NEW APPROACHES FOR DESIGN OF HIGH-MIX LOW-VOLUME FACILITIES

DISSERTATION

Presented in Partial Fulfillment of the Requirements for

the Degree Doctor of Philosophy in the Graduate

School of The Ohio State University

By

Smart Khaewsukkho, M.S.

* * * * *

The Ohio State University
2008

Dissertation Committee:

Professor Shahrukh A. Irani, Advisor

Professor Clark A. Mount-Campbell

Professor Ponnuswamy Sadayappan

Approved by

____

Advisor
Industrial and Systems Engineering
Graduate Program

ABSTRACT

A jobshop is a manufacturing system well-known for its operational complexity due to the conditions under which it operates, such as instability of order volumes, product mix, product routings, customer bases, and etc. A fundamental approach to improve the performance and efficiency of jobshops has been to transform the Functional Layout that they traditionally use into a Cellular Layout. Functional Layouts have noticeable advantages such as flexibility and machine utilization; however, they also have major disadvantages such as high production lead times, high WIP (work-in-progress) inventory levels, and complex scheduling. In comparison, Cellular Layouts provide shorter production lead times, lower WIP inventory levels, simpler scheduling, and better control of product quality.

However, Cellular Layouts have major disadvantages such as low machine utilization, high cost of cell reconfiguration when demand or product mix change, and high risk of disruption to production due to machine breakdowns and operator absenteeism. These disadvantages are especially harmful for jobshops because they operate in a high-mix low-volume (HMLV) environment. Therefore, a complete reorganization of the Functional Layout of a jobshop into a Cellular Layouts, especially when operating in a HMLV environment, is never advisable. At the same time, retaining

the existing Functional Layout does not make a jobshop-type manufacturer competitive. These major shortcomings of these two extremes of facility layouts for jobshops encouraged as to investigate alternative layouts that retain the advantages of both layouts, and mitigate their disadvantages and the weaknesses.

In this dissertation, two novel layouts that integrate the attributes of the traditional Functional, Cellular and Flowline layouts are introduced. These layouts are a Modular Layout and a Hybrid Flowshop Layout. A modular layout derives its flexibility and efficiency by decomposing a complex material flow network into a network of layout modules that exhibit flow pattern characteristics of well-known standard types of layout. A hybrid flowshop layout derives its simplicity by transforming a complex material flow network into a flowline-like material flow network so that the simple factory logistics of the Toyota Production System could potentially be applied in the jobshop. This dissertation introduces the mathematical models, optimization methods and heuristics for design of these two novel layout configurations. Performance analyses were done to compare the traditional and proposed layouts. The role of the PFAST software, which was developed during the course of the dissertation, in a man-machine interactive process to design any facility layout is explained.

Dedicated to my parents

ACKNOWLEDGMENTS

I would like express my deepest sincere appreciation and gratitude to Dr. Shahrukh Irani, my advisor and mentor, excellent guidance and encouragement throughout this research work, and for his financial support for my graduate study at The Ohio State University. Without his support, both mentally and financially, this dissertation would not have been accomplished.

I would like to thank Dr. Clark A. Mount-Campbell and Dr. Ponnuswamy Sadayappan, for their helpful suggestions, comments and advice that helped correct scientific errors in this dissertation.

I would also like to thank my friends Sarawoot Watechagit, Attawit Ourairat, and Jin Zhou for their friendship, Partchapol Sartkulvanich for proofreading chapters in my dissertation, and Wichi Hanittina for being my big brother all these years at this university.

I wish to thank my only sister Namkhang for her support. I wish to thank my dearest beautiful wife Rattana for her true love, kindness and support throughout all these years of my doctoral program. I wish to thank my son Thatchapong (O_5), the most precious thing in my life, for lifting up my world with his smile.

Finally, I wish to thank my parents for their love, sacrifice and support. They do not know how to read a single word in this dissertation but they do know how to raise their son to become a person who can fulfill his dreams. This is for their endless love.

VITA

1972…………………………… Born, Samutprakarn, Thailand

1993…………………………… B.S. Mechanical Engineering, King Mongkut Instute of Technology Thonburi, Bangkok, Thailand

1999…………………………… M.S. Industrial and Systems Engineering, The Ohio State University

1999-present…………………... Graduate Research Assistant, The Ohio State Univeristy

FILED OF STUDY

Major Field: Industrial and Systems Engineering

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Surveys conducted by the Small Business Administration (SBA) show that about 99 percent of all manufacturing companies in America are small-and-medium manufacturing companies (SMEs) [Conner, 2001]. These companies are suppliers to larger companies and may have as few as 50 or more than 200 customers and do not make the same products every day. They are also known as job shops that are typically operating in a high-mix low-volume (HMLV) manufacturing environment. An HMLV manufacturing environment has complex operating characteristics, such as high variety in product mix, uncertainty in production demand, and many different manufacturing routings resulting in a complex material flow network. In order to survive and compete in the business these days, these SMEs have begun to adopt Lean Manufacturing approach since this approach has proven to vastly reduce the production cost and improve delivery performance [Achanga et al., 2005]. Lean manufacturing is a manufacturing philosophy that aims to shorten lead time from the point in time when an order is received to the point in time when the order is shipped by eliminating manufacturing wastes (non-value added activities) during the flow of the order through the factory [Womack and Jones, 1996]. There are Seven Forms of Waste that can be categorized as (1) Overproduction, (2) Underperformance due to non-standardized work, (3) Queue time, (4) Transportation (material handling) time, (5) Inventory (raw material, work-in-process inventory, and finished foods), (6) Unnecessary motion and travel by resources (tools, operations, etc.), and (7) Defective products. Among these wastes, queue time and transportation time are primary factors that increase production lead time in conventional jobshops, since a part may spend up to 95% of the total production time waiting or traveling as illustrated in Figure 1.1 [Crowson, 2005].

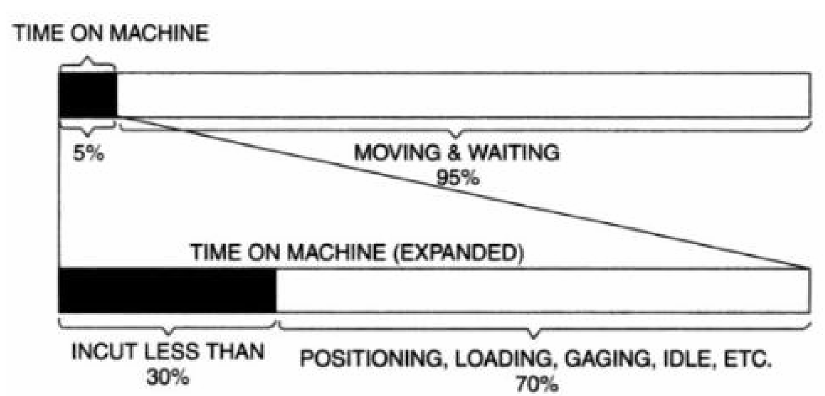


Figure 1.1: Production time for a part in a jobshop
[Source: Crowson, 2005, p. 170]

To shorten production lead time, these two dominant wastes must be minimized or eliminated so that the velocity of material flows can increase, inventories can be reduced, and operating costs can be reduced. A well-designed facility layout that is flow-efficient can help HMLV manufacturers to eliminate these major wastes and shorten production lead times.

Facility layout design determines how to arrange, locate and distribute the equipment and support services in a manufacturing facility to serve physical flow relationships between manufacturing activities. Since flows of materials, personnel, tools and related resources in a manufacturing facility are physical moves, without physical changes in the layout, the production flow in a facility can hardly improve. Lean practitioners and manufacturers have long understood that the facility layout is a fundamental reason for the complexity of flow in a facility. For decades, to facilitate the flow, converting an existing functional layout to a cellular layout has been promoted. Cellular layouts have made the production flows in any facilities more efficient [Wemmerlov and Hyer, 1989]. Once the production flow is "fixed", Lean Manufacturing can be adopted effectively.

However, in today's market, customers are demanding more customized, better quality products at low cost. With demand characteristics changing and small-niche market glowing, product life cycles are rapidly shortening. Manufacturing companies are forced to offer more wide variety of products with arrays of options and features produced in small batches or even one-of-a-kind. This leads to higher complexity in the production flows in the facilities of these manufacturing companies. The cellular layout is no longer an efficient solution for these companies because manufacturing cells are designed to produce specific sets of products (aka product families). In addition, manufacturing cells demand sufficient production volumes in order to operate efficiently. In volatile environments where product mix changes and demand volumes are not stable, the cellular layout is not flexible enough to maintain its efficiency and it is often found to have inferior performance compared to a functional layout in this situation. Therefore, the manufacturing companies that are operating in an HMLV environment these days need to evaluate new approaches for designing their layouts to facilitate the flows and allow them to exploit the benefits of implementing Lean Manufacturing, even in high-mix low-volume (HMLV) conditions.

In summary, Lean Manufacturing is the most productive approach that in manufacturing industries and manufacturing companies around the world have adopted to improve performance of their facilities. A core approach for Lean Manufacturing is to facilitate efficient production flow in a facility by removing wastes that delay production flow. The layout of a facility has a profound and direct effect in the production flow in a facility; without designing a suitable layout to facilitate the flow, adopting Lean Manufacturing may not achieve maximum benefits.

1.2 Objectives of This Research

The primary objectives in our research are two-fold. The first objective is to introduce new approaches to design two novel layouts that are suitable for manufacturing companies operating in an HMLV environment. These two layouts are a Modular Layout and a Hybrid Flowshop Layout. These hybrid layouts integrate the attributes of the traditional Functional, Cellular and Flowline layouts.

A modular layout derives its flexibility and efficiency by decomposing a complex material flow network into a network of layout modules that exhibit flow pattern characteristics of well-known standard types of layout. These layout modules are similar to small (fractional) manufacturing cells but they are not dedicated to specific product families and machines are allowed to be shared across multiple product families. Since the machines in any module are shared by more than one product family, that gives a modular layout the flexibility that a Functional Layout also does provide. And since the layout modules are pseudo-cells, a modular layout also inherits some of the benefits of having a Cellular Layout such as short travel distances and improved control of product quality. Therefore, this new layout gains several advantages by combining multiple structures of traditional layouts in one layout.

Since a modular layout is designed to integrate multiple layout characteristics contained in a single facility, theoretically it should perform efficiently in a complex, volatile HMLV environment. However, in practice operational difficulties could arise with this layout since it involves operating several sub-factories simultaneously in one facility. Without having an efficient material control strategy, chaos could result. This downside of the modular layout has led us to develop another layout which is still a modular layout but exhibits more simplified flow. This layout is called a Hybrid Flowshop Layout, one of the main contributions of our research to both the archival literature and practice of facility layout. This layout retains the advantages of having a modular structure and being simply it transforms a complex network of material flows into a flowline-like material flow network.

A hybrid flowshop layout allocates machines into several groups (called stages) similar to layout modules in a modular layout; however, these groups are arranged in a line. The majority of production flows in this layout are forward (either in-sequence or bypass) while backtracking flows are minimized or eliminated if possible. This layout imitates a flowline layout and allows the production control of flowline production systems, such as Toyota Production System (TPS) to be effectively used.

Along with this layout, a new material control system called the Next Stage Awareness System (NSAS) has been developed and our experiments show that when using this system, a hybrid flowshop layout outperforms other layouts in most cases. The hybrid flowshop layout with NSAS can be used by HMLV manufacturing companies to design flowline-like production facilities to simplify their operational controls and increase their production efficiencies.

The second objective in our research is to provide a systematic approach for selecting and designing appropriate layouts for HMLV manufacturing companies. The design and selection of an appropriate layout for a HMLV facility poses a major challenge since a layout can serve different purposes and provides different advantages and disadvantages based upon different criteria being considered. There is no absolute method or solution for layout design that suits different flow networks, product variety, production volume, and demand stability. Therefore, we introduce a range of conceptual layout configurations and provide a flowchart for selecting and designing both traditional and non-traditional layouts for HMLV facilities using the PFAST software that has been developed at The Ohio State University.

1.3 Organizations of This Research

The remaining of this dissertation is organized as follows. Chapter 2 provides an introduction and background on traditional facility layouts used in HMLV manufacturing environments. The modular layout and hybrid flowshop layout are also introduced briefly.

Chapter 3 introduces the details of a modular layout and layout modules. Flow pattern characteristics of each module type are introduced. A mathematical model for the problem of designing modular layouts is developed and solved. The original problem is then divided into two phases with some simplifications and assumptions made to reduce its complexity.

Chapter 4 presents a cut-tree based heuristic approach proposed and developed for solving the real-world instances of the modular layout problem.

Chapter 5 introduces the details of a hybrid flowshop layout and its advantages. A mathematical model for the problem of designing hybrid flowshop layouts is developed and solved. The original problem is simplified similar to what was done for the modular layout problem. A ratio cut partitioning heuristic approach is proposed for solving large real-world instances of this layout design problem.

Chapter 6 presents the current man-machine approach for facilities design in a HMLV environment using Production Flow Analysis (PFA). The PFAST supports this current man-machine approach to design facility layouts.

Chapter 7 presents an experimental evaluation and performance comparison between the traditional layouts and our proposed layouts using simulation. Our simulation model is based on the experimental study using simulation approach to investigate the performances of different manufacturing layouts [Suresh and Slomp, 2005]. In our experiments, the performance of functional layouts, cellular layouts, modular layouts and hybrid flowshop layouts are evaluated using the Arena™ discrete simulation software.

Chapter 8 contains conclusions, contributions and recommendations for future research based on this study.

Chapter 9 introduces a new concept for a material control system called the next stage awareness system developed for a hybrid flowshop layout to promote pull scheduling in high-mix low-volume facilities.

References

1. Achanga, P., Shehab, E., Roy, R. and Nelder, G., "Critical success factors for lean implementation within SMEs," *Journal of Manufacturing Technology Management*, Vol. 17, No. 4, pp.460-471 (2006)

2. Conner, G., *Lean Manufacturing for the Small Shop*, Society of Manufacturing Engineers, Dearborn, Michigan (2001)

3. Crowson, R., *Product Design and Factory Development (Handbook of Manufacturing Engineering, Second Edition)*, CRC Press, Boca Raton, Florida (2005).

4. Miltenburg, J., *Manufacturing Strategy*, Productivity Press, Portland, Oregon (1995).

5. Suresh, N.C., Slomp, J., "Performance of virtual cellular manufacturing with functional and cellular layouts in DRC settings," *International Journal of Production Research*, Vol. 43, No. 5, pp. 945-979 (2005).

6. Wemmerlov, U. and Hyer, N.L., "Cellular manufacturing in the US industry: a survey of users," International Journal of Production Research, Vol. 27, pp. 1530-1551 (1989)

7. Womack, J.P. and Jones, D.T., *Lean Thinking*, Simom & Schuster, New York, NY (1996).

CHAPTER 2

BACKGROUND

2.1 Traditional Layouts

Manufacturing companies nowadays are facing increased demand from their customers for prompt response and product customization. This leads to increased complexity and diversification of the production processes and manufacturing systems in the companies. As the complexity and diversity of the product mix increases, companies migrate towards HMLV operating environment. In order to be competitive, the manufacturing companies have to maintain manufacturing flexibility but keep their processes and management practices simple despite the complications of diversity.

The layouts of manufacturing companies have profound effects on the complexity of their production processes and flexibility of their manufacturing systems. As a result, the problem of designing layouts for manufacturing facilities is one of the most recognized and critical problems in Industrial Engineering with considerable research done in this area. [Meller and Gau, 1996] A facility layout determines how to arrange, locate, and distribute machines, equipment, and support services in a large area. An effective facility layout helps to maximize product throughputs. Traditionally, there are three types of facility layouts that have been used for decades to layout discrete manufacturing systems. They are Functional, Cellular, and Flowline layouts. These layouts allocate machines, equipment, and support services in a manufacturing facility based on different criteria. Therefore, each of these has advantages and disadvantages with no single layout always being superior to the other two.

Functional Layout (Figure 2.1) is a layout in which machines, equipment, and support services are arranged based on their functional capabilities. It is also called a Process Layout because machines, equipment, and support services with similar process and operational functions are grouped into one area (or department). These departments are located in such a way that the total material handling costs for material flows between all departments the facility are minimized.

Cellular Layout (Figure 2.2) is a layout that has the arrangement of machines, equipment, and tooling is based of product families. This layout is referred to as Product Layout. In contrast to a functional layout, machines, equipment, and support services in a cellular layout are segmented into different groups, called manufacturing cells. Each manufacturing cell is responsible for producing a group of similar products, called a product family. Each manufacturing cell can generally process different operation sequences required to manufacture the products in the product family. Because each manufacturing cell focuses on similar products and the cell has a small footprint, material transfers within the cell can be done using one-piece flow. The benefits of a cellular layout are excellent quality control and low material handling costs.

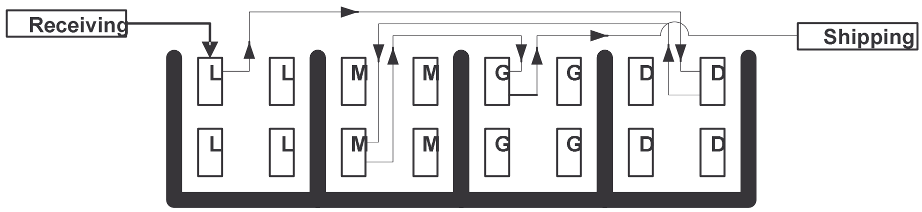


Figure 2.1: Functional Layout

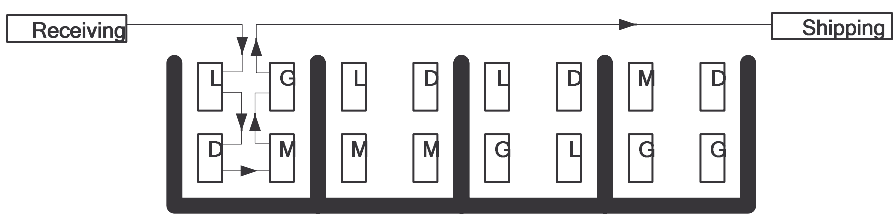


Figure 2.2: Cellular Layout

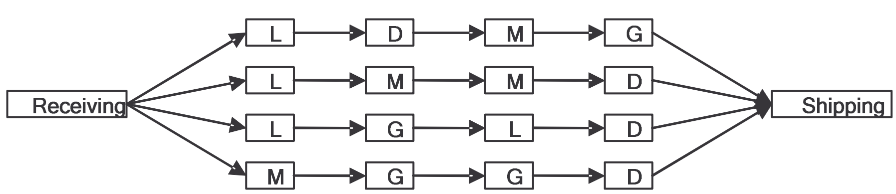


Figure 2.3: Flowline Layout

Flowline Layout (Figure 2.3) is similar to a cellular layout and is essentially a product-oriented layout. In this layout, machines, equipment, and services are once again allocated according to the needs of one product or a small product family. However, they are arranged in a linear layout according to the operation sequences of the product or the product family. The highest production efficiencies and the product quality can be achieved using this type of layout. Nevertheless, this type of layout can be justify mostly for a high volume of production of a low-variety product mix in order to justify the use of expensive dedicated machines and equipment.

2.2 HMLV Facilities

The three traditional layouts described earlier perform differently on performance criteria, such as costs, quality, output, delivery, flexibility, and adaptability. Each of these layouts is suited best for only a certain manufacturing environment, e.g. either high-mix low-volume or low-mix high-volume manufacturing environment. Figure 2.4 shows how the traditional layouts are compared by their different performance criteria such as product variety, production volumes, etc. The manufacturing outputs as shown the figure, which can be categorized into delivery, cost, quality, performance, flexibility, and innovativeness, are influenced significantly by the choice of employing different layouts. In this research, our focus was on the first three outputs (delivery, cost, and quality), which are often lacking but considered the most important outputs for HMLV manufacturing companies. As seen in this figure, flowline layouts (which are line-flow operator paced, line-flow equipment paced, and continuous flow) are the most preferable because they are proven to achieve all three key outputs. However these layouts are suited only for manufacturing companies operating in low-mix high-volume environments. For HMLV manufacturing companies, flowline layouts are impractical. Thus, achieving the three key outputs for HMLV manufacturing companies seems to be a difficult and challenging goal.

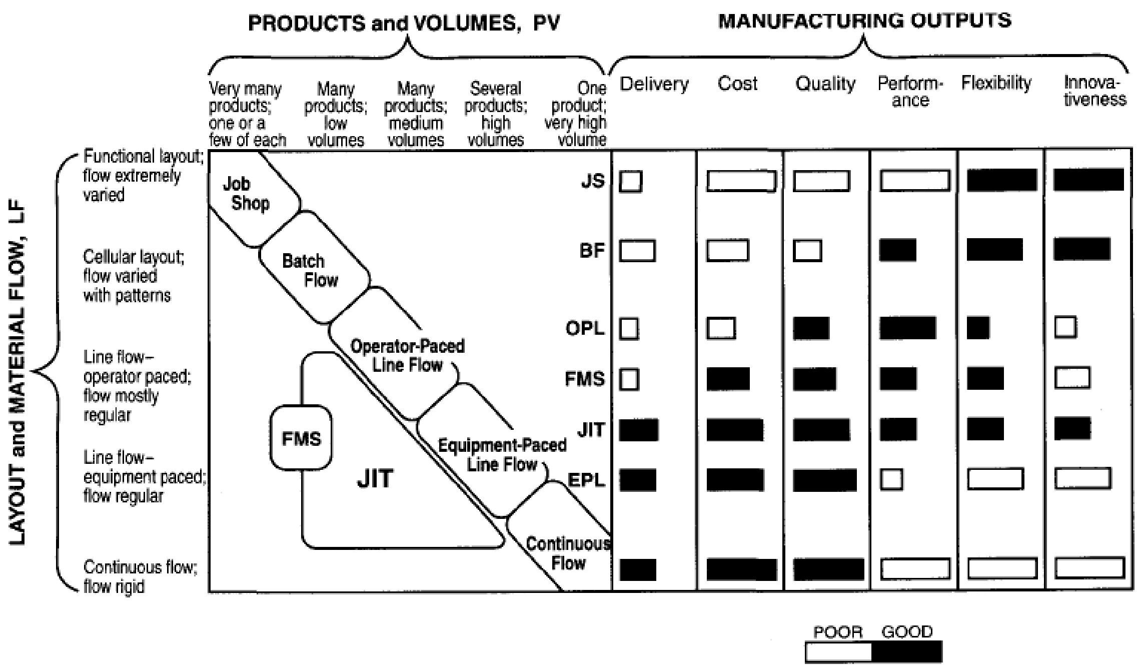


Figure 2.4: The product/volume-layout/flow matrix and its relationship to manufacturing outputs [source: Miltonburg, 1995, p. 41]

In Figure 2.4, a jobshop (JS) production system is positioned at the left top corner of the chart. The attributes of this production system are wide variety products, small lot (or batch) production, and demand volume ranging from one to a few. Products in a jobshop production system mostly require a unique set-up and sequence of processing steps. Examples of a jobshop manufacturing companies (so-called jobshops) include machine shops, forge shops, paint shops, and other facilities that make products that require customization in small lot sizes. Because of the complexity and the need of flexibility, functional layouts are the most preferred layouts by jobshops because flexibility and highly skilled operators are needed in complex HMLV manufacturing facilities. Since the jobshop production system and functional layouts score poorly on the three key outputs (cost, quality, and delivery), in order to improve their performance, jobshops has been attempting to adopt lean manufacturing to help improving their performances and reduce their production costs.

2.3 Lean Manufacturing

Lean manufacturing was probably originated in 1920s when Henry Ford and his associates recognized the potential of flow improvement in Ford automotive production. Henry implemented synchronized assembly line production to promote continuous flow from raw materials to shipments of finished car for their Model T Ford automobile. His method has shown significant productivity leap but has been only in the *special case* of large-lot high-volume high-speed assembly production. Three quarters of a century later, Ohno, a chief engineer at Toyota Motor Company, conducted a thorough study of large-lot American production systems in order to improve his company's production system to compete and survive in the global market competition. [Ohno, 1988] It is understood that Japanese corporations cannot afford large amounts of land to warehouse finished products and parts. Adopting large-lot production to achieve economic lot size would cost even more. Therefore, he developed an alternative approach to promote continuous flow for small-lot production in Toyota plants and soon his developed production system has become one of the most productive systems in the global manufacturing industry. This production system is referred to as Toyota Production System (TPS). It is a special line flow production system that can produce more variety of products in lower volumes or smaller lot-sizes than the traditional large-lot American production system. [Miltenburg, 1995]

The small-lot production allows TPS to operate under low inventories. When the inventories decrease, problems or wastes are exposed. Once these problems or wastes are found, they are removed and the production system improves. This is referred to as a waste identification and removal strategy in TPS. The system continues to reduce the inventories and the next problems or wastes are then identified and removed. The repeating process of identifying and eliminating wastes is referred to as a continuous improvement strategy in TPS. Therefore, this manufacturing philosophy aims directly to attack any form of waste in the production process through continuous improvement in pursuit of perfection.

TPS best practice has brought lean approach becoming the most productive approach in the manufacturing industry. It has become the *general case* of lean manufacturing these days. The values of the system and its strategies have proven to global manufacturing industry and a considerable number of manufacturing companies are keen to adopt this model to their own production systems. [Papadopoula and Ozbayrak, 2005]

2.4 Lean Manufacturing in HMLV Facilities

Although lean manufacturing can be a universal tool for all manufacturing companies, lean strategies and techniques in TPS, may not fit to all types of manufacturing companies. According to a lean practitioner who has 20 years of lean manufacturing experience in both low-mix high-volume and high-mix low-volume environments, Lean Manufacturing and TPS are terms that can be referred interchangeably but they may *not* be interchangeable. [Nelson] TPS is suited for the manufacturing companies in a low-mix high-volume environment where flow patterns are simplified and recognized and flowline layouts are most suitable for these companies. In a HMLV environment, most manufacturing companies are jobshops. Their products are wide variety and their production volumes are very small compared to the manufacturing environment in which TPS was developed. Functional layouts are therefore most implemented in these companies to handle the complexity of the production flows. While their production flows are very complex and their layouts are not supporting and smoothing the production flow, adopting TPS to these manufacturing companies without properly adapting is akin to putting smaller-size shoes without adjusting.

The adaptation of the layout when implementing lean manufacturing in most manufacturing companies has traditionally been done by converting from the existing functional layouts to cellular layouts. However, researchers and practitioners have found that it is ill-advised to completely convert the functional layout of a jobshop operating in a HMLV environment into a cellular layout. The main reason is that the inherent inflexibility of manufacturing cells cannot adapt to changing capacity requirements (machine and labor), product mix, and demand volume. Therefore, the design of facility layouts for Lean implementation in the jobshops operating in this complex environment needs a new research.

In this research, we have developed two new layouts for the manufacturing companies operating in a HMLV environment. The two layouts are a Modular Layout and a Hybrid Flowshop Layout that will be briefly introduced next.

2.5 Modular Layout

In a cellular layout, the majority of inter-machine flows are intra-cell with some inter-cell flows between different manufacturing cells. The ideal case for cellular manufacturing is to have completely disjointed cells where there is no flow among the cells. That ideal rarely happens in HMLV facilities because product mix and order quantities change, and machines in the cells are always shared. Even though machine duplication could make cells disjoint, the duplication cost is always a huge factor for the small and medium-sized HMLV companies. Modular Layout is similar in concept to a cellular layout. It is layout containing small-size manufacturing cells; however, products are not required to be manufactured in only one cell and machines are allowed to be shared across multiple product families. This small-size manufacturing cell is called a layout module. In a modular layout, machines are more flexible than they are in a cellular layout. At the same time, flow distances are shorter than in a functional layout resulting in less material handling cost. Therefore, the objective for designing modular layouts is to design several machine groups that absorb the highest flows within different groups and have low volumes of flows between the different groups. The graphical representation of a modular layout and other traditional layouts are shown in Figure 2.5.

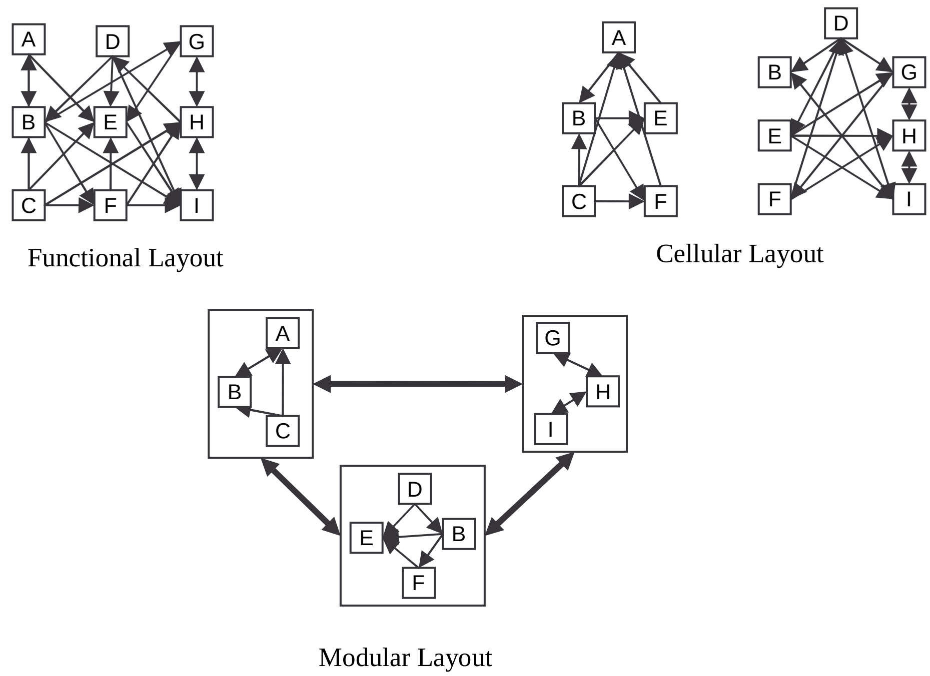
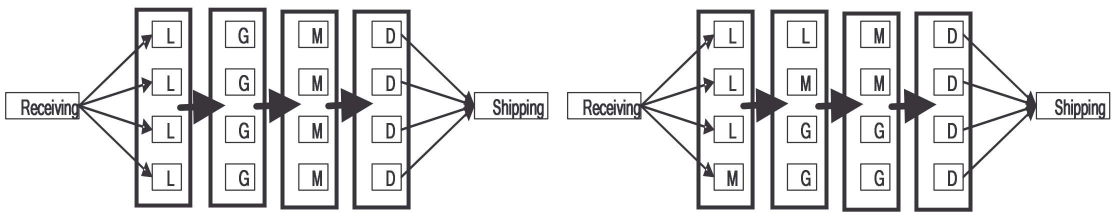


Figure 2.5: Modular layout, Functional layout and Cellular Layout

2.6 Hybrid Flowshop Layout

In a general flowshop layout, machines are allocated into several groups (called stages) and these groups are arranged in a flowline. Generally, product routings in this layout are identical and machines within each stage are identical or have similar process capabilities. Products can flow in forward direction without backtrackings in this layout. In a *hybrid* flowshop layout, product routings are not similar and machines are not necessary identical; however, the products can still flow in one direction from beginning stages to finishing stages without backtracking flows (or less backtracking flows), as shown in Figure 2.6.



(a) Pure Flowshop (b) Hybrid Flowshop

Figure 2.6: Pure flowshop and hybrid flowshop

Each stage in a hybrid flowshop layout is a group of machines similar to a manufacturing cell or a layout module. However, unlike a manufacturing cell or a layout module described earlier, it does not contain a group of machines that produces a product family as a manufacturing cell. It does not contain high traffic machines as a layout module either. Each stage in a hybrid flowshop layout contains machines that can perform one or more consecutive operations occurring in the operation sequences of a large number of parts. Importantly, when aligning these groups of machines (stages) in a line, it must assure that most parts do not need to travel back to preceding stages to perform their remaining operations. The remaining operations will always be performed by machines within the current stages or the next stages. The graphical representation of a hybrid flowshop layout and other traditional layouts are shown in Figure 2.7.

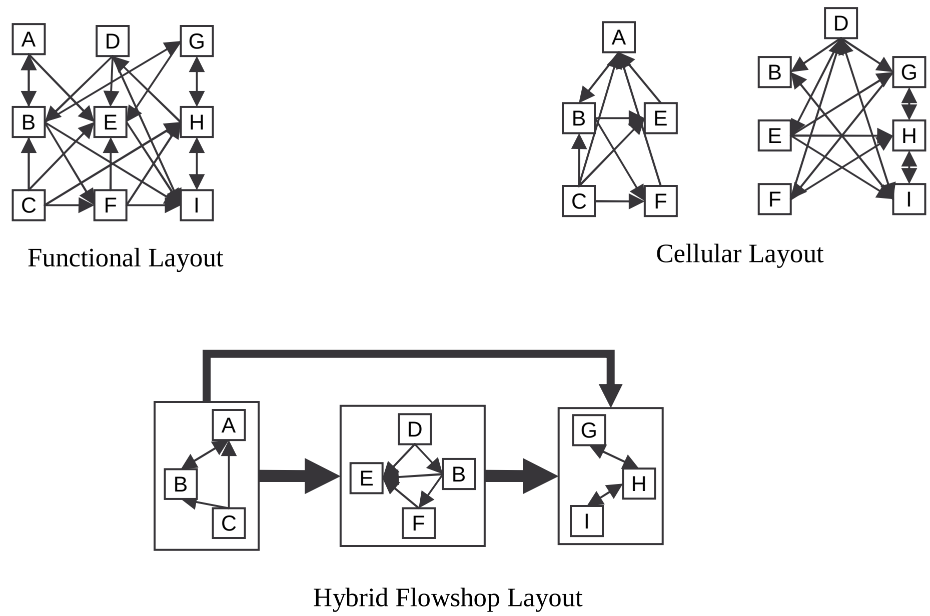


Figure 2.7: Hybrid Flowshop layout, Functional layout and Cellular layout

In the next chapters, the mathematical models for the problems of designing a modular layout and a hybrid flowshop layout are presented. Optimization and heuristic approaches for solving these layout problems are developed and exercised with an example data set from literature.

References

1. Meller, R.D. and Gau, K., "The facility layout problem: recent and emerging trends and perspectives," *Journal of Manufacturing Systems*, Vol. 15, No. 5, pp. 351-366 (1996).

2. Miltenburg, J., *Manufacturing Strategy*, Productivity Press, Portland, Oreron (1995).

3. Nelson, K., *Lean Manufacturing: Adapting as Important as Adopting*, www.iSixSigma.com

4. Ohno, T., *The Toyota Production System: Beyond Large-Scale Production*, Productivity Press, Portland (1988).

5. Papadopoula, T.C. and Ozbayrak, M., "Leanness: experience from the journey to date," *Journal of Manufacturing Technology Management*, Vol. 16, No. 7, pp. 784-807 (2005).

CHAPTER 3

DESIGN OF MODULAR LAYOUTS: A REVISED AND SIMPLIFIED METHOD

3.1 Introduction

Considerable research has been done in designing a flexible layout to hedge against future changes in product mix, product routings and demand volumes in these days manufacturing environment. A modular layout is one of the flexible layouts developed to combine the attributes and advantages derived by different traditional layouts [Irani and Haung, 2000]. In a modular layout, the material flow network in a manufacturing facility is decomposed into a network of segments called Layout Modules. A layout module is essentially a group of machines connected by a material flow network that exhibits a flow pattern characteristic of a specific type of layout, such as the Flowline, Cellular or Functional Layout.

A layout module is similar to a manufacturing cell or it can be considered as a fractional manufacturing cell. The differences are that products may not necessarily be completely manufactured in one module and machines are allowed to be shared by multiple product families. The objective of designing a modular layout is to select the best set of machines for each layout module and assign each product to the best combination of layout module(s), such that the total production cost incurred is minimized and all resource constraints are satisfied. Intuitively, since this new layout is a hybrid combination of traditional layouts, it can shorten throughput time, lower WIP inventory levels, and incur less material handling cost while retaining flexibility and high machine utilization.

Among all flexible layouts, a modular layout is different from others because it combines different types of traditional layouts and deploys them together in the same facility [Balakrishnan and Cheng, 2007]. Conceptually, it should perform better than other layouts because of all the advantages that can be expected from selecting, combining and deploying several types of layouts that are suited best to different areas in the same facility. The advantages of this layout are very promising but the application of this layout in the industry is limited. It can be the reasons that the layout solution is not validated and also the problem of designing a modular layout is very complex. The modular layout problem has proven to fall into the class of NP-complete problem. In addition, this designing problem required the great detail of data input which is very difficult in practical for manufacturing companies to acquire. These two issues can be the most obstacles for adopting this layout to the industry. Therefore, in this research, a recent mathematical model for designing a modular layout is revised and the layout solution is validated. Then the current mathematical model is simplified to be suited to practical conditions where the availability and accuracy of data input is difficult to acquire, especially in the manufacturing facilities that are operating in high-mix low-volume environment where a modular layout was intentionally designed to suit for.

This chapter is organized as follows. Literature review on flexible layouts is presented. The prior research and mathematical model for the problem of designing a modular layout are described. The problem model of designing a modular layout is revised. An optimization technique using CPLEX, a commercial IP solver, is used to demonstrate and validate a solution approach for solving a modular layout problem optimally. Then a simplified version of the current mathematical model with a two-stage solution approach is introduced. The performance comparison between the original approach and the simplified approach is discussed and concluded.

3.2 Literature Review

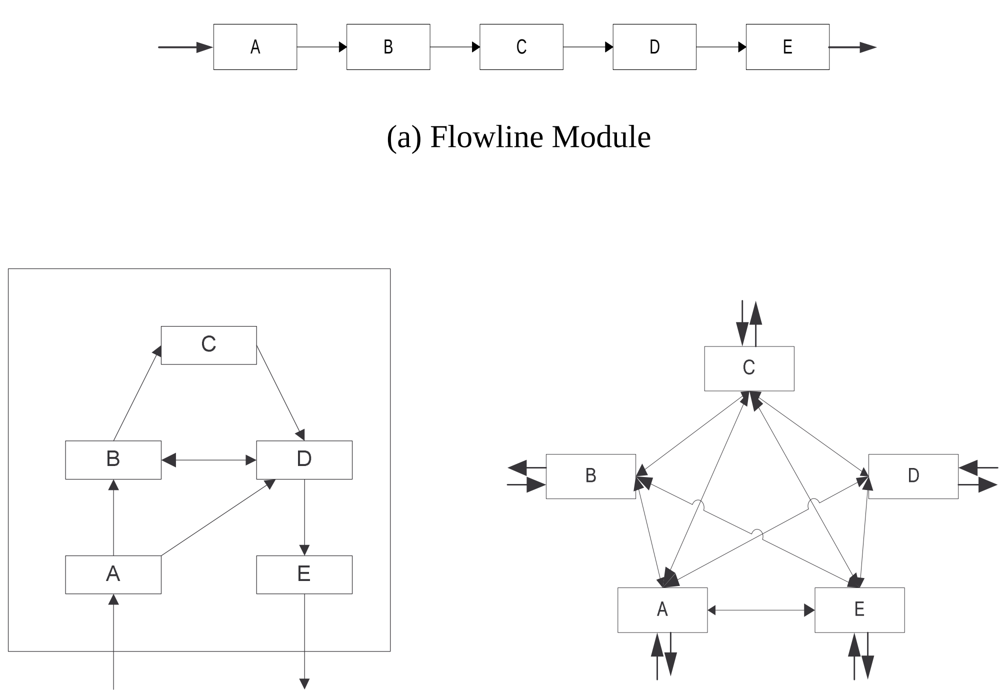
The design of flexible facility layouts has been recognized in recent research tremendously. Examples of these layouts are fractal layouts, virtual cellular layouts, distributed layouts, and modular layouts. In typical, flexible layouts are designed to hedge against the variability and variety of products and demand in manufacturing facilities. *Fractal Layouts* [Askin, Ciarallo and Lundgren, 1999; Venkatadri, Rardin, and Montreuil, 1997] divide the facility into a material flow network of fractals. Fractals are considered as small factories within a factory. It can be considered as a small manufacturing cell; however, these fractals are capable of producing a wide variety of products than a manufacturing cell. Because each fractal is more like a complete factory than a complete cell, they have the material handling, scheduling and teamwork advantages of manufacturing cells, but are more flexible to demand and product (mix) changes. *Virtual Cellular Layouts* [Irani et al, 1993; Babu et al, 2000; Prince and Kay, 2003] implement product family-oriented scheduling in a functional layout facility. Unlike typical manufacturing cellular layout, the physical rearrangement of machines in a facility to form manufacturing cells is not necessary. Rather, the manufacturing cells are formed logically and products within the same families are routed through appropriate logical cells. While this makes scheduling complex, the flexibility and ability to adapt to the changes of product mix and demand for this layout is exceptional because there is never the need to dissolve and relocate the machines in the cells when the product families change. *Distributed Layouts* [Lahmar and Benjaafar, 2005] disaggregate departments in a functional layout into subdepartments and statically distribute them throughout a facility. The degrees of distribution range from a non-distributed layout, which is a functional layout, to a completely distributed layout, where the departments are down to single machines and placed individually in the facility. The completely distributed layout is similar to the concept of *Holonic Layouts* [Montreuil et al, 1993] where individual machines are placed throughout a facility with an objective to provide efficient process routes for any products with a minimum delay. Both distributed and holonic layouts aim to reduce the long travel distances in functional layouts and the inflexibility of cellular layouts. *Modular Layouts* [Irani and Huang, 2000 and 2005] decompose the overall material flow network for a facility into a network of one or more different layout modules. Each module has a unique layout, material flow pattern and scheduling characteristics, such as single machining center, flowline, branched flowline, cell, flowshop, and jobshop. Thereby, different areas of the facility have a layout that best suits the material flow network for the machines and products that comprise that area. Layout modules can be categorized generally as follows:

- *Flowline Module* (Figure 3.1a): A Flowline module is a linear arrangement of machines such that all inter-machine moves for consecutive pairs of operations on any product moving through the line would be forward, either in-sequence or bypass. In case of backtracks, due to multiple non-consecutive operations on the same machine, a decision could be taken to (a) modify the linear shape into linear segments with circular/loop segments separating pairs of consecutive linear segments, (b) retain the linear shape but utilize a bi-directional material handling system for backtrack moves, or (c) duplicate the same machine at multiple locations to convert backtrack moves into forward moves.

- *Cell Module* (Figure 3.1b): Similar to a Flowline module, a Cell module is a set of dissimilar machines which, if placed together, could produce a family of parts or products without the products requiring to visiting any additional departments or machines external to the module. Although the parts in a family may not use all the machines and/or have the same sequence of operations, their operation sequences have high commonality of machine requirements and high similarity of operation sequences.

- *Functional Module* (Figure 3.1c): A Functional module is analogous to the process-focussed department in a traditional Functional layout in which material flows are random. The random flows are due to the absence of any flow dominance or patterns in the sequences in which the different machines within the module are used by different parts.

In this research, the concept of modular layouts has been studied further. A revised mathematical model for this problem is presented in the next section.



(a) Flowline Module

(b) Cell Module

(c) Functional Module

Figure 3.1: Three types of layout modules

3.3 Problem Formulation and Recent Mathematical Model

The problem of designing a modular layout is a multi-criteria decision problem that can be described as follows:

Given

- Production volume of each product in a single planning period
- Operation sequences for each product (i.e. machine routings) and processing time for each operation
- The production quantity of each product type in the planning period
- The transfer batch size of each product type
- The material handling cost per transfer batch of each product type traveling between each pair of machine types in the planning period
- There is a fixed rate penalty for products traveling between layout modules
- The rates of the setup cost and processing cost of each product type at each machine type in the planning period
- The loading/unloading cost of each product type at each machine is negligible and can be ignorable
- The amortized purchase price per unit of each machine type in the planning period
- Number of machine type available in the planning
- Penalty factor for product traveling between modules

Determine

- Number of layout modules, denoted by *m*
- To which module in the existing layout each machine is allocated
- To which machine each operation for each product is assigned

Minimize

- Material handling cost
- Machine purchasing cost for additional machine that may be required
- Production cost

  Such that

  - Each operation is assigned to one and only one machine
  - Machine capacity constraints are satisfied

The following model is the revised version of the original non-linear mixed integer programming model for the problem of designing a modular layout [Irani and Huang in 2005]:

$$\textbf{Minimize} \ \sum_{k=1}^{K} E_k \cdot \left(\sum_{m=1}^{M} r_{km} - L_k \right) \qquad (1)$$

$$+ \left(\sum_{i=1}^{T} \sum_{j=1}^{N_i} \sum_{k=1}^{K} \sum_{m=1}^{M} \left\lceil \frac{Q_i}{B_i} \right\rceil \cdot S_{ijk} \cdot x_{ijkm} \right.$$

$$\left. + \sum_{i=1}^{T} \sum_{j=1}^{N_i} \sum_{k=1}^{K} \sum_{m=1}^{M} Q_i \cdot U_{ijk} \cdot P_{ijk} \cdot x_{ijkm} \right)$$

$$+\left(\sum_{i=1}^{T}\sum_{j=1}^{N_i-1}\sum_{k=1}^{K}\sum_{\substack{l=1\\l\neq k}}^{K}\sum_{m=1}^{M}HUC_{ijk}\cdot x_{ijkm}\cdot x_{i(j+1)lm}\right.$$

$$\left.+\mu\cdot\sum_{i=1}^{T}\sum_{j=1}^{N_i-1}\sum_{k=1}^{K}\sum_{\substack{l=1\\l\neq k}}^{K}\sum_{m=1}^{M}\sum_{\substack{n=1\\n\neq m}}^{M}HBC_{ikl}\cdot\left\lceil\frac{Q_i}{B_i}\right\rceil\cdot x_{ijkm}\cdot x_{i(j+1)ln}\right)$$

subject to:

$$\sum_{k=1}^{K}\sum_{m=1}^{M}x_{ijkm}=1, \text{ for each } (i, j) \tag{2}$$

$$\frac{\sum_{i=1}^{T}\sum_{j=1}^{N_i}Q_i\cdot P_{ijk}\cdot x_{ijkm}}{F_k}\leq r_{km}, \text{ for each } (k, m) \tag{3}$$

$$\sum_{k=1}^{K}r_{km}\geq 1, \text{ for each } k \tag{4}$$

$$\sum_{m=1}^{M}r_{km}\leq MaxL_k, \text{ for each } k \tag{5}$$

$$r_{km}\leq BN\,.\,w_{km}, \text{ for each } (k, m) \tag{6}$$

$$\sum_{k=1}^{K}w_{km}\leq SIZE, \text{ for each } m \tag{7}$$

$$r_{km}\geq 0 \text{ and integral, for each } (k, m) \tag{8}$$

$$x_{ijkm} \text{ binary, for each } (i, j, k, m) \tag{9}$$

$$w_{km} \text{ binary, for each } (k, m) \tag{10}$$

$\lceil z \rceil$ denotes the smallest integer that is larger than or equal to z

Variables:

r_{km} = Number of units of machine type k assigned to module m.

x_{ijkm} = 1 if the j^{th} operation of product type i is processed at machine type k in module m;

= 0 otherwise.

w_{km} = 1 if machine type k is assigned to module m;

= 0 otherwise.

Parameters:

μ = Penalty factor for products traveling between modules

B_i = Transfer batch size of product type i

BN = Any big number

E_k = Annualized cost of purchasing a unit of machine type k

F_k = Annual production time available per unit of machine type k

HBC_{ikl} = Material handling cost for transferring a *batch* of product type i from machine type k to machine type l

HUC_{ikl} = Material handling cost for transferring a single *unit* of product type i from machine type k to machine type l

K = Number of machine types

L_k = Existing number of units of machine type k

$MaxL_k$ = Maximum number of units of machine type k allowed

M = Maximum number of modules in the layout

N_i = Number of operations in the routing of product type i

P_{ijk} = Processing time of the j^{th} operation of product type i at machine type k

S_{ijk} = Setup cost for the j^{th} operation of product type i at machine type k

U_{ijk} = Cost per unit time of processing the j^{th} operation of product type i at machine type k

Q_i = Annual production quantity of product type i

$SIZE$ = Maximum number of machine types in one module

T = Number of product types

The objective function (1) minimizes the sum of prorated machine purchasing cost, production cost and material handling cost. The machine purchasing cost in the first part of the objective function (1) is the amortized cost of purchasing extra machines in the planning period. The second part of (1) calculates the production cost that consists of setup cost and processing cost. The processing cost is the cost of producing all units of products at their designated machines. The setup cost is the cost of setups required by batches of products at their designated machines. The third part of (1) calculates the material handling cost that consists of intra-module and inter-module material handling costs. The intra-module material handling cost is the cost of moving a unit of product from one machine to another machine inside a module. HUC_{ikl} is the cost of moving a unit of product i from machine k to machine l where these two machines are in the same module. The inter-module material handling cost is the cost of moving products normally in batch from one module to another module. HBC_{ikl} is the cost of moving a *batch* of product i from machine k to machine l when these two machines are located in different modules.

Constraint set (2) ensures that each operation in the routing of each product is performed at one and only one machine. Constraint set (3) guarantees enough capacity for operations performed on each machine type in each layout module. This constraint set implies that the number of machines of the same type allocated to one or more modules is constrained by the total number of machines of that type currently available on the shop floor i.e. acquisition of extra machines incurs a capital expense. Constraint set (4) ensures that at least one machine of the same type is assigned to a layout module. Constraint set (5) ensures that the number of machines for each type is not larger than it is allowed. Constraint sets (6) and (7) ensure that the number of machine types in a module will not exceed its maximum size. The size of module is defined by the number of different machine types in the module instead of the number of the copies of machines. This is because there will be no move of parts between machines of the same type. Therefore, the multiple copies of the same machine type in a module are grouped and considered as one unit machine in a modular layout. Constraint sets (8), (9) and (10) ensure nonnegativity, binary and integer requirements for the decision variables.

There are three types of cost embedded in the objective function. The material handling cost is the only cost affected directly by the different arrangements of machines in various layout solutions. The production cost, consisting of the setup cost and processing cost, is affected mostly by the performance of operations on the machines, not by the different arrangements of machines. Also, the cost of purchasing extra machines could be affected mostly by the limited capacity of machines. In practice, the capacity problem can be managed by reducing setup times, reducing operation times, adding over-time, or having alternate routes to move operations away from overloaded machine types. Therefore, the material handling cost is core to the problem, especially the inter-module material handling cost. Not only do the inter-module trips result in higher material handling costs, they also imply a batching process that leads to queuing delays, non-uniform machines loads and high work-in-process inventory levels. This is the reason for having parameter μ in the formulation to allow the inter-module material handling cost to reflect the hidden costs of queuing delays and WIP inventory levels in the model.

