots, m_{(u,1)}, \dots, m_{(\tau,1)}\}, \dots, \{m_{(1,l)}, m_{(2,l)}, \dots, m_{(u,l)}, \dots, m_{(\tau,l)}\}, \dots \{m_{(1,(\tau-1)!)}, m_{(2,(\tau-1)!)}, \dots, m_{(u,(\tau-1)!)}, \dots, m_{(\tau,(\tau-1)!)}\}\right\} \qquad (5.21).$$

Equation (5.16) is the objective function which was described in previous sub-section. Equation (5.17) is the first set of constraints of the model ensuring that master operations sequence has all the operations shared among the operations sequences of the existing variants. Equation (5.18) guarantees that no directed loop can exist in the master operations sequence. In order to make sure that no directed loop will happen, for each set of operations, this constraint should be applied. In fact, Equation (5.18) is a set of anti-loop constraint sets and is similar to sub-tour elimination constraint in Travelling Salesman Problem (TSP) (Hoffman, 2013). Equation (5.18) prevents performing an operation that has already been carried out. For instance, a given part which has already undergone "Saw Cutting" operation should not again undergo the same "Saw Cutting" operation. Note that, in reality, a part/product may require several milling, drilling, etc. operations in different steps but they should be identified under different operation names (such as milling1, milling2,…) to be distinguished from the addressed directed loop issues. Figure 5.4 shows some examples of invalid master operations sequences with directed loops.



[redacted]

In all above examples, there are directed loops which make the generated operations sequence invalid. In the first case, the directed loop is: 1→3→1.Note that Operations 2, 3 and 4 in the first case do not make a directed loop; because their relationship indicates that Operation 2 should be performed after Operations 3 and 4 and Operation 4 should be performed after Operation 3 and there is no directed loop among them. In fact, precedence constraint of 3→2 is redundant. In the second case, 1→2→4→1 and in the third case, 1→2→5→6→4→1 make the directed loops. Equation (5.19) is a logical constraint, with the same concept as Equation (5.18), which avoids the directed loops for each single operation. Finally, Equation (5.20) ensures that X_{ij} is a binary variable.

5.5 The Proposed Master Generating Algorithm

As the total number of operations increases, the time efficiency of the developed models decreases. In this section, a novel algorithm is proposed which is not only capable of finding good (and even optimal) solutions, but it is also quite time efficient for any sizes of problems.

Considering the objective function, it can be interpreted that the appearance of each edge in the master operations sequence is in fact based on the frequency of that edge among all variants as long as it satisfies the objective function. Having reasonable number of a particular edge among variants does not guarantee that it should also be appeared in the master operations sequence. For instance, suppose that the edge 1→4 has been appeared in five out of thirteen operations sequences of existing variants. In addition, suppose that Operations 1 and 4 are both appeared in eleven out of these thirteen variants. Despite the fact that the edge 1→4 has a reasonable frequency (five), there are six other variants (11-5=6) which have both Operations 1 and 4 but do not have that edge (i.e. there is no immediate relationship between the two operations in those six variants). Therefore, the edge 1→4 should not be appeared in the master operations sequence since adding it to the master operations sequence corresponds to deterioration in the objective function. On the other hand, there might be edges with low frequency among variants and still they should be appeared in the master operations sequence as they improve the objective function.

In the proposed master generating algorithm, after finding the frequency for each edge, those edges with zero frequency and those that worsen (increase the value of) the objective function are eliminated. The remaining edges are sorted in descending order. Next, one edge at a time is considered to be added to the master operations sequence. If the new edge (which is going to be added) makes a directed sub-tour with any sets of previously added edges, it will not be added to the master operations sequence. This is similar to Constraint (5.18) for sub-tour elimination in the second mathematical model. For this purpose, in the proposed algorithm, each operation has a list of Tabu operations which is empty at the beginning. Whenever, an edge is going to be added to the master graph, this edge is first checked to make sure that it is not a non-permitted edge and after adding the edge, Tabu lists are updated accordingly. For instance, suppose that the edge 3→5 is the first edge that is added to the master operations sequence. Therefore, Set 3 is updated from 3={∅} to 3={5} meaning that the edge 5→3 is not permitted anymore to be in the master graph. Now, suppose that the edge 5→2 is the second candidate edge (in terms of frequency) to be added to the master graph. Therefore, Set 5 is updated from 5={∅} to 5={2} and consequently Set 3 is updated from 3={5} to 3={5,2} as Set 5 is actually an element of Set 3. It means that, in addition to the edge 5→3, the edges 2→5 and 2→3 are also not permitted to be in the master operations sequence since they lead to having a directed loop in the master operations sequence. This procedure continues until all remaining edges are assessed to be either added to or not considered in the master operations sequence.

The proposed master generating algorithm can deal with variants having serial, networked or combination of both structures and the obtained master operations sequence can be serial or networked.

The following is the pseudo code of the proposed master generating algorithm:

Begin algorithm

Set n= total number of existing variants, m=maximum number of operations among all the existing variants.
Set $P_1, P_2, \ldots P_n$: matrices of operations sequences of Variant 1 to Variant n with the elements of $y[i][j][k]=1$ if edge $i \rightarrow j$ exists in Variant k and 0 otherwise.
Set Matrix X: matrix of master operations sequence (master matrix) with its all $m\times m$ elements equal to zero.
Set $O_1=O_2=\ldots=O_m=\{\emptyset\}$, Tabu list of each operation in the master operations sequence.
Set $e \leftarrow 0$, $w \leftarrow 1$, and $q \leftarrow 1$, as counters.

For all possible edges $(m\times(m\text{-}1),\ i\neq j)$: **finding the frequency of each edge* over all existing variants*

a. Set $Frq[i][j] = \sum_{k=1}^{n} y[i][j][k]$

Endfor

For all possible edges $(m\times(m\text{-}1),\ i\neq j)$: **finding edges deteriorating OF and setting them zero**

If $Frq[i][j] \neq 0$ **and** $Frq[i][j] < 0.5\times n$

a. Set l=total number of existing variants having both Operations i and j

b. **If** $Frq[i][j] < 0.5\times l$

- $Set\ Frq[i][j] \leftarrow 0$

Endif

Endif

Endfor

For all possible edges $(m\times(m\text{-}1),\ i\neq j)$: **eliminating initial zero-frequency and deteriorating edges**

If $Frq[i][j]\neq 0$

a. Set $Frqno[g][h] \leftarrow Frq[i][j]$

b. $g\leftarrow i$

c. $h\leftarrow j$

d. $e\leftarrow e+1$ ** finding total number of non-zero edges after elimination of deteriorating edges**

Endif

Endfor

While $w \leq e$: **sorting edges in descending order **

a. Set $Frqnosort[w] = w^{th}$ highest frequency among *Frqno*

b. Identify the corresponding edge placed in rank w

c. $w\leftarrow w+1$

Endwhile

While $q \leq e$:

a. Select the edge corresponding to $Frqnosort[q]$ (edge $c\rightarrow d$ for instance)

b. **If** edge $c\rightarrow d$ is a permitted edge ($c \notin O_d$)

i. add it to the master matrix ($x[c][d] \leftarrow 1$)

ii. update Tabu lists accordingly

Endif

c. $q\leftarrow q+1$

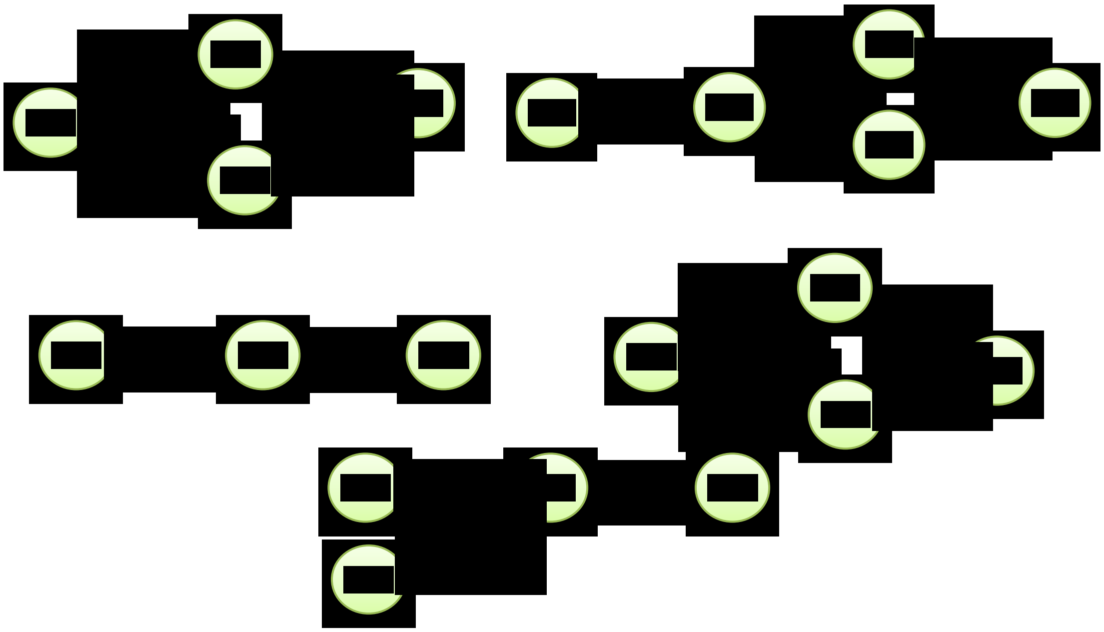
Endwhile

Calculate objective function based on the elements of Matrix X.

End algorithm.

5.6 Illustrative Example

In this section, a hypothetical example is provided to show how the master operations sequence is generated from the existing variants and how it is used to extract the operations sequence of a new variant. Suppose that there are five existing variants with the total number of seven operations. The operations sequences of the existing variants are available. Figure 5.5 illustrates the operation precedence graphs of the existing variants.



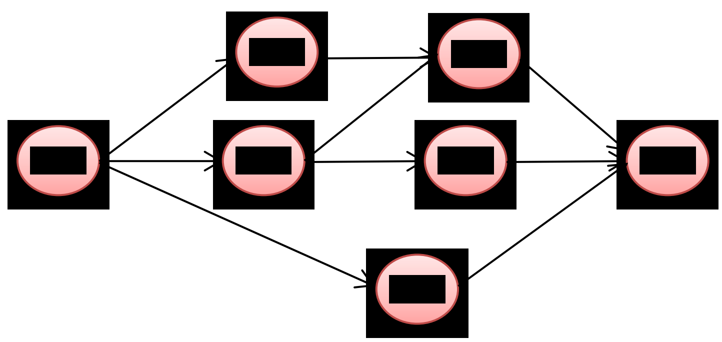
[redacted]

Since the above example includes a combination of both serial and networked operations sequences, the generalized master operations sequence generating model is considered to find the master operations sequence. For this purpose, GAMS optimization solver is used to find the optimum solution. The obtained master matrix has been illustrated in Figure 5.6 with an optimum objective function of 3. It took 6.27 seconds for the model to obtain the solution under a PC with 3.4 GHz Intel Core i7-4770 CPU processor with the Windows 8.1 operating system and 16 GB of RAM. Applying the proposed master generating algorithm also yields the same solution and objective function. Obtaining non-zero objective function indicates that some conflicts (dissimilarities) exist between the existing variants. For instance, in Figure 5.5, in the second variant, there is direct precedence relationship between Operations 2 and 3 (2→3) while in the fourth and fifth variants, Operations 2 and 3 do not have such precedence constraint. This phenomenon is not uncommon in reality and variants belonging to the same family may have some precedence dissimilarities depending on their features.

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

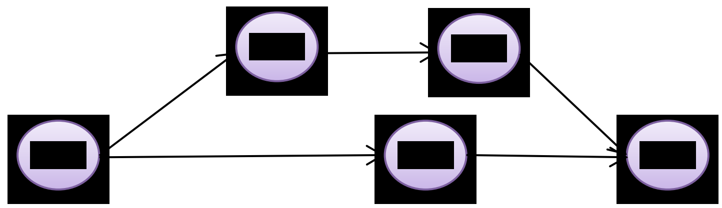
Figure 5.6 Master operation sequence matrix obtained from the second mathematical model

The master operations sequence can be extracted from the obtained master matrix. Figure 5.7 shows the master operations precedence graph.



