공학박사 학위논문

재구성가능 생산셀의 통합 의사결정시스템에 관한 연구

A Study on Integrative Decision-making System for Reconfigurable Manufacturing Cells

2014년 2월

서울대학교 대학원

산업공학과

서진우

재구성가능 생산셀의 통합 의사결정시스템에 관한 연구

지도교수 박 진 우

이 논문을 공학박사 학위논문으로 제출함
2013년 11월

서울대학교 대학원
산업공학과
서 진 우

서진우의 공학박사 학위논문을 인준함
2013년 12월

위 원 장 ______________________ (인)

부위원장 ______________________ (인)

위　　원 ______________________ (인)

위　　원 ______________________ (인)

위　　원 ______________________ (인)

Abstract

A Study on Integrative Decision-making System for Reconfigurable Manufacturing Cells

Jinwu Seo

Department of Industrial Engineering

The Graduate School

Seoul National University

RMCs (Reconfigurable Manufacturing Cells) are production systems which can rapidly change hardware configurations to respond to market variance. The practical range of such change time is one or two days. In the operation of RMCs, managers encounter two decision problems with regard to how reconfiguring capability of manufacturing systems would be utilized.

The first problem is to decide whether and how hardware configurations of manufacturing systems are to be changed. An alternative configuration may improve manufacturing performance in response to demand, but detailed resource allocation needs to be examined together with capacity loss during reconfiguration for accurate evaluation so that managers may make decisions with certainty. However, most of previous studies on hardware reconfiguration dealt it as a tactical problem rather than an operational problem, not providing accurate performance evaluation of manufacturing systems.

The second problem is about purchase of equipment. Managers want to know whether their current systems may catch up with future demand or additional equipment needs to be purchased. In this case, reconfiguring capability of manufacturing systems should be considered to satisfy demand with minimized purchasing cost.

This study aims to develop an integrative decision-making system for those decision problems. To provide scheduling-level resolution of hardware reconfiguration, a scheduling algorithm appropriate for RMC environment is developed. And using the developed scheduling algorithm, a mathematical model for hardware reconfiguration is presented in consideration with capacity loss during reconfiguration. To solve the model, a relaxation based branch-and-bound algorithm is proposed. Finally, to resolve the equipment purchase problem, an algorithm which minimizes the net-present-value of purchasing cost is suggested.

Keywords: Reconfigurable Manufacturing Cell, Scheduling, Hardware Reconfiguration, Algorithm
Student Number: 2008-21225

Contents

영문초록 i

1. Introduction 1

1.1. Environment of Research 1

1.2. Purpose of Research 5

2. Related Work 7

2.1. Previous Study on Hardware Reconfiguration........................9

2.2. Previous Study on FMS Scheduling 14

2.3. Summary of Related Work 17

3. Problem Definition: Managerial Aspect.......................19

3.1. Description of Production Environment 19

3.2. Definition of Managerial Problems......................................22

4. Scheduling Algorithm for RMC....................................26

4.1. Scheduling Environment..28

4.2. Mathematical Model 31

4.3. Development of Algorithm 35

4.3.1. An Example for the Algorithm.. 42

4.4. Experiment 47

4.5. Result and Analysis...49

5. A Framework for Hardware Reconfiguration.............57

5.1. Mathematical Model 59

5.2. Solution Procedure for Hardware Reconfiguration: Branch-and-bound Approach...64

5.2.1. Relaxation of the Problem 64

5.2.2. Relaxation based Branch-and-bound Algorithm 74
5.3. Experiment 84
5.4. Result and Analysis 86
6. RMC Equipment Purchase Planning 93
6.1. Mathematical Model 95
6.2. Development of Algorithm 99
6.3. Experiment 102
6.4. Result and Analysis 103
7. Conclusion 111
Appendix 114
References 115
국문초록 **128**

List of Tables

<Table 3.1> The summary of managerial problems and research direction.....24
<Table 4.1> Dispatching information for the example of the proposed algorithm45
<Table 4.2> Dispatching rules used in experiment as alternatives .48
<Table 4.3> The experimental result for the performance measure51
<Table 4.4> The experimental result for computational time 53
<Table 5.1> The experimental result 87
<Table 5.2> The result of hardware reconfiguration problem 90
<Table 6.1> An example of the purchasing alternative list............101
<Table 6.2> The experimental result 104
<Table 6.3> The result of reconfigurable equipment purchase problem 106
<Table 6.4> Cost of equipment purchase 109
<Table 6.5> Benefit of equipment purchase: Reduced production delay110

List of Figures

<Figure 1.1> A typical RMC 1
<Figure 1.2> The concept of reconfiguring 2
<Figure 1.3> The reconfiguring problem 3
<Figure 1.4> Manufacturing performance of a configuration and RMC schedules .3
<Figure 3.1> Production environment: A typical layout of a factory21
<Figure 3.2> The relationship among the managerial problems 25
<Figure 4.1> The relationship between the scheduling problem and the reconfiguring problem 27
<Figure 4.2> An example of the pallet-fixture assignment 29
<Figure 4.3> Impact of resource idleness 37
<Figure 4.4> The work-in-process projection...................................38
<Figure 4.5> The example setting 43
<Figure 4.6> The experimental result for the performance measure55
<Figure 4.7> The experimental result for the performance measure55
<Figure 4.8> The experimental result for computational time 56
<Figure 5.1> The case when hardware reconfiguration is considered.....57
<Figure 5.2> The capacity change around hardware reconfiguration58
<Figure 5.3> Reconfiguring time according to change of cell configurations....60
<Figure 5.4> The flow chart of RPX algorithm................................74
<Figure 5.5> The flow chart of the branch-and-bound algorithm..75
<Figure 5.6> Pseudo code of the neighborhood search algorithm..82
<Figure 5.7> The result of hardware reconfiguration problem in graphical representation 91
<Figure 5.8> The result of hardware reconfiguration problem in change of schedules92

<Figure 6.1> The flow chart of REPP algorithm............................100

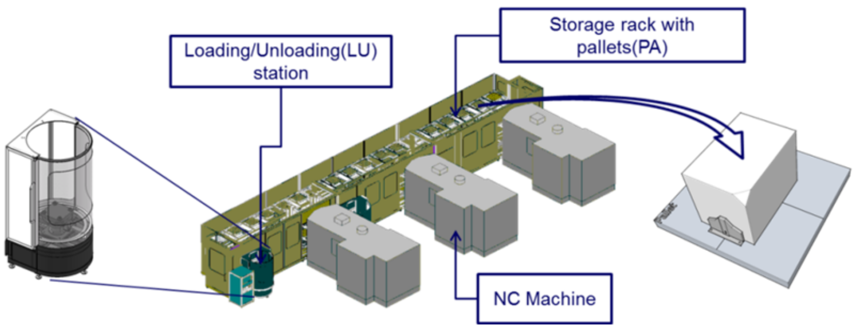
<Figure 6.2> The result of reconfigurable equipment purchase problem in graphical representation..108

<Figure A.0.1> The optimal schedules: the non-permutation schedule (a) and the permutation schedule (b) 114

1. Introduction

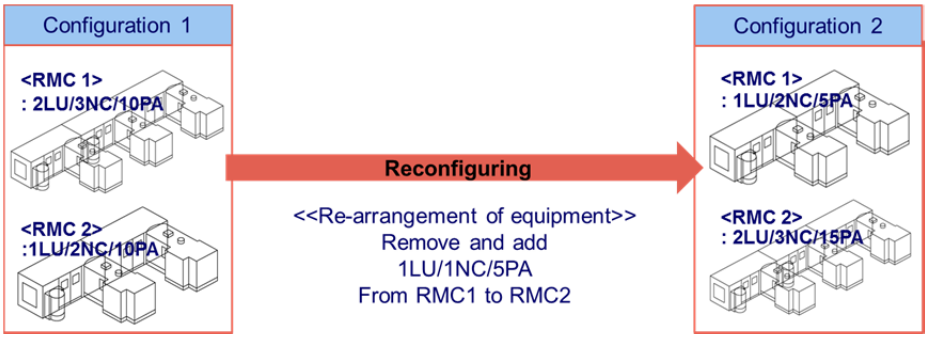
1.1. Environment of Research

RMCs (**Reconfigurable Manufacturing Cells**) are production systems which can rapidly change hardware configurations to respond to market variance. An RMC consists of loading/unloading (L/U) stations, NC machines (e.g. horizontal machining centers), material handling devices (e.g. AGVs), and storage rack of pallets. <Figure 1.1> shows a typical RMC in development.



<Figure 1.1> A typical RMC

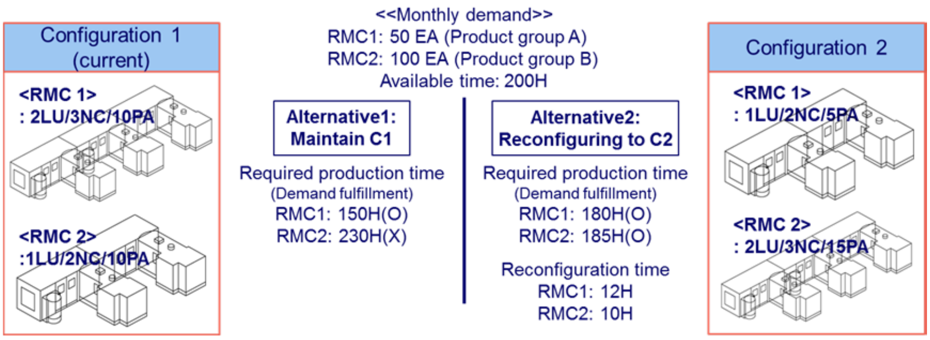
Reconfiguring is re-arrangement of equipment among RMCs, to adjust production capacity among product groups. Reconfigurable equipment is L/U stations, NC machines, and pallets. <Figure 1.2> indicates the concept of reconfiguring. Note that the total number of equipment of the whole factory does not change.



<Figure 1.2> The concept of reconfiguring

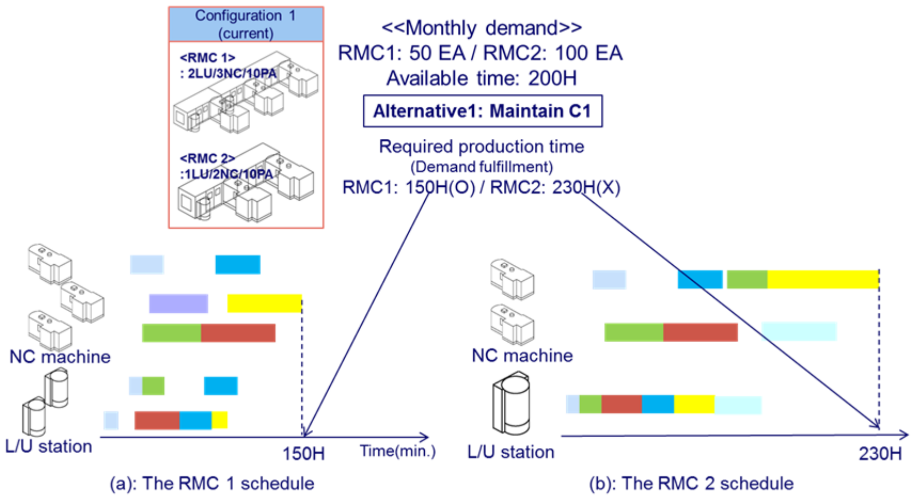
Since market demand is ever-changing, the current configuration may not be appropriate to respond to demand. In this case, production managers are concerned with performance improvement of their manufacturing systems through hardware reconfiguration. <Figure 1.3> indicates different manufacturing performance according to hardware configurations. In <Figure 1.3>, 'Configuration 2'(C2) performs better than 'Configuration 1'(C1), since with C2 both cells may finish production within the available time, while production delay may happen in RMC 1 with C1. Although it takes time to change the hardware configuration from C1 to C2, as in <Figure 1.3>, C2 may satisfy demand on time.

Since various alternative configurations may exist, production managers wonder which hardware configuration would perform best. The **Reconfiguring problem** is to find the best hardware configuration of RMCs in response to demand.



<Figure 1.3> The reconfiguring problem

To evaluate manufacturing performance of a configuration, scheduling may be required. The required production time of each RMC in <Figure 1.3> was obtained by generating schedules given a hardware configuration, as shown in <Figure 1.4>.



<Figure 1.4> Manufacturing performance of a configuration and RMC schedules

Scheduling-level evaluation of hardware configurations provides reliable information for decision-making. The schedule specifies detailed resource allocation over time and close to actual production, so it would be reliable to compare configuration alternatives based on their schedules. With scheduling-level evaluation, therefore, managers may make decisions with certainty.

Scheduling-level evaluation becomes more practical with the advent of RMCs. When the concept of RMS (reconfigurable manufacturing system) was introduced in late 1990s, it took one month or more to change hardware configurations. So reconfiguration decisions have been considered as long-term (over a year) decisions [18][52][80][103]. In order to respond to ever-changing demand more quickly, development of manufacturing systems with small changeover cost was requested. In response, a Korean governmental research project for the development of RMC (reconfigurable manufacturing cell) was launched in 2009 [42]. In the manufacturing system with RMCs, reconfiguration happens not throughout the entire factory but within RMCs, and components consisting of RMCs are highly modularized. As a result, time required for hardware reconfiguration has been reduced to few days. (The final objective of changeover time in the research project is one day.) As short-term demand forecast is relatively accurate and difference from actual production is relatively small, scheduling-level evaluation of RMCs would be meaningful for reliable decision-making.

1.2. Purpose of Research

This study aims to develop an integrative decision-making system for RMC environment. The system provides following functions:

- Scheduling-level resolution of the RMC reconfiguring problem: Here, manufacturing performance in response to market demand is considered as the primary objective, and capacity loss during reconfiguration is examined. Then reconfiguring cost is considered as the secondary objective.

- RMC equipment purchase planning: In long-term aspect, it may be required to purchase additional equipment. Considering reconfiguring capability of manufacturing systems, the best purchase plan, with which demand may be fulfilled in minimized purchasing cost, is considered. This is an extension of the reconfiguring problem.

This thesis is organized as follows. Chapter 2 reviews previous research on the research problem. In section 3.1, production environment is explained and in section 3.2, the research problem is redefined in managerial aspect. In chapter 4, a scheduling algorithm appropriate for hardware reconfiguration is developed as a basic component of the overall problem-solving process. Then in chapter 5, using the developed scheduling algorithm, a mathematical model for the reconfiguring problem is presented and a relaxation based branch-and-bound algorithm is proposed to solve that model. In chapter 6, an algorithm for minimizing net-present-value of purchasing cost is suggested for equipment purchase planning. Finally it ends up with conclusion in chapter 7.

2. Related Work

The concept and design of manufacturing systems with re-configurability was investigated by early researchers from mid 1990s. Adlemo et al. [4] classified flexibility of manufacturing systems into long time-scale, shorter time-scale and very short time-scale flexibility according to the degree and lasting duration of system change, and presented the corresponding software structure and a product model to support introduction of new equipment and products to existing systems. Koren et al. [35] (refer also [33]), Mehrabi et al. [52] and ElMaraghy [18] suggested the concept of reconfigurable manufacturing system(RMS)s, and each proposed architecture to design and operate those systems. Makino and Trai [48] categorized RMSs as static and dynamic according to whether resources are just used like building blocks or advanced material handling systems support production.

Abdi and Labib [1] presented an AHP model for a design strategy of RMSs, where various planning horizons and actors were considered together with manufacturing objectives and criteria, and the authors extended the subject to grouping products into product families based on operation similarities in [2], and to selecting hardware configurations in [3]. Mehrabi et al. [53] investigated on the development direction of RMSs in aspects of system design and hardware/software architecture. Mehrabi et al. [54] conducted a survey to understand needs and expectations in development of RMSs.

Tilbury and Kota [90] investigated on design of reconfigurable machine tools to manufacture a given part family together with associated control modules. Matt and Rauch [51] presented functional requirements and design parameters for scalable modular manufacturing systems to support geographically distributed production of mass customized goods.

This study is concerned with hardware reconfiguration of manufacturing systems. Previous studies on the reconfiguring problem are reviewed in the next section. Studies that approach the problem in the scheduling level are discussed separately. Then studies on pallet adjustment follow. Since production in RMCs has similarity with that in FMSs and flowshops, relevant studies on those topics are also discussed.

2.1. Previous Study on Hardware Reconfiguration

There have been studies that were concerned with optimization and/or selection of hardware configurations of manufacturing systems.

Koren et al. [34] considered system level reconfiguring - changing the number of production stages and the number of machines for each stage - and analyzed the effect of different configurations on manufacturing performance in terms of expected productivity, initial setup cost, and scalability cost. Koren and Shpitalni [31] further investigated system level reconfiguring, focusing on eliminating impractical or unsatisfactory configurations from all possible alternatives. Wang and Koren [91] proposed scalability planning problems of RMSs to minimize the number of new machines required, and suggested a solution procedure using the genetic algorithm. Tang et al. [88] presented the computer-aided reconfiguration plan framework, where the reconfiguration plan problem was modeled as a network of potential reconfiguring activities and both system and machine level reconfiguring was considered. Here machine level reconfiguring indicates adjusting additional capability of equipment (by adding/removing spindles) and setting fixtures. (Refer also [32].) Using the result of [86][99], in [87], selection principles of manufacturing systems was presented based on the cost model and the utilization rate. Youssef and ElMaraghy [103] proposed automatic generation methods of system and machine level hardware reconfiguration considering operation clusters setups and hardware space limitation. In [104], the reconfiguration smoothness metric considering anticipated reconfiguration process was defined in market, system and machine perspectives. And combining the result of [103][104], in [106], RMC configuration selection approach was developed using meta-heuristic algorithms. (Refer also [102].) Spicer and Carlo [80] investigated on the reconfiguring cost structure, and suggested a dynamic programming based algorithm to determine the cost-minimizing hardware reconfiguration path for multi periods. Xiaobo et al. [94] considered manufacturing systems which change hardware configurations according to product groups to be manufactured and proposed a reconfiguration framework based on stochastic arrival rates and processing times and investigated it further in [95][96][97]. In [107], selection of the optimal configuration was dealt with machine availability using the universal generating function (UGF), which was extended from [105]. Huang et al. [27] investigated an optimization model for configuration selection using the characteristic state equation. Bensmaine et al. [9] dealt with machine level reconfiguration considering tool approach directions of machines and multi objectives using the non-dominated sorting genetic algorithm, which Kumar et al. also employed for optimal configuration selection in [36]. Asl and Ulsoy [5] and Deif and EIMaraghy [14] applied approaches of feedback control for hardware reconfiguration. In [81], the effect of reconfiguration on throughput was investigated with consideration of machine availability. Meng [55] presented the petri-net model for RMSs which is divided into system level, process level, and cells level.

Studies that provide scheduling-level resolution of hardware reconfiguration are relatively few.

Ye and Liang [100] proposed an integrated model for scheduling of modular products and reconfiguring of manufacturing cells and suggested a genetic algorithm which determines cell configurations and job sequences simultaneously, but it did not consider capacity scalability. Nan et al. [58] and Zheng et al. [109] used the timed petri-net to generate the reconfiguration plan in combination with the corresponding schedule, considering reconfiguring time. The petri-net model, however, has limitation that its complexity dramatically increases as does the problem size. The suggested search technique based on the reachability graph was also appropriate for the small size problems. Seo and Park [70] and Seo et al. [71] investigated enhanced flexibility of RMCs compared to conventional FMCs using the constraint programming and sought to make integrative decisions for the reconfiguring problem in the scheduling level, but a systemic procedure to find good solutions were missing.

Other studies relevant to hardware reconfiguration are as follows.

Zhong et al. [110] considered quality in performance analysis of different system configurations based on the homogeneous transformation matrix [77]. Maier-Speredelozzi et al. [47] presented measures for convertibility of manufacturing systems, which is capability to adjust capacity and/or functionality of systems. Mun el. al [56] proposed software architecture for RMSs to facilitate a dynamic reconfiguration of system elements (i.e. fractals). Lee [41] presented design rules to minimize material handling cost where time required for reconfiguration was considered. Hasan et al. [26] investigated on an optimum sequence of a part family considering reconfiguring cost, and stochastic demand. Yigit et al. [101] was concerned with the optimum selection of module instances for products manufactured in RMSs considering the trade-off between quality and reconfiguring cost. Yamada et al. [98] investigated into optimization of the layout and allocation of transport robots and suggested meta-heuristic algorithms based on particle swarm optimization. Shah et al. [74] considered reconfiguration of the logic controllers for a small-scale manufacturing line.

Pallets are often critical resources in flexible manufacturing systems. There were studies on pallet adjustment.

Rahimifard and Newman [65] considered simultaneous scheduling of workpieces, fixtures and cutting tools and Seo and Park [68] discussed about how to improve the RMC scheduler’s practicality and presented a corresponding database structure, yet both of them were conceptual. Shalev-Oren and Schweitzer [75] extended [85] and presented an analytic model based on the mean value analysis to measure performance of FMSs according to priority rules considering pallets and fixtures and material handling time. The result was compared with the simulation model and appeared to have difference of around 4-13% for small size problems. Stecke and Kim [84], Solot and Bostos [79], and Solot [78] investigated into the optimal number of pallets for several part types using the queuing network. Mashaei et al. [50] suggested an analytic mathematical model and an IP model to determine the minimum number of pallets attaining the minimum cycle time, given a cyclic schedule (sequence) and derived inequalities about the minimum number and verified the result with the simulation optimization result, yet the fixtures were not considered. Denzler et al. [16] was concerned with impact of distribution of dedicated pallets on scheduling performance. Newman et al. [61] compared dedicated and general (shared) pallets considering the trade-off between the scrap rate and the utilization efficiency. Han et al. [21] dealt the case where the distribution of dedicated pallets is decidable and tried to improve manufacturing performance in an integrative manner.

2.2. Previous Study on FMS Scheduling

Given hardware configurations, the RMC scheduling problems have similar characteristics with the FMS scheduling problems. Below are researches about the FMS scheduling problems.

Sethi et al. [73] verified performance of the Gilmore-Gomory's algorithm in [20] for pallet-constrained flow shops. Yu et al. [108] dealt with scheduling problems of reconfigurable manufacturing cell, divided decisions into input sequencing, operation/machine selection, and part sequencing. It evaluated performance of various combinations of dispatching-based rules. Park et al. [62] considered the FMC scheduling problem under pallet constraints and presented a heuristic algorithm which reduces the setup delay. Na et al. [57] investigated the FMC scheduling and rescheduling problems during the transient disturbance period with pallet constraints. Mashaei and Lennartson [49] also dealt the pallet-constrained flowshops with purpose of reduction of energy consumption.

Above studies reflected pallet requirement for generation of schedules. The below studies did not consider pallets as scarce resources.

Park and Woo [63] examined robustness of the queuing network model in the FMS scheduling problem. Jang et al. [29] suggested an integrated decision support system for FMSs and proposed a periodic scheduling algorithm. Lee and Ryo [39] presented an algorithm responsive to shop situations applying different dispatching rules to machines. Woo et al. [93] solved the FMS loading and the FMS scheduling problems in an integrative manner. Chung et al. [11] developed a mixed integer programming model for the FMS scheduling problem in the control level and suggested control software architecture. Gultekin et al. [22] presented a cyclic scheduling algorithm for the FMC scheduling problem where the material handling time of a robot is critical. Lin and Lee [43] proposed a petri-net based scheduling scheme for FMCs to integrate scheduling and control.

Flexible manufacturing systems like RMCs are often modeled as flowshops in which parts are processed along machines in sequence.

Santos et al. [67] presented a lower bound scheme for flexible flowshops, meaningful for flowshops beyond three stages. Rajendran and Chaudhuri [66] suggested a branch-and-bound algorithm for flexible flowshops forcing permutation schedules. Azizoglu et al. [6] suggested a branch-and-bound algorithm for flexible flowshops not forcing permutation schedules. Guinet et al. [21] proved the NP-hardness of scheduling problems of two-stage flexible flowshops and suggested the 'sequence-first and allocate-second' heuristic, of which performance is within 0.73% gap to the three, presented lower bounds. Lee and Vairaktarakis [37] presented the $2-\frac{1}{m}$ approximation algorithm for hybrid flowshops and showed performance numerically that the gap is within the average of 1.6% and the maximum of 3%. Sriskandarajah and Sethi [82] presented an analytic model for flexible flowshops and proved that the list scheduling algorithm is the $3 - \frac{1}{m}$ approximation algorithm and suggested a heuristic algorithm. However it has limitation that the considered case is too specific. Narasimhan and Mangiameli [59] investigated impact of job allocations on the future status of machine idleness and waiting time in two-stage hybrid flowshops and suggested the generalized cumulative minimum deviation rule(GCMD) assuming permutation schedules, which is an extension of Narastmhan and Panwalkar [60]'s CMD rule which is for two-stage hybrid flowshops with a single machine on the first stage.

2.3. Summary of Related Work

Previous research considered the reconfiguring problem as a tactical problem for long-term demand rather than an operation problem for short to mid-term demand as Deif and EIMaraghy [14] pointed out that "the rate of variation in demand is usually much higher than the rate at which capacity can be changed and reconfiguration decision is tactical rather than operational". Capacity loss during reconfiguration was rarely considered, and systems were roughly modeled using the queuing network and/or the simple calculation to estimate manufacturing performance of hardware configurations. Improvement in reconfiguring cost and reconfiguring time with the advent of RMCs has not been reflected yet.

For researches about FMS scheduling problems, there has been no research that re-configurability is concerned in the development of scheduling algorithms. Such a scheduling algorithm is hardly found that explicitly considers environment where both hardware configurations and demand are ever-changing.

Our study deals with manufacturing environment where hardware reconfiguration may happen frequently for short-term demand and therefore scheduling-level evaluation is required for reliable decision making. In the development of the scheduling algorithm for such environment, it should be considered that both hardware configurations and demand are variable.

In the next chapter, production environment is explained and the reconfiguring problem is redefined in managerial aspect.

3. Problem Definition: Managerial Aspect

In this chapter, the reconfiguring problem is redefined in the view of production managers. At first, production environment is described.

3.1. Description of Production Environment

Description of production environment is as follows:

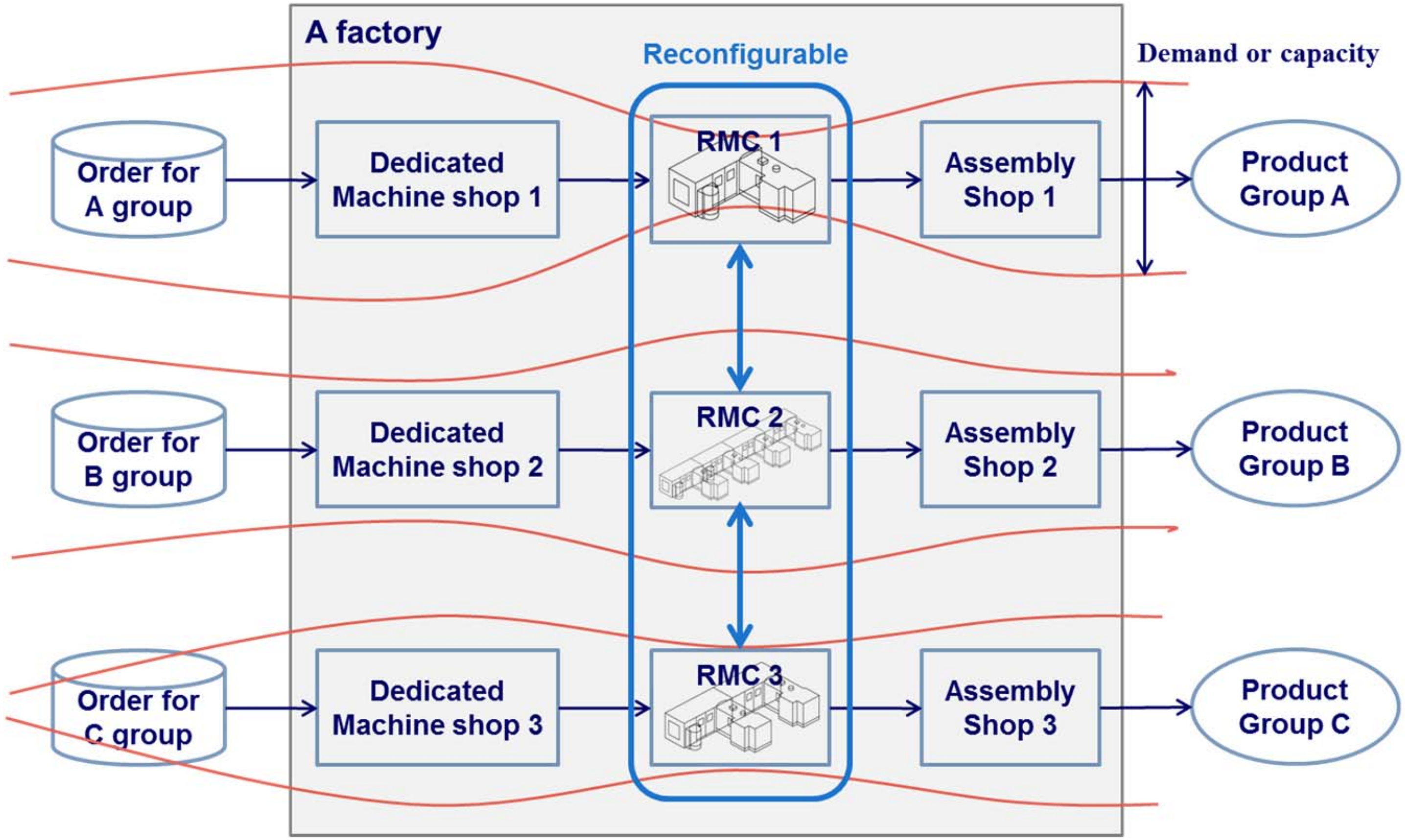
There exist multiple cells of RMCs in the factory, each for a production line and a production line is made up of dedicated and assembly shops together with an RMC. <Figure 3.1> shows a typical layout of a factory in consideration. The exemplary factory is divided into 3 production lines according to product groups, and each line consists of 3 stages including a single RMC. Though RMCs are available for a variety of machining operations, the factory may still maintain the traditional system of dedicated machines for various reasons such as high productivity and cost reduction. According to Koren and Shpitalni [31], it's better to employ the dedicated machine shop for repetitive and common operations. So it is more realistic that RMCs are used as a part of the production line. This shop environment is sometimes called a cellular reconfigurable manufacturing system (RMS) [108], or may be named as a multi-cell RMS according to [46].

An RMC consists of L/U stations, NC machines and pallets and those components are identical among the same types of components. Therefore an

RMC is defined by the number of L/U stations, NC machines and pallets. And each RMC is given a production order for a pre-defined demand period (e.g. monthly demand), which is defined by product types and volumes.

To respond demand, RMCs are reconfigurable: L/U stations and NC machines are interchangeable among RMCs in the factory. Pallets which support operations are also deliverable from one RMC to another. Reconfiguration includes removing or attaching of equipment, tool changes, NC program uploads, precision tests and adjustment, and all the other activities required to ramp up the manufacturing system with the new hardware configuration. Hardware reconfiguration is conducted before production begins. The other parts of the manufacturing system except RMCs are assumed to have fixed capacity.

Though it may be possible to redistribute work among RMCs, transferring work or raw material from one to another production line usually brings inefficiency and sometimes causes errors, because of physical constraints such as the factory layout. Therefore work redistribution among RMCs is not allowed. In this context the part-grouping problem [19][24][84] is not concerned and related decisions are assumed given.



<Figure 3.1> Production environment: A typical layout of a factory

3.2. Definition of Managerial Problems

Provided with the production environment, production managers in charge of operating RMCs may incur following decision problems.

The first one is the **configuration evaluation problem** (CEP), which is to check manufacturing performance of the current configuration. Given production demand, managers may want know whether the factory may satisfy demand with the current configuration or not. The result of CEP is manufacturing performance (e.g. demand fulfillment) of each RMC and corresponding schedules. Based on CEP result, managers may decide whether to maintain or change the current configuration. In addition, they may necessitate information about how much they have excess or shortage of capacity to make further decisions.

The second problem is the **reconfiguring problem** (RP) which is to find the best alternative configuration to improve manufacturing performance. The reconfiguring problem needs to be solved when it is verified through CEP that the current configuration may not meet demand, and there is excess capacity in some RMCs while other RMCs suffer from lack of capacity. The result of RP is the alternative configuration together with improved manufacturing performance and reconfiguring cost at that time. Based on this, managers may decide how to change the existing configuration.

Above two problems are related to response to short-term demand. Based on long-term demand forecast, managers may wonder if their current system is sufficient for production of future periods. If managers are not satisfied with the current system, they should find a way to enhance manufacturing performance and this may be done by purchasing additional equipment.

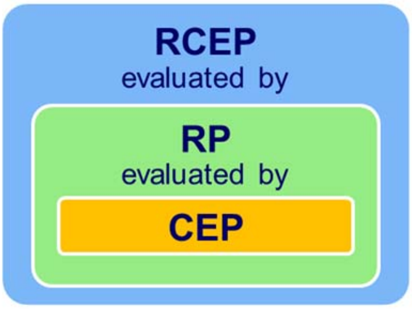
The **reconfigurable equipment phase problem** (REPP) is to find the best purchase plan for long-term demand. The result of REPP is the purchase plan that specifies which and how many equipment would be bought and when, together with the purchasing cost and manufacturing performance under that plan. Based on the result, managers may release purchase orders or make a budget for purchase of equipment. REPP is distinguished from the conventional equipment purchase problem in that reconfiguration of manufacturing systems is considered.

<Table 3.1> summarizes the managerial problems described above. <Table 3.1> also includes the research direction to support the managerial problems, together with relevant chapters.

<Table 3.1> The summary of managerial problems and research direction

	What production managers want to			***Direction of development to support managers***
	Know	See (metrics)	Decide	
Configuration Evaluation Problem	Manufacturing performance of the current system	Demand fulfillment	Whether to maintain or change the current system	Scheduling algorithm (Heuristic) → Chapter 4
Reconfiguring Problem	Costs and benefits of hardware reconfiguration	Demand fulfillment, Reconfiguring cost	How to change the current system	Configuration search algorithm (Relaxation based B&B) → Chapter 5
Reconfigurable Equipment Purchase Problem	Costs and benefits of equipment purchase	Demand fulfillment (long term) Purchasing cost	Whether/Which/ How many/When to purchase equipment	Purchase plan search algorithm (Exact) → Chapter 6

<Figure 3.2> shows the relationship among the managerial problems defined in the previous section. The reconfiguring problem relies on the result of the configuration evaluation problem, since manufacturing performance of every investigated configuration in the reconfiguring problem needs to be evaluated for comparison. On the other hand the reconfiguring problem may be a building block for the reconfigurable equipment purchase problem, for hardware reconfiguration should be considered to evaluate the effect of purchased equipment on manufacturing performance.



<Figure 3.2> The relationship among the managerial problems

As shown in <Figure 3.2>, the configuration evaluation problem is the most basic problem. Development of a scheduling algorithm for hardware reconfiguration is therefore very important as the configuration evaluation problem is solved by generating schedules. The scheduling algorithm may be considered as a basic component of the decision-making system this study aims to develop.

In this context, the scheduling algorithm is introduced in the next chapter.