3.4 Problem Complexity

In this section, the proof of problem complexity is given for the IP model of a modular layout problem that falls into the class of NP-complete problem. The 3-Portion Problem which is known to be an NP-complete problem is used in this proof as follows:

"Set A of $3m$ elements, a bound $B \in Z^+$, and a size $s(a) \in Z^+$ for each $a \in A$ such that $B/4 < s(a) < B/2$ and such that $\sum_{a \in A} s(a) = mB$. Can A be partitioned into m disjoint sets $A_1, A_2, \ldots A_m$ such that , for $1 \leq i \leq m$, $\sum_{a \in A_i} s(a) = B$ (note that each A_i must therefore contain exactly three elements from A)?" [Garey and Johnson, 1979]

In this chapter our proof follows the procedures presented [Logendran and Ramakrishna, 1995]. The problem of designing a modular layout was constructed by reducing the original problem to the problem of selecting and assigning products to machines in pre-determined modules. The instance of the special problem is described as follows:

Let Q be a layout problem that contains $3N$ products and N machine types. There are N machines of each type and the machines of the same type are assigned to the same module. There are N fixed modules in this problem. Processing time for product i where $i = 1, 2, \ldots, 3N$ at assigned machine j where $j = 1, 2, \ldots, N$ is a_{ij} and available processing time on machine type j is b_j. There is only one batch for each product and ϕ is the cost of moving a batch of product from one module to another module. Each product requires a unique operation to be performed on every machine type; however, there is no specific order for the operation. The problem is to minimize the total inter-module moves of the products subject to machine capacity constraints.

It can be noticed from the problem description that modules are predetermined and constructed as Functional modules containing N machines of the same type. The objective is then to select and assign all products to machines in N modules where the machine capacity is satisfied and total inter-module moves are minimized.

Let S be a solution of problem Q and $T(S)$ is a function to calculate the total cost of the inter-module moves of solution S. It can be clearly shown that the maximum number of moves is $3N(N-1)$ since there $3N$ products and each product requires N-1 moves for N operations. Therefore, $T(S) \leq 3N\phi(N-1)$ should be satisfied by any solution S of problem Q. So the problem Q can be stated that "given a_{ij}, B_j and ϕ, is there a solution that $T(S) \leq 3N\phi(N-1)$?

The next step in the procedure is to transform the instance of the 3-partition problem into the instance of the problem Q. So let m, B, and $A = \{a_1, a_2, \ldots, a_{3m}\}$ be any instance of the 3-partition problem. The notations on both problems are transformed as follows:

N (problem Q)	=	m (3-partition)
a_{ij} (problem Q)	=	a_i (3-partition) where $i = 1,2,\ldots,3m$ and $j = 1,2,\ldots,m$
B_j (problem Q)	=	B (3-partition)
a (problem Q)	=	$s(a)$ (3-partition)

From the transformation, there are m machines and $3m$ products in the problem. Each product i requires the same amount of processing time a_i on any machine j. Each machine j has the same amount of available time B. In addition, function $s(a)$ in the 3-partition problem is equal to the value of a and the total processing time required by all the operations of products is mB. It can be realized that the entire transformation above is polynomially bounded. Therefore the next step is to show that there is a solution S for the problem Q *if* and *only if* there exists a solution to the 3-partition problem. The proof is as follows:

***If there exists a solution*:** Suppose there is a three partition solution of m disjoint sets A_1, A_2, … A_m such that $\sum_{a \in A_i} s(a) = B$. The solution can be:

$A_1 = \{a_1,a_2,a_3\}$ where $a_1 + a_2 + a_3 = B$

$A_2 = \{a_4,a_5,a_6\}$ where $a_4 + a_5 + a_6 = B$

⋮

$A_m = \{a_{3m-2},a_{3m-1},a_{3m}\}$ where $a_{3m-2} + a_{3m-1} + a_{3m} = B$

From the solution of the 3-partition problem above, it can be seen that we can assign the first operation of products 1,2,3 to machine #1, products 4,5,6, to machine #2 and so on to finally products $3m$-2, $3m$-1,$3m$ are assigned to machine m in the first module. The assignment procedure repeats until all the remaining operations of the products are assigned to remaining machines in remaining modules. Each product requires m-1 moves for m operations and each move costs ϕ. Therefore the total cost of inter-module moves will be equal to $3m(m-1)$ which is satisfied the condition $T(S) \leq 3m(m-1)$ of the problem Q. Accordingly, it can be seen that any instance of 3-partition problem can provide a solution to the problem Q. Hence this part is proven.

Only if there exists a solution: In the 3-partition problem, the following conditions must be satisfied for any instance.

$$B/4 < s(a) < B/2 \text{ and } \sum_{a \in A} s(a) = mB$$

Since $s(a)$ is equal to a, it can be seen that a machine cannot be assigned to 2 or less than 2 products, since there will be one or more remaining machines that needs to be assigned to 4 or more products. If there is one machine assigned to 4 products and each product requires a processing time larger than $B/4$, then this machine must have an available time greater than B. If that happens, it will violate the capacity constrains because each machine has only B available time. In contrast, any machine cannot be assigned to 4 or more products as it violates the problem constraints as well. Therefore, there must be exactly 3 products assigned to each machine proving that there exists a 3-partition solution to the 3-partition problem. Hence this part is proven and we can conclude that the problem Q falls into the class of NP-complete problem so as the original modular layout problem. □

The next section illustrates and validates the optimization approach to solve the problem of designing a modular layout with the revised mathematical model presented earlier. CPLEX which is a commercial integer programming solver is used in this experimental study.

3.5 Experimental Study

To illustrate the procedure of formulating and solving the problem of designing a modular layout, the Vakharia dataset containing 19 products and 12 machines shown in Table 3.1 was used to setup the problem. The production quantity for each product from this table is the demand per week so each value was multiplied be 50 to obtain a production figure for one year.

Product	Quantity	Routing	# of Operations (N_i)
1	2	1→4→8→9	4
2	34	1→4→7→4→8→7	6
3	23	1→2→4→7→8→9	6
4	12	1→4→7→9	4
5	65	1→6→10→7→9	5
6	98	6→10→7→8→9	5
7	34	6→4→8→9	4
8	87	3→5→2→6→4→8→9	7
9	45	3→5→6→4→8→9	6
10	12	4→7→4→8	4
11	67	6	1
12	34	11→7→12	3
13	7	11→12	2
14	26	11→7→10	3
15	34	1→7→11→10→11→12	6
16	89	1→7→11→10→11→12	6
17	45	11→7→12	3
18	23	6→7→10	3
19	23	12	1

Table 3.1: Product routings

Table 3.2 shows the available number of machines and their purchase costs. From the table, there are 3 machines with low duplication costs. This simply implies that these 3 machines are preferred to be duplicated first if necessary. While the other machines have much higher cost of duplication, any of these machines could still be duplicated if its duplication provides a better solution to the problem.

Machine Type	Quantity (*Lk*)	Cost (*Ek*)	Avail. Time (*Fk*)
1	2	300	2,000
2	1	300	2,000
3	1	300	2,000
4	2	10	2,000
5	1	300	2,000
6	2	300	2,000
7	4	10	2,000
8	1	300	2,000
9	2	300	2,000
10	3	10	2,000
11	3	300	2,000
12	1	300	2,000

Table 3.2: The number of machines available and their purchase costs

The annual production time available for each machine and processing time for each operation of each product were missing in the original dataset. Therefore, we use a standard approximation technique to calculate the missing data as follows. For example, there are 2, 34, 23, 12, 65, 34 and 89 units of products 1, 2, 3, 4, 5, 6, 16 and 17 being produced at machine # 1 respectively. So there are 259 units of products/week × 50 weeks/year = 12,950 products being produced at this machine in one year of operation. Therefore, a processing time of the j^{th} operation of product i at machine #1 (P_{ij1}) = 2 copies of machines #1 × 2,000 hours of availability in one year ⁄ 12,950 products produced in one year = 0.309 hour. With the same calculation, the processing time for the remaining products at other machines can be obtained as shown in Table 3.3.

Part No.	Operations						
	1	2	3	4	5	6	7
1	0.309	0.271	0.119	0.219			
2	0.309	0.271	0.302	0.271	0.119	0.302	
3	0.309	0.364	0.271	0.302	0.119	0.219	
4	0.309	0.271	0.302	0.219			
5	0.309	0.191	0.358	0.302	0.219		
6	0.191	0.358	0.302	0.119	0.219		
7	0.191	0.271	0.119	0.219			
8	0.303	0.303	0.364	0.191	0.271	0.119	0.219
9	0.303	0.303	0.191	0.271	0.119	0.219	
10	0.271	0.302	0.271	0.119			
11	0.191						
12	0.335	0.302	0.172				
13	0.335	0.172					
14	0.335	0.302	0.358				
15	0.309	0.302	0.335	0.358	0.335	0.172	
16	0.309	0.302	0.335	0.358	0.335	0.172	
17	0.335	0.302	0.172				
18	0.191	0.302	0.358				
19	0.172						

Table 3.3: The processing time (hour/piece) for each operation

For the sake of simplicity, most parameters are set to 1 in this experimental study. The maximum number of machine types in any module (*SIZE*) is 4. That means each module can contain a maximum of 4 different types of machines. As mentioned, we use the type of machine instead of the number of copies of machine because of the assumption that there is no cost of material handling within the group of the machines of the same type. Therefore, the machines of the same type can be considered as a single unit of machine. The remaining parameters of the problem are as follows:

μ	B_i	BN	HBC_{ikl}	K	M	$SIZE$	T
1	1	1,000	1	12	5	4	19

The maximum number of machine types, referred to as the size of a layout module, can vary and need to be set carefully. For example, if module size is set too large, it will cause machines within a module to be located far apart from each other. For instance, if the module size is unconstraint, a layout solution can converge to a functional layout where a single super module containing all machines is created. If the module size is too small, the layout module becomes the extreme degree of distributed layout where inter-module flows will increase and machine duplication will be required greatly to reduce the flows between small modules. Thus, caution and experience may be needed to ensure an appropriate module size.

Another important parameter is the maximum number of layout modules (M). This parameter plays a significant role in the computational process for solving the problem. The maximum number of layout modules acts as an upper bound for the size of the problem. If this given bound is too tight, then a solution may fall into a local optimum. However, if the given bound is too loose, then a problem size may be too large and the solution process can take days to solve or cannot be solved at all. An approximate bound can be calculated by using the total number of machines in a problem divided by a module size. For this example problem, the approximate bound for the numbers of layout modules can be 12 / 4 = 3 modules. But, this bound does not consider the impact of machine duplication in the optimal solution. So this technique gives a too tight bound and can force the IP solver to give a local optimal solution. To improve this bound, we can relax it by increasing the size of the bound by 50% or even 100% in order to take the effect of machine duplication into account. For this example problem, we can use the value of 3 modules + 100% = 6 modules as the upper bound of the maximum number of layout modules. We can give a larger bound to make sure that the solver will not fall into local optima. However, for this example problem which is considered as a very small problem, when the maximum number of modules was set to 8, the solver ran for days and still could not reach a solution. Care must be taken to set this bound correctly to avoid such computational burden when using this optimization approach.

In the next section, CPLEX an IP solver is used to solve this problem of designing a modular layout. When using commercial IP solvers to solve a non-linear mixed integer problem, it helps the solvers to perform better when all terms in the problem model are linearized. It can be recognized from the revised problem model that there is a quadratic term, a non-linear function that represents the material handling cost as shown below:

$$\sum_{i=1}^{T}\sum_{j=1}^{N_i-1}\sum_{k=1}^{K}\sum_{\substack{l=1\\ l\neq k}}^{K}\sum_{m=1}^{M}\sum_{\substack{n=1\\ n\neq m}}^{M}\mu\cdot HBC_{ikl}\cdot\left\lceil\frac{Q_i}{B_i}\right\rceil\cdot x_{ijkm}\cdot x_{i(j+1)ln}$$

The product of two variables x_{ijkm} and $x_{i(j+1)ln}$ is a non-linear term that can be linearized by using a technique called "usual linearization" [Hammer and Rudeanu, 1968]. The product of the two variables can be replaced with a new variable $y_{ij(j+1)klmn}$ that is introduced into the model with the following constraints added:

$$y_{ij(j+1)klmn}\leq x_{ijkm}$$

$$y_{ij(j+1)klmn}\leq x_{i(j+1)\ln}$$

$$y_{ij(j+1)klmn}\geq x_{ijkm}+x_{i(j+1)\ln}-1$$

$y_{ij(j+1)klmn}$ is binary for each i=1,2,…,T; j=1,2,…,N_i-1; k=1,2,…,K; m and n=1,2,…,M

From the above constraints, if either x_{ijkm} or $x_{i(j+1)\ln}$ is 0, then $y_{ij(j+1)klmn}$ is forced to be 0. That means there is no operation *j* of product *i* performed at machine *k* in module *m* or operation *j*+1 of the same product performed at machine *l* in another module *n*. If both x_{ijkm} and $x_{i(j+1)\ln}$ are 1, then $y_{ij(j+1)klmn}$ is 1 meaning that operation *j* of product *i* is performed at machine *k* in module *m* and operation *j*+1 for the same product is performed at machine *l* in another module *n*.

3.6 Experimental Result

The instance of a modular layout problem from the Vakharia data set was constructed and solved by CPLEX running on Pentium 4, 2.53GHz, 1GB Ram Windows-based computer. This instance problem took 12,786 seconds to solve. The result is shown in Figure 3.2. From the result, there are 4 modules created. Module 1 contains machines 1, 2, 3 and 5. Module 2 contains machines 7, 10, 11, 12. Module 3 contains machines 6, 7 and 10. Module 4 contains machines 4, 7, 8 and 9. It can be observed that machines 7 and 10 are duplicated and placed in different modules. The total material handling cost for the original functional layout for this Vakharia data set is **2472**. This can be considered as the total inter-module flow since the material handling cost is normalized and batch size is 1. The total inter-module flow for the modular layout is **498** which is **79.8%** less than the functional layout. The reduction was achieved by grouping the high traffic machines into the modules with two additional machines being purchased.

	Machine #											
Product #	1	2	3	4	5	6	7	8	9	10	11	12
1	0.02			0.01				0.01	0.01			
2	0.26			0.46			0.51	0.10				
3	0.18	0.21		0.16			0.17	0.07	0.13			
4	0.09			0.08			0.09		0.07			
5	0.50					0.31	0.49		0.36	0.58		
6						0.47	0.74	0.29	0.54	0.88		
7				0.23		0.16		0.10	0.19			
8		0.79	0.66	0.59	0.66	0.42		0.26	0.48			
9			0.34	0.31	0.34	0.21		0.13	0.25			
10				0.16			0.09	0.04				
11						0.32						
12							0.26				0.28	0.15
13											0.06	0.03
14							0.20			0.23	0.22	
15	0.26						0.26			0.30	0.57	0.15
16	0.69						0.67			0.80	1.49	0.38
17							0.34				0.38	0.19
18						0.11	0.17			0.21		
19												0.10
Module 1	2.00	1.00	1.00		1.00							
Module 2							1.72			1.33	3.00	1.00
Module 3						2.00	1.41			1.67		
Module 4				2.00			0.87	1.00	2.00			
M/Cs Required	2	1	1	2	1	2	5	1	2	4	3	1
M/Cs Available	2	1	1	2	1	2	4	1	2	3	3	1
Extra M/Cs							1			1		

Figure 3.2: Results from the optimization for Vakharia dataset

Figure 3.3 shows the routings in the problem updated with the original machines are replaced with the layout modules to which these machines belong. As can be seen from the updated routings, products are now moving less than they were in the original routings. For example, part 1 in the original routing moved from machine 1 to 4 to 8 to 9. In the updated routings in the modular layout solution, this part moves only from module 1 to module 4. This is a major reason why the total material handling cost decreases dramatically in the modular layout, compared to the functional layout.

Product	Orignal Routing						
1	1	4	8	9			
2	1	4	7	4	8	7	
3	1	2	4	7	8	9	
4	1	4	7	9			
5	1	6	10	7	9		
6	6	10	7	8	9		
7	6	4	8	9			
8	3	5	2	6	4	8	9
9	3	5	6	4	8	9	
10	4	7	4	8			
11	6						
12	11	7	12				
13	11	12					
14	11	7	10				
15	1	7	11	10	11	12	
16	1	7	11	10	11	12	
17	11	7	12				
18	6	7	10				
19	12						

Module Routing						
1	4	4	4			
1	4	4	4	4	4	
1	1	4	4	4	4	
1	4	4	4			
1	3	3	3	4		
3	3	3	4	4		
1	4	4	4			
1	1	1	3	4	4	4
1	1	3	4	4	4	
4	4	4	4			
3						
2	2	2				
2	2					
2	2	2				
1	2	2	2	2	2	
1	2	2	2	2	2	
2	2	2				
3	2	2				
2						

Final Routing		
1	**4**	
1	**4**	
1	**4**	
1	**4**	
1	**3**	**4**
3	**4**	
1	**4**	
1	**3**	**4**
1	**3**	**4**
4		
3		
2		
2		
2		
1	**2**	
1	**2**	
2		
3	**2**	
2		

Figure 3.3: Updated routings with machines replaced by layout modules

When the problem of designing a layout module has been solved, a detailed layout for each module can then be constructed. Typically a layout module can fall into one of the three traditional types—functional, cell, and flowline modules. The layout modules for this solution can be constructed as follows:

- The modules 1 and 4 can be constructed as “Functional Layout Modules.” These two modules are used randomly in the routings. Therefore, these modules are best laid out as Functional Layout Modules.
- The module 2 can be constructed as “Cell Module.” It performs the complete sets of operations for 5 parts including parts 12,13,14,17, and 19. It also performs most operations required by the other parts that use this module.
- The module 3 can be constructed as “Flowline Module since flows are only in forward direction as seen in Figure 3..

Figure 3.4 shows the material flow network of the functional layout and Figure 3.5 shows the material flow network of the modular layout for the same Vakharia data set. It can be noticed that the complexity of the material flow network as well as the total travel distance of material flow for the modular layout is much less than the functional layout. However, the complexity is reduced not because the material movements are reduced. It is because most movements with high traffics now have shorter distances. That is the key to simplify the material flow network in a functional layout by changing to a modular layout.

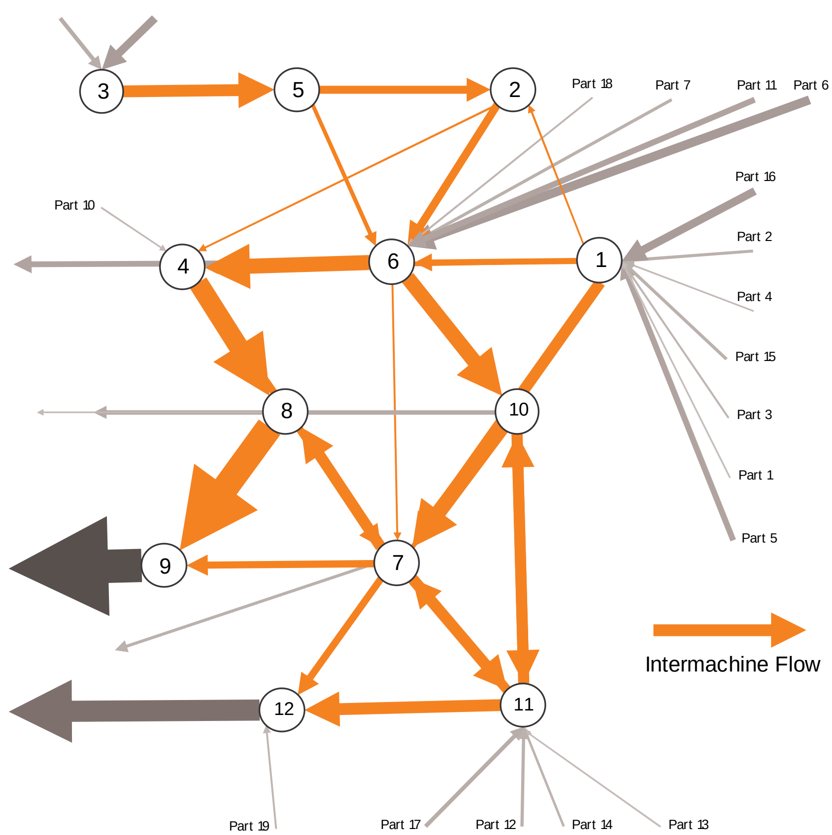


Figure 3.4: Material flow network for the functional layout for Vakharia dataset

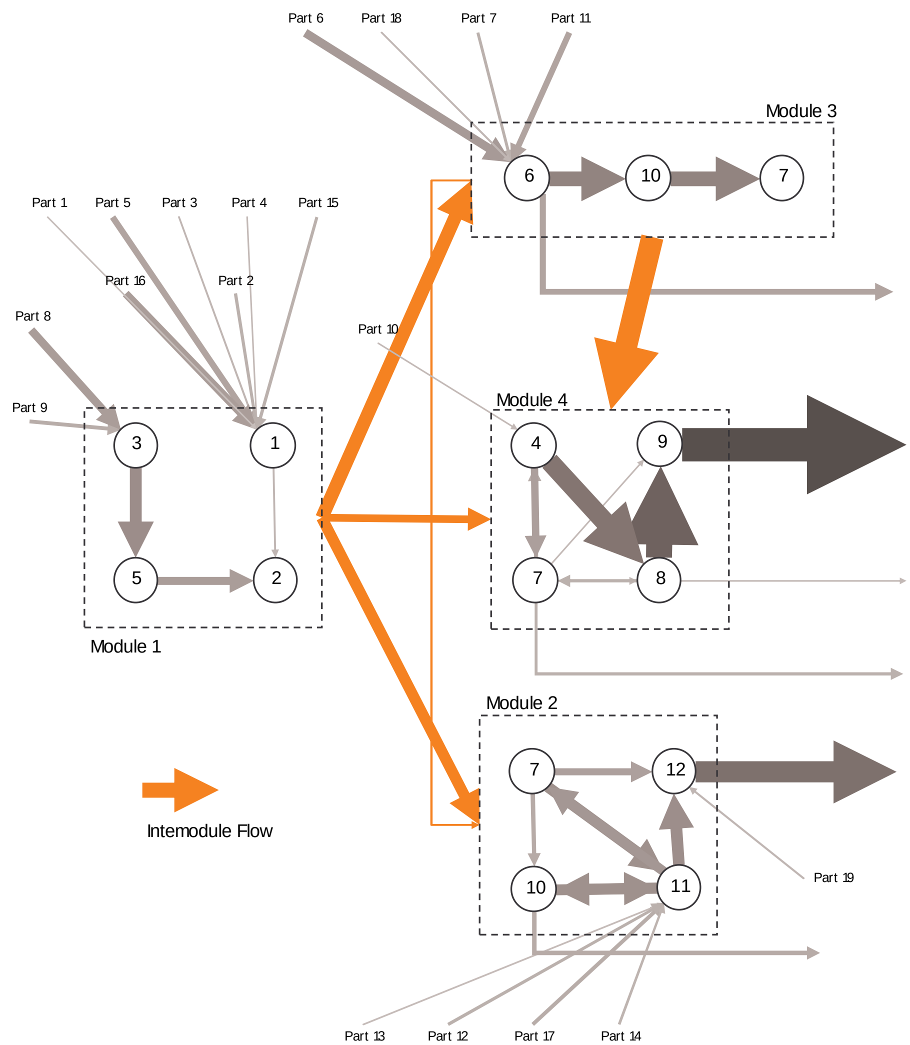


Figure 3.5: Material flow network for the modular layout for Vakharia dataset

3.7 The Problem Model Simplification

The problem of designing a modular layout falls into a category of NP-complete problem as proven in Appendix A. In addition to the complexity of the problem that is known to be very difficult to solve, the mathematical model for the designing problem shown in previous section requires a great detail of production information as for input data. This is one of the major reasons that make a modular layout be difficult to implement in practice. Therefore, the original mathematical model for solving the problem of designing a modular layout is being simplified by eliminating some insignificant parameters and relaxing some constraints so that the data requirement as well as the size of the problem is reduced. There are three areas of simplification including (1) setup time and processing time removal, (2) intra-module material cost removal, and (3) problem size reduction using a two-stage approach.

3.7.1 Setup Time and Processing Time Removal

It is typical to have hundreds to thousands of routings in a small HMLV facility and each routing may contain ten or more operations and setups. Tremendous resources and investment are needed solely to acquire this data, which is unlikely to be worth the effort from management point of view. Therefore, it helps when employing this designing model in practice if it is possible to not acquire the complete information of setup times and processing times.

From the original model, it can be realized that S_{ijk} and P_{ijk} (setup time and processing time) need to be provided. These two parameters do not vary by the different arrangements of machines in layout modules. Therefore, the total production cost in the objective function $\sum_{i=1}^{T}\sum_{j=1}^{N_i}\sum_{k=1}^{K}\sum_{m=1}^{M}\left\lceil \frac{Q_i}{B_i} \right\rceil \cdot S_{ijk} \cdot x_{ijkm} + \sum_{i=1}^{T}\sum_{j=1}^{N_i}\sum_{k=1}^{K}\sum_{m=1}^{M} Q_i \cdot U_{ijk} \cdot P_{ijk} \cdot x_{ijkm}$ does not change and can be omitted from the model.

It seems like with the production cost the objective function, the processing time and the setup time are not needed. However, the cost function is not an only place in the model that requires the processing time P_{ijk}. There is also a constrain set governing the number of machines required in the original model that requires P_{ijk}. This constrain set can not be omitted. However, with the approximation of the processing time that has been used conventionally and conveniently in the real world cases, this constrain set can be simplified. The detail will be described in the section 3.6.3 where a two-stage approach for the model is introduced.

3.7.2 Intra-Module Material Handling Cost Removal

With the removal of the production cost, the other costs remaining in the problem model are machine duplication cost and material handling cost. In typical job shops, with their tight budgets, the cost of machine duplication is always significant. Therefore, this cost in the objective function of the model does not much vary compared to the material handling cost. The cost of material handling has the greatest impact to the layout problem model. There are two types of material handing costs—intra-module and inter-module material handling costs. The intra-module material handling cost in a modular layout can be neglected since the material movements inside a module can be done easily by machine operators working in the module. This leads to a simplified mathematical model where the intra-module material handling cost can be omitted and the production cost is also omitted as described in 3.6.1. Therefore, the objective function of the original model is reduced as shown below:

$$\text{Minimize } \sum_{k=1}^{K} E_k \cdot \left(\sum_{m=1}^{M} r_{km} - L_k \right)$$

$$+ \mu \cdot \sum_{i=1}^{T} \sum_{j=1}^{N_i - 1} \sum_{k=1}^{K} \sum_{\substack{l=1 \\ l \neq k}}^{K} \sum_{m=1}^{M} \sum_{\substack{n=1 \\ n \neq m}}^{M} HBC_{ikl} \cdot \left\lceil \frac{Q_i}{B_i} \right\rceil \cdot x_{ijkm} \cdot x_{i(j+1)ln}$$

This new objective function becomes a pure trade-off between the cost of purchasing extra machines and the intra-module material handling cost while μ is serving as the balancing factor that makes the two costs comparable.

3.7.3 Two-Stage Approach for Problem Size Reduction

In the formulation, the processing time P_{ijk} in the machine capacity calculation constrain set is needed to determine the number of machines required in each layout module. This parameter cannot be omitted from the model; however, when the processing time is approximated as it is done in the experimental study section, this constraint set can be simplified and the size of the problem can be greatly reduced.

Since there can be several thousands of distinct operations required in even a small HMLV manufacturing facility, obtaining the exact value of processing time for each operation is such a tremendous work. Therefore, it is typical to see that the layout analysts use the approximation technique to approximate the value of processing time for each operation in the designing process. When the processing time is approximated using this technique, we have found that the machine capacity calculation in the modular layout problem model can be done independently using a two-stage approach. The following statement shows the motive of this proposed technique.

"Suppose there are *X* machines of type *Y* and this machine type needs to be allocated in two modules *A* and *B*. Module *A* requires *a* machines and module *B* requires *b* machines where $a + b = X$ machines. Since the number of machines needs to be an integer number, module *A* must have $\lceil a \rceil$ machines and module *B* must have $\lceil b \rceil$ machines where $\lceil z \rceil$ denotes the smallest integer that is larger than or equal to *z*. As a result, if *X* machines of type *Y* need to be located in two modules, then there will be at most *X*+1 machines required. If there are *N* modules that require the machine of the same type, then there will be at most *X*+*N*-1 machines required for *N* modules."

The statement above always holds if a machine needs to be duplicated and allocated into two locations. For example, if there 3 machines of type Y, and two modules require 1.3 and 1.7 units of machine type Y, then the total number of machines required will be $\lceil 1.3 \rceil + \lceil 1.7 \rceil$ = 2+2 = 4 (3+1) machines. The only case that two modules need only 3 machines is that when one module requires exactly 1.00 machine and the other requires 2.00 machines, or vice versa. The possibility of such case would probably be the same as that when a module requires 1.9999 and the other module needs 0.9999 machines. If a machine needs to be duplicated and allocated into more than three or more locations, the machine capacity will always be enough capacity by *X*+*N*-1 machines. That means the layout solution will always be feasible with this approximation but it may not guarantee the optimality of the solution. However, this approximation leads to a very simple problem model that also reduces tremendous burden of computational time for solving this problem which is worthwhile in practical situations.

Given that the processing time is approximated by the total number of machines available, the solution approach can be modified and solved in two stages. In the first stage, the problem will be solved without using the detailed processing time and machine capacity. After the first stage is solved, the machine capacity allocation and the exact number of machines required will be determined in the second stage. The simplified mathematical model for designing a modular in the first stage is shown as follows:

Minimize $\sum_{k=1}^{K} E_k \cdot \left(\sum_{m=1}^{M} r'_{km} - 1 \right)$

$$+\mu \cdot \sum_{i=1}^{T} \sum_{j=1}^{N_i - 1} \sum_{k=1}^{K} \sum_{\substack{l=1 \\ l \neq k}}^{K} \sum_{m=1}^{M} \sum_{\substack{n=1 \\ n \neq m}}^{M} HBC_{ikl} \cdot \left\lceil \frac{Q_i}{B_i} \right\rceil \cdot y_{ij(j+1)klmn}$$

subject to:

$\sum_{k=1}^{K} \sum_{m=1}^{M} x_{ijkm} = 1$, for each (i, j)

$\sum_{k=1}^{K} r'_{km} \leq SIZE$, for each m

$\sum_{i=1}^{T} \sum_{j=1}^{N_i} x_{ijkm} \leq BN \, . \, r'_{km}$, for each (k, m)

$\sum_{m=1}^{M} r'_{km} \leq MaxL_k - L_k + 1$, for each k

$y_{ij(j+1)klmn} \leq x_{ijkm}$, for each (i, k, l, m, n) and $j=1, 2, \ldots, N_i-1$

$y_{ij(j+1)klmn} \le x_{i(j+1)ln}$, for each ($i$, k, l, m, n) and j=1, 2, …, N_i-1

$y_{ij(j+1)klmn} \ge x_{ijkm} + x_{i(j+1)ln} - 1$, for each ($i$, k, l, m, n) and j=1, 2, …, N_i-1

r'_{km} binary, for each (k, m)

x_{ijkm} binary, for each (i, j, k, m)

$y_{ij(j+1)klmn}$ binary, for each (i, k, l, m, n) and j=1, 2, …, N_i-1

$\lceil z \rceil$ denotes the smallest integer that is larger than or equal to z

Variables:

r'_{km} = 1 if machine type k is assigned to module m;

= 0 otherwise.

x_{ijkm} = 1 if the j^{th} operation of product i is processed at machine type k in module m;

= 0 otherwise.

$y_{ij(j+1)klmn}$ = 1 if the j^{th} operation of product i is processed at machine type k in module m and $(j+1)^{th}$ operation of the same product i is processed at machine type l in module n

= 0 otherwise.

The most significant change that has been done to the model is the replacement of variable r_{km} which is an *integral* number representing the *number* of units of machine type k assigned to module m in the original model. With the proposed two-stage approach, this variable has been replaced with r'_{km} which is a *binary* number representing the presence of machine type k in module m. So instead of having an integer variable, the simplified model has replaced this variable with a bounded binary variable. Intuitively, this replacement should dramatically reduce the size of the problem model and reduce the computational effort for solving the problem of designing a modular layout.

Since r_{km} have been replaced with r'_{km}, the capacity constraint set $\frac{\sum_{i=1}^{T}\sum_{j=1}^{N_i} Q_i \cdot P_{ijk} \cdot x_{ijkm}}{F_k} \leq r_{km}$ from the original model has to be revised. This constraint set was used to determine the number of machines required r_{km}. In the revised model, the formula does not need to identify the exact. It needs only to know whether a machine is used in a certain module. Thus, this constraint set is not needed in the revised model. However, the model still needs constraints that tide both variables x_{ijkm} and r'_{km} together. Therefore, additional constraints $\sum_{i=1}^{T}\sum_{j=1}^{N_i} x_{ijkm} \leq BN \, . \, r'_{km}$ to tide these two variables have been introduced where BN is any big number that forces r'_{km} to be greater than zero whenever $\sum_{i=1}^{T}\sum_{j=1}^{N_i} x_{ijkm}$ is greater than 0 which means there exists at least one machine type k in module m.

The revised model as described above provides only partial solutions to the problem. The remaining part of the solution that needs to be solved is the allocation of machine capacity that also results in the number of machines required in each module. This is the post process of the two-stage approach. Once the initial solution from the revised model from the first stage is obtained, the detailed calculation to determine the number of machines and the machine capacity required is executed. Since each the operation at each machine is each module is already obtained from the first stage, the capacity calculation is straightforward. By obtaining all the operations at the same machine type in the same module and aggregating the processing times for these operations, the total time required for this machine can be obtained and then can be easily converted to the number of machines required for each module.

3.8 Experimental Study for the Simplified Model

It can be seen from the previous section that the formulation of the problem model has become simpler when the capacity constraints are not included. When using this simplified model to setup a modular layout problem with the same Vakharia data set, the solution from this simplified model is shown in **Figure 3.6: Results from the simplified model**Figure 3. and the flow diagram for this layout solution is shown in Figure 3.. There are 4 modules including [1,2,6,10], [7,10,11,12], [3,5], and [4,7,8,9] where machines 7 and 10 are duplicated and allocated into two different modules. After the first stage solution is obtained, the second stage for allocating the exact number of machines and the capacities required for each module is executed.

Product	Machine 1	2	3	4	5	6	7	8	9	10	11	12
1	0.02			0.01				0.01	0.01			
2	0.26			0.46			0.51	0.10				
3	0.18	0.21		0.16			0.17	0.07	0.13			
4	0.09			0.08			0.09		0.07			
5	0.50					0.31	0.49		0.36	0.58		
6						0.47	0.74	0.29	0.54	0.88		
7				0.23		0.16		0.10	0.19			
8		0.79	0.66	0.59	0.66	0.42		0.26	0.48			
9			0.34	0.31	0.34	0.21		0.13	0.25			
10				0.16			0.09	0.04				
11						0.32						
12							0.26				0.28	0.15
13											0.06	0.03
14							0.20			0.23	0.22	
15	0.26						0.26			0.30	0.57	0.15
16	0.69						0.67			0.80	1.49	0.38
17							0.34				0.38	0.19
18						0.11	0.17			0.21		
19												0.10
Module 1	2.00	1.00				2.00				1.46		
Module 2			1.00		1.00		1.90			1.54	3.00	1.00
Module 3												
Module 4				2.00			2.10	1.00	2.00			
M/Cs Required	2	1	1	2	1	2	5	1	2	4	3	1
M/Cs Available	2	1	1	2	1	2	4	1	2	3	3	1
Extra M/Cs							1			1		

Figure 3.6: Results from the simplified model

The total inter-module flow for the modular layout is reduced to **678** which is **72.5%** less than the total flow for the functional layout. The percentage of flow reduction for this simplified model is near the optimal solution obtained from the original problem model which is **79.8**%. However, the most recognized improvement with this simplified model is that the computational time to solve the same problem was reduced from **12,786** seconds for the original problem model to only **32** seconds for the simplified model. The most significant factor that contributes to the computation time reduction is the replacement of variable r_{km} which is an *integral* number to r'_{km} which is a *binary* number. By using the approximation technique for processing time, the simplified model does not need to calculate the exact number of machine required in each module. It only needs to identify the presence of machine in each module. Then if it happens that machine A is needed in X modules, there will be X-1 additional copies of machine A required. Therefore, the complexity and the size of the problem are reduced dramatically with this simplification technique.

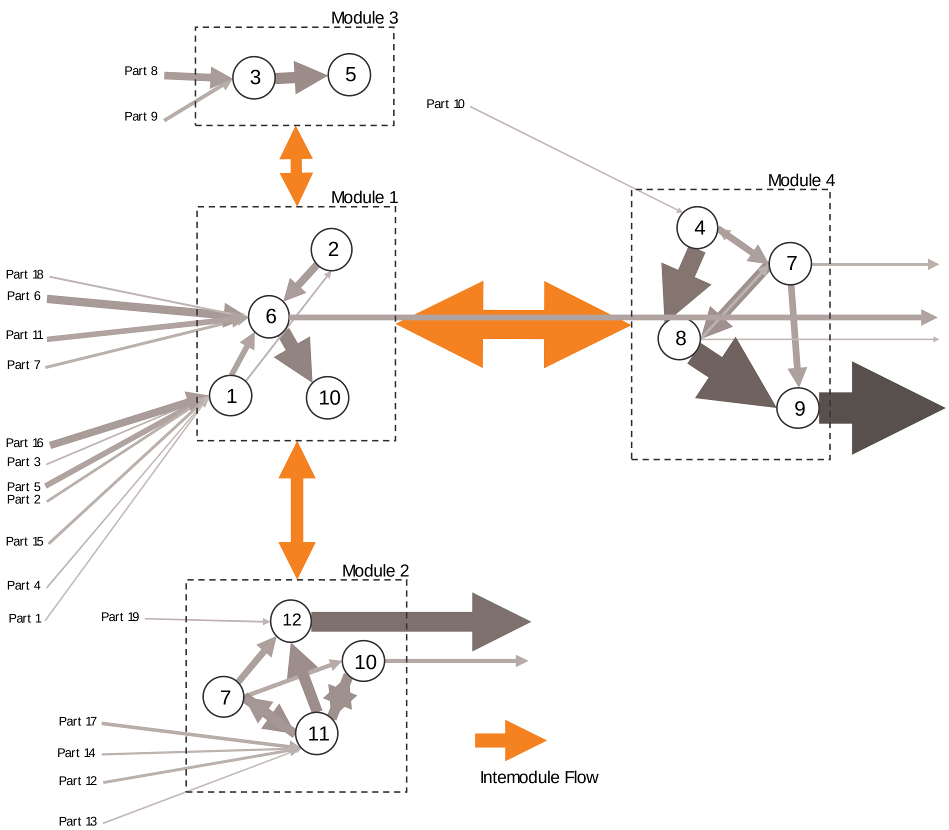


Figure 3.7: Flow diagram for the modular layout from the simplified model

3.9 Conclusion and Discussion

In this chapter, we introduced the concept of a modular layout designing that combines the attributes of the traditional layouts—Functional, Cellular and Flowline—by decomposing a complex material network into smaller sub-networks that exhibit flow patterns observed in different the traditional layouts and scheduling problems. Therefore, instead of designing a single layout that suits only a portion of the material flow network in an entire manufacturing facility, a modular layout can decompose the entire network into several layout types fit the flow patterns of different portions of the entire network. Therefore, this layout is one of the most flexible and appropriate layouts to be suggested for complex high-mix low-volume facilities.

The problem of designing a modular layout was proven to fall into a class of NP-complete problem. The mathematical model for this problem was described, revised and simplified using a two-stage approach proposed in this research. When processing times are approximated, machine capacity requirement and machine allocation for each module can be done independently after the modules are formed. With this two-stage approach, the original problem can be simplified and the problem size can be reduced greatly. The experiment study has shown that the computational time to solve the same modular layout problem using the simplified is reduced dramatically compared to the original model, while the correctness of the solution is not much worsening.

References

1. Askin, R.G., Lundgren, N.H. and Ciarallo, F., "A material flow based evaluation of layout alternatives for agile manufacturing," in *Progress in Material Handling Research* (Ed: Graves R.J. et al), Ann Arbor, MI: Material Handling Institute, Braun-Brumfield Inc., pp. 71-90 (1996).

2. Babu, S.A., Nandurkar, K.N., Thomas, A., "Development of virtual cellular manufacturing systems for SMEs." Logistics Information Management, Vol. 13, No.4, pp. 228-243 (2000)

3. Balakrishnan, J. and Cheng, C.H., "Multi-period planning and uncertainty issues in cellular manufacturing: A review and future directions," *European Journal of Operational Research*, Vol. 177, pp. 281-309 (2007).

4. Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Company, New York (1979).

5. Irani, S.A., Cavalier, T.M., Cohen, P.H., "Virtual manufacturing cells: Exploiting layout design and intercell flows for the machine sharing problem." *International Journal of Production Research*, Vol. 31, No. 4, pp. 2901-2923(1993).

6. Irani, S.A. and Huang, H., "Custom design of facility layouts for multi-product facilities using Layout Modules," *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 3, pp. 259-267 (2000).

7. Irani, S.A. and Huang, H., "Cascading flowlines and layout modules: Practical strategies for machine duplication in facility layouts," *International Journal of Flexible Manufacturing Systems*, Vol. 17, No. 2, pp. 119-149 (2005).

8. Lahmar, M. and Benjaafar, S., "Design of distributed layouts," IIE Transactions, Vol. 37, pp. 303-318 (2005).

9. Logendran, R. and Ramakrisha, P., "Manufacturing cell formation in the presence of lot splitting and multiple units of the same machine," *International Journal of Production Research*, Vol.33, No. 3, pp. 675-693 (1995).

10. Montreuil, B., Lefrancois, P, Marcotte, S. and Venkatadri, U., "Holographic layout of manufacturing systems operating in chaotic environments. *Technical Report* 93-53, Laval University, Québec, Canada (1993)

11. Prince, J. and Kay, J.M., "Combining lean and agile characteristics: creation of virtual groups by enhanced production flow analysis," *International Journal of Production Economics*, Vol. 85, pp. 305-318 (2003).

12. Venkatadri, U., Rardin, R. and Montreuil, B., "A design methodology for fractal layout organization," *IIE Transactions*, Vol. 29, pp. 911-924 (1997).

CHAPTER 4

DESIGN OF MODULAR LAYOUTS USING A NEW CUT-TREE ALGORITHM

4.1 Introduction

The two-stage approach for solving the problem of designing a modular presented in previous chapter can provide very good results and can greatly reduce the complexity of the original problem model. However, the approach is still a non-linear mix integer programming based model which is proven to have no technique to solve in polynomial time [Garey and Johnson, 1979]. Therefore, this approach is not applicable for real-world size problems. A heuristic approach to tackle the large-size problems of designing a modular layout is developed.

4.2 Heuristic Solution Approach for Design of Modular Layouts

Our heuristic approach for solving the problem of designing a modular layout is based on a graph theoretic approach which is one of the most commonly used approaches for solving layout design problems. The graph theoretic approach models a facility in a layout problem as a simple graph where its vertices represent departments (or machines) and its edges represent material flows among departments (or machines). In our approach, a flow network graph problem is constructed to represent a modular layout problem. Cut Tree algorithm is used to find strongly connected sets of nodes in the flow network graph and form these nodes into layout modules. The cut tree algorithm has been applied successfully in solving a functional layout problem [Montreuil and Ratliff, 1989] and a cellular layout problem [Kandiller, 1998, and Lee and Chiang, 2001]. Since the modular layout and the cellular layout design problems have many similar characteristics, with some modification, the cut tree algorithm can be also used for solving the modular layout design problem.

The heuristic approach proposed for designing a modular layout uses similar strategy as used in the two-stage optimization approach where grouping machines into layout modules is executed first and allocating machine capacity to each module is later. The approach consists of 3 phases as follows:

(1) Network flow problem formulation where a network flow graph representing a modular layout problem is constructed,

(2) Cut-tree transformation where a cut tree is created from the network flow graph in phase 2, and

(3) Layout module construction where nodes representing machines in the cut tree are grouped into layout modules.

The detail of the 3-phase heuristic approach is described next.

4.2.1 Problem Formulation of a Flow Network Graph

The first phase of the heuristic approach is to construct a flow network graph representing the modular layout problem. The flow network graph is an undirected graph *G(N,A)* where *N* is a set of nodes representing machines and *A* is a set of arcs representing flows between a pair of machines. For any undirected arc in the graph, F_{ij} represents a flow from machine *i* to machine j which is equal to F_{ji}, a flow from machine *j* to machine *i*. The Vakharia dataset of 12 machines and 19 products is used to demonstrate how this heuristic approach works. The input data needed for this first phase are product routings and production quantities as shown in Figure 4.1. After the routings and production quantities are obtained, a single line graph representing a product routing is created. Each edge in this graph has a weight (or flow) which is equal to the production quantity of a product of this edge. For example, the routing of product 1 is 1➔4➔8➔9; therefore a single-line graph representing this product routing will be node (vertex) 1 connects to vertex 4, node 4 connects to node 8, and node 8 connects to node 9. Each pair of nodes is connected with an arc (edge) with the weight of 2 which is equal to the production quantity of product 1. The single-line graphs for all the routings created for this Vakharia dataset are as shown in Figure 4.2.

Product	Quantity	Routing
1	2	1→4→8→9
2	34	1→4→7→4→8→7
3	23	1→2→4→7→8→9
4	12	1→4→7→9
5	65	1→6→10→7→9

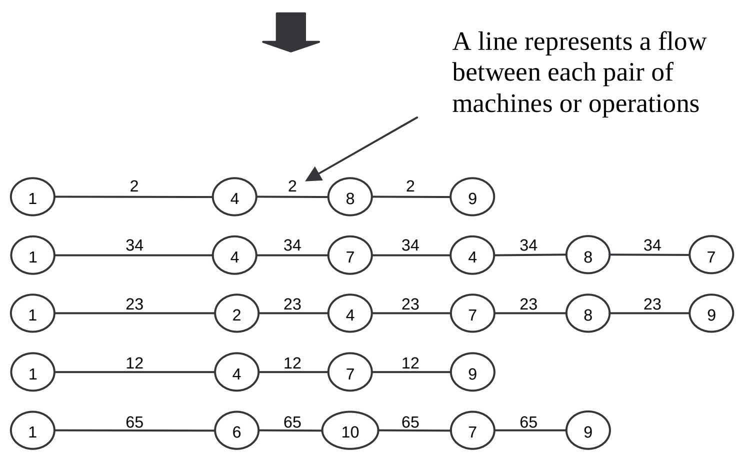


Figure 4.1: The construction of single-line graphs representing the product routings

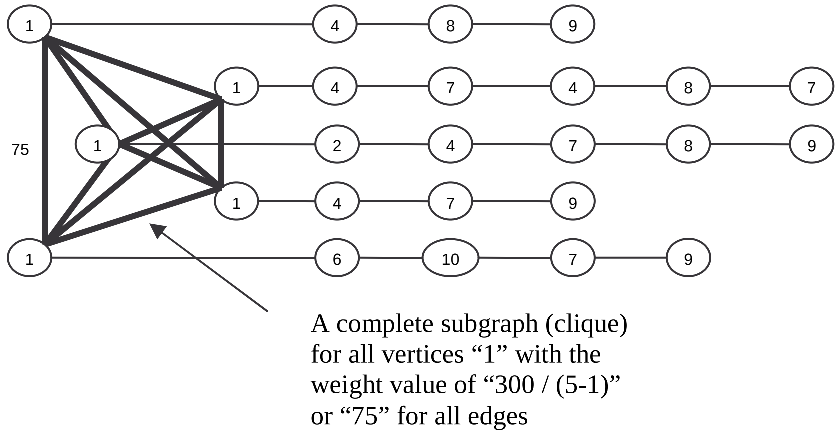


Figure 4.2: The complete subgraph creating all identical nodes due to the duplication of machine type 1

The next step in this phase is to connect all identical machine nodes (vertices) together. When the identical machine nodes are connected, they forms a number of cliques, which are complete subgraphs residing in a parent graph. All arcs (edges) that connect the identical nodes are weighted with a value reflecting the cost of machine duplication. In our approach, the weight value for identical machine nodes is equal to the cost of node (machine) duplication divided by the total number of identical nodes minus 1. This value is called “a weight value of machine duplication cost.” An example for a clique constructed by connecting all identical machine nodes 1 in the problem graph is shown in Figure 4.8. As can be seen from the figure, machine nodes “1” are all connected by arcs with the weight value of 300/(5-1) = 75. Similarly, a similar clique will be created for each machine type that occurs in multiple nodes across the flow network graph.

The weight value of duplication cost plays a significant role in this heuristic approach. It has a considerable impact on the solutions of a modular layout problem. If the weight values are too high, then all the identical nodes that are connected with these heavy weight values will not be broken up. It implies that there will be no machine duplication allowed in the layout solution. That means the approach will eventually create a functional layout that does not contain any duplicated machines. In contrast, if the weight value is too low, the approach could eventually result back to the original single-line graphs constructed in the first step of this approach. If that is the case, the approach will give a solution that is a flowline layout (flowline module) for each product of a problem. Therefore, to ensure good results, iterative experimentation based on experience is needed to determine appropriate values for this weight.

Once all the identical nodes are connected, the single-line graphs in the first step of this approach are supposed to connect and construct a flow network graph for the problem. After the flow network graph is constructed, the next phase of the heuristic approach is a “cut-tree network transformation phase” where the network flow graph is transformed into a cut-tree. A cut tree is a spanning tree where all nodes in a flow network graph are retained but these nodes are linked by newly introduced arcs with specific weights. Next, the cut-tree algorithm, an algorithm to generate the cut tree based on a multi-terminal network model [Gomory and Hu, 1961], is described.

4.2.2 Cut-Tree Transformation

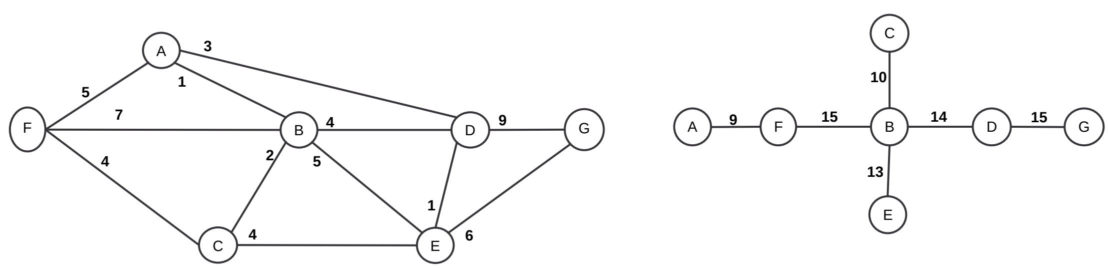
Usually, when solving a maximum flow problem, the problem network must contain only one sink node and one source node. If we want to find a maximum flow between each pair of nodes in a network with undirected arcs, then we need to solve $\binom{n}{2}$ = $n(n\text{-}1)/2$ maximum flow problems for all pairs of nodes. This particular type of network problem is called a multi-terminal network problem. Gomory and Hu (1961) has shown that for n-node multi-terminal network, $n(n\text{-}1)/2$ maximum flows can be found by solving only n-1 maximum flow problems. The n-1 maximum flows from Gomory and Hu's algorithm can then constructs a spanning tree that consists of n original nodes and n-1 arcs obtained from n-1 maximum flow problems. Each node in the spanning tree represents the original node of the original multi-terminal network. Each arc in the spanning tree represents a maximum flow or a minimum cut value in the original network where the two nodes, which are linked by this arc, are the source and the sink in the original network. This spanning tree is the called a cut tree and it is going to be used as a design skeleton to construct a modular layout in our heuristic approach.