[redacted]

Now, suppose that a new variant requires operations 1,2,4,5 and 7. Observing Figure 5.5 indicates that none of the existing variants have all these operations. The operations sequence of the new variant can be obtained by excluding the operations (and their edges) from the master operations sequence that do not appear in the new variant. The operation precedence graph for the new variant is shown in Figure 5.8.



[redacted]

After obtaining the operations sequence for the new variant, it needs to be reviewed by planners in order to apply required modification. For instance, the new variant might require a new operation which none of the existing variants requires it. Therefore, planners should modify the obtained operations sequence in a way that the new operation also appears on it. This modification is more important when the obtained master operations sequence has non-zero dissimilarity distance from the existing variants. It indicates that there are some conflicts between the existing variants with respect to some operations steps and these operations steps might also appear in new variants. In such case, it is advised to first check if the new variants have those conflicting operations steps. If the answer is yes, then, those conflicting features of new variants should be examined, by considering other features, and determining the appropriate way of performing them. It should be noted that this research is based on the assumption that all the product variants, including the new variants, belong to the same product family and share many similarities and hence, only few aspects of them may vary from each other.

5.7 Numerical Results

The system used for the experiment in this section is the same as the one described in Section 5.6. Several numerical tests were conducted and the results obtained from the developed mathematical models were compared with the solutions obtained from the proposed algorithm. Both serial and generalized MIP models were implemented in GAMS optimization software and solved with CPLEX 12.0.1 MILP solver. For the proposed master generating algorithm, Borland C++ compiler version 5.02 was applied.

Different sets of problems were considered for each model to encompass small, medium and large sizes of problems. For the number of operations *(o)*, 3, 5, 7, 10, 15, 25 and 50 were selected while the number of variants *(n)* was set to be 5, 10, 20, and 50. For each combination, 10 replicates were conducted and examined. Inasmuch as GAMS optimization software is not able to find the exact solutions for large sizes of problems in a reasonable computing time, it was terminated after 60 minutes for some cases. Regarding the performance measures, two important criteria namely relative average percentage error (%RE) and run time were considered to assess the performance of each model and the proposed algorithm. The relative percentage of error can be obtained using the following equation:

$$\%RE = \frac{\text{the solution found by proposed algorithm (or by GAMS)} - \text{best solution)}}{\text{best solution}} \times 100 \qquad (5.22).$$

The best solution in above formula can be obtained by either GAMS or the proposed algorithm. In large problem sizes, GAMS may not be able to find the exact solution in 60 minutes and in such case, the solution obtained from the proposed algorithm might be better than GAMS's solution.

Each mathematical model is compared with the proposed master generating algorithm separately. Tables 5.1 and 5.2 summarize the results.

Table 5.1 Comparison of solutions of the first MIP model using GAMS and the proposed master generating algorithm

		GAMS (Serial MIP model)		Proposed master generating algorithm	
o (operations)	n (variants)	%RE	Avg. time (Sec.)	%RE	Avg. time (Sec.)
3	5	0	0.666	0	< 0.000001
	10		0.651		
	20		0.65		
	50		0.654		
5	5		0.658		
	10		0.646		
	20		0.645		
	50		0.781		
7	5		0.641		
	10		0.648		
	20		0.771		

	50		0.652		
10	5		0.676		
	10		0.671		
	20		0.802		
	50		0.834		
15	5		2.045		
	10		1.95		
	20		1.308		
	50		1.702		
25	5		26.833		
	10		29.566		
	20		23.607		
	50		24.758		
50	5	125	3600		0.031
	10	68.75	3600		0.031
	20	310	3600		0.032
	50	415	3600		0.036

Table 5.2 Comparison of solutions of the second model using GAMS and the proposed master generating algorithm

		GAMS (Generalized MIP model)		Proposed master generating algorithm	
o (operations)	n (variants)	%RE	Avg. time (Sec.)	%RE	Avg. time(Sec.)
3	5	0	0.846	0	< 0.000001
	10		0.851		
	20		0.853		
	50		0.86		
5	5		0.869		
	10		0.801		
	20		0.853		
	50		0.961		
7	5		6.018		
	10		5.965		
	20		5.975		
	50		6.089		
10	5	out of memory			
	10				
	20				
	50				

The most significant finding from these two tables is that the proposed master generating algorithm was able to find the exact solutions for all the studied problems in a very short time (less than 0.000001 seconds in most cases). It verifies the efficiency of the proposed algorithm in terms of accuracy and run time. Besides, finding the same solutions by both GAMS and the proposed master generating algorithm verifies the accuracy of both developed serial and generalized mathematical models.

It is also observable from Table 1 that the average computing time is not sensitive to the number of variants and in fact it is the number of operations which significantly affects the computing time for the mathematical models.

According to Table 5.1 and the serial mathematical model, GAMS is capable of finding exact solutions in reasonable computing time until o=25. When the number of operations increases to 50, regardless of the number of variants, GAMS cannot find the solution in 3600 seconds and it has been then terminated and the best solution obtained until that time has been recorded. For that reason, the relative percentage of error for GAMS is not zero when o=50.

Table 5.2 refers to the comparison of solutions obtained from the generalized mathematical model and the proposed master generating algorithm. As mentioned before, the second mathematical model is more complicated than the first model as it can apply for variants having serial, networked or combination of both operations sequences. The number of operations corresponds to the number of constraints used to prevent directed loops in the generalized mathematical model.Thus, as soon as the number of operations rises to 10, GAMS software yields “out of memory” message. In contrast, the proposed master generating algorithm, even when the number of operations is high, can find the solutions in a very short time.

5.8 Case Study

In this section, two case studies, one for assembly application and the other for fabricating application, are provided to demonstrate how a master operations sequence is derived from existing variants and used for constructing the operations sequences of new variants in real manufacturing environment.

5.8.1 Case Study 1: Assembly Application

This case study considers five variants of pilot control valves. A pilot control valve is a small valve which controls the flow and typically has emergency and safety applications. These valves are used in different industries such as food, beverage, paper and pharmaceutical industries. In this case study, all information about the five variants has been retrieved from Dorot® control valves catalogs (http://www.dorot.com/) with minor changes in some variants for better illustration of the model application. The first four variants are assumed to be the existing variants and their operations sequences are available in advance while the last pilot control valve is considered a new variant and there is no information about its operations sequence. In this case study "operation" mostly refers to an assembly process. Figure 5.9 illustrates the four existing variants of pilot control valves.

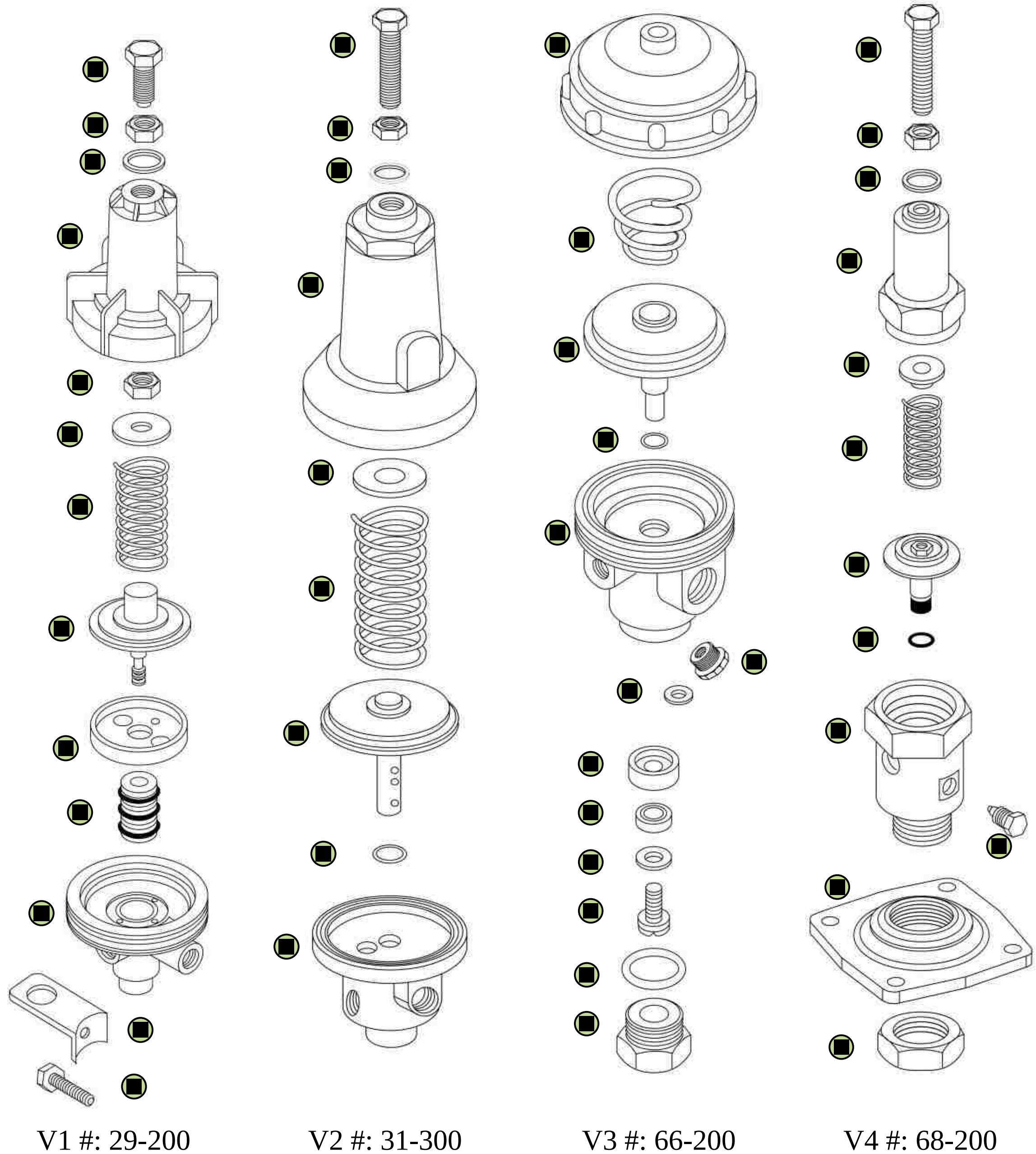


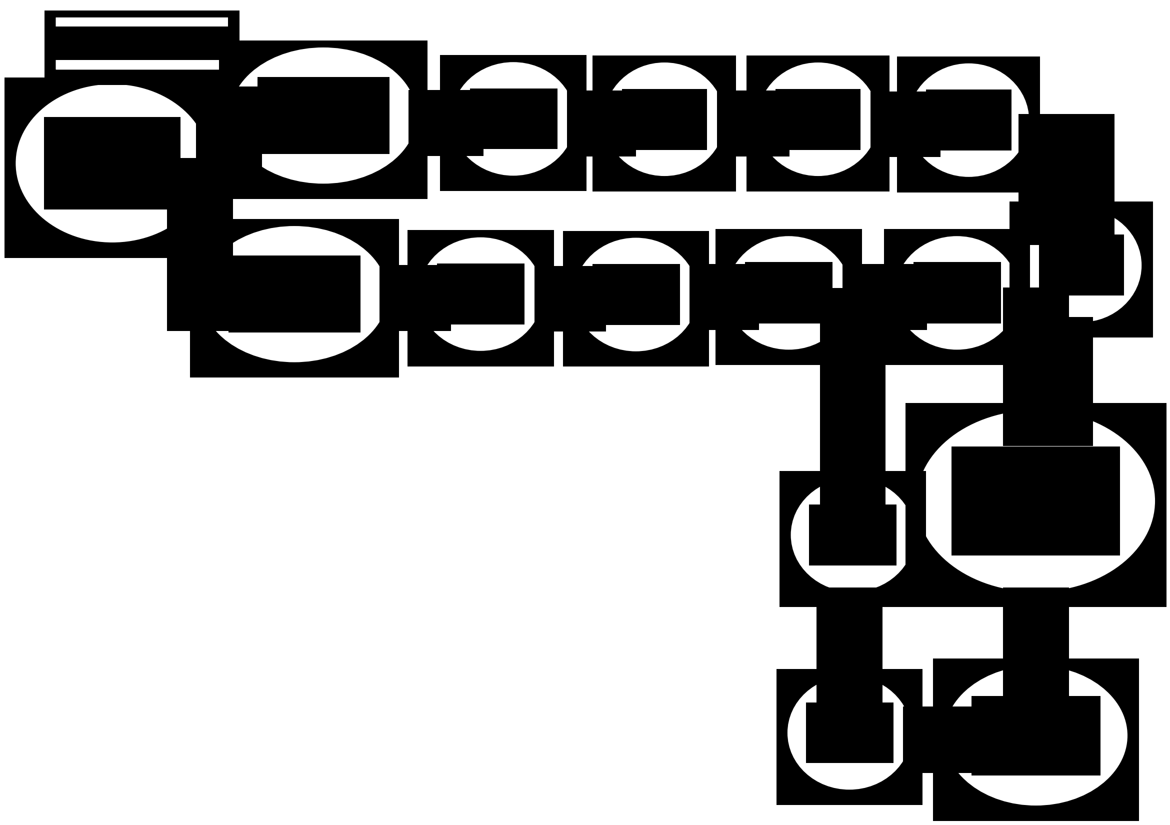
Figure 5.9 Four existing variants of pilot control valves for the case study retrieved from Dorot® control valves catalogs (http://www.dorot.com/)