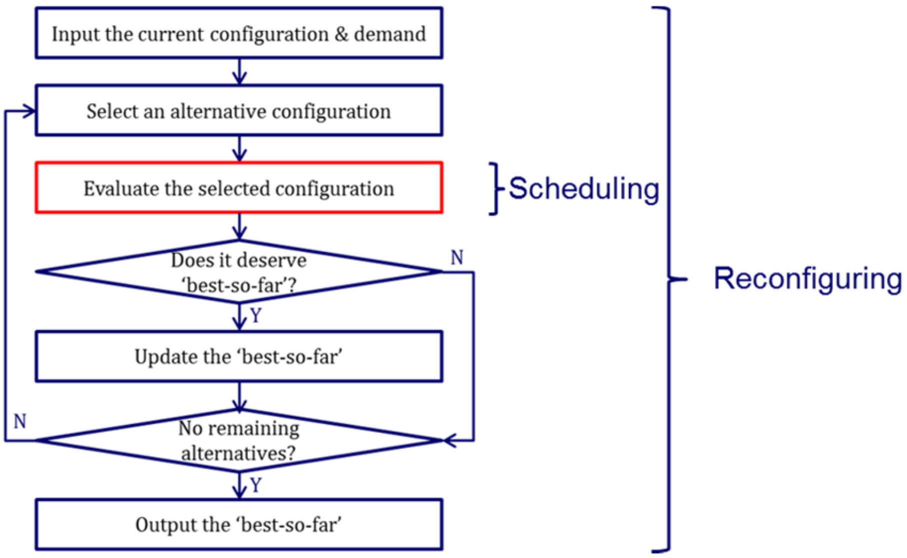
4. Scheduling Algorithm for RMC

This chapter is concerned with development of a scheduling algorithm to resolve the configuration evaluation problem (CEP). In CEP, manufacturing performance of the current system is evaluated. Therefore throughout this chapter, it is assumed the hardware configuration is given. Development of a scheduling algorithm appropriate for RMCs would provide managers with scheduling-level evaluation of manufacturing performance for decision making.

In RMC environment, the hardware configuration may be changing frequently. Therefore the scheduling algorithm for RMCs should be able to perform consistently for a variety of hardware configurations.

On the other hand, the schedules need to be generated as many as the number of RMCs in the manufacturing system, so it is required for the scheduling algorithm to generate a number of schedules in short time.

These requirements for the scheduling algorithm – performance consistency and time efficiency - are even more imperative when considering the reconfiguring problem, where a large number of configurations may necessitate scheduling for evaluation. The relationship between the scheduling problem and the reconfiguring problem is depicted in <Figure 4.1>.



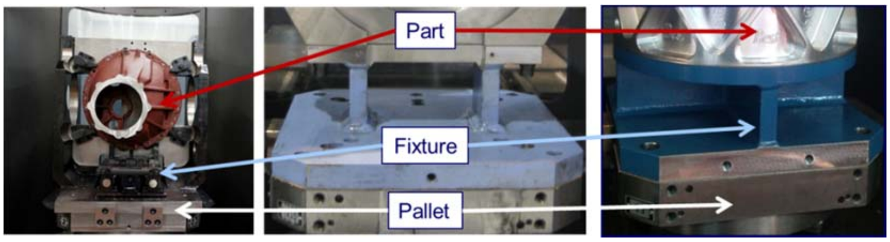
<Figure 4.1> The relationship between the scheduling problem and the reconfiguring problem

In this back ground, in this chapter, we are going to develop a scheduling algorithm appropriate for RMC environment. In the next section scheduling environment is described, and then the mathematical model is introduced. After that a heuristic algorithm based on lower bound prediction is proposed and an experiment is conducted to verify the performance of the proposed algorithm.

4.1. Scheduling Environment

Based on the production environment described in chapter 3, now let's see how operations are performed in an RMC.

An RMC processes production orders for a demand period (e.g. monthly demand). Each order indicates manufacturing parts of a certain type. Each part is produced through two consequent operations: loading (or setup) and machining. Loading operations are performed in L/U stations and machining operations in NC machines. A loading operation includes removing of the completed part and mounting of a new part on a pallet. After loading, parts may be processed in one of available machines directly or wait in a buffer. Processing times of loading and machining operations may differ from part types, and all L/U stations and machines are assumed identical.
For any operation, an empty pallet is required. Every pallet is the same, but distinguished by the fixture installed on it. And the installed fixture determines the supporting part type of the pallet. The 'pallet-fixture assignment' specifies which fixtures are to be installed on pallets. <Figure 4.2> indicates the example of the pallet-fixture assignment. The pallet-fixture assignment is given before production begins.



<Figure 4.2> An example of the pallet-fixture assignment

After machining, processed parts are extracted from the cell through L/U stations and returns pallets for processing of next parts.
It is assumed that cutting tools are not scarce resource and there exist enough capacity of the tool magazine. Therefore the part-loading problem or part-mix allocation problem [38] is not considered here. Besides, deterministic operation time is assumed and material handling time and the size of buffers are not considered.
Managerial objective is the production completion time or the makespan. It is known that the makespan objective guarantees high resource utilization [64]. Based on the makespan, managers may estimate the production time of an RMC with the current configuration.

Liu and MacCarthy [44] presented a classification scheme for FMS scheduling problems according to the FMS type, capacity constraints, job description, production environment, and scheduling criteria. According to it, the scheduling environment that this study is concerned is 'FMC/lim:LU,M,PL/JC1/pr,+pt/C_{max}' which means the FMS type is the FMC which a group of a single flexible machine(SFM)s sharing one common material handling device, with limited (denoted by ‘lim’) number of L/U stations, machines and pallets, each job consists of just one operation (‘JC1’), and orders are handled periodically (‘pr’), more than one part per type (‘pt’) for makespan objective.

It should be noted that RMC described above can be modelled as a two-stage pallet-constrained flexible flowshop (PcFF): The first stage is loading (setup) and the second stage is machining. As a flowshop, each part flows in the same sequence along the stages. However, pallets do not flow but circulate in the PcFF: After machining of one part, the associated pallet can be used for loading of another part. The two-stage pallet-constrained flexible flowshop is dealt in [73], where the shop with a single processor for each stage, and with commonly shared pallets regardless of part types or fixtures, was investigated. In our research, multiple processors for each stage and the distinguished pallets according to pallet-fixture assignment are considered.

4.2. Mathematical Model

Based on the problem description, following mathematical model is built:

PcFF

· Input parameters

i : Index of part types

l_i: Part i's loading (or setup) time

m_i: Part i's machining time

q_i: Production quantity (or demand) for part i

L : The number of L/U stations

M: The number of NC machines

p_i : The number of fixtured pallets which support part type i

T: The upper bound of the maximum completion time

t: Time index (t=1,2,…, T)

W : The large number

Both the loading and machining times (l_i and m_i) are assumed to be of integer multiples. The upper bound (T) of the makespan would be obtained by applying the simple dispatching rule such as SPT.

· Decision variables

$S^{\mathrm{l}}_{it}, S^{\mathrm{m}}_{it}$: The number of parts of type i which start its loading or machining operations at time t

· Dependent variables

$C_{it}^{\mathrm{l}}, C_{it}^{\mathrm{m}}$: The number of parts of type i which complete its loading or machining operations at time t

R_t^{l}, R_t^{m} : The number of L/U stations or NC machines which process parts at time t

A_{it}: The number of pallets for type i which are occupied by parts for processing at time t

x_{it}: Binary variable indicating whether there is any part of type i which completes its machining operation at time t (1: exist, 0: not exist).

$\mathrm{C}_{\max}$: The maximum completion time or makespan

The expected range of the makespan ($\mathrm{C}_{\max}$) is from few days to few weeks.

· Constraints

$\sum_t S_{it}^{\mathrm{l}} = \sum_t S_{it}^{\mathrm{m}} = q_t\ \forall i$ (1)

$S_{it}^{\mathrm{l}} = 0\ \forall i, t \in \{T - l_i + 2, T - l_i + 3, \dots, T\}$ (2)

$S_{it}^{\mathrm{m}} = 0\ \forall i, t \in \{T - m_i + 2, T - m_i + 3, \dots, T\}$ (3)

$S_{it}^{\mathrm{l}} = C_{i(t+l_i-1)}^{\mathrm{l}}\ \forall i, t \in \{1,2, \dots, T - l_i + 1\}$ (4)

$S_{it}^{\mathrm{m}} = C_{i(t+m_i-1)}^{\mathrm{m}}\ \forall i, t \in \{1,2, \dots, T - m_i + 1\}$ (5)

$\sum_k^t C_{ik}^{\mathrm{l}} \geq \sum_k^{(t+1)} S_{ik}^{\mathrm{m}}\ \forall i, t\ \in \{1,2, \dots, T - 1\}$ (6)

$R_t^{\mathrm{l}} = R_{t-1}^{\mathrm{l}} - \sum_i C_{i(t-1)}^{\mathrm{l}} + \sum_i S_{it}^{\mathrm{l}} \quad \forall\ t\ \in \{2, \dots, T\}$ (7)

$R_t^{\mathrm{m}} = R_{t-1}^{\mathrm{m}} - \sum_i C_{i(t-1)}^{\mathrm{m}} + \sum_i S_{it}^{\mathrm{m}} \quad \forall\ t\ \in \{2, \dots, T\}$ (8)

$R_1^{\mathrm{l}} = \sum_i S_{i1}^{\mathrm{l}}$ (9)

$R_1^{\mathrm{m}} = 0$ (10)

$A_{it} = A_{i(t-1)} - C_{i(t-1)}^{\mathrm{m}} + S_{it}^{\mathrm{l}} \quad \forall\ i, t\ \in \{2, \dots, T\}$ (11)

$A_{i1} = S_{i1}^{\mathrm{l}} \quad \forall i$ (12)

$R_t^{\mathrm{l}} \leq L, R_t^{\mathrm{m}} \leq M\ \forall\ t$ (13)

$A_{it} \leq p_i\ \forall i, t$ (14)

$W x_{it} \geq C_{it}^{\mathrm{m}}\ \forall i, t$ (15)

$\mathrm{C}_{\max} = \max_{i,t} \{x_{it} * t\}$ (16)

$S_{it}^{\mathrm{l}}, S_{it}^{\mathrm{m}} \in \mathrm{Z}^+, x_{it} \in \{0,1\}\ \forall\ i, t$ (17)

· Objective function

$z^{\mathbf{PcFF}} = \min \mathrm{C}_{\max}$ (18)

(1) - (3) means every part must start its loading and machining operations and operations should start so that it may finish within the upper time limit. (4), (5)

indicates relation between start and completion times of operations, and (6) forces the machining operation to start after the loading operation. (7) - (12) shows occupation of resources—both equipment and pallets—along time and (13), (14) says it should be less than or equal to resource capacity. Through (15), (16) the maximum completion time can be calculated. (17) indicates the variable conditions. Finally (18) indicates the objective function is the maximum completion time.

The complexity of the problem is NP-complete, for it can be reduced to the two-stage hybrid flowshop scheduling problem, which is NP-complete [23], by allowing enough pallets for each part type. When the problem was formulated in the exact search engine (iLOG CPLEX), the small size problem of around 10 jobs with 5 part types were taken over one and half hour, to find an optimal solution. It is practically impossible to apply an analytic approach. Therefore a heuristic approach may be a practical alternative.

From the next section it will be discussed about the algorithm for resolution of the scheduling problem.

4.3. Development of Algorithm

Before describing the developed algorithm, it needs to understand facts around the problem. First, the makespan ($z^{*\mathbf{PcFF}}$) is always larger than or equal to the total work amount allocated to each resource type divided by that resource capacity (or the number of resources). It means that the work amounts of various resource types constitute the initial lower bounds of the problem. In the problem are three kinds of resources – L/U station, NC machines, and pallets, and each constitutes a lower bound. In detail, they are given like below:

a) The lower bound in L/U station bottleneck case

$$LB_1^0 = \frac{\sum l_i \times q_i}{\min(L,\ \sum q_i)} + \min_i m_i$$

The superscript 0 in variable name means it is the initial lower bound calculated before the schedule is generated (i.e. no decision is made).

b) The lower bound in NC machine bottleneck case

$$LB_2^0 = \min_i l_i \ + \frac{\sum m_i \times q_i}{\min(M,\ \sum q_i)}$$

The lower bound for pallet bottleneck case is a little bit different from the other cases. Because pallets are distinguished by their supporting part types, the lower bound should be calculated for each of part type. The lower bound of pallets for part type k is given like that:

$$lb^{0}_{\mathrm{PA}(k)} = (l_k + m_k) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

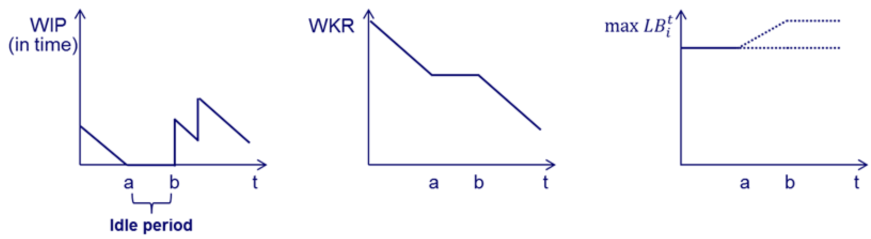
Finally the pallet-based lower bound is like below:

c) The lower bound in pallet bottleneck case

$$LB^{0}_{3} = \max_{\mathrm{k}} lb^{0}_{\mathrm{PA}(k)} = \max_{\mathrm{k}} (l_k + m_k) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

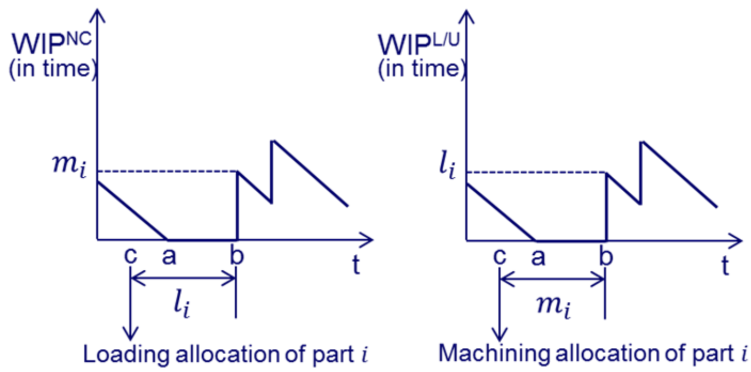
Definitely, $\mathrm{z}^{*\mathbf{PcFF}} \geq \max_{i} LB^{0}_{i}$. This information of lower bounds may be used as an estimator for $\mathrm{z}^{*\mathbf{PcFF}}$.

Each LB^{0}_{i} assumes no or minimum idleness of resources. But as a schedule is generated, resource idleness may occur. This is when there is no waiting job or work-in-process (WIP) in the queue, while resource is available. The resource idleness increases the corresponding LB^{t}_{i} and may increase the $\max LB^{t}_{i}$, according to whether the idle resource is bottleneck. (Superscript t in LB^{t}_{i} means the lower bound updated with regard to the schedule generated by time t.) <Figure 4.3> indicates impact of resource idleness on remaining work amount (WKR) and the updated lower bound. During the resource idle period, $a \leq t \leq b$ where WIP=0, remaining work amount is stagnant and it may lead to the increase of the lower bound.



<Figure 4.3> Impact of resource idleness

On the other hand, as the second fact, job allocation decisions affect the future status of the system, especially WIP. The impact of decisions on the future status is predictable via projection. For example, as in <Figure 4.4>, if a loading operation of part type i is assigned to a L/U station at $t = c$, then after the loading time (l_i), at $t = b = c + l_i$, the amount of WIP waiting for machining operations increases as much as the machining time (m_i). Conversely, if a machining operation of part type i is allocated to a NC machine, then after the machining time (m_i), the amount of WIP waiting for loading operations may increase, since the corresponding pallet is released for the next part, if exists. Through the WIP projection, the remaining work amount and the lower bound in the future time are also predictable.



<Figure 4.4> The work-in-process projection

Another implication of the second fact is that: It may not be enough to consider only permutation schedules. In two-stage (flexible) flowshop, it is known that there always exists an optimal schedule where the job sequence in both stages is identical [7]. In the pallet constraint flowshop, however, this property does not always stand.

Property 4.1 In pallet-constraint flowshops, where different pallets are required for different part types, permutation schedules may not constitute a dominant set.

Proof. See the counter example in the Appendix.

This property comes from the fact that the decisions in the second stage affect the future decisions in the first stage, since pallets circulate between stages.
Combining the two facts discussed above, it may be concluded that: During the schedule generation process, job allocation decisions affect WIP of

resources in the future time, resulting in the probable increase of the lower bound with respect to that time point. A good schedule may be obtained, if decisions are made so that the increment from the initial lower bound ($\max_i LB_i^0$) is as small as possible. This philosophy led us to the following algorithm.

Note that the following scheduling algorithm takes the dispatching approach [7] where a partial schedule is generated at every decision point (i.e. when a resource becomes available) in chronological order and generation of the entire schedule is completed when all operations are assigned according to production requirement.

Algorithm Whenever a resource becomes available, allocate an operation such that its allocation will result in the minimum projected lower bound in the future. Detailed steps are as follows:

STEP1 (Initialization)	Initialize the partial schedule (S) as an empty schedule. Set the projection time (τ) to 0. Go to step 2.
STEP2 (Dispatching)	Based on S, find the minimum time t when equipment (L/U station or NC machine) becomes available. 2.1. Find operations assignable to that equipment. Set those operations as J. 2.2. For each operation $j \in J$, calculate the projected lower bound ($\max_k LB_k^{\tau'}(j)$) for the future time $\tau' = \max\{\tau, \mathrm{t} + l_j(\text{or } m_j)\}$ 2.3. Assign an operation $p = \operatorname{argmin}_{j \in J}\{\max_k LB_k^{\tau'}(j)\}$ to available equipment. If tie happens, assign the operation with larger τ'. 2.4. Set $\tau = \max\{\tau, \tau'\}$ and go to step 3.
STEP3 (Termination)	Terminate if all operations are assigned. Output S as the final schedule. Otherwise go to step 2.

$LB_k^{\tau'}(j)$ is the projected lower bound of k^{th} type for time τ', if an operation $j \in J$ is assigned to available equipment at the decision point, based on decisions made so far. Detail information is given below:

$$LB_1^{\tau}(j) = \frac{\sum l_i \times q_i^l(j,\tau) + \sum r_k^l(j,\tau)}{\min\left(L,\ \sum q_i^l(j,\tau) + \sum u_k^l(j,\tau)\right)} + \min_{i:q_i^m(j,\tau)>0} m_i$$

$$LB_2^{\tau}(j) = \begin{cases} \min_k r_k^l(j,\tau) + \dfrac{\sum m_i \times q_i^m(j,\tau) + \sum r_k^m(j,\tau)}{\min\left(M,\ \sum q_i^m(j,\tau) + \sum u_k^m(j,\tau)\right)}, if \sum u_k^m(j,\tau) > 0 \\ \min_{i:q_i^l(j,\tau)>0} l_i + \dfrac{\sum m_i \times q_i^m(j,\tau) + \sum r_k^m(j,\tau)}{\min\left(M,\ \sum q_i^m(j,\tau) + \sum u_k^m(j,\tau)\right)}, otherwise \end{cases}$$

$$LB_3^{\tau}(j) = \max_k \left(l_k \times q_k^l(j,\tau) + \sum_{v:\mathrm{F}(v)=k} r_v^l(j,\tau) + m_i \times q_i^m(j,\tau) + \sum_{v:\mathrm{F}(v)=k} r_v^m(j,\tau) \right) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

Where

$u_k^l(j,\tau), u_k^m(j,\tau)$: Binary variable indicating whether there would exist an operation in processing at a L/U station or NC machine k at time τ, if part j is assigned at the current decision point (1: exist, 0: non-exists).

$r_k^l(j,\tau), r_k^m(j,\tau)$: Projected remaining operation time of a L/U station or NC machine k with respect to time τ, if an operation j is assigned at the current decision point.

$q_i^l(j,\tau), q_i^m(j,\tau)$: Projected remaining quantity of loading operations or machining operations with respect to time τ, if an operation j is assigned at the current decision point.

$\mathrm{F}(k)$: Part index of an operation which is being processed in a L/U station or NC machine k.

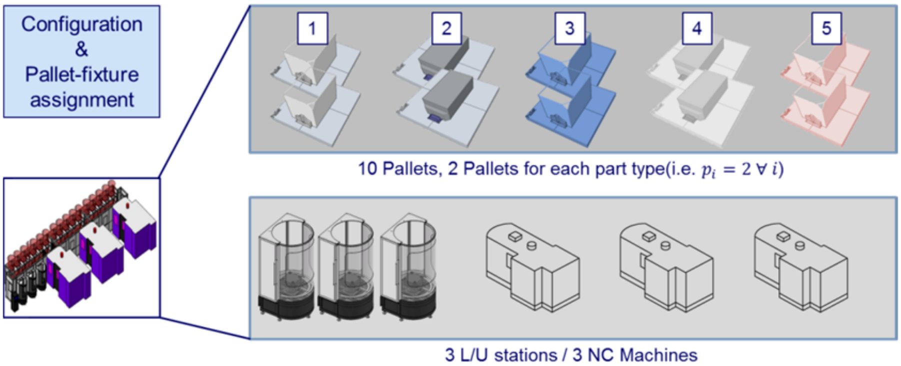
In calculation of $LB_k^{\tau'}(j)$, LB_k^0 is extended so that it would reflect the future system status which is derived by work-in-process projection, explained in <Figure 4.3> and <Figure 4.4>.

4.3.1. An Example for the Algorithm

For better understanding of the proposed algorithm, an example is presented, setting of which is depicted in <Figure 4.5>.

Master Data & Production quantity

Part Type (No.)	1	2	3	4	5
Loading Time (l_i)	42 min.	18	33	29	30
Machining Time(m_i)	60 min.	60	20	15	53
Production quantity(q_i)	4 units	3	1	3	3



<Figure 4.5> The example setting

For calculation of the initial lower bound, $LB_i{}^0$ is calculated by following:

$$LB_1^0 = \frac{\sum l_i \times q_i}{\min(L,\ \sum q_i)} + \min_i m_i \ = \frac{432}{\min(3,14)} + 15 = 159$$

$$LB_2^0 = \min_i l_i \ + \frac{\sum m_i \times q_i}{\min(M,\ \sum q_i)} = 18 + \frac{644}{\min(3,14)} = 232.67$$

$$lb_{\mathrm{PA}(1)}^0 = (l_1 + m_1) * \left\lceil \frac{q_1}{p_1} \right\rceil = (42 + 60) * \left\lceil \frac{4}{2} \right\rceil = 204$$

$$lb_{\mathrm{PA}(2)}^0 = (18 + 60) * \left\lceil \frac{3}{2} \right\rceil = 156$$

$$lb_{\mathrm{PA}(3)}^0 = (33 + 20) * \left\lceil \frac{1}{2} \right\rceil = 53$$

$$lb_{\mathrm{PA}(4)}^0 = (29 + 15) * \left\lceil \frac{3}{2} \right\rceil = 132$$

$$lb^0_{\mathrm{PA}(5)} = (30 + 53) * \left\lceil \frac{3}{2} \right\rceil = 166$$

$$LB^0_3 = \max_{\mathrm{k}} lb^0_{\mathrm{PA}(k)} = 204$$

So the initial lower bound of the problem is $\max_{\mathrm{i}} LB_i{}^0 = 232.67$.

For initialization, set $\mathrm{\tau} = 0$.

When $\mathrm{t} = 0$, all three L/U stations are available. For job allocation of the first L/U station, all types of parts are assignable, which means $J = \{1,2,3,4,5\}$.

For each assignable part $j \in J$, $\mathrm{\tau}'$ and the projected lower bound ($\max_{k} LB^{\tau'}_k(j)$) is like below:

For $j = 1$, $\mathrm{\tau}' = \max\{\mathrm{\tau},\ \mathrm{t} + l_1\} = \max\{0,0 + 42\} = 42$,

$$u^l_k(1,42), u^m_k(1,42), r^l_k(1,42), r^m_k(1,42) = 0\ \forall k$$

$$q^l_1(1,42) = 3$$

$$q^l_i(1,42), q^m_i(1,42) = q_i\ for\ i = 2,3,4,5$$

$$\max_k LB^{42}_k(1) = LB^{42}_2(1) = 263.5$$

For $j = 2$, $\mathrm{\tau}' = 18$, $max_k\, LB^{18}_k(2) = 240$

For $j = 3$, $\mathrm{\tau}' = 33$, $max_k\, LB^{33}_k(3) = 269$

For $j = 4$, $\mathrm{\tau}' = 29$, $max_k\, LB^{29}_k(4) = 248.33$

For $j = 5$, $\mathrm{\tau}' = 30$, $max_k\, LB^{30}_k(5) = 248$

Therefore, part 2 $= \operatorname{argmin}_{j \in J}\{\max_{k} LB^{\tau'}_k(j)\}$ is allocated to the L/U station. And $\mathrm{\tau}$ is updated $\max\{\mathrm{\tau}, \mathrm{\tau}'\} = \max\{0,42\} = 42$.

For decisions afterward, dispatching information is depicted in <Table 4.1>.

<Table 4.1> Dispatching information for the example of the proposed algorithm

t	Available resource	Assignable jobs J	Selected Job $p = \operatorname{argmin}_{j \in J}\{\max_k LB_k^{\tau'}(j)\}$	Projected lower bound $\max_k LB_k^{\tau'}(p)$	Projection time τ
0	L/U 2	1,2,3,4,5	2	236.3	18
0	L/U 3	1,3,4,5	4	236.3	29
18	L/U 1	1,3,4,5	1	243	60
18	L/U 1	1,3,4,5	1	241.67	60
18	MC 1	2	2	244	78
18	MC 2	2	2	244	78
29	L/U 3	3,4,5	5	241.33	78
29	MC 3	4	4	241.33	78
59	L/U 3	3,4,5	5	266.33	89
59	MC 3	5	5	274	112
60	L/U 1	3,4	4	274	112
60	L/U 2	3,4	3	274	112
78	MC 1	1	1	270	138
78	MC 2	1	1	242.67	138
89	L/U 1	2,4	2	252.33	138
89	L/U 3	4	4	241.33	138
112	L/U 1	5	5	264.33	142
112	MC 3	2,3,4,5	4	244	142
127	MC 3	2,3,4,5	4	256.33	142
138	L/U 2	1	1	261	180

138	L/U 3	1	1	241.33	180
138	MC 1	2,3,5	3	241.33	180
138	MC 2	2,5	5	251	191
142	MC 3	2,5	5	255	195
158	MC 1	2	2	278	218
191	MC 2	1	1	284	251
195	MC 3	1	1	259	255

The algorithm ends up with completion time 255 (min.), about 10% larger than the initial lower bound.

4.4. Experiment

To verify the performance of the proposed algorithm, an experiment was conducted.

Currently, hardware of a single RMC in development in the Korean NC machine maker supports up to 4 L/U stations and 9 NC machines. The number of supporting NC machines is larger than that of L/U stations, because usually machining operations take longer than loading operations. And this is reflected in the experiment.

Five experimental factors were chosen to experiment various shop configurations and demands: the number of L/U stations, machines and pallets, and the number of part types and production volume for each part type. Parameter information about processing times (l_i and m_i) were obtained from field data so that $l_i \sim \mathrm{UNIF}(10{,}50)$min. and $m_i \sim \mathrm{UNIF}(10{,}100)$min., respectively. As for the pallet-fixture assignment, the same number of total pallets was equally assigned for each part type and the remainder was assigned randomly.

As alternatives to be prepared with the proposed algorithm, various dispatching rules and a heuristic search engine, iLOG CP, were taken into account. iLOG CP is a constraint programming-based search engine that is known more appropriate for problems with disjunctive constraints such as scheduling problems, than exact algorithm based search engines, for example, iLOG CPLEX [28]. (It is also observed in the pilot study that iLOG CP showed superior performance than iLOG CPLEX.) Description of alternative dispatching rules is depicted in <Table 4.2>. Additionally, the failure limit of iLOG CP was set to ten thousands.

Performance was measured in the percentage gap from the initial lower bound ($\max LB_i^0$) given like below:

$$\%\text{Gap} = \frac{C_{max}(r) - \max_i LB_i^0}{\max_i LB_i^0} \times 100$$

where $C_{max}(r)$ is the makespan obtained from applying a dispatching rule or an algorithm r.

Finally for each experimental condition, the average performance from 10 replications was recorded, together with the computational time spent for generating the schedules.

<Table 4.2> Dispatching rules used in experiment as alternatives

Rule	**Description**
SPT	Shortest processing time job first
LPT	Longest processing time job first
MRK	Most work remaining job first
LRK	Least work remaining job first
LVP	Least vacant pallets job first

4.5. Result and Analysis

<Table 4.3> and <Table 4.4> show the result of the experiment for the performance measure and computational time, respectively. Figures from <Figure 4.6> to <Figure 4.8> are graphical representation. A*(non-p.) indicates the proposed algorithm which is applied to determining both loading and machining sequence, allowing generation of non-permutation schedules. A*(perm.) means the algorithm is applied to sequencing only loading operations, forcing the same sequence of loading and machining operations (i.e. permutation schedules).

The result shows that the proposed non-permutation algorithm (A*(non-p.)) consistently performs well over the alternative dispatching rules and iLOG CP for various shop configurations and demand settings. The performance of the least work remaining rule (LRK) and iLOG CP are relatively good, but the statistics — average, maximum and minimum performance value, and standard deviation— says that the proposed algorithm is superior to those alternatives as well.

On the other hand, the proposed permutation algorithm (A*(perm.)) appeared not to performs well, proving the nature of pallet-constrained flexible flowshops in which permutation schedules are not enough to find efficient solutions.

For computational time, as shown in <Table 4.4> and <Figure 4.8>, the proposed algorithm appeared to generate schedules as fast as the alternative dispatching rules, while the heuristic search engine iLOG CP spent about 10 to 100 times more to do the same job.

This result implies that the proposed algorithm performs consistent for various hardware configurations or demands and may generate a number of schedules quickly, therefore appropriate for hardware reconfiguration where various configurations need to be evaluated to find the best one. Moreover, the proposed algorithm may be applicable to manufacturing execution as well, where real-time decision making is required.

<Table 4.3> The experimental result for the performance measure (%Gap)

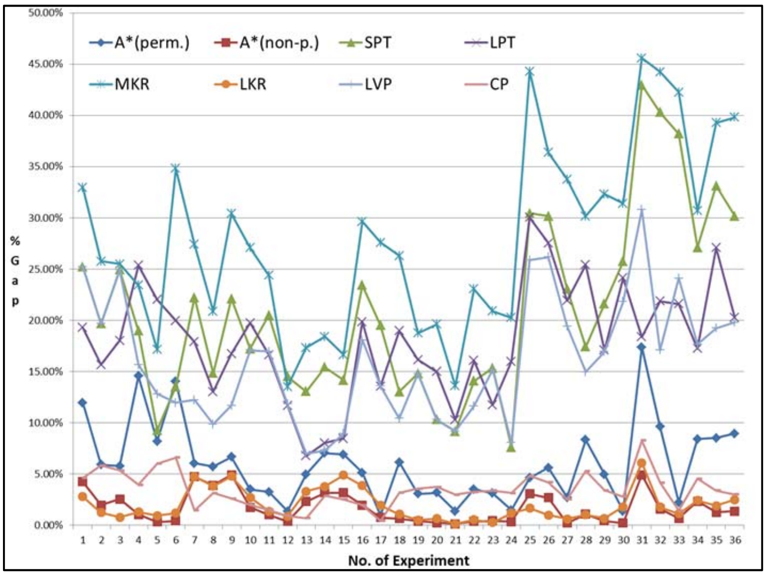
No.	L/U	NC	Pallet	Part	Vol.	A* (perm.)	A* (non-p.)	SPT	LPT	MKR	LKR	LVP	CP
1	2	3	10	10	10-30	11.94	4.23	25.24	19.29	32.96	2.80	25.24	4.63
2	2	3	10	10	30-60	5.90	1.96	19.67	15.67	25.77	1.22	19.67	5.82
3	2	3	10	10	60-90	5.78	2.56	24.92	18.02	25.49	0.79	24.92	5.35
4	2	3	20	10	10-30	14.55	1.05	18.96	25.37	23.42	1.30	15.66	3.95
5	2	3	20	10	30-60	8.20	0.33	9.27	22.01	17.15	0.93	12.81	6.01
6	2	3	20	10	60-90	14.05	0.43	13.51	19.96	34.82	1.18	11.95	6.62
7	3	6	15	10	10-30	6.05	4.72	22.22	17.88	27.41	4.76	12.22	1.44
8	3	6	15	10	30-60	5.71	3.89	14.82	13.02	20.86	3.82	9.87	3.15
9	3	6	15	10	60-90	6.65	4.91	22.12	16.75	30.40	4.79	11.68	2.60
10	3	6	30	10	10-30	3.50	1.75	17.22	19.75	27.09	2.72	17.07	1.99
11	3	6	30	10	30-60	3.26	1.02	20.49	16.63	24.40	1.37	16.94	1.50
12	3	6	30	10	60-90	1.36	0.38	14.50	11.67	13.50	0.89	11.83	0.92
13	4	9	20	10	10-30	4.96	2.30	13.05	6.77	17.32	3.30	6.97	0.68
14	4	9	20	10	30-60	7.05	3.18	15.41	8.03	18.39	3.78	7.32	2.87
15	4	9	20	10	60-90	6.90	3.16	14.16	8.47	16.63	4.88	8.94	2.52
16	4	9	40	10	10-30	5.14	1.97	23.43	19.85	29.63	3.87	18.09	1.97
17	4	9	40	10	30-60	0.93	0.75	19.52	13.56	27.60	1.95	13.59	0.42
18	4	9	40	10	60-90	6.16	0.65	13.04	18.95	26.30	1.06	10.42	3.16
19	2	3	20	20	10-30	3.07	0.39	14.80	16.13	18.76	0.51	14.80	3.58
20	2	3	20	20	30-60	3.17	0.25	10.32	14.99	19.60	0.65	10.32	3.73
21	2	3	20	20	60-90	1.37	0.12	9.15	10.30	13.58	0.14	9.15	2.99
22	2	3	30	20	10-30	3.53	0.38	14.07	16.05	23.04	0.55	11.63	3.25
23	2	3	30	20	30-60	3.12	0.42	15.34	11.73	20.91	0.25	15.17	3.39
24	2	3	30	20	60-90	1.48	0.34	7.60	15.97	20.24	1.19	8.07	3.15
25	3	6	30	20	10-30	4.62	3.06	30.45	30.02	44.29	1.67	25.88	4.81
26	3	6	30	20	30-60	5.62	2.69	30.18	27.55	36.37	0.98	26.15	4.20

27	3	6	30	20	60-90	2.72	0.20	23.08	21.91	33.78	0.61	19.43	2.53
28	3	6	40	20	10-30	8.33	1.06	17.41	25.43	30.18	1.02	14.97	5.30
29	3	6	40	20	30-60	4.96	0.42	21.59	17.09	32.32	0.67	16.94	3.41
30	3	6	40	20	60-90	1.36	0.20	25.74	24.12	31.39	1.80	21.83	2.80
31	4	9	40	20	10-30	17.37	4.92	42.95	18.36	45.59	6.10	30.78	8.27
32	4	9	40	20	30-60	9.61	1.57	40.29	21.86	44.26	1.77	17.09	4.16
33	4	9	40	20	60-90	2.23	0.68	38.21	21.59	42.25	1.13	24.09	1.41
34	4	9	50	20	10-30	8.40	2.33	27.10	17.26	30.68	2.43	17.65	4.53
35	4	9	50	20	30-60	8.52	1.27	33.11	27.09	39.27	1.90	19.24	3.39
36	4	9	50	20	60-90	8.94	1.38	30.16	20.24	39.80	2.49	19.78	3.00
Average (%)						6.01	**1.69**	20.92	18.04	27.93	1.98	16.06	3.43
Max value (%)						17.37	**4.92**	42.95	30.02	45.59	6.10	30.78	8.27
Min value (%)						0.93	**0.12**	7.60	6.77	13.50	0.14	6.97	0.42
Standard deviation (%)						3.91	**1.49**	8.83	5.54	8.94	1.52	6.02	1.70