The step procedures for constructing a cut tree from a multi-model network are as follows:

1) Select a pair of nodes n_i and n_j in an original flow network Q and perform a max-flow min-cut computation to obtain a maximum flow f_{ij} and cut sets C_i and C_j.
2) Construct an initial cut tree T from the cut sets C_i and C_j where the cut sets are new super nodes (a node that contains more than one node in itself) N_i and N_j

of the tree *T*. These two super nodes are linked by an arc with its weight is equal to the maximum flow f_{ij} calculated in step 1.

3) If there is any super node *NN* in the cut tree *T*, select a pair of single nodes n_k and n_l inside the super node *NN*. Perform another max-flow min-cut computation from the original network *Q* to obtain a maximum flow f_{kl} where other nodes outside of *NN* in the original network *Q* are replaced by their corresponding nodes in the current cut tree *T*.
4) Obtain the new cut sets C_k and C_l and update the cut tree *T*.
5) Repeat from step 3 until there is no super node remaining in the cut tree *T*.
6) The process stops, the final cut tree *T* is updated and it is the result to the cut-tree network transformation procedure.



a) Original Flow Network

b) Cut Tree

Figure 4.3: Example of a cut tree created from a flow network

Figure 4.3 shows an example of a cut tree created from an original flow network. Once the cut tree is constructed, the next step is to generate the sets of machines or so-called layout modules in a modular layout. The process of generating the layout modules is described next.

4.2.3 Construction of Layout Modules

A cut tree created from a network flow problem has a unique property. If we separate the nodes in a cut tree into two disjoint sets of two connected sub-trees e.g. {A,B,C,E,F} and {D,G} of the cut tree in Figure 4.9 (b). The total flow between these two sets of nodes in the original network graph (shown in Figure 4.9 (a)) is equal to the smallest value of flow of arc that links the two sub-trees i.e. the arc that links nodes B and D which its weight is 14. Therefore, the total flow between the two sets of nodes in the network (a) in Figure 4.10 is also 14. With this unique property, if we want to find the two sets of nodes in the original network that give the smallest total flow, we can look at the cut tree of this original network and seek for smallest value of arc in the cut tree. Accordingly, suppose we want to create a flow network of a material flow in a facility and we want to find the smallest flow between two sets of machines, we can use the cut tree technique as described above. Therefore, the flow network can be used to represent a material flow network in a modular layout problem and the cut tree can be used to provide the minimum values of flows between pairs of machine groups in the problem.

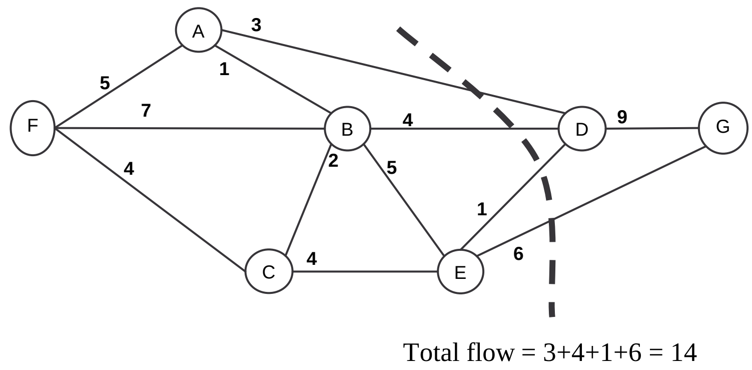


Figure 4.4: Example cut

If nodes in a cut tree that are connected with high weights (flows) are merged into groups of nodes, these groups are then connected to each other with remaining arcs that contain lighter flows. This is comparable to the concept of designing a modular layout where layout modules internalize and co-relate the sets of machines with heavy flow connectivity. At the same time, flows that occur between the modules are minimized. Therefore, by merging the nodes in the cut tree of a modular layout problem that are connected by heavy flows into groups of nodes and leaving the light flows outside of these groups of nodes, we are generating layout modules that contain high traffic volumes inside the modules and leave all the light traffic volumes (less interconnections) outside the modules.

Finding the right nodes in the cut tree of a modular layout problem to merge to a layout module is equivalent to the problem of finding a k-capacitated minimum spanning tree which again belongs to the class of NP-complete [Garey and Johnson, 1979]. So a greedy method will be used here to assure good results with a reasonable computational time spent on the merging process. The greedy method to selecting and merging heavily linked nodes in the cut tree to form the layout modules is described as follows:

1) List all arcs *L* in the initial cut tree *T* and sort them in descending order by their weights.
2) Mark all arcs in the list *L* as "Unselected".
3) Search through the list *L*. If a pair of nodes n_i and $n_{j,}$ connected by an arc *l* in the list *L* correspond to the same machine *k*, merge these two nodes n_i and n_j into one super node $\{n_i, n_j\}$. Remove the arc *l* from the list *L* and update the cut tree *T*.
4) Repeat step 3 until no nodes are left to merge.
5) From the list *L*, select the largest weighted arc *l* which is marked as "Unselected". If a pair of nodes n_i and n_j linked by this arc *l* can be merged[1], then merge these two nodes into one super node "*Ni* and *Nj*". Remove the arc *l* from the list *L* and update the cut tree *T*. If these two nodes cannot merge, mark this arc *l* as "Selected" and perform the next step.
6) Repeat step 3 until all arcs in the list *L* have been marked "Selected" or have been removed.
7) Check all the nodes, either single nodes or super nodes, whether they can be merged further to reduce the number of machines duplicated in the final solution.
8) If step 7 yields, then merged nodes that can be merged until no more "merge-able" nodes remain. When this procedure stops, all single and super nodes remaining in the cut tree are the layout modules for the modular layout.

The last phase of the heuristic approach provides a feasible set of layout modules and also the allocation of products to machines in the modules. This 3-phase cut tree heuristic approach for designing a modular layout including the last phase described above is expressed as a flow chart in Figure 4.5.

What remains for the solution is to calculate the machine capacity and the number of machine required for each module. This process is identical the process used in the second phase of the two-stage optimization approach described in chapter 3. An illustration of how the cut tree heuristic approach works is described next.

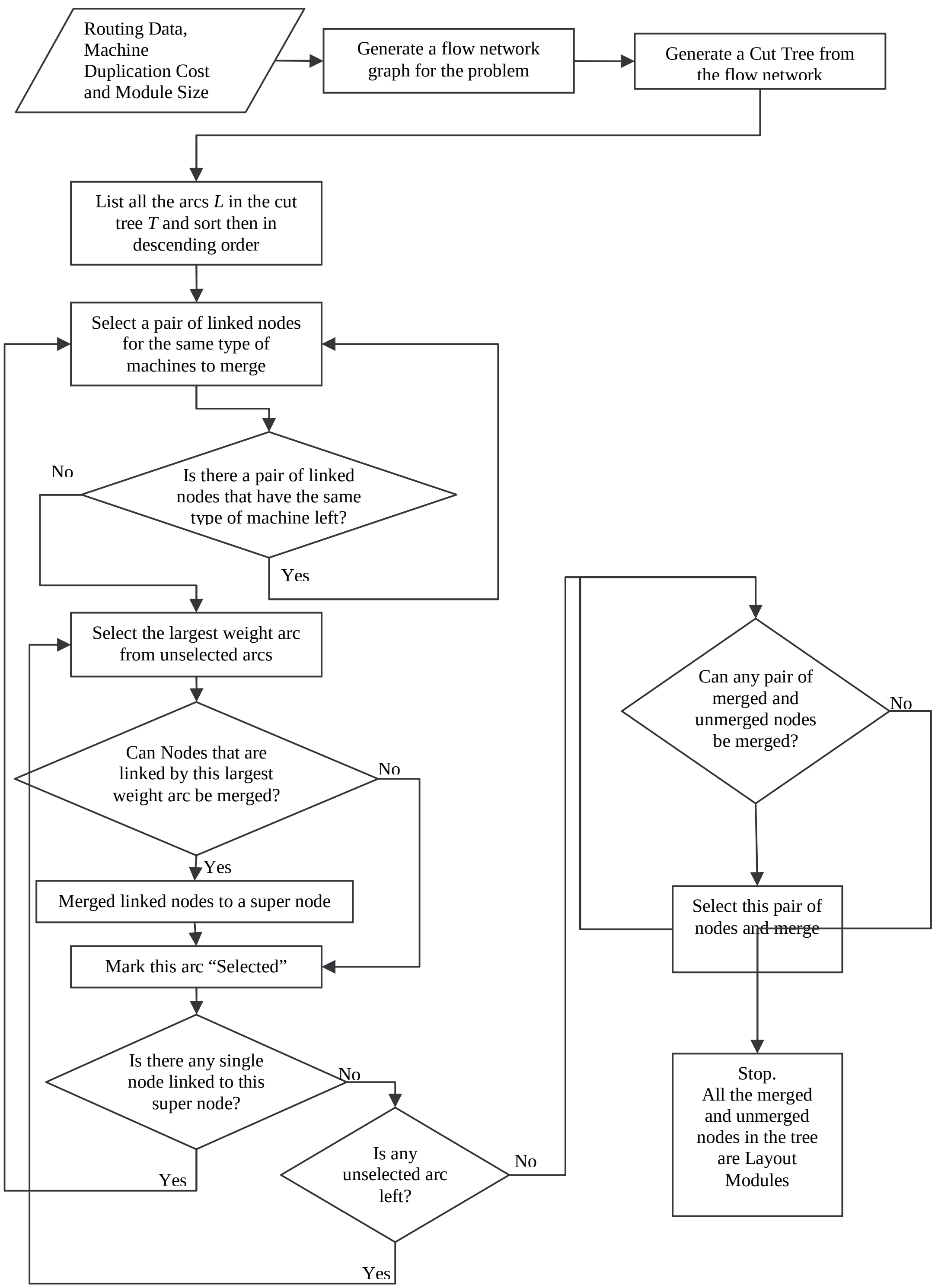


Figure 4.5: The flow chart for the heuristic approach for design of a modular layout

4.3 Illustration of the Cut Tree Heuristic Approach

The Vakharia dataset is used here to demonstrate how the cut tree heuristic approach works. The first step is to generate a flow network problem shown in Figure 4.6. For each node in this flow network, there are two numbers separated by "|". The first number is corresponding to a product number and the second number is corresponding to a machine number. Nodes that correspond to the same product are connected to form the single-line graphs that represent the product routings. Arcs that connect these nodes have their weights equal to the production volumes of the corresponding products. In Figure 4.8, nodes that represent machine 8 (node labeled as "|8") are the only nodes shown completely connected. The other nodes that represent the same machines need to be connected completely as well. However, they are not shown in this figure for the purpose of clarity and readability. Each link that connects a pair of nodes corresponding to the same machine type has a weight equal to the cost of machine duplication divided by (n-1), where n is the total number of nodes of the same machine type. For example, there are 7 nodes in the flow network of machine type 8. The cost of duplication for machine 8 is 300. Therefore, the weight value of each link that connects the nodes of machine 8 is equal to 300 / (7-1) = 50.

After the flow network is constructed, it will have to be transformed into a cut tree. Figure 4.7 shows the cut tree for this flow network. After the cut tree is created, the process for constructing layout modules starts as described next.

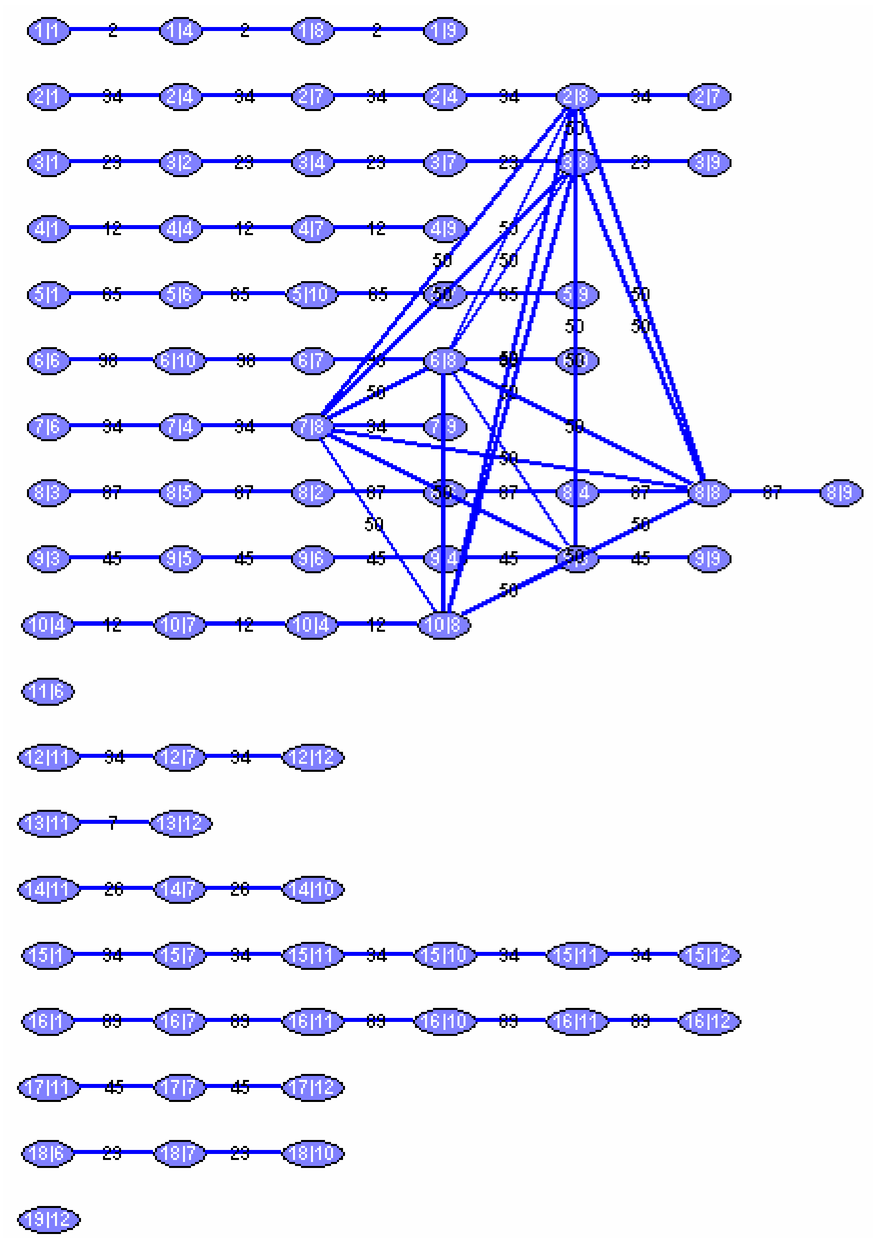


Figure 4.6: Flow network for Vakharia dataset with completely connected graph (clique) for all nodes of machine type 8

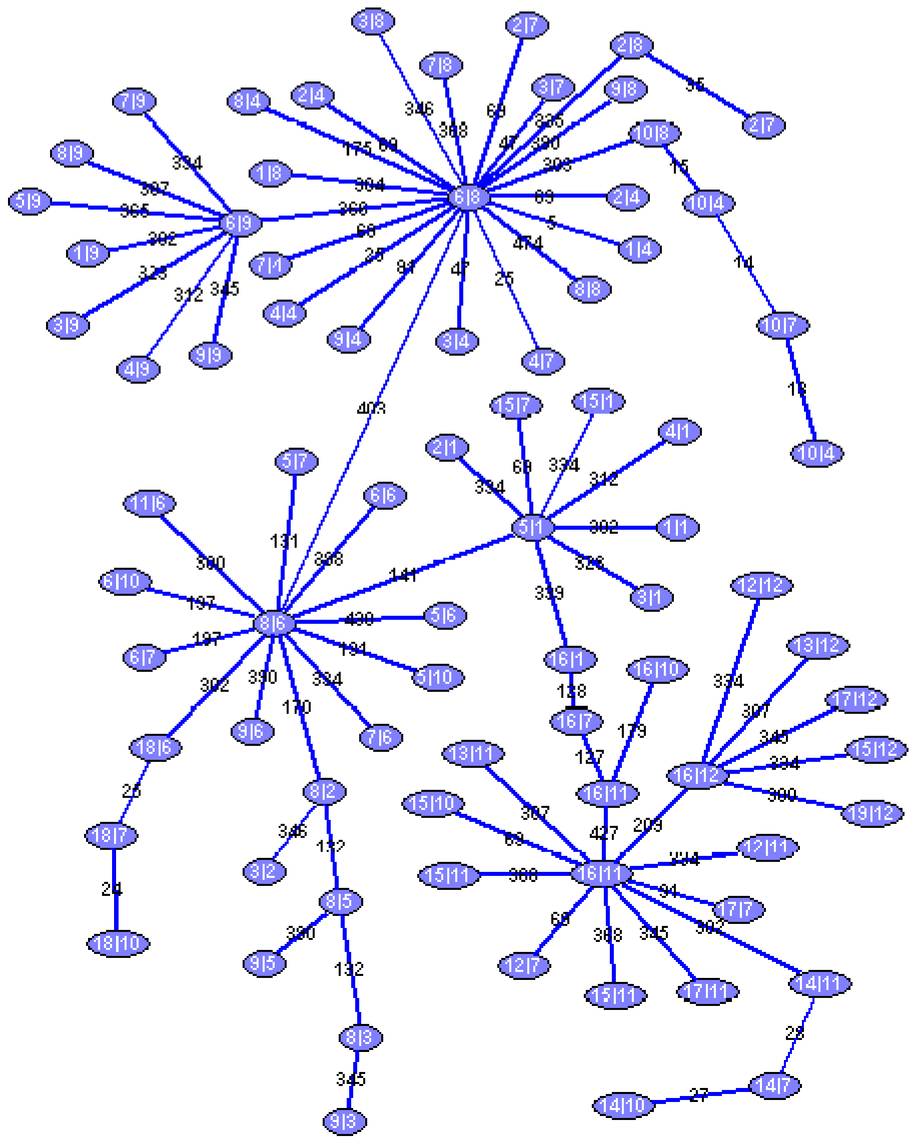


Figure 4.7: A cut tree for Vakharia dataset

The first step in the procedure for constructing the layout modules is to enumerate all arcs in the initial cut tree and sort them in descending order by their weights. As shown in Figure 4.7, there are 79 nodes and 78 arcs in the cut tree for this problem. Table 4.1 shows the list of 78 arcs in the cut tree ordering by their weight values with all the arcs are marked as “Unselected”.

No.	Weight	Selected
1	474	☐
2	464	☐
3	430	☐
4	422	☐
5	398	☐
6	390	☐
7	390	☐
8	390	☐
9	387	☐
⋮	⋮	⋮
76	32	☐
77	22	☐
78	14	☐

Table 4.4: List of arcs – 1st update

From the list in Table 4.1, if there is any arc that connects a pair of nodes of the same machine type, these nodes need to be merged and this arc needs to be removed from the list. The cut tree is then updated. For example, there are several arcs that connect the pairs of nodes of the same machines in the cut tree shown in Figure 4.7. Arc 1, with the largest weight value of 474, connects nodes {8|8} and {6|8} together. These two nodes {8|8} and {6|8} represent machine type 8 used by product 8 and machine type 8 used by product 6, respectively. These two nodes represent the same machine 8; therefore, they can be merged. Figure 4.8 shows the updated cut tree after all the nodes of the same machines have been merged. 37 nodes have been merged and 36 arcs have been removed. Table 4.2 shows the remaining 36 arcs left after this step.

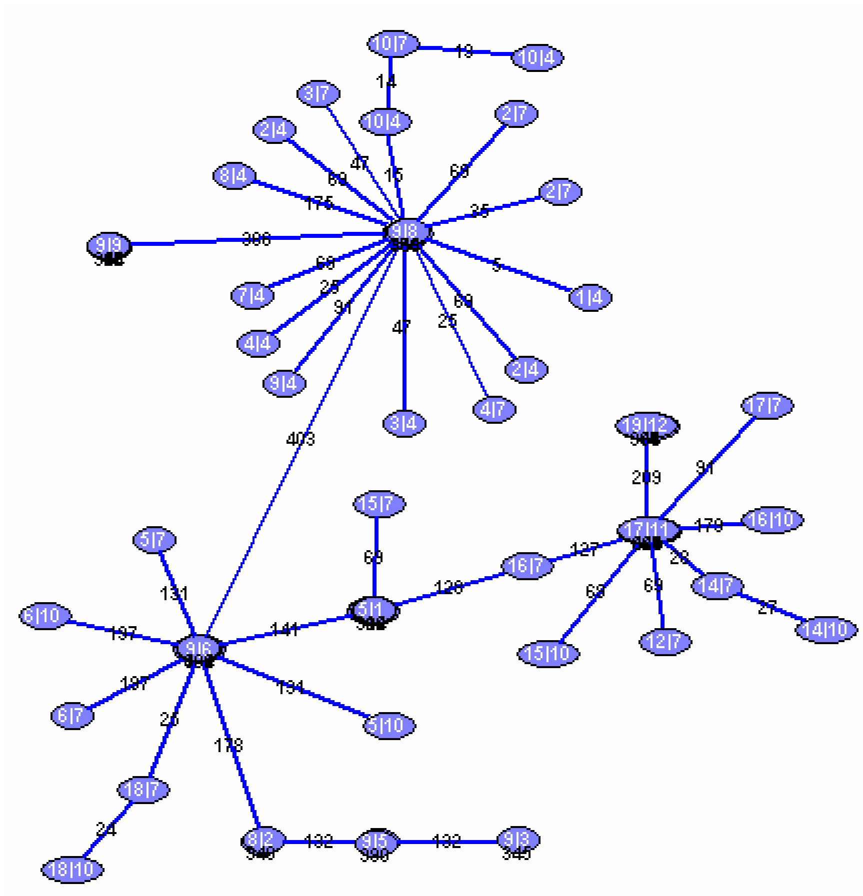


Figure 4.8: The cut tree updated after the nodes corresponding to identical machines are merged

No.	Weight	Selected
1	403	☑
2	366	☐
3	209	☐
4	206	☐
5	206	☐
6	188	☐
7	187	☐
8	184	☐
⋮	⋮	⋮
34	32	☐
35	22	☐
36	14	☐

Table 4.5: List of arcs – 2nd update

The next step is to select the heaviest arc from the remaining set of arcs in the list to continue the merging process. From Table 4.2, arc 1 is the heaviest arc and is chosen and marked as selected. From the cut tree in Figure 4.8, nodes that are connected by the arc 1 are nodes {9|8} and {9|6}. These two nodes need to merge and the arc #1 needs to be removed. It is possible that the two nodes cannot merge if they create a super node that is larger than the module size which is set to 4 for this case. An example of such a case can be where there is a super node that contains 3 machines and another super node that contains 2 machines. When these two super nodes merge, the number of different machines contained in the new super node could be 5. Since the module size is 4, the two nodes cannot be merged.

Since node {9|8} contains machine 8 and node {9|6} contains machine 6, these two nodes can merge into a super node that contains only two different machine types, 8 and 6. After these two nodes have merged, Arc 1 needs to be removed from the list and the cut tree needs to be updated. Table 4.3 shows the updated list. Figure 4.9 shows the updated cut tree after arc 1 is removed and nodes {9|8} and {9|6} have merged in to a super node {9|8, 9|6}. As a result, if there are nodes linked to this new super node, containing either machine 6 or machine 8, all these nodes need to merge into the super node {9|8, 9|6}. Apparently, there is no such node at this stage; therefore, the procedure repeats from the step of selecting the next heaviest "Unselected" arc which is the first arc with the weight of 366 as shown in Table 4.3.

No.	Weight	Selected
1	366	☑
2	209	☐
3	206	☐
4	206	☐
5	188	☐
6	187	☐
7	184	☐
8	178	☐
⋮	⋮	⋮
33	32	☐
34	22	☐
35	14	☐

Table 4.6: List of arcs – 3rd update

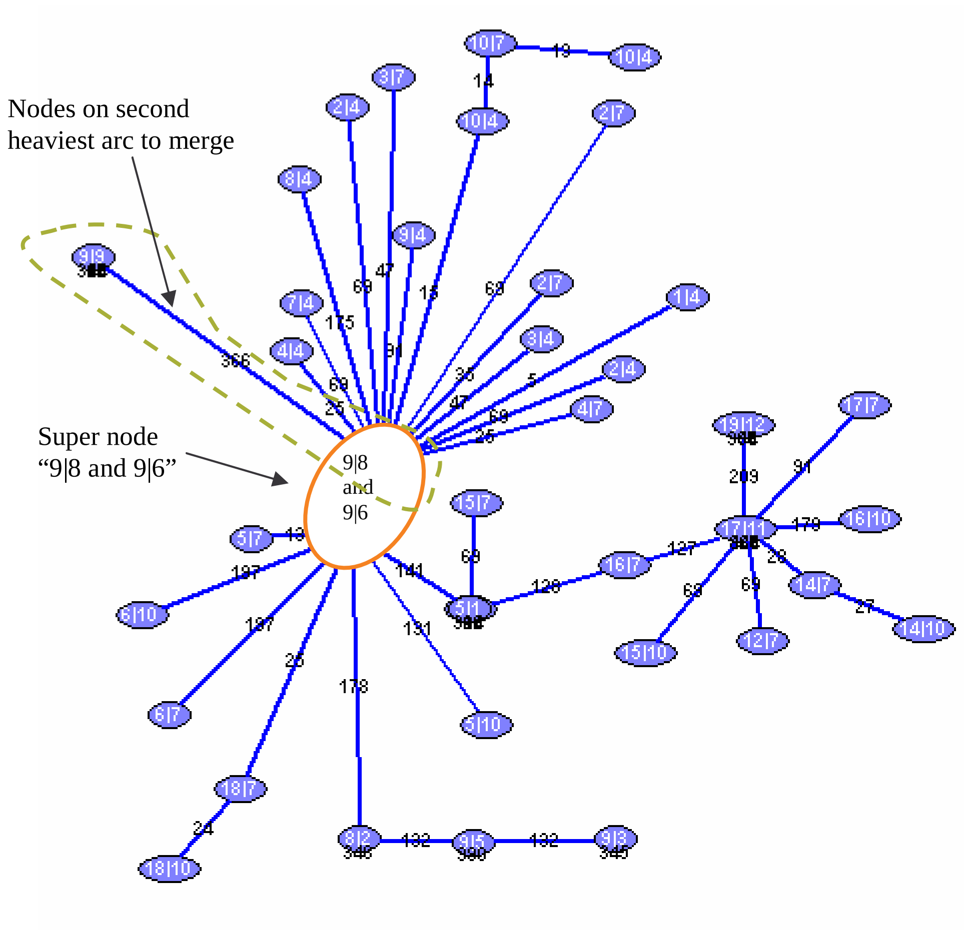


Figure 4.9: The procedure for grouping nodes linked by large arc weights

As can be seen from the updated cut tree in Figure 4.9, the heaviest arc left is the arc with the weight of 366. This arc links node {9|9} and super node {9|6, 9|8} together. These two nodes need to merge. After merging, the final size of the new super node becomes 3 since it contains three different machines—6, 8 and 9. The new super node created is {9|6, 9|8, 9|9}.

The procedure repeats until all the arcs are selected and processed. Some arcs may have been removed and some may still remain in the cut tree. Figure 4.10 shows the updated cut tree at this stage. As can be seen in the figure, there are 5 super nodes created and 11 single nodes left in the current cut tree. At this stage which is the final step of the procedure, all the remaining nodes starting with all the single nodes have to merge if possible. The goal of this step is to reduce the number of machine duplications as much as possible. In Figure 4.10, the single nodes in the cut tree contain either machine 4 or machine 10. Therefore, all the single nodes that contain machine 4 can merge with super node {10|7, 10|4} because this super node already contains machine 4. The same applies for all the single nodes that contain machine 10, i.e. they can merge with super node {17|7, 16|10, 17|11, 19|12}.

After all the single nodes have merged, the next step is to merge the remaining super nodes if they will not create oversize super nodes. From Figure 4.16, the super node {10|7, 10|4} can merge with either the super node {5|1, 15|7} or the super node {17|7, 16|10, 17|11, 19|12}. It does not matter which one to choose during this phase of layout design but it will matter during the implementation phase when workload balancing is considered. In this case, the super node {5|1, 15|7} is selected to merge with the super node {10|7, 10|4} in order to balance the sizes of the different layout modules in the final solution. If no other nodes can merge further, the procedure stops and the final cut tree is layout solution to the modular layout problem. The final cut tree is shown in Figure 4.11.

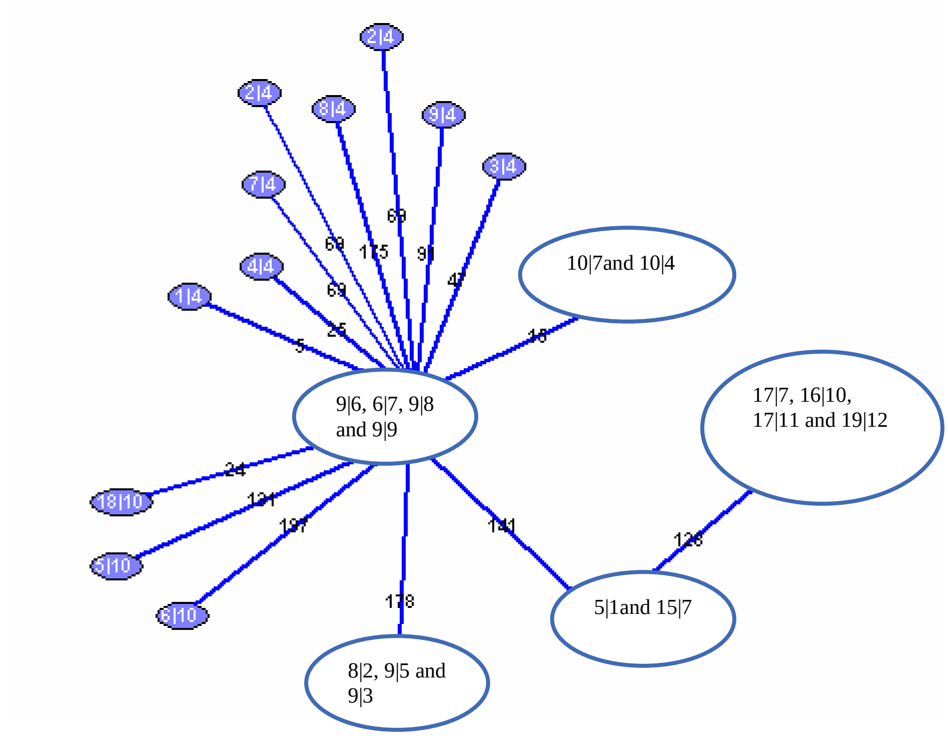


Figure 4.10: The updated cut tree after all arcs have been selected and processed

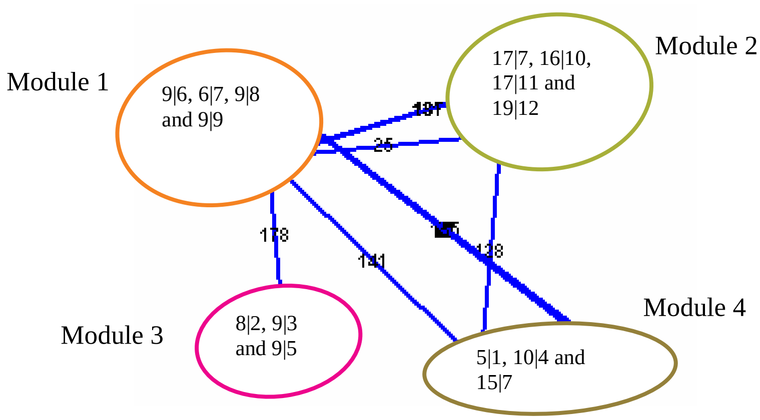


Figure 4.11: Final result – the updated cut tree representing the modular layout

4.4 Results from Heuristic Approach

The heuristic approach described in the previous section was programmed in C++ on Pentium 4, 2.53GHz, 1GB Ram Windows-based computer. It took less than 1 second to solve the modular layout problem of the Vakharia dataset. The result contains 4 modules including [6,7,8,9], [7,10,11,12], [2,3,5], and [1,4,7]. For this solution, there is machine 7 that is required in three modules 1, 2 and 4. After the second phase for allocating the machine capacity, module 1 requires 2.19 ≈ 3 machines, module 2 requires 1.09 ≈ 1 machines, and module 4 requires 0.79 ≈ 1 machine. Therefore, one additional copy of machine 7 needs to be purchased as per this solution. The total production flow for the modular layout from this approach decreases from **2,472** in the original functional layout to **1,198** which is ≈ **52%** reduction compared to the functional layout. Despite the facts that the problem of designing a modular layout is NP-complete, the cut tree heuristic approach can solve the problem and provides results that are acceptable with less computational time.

4.5 Comparison Study

The results from both optimization approach and cut-tree heuristic approach are shown in Table 4.4. There are 6 different datasets compared in this study. The details for these datasets are shown in Appendix B. As can be seen from the performance comparison table, the cut-tree heuristic can perform well acceptable in several cases but for some cases the performance is not impressive compared to the optimization approach.

In the cases that the cut-tree approach cannot duplicate the machines as many as it is supposed to such as Mettler_25x13, Purcheck_28x18 and Sekine_13x12, the performances for these cases are quite poor compared to the other cases. The reason can be that the cut-tree approach has its weakness. Whenever the cut-tree generated from the routing graph is more like a linked chain or spanning tree, reasonable good results should be expected. This is because the process of grouping and merging nodes to form modules can be done efficiently. Whenever, the cut-tree generated has a star structure, meaning that there are too many singleton nodes branching from one node, poor results can be expected. This is because when the middle node has been merged and form a layout module, the remaining singleton nodes cannot be able to form another module. They will have to merge with other layout modules that are already formed. For that reason, the machines are not appropriately duplicated in this case.

Dataset	Total Production Flow			Reduction (%)		CPU Time (s)		No. of M/Cs duplicated	
	FL	MDL (OPT)	MDL (CUT)	OPT	CUT	OPT	CUT	OPT	CUT
Vakharia_19x12	2742	678	1198	75.27	56.30	32	<1	2	1
ABB_50x25	3934	2342*	2370	40.46	39.75	3600*	<1	8	17
Mettler_25x13	4522	1303*	2638	71.18	41.66	3600*	<1	3	0
Purcheck_28x18	3223	1556	2326	51.72	27.83	3494	<1	5	2
Sekine_13x12	1247	308	528	75.30	57.65	9	<1	6	1
Tecomet_42x15	180793	43606	61426	75.88	66.02	3600*	<1	5	1

Remarks: FL = Functional Layout
MDL = Modular Layout
OPT = Optimization Approach
CUT = Cut Tree Heuristic Approach
* = Optimal solution is not obtained. The solution time is set to 3600 seconds only.

Table 4.7: Comparison of both optimization and cut tree heuristic approaches

Although the performances in some cases are not very well acceptable for the cut-tree heuristic approach, the heuristic approach always provide a good upper bound for the optimization approach for the same problem. For example in Vakharia_19x12 case, when the upper bound was set to 6, CPLEX took less than minutes to solve. When it was set to 8, CPLEX could not provide a solution since the computer ran out of memory and CPLEX failed to continue. The layout solution for the same problem from the heuristic approach contains 4 modules. Therefore, we could estimate the number of modules in the final solution to be around 4 and 6 was the upper bound value that we have chosen for the optimization approach for the problem of the Vakharia data set.

4.6 Discussion and Conclusions

Although the two-stage approach helps to simplify the complexity of the problem, the optimization approach by itself cannot handle large-size problems. Therefore, a heuristic approach using the cut tree algorithm for solving a modular layout problem has been developed. This heuristic approach follows the two-stage approach where layout modules are formed first and machines are allocated to each layout module later. In spite of the fact that the problem of designing a modular layout still falls into the class of NP-complete problem, the cut tree heuristic provides acceptable results without significant computational times, compared with the optimization approach.

References

1. Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Company, New York (1979).

2. Gomory, R.E. and Hu, T.C., "*Multi-terminal network flows*," J. SIAM, Vol. 9, No. 4, pp.551-570 (1961).

3. Kandiler, L., "A cell formation algorithm: Hypergraph approximatation – Cut tree," European Journal of Operational Research, Vol. 109, pp. 686-702 (1998).

4. Lee, S. and Chiang, C., "A cut-tree-based approach for clustering machine cells in the bidirectional linear flow layout," International Journal of Production Research, Vol. 39, No.15, pp. 3491-3512 (2001).

5. Montreuil,B. and Ratliff, H.D., "Utilizing cut trees as design skeletons for facility layout," IIE Transactions, Vol. 21, pp. 136-143 (1989).

CHAPTER 5

A NEW APPROACH FOR DESIGNING A HYBRID FLOWSHOP LAYOUT

5.1 Introduction

Flow in manufacturing can be defined as "*the progressive movement of a product/s through a facility from the receiving of raw material/s to the shipping of the finished product/s without stoppages at any point in time due to back flows, an inefficient layout, machine breakdowns, scrap of other production delays*" [Suzaki, 1987]. To ensure the progressive movement, there are three principles for effective flow planning within a facility [Tompskin et al, 1996]:

(1) Minimize flow. Flow can be minimized if the consecutive pairs of operations take place over short travel distances. Short travel distances lead to minimum delay in inter-machine travel. Accordingly, minimum delay in inter machine travel lead to minimal stoppages of flow along its path.

(2) Maximize directed flow. Flow in a facility can be classified as forward flow (in-sequence or bypass), backtracking flow and cross flow. The last two are the least desirable types of flow because they cause interruptions and stoppages for flow in a facility. The interruptions and stoppages can result in high WIP (work in progress), high throughput times, idle machines, and scheduling difficulties [Sarker et al, 1995]. Minimize the least desirable backtracking flow and cross flow and maximize the directed flow can ensure the progressive movement.

(3) Minimize cost of flow. Cost of material handling can be minimized by conducting the previous two principles and eliminating manual handling by mechanizing or automating flow.

From manufacturing layout perspective, these three principles can be observed in product flowline layouts that are employed typically in low-mix high-volume facilities such as the manufacturing facilities in automotive industry. In high-mix low-volume (HMLV) environments where typically functional layouts are employed, these three principles can hardly be observed. In order to obtain efficient flow in HMLV facilities, approaches such as the conversion of functional layout to flowline layout or the employment of manufacturing cell with a flowline layout have been suggested. The latter approach was a traditional approach for most HMLV facilities. When employing a manufacturing cell, products with similar operation sequences are grouped into a product family. Machines are grouped into a manufacturing cell which is assigned dedicatedly to produce a particular product family. If machines in a manufacturing cell can be arranged in such a way that a sequential production can be performed and material flows are in unidirectional movements, a product-based flowline layout can be constructed as a layout for this manufacturing cell. When a flowline layout is implemented, parts can flow efficiently, unidirectional from one machine to another with minimum distances and minimum handling costs. Thus, flow is considered efficient as “directed and minimized” in this case.

Obtaining efficient flows by implementing product-based flowline layouts as manufacturing cells in HMLV facilities could yield efficient flows and bring in advantages. However, there are many limitations as follows [Stockton, 1994]:

- Product families are difficult to identify and manufacturing cells can hardly form.
- Production volumes are not sufficient to accommodate efficient flows in flowline-layout manufacturing cells.
- Manufacturing cells cannot afford interruptions from outside products because of machine sharing. This can result in inefficient operation and underutilized machines in the cells.
- Flowline-layout manufacturing cells provide insufficient flexibility and less reliability when changes in product mix and production demands as well as machine failures occur.

Consequently, adopting flowline layouts as manufacturing cells to exploit the benefits of having efficient flow in HMLV facilities is limited. Therefore, a hybrid flowshop layout is introduced as an alternative solution. In the literature, a hybrid flow shop is defined as a manufacturing facility that consists of series of production stages where each stage contains single or multiple machines of same or different types and products flows through the shop in one direction [Artiba and Elmaghraby, 1997]. Process industries such as cosmetics, pharmaceuticals, textiles and food industries are the example application of hybrid flow shops.

In our research, a hybrid flowshop layout is defined as a layout that allocates machines into several groups (called stages) and these groups are arranged in a line. Unlike a traditional manufacturing cell, each group of machines in this layout does not process a family of products. Only portions of one or more consecutive operations occurring in the routings of products are performed. All flows are forward (either in-sequence or bypass) and backtracking flows are minimized or diminished if possible in this layout. Therefore, the progressive movements of flow process production are promoted without restrictions to the existence of product families and manufacturing cells.

This chapter is organized as follows. Literature view on transforming functional layout to flowline layout is presented. The problem model of designing a hybrid flowshop layout is constructed. The complexity of this problem that falls into NP-complete category is proven. CPLEX is used to demonstrate the solution approach for solving a modular layout problem optimally. Then a heuristic approach using Ratio Cut algorithm is presented. The demonstration of solution approach using this heuristic is presented. The performance comparison between these two approaches is shown and the results are discussed and concluded.

5.2 Literature Review

The flow structure of a production process used in a manufacturing facility to make or produce a product can be classified into two categories as either a job shop production or flow shop production (also known as flow process production). In a job shop production, flow is not organized and is complex but it can be very flexible. A flow shop production is opposite. Flow in this production type is not flexible since it has a fixed path but it can be much efficient than flow in a job shop production. Because of its efficient flow, a flow shop production has several advantages such as (a) less material handling cost, (b) minimal backtracking flow, (c) ease of control and scheduling, and (d) applicable demand-pull production control [Framinan, 2005]. With these advantages especially being able to adopt a demand-pull production control, an important element of Lean Manufacturing, the employment of the flow production in jobshops has shown to improve the performance of jobshop manufacturing companies [Li and Barnes, 2000].

Most jobshop manufacturing companies tend to adopt a functional layout for their facilities because flexibility is the most important for their companies to survive in volatile and chaotic manufacturing environments. In contrast, most flowshop companies exploit a flowline layout for their facilities because this layout fits well and it provides the most efficient flow among other layouts. One approach to adopt the flow shop production to a jobshop is to convert a jobshop layout (which is typically a functional layout) to a flowshop layout (which is basically a flowline layout) [Framinan and Ruiz-Usano, 2002].

The problem of transforming a jobshop to flowshop (TJF) can be categorized in two criteria: (1) a single-machine flowline or multiple-machine flowline (flowshop) and

(2) backtrackings minimized or no backtrackings allowed. For the TJF problem with a single machine flowline and no backtracking allowed, this problem was modeled as a linear integer programming with the objective function to minimize the cost of machine duplication and to obtain the minimal length of flowline [Framinan and Ruiz-Usano, 2002]. This problem was solved optimally using a branch and bound technique with the largest size of problem instances of products ≤ 5 and machines ≤ 4. This problem is also considered as a special case of a well-known classical Shortest Common Supersequence (SCS) stringology problem. Since the SCS problem belongs to the class of NP-hard problem [Raiha and Ukkonen, 1981], solving this problem optimally is not practical. Several heuristic approaches to solve this SCS problem were developed [Askin and Zhou, 1997], [Framinan, 2005].

The results of TJF problem with a single-machine flowline and no backtracking allowed can contain a substantial number of machines duplicated. Such results may not be applicable for jobshops especially under their constrained budgets. Therefore, the different version of a TJF problem with a single machine flowline remains the same, backtracking flows are allowed but minimized, and no machine duplication allowed has been suggested. This problem was modeled as a quadratic assignment problem (QAP) with the objective function to minimize the backtrackings as well as total traveling distances in a flowline while using only existing machines available [Sarker et al., 1995] Note that QAP belongs to the class of NP-complete problem, several heuristic approaches were developed to tackle this problem as well [Kouvelis and Chieng, 1992], [Sarker et al, 1995], [Ho and Moodie, 1998]. A more generalized problem model with machine duplication allowed under budget constraint was also proposed. This generalized problem model has the objective function of minimizing the backtracking flows as well as machine duplication costs in a single machine flowline layout. It was modeled as a QAP and solved optimally for small-size problem instances. The heuristic approach of two sequential phases: forming SCS and compressing the SCS, was developed to tackle the large-size instances of this problem [Ouriarat, 2000].

The last version of TJF problem which is the main problem of this chapter is the problem with a multiple-machine flowline and minimum backtrackings. This problem is more generalized and more applicable to jobshops in a sense that a production line is shorter because each spot in the line is not an individual machine. Instead, each spot consists of machines either the same types or different types. In addition, the numbers of machines duplicated can be less because backtrackings are allowed but minimized.

There has been only a few research conducting for the above problem. The most relevant research is an approach to cluster machine cells in the bidirectional linear flow layout [Lee and Chiang, 2001]. This research introduced a cut-tree-based heuristic approach to form manufacturing cells and simultaneously arrange these cells into a single line. The objective is to minimize inter-cell and intra-cell flows among these cells. However, machine duplication is not allowed is this approach. Another research which has a similar concept but also allows machine duplication in a layout solution was introduced in 1994. [Stockton and Lindley, 1994] This approach creates a layout called a process sequence cell layout (PSCL). The PSCL is composed of individual manufacturing cells where each cell represents an individual stage of operations in product routings. The method to design the PSCL uses a simple sorting technique and a left-right justification technique that are available in most commercial spreadsheets. The design objective of PSCL is to have all production flows moving in one direction with machines are allowed to be duplicated without any restriction. In the PSCL, each operation in a product routing is assigned to a cell. A machine or equipment required for each operation in each cell is then allocated. It has shown that when implementing the PSCL in conjunction with a Kanban scheduling control, it helps the performance of manufacturing companies operating in batch processing environments [Stockton and Lindley, 1998].

In our research, the TJF problem was redeveloped and generalized. Our version is to generate production stages where each stage may contain multiple machines of same or different types. The objective of our TJF problem is to minimize the machine purchasing cost (duplication cost), the production cost, and the penalty cost of having backtracking flows where each stage has a limited size and machine capacity constraints have to be satisfied. The detailed description for our version of TJF problem is described next.

5.3 Problem Formulation

The motivation of designing a hybrid flowshop layout is to convert traditional functional layouts in HMLV facilities to flowline-like layouts. As mentioned, the current approaches of converting a functional layout to a flowline layout resulting in a large number of machines duplicated. On the other hand, without a number of machines duplicated, it is difficult to obtain smooth, unidirectional flows in HMLV. Thus, a hybrid flowshop layout, also known as a flexible flowline layout, is considered in this research to cope with the limitations of adopting a single flowline layout in HMLV facilities.

A hybrid flowshop layout is a flowline layout where each spot, called a stage, in a production line may contain multiple machines/operations of same or different types. This layout appears to be more flexible than a single flowline layout while maintain its characteristic of flow process production. Therefore, the problem of designing a hybrid flowshop layout is to design a layout that contains the groups of machines (stages) that can absorb the majority of complex, bidirectional material flows while promoting unidirectional material flows in a facility with minimal machine duplication. The resulting layout of this problem therefore exhibits the characteristics of a flexible flowline layout and implicates that the advantages of flow process production can possibly be obtained from manufacturing facilities operating in HMLV environments. The description of the problem is as follows:

Given

- Production volume of each product in the planning period
- Operation sequences for each product, in form of routing of machines, and processing time for each operation
- The production quantity of each product type in the planning period is known
- The transfer batch size of each product type is known
- The rates of the setup cost and processing cost of each product type at each machine type in the planning period are known

- The loading/unloading cost of each product type at each machine is ignorable
- The amortized purchasing price per unit of each machine type in the planning period is known
- Current number of each type of machine
- Balancing factors for machine duplication cost, production cost, and the penalty cost of backtracking flows

Determine

- Number of stages, denoted by m
- To which stage each machine is allocated
- To which machine each operation for each product is assigned

To Minimize

- Machines purchasing cost
- Production cost
- Backtracking flows

Such that

- Each operation is assigned to one and only one machine
- Machine capacity constraints are satisfied

It can be noticed that the problem of designing a hybrid flowshop layout is similar to a modular layout problem with the distinction of the minimization of backtracking flows introduced and the material handling costs omitted in this hybrid flowshop layout problem. The minimization of backtrackings promotes unidirectional material flows in a hybrid flowshop layout. The absence of material handling cost in the problem description does not imply that there is no cost of material handling in this layout. There is still such cost; however, the focus is more of the direction of material flows than the material handling cost in the layout. Another reason is that there is no evidence to support that either having a large volume of intra-stage flows (flows among machines within a stage) and minimizing inter-stage flows (flows among machines between stages) which of these would provide a better performance to a hybrid flowshop layout. Therefore, the cost of material handling is omitted at this present and left for the future study to conduct further research on this issue.

Intuitively, a hybrid flowshop layout can be analogous to a sequence of layout modules where flows from upstream modules to downstream modules are only encouraged. Hence, based on the mathematical model of a modular layout problem in the previous chapter, the following formulation of a mathematical model for a hybrid flowshop layout problem can be constructed as follows:

Minimize $\mu_1 \cdot \sum_{k=1}^{K} E_k \cdot \left(\sum_{m=1}^{MAX} r_{km} - L_k \right)$ (1)

$$+\mu_2 \cdot \left(\sum_{i=1}^{T}\sum_{j=1}^{N_i}\sum_{k=1}^{K}\sum_{m=1}^{M} \left\lceil \frac{Q_i}{B_i} \right\rceil \cdot S_{ijk} \cdot x_{ijkm} \right.$$

$$\left. + \sum_{i=1}^{T}\sum_{j=1}^{N_i}\sum_{k=1}^{K}\sum_{m=1}^{M} Q_i \cdot U_{ijk} \cdot P_{ijk} \cdot x_{ijkm} \right)$$

$$+\mu_3 \cdot \sum_{i=1}^{T}\sum_{j=1}^{N_i-1}\sum_{k=1}^{K}\sum_{\substack{l=1 \\ l\neq k}}^{K}\sum_{m=2}^{MAX}\sum_{n=1}^{m-1} HBC_{ikl} \cdot \left\lceil \frac{Q_i}{B_i} \right\rceil . x_{ijkm} . x_{i(j+1)ln}$$

Subject to: $\sum_{k=1}^{K}\sum_{m=1}^{MAX} x_{ijkm} = 1$, for each (i, j) (2)

$$\frac{\sum_{i=1}^{T}\sum_{j=1}^{N_i} Q_i \cdot P_{ijk} \cdot x_{ijkm}}{F_k} \le r_{km}, \text{ for each } (k, m) \quad (3)$$

$$\sum_{k=1}^{K} r_{km} \ge 1, \text{ for each } k \quad (4)$$

$$\sum_{m=1}^{M} r_{km} \le MaxL_k, \text{ for each } k \quad (5)$$

$$r_{km} \le BN \,.\, w_{km}, \text{ for each } (k, m) \quad (6)$$

$$\sum_{k=1}^{K} w_{km} \le SIZE, \text{ for each } m \quad (7)$$

$r_{km} \ge 0$ and integral, for each (k, m) (8)

x_{ijkm} binary, for each (i, j, k, m) (9)

w_{km} binary, for each (k, m) (10)

$\lceil z \rceil$ denotes the smallest integer that is larger than or equal to z

Variables:

r_{km} = Number of units of machine type k assigned to stage m.

x_{ijkm} = 1 if the j^{th} operation of product i is processed at machine type k in stage m;

= 0 otherwise.

w_{km} = 1 if machine type k is in stage m;

= 0 otherwise.