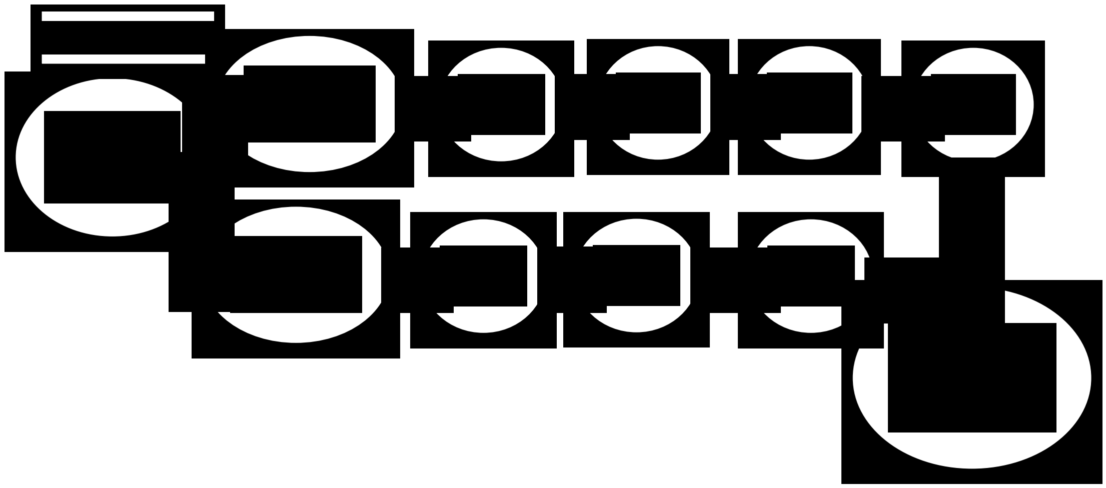
Table 5.3 also shows the name of each part used in the existing pilot control valves.

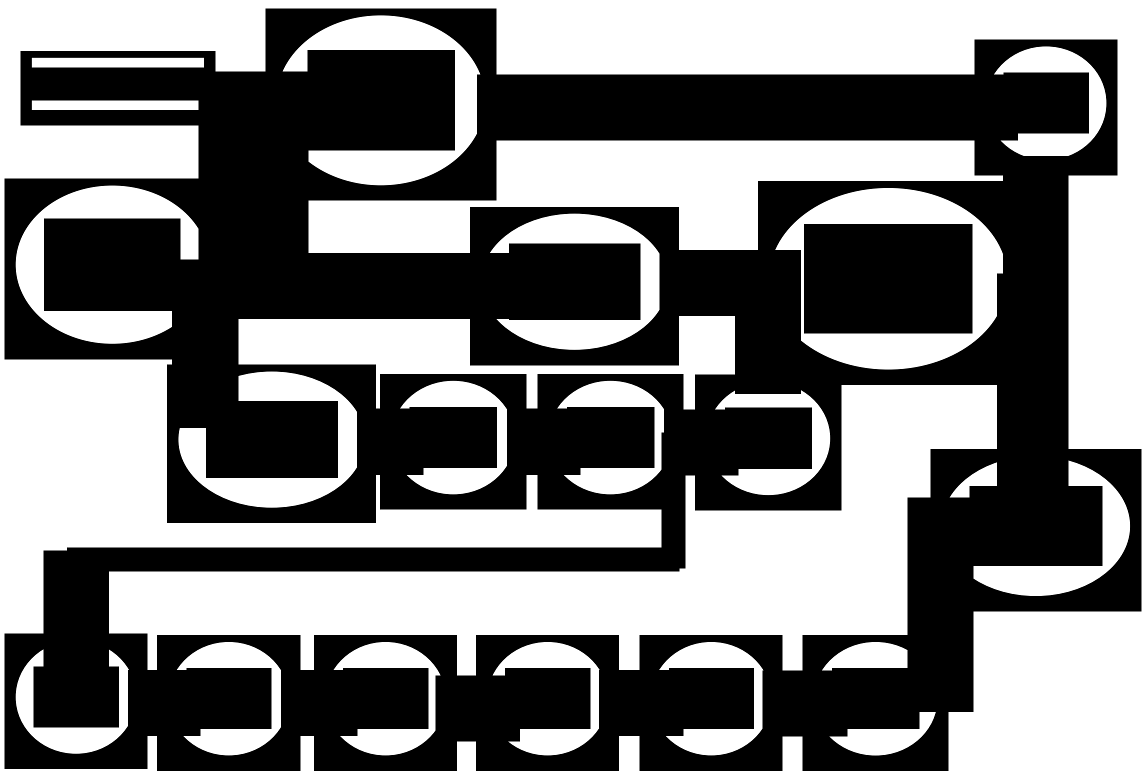
Table 5.3 The name of the components used in the existing variants (case study)

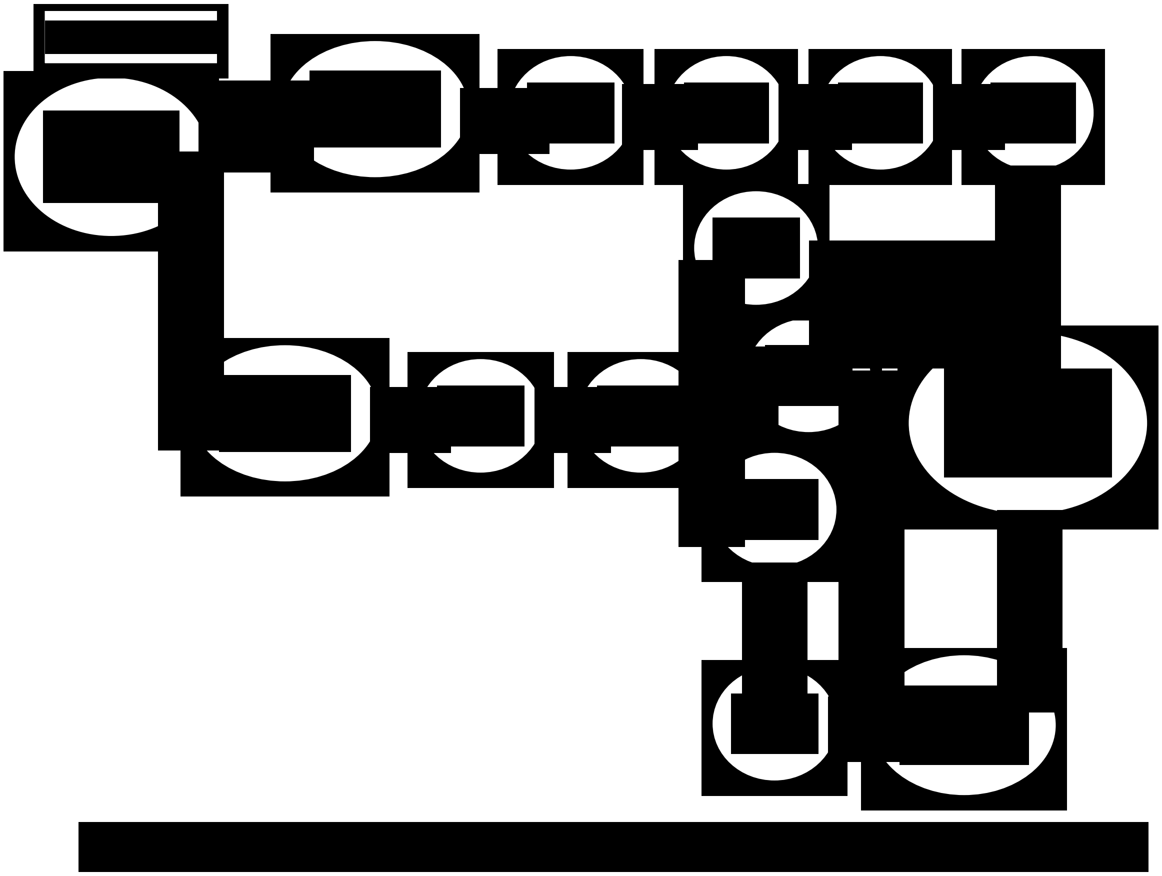
Part #	Part Name	**Part #**	Part Name
1	Adjusting Bolt	14	Body O-Ring
2	Locking Nut	15	Plug
3	Spring ID Ring	16	Seal for Plug
4	Bonnet	17	Seal Housing
5	Bonnet Nut	18	Seal
6	Spring Disc	19	Seal Disc
7	Spring	20	Seal Bolt
8	Diaphragm Assembly	21	Seat Washer
9	Diaphragm Bowl	22	Seat
10	Ports Selection Body	23	Nozzle Bolt
11	Body	24	2"-QR Bonnet
12	Bracket	25	Pilot Locking Nut
13	Bracket Bolt		

Figure 5.10 shows the operations precedence graphs of the existing variants which are assumed to be available in advance.