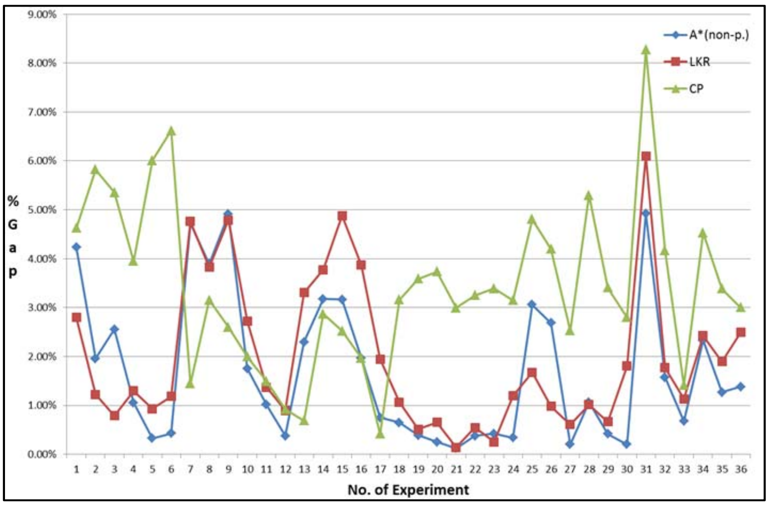
<Table 4.4> The experimental result for computational time (millisecond)

No	LU	NC	PA	Part	Vol.	A* (perm.)	A* (non-p.)	SPT	LPT	MKR	LKR	LVP	CP
1	2	3	10	10	10-30	139	172	106	185	141	145	103	6911
2	2	3	10	10	30-60	476	523	357	626	519	534	349	30225
3	2	3	10	10	60-90	2440	2850	1852	3616	2225	2795	1858	146109
4	2	3	20	10	10-30	201	207	129	225	183	145	116	6328
5	2	3	20	10	30-60	1018	842	479	1371	733	851	496	27598
6	2	3	20	10	60-90	2278	1933	1088	3286	1946	2176	1021	154064
7	3	6	15	10	10-30	123	121	110	122	128	93	96	5701
8	3	6	15	10	30-60	638	605	590	621	676	491	472	24228
9	3	6	15	10	60-90	1083	903	1145	1278	1486	822	796	147648
10	3	6	30	10	10-30	166	242	58	108	95	67	67	4054
11	3	6	30	10	30-60	552	713	267	407	387	184	294	17183
12	3	6	30	10	60-90	1428	1420	744	1048	835	802	659	108482
13	4	9	20	10	10-30	206	202	190	198	218	149	141	5309
14	4	9	20	10	30-60	960	930	871	917	1050	741	711	41292
15	4	9	20	10	60-90	2152	1725	2197	1681	2141	1532	1648	77113
16	4	9	40	10	10-30	303	379	75	115	118	50	69	4006
17	4	9	40	10	30-60	828	942	256	351	383	242	280	11935
18	4	9	40	10	60-90	1841	1998	859	1622	1408	944	802	49697
19	2	3	20	20	10-30	748	1340	357	1029	645	879	373	21996
20	2	3	20	20	30-60	3411	5260	1629	5618	2598	4335	1653	184726
21	2	3	20	20	60-90	11582	15444	5702	16957	8832	13885	5729	703562
22	2	3	30	20	10-30	772	1319	389	1063	651	780	369	18327
23	2	3	30	20	30-60	4594	6863	2776	6840	4475	5513	2705	144962
24	2	3	30	20	60-90	9228	13520	4563	13752	8075	12302	4522	652904
25	3	6	30	20	10-30	568	770	233	324	392	188	214	9813
26	3	6	30	20	30-60	1732	2227	1144	2005	1720	1160	997	122091

27	3	6	30	20	60-90	4654	5583	3089	4653	4750	3768	2774	471925
28	3	6	40	20	10-30	1004	1257	285	652	500	379	307	12625
29	3	6	40	20	30-60	3337	3483	1354	2545	2049	1877	1460	78150
30	3	6	40	20	60-90	5084	5647	2913	7253	4939	3654	3545	475142
31	4	9	40	20	10-30	998	1353	413	476	534	336	356	10144
32	4	9	40	20	30-60	2637	3218	1539	1778	1824	978	1009	70056
33	4	9	40	20	60-90	5634	5766	3643	4808	4528	2786	2798	255274
34	4	9	50	20	10-30	1387	1609	298	403	364	231	268	10073
35	4	9	50	20	30-60	3662	4274	1345	2228	1845	1066	1230	64893
36	4	9	50	20	60-90	7185	7447	3157	4994	4108	3350	3305	337145

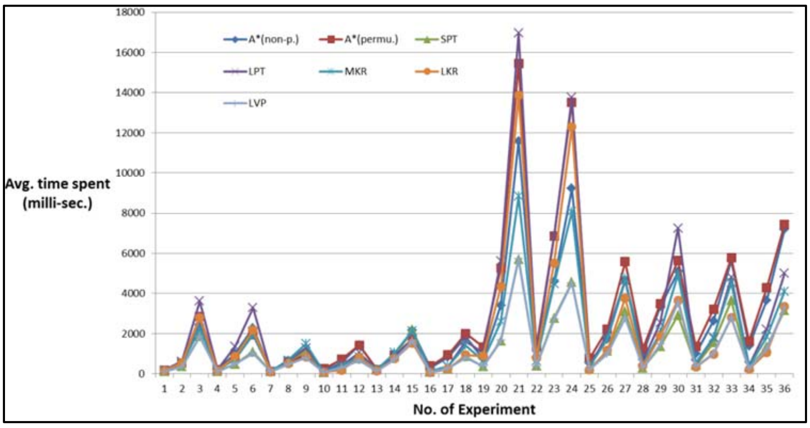


<Figure 4.6> The experimental result for the performance measure (%Gap)



<Figure 4.7> The experimental result for the performance measure

(%Gap, selected and zoomed)



<Figure 4.8> The experimental result for computational time (excluding iLOG CP)

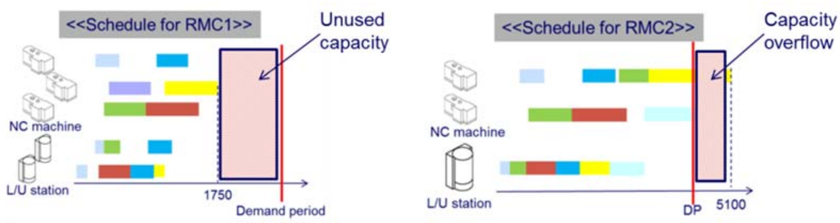
The robustness of the proposed algorithm may be originated from the fact that the algorithm reflects shop configurations and demand information in its reasoning scheme (i.e. calculation of LB_i^τ) and also looks ahead of the future status of the shop via work-in-process projection, unlike the conventional dispatching rules.

In the next chapter, an integrative framework for the reconfiguring problem using the developed scheduling algorithm is discussed.

5. A Framework for Hardware Reconfiguration

The existing manufacturing system may fail to respond to demand. In this case managers may consider: rejection of some orders [40], negotiation [30], or out-sourcing [10]. In RMC environment hardware reconfiguration may be considered as well. It is especially when there is unused capacity in some parts while capacity overflow occurs in other parts, as depicted in <Figure 5.1>.

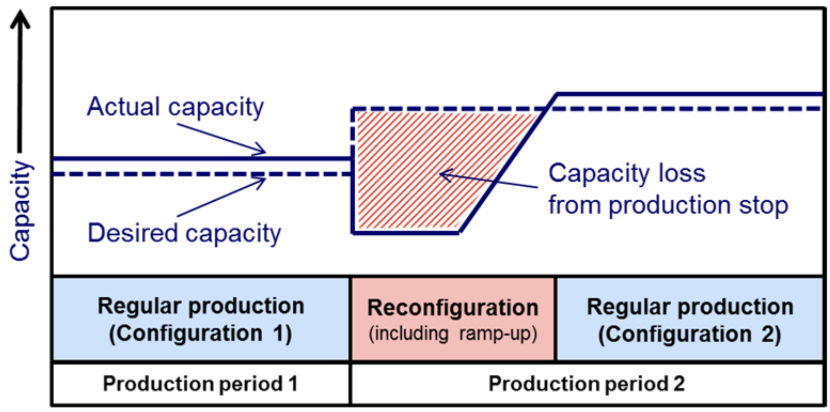
In this case, managers may have questions like this: “How does the current system need to be changed to satisfy demand or to reduce production delay?” The **Reconfiguring problem** is to decide the best hardware configuration to improve manufacturing performance for given demand. Resolution may not be easy, for the number of possible alternatives is huge. Therefore a systematic framework for hardware reconfiguration needs to be developed.



<Figure 5.1> The case when hardware reconfiguration is considered

On the other hand, it takes time for hardware reconfiguration. Production stops during reconfiguration. Reconfiguring time directly affects manufacturing performance. Capacity loss from the production stop during hardware reconfiguration is depicted in <Figure 5.2>. (Note that here capacity means that of a single RMC, since total capacity of the factory does not change.) If such capacity loss is too much, it may not be desirable to change the current configuration. Therefore, impact of reconfiguring time should be considered for accurate evaluation of manufacturing performance.

Monetary cost also occurs for reconfiguration such as labor cost, but this study does not consider monetary cost explicitly. Since monetary cost tends proportional to reconfiguring time [80], monetary cost may be considered implicitly by considering time required for reconfiguration.



<Figure 5.2> The capacity change around hardware reconfiguration

(Edited from Spicer & Carlo [80])

In this background, we are going to develop an integrative framework for the reconfiguring problem. A relaxation based branch-and-bound algorithm is suggested for configuration search. The scheduling algorithm suggested in the previous chapter will be used for configuration evaluation as a sub-component.

5.1. Mathematical Model

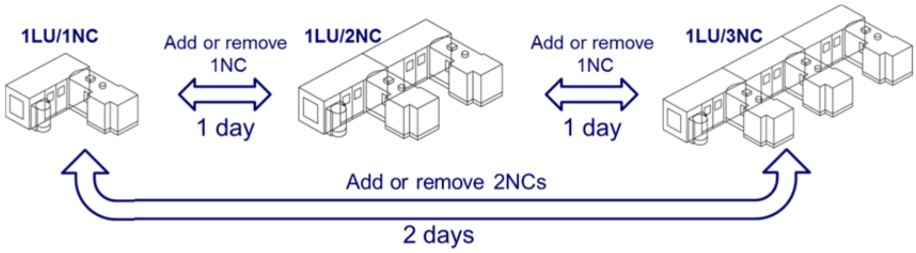
Description about production environment follows the one explained in chapter 3.

As described in <Figure 3.1>, there exist multiple production lines in the factory, according to product groups. And there exist multiple RMCs, each being defined with the number of L/U stations, NC machines, and fixtured pallets according to part types. Each RMC is given a production order for a pre-defined demand period. A production order consists of part types and volumes.

Reconfigurable components are L/U stations, NC machines, and pallets. Those components are interchangeable among RMCs via reconfiguration. Pallets are also deliverable from one RMC to another.

Reconfiguration includes removing or attaching of equipment, tool changes, NC program uploads, precision tests and adjustment, and all the other activities required to ramp up the manufacturing system with the new hardware configuration.

Change from one hardware configuration to another takes time. Production interruption of an RMC from hardware reconfiguration and ramp-up is independent each other: The duration of production interruption is proportional to the degree of change from the initial hardware configuration, as depicted in <Figure 5.3>.



<Figure 5.3> Reconfiguring time according to change of cell configurations

Time required for delivering pallets to another cells and changing the pallet-fixture assignment is relatively small compared to time for exchanging L/U stations and NC machines, and it may be performed in parallel. Therefore it is assumed that the time required for pallet adjustment is included in reconfiguring time of other equipment.

Managers' primary objective is demand fulfillment. Then managers are concerned with reconfiguring cost, as secondary objectives. In the case where it may not be possible to complete production within the available time, managers seek to minimize total delay – difference from the available time and the estimated completion time.

Description of operation environment of RMCs is following that described in chapter 3.

Based on the description of the previous section, the following mathematical model for the reconfiguring problem (RP) is built:

RP

· Input parameters (New or redefined variables only)

c : Index of the cells

q_{ci}: The production quantity (or demand) for part i of cell c

w_c : The relative weight of cell c

L_c^0 : The initial number of L/U stations of cell c

M_c^0: The initial number of NC machines of cell c

α : The unit time required for reconfiguration of L/U station

β : The unit time required for reconfiguration of NC machine

H: The demand period or the available time (e.g. few weeks or a month)

The demand period (H) should be adjusted so that it has the same unit with the time index.

· Decision variables

L_c : The number of L/U stations of cell c after reconfiguring

M_c: The number of NC machines of cell c after reconfiguring

· Dependent variables

$\underline{p_{ci}}$: The minimal number of pallets which support part type i in cell c after reconfiguring

$\mathrm{C}_{\max}^{c}$: The required production time of the cell c

$z^{*\mathbf{PcFF}}(L_c, M_c, p_{ci}, q_{ci})$: The optimal makespan of the pallet-constrained flexible flowshop with a configuration consisting of $L = L_c, M = M_c, p_i = p_{ci}\ \forall i$ and demand $q_i = q_{ci}\ \forall i$

D_c: Reconfiguring time of the cell c

y_c: The binary variable indicating demand fulfillment of the cell c (1: fulfilled, 0: not fulfilled)

· Constraints

$$\sum_c L_c = \sum_c L_c^0, \sum_c M_c = \sum_c M_c^0 \quad (1)$$

$$\mathrm{C}_{\max}^c = z^{*\mathbf{PcFF}}(L_c, M_c, \underline{p_{ci}}, q_{ci}) \forall c \quad (2)$$

$$D_c = \alpha|L_c - L_c^0| + \beta|M_c - M_c^0| \forall c \quad (3)$$

$$W(1 - y_c) \geq \mathrm{C}_{\max}^c + D_c - H\ \forall c \quad (4)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}}} \right\rceil \leq \max\{LB_1^c, LB_2^c\}\ \ \forall c, i \quad (5)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}} - 1} \right\rceil > \max\{LB_1^c, LB_2^c\}\ \forall c, i \quad (6)$$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c, \sum q_{ci})} + \min_i m_i \quad (7)$$

$$LB_2^c = \min_i l_i\ \ + \frac{\sum m_i \times q_{ci}}{\min(M_c, \sum q_{ci})} \quad (8)$$

$$L_c, M_c, \underline{p_{ci}} \in \mathrm{Z}^+, y_c \in \{0,1\}\ \forall c, i \quad (9)$$

· Objective function

$$\mathrm{z}^{\mathbf{RP}} = \min \sum_c \{w_c * (1 - y_c) * (\mathrm{C}_{\max}^c + D_c - H)\} \quad (10)$$

(1) indicates the total number of equipment—L/U stations and NC machines—of the whole factory is fixed (i.e. purchasing of equipment is not

considered). (2) indicates the required production time of each cell after reconfiguring is calculated by scheduling and through (3), the corresponding reconfiguring time can be calculated. (4) is for checking on-time production, whether whole production of the cell is finished within the available time. (5) and (6) implies that pallets are determined minimal, not causing bottleneck after reconfiguration. (7) and (8) indicates corresponding lower bounds in the case of L/U station and machine bottleneck, respectively. Note that through **RP**, only the minimal number of pallets is determined. The actual number of pallets for each part type of each RMC is determined after solving **RP**, which is discussed in the later section. (9) indicates the integer or binary conditions of variables. Finally the objective function (10) indicates the sum of production delay of whole RMCs.

Note that it is practically impossible to obtain $z^{*\mathbf{PcFF}}(L_c, M_c, p_i, q_i)$ for a certain configuration, as it is a NP-hard problem. Instead of seeking $z^{*\mathbf{PcFF}}(L_c, M_c, p_i, q_i)$, we are going to use the makespan obtained by applying the scheduling algorithm suggested in the previous chapter, denoted by $z^{\mathbf{PcFF(A)}}(L_c, M_c, p_i, q_i)$. Therefore the constraint (2) in **RP** would be substituted like below:

$$\mathrm{C}_{\max}^{c} = z^{\mathbf{PcFF(A)}}(L_c, M_c, \underline{p_{ci}}, q_{ci})\forall c \quad (2')$$

Since the developed scheduling algorithm appeared to perform close to the lower bound and robust for various configurations, it would be the best alternative minimizing performance deterioration.

5.2. Solution Procedure for Hardware Reconfiguration: Branch-and-bound Approach

The most common way to solve integer programs is the branch-and-bound (B&B) [92]. It is often found in literature to employ B&B to solve the scheduling problems [7][8].

B&B seeks to reduce search space by calculating lower bounds of sub-problems. Tightness of the lower bounding scheme determines effectiveness of B&B.

A relaxation problem is often used to obtain the lower bound [45][92]. Linear relaxation is common but in the case of **RP**, it may not be applied because i) **RP** is not linear and ii) even if it is possible to obtain a real-value optimal solution, it may provide loose bounds.

Therefore it is required to design a relaxation problem being solved fast and providing tight lower bounds. In the next section it will be discussed about the relaxation problem.

5.2.1. Relaxation of the Problem

The relaxation problem of the original reconfiguring problem (**RPX**) is described below:

First, the required production time of each cell ($C^c_{\max}$) is redefined like below:

$$C^c_{\max} = \max\{ LB^c_1 + \alpha|L_c - L^0_c|, LB^c_2 + \beta|M_c - M^0_c|\}\ \forall c \quad (2x)$$

Second, in **RPX**, the (3) in **RP** is substituted like blow:

$D_c = 0 \ \forall c$ (3x)

Note that instead of setting $D_c = 0$, the required production time partially includes the reconfiguring time.

Proposition 5.1 RPX is a relaxation problem of **RP**.

Proof. The solution space of decision variables of **RPX** is the same with **RP** and the objective value of **RPX** is always smaller than **RP** for every feasible solution, since $z^{\mathbf{PcFF}(\mathrm{A})}(L_c, M_c, p_{ci}, q_{ci}) + D_c \geq \max\{LB_1^c + \alpha|L_c - L_c^0|, LB_2^c + \beta|M_c - M_c^0|\}$ ■

Note that the relaxed terms ($\mathrm{C}_{\max}^c$, D_c) are all related to the objective function. A relaxation problem is usually designed so that some of constraints in the original problem is removed or relaxed and the feasible region of the problem is extended. The designed relaxation problem (RPX), however, only seeks to simplify calculation of the objective function.

For **RPX** to be meaningful to solve **RP**, it should have properties such that it can be solved fast. In the next two sections such properties and how to solve **RPX** are investigated.

5.2.1.1. Dimension Reduction of the Relaxation Problem

If we are going to solve **RPX** via complete enumeration, the number of possible configurations that should be investigated may be too large.

The number of possible configurations is given like below:

$$_{\sum L_c - 1}\mathrm{C}_{\mathrm{N}-1} \times {}_{\sum M_c - 1}\mathrm{C}_{\mathrm{N}-1}$$

where N is the number of cells in the factory.

Considering that the expected range of the number of equipment and cells in a factory is $3 \le \mathrm{N} \le 10$, $5 \le \sum L_c \le 30$, and $10 \le \sum M_c \le 50$, above number is too large to enumerate completely. To reduce the problem space, we consider followings:

First, the ideal number of equipment is defined.

Definition:

'*The ideal number of equipment*' of a cell is defined the minimal number which satisfies the following conditions:

$$H \ge \frac{\sum l_i \times q_{ci}}{\min(L_c^I,\ \sum q_{ci})} + \min_i m_i + \alpha |L_c^I - L_c^0|$$

$$H \ge \min_i l_i + \frac{\sum m_i \times q_{ci}}{\min(M_c^I,\ \sum q_{ci})} + \beta |M_c^I - M_c^0|$$

L_c^I and M_c^I denote the ideal number of L/U stations, NC machines of cell c, respectively. With the ideal number of equipment, the lower bounds of the estimated completion time are always smaller than or equal to the available time. Then by definition following properties hold.

Property 5.1

For $a \in Z^{+}$,

$$\mathrm{C}_{\max}^{c}(L_{c}^{I}-a, M_{c}) \geq \mathrm{C}_{\max}^{c}(L_{c}^{I}, M_{c}) \leq H$$

And for $a \in Z^{+} \cup \{0\}$ and $a \leq L_{c}^{0}-L_{c}^{I}$,

$$\mathrm{C}_{\max}^{c}(L_{c}^{I}+a, M) = \mathrm{C}_{\max}^{c}(L_{c}^{I}, M) \leq H$$

Property 5.2

For $a \in Z^{+}$,

$$\mathrm{C}_{\max}^{c}(L_{c}, M_{c}^{I}-a) \geq \mathrm{C}_{\max}^{c}(L_{c}, M_{c}^{I})$$

And for $a \in Z^{+} \cup \{0\}$ and $a \leq M_{c}^{0}-M_{c}^{I}$,

$$\mathrm{C}_{\max}^{c}(L_{c}, M_{c}^{I}+a) = \mathrm{C}_{\max}^{c}(L_{c}, M_{c}^{I})$$

Property 5.1 and **Property 5.2** hold by definition and by the fact that the relaxed reconfiguring time ($\alpha|L_{c}^{I}-L_{c}^{0}|$ and $\beta|M_{c}^{I}-M_{c}^{0}|$) does not increase (rather maybe decrease) as the L_{c}, M_{c} approaches to the initial configuration. Using the properties, following proposition is derived.

Proposition 5.2 If $\sum(L_{c}^{0}-L_{c}^{I}) \geq 0$, $z^{\mathbf{RPX}}(L_{c}^{*}, M_{c}^{*})$ is the same with a sub-problem with decision variable $L_{c}=L_{c}^{I}+a_{c}$, $a_{c} \geq 0$ ∀c, and a_{c} also can be derived.

Proof. For $c: L_{c}^{*}<L_{c}^{I}$, setting $L_{c}^{\prime}=L_{c}^{I}$ would not increase the objective

value, and for $c: L_c^* > L_c^I$, setting $L_c' = L_c^I$ would also not increase the objective value by **Property 5.1**.

And suppose $a = \sum(L_c^0 - L_c^I) \geq 0$. Then it is possible to modify $L_c' = L_c^I + a_c$, $0 \leq a_c \leq L_c^0 - L_c^I$ for $c: L_c^0 > L_c^I$ so that $\sum a_c = a$**,** not increasing the objective value, by **Property 5.1.** Now L_c' is a feasible solution. Therefore the following holds:

$z^{\mathbf{RPX}}(L_c^*,\ M_c^*) = z^{\mathbf{RPX}}(L',\ M_c^*) = z^{\mathbf{RPX}}(L_c^I,\ M_c^*) = z^{\mathbf{RPX}}(L_c^I + a_c,\ M_c^*)$ ■

Proposition 5.2 is for L/U stations. A similar proposition holds for NC machines.

Proposition 5.3 If $\sum(M_c^0 - M_c^I) \geq 0$, $z^{\mathbf{RPX}}(L_c^*, M_c^*)$ is the same with a sub-problem with decision variable $M_c = M_c^I + a_c\, , a_c \geq 0\ \forall$c, and a_c also can be derived.

Proof. The same with **Proposition** 5.2 ■

Combining **Proposition 5.2** and **Proposition 5.3**, the following proposition is derived:

Proposition 5.4 If $\sum(L_c^0 - L_c^I) \geq 0$, $\sum(M_c^0 - M_c^I) \geq\ 0$, and then $z^{\mathbf{RPX}}(L_c^*, M_c^*) = z^{\mathbf{RPX}}(L_c^I, M_c^I) = 0$.

It is probable that only a single kind of resource is in shortage. In that case, the problem dimension may be reduced by half using **Proposition 5.2** and

Proposition 5.3. And if there is no resource type in shortage, and then the optimal objective value can directly be derived by **Proposition 5.4**. Note that the above propositions may be very useful while solving sub-problems in the search tree of B&B, even if the original problem does not satisfy any of the conditions.

5.2.1.2. Alternative Formulation to Reduce the Domain of Variables

The concept of the ideal number of equipment provides additional advantage. Consider following propositions.

Proposition 5.5 If $\sum(L_c^I - L_c^0) \geq 0$, there is an optimal solution with$L_c \leq L_c^I \forall \mathrm{c}$

Proof. Suppose an optimal solution L_c^*, $M_c^* \forall c$ and for some cell k, $L_k^* > L_k^I = L_k^* - a$.

Then by setting $L_k' = L_k^I$ and $L_c' = L_c^* \forall c/\{k\}$, an alternative solution L_c', M_c^* can be obtained without increase of the objective value, by **Property 5.1**. And it is possible to modify $L_c' = L_c^I + a_c$, $0 \leq a_c \leq L_c^0 - L_c^I$ for $c\colon L_c^0 > L_c^I$ so that $\sum a_c = a$**,** not increasing the objective value, by **Property 5.1.** And now L_c' is a feasible solution. ■

Proposition 5.5 is for L/U stations. A similar proposition holds for NC machines.

Proposition 5.6 If $\sum(M_c^I - M_c^0) \geq 0$, there is an optimal solution with $M_c \leq M_c^I \forall c$

By **Proposition 5.5** and **Proposition 5.6**, the domain of decision variables would be reduced. To reflect the reduced variable domain in the model, an alternative formulation of **RPX** is built, named **A-RPX.**

A-RPX

· Input parameters (New or redefined variables only)

L_c^I : The ideal number of L/U stations of cell c (based on the definition in the previous section)

M_c^I: The ideal number of NC machines of cell c

· Decision variables

L_c^- : The shortage of L/U stations of cell c from the ideal number (L_c^I) after reconfiguring

M_c^-: The shortage of NC machines of cell c from the ideal number (M_c^I) after reconfiguring

· Constraints

$$\sum_c (L_c^I - L_c^-) = \sum_c L_c^0 , \sum_c (M_c^I - M_c^-) = \sum_c M_c^0 \quad (1x\text{-}a)$$

$$\mathrm{C}_{\max}^c = \max\{ LB_1^c + \alpha |L_c^I - L_c^- - L_c^0|, LB_2^c + \beta |M_c^I - M_c^- - M_c^0| \} \;\forall c \quad (2x\text{-}a)$$

$$D_c = 0 \forall c \quad (3x)$$

$$W(1 - y_c) \geq \mathrm{C}_{\max}^c + D_c - H \;\forall c \quad (4)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}}} \right\rceil \leq \max\{ LB_1^c, LB_2^c \} \;\; \forall c, i \quad (5)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}} - 1} \right\rceil > \max\{ LB_1^c, LB_2^c \} \;\forall c, i \quad (6)$$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c^I - L_c^-, \sum q_{ci})} + \min_i m_i \quad (7x\text{-}a)$$

$$LB_2^c = \min_i l_i \;\; + \frac{\sum m_i \times q_{ci}}{\min(M_c^I - M_c^-, \sum q_{ci})} \quad (8x\text{-}a)$$

$$L_c^- < L_c^I, M_c^- < M_c^I, L_c^-, M_c^- \in \mathrm{Z}^+ \cup \{0\}, \underline{p_{ci}} \in \mathrm{Z}^+ , y_c \in \{0,1\} \;\forall c, i \quad (9x\text{-}a)$$

· Objective function

$$\mathrm{z}^{\mathbf{A-RPX}} = \min \sum_c \{ w_c * (1 - y_c) * (\mathrm{C}_{\max}^c - H) \} \quad (10)$$

A-RPX is the same with **RPX** except that the original decision variables (L_c, M_c) are substituted with the shortage from the ideal number of equipment, $\sum_c (L_c^I - L_c^-)$, $\sum_c (M_c^I - M_c^-)$, and the domain reduction by **Proposition 5.5** and **Proposition 5.6** are apparently expressed in (9x-a).

Proposition 5.7 $\mathrm{z}^{*\mathbf{A-RPX}} = \mathrm{z}^{*\mathbf{RPX}}$

Proof. The solution space of **RPX** is definitely larger than that of **A-RPX**. Therefore, it is enough to show that every optimal solution in **RPX** is convertible to the solution in **A-RPX**, without the decrease of the objective value. Support the optimal solution in **RPX** is L_c^*, $M_c^* \forall c$. For any c, if $L_c^* - L_c^I = a \le 0$, it can be converted to $L_c^- = -a$, and $L_c^* = L_c^I - L_c^-$ holds. Otherwise if $L_c^* - L_c^I = a > 0$, then $L_c^* = L_c^I + a$ and the alternative solution L_c' in that L_c^* is substituted with $L_c' = L_c^I$ have still the same objective value by **Proposition 5.1**, then again $L_c' - L_c^I = a \le 0$ holds and converted to the solution in **A-RPX**. And the converted solution is a feasible with the same objective values, since the objective functions of the two problems are same, that is what was going to be shown. ■

Remember that in **RP** and **RPX**, the problem is to distribute the total number equipment among the cells. In **A-RPX**, however, the problem is to distribute the total shortage $\sum(L_c^I - L_c^0)$, $\sum(M_c^I - M_c^0)$ among the cells.

This alternative model has a different variable domain. From (1') the following is derived:

$$\sum L_c^- = \sum (L_c^I - L_c^0)$$

$$\sum M_c^- = \sum (M_c^I - M_c^0)$$

It is expected that by solving **A-RPX** instead of **RPX**, the domain of decision variables is reduced significantly. In **A-RPX**, the number of possible combination is given:

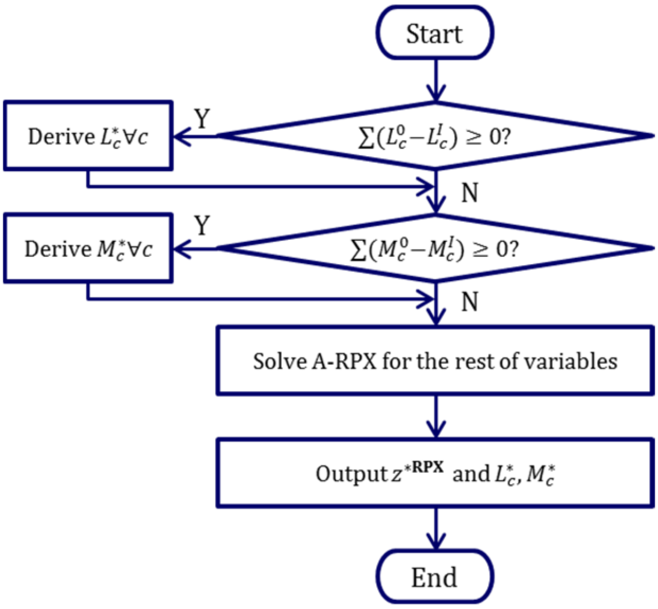
$$_{\sum(L_c^I - L_c^0)+N-1}\mathrm{C}_{\mathrm{N-1}} \times {}_{\sum(M_c^I - M_c^0)+N-1}\mathrm{C}_{\mathrm{N-1}}$$

Note that the above equation holds when the total shortage is greater than 0 ($\sum L_c^- = \sum(L_c^I - L_c^0) > 0$, $\sum M_c^- = \sum(M_c^I - M_c^0) > 0$). Otherwise, when the total number equipment exceeds the ideal number, it becomes 1 as discussed in the previous section.

<Figure 5.4> summarizes the solving process of **RPX** to obtain the lower bounds of **RP**.

So far we have discussed about the relaxation model and how it can be efficiently solved by reduction of the dimension and the domain of variables. The proposed relaxation model is expected to provide a tight lower bound to the sub-problem fast so that it may be possible to implicitly enumerate possible combinations using the branch-and-bound algorithm.
Note, however, that the proposed relaxation does not guarantee it will be solved fast almost every time as does the linear relaxation [92]. The complexity may not be reduced compared to the original problem for certain conditions of problems. The efficacy of the proposed relaxation should be verified through an experiment.

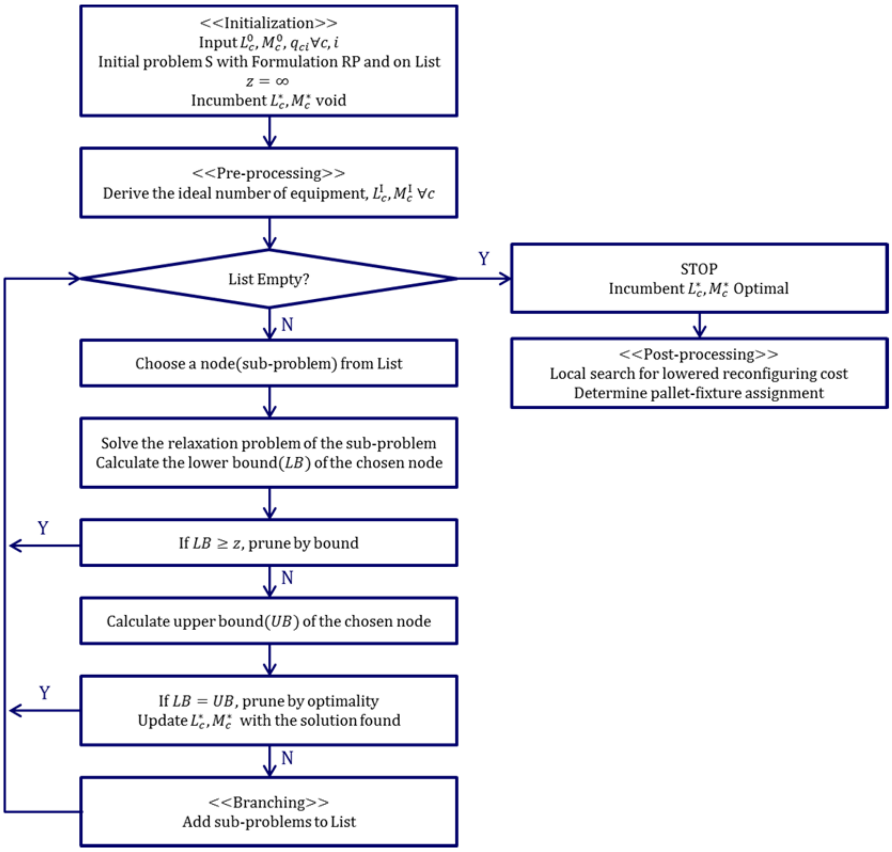
From the next section, the branch-and-bound algorithm using the proposed relaxation model is presented.



<Figure 5.4> The flow chart of RPX algorithm

5.2.2. Relaxation based Branch-and-bound Algorithm

The overall flow chart of the branch-and-bound algorithm is depicted in <Figure 5.5>.



<Figure 5.5> The flow chart of the branch-and-bound algorithm

5.2.2.1. Pre-processing: Derive the Ideal Number of Equipment

Before starting node search, derive the ideal number of equipment using **RPX** as defined in the previous section. From the ideal number, the net shortage is also calculated for each resource type. That is

$$\sum (L_c^I - L_c^0)$$

$$\sum (M_c^I - M_c^0)$$

for L/U stations and NC machines, respectively.