Parameters:

μ_1, μ_2, μ_3 = Cost balancing factor

B_i = Transfer batch size of product type i

BN = Any big number

E_k = Annualized cost of purchasing a unit of machine type k

F_k = Annual production time available per unit of machine type k

HBC_{ikl} = Material handling cost for transferring a *batch* of product type i from machine type k to machine type l

K = Number of machine types

L_k = Existing number of units of machine type k

$MaxL_k$ = Maximum number of units of machine type k allowed

M = Maximum number of stages in the layout

N_i = Number of operations in the routing of product type i

P_{ijk} = Processing time of the j^{th} operation of product type i at machine type k

Q_i = Annual production quantity of product type i

S_{ijk} = Setup cost for the j^{th} operation of product type i at machine type k

$SIZE$ = Maximum number of machine types in one stage

T = Number of product types

The objective function (1) minimizes the sum of prorated machine purchasing cost, production cost and the penalty cost of having backtracking flows. The machine purchasing cost in the first part of (1) gives the amortized cost of purchasing extra machines in the planning period. The second part of (1) calculates the production cost that consists of setup cost and processing cost. The processing cost is the cost of producing all units of products at their designated machines. The setup cost is the cost of setups required by batches of products at their assigned machines. The third part of (1) penalizes the problem model if backtracking flows occur in the layout solutions. The balancing factors μ is given to make all three costs in (1) comparable.

Constraint set (2) ensures that each operation in the routing of each product is performed at one and only one machine. Constraint set (3) guarantees enough capacity for operations performed on each machine type in each stage. This set implies that the integer allocation of machines of the same type assigned to one or more stages is constrained by the total number of machines of that type i.e. acquisition of extra machines incurs a capital expense. Constraint set (4) ensures that at least one machine of each type is assigned to a stage. Constraint set (5) ensures that the number of machines for each type if not larger than it is allowed. Constraint sets (6) and (7) ensure that the number of machine types in a stage will not exceed the maximum size of this stage. The model counts machine types instead of the number of copies of machines because there will be no movement of parts between the same machines of the same type. Therefore, the multiple copies of the same machine type are grouped together into one unit of the machine type in a stage. Constraint sets (8), (9) and (10) ensure nonnegativity, binary and integer requirements for the decision variables.

In summary, the mathematic model for solving a hybrid flowshop layout problem does (1) partitioning and duplicating, if necessary, machines and placing them into several stages, (2) arranging these stages in a sequence, and (3) assigning each operation in product routings to a specific machine in a specific stage while each machine must not be over utilized. The objective function is to minimize the production cost, the machine purchasing cost, and the total volume of production flows in backward direction.

5.4 Problem Complexity

The problem of designing a hybrid flowshop layout can be reducible to the same special case problem that was used to prove the complexity of a modular layout problem. In a modular layout problem, its special case problem is the problem of selecting and assigning products to machines in pre-determined modules. Instead of modules, we can change them to stages. Therefore, when using the same procedure of proving the complexity of a modular layout problem using the 3-partition problem, we can conclude that a hybrid flowshop layout problem also falls into the class of NP-complete. □

Since a hybrid flowshop layout problem falls into the class of NP-hard problem, it can be helpful to if this problem can be simplified further in any forms without scarifying its correctness. The same simplifications for a modular layout problem are applied to this hybrid flowshop layout problem as described next.

5.5 Problem Model Simplification

In a modular layout problem, the production cost that consists of the setup cost and processing cost can be ignored if S_{ijk} and P_{ijk} are independent from the different arrangement and assignment of machines in different modules. Similarly, in a hybrid flowshop layout, the different arrangements and assignments of machines to different stages do not reflect to changes in the setup time and processing time for each operation in the product routings. Therefore, we will omit the production cost from the objective function in a hybrid flowshop layout problem assuming that S_{ijk} and P_{ijk} are the same for any k^{th} stage.

There is also another consideration for simplifying the problem. When processing time P_{ijk} is approximated similarly to what it was done in a modular layout problem, machine duplication can be done independently apart from the main solution approach without losing the correctness of the layout solutions. Given that the processing time is approximated by the total number of machines available, the solution approach for solving a hybrid flowshop layout can be modified and solved in two stages. In the first stage, the problem will be solved without the capacity constraints. When the first stage is solved, the detail processing time and machine capacity will be used in the second stage for machine allocation process. With the two considerations for omitting the production cost and deploying the two-stage approach, the problem model can be simplified and linearized as follow:

Minimize $\sum_{k=1}^{K} E_k \cdot \left(\sum_{m=1}^{MAX} r_{km} - 1 \right)$ (1)

$$+ \mu \cdot \sum_{i=1}^{T} \sum_{j=1}^{N_i-1} \sum_{k=1}^{K} \sum_{\substack{l=1 \\ l \neq k}}^{K} \sum_{m=2}^{MAX} \sum_{n=1}^{m-1} HBC_{ikl} \cdot \left\lceil \frac{Q_i}{B_i} \right\rceil . y_{ij(j+1)klmn}$$

Subject to: $\sum_{k=1}^{K} \sum_{m=1}^{MAX} x_{ijkm} = 1$, for each (i, j) (2)

$\sum_{k=1}^{K} r_{km} \leq SIZE$, for each m (3)

$\sum_{m=1}^{M} r_{km} \leq MaxL_k - L_k + 1$, for each k (4)

$\sum_{i=1}^{T} \sum_{j=1}^{N_i} x_{ijkm} \leq BN \, . \, r_{km}$, for each (k, m) (5)

$y_{ij(j+1)klmn} \leq x_{ijkm}$, for each (i, k, l, m, n) and j=1, 2, …, N_i-1 (6)

$y_{ij(j+1)klmn} \leq x_{i(j+1)ln}$, for each (i, k, l, m, n) and j=1, 2, …, N_i-1 (7)

$y_{ij(j+1)klmn} \geq x_{ijkm} + x_{i(j+1)ln} - 1$, for each ($i$, k, l, m, n)

and j=1, 2, …, N_i-1 (8)

r_{km} binary, for each (k, m) (9)

x_{ijkm} binary, for each (i, j, k, m) (10)

w_{km} binary, for each (k, m) (9)

$\lceil z \rceil$ denotes the smallest integer that is larger than or equal to z

Variables:

r_{km} = Number of units of machine type k assigned to stage m.

x_{ijkm} = 1 if the j^{th} operation of product i is processed at machine type k in stage m;

= 0 otherwise.

$y_{ij(j+1)klmn}$ = 1 if the j^{th} operation of product i is processed at machine type k in stage m and $(j+1)^{th}$ operation of the same product i is processed at machine type l in stage n

= 0 otherwise.

This simplified mathematical model for solving a hybrid flowshop layout problem is similar to the problem model of a single machine flowline problem proposed by Framinan and Ruiz-Uzano in 2002. The differences are that, for the model of a single flowline problem, (1) the objective function is solely to minimize the number of machines duplicated, (2) backtracking flows are completely not allowed in a flowshop, and (3) each stage or shop can contain only one machine. In our modified problem model of a hybrid flowshop layout, machines are weighted with their purchasing costs, backtracking flows with production volumes associated are allowed to some extent, and each stage (shop) is allowed to contain more than one machine of the same type or different types. This problem model can be analogous to a capacitated partitioning problem. In addition, without a linearization applied, it can be seen that the problem model is a quadratic assignment problem which belongs to the class of NP-complete problem. [Garey and Johnson, 1979] It is clearly shown that a hybrid flowshop layout problem can not be solved optimally for real-world problem instances.

For a single flowline problem, the largest size of problem instances that was reportedly solved with an optimization technique is the problem of 5 products and 4 machines. Accordingly, solving a hybrid flowshop layout problem optimally can only be done for small-size problems. CPLEX is again used with the Vakharia dataset that was used in the previous chapter to demonstrate the optimization approach for solving a hybrid flowshop layout problem. A heuristic solution approach to solve large-size problems will be discussed later in this chapter.

5.6 Optimization Solution Approach Using CPLEX

To illustrate the procedure of formulating and solving a hybrid flowshop layout problem, the Vakharia dataset containing 19 parts and 12 machines is used to setup the problem. For reference and comparison purposes, the problem instance for a hybrid flowshop layout from this dataset will be constructed as comparable as the problem instance of a single flowline problem. Therefore, our focus will be for a layout solution that requires the minimum number of machines duplicated while backtracking flows are completely disallowed. Again, in a hybrid flowshop layout problem, each stage can contain more than one machine so that machines will be partitioned and placed in such a way that the directions of flows among the machines inside each stage are not restricted but among stages (groups of machines), flows are unidirectional.

The routings of this dataset are shown in Table 5.1. The production quantity of each product is the demand per week so each value was multiplied by 50 weeks to obtain a production figure for one year.

Product	Quantity	Routing	# of Operations (N_i)
1	2	1→4→8→9	4
2	34	1→4→7→4→8→7	6
3	23	1→2→4→7→8→9	6
4	12	1→4→7→9	4
5	65	1→6→10→7→9	5
6	98	6→10→7→8→9	5
7	34	6→4→8→9	4
8	87	3→5→2→6→4→8→9	7
9	45	3→5→6→4→8→9	6
10	12	4→7→4→8	4
11	67	6	1
12	34	11→7→12	3
13	7	11→12	2
14	26	11→7→10	3
15	34	1→7→11→10→11→12	6
16	89	1→7→11→10→11→12	6
17	45	11→7→12	3
18	23	6→7→10	3
19	23	12	1

Table 5.1: Product routings from Vakharia dataset

Table 5.2 shows the available number of machines and their purchase costs. Since this problem instance is focused solely for minimizing the total number of machines duplicated, cost for purchasing a machine is assigned as to be the same for every machine. The annual production time available for each machine and processing time for each operation of each product that is missing for the original dataset will be approximated by the same approximation technique used in a modular layout problem. The remaining parameters of the problem are shown in Table 5.3.

Machine Type	**Quantity (L_k)**	**Cost (E_k)**	**Avail. Time (F_k)**
1	2	1	2,000
2	1	1	2,000
3	1	1	2,000
4	2	1	2,000
5	1	1	2,000
6	2	1	2,000
7	4	1	2,000
8	1	1	2,000
9	2	1	2,000
10	3	1	2,000
11	3	1	2,000
12	1	1	2,000

Table 5.2: The number of machines available and their purchase costs

μ	B_i	HBC_{ikl}	K	MAX	$SIZE$	T
100	1	1	12	5	4	19

Table 5.3: Problem parameters

Most parameters are set to 1 in this experimental study for the sake of simplicity. The balancing factor μ is set to a large number meaning that backtracking flows will be too costly to be allowed in the layout solutions. The maximum number of machine type (*SIZE*) for each stage is set to 4. That means that each stage can contain at most 4 different machine types. However, each machine type may contain more than one machine. Similar to a modular layout problem, we use the machine type instead of the number of machines because of the assumption that there is no material movement between machines of the same type. So in this problem, machines of the same type located in the same stage are grouped as a single unit machine of that type.

The maximum number of machine types in each stage (referred as a stage size) can vary depending on several conditions. If the stage size is too large, layout solutions may contain a very few stages. With a less number of stages, unidirectional flows are not fully promoted since much complex, bidirectional flows are absorbed within each big stage. It implies that we are not healing the problem of having complex flows in a facility. Instead, we are creating several subproblems from the main problem and making the problem even more difficult to handle. In contrast, if the stage size is set too small, a large number of machines duplicated can occur and the layout solution can converge closely to a single flowline layout. Therefore, exercises may be needed to ensure that an appropriate stage size is assigned.

Another important parameter is the maximum number of stages (*MAX*). This parameter plays a significant role the same way it does in a modular layout problem. The size of an instance of a hybrid flowshop layout problem can be defined by the number of decision variables that are r_{il} and $y_{ij(j+1)klmn}$ in the problem model. Supposedly, the problem instance contains m machine types, m operations per routing, and n routings total. The upper bound for the number of stages can be approximated as the product of the number of machine types and the number of total jobs ($m \times n$) for the worst case scenario according to Framinan and Ruiz-Usano [2002]. Therefore, the number of variables in this problem instance can be equal to $(n \times m \times n) + (n \times m \times m \times (m \times n) \times m \times (m \times n) \times m) \approx O(m^6n^3)$. It can be noticed that the number of variables grows in an order of m^3n^2. As the solution space grows in an order of $2^{O(m^6n^3)}$, if a given bound is too loose, then the problem size that grows exponentially can be too large and can not be solved eventually. However, if a bound given is too tight, then a solution may fall into a local optimum. In a hybrid flowshop problem, each stage (shop) can contain more than one machine. So using $m \times n$ as an upper bound can be too loose. An approximate upper bound given by solutions obtained by the heuristic approach can be used as a guideline. As for this problem instance, the stage size is 4 according to the result obtained from the proposed heuristic approach which will be described later in this chapter.

5.7 Results from Optimization Approach

The problem instance from Vakharia dataset was solved by CPLEX solver on Pentium 4, 2.53GHz, 1GB Ram windows-based machine. The computation time for solving this instance problem of 19 products and 12 machines with 5 maximum stages is 32 seconds. The resulting flowshop layout for this problem is shown in Figure 5.1.

	Machine											
Product	1	2	3	4	5	6	7	8	9	10	11	12
1	0.02			0.01				0.01	0.01			
2	0.26			0.46			0.51	0.10				
3	0.18	0.21		0.16			0.17	0.07	0.13			
4	0.09			0.08			0.09		0.07			
5	0.50					0.31	0.49		0.36	0.58		
6						0.47	0.74	0.29	0.54	0.88		
7				0.23		0.16		0.10	0.19			
8		0.79	0.66	0.59	0.66	0.42		0.26	0.48			
9			0.34	0.31	0.34	0.21		0.13	0.25			
10				0.16			0.09	0.04				
11						0.32						
12							0.26				0.28	0.15
13											0.06	0.03
14							0.20			0.23	0.22	
15	0.26						0.26			0.30	0.57	0.15
16	0.69						0.67			0.80	1.49	0.38
17							0.34				0.38	0.19
18						0.11	0.17			0.21		
19												0.10
Stage 1	2.00	1.00	1.00		1.00							
Stage 2						2.00	3.13			3.00	3.00	
Stage 3				2.00			0.87	1.00				
Stage 4									2.00			1.00
M/Cs Required	2	1	1	2	1	2	5	1	2	3	3	1
M/Cs Available	2	1	1	2	1	2	4	1	2	3	3	1
Extra M/Cs							1					

Figure 5.1: Results from the optimization approach for Vakharia dataset

The layout solution contains 4 stages including {1,2,3,5}, {6,7,10,11}, {4,7,8}, and {9,12} with only machine 7 needed to be placed in two different stages. The update routings are shown in Figure 5.2. As can be seen in the table, flows are strictly forward from one stage to another stage. Some flows are in-sequenced and some are bypass. For example, product 8, its routings in the flowshop layuot is 1➔2➔3➔4 from stage 1 through 4 without skipping. For product 1, its routing in the flowshop layout is 1➔3➔4 from stage 1 through 4 and skipping stage 2. So with only one machine 7 duplicated, the existing functional layout can be transformed to a flowshop layout without backtracking flows occured.

Product	Orignal Routing						
1	1	4	8	9			
2	1	4	7	4	8	7	
3	1	2	4	7	8	9	
4	1	4	7	9			
5	1	6	10	7	9		
6	6	10	7	8	9		
7	6	4	8	9			
8	3	5	2	6	4	8	9
9	3	5	6	4	8	9	
10	4	7	4	8			
11	6						
12	11	7	12				
13	11	12					
14	11	7	10				
15	1	7	11	10	11	12	
16	1	7	11	10	11	12	
17	11	7	12				
18	6	7	10				
19	12						

Hybrid Flowshop Routing

Product / Stage	1			2				3					4
1	1							4	8				9
2	1							4	7	4	8	7	
3	1	2						4	7	8			9
4	1							4	7				9
5	1			6	10	7							9
6				6	10	7		8					9
7				6				4	8				9
8	3	5	2	6				4	8				9
9	3	5		6				4	8				9
10								4	7	4	8		
11				6									
12				11	7								12
13				11									12
14				11	7	10							
15	1			7	11	10	11						12
16	1			7	11	10	11						12
17				11	7								12
18				6	7	10							
19													12

Figure 5.2: Updated routings with flowshop replacements for Vakharia dataset

Figure 5.3 shows the flow diagram of the original functional layout for this Vakharia data set. Figure 5.4 shows the flow diagram for the hybrid flowshop layout for this data set and Figure 5.5 shows the same layout where only flows between stages are shown. As can be seen from the flow diagrams for the functional layout and the hybrid flowshop layout, the complexity can be reduced with vastly when transforming from the functional layout to the hybrid flowshop layout.

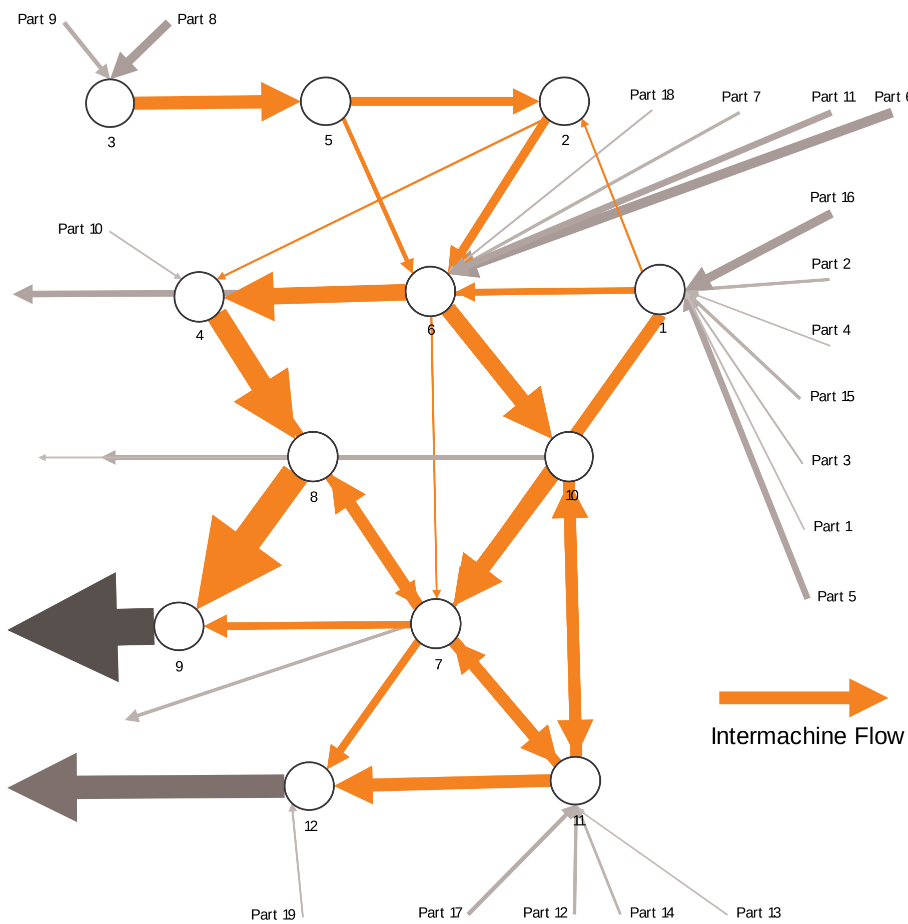


Figure 5.3: Material flow network of the functional layout for Vakahria dataset

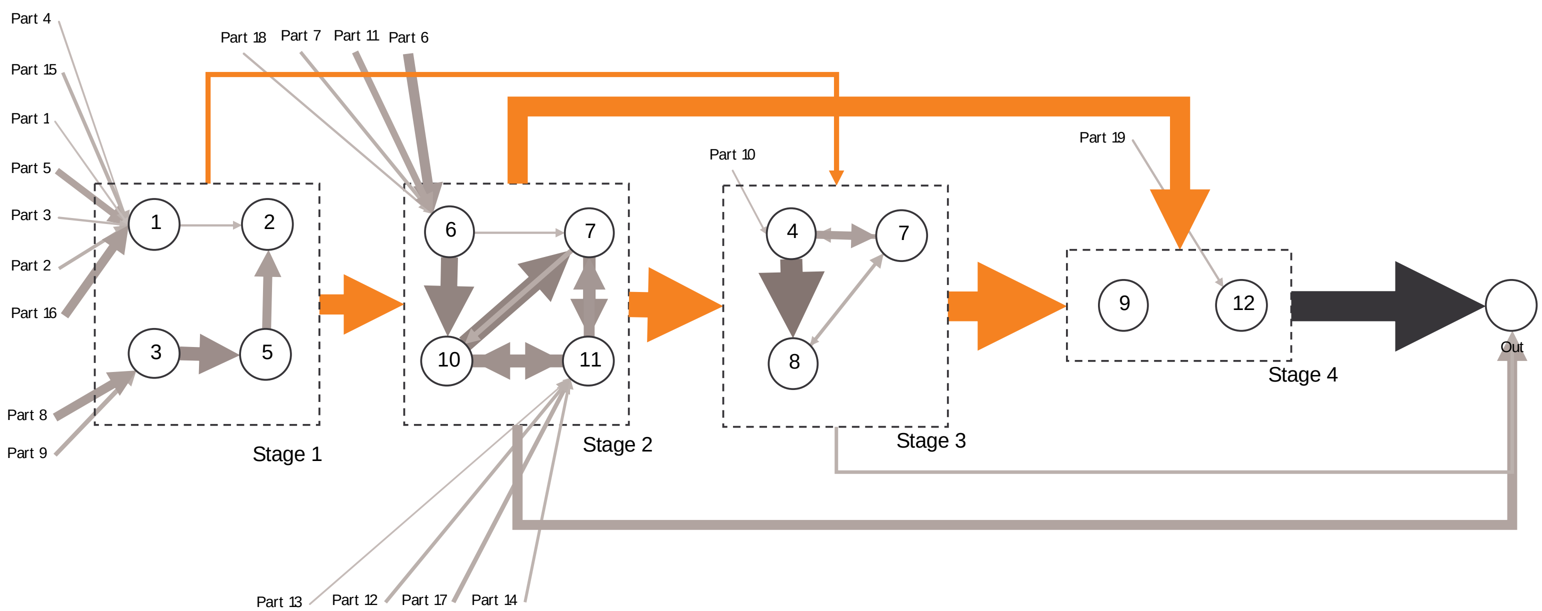


Figure 5.4: Flowshop Hybrid Layout – All flows are shown

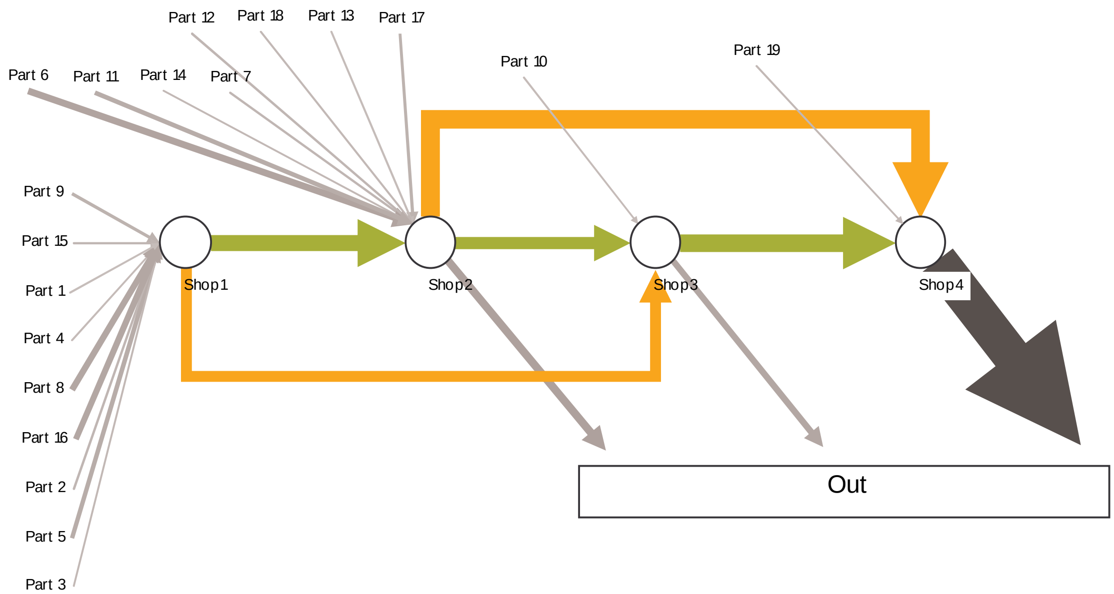


Figure 5.5: Hybrid Flowshop Layout – only flows between stages are shown

The optimization approach has been demonstrated. Next, a heuristic approach using a graph partition technique is described.

5.8 A Heuristic Solution Approach for Design of Hybrid Flowshop Layouts

A layout problem can be typically modeled as a graph problem and a hybrid flowshop layout problem can be modeled as a partitioning problem. Therefore, a graph partitioning problem can be used to model this hybrid flowshop layout problem. The complexity of a graph partitioning problem with the size constraint for each subset of nodes is known to be NP-complete. [Garey and Johnson, 1979] Therefore, several heuristic approaches have been developed for solving a graph partitioning problem. In circuit partitioning, an application of a graph partitioning for circuit board design that has been used successfully is called a "ratio cut" partitioning technique. [Wei and Cheng, 1991]. A ratio cut is described as follow:

"For a given graph $G = (V, E)$ where V is the set of nodes and E is the set of arcs, let c_{ij} be the capacity of an arc connecting node i and node j. A cut is a set of arcs that separates a set of nodes V of graph G into two disjointed sets A and A' where $A' = V - A$. A cut capacity is defined as a summation of arc capacities in the cut which is equal to $C_{AA'} = \sum_{i \in A} \sum_{j \in A'} c_{ij}$. The ratio of this cut is then defined as $R_{AA'} = (C_{AA'} / |A| \cdot |A'|)$, where $|A|$ and $|A'|$ denote the cardinalities of subsets A and A', respectively. The ratio cut is the cut that gives the minimum ratio among all cuts in the graph."

The ratio-cut technique provides two balanced subgraphs when using to bi-partition a graph. So we are going to adopt the ratio-cut technique in our heuristic approach to solve this hybrid flowshop layout problem. The proposed heuristic approach consists of three major phases: (1) problem graph transformation phase, (2) iterative bi-partitioning phase using the ratio-cut technique, and (3) machine allocation phase.

5.8.1 Problem Graph Transformation

At the first phase of the heuristic approach, a directed graph $G(N,A)$ representing each product routing in a layout problem needs to be constructed. The procedure of constructing a graph representing product routings in a hybrid flowshop layout problem is similar to the flow network transformation of the heuristic approach for a modular layout problem. However, in a modular layout problem arcs are not directed but they are in a hybrid flowshop layout problem. A directed graph constructed contains N nodes (vertices) that are representing machines in a problem and *A* arcs (edges) representing flows between a pair of machines. The same instance problem for the optimization approach will be used in this heuristic approach. The data that is needed at this phase consists of product routings and production volumes as shown in the table in Figure 5.6.

When all the part routings are obtained, a single line graph representing each part routing will be created where each arc in this graph is directed. For example, the routing of part 1 is 1➔4➔8➔9; therefore a single-line graph representing this part routing will be node 1 connects to node 4, node 4 connects to node 8, and node 8 connects to node 9. Each pair of nodes is connected with a directed arc according to its flow direction in the routing.

Product	Volume	Routing
1	2	1→4→8→9
2	34	1→4→7→4→8→7
3	23	1→2→4→7→8→9
4	12	1→4→7→9
5	65	1→6→10→7→9

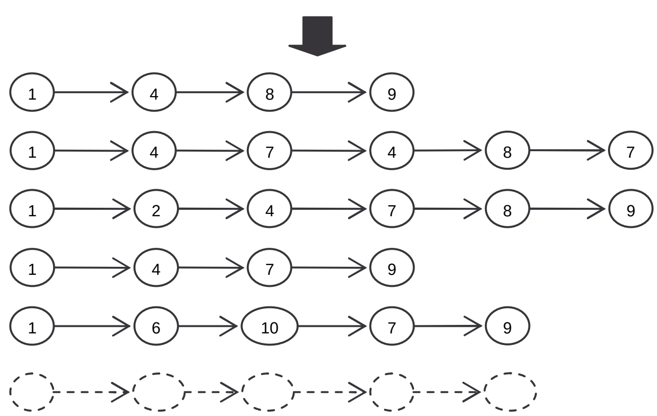


Figure 5.6: The construction of a problem graph representing the product routings

To ease the calculation in the demonstration of this approach, each arc in the problem graph is weighted as 1 and will not be shown in the network graph for clarity purposes. Since there is no backtracking flow allowed for this problem, it is not necessary at this stage to assign the weights of the production volumes to all the arcs in the problem graph. The production volumes will be required when the machine allocation is executed which is in the last phase of this heuristic approach. However, if an instance problem does not have a flow restriction, meaning that backtracking flows are allowed to some extent, then each arc needs to be weighted by its corresponding production volume. The approach still follows the same procedures whether arcs are weighted or not. The construction of a problem graph representing all the product routings of this instance problem is shown in Figure 5.6.

5.8.2 Iterative Bi-Partitioning

Once the problem graph of a hybrid flowshop layout problem has been constructed, the next phase is to iteratively bi-partition a set of nodes in this graph into smaller sets or so-called stages. The bi-partitioning process repeats until the size of each stage is met the size constraint. The ratio-cut technique will be used to locate two balanced sets of nodes. However, the problem of finding a ratio cut in a general graph belongs to the class of NP-complete [Matula and Shahrokhi, 1986]. A heuristic technique called *Shifting* technique is going to be used along with the ratio-cut technique to obtain good partitions with reasonable computational time [Wai and Cheng, 1991]. The shifting technique is similar to a pair-wise interchange technique where nodes are swapped in between different sets in order to seek for improvements.

Conceptually, the shifting technique would have to perform on the nodes in a problem graph. However, in our approach, the number of nodes in a problem graph is exploded by the total number of operations in the product routings. There can be a tremendous effort to perform this swapping-like technique for all the nodes in the graph Therefore, the shifting technique in our approach performs on a temporary set of machines that is constructed to represent all machine nodes in the problem graph at each step of partitioning process. A shifting operation performed on the temporary machine set will then be translated to an operation which will be performed on the original problem graph. Once the translated operation has been performed on the original problem graph, it generates partitions and the cut ratio value of these partitions can then be calculated. After that, this cut ratio value is fed back to guide the shifting to perform the next operation on the temporary machine set. The operations in the bi-partitioning process are as follows:

(1) Enumerate all machines in the problem graph (G) and construct two temporary balanced, disjoint machine sets, referred as LEFT set (M_L) and RIGHT set (M_R).

(2) Partition the problem graph into two partitions, referred as LEFT partition (A) and RIGHT partition ($A^{'}$) according to the LEFT set and RIGHT set of machine sets in step (1) respectively.

(3) Calculate the cut ratio $R_{AA'}$ for the current partitions A and $A^{'}$.

(4) Randomly select a machine *m* in M_R and temporarily move this machine to M_L. Update the partitions *A* and $A^{'}$ and recalculate the cut ratio $R_{AA'}$. This step is referred as “right shifting operation.”

(5) Repeat step (4) until all machines in M_R are performed. If there is any machine *m* that its movement a better cut ratio (less value) that the current cut ratio from step (3), then remove machine m from M_R and insert this machine to M_L.

(6) Repeat from step (4) until no improvement is found. Proceed to the next step.

(7) Randomly select a machine *m* in M_L and temporarily move this machine to M_R. Update the partitions *A* and $A^{'}$ and recalculate the cut ratio $R_{AA'}$. This step is referred as “left shifting operation.”

(8) Repeat step (4) until all machines in M_L are performed. If there is any machine *m* that its movement a better cut ratio (less value) that the current cut ratio from step (3), then remove machine m from M_L and insert this machine to M_R.

(9) Repeat from step (4) until no improvement is found. Proceed to the next step.

(10) Repeat the right shifting operation from step (4) and left shifting operation from step (7) back and forth until no improvement can be found. The final partitions are the solution of this bi-partitioning phase.

Figure 5.7 illustrates how the shifting technique on the machine set and the original problem graph are related. The bi-partitioning process described above generates two balanced subgraphs of the original graph problem. These subgraphs are representing the possible stages of a final layout solution. However, if there is any stage containing more machine types than the maximum stage size of the problem, this stage will have to be partitioned further. If there is such case, the subgraph of the overlarge stage will be partitioned by the bi-partitioning process described above. The process repeats until all the subgraphs (stages) are met the size constraint. The final set of stages that are automatically in sequence is a final layout solution to an original hybrid flowshop layout problem.

The following paragraphs demonstrate how the bi-partitioning process works. The problem graph in Figure 5.6 is used in this demonstration. First, in bi-partitioning phase, all machines in the problem need to be enumerated and arbitrarily assigned into two sets. There are 8 machines containing machines 1, 2, 4, 6, 7, 8, 9 and 10 that can be arbitrarily assigned into two sets $M_L = \{1,2,4,6\}$ and $M_R = \{7,8,9,10\}$. Then all nodes in the problem graph can be partitioned into two partitions A and A' corresponding to the two machine sets M_L and M_R. The partition A and A' are called the LEFT and RIGHT partitions respectively. As a result, the nodes in the problem graph are partitioned into two disjoint partitions A and A' while all the arcs connecting between all these nodes still remain as shown in Figure 5.7. From the figure, arcs that come from the LEFT partition to the RIGHT partition are called forward arcs. Then arcs come from the RIGHT partition to the LEFT partition are backward arcs.

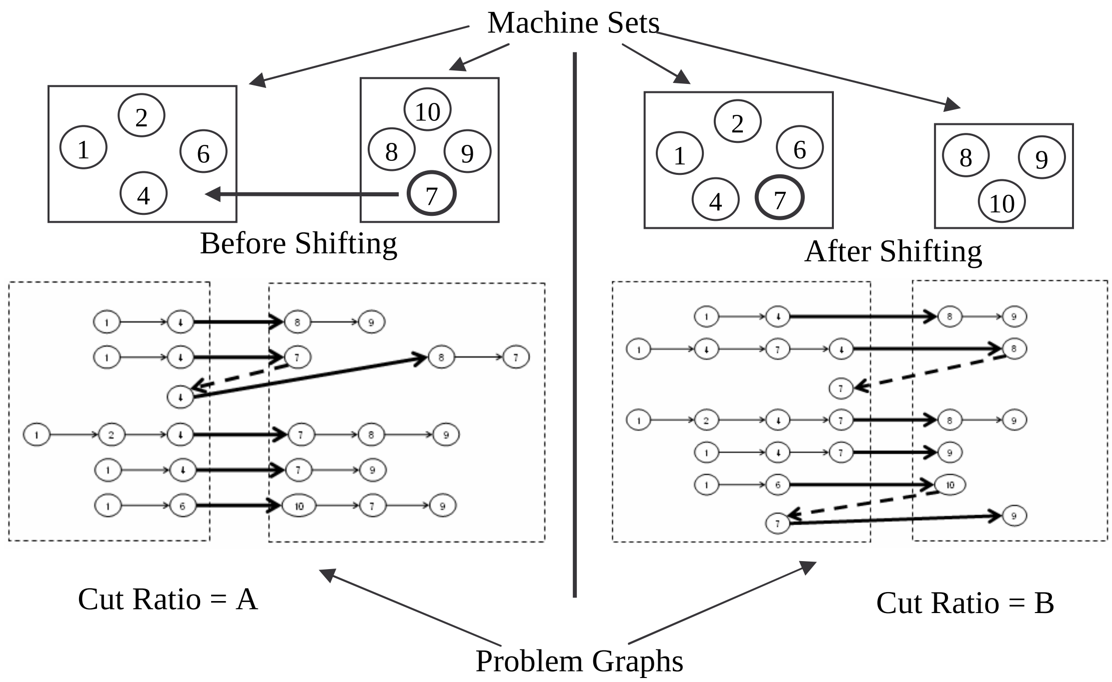


Figure 5.7: The relation between machine sets and problem graph partitions

The set of arcs connecting the nodes between *A* and *A*′ is called a cut. After the cut is found, a cut ratio for this cut can be calculated. The cut ratio in our heuristic approach is slightly different from its original calculation. Because in a hybrid flowshop layout problem, the machine purchasing cost as well as the penalty cost of having backtracking flows are also considered; therefore, the cut ratio has to take these two costs into account. The cut ratio for our heuristic approach is defined as:

$$R_{AA'} = \frac{D_A + \mu . E^-}{|A| . |A'| . E^+}, \text{ where}$$

D_A ($D_{A'}$) is the cost of machines purchasing,

$|A|$ ($|A'|$) is the number of machine types in A (A') from which D_A ($D_{A'}$) is obtained and selected,

E^- is the total backward flow between A and A',

E^+ is the total forward flow between A and A',

μ is the balancing factor.

The numerator $D_A + \mu . E^-$ of the ratio cut formula is the objective function of the hybrid flowshop layout problem model containing the cost of machines purchasing and the penalty cost of backtracking flows. The $|A|.|A'|$ term in the denominator is adopted straightforwardly from the original ratio-cut technique where it governs the cut to generate two balanced partitions. The total forward flow E^+, which is an additional multiplier to the denominator, ensures that the cut set contains the majority of forward arcs.

During the shifting operation, all the nodes in the problem graph will be partitioned into two sets residing in partitions A and A' according to the temporary machine sets as described. A cut which is the set of arcs connecting the nodes between A and A' is used to calculate a cut ratio. In our approach, whenever the calculation of a cut ratio is performed, we also seek to improve this value and the best cut ratio value gives the final cut ratio for this cut. There can be 3 following ways that influence the value of the final cut ratio for each cut.

(1) If there is no backtracking flow between *A* and *A′*, then this cut gives a final cut ratio. For this case, E^- is zero and there is no need to make further adjustment for this cut.

(2) If there are backtracking flows and it is possible that these backtracking flows can be removed by duplicating certain machines, then the trial process of removing the backtracking flows is executed. If it appears that the removal of backtracking flows by duplicating certain machines *does not improve* the value of the cut ratio, then the current value of cut ratio remains unchanged. For this case, E^- is non-zero and there is no need to make further adjustment for this cut.

(3) If (2) but it appears that the removal of backtracking flows by duplicating certain machines *improves* the value of the cut ratio, then the current value of cut ratio changes to the improved one. For this case E^- is zero, D_A ($D_{A'}$) increases, and *A* and *A′* need to be adjusted according to the removal of backtracking flows.

Next, we are going to describe how to obtain D_A ($D_{A'}$) and E^- according to the above statements. Whenever there are backward arcs occurring between two partitions, we can assume that these arcs are not preferred and need to be removed if feasible. To get rid of all the backward arcs, nodes that are connected to these arcs will have to move from one partition to another partition. Nodes that are connected to the backward arcs can reside in both RIGHT partition and LEFT partition. Choosing nodes from either the RIGHT partition or the LEFT partition to move can affect the total cost of machine purchasing. When a particular node moves from one partition to another partition, it can create a new backward arc. If such situation happens, to get rid of the new backward arc, the node that is connected to the new backward arc will have to move and so on. With the wrong node chosen, it could end up that all machines in a routing may have to move to another partition. That will require several machines to be duplicated as shown in Figure 5.8. Hence, all nodes in both partitions that are connected to the backward arcs need to be selected and exercised. The ones that their moves give the least amount of machine purchasing cost are the ones that account for the value of D_A ($D_{A'}$).

We are going to illustrate the process of determining the machine purchasing cost D_A ($D_{A'}$) as stated above. As can be seen in Figure 5.9, there is 6 forward arcs and 1 backward arc. Since each arc has its weight equal to 1, we can use the cardinality of arcs as the total flow and also the cost of machine duplication. The current cut ratio $R_{AA'} = (0 + 100 \times 1) / (4 \times 4 \times 6)$ where:

D_A ($D_{A'}$), the cost of machine purchasing for this cut = 0,

$|A|$ ($|A'|$), the number of machine types = 4 in A (1,2,4,6) and 4 in A' (7,8,9,10),

E^-, the total backward flows = 1, and

E^+, the total forward flows = 6.

The ratio for the cut demonstrated in Figure 5.9 is equal to 100/96 = 1.04.

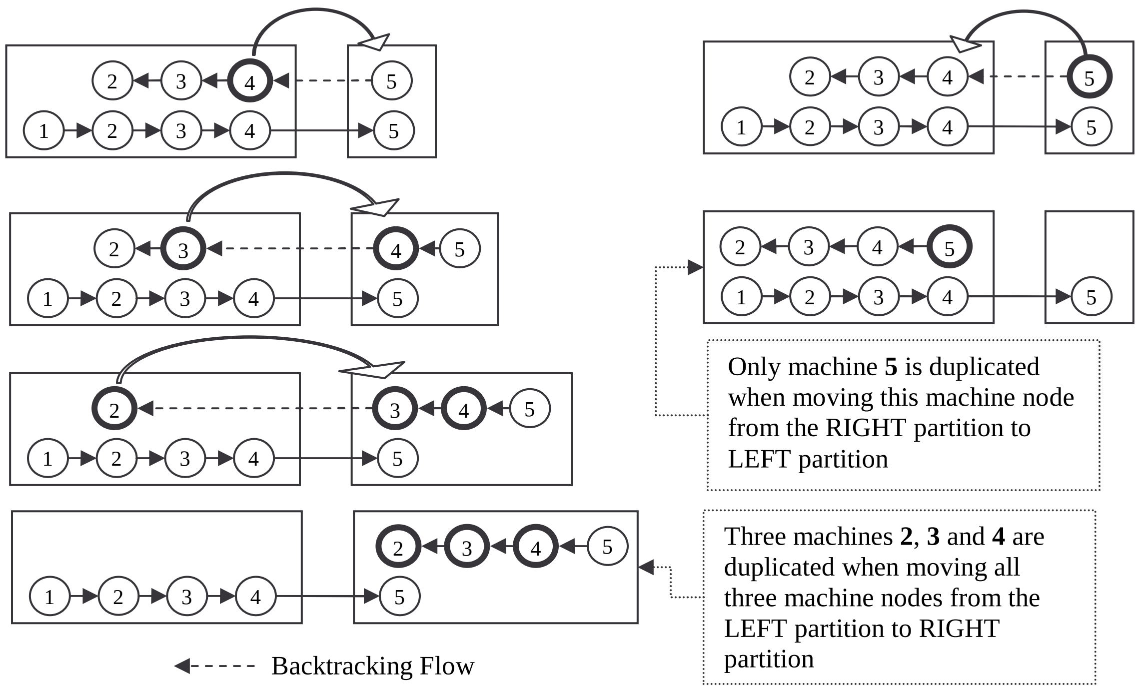


Figure 5.8: The different results when choosing which partition to move a node

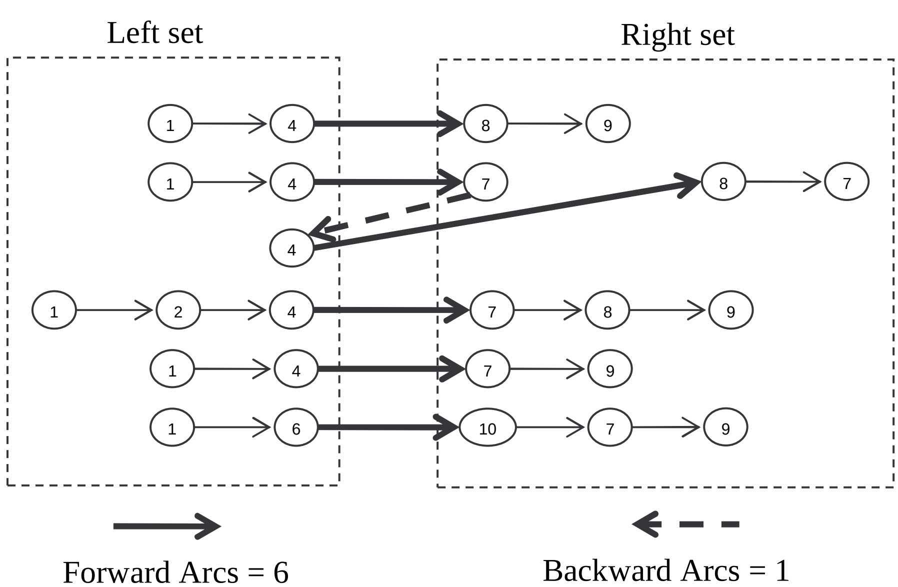


Figure 5.9: Initial cut

In order to get rid of the backward arc between the two partitions in Figure 5.9, either node 4 or node 7 that are connected to this arc had to move to another partition. Figure 5.10 shows the procedure to move node 4 from the LEFT partition to the RIGHT partition. After this node has been moved, all flows between the two partitions are forward. The total cast of machine purchasing, which is equivalent to the number of machines needed to get rid of the backward arc, is equal to 1, where machine 4 is being added to the RIGHT partition that originally contains only machines 7, 8, 9 and 10. Another way to get rid of the backward arc is to move node 7 from the RIGHT partition to the LEFT partition as shown in Figure 5.7. The number of machines needed for this move is also 1, where machine 7 is being added to the LEFT partition which originally contains only machines 1, 2, 4 and 6. Since the number of machines duplicated needed for either move is 1, we can choose either way to perform the calculation of the final cut ratio. Suppose, we are going to choose to add machine 4 to the RIGHT partition to get rid of the backward arc, the final cut ratio for the current partitioning is equal to $R_{AA'} = (1 + 100 \times 0) / (4 \times 5 \times 7)$ where

D_A ($D_{A'}$), the smallest cost of machine purchasing for this cut = 1,

$|A|$ ($|A'|$), the number of machine types = 4 in A (1,2,4,6) and 5 in A' (4,7,8,9,10),

E^-, the total backward flow = 0, and

E^+, the total forward flow = 7.

The ratio for the cut demonstrated in Figure 5.10 is equal to 1/140 = 0.0007.

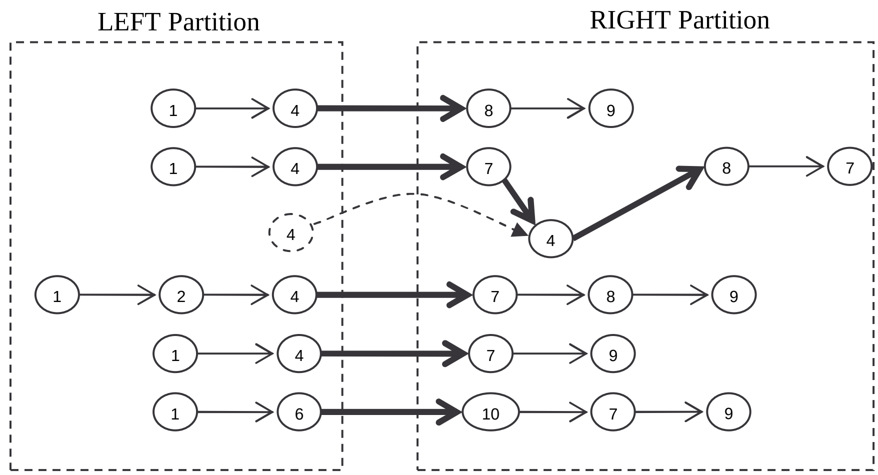


Figure 5.10: Initial with node 4 moved to the RIGHT partition

For the initial cut described above, the cut ratio without moving any machine node is 1.04. The cut ratio after a machine node moved and a machine needs to be duplicated is 0.0007. Therefore, the final value of the cut ratio for this initial cut is 0.0007. The final initial cut therefore contains no backward flows and the node of machine 4 is added to the RIGHT partition of this cut.

After the initial cut has been completely constructed, the shifting technique will perform in order to improve the value of the cut ratio by moving machine nodes from left to right and vice versa. The right shifting operation is going to perform first at this stage by moving machines 7, 8, 9 and 10, one at a time from the RIGHT partition to the LEFT partition. Each time that a machine moves, a new cut is generated and a cut ratio for this cut is calculated, the same way it was calculated for the initial cut. If there is a move that gives a better value of the cut ratio than the initial cut ratio, the machine nodes of this move are removed from the RIGHT partition and added to the LEFT partition permanently. The right shifting operation keeps performing until no improvement occurs. Then the left shifting operation performs afterward.

Both shifting operations repeat back and forth until no improvement can be found. The final cut obtained at the last step is the solution of this bi-partitioning phase which will be the two disjoint subgraphs obtained from the original problem graph. Once the first bi-partition is obtained, if there is any subgraph (stage) that contains more machine types than the stage size constraint, this subgraph has to be bi-partitioned further using the same partitioning technique. The process repeats iteratively until no subgraph is larger than the stage size. The partitioning process stops and all the subgraphs obtained at this last step are final stages in the layout solution of a hybrid flowshop layout problem.

When the layout solution is obtained, the last step is to allocate and assign machines to each stage and each operation being performed in each stage in the final layout solution. The same machine allocation technique in a modular layout problem is used here at this phase. All detailed description is already described in the previous chapter. Readers shall refer to the previous chapter for the detail of this technique.

In the next section, the demonstration of how the ratio-cut heuristic approach works is presented. The Vakharia dataset is used and the performance comparison between the heuristic approach and the optimization is presented.

5.9 Illustration of the Ratio-Cut Partitioning Heuristic Approach

The Vakharia data set is used to demonstrate the procedures of the ratio cut partitioning heuristic approach for the problem of designing a hybrid flowshop layout. The first step is to transform all routings in this problem into a problem graph as shown in Figure 5.11. For each node in the problem graph, there are two numbers separated by "|" shown inside each node. The first number is corresponding to a product number and the second number is corresponding to a machine number. Nodes that correspond to the same product are connected to form the single-line graphs representing the product routings. Arcs that connect these nodes have their weights equal to the corresponding production quantities. Since this problem all the weights is set to 1 so that arc weights are not shown in the figure.