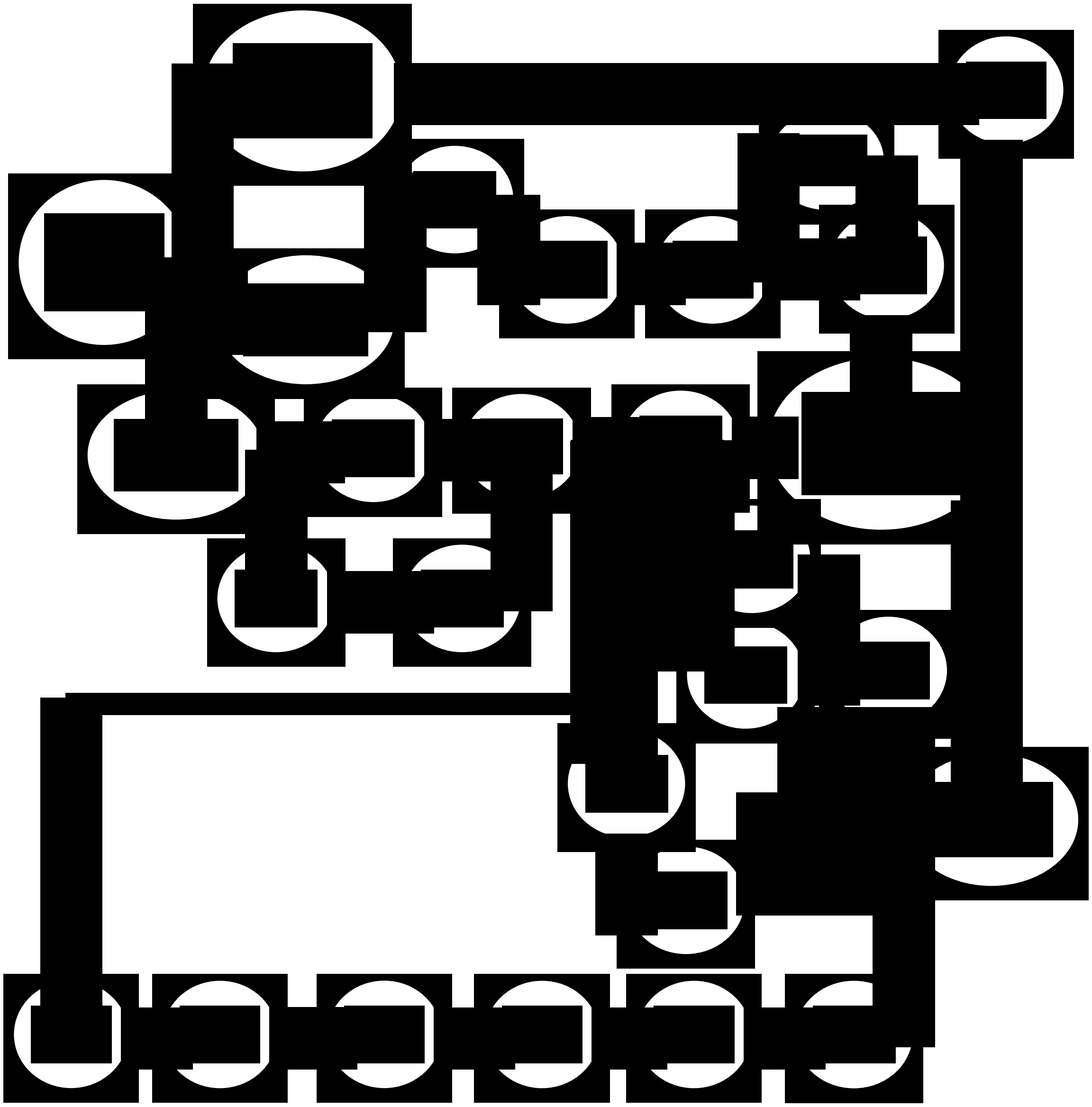






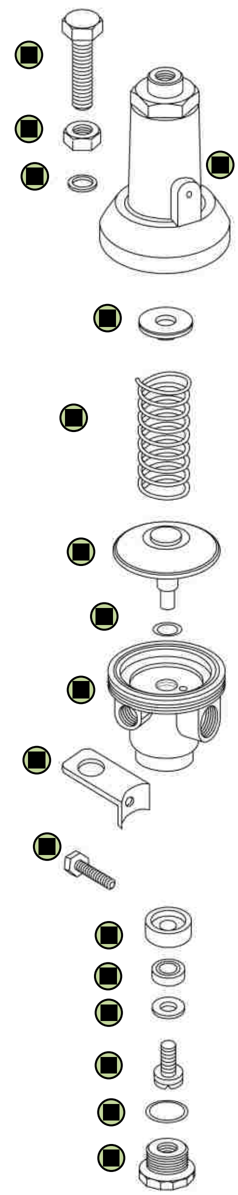


In Figure 5.10, “ASM” and “Sub-ASSY” refer to “assembling” and “sub-assembly” respectively. According to the graphs in Figure 5.10, each product variant consists of several components that may or may not appear in other variants. In addition, depending on the features and components, the assembly steps may differ between variants. Using these operation precedence graphs, the master operations sequence is generated by applying the proposed master generating algorithm. It took less than 0.000001 seconds to generate the master graph. Figure 5.11 shows the obtained master operation precedence graph which has an objective function value of 3 indicating that there are some conflicts/inconsistencies among the four existing variants. For instance, assembling Component 4 in Variant 3 is different from the ones in other variants.



[redacted]

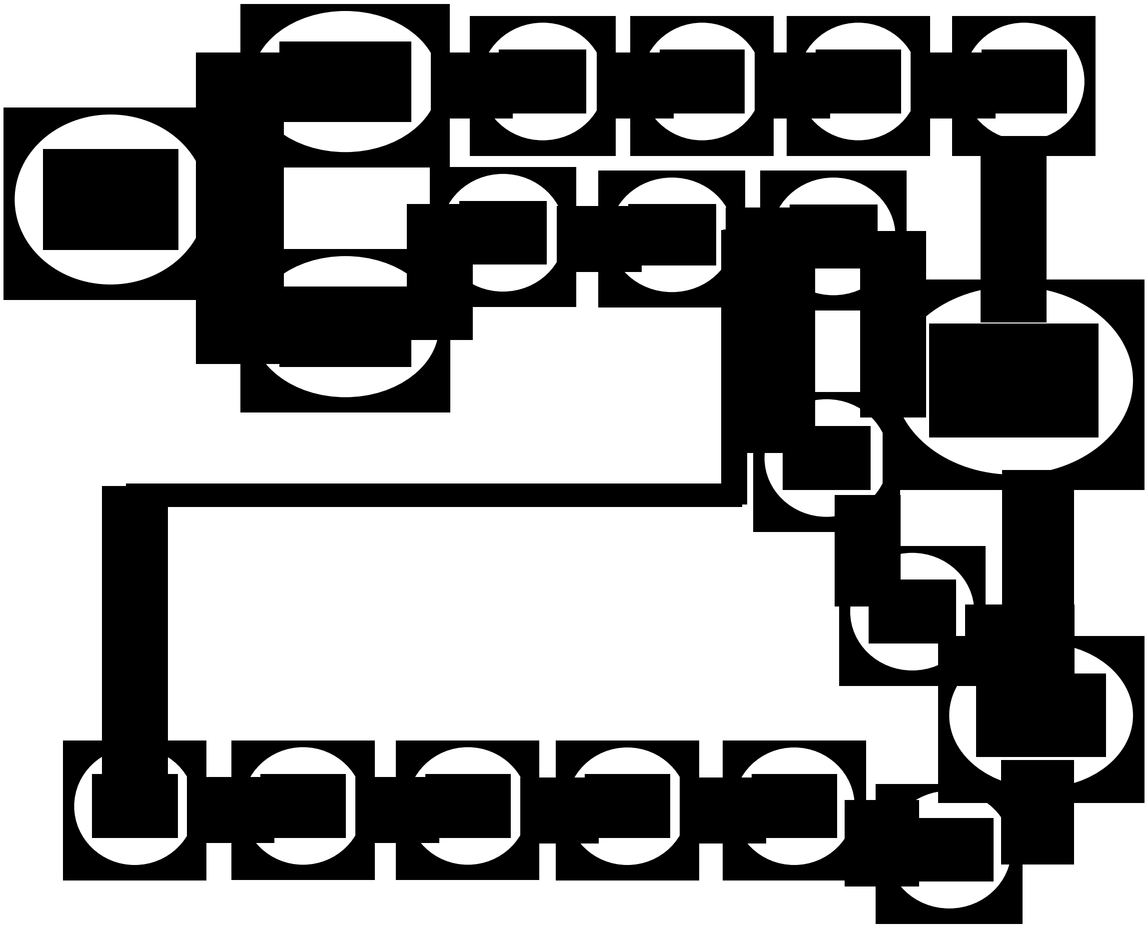
Now, suppose that a new variant (V5#: 68-700) is introduced which is illustrated in Figure 5.12. No information regarding the operations sequence of this variant is available.



V5#: 68-700

Figure 5.12 Components of the new poilot control valve in the case study retrieved from Dorot® control valves catalogs (http://www.dorot.com/)

By using the obtained master operations sequence, the new variant can be processed in line with the existing variants. By removing the components not appeared in the new variant from the generated master operations sequence, the operations sequence for the new variant can be obtained as shown in Figure 5.13.



[redacted]

Comparing the obtained operations sequence of the new variant with its components (Figure 5.12) and also with the operations sequences of the existing variants verifies that the assembly steps obtained for the new variant from the proposed master generating algorithm are logical and consistent with the assembly steps of existing variants. As can be observed in Figure 5.12, the components of the new variant do not all appear in a single existing variant; some of its components appear in some existing variants while the remaining components appear in other existing variants. Therefore, the assembly steps for the new variant would be as follows: for Parts 1,2,3,4 and 6 the assembly steps would be almost the same as the ones for Variants 1,2 and 4, for Parts 7, 8, 11 and 14 should be similar to the ones for Variants 2, 3 and 4 while for Parts 12 and 13 it should be the same as the ones for Variant 1. Finally, Parts 17 to 22 only appear in Variant 3 and hence, the assembly steps for these parts would be similar to the assembly steps of Variant 3. This is what exactly the generated master operations sequence does; i.e. aggregating the information of all existing variants to be used for retrieving the operations steps of new variants.

By using the proposed algorithm in generating master operations sequence which takes less than 0.000001 seconds in most cases and then, removing the components not appearing in the new variant from the generated master operations sequence, the time of retrieval would be less than a minute. This will result in less time and effort in developing the operations sequence of the new variant from the scratch and less reconfiguration in system level. It should be noted that it is still required that the obtained operations sequence for the new variant is reviewed by the planner and required modifications are applied specially when the new variant has some new components that did not exist in any of the existing variants.

5.8.2 Case Study 2: Machining Application

This case study considers nine variants of mostly ejecting and coupling parts/components produced by Rabourdin Industry (http://www.rabourdin.fr/). These parts can be used in different applications such as automotive, power transmission, locomotive, manufacturing machinery and conveyors. Figure 5.14 illustrates the first eight variants with their names which are used as existing variants. The ninth variant is considered the new variant. Most of the required operations in this case study are machining processes.