5.2.2.2. Choose a Node

After pre-processing, choose a node for search. Each node in B&B represents a sub-problem where decision variables are partially decided. The starting node is the node with depth 0 in the search tree, indicating the whole problem where none of decision variables is decided. If the depth of the chosen node is $\mathrm{k} \leq 2\mathrm{N}$ (where N is the number of the cells), this indicates the sub-problem in which k of decision variables are decided, and $2\mathrm{N}-\mathrm{k}$ variables should be further investigated. When the depth of the node in the search tree is equal to the number of decision variables (i.e. $2\mathrm{N}$), this mean that the complete solution is obtained.

5.2.2.3. Calculate the Lower Bound of the Chosen Node

A node represents a sub-problem where decision variables are partially decided. In the sub-problem, a cell may be categorized one of three: i) a cell where both decision variables (L_c, M_c) are decided, ii) a cell where only one of decision variables is decided, and iii) a cell none of decision variables is decided. Denote the set of those cells as A, B, and C, respectively. And without loss of generality, assume that the variable is decided in sequence of L_c, and then M_c.

To calculate the lower bound for cells $c \in \mathrm{B} \cup \mathrm{C}$, the relaxation problem (**RPX**) of the sub-problem should be solved. However, it may have no advantage to solve **RPX** instead of the original problem (**RP**). Therefore **RPX** is solved only if significant problem space reduction is guaranteed. The criterion to solve **RPX** is like below:

$$\text{Combinations}(\mathbf{RPX}) \leq \mathrm{X} \times \mathrm{Y} \leq {}_{\sum L_c - 1}\mathrm{C}_{\mathrm{N}-1} \times {}_{\sum M_c - 1}\mathrm{C}_{\mathrm{N}-1} = \text{Combinations}(\mathbf{RP})$$

$$\mathrm{X} = \begin{cases} 1, \text{ if} \sum (L_c^0 - L_c^I) \geq 0 \\ {}_{\sum (L_c^I - L_c^0) + N - 1}\mathrm{C}_{\mathrm{N}-1}, \text{ othersiwe} \end{cases} \quad Y = \begin{cases} 1, \, if \sum (M_c^0 - M_c^I) \geq 0 \\ {}_{\sum (M_c^I - M_c^0) + N - 1}C_{N-1}, \, othersiwe \end{cases}$$

Now the lower bound of the sub-problem can be calculated like below:

For $c \in \mathrm{A}$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c, \sum q_{ci})} + \min_i m_i$$

$$LB_2^c = \min_i l_i + \frac{\sum m_i \times q_{ci}}{\min(M_c, \sum q_{ci})}$$

$$D_c = \alpha |L_c - L_c^0| + \beta |M_c - M_c^0|$$

$$LB_c = \mathrm{T}(\max\{LB_1^c, LB_2^c\} + D_c - H)$$

where $\mathrm{T(X)}$ is the tardiness function that which outputs X when $\mathrm{X} \geq 0$, and

0, otherwise.

Note that the values of L_c and M_c are known for $c \in$ A.

For $c \in$ B

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c,\ \sum q_{ci})} + \min_i m_i$$

$$LB_2^c = \min_i l_i \ + \frac{\sum m_i \times q_{ci}}{\min(M_c^{*\mathbf{RPX}},\ \sum q_{ci})} + \beta\left|M_c^{*\mathbf{RPX}} - M_c^0\right|$$

$$D_c = \alpha|L_c - L_c^0|$$

$$LB_c = \mathrm{T}(\max\{LB_1^c + D_c, LB_2^c\} - H)$$

where $M_c^{*\mathbf{RPX}}$ is the optimal value obtained by solving the sub-problem with the relaxation model.

Note that the value of L_c is known for $c \in$ B.

For $c \in$ C

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c^{*\mathbf{RPX}},\ \sum q_{ci})} + \min_i m_i \ + \alpha\left|L_c^{*\mathbf{RPX}} - L_c^0\right|$$

$$LB_2^c = \min_i l_i + \frac{\sum m_i \times q_{ci}}{\min(M_c^{*\mathbf{RPX}}, \sum q_{ci})} + \beta\left|M_c^{*\mathbf{RPX}} - M_c^0\right|$$

$$LB_c = \mathrm{T}(\max\{LB_1^c, LB_2^c\} - H)$$

where $L_c^{*\mathbf{RPX}}$, $M_c^{*\mathbf{RPX}}$ is the optimal value obtained by solving the sub-problem with the relaxation model.

In the case where the relaxation problem is not solved, set related variables equal to 0. Even if the relaxation is not useful in one sub-problem, it may be useful in another sub-problem inside the chose node. Moreover though relaxation does not work at all, still the lower bound can be calculated for the cells $c \in A$.

Finally the lower bound of the sub-problem is given below:

$$LB = \sum_{c \in A \cup B \cup C} LB_c$$

5.2.2.4. Determine Whether to Prune the Chosen Node

If the calculated lower bound of the chosen node is greater than or equal to the upper bound of the whole problem, there is no reason for further investigation. Therefor prune the node ('prune by bound'). On the other hand, if the lower bound is smaller than the upper bound, there is still possibility that an optimal solution may exist in the sub-tree below the chosen node.

5.2.2.5. Calculate the Upper Bound of the Chosen Node

This stage happens when it is not decided to prune the chosen node.

To calculate the upper bound, the scheduling algorithm should be applied to each cell and therefore the complete solution is required. It is when the depth of the chosen node is equal to the number of variables.

On the other hand, it is also possible to obtain the complete solution of the relaxation problem before the search reaches leaf nodes. And if the upper bound value is the same with the lower bound, it implies that the optimal solution of the relaxation problem is the optimal solution of the original sub-problem, and it needs not to search further into branches of the node.

However, it is hard to expect meet such a pessimistic case, and it takes time to generate the schedules to obtain C_{max}^{c}. Therefore the scheduling algorithm is applied to the special case when the objective value of the relaxation problem equals to 0 excluding cells $c \in A$.

If the calculated upper bound is smaller than the upper bound of the whole problem, it means the corresponding solution is the best found so far. In that case update the value of the best solution and the upper bound with that of the chosen node.

Moreover if the calculated upper bound is the same with the lower bound, it means the derived solution of the chosen node is the best one among the solutions in the sub-tree below the chosen node, and therefore further investigation is not required. In that case prune the node ('prune by optimality').

5.2.2.6. Branch for Further Investigation

If no pruning decision has been made in previous stages, and the chosen node is not the leap node, branch the node for further investigation. The number of branches is determined by the remaining number of equipment which is the total number of equipment minus the sum of the value of decision variables above the chosen node.

5.2.2.7. Terminate the algorithm

If there is left no node to investigate, terminate the algorithm and output the best solution and the corresponding objective value.

5.2.2.8. Post-processing

After solving **RP** using the suggested B&B algorithm, the L_c^*, M_c^* ∀c are decided together with $z^{*\mathbf{RP}}$. Though the algorithm considered reconfiguring time (D_c), there may be alternative solutions with the same $z^{*\mathbf{RP}}$ but with lowered reconfiguring cost. To find such alternative solutions, a neighborhood search algorithm is designed of which pseudo code is depicted in <Figure 5.6>. The search is conducted until no improvement can be made.

Algorithm NeighborhoodSearch
$D = \sum D_c(L_c^*, M_c^*)$; //initial reconfiguring cost. Note $L_c^*, M_c^* \forall c$ are given
do
  foreach cell c
    if $L_c^0 < L_c^*$
      $L_c' \leftarrow L_c^* - 1$;
      if $C_{\max}^c(L_c', M_c^*) + D_c(L_c', M_c^*) \leq H$ **or**
          $C_{\max}^c(L_c', M_c^*) + D_c(L_c', M_c^*) \leq C_{\max}^c(L_c^*, M_c^*) + D_c(L_c^*, M_c^*)$
      //the condition above indicates $z^{*\mathbf{RP}}$ does not decreased with L_c'
        find k such that $L_k^0 > L_k^*$ //at least one exists
        $L_c^* \leftarrow L_c^* - 1$; //this reduces reconfiguring cost, not decreasing $z^{*\mathbf{RP}}$
        $L_k^* \leftarrow L_k^* + 1$; //this also reduces reconfiguring cost, not decreasing $z^{*\mathbf{RP}}$
      endif
    endif
    if $M_c^0 < M_c^*$ //the same for L_c^*
      $M_c' \leftarrow M_c^* - 1$;
      if $C_{\max}^c(L_c^*, M_c') + D_c(L_c^*, M_c') \leq H$ **or**
          $C_{\max}^c(L_c^*, M_c') + D_c(L_c^*, M_c') \leq C_{\max}^c(L_c^*, M_c^*) + D_c(L_c^*, M_c^*)$
        find k such that $M_k^0 > M_k^*$
        $M_c^* \leftarrow M_c^* - 1$;
        $M_k^* \leftarrow M_k^* + 1$;
      endif
    endif
  endforeach
while $D \neq \sum D_c(L_c^*, M_c^*)$ //until no improvement is made
end do while

<Figure 5.6> Pseudo code of the neighborhood search algorithm

After improved solution is obtained, the value of the pallet-fixture assignment

$\underline{p_{ci}}$ is derived from L_c^*, M_c^*. Remember the value is the minimal number such that the corresponding part type does not cause pallet-bottleneck. Underlying assumption in this approach is that the total number of pallets of the whole factory is enough to satisfy minimal requirement and that it is possible to resolve a temporary pallet-bottleneck case via adjusting pallets.

According to the derived pallet-fixture assignment, it is determined whether it is possible to adjust the current pallet-fixture assignment with pallets inside the cell or additional pallets should be provided. In the latter case, additional pallets are delivered from the cells which have excess pallets above the minimal number, to the cell in lacking.

5.3. Experiment

To verify the performance of the proposed algorithm, an experiment was conducted.

To experiment various shop conditions, three experimental factors were considered: the number of RMCs, total number of L/U stations and total number of NC machines. For an experimental condition, the initial configuration was generated randomly.

Moreover, the unit reconfiguring time was set 480 min. for L/U station ($\alpha = 480$), and 720 min. for NC machine ($\beta = 720$).

The demand period or the available time (H) was set to 200 hour or 12000 min.(8 hour/day $\times$ 25days $\times$ 60min./hour). The number of parts per each cell was set to 10. The production volume for each part type (q_{ci}) was generated randomly but the initial configuration was reflected so that the following equation holds:

$$\frac{\sum m_i \times q_{ci}}{M_c^0} \cong H \times (1 - \mathrm{UNIF}(-\delta, \delta))$$

where δ is the demand fluctuation factor, set to 0.3 as default in the experiment. $\mathrm{UNIF}(-\delta, \delta)$ is a random variable generated from the uniform distribution with the minimum value of $-\delta$ and the maximum value of δ. Weights of cells were set the same for all cell ($w_c = 1 \forall\, c$). The rest of parameters were generated as in chapter 4.

For benchmarking, the general B&B algorithm was considered which does not utilize the relaxation models to obtain the lower and upper bounds, but uses a simple lower bound scheme.

As performance measures, the number of searched nodes, the search time to reach the optimum and the number of calls of scheduling procedure (CoS) were recorded for both the proposed and general B&B algorithms. Note that the scheduling procedure requires relatively large amount of time and it is desired to reduce the number of calls of it [69].

Finally for each experimental condition, the average performance from 10 replications was recorded.

5.4. Result and Analysis

<Table 5.1> indicates the experimental result. The result showed that through the proposed B&B algorithm, the search effort is significantly reduced compared to the general B&B algorithm. The proposed B&B showed improved performance for all measures - the number of searched nodes and the CoS measure as well as the search time. It was also observed that the proposed B&B appeared to work more efficiently as the problem size gets larger.

Experimental result verifies the efficacy of the relaxation model (**RPX**) and the proposed B&B algorithm based on the relaxation model.

Among performance measures, improvement in search time was relatively smaller than that in the other measures. It was because it takes time to solve the relaxation problem itself. Remember that the relaxation problem is solved almost every time a chosen node (a partial solution) is examined. If this redundancy is eliminated in solving the relaxation problem, more improvement would be made.

<Table 5.1> The experimental result

Cell	***L/U***	***NC***	***Suggested B&B***			***General B&B***			***%Improvement***		
			Searched nodes	Search time(sec.)	CoS	Searched nodes	Search time(sec.)	CoS	Search time	Searched nodes	CoS
4	8	12	706	54	19	2099	70	33	22	66	42
4	8	16	2442	75	36	6153	81	49	8	60	27
4	8	20	5704	156	67	18056	159	75	2	68	10
4	12	12	1105	189	29	7301	268	42	30	85	31
4	12	16	8874	466	54	31363	522	64	11	72	16
4	12	20	4556	270	59	17859	312	83	14	74	29
4	16	12	3545	602	45	24437	746	58	19	85	21
4	16	16	4022	579	46	38036	1043	75	45	89	38
4	16	20	3193	556	62	13100	670	81	17	76	23
6	12	18	1659	53	27	19036	79	59	33	91	55
6	12	24	15884	85	55	434409	132	92	36	96	40
6	12	30	67222	178	92	2266121	297	137	40	97	32

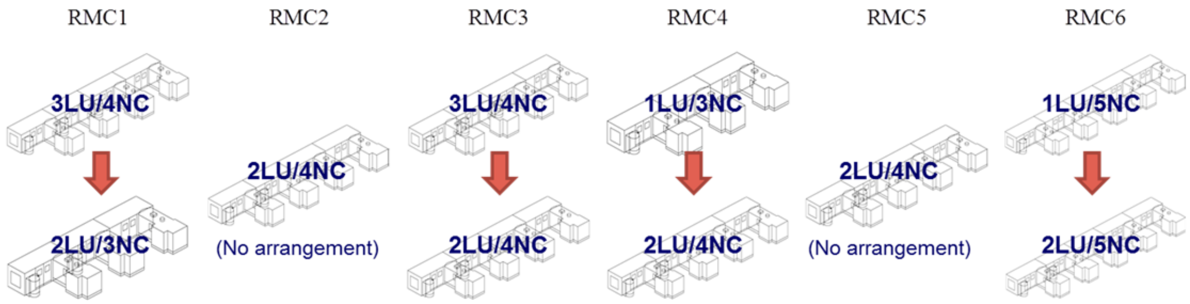
6	18	18	3017	93	24	143870	296	61	69	98	62
6	18	24	4781	175	49	129001	295	95	41	96	49
6	18	30	11547	221	77	1339292	376	147	41	99	48
6	24	18	24882	555	44	1302117	883	67	37	98	34
6	24	24	3282	224	29	81657	930	91	76	96	68
6	24	30	33582	1444	116	7748436	1827	142	21	100	18
10	20	30	10407656	1662	43	98187443	3704	111	55	89	61
								Average(%):	**32**	**86**	**37**

<Table 5.2> and <Figure 5.7>, and <Figure 5.8> show an exemplary result of **RP.** In <Table 5.2>, arrangement of equipment and required time for such arrangement are presented for each RMC, together with change of manufacturing performance (i.e. production delay) before and after hardware reconfiguration. Note that the values about pallets are the minimal number not causing bottleneck. <Figure 5.7> indicates graphical representation of hardware reconfiguration. Change of the schedule with respect to arrange of equipment is depicted in <Figure 5.8>. (Only parts of the schedule are depicted because the length of the entire schedules is too large)

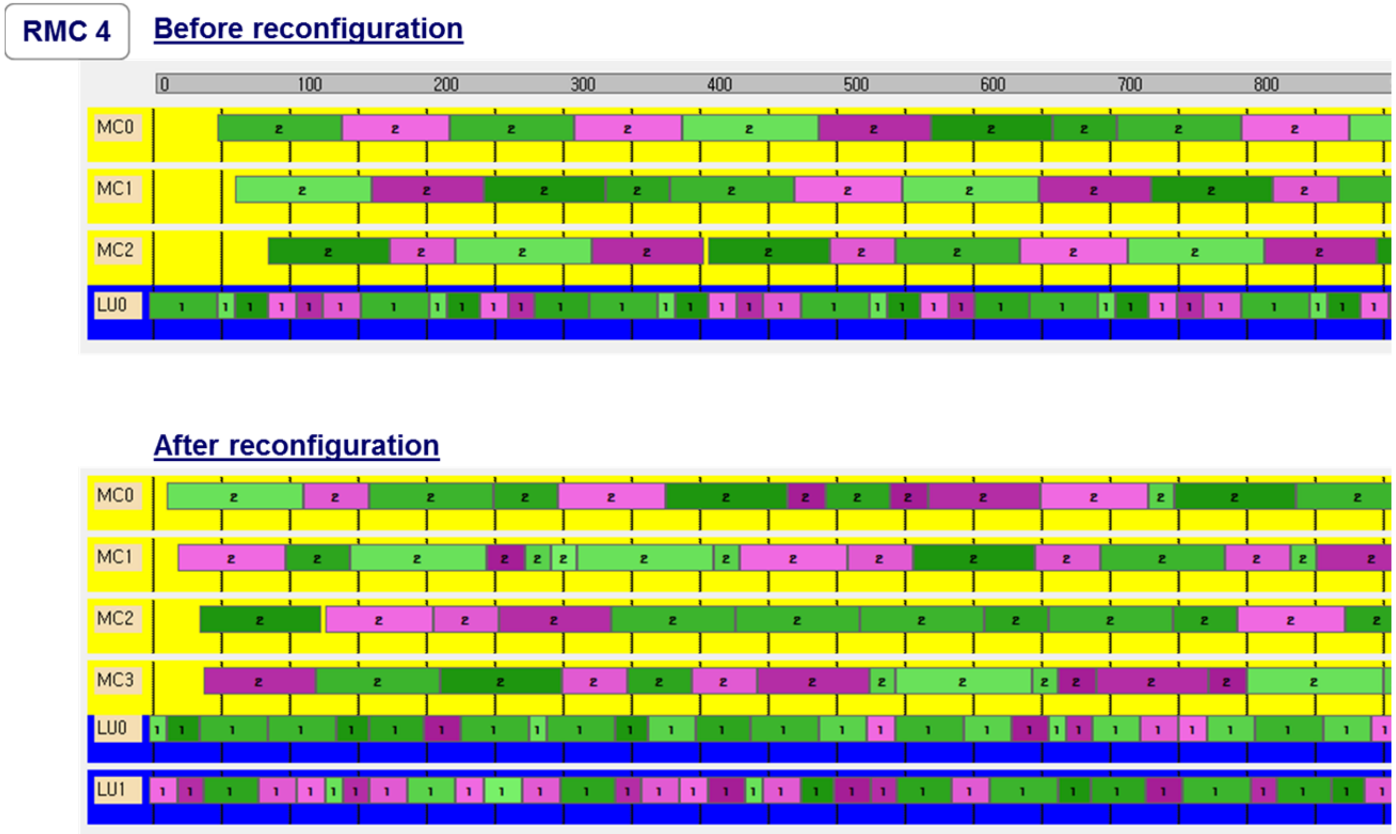
With the proposed algorithm, and derived result, managers may obtain information for hardware reconfiguration and be able to make decisions for response to demand.

<Table 5.2> The result of hardware reconfiguration problem (example)

	RMC1	*RMC2*	*RMC3*	*RMC4*	*RMC5*	*RMC 6*
L/U	3→2	No arrangement	3→2	1→2	No arrangement	1→2
NC	4→3		4→4	3→4		5
PA(min#)	>10	>10	>10	>11	>10	>11
Reconf. time(hour)	20	-	8	20	-	8
Prod. delay(hour)	0→26	0	0→0	124→8	0	240→28



<Figure 5.7> The result of hardware reconfiguration problem in graphical representation (an example)



<Figure 5.8> The result of hardware reconfiguration problem in change of schedules (an example, partially presented)

6. RMC Equipment Purchase Planning

In long term, production managers are concerned with whether the current system may catch up with the demand pattern. If they judge the current system is not enough to cover future demand, then they may consider providing additional capacity by purchasing equipment or outsourcing demand in part. Our research considers the former.

The **reconfigurable equipment purchase problem** (REPP) is defined as: given demand forecast, to find the best purchase plan which specifies whether, which and how many equipment would be brought and when, to catch up with future demand.

REPP is often referred in previous studies as the capacity scalability problem [13][14][15], or the scalability planning problem [91]. Unlike the traditional FMS design problem [83], the capacity scalability problem is concerned with dynamic changes of production capacity along demand periods, including both capacity expansion and reduction. Here capacity reduction indicates decreasing capacity of some parts of a production system while capacity of other parts is increased. This —dynamic changes of production capacity — is enabled via reconfigurable capability of reconfigurable manufacturing systems (RMSs), therefore hardware reconfiguration should be considered together in the resolution of the capacity scalability problem.

Traditionally equipment purchasing decisions were usually made in strategic-level of detail [12][76][89]. However, in RMSs, hardware reconfiguration happens frequently in short term and reconfiguration decisions are made based on operational-level information (i.e. schedule information in our study). Therefore the equipment purchasing in RMS needs to be considered in operational level in relation with hardware reconfiguration. This is more practical as reconfiguring capability of RMS is evolved and time required for hardware reconfiguration is reduced. (The final objective for system-setup (i.e. hardware reconfiguration of existing and purchased equipment) time in the governmental development project for RMCs [42] is one day.)

In this background, this chapter aims to define and resolve the reconfigurable capacity planning problem in operational level of detail. In the next section extended production environment is explained and in the following section a mathematical model is introduced. After that the solution algorithm to find the best purchase plan is suggested. Then an experiment is conducted to verify the suggested algorithm together with analysis of result.

6.1. Mathematical Model

Production environment is basically the same as in chapter 3, but a little bit extended.

A factory consists of multiple production lines and each has an RMC as a part of it. Each RMC is defined by the cell configuration, by the number of L/U stations, NC machines and pallets.

Demand forecast is provided in advance by the marketing department. Demand forecast contains expected demand for multi periods. For each period, production volume is specified along product types. By this way the correlation inside and among product groups would be considered implicitly.

According to demand forecast, equipment purchase is considered. Purchasable components are: L/U stations, NC machines, and pallets. Performance and purchasing cost is assumed identical for components of the same type. And purchased equipment is assumed to be used for production just after introduction without the testing period.

Managerial concern is to cover future demand for entire periods with minimized purchasing cost. Most of studies on the FMS design or the RMS capacity scalability problem set demand satisfaction as a hard constraint and the capital cost as the objective function [15][17][76][89], and the capital cost consists of the investment cost(i.e. purchasing cost) and the operation cost. In this study, the operation cost is assumed relatively small compared to the investment cost, therefore is neglected. This assumption is originated from the fact that target RMCs can rapidly change hardware configurations with small reconfiguration cost which is one of major cost in the operation of an RMS. Besides, information about operations of RMCs is described in chapter 4.

Based on the description of production environment, the following mathematical model for the reconfigurable equipment purchase problem (**REPP**) is built:

REPP

· Input parameters (New or redefined variables only)

t : Index of production period

q_{cit}: Demand forecast of part i of cell c for production period t

L_{c0}^{0} : The initial number of L/U stations of cell c at the starting period

M_{c0}^{0}: The initial number of NC machines of cell c at the starting period

E^{LU}: Unit purchasing cost of L/U station

E^{NC}: Unit purchasing cost of NC machine

ε: The risk-free interest rate

· Decision variables

L_t^{+} : The number of L/U stations which is to be bought at time *t*

M_t^{+}: The number of NC machines which is to be bought at time *t*

· Dependent variables

L_{ct}^{0}: The input value of L_c^0 for the problem $\mathbf{RP}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit})$

M_{ct}^{*}: The input value of M_c^0 for the problem $\mathbf{RP}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit})$

L_{ct}^{*}: The optimal value of L_c of the problem $\mathbf{RP}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit})$

M_{ct}^{*}: The optimal value of M_c of the problem $\mathbf{RP}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit})$

$\mathbf{RP}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit})$ indicates an instance of the reconfiguring problem with input parameters of $L_c^0 = L_{ct}^0, M_c^0 = M_{ct}^0, q_{ci} = q_{cit}, \sum_c L_c^0 = \sum_c L_{ct}^0 + L_t^+, \sum_c M_c^0 = \sum_c M_{ct}^0 + M_t^+$. Note that the last two conditions are to modify the constraint (1) in **RP** to reflect introduction of purchased equipment.

· Constraints

$\mathrm{z}^{*\mathbf{RP}}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit}) \leq 0 \forall\, t$ (1)

$L_{\mathrm{ct}}^{0} = L_{\mathrm{c(t-1)}}^{*}, M_{\mathrm{ct}}^{0} = M_{\mathrm{c(t-1)}}^{*} \forall \mathrm{t}$ (2)

$L_t^+, M_t^+ \in \mathrm{Z}^+ \cup \{0\} \forall t$ (3)

· Objective function

$$z^{\mathbf{REPP}} = \min \sum_t \left(E^{LU} \times \frac{L_t^+}{(1+\varepsilon)^{t-1}} + E^{NC} \times \frac{M_t^+}{(1+\varepsilon)^{t-1}} \right) \qquad (4)$$

(1) indicates the production capacity should cover demand forecast of every production periods. (2) indicates the result of hardware reconfiguration of the former period becomes the initial configuration of the following period. (3) indicates the integrality condition. Finally (5) represents the net-present-value

(NPV) of total purchasing cost considering the risk-free rate.

Note that although the above model seems not considering purchase of pallets, the purchase amount of pallets is also derived through **REPP**. Remember the minimal number of pallets are determined through **RP** whenever the number of equipment (L^*_{ct}, M^*_{ct}) is decided. Additional pallets need to be purchased when the sum of the minimal pallets exceeds the total pallets of the factory. On the other hand, purchasing cost of pallets is not considered in the objective function, because the unit cost of pallets is considered small compared to that for L/U stations or NC machines.

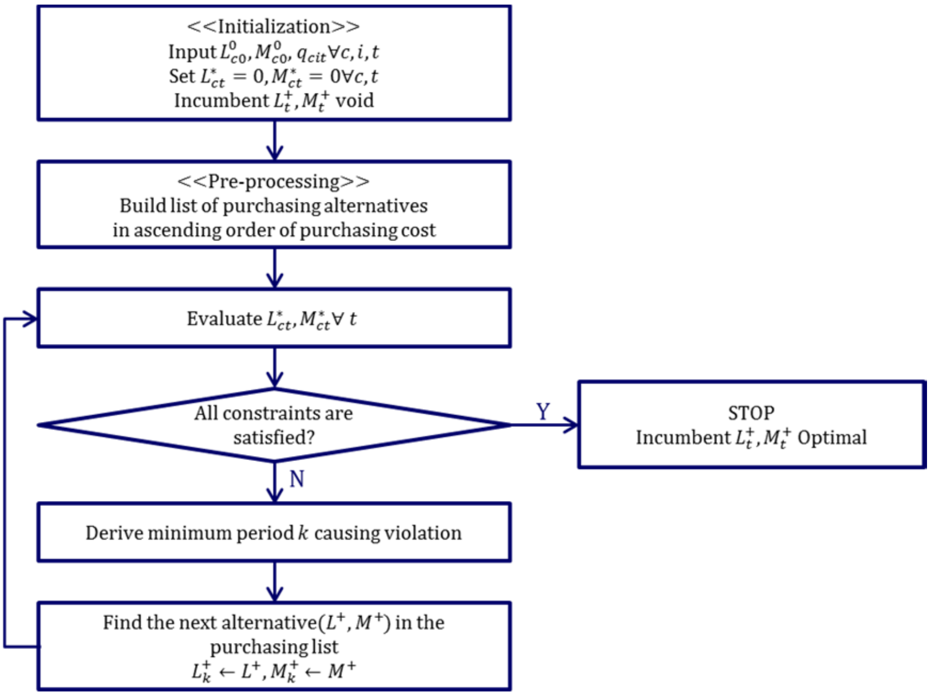
6.2. Development of Algorithm

In this section the algorithm for **REPP** is discussed.

Considering that the objective is cost-minimization, the following facts are derived:

i) It would be better to purchase minimum number of equipment
ii) If it needs to purchase the same number of equipment, it would be better to purchase the cheaper equipment as many as possible
iii) If the number and the type of equipment is the same, It would be better to purchase as late as possible

Considering the facts above, the following algorithm is designed. The flow chart of the algorithm is depicted in <Figure 6.1>.



<Figure 6.1> The flow chart of REPP algorithm

Description of the algorithm is like below:

First, as pre-processing, build the list of purchasing alternatives.

The best purchasing alternative is no purchasing. The next would be to purchase cheaper equipment by one. Like this, the list may be built for all purchasing alternatives, considering the unit purchasing cost of L/U station and NC machine. <Table 6.1> shows the example of such list.

<Table 6.1> An example of the purchasing alternative list

L^+	0	1	0	1	2	0	3	…
M^+	0	0	1	1	0	2	0	…
Purchasing cost ($E^{LU} \times L^+ + E^{NC} \times M^+$)	0	150	200	350	300	400	450	…

Then, stating with the current solution of no purchasing (i.e. $L_t^+ = M_t^+ = 0 \forall t$), evaluate constraints with the current solution.
Constraints would be evaluated by solving **RP** in sequence from earlier periods to later periods. If the current solution does not violate any constraint, then the current solution would be the best solution, since the solution is evaluated ascending order of the objective function.

If some of constraints are violated, then the minimum period t can be derived where the violation happens. To resolve the violation, the current solution L_t^+, M_t^+ should be updated. The values are chosen the first pair from the alternative list excluding selected values.

Through the suggested algorithm, the optimal purchase plan can be derived.

6.3. Experiment

To verify the performance of the **REPP** algorithm, an experiment was conducted.

The experiment was conducted varying the number of cells, total number of L/U stations and total number of NC machines. For given an experimental condition, the initial configuration (L_{c0}^{0} M_{c0}^{0}) of the starting period was generated randomly.

Demand forecast (q_{cit}) is generated for upcoming 12 periods by weighted moving average method from past 6 periods. Each period's demand indicates the expected monthly demand, and past demand is generated randomly.

Moreover, the unit purchasing cost set 100 million KRW for L/U station ($E^{LU} = 100$), and 300 million KRW for NC machine ($E^{NC} = 300$).

The demand period or the available time for each period (H) was set to 200 hour (1 month) in accordance with demand forecast. The number of part types per each cell was set to 10. The rest of parameters were generated as in chapter 4.

As a performance measure, the search time is gathered.

Finally for each experimental condition, the average performance from 5 replications was recorded.

6.4. Result and Analysis

<Table 6.2> shows the experimental result. It took a little longer to solve the equipment purchasing problem than to solve the reconfiguring problem, since it should solve the reconfiguration problem for multi periods and in an iterative manner. But the search time is still acceptable. Note that the result of the problem includes not only the purchase plan which specifies equipment to be purchased, proper time of purchasing, and total purchasing cost, but also reconfiguration plan with the purchased equipment for entire periods, and the corresponding schedules.

<Table 6.2> The experimental result

Cell	***L/U***	***NC***	***Search time (sec.)***
4	8	12	58
4	8	16	105
4	12	12	121
4	12	16	287
6	12	18	104
6	12	24	125
6	18	18	237
6	18	24	181
10	20	30	196
10	20	40	550
10	30	30	1066
10	30	40	910

<Table 6.3>-<Table 6.5> and <Figure 6.2> show an exemplary result of **REPP**. In <Table 6.3>, recommendation for purchase of equipment and hardware reconfiguration with purchased equipment are presented for each production period, and its graphical representation for a single period is depicted in <Figure 6.2>. The reason why it does not require hardware reconfiguration in later periods is that demand forecast is generated by moving average method, so demand is less fluctuated in later periods. Information about purchasing cost is depicted in <Table 6.4>. Benefit of equipment purchase is of improved manufacturing performance (i.e. reduced production delay) which is depicted in <Table 6.5>. Each value in <Table 6.5> indicates estimated production delay (hour) of each RMC. For comparison, manufacturing performance in the case without equipment purchase is also presented. From difference of production delay between two cases (with and without equipment purchase), return-on-investment (ROI) of purchased equipment can be calculated like below:

$$\mathrm{ROI} = \frac{\sum_t \frac{\mathrm{OC}}{(1+\varepsilon)^{t-1}} \times \left(\mathrm{z}^{*\mathbf{RP}}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit}) - \mathrm{z}^{*\mathbf{RP}}(L_{ct}^0, M_{ct}^0, 0,0, q_{cit})\right)}{\mathrm{z}^{*\mathbf{REPP}}}$$

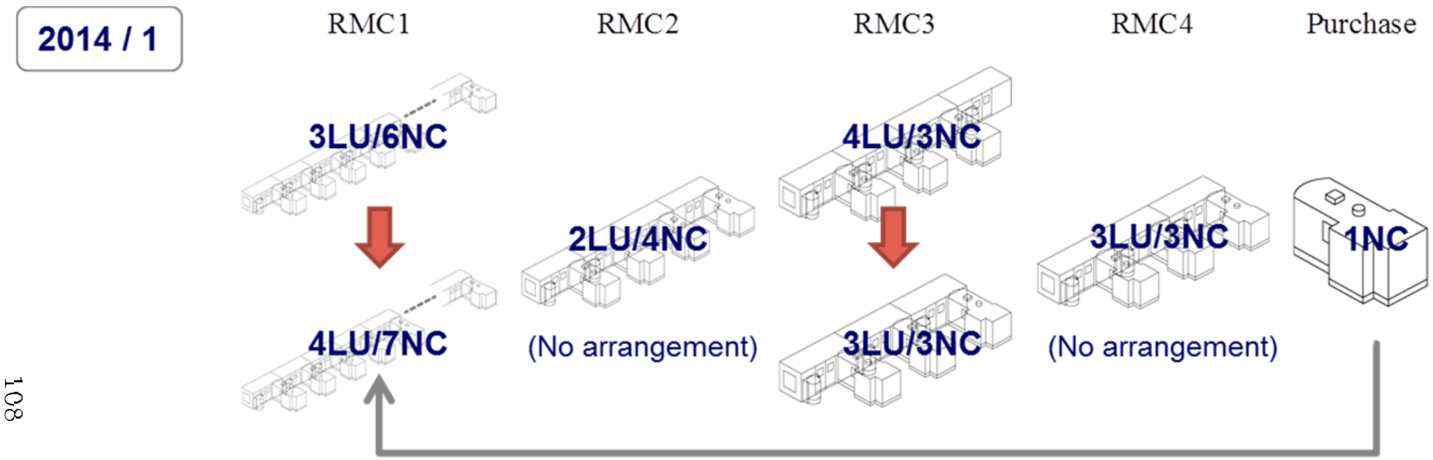
, where OC is the operation cost of an RMC per an hour.

With the proposed algorithm, managers may obtain information for equipment purchase in long-term aspect.