Once the problem graph is constructed, all machines in this graph which are machines 1 through 12 are listed. This machine list is arbitrarily divided into two disjoint sets. Suppose, the two machine sets are {1,2,3,4,5,6} and {7,8,9,10,11,12}. The problem graph is then partitioned into two subgraphs where nodes corresponding to the machines in the same set are located in the same subgraph (partition) as shown in Figure 5.12. The two subgraphs, referred as the LEFT partition and RIGHT partition, are connected with the set of arcs which is called a cut. This cut containing both forward arcs and backward arcs as shown is used to calculate an initial cut ratio for the current partitioning which is equal to (0 + 100*2) / (6*6*15) = 0.37.

The next step is to examine whether the initial cut ration can be improved. As can be seen from Figure 5.12, there are few backward flows connecting between the two partitions that can be removed by moving either the nodes of machine 4 or machine 7 from one partition to another partition. Suppose the machine nodes 7 that are connected by the backward arcs are chosen to move from the RIGHT partition to the LEFT partition. After the machine nodes 7 have been moved, the current arcs connecting the two partitions are now all forward arcs as shown in Figure 5.13. The new cut ration is equal to (1 + 100*0) / (7*6*13) = .0018 which is less than the original initial cut ratio. Therefore, the new ratio cut is the final value of the initial cut ratio and the problem graph is updated to as the graph that contains two subgraphs where all flows connecting these two subgraphs are forward flows.

Once the initial cut is completely constructed, the shifting technique of right operation and left operation is performed. However, for this example problem, there is no better solution than this initial cut. Therefore, the first layout solution is found where there are two stages containing machine sets {1,2,3,4,5,6,7} and {7,8,9,10,11,12} created as shown in Figure 5.13. Since the stage size of this problem was set to 4 so both stages will have to be partitioned further using the same bi-partitioning technique described.

After iteratively bi-partitioning the over-size stages until all stages are reduced to meet the stage size constraint, the final solution is obtained as shown in Figure 5.14. There are 5 stages in the final hybrid flowshop layout of the Vakharia data set. All flows between stages are forward flows. This final layout requires only machine 7 to be placed in two different stages. After the final layout is obtained, it can be seen clearly that operations in each routing are already allocated to specific machines in each stage. Therefore, the process to allocate machines, calculate workloads and obtain the number of additional machines required can be done straightforwardly after this step.

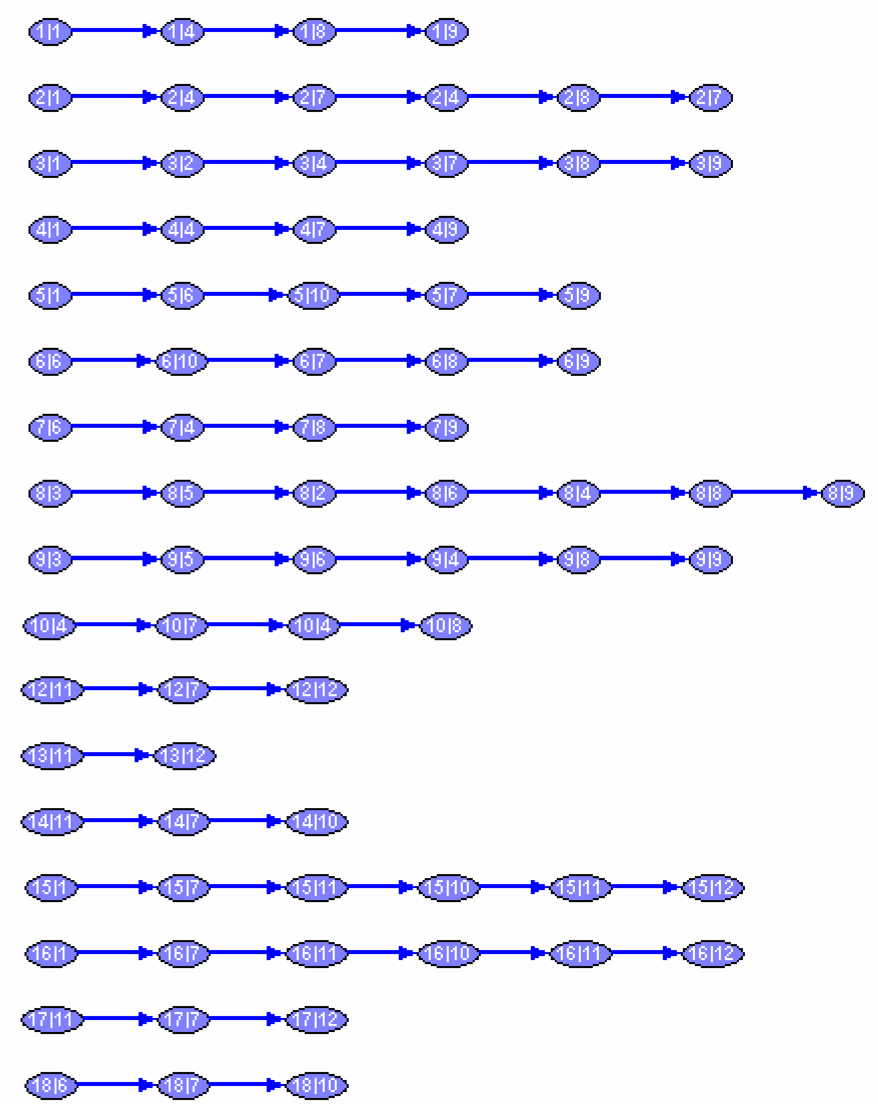


Figure 5.11: The problem graph representing product routings

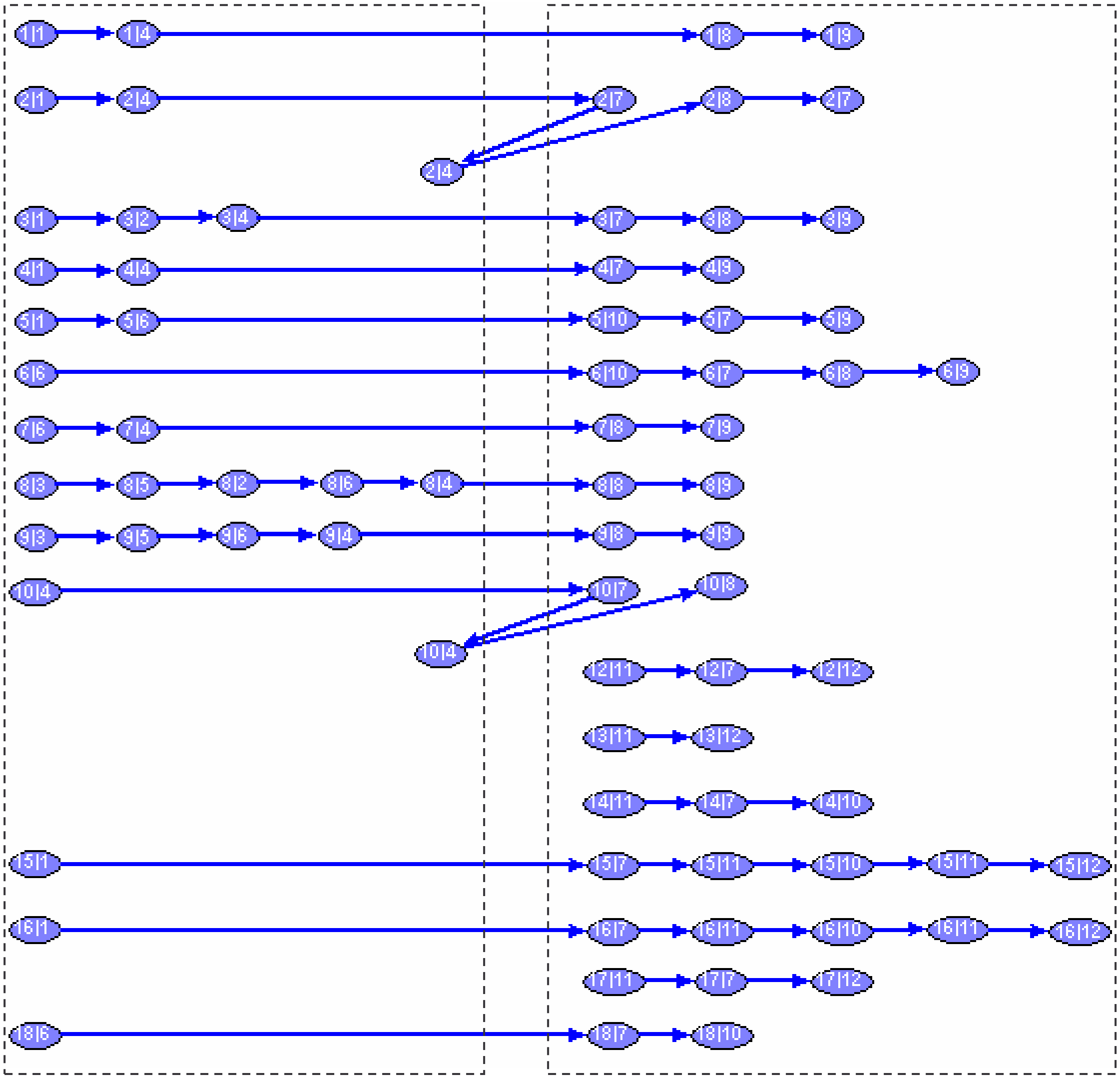


Figure 5.12: Two partitions from the first cut

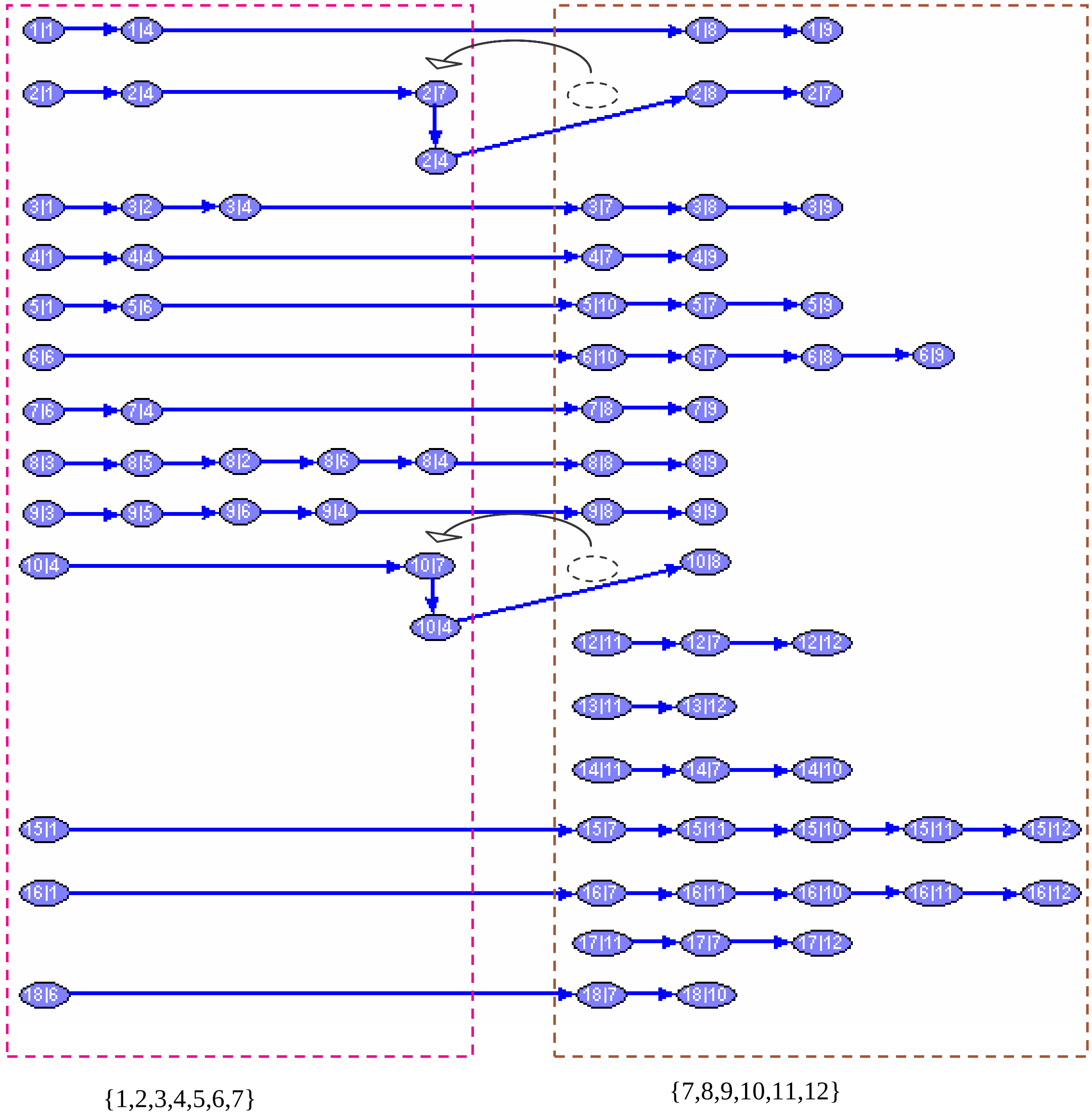


Figure 5.13: Initial cut with nodes of machine 7 moved to the LEFT partition

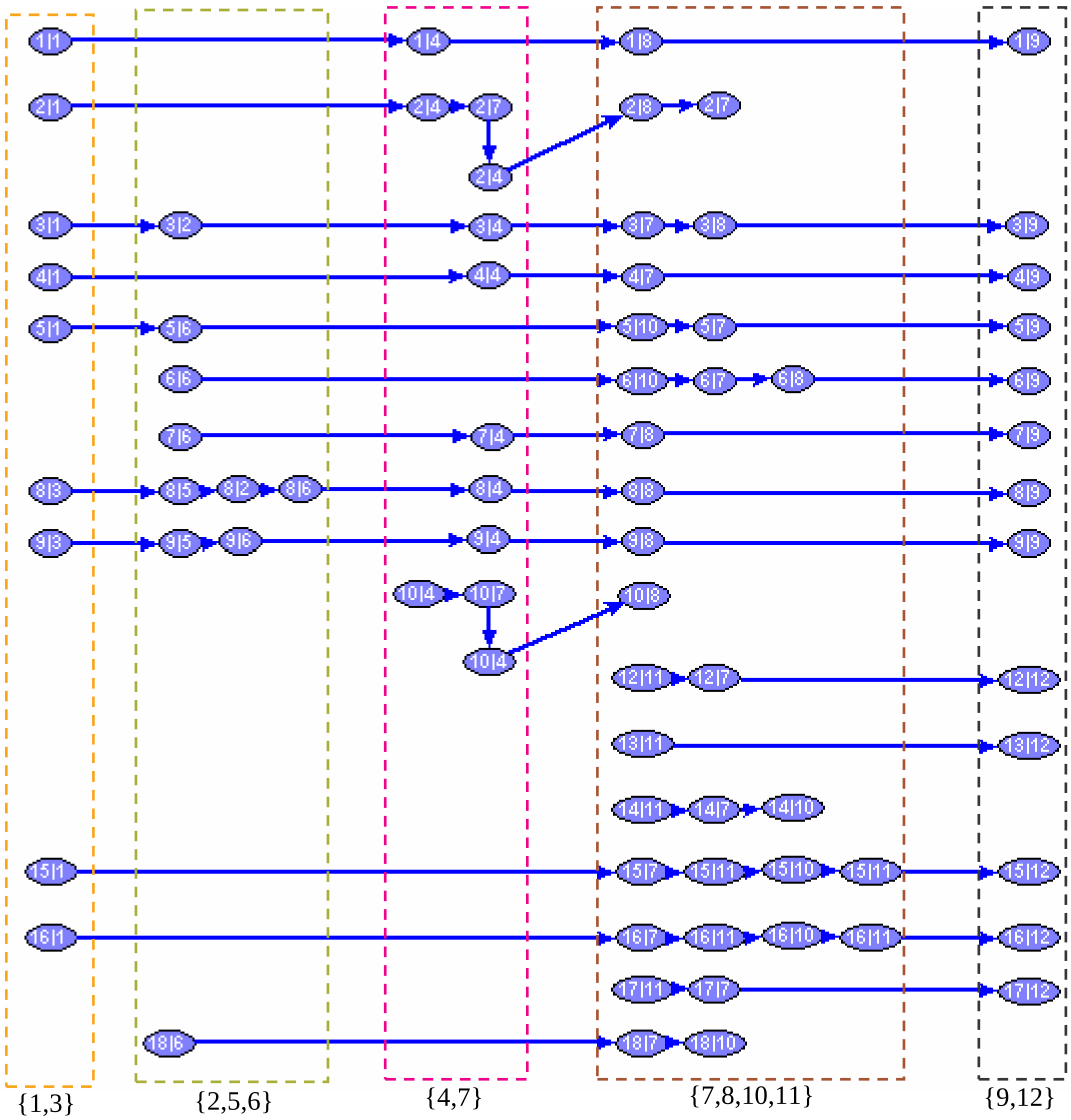


Figure 5.14: Final solution

Next, the result from the heuristic approach and the comparison performance between the optimization approach and heuristic approach are presented.

5.10 Results from Heuristic Approach

The heuristic approach described in previous section was programmed in C++ on Pentium 4, 2.53GHz, 1GB Ram Windows-based machine. The problem instance of a hybrid flowshop layout problem was constructed from the Vakharia dataset containing 19 products and 12 machines. The heuristic approach took less than 1 second to solve this problem instance. The flowshop layout solution contains 5 stages including {1,3}, {2,5,6}, {4,7}, {7,8,10,11}, and {9,12}. This layout solution can be revised manually further by merging small-size stages together. The revised layout solution can then contain only 4 stages including {1,2,3,5}, {4,6,7}, {7,8,10,11} and {9,12} where only machine 7 is placed in two different stages.

Parts	Stage 1	Stage 2	Stage 3	Stage 4
1	1	4	8	9
2	1	4 7 4	8	7
3	1 2	4 7	8	9
4	1	4 7		9
5	1	6	10 7	9
6		6	10 7 8	9
7		6 4	8	9
8	3 5 2	6 4	8	9
9	3 5	6 4	8	9
10		4 7 4	8	
11		6		
12			11 7	12
13			11	12
14			11 7 10	
15	1		7 11 10 11	12
16	1		7 11 10 11	12
17			11 7	12
18		6	7 10	
19				12

Figure 5.15: Result from the heuristic approach

After the result is obtained from the heuristic approach, the work load calculation and machine allocation can be performed. The result is shown in Figure 5.16. From the calculation, there is 1 copy of machine 7 required for stage 2 and 4 copies required for stage 3. There are only 4 copies of machine 7 available; therefore, 5 copies of machine 7 need to be purchased additionally to satisfy the capacity requirement of each stage in the solution. The performance comparison between the heuristic approach and optimization approach is presented next.

Product	Machine 1	2	3	4	5	6	7	8	9	10	11	12
1	0.02			0.01				0.01	0.01			
2	0.26			0.46			0.51	0.10				
3	0.18	0.21		0.16			0.17	0.07	0.13			
4	0.09			0.08			0.09		0.07			
5	0.50					0.31	0.49		0.36	0.58		
6						0.47	0.74	0.29	0.54	0.88		
7				0.23		0.16		0.10	0.19			
8		0.79	0.66	0.59	0.66	0.42		0.26	0.48			
9			0.34	0.31	0.34	0.21		0.13	0.25			
10				0.16			0.09	0.04				
11						0.32						
12							0.26				0.28	0.15
13											0.06	0.03
14							0.20			0.23	0.22	
15	0.26						0.26			0.30	0.57	0.15
16	0.69						0.67			0.80	1.49	0.38
17							0.34				0.38	0.19
18						0.11	0.17			0.21		
19												0.10
Stage 1	2.00	1.00	1.00		1.00							
Stage 2				2.00		2.00	0.87					
Stage 3							3.13	1.00		3.00	3.00	
Stage 4									2.00			1.00
M/Cs Required	2	1	1	2	1	2	5	1	2	3	3	1
M/Cs Available	2	1	1	2	1	2	4	1	2	3	3	1
Extra M/Cs							1					

Figure 5.16: Machine allocation for the heuristic approach

5.11 Comparison Study

The results from both optimization approach and ratio-cut heuristic approach have shown in Table 5.4. The study has been done for 6 different data sets, same as in the experimental study of a modular layout problem. The details for these datasets are shown in Appendix B.

Dataset	CPU Time (s)		No. of M/Cs duplicated	
	OPT	R/C	OPT	R/C
Vakharia_19x12	32	<1	1	1
ABB_50x25	13,412	<1	2	3
Mettler_25x13	14,074	<1	4	6
Purcheck_28x18	20	<1	0	0
Sekine_13x12	<1	<1	0	0
Tecomet_42x15	441	<1	1	2

Remarks: PT = Optimization Approach
R/C = Ratio Cut Heuristic Approach

Table 5.4: Performance comparison between the optimization and the ratio-cut heuristic approaches

As can be seen from the performance comparison table, the heuristic approach has performed reasonably well providing the same solutions as optimization approach from 3 out of 6 cases. The cases that the heuristic approach can not provide the optimal solutions are the cases of ABB_50x25, Mettler_25x13 and Tacomet_42x15. These cases are more complex than the others since they require a certain number of machines duplicated.

Although, the heuristic approach cannot obtain the optimal solutions for such cases, its solution is acceptable when considering its computation time which is less than a second while the optimization approach takes more than 3 hours in two cases. The enhancement approaches such as random search techniques can absolutely help improving the quality of solution from this heuristic approach.

5.12 Discussion and Conclusions

In this chapter, a hybrid flowshop layout was introduced to address the lack of efficient layout that can exploit the benefits of flow process production in HMLV facilities. The concept of this layout is to divide, duplicate and reorder existing machines in a functional layout and transform them into a flowline-like layout with minimum additional machines required. Machines in a hybrid flowshop layout are divided into disjoint groups called stages, all stages are arranged in a sequence, and forward flows, either in-sequence or bypass, between these stages in the layout are promoted. Each stage may contain machines of the same types or different types. It is similar to a layout module; however, in a modular layout, intra-module and inter-module flows are more focused. In a hybrid flowshop layout, flow direction and machine duplication are more focused.

The problem of designing a hybrid flowshop layout was proven to belong to the class of NP-complete problem. Thus, solving a hybrid flowshop layout problem optimally is applicable only for small problem instances. For large problem instances, a heuristic approach using the ratio-cut partitioning technique was developed. The results of this heuristic has shown that it can provide good results with less computational efforts compared to the optimization approach. Incorporating with enhancement approaches such as random search techniques should help improving the solution quality of the solution from this ratio-cut approach.

References

1. Artiba, A. and Elmaghraby, S.E., *The planning and scheduling of production systems : methodologies and applications*, Chapman & Hall, New York (1997)

2. Askin, R.G. and Zhou, M., "Formation of independent flowline cells based on operation requirements and machine capacities," *IIE Transactions*, Vol. 30, pp. 319-329 (1998)

3. C. M. Fiduccia and R. M. Mattheyses, "A linear time heuristic for improving network partitions," in *Proceeding 19th Design Automation Conference*, pp. 175-181 (1982)

4. D. W. Matula and F. Shahrokhi, "The maximum concurrent flow problem and sparsest cuts," *Technical Report*, Southem Methodist University (1986)

5. Framinan, J.M., "Efficient heuristic approaches to transform job shops into flow shops," *IIE transactions*, Vol. 37, pp.441-451 (2005)

6. Framinan, J.M. and Ruiz-Usano, R., "On transforming job-shops into flow-shops," *Production Planning & Control,* Vol.13, No. 2, pp. 166-174 (2002)

7. Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Company, New York (1979)

8. Kadipasaoglu, S.N., Xiang, W. and Khumawala, B.M., "Batch Scheduling in Multistage, Multiproduct Manufacturing Systems - An Application", *International Journal of Operations and Production Management*, Vol. 19, No. 3, pp. 421-436 (1999)

9. Kouvelis, P. and Chiang, W.C., "A Simulated Annealing Procedure for Single Row Layout Problems in Flexible Manufacturing Systems," *International Journal of Production Research*, Vol. 30, No.4, pp. 717-732 (1992)

10. Ho, Y. and Moodie, C.L., "Machine layout with a linear single-row flow path in an automated manufacturing system," *Journal of Manufacturing Systems*, Vol. 17, No. 1, pp. 1-22 (1998)

11. Lee, S. and Chiang, C., A cut-tree-based approach for clustering machine cells in the bidirectional linear flow layout, *International Journal Production Research*, Vol. 39, No. 15, pp. 3491-3512 (2001)

12. Li, J. and Barnes, D.J., "Investigating the factors influencing the shop performance in a job shop environment with kanban-based production control," *International Journal of Production Research,* Vol. 38, No. 18, pp. 4683-4699 (2000)

13. Matula, D. W. and Shahrokhi, F., "The maximum concurrent flow problem and sparsest cuts," *Technical Report*, Southem Methodist University (1986)

14. Ourairat, A., *The Machine Duplication Problem (MDP) in Design of Flowline Layouts*, Ph.D. Dissertation, The Ohio State University (2000)

15. Raiha, K.J. and Ukkonen, E., "The shortest common supersequence problem over binary alphabet is NP-complete, *Theoretical Computer Science*, Vol. 16, pp. 187-198 (1981)

16. Stock, D.J. and Lindley, R.J., "Implementing kanbans within high variety/low volume manufacturing environments," *International Journal of Operations and Production Management*, Vol. 15, No. 7, pp. 47-59 (1994)

17. Stock, D.J. and Lindley, R.J., "Batch production using process sequence cells," *International Journal of Production Research*, Vol.36, No. 9, 2419-2433 (1998)

18. Suzaki, K., *The New Manufacturing Challenge: Techniques for Continuous Improvement*. New York, NY: The Free Press (1987)

19. Tompkins, J.A., While, F.A., Bozer, Y.A., Frazelle, E.H., Tanchoco, J.M.A., and Trevino, J., *Facilities Planning*, New York, NY: John Wiley (1996)

20. Sarker, B.R., Wilhelm, W.E., Hogg, G.L. and Han, M., Backtracking of Jobs in One-Dimensionla Machine Location Problems, *European Journal of Operational Research*, Vol. 85, pp. 593-609 (1995)

21. Wei, Y. and Chieng, C., "Ratio cut partitioning for hierarchical design," *IEEE Transactions On Computer-Aided Design*, Vol. 10. No. 7, pp. 911-921 (1991)

CHAPTER 6

LAYOUT DESIGN FOR JOBSHOPS IN HIGH-MIX LOW-VOLUME MANUFACTURING FACILITIES USING PFAST

6.1 Introduction

In this chapter, we describe current methods to design the layouts, including the proposed layouts, for jobshops operating in a HMLV environment. The use of PFAST (Production Flow Analysis and Simplification Toolkit) software to design these layouts is demonstrated using a case study to describe various practical concepts and approaches for layout design. There are several criteria influencing the process of selecting and designing a layout that is a "best fit" for a HMLV facility. It seems that there is no absolute method or solution that suits every criterion and different configurations of product variety, production volume, and demand stability. Hence, we introduce a range of conceptual layout configurations and methods for selecting and designing both traditional and non-traditional layouts that can suit different HMLV facilities. The software PFAST was used in a real-world case study based to demonstrate how to design layouts for any complex HMLV jobshop facility.

6.2 Layout Design with PFAST

One of the most successful approaches for designing a layout for a HMLV facility is Production Flow Analysis (PFA), introduced by Burbridge (1963). Its objective was to provide an efficient method of transforming a "process focused organization" to "product focused organization." It is an approach derived from the concept of Group Technology (GT) which seeks to identify and group together similar parts to take advantage of their similarities in manufacturing and design. Using PFA, complex material flows resulting from process oriented layouts, or functional layouts, are converted into more organized and efficient flows via transformation to product oriented layouts, which are either cellular or flowline layouts. However, this approach is only suitable when the complexity of the production flow is not too high and the product mix clearly contains product families. When the complexity is very high and product families are unclear as of most HMLV facilities and jobshops, traditional manual PFA can be difficult to apply. PFAST, developed by Irani et al (1999), extends the manual methods of PFA to enable the study of production flows in complex HMLV environments when the manual methods of PFA cannot be used.

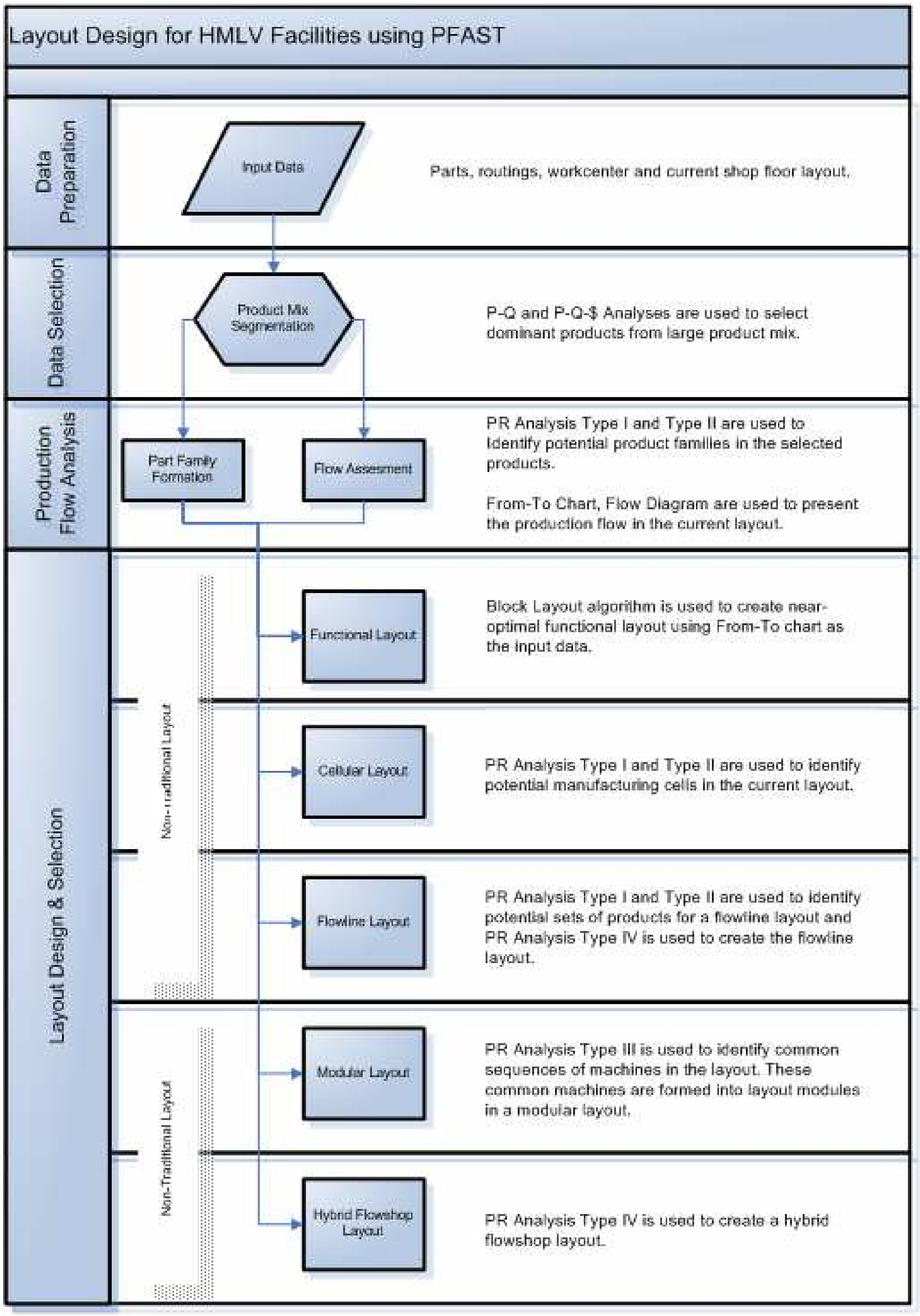


Figure 6.1: An overall framework of production flow analysis (PFA) and layout design using PFAST

PFAST provides an effective integrated suite of algorithms for production flow analysis and layout design. Key benefits of using PFAST are simplifying the flow complexity in real-world HMLV environments by converting complex flows into more organized and efficient flows and selecting the most appropriate layout on a case-by-case basis.

The current version of PFAST consists of several algorithms which can be categorized into the following modules:

a. Data Collection
b. Product Mix Segmentation
c. Flow Assessment
d. Product Family Formation
e. Layout Design (Traditional Layouts and Non-Traditional Layouts)

Figure 6.1 shows the systematic use of the above modules to simplify production flows and design facility layouts using PFAST. The flow chart starts with data collection. Input data required by PFAST include information of the products, machines, and the current layout of the facility is gathered and fed into the software. The Product Mix Segmentation module is used to identify critical products based on production volume & revenue especially when the size of product mix is too large to analyze. The Flow Assessment module coupled with Product Family Formation help to assess the complexity of the current production flow network. At this step, PFAST suggests the possibility of a cellular layout for a portion of a product mix if there is evidence of the existence of a few clear-cut product families in the product mix. As shown in Figure 6.5, the Layout Design module helps analysts to design different types of layouts. The design of a Functional Layout is done using from-to chart analyses and a Block Layout algorithm. The Product-Process Matrix Clustering (PR-I Analysis) and the product-routing Hierarchical Clustering technique (PR-II Analysis) are used to design a Cellular Layout. To design a Flowline Layout, approaches used to design a Cellular Layout can be coupled with the Modified Multi-Product Process Chart (MM-PPC) or PR-IV Analysis. Modular layouts can be designed by using a substring-clustering technique (PR-III Analysis) coupled with Hierarchical Clustering (PR-II Analysis). In addition, a Hybrid Flowshop layout can be designed by using the MM-PPC (PR-IV Analysis).

6.3 Case Study

This section presents an industrial application using PFAST to design manufacturing layouts in a HMLV manufacturing environment. Industrial data were collected from a forging company. This is a job shop company manufacturing a product mix that contains 79 products and the routings of which range from 3 to 15 operations. Details of this industrial data are described in Appendix A.

6.3.1 Data Collection

In order to run PFAST, three input data files are required containing the following data:

1) Products

2) Machines

3) Current layout

This input data is also known as P-Q-R-$ data because it consists of: a) **P**art (or product) # and description (optional), b) annual production **Q**uantity, c) manufacturing **R**outing, and d) annual **$**ales (or profit or revenue).

Each operation in a manufacturing routing must provide a specific workcenter, a group of identical machines, to process the part. Machine information includes the list of all manufacturing workcenters and supporting equipment that appear in the routings of parts produced in the manufacturing facility. In addition, the list of machines, area requirements of each machine footprint, and their layout attributes should be provided. Examples of these attributes are:

i) Whether a machine is a monument, i.e. would it be very expensive to relocate that piece of equipment,

ii) Whether additional copies of the equipment needed in different manufacturing cells can be purchased at reasonable cost, and

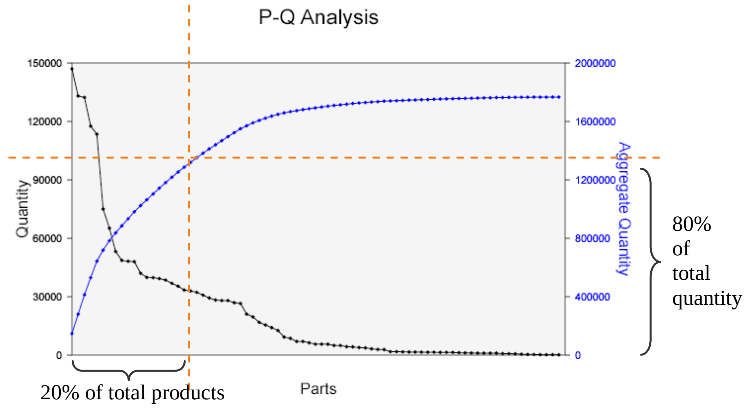
iii) Whether a machine is interchangeable in its capabilities with any other machines.

The drawing of the current layout displays the locations of all manufacturing workcenters and support services utilized in the facility. This information will be needed to visualize the production flow in the current layout graphically and it also serves as a template to design new layouts.

6.3.2 Product Mix Segmentation

In order to study and simplify production flows in a manufacturing facility, the first step is to identify and separate the significant from the insignificant products, thereby; focusing the products contributing to the dominant flows. Product Mix Segmentation can be done using (i) P-Q Analysis which is also known as an ABC Analysis or Pareto Analysis and (ii) P-Q-$ Analysis which is a bi-criteria extension of the P-Q Analysis. By using either of these two techniques, dominant products can be quickly identified using an 80-20 rule. Depending upon the criterion to choose, either Production Quantity Alone or both Production Quantity and Revenue, the 80-20 rule seeks the sample of products that contributes 80% of total Quantity using P-Q Analysis, or 80% of both Quantity and Revenue using P-Q-$ analysis. Figure 6.2 shows an example of how the dominant products can be selected by looking in the Aggregated Quantity column in P-Q Analysis, or both the Aggregated Quantity and Aggregated Revenue columns in Figure 6.3.

The product mix segmentation described above can be very helpful when working with very large datasets. In this case study, since there are only 79 products in the mix, this analysis was ignored. However, if the data had ≥ 1,000, then product mix segmentation would have been necessary.

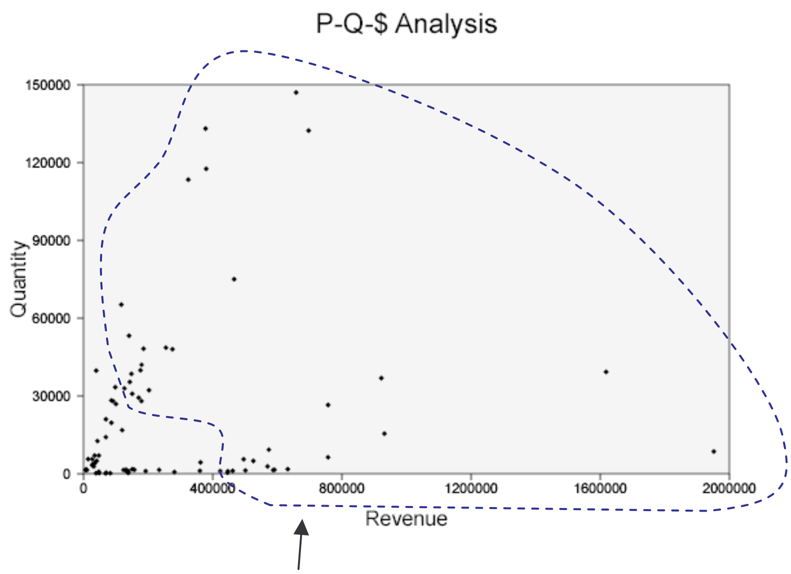


Dominant set of products

Part	Quantity	Agg. Qty.	Agg. Qty. %
80-671391	147000	147000	8.3
80-4012179	133070	280070	15.9
80-35-B357	132314	412384	23.3
80-4010346	117614	529998	30
80-4012174	113400	643398	36.4
80-671635-00	75012	718410	40.7
80-4030339	65198	783608	44.4
80-4030341	53200	836808	47.4
80-051-1	48580	885388	50.1
80-B113-1001	48132	933520	52.8
80-121018-00	47950	981470	55.6
80-4041707	42000	1023470	57.9
80-4009263	39886	1063356	60.2
80-S113-1001	39732	1103088	62.4
80-4030011870964	39256	1142344	64.7
80-4010350	38500	1180844	66.8
80-9033023-303	36848	1217692	68.9
80-121148	35350	1253042	70.9
80-4012169	33362	1286404	72.8
80-4009270	32900	1319304	74.7
80-121188-002	32200	1351504	76.5
80-4009121	30800	1382304	78.3
80-121009-00	29288	1411592	79.9
80-4035149	28252	1439844	81.5
80-3249869	28014	1467858	83.1
80-4011714	28000	1495858	84.7
80-191820	26866	1522724	86.2
80-4067179	26502	1549226	87.7

About 80% of total quantity

Figure 6.2: P-Q Analysis



Dominant set of products based on both quantity and revenue

Part	Revenue	Quantity	Agg. Rev.	Agg. Qty.	Agg. Rev. %	Agg. Qty. %
80-671391	658560	147000	658560	147000	2.9	8.3
80-27708-302UP	1952230	8540	2610790	155540	11.6	8.8
80-35-B357	697298	132314	3308088	287854	14.7	16.3
80-4012179	377916	133070	3686004	420924	16.4	23.8
80-4030011870964	1618932	39256	5304936	460180	23.6	26.1
80-4010346	379806	117614	5684742	577794	25.3	32.7
80-4012174	324324	113400	6009066	691194	26.7	39.1
80-671635-00	466340	75012	6475406	766206	28.8	43.4
80-9033023-303	922670	36848	7398076	803054	32.9	45.5
80-27750-01	931854	15428	8329930	818482	37.1	46.3
80-4030339	117208	65198	8447138	883680	37.6	50
80-4067179	757428	26502	9204566	910182	41	51.5
80-4030007296091	757190	6356	9961756	916538	44.3	51.9
80-4030341	140980	53200	10102736	969738	45	54.9
80-121018-00	275716	47950	10378452	1017688	46.2	57.6
80-051-1	255052	48580	10633504	1066268	47.3	60.4
80-B113-1001	186116	48132	10819620	1114400	48.1	63.1
80-NL150T060LT	632744	1764	11452364	1116164	51	63.2
80-NL150T072LT	591752	1540	12044116	1117704	53.6	63.3
80-150T084LT	587664	1344	12631780	1119048	56.2	63.3
80-4041707	180180	42000	12811960	1161048	57	65.7
80-4030007296094	574084	9240	13386044	1170288	59.6	66.2
80-3260-041	569842	2828	13955886	1173116	62.1	66.4
80-4009263	176302	39886	14132188	1213002	62.9	68.7
80-4010350	148610	38500	14280798	1251502	63.5	70.8
80-S113-1001	39732	39732	14320530	1291234	63.7	73.1
80-921790	526120	4914	14846650	1296148	66.1	73.4
80-37355-1072	501774	1204	15348424	1297352	68.3	73.4
80-C27416-2	495992	5614	15844416	1302966	70.5	73.8
80-121148	143444	35350	15987860	1338316	71.1	75.8
80-121188-002	202860	32200	16190720	1370516	72	77.6
80-9627713-301UP	462378	1050	16653098	1371566	74.1	77.6
80-4009270	126994	32900	16780092	1404466	74.7	79.5
80-4012169	98756	33362	16878848	1437828	75.1	81.4
80-37355-1084	447062	952	17325910	1438780	77.1	81.4
80-W101-2006	445858	462	17771768	1439242	79.1	81.5
80-4009121	151228	30800	17922996	1470042	79.7	83.2
80-121009-00	171332	29288	18094328	1499330	80.5	84.9
80-9627715-301UP	423766	1022	18518094	1500352	82.4	84.9
80-3249869	179200	28014	18697294	1528366	83.2	86.5
80-4035149	86170	28252	18783464	1556618	83.6	88.1
80-4011714	91560	28000	18875024	1584618	84	89.7
80-191820	100744	26866	18975768	1611484	84.4	91.2
80-C46806-1	362474	4354	19338242	1615838	86	91.5
80-9627714-301UP	360430	1078	19698672	1616916	87.6	91.5
80-4010348	69300	21000	19767972	1637916	88	92.7

About 80% of total quantity and total revenue

Figure 6.3: P-Q-$ Analysis

6.3.3 Flow Assessment

Once the dominant products are identified and separated, PFAST helps to visualize their production flows using both graphical and quantitative analyses. The graphical analysis is called a Flow Diagram or a Spaghetti Diagram. There are 3 types of diagrams, Q-Type, $-Type, and ƒ-Type representing volume flows, sales flows, and the number of different products moving between machines, respectively. Figure 6.4 shows the Q-Type diagram mapped on the current layout of this facility.

Corresponding to each graphical analysis, a quantitative analysis called From-To Chart, each chart being a quantitative representation of the corresponding spaghetti diagram. Figure 6.5 shows the Q-Type From-To Chart of this case study. Particular flows between pairs of machines in the Q-Type spaghetti diagram in Figure 6.4 can be read from this chart. For example, there is a thick arrow from machine 1 to machine 26. This flow can be read from the From-To chart by finding an entry in the chart from the row for machine 1 and the column for machine 26. The appropriate entry in the chart is 365,806 meaning that a total of 365,806 products are traveling from machine 1 to machine 26 annually.

Both graphical and quantitative analyses as described enable analysts to investigate the complexity of the flows and to recognize suitable layout solutions for reducing the complexity of the current production flows. As can be seen in the Q-Type spaghetti diagram for this case study, all flows in the current layout are chaotic and crisscrossing the entire facility layout. Most HMLV manufacturing companies exhibits this kind of disorganized flows in their facilities. Many thick lines that represent high volume flows traverse across buildings or machines within any building. Functional layouts, where similar machines tend to be placed in the same departments (or buildings) will always exhibit there chaotic flows.

If machines are not placed close to each other, products that are traversing from one machine to the other machine will be batched placed in large containers that material handlers will then transport between the machines at large intervals. The disadvantage of batch manufacturing is that there is a discontinuity in material flows and significant inter-machine travel delays between machines. Inside a building, batches often take about a day or less to travel from one machine to another. Between buildings, they may take several days or a week. Thus, batch processes often lead high production lead time and high work-in-process (WIP) inventory at machine machines.

Besides the high production lead time, high WIP inventory, and high material handling cost, when the flows are chaotic in any facility that means the scheduling and product tracking become more complicated. Among three traditional layouts—a functional layout, a cellular layout, and a flowline layout, material flows can be scheduled and tracked much easier in the flowline layout and the cellular layout compare to the functional layout. Thus, many HMLV companies tend to deploy either cellular or flowline layouts, both of which are essentially product-oriented layouts.

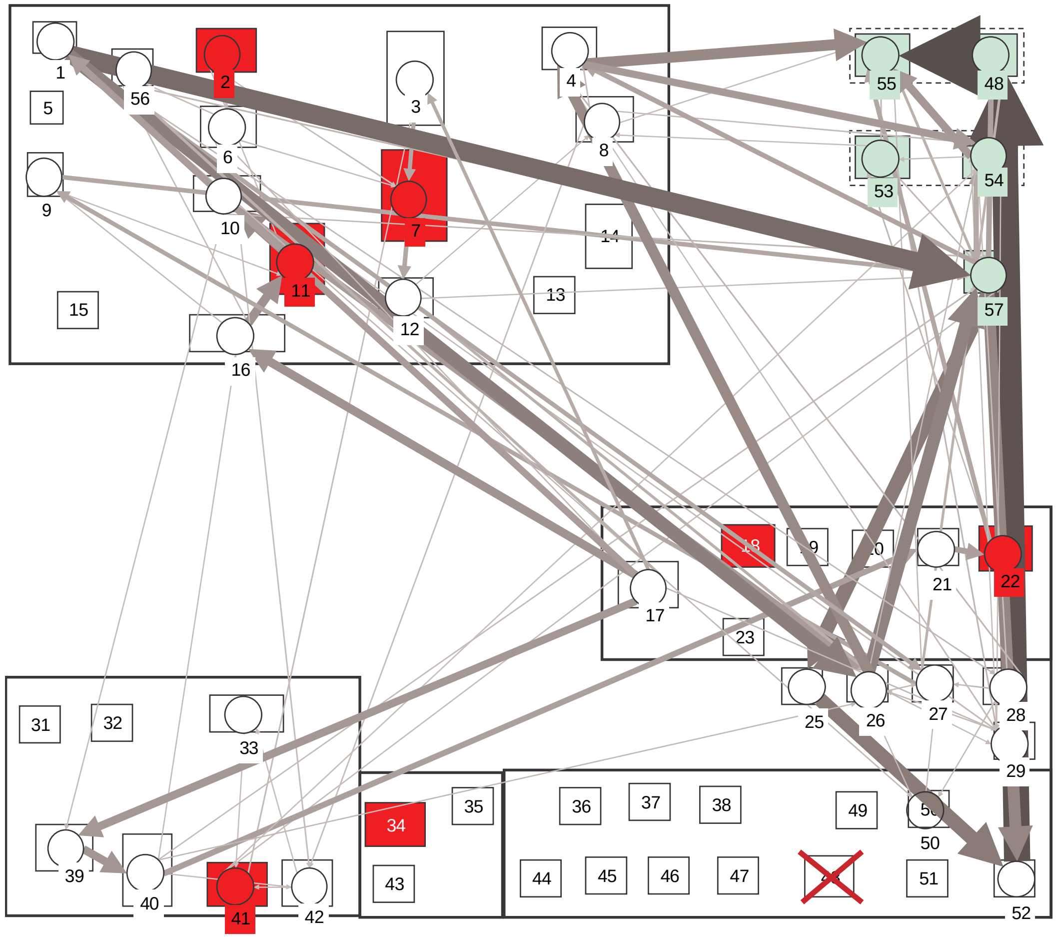


Figure 6.4: Q-Type spaghetti diagram displaying material flow in the current layout

W/C	1	2	3	4	6	7	8	9	10	11	12	16	17	21	22	25
1													28014			
2						8008				20706						
3						112574										
4																
6		29176				5194										
7											125776					
8				17220												
9										8540						
10					5194											
11									267890							
12							35336									
16								8540		238644						
17	180894		97146		34370							233450				
21															161728	
22																
25																
26				284704												
27								118398								
28				44170												
29				17220												
33																
39																
40												8540		161728		
41			15428													
42																
48																
50																
52																
53							5194									
54																
55																
56	28014											5194				
57				132776												389200

W/C	26	27	28	29	33	39	40	41	42	48	50	52	53	54	55	56	57
1	365806	118398	35252								35350						521514
2									462								
3																	
4														198114	297976		
6																5194	
7																	
8									8008					10108	5194		
9																	118398
10	193354			20706		8540											40096
11																	
12	75012																15428
16																	
17				28014		212198										76146	
21																	
22													39256		122472		
25												389200					
26		35350		32242										28014			343728
27										67494					35350		
28		32242									35252			4522			
29	28014		80934											16184			
33								462									
39							220738										
40	26502								15428								8540
41														462			8008
42					462			23436									
48															741678		
50	35350	35252															
52										706426							
53				39256											40096		
54				22134									5194		212506		101542
55																	
56																	48132
57										146412		317226	40096	140644	75880		

Figure 6.5: Q-Type (production quantity) From-To chart

6.3.4 Product Family Formation

In addition to the visualization of the complexity of material flow networks, an important capability of PFA is product family formation. Thereby, PFA simplifies the material flow network by segmenting a complex material flow network into smaller subnetworks for families of similar products. Finding product families hidden in a current product mix is a very important application of PFA. In PFAST, a product family is defined as a group of products that have similar routings containing a group of machines to perform the operations needed by the product family. PFAST identifies and groups similar products based on their routings using a matrix clustering technique as well as a hierarchical clustering technique.