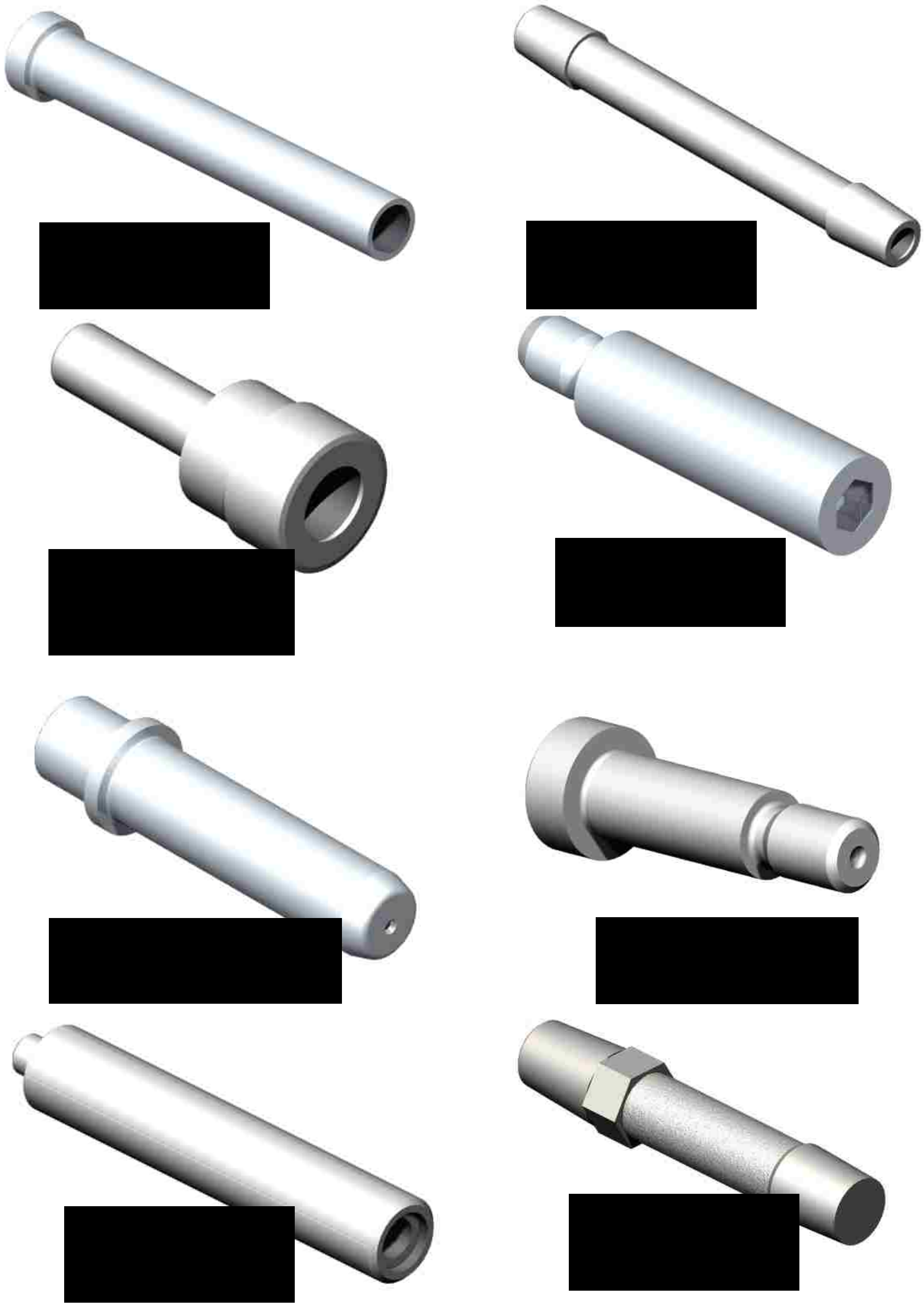
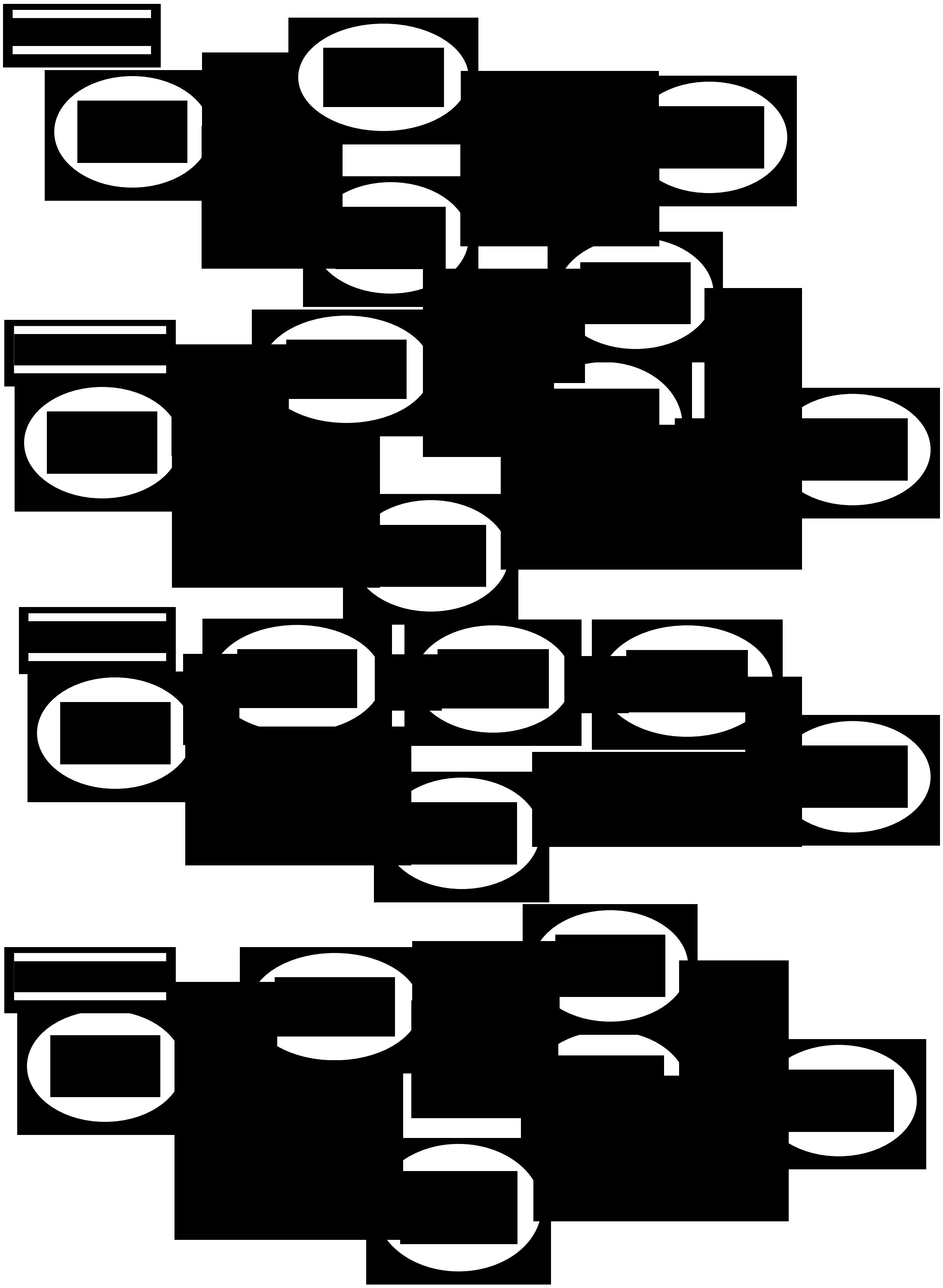


Figure 5.14 Eight existing variants of different parts manufactured by Rabourdin Industry (http://www.rabourdin.fr/)

Figure 5.15 shows the operations precedence graphs of the eight existing variants based on the operations listed in Table 5.4. Explanation is provided for some operations for clarity.

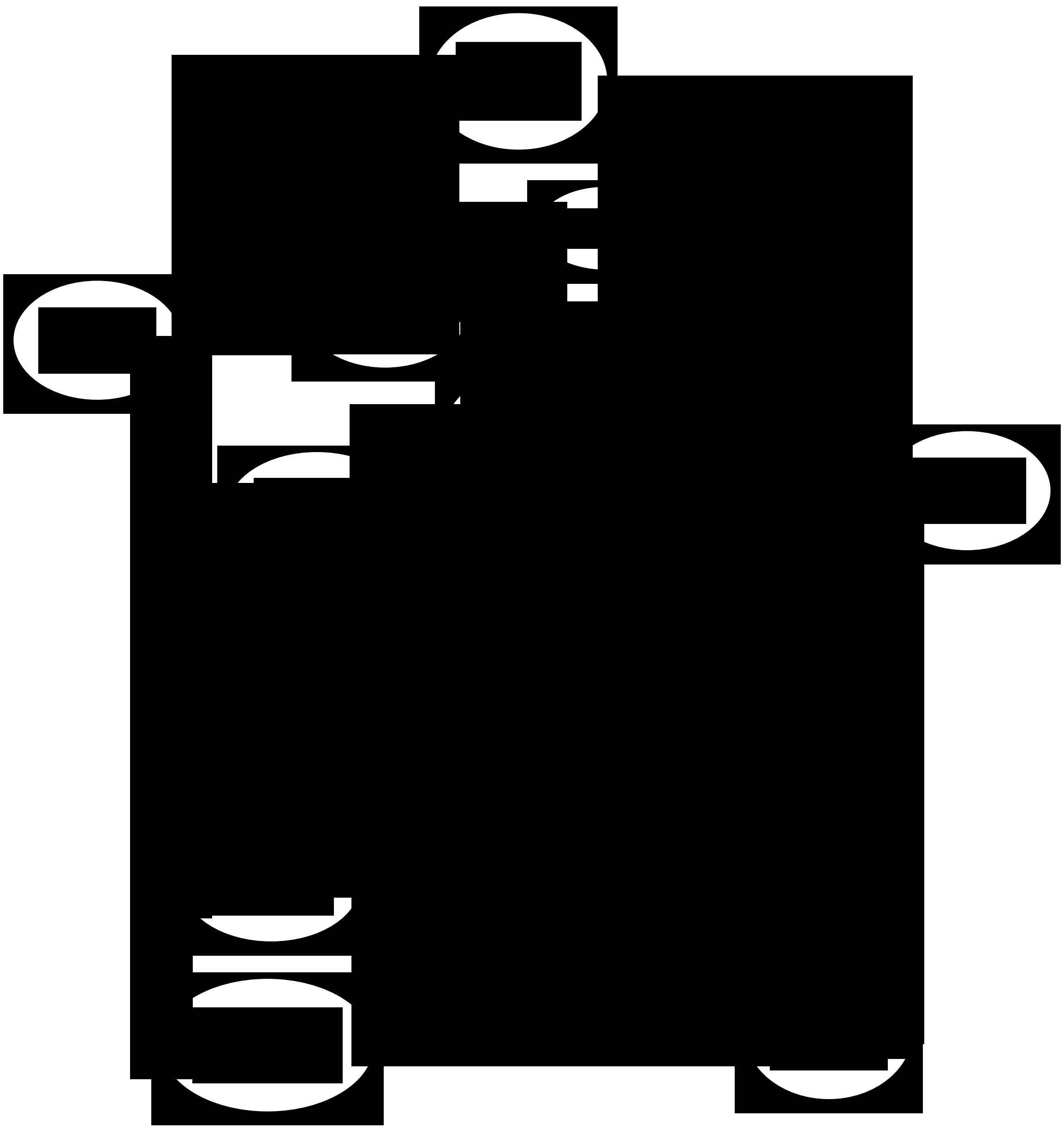
Table 5.4 List of operations' names used in the case study

<table>
<tr><th>No.</th><th>Operation's name and description</th><th>No.</th><th>Operation's name and description</th></tr>
<tr><td>1</td><td>Saw Cutting</td><td>8</td><td>Taper Turning</td></tr>
<tr><td>2</td><td>Cylindrical Turning 1 (first round of stepping)</td><td>9</td><td>Taper Turning-other side</td></tr>
<tr><td>3</td><td>Cylindrical Turning -other side</td><td>10</td><td>Chamfering</td></tr>
<tr><td rowspan="2">4</td><td rowspan="2">Cylindrical Turning 2 (second round of stepping):
This operation is done on a surface, which has already undergone cylindrical turning, to have a deeper cutting.</td><td>11</td><td>Under Cutting</td></tr>
<tr><td>12</td><td>Fillet</td></tr>
<tr><td>5</td><td>Gun Drilling:
This operation is applied for deep drilling.</td><td>13</td><td>Boring</td></tr>
<tr><td>6</td><td>Hexagon Hole Drilling</td><td rowspan="2">14</td><td rowspan="2">Quality Control:
This operation is for checking and comparing the actual dimensions of the finalized part with specifications.</td></tr>
<tr><td>7</td><td>Horizontal Milling</td></tr>
</table>

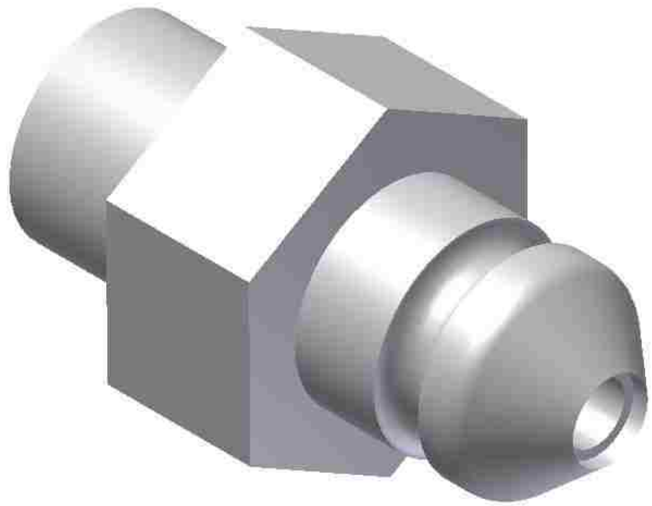


[redacted]

Using the proposed algorithm, a master operations sequence based on the eight existing variants is constructed. The obtained master operations sequence has an objective value of 8 (non-zero) meaning that some conflicts exist among the existing variants regarding their operations sequences. In such cases, more attention is required after extracting the operations sequence for a new variant as it may require some modifications and revisions based on those conflicting features. Figure 5.16 illustrates the obtained master operations sequence from the eight existing variants.