<Table 6.3> The result of reconfigurable equipment purchase problem (an example)

Period (Month)	*Resource type*	*RMC1*	*RMC2*	*RMC3*	*RMC4*	*Purchase*
2014 / 1	L/U	3→4	No arrangement	4→3	No Arr.	0
	NC	6→7		3→3		1
2	L/U	4→3	2→2	3→4	No Arr.	0
	NC	7→7	4→5	3→3		1
3	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
4	L/U	No Arr.	No Arr.	No Arr.	3→3	0
	NC				3→4	1
5	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
6	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
7	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
8	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
9	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0

10	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0
11	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0
12	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0



<Figure 6.2> The result of reconfigurable equipment purchase problem in graphical representation (example)

<Table 6.4> Cost of equipment purchase (an example)

Period (Month)	***Purchase - L/U station***	***Purchase - NC machine***	***Purchasing cost (million KRW)***
2014/ 1	-	1	300
2	-	1	299
3	-	-	-
4	-	1	296
5	-	-	-
6	-	-	-
7	-	-	-
8	-	-	-
9	-	-	-
10	-	-	-
11	-	-	-
12	-	-	-
Total	**0**	**3**	**895**

<Table 6.5> Benefit of equipment purchase: Reduced production delay (an example)

Period **(Month)**	***Equipmenddt purchase***				***No purchase***				***Time saving (hour)***
	RMC1	*RMC2*	*RMC3*	*RMC4*	*RMC1*	*RMC2*	*RMC3*	*RMC4*	
2014/1	0	0	0	0	25	0	0	0	25
2	0	0	0	0	21	1	0	0	22
3	0	0	0	0	22	1	0	0	23
4	0	0	0	0	22	4	0	1	27
5	0	0	0	0	21	8	0	2	31
6	0	0	0	0	21	3	0	0	24
7	0	0	0	0	21	3	0	0	23
8	0	0	0	0	21	3	0	0	24
9	0	0	0	0	21	3	0	0	24
10	0	0	0	0	21	3	0	0	24
11	0	0	0	0	21	3	0	0	24
12	0	0	0	0	21	3	0	0	24
					Average(hour/month):				**25**

7. Conclusion

This study has aimed to develop an integrative decision-making system for RMC environment, a manufacturing system with hardware reconfiguring capability. For reconfiguring capability of RMCs, 1-2 days of reconfiguring time has been considered. The capability is still advancing.

Managerial problems encountered in the operation of RMCs have been defined in chapter 3. In chapter 4, a scheduling algorithm appropriate for those problems has been developed as a basic component of the decision-making system. Using the developed scheduling algorithm, in chapter 5, a mathematical model is suggested to maximize demand fulfillment for the reconfiguring problem. In the model, impact of reconfiguring time on manufacturing performance is considered. For resolution of the reconfiguring problem, a relaxation based branch-and-bound algorithm is proposed, and properties for reduction of the problem dimension and reduction in the domain of variables of the relaxation problem have been investigated. A local (neighborhood) search is also included in the branch-and-bound algorithm to find solutions with reduced reconfiguring cost while maintaining the optimum level of demand fulfillment. Finally, in chapter 6, equipment purchasing is considered to respond to long-term demand forecast. A multi-period model where each period represents a reconfiguring problem is proposed and an exact algorithm for that model is suggested.

For the development of a scheduling algorithm, the lower bound prediction scheme based on work-in-process projection works consistently and effectively for the measure of makespan, with small computational cost. The proposed approach is somewhat compensating the limitation of the greedy (dispatching) approach by looking ahead of the future status of manufacturing systems. Consideration of the lower bound is related to bottleneck (i.e. constrained resource) control.

For the reconfiguring problem, difficulty has come from numerous configurations that should be considered. Some constraints such as the reconfiguring time were obstacles to simplification of the problem. To overcome such a problem structure, the relaxation model has helped eliminate unpromising or redundant alternatives and enabled efficient search. Through the alternative formulation, the benefit has been expressed apparently.

The reconfigurable equipment purchase problem, though it is the extended one of the reconfiguring problem, could been solved plainly (relative to the other problems) relying on the resolution technique of the reconfiguring problem. The reconfiguring problem itself has taken advantage from the scheduling algorithm which is consistent and fast.

Investigation on impact of the pallet-fixture assignment change is remained for the future study. This may be valuable when short-term (e.g. a week) response of production system is concerned, after the hardware configuration is decided. Moreover, in this study, a single type of machining equipment is assumed, since the RMC in development consists of a single type of machines (i.e. horizontal machining centers). To cover more of the real problems, hardware reconfiguration with more than one type of machines such as vertical machines needs to be considered. And this study prohibited alternative material flows among RMCs in different production lines. In real situation, however, some of RMCs may be assigned to process product groups in different lines, while still others may not be used to do so. Finally in this study the required production time (i.e. makespan) is used as the performance measure to evaluate a given hardware configuration. Though the makespan may be a practical measure for production systems where utilization and productivity are important, there are other production systems where due date fulfillment of individual jobs are more important than productivity. Therefore evaluation of manufacturing performance with due date-related measures may help improving the suggested decision-making system.

There exists another challenge. Throughout the study, deterministic demand or demand forecast is considered. In reality, uncertainty exists. Nevertheless, managers should make decisions. Information gathering to resolve uncertainty should be aligned with hardware reconfiguration, toward the objective.

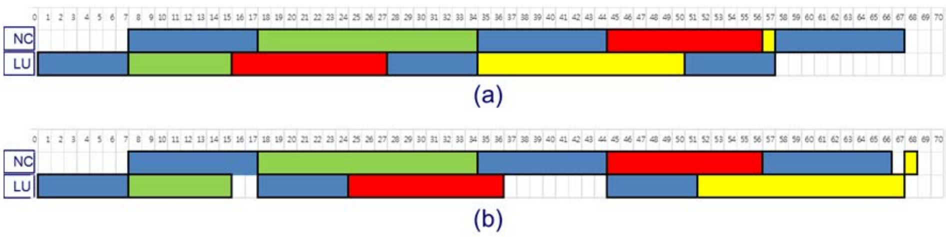
Appendix

Proof for property in section 4.1

Consider a case of 1 L/U station and 1 NC machine and 4 part types with (loading time(l_i), machining time(m_i), production quantity(q_i))-(12,12,1),(7,10,3),(8,17,1),(16,1,1), and that there is only one pallet for each part type (i.e. $p_i = 1 \; \forall \; i$).

For non-permutation schedules is the optimal schedule with the makespan of 67, indicated in <Figure A.0.1> (a).

For permutation schedules is the optimal schedule with the makespan of 68, indicated in <Figure A.0.1> (b). ■



<Figure A.0.1> The optimal schedules:

the non-permutation schedule (a) and the permutation schedule (b)

References

[1] M. R. Abdi and A. W. Labib, "A design strategy for reconfigurable manufacturing systems (RMSs) using analytical hierarchical process (AHP): a case study," *Int. J. Prod. Res.*, vol. 41, no. 10, pp. 2273-2299, 2003.

[2] M. R. Abdi and A. W. Labib, "Grouping and selecting products: the design key of reconfigurable manufacturing systems (RMSs)," *Int. J. Prod. Res.*, vol. 42, no. 3, pp. 521-546, 2004.

[3] M. R. Abdi, "Selection of a layout configuration for reconfigurable manufacturing systems using the AHP," in *Int. Symp. Analytic Hierarchical Process*, Honolulu, HI, 2005.

[4] A. Adlemo et al., "Towards a truly flexible manufacturing system," *Control Eng. Practice*, vol. 3, no. 4, pp. 545-554, 1995.

[5] F. M. Asl and A. G. Ulsoy, "Capacity management via feedback control in reconfigurable manufacturing systems," in *Proc. Japan-USA Symp. Flexible Automation*, Hiroshima, Japan, 2002.

[6] M. Azizoglu et al., "A flexible flowshop problem with total flow time minimization," *Euro. J. Opl. Res.*, vol. 132, pp. 528-538, 2001.

[7] K. R. Baker, *Introduction to sequencing and scheduling.* New York, NY: John Wiley and Sons, 1974.

[8] K. R. Baker, *Principles of sequencing and scheduling.* New York, NY: John Wiley and Sons, 2009.

[9] A. Bensmaine et al., “Design of reconfigurable manufacturing systems: optimal machines selection using non dominated sorting genetic algorithm (NSGA-II),” in *Proc. 41st Int. Conf. Computers and Industrial Engineering*, Los Angeles, CA, 2011, pp. 110-115.

[10] D. Chung., “A study on job shop scheduling problems considering production capacity adjustment,” Ph.D.'s Dissertation, Dept. Industrial Eng., Seoul Natl. Univ., Korea, 1999.

[11] D. Chung et al., “Developing a shop floor scheduling and control software for an FMS,” *Comput. Industrial Eng.*, vol. 30, no. 3, pp. 557-568, 1996.

[12] Y. Dallery and Y. Frein, “An efficient method to determine the optimal configuration of a flexible manufectudng system,” *Ann. Ops. Res.*, vol. 15, pp. 207-225, 1988.

[13] A. M. Deif and H. A. ElMaraghy, “Assessing capacity scalability policies in RMS using system dynamics,” *Int. J. Flex. Manuf. Syst.*, vol. 19, pp. 128-150, 2007.

[14] A. M. Deif and W. H. EIMaraghy, “A control approach to explore the dynamics of capacity scalability in reconfigurable manufacturing systems,” *J. Manuf. Syst.*, vol. 25, no. 1, pp. 12-24, 2006.

[15] A. M. Deif and W. H. ElMaraghy, “Investigating optimal capacity scalability scheduling in a reconfigurable manufacturing system,” *Int. J. Adv. Manuf. Technol.*, vol. 32, pp. 557-562, 2007.

[16] D. R. Denzler et al., “An experimental investigation of FMS scheduling rules under uncertainty,” *J. Ops. Manage.*, vol. 7, no. 1-2, pp. 139-151,

1987.

[17] J. Dou et al., “A GA-based approach for optimizing single-part flow-line configurations of RMS,” *J. Intell. Manuf.* vol. 22, pp. 301-317, 2011.

[18] H. A. ElMaraghy, “Flexible and reconfigurable manufacturing systems paradigms,” *Int. J. Flex. Manuf. Syst.*, vol. 17, pp. 261-276, 2006.

[19] R. Galan et al., “A systematic approach for product families formation in reconfigurable manufacturing systems,” *Robotics Computer-Integrated Manuf.*, vol. 23, pp. 489–502, 2007.

[20] P. Gilmore and R. Gomory, “Sequencing a one state-variable machine: a solvable case of the traveling salesman problem,” *Ops. Res.*, vol. 12, no. 5, pp. 655-679, 1964.

[21] A. Guinet et al., “A computational study of heuristics for two-stage flexible flowshops,” *Int. J. Prod. Res.*, vol. 34, no. 5, pp. 1399-1415, 1996.

[22] H. Gultekin, et al., “Scheduling in a three-machine robotic flexible manufacturing cell,” *Comput. Ops. Res.*, vol. 34, no. 3, pp. 2463-2477, 2007.

[23] J. N. D. Gupta, “Hybrid flowshop scheduling problem,” *J. Opl. Res. Soc.*, vol. 39, no. 4, pp. 359-364, 1988.

[24] Y. Gupta et al., “A genetic algorithm-based approach to cell composition and layout design problems,” *Int. J. Prod. Res.*, vol. 34, no. 2, pp. 447-482, 1996.

[25] S. Han et al., “Pallet-fixture allocation in reconfigurable manufacturing cells: an integrative approach,” *J. Korean Soc. Precision Eng.*, vol. 20, pp.

357-366, 2012.

[26] F. Hasan et al., “Optimum configuration selection in reconfigurable manufacturing system involving multiple part families,” *OPSEARCH*, DOI 10.1007/s12597-013-0146-1, 2013.

[27] L. Huang et al., “Configuration selection for reconfigurable manufacturing systems by means of characteristic state space,” *Chinese J. Mech. Eng.*, vol. 23, pp. 1-10, 2010.

[28] IBM Corp.(2009) *IBM ILOG OPL Language User's Manual* [Online]. Available: http://pic.dhe.ibm.com/infocenter/odmeinfo/v3r4/index.jsp?topic=%2Filog.odms.ide.odme.help%2FContent%2FOptimization%2FDocumentation%2FODME%2F_pubskel%2FODME_pubskels%2Fstartall_ODME34_Eclipse991.html.

[29] S. Jang et al., “An integrated decision support system for FMS production planning and scheduling problems,” *Int. J. Manuf. Technol.*, vol. 11, pp. 101-110, 1996.

[30] H. Jeong et al., “A study about integrated supply chain planning model with considering upstream and down-stream negotiation for maximizing profit in open business environment,” *Entrue J. Inform. Technol.*, vol. 11, no. 2, pp. 135-153, 2012.

[31] Y. Koren and M. Shpitalni, “Design of reconfigurable manufacturing systems,” *J. Manuf. Syst.*, vol. 29, pp. 130–141, 2010.

[32] Y. Koren and A. G. Ulsoy. “Reconfigurable manufacturing system having a production capacity method for designing same and method for

changing its production capacity," U.S. Patent 6 349 237, Feb 19, 2002.

[33] Y. Koren and A. G. Ulsoy, "Vision, principles and impact of reconfigurable manufacturing systems," *Powertrain Int.*, vol. 5, no. 3, pp. 14-21, 2002.

[34] Y. Koren et al., "Impact of manufacturing system configuration on performance". *Ann. CIRP*, vol. 47, no. 1, pp. 369-372, 1998.

[35] Y. Koren et al., "Reconfigurable manufacturing systems," *Ann. CIRP*, vol. 48, pp 1-14., 1999.

[36] K. Kumar et al., "Optimal configuration selection for reconfigurable manufacturing system using NSGA II and TOPSIS," *Int. J. Prod. Res.*, vol. 50, no. 15, pp. 4175-4191, 2012.

[37] C. Lee and G. L. Vairaktarakis, "Minimizing makespan in hybrid flowshops," *Ops. Res. Letters*, vol. 16, pp. 149-158, 1994.

[38] D. Lee and Y. Kim, "Part-mix allocation in a hybrid manufacturing system with a flexible manufacturing cell and a conventional jobshop," *Int. J. Prod. Res.*, vol. 34, no. 5, pp. 1347-1360, 1996.

[39] D. Lee and I. Ryo, "A study on scheduling by mixed dispatching rule in flexible manufacturing systems," *J. Soc. Korea Industrial Syst. Eng.*, vol. 47, pp. 35-45, 1998.

[40] D. Lee et al., "A study on integrated supply chain planning in open business environment," *Entrue J. Info. Technol.*, vol. 9, no.2, pp. 123-141, 2010.

[41] G. H. Lee, "Reconfigurability consideration design of components and manufacturing systems," *Int. J. Adv. Manuf. Tech.*, vol. 13, pp. 376-386,

1997.

[42] S. Lee et al., "Development of integrated operation technology for autonomous reconfigurable production system," in *Proc. 2010 Fall Korean Society of Machine Tool Engineers Conf.*, Korea, 2010, pp. 167-169.

[43] J. Lin and C. Lee, "A petri net-based integrated control and scheduling scheme for flexible manufacturing cells," *Comput. Integrated Manuf. Syst.*, vol. 10 pp. 143-160, 1997.

[44] J. Liu and B. L. MacCarthy, "The classification of FMS scheduling problems," *Int. J. Prod. Res.*, vol. 34, no. 3, pp. 647-656, 1996.

[45] P. B Luh et al., "Schedule generation and reconfiguration for parallel machines," *IEEE Trans. Robot. Autom.,* vol. 6, no. 6, pp. 687-696, 1990.

[46] B. L. MacCarthy and J. Liu, "A new classification scheme for flexible manufacturing systems," *Int. J. Prod. Res.*, vol. 31, no. 2, pp. 229-309, 1993.

[47] V. Maier-Speredelozzi et al., "Convertibility measures for manufacturing systems," *Ann CIRP*, vol. 52, no. 1, pp. 367–370, 2003.

[48] H. Makino and T. Trai, "New developments in assembly systems," *Ann. CIRP*, vol. 43, pp. 501-522, 1994.

[49] M. Mashaei and B. Lennartson, "Energy reduction in a pallet-constrained flow shop through on–off control of idle machines," *IEEE Trans. Automation Sci. Eng*, vol. 10, no. 1, pp. 45-56, 2013.

[50] M. Mashaei et al., "Optimal number of pallets for reconfigurable cyclic manufacturing plants," in 14th *IEEE Conf. Emerging Technologies and*

Factory Automation, Mallorca, Spain, 2009, pp. 1-8.

[51] D. T. Matt and E. Rauch, "Design of a network of scalable modular manufacturing systems to support geographically distributed production of mass customized goods," *Procedia CIRP*, vol. 12, pp. 438-443, 2013.

[52] M. G. Mehrabi et al., "Reconfigurable manufacturing systems: key to future manufacturing," *J. Intell. Manuf.*, vol. 11 pp. 403-419, 2000.

[53] M. G. Mehrabi et al., "Reconfigurable manufacturing systems and their enabling technologies," *Int. J. Manuf. Tech. Manage*, vol. 1, no. 1, pp. 114–131, 2000.

[54] M. G. Mehrabi et al., "Trends and perspectives in flexible and reconfigurable manufacturing systems," *J. Intell. Manuf.*, vol. 13, no.2, pp. 135-146, 2002.

[55] X. Meng, "Modeling of reconfigurable manufacturing systems based on colored timed object-oriented petri nets," *J. Manuf. Syst.*, vol. 29, pp. 81-90, 2010.

[56] J. Mun, K. Ryu, and M. Jung, "Self-reconfigurable software architecture: design and implementation," *Comput. Industrial Eng.*, vol. 1, pp. 163–173, 2006.

[57] H. Na et al., "A study on scheduling and rescheduling problem of the FMC during transient disturbance period with pallet constraints," in *Proc. 21st Int. Conf. Production Research*, Stuttgart, Germany, 2011.

[58] X. Nan et al., "Product scheduling and manufacturing line reconfiguration using petri nets and heuristic search," in *Proc. 2007 IEEE Int. Conf. Robotics and Biomimetics*, Sanya, China, 2007, pp.1721-1726.

[59] S. L. Narasimhan and P. M. Mangiameli, “A comparison of sequencing rules for a two-stage hybrid flow shop,” *Decision Sci.*, vol. 18, pp. 250-265, 1987.

[60] S. L. Narasimhan and S. S. Panwalkar, “Scheduling in a two-stage manufacturing process,” *Int. J. Prod. Res.*, vol. 22, no. 4, pp. 555-564, 1984.

[61] W. E. Newman et al., “Examining the use of dedicated and general purpose pallets in a dedicated flexible manufacturing system,” *Int. J. Prod. Res.*, vol. 29, no. 10, pp. 2117-2133, 1991.

[62] J. Park et al., “RMC scheduling considering setup and pallet constraint,” in *Proc. 2010 Joint Conf. Korean Institute of Industrial Engineering and Operations Research and Management Science Society*, Korea, 2010, pp. 1087-1093.

[63] J. Park and S. Woo, “The robustness of queuing network models in FMS production plans,” *J. Korea Soc. Simulation*, vol. 12, pp. 48-54, 1992.

[64] M. Penedo, *Scheduling - theory, algorithms, and systems,* 4th ed. NJ: Prentice Hall, 1995.

[65] S. Rahimifard and S. T. Newman, “The application of information systems for the design and operation of flexible machining cells,” *J. Intell. Manuf.,* vol. 10, pp. 21-27, 1999.

[66] C. Rajendran and D. Chaudhuri, “A multi-stage parallel-processor flowshop,” *Euro. J. Opl. Res.*, vol. 57, pp. 111-122, 1992.

[67] D. L. Santos et al., “Global lower bounds for flow shops with multiple processors,” *Euro. J. Opl. Res.*, vol. 80, pp. 112-120, 1995.

[68] J. Seo and J. Park, “Developing a practical machine scheduler for worker-involved systems,” in *Proc. Asia Simulation Conf.* Korea, 2011, pp. 316-325.

[69] J. Seo and J. Park, “An integrative scheduling framework for reconfigurable manufacturing cell,” in *Proc. 2012 Joint Conf. Korean Institute of Industrial Engineering and Operations Research and Management Science Society*, Korea, 2012, pp. 1661-1665.

[70] J. Seo and J. Park, “The effect of enhanced flexibility in the reconfigurable manufacturing cell,” in *Proc. 22nd Int. Conf. Production Research*, Iguassu, Brazil, 2013.

[71] J. Seo et al., “Development of capacity simulator for reconfigurable manufacturing system,” in *Proc. 2010 Spring Conf. Korean Society of Precision Eng.*, Korea, 2010, pp. 251-252.

[72] J. Seo et al., “A solution procedure for integrated supply chain planning problem in open business environment using genetic algorithm,” *Int. J. Adv. Manuf. Technol.*, vol. 62, no .9-12, pp. 1115-1133, 2012.

[73] S. P. Sethi et al., “Minimizing makespan in a pallet-constrained flowshop,” *J. Scheduling*, vol. 2, pp. 115-133, 1999.

[74] S. S. Shah et al., “Reconfigurable logic control using modular FSMs: design, verification, implementation, and integrated error handling,” in *Proc. American Control Conf.*, Anchorage, AK, 2002, vol.5, pp.4153-4158.

[75] S. Shalev-Oren et al., “Analysis of flexible manufacturing systems with priority scheduling: PMVA,” *Ann. Ops. Res.*, vol. 3, pp. 115-139, 1985.

[76] H. Shin et al., “A decision support model for the initial design of FMS,” *Comput. Industrial Eng*., vol. 33, no. 3-4, pp. 549-552, 1997

[77] A. Slocum, *Precision machine design.* NJ: Prentice Hall, 1992.

[78] P. Solot, “A heuristic method to determine the number of pallets in a flexible manufacturing system with several pallet types,” *Int. J. Flex. Manuf. Syst.*, vol. 2, pp. 191-216, 1990.

[79] P. Solot and J. M. Bastos, “MULTIQ: a queueing model for FMSs with several pallet types,” *J. Opl. Res. Soc.*, vol. 39, no. 9, pp. 811-821, 1988.

[80] P. Spicer and H. J. Carlo, “Integrating reconfiguration cost into the design of multi-period scalable reconfigurable manufacturing systems,” *J. Manuf. Sci. Eng.*, vol. 129, pp. 202-210, 2007.

[81] P. Spicer et al., “Design principles for machining system configurations,” *Ann. CIRP*, vol. 51, no. 1, pp. 275–280, 2002.

[82] C. Sriskandarajah and S.P. Sethi, “Scheduling algorithms for flexible flowshops: worst and average case performance,” *Euro. J. Opl. Res.*, vol. 43, pp. 143-160, 1989.

[83] K. E. Stecke, “Design, planning, scheduling, and control problems of flexible manufacturing systems,” *Ann. Ops. Res.*, vol. 3, pp. 3-12, 1985.

[84] K. E. Stecke and I. Kim, “A study of FMS part type selection approaches for short-term production planning,” *Int. J. Flex. Manuf. Syst.*, vol. 1, pp. 7-29, 1988.

[85] R. Suri and R. R. Hildebrant, “Modelling flexible manufacturing systems using mean-value analysis,” *J. Manuf. Syst.*, vol. 3, no. 1, pp. 27-38, 1984.

[86] L. Tang et al., “Concurrent line-balancing, equipment selection and

throughput analysis for multi-part optimal line design," *Int. J. Manuf. Sci. Prod.*, vol. 6, no. 1-2, pp. 71-81, 2004.

[87] L. Tang et al., "Selection principles on manufacturing system for part family," in *Proc. CIRP 3rd Int. Conf. Reconfigurable Manufacturing*, Ann Arbor, MI, 2005.

[88] L. Tang et al., "Computer-aided reconfiguration planning: an artificial intelligence-based approach," *Trans. ASME*, vol. 6, pp. 230-240, 2006.

[89] U. A. W. Tetzlaff, "A model for the minimum cost configuration problem in flexible manufacturing systems," *Int. J. Flex. Manuf. Sys.*, vol. 7, pp. 127-146,1995.

[90] D. M. Tilbury and S. Kota, "Integrated machine and control design for reconfigurable machine tools," in *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics*, Atlanta, GA, 1999, pp. 629-634.

[91] W. Wang and Y. Koren, "Scalability planning for reconfigurable manufacturing systems," *J. Manuf. Syst.*, vol. 31, pp. 83-91, 2012.

[92] L. A. Wolsey, *Integer Programming.* New York, NY*:* John Wiley and Sons, 1998.

[93] S. Woo et al., "An integrated approach for loading and scheduling of a flexible manufacturing system," *J. Korean Inst. Industrial Engineers*, vol. 25, no. 3, pp. 298-309, 1999.

[94] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system part1: a framework," *Int. J. Prod. Res.*, vol. 38, no. 10, pp. 2273-2285, 2000.

[95] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing

system part 2: optimal configurations," *Int. J. Prod. Res.*, vol. 38, no. 12, pp. 2829-2842, 2000.

[96] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system part 3: optimal selection policy," *Int. J. Prod. Res.*, vol. 39, no. 4, pp. 747-758, 2001.

[97] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system - part 4: performance measure," *Int. J. Prod. Res.*, vol. 39, no. 6, pp. 1113-1126, 2001.

[98] Y. Yamada et al., "Layout optimization of manufacturing cells and allocation optimization of transport robots in reconfigurable manufacturing systems using particle swarm optimization," in *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, 2003, vol. 2, pp. 2049-2054.

[99] S. Yang et al., "Modeling and analysis of multi-stage transfer lines with unreliable machines and finite buffers," *Ann. Oper. Res*, vol. 93, pp. 405-421, 2000.

[100] H. Ye and M. Liang, "Simultaneous modular product scheduling and manufacturing cell reconfiguration using a genetic algorithm," *Trans. ASME*, vol. 128, pp. 984-995, 2006.

[101] A. S. Yigit et al., "Optimizing modular product design for reconfigurable manufacturing," *J. Intell. Manuf.* vol. 13, no. 4. pp. 309-16, 2002.

[102] A. M. A. Youssef, "Optimal configuration selection for reconfigurable manufacturing systems," Ph. D. dissertation, Univ. of Windsor, Canada, 2006.

[103] A. M. A. Youssef and H. A. ElMaraghy, “Modelling and optimization of multiple-aspect RMS configurations,” *Int. J. Prod. Res.*, vol. 44, no. 22, pp. 4929-4958, 2006.

[104] A. M. A. Youssef and H. A. ElMaraghy, “Assessment of manufacturing systems reconfiguration smoothness,” *Int. J. Adv. Manuf. Technol.*, vol. 30, pp. 174-193, 2006.

[105] A. M. A. Youssef and H. A. ElMaraghy, “Availability assessment of multi-state manufacturing systems using universal generating function,” *Ann. CIRP*, vol. 55, no. 1, pp. 445–448, 2006.

[106] A. M. A. Youssef and H. A. ElMaraghy, “Optimal configuration selection for reconfigurable manufacturing systems,” *Int. J. Flex. Manuf. Syst.*, vol. 19, pp. 67-106, 2007.

[107] A. M. A. Youssef and H. A. ElMaraghy, “Availability consideration in the optimal selection of multiple-aspect RMS configurations,” *Int. J. Prod. Res.*, vol. 46, no. 21, pp. 5849-5882, 2008.

[108] J. Yu, et al., “Iterative algorithms for part grouping and loading in cellular reconfigurable manufacturing system,” *J. Opl. Res. Soc.*, vol. 63, pp. 1635-1644, 2012.

[109] B. Zheng et al., “Configuration optimization of manufacturing system based resource reconfiguration,” in *Proc. 2nd Int. Conf. Mechanical and Electronics Engineering*, London, UK, 2010, vol. 1, pp. 193-195.

[110] W Zhong et al., “Performance analysis of machining systems with different configurations,” in *Proc. Japan-USA Symp. Flex. Manuf.* Ann Arbor, MI, 2000, pp. 1-8.

초 록

재구성가능 생산셀(Reconfigurable Manufacturing Cell, RMC)은 시장 수요의 변화에 대응하기 위해 하드웨어 구성을 빠르게 변화시킬 수 있는 생산시스템이다. 현재 이러한 재구성에 걸리는 시간은 1-2일 정도이다. RMC의 운영에 있어 관리자는 제조시스템의 재구성 능력을 어떻게 활용할 것인지에 관련하여 두 가지 의사결정문제에 직면하게 된다.

첫 번째는 하드웨어 구성을 변경할 것인지 또 변경한다면 어떻게 변경할 것인지에 관한 문제이다. 하드웨어 구성을 변경하면 수요에 대한 생산 시스템의 성능이 향상될 수 있으나, 의사결정의 신뢰도를 높이기 위해서는 상세한 자원 할당과 재구성 기간 동안 발생하는 생산 능력의 손실에 대한 정확한 평가가 이루어져야 한다. 그러나 하드웨어 재구성에 관한 기존 연구들은 재구성 문제를 운영적 문제가 아닌 전략적 문제로 다루었고 제조시스템의 정확한 성능 평가를 제공하지 못하였다.

두 번째 문제는 장비 구매에 관한 것이다. 관리자는 현재 시스템이 미래의 수요를 충족시킬 수 있을 것인지, 아니면 추가적인 장비 구매가 필요한지 알기 원한다. 이 때 수요를 충족시키면서 구매

비용을 최소화하기 위해서는 제조시스템의 재구성 능력이 함께 고려되어야 한다.

본 연구는 이상의 두 가지 문제를 위한 통합적인 의사결정시스템을 개발하고자 한다. 하드웨어 재구성 문제를 스케줄링 수준에서 해결하기 위해, RMC 환경에 알맞은 스케줄링 알고리즘을 개발한다. 그리고 개발한 스케줄링 알고리즘을 이용하고 재구성 기간 동안의 생산 능력의 손실을 고려하여, 하드웨어 재구성을 위한 수리 모델을 제시한다. 제시한 모델을 풀기 위하여, 완화에 기초한 분기한정 알고리즘을 제안한다. 마지막으로 장비 구매 문제를 해결하기 위하여 구매 비용의 현재가치를 최소화하는 알고리즘을 제안한다.

주요어: Reconfigurable Manufacturing Cell, Scheduling, Hardware Reconfiguration, Algorithm
학 번: 2008-21225

공학박사 학위논문

재구성가능 생산셀의 통합 의사결정시스템에 관한 연구

A Study on Integrative Decision-making System

for Reconfigurable Manufacturing Cells

2014년 2월

서울대학교 대학원

산업공학과

서진우

재구성가능 생산셀의 통합 의사결정시스템에 관한 연구

지도교수 박 진 우

이 논문을 공학박사 학위논문으로 제출함
2013년 11월

서울대학교 대학원
산업공학과
서 진 우

서진우의 공학박사 학위논문을 인준함
2013년 12월

위 원 장 ______________________ (인)

부위원장 ______________________ (인)

위　　원 ______________________ (인)

위　　원 ______________________ (인)

위　　원 ______________________ (인)

Abstract

A Study on Integrative Decision-making System for Reconfigurable Manufacturing Cells

Jinwu Seo

Department of Industrial Engineering

The Graduate School

Seoul National University

RMCs (Reconfigurable Manufacturing Cells) are production systems which can rapidly change hardware configurations to respond to market variance. The practical range of such change time is one or two days. In the operation of RMCs, managers encounter two decision problems with regard to how reconfiguring capability of manufacturing systems would be utilized.

The first problem is to decide whether and how hardware configurations of manufacturing systems are to be changed. An alternative configuration may improve manufacturing performance in response to demand, but detailed resource allocation needs to be examined together with capacity loss during reconfiguration for accurate evaluation so that managers may make decisions with certainty. However, most of previous studies on hardware reconfiguration dealt it as a tactical problem rather than an operational problem, not providing accurate performance evaluation of manufacturing systems.

The second problem is about purchase of equipment. Managers want to know whether their current systems may catch up with future demand or additional equipment needs to be purchased. In this case, reconfiguring capability of manufacturing systems should be considered to satisfy demand with minimized purchasing cost.

This study aims to develop an integrative decision-making system for those decision problems. To provide scheduling-level resolution of hardware reconfiguration, a scheduling algorithm appropriate for RMC environment is developed. And using the developed scheduling algorithm, a mathematical model for hardware reconfiguration is presented in consideration with capacity loss during reconfiguration. To solve the model, a relaxation based branch-and-bound algorithm is proposed. Finally, to resolve the equipment purchase problem, an algorithm which minimizes the net-present-value of purchasing cost is suggested.

Keywords: Reconfigurable Manufacturing Cell, Scheduling, Hardware Reconfiguration, Algorithm
Student Number: 2008-21225

Contents

영문초록 i

1. Introduction 1

1.1. Environment of Research 1

1.2. Purpose of Research 5

2. Related Work 7

2.1. Previous Study on Hardware Reconfiguration........................9

2.2. Previous Study on FMS Scheduling 14

2.3. Summary of Related Work 17

3. Problem Definition: Managerial Aspect.......................19

3.1. Description of Production Environment 19

3.2. Definition of Managerial Problems.......................................22

4. Scheduling Algorithm for RMC....................................26

4.1. Scheduling Environment..28

4.2. Mathematical Model 31

4.3. Development of Algorithm 35

4.3.1. An Example for the Algorithm.. 42

4.4. Experiment..47

4.5. Result and Analysis...49

5. A Framework for Hardware Reconfiguration.............57

5.1. Mathematical Model 59

5.2. Solution Procedure for Hardware Reconfiguration: Branch-and-bound Approach...64

5.2.1. Relaxation of the Problem 64

5.2.2. Relaxation based Branch-and-bound Algorithm 74
5.3. Experiment 84
5.4. Result and Analysis 86
6. RMC Equipment Purchase Planning 93
6.1. Mathematical Model 95
6.2. Development of Algorithm 99
6.3. Experiment 102
6.4. Result and Analysis 103
7. Conclusion 111
Appendix 114
References 115
국문초록 **128**

List of Tables

<Table 3.1> The summary of managerial problems and research direction.....24

<Table 4.1> Dispatching information for the example of the proposed algorithm45

<Table 4.2> Dispatching rules used in experiment as alternatives .48

<Table 4.3> The experimental result for the performance measure51

<Table 4.4> The experimental result for computational time 53

<Table 5.1> The experimental result 87

<Table 5.2> The result of hardware reconfiguration problem 90

<Table 6.1> An example of the purchasing alternative list............101

<Table 6.2> The experimental result 104

<Table 6.3> The result of reconfigurable equipment purchase problem 106

<Table 6.4> Cost of equipment purchase 109

<Table 6.5> Benefit of equipment purchase: Reduced production delay110

List of Figures

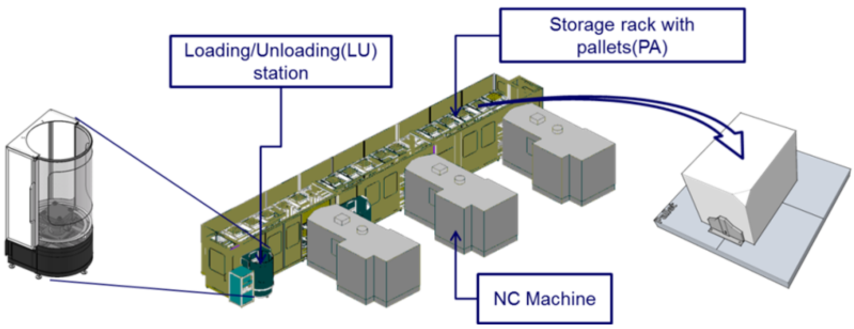
<Figure 1.1> A typical RMC..1
<Figure 1.2> The concept of reconfiguring 2
<Figure 1.3> The reconfiguring problem 3
<Figure 1.4> Manufacturing performance of a configuration and RMC schedules .3
<Figure 3.1> Production environment: A typical layout of a factory21
<Figure 3.2> The relationship among the managerial problems 25
<Figure 4.1> The relationship between the scheduling problem and the reconfiguring problem 27
<Figure 4.2> An example of the pallet-fixture assignment 29
<Figure 4.3> Impact of resource idleness 37
<Figure 4.4> The work-in-process projection...................................38
<Figure 4.5> The example setting 43
<Figure 4.6> The experimental result for the performance measure55
<Figure 4.7> The experimental result for the performance measure55
<Figure 4.8> The experimental result for computational time 56
<Figure 5.1> The case when hardware reconfiguration is considered.....57
<Figure 5.2> The capacity change around hardware reconfiguration58
<Figure 5.3> Reconfiguring time according to change of cell configurations....60
<Figure 5.4> The flow chart of RPX algorithm................................74
<Figure 5.5> The flow chart of the branch-and-bound algorithm..75
<Figure 5.6> Pseudo code of the neighborhood search algorithm ..82
<Figure 5.7> The result of hardware reconfiguration problem in graphical representation 91
<Figure 5.8> The result of hardware reconfiguration problem in change of schedules92

<Figure 6.1> The flow chart of REPP algorithm............................100
<Figure 6.2> The result of reconfigurable equipment purchase problem in graphical representation..108
<Figure A.0.1> The optimal schedules: the non-permutation schedule (a) and the permutation schedule (b) 114

1. Introduction

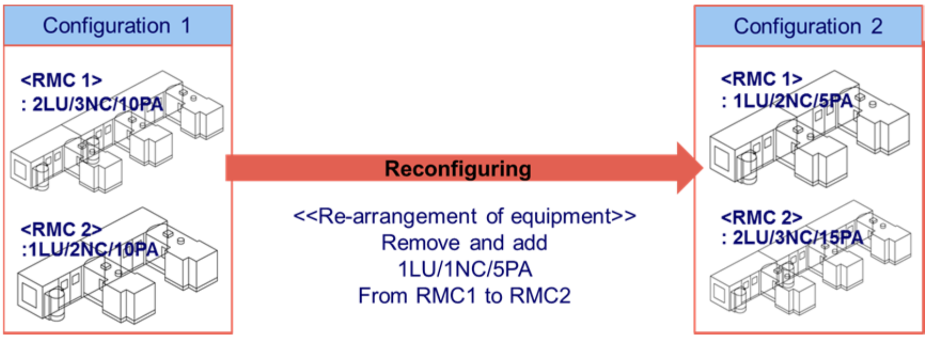
1.1. Environment of Research

RMCs (**Reconfigurable Manufacturing Cells**) are production systems which can rapidly change hardware configurations to respond to market variance. An RMC consists of loading/unloading (L/U) stations, NC machines (e.g. horizontal machining centers), material handling devices (e.g. AGVs), and storage rack of pallets. <Figure 1.1> shows a typical RMC in development.