Cluster Analysis is one of the most recognized techniques for product family and cell formation in Group Technology used in PFAST. This technique clusters objects into several groups based on their features. When applying the clustering technique in PFA, parts or products and machines are formed into part families and manufacturing machine cells based on criteria such as product or machine similarities. In order to logically group the products and machines into a number of cells, a product family identification and cell formation problem has to be solved. The most general formulation of the cell formation is matrix clustering using row and column permutations.

The matrix formulation for PFA constructs a matrix called a part-machine (or product-machine) matrix that represents the relationship between parts and machines as captured in their routings. The entries in this matrix are binary digits, 0 or 1. (Zero entries in part-machine matrices are normally represented by blank entries.) An element (i,j) of the matrix is "1" if machine i is used by part j, "0" otherwise. When using the matrix clustering method, an initial matrix with unstructured entries is transformed to a more structured form called a block diagonal form. Figure 6.6 shows the original unsorted matrix and sorted part-machine matrices using matrix clustering technique where product families can be derived from the diagonal blocks.

Ideally, each product family should contain products that require a set of machines and each set of machines that does not overlap with any other family, i.e. each machine set should be unique to any one corresponding product family. However, in practice, there may be some overlaps of machines between two or more product families. For example, in Figure 6.7, part family 1 contains products 1, 6, and 3 and utilizes machines 2, 8, 4, and 1. However, the element (1, 8) of the matrix for machine 1 and product 8 technically belongs to part family 2. However, it also requires machine 1 which is routinely used to manufacture the parts in part family 1. Similarly, for the element (10, 2) of the matrix for product 2, it shares machine 10 which is mainly used for the parts in the part family 6. These overlapping machines are called shared machines since they are being shared among different part (or product) families. The more the number of machines is shared in cellular manufacturing layouts, the less efficiency can be expected because operators in a cell will get interrupted by parts or products that do not belong to its product family. Thus, one of the challenging tasks for designing a cellular layout is to minimize intercell flows and cell overlaps due to the sharing of key machines.

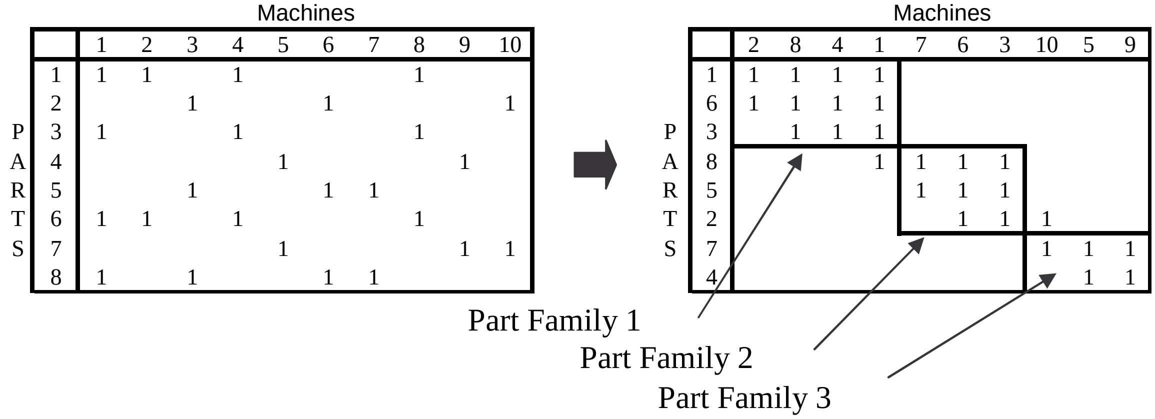


Figure 6.6: Unsorted and sorted part-machine matrices

In PFAST, a part-machine matrix is generated by an algorithm called PR-I Analysis that uses a matrix clustering heuristic. The result of this algorithm for the case study is the part-machine matrix in Figure 6.7. In addition to PR-I Analysis, PR-II Analysis uses a hierarchical clustering to produce the dendogram this figure. The dendogram provides the analyst a different view to identify the product families. In the dendogram, a vertical line shows a group of parts that are grouped into a cluster. The position of the line represents the similarity among the clustered parts. The closer vertical line is to the right hand side, the more similar are in the parts in the cluster. For example, the first vertical line in the dendogram in this figure is the line that clusters the first 13 parts because all of them use the same set of machines 55, 57, 25, 52, 48 and 1. As their lines move to the left, the part, as clusters of parts, being grouped then to be dissimilar.. When different product families share a large number of machines, it becomes difficult to visualize diagonal blocks to create independent product families using just the part-machine matrix. The dendogram provides a better structural view of cluster relationships to help analysts define the boundaries between product families if the part-machine matrix representation is ambiguous.

In PFAST, the 0-1 relationships between machines and parts is the only attribute used to construct both the part-machine matrix and dendogram. There are numerous other attributes such as operation sequences, setup cost/times, cell/part family size constraints, material handling costs, machine capacity constraints, machine investment cost that must be used to determine the product families and machine cells. These attributes are beyond the 0-1 part-machine matrix and the tree-like dendogram to capture. PFAST needs the analysts to use experience and judgment, something that no algorithm for product family formation could ever do.

In Figure 6.7, based on visual and experimental decisions, four products families can be formed with several shared machines. The four product families may need to be re-organized later to suit final layout since the family compositions play a significant role in the design of a cellular layout or flowline layout, as described in the next section. At this point, the identification of potential product families and the machine sharing relationships among these families is essential to even proceed to the next step in the layout design process.

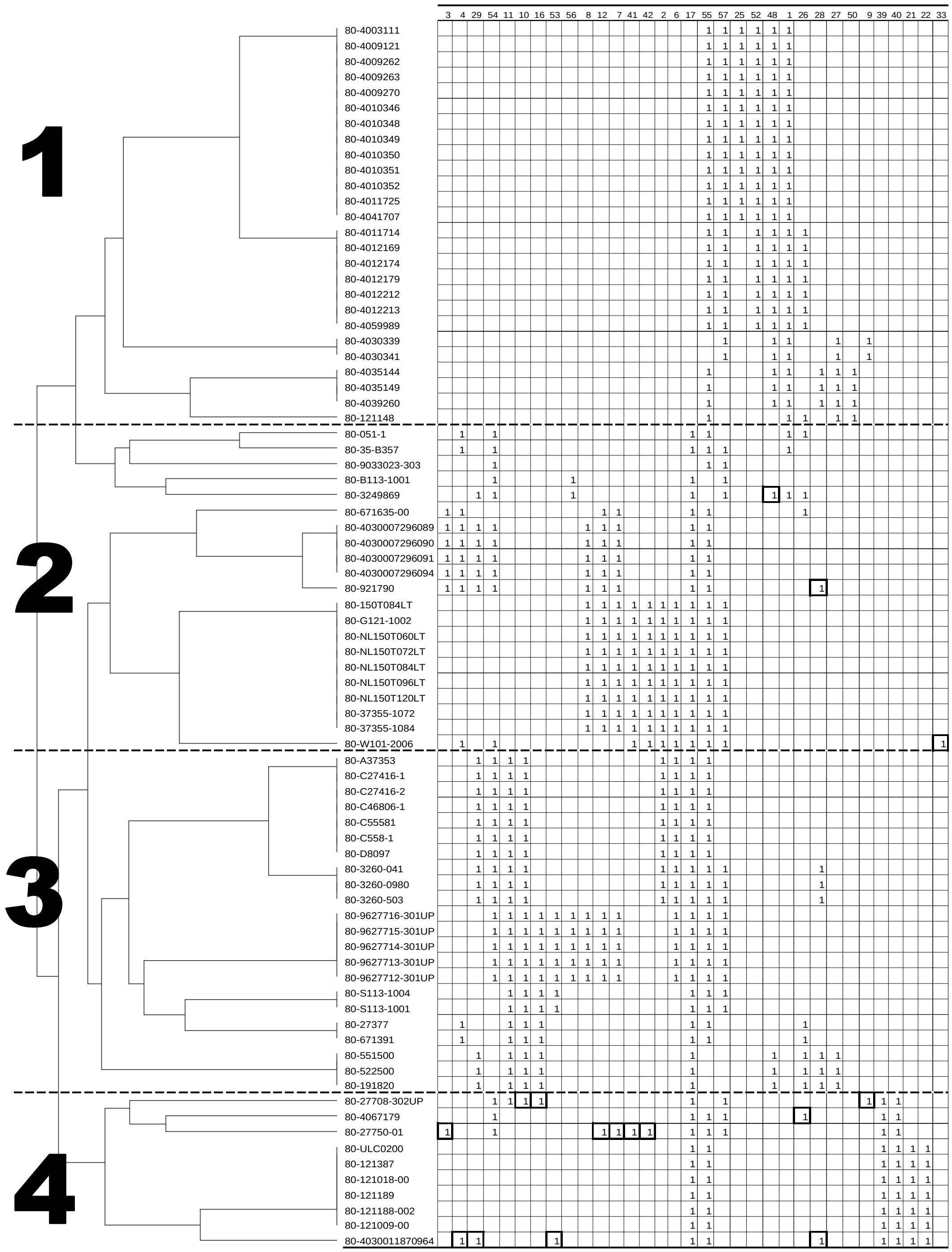


Figure 6.7: Product family identification using PR-I & PR-II

6.3.5 Layout Design

Using the combination of PR-I and PR-II Analyses, the analysts can quickly see whether product families exist, partially exist, or none exist. If there is no evidence of the existence of product families, then functional layout with updating the relocation of the machines with high traffic can be the best choice. If there is only one dominant family of products, then a flowline layout may be the most appropriate choice. If several potential product families are obvious, then cellular manufacturing with machine sharing or duplication can be a solution. When product families are observed but cannot be clearly distinguished as several machines may be heavily shared by different product families, then hybrid layouts that are a combination of functional layout, cellular layout, and flowline layout can be a solution. Especially in the case non-traditional layouts such as modular layouts and hybrid flowshop layouts can also be considered.

In PFAST, there are algorithms that helps to design five types of layouts—Functional Layout, Cellular Layout, Flowline Layout, Modular Layout, and Hybrid Flowshop Layout. The following section describes these layout design algorithms in detail.

6.3.5a Design of Functional Layout

The problem of designing a functional layout is, in its simplest-form, a Quadratic Assignment Problem (QAP) that can be formulated as follows:

$$Min \sum_{i} \sum_{j} c_{ij} . f_{ij} . d_{ij}$$

where

c_{ij} = the cost of moving one unit load per unit distance from department i to department j,

f_{ij} = the flow or frequency of materials move from department i to j normally measured in a number of trips per a period of time, and

d_{ij} = the distance of moving materials from department i to j.

Functional Layouts are convenient whether product mix and production volumes change because customer requirements change frequently especially for HMLV manufacturers. PFAST contains a block layout algorithm to design a functional layout. PFAST uses a design skeleton approach coupled with a Genetic Algorithm to solve this problem heuristically. While there is no guarantee of optimal solutions, PFAST provides good solutions in reasonable time, especially when working with large and complex datasets.

The block layout for this case study produced by PFAST is shown in Figure 6.8 where machines with heavy traffic between them are placed next to each other as much as possible. The thick arrows in the block layout represent the high volumes of material flow. Thereafter, a company if they can relocate the high traffic machines since it is usually costly to change an existing layout.

In the case study, it is clear that there are several pairs of high traffic machines that are located far apart. For example, machine 1 and 26 are located in different buildings. This pair of machines is a good example to consider relocating them in order to reduce the travel distance between them since that will eventually result in reduction of material handling and WIP costs incurred by the current layout.

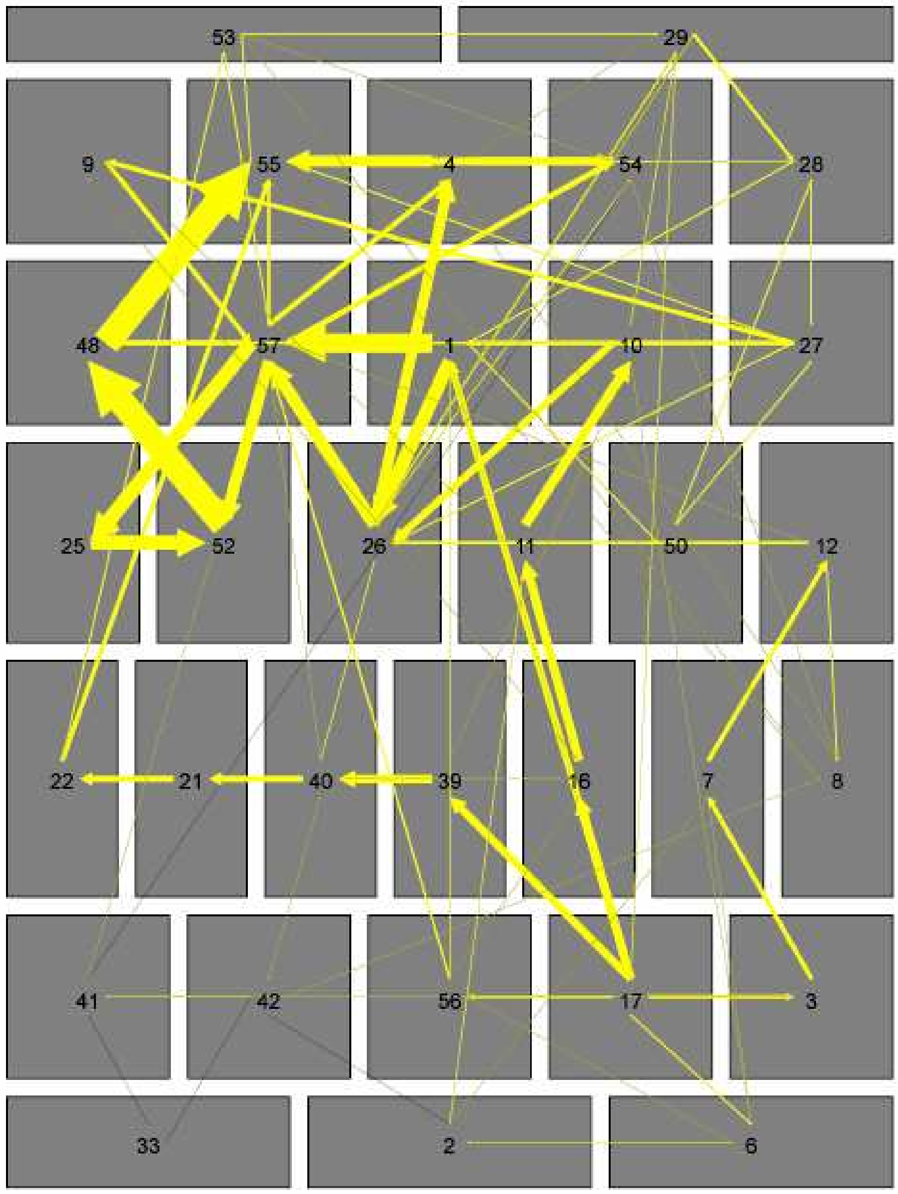


Figure 6.8: Block layout

6.3.5b Design of Cellular Layout

If product families exist, manufacturing companies can seek to replace a portion of their current functional layout by a cellular layout. A cellular layout is best implemented in the following situations: (1) if one or more distinct product families can be identified, (2) if the number of exception operation and machines shared by two of more cells is minimal, and (3) if the cost of duplication of machines of the same type among multiple cells does not exceed the capital investment budget. In this case study, the four product families were identified; however, these product families require several machines to be shared among them ex machines 17, 55, and 57 as shown in Figure 6.7. Hence, it is not possible to completely convert the current layout into a cellular layout unless many machines are duplicated, some machines are moved, and buildings consolidated. The feasible allocation is to implement a manufacturing cell for the most distinct product family and produce the remaining products using the current functional layout. From Figure 6.7, product family 1 is the best candidate for producing in a cell.

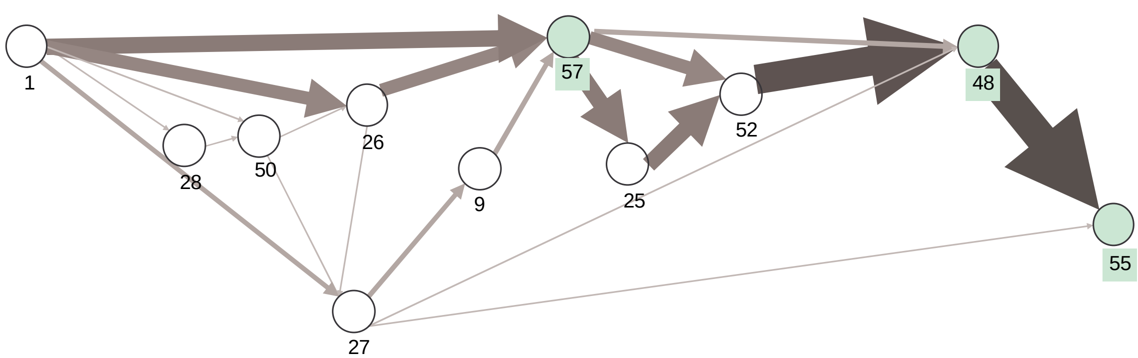


Figure 6.9: Flow diagram for product family 1

Figure 6.9 shows the Flow Diagram for product family 1. This product family does not contain monument machines or expensive machines (red colored machines) so it is quite possible to setup a manufacturing cell to produce this product family. There are some heavily shared machines that merit attention. These machines are machines 55 and 57. Machine 55 is used for the least operation in the routings for product family1. Therefore, interruptions in product family 1 resulted by machine 55 being shared with other product families can be neglected. However, machine 57 is used in the middle of the routings of product family 1; therefore, this machine may need to be duplicated to contain the flows of the parts in this product family in its manufacturing cell.

For this case study, Figure 6.10 shows a possible implementation of a manufacturing cell in the current layout. If feasible, there are two machines that need to be relocated or duplicated. Machine 1 and machine 9 need to move from building #1 to building #2. There are also three machines/operations that are external operations meaning that these products or parts will need to be manufactured outside the facility and then return to the cell for the remaining operations in their routings. To prevent delays and flow interruptions because of the external operations, the company may need to consider bringing back some of these outsourced operations and doing them inside the cell. If that is not an option, then a two-bin kanban or some type of buffer inventory may need to be established to keep all the operations in the cell running efficiently.

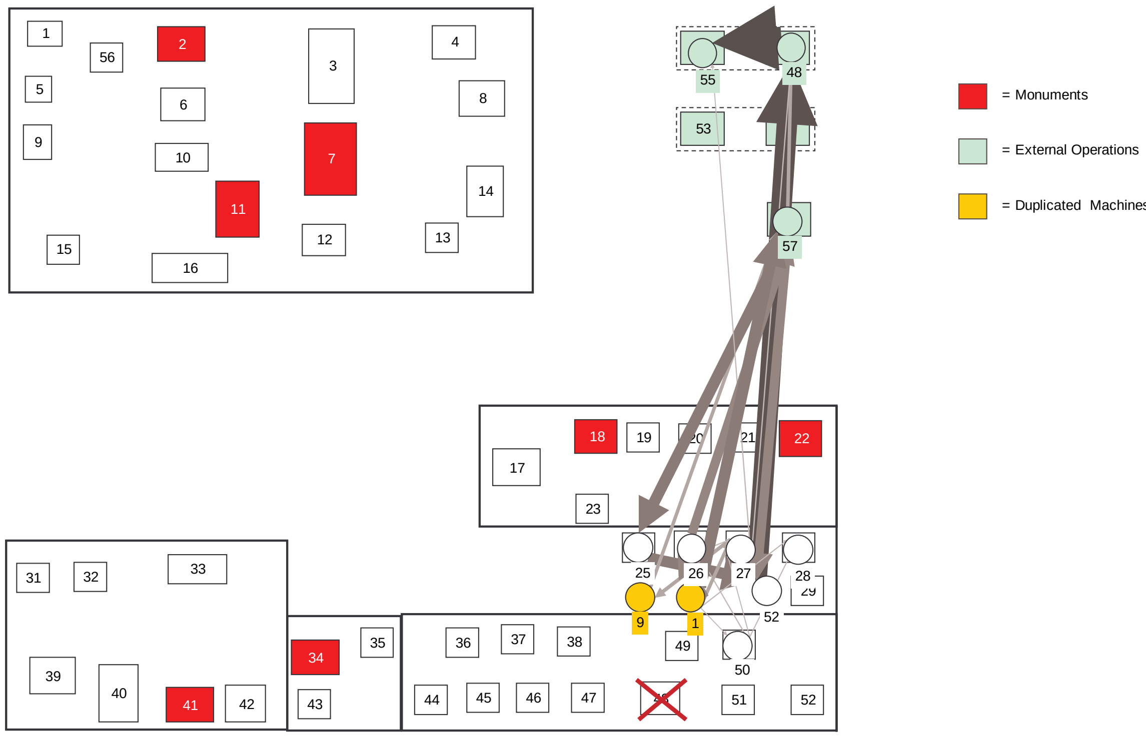


Figure 6.10: A manufacturing cell for product family 1

6.3.5c Design of Flowline Layout

Flowline layouts are the least preferred type of layout for HMLV manufacturing facilities. In order to implement and operate flowline layouts efficiently, manufacturing companies need a stable product mix, stable demand, and high production volume. It can be very difficult to find any HMLV facilities that are suited for flowline layouts. However, there can be a possibility that there is a portion of products or a product family that have high production volume and identical, or very similar, routings. In this situation, a flowline layout and can be designed for the cell to produce this product family.

In order to find a product family that is suited for a flowline layout, all product families needed to be first identified and mapped. Once the product families are identified, the flows of these product families need to be mapped as shown in Figure 6.11. The mapping of the flow network for each product family shows the linearity of flows ad number of machines in the cell. In this case study, it can be noticed that product families 3 and 4 have the majority of flows streamlined and in the forwarded direction. However, production volumes for these two product families are very low. It may not be economically feasible to dedicate a set of machines to implement a flowline layout to produce either of these product families. Instead, product family 1 seems to have the largest amount of production volumes, about 50% of total volume, and all major flows of its products are simple and forward. Therefore, this product family should be examined for the feasibility of implementing a flowline layout. However, before proceeding to a design process, unmanageable flows such as backtracking flows should be removed. Such flows prevent flowline layouts from operating efficiently. The process for identifying and removing such unmanageable flows can be done using PR-IV Analysis in PFAST.

PR-IV Analysis does an alignment of product routings in such a way that misfit routings in the product family can be identified easily. These misfit routings that create the non-conforming flows can be recognized and removed from the product families as shown in Figure 6.12. When all the misfit routings are removed, there are 20 products left and this set of products requires about 7 machines. Once all unmanageable flows are removed and the remaining flows are unified, a flowline layout for this product family can be implemented as shown in Figure 6.13 assuming that machine 57 is brought in to the facility to smoothen the operations in the flowline.

The flowline layout implemented for this case study produces 20 products that account for 706,426 units of production volume, which is about 40% of total production volume. If any HMLV facility can partition their complex production system to be able to manage 40% of their product mix efficiently using flowline production, this would be a tremendous improvement in their existing operations.

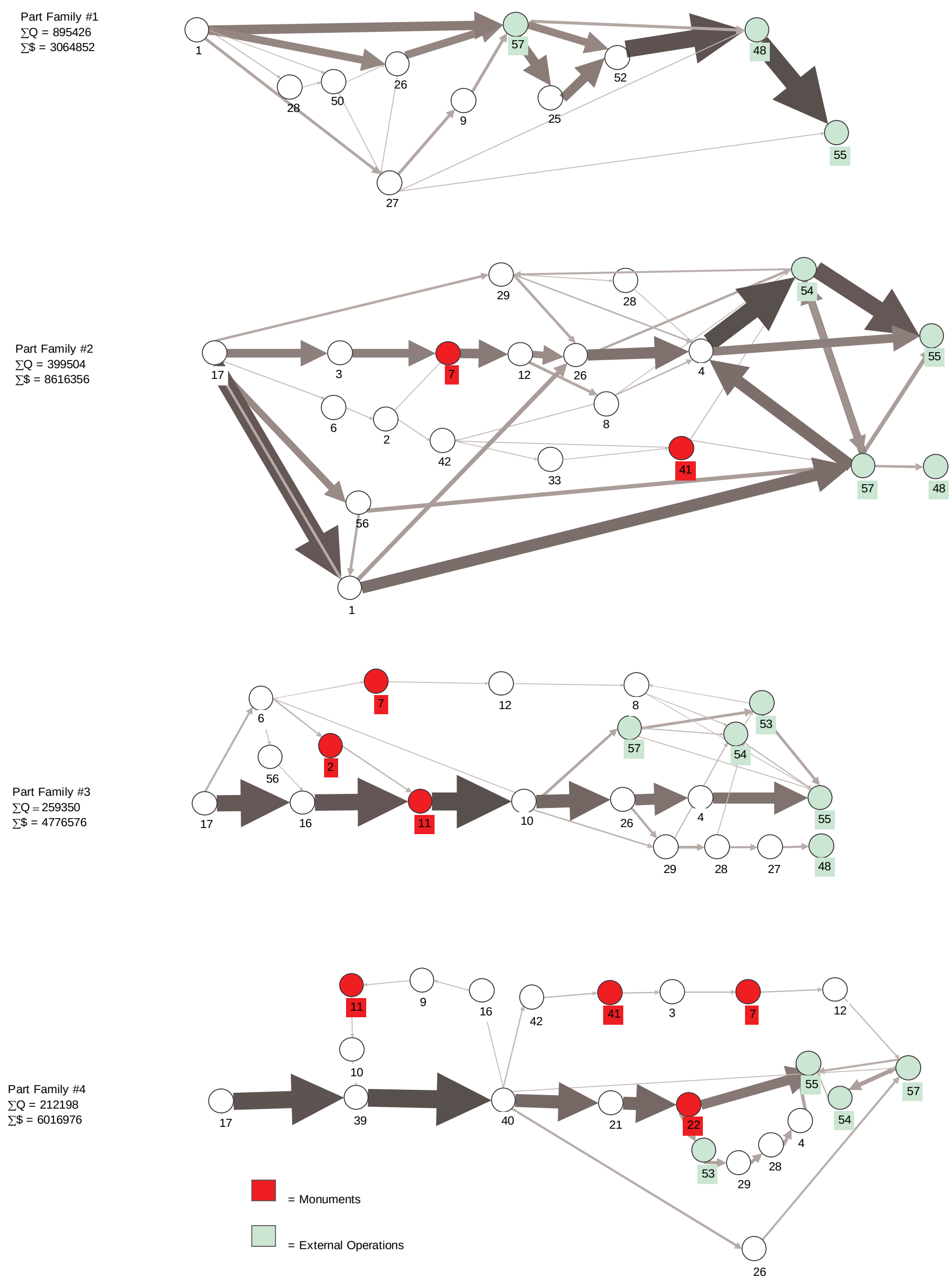


Figure 6.11: Flow mapping diagrams for each product family

Part										
80-121148	1					50	26	27		55
80-4035144	1		28			50		27	48	55
80-4035149	1		28			50		27	48	55
80-4039260	1		28			50		27	48	55
80-4003111	1					57	25	52	48	55
80-4009121	1					57	25	52	48	55
80-4009262	1					57	25	52	48	55
80-4009263	1					57	25	52	48	55
80-4009270	1					57	25	52	48	55
80-4010346	1					57	25	52	48	55
80-4010348	1					57	25	52	48	55
80-4010349	1					57	25	52	48	55
80-4010350	1					57	25	52	48	55
80-4010351	1					57	25	52	48	55
80-4010352	1					57	25	52	48	55
80-4011725	1					57	25	52	48	55
80-4041707	1					57	25	52	48	55
80-4011714	1	26				57		52	48	55
80-4012169	1	26				57		52	48	55
80-4012174	1	26				57		52	48	55
80-4012179	1	26				57		52	48	55
80-4012212	1	26				57		52	48	55
80-4012213	1	26				57		52	48	55
80-4059989	1	26				57		52	48	55
80-4030339	1			27	9	57			48	
80-4030341	1			27	9	57			48	

Misfit Routings

Figure 6.12: PR-IV for product family 1 and misfit routings

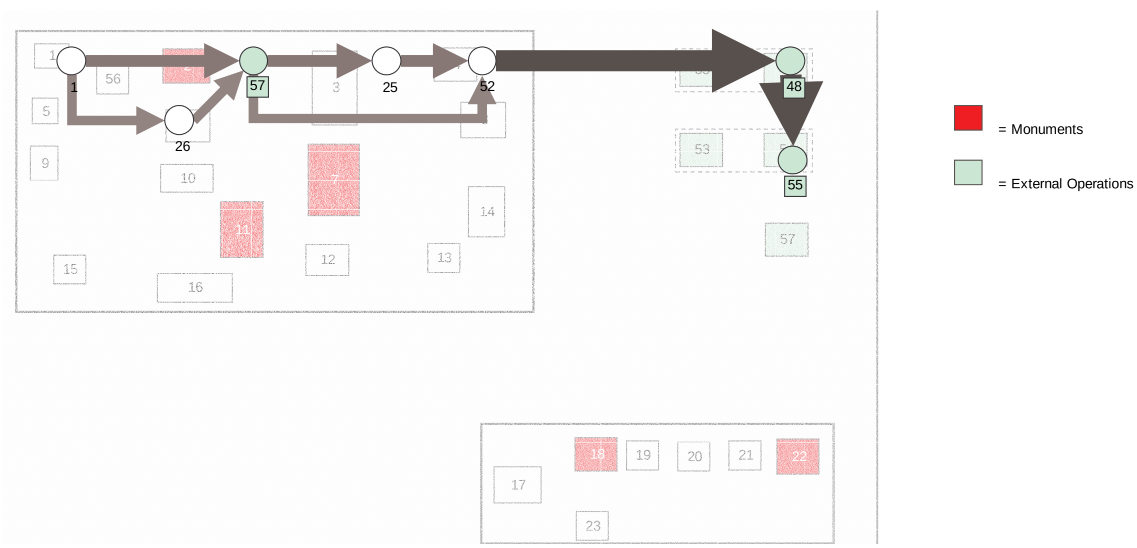


Figure 6.13: The implementation of the flowline layout over the current layout

6.3.5d Design of Modular Layout

The approach for designing a modular layout using PFAST is as follows:

1. Define sets of strongly connected machines using PR-III Analysis as prospective layout modules.
2. Replace the original substrings of operations in product routings with layout modules.
3. Map the new routings on the current layout with the machines grouped into modules.

The first step to define a set of strongly connected machines can be done by PR-III Analysis, as shown in Figure 6.14, which shows different substrings of operations that occur in manufacturing routings. The first column in indicates the size of each substring ranging from 2 machines to at most the length of the longest routing, which is 15 in this case study. The frequency column is the number of occurrences of each substring. The percentages columns indicate the product quantity and revenue proportion contributed by each substring. The dendogram, on the right-hand-side of the table is essentially a PR-II Analysis of the common substrings of operations instead of the complete routings. Figure 6.15 shows examples of layout modules generated by merging sets of substrings together based on their sequence similarities, quantity, and revenue of products being processed by these sets of common machines. The red-dashed ovals in this figure represent final layout modules that are going to be implemented. The main factors that drive the selection of the final layout modules are the size of each layout module, availability of machine to allocate to the modules, and cost of machine duplication.

Once the layout modules are generated, the original product routings will need to be modified. For example, part 80-4035144 in this case study has its routing as 1→28→50→27→48→55. Using the final layout modules obtained earlier, there are two available layout modules—module (29-28) and (48-55)—that can replace some of the machines that appear in the routing. The new routing for part 80-4035144 becomes 1→(29-28)→50→27→(48-55). Figure 6.16 shows all routings updated to show the sets of machines in each routing aggregated into layout modules.

After updating the original product routings, the next step is to map the new routings and layout modules on the current layout. The modular layout that could be implemented using the updated routings and showing the layout modules mapped on the current layout is shown in Figure 6.17.

It is obvious that material flow network in the modular layout is much cleaner and clearer then the original layout. Most of the low-volume crisscrossing flows in the original layout have been eliminated. This reduction in the complexity of material flows in a functional layout can always be expected in a modular layout.

In this case study, the modular layout needs only one machine, machine 17, to be duplicated and the remaining machines requiring to be relocated into layout modules. Based on standard distance calculations, the total travel distance reduces from 2.57×10^9 in the current layout to 0.77×10^9 in the modular layout, i.e. a significant improvement (70% reduction in traveling distance) that can be expected from this layout change. The detailed calculations of total distance scores are described in Appendix C.

Length	Frequency	% Qty.	% Rev.	Substring
2	6	3.82	1.09	27->48
6	3	2.00	0.48	1->28->50->27->48->55
2	23	41.99	11.85	48->55
3	20	39.99	11.37	52->48->55
6	7	17.96	4.32	1->26->57->52->48->55
2	3	8.29	1.95	57->48
5	2	6.70	1.15	1->27->9->57->48
2	14	29.52	10.15	1->57
6	13	22.03	7.05	1->57->25->52->48->55
2	2	7.52	5.09	57->4
2	8	19.46	7.69	26->57
2	10	7.96	28.16	57->54
2	12	5.75	20.73	54->57
3	5	3.84	10.92	54->57->55
4	2	3.59	7.48	57->54->57->55
2	10	12.03	14.82	54->55
2	14	4.30	25.76	57->55
10	9	0.45	14.83	17->6->2->7->12->8->42->41->57->55
4	2	2.50	9.54	29->28->4->55
2	8	4.58	13.61	29->28
2	5	1.25	9.48	54->29
2	6	11.22	11.38	4->54
3	2	10.24	4.24	4->54->55
2	10	16.87	23.98	4->55
3	3	13.37	5.31	26->4->55
2	4	16.12	6.45	26->4
2	8	20.71	5.46	1->26
10	4	0.97	7.14	17->3->7->12->8->4->54->29->4->55
5	5	1.25	9.48	17->3->7->12->8
4	6	5.50	11.56	17->3->7->12
3	7	6.37	15.70	3->7->12
2	21	7.12	37.55	7->12
3	19	2.00	31.33	7->12->8
4	6	0.57	9.36	7->12->8->54
16	5	0.29	7.02	17->6->56->16->11->10->6->7->12->8->54->57->54->53->8->55
9	3	1.83	0.62	17->16->11->10->26->29->28->27->48
2	23	15.17	29.94	11->10
3	12	13.51	11.37	16->11->10
4	7	13.22	4.34	17->16->11->10
5	5	10.95	3.85	17->16->11->10->26
7	2	9.12	3.24	17->16->11->10->26->4->55
7	2	2.27	0.49	17->16->11->10->57->53->55
10	3	0.26	3.45	17->6->2->11->10->29->28->54->57->55
2	25	1.95	33.72	17->6
3	20	1.65	26.70	17->6->2
6	10	1.17	9.89	17->6->2->11->10->29
8	7	0.92	6.44	17->6->2->11->10->29->54->55
2	2	4.31	1.63	17->56
2	2	10.24	4.24	17->1
6	6	6.93	3.37	17->39->40->21->22->55
5	7	9.16	10.57	17->39->40->21->22
3	10	12.01	26.77	17->39->40
2	10	12.01	26.77	39->40
2	10	1.33	18.98	42->41

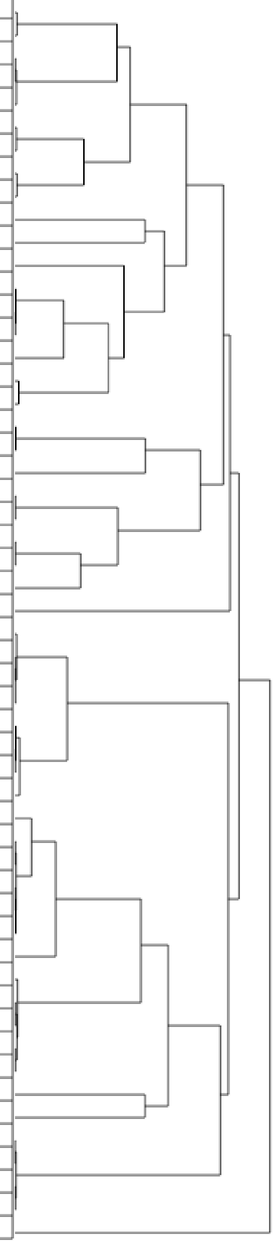


Figure 6.14: PR-III Result

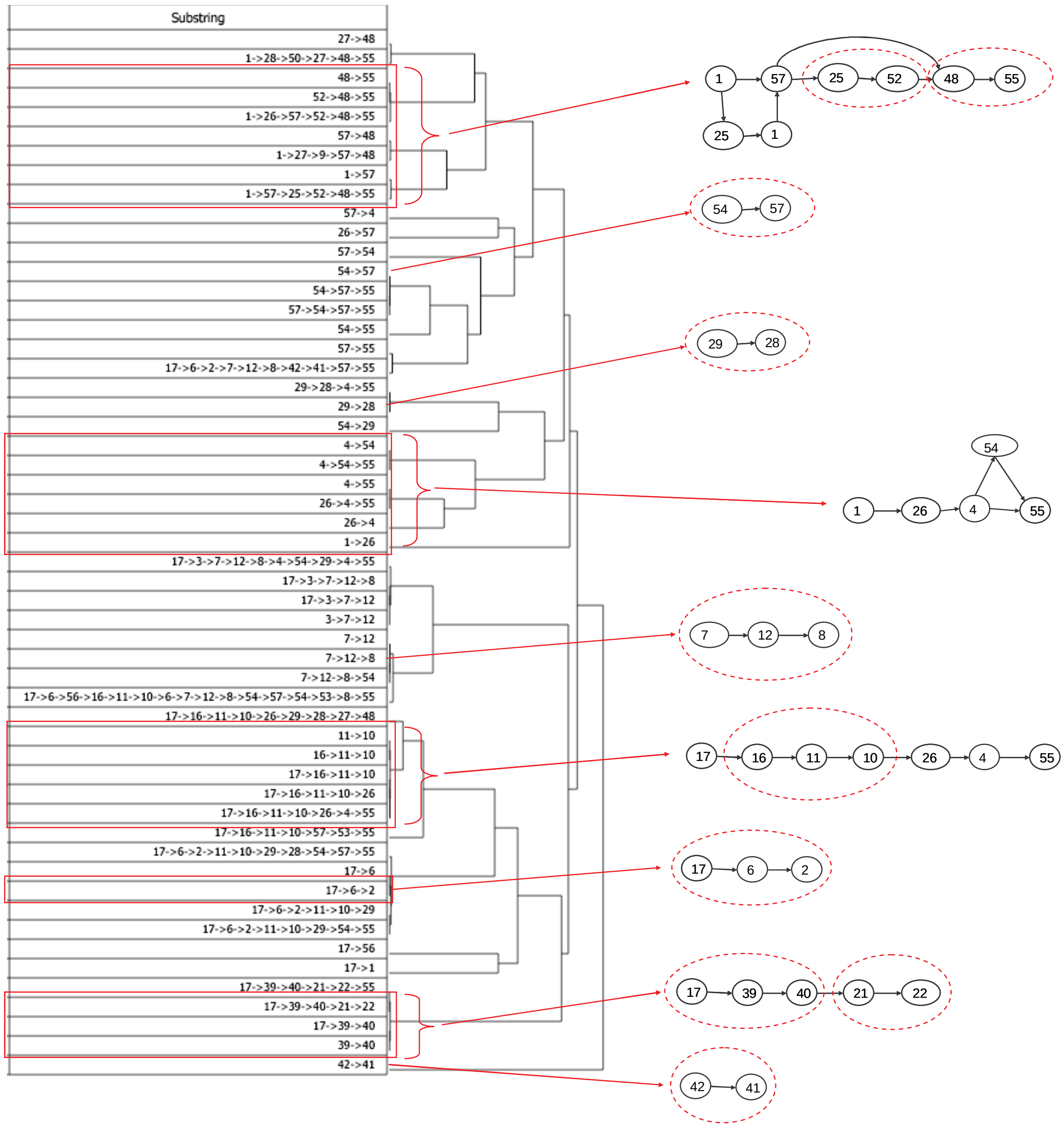


Figure 6.15: Layout module formation using PR-III

Parts									
80-4035144	1	29-28	50	27	48-55				
80-4035149	1	29-28	50	27	48-55				
80-4039260	1	29-28	50	27	48-55				
80-4003111	1	54-57	25-52	48-55					
80-4009121	1	54-57	25-52	48-55					
80-4009262	1	54-57	25-52	48-55					
80-4009263	1	54-57	25-52	48-55					
80-4009270	1	54-57	25-52	48-55					
80-4010346	1	54-57	25-52	48-55					
80-4010348	1	54-57	25-52	48-55					
80-4010349	1	54-57	25-52	48-55					
80-4010350	1	54-57	25-52	48-55					
80-4010351	1	54-57	25-52	48-55					
80-4010352	1	54-57	25-52	48-55					
80-4011725	1	54-57	25-52	48-55					
80-4041707	1	54-57	25-52	48-55					
80-4011714	1	26	54-57	25-52	48-55				
80-4012169	1	26	54-57	25-52	48-55				
80-4012174	1	26	54-57	25-52	48-55				
80-4012179	1	26	54-57	25-52	48-55				
80-4012212	1	26	54-57	25-52	48-55				
80-4012213	1	26	54-57	25-52	48-55				
80-4059989	1	26	54-57	25-52	48-55				
80-4030339	1	27	9	54-57	48-55				
80-4030341	1	27	9	54-57	48-55				
80-191820	17-6-2	16-11-10	26	29-28	27	48-55			
80-522500	17-6-2	16-11-10	26	29-28	27	48-55			
80-551500	17-6-2	16-11-10	26	29-28	27	48-55			
80-27377	17-6-2	16-11-10	26	4	48-55				
80-671391	17-6-2	16-11-10	26	4	48-55				
80-671635-00	17-6-2	3	7-12-8	26	4	48-55			
80-921790	17-6-2	3	7-12-8	54-57	29-28	4	48-55		
80-4030007296094	17-6-2	3	7-12-8	4	54-57	29-28	4	48-55	
80-4030007296091	17-6-2	3	7-12-8	4	54-57	29-28	4	48-55	
80-4030007296090	17-6-2	3	7-12-8	4	54-57	29-28	4	48-55	
80-4030007296089	17-6-2	3	7-12-8	4	54-57	29-28	4	48-55	
80-150T084LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-37355-1072	17-6-2	7-12-8	42-41	54-57	48-55				
80-37355-1084	17-6-2	7-12-8	42-41	54-57	48-55				
80-G121-1002	17-6-2	7-12-8	42-41	54-57	48-55				
80-NL150T060LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-NL150T072LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-NL150T084LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-NL150T096LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-NL150T120LT	17-6-2	7-12-8	42-41	54-57	48-55				
80-D8097	17-6-2	16-11-10	29-28	54-57	48-55				
80-C558-1	17-6-2	16-11-10	29-28	54-57	48-55				
80-C55581	17-6-2	16-11-10	29-28	54-57	48-55				
80-C46806-1	17-6-2	16-11-10	29-28	54-57	48-55				
80-C27416-2	17-6-2	16-11-10	29-28	54-57	48-55				
80-C27416-1	17-6-2	16-11-10	29-28	54-57	48-55				
80-A37353	17-6-2	16-11-10	29-28	54-57	48-55				
80-3260-503	17-6-2	16-11-10	29-28	54-57	48-55				
80-3260-0980	17-6-2	16-11-10	29-28	54-57	48-55				
80-3260-041	17-6-2	16-11-10	29-28	54-57	48-55				
80-9627712-301UP	17-6-2	56	16-11-10	17-6-2	7-12-8	54-57	53	7-12-8	48-55
80-9627713-301UP	17-6-2	56	16-11-10	17-6-2	7-12-8	54-57	53	7-12-8	48-55
80-9627714-301UP	17-6-2	56	16-11-10	17-6-2	7-12-8	54-57	53	7-12-8	48-55
80-9627715-301UP	17-6-2	56	16-11-10	17-6-2	7-12-8	54-57	53	7-12-8	48-55
80-9627716-301UP	17-6-2	56	16-11-10	17-6-2	7-12-8	54-57	53	7-12-8	48-55
80-S113-1001	17-6-2	16-11-10	54-57	53	48-55				
80-S113-1004	17-6-2	16-11-10	54-57	53	48-55				
80-121009-00	17-39-40	21-22	48-55						
80-121018-00	17-39-40	21-22	48-55						
80-121188-002	17-39-40	21-22	48-55						
80-121189	17-39-40	21-22	48-55						
80-121387	17-39-40	21-22	48-55						
80-ULC0200	17-39-40	21-22	48-55						
80-9033023-303	54-57	48-55							

Figure 6.16: Layout module replacement in the routings

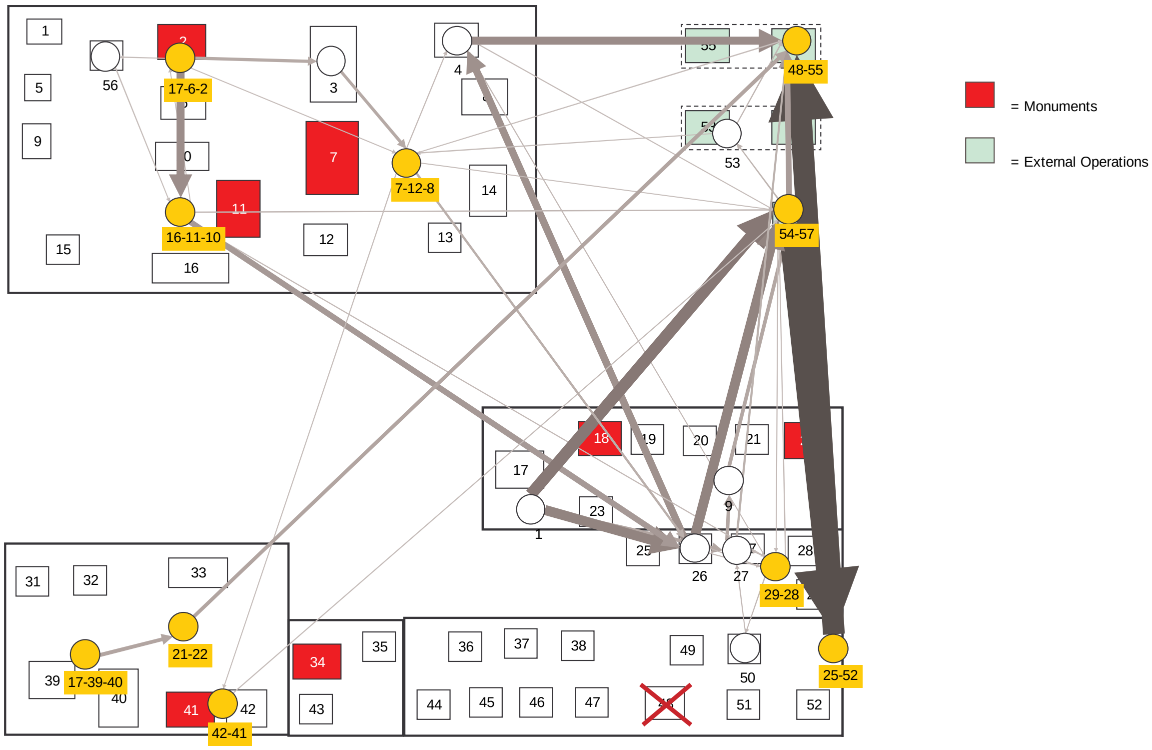


Figure 6.17: Material flow in the modular layout

6.3.5e Design of Hybrid Flowshop Layout

PR-IV Analysis in PFAST as shown in Figure 6.18 is used to design a hybrid flowshop layout. This analysis, also called MM-PPC, is a table-like chart that has part numbers listed in the first column and the operations in the routing of each part listed in each row according to the part number. The most interesting feature of this chart is that the parts that have similar operation sequences are placed next to each other and machines/operations that are identical are placed in the same column as much as possible.

Once the MM-PPC is generated, parts that create misfit routings as highlighted in gray color shown in the chart need to be removed. These misfit routings can cause unnecessary machine duplications in a final flowshop layout. These misfits can be recognized easily, for example, part “80-W101-2006” is the only part in the product mix that requires machine #33. If this part is not removed, a final flowshop layout must need an extra machine #33 for this part only. Therefore, misfit routings like this need to be removed.

Once all the misfit routings have been removed and MM-PPC chart has been updated, the next step is to form the stages from the updated MM-PPC chart. The stages should be generated under the following criteria:

1. Each stage has its size constraint meaning that it can contain only a limited number of machine types.

2. A final flowshop layout should have a minimum number of machines duplicated.

3. A part performing an operation in its current stage can perform its next operations only in the current stage or the succeeding stages. It cannot go back to perform its next operations in any preceding stages.

Supposed that each stage can contain no more than 4 machines in this case study, Figure 6.19 shows the results of the flowshop layout design using the updated MM-PPC chart. There are three machines that need to be duplicated including machines 6, 4 and 54. With these three machines duplicated, the flow diagram of the final hybrid flowshop layout can be drawn as shown in Figure 6.20. As can be seen from the figure, the hybrid flowshop layout can transform such a complex, crisscrossing flows in the functional layout into a very simple flowline-like layout.

Figure 6.21 shows the hybrid flowshop layout implemented on the current layout. When comparing the hybrid flowshop layout with the modular layout for this case study, the hybrid flowshop layout seems to provide much simplified flows within its facility. Although, the hybrid flowshop layout may not outperform the modular layout in terms of traveling distance reduction in this case study, the simplicity of the material flow network that leads to the ease of scheduling control should be expected in a hybrid flowshop layout, compared to a functional layout and a modular layout.