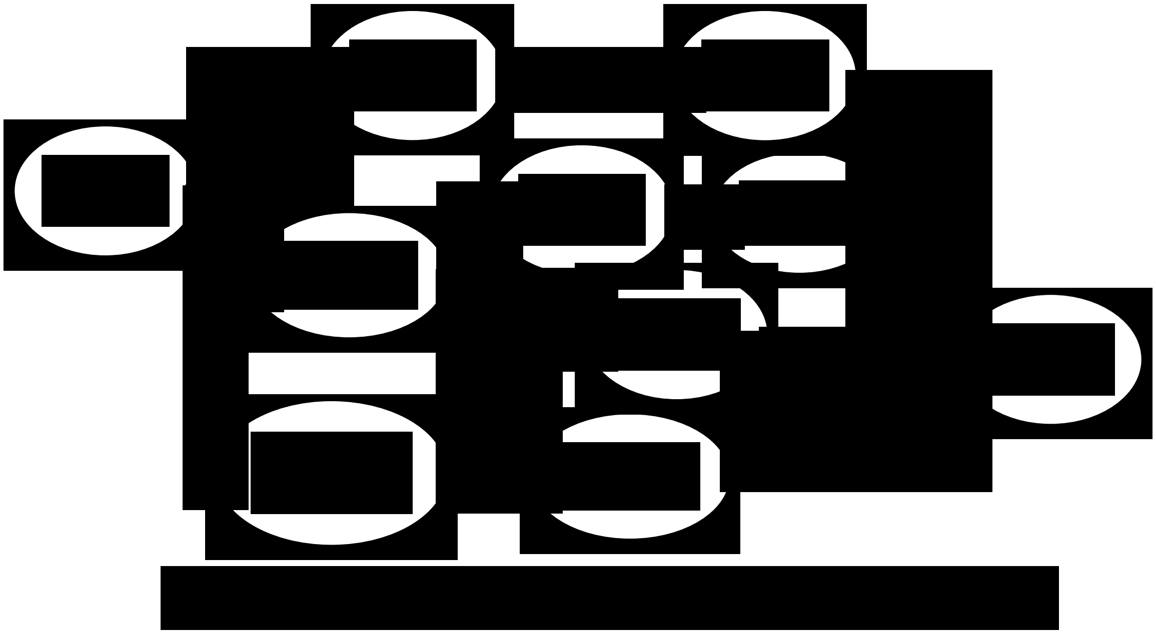


[redacted]



[redacted]

The operations sequence of the new variant (Bushed ejector rod) (Figure 5.17) is not available in advance. However, the processes that this variant undergoes are known as follows: Saw cutting, Gun drilling, Horizontal milling, Cylindrical turning 1, Quality control, Taper turning, Under cutting, Boring, and Fillet. The operations sequence of the new variant is obtained by removing those processes (manually or using a program) that are not required for the new variant from the generated master operations sequence. Here, the unnecessary operations were removed manually. Figure 5.18 shows the operations sequence of the new variant obtained from the master operations sequence. Comparing the obtained operations sequence in Figure 5.18 with the operations sequences of the existing variants in Figure 5.15 verifies the fact that the operations sequence of the new variant is consistent with the operations sequences of the existing variants. As mentioned before, the operations sequence for the new variant obtained by the developed retrieval method should be reviewed by the planner and based on features of the new variant, required modifications should be applied.



5.9 Summary

In this chapter, two MIP models were developed to generate a master operations sequence from operations sequences of existing variants of a part/product family. Review of the previous works reveals lack of models addressing this issue. The first MIP model was developed for product variants with serial structure of operations sequences. The second MIP model is a generalized form of the first model which can deal with serial and networked operations sequences or combination of both structures. Therefore, the derived master operations sequence from the generalized model can be serial or networked. The obtained master operations sequence has the minimum total dissimilarity distance from the existing variant. Inasmuch as the developed mathematical models can find a master operations sequence for small to medium sizes of problems (in terms of total number of operations), a novel master generating algorithm was proposed which is capable of finding the optimum master operations sequence in all sizes of the studied problems in less than 0.036 seconds. Two case studies were also used to show the application of the developed models and the proposed algorithm to both assembly and fabricating applications. The obtained master operations sequence has several advantages. It helps planners to spend less time and effort in development of operations sequences of new variants which are within or significantly overlap with the scope of the existing variants. It also guarantees that the operation steps of a new variant are in line with those of the existing product variants leading to less changeover time between variants on the shop floor.

Chapter 6
Conclusions and Future Work

6.1 Research Significance and Achievements

In this research, innovative mathematical models have been developed and novel algorithms have been proposed to fill identified research gaps existing in four different manufacturing systems domains namely production sequencing, product family formation, production flow, and operations sequence retrieval.

In some flow shop environments, total setup times between product variants have an important role and are considered more significant than processing times. Therefore, optimal sequence of the various product variants is based on improving machine utilization. In the first part of the research, a permutation flow shop environment was considered where bypassing was not permitted. A mathematical model was developed and solved using GAMS software. The non-linear model was then reformulated to find the exact solutions. A novel policy for sequencing product variants was proposed and its results were compared to the exact solutions obtained from GAMS. This sequencing policy capitalizes on the commonality between product variants to increase sequencing efficiency. The developed sequencing policy is simple and easy to apply in manufacturing environments. A case study in the labels stickers making industry was used for demonstration and validation. Numerical results indicate that the proposed sequencing policy is capable of finding good solutions, and optimum ones in some cases, in less than 0.02 seconds for all small, medium and large studied problem sizes. The solutions obtained using the proposed sequencing policy has a total average of 1.2% relative error which is quite comparable with the solutions obtained using GAMS. Besides, according to the case study of six variants of label stickers, the obtained sequence from the proposed sequencing policy could reduce the total setup time by 42% and improve the productivity by 12% when compared with a random sequence. Using the results obtained in the first phase of this research can help production planners to benefit from the proposed sequencing policy and try to decrease the changeover time and improve utilization in the system.

Operations flow based similarity is an important criterion for grouping variants. Similarity coefficient for product variants with networked sequence of operations has not been considered in the literature. Previously proposed similarity coefficients, which are based on operation/assembly sequence, focused on variants with serial operations sequences where the order of processing operations is fixed; while in practice, there are many part/product variants with flexible operations sequence options. In the second part of the research, a novel similarity coefficient for part/product variants was proposed based on the networked operations sequence similarity inspired by the analysis used in the field of biology (e.g. enzymes structures comparison). An extension of the proposed coefficient was also presented with an example for illustration. A more comprehensive similarity coefficient was developed by including operations similarity and production volume criteria. The popular operations similarity coefficient, called Jaccard's similarity, was applied and extended. A new coefficient using production volume similarity criterion was also developed. Part/product variants are then clustered and grouped based on the integrated similarity coefficient using the average linkage clustering (ALC) algorithm. The main applications of the proposed similarity coefficient were addressed. The grouped variants were sequenced as a secondary application of the proposed similarity coefficient. The sequence obtained from the proposed approach was compared with that obtained from a developed mathematical model. The result showed the accuracy of the proposed sequencing approach and can serve as a good preliminary sequence. A case study was also provided for demonstration with considering seven different product variants. The case study shows that the sequence obtained from the proposed sequencing approach improves the objective function (combination of changeover time and utilization) by 15.7% on average and it yields the optimum solution verified by the solution obtained from the mathematical model. Using the similarity coefficient proposed in the second phase of this research can help the manufacturers to decrease the changeover time and intercellular movement and improve utilization and productivity of the system by grouping similar variants and assigning them to a single cell. It can also help production planners to sequence variants based on the proposed similarity coefficient, if the aforementioned conditions are met, leading to less changeover time.

After grouping product variants based on their similarity, facility assignment with respect to backtracking minimization was taken into consideration in the third phase of the research. Minimizing backtracking can reduce the wasted time and increase availability of parts/products to machines leading to improving machine utilization and throughput of the system. Therefore, a novel mathematical model was developed to minimize total backtracking distance by considering product variants having networked operations sequences and also considering their production volumes. In this regard, three problems were solved simultaneously in one model: (1) how to locate machines in candidate places, (2) how to assign operations to machines based on machine capability, and (3) how the final sequence of performing operations should be. The developed model was non-linear at first which was then reformulated to its equivalent MIP model in order to use LP solvers to find the exact solutions. A case study of three different variants of engine blocks was also provided for demonstration. Based on the obtained results, backtracking minimization improved the total throughput by 7.79% in the case study. Therefore, using the developed model can help manufacturers to increase availability of jobs to machines and to improve total throughputs of their systems.

In the last phase of the research, two new mixed integer programing (MIP) models were developed and a novel algorithm was proposed for generating master operations sequence based on available operations sequences of a family of part/product variants. The generated master operations sequence can be used to construct the operations sequence for new variants falling within or significantly overlapping with the boundary of the considered family. Literature review indicates that no research has been carried out for generating master operations sequence. The main advantages of the master operations sequence include reducing time, cost and effort required for developing operations sequences of new variants and hence, improving productivity of the system. It also guarantees that the operation steps of the new variant are consistent with the operation steps of the existing variants resulting in less changeover time from processing existing variants to the new one. The first MIP model was developed for variants having serial operations sequence while the second model was a generalized model encompassing variants with serial or networked operations sequences or the combination of both structures. The proposed master generating algorithm was able to deal with variants with various structures. The developed models and algorithm are used to find a master operations sequence which has the minimum total dissimilarity distance from existing variants. A case study was also provided for demonstration of the problem. By using the proposed algorithm in generating master operations sequence which takes less than 0.000001 seconds in most cases (up to 50 variants and 25 operations) and then, removing the components not appearing in the new variant from the generated master operations sequence, the time of retrieval would be less than a minute. It helps planners to obtain the operations sequences of new variants much faster than starting from scratch. It also ensures that the operation steps of the new variant are consistent with the operation steps of the existing variants resulting in less changeover time and improving productivity.

6.2 Contributions

In this section, the contributions of the research made at each topic are presented. As mentioned before, this research included four different manufacturing systems domains namely production sequencing, product family formation, production flow, and operations sequence retrieval.

6.2.1 Production Sequencing

This topic was presented in Chapter 2. The contributions are as follows:

- A new mathematical model was developed for a permutation flow shop.
- For the first time, a novel setup similarity coefficient was proposed for this problem.
- A novel sequencing policy based on the proposed setup similarity coefficient was developed for the first time which is quite efficient in terms of accuracy and runtime.

6.2.2 Product Family Formation

This topic was presented in Chapter 3. The contributions are as follows:

- A novel operations flow based similarity coefficient inspired by the analysis used in the field of biology was proposed for the first time for variants having network structure.
- The proposed coefficient then was extended to the cases in which the connections between stations have different importance weight.
- An extension to Jaccard's similarity coefficient was proposed for cases that different operations have different importance weight.
- A novel production volume similarity was proposed to obviate the drawback of the most common coefficient in literature.
- A mathematical model was developed for sequencing application (secondary application) of the integrated similarity coefficient.
- The sequencing policy used for the first topic was applied and its results were compared with the mathematical model indicating the efficiency of the policy in terms of time and accuracy.

6.2.3 Production Flow

This topic was presented in Chapter 4. The contributions are as follows:

- Minimizing backtracking in a manufacturing system where product variants have networked structures and operations should be assigned to machines based on machine capability has never been addressed in literature.
- A novel mathematical model was developed capable of dealing with three decision variables simultaneously: (1) machine location, (2) operation assignments to machines and (3) final sequence of performing operations.

6.2.4 Operations Sequence Retrieval

This topic was presented in Chapter 5. The contributions are as follows:

- The considered problem of generating a master operations sequence has never been addressed in literature. Operation refers to both assembly and fabrication processes.

- Two new mathematical models were developed for generating master operations sequence from a family of existing variants. The first model was developed for variants having serial operations sequence while the second model, the generalized form of the first model, was developed for variants having different structure; i.e. serial, networked or combinations of both.
- A novel algorithm was also proposed for generating master operations sequence in much less time than the developed mathematical models.

6.3 Limitations and Future Work

In this section the limitations of the current research are mentioned and the potential directions for future work are discussed.

In Chapter 2, the proposed similarity coefficient and the sequencing policy were based on the setup criterion only. Other criteria that can be taken into consideration for future related research include process commonality in terms of lot sizing and processing time, due dates, job priorities and so on. In addition, some uncertainty such as stochastic setup times can be taken into consideration for future works and accordingly, a robust algorithm combined with simulation could be used to check the sensitivity of the algorithm.

In Chapter 3, three important similarity criteria were considered. Yet, the developed similarity coefficient may be extended in the future to include additional criteria such as alternate routing to have a more comprehensive similarity coefficient. In addition, the developed coefficient can also be extended for application in grouping and assigning manufacturing machines.

In Chapter 4, a mathematical model was developed to minimize total backtracking distance. Nevertheless, as the size of the problem increases, especially in terms of number of operations and machines (e.g. 30 operations and 6 machines), the efficiency of the developed model to solve the problem in a reasonable time decreases. Therefore, for future work, a heuristic or a Meta Heuristic algorithm can be used to deal with large sized problems.

In Chapter 5, two MIP models were developed and an algorithm was proposed to generate a master operations sequence. However, the obtained master operations sequence in this research is based on the available information of the existing variants and does not consider new features/components which may appear in new variants. This can be a subject for future research.

Moreover, the proposed algorithms and developed mathematical models have not been implemented in real manufacturing systems which can be done in the future for validation purposes. Besides, their accuracy versus runtime can be examined in industrial applications.