<Figure 1.1> A typical RMC

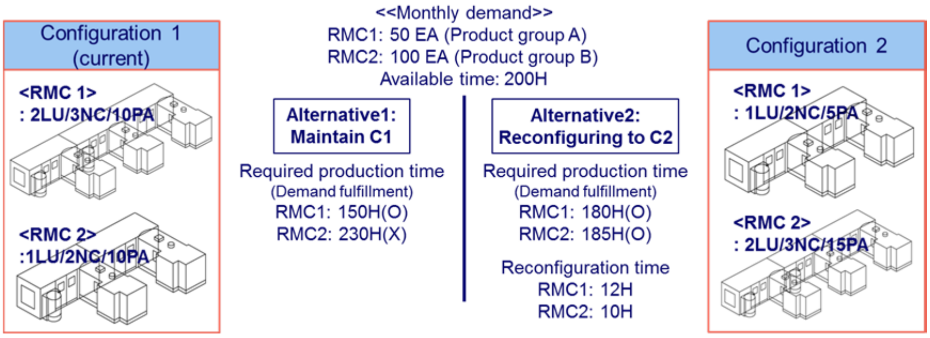
Reconfiguring is re-arrangement of equipment among RMCs, to adjust production capacity among product groups. Reconfigurable equipment is L/U stations, NC machines, and pallets. <Figure 1.2> indicates the concept of reconfiguring. Note that the total number of equipment of the whole factory does not change.



<Figure 1.2> The concept of reconfiguring

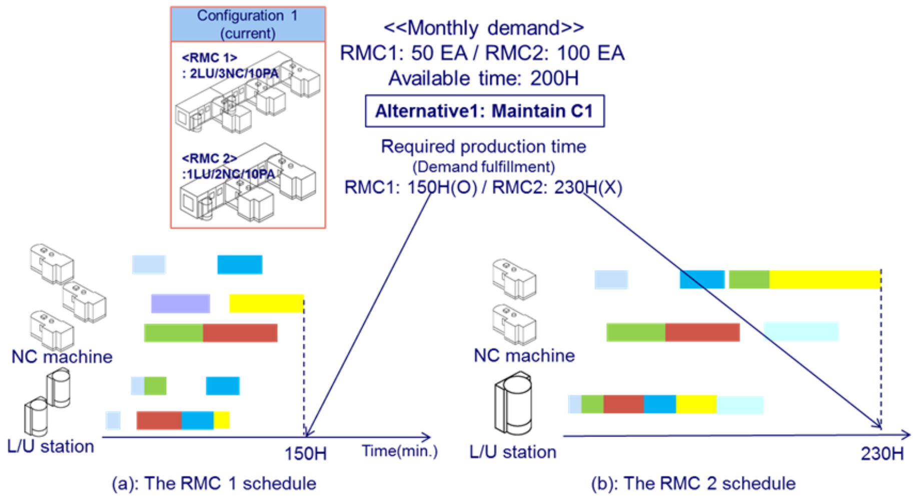
Since market demand is ever-changing, the current configuration may not be appropriate to respond to demand. In this case, production managers are concerned with performance improvement of their manufacturing systems through hardware reconfiguration. <Figure 1.3> indicates different manufacturing performance according to hardware configurations. In <Figure 1.3>, ‘Configuration 2’(C2) performs better than ‘Configuration 1’(C1), since with C2 both cells may finish production within the available time, while production delay may happen in RMC 1 with C1. Although it takes time to change the hardware configuration from C1 to C2, as in <Figure 1.3>, C2 may satisfy demand on time.

Since various alternative configurations may exist, production managers wonder which hardware configuration would perform best. The **Reconfiguring problem** is to find the best hardware configuration of RMCs in response to demand.



<Figure 1.3> The reconfiguring problem

To evaluate manufacturing performance of a configuration, scheduling may be required. The required production time of each RMC in <Figure 1.3> was obtained by generating schedules given a hardware configuration, as shown in <Figure 1.4>.



<Figure 1.4> Manufacturing performance of a configuration and RMC schedules

Scheduling-level evaluation of hardware configurations provides reliable information for decision-making. The schedule specifies detailed resource allocation over time and close to actual production, so it would be reliable to compare configuration alternatives based on their schedules. With scheduling-level evaluation, therefore, managers may make decisions with certainty.

Scheduling-level evaluation becomes more practical with the advent of RMCs. When the concept of RMS (reconfigurable manufacturing system) was introduced in late 1990s, it took one month or more to change hardware configurations. So reconfiguration decisions have been considered as long-term (over a year) decisions [18][52][80][103]. In order to respond to ever-changing demand more quickly, development of manufacturing systems with small changeover cost was requested. In response, a Korean governmental research project for the development of RMC (reconfigurable manufacturing cell) was launched in 2009 [42]. In the manufacturing system with RMCs, reconfiguration happens not throughout the entire factory but within RMCs, and components consisting of RMCs are highly modularized. As a result, time required for hardware reconfiguration has been reduced to few days. (The final objective of changeover time in the research project is one day.) As short-term demand forecast is relatively accurate and difference from actual production is relatively small, scheduling-level evaluation of RMCs would be meaningful for reliable decision-making.

1.2. Purpose of Research

This study aims to develop an integrative decision-making system for RMC environment. The system provides following functions:

- Scheduling-level resolution of the RMC reconfiguring problem: Here, manufacturing performance in response to market demand is considered as the primary objective, and capacity loss during reconfiguration is examined. Then reconfiguring cost is considered as the secondary objective.

- RMC equipment purchase planning: In long-term aspect, it may be required to purchase additional equipment. Considering reconfiguring capability of manufacturing systems, the best purchase plan, with which demand may be fulfilled in minimized purchasing cost, is considered. This is an extension of the reconfiguring problem.

This thesis is organized as follows. Chapter 2 reviews previous research on the research problem. In section 3.1, production environment is explained and in section 3.2, the research problem is redefined in managerial aspect. In chapter 4, a scheduling algorithm appropriate for hardware reconfiguration is developed as a basic component of the overall problem-solving process. Then in chapter 5, using the developed scheduling algorithm, a mathematical model for the reconfiguring problem is presented and a relaxation based branch-and-bound algorithm is proposed to solve that model. In chapter 6, an algorithm for minimizing net-present-value of purchasing cost is suggested for equipment purchase planning. Finally it ends up with conclusion in chapter 7.

2. Related Work

The concept and design of manufacturing systems with re-configurability was investigated by early researchers from mid 1990s. Adlemo et al. [4] classified flexibility of manufacturing systems into long time-scale, shorter time-scale and very short time-scale flexibility according to the degree and lasting duration of system change, and presented the corresponding software structure and a product model to support introduction of new equipment and products to existing systems. Koren et al. [35] (refer also [33]), Mehrabi et al. [52] and ElMaraghy [18] suggested the concept of reconfigurable manufacturing system(RMS)s, and each proposed architecture to design and operate those systems. Makino and Trai [48] categorized RMSs as static and dynamic according to whether resources are just used like building blocks or advanced material handling systems support production.

Abdi and Labib [1] presented an AHP model for a design strategy of RMSs, where various planning horizons and actors were considered together with manufacturing objectives and criteria, and the authors extended the subject to grouping products into product families based on operation similarities in [2], and to selecting hardware configurations in [3]. Mehrabi et al. [53] investigated on the development direction of RMSs in aspects of system design and hardware/software architecture. Mehrabi et al. [54] conducted a survey to understand needs and expectations in development of RMSs.

Tilbury and Kota [90] investigated on design of reconfigurable machine tools to manufacture a given part family together with associated control modules. Matt and Rauch [51] presented functional requirements and design parameters for scalable modular manufacturing systems to support geographically distributed production of mass customized goods.

This study is concerned with hardware reconfiguration of manufacturing systems. Previous studies on the reconfiguring problem are reviewed in the next section. Studies that approach the problem in the scheduling level are discussed separately. Then studies on pallet adjustment follow. Since production in RMCs has similarity with that in FMSs and flowshops, relevant studies on those topics are also discussed.

2.1. Previous Study on Hardware Reconfiguration

There have been studies that were concerned with optimization and/or selection of hardware configurations of manufacturing systems.

Koren et al. [34] considered system level reconfiguring - changing the number of production stages and the number of machines for each stage - and analyzed the effect of different configurations on manufacturing performance in terms of expected productivity, initial setup cost, and scalability cost. Koren and Shpitalni [31] further investigated system level reconfiguring, focusing on eliminating impractical or unsatisfactory configurations from all possible alternatives. Wang and Koren [91] proposed scalability planning problems of RMSs to minimize the number of new machines required, and suggested a solution procedure using the genetic algorithm. Tang et al. [88] presented the computer-aided reconfiguration plan framework, where the reconfiguration plan problem was modeled as a network of potential reconfiguring activities and both system and machine level reconfiguring was considered. Here machine level reconfiguring indicates adjusting additional capability of equipment (by adding/removing spindles) and setting fixtures. (Refer also [32].) Using the result of [86][99], in [87], selection principles of manufacturing systems was presented based on the cost model and the utilization rate. Youssef and ElMaraghy [103] proposed automatic generation methods of system and machine level hardware reconfiguration considering operation clusters setups and hardware space limitation. In [104], the reconfiguration smoothness metric considering anticipated reconfiguration process was defined in market, system and machine perspectives. And combining the result of [103][104], in [106], RMC configuration selection approach was developed using meta-heuristic algorithms. (Refer also [102].) Spicer and Carlo [80] investigated on the reconfiguring cost structure, and suggested a dynamic programming based algorithm to determine the cost-minimizing hardware reconfiguration path for multi periods. Xiaobo et al. [94] considered manufacturing systems which change hardware configurations according to product groups to be manufactured and proposed a reconfiguration framework based on stochastic arrival rates and processing times and investigated it further in [95][96][97]. In [107], selection of the optimal configuration was dealt with machine availability using the universal generating function (UGF), which was extended from [105]. Huang et al. [27] investigated an optimization model for configuration selection using the characteristic state equation. Bensmaine et al. [9] dealt with machine level reconfiguration considering tool approach directions of machines and multi objectives using the non-dominated sorting genetic algorithm, which Kumar et al. also employed for optimal configuration selection in [36]. Asl and Ulsoy [5] and Deif and EIMaraghy [14] applied approaches of feedback control for hardware reconfiguration. In [81], the effect of reconfiguration on throughput was investigated with consideration of machine availability. Meng [55] presented the petri-net model for RMSs which is divided into system level, process level, and cells level.

Studies that provide scheduling-level resolution of hardware reconfiguration are relatively few.

Ye and Liang [100] proposed an integrated model for scheduling of modular products and reconfiguring of manufacturing cells and suggested a genetic algorithm which determines cell configurations and job sequences simultaneously, but it did not consider capacity scalability. Nan et al. [58] and Zheng et al. [109] used the timed petri-net to generate the reconfiguration plan in combination with the corresponding schedule, considering reconfiguring time. The petri-net model, however, has limitation that its complexity dramatically increases as does the problem size. The suggested search technique based on the reachability graph was also appropriate for the small size problems. Seo and Park [70] and Seo et al. [71] investigated enhanced flexibility of RMCs compared to conventional FMCs using the constraint programming and sought to make integrative decisions for the reconfiguring problem in the scheduling level, but a systemic procedure to find good solutions were missing.

Other studies relevant to hardware reconfiguration are as follows.

Zhong et al. [110] considered quality in performance analysis of different system configurations based on the homogeneous transformation matrix [77]. Maier-Speredelozzi et al. [47] presented measures for convertibility of manufacturing systems, which is capability to adjust capacity and/or functionality of systems. Mun el. al [56] proposed software architecture for RMSs to facilitate a dynamic reconfiguration of system elements (i.e. fractals). Lee [41] presented design rules to minimize material handling cost where time required for reconfiguration was considered. Hasan et al. [26] investigated on an optimum sequence of a part family considering reconfiguring cost, and stochastic demand. Yigit et al. [101] was concerned with the optimum selection of module instances for products manufactured in RMSs considering the trade-off between quality and reconfiguring cost. Yamada et al. [98] investigated into optimization of the layout and allocation of transport robots and suggested meta-heuristic algorithms based on particle swarm optimization. Shah et al. [74] considered reconfiguration of the logic controllers for a small-scale manufacturing line.

Pallets are often critical resources in flexible manufacturing systems. There were studies on pallet adjustment.

Rahimifard and Newman [65] considered simultaneous scheduling of workpieces, fixtures and cutting tools and Seo and Park [68] discussed about how to improve the RMC scheduler's practicality and presented a corresponding database structure, yet both of them were conceptual. Shalev-Oren and Schweitzer [75] extended [85] and presented an analytic model based on the mean value analysis to measure performance of FMSs according to priority rules considering pallets and fixtures and material handling time. The result was compared with the simulation model and appeared to have difference of around 4-13% for small size problems. Stecke and Kim [84], Solot and Bostos [79], and Solot [78] investigated into the optimal number of pallets for several part types using the queuing network. Mashaei et al. [50] suggested an analytic mathematical model and an IP model to determine the minimum number of pallets attaining the minimum cycle time, given a cyclic schedule (sequence) and derived inequalities about the minimum number and verified the result with the simulation optimization result, yet the fixtures were not considered. Denzler et al. [16] was concerned with impact of distribution of dedicated pallets on scheduling performance. Newman et al. [61] compared dedicated and general (shared) pallets considering the trade-off between the scrap rate and the utilization efficiency. Han et al. [21] dealt the case where the distribution of dedicated pallets is decidable and tried to improve manufacturing performance in an integrative manner.

2.2. Previous Study on FMS Scheduling

Given hardware configurations, the RMC scheduling problems have similar characteristics with the FMS scheduling problems. Below are researches about the FMS scheduling problems.

Sethi et al. [73] verified performance of the Gilmore-Gomory's algorithm in [20] for pallet-constrained flow shops. Yu et al. [108] dealt with scheduling problems of reconfigurable manufacturing cell, divided decisions into input sequencing, operation/machine selection, and part sequencing. It evaluated performance of various combinations of dispatching-based rules. Park et al. [62] considered the FMC scheduling problem under pallet constraints and presented a heuristic algorithm which reduces the setup delay. Na et al. [57] investigated the FMC scheduling and rescheduling problems during the transient disturbance period with pallet constraints. Mashaei and Lennartson [49] also dealt the pallet-constrained flowshops with purpose of reduction of energy consumption.

Above studies reflected pallet requirement for generation of schedules. The below studies did not consider pallets as scarce resources.

Park and Woo [63] examined robustness of the queuing network model in the FMS scheduling problem. Jang et al. [29] suggested an integrated decision support system for FMSs and proposed a periodic scheduling algorithm. Lee and Ryo [39] presented an algorithm responsive to shop situations applying different dispatching rules to machines. Woo et al. [93] solved the FMS loading and the FMS scheduling problems in an integrative manner. Chung et al. [11] developed a mixed integer programming model for the FMS scheduling problem in the control level and suggested control software architecture. Gultekin et al. [22] presented a cyclic scheduling algorithm for the FMC scheduling problem where the material handling time of a robot is critical. Lin and Lee [43] proposed a petri-net based scheduling scheme for FMCs to integrate scheduling and control.

Flexible manufacturing systems like RMCs are often modeled as flowshops in which parts are processed along machines in sequence.

Santos et al. [67] presented a lower bound scheme for flexible flowshops, meaningful for flowshops beyond three stages. Rajendran and Chaudhuri [66] suggested a branch-and-bound algorithm for flexible flowshops forcing permutation schedules. Azizoglu et al. [6] suggested a branch-and-bound algorithm for flexible flowshops not forcing permutation schedules. Guinet et al. [21] proved the NP-hardness of scheduling problems of two-stage flexible flowshops and suggested the 'sequence-first and allocate-second' heuristic, of which performance is within 0.73% gap to the three, presented lower bounds. Lee and Vairaktarakis [37] presented the $2-\frac{1}{m}$ approximation algorithm for hybrid flowshops and showed performance numerically that the gap is within the average of 1.6% and the maximum of 3%. Sriskandarajah and Sethi [82] presented an analytic model for flexible flowshops and proved that the list scheduling algorithm is the $3-\frac{1}{m}$ approximation algorithm and suggested a heuristic algorithm. However it has limitation that the considered case is too specific. Narasimhan and Mangiameli [59] investigated impact of job allocations on the future status of machine idleness and waiting time in two-stage hybrid flowshops and suggested the generalized cumulative minimum deviation rule(GCMD) assuming permutation schedules, which is an extension of Narastmhan and Panwalkar [60]'s CMD rule which is for two-stage hybrid flowshops with a single machine on the first stage.

2.3. Summary of Related Work

Previous research considered the reconfiguring problem as a tactical problem for long-term demand rather than an operation problem for short to mid-term demand as Deif and EIMaraghy [14] pointed out that “the rate of variation in demand is usually much higher than the rate at which capacity can be changed and reconfiguration decision is tactical rather than operational”. Capacity loss during reconfiguration was rarely considered, and systems were roughly modeled using the queuing network and/or the simple calculation to estimate manufacturing performance of hardware configurations. Improvement in reconfiguring cost and reconfiguring time with the advent of RMCs has not been reflected yet.

For researches about FMS scheduling problems, there has been no research that re-configurability is concerned in the development of scheduling algorithms. Such a scheduling algorithm is hardly found that explicitly considers environment where both hardware configurations and demand are ever-changing.

Our study deals with manufacturing environment where hardware reconfiguration may happen frequently for short-term demand and therefore scheduling-level evaluation is required for reliable decision making. In the development of the scheduling algorithm for such environment, it should be considered that both hardware configurations and demand are variable.

In the next chapter, production environment is explained and the reconfiguring problem is redefined in managerial aspect.

3. Problem Definition: Managerial Aspect

In this chapter, the reconfiguring problem is redefined in the view of production managers. At first, production environment is described.

3.1. Description of Production Environment

Description of production environment is as follows:

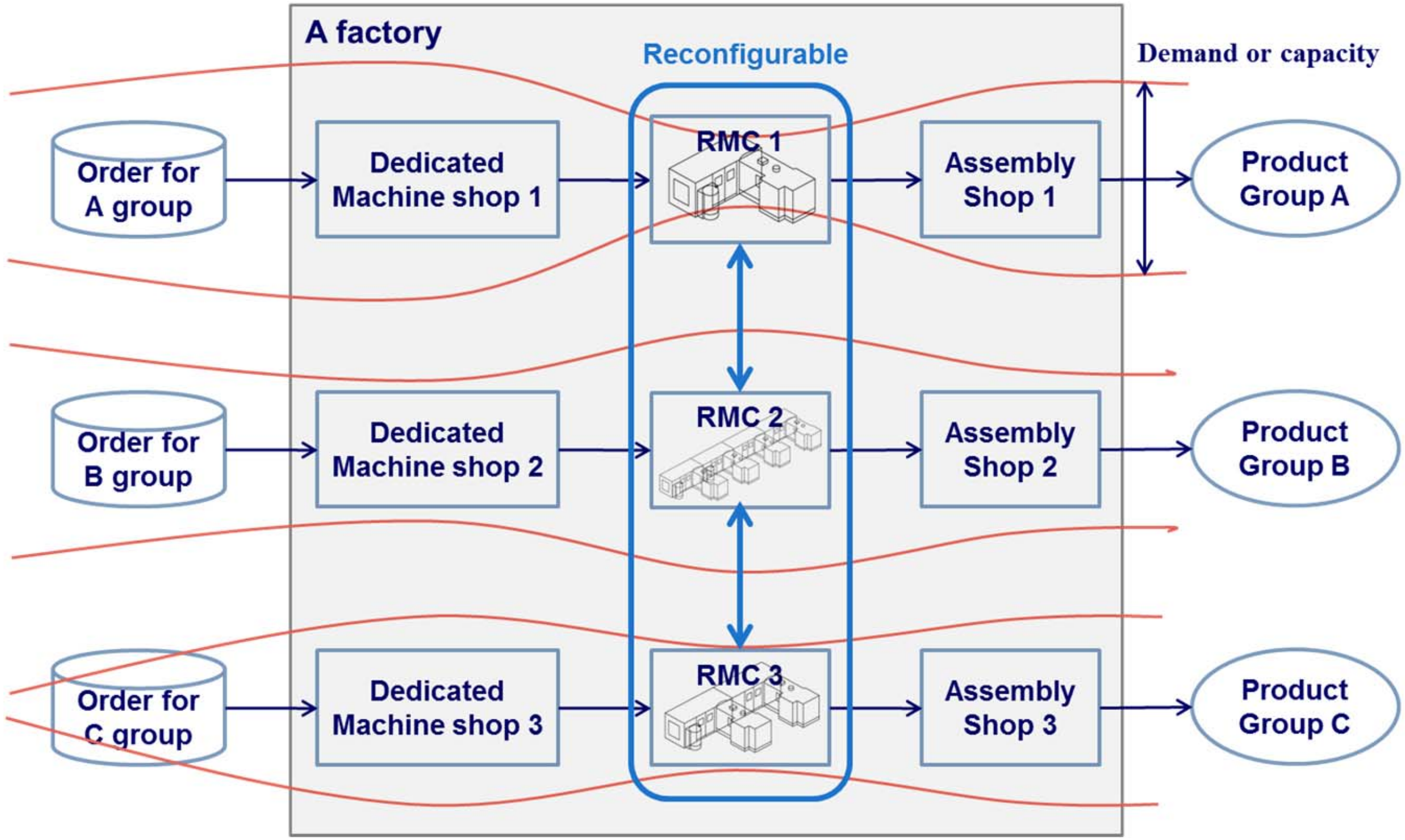
There exist multiple cells of RMCs in the factory, each for a production line and a production line is made up of dedicated and assembly shops together with an RMC. <Figure 3.1> shows a typical layout of a factory in consideration. The exemplary factory is divided into 3 production lines according to product groups, and each line consists of 3 stages including a single RMC. Though RMCs are available for a variety of machining operations, the factory may still maintain the traditional system of dedicated machines for various reasons such as high productivity and cost reduction. According to Koren and Shpitalni [31], it’s better to employ the dedicated machine shop for repetitive and common operations. So it is more realistic that RMCs are used as a part of the production line. This shop environment is sometimes called a cellular reconfigurable manufacturing system (RMS) [108], or may be named as a multi-cell RMS according to [46].

An RMC consists of L/U stations, NC machines and pallets and those components are identical among the same types of components. Therefore an

RMC is defined by the number of L/U stations, NC machines and pallets. And each RMC is given a production order for a pre-defined demand period (e.g. monthly demand), which is defined by product types and volumes.

To respond demand, RMCs are reconfigurable: L/U stations and NC machines are interchangeable among RMCs in the factory. Pallets which support operations are also deliverable from one RMC to another. Reconfiguration includes removing or attaching of equipment, tool changes, NC program uploads, precision tests and adjustment, and all the other activities required to ramp up the manufacturing system with the new hardware configuration. Hardware reconfiguration is conducted before production begins. The other parts of the manufacturing system except RMCs are assumed to have fixed capacity.

Though it may be possible to redistribute work among RMCs, transferring work or raw material from one to another production line usually brings inefficiency and sometimes causes errors, because of physical constraints such as the factory layout. Therefore work redistribution among RMCs is not allowed. In this context the part-grouping problem [19][24][84] is not concerned and related decisions are assumed given.



<Figure 3.1> Production environment: A typical layout of a factory

3.2. Definition of Managerial Problems

Provided with the production environment, production managers in charge of operating RMCs may incur following decision problems.

The first one is the **configuration evaluation problem** (CEP), which is to check manufacturing performance of the current configuration. Given production demand, managers may want know whether the factory may satisfy demand with the current configuration or not. The result of CEP is manufacturing performance (e.g. demand fulfillment) of each RMC and corresponding schedules. Based on CEP result, managers may decide whether to maintain or change the current configuration. In addition, they may necessitate information about how much they have excess or shortage of capacity to make further decisions.

The second problem is the **reconfiguring problem** (RP) which is to find the best alternative configuration to improve manufacturing performance. The reconfiguring problem needs to be solved when it is verified through CEP that the current configuration may not meet demand, and there is excess capacity in some RMCs while other RMCs suffer from lack of capacity. The result of RP is the alternative configuration together with improved manufacturing performance and reconfiguring cost at that time. Based on this, managers may decide how to change the existing configuration.

Above two problems are related to response to short-term demand. Based on long-term demand forecast, managers may wonder if their current system is sufficient for production of future periods. If managers are not satisfied with the current system, they should find a way to enhance manufacturing performance and this may be done by purchasing additional equipment.

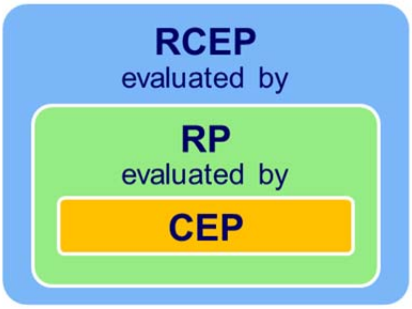
The **reconfigurable equipment phase problem** (REPP) is to find the best purchase plan for long-term demand. The result of REPP is the purchase plan that specifies which and how many equipment would be bought and when, together with the purchasing cost and manufacturing performance under that plan. Based on the result, managers may release purchase orders or make a budget for purchase of equipment. REPP is distinguished from the conventional equipment purchase problem in that reconfiguration of manufacturing systems is considered.

<Table 3.1> summarizes the managerial problems described above. <Table 3.1> also includes the research direction to support the managerial problems, together with relevant chapters.

<Table 3.1> The summary of managerial problems and research direction

	What production managers want to			***Direction of development to support managers***
	Know	See (metrics)	Decide	
Configuration Evaluation Problem	Manufacturing performance of the current system	Demand fulfillment	Whether to maintain or change the current system	Scheduling algorithm (Heuristic) → Chapter 4
Reconfiguring Problem	Costs and benefits of hardware reconfiguration	Demand fulfillment, Reconfiguring cost	How to change the current system	Configuration search algorithm (Relaxation based B&B) → Chapter 5
Reconfigurable Equipment Purchase Problem	Costs and benefits of equipment purchase	Demand fulfillment (long term) Purchasing cost	Whether/Which/ How many/When to purchase equipment	Purchase plan search algorithm (Exact) → Chapter 6

<Figure 3.2> shows the relationship among the managerial problems defined in the previous section. The reconfiguring problem relies on the result of the configuration evaluation problem, since manufacturing performance of every investigated configuration in the reconfiguring problem needs to be evaluated for comparison. On the other hand the reconfiguring problem may be a building block for the reconfigurable equipment purchase problem, for hardware reconfiguration should be considered to evaluate the effect of purchased equipment on manufacturing performance.



<Figure 3.2> The relationship among the managerial problems

As shown in <Figure 3.2>, the configuration evaluation problem is the most basic problem. Development of a scheduling algorithm for hardware reconfiguration is therefore very important as the configuration evaluation problem is solved by generating schedules. The scheduling algorithm may be considered as a basic component of the decision-making system this study aims to develop.

In this context, the scheduling algorithm is introduced in the next chapter.

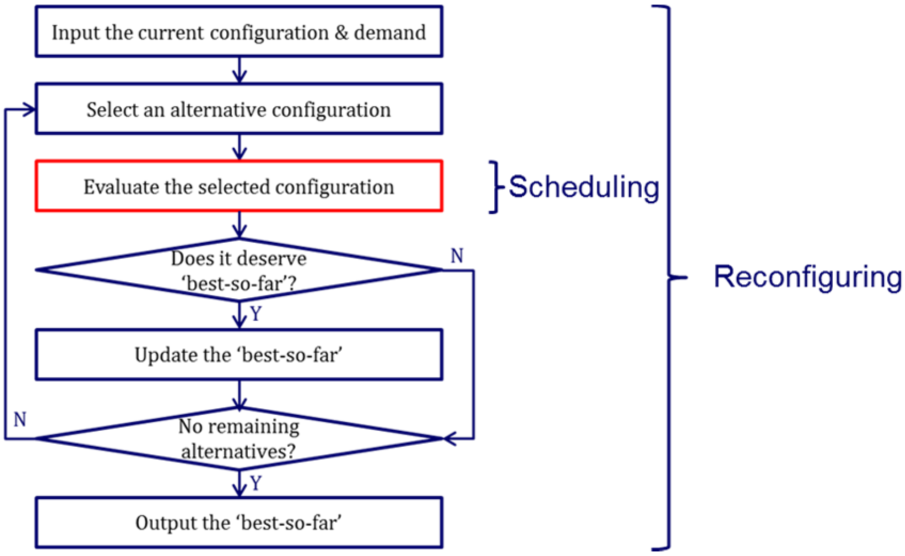
4. Scheduling Algorithm for RMC

This chapter is concerned with development of a scheduling algorithm to resolve the configuration evaluation problem (CEP). In CEP, manufacturing performance of the current system is evaluated. Therefore throughout this chapter, it is assumed the hardware configuration is given. Development of a scheduling algorithm appropriate for RMCs would provide managers with scheduling-level evaluation of manufacturing performance for decision making.

In RMC environment, the hardware configuration may be changing frequently. Therefore the scheduling algorithm for RMCs should be able to perform consistently for a variety of hardware configurations.

On the other hand, the schedules need to be generated as many as the number of RMCs in the manufacturing system, so it is required for the scheduling algorithm to generate a number of schedules in short time.

These requirements for the scheduling algorithm – performance consistency and time efficiency - are even more imperative when considering the reconfiguring problem, where a large number of configurations may necessitate scheduling for evaluation. The relationship between the scheduling problem and the reconfiguring problem is depicted in <Figure 4.1>.



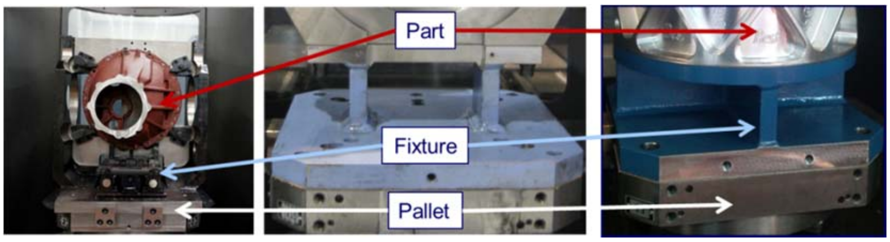
<Figure 4.1> The relationship between the scheduling problem and the reconfiguring problem

In this back ground, in this chapter, we are going to develop a scheduling algorithm appropriate for RMC environment. In the next section scheduling environment is described, and then the mathematical model is introduced. After that a heuristic algorithm based on lower bound prediction is proposed and an experiment is conducted to verify the performance of the proposed algorithm.

4.1. Scheduling Environment

Based on the production environment described in chapter 3, now let's see how operations are performed in an RMC.

An RMC processes production orders for a demand period (e.g. monthly demand). Each order indicates manufacturing parts of a certain type. Each part is produced through two consequent operations: loading (or setup) and machining. Loading operations are performed in L/U stations and machining operations in NC machines. A loading operation includes removing of the completed part and mounting of a new part on a pallet. After loading, parts may be processed in one of available machines directly or wait in a buffer. Processing times of loading and machining operations may differ from part types, and all L/U stations and machines are assumed identical.
For any operation, an empty pallet is required. Every pallet is the same, but distinguished by the fixture installed on it. And the installed fixture determines the supporting part type of the pallet. The 'pallet-fixture assignment' specifies which fixtures are to be installed on pallets. <Figure 4.2> indicates the example of the pallet-fixture assignment. The pallet-fixture assignment is given before production begins.



<Figure 4.2> An example of the pallet-fixture assignment

After machining, processed parts are extracted from the cell through L/U stations and returns pallets for processing of next parts.

It is assumed that cutting tools are not scarce resource and there exist enough capacity of the tool magazine. Therefore the part-loading problem or part-mix allocation problem [38] is not considered here. Besides, deterministic operation time is assumed and material handling time and the size of buffers are not considered.

Managerial objective is the production completion time or the makespan. It is known that the makespan objective guarantees high resource utilization [64]. Based on the makespan, managers may estimate the production time of an RMC with the current configuration.

Liu and MacCarthy [44] presented a classification scheme for FMS scheduling problems according to the FMS type, capacity constraints, job description, production environment, and scheduling criteria. According to it, the scheduling environment that this study is concerned is 'FMC/lim:LU,M,PL/JC1/pr,+pt/C_{max}' which means the FMS type is the FMC which a group of a single flexible machine(SFM)s sharing one common material handling device, with limited (denoted by 'lim') number of L/U stations, machines and pallets, each job consists of just one operation ('JC1'), and orders are handled periodically ('pr'), more than one part per type ('pt') for makespan objective.

It should be noted that RMC described above can be modelled as a two-stage pallet-constrained flexible flowshop (PcFF): The first stage is loading (setup) and the second stage is machining. As a flowshop, each part flows in the same sequence along the stages. However, pallets do not flow but circulate in the PcFF: After machining of one part, the associated pallet can be used for loading of another part. The two-stage pallet-constrained flexible flowshop is dealt in [73], where the shop with a single processor for each stage, and with commonly shared pallets regardless of part types or fixtures, was investigated. In our research, multiple processors for each stage and the distinguished pallets according to pallet-fixture assignment are considered.

4.2. Mathematical Model

Based on the problem description, following mathematical model is built:

PcFF

· Input parameters

i : Index of part types

l_i: Part i's loading (or setup) time

m_i: Part i's machining time

q_i: Production quantity (or demand) for part i

L : The number of L/U stations

M: The number of NC machines

p_i : The number of fixtured pallets which support part type i

T: The upper bound of the maximum completion time

t: Time index (t=1,2,…, T)

W : The large number

Both the loading and machining times (l_i and m_i) are assumed to be of integer multiples. The upper bound (T) of the makespan would be obtained by applying the simple dispatching rule such as SPT.