Parts																														
80-121148	1																								50	26	27		55
80-4035144	1														28										50		27	48	55
80-4035149	1														28										50		27	48	55
80-4039260	1														28										50		27	48	55
80-4003111	1																					57				25	52	48	55
80-4009121	1																					57				25	52	48	55
80-4009262	1																					57				25	52	48	55
80-4009263	1																					57				25	52	48	55
80-4009270	1																					57				25	52	48	55
80-4010346	1																					57				25	52	48	55
80-4010348	1																					57				25	52	48	55
80-4010349	1																					57				25	52	48	55
80-4010350	1																					57				25	52	48	55
80-4010351	1																					57				25	52	48	55
80-4010352	1																					57				25	52	48	55
80-4011725	1																					57				25	52	48	55
80-4041707	1																					57				25	52	48	55
80-4011714	1										26											57					52	48	55
80-4012169	1										26											57					52	48	55
80-4012174	1										26											57					52	48	55
80-4012179	1										26											57					52	48	55
80-4012212	1										26											57					52	48	55
80-4012213	1										26											57					52	48	55
80-4059989	1										26											57					52	48	55
80-4030339	1																27	9					57						48	
80-4030341	1																27	9					57						48	
80-191820	17				16	11	10					26			29	28	27												48	
80-522500	17				16	11	10					26			29	28	27												48	
80-551500	17				16	11	10					26			29	28	27												48	
80-27377	17				16	11	10					26	4																	55
80-671391	17				16	11	10					26	4																	55
80-671635-00	17		3						7	12		26	4																	55
80-921790	17		3						7	12	8			54	29	28									4					55
80-4030007296094	17		3						7	12	8		4	54	29										4					55
80-4030007296091	17		3						7	12	8		4	54	29										4					55
80-4030007296090	17		3						7	12	8		4	54	29										4					55
80-4030007296089	17		3						7	12	8		4	54	29										4					55
80-150T084LT	17	6	2						7	12	8							42		41			57							55
80-37355-1072	17	6	2						7	12	8							42		41			57							55
80-37355-1084	17	6	2						7	12	8							42		41			57							55
80-G121-1002	17	6	2						7	12	8							42		41			57							55
80-NL150T060LT	17	6	2						7	12	8							42		41			57							55
80-NL150T072LT	17	6	2						7	12	8							42		41			57							55
80-NL150T084LT	17	6	2						7	12	8							42		41			57							55
80-NL150T096LT	17	6	2						7	12	8							42		41			57							55
80-NL150T120LT	17	6	2						7	12	8							42		41			57							55
80-W101-2006	17	6	2															42	33	41		54	57		4					55
80-D8097	17	6	2			11	10								29							54								55
80-C558-1	17	6	2			11	10								29							54								55
80-C55581	17	6	2			11	10								29							54								55
80-C46806-1	17	6	2			11	10								29							54								55
80-C27416-2	17	6	2			11	10								29							54								55
80-C27416-1	17	6	2			11	10								29							54								55
80-A37353	17	6	2			11	10								29							54								55
80-3260-503	17	6	2			11	10								29	28						54	57							55
80-3260-0980	17	6	2			11	10								29	28						54	57							55
80-3260-041	17	6	2			11	10								29	28						54	57							55
80-9627712-301UP	17	6		56	16	11	10	6	7	12	8											54	57	54		53	8			55
80-9627713-301UP	17	6		56	16	11	10	6	7	12	8											54	57	54		53	8			55
80-9627714-301UP	17	6		56	16	11	10	6	7	12	8											54	57	54		53	8			55
80-9627715-301UP	17	6		56	16	11	10	6	7	12	8											54	57	54		53	8			55
80-9627716-301UP	17	6		56	16	11	10	6	7	12	8											54	57	54		53	8			55
80-S113-1001	17				16	11	10																57			53				55
80-S113-1004	17				16	11	10																57			53				55
80-27708-302UP	17	39	40		16	9	11	10	39	40											57	54								
80-27750-01	17	39	40			42	41	3	7	12											57	54								55
80-4067179	17	39	40									26									57	54	57							55
80-3249869	17			56				1	17	29		26										54	57							48
80-051-1	17									1		26	4									54								55
80-35-B357	17									1	57		4									54								55
80-B113-1001	17			56							57											54								
80-121009-00	17	39	40				21	22																						55
80-121018-00	17	39	40				21	22																						55
80-121188-002	17	39	40				21	22																						55
80-121189	17	39	40				21	22																						55
80-121387	17	39	40				21	22																						55
80-4030011870964	17	39	40				21	22	53	29	28		4																	55
80-ULC0200	17	39	40				21	22																						55
80-9033023-303																					57	54	57							55

Figure 6.18: PR-IV and misfit routings

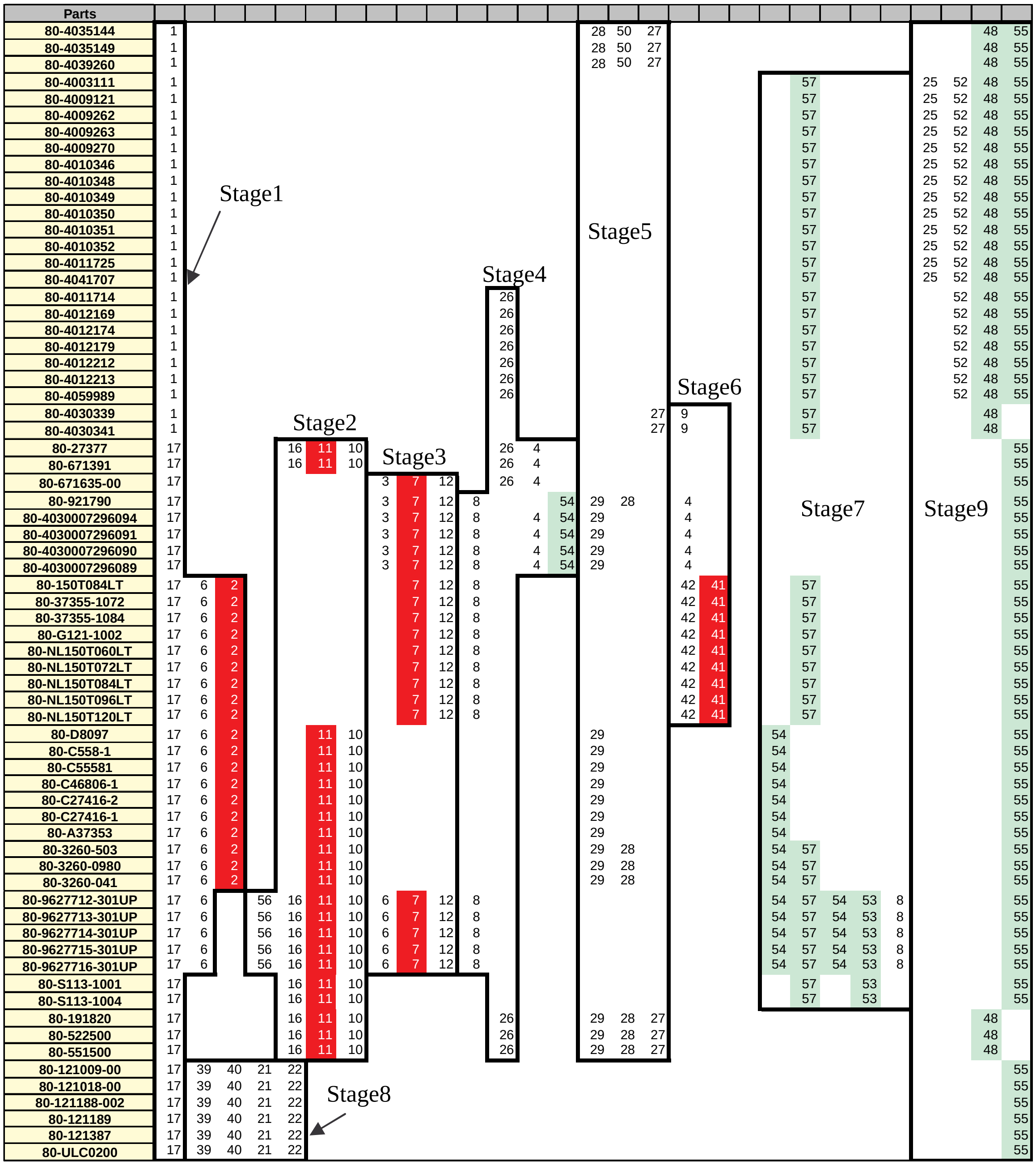


Figure 6.19: Flow shops formation using PR-IV

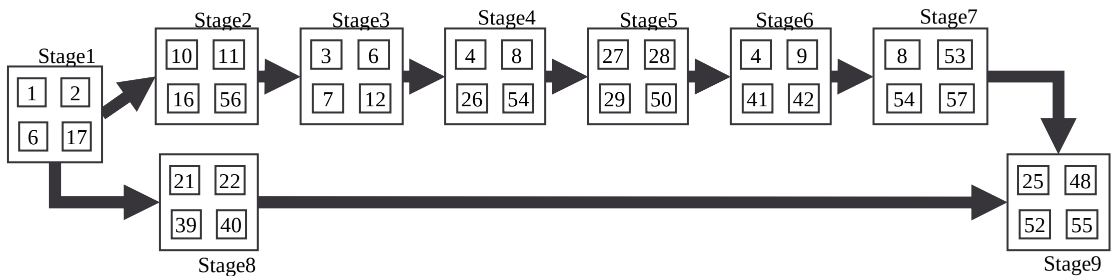


Figure 6.20: Flow Diagram of Hybrid Flowshop

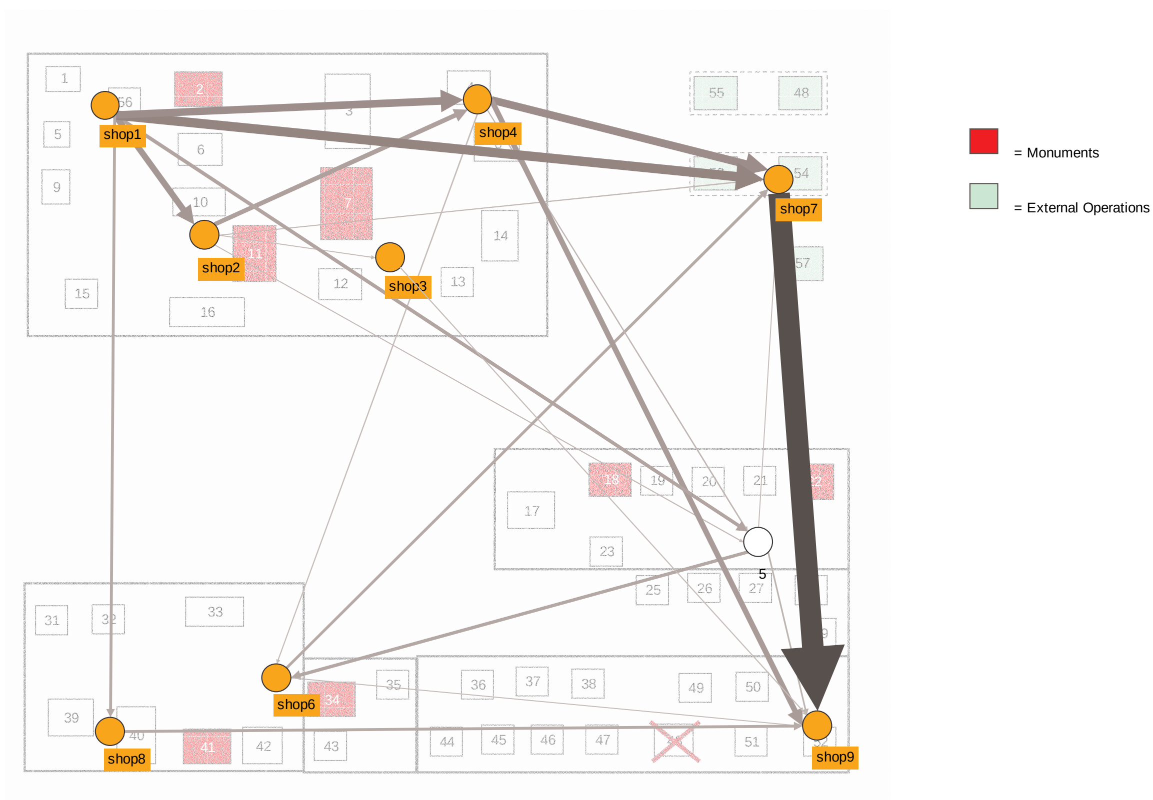


Figure 6.21: Hybrid flowshop layout implemented on the current facility

6.4 Conclusion

In this chapter, we introduced a real-world case of high-mix low-volume facility that manufactures a large variety of products with complex and dissimilar routings. The implementation of Lean Manufacturing to improve production activities and performance for a HMLV facility requires an efficient and appropriate layout to operate. Since HMLV facilities pose unique characteristics, they always give challenges to analysts for understanding their complex material networks and seeking the most efficient and appropriate layouts. To cope with these challenges, we introduced PFAST software that contains several PFA algorithms that can decompose and simplify the complex material networks in HMLV facilities. Then we gave the approach that allows analysts to appropriately select and design different layouts that are suited for these HMLV facilities. These layouts include (1) the three traditional layouts—Functional Layout, Cellular Layout, and Flowline Layout, (2) a mix of traditional layouts, and (3) the new conceptual modular layout and hybrid flowshop layout. The mind set of PFAST is to simplify the complex material flow networks and transform them to manageable flow networks. Once these complex material flow networks were organized, the layout design module in PFAST gives analysts the ability to select and design an efficient and appropriate layout that are suited for these particular material flow networks.

References

1. Burbridge, J.L., “Production Flow Analysis,” *Production Engineer*, Vol. 42, pp. 742-752 (1963).

2. Evans, J.R., *Applied Production and Operations Management*, West Publishing Company, St. Paul, MN (1983).

3. Fogarty, D.W., Hoffmann, T.R. and Stonebraker, P.W., *Production and operations management*, South-Western Pub. Co., Cincinnati, OH (1989).

4. Ham, I., Hitomi, K., and Yoshida, T., *Group Technology*, Kulwer-Nijhoff Publishing, Boston, MA (1985).

5. Harhalakis, G., Nagi, R. and Proth, J. M., “An Efficient Heuristic in Manufacturing Cell Formation for Group Technology Application.” *International Journal of Production Research*, Vol. 28, pp. 185-198 (1990).

6. Harhalakis, G., Lu, T., Minis, I. and Nagi, R., “A Practical Method for Design of Hybrid-Type Production Facilities,” *International Journal of Production Research*, Vol. 34, pp. 897-918 (1996)

7. Irani, S.A. and Huang, H., “Layout Modules: A Novel Extension of hybrid Cellular Layouts,” *Proceedings of the 1998 ASME International Mechanical Engineering Congress & Exposition and Winter Annual Meeting of the ASME, Anheim, CA*: American Society of Mechanical Engineerings, Paper # 98-WA/MH 1, Novermber 15-20 (1889).

8. Irani, S.A., Cavalier, T.M. and Cohen, P.H., “Virtual manufacturing cells: exploiting layout design and inter-cell flows for the machine sharing problem,” *International Journal of Production Research*, Vol. 31, pp. 791-810 (1993).

9. Kannan, V.R. and Ghosh, S., “Cellular manufacturing using virtual cells,” *International journal of operations & production management*, Vol. 16, pp. 99-112 (1996).

10. Levassuer, G. A., Helms, M. M. and Zink, A. A., “Conversation from a functional to a cellular manufacturing layout at Steward, inc.,” *Production and Inventory Management journal*, Vol. 36, pp. 37-42 (1995)

11. McLean, C.R., Bloom, H.M. and Hopp, T.H., “The virtual manufacturing cell,” *Proceedings of the 4th IFAC/IFIP Conference on Information Control Problems in Manufacturing Technology*, pp. 105-111 (1982).

12. Meller, R.D. and Gau, K., "The facility layout problem: recent and emerging trends and perspectives," *Journal of Manufacturing Systems*, Vol. 15, No. 5, pp. 351-366 (1996).

13. Prince, J. and Kay, J.M., "Combining lean and agile characteristics: creation of virtual groups by enhanced production flow analysis," *International Journal of Production Economics*, Vol. 85, pp. 305-318 (2003).

14. Shukla, J., *Hybrid and Progressive Cellular Layouts for Jobshops,* Master of Science Thesis, Department of Mechanical Engineering, University of Minnesota, Minneapolis, MN (1995).

15. Wemmerlov, U. and Hyer, N.L., "Cellular manufacturing in the US industry – a survey of users," *International Journal of Production Research*, Vol. 27, pp. 1530-1551 (1989).

16. Womack, J.P. and Jones, D.T., *Lean Thinking*, Simom & Schuster, New York, NY (1996).

CHAPTER 7

PERFORMANCE EVALUATION OF THE PROPOSED LAYOUTS USING SIMULATION APPROACH

7.1 Experimental Design

In this chapter, an experimental study is conducted to investigate the performance of our conceptual layouts comparing to the traditional layouts when operating in a HMLV environment. Arena simulation software is used in this study. The experimental settings follow the setups described in “Performance of virtual cellular manufacturing with functional and cellular layouts in DRC settings” with some modifications to fit our study [Suresh and Slomp, 2005]. There are 4 main factors in this experimental study including (1) layout types, (2) batch sizes, (3) machine capacity, and (4) move times as shown in Table 7.1.

There are 5 cases of layouts including a functional layout, cellular layout, modular layout and hybrid flowshop layout with and without the next stage awareness system[†]. The batch sizes range from 5 to 80. The machine capacity factor has two levels—unlimited and limited number of machines available. In unlimited case, the number of available machines was set to a very large number to make sure that queues do not occur in the system.

Factors	**Levels**
Layouts	- FL (Functional Layout) - CM (Cellular Manufacturing Layout) - ML (Modular Layout) - HFL (Hybrid Flowshop Layout) - HFL-NXT (Hybrid Flowshop Layout with Next Stage Awareness System
Batch Size (q)	5,10,15,20, 30,50,80
Machine Capacity	-No limit -Limited
Move times	- Vary by different layouts - Fix for all layouts

Table 7.1: Main factors

The remaining parameters are set as shown in Table 7.2. The configurations of all layouts are shown in Figure 7.1. Each layout produces 19 part types and contains 12 machine types. Part routings are shown in Table 7.3. For the limited case, the number of available machines is set as shown in Table 7.4. There is an annual demand for each part

[†] The next stage awareness system is the material control developed in this research to promote pull scheduling in HMLV facilities. This system is described in Chapter 9.

type ranging uniformly between 1752-2628 pieces. The number of part per one part type for each order will be equal to the batch size. Therefore, the number of order in one year (3600 hours) will be about (1752-2628) / (batch sizes) orders. Since we do not assume well-defined product families in HMLV facilities, setup times are none for all layouts. Move times are set as shown in the table. They vary depending upon the different assumptions made such as the different sizes of machine groups, difficulties in locating materials and parts, and the availability of material handlers in the different layouts. The rule for job selection is first-come first-serve.

For each combination of 4 factors (5 layouts $\times$ 7 batch sizes $\times$ 2 levels of machine capacity $\times$ 2 levels of traveling time $\approx$ 140 combinations), the simulation runs for 360 days with 10 replications regenerated for each run. For each replication, there is a 180 days warm-up period meaning that the simulation will run without collecting data for 180 days and then run for 360 days period of simulation time to collect the statistical data.

Parameters	Values
Number of part types	19
Number of machine types	12
Annual demand (d)	1752-2628
Job Inter-arrival times	~ Uniform [d/q * 3600] hours
Setup time	0
Processing times	~ Exponential [0.1] hours
Move times	
Varied	
– FL	~Uniform [60-120] minutes
– CM	~Uniform [30-60] minutes – between cells ~Uniform [3-6] minutes – within the same cells
– ML & HFL	~Uniform [30-60] minutes – between modules/stages ~Uniform[1-2] minutes – within the same modules/stages
Fixed	2 hours
Job selection	First-Come First-Serve
Simulation time	360 days – 10 hours / day
Warm-up period	180 days

Table 7.2: Experimental parameters

Product	Routing
1	1→4→8→9
2	1→4→7→4→8→7
3	1→2→4→7→8→9
4	1→4→7→9
5	1→6→10→7→9
6	6→10→7→8→9
7	6→4→8→9
8	3→5→2→6→4→8→9
9	3→5→6→4→8→9
10	4→7→4→8
11	6
12	11→7→12
13	11→12
14	11→7→10
15	1→7→11→10→11→12
16	1→7→11→10→11→12
17	11→7→12
18	6→7→10
19	12

Table 7.3: Product routings

Machine Type	No. Available
1	2
2	1
3	1
4	2
5	1
6	2
7	3
8	1
9	1
10	2
11	1
12	1

Table 7.4: The number of machines available for each type

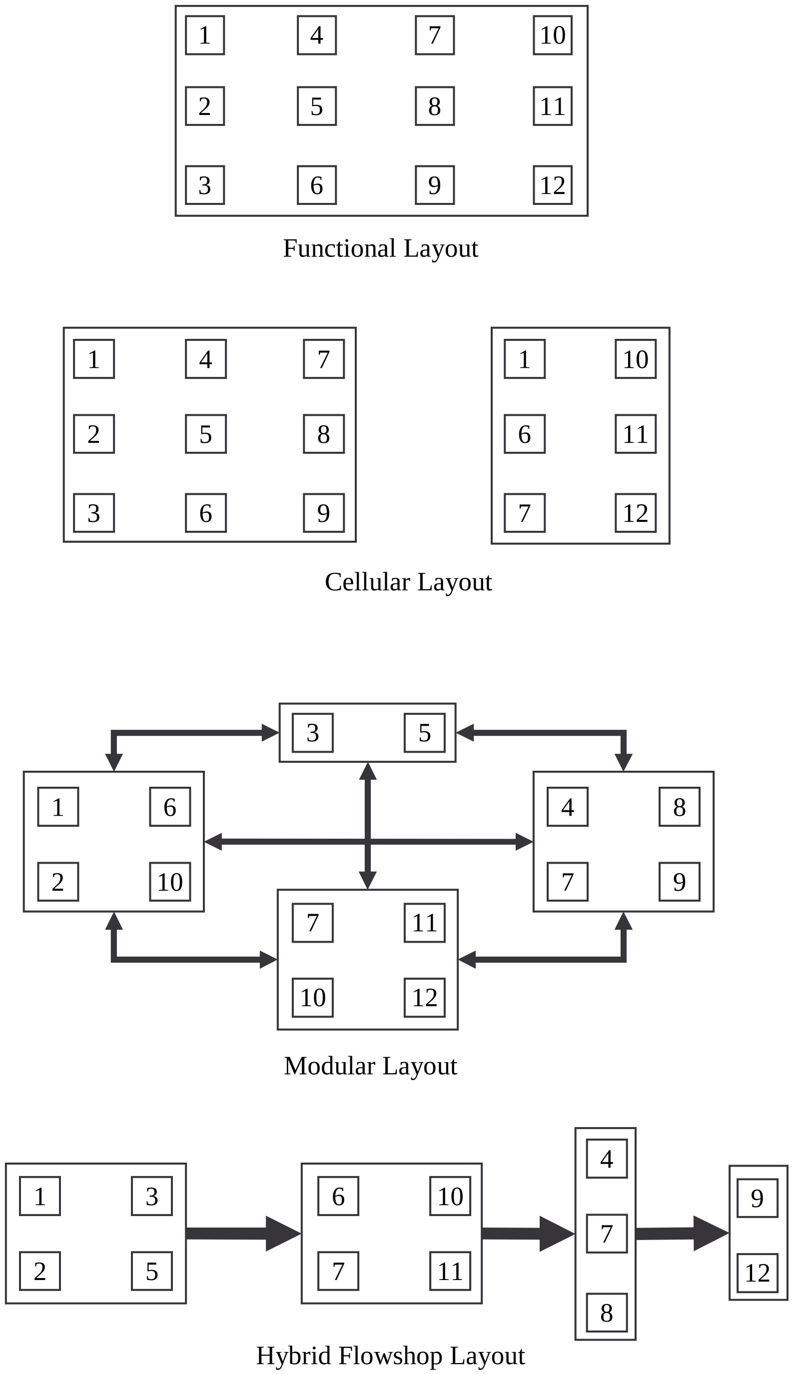


Figure 7.1: Layout configurations

7.2 Results and Discussions

As realized in a jobshop, a part may spend up to 95% of the total production time for waiting or traveling. Therefore, we are going to investigate the performance of different layouts in two most aspects—traveling time and waiting time in the layout.

7.2.1 Traveling Time Experimentation

High traveling times normally come from long traveling distances for moving materials between pairs of machines in a layout. The long traveling distances are driven directly by the arrangements and configurations of the layout. The total production time is mainly incurred by both waiting and traveling times with the small portion of processing time. If we want to investigate the traveling time, we have to omit the waiting time caused by the queuing delays from the production system. Therefore, the first set of experiments was analyzed without machine capacity constraints so queuing delays would not much occur. The purpose is to solely investigate the effects of traveling distance to the total production times for different layouts.

Table 7.5 shows the results from this first set of experiments. Batch sizes ranging from 5 to 80 were used in this set of experiments. Results contain (1) average completion times in hours per part, (2) average works-in-process (WIP) which are the average numbers of parts being processed, (3) average values for machine utilization, and (4) average queue lengths in the system. As can be seen from the simulation results in Table 7.5 and the performance comparison charts in Figure 7.2 and Figure 7.3, the functional layout was outperformed by other layouts, especially when the batch size is small. This is simply because the traveling distances in this layout are much higher than the others. The cellular layout has the lowest completion times and WIP. While the completion times and WIP for modular layout and hybrid flowshop layout are a bit higher than the cellular layout, these two layouts have performed much better than the functional layout.

Batch Size	Model	WIP	Completion Time per Part	Utilization	Queue
5	FL	3.22	5.52	0.27	0.00
	CM	2.11	3.62	0.22	0.00
	ML	2.17	3.72	0.24	0.00
	HFL	2.30	3.94	0.25	0.00
	HFL-NXT	2.30	3.94	0.25	0.00
10	FL	1.74	2.98	0.15	0.00
	CM	1.18	2.02	0.12	0.00
	ML	1.21	2.07	0.14	0.00
	HFL	1.27	2.18	0.14	0.00
	HFL-NXT	1.27	2.18	0.14	0.00
15	FL	1.25	2.13	0.12	0.00
	CM	0.87	1.48	0.09	0.00
	ML	0.89	1.52	0.10	0.00
	HFL	0.93	1.59	0.11	0.00
	HFL-NXT	0.93	1.59	0.11	0.00
20	FL	0.99	1.70	0.10	0.00
	CM	0.71	1.22	0.08	0.00
	ML	0.73	1.24	0.08	0.00
	HFL	0.76	1.30	0.09	0.00
	HFL-NXT	0.76	1.30	0.09	0.00
30	FL	0.72	1.23	0.08	0.00
	CM	0.56	0.95	0.06	0.00
	ML	0.57	0.97	0.07	0.00
	HFL	0.59	1.01	0.07	0.00
	HFL-NXT	0.59	1.01	0.07	0.00
50	FL	0.48	0.82	0.06	0.00
	CM	0.43	0.73	0.05	0.00
	ML	0.44	0.75	0.05	0.00
	HFL	0.45	0.77	0.06	0.00
	HFL-NXT	0.45	0.77	0.06	0.00
80	FL	0.39	0.66	0.05	0.00
	CM	0.36	0.61	0.04	0.00
	ML	0.36	0.62	0.05	0.00
	HFL	0.37	0.64	0.05	0.00
	HFL-NXT	0.37	0.64	0.05	0.00

Table 7.5: Experimental results for traveling time factor

It can be noticed that the performance of both hybrid flowshop layouts with and without the next stage awareness system is almost the same in all experiments. There is only one case where the batch size is 2 that the hybrid flowshop layout with the next stage awareness system can perform slightly better. This is because the next stage awareness system is active and working effectively only when the system contains some levels of queue. Without parts waiting in queues, the next stage awareness system will not be functioning.

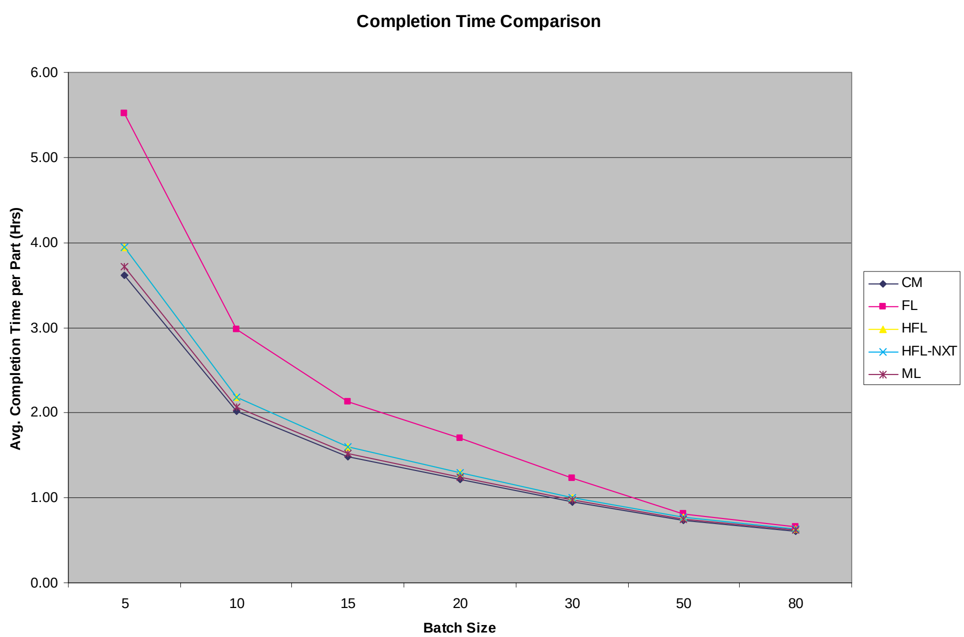


Figure 7.2: Average completion times comparison for traveling time factor

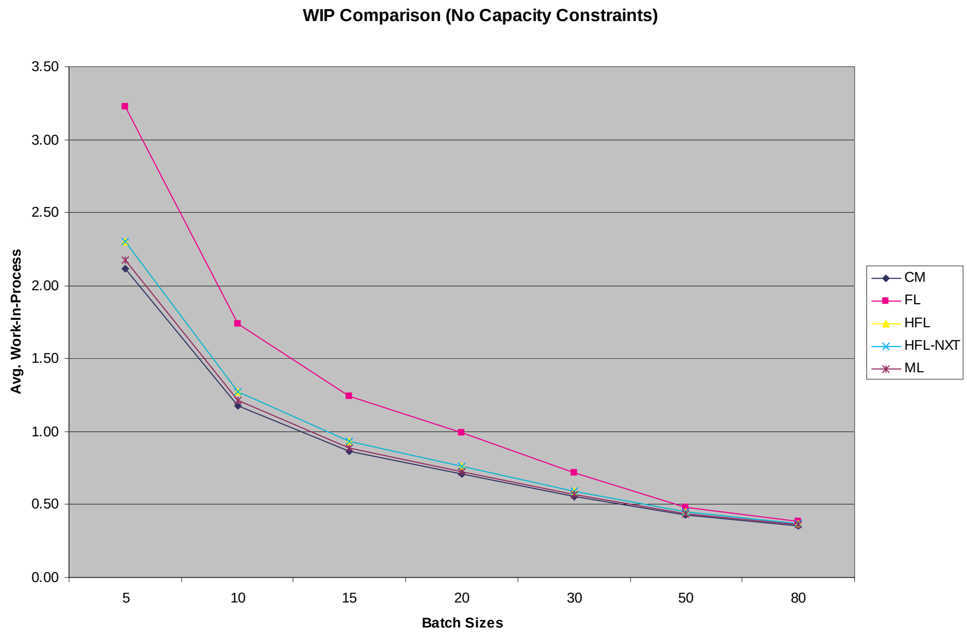


Figure 7.3: Average WIPs comparison for traveling time factor

7.2.2 Waiting Time Experimentation

The second set of experiments was analyzed with the limited number of machines available. When each layout has limited numbers of machines of each type, delays occur and queues in front of machines begin to build up. The completion times in this case should be contributed by both traveling time and queuing time. Since we fix the move times for each layout to be the same; therefore, the different in the average completion times of the layouts come directly from the queuing delays occurring in the different layouts. Table 7.6 shows the results from this set of experiments. The graphical representations of these results are shown in Figure 7.4 and Figure 7.5.

It can be seen from the results that this case is quite opposite to the previous one. In this case, the cellular layout, that had the best performance in previous case, was out performed by the other layouts in all experiments in this case. This is because some machines of the same types in the cellular layout are separated and allocated into disjoint groups. The disjoint locations for these machines of the same types made the cellular layout less flexible compared to other layouts. If machines of the same type are put together as they are in the functional layout, they should have more capability and flexibility to handle different jobs with different demands. The cellular layout has 4 machines of the same types separated into two different cells while the functional layout has no machine separated, the hybrid flowshop layout has 1 machine separated, and the modular layout has 2 machines separated. So from this information alone, we can assume that the functional layout should outperform all other layouts. This is correct if we do not take the hybrid flowshop layout with the next stage awareness system in to the consideration. Without the awareness system, the hybrid flowshop layout performed moderately as shown in the results. With the awareness system incorporated, this layout has shown its superiority over the other layouts in every experiment.

As we mentioned, when queues occur in the system, the awareness system will be functioning and it acts like a catalyst to boost up the performance of the hybrid flowshop layout to out gain the others. This is the benefit of having to the operators working in each stage to dynamically schedule their jobs. With this system, the operators are aware of the availability of machines in the next stages and they can decide which jobs they should perform first in order to help processing these jobs to be completed quicker. That is the main reason why this system is very effective.

Batch Size	Model	WIP	Completion Time per Part	M/C Utilization	Queue
5	FL	777.60	532.55	0.95	>800
	CM	786.51	538.28	0.98	>800
	ML	778.25	532.40	0.93	>800
	HFL	777.68	532.95	0.95	>800
	HFL-NXT	739.37	480.23	0.98	>800
10	FL	235.04	201.16	0.91	>200
	CM	242.36	207.37	0.94	>200
	ML	236.86	202.40	0.89	>200
	HFL	235.25	201.29	0.91	>200
	HFL-NXT	211.63	181.19	0.94	>200
15	FL	77.17	88.02	0.77	>60
	CM	79.19	90.41	0.81	>60
	ML	78.87	90.09	0.77	>60
	HFL	77.23	88.08	0.78	>60
	HFL-NXT	66.96	76.43	0.77	>60
20	FL	31.70	40.66	0.68	>20
	CM	31.84	40.86	0.70	>20
	ML	31.77	40.73	0.68	>20
	HFL	31.76	40.75	0.68	>20
	HFL-NXT	28.10	36.09	0.66	>20
30	FL	0.88	1.50	0.54	<0.5
	CM	0.93	1.58	0.54	<0.5
	ML	0.91	1.56	0.53	<0.5
	HFL	0.89	1.52	0.54	<0.5
	HFL-NXT	0.78	1.33	0.52	<0.5
50	FL	0.53	0.91	0.43	<0.2
	CM	0.55	0.95	0.44	<0.2
	ML	0.53	0.92	0.43	<0.2
	HFL	0.54	0.92	0.43	<0.2
	HFL-NXT	0.50	0.87	0.42	<0.2
80	FL	0.43	0.73	0.37	<0.15
	CM	0.46	0.79	0.38	<0.15
	ML	0.44	0.75	0.36	<0.15
	HFL	0.43	0.74	0.37	<0.15
	HFL-NXT	0.42	0.72	0.36	<0.15

Table 7.6: Experimental results for waiting time factor

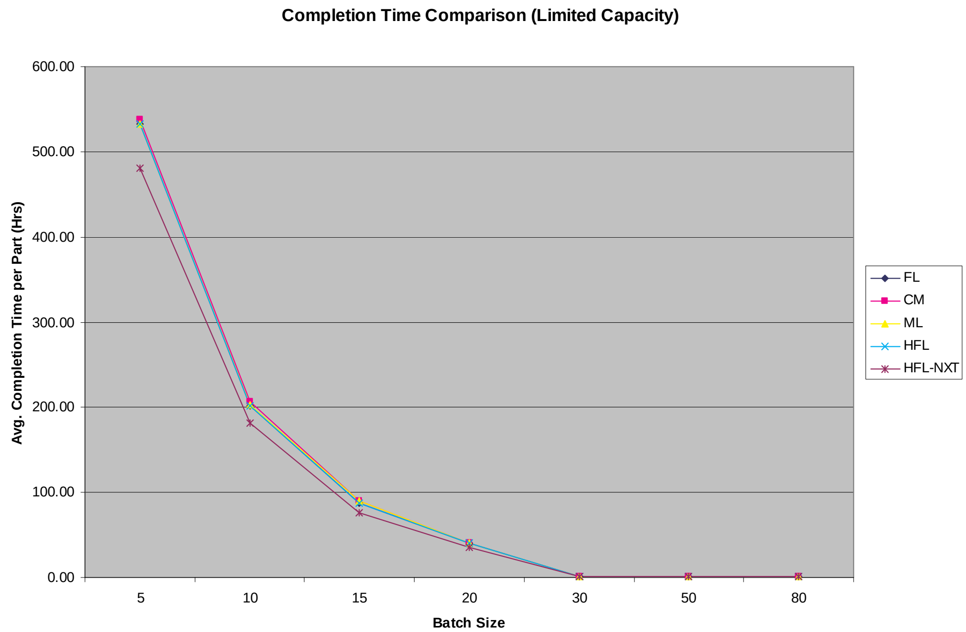


Figure 7.4: Average completion times comparison for waiting time factor

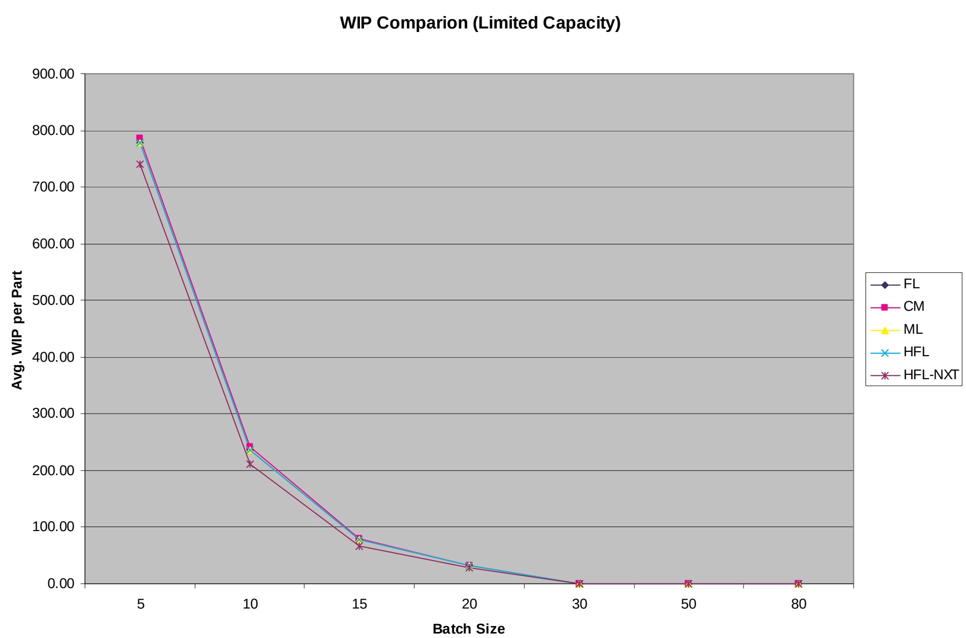


Figure 7.5: Average WIPs comparison for waiting time factor

7.2.3 Traveling Time and Waiting Time Experimentation

In this last case, we let the traveling time vary along with the limited number of machines in the layouts and re-perform the experimental study to investigate the performances of the layouts. The results for this case are shown in Table 7.7 and their graphical representations are shown in Figure 7.6 and Figure 7.7. As can be seen from the results, in some experiments the functional layout was outperformed and in the other experiments, other layouts were outperformed. This is because the different levels of traveling times and waiting times in the layouts. If the traveling time contributes more than the waiting time to the total production time, the cellular layout tends to perform better. If the waiting time contributes more than the traveling time to the total production time, the functional layout tends to perform well. The modular layout and the hybrid flowshop layout, in most experiments, performed moderately between these two layouts since they can absorb the impact of both traveling time and waiting time. However, as can be expected, the hybrid flowshop layout with the awareness system outperformed other layouts in every experiment. This is such evidence to show how effective of this layout and the awareness system compared to other layouts. It can be very promising that this layout should perform effectively in real-world cases.

Batch Size	Model	WIP	Completion Time per Part	M/C Utilization	Queue
5	FL	235.35	402.98	0.91	>200
	CM	239.56	410.22	0.91	>200
	ML	236.78	405.32	0.89	>200
	HFL	235.20	402.74	0.91	>200
	HFL - NXT	211.19	362.12	0.94	>200
10	FL	77.76	132.97	0.77	>60
	CM	77.27	132.53	0.79	>60
	ML	78.70	134.66	0.77	>60
	HFL	77.18	132.07	0.78	>60
	HFL - NXT	67.02	114.81	0.77	>60
15	FL	32.08	54.96	0.67	>20
	CM	32.07	54.96	0.67	>20
	ML	31.92	54.74	0.68	>20
	HFL	32.30	55.37	0.68	>20
	HFL - NXT	28.46	48.72	0.66	>20
20	FL	12.74	21.82	0.62	>10
	CM	12.83	21.96	0.61	>10
	ML	12.69	21.71	0.62	>10
	HFL	12.75	21.84	0.62	>10
	HFL - NXT	11.24	19.30	0.60	>10
30	FL	1.12	1.91	0.54	<0.5
	CM	0.97	1.67	0.52	<0.5
	ML	1.01	1.72	0.54	<0.5
	HFL	1.03	1.76	0.54	<0.5
	HFL - NXT	0.92	1.57	0.52	<0.5
50	FL	0.61	1.04	0.43	<0.2
	CM	0.57	0.98	0.41	<0.2
	ML	0.59	1.02	0.42	<0.2
	HFL	0.60	1.02	0.43	<0.2
	HFL - NXT	0.56	0.96	0.42	<0.2
80	FL	0.47	0.81	0.37	<0.15
	CM	0.47	0.81	0.36	<0.15
	ML	0.46	0.79	0.36	<0.15
	HFL	0.47	0.80	0.37	<0.15
	HFL - NXT	0.45	0.77	0.36	<0.15

Table 7.7: Experimental results for both waiting time and traveling time factors

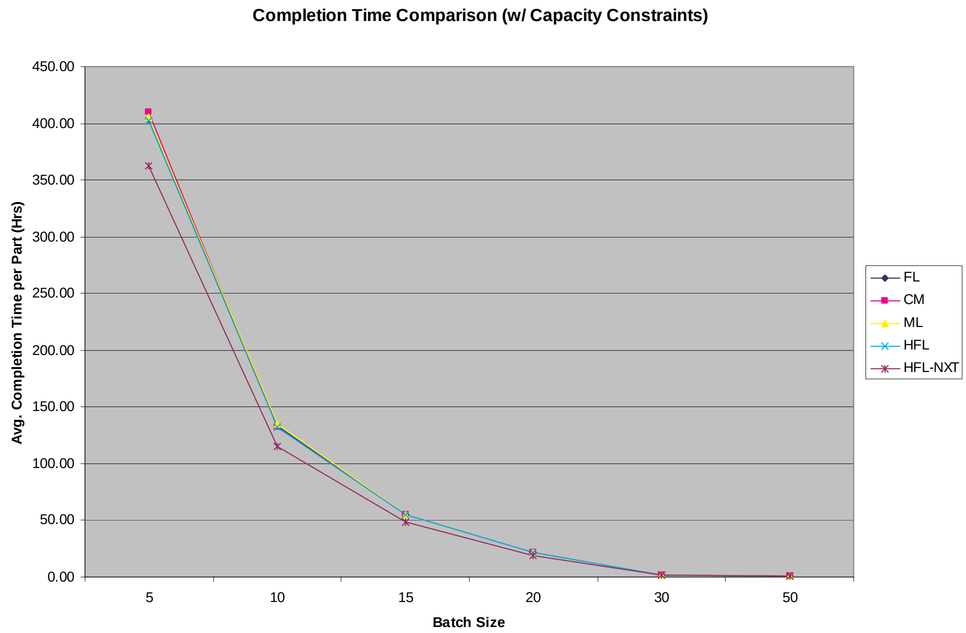


Figure 7.6: Average completion times for both waiting time and traveling time factors

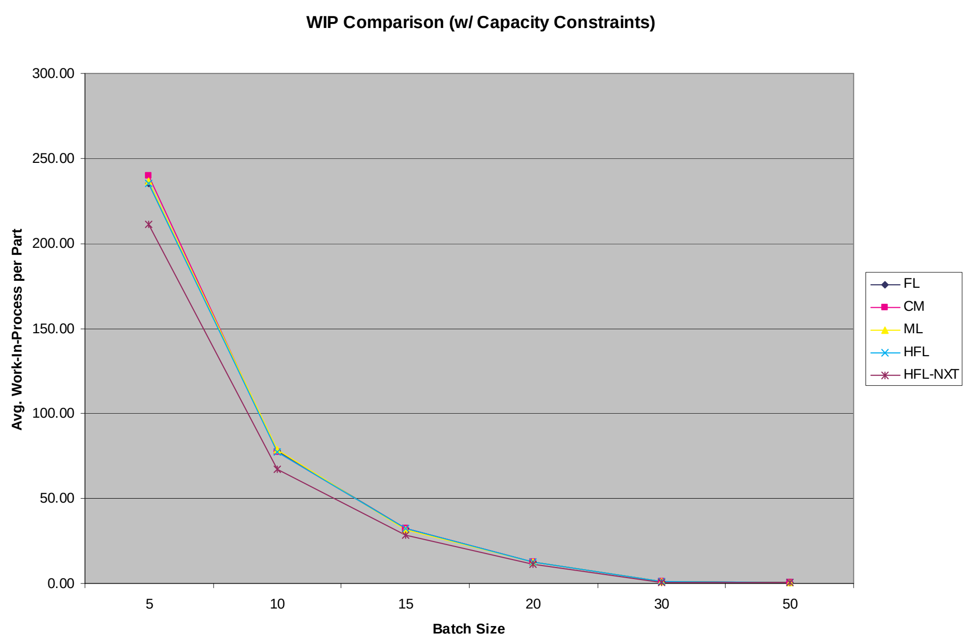


Figure 7.7: Average WIPs for both waiting time and traveling time factors

7.3 Conclusion

From the experimental study, it can be concluded that the performance of the two proposed layouts is promising. A modular layout can vastly reduce an amount of traveling distances as well as a cellular layout when different machine types are grouped. At the same time, with some machines of the same type are still grouped together and shared across the product mix, these machines are more flexible to perform different jobs with fluctuating demands. The modular layout may not outperform a functional layout in term of flexibility and cellular layout in term of flow distance reduction. However, this layout can leverage the advantages of both traditional layouts. Therefore, for HMLV facilities where a cellular layout may not be a solution, this modular layout can be an alternative layout that brings both the flexibility and traveling distance reduction to the HMLV facilities.

On the other hand, a hybrid flowshop layout has the same advantages and disadvantages as a modular layout since these two layouts are very similar in concept. However, when engaging a hybrid flowshop layout with the pull scheduling technique that we have developed, this layout has emerged a significant advantage over the other layouts.

References

1. Suresh, N.C., Slomp, J., “Performance of virtual cellular manufacturing with functional and cellular layouts in DRC settings,” *International Journal of Production Research*, Vol. 43, No. 5, pp. 945-979 (2005).

CHAPTER 8

CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH

8.1 Conclusions

In this dissertation, we have addressed the necessity that small and medium manufacturing companies change their layouts to facilitate the production flows a prelude to implement Lean Manufacturing. We have addressed these needs by introducing two novel layouts—a modular layout and a hybrid flowshop layout. The concept and design approach for a modular layout was introduced in Chapter 3. An earlier mathematical model in the literature for the problem of designing a modular layout was improved. We optimally solved a small dataset for a modular layout problem using this mathematical model with CPLEX, a commercial IP solver. However, the complexity of this problem falls in the class of NP-complete problems (the proof was also given in this chapter). Therefore, we have simplified the mathematical model by removing some constraints and the two-stage solution approach has been introduced in order to solve this layout problem more efficiently. The modified model was solved with the same dataset and the results showed that the differences between the original model and the modified version are not significant. A heuristic approach using cut-tree algorithm has been developed to solve large-size problems and presented in Chapter 4. Its performance has been compared with the optimization approach and the results showed that the heuristic approach can provide reasonably good results with significantly less computational effort.

In Chapter 5, the detailed concept of a hybrid flowshop layout was introduced. The mathematical model for the problem of designing a hybrid flowshop layout was constructed based on the model for the modular layout problem. Since the hybrid flowshop layout problem also falls into the class of NP-complete problem (the proof was given in this chapter), we have simplified the mathematical model for this layout problem just like we did for the modular layout problem. The same dataset was solved optimally for this layout problem using CPLEX. A heuristic approach using ratio-cut was developed to tackle the real-world size problems. The results have shown that the heuristic approach has performed efficiently providing near-optimal solutions with less amount of computational effort compared to the optimization approach.

We presented a systematic approach to design a layout for a HMLV facility using PFAST in Chapter 6. This systematic approach has been developed based on years of real project experiences with small and medium-size manufacturing companies. A real-world case study was used to describe all the steps in our approach, from collecting data until all alternative layouts have been constructed.

Finally, in Chapter 7 we presented an experimental study to compare the performance our proposed layouts and the traditional layouts when operating in an HMLV environment. We have found that each layout could have advantages and disadvantages over the others when different parameters and factors change. However, when a hybrid flowshop layout has performed with the Next Stage Awareness System (described in Chapter 9) for material control, this layout outperformed other layouts in most cases where queuing delays and work-in-progress inventories exist in the system.

Based on the experiments conducted, we realized that the two novel layouts we proposed showed potential fit with the manufacturing facilities operating in an HMLV environment.

8.2 Research Contributions

The research contributions from this dissertation can be summarized as follows:

1. The hybrid flowshop layout and its design approach is probably the most important contribution of this research. The concept of this layout was inspired by the term "hybrid flow shop" found commonly in literature on scheduling problems but had never been incorporated into layout design problems. We have adapted this concept of hybrid flow shop and developed a new conceptual manufacturing layout called a hybrid flowshop layout. This layout is designed to have its production flow mainly in a forward direction from the beginning stage to the finishing stage regardless of the dissimilarity or complexity of jobs. We have developed the approaches to design this layout using both optimization and heuristic methods. We have also shown that the complexity of production flow in manufacturing facilities operating in a HMLV environment can be significantly reduced by implementing this layout instead of the traditional Functional Layout observed in most companies.

Streamlined production flow is core to the implementation of Lean Manufacturing. The facility layout is core to the complexity of the production flow. Without designing the appropriate layout to facilitate and simplify the production flow, Lean Manufacturing can be very difficult to implement effectively for these manufacturing companies operating in an HMLV environment. With the hybrid flowshop layout, these manufacturing companies can possibly turn their chaotic facilities into flowline-like production facilities and efficiently apply Lean Manufacturing to reduce their production costs and increase their production efficiencies.

2. The second most important contribution is that we have developed a new material control system called the Next Stage Awareness System (NSAS) that can be used effectively with a hybrid flowshop layout. Once the production flow in a HMLV facility gets simplified and organized by adopting a hybrid flowshop layout, a Pull Scheduling technique, which is an important component of Lean Manufacturing, can then be applied. There are several available scheduling techniques either pure Pull or hybrid Push-Pull; however, none of them seems to fit well in HMLV environments. The NSAS is therefore developed to address this problem.

It is possible to see that production flow in an HMLV manufacturing environment is analogous to traffic in a crowded city. On any given day of operation, there can be traffic problems at any point anywhere in the facility and the problem points can shift from time to time. This awareness system works like an the traffic alert displays on highways that warn drivers if there are problems ahead, then suggest to the drivers to take different roads if possible. In a facility with a hybrid flowshop layout, the awareness system tells workers in any stage if there are queues and delays at certain machines in the next stages, then it suggests to these workers to start different jobs, if possible, that require other available machines for their next operations. This system is designed to be as simple as possible, yet proven to be very efficient as shown by our experiment.