In some chapters, the developed mathematical models were not able to find the solutions for large size problems. One of the reasons for this drawback is the capability of the operating system by which the experiments were conducted. Usually, industrial problems are large sizes. Therefore, for such cases, using more powerful computers might be an option to obviate this issue.

REFERENCES

Algeddawy, T. and ElMaraghy, H., 2010a. Assembly Systems Layout Design Model For Delayed Products Differentiation. *International Journal Of Production Research,* 48, 5281-5305.

Algeddawy, T. and ElMaraghy, H., 2010b. Design of single assembly line for the delayed differentiation of product variants. *Flexible Services and Manufacturing Journal*, 22, 163-182.

Algeddawy, T. and ElMaraghy, H., 2012. Product variety management in design and manufacturing: challenges and strategies. *Enabling Manufacturing Competitiveness and Economic Sustainability.* Springer Berlin Heidelberg, 518-523.

Alhourani, F., 2013. Clustering algorithm for solving group technology problem with multiple process routings. *Computers & Industrial Engineering*,66, 781-790.

Alhourani, F. and Seifoddini, H., 2007. Machine cell formation for production management in cellular manufacturing systems. *International Journal of Production Research*,45 (4), 913-934.

Armentano, V.A. and Arroyo, J.C., 2004. An Application of a Multi-Objective Tabu Search Algorithm to a Bicriteria Flowshop Problem. *Journal of Heuristics*, 10, 463-481.

Aneke, N. and Carrie, A. 1986. A design technique for the layout of multi-product flowlines. *International Journal of Production Research,* 24**,** 471-481.

Askin, R.G. and Zhou, M., 1998. Formation of independent flowline cells based on operation requirements and machine capabilities. *IIE Transactions*, 30, 319–329.

Azab, A. and ElMaraghy, H., 2007 a. Mathematical modeling for reconfigurable process planning, *CIRP Annals-Manufacturing Technology*, 56, 467-472.

Azab, A. and ElMaraghy, H., 2007 b. Sequential process planning: A hybrid optimal macro-level approach, *Journal of Manufacturing Systems*, 26, 147-160.

Badrous, S.N.S., 2011. Complexity of products and their assembly systems [Doctoral Dissertation], University of Windsor.

Baker, K. and College, D., 2008. Sequencing: The Shortest Processing Time Rule. *In:* CHHAJED, D. & LOWE, T. (eds.) *Building Intuition.* Springer US.

Bhatnagar, R. and Saddikuti, V., 2010. Models for cellular manufacturing systems design: matching processing requirements and operator capabilities. *Journal of Operation Research Society*,61, 827-839.

Chang, C.C., Wu, T.H. and Wu, C.W., 2013. An efficient approach to determine cell formation, cell layout and intracellular machine sequence in cellular manufacturing systems. *Computers & Industrial Engineering*, 66(2), 438-450.

Choobineh, F., 1988. A framework for the design of cellular manufacturing systems. *International Journal of Production Research*, 26, 1161–1172.

Daie, P. and Li, S., 2016. Managing product variety through configuration of pre-assembled vanilla boxes using hierarchical clustering. *International Journal of Production Research*,1-12.

Davies, A., John, E. and Thomas, A. 2013. A flow direction weighting scheme for facility layout in cellular manufacturing. *International Journal of Productivity and Performance Management*, 62**,** 185-203.

Eguia, I., Lozano, S., Racero, J. and Guerrero, F., 2011. A methodological approach for designing and sequencing product families in Reconfigurable Disassembly Systems. *Journal of Industrial Engineering and Management*, 4, 418-435.

Eguia, I., Racero, J., Guerrero, F. and Lozano, S., 2013. Cell formation and scheduling of part families for reconfigurable cellular manufacturing systems using Tabu search. *Simulation*, 89, 1056-1072.

ElMaraghy, H.A. 2005. Flexible and reconfigurable manufacturing systems paradigms. *International Journal of Flexible Manufacturing Systems*, 17(4):261-276.

ElMaraghy, H. and Laperrière, L. 1992. Modelling and sequence generation for robotized mechanical assembly. *Robotics and Autonomous Systems*, 9(3): 137-147.

ElMaraghy, H., Schuh, G., ElMaraghy, W., Piller, F., Schönsleben, P., Tseng, M. and Bernard, A., 2013. Product variety management. *CIRP Annals - Manufacturing Technology*,62, 629-652.

Ferris, MC., and Voelker, MM. 2002. Slice models in GAMS. *In Operations Research Proceedings 2001*, 239-246, Springer Berlin Heidelberg.

Fogliatto, F. S., Da Silveira, G. J. C. and Borenstein, D., 2012. The mass customization decade: An updated review of the literature. *International Journal of Production Economics*, 138, 14-25.

Forghani, K., Mohammadi, M. and Ghezavati, V. 2015. Integrated cell formation and layout problem considering multi-row machine arrangement and continuous cell layout with aisle distance. *The International Journal of Advanced Manufacturing Technology*, 78**,** 687-705.

Galan, R., Racero, J., Eguia, I. and Canca, D., 2007a. A methodology for facilitating reconfiguration in manufacturing: The move towards reconfigurable manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 33, 345-353.

Galan, R., Racero, J., Eguia, I. and Garcia, J. M., 2007b. A systematic approach for product families formation in Reconfigurable Manufacturing Systems. *Robotics and Computer-Integrated Manufacturing*, 23, 489-502.

Golmohammadi, A., Bani-Asadi, H., Esmaeeli, H., Hadian, H. and Bagheri, F. 2016. Facility layout for cellular manufacturing system under dynamic conditions. *Decision Science Letters,* 5**,** 407-416.

Gong, D., Yamazaki, G., Gen, M. and Xu, W. 1999. A genetic algorithm method for one-dimensional machine location problems. *International Journal of Production Economics,* 60**,** 337-342.

Goyal, K.K., Jain, P.K. and Jain, M., 2012. Operation sequence based similarity coefficient for RMS. *In : Proceedings of the 23rd International DAAAM Symposium*, 23(1), 281–284, Vienna, Austria.

Goyal, K.K., Jain, P.K. and Jain, M., 2013. A comprehensive approach to operation sequence similarity based part family formation in the reconfigurable manufacturing system. *International Journal of Production Research*, 51(6), 1762–1776.

Gupta, T., 1993. Design of manufacturing cells for flexible environment considering alternative routing. *International Journal of Production Research*, 31, 1259–1273.

Hamming, R.W., 1950. Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2),147–160.

He, H., 2016. Minimization of maximum lateness in an m-machine permutation flow shop with a general exponential learning effect. *Computers & Industrial Engineering*, *97*,73-83.

Hegge, H.M.H. and Wortmann, J.C., 1991. Generic bill-of-material: a new product model. *International Journal of Production Economics,* 23,117–128.

Heragu, S. S. and Kusiak, A. 1988. Machine layout problem in flexible manufacturing systems. *Operations Research*, 36**,** 258-268.

Heymans, M. and Singh, A., 2003. Deriving phylogenetic trees from the similarity analysis of metabolic pathways. *BIOINFORMATICS*,19, 138-146.

Hoffman, K., Padberg, M. and Rinaldi, G., 2013. Traveling Salesman Problem. *Encyclopedia of Operations Research and Management Science*. S. Gass and M. Fu, Springer US: 1573-1578.

Hogg, G. L. 2012. Backtracking of jobs. *Material Handling'90,* 2**,** 117.

Hu, S. J., Ko, J., Weyand, L., ElMaraghy, H. A., Lien, T. K., Koren, Y., Bley, H., Chryssolouris, G., Nasr, N. and Shpitalni, M., 2011. Assembly system design and operations for product variety. *CIRP Annals - Manufacturing Technology*, 60/2, 715-733.

Ilic, O.R., 2014. An e-Learning tool considering similarity measures for manufacturing cell formation. *International Journal of Advanced Manufacturing Technology*, 25, 617–628.

Ishibuchi, H., Yoshida, T. and Murata, T., 2003. Balance Between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flowshop Scheduling. *IEEE Transactions on Evolutionary Computation*, 7, 204-223.

Jiao, J., Tseng, M.M., Ma, Q. and Zou, Y., 2000. Generic Bill-of-Materials-and-Operations for high-variety production management. *Concurrent Engineering Research & Applications*, 8(4), 297-321.

Kashkoush, M. and ElMaraghy, H., 2014a. Product family formation for reconfigurable assembly systems. *In: Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 302-307, Windsor, Canada.

Kashkoush, M. and ElMaraghy, H., 2014b. Consensus tree method for generating master assembly sequence. *Production Engineering. Research and Development*, 8,233-242.

Kashkoush, M. and ElMaraghy, H., 2015. Knowledge-based model for constructing master assembly sequence. *Journal of Manufacturing Systems*, 34,43-52.

Khan, BSH. and Govindan, K., 2011. A multi-objective simulated annealing algorithm for permutation flow shop scheduling problem. *International Journal of Advanced Operations Management*, 3(1),88–100.

Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G. and Van Brussel, H., 1999. Reconfigurable manufacturing systems. *CIRP Annals - Manufacturing Technology*, 48/2, 527-540.

Koren, Y. and Shpitalni, M., 2010. Design of reconfigurable manufacturing systems. *Journal of Manufacturing Systems*, 29, 130-141.

Kumar, K. R. and Narendran, T., 1997. A heuristic for sequencing PCBs with due-dates. *International Journal of Operations & Production Management*, 17, 446-467.

Lai, H.Y., and Huang, C.T., 2003. Integrated assembly plan generation system for grouped product families. *International Journal of Production Research*, 41,4041–61.

Lee, S.-D. and Chiang, C.-P. 2002. Cell formations in the uni-directional loop material handling environment. *European Journal of Operational Research*, 137**,** 401-420.

Lei, D. and Wu, Z., 2005. Tabu search approach based on a similarity coefficient for cell formation in generalized group technology. *International Journal of Production Research*, 43 (19), 4035–4047.

Lemesre, J., Dhaenens, C. and Talbi, E.G., 2007. An exact parallel method for a bi-objective permutation flowshop problem. *European Journal of Operational Research*, 177, 1641-1655.

Li, T. and Li, Y., 2007. Constructive backtracking heuristic for hybrid flowshop scheduling with limited waiting times. In *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, IEEE, 6671-6674.

Lin, H. and Liao, C.J., 2003. A case study in a two-stage hybrid flow shop with setup time and dedicated machines. *International Journal of Production Economics,* 86, 133-143.

Lin, B.M.T. and Wu, J.M., 2006. Bicriteria scheduling in a two-machine permutation flowshop. *International Journal of Production Research,* 44, 2299-2312.

Lin, Q., Gao, L., Li, X. and Zhang, C., 2015. A hybrid backtracking search algorithm for permutation flow-shop scheduling problem. Computers & Industrial Engineering, 85, 437-446.

Lin, S.W. and Ying, K.C., 2013. Minimizing makespan and total flow time in permutation flowshops by a bi-objective multi-start simulated-annealing algorithm. *Computers & Operations Research,* 40, 1625-1647.

Lin, S.W., Ying, K.C. and Lee, Z.J., 2010. Part-machine cell formation in group technology using a simulated annealing-based meta-heuristic. *International Journal of Production Research,* 48 (12), 3579–3591.

Luo, H., Zhang, A. and Huang, G.Q., 2015. Active scheduling for hybrid flowshop with family setup time and inconsistent family formation. *Journal of Intelligent Manufacturing,* 26, 169-187.

Mahdavi, I., Shirazi, B. and Paydar, M. M. 2008. A flow matrix-based heuristic algorithm for cell formation and layout design in cellular manufacturing system. *The International Journal of Advanced Manufacturing Technology,* 39**,** 943-953.

Mahdavi, I., Teymourian, E., Baher, N. T. and Kayvanfar, V. 2013. An integrated model for solving cell formation and cell layout problem simultaneously considering new situations. *Journal of Manufacturing Systems,* 32**,** 655-663.

Martinez, M.T., Favrel, J. and Ghodous, P., 2000. Product family manufacturing plan generation and classification. *Concurrent Engineering Research & Applications*, 8,12–23.

McAuley, J., 1972. Machine grouping for efficient production. *The Production Engineer,* 51, 53–57.

M'Hallah, R., 2014. Minimizing total earliness and tardiness on a permutation flow shop using VNS and MIP. Computers & Industrial Engineering, 75, 142-156.