· Decision variables

$S_{it}^{\mathrm{l}}, S_{it}^{\mathrm{m}}$: The number of parts of type i which start its loading or machining operations at time t

· Dependent variables

$C_{it}^{\mathrm{l}}, C_{it}^{\mathrm{m}}$: The number of parts of type i which complete its loading or machining operations at time t

$R_t^{\mathrm{l}},\ R_t^{\mathrm{m}}$: The number of L/U stations or NC machines which process parts at time t

A_{it}: The number of pallets for type i which are occupied by parts for processing at time t

x_{it}: Binary variable indicating whether there is any part of type i which completes its machining operation at time t (1: exist, 0: not exist).

$\mathrm{C_{max}}$: The maximum completion time or makespan

The expected range of the makespan ($\mathrm{C_{max}}$) is from few days to few weeks.

· Constraints

$\sum_t S_{it}^{\mathrm{l}} = \sum_t S_{it}^{\mathrm{m}} = q_t \ \forall i$ (1)

$S_{it}^{\mathrm{l}} = 0 \ \forall i, t \in \{T - l_i + 2, T - l_i + 3, \dots, T\}$ (2)

$S_{it}^{\mathrm{m}} = 0 \ \forall i, t \in \{T - m_i + 2, T - m_i + 3, \dots, T\}$ (3)

$S_{it}^{\mathrm{l}} = C_{i(t+l_i-1)}^{\mathrm{l}} \ \forall i, t \in \{1,2, \dots, T - l_i + 1\}$ (4)

$S_{it}^{\mathrm{m}} = C_{i(t+m_i-1)}^{\mathrm{m}} \ \forall i, t \in \{1,2, \dots, T - m_i + 1\}$ (5)

$\sum_k^t C_{ik}^{\mathrm{l}} \geq \sum_k^{(t+1)} S_{ik}^{\mathrm{m}} \ \forall i, t \ \in \{1,2, \dots, T-1\}$ (6)

$R_t^{\mathrm{l}} = R_{t-1}^{\mathrm{l}} - \sum_i C_{i(t-1)}^{\mathrm{l}} + \sum_i S_{it}^{\mathrm{l}} \quad \forall \ t \ \in \{2, \dots, T\}$ (7)

$R_t^{\mathrm{m}} = R_{t-1}^{\mathrm{m}} - \sum_i C_{i(t-1)}^{\mathrm{m}} + \sum_i S_{it}^{\mathrm{m}} \quad \forall \ t \ \in \{2, \dots, T\}$ (8)

$R_1^{\mathrm{l}} = \sum_i S_{i1}^{\mathrm{l}}$ (9)

$R_1^{\mathrm{m}} = 0$ (10)

$A_{it} = A_{i(t-1)} - C_{i(t-1)}^{\mathrm{m}} + S_{it}^{\mathrm{l}} \quad \forall \ i, t \ \in \{2, \dots, T\}$ (11)

$A_{i1} = S_{i1}^{\mathrm{l}} \quad \forall i$ (12)

$R_t^{\mathrm{l}} \leq L, R_t^{\mathrm{m}} \leq M \ \forall \ t$ (13)

$A_{it} \leq p_i \ \forall i, t$ (14)

$W x_{it} \geq C_{it}^{\mathrm{m}} \ \forall i, t$ (15)

$\mathrm{C}_{\max} = \max_{i,t} \{x_{it} * t\}$ (16)

$S_{it}^{\mathrm{l}}, S_{it}^{\mathrm{m}} \in \mathrm{Z}^+, x_{it} \in \{0,1\} \ \forall \ i, t$ (17)

· Objective function

$z^{\mathbf{PcFF}} = \min \mathrm{C}_{\max}$ (18)

(1) - (3) means every part must start its loading and machining operations and operations should start so that it may finish within the upper time limit. (4), (5)

indicates relation between start and completion times of operations, and (6) forces the machining operation to start after the loading operation. (7) - (12) shows occupation of resources—both equipment and pallets—along time and (13), (14) says it should be less than or equal to resource capacity. Through (15), (16) the maximum completion time can be calculated. (17) indicates the variable conditions. Finally (18) indicates the objective function is the maximum completion time.

The complexity of the problem is NP-complete, for it can be reduced to the two-stage hybrid flowshop scheduling problem, which is NP-complete [23], by allowing enough pallets for each part type. When the problem was formulated in the exact search engine (iLOG CPLEX), the small size problem of around 10 jobs with 5 part types were taken over one and half hour, to find an optimal solution. It is practically impossible to apply an analytic approach. Therefore a heuristic approach may be a practical alternative.

From the next section it will be discussed about the algorithm for resolution of the scheduling problem.

4.3. Development of Algorithm

Before describing the developed algorithm, it needs to understand facts around the problem. First, the makespan ($z^{*\mathbf{PcFF}}$) is always larger than or equal to the total work amount allocated to each resource type divided by that resource capacity (or the number of resources). It means that the work amounts of various resource types constitute the initial lower bounds of the problem. In the problem are three kinds of resources – L/U station, NC machines, and pallets, and each constitutes a lower bound. In detail, they are given like below:

a) The lower bound in L/U station bottleneck case

$$LB_1^0 = \frac{\sum l_i \times q_i}{\min(L,\ \sum q_i)} + \min_i m_i$$

The superscript 0 in variable name means it is the initial lower bound calculated before the schedule is generated (i.e. no decision is made).

b) The lower bound in NC machine bottleneck case

$$LB_2^0 = \min_i l_i \ + \frac{\sum m_i \times q_i}{\min(M,\ \sum q_i)}$$

The lower bound for pallet bottleneck case is a little bit different from the other cases. Because pallets are distinguished by their supporting part types, the lower bound should be calculated for each of part type. The lower bound of pallets for part type k is given like that:

$$lb^0_{\mathrm{PA}(k)} = (l_k + m_k) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

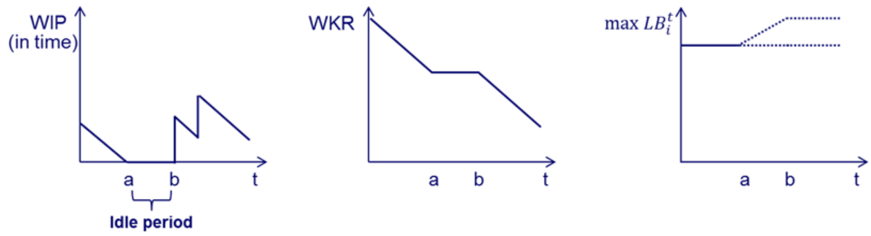
Finally the pallet-based lower bound is like below:

c) The lower bound in pallet bottleneck case

$$LB^0_3 = \max_{\mathrm{k}} lb^0_{\mathrm{PA}(k)} = \max_{\mathrm{k}} (l_k + m_k) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

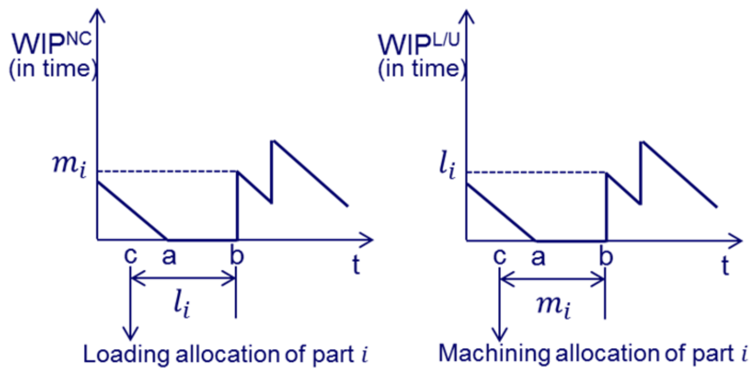
Definitely, $\mathrm{z}^{*\mathbf{PcFF}} \geq \max_i LB^0_i$. This information of lower bounds may be used as an estimator for $\mathrm{z}^{*\mathbf{PcFF}}$.

Each LB^0_i assumes no or minimum idleness of resources. But as a schedule is generated, resource idleness may occur. This is when there is no waiting job or work-in-process (WIP) in the queue, while resource is available. The resource idleness increases the corresponding LB^t_i and may increase the $\max LB^t_i$, according to whether the idle resource is bottleneck. (Superscript t in LB^t_i means the lower bound updated with regard to the schedule generated by time t.) <Figure 4.3> indicates impact of resource idleness on remaining work amount (WKR) and the updated lower bound. During the resource idle period, $a \leq t \leq b$ where WIP=0, remaining work amount is stagnant and it may lead to the increase of the lower bound.



<Figure 4.3> Impact of resource idleness

On the other hand, as the second fact, job allocation decisions affect the future status of the system, especially WIP. The impact of decisions on the future status is predictable via projection. For example, as in <Figure 4.4>, if a loading operation of part type i is assigned to a L/U station at $t = c$, then after the loading time (l_i), at $t = b = c + l_i$, the amount of WIP waiting for machining operations increases as much as the machining time (m_i). Conversely, if a machining operation of part type i is allocated to a NC machine, then after the machining time (m_i), the amount of WIP waiting for loading operations may increase, since the corresponding pallet is released for the next part, if exists. Through the WIP projection, the remaining work amount and the lower bound in the future time are also predictable.



<Figure 4.4> The work-in-process projection

Another implication of the second fact is that: It may not be enough to consider only permutation schedules. In two-stage (flexible) flowshop, it is known that there always exists an optimal schedule where the job sequence in both stages is identical [7]. In the pallet constraint flowshop, however, this property does not always stand.

Property 4.1 In pallet-constraint flowshops, where different pallets are required for different part types, permutation schedules may not constitute a dominant set.

Proof. See the counter example in the Appendix.

This property comes from the fact that the decisions in the second stage affect the future decisions in the first stage, since pallets circulate between stages.

Combining the two facts discussed above, it may be concluded that: During the schedule generation process, job allocation decisions affect WIP of resources in the future time, resulting in the probable increase of the lower bound with respect to that time point. A good schedule may be obtained, if decisions are made so that the increment from the initial lower bound ($\max_i LB_i^0$) is as small as possible. This philosophy led us to the following algorithm.

Note that the following scheduling algorithm takes the dispatching approach [7] where a partial schedule is generated at every decision point (i.e. when a resource becomes available) in chronological order and generation of the entire schedule is completed when all operations are assigned according to production requirement.

Algorithm Whenever a resource becomes available, allocate an operation such that its allocation will result in the minimum projected lower bound in the future. Detailed steps are as follows:

STEP1 (Initialization)	Initialize the partial schedule (S) as an empty schedule. Set the projection time (τ) to 0. Go to step 2.
STEP2 (Dispatching)	Based on S, find the minimum time t when equipment (L/U station or NC machine) becomes available. 2.1. Find operations assignable to that equipment. Set those operations as J. 2.2. For each operation $j \in J$, calculate the projected lower bound ($\max_k LB_k^{\tau'}(j)$) for the future time $\tau' = \max\{\tau, \mathrm{t} + l_j(\text{or } m_j)\}$ 2.3. Assign an operation $p = \underset{j \in J}{\operatorname{argmin}}\{\max_k LB_k^{\tau'}(j)\}$ to available equipment. If tie happens, assign the operation with larger τ'. 2.4. Set $\tau = \max\{\tau, \tau'\}$ and go to step 3.
STEP3 (Termination)	Terminate if all operations are assigned. Output S as the final schedule. Otherwise go to step 2.

$LB_k^{\tau'}(j)$ is the projected lower bound of k^{th} type for time τ', if an operation $j \in J$ is assigned to available equipment at the decision point, based on decisions made so far. Detail information is given below:

$$LB_1^{\tau}(j) = \frac{\sum l_i \times q_i^l(j,\tau) + \sum r_k^l(j,\tau)}{\min\left(L,\ \sum q_i^l(j,\tau) + \sum u_k^l(j,\tau)\right)} + \min_{i:q_i^m(j,\tau)>0} m_i$$

$$LB_2^{\tau}(j) = \begin{cases} \min\limits_k r_k^l(j,\tau) + \dfrac{\sum m_i \times q_i^m(j,\tau) + \sum r_k^m(j,\tau)}{\min\left(M,\ \sum q_i^m(j,\tau) + \sum u_k^m(j,\tau)\right)}, if \sum u_k^m(j,\tau) > 0 \\ \min\limits_{i:q_i^l(j,\tau)>0} l_i + \dfrac{\sum m_i \times q_i^m(j,\tau) + \sum r_k^m(j,\tau)}{\min\left(M,\ \sum q_i^m(j,\tau) + \sum u_k^m(j,\tau)\right)}, otherwise \end{cases}$$

$$LB_3^{\tau}(j) = \max_k \left(l_k \times q_k^l(j,\tau) + \sum_{v:\mathrm{F}(v)=k} r_v^l(j,\tau) + m_i \times q_i^m(j,\tau) + \sum_{v:\mathrm{F}(v)=k} r_v^m(j,\tau) \right) * \left\lceil \frac{q_k}{p_k} \right\rceil$$

Where

$u_k^l(j,\tau), u_k^m(j,\tau)$: Binary variable indicating whether there would exist an operation in processing at a L/U station or NC machine k at time τ, if part j is assigned at the current decision point (1: exist, 0: non-exists).

$r_k^l(j,\tau), r_k^m(j,\tau)$: Projected remaining operation time of a L/U station or NC machine k with respect to time τ, if an operation j is assigned at the current decision point.

$q_i^l(j,\tau), q_i^m(j,\tau)$: Projected remaining quantity of loading operations or machining operations with respect to time τ, if an operation j is assigned at the current decision point.

$\mathrm{F}(k)$: Part index of an operation which is being processed in a L/U station or NC machine k.

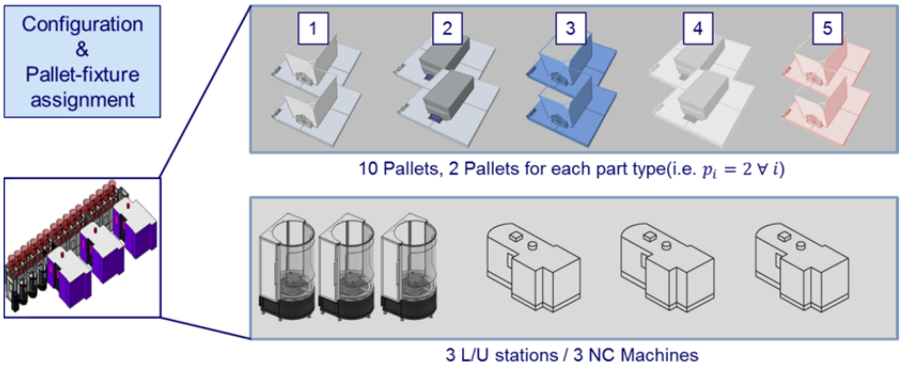
In calculation of $LB_k^{\tau'}(j)$, LB_k^0 is extended so that it would reflect the future system status which is derived by work-in-process projection, explained in <Figure 4.3> and <Figure 4.4>.

4.3.1. An Example for the Algorithm

For better understanding of the proposed algorithm, an example is presented, setting of which is depicted in <Figure 4.5>.

Master Data & Production quantity

Part Type (No.)	1	2	3	4	5
Loading Time (l_i)	42 min.	18	33	29	30
Machining Time(m_i)	60 min.	60	20	15	53
Production quantity(q_i)	4 units	3	1	3	3



<Figure 4.5> The example setting

For calculation of the initial lower bound, $LB_i{}^0$ is calculated by following:

$$LB_1^0 = \frac{\sum l_i \times q_i}{\min(L,\ \sum q_i)} + \min_i m_i \ = \frac{432}{\min(3,14)} + 15 = 159$$

$$LB_2^0 = \min_i l_i \ + \frac{\sum m_i \times q_i}{\min(M,\ \sum q_i)} = 18 + \frac{644}{\min(3,14)} = 232.67$$

$$lb_{\mathrm{PA}(1)}^0 = (l_1 + m_1) * \left\lceil \frac{q_1}{p_1} \right\rceil = (42 + 60) * \left\lceil \frac{4}{2} \right\rceil = 204$$

$$lb_{\mathrm{PA}(2)}^0 = (18 + 60) * \left\lceil \frac{3}{2} \right\rceil = 156$$

$$lb_{\mathrm{PA}(3)}^0 = (33 + 20) * \left\lceil \frac{1}{2} \right\rceil = 53$$

$$lb_{\mathrm{PA}(4)}^0 = (29 + 15) * \left\lceil \frac{3}{2} \right\rceil = 132$$

$$lb^0_{\mathrm{PA}(5)} = (30+53) * \left\lceil \frac{3}{2} \right\rceil = 166$$

$$LB^0_3 = \max_{\mathrm{k}} lb^0_{\mathrm{PA}(k)} = 204$$

So the initial lower bound of the problem is $\max_{\mathrm{i}} {LB_i}^0 = 232.67$.

For initialization, set $\tau = 0$.

When $\mathrm{t} = 0$, all three L/U stations are available. For job allocation of the first L/U station, all types of parts are assignable, which means $J = \{1,2,3,4,5\}$.

For each assignable part $j \in J$, τ' and the projected lower bound ($\max_{k} LB_k^{\tau'}(j)$) is like below:

For $j = 1$, $\tau' = \max\{\tau, \mathrm{t} + l_1\} = \max\{0, 0 + 42\} = 42$,

$$u^l_k(1,42), u^m_k(1,42), r^l_k(1,42), r^m_k(1,42) = 0 \ \forall k$$

$$q^l_1(1,42) = 3$$

$$q^l_i(1,42), q^m_i(1,42) = q_i \ for \ i = 2,3,4,5$$

$$\max_{k} LB^{42}_k(1) = LB^{42}_2(1) = 263.5$$

For $j = 2$, $\tau' = 18$, $max_k \, LB^{18}_k(2) = 240$

For $j = 3$, $\tau' = 33$, $max_k \, LB^{33}_k(3) = 269$

For $j = 4$, $\tau' = 29$, $max_k \, LB^{29}_k(4) = 248.33$

For $j = 5$, $\tau' = 30$, $max_k \, LB^{30}_k(5) = 248$

Therefore, part 2 $= \operatorname{argmin}_{j \in J}\{\max_{k} LB^{\tau'}_k(j)\}$ is allocated to the L/U station. And τ is updated $\max\{\tau, \tau'\} = \max\{0, 42\} = 42$.

For decisions afterward, dispatching information is depicted in <Table 4.1>.

<Table 4.1> Dispatching information for the example of the proposed algorithm

t	Available resource	Assignable jobs J	Selected Job $p = \underset{j \in J}{\mathrm{argmin}}\{\underset{k}{max}\, LB_k^{\tau'}(j)\}$	Projected lower bound $\underset{k}{max}\, LB_k^{\tau'}(p)$	Projection time τ
0	L/U 2	1,2,3,4,5	2	236.3	18
0	L/U 3	1,3,4,5	4	236.3	29
18	L/U 1	1,3,4,5	1	243	60
18	L/U 1	1,3,4,5	1	241.67	60
18	MC 1	2	2	244	78
18	MC 2	2	2	244	78
29	L/U 3	3,4,5	5	241.33	78
29	MC 3	4	4	241.33	78
59	L/U 3	3,4,5	5	266.33	89
59	MC 3	5	5	274	112
60	L/U 1	3,4	4	274	112
60	L/U 2	3,4	3	274	112
78	MC 1	1	1	270	138
78	MC 2	1	1	242.67	138
89	L/U 1	2,4	2	252.33	138
89	L/U 3	4	4	241.33	138
112	L/U 1	5	5	264.33	142
112	MC 3	2,3,4,5	4	244	142
127	MC 3	2,3,4,5	4	256.33	142
138	L/U 2	1	1	261	180

138	L/U 3	1	1	241.33	180
138	MC 1	2,3,5	3	241.33	180
138	MC 2	2,5	5	251	191
142	MC 3	2,5	5	255	195
158	MC 1	2	2	278	218
191	MC 2	1	1	284	251
195	MC 3	1	1	259	255

The algorithm ends up with completion time 255 (min.), about 10% larger than the initial lower bound.

4.4. Experiment

To verify the performance of the proposed algorithm, an experiment was conducted.

Currently, hardware of a single RMC in development in the Korean NC machine maker supports up to 4 L/U stations and 9 NC machines. The number of supporting NC machines is larger than that of L/U stations, because usually machining operations take longer than loading operations. And this is reflected in the experiment.

Five experimental factors were chosen to experiment various shop configurations and demands: the number of L/U stations, machines and pallets, and the number of part types and production volume for each part type. Parameter information about processing times (l_i and m_i) were obtained from field data so that $l_i \sim \mathrm{UNIF}(10{,}50)$min. and $m_i \sim \mathrm{UNIF}(10{,}100)$min., respectively. As for the pallet-fixture assignment, the same number of total pallets was equally assigned for each part type and the remainder was assigned randomly.

As alternatives to be prepared with the proposed algorithm, various dispatching rules and a heuristic search engine, iLOG CP, were taken into account. iLOG CP is a constraint programming-based search engine that is known more appropriate for problems with disjunctive constraints such as scheduling problems, than exact algorithm based search engines, for example, iLOG CPLEX [28]. (It is also observed in the pilot study that iLOG CP showed superior performance than iLOG CPLEX.) Description of alternative dispatching rules is depicted in <Table 4.2>. Additionally, the failure limit of iLOG CP was set to ten thousands.

Performance was measured in the percentage gap from the initial lower bound (max LB_i^0) given like below:

$$\%\text{Gap} = \frac{C_{max}(r) - \max_i LB_i^0}{\max_i LB_i^0} \times 100$$

where $C_{max}(r)$ is the makespan obtained from applying a dispatching rule or an algorithm r.

Finally for each experimental condition, the average performance from 10 replications was recorded, together with the computational time spent for generating the schedules.

<Table 4.2> Dispatching rules used in experiment as alternatives

Rule	Description
SPT	Shortest processing time job first
LPT	Longest processing time job first
MRK	Most work remaining job first
LRK	Least work remaining job first
LVP	Least vacant pallets job first

4.5. Result and Analysis

<Table 4.3> and <Table 4.4> show the result of the experiment for the performance measure and computational time, respectively. Figures from <Figure 4.6> to <Figure 4.8> are graphical representation. A*(non-p.) indicates the proposed algorithm which is applied to determining both loading and machining sequence, allowing generation of non-permutation schedules. A*(perm.) means the algorithm is applied to sequencing only loading operations, forcing the same sequence of loading and machining operations (i.e. permutation schedules).

The result shows that the proposed non-permutation algorithm (A*(non-p.)) consistently performs well over the alternative dispatching rules and iLOG CP for various shop configurations and demand settings. The performance of the least work remaining rule (LRK) and iLOG CP are relatively good, but the statistics — average, maximum and minimum performance value, and standard deviation— says that the proposed algorithm is superior to those alternatives as well.

On the other hand, the proposed permutation algorithm (A*(perm.)) appeared not to performs well, proving the nature of pallet-constrained flexible flowshops in which permutation schedules are not enough to find efficient solutions.

For computational time, as shown in <Table 4.4> and <Figure 4.8>, the proposed algorithm appeared to generate schedules as fast as the alternative dispatching rules, while the heuristic search engine iLOG CP spent about 10 to 100 times more to do the same job.

This result implies that the proposed algorithm performs consistent for various hardware configurations or demands and may generate a number of schedules quickly, therefore appropriate for hardware reconfiguration where various configurations need to be evaluated to find the best one. Moreover, the proposed algorithm may be applicable to manufacturing execution as well, where real-time decision making is required.

<Table 4.3> The experimental result for the performance measure (%Gap)

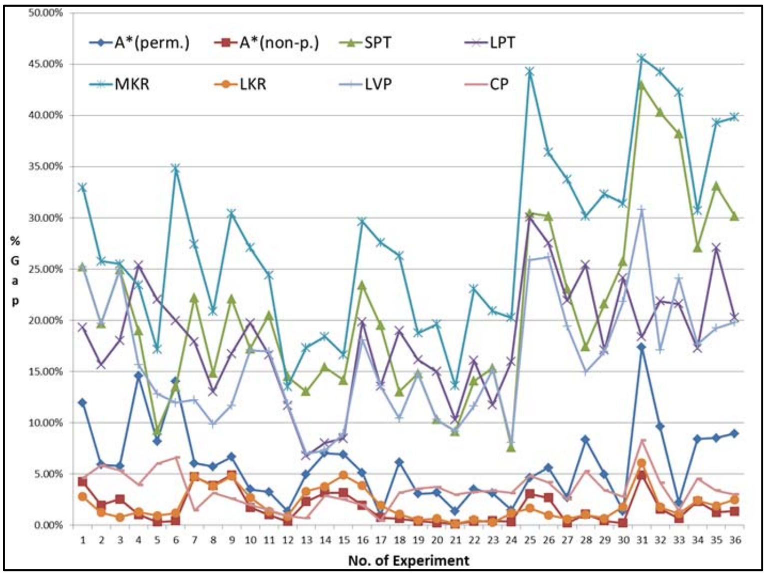
No.	L/U	NC	Pallet	Part	Vol.	A* (perm.)	A* (non-p.)	SPT	LPT	MKR	LKR	LVP	CP
1	2	3	10	10	10-30	11.94	4.23	25.24	19.29	32.96	2.80	25.24	4.63
2	2	3	10	10	30-60	5.90	1.96	19.67	15.67	25.77	1.22	19.67	5.82
3	2	3	10	10	60-90	5.78	2.56	24.92	18.02	25.49	0.79	24.92	5.35
4	2	3	20	10	10-30	14.55	1.05	18.96	25.37	23.42	1.30	15.66	3.95
5	2	3	20	10	30-60	8.20	0.33	9.27	22.01	17.15	0.93	12.81	6.01
6	2	3	20	10	60-90	14.05	0.43	13.51	19.96	34.82	1.18	11.95	6.62
7	3	6	15	10	10-30	6.05	4.72	22.22	17.88	27.41	4.76	12.22	1.44
8	3	6	15	10	30-60	5.71	3.89	14.82	13.02	20.86	3.82	9.87	3.15
9	3	6	15	10	60-90	6.65	4.91	22.12	16.75	30.40	4.79	11.68	2.60
10	3	6	30	10	10-30	3.50	1.75	17.22	19.75	27.09	2.72	17.07	1.99
11	3	6	30	10	30-60	3.26	1.02	20.49	16.63	24.40	1.37	16.94	1.50
12	3	6	30	10	60-90	1.36	0.38	14.50	11.67	13.50	0.89	11.83	0.92
13	4	9	20	10	10-30	4.96	2.30	13.05	6.77	17.32	3.30	6.97	0.68
14	4	9	20	10	30-60	7.05	3.18	15.41	8.03	18.39	3.78	7.32	2.87
15	4	9	20	10	60-90	6.90	3.16	14.16	8.47	16.63	4.88	8.94	2.52
16	4	9	40	10	10-30	5.14	1.97	23.43	19.85	29.63	3.87	18.09	1.97
17	4	9	40	10	30-60	0.93	0.75	19.52	13.56	27.60	1.95	13.59	0.42
18	4	9	40	10	60-90	6.16	0.65	13.04	18.95	26.30	1.06	10.42	3.16
19	2	3	20	20	10-30	3.07	0.39	14.80	16.13	18.76	0.51	14.80	3.58
20	2	3	20	20	30-60	3.17	0.25	10.32	14.99	19.60	0.65	10.32	3.73
21	2	3	20	20	60-90	1.37	0.12	9.15	10.30	13.58	0.14	9.15	2.99
22	2	3	30	20	10-30	3.53	0.38	14.07	16.05	23.04	0.55	11.63	3.25
23	2	3	30	20	30-60	3.12	0.42	15.34	11.73	20.91	0.25	15.17	3.39
24	2	3	30	20	60-90	1.48	0.34	7.60	15.97	20.24	1.19	8.07	3.15
25	3	6	30	20	10-30	4.62	3.06	30.45	30.02	44.29	1.67	25.88	4.81
26	3	6	30	20	30-60	5.62	2.69	30.18	27.55	36.37	0.98	26.15	4.20

27	3	6	30	20	60-90	2.72	0.20	23.08	21.91	33.78	0.61	19.43	2.53
28	3	6	40	20	10-30	8.33	1.06	17.41	25.43	30.18	1.02	14.97	5.30
29	3	6	40	20	30-60	4.96	0.42	21.59	17.09	32.32	0.67	16.94	3.41
30	3	6	40	20	60-90	1.36	0.20	25.74	24.12	31.39	1.80	21.83	2.80
31	4	9	40	20	10-30	17.37	4.92	42.95	18.36	45.59	6.10	30.78	8.27
32	4	9	40	20	30-60	9.61	1.57	40.29	21.86	44.26	1.77	17.09	4.16
33	4	9	40	20	60-90	2.23	0.68	38.21	21.59	42.25	1.13	24.09	1.41
34	4	9	50	20	10-30	8.40	2.33	27.10	17.26	30.68	2.43	17.65	4.53
35	4	9	50	20	30-60	8.52	1.27	33.11	27.09	39.27	1.90	19.24	3.39
36	4	9	50	20	60-90	8.94	1.38	30.16	20.24	39.80	2.49	19.78	3.00
Average (%)						6.01	**<u>1.69</u>**	20.92	18.04	27.93	1.98	16.06	3.43
Max value (%)						17.37	**<u>4.92</u>**	42.95	30.02	45.59	6.10	30.78	8.27
Min value (%)						0.93	**<u>0.12</u>**	7.60	6.77	13.50	0.14	6.97	0.42
Standard deviation (%)						3.91	**<u>1.49</u>**	8.83	5.54	8.94	1.52	6.02	1.70

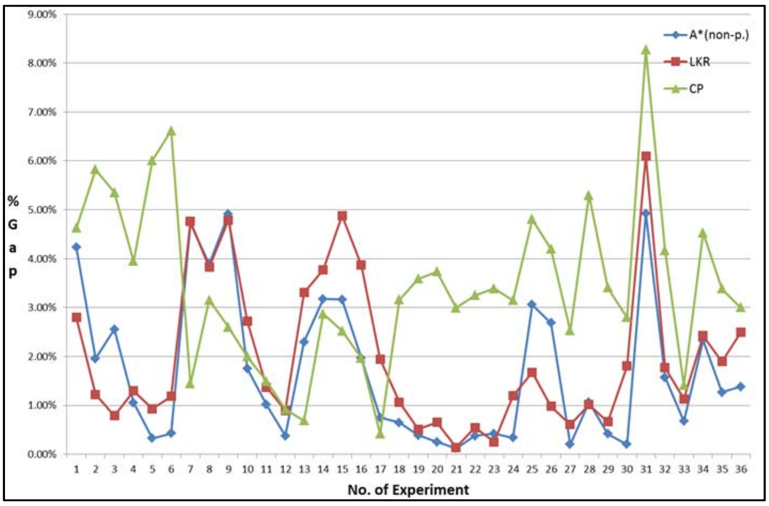
<Table 4.4> The experimental result for computational time (millisecond)

No	LU	NC	PA	Part	Vol.	A* (perm.)	A* (non-p.)	SPT	LPT	MKR	LKR	LVP	CP
1	2	3	10	10	10-30	139	172	106	185	141	145	103	6911
2	2	3	10	10	30-60	476	523	357	626	519	534	349	30225
3	2	3	10	10	60-90	2440	2850	1852	3616	2225	2795	1858	146109
4	2	3	20	10	10-30	201	207	129	225	183	145	116	6328
5	2	3	20	10	30-60	1018	842	479	1371	733	851	496	27598
6	2	3	20	10	60-90	2278	1933	1088	3286	1946	2176	1021	154064
7	3	6	15	10	10-30	123	121	110	122	128	93	96	5701
8	3	6	15	10	30-60	638	605	590	621	676	491	472	24228
9	3	6	15	10	60-90	1083	903	1145	1278	1486	822	796	147648
10	3	6	30	10	10-30	166	242	58	108	95	67	67	4054
11	3	6	30	10	30-60	552	713	267	407	387	184	294	17183
12	3	6	30	10	60-90	1428	1420	744	1048	835	802	659	108482
13	4	9	20	10	10-30	206	202	190	198	218	149	141	5309
14	4	9	20	10	30-60	960	930	871	917	1050	741	711	41292
15	4	9	20	10	60-90	2152	1725	2197	1681	2141	1532	1648	77113
16	4	9	40	10	10-30	303	379	75	115	118	50	69	4006
17	4	9	40	10	30-60	828	942	256	351	383	242	280	11935
18	4	9	40	10	60-90	1841	1998	859	1622	1408	944	802	49697
19	2	3	20	20	10-30	748	1340	357	1029	645	879	373	21996
20	2	3	20	20	30-60	3411	5260	1629	5618	2598	4335	1653	184726
21	2	3	20	20	60-90	11582	15444	5702	16957	8832	13885	5729	703562
22	2	3	30	20	10-30	772	1319	389	1063	651	780	369	18327
23	2	3	30	20	30-60	4594	6863	2776	6840	4475	5513	2705	144962
24	2	3	30	20	60-90	9228	13520	4563	13752	8075	12302	4522	652904
25	3	6	30	20	10-30	568	770	233	324	392	188	214	9813
26	3	6	30	20	30-60	1732	2227	1144	2005	1720	1160	997	122091

27	3	6	30	20	60-90	4654	5583	3089	4653	4750	3768	2774	471925
28	3	6	40	20	10-30	1004	1257	285	652	500	379	307	12625
29	3	6	40	20	30-60	3337	3483	1354	2545	2049	1877	1460	78150
30	3	6	40	20	60-90	5084	5647	2913	7253	4939	3654	3545	475142
31	4	9	40	20	10-30	998	1353	413	476	534	336	356	10144
32	4	9	40	20	30-60	2637	3218	1539	1778	1824	978	1009	70056
33	4	9	40	20	60-90	5634	5766	3643	4808	4528	2786	2798	255274
34	4	9	50	20	10-30	1387	1609	298	403	364	231	268	10073
35	4	9	50	20	30-60	3662	4274	1345	2228	1845	1066	1230	64893
36	4	9	50	20	60-90	7185	7447	3157	4994	4108	3350	3305	337145

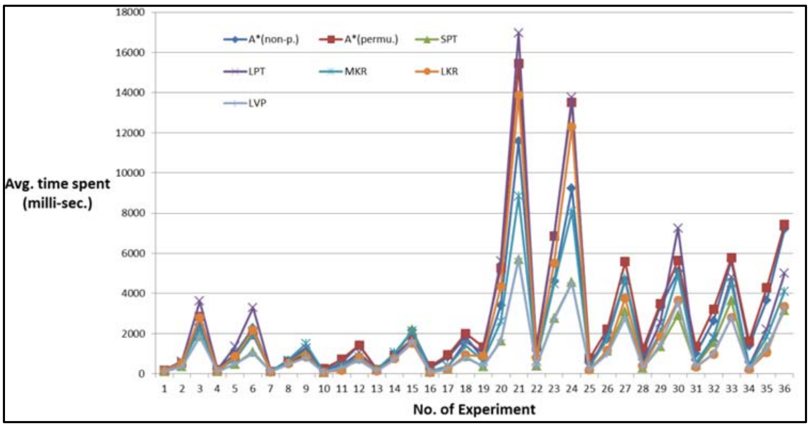


<Figure 4.6> The experimental result for the performance measure (%Gap)



<Figure 4.7> The experimental result for the performance measure

(%Gap, selected and zoomed)



<Figure 4.8> The experimental result for computational time (excluding iLOG CP)

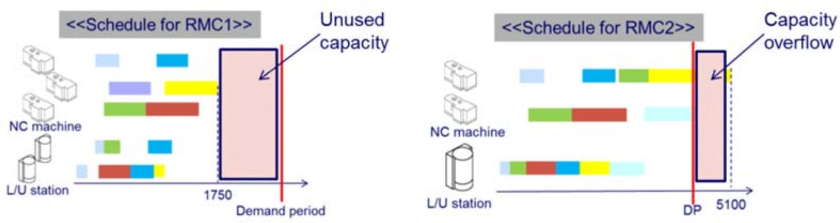
The robustness of the proposed algorithm may be originated from the fact that the algorithm reflects shop configurations and demand information in its reasoning scheme (i.e. calculation of LB_i^τ) and also looks ahead of the future status of the shop via work-in-process projection, unlike the conventional dispatching rules.