Although, scheduling and material control are not our main focus of this research and the description of this awareness system is limited, this system showed its potential in our experimental study. It can be a research topic that future researchers can improve upon to produce a new control system that manufacturing companies operating in an HMLV environment can use to improve their performance when queuing delays and work-in-process exist. That is a key reason why we consider this “alert system” to also be one of the top contributions of our research.

3. Another important contribution is a new approach for designing a modular layout. We have improved the original mathematical model for designing this layout. We have simplified that mathematical model without worsening its accuracy. We have also solved the problem of designing this layout both optimally and heuristically. Our cut-tree heuristic approach has performed reasonably well with less computational effort compared to the optimization approach. Advance search techniques such as pair-wise interchange, simulated annealing, and other random-search heuristics can definitely improve the solution quality of our cut-tree approach.

In addition, the modular layout problem originally proposed uses a string-based approach as its solution approach. We have found that the string based approach is not flexible and it can result in a large number of duplicated machines. For example, suppose there are products using 3 machines A, B and C. Consider the following routings:

A-B-C

A-C-B

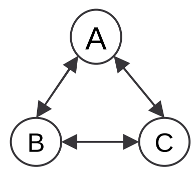
B-A-C

B-C-A

C-A-B

C-B-A

These routings would all be considered as unique and different in the string-based approach. However, consider the completely connected graph incident on these 3 machines as follows:



All of the above routings can be mapped onto this one machine incidence graph. Therefore, our cut-tree heuristic approach using a graph network reduces the problem of excessive machine duplication in the original method.

4. We have conducted an experimental study to compare the performance of different layouts and we have found that our results have clarified some of our concepts about layout design with machine duplication. We had the idea that disaggregating and distributing a number of machines strategically in a facility could increase routing flexibility, improve production efficiency, and reduce material handling costs. From our experimental study, the material handling costs can be reduced but the flexibility of layouts to handle fluctuations in production demand and product mix changes may also be reduced if the machines or departments are not correctly defined. The simulation

results have shown that the more we spread out the machines of the same types to different locations, the less we can handle the mix and volume fluctuation. The distribution of the machines causes the total capacity to be dispersed and without well-managed scheduling technique and capacity allocation strategies, it could happen that the machines in one location are overloaded while the others are under-utilized. Unless these machine types have no capacity problem, then the desegregation and distribution of these machines would help improving the flexibility and efficiency of the layout.

5. We have over a decade of experience working on real-world projects for small and medium-size manufacturing companies in order to help them simplify their complex production flows and provide them better layouts for their facilities. The PFAST software has been developed and has been used for many years on these projects. Flow simplification as a prelude to layout design is the key capability of this software. Therefore, we have reviewed classical and contemporary approaches and techniques based on our experience and introduced a new systematic approach for designing layouts for HMLV facilities using PFAST. Our intention is to share the current technology, its capabilities, and how we use it as part of a man-machine approach to design appropriate layouts for each small and medium-size manufacturing company.

6. We have used a graph network model to represent the routings where machines of the same types are connected with machine duplication costs (machine purchasing costs). We use this model to design both modular layout and hybrid flowshop layout. Therefore, it is possible that we could have used this network graph for designing all types of facility layouts. Suppose we have the network graph that is constructed using the routings for a layout problem as shown below.

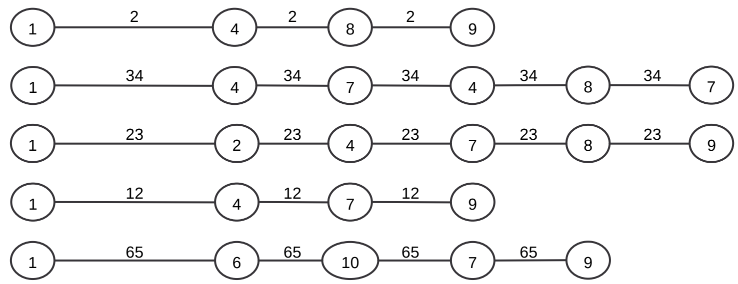


Figure 8.1: A graph network representing a set of routings

If the resulting graph shows that all nodes representing the same types of machines are connected, we have a Functional Layout as shown below:

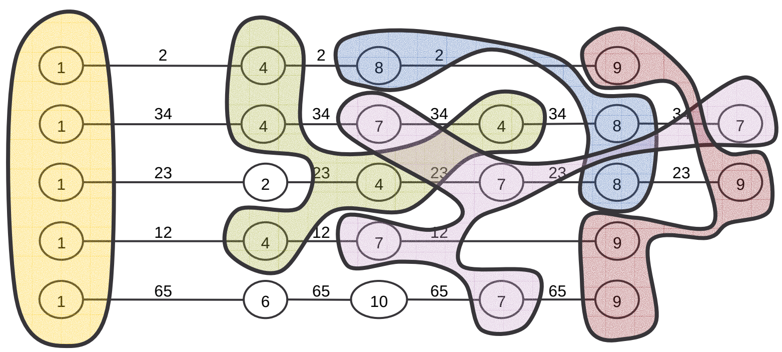


Figure 8.2: The resulting graph of a functional layout

If the resulting graph is partitioned horizontally into several partitions where machines of the same types are allocated in different partitions, we have a Cellular Layout for this solution as shown below:

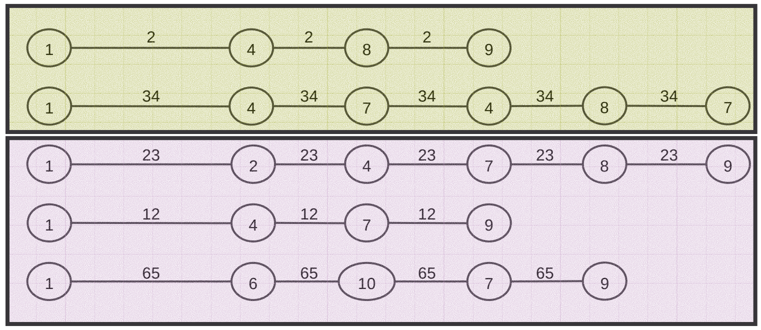


Figure 8.3: The resulting graph of a cellular layout

If the resulting graph is partitioned vertically into several partitions where partitions are not overlapped, we have a hybrid flowshop layout as shown below:

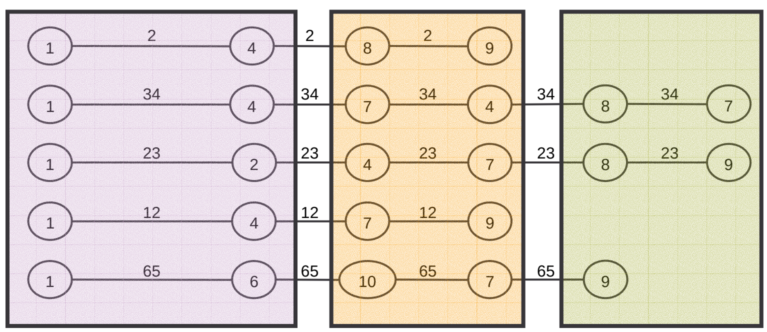


Figure 8.4: The resulting graph of a hybrid flowshop layout

If the resulting graph is partitioned into several partitions, we have a modular layout as shown below:

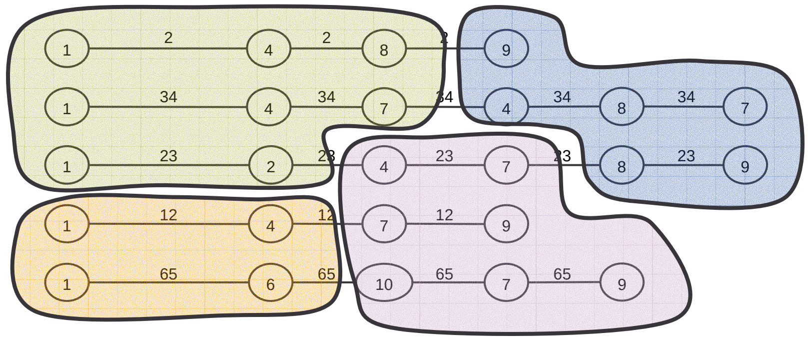


Figure 8.5: The resulting graph of a modular layout

This could lead to a unified approach for designing multiple types of manufacturing layouts simultaneously. Layout analysts can benefit from this approach since it is not only designing a layout, but it is also comparing the different types of layout and selecting one which is the most suitable to the particular dataset.

7. Finally, a possibility of applying Lean Manufacturing to HMLV facilities using the layout design methods developed in this research is our last contributions. The novel layouts with a new material control system, and a systematic approach for layout design using PFAST could potentially answer the needs of small and medium manufacturing companies that are facing difficulties when they try to implement Lean Manufacturing but lack the ability to analyze a diverse product mix.

8.3 Recommendations for Future Research

Some recommendations for future work to extend this research are:

- Conduct more detailed research against the worst case scenario where the products are completely dissimilar. This situation may not happen in the real world; however, it provides the ultimate “stress test” for these two layouts as to how well they can handle the chaos compared to the functional layout which is considered the best layout in this case of total mix uncertainty.
- Improve the problem models for designing the two proposed layouts to incorporate machine relocation costs. We have not considered the fact that the cost of changing the existing layout to the new layout should be incorporated in our research.
- Conduct more detailed research in the case where backtracking flows are allowed in a hybrid flowshop layout and how well the Next Stage Awareness System can perform in this case.
- Use different partitioning approaches beside cut-tree and ratio-cut to see if computational performance and quality of results could both be improved.
- Use random search heuristic techniques, such as pair-wise swapping or simulated annealing, to further improve both layout design methods.
- Conduct an exhaustive experimental study involving a really large product mix obtained from industry. In this research, we used the small problem for the simulation experiment and the layout design problems, in order to be consistent. This problem size is too small compared to real-world problem sizes. With the

real-world problems, the differences in the performance of different layouts could be magnified and we could get different results and conclusions possibly.

- Use Analytic Hierarchy Process (AHP) in helping the selection of the alternative layouts designed by PFAST. In our systematic approach, we do not have an exact approach to select the best layout from among several alternative layouts. Human judgment is used currently. The AHP is similar to decision tree logic and would provide a better approach for layout selection.

CHAPTER 9

FUTURE STUDY—THE NEXT STAGE AWARENESS SYSTEM FOR MATERIAL CONTROL

When the production flow in a hybrid flowshop layout is unidirectional, pull scheduling system which is one important component in Lean Manufacturing can be applied. In this dissertation, we have developed and proposed a new material control system called the Next Stage Awareness System to promote pull scheduling for a hybrid flowshop layout in an HMLV facility. This system allows operators in the current stage in a hybrid flowshop layout being aware of machine availability in the next stages. It works like traffic alert displays on a highway that warn drivers if there are problems ahead so drivers can decide to take the next roads or detour to other roads. This system allows operators to prioritize jobs based on the availability of the machines that are required to perform the next operations of these jobs. For example, suppose there are two products 1 and 2 coming in to be processed as machine *A*. After the processes are done at machine *A*, the product 1 will go to machine *B* and product 2 will go to machine *C* for their next operations. Machine *B* is not available currently and machine C is ready to take another job. If this operator at machine *A* does not realize the availability of machines *B* and *C*, he will take product 1 to perform and when the job for product 1 is done, product 1 will be sent to machine *B* and have to wait until machine *B* is available before this its next operation can be performed. If the operator knows the availability of both machines, then product 2 will be wisely chosen by this operator to be performed first. When the job for product 2 is done, its next operation can be performed by machine *C* promptly since this machine is ready to take the job. At the mean time, machine *B* would also be ready to take another job by the time that the job for product 2 is done at machine *A*. So both products 1 and 2 can be completed faster in this scenario than the previous scenario. This is why the next stage awareness system would help reducing lead time and also controlling WIP in a hybrid flowshop layout.

The illustration of the awareness system for a hybrid flowshop layout is shown in Figure 9.1. The Vakharia data set is used to setup this illustration. In this figure, there is a post stand located in front of each stage in the layout. Each post stand contains slots for cards that representing the availability of all machines in the next stages. If cards for certain machines are present, it means that these machines are available to take a job. At the starting period, all cards are available as shown in this figure. Suppose there are three jobs (products) coming in to stage 1. For their next operations, these products require machine 6 at stage 2, machine 4 at stage 3, and machine 9 at stage 4. Operators then pick cards 6, 4 and 9 and put into containers where the corresponding jobs will be stored as shown in the figure.

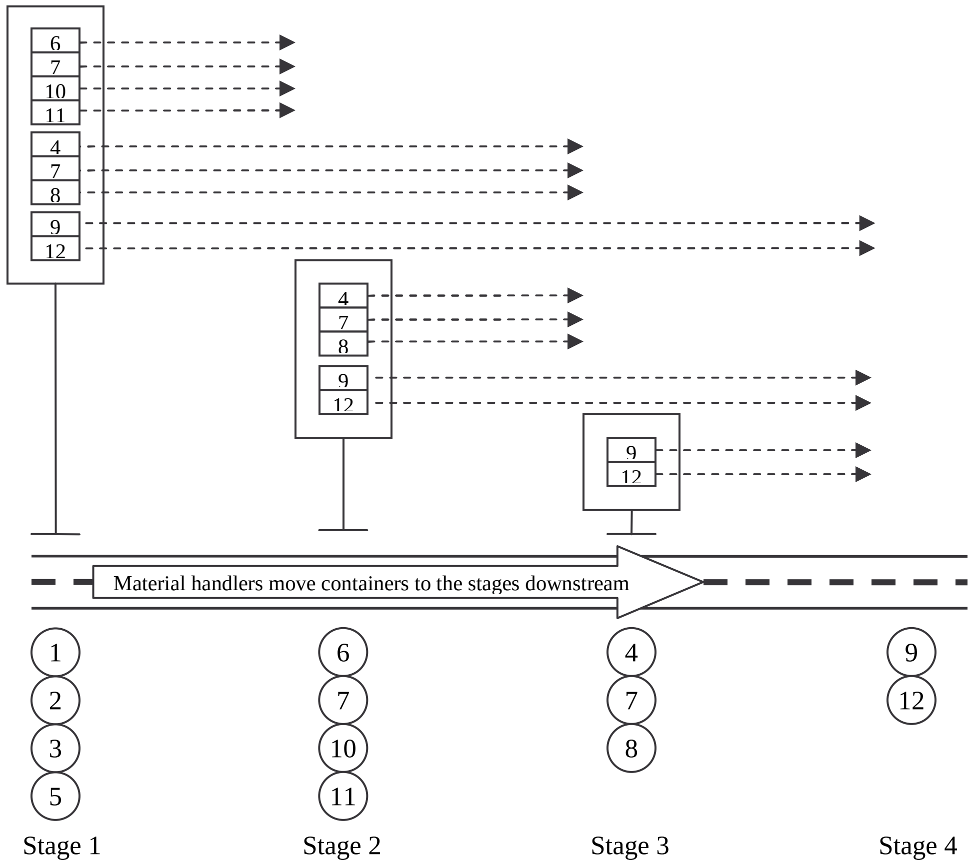


Figure 9.1: The next stage awareness system – Material handlers move materials from upstream to downstream

When jobs are done, material handlers move these containers to the designated stages. Suppose that there are more jobs coming in at stage 1 and the next operations of these jobs require machines 6, 4 and 9, these jobs will have the lowest priority to be selected. Because there are no more slots for jobs that are going to the machines 6, 4 and 9, it means that these machines are not available and ready to take any more jobs. Unless the cards of these machines returning back to the stage 1, then the new jobs for these machines 6, 4 and 9 can be started.

At the next stages, when the jobs in the containers get started, the cards in these containers will be removed to return back to the preceding stages and placed in the slots to which these cards belong. The material handlers pick up the cards and then put them into their original slots as shown in Figure 9.2. This triggers the availability of the machines for the next stages and the process repeats as described.

In a hybrid flowshop layout, since flow is mostly unidirectional from upstream to downstream, material handlers can work more efficiently. When they travel from stages upstream to downstream, they look for moving materials in the containers from the current stages to the next stages. When they travel back from the stages downstream to upstream, they look for picking up the cards in the containers that are ready to be returned back to the stages downstream. This is why the scheduling and material handling in a hybrid flowshop layout with this awareness system can be much simplified compared to a functional layout.

The awareness system developed for a hybrid flowshop layout is adapted from POLCA (Paired-cell Overlapping Loops of Cards with Authorization), a material control system that has been introduced by Suri (1998). The different between POLCA and our system is that POLCA is a cell-based controlling system and our system is machine-based controlling system. In HMLV facilities, manufacturing cells can hardly form; therefore, POLCA can not apply straightforwardly to these facilities. If an individual machine in a facility is treated as a manufacturing cell, POLCA can possibly apply. However, if there are 12 machines in a facility, there will be $12\times11/2 = 66$ pairs of machines to control using POLCA cards. In our system, if 12 machines can be constructed as the 4-stage hybrid flowshop layout shown in Figure 9.2, then there will be 16 cards needed for the awareness system.

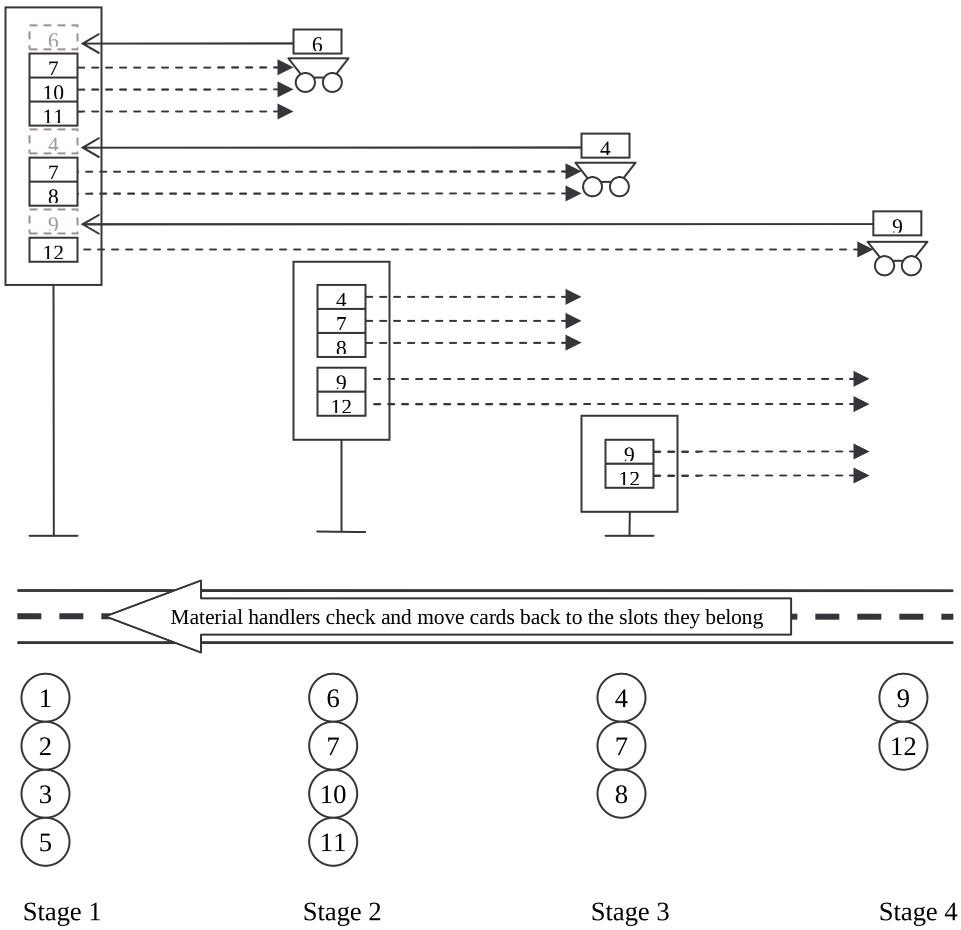


Figure 9.2: The next stage awareness system – Material handlers move materials from upstream to downstream

One can argue that if the 12 machines can be formed into 4 manufacturing cells, then there will be only 4×3/2 = 6 POLCA cards needed for all pairs of these cells. It can be true; however, these cells are not completed cells, meaning that all machines inside each cell do not perform their operations in series or patterns. Most of the time, the operations of products will be performed randomly. As a result, machines in a particular cell can be busy or available randomly and independently. Suppose that there is a POLCA card for a loop between cell *A* and cell *B*. If certain machines in cell *B* are ready, it does not mean that cell *B* can take any job from cell *A*. For example, if machine 1 in cell *B* is ready, cell *B* can take a job from cell *A* only if this job requires machine 1 in cell *B* to perform. So in HMLV facilities, everything has to come down to individual machines, not completed cells, and this is a reason why POLCA may not work effectively in these facilities.

As can be seen from the performance evaluation using simulation in Chapter 7, the results has shown that a hybrid flowshop layout outperforms the other layouts in every experiment when queues exist in the system. This is because a hybrid flowshop layout has operators involved in dynamically adjusting the schedules of jobs based on the current situation in the shop floor as we have described. Although this is not a new technology to the manufacturing system, it is a new approach to HMLV companies that allows these companies to feasibly apply Lean Manufacturing and a pull scheduling technique to their facilities where operating in a chaotic environment.

References

1. Suri, R., *Quick Response Manufacturing: A Companywide Approach to Reducing Leadtimes*, Productivity Press, Portland, OR (1998).

APPENDIX A

INPUT DATA FOR CASE STUDY

No.	Part	Quantity	Revenue	Routings															
1	80-A37353	728	47320	17	6	2	11	10	29	54	55								
2	80-C27416-1	1456	124054	17	6	2	11	10	29	54	55								
3	80-C27416-2	5614	495992	17	6	2	11	10	29	54	55								
4	80-C46806-1	4354	362474	17	6	2	11	10	29	54	55								
5	80-C55581	1750	151284	17	6	2	11	10	29	54	55								
6	80-C558-1	1526	131922	17	6	2	11	10	29	54	55								
7	80-D8097	756	133882	17	6	2	11	10	29	54	55								
8	80-B113-1001	48132	186116	17	56	57	54												
9	80-4003111	4900	41216	1	57	25	52	48	55										
10	80-4009121	30800	151228	1	57	25	52	48	55										
11	80-4009262	5600	27048	1	57	25	52	48	55										
12	80-4009263	39886	176302	1	57	25	52	48	55										
13	80-4009270	32900	126994	1	57	25	52	48	55										
14	80-4010346	117614	379806	1	57	25	52	48	55										
15	80-4010348	21000	69300	1	57	25	52	48	55										
16	80-4010349	12600	43092	1	57	25	52	48	55										
17	80-4010350	38500	148610	1	57	25	52	48	55										
18	80-4010351	19600	86632	1	57	25	52	48	55										
19	80-4010352	7000	34790	1	57	25	52	48	55										
20	80-4011714	28000	91560	1	26	57	52	48	55										
21	80-4011725	16800	119448	1	57	25	52	48	55										
22	80-4012169	33362	98756	1	26	57	52	48	55										
23	80-4012174	113400	324324	1	26	57	52	48	55										
24	80-4012179	133070	377916	1	26	57	52	48	55										
25	80-4012212	1400	10682	1	26	57	52	48	55										
26	80-4012213	4144	34846	1	26	57	52	48	55										
27	80-4030339	65198	117208	1	27	9	57	48	55										
28	80-4030341	53200	140980	1	27	9	57	48	55										
29	80-4035144	5600	14952	1	28	50	27	48	55										
30	80-4035149	28252	86170	1	28	50	27	48	55										
31	80-4039260	1400	6062	1	28	50	27	48	55										
32	80-4041707	42000	180180	1	57	25	52	48	55										
33	80-4059989	3850	33306	1	26	57	52	48	55										
34	80-4067179	26502	757428	17	39	40	26	57	54	57	55								
35	80-4030011870964	39256	1618932	17	39	40	21	22	53	29	28	4	55						
36	80-150T084LT	1344	587664	17	6	2	7	12	8	42	41	57	55						
37	80-G121-1002	280	138768	17	6	2	7	12	8	42	41	57	55						
38	80-NL150T060LT	1764	632744	17	6	2	7	12	8	42	41	57	55						
39	80-NL150T072LT	1540	591752	17	6	2	7	12	8	42	41	57	55						
40	80-NL150T084LT	644	281596	17	6	2	7	12	8	42	41	57	55						
41	80-NL150T096LT	168	83258	17	6	2	7	12	8	42	41	57	55						
42	80-NL150T120LT	112	68572	17	6	2	7	12	8	42	41	57	55						
43	80-3249869	28014	179200	17	56	1	17	29	26	54	57	48	55						
44	80-121009-00	29288	171332	17	39	40	21	22	55										
45	80-121188-002	32200	202860	17	39	40	21	22	55										
46	80-121189	7014	47348	17	39	40	21	22	55										
47	80-671391	147000	658560	17	16	11	10	26	4	55									
48	80-121018-00	47950	275716	17	39	40	21	22	55										
49	80-121148	35350	143444	1	50	26	27	55											
50	80-121387	3220	27720	17	39	40	21	22	55										
51	80-ULC0200	2800	31556	17	39	40	21	22	55										
52	80-35-B357	132314	697298	17	1	57	4	54	55										
53	80-27750-01	15428	931854	17	39	40	42	41	3	7	12	57	54	55					
54	80-37355-1072	1204	501774	17	6	2	7	12	8	42	41	57	55						
55	80-37355-1084	952	447062	17	6	2	7	12	8	42	41	57	55						
56	80-051-1	48580	255052	17	1	26	4	54	55										
57	80-191820	26866	100744	17	16	11	10	26	29	28	27	48	55						
58	80-522500	1652	8428	17	16	11	10	26	29	28	27	48	55						
59	80-551500	3724	29050	17	16	11	10	26	29	28	27	48	55						
60	80-S113-1001	39732	39732	17	16	11	10	57	53	55									
61	80-S113-1004	364	70532	17	16	11	10	57	53	55									
62	80-27708-302UP	8540	1952230	17	39	40	16	9	11	10	39	40	57	54					
63	80-9033023-303	36848	922670	57	54	57	55												
64	80-9627712-301UP	1022	192276	17	6	56	16	11	10	6	7	12	8	54	57	54	53	8	55
65	80-9627713-301UP	1050	462378	17	6	56	16	11	10	6	7	12	8	54	57	54	53	8	55
66	80-9627714-301UP	1078	360430	17	6	56	16	11	10	6	7	12	8	54	57	54	53	8	55
67	80-9627715-301UP	1022	423766	17	6	56	16	11	10	6	7	12	8	54	57	54	53	8	55
68	80-9627716-301UP	1022	139258	17	6	56	16	11	10	6	7	12	8	54	57	54	53	8	55
69	80-3260-041	2828	569842	17	6	2	11	10	29	28	54	57	55						
70	80-3260-0980	1512	157024	17	6	2	11	10	29	28	54	57	55						
71	80-3260-503	182	48342	17	6	2	11	10	29	28	54	57	55						
72	80-671635-00	75012	466340	17	3	7	12	26	4	55									
73	80-4030007296089	168	38892	17	3	7	12	8	4	54	29	4	55						
74	80-4030007296090	1456	234346	17	3	7	12	8	4	54	29	4	55						
75	80-4030007296091	6356	757190	17	3	7	12	8	4	54	29	4	55						
76	80-4030007296094	9240	574084	17	3	7	12	8	4	54	29	4	55						
77	80-27377	14112	69286	17	16	11	10	26	4	55									
78	80-921790	4914	526120	17	3	7	12	8	54	29	28	4	55						
79	80-W101-2006	462	445858	17	6	2	42	33	41	54	57	4	55						

Figure A.1: Product Information

W/C #	DESCRIPTION OF EQUIPMENT	RELOCATABLE?	COST OF DUPLICATION
1	700 TON PRESS	YES	EXPENSIVE
2	**5" UPSETTER**	**NO**	**EXPENSIVE**
3	5000# Area FURNACE	YES	
4	LARGE ROTOBLASTER	YES	
5	350 TON PRESS	YES	EXPENSIVE
6	5" UPSETTER FURNACE	YES	
7	**5000# Area HAMMER**	**NO**	**EXPENSIVE**
8	GRINDING TABLE	YES	
9	60 TON PRESS	YES	
10	150 TON TRIM PRESS	YES	EXPENSIVE
11	**3000# Area HAMMER**	**NO**	**EXPENSIVE**
12	158 TON TRIM PRESS	YES	EXPENSIVE
13	HYDRAULIC BENDER	YES	
14	4" THREADER	YES	
15	4" BELT GRINDER	YES	
16	3000# Area FURNACE	YES	
17	BAND SAWS	YES	
18	**200# Area OPEN DIE HAMMER**	**NO**	**EXPENSIVE**
19	400# Area FURNACE	YES	
20	**400# Area OPEN DIE HAMMER**	**NO**	**EXPENSIVE**
21	600# Area FURNACE	YES	
22	**600# Area OPEN DIE HAMMER**	**NO**	**EXPENSIVE**
23	STONE GRINDER	YES	
24	HYDRAULIC BENDER	YES	
25	DUAL BELT GRINDER	YES	
26	BELT GRINDER	YES	
27	SMALL ROTOBLASTER	YES	
28	TEMPER FURNACE	YES	
29	QUENCH FURNACE	YES	
30	HORIZONTAL BORING MACHINE	YES	
31	3 POST HYDRAULIC PRESS	YES	
32	INDUCTION HEATER	YES	
33	3 POST HYDRAULIC BENDER	YES	
34	**DIE MILLING MACHINE**	**NO**	**EXPENSIVE**
35	DRILL PRESS	YES	
36	VERTICAL LATHE	YES	
37	VERTICAL MILL	YES	
38	TOOL GRINDER	YES	
39	SLOT FURNACE	YES	
40	2.5" UPSETTER	YES	EXPENSIVE
41	**1500# Area OPEN DIE HAMMER**	**NO**	**EXPENSIVE**
42	1500# Area FURNACE	YES	
43	DIE MILLING MACHINE	YES	
44	EDM MACHINE	YES	
45	VERTICAL MILL	YES	
46	TURRET LATHE	YES	
47	ENGINE LATHE	YES	
49	H.T. TESTING AREA	YES	
50	SMALL TUMBLER	YES	
51	LARGE TUMBLER	YES	
52	CLEAR COAT DIP TANK	YES	
53	MANUAL MACHINE SHOP	YES	EXPENSIVE
54	CNC MACHINE SHOP	YES	EXPENSIVE
48	SHIPPING DESK	YES	
55	SHIPPING AREA	YES	
56	1.5" UPSETTER	YES	EXPENSIVE
57	OUTSIDE PROCESSING	YES	

= Monuments

= External Operations

Machine shop is located about 10 miles away from forge shop

Combine Workcenter #48 and Workcenter #55

Figure A.2: Workcenter Information

Figure A.3: Current Layout

APPENDIX B

EXAMPLE DATASETS

1. ABB_50×25

No.	Part	Quantity	Routing
1	121813002	13	25→10→7→13→1→15
2	121824001	6	5→2→16
3	121957001	24	10→3→7→16
4	122004006	5	9→15→16
5	122894001	10	10→3
6	123101001	11	10→1→14→15→16
7	123309002	7	10→8→16
8	123309003	10	9→10→8→16
9	123519003	64	10→8→3→16
10	123799002	4	10→7→2→14→15→16
11	125232002	6	9→25→1→7→2→15→16
12	125883001	90	10→1→2→16
13	126558002	42	8→3→23→16
14	126560001	63	9→10→8→2→21→23→24
15	126566001	55	10→8→3→2→6→16
16	126568002	55	9→1→2→23→16
17	126570002	55	3→2→23→16
18	126574001	63	10→1→2→23→16
19	126575001	5	10→8→3→23→16
20	126576002	35	8→1→2→23→16
21	126577001	35	10→8→1→21→19→16
22	126579001	55	8→24→16
23	127539001	8	10→7→2
24	127552001	6	10→3→7→2→22→16
25	127646001	32	10→3→15→16
26	128240001	8	9→25→3→13→17→16
27	128412001	50	10→5→16
28	128479001	14	3→16
29	128551001	7	9→4→17→2→12→16
30	128551002	10	9→4→17→2→16→12
31	128591001	21	10→1→18→16
32	129684001	4	10→7→13→17
33	129736001	9	9→4→14→15→16
34	130851001	2	10→7→16
35	130965001	39	10→3→16
36	131853001	2	10→7→11
37	132008001	22	10→1→20
38	62076001	4	9→10→16
39	632420019	27	9→5→7→2→16
40	6330001	22	10→3→2→3→8→2→16
41	68800001	10	2→13→1→2→14→15→16
42	69307001	2	25→3→2→18
43	69308001	1	25→3→2→18
44	72189001	40	10→1→14→15→16
45	72191001	7	10→2→3→2→18→2→16
46	74909001	6	10→7→3→2→18→15→16
47	82546001	7	10→2→3→2→18→2→16
48	82551005	3	9→6→18
49	82556001	5	10→1→2→18→2→16
50	85883001	8	9→25→4→3→2→20→16

Table B.1: Product routings - ABB_50×25

Work Center No	Description	Cost
1	1	1
2	2	1
3	3	100
4	4	100
5	5	100
6	6	100
7	7	100
8	8	100
9	9	100
10	10	1
11	11	100
12	12	100
13	13	100
14	14	100
15	15	100
16	16	1
17	17	100
18	18	100
19	19	100
20	20	100
21	21	100
22	22	100
23	23	100
24	24	100
25	25	100

Table B.2: Machine purchasing costs - ABB_50×25

2. Carrie_24×36

No.	Part	Quantity	Routing
1	1325500A	20	12→2→3→5→6→2→10
2	13255400A	90	12→2→3→5→6→9→10
3	14826100A	1	12→6→2→3→2→4→10
4	1516700A	48	1→2→3→5→4→8→6→8→10
5	15185300A	48	1→2→3→5→6→10
6	1521500A	36	12→2→3→10
7	15252500A	500	12→2→10
8	15538500A	19	12→2→3→10
9	6K01C000301	6	1→2→3→4→5→8→6→5→7→10
10	6K01C000406	5	1→2→3→4→5→8→6→5→7→10
11	A14519900A	16	1→2→14→4→5→6→9→10
12	A14529000A	1	12→2→13→3→2→9→10
13	A14639800A	46	12→2→3→9→10
14	A14734400A	1	12→2→6→3→10
15	A14827200A	1	1→2→8→9→2→4→10
16	A15322300A	98	12→2→3→5→4→6→9→10
17	A90227900A	69	1→2→5→6→4→9→10
18	B14519800A	16	1→2→3→4→5→6→7→10
19	B14528900A	1	1→2→3→11→4→8→10
20	B14829300A	1	2→3→5→4→6→7→6→7→10
21	B14829400A	1	2→3→5→4→6→10
22	B15165200A	48	1→2→13→3→6→5→9→10
23	B9023200A	6	1→2→3→4→5→6→7→5→10
24	C14465700A	1	1→2→3→4→5→6→7→8→9→10
25	D13256900A	8	12→2→3→8→10

Table B.3: Product routings - Carrie_24×36

Work Center No	Description	Cost
1	1	1000
2	2	1000
3	3	1
4	4	1000
5	5	1
6	6	1000
7	7	1000
8	8	1000
9	9	1000
10	10	1000
11	11	1000
12	12	1000
13	13	1000
14	14	1000

Table B.4: Machine purchasing costs - Carrie_24×36

3. Purcheck_28×18

No.	Part	Quantity	Routing
1	1	23	1->2->3->5->7->8->19->24
2	10	10	1->3->5->6->7->9->10->19->24
3	11	5	1->2->5->8->19
4	12	64	1->5->8->19
5	13	76	1->8->19
6	14	23	1->7->8->19->20
7	15	64	1->3->8->13->19
8	16	23	1->2->5->19
9	17	98	1->5->6->7->10
10	18	25	1->5->7->10
11	19	75	1->19->22
12	2	34	1->6
13	20	23	1->10->11->12->23
14	21	54	1->2->7->8->19
15	22	12	1->5->12
16	23	52	1->7->8->19
17	24	12	1->7->19->21
18	25	98	1->5
19	26	23	1->2->5->8->9->10->12->19
20	27	9	1->5->8->19->9
21	28	12	1->8->9->19
22	3	65	1->6->7->19
23	4	34	1->6->19->20
24	5	97	6->22
25	6	23	1->6->19
26	7	65	1->6->20
27	8	21	1->6->10->20
28	9	31	1->5->19

Table B.5: Product routings - Purcheck_28×18

Work Center No	Description	Cost
1	1	1
2	2	1000
3	3	1000
5	5	1000
6	6	1000
7	7	1
8	8	1000
9	9	1000
10	10	1000
11	11	1000
12	12	1000
13	13	1000
19	19	1000
20	20	1000
21	21	1000
22	22	1000
23	23	1000
24	24	1000

Table B.6: Machine purchasing costs - Purcheck_28×18

4. Sekine_13×12

No.	Part	Quantity	Routing
1	1	10	6->9->10->11->12
2	10	21	2->9
3	11	10	3->9->10->12
4	12	43	3->6->4->10->12
5	13	87	4->6->4->10->12
6	2	21	4->6->9->10->11->12
7	3	4	5->8->9->10
8	4	13	4->7->9->10
9	5	76	3->7->10->12
10	6	23	1->7->9->10
11	7	47	1->8->9->10
12	8	96	4->7->9
13	9	34	2->7->9

Table B.7: Product routing - Sekine_13×12

Work Center No	Description	Cost
1	1	100
10	10	100
11	11	100
12	12	100
2	2	100
3	3	100
4	4	100
5	5	100
6	6	100
7	7	1
8	8	100
9	9	1

Table B.8: Machine purchasing costs - Sekine_13×12

5. Tacomet_42×15

No.	Part	Quantity	Routing
1	18150	900	8->94
2	18164	720	8->10->8->O.V.->94
3	18179	720	8->10->O.V.->18->22->94
4	21097	900	18->96->92->96->92->18->94
5	21275	900	91->90->O.V.->18->96->92->96->92->8->94
6	212751	900	91->25->94
7	212752	113	8->94
8	212753	113	8->94
9	212754	900	8->94
10	212755	900	8->94
11	21306	5603	8->O.V.->18->96->92->96->92->8->10->94
12	25043	720	8->10->94
13	25896	720	20->91->O.V.->18->94
14	258961	720	8->10->8->94
15	258962	720	8->10->92->94
16	259863	720	8->94
17	26033	900	8->10->O.V.->18->96->92->96->92->8->94
18	26034	900	8->10->O.V.->18->96->92->96->92->8->94
19	26035	900	91->18->94
20	260351	900	8->10->O.V.->94
21	260352	7200	8->94
22	260353	900	8->10->94
23	26036	900	8->10->O.V.->18->96->92->96->92->8->94
24	26037	900	91->8->94
25	260371	900	8->10->O.V.->94
26	260372	7200	8->94
27	26038	900	8->94
28	26039	900	8->10->O.V.->18->96->92->96->92->8->94
29	26083	900	8->10->DFLOW->O.V.->18->94
30	26392	900	91->90->O.V.->18->96->92->96->92->8->94
31	263921	900	91->25->94
32	263922	900	8->94
33	263923	900	8->94
34	263924	900	8->94
35	263925	900	8->94
36	26440	900	8->19->O.V.->18->25->94
37	26610	180	19->92->8->10->O.V.->18->94
38	270373	900	8->10->DFLOW->94
39	27464	720	91->8->92->8->10->22->94
40	274641	720	8->10->O.V.->94
41	274642	720	8->10->19->8->23->94
42	2764643	720	19->25->94

Table B.9: Product routings - Tecomet_42×15

Work Center No	Description	Cost
8	8	1
10	10	1000
18	18	1000
19	19	1000
20	20	1000
22	22	1000
23	23	1000
25	25	1000
90	90	1000
91	91	1000
92	92	1000
94	94	1000
96	96	1000
DFLOW	DFLOW	1000
O.V.	O.V.	1

Table B.10: Machine purchasing costs - Tecomet_42×15

BIBILOGRAPHY

1. Achanga, P., Shehab, E., Roy, R. and Nelder, G., "Critical success factors for lean implementation within SMEs," *Journal of Manufacturing Technology Management*, Vol. 17, No. 4, pp.460-471 (2006)

2. Artiba, A. and Elmaghraby, S.E., *The planning and scheduling of production systems : methodologies and applications*, Chapman & Hall, New York (1997)

3. Askin, R.G. and Zhou, M., “Formation of independent flowline cells based on operation requirements and machine capacities,” *IIE Transactions*, Vol. 30, pp. 319-329 (1998)

4. Askin, R.G., Lundgren, N.H. and Ciarallo, F., “A material flow based evaluation of layout alternatives for agile manufacturing,” in *Progress in Material Handling Research* (Ed: Graves R.J. et al), Ann Arbor, MI: Material Handling Institute, Braun-Brumfield Inc., pp. 71-90 (1996)

5. Babu, S.A., Nandurkar, K.N., Thomas, A., “Development of virtual cellular manufacturing systems for SMEs.” Logistics Information Management, Vol. 13, No.4, pp. 228-243 (2000)

6. Balakrishnan, J. and Cheng, C.H., “Multi-period planning and uncertainty issues in cellular manufacturing: A review and future directions,” *European Journal of Operational Research*, Vol. 177, pp. 281-309 (2007)

7. Burbridge, J.L., “Production Flow Analysis,” *Production Engineer*, Vol. 42, pp. 742-752 (1963)

8. C. M. Fiduccia and R. M. Mattheyses, “A linear time heuristic for improving network partitions,” in *Proceeding 19th Design Automation Conference*, pp. 175-181 (1982)

9. Conner, G., *Lean Manufacturing for the Small Shop*, Society of Manufacturing Engineers, Dearborn, Michigan (2001)

10. Crowson, R., *Product Design and Factory Development (Handbook of Manufacturing Engineering, Second Edition)*, CRC Press, Boca Raton, Florida (2005)

11. D. W. Matula and F. Shahrokhi, “The maximum concurrent flow problem and sparsest cuts,” *Technical Report*, Southem Methodist University (1986)

12. Evans, J.R., *Applied Production and Operations Management*, West Publishing Company, St. Paul, MN (1983)

13. Fogarty, D.W., Hoffmann, T.R. and Stonebraker, P.W., *Production and operations management*, South-Western Pub. Co., Cincinnati, OH (1989)

14. Framinan, J.M. and Ruiz-Usano, R., “On transforming job-shops into flow-shops,” *Production Planning & Control*, Vol.13, No. 2, pp. 166-174 (2002)

15. Framinan, J.M., “Efficient heuristic approaches to transform job shops into flow shops,” *IIE transactions*, Vol. 37, pp.441-451 (2005)

16. Garey, M.R. and Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Company, New York (1979)

17. Gomory, R.E. and Hu, T.C., “*Multi-terminal network flows*,” J. SIAM, Vol. 9, No. 4, pp.551-570 (1961)

18. Ham, I., Hitomi, K., and Yoshida, T., *Group Technology*, Kulwer-Nijhoff Publishing, Boston, MA (1985)

19. Harhalakis, G., Lu, T., Minis, I. and Nagi, R., “A Practical Method for Design of Hybrid-Type Production Facilities,” *International Journal of Production Research*, Vol. 34, pp. 897-918 (1996)

20. Harhalakis, G., Nagi, R. and Proth, J. M., “An Efficient Heuristic in Manufacturing Cell Formation for Group Technology Application.” *International Journal of Production Research*, Vol. 28, pp. 185-198 (1990)

21. Ho, Y. and Moodie, C.L., “Machine layout with a linear single-row flow path in an automated manufacturing system,” *Journal of Manufacturing Systems*, Vol. 17, No. 1, pp. 1-22 (1998)

22. Irani, S.A. and Huang, H., “Cascading flowlines and layout modules: Practical strategies for machine duplication in facility layouts,” *International Journal of Flexible Manufacturing Systems*, Vol. 17, No. 2, pp. 119-149 (2005)

23. Irani, S.A. and Huang, H., “Custom design of facility layouts for multi-product facilities using Layout Modules,” *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 3, pp. 259-267 (2000)

24. Irani, S.A. and Huang, H., “Layout Modules: A Novel Extension of hybrid Cellular Layouts,” *Proceedings of the 1998 ASME International Mechanical Engineering Congress & Exposition and Winter Annual Meeting of the ASME, Anheim, CA*: American Society of Mechanical Engineerings, Paper # 98-WA/MH 1, Novermber 15-20 (1889)

25. Irani, S.A., Cavalier, T.M., Cohen, P.H., “Virtual manufacturing cells: Exploiting layout design and intercell flows for the machine sharing problem.” *International Journal of Production Research*, Vol. 31, No. 4, pp. 2901-2923(1993)

26. Kadipasaoglu, S.N., Xiang, W. and Khumawala, B.M., "Batch Scheduling in Multistage, Multiproduct Manufacturing Systems - An Application”, *International Journal of Operations and Production Management*, Vol. 19, No. 3, pp. 421-436 (1999)

27. Kandiler, L., “A cell formation algorithm: Hypergraph approximatation – Cut tree,” European Journal of Operational Research, Vol. 109, pp. 686-702 (1998)

28. Kannan, V.R. and Ghosh, S., “Cellular manufacturing using virtual cells,” *International journal of operations & production management*, Vol. 16, pp. 99-112 (1996)

29. Kouvelis, P. and Chiang, W.C., “A Simulated Annealing Procedure for Single Row Layout Problems in Flexible Manufacturing Systems,” *International Journal of Production Research*, Vol. 30, No.4, pp. 717-732 (1992)

30. Lahmar, M. and Benjaafar, S., “Design of distributed layouts,” IIE Transactions, Vol. 37, pp. 303-318 (2005)

31. Lee, S. and Chiang, C., “A cut-tree-based approach for clustering machine cells in the bidirectional linear flow layout,” International Journal of Production Research, Vol. 39, No.15, pp. 3491-3512 (2001)

32. Levassuer, G. A., Helms, M. M. and Zink, A. A., “Conversation from a functional to a cellular manufacturing layout at Steward, inc.,” *Production and Inventory Management journal*, Vol. 36, pp. 37-42 (1995)

33. Li, J. and Barnes, D.J., “Investigating the factors influencing the shop performance in a job shop environment with kanban-based production control,” *International Journal of Production Research*, Vol. 38, No. 18, pp. 4683-4699 (2000)

34. Logendran, R. and Ramakrisha, P., “Manufacturing cell formation in the presence of lot splitting and multiple units of the same machine,” *International Journal of Production Research*, Vol.33, No. 3, pp. 675-693 (1995)

35. Matula, D. W. and Shahrokhi, F., “The maximum concurrent flow problem and sparsest cuts,” *Technical Report*, Southem Methodist University (1986)

36. McLean, C.R., Bloom, H.M. and Hopp, T.H., “The virtual manufacturing cell,” *Proceedings of the 4th IFAC/IFIP Conference on Information Control Problems in Manufacturing Technology*, pp. 105-111 (1982)

37. Meller, R.D. and Gau, K., “The facility layout problem: recent and emerging trends and perspectives,” *Journal of Manufacturing Systems*, Vol. 15, No. 5, pp. 351-366 (1996)

38. Miltenburg, J., *Manufacturing Strategy*, Productivity Press, Portland, Oregon (1995)

39. Montreuil, B., Lefrancois, P, Marcotte, S. and Venkatadri, U., “Holographic layout of manufacturing systems operating in chaotic environments. *Technical Report* 93-53, Laval University, Québec, Canada (1993)

40. Montreuil,B. and Ratliff, H.D., “Utilizing cut trees as design skeletons for facility layout,” IIE Transactions, Vol. 21, pp. 136-143 (1989)

41. Nelson, K., Lean Manufacturing: Adapting as Important as Adopting, www.iSixSigma.com

42. Ohno, T., *The Toyota Production System: Beyond Large-Scale Production*, Productivity Press, Portland (1988)

43. Ourairat, A., *The Machine Duplication Problem (MDP) in Design of Flowline Layouts*, Ph.D. Dissertation, The Ohio State University (2000)

44. Papadopoula, T.C. and Ozbayrak, M., “Leanness: experience from the journey to date,” *Journal of Manufacturing Technology Management*, Vol. 16, No. 7, pp. 784-807 (2005)

45. Prince, J. and Kay, J.M., “Combining lean and agile characteristics: creation of virtual groups by enhanced production flow analysis,” *International Journal of Production Economics*, Vol. 85, pp. 305-318 (2003)

46. Prince, J. and Kay, J.M., “Combining lean and agile characteristics: creation of virtual groups by enhanced production flow analysis,” *International Journal of Production Economics*, Vol. 85, pp. 305-318 (2003)

47. Raiha, K.J. and Ukkonen, E., “The shortest common supersequence problem over binary alphabet is NP-complete, *Theoretical Computer Science*, Vol. 16, pp. 187-198 (1981)

48. Sarker, B.R., Wilhelm, W.E., Hogg, G.L. and Han, M., Backtracking of Jobs in One-Dimensionla Machine Location Problems, *European Journal of Operational Research*, Vol. 85, pp. 593-609 (1995)

49. Shukla, J., *Hybrid and Progressive Cellular Layouts for Jobshops*, Master of Science Thesis, Department of Mechanical Engineering, University of Minnesota, Minneapolis, MN (1995)

50. Stock, D.J. and Lindley, R.J., “Batch production using process sequence cells,” *International Journal of Production Research*, Vol.36, No. 9, 2419-2433 (1998)

51. Stock, D.J. and Lindley, R.J., “Implementing kanbans within high variety/low volume manufacturing environments,” *International Journal of Operations and Production Management*, Vol. 15, No. 7, pp. 47-59 (1994)

52. Suresh, N.C., Slomp, J., “Performance of virtual cellular manufacturing with functional and cellular layouts in DRC settings,” *International Journal of Production Research*, Vol. 43, No. 5, pp. 945-979 (2005)

53. Suri, R., *Quick Response Manufacturing: A Companywide Approach to Reducing Leadtimes*, Productivity Press, Portland, OR (1998)

54. Suzaki, K., *The New Manufacturing Challenge: Techniques for Continuous Improvement*. New York, NY: The Free Press (1987)

55. Tompkins, J.A., While, F.A., Bozer, Y.A., Frazelle, E.H., Tanchoco, J.M.A., and Trevino, J., *Facilities Planning*, New York, NY: John Wiley (1996)

56. Venkatadri, U., Rardin, R. and Montreuil, B., “A design methodology for fractal layout organization,” *IIE Transactions*, Vol. 29, pp. 911-924 (1997)

57. Wei, Y. and Chieng, C., “Ratio cut partitioning for hierarchical design,” *IEEE Transactions On Computer-Aided Design*, Vol. 10. No. 7, pp. 911-921 (1991)

58. Wemmerlov, U. and Hyer, N.L., "Cellular manufacturing in the US industry: a survey of users," International Journal of Production Research, Vol. 27, pp. 1530-1551 (1989)

59. Womack, J.P. and Jones, D.T., *Lean Thinking*, Simom & Schuster, New York, NY (1996)