Moslehi, G., Mirzaee, M., Vasei, M., Modarres, M. and Azaron, A., 2009. Two-machine flow shop scheduling to minimize the sum of maximum earliness and tardiness. I*nternational Journal of Production Economics*, 122(2):763-773.

Mozdgir, A., Fatemi Ghomi, S. M. T., Jolai, F. and Navaei, J., 2013. Two-stage assembly flow-shop scheduling problem with non-identical assembly machines considering setup times. *International Journal of Production Research,* 51, 3625-3642.

Navaei, J. and ElMaraghy, H., 2016a. Grouping and sequencing product variants based on setup similarity. *International Journal of Computer Integrated Manufacturing*, DOI:10.1080/0951192X.2016.1190464, in press.

Navaei, J. and ElMaraghy, H., 2016b. Grouping part/product variants based on networked operations sequence. *Journal of Manufacturing Systems*, 38, 63-76.

Navaei, J. and ElMaraghy, H., 2014. Grouping product variants based on alternate machines for each operation. *In: Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 61-66, Canada.

Navaei, J., Fatemi Ghomi, S. M. T., Jolai, F., Shiraqai, M. E. and Hidaji, H., 2013. Two-stage flow-shop scheduling problem with non-identical second stage assembly machines. *International Journal of Advanced Manufacturing Technology,* 69, 2215-2226.

Nouri, H. and Hong, T.S., 2012. A bacteria foraging algorithm based cell formation considering operation time. *Journal of Manufacturing Systems*, 31, 326-336.

Ogata, H., Fujibuchi, W., Goto, S. and Kanehisa, M., 2000. A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Research*,28 (20), 4021-4028.

Pattanaik, L.N. and Kumar, V., 2010. Multiple levels of reconfiguration for robust cells formed using modular machines. *International Journal of Industrial and Systems Engineering,* 5 (4), 424–441.

Paydar, M. M., Mahdavi, I., Sharafuddin, I. and Solimanpur, M. 2010. Applying simulated annealing for designing cellular manufacturing systems using MDmTSP. *Computers & industrial engineering*, 59**,** 929-936.

Paydar, M.M., Saidi-Mehrabad, M. and Teimoury, E., 2014. A robust optimisation model for generalised cell formation problem considering machine layout and supplier selection. *International Journal of Computer Integrated Manufacturing,* 27, 772–786.

Piller, F., 2004. Mass Customization: Reflections on the State of the Concept. *The International Journal of Flexible Manufacturing Systems,* 16, 313-334.

Pinedo, M., 2012. *Scheduling: theory, algorithms, and systems*, Springer Science & Business Media.

Rifai, A.P., Nguyen, H.T. and Dawal, S.Z.M., 2016. Multi-objective adaptive large neighborhood search for distributed reentrant permutation flow shop scheduling. *Applied Soft Computing*, *40*,42-57.

Rahimi-Vahed, A.R. and Mirghorbani, S.M., 2007. A multi-objective particle swarm for a flow shop scheduling problem. *Journal of Combinatorial Optimization,* 13, 79-102.

Raja, A. and Anbumalar, V., 2016. A new method for integrating feasibility assessment and cell formation problem considering operation sequence. *International journal of Advanced Engineering Technology*,7(1), 544-551.

Rajkumar, K. and Narendran, T., 1998. A heuristic for sequencing PCB assembly to minimize set-up times. *Production Planning & Control,* 9, 465-476.

Romanowski, C.J. and Nagi, R., 2004. A data mining approach to forming generic bills of materials in support of variant design activities. *Journal of Computing and Information Science in Engineering*,4,316–328.

Saaty, T.L., 1980. The analytic hierarchy process. *New York: McGraw-Hill*.

Sarker, B. R., Wilhelm, W. E., Hogg, G. L. and Han, M.-H. 1991. Backtracking of jobs and machine location problems. *Material Handling'90.* Springer.

Sarker, B. R., Wilhelm, W. E., Hogg, G. L. and Han, M.-H. 1995. Backtracking of jobs in one-dimensional machine location problems. *European Journal of Operational Research,* 85**,** 593-609.

Shtub, A. and Maimon, O., 1992. Role of similarity measures in PCB grouping procedures. *International Journal of Production Research*, 30, 973-983.

Shu, T., Shou, C., Wang, S. and Lai, K., 2014. GBOM-oriented management of production disruption risk and optimization of supply chain construction. *Expert Systems with Applications,* 41,59-68.

Sodhi, M. S. and Sarker, B. R. 2003. Configuring flexible flowlines. *International journal of production research,* 41**,** 1689-1706.

Spiliopoulos, K. and Sofianopoulou, S., 2007. Manufacturing cell design with alternative routings in generalized group technology: reducing the complexity of the solution space. *International Journal of Production Research,* 45 (6), 1355–1367.

Tam, K.Y., 1990. An operation sequence based similarity coefficient for part families formations. *Journal of Manufacturing Systems,* 9 (1), 55–68.

Tambuskar, D.P., Narkhede, B.E. & Mahapatra, S.S, 2015. A novel algorithm for virtual cellular manufacturing considering real life production factors. *International Journal of Services and Operations Management,* 20 (2), 246–269.

Toktas, b., Azizoglu, M. and Koksalan, S.K., 2004. Two-machine flow shop scheduling with twocriteria: Maximum earliness and makespan. *European Journal of Operational Research,* 157, 286–295.

Triantaphyllou, E., 2000. Multi-Criteria Decision Making: A Comparative Study. Dordrecht, *The Netherlands: Springer*. p. 320.

Tseng, M. M. and Hu, S. J. 2014. Mass customization. *Cirp Encyclopedia of Production Engineering.* Springer.

Tseng, F.T., Stafford Jr., F. and Gupta, J.N.D., 2004. An empirical analysis of integer programming formulations for the permutation flowshop. *Omega,* 32, 285-293.

Tseng, M.M., Jiao, R. J. and Wang, C., 2010. Design for mass personalization. *CIRP Annals - Manufacturing Technology,* 59, 175-178.

Vakharia, A. J. and Wemmerlöv, U., 1995. A comparative investigation of hierarchical clustering techniques and dissimilarity measures applied to the cell formation problem. *Journal of Operations Management,* 13, 117-138.

Wang, S., Liu, M. and Chu, C., 2015. A branch-and-bound algorithm for two-stage no-wait hybrid flow-shop scheduling. *International Journal of Production Research,* 53, 1143-1167.

Won, Y. and Logendran, R., 2015. Effective two-phase p-median approach for the balanced cell formation in the design of cellular manufacturing system. *International Journal of Production Research,* 53(9), 2730-2750.

Wu, L. and Suzuki, S., 2015. Cell formation design with improved similarity coefficient methodand decomposed mathematical model. *International Journal of Advanced Manufacturing Technology,* 79:1335-1352.

Yagmahan, B. and Yenisey, M.M., 2010. A multi-objective ant colony system algorithm for flow shop scheduling problem. *Expert Systems with Applications,* 37, 1361-1368.

Yang, Q.H., QI, G.N., LU, Y.J. and GU, X.J., 2007. Applying mass customization to the production of industrial steam turbines. *International Journal of Computer Integrated Manufacturing,* 20, 178–188.

Yenisey, M.M. and Yagmahan, B., 2014. Multi-objective permutation flow shop scheduling problem: Literature review, classification and current trends. *Omega,* 45, 119-135.

Yin, Y. and Yasuda, K., 2002. Manufacturing cells' design in consideration of various production factors. *International Journal of Production Research,* 40 (4), 885–906.

Yin, Y. and Yasuda, K., 2006. Similarity coefficient methods applied to the cell formation problem: A taxonomy and review. *International Journal of Production Economics,* 101, 329–352.

Youssef, A.M.A. and ElMaraghy, H., 2006. Assessment of manufacturing systems reconfiguration smoothness. *International Journal of Advanced Manufacturing Technology,* 30, 174–193.

Zager, L., 2003. Graph similarity and matching. *Master thesis,* Massachusetts Insttitute of Technology.

Zhao, F., Zhang, J., Wang, J. and Zhang, C., 2015. A shuffled complex evolution algorithm with opposition-based learning for a permutation flow shop scheduling problem. *International Journal of Computer Integrated Manufacturing,* 28(11): 1220-1235.

VITA AUCTORIS

NAME:	Javad Navaei
PLACE OF BIRTH:	Tehran, Iran
YEAR OF BIRTH:	1986
EDUCATION:	Roshd High School, Tehran, Iran, 2005
	B.Sc. in Industrial Engineering, K.N.Toosi University of Technology, Tehran, Iran, 2009
	M.Sc. in Industrial Engineering, Amirkabir University of Technology, Tehran, Iran, 2011

PUBLICATIONS: **Journal Publications:**

[1] **Navaei, J.** and ElMaraghy, H., 2016. Optimal assignments of facilities in backtracking of product variants with networked operations sequence, *CIRP Journal of Manufacturing Science and Technology* (Under review).

[2] **Navaei, J.** and ElMaraghy, H., 2016. Mathematical models for generating master operations sequence, *Expert Systems with Applications* (Submitted).

[3] **Navaei, J.** and ElMaraghy, H., 2016. Operations sequence retrieval by construction of master operations sequence using a mathematical model and a novel algorithm, *European Journal of Operational Research* (To be submitted).

[4] **Navaei, J.** and ElMaraghy, H., 2016. Grouping and sequencing product variants based on setup similarity, *International Journal of Computer Integrated Manufacturing (IJCIM)*, DOI:10.1080/0951192X.2016.1190464, in press.

[5] **Navaei, J.** and ElMaraghy, H., 2016. Grouping part/product variants based on networked operations sequence, *Journal of Manufacturing Systems (JMS)*, 38, 63-76.

[6] **Navaei, J.;** Fatemi Ghomi, S.M.T.; Jolai, F.; Mozdgir, A., 2014. Heuristics for an assembly flow-shop with non-identical assembly machines and sequence dependant setup times to minimize sum of holding and delay costs, *Computers & Operations Research (COR)*, 44, 52-65.

[7] Mozdgir, A.; Fatemi Ghomi, S.M.T.; Jolai, F.; **Navaei, J.,** 2014. Three meta-heuristics to solve the no-wait two-stage assembly flow shop scheduling problem, *Scientia Iranica*, 20(6), 2275-2283.

[8] **Navaei, J.;** Fatemi Ghomi, S.M.T.; Jolai, F.; Shiraqai. M.E.; Hidaji, H., 2013. Two-stage flow-shop scheduling problem with non-identical second stage assembly machines, *International Journal of Advanced Manufacturing Technology (IJAMT)*, 69, 2215-2226.

[9] Mozdgir, A.; Fatemi Ghomi, S.M.T.; Jolai, F.; **Navaei, J.,** 2013. Two-stage assembly flow-shop scheduling problem with non-identical assembly machines considering setup times, *International Journal of Production Research (IJPR)*, 51(12), 3625-3642.

[10] Azadeh, A.; Zahedi Anaraki, A.; Jeihoonian, M.; **Navaei, J.,** 2012. Multi-criteria no-wait three-stage assembly flow-shop scheduling problem with sequence-dependent setup times. *Under Review in International Journal of Advanced Manufacturing Technology (IJAMT)*.

Conference Publications:

[1] **Navaei, J.** and ElMaraghy, H., 2014. Grouping product variants based on alternate machines for each operation. In proceeding of the 47th CIRP Conference on Manufacturing Systems (CMS), Windsor, ON, Canada, Procedia CIRP, 17, 61-66.

[2] Zahedi Anaraki, A.; Jeihoonian, M.; Pazoki, M.; **Navaei, J.,** 2011. A Benders Decomposition Approach for a Production-Distribution Network Design Problem. In proceeding of 41th International Conference on Computers & Industrial Engineering (CIE 41), University of Southern California, Los Angeles, USA, 50-55.

[3] Pazoki, M.; **Navaei, J.;** Fahimi, K.; Hidaji, H., 2011. Designing a Semi-obnoxious Production-Distribution Network. In proceedings of the 16th Annual International

Conference on Industrial Engineering Theory, Applications and Practice (IJIE 16), Stuttgart, Germany.

[4] **Navaei, J.;** Mozdgir, A.; Hidaji, H., 2010. Two-stage assembly flow-shop scheduling problem with bi-objective of number of tardy and makespan minimization. In proceedings of the 2010 International Conference on Artificial Intelligence (ICAI ’10), Las Vegas, Nevada, USA, 704-709.