In the next chapter, an integrative framework for the reconfiguring problem using the developed scheduling algorithm is discussed.

5. A Framework for Hardware Reconfiguration

The existing manufacturing system may fail to respond to demand. In this case managers may consider: rejection of some orders [40], negotiation [30], or out-sourcing [10]. In RMC environment hardware reconfiguration may be considered as well. It is especially when there is unused capacity in some parts while capacity overflow occurs in other parts, as depicted in <Figure 5.1>.

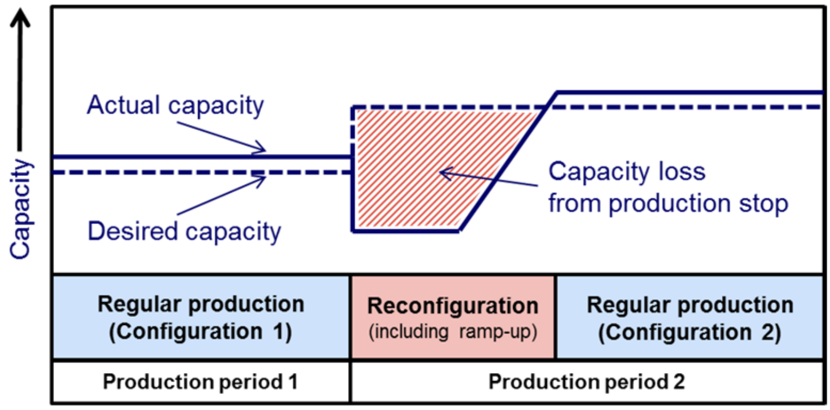
In this case, managers may have questions like this: "How does the current system need to be changed to satisfy demand or to reduce production delay?" The **Reconfiguring problem** is to decide the best hardware configuration to improve manufacturing performance for given demand. Resolution may not be easy, for the number of possible alternatives is huge. Therefore a systematic framework for hardware reconfiguration needs to be developed.



<Figure 5.1> The case when hardware reconfiguration is considered

On the other hand, it takes time for hardware reconfiguration. Production stops during reconfiguration. Reconfiguring time directly affects manufacturing performance. Capacity loss from the production stop during hardware reconfiguration is depicted in <Figure 5.2>. (Note that here capacity means that of a single RMC, since total capacity of the factory does not change.) If such capacity loss is too much, it may not be desirable to change the current configuration. Therefore, impact of reconfiguring time should be considered for accurate evaluation of manufacturing performance.

Monetary cost also occurs for reconfiguration such as labor cost, but this study does not consider monetary cost explicitly. Since monetary cost tends proportional to reconfiguring time [80], monetary cost may be considered implicitly by considering time required for reconfiguration.



<Figure 5.2> The capacity change around hardware reconfiguration

(Edited from Spicer & Carlo [80])

In this background, we are going to develop an integrative framework for the reconfiguring problem. A relaxation based branch-and-bound algorithm is suggested for configuration search. The scheduling algorithm suggested in the previous chapter will be used for configuration evaluation as a sub-component.

5.1. Mathematical Model

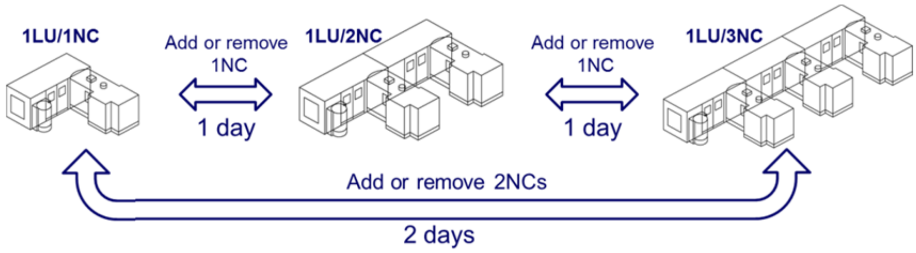
Description about production environment follows the one explained in chapter 3.

As described in <Figure 3.1>, there exist multiple production lines in the factory, according to product groups. And there exist multiple RMCs, each being defined with the number of L/U stations, NC machines, and fixtured pallets according to part types. Each RMC is given a production order for a pre-defined demand period. A production order consists of part types and volumes.

Reconfigurable components are L/U stations, NC machines, and pallets. Those components are interchangeable among RMCs via reconfiguration. Pallets are also deliverable from one RMC to another.

Reconfiguration includes removing or attaching of equipment, tool changes, NC program uploads, precision tests and adjustment, and all the other activities required to ramp up the manufacturing system with the new hardware configuration.

Change from one hardware configuration to another takes time. Production interruption of an RMC from hardware reconfiguration and ramp-up is independent each other: The duration of production interruption is proportional to the degree of change from the initial hardware configuration, as depicted in <Figure 5.3>.



<Figure 5.3> Reconfiguring time according to change of cell configurations

Time required for delivering pallets to another cells and changing the pallet-fixture assignment is relatively small compared to time for exchanging L/U stations and NC machines, and it may be performed in parallel. Therefore it is assumed that the time required for pallet adjustment is included in reconfiguring time of other equipment.

Managers' primary objective is demand fulfillment. Then managers are concerned with reconfiguring cost, as secondary objectives. In the case where it may not be possible to complete production within the available time, managers seek to minimize total delay – difference from the available time and the estimated completion time.

Description of operation environment of RMCs is following that described in chapter 3.

Based on the description of the previous section, the following mathematical model for the reconfiguring problem (RP) is built:

RP

· Input parameters (New or redefined variables only)

c : Index of the cells

q_{ci}: The production quantity (or demand) for part i of cell c

w_c : The relative weight of cell c

L_c^0 : The initial number of L/U stations of cell c

M_c^0: The initial number of NC machines of cell c

α : The unit time required for reconfiguration of L/U station

β : The unit time required for reconfiguration of NC machine

H: The demand period or the available time (e.g. few weeks or a month)

The demand period (H) should be adjusted so that it has the same unit with the time index.

· Decision variables

L_c : The number of L/U stations of cell c after reconfiguring

M_c: The number of NC machines of cell c after reconfiguring

· Dependent variables

$\underline{p_{ci}}$: The minimal number of pallets which support part type i in cell c after reconfiguring

$\mathrm{C}_{\max}^{c}$: The required production time of the cell c

$z^{*\mathbf{PcFF}}(L_c, M_c, p_{ci}, q_{ci})$: The optimal makespan of the pallet-constrained flexible flowshop with a configuration consisting of $L = L_c, M = M_c, p_i = p_{ci}\ \forall i$ and demand $q_i = q_{ci}\ \forall i$

D_c: Reconfiguring time of the cell c

y_c: The binary variable indicating demand fulfillment of the cell c (1: fulfilled, 0: not fulfilled)

· Constraints

$$\sum_c L_c = \sum_c L_c^0, \sum_c M_c = \sum_c M_c^0 \quad (1)$$

$$C_{\max}^c = z^{*\mathbf{PcFF}}(L_c, M_c, \underline{p_{ci}}, q_{ci}) \forall c \quad (2)$$

$$D_c = \alpha|L_c - L_c^0| + \beta|M_c - M_c^0| \forall c \quad (3)$$

$$W(1 - y_c) \geq C_{\max}^c + D_c - H\ \forall c \quad (4)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}}} \right\rceil \leq \max\{LB_1^c, LB_2^c\}\ \ \forall c, i \quad (5)$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}} - 1} \right\rceil > \max\{LB_1^c, LB_2^c\} \forall c, i \quad (6)$$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c, \sum q_{ci})} + \min_i m_i \quad (7)$$

$$LB_2^c = \min_i l_i\ + \frac{\sum m_i \times q_{ci}}{\min(M_c, \sum q_{ci})} \quad (8)$$

$$L_c, M_c, \underline{p_{ci}} \in Z^+, y_c \in \{0,1\} \forall c, i \quad (9)$$

· Objective function

$$z^{\mathbf{RP}} = \min \sum_c \{w_c * (1 - y_c) * (C_{\max}^c + D_c - H)\} \quad (10)$$

(1) indicates the total number of equipment—L/U stations and NC machines—of the whole factory is fixed (i.e. purchasing of equipment is not

considered). (2) indicates the required production time of each cell after reconfiguring is calculated by scheduling and through (3), the corresponding reconfiguring time can be calculated. (4) is for checking on-time production, whether whole production of the cell is finished within the available time. (5) and (6) implies that pallets are determined minimal, not causing bottleneck after reconfiguration. (7) and (8) indicates corresponding lower bounds in the case of L/U station and machine bottleneck, respectively. Note that through **RP**, only the minimal number of pallets is determined. The actual number of pallets for each part type of each RMC is determined after solving **RP**, which is discussed in the later section. (9) indicates the integer or binary conditions of variables. Finally the objective function (10) indicates the sum of production delay of whole RMCs.

Note that it is practically impossible to obtain $z^{*\mathbf{PcFF}}(L_c, M_c, p_i, q_i)$ for a certain configuration, as it is a NP-hard problem. Instead of seeking $z^{*\mathbf{PcFF}}(L_c, M_c, p_i, q_i)$, we are going to use the makespan obtained by applying the scheduling algorithm suggested in the previous chapter, denoted by $z^{\mathbf{PcFF(A)}}(L_c, M_c, p_i, q_i)$. Therefore the constraint (2) in **RP** would be substituted like below:

$$\mathrm{C}_{\max}^{c} = z^{\mathbf{PcFF(A)}}(L_c, M_c, \underline{p_{ci}}, q_{ci}) \forall c \quad (2')$$

Since the developed scheduling algorithm appeared to perform close to the lower bound and robust for various configurations, it would be the best alternative minimizing performance deterioration.

5.2. Solution Procedure for Hardware Reconfiguration: Branch-and-bound Approach

The most common way to solve integer programs is the branch-and-bound (B&B) [92]. It is often found in literature to employ B&B to solve the scheduling problems [7][8].

B&B seeks to reduce search space by calculating lower bounds of sub-problems. Tightness of the lower bounding scheme determines effectiveness of B&B.

A relaxation problem is often used to obtain the lower bound [45][92]. Linear relaxation is common but in the case of **RP**, it may not be applied because i) **RP** is not linear and ii) even if it is possible to obtain a real-value optimal solution, it may provide loose bounds.

Therefore it is required to design a relaxation problem being solved fast and providing tight lower bounds. In the next section it will be discussed about the relaxation problem.

5.2.1. Relaxation of the Problem

The relaxation problem of the original reconfiguring problem (**RPX**) is described below:

First, the required production time of each cell ($C^c_{\max}$) is redefined like below:

$$C^c_{\max} = \max\{ LB^c_1 + \alpha|L_c - L^0_c|, LB^c_2 + \beta|M_c - M^0_c|\} \; \forall c \quad (2x)$$

Second, in **RPX**, the (3) in **RP** is substituted like blow:

$D_c = 0 \ \forall c \qquad (3x)$

Note that instead of setting $D_c = 0$, the required production time partially includes the reconfiguring time.

Proposition 5.1 RPX is a relaxation problem of **RP**.

Proof. The solution space of decision variables of **RPX** is the same with **RP** and the objective value of **RPX** is always smaller than **RP** for every feasible solution, since $z^{\mathbf{PcFF}(\mathrm{A})}(L_c, M_c, p_{ci}, q_{ci}) + D_c \geq \max\{LB_1^c + \alpha|L_c - L_c^0|, LB_2^c + \beta|M_c - M_c^0|\}$ ■

Note that the relaxed terms ($C_{\max}^c$, D_c) are all related to the objective function. A relaxation problem is usually designed so that some of constraints in the original problem is removed or relaxed and the feasible region of the problem is extended. The designed relaxation problem (RPX), however, only seeks to simplify calculation of the objective function.

For **RPX** to be meaningful to solve **RP**, it should have properties such that it can be solved fast. In the next two sections such properties and how to solve **RPX** are investigated.

5.2.1.1. Dimension Reduction of the Relaxation Problem

If we are going to solve **RPX** via complete enumeration, the number of possible configurations that should be investigated may be too large.

The number of possible configurations is given like below:

$$ {}_{\sum L_c - 1}\mathrm{C}_{\mathrm{N}-1} \times {}_{\sum M_c - 1}\mathrm{C}_{\mathrm{N}-1} $$

where N is the number of cells in the factory.

Considering that the expected range of the number of equipment and cells in a factory is $3 \le \mathrm{N} \le 10$, $5 \le \sum L_c \le 30$, and $10 \le \sum M_c \le 50$, above number is too large to enumerate completely. To reduce the problem space, we consider followings:

First, the ideal number of equipment is defined.

Definition:

'*The ideal number of equipment*' of a cell is defined the minimal number which satisfies the following conditions:

$$ H \ge \frac{\sum l_i \times q_{ci}}{\min(L_c^I,\ \sum q_{ci})} + \min_i m_i + \alpha |L_c^I - L_c^0| $$

$$ H \ge \min_i l_i + \frac{\sum m_i \times q_{ci}}{\min(M_c^I,\ \sum q_{ci})} + \beta |M_c^I - M_c^0| $$

L_c^I and M_c^I denote the ideal number of L/U stations, NC machines of cell c, respectively. With the ideal number of equipment, the lower bounds of the estimated completion time are always smaller than or equal to the available time. Then by definition following properties hold.

Property 5.1

For $a \in Z^{+}$,

$$\mathrm{C}_{\max}^{c}(L_c^I - a, M_c) \geq \mathrm{C}_{\max}^{c}(L_c^I, M_c) \leq H$$

And for $a \in Z^{+} \cup \{0\}$ and $a \leq L_c^0 - L_c^I$,

$$\mathrm{C}_{\max}^{c}(L_c^I + a, M) = \mathrm{C}_{\max}^{c}(L_c^I, M) \leq H$$

Property 5.2

For $a \in Z^{+}$,

$$\mathrm{C}_{\max}^{c}(L_c, M_c^I - a) \geq \mathrm{C}_{\max}^{c}(L_c, M_c^I)$$

And for $a \in Z^{+} \cup \{0\}$ and $a \leq M_c^0 - M_c^I$,

$$\mathrm{C}_{\max}^{c}(L_c, M_c^I + a) = \mathrm{C}_{\max}^{c}(L_c, M_c^I)$$

Property 5.1 and **Property 5.2** hold by definition and by the fact that the relaxed reconfiguring time ($\alpha|L_c^I - L_c^0|$ and $\beta|M_c^I - M_c^0|$) does not increase (rather maybe decrease) as the L_c, M_c approaches to the initial configuration. Using the properties, following proposition is derived.

Proposition 5.2 If $\sum(L_c^0 - L_c^I) \geq 0$, $z^{\mathbf{RPX}}(L_c^*, M_c^*)$ is the same with a sub-problem with decision variable $L_c = L_c^I + a_c, a_c \geq 0$ ∀c, and a_c also can be derived.

Proof. For $c: L_c^* < L_c^I$, setting $L_c' = L_c^I$ would not increase the objective value, and for $c{:}\, L_c^* > L_c^I$, setting $L_c' = L_c^I$ would also not increase the objective value by **Property 5.1**.

And suppose $a = \sum(L_c^0 - L_c^I) \geq 0$. Then it is possible to modify $L_c' = L_c^I + a_c$, $0 \leq a_c \leq L_c^0 - L_c^I$ for $c{:}\, L_c^0 > L_c^I$ so that $\sum a_c = a$**,** not increasing the objective value, by **Property 5.1.** Now L_c' is a feasible solution. Therefore the following holds:

$z^{\mathbf{RPX}}(L_c^*,\ M_c^*) = z^{\mathbf{RPX}}(L',\ M_c^*) = z^{\mathbf{RPX}}(L_c^I,\ M_c^*) = z^{\mathbf{RPX}}(L_c^I + a_c,\ M_c^*)$ ■

Proposition 5.2 is for L/U stations. A similar proposition holds for NC machines.

Proposition 5.3 If $\sum(M_c^0 - M_c^I) \geq 0$, $z^{\mathbf{RPX}}(L_c^*, M_c^*)$ is the same with a sub-problem with decision variable $M_c = M_c^I + a_c\,, a_c \geq 0$ ∀c, and a_c also can be derived.

Proof. The same with **Proposition** 5.2 ■

Combining **Proposition 5.2** and **Proposition 5.3**, the following proposition is derived:

Proposition 5.4 If $\sum(L_c^0 - L_c^I) \geq 0$, $\sum(M_c^0 - M_c^I) \geq 0$, and then $z^{\mathbf{RPX}}(L_c^*, M_c^*) = z^{\mathbf{RPX}}(L_c^I, M_c^I) = 0$.

It is probable that only a single kind of resource is in shortage. In that case, the problem dimension may be reduced by half using **Proposition 5.2** and

Proposition 5.3. And if there is no resource type in shortage, and then the optimal objective value can directly be derived by **Proposition 5.4**. Note that the above propositions may be very useful while solving sub-problems in the search tree of B&B, even if the original problem does not satisfy any of the conditions.

5.2.1.2. Alternative Formulation to Reduce the Domain of Variables

The concept of the ideal number of equipment provides additional advantage. Consider following propositions.

Proposition 5.5 If $\sum(L_c^I - L_c^0) \geq 0$, there is an optimal solution with $L_c \leq L_c^I \forall c$

Proof. Suppose an optimal solution L_c^*, $M_c^* \forall c$ and for some cell k, $L_k^* > L_k^I = L_k^* - a$.

Then by setting $L_k' = L_k^I$ and $L_c' = L_c^* \forall c/\{k\}$, an alternative solution L_c', M_c^* can be obtained without increase of the objective value, by **Property 5.1**. And it is possible to modify $L_c' = L_c^I + a_c,\ 0 \leq a_c \leq L_c^0 - L_c^I$ for $c: L_c^0 > L_c^I$ so that $\sum a_c = a$, not increasing the objective value, by **Property 5.1.** And now L_c' is a feasible solution. ■

Proposition 5.5 is for L/U stations. A similar proposition holds for NC machines.

Proposition 5.6 If $\sum(M_c^I - M_c^0) \geq 0$, there is an optimal solution with $M_c \leq M_c^I \forall c$

By **Proposition 5.5** and **Proposition 5.6**, the domain of decision variables would be reduced. To reflect the reduced variable domain in the model, an alternative formulation of **RPX** is built, named **A-RPX.**

A-RPX

· Input parameters (New or redefined variables only)

L_c^I : The ideal number of L/U stations of cell c (based on the definition in the previous section)

M_c^I: The ideal number of NC machines of cell c

· Decision variables

L_c^- : The shortage of L/U stations of cell c from the ideal number (L_c^I) after reconfiguring

M_c^-: The shortage of NC machines of cell c from the ideal number (M_c^I) after reconfiguring

· Constraints

$$\textstyle\sum_c (L_c^I - L_c^-) = \sum_c L_c^0 , \sum_c (M_c^I - M_c^-) = \sum_c M_c^0 \quad \text{(1x-a)}$$

$$\mathrm{C}_{\max}^c = \max\{ LB_1^c + \alpha |L_c^I - L_c^- - L_c^0|, LB_2^c + \beta |M_c^I - M_c^- - M_c^0| \} \, \forall c \quad \text{(2x-a)}$$

$$D_c = 0 \forall c \quad \text{(3x)}$$

$$W(1 - y_c) \geq \mathrm{C}_{\max}^c + D_c - H \; \forall c \quad \text{(4)}$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}}} \right\rceil \leq \max\{ LB_1^c, LB_2^c \} \;\; \forall c, i \quad \text{(5)}$$

$$(l_i + m_i) * \left\lceil \frac{q_{ci}}{\underline{p_{ci}} - 1} \right\rceil > \max\{ LB_1^c, LB_2^c \} \, \forall c, i \quad \text{(6)}$$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c^I - L_c^-, \sum q_{ci})} + \min_i m_i \quad \text{(7x-a)}$$

$$LB_2^c = \min_i l_i \; + \frac{\sum m_i \times q_{ci}}{\min(M_c^I - M_c^-, \sum q_{ci})} \quad \text{(8x-a)}$$

$$L_c^- < L_c^I, M_c^- < M_c^I, L_c^-, M_c^- \in \mathrm{Z}^+ \cup \{0\}, \underline{p_{ci}} \in \mathrm{Z}^+ , y_c \in \{0,1\} \, \forall c, i \quad \text{(9x-a)}$$

· Objective function

$$\mathrm{z}^{\mathbf{A-RPX}} = \min \textstyle\sum_c \{ w_c * (1 - y_c) * (\mathrm{C}_{\max}^c - H) \} \quad \text{(10)}$$

A-RPX is the same with **RPX** except that the original decision variables (L_c, M_c) are substituted with the shortage from the ideal number of equipment, $\sum_c (L_c^I - L_c^-)$, $\sum_c (M_c^I - M_c^-)$, and the domain reduction by **Proposition 5.5** and **Proposition 5.6** are apparently expressed in (9x-a).

Proposition 5.7 $\mathrm{z}^{*\mathbf{A-RPX}} = \mathrm{z}^{*\mathbf{RPX}}$

Proof. The solution space of **RPX** is definitely larger than that of **A-RPX**. Therefore, it is enough to show that every optimal solution in **RPX** is convertible to the solution in **A-RPX**, without the decrease of the objective value. Support the optimal solution in **RPX** is L_c^*, $M_c^* \forall$c. For any c, if $L_c^* - L_c^I = a \leq 0$, it can be converted to $L_c^- = -a$, and $L_c^* = L_c^I - L_c^-$ holds. Otherwise if $L_c^* - L_c^I = a > 0$, then $L_c^* = L_c^I + a$ and the alternative solution L'_c in that L_c^* is substituted with $L'_c = L_c^I$ have still the same objective value by **Proposition 5.1**, then again $L'_c - L_c^I = a \leq 0$ holds and converted to the solution in **A-RPX**. And the converted solution is a feasible with the same objective values, since the objective functions of the two problems are same, that is what was going to be shown. ■

Remember that in **RP** and **RPX**, the problem is to distribute the total number equipment among the cells. In **A-RPX**, however, the problem is to distribute the total shortage $\sum(L_c^I - L_c^0)$, $\sum(M_c^I - M_c^0)$ among the cells.

This alternative model has a different variable domain. From (1') the following is derived:

$$\sum L_c^- = \sum (L_c^I - L_c^0)$$

$$\sum M_c^- = \sum (M_c^I - M_c^0)$$

It is expected that by solving **A-RPX** instead of **RPX**, the domain of decision variables is reduced significantly. In **A-RPX**, the number of possible combination is given:

$$_{\sum(L_c^I - L_c^0)+N-1}C_{N-1} \times {}_{\sum(M_c^I - M_c^0)+N-1}C_{N-1}$$

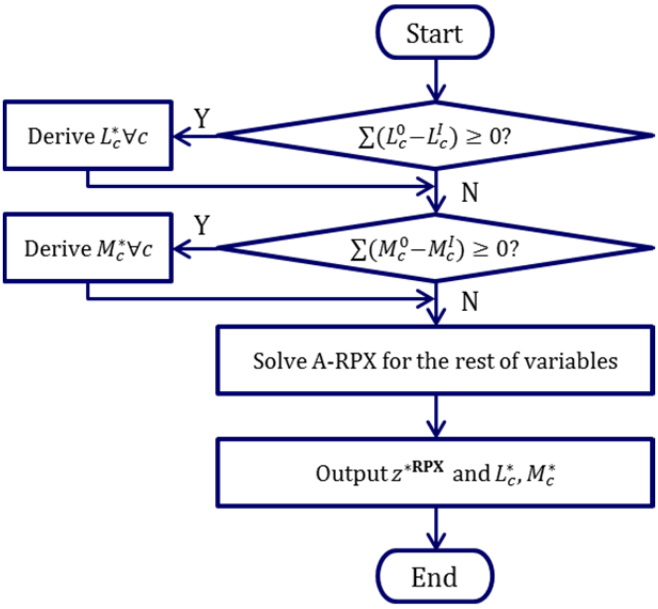
Note that the above equation holds when the total shortage is greater than 0 ($\sum L_c^- = \sum(L_c^I - L_c^0) > 0$, $\sum M_c^- = \sum(M_c^I - M_c^0) > 0$). Otherwise, when the total number equipment exceeds the ideal number, it becomes 1 as discussed in the previous section.

<Figure 5.4> summarizes the solving process of **RPX** to obtain the lower bounds of **RP**.

So far we have discussed about the relaxation model and how it can be efficiently solved by reduction of the dimension and the domain of variables. The proposed relaxation model is expected to provide a tight lower bound to the sub-problem fast so that it may be possible to implicitly enumerate possible combinations using the branch-and-bound algorithm.

Note, however, that the proposed relaxation does not guarantee it will be solved fast almost every time as does the linear relaxation [92]. The complexity may not be reduced compared to the original problem for certain conditions of problems. The efficacy of the proposed relaxation should be verified through an experiment.

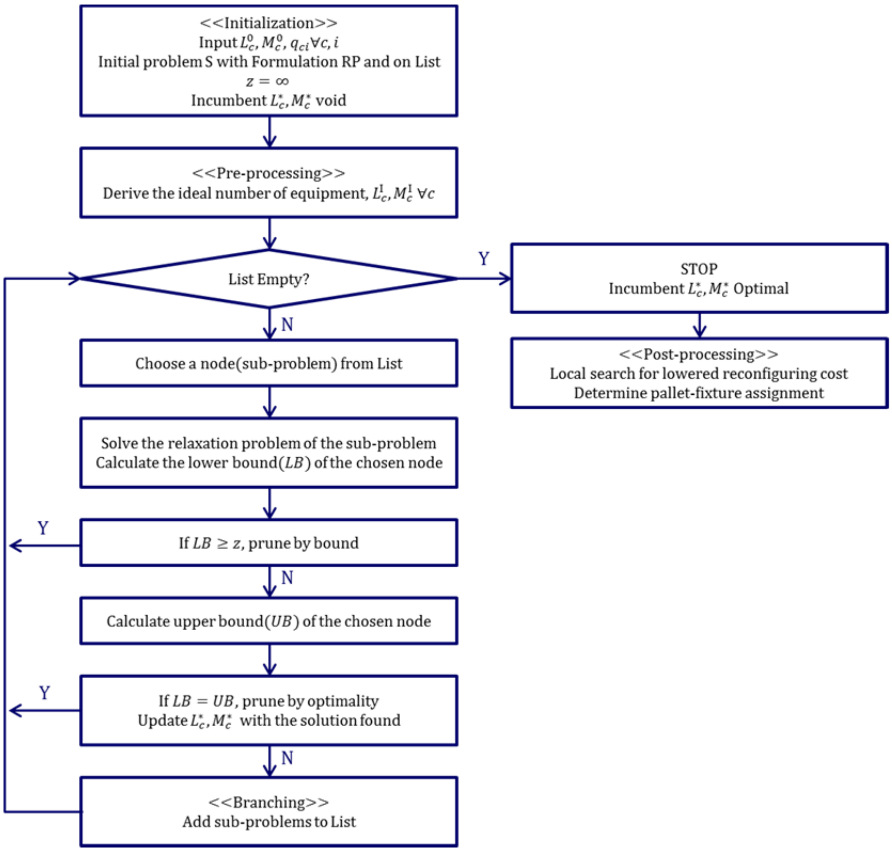
From the next section, the branch-and-bound algorithm using the proposed relaxation model is presented.



<Figure 5.4> The flow chart of RPX algorithm

5.2.2. Relaxation based Branch-and-bound Algorithm

The overall flow chart of the branch-and-bound algorithm is depicted in <Figure 5.5>.



<Figure 5.5> The flow chart of the branch-and-bound algorithm

5.2.2.1. Pre-processing: Derive the Ideal Number of Equipment

Before starting node search, derive the ideal number of equipment using **RPX** as defined in the previous section. From the ideal number, the net shortage is also calculated for each resource type. That is

$$\sum (L_c^I - L_c^0)$$

$$\sum (M_c^I - M_c^0)$$

for L/U stations and NC machines, respectively.

5.2.2.2. Choose a Node

After pre-processing, choose a node for search. Each node in B&B represents a sub-problem where decision variables are partially decided. The starting node is the node with depth 0 in the search tree, indicating the whole problem where none of decision variables is decided. If the depth of the chosen node is $k \leq 2N$ (where N is the number of the cells), this indicates the sub-problem in which k of decision variables are decided, and $2N - k$ variables should be further investigated. When the depth of the node in the search tree is equal to the number of decision variables (i.e. $2N$), this mean that the complete solution is obtained.

5.2.2.3. Calculate the Lower Bound of the Chosen Node

A node represents a sub-problem where decision variables are partially decided. In the sub-problem, a cell may be categorized one of three: i) a cell where both decision variables (L_c, M_c) are decided, ii) a cell where only one of decision variables is decided, and iii) a cell none of decision variables is decided. Denote the set of those cells as A, B, and C, respectively. And without loss of generality, assume that the variable is decided in sequence of L_c, and then M_c.

To calculate the lower bound for cells $c \in B \cup C$, the relaxation problem (**RPX**) of the sub-problem should be solved. However, it may have no advantage to solve **RPX** instead of the original problem (**RP**). Therefore **RPX** is solved only if significant problem space reduction is guaranteed. The criterion to solve **RPX** is like below:

$$\text{Combinations}(\mathbf{RPX}) \leq \mathrm{X} \times \mathrm{Y} \leq {}_{\sum L_c - 1}\mathrm{C}_{\mathrm{N}-1} \times {}_{\sum M_c - 1}\mathrm{C}_{\mathrm{N}-1} = \text{Combinations}(\mathbf{RP})$$

$$\mathrm{X} = \begin{cases} 1, \text{ if} \sum (L_c^0 - L_c^I) \geq 0 \\ {}_{\sum (L_c^I - L_c^0) + N - 1}\mathrm{C}_{\mathrm{N}-1}, \text{ othersiwe} \end{cases} \quad Y = \begin{cases} 1, \ if \sum (M_c^0 - M_c^I) \geq 0 \\ {}_{\sum (M_c^I - M_c^0) + N - 1}C_{N-1}, \ othersiwe \end{cases}$$

Now the lower bound of the sub-problem can be calculated like below:

For $c \in \mathrm{A}$

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c, \ \sum q_{ci})} + \min_i m_i$$

$$LB_2^c = \min_i l_i \ + \frac{\sum m_i \times q_{ci}}{\min(M_c, \ \sum q_{ci})}$$

$$D_c = \alpha |L_c - L_c^0| + \beta |M_c - M_c^0|$$

$$LB_c = \mathrm{T}(\max\{ LB_1^c, LB_2^c\} + D_c - H)$$

where $\mathrm{T(X)}$ is the tardiness function that which outputs X when $\mathrm{X} \geq 0$, and

0, otherwise.

Note that the values of L_c and M_c are known for $c \in$ A.

For $c \in$ B

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c,\ \sum q_{ci})} + \min_i m_i$$

$$LB_2^c = \min_i l_i \ + \frac{\sum m_i \times q_{ci}}{\min(M_c^{*\mathbf{RPX}},\ \sum q_{ci})} + \beta\left|M_c^{*\mathbf{RPX}} - M_c^0\right|$$

$$D_c = \alpha|L_c - L_c^0|$$

$$LB_c = \mathrm{T}(\max\{LB_1^c + D_c, LB_2^c\} - H)$$

where $M_c^{*\mathbf{RPX}}$ is the optimal value obtained by solving the sub-problem with the relaxation model.

Note that the value of L_c is known for $c \in$ B.

For $c \in$ C

$$LB_1^c = \frac{\sum l_i \times q_{ci}}{\min(L_c^{*\mathbf{RPX}},\ \sum q_{ci})} + \min_i m_i \ + \alpha\left|L_c^{*\mathbf{RPX}} - L_c^0\right|$$

$$LB_2^c = \min_i l_i \ + \frac{\sum m_i \times q_{ci}}{\min(M_c^{*\mathbf{RPX}},\ \sum q_{ci})} + \beta \left| M_c^{*\mathbf{RPX}} - M_c^0 \right|$$

$$LB_c = \mathrm{T}(\max\{ LB_1^c, LB_2^c\} - H)$$

where $L_c^{*\mathbf{RPX}}$, $M_c^{*\mathbf{RPX}}$ is the optimal value obtained by solving the sub-problem with the relaxation model.

In the case where the relaxation problem is not solved, set related variables equal to 0. Even if the relaxation is not useful in one sub-problem, it may be useful in another sub-problem inside the chose node. Moreover though relaxation does not work at all, still the lower bound can be calculated for the cells $c \in A$.

Finally the lower bound of the sub-problem is given below:

$$LB = \sum_{c \in A \cup B \cup C} LB_c$$

5.2.2.4. Determine Whether to Prune the Chosen Node

If the calculated lower bound of the chosen node is greater than or equal to the upper bound of the whole problem, there is no reason for further investigation. Therefor prune the node ('prune by bound'). On the other hand, if the lower bound is smaller than the upper bound, there is still possibility that an optimal solution may exist in the sub-tree below the chosen node.

5.2.2.5. Calculate the Upper Bound of the Chosen Node

This stage happens when it is not decided to prune the chosen node.

To calculate the upper bound, the scheduling algorithm should be applied to each cell and therefore the complete solution is required. It is when the depth of the chosen node is equal to the number of variables.

On the other hand, it is also possible to obtain the complete solution of the relaxation problem before the search reaches leaf nodes. And if the upper bound value is the same with the lower bound, it implies that the optimal solution of the relaxation problem is the optimal solution of the original sub-problem, and it needs not to search further into branches of the node.

However, it is hard to expect meet such a pessimistic case, and it takes time to generate the schedules to obtain C_{max}^{c}. Therefore the scheduling algorithm is applied to the special case when the objective value of the relaxation problem equals to 0 excluding cells $c \in A$.

If the calculated upper bound is smaller than the upper bound of the whole problem, it means the corresponding solution is the best found so far. In that case update the value of the best solution and the upper bound with that of the chosen node.

Moreover if the calculated upper bound is the same with the lower bound, it means the derived solution of the chosen node is the best one among the solutions in the sub-tree below the chosen node, and therefore further investigation is not required. In that case prune the node (‘prune by optimality’).

5.2.2.6. Branch for Further Investigation

If no pruning decision has been made in previous stages, and the chosen node is not the leap node, branch the node for further investigation. The number of branches is determined by the remaining number of equipment which is the total number of equipment minus the sum of the value of decision variables above the chosen node.

5.2.2.7. Terminate the algorithm

If there is left no node to investigate, terminate the algorithm and output the best solution and the corresponding objective value.

5.2.2.8. Post-processing

After solving **RP** using the suggested B&B algorithm, the L_c^*, M_c^* ∀c are decided together with $z^{*\mathbf{RP}}$. Though the algorithm considered reconfiguring time (D_c), there may be alternative solutions with the same $z^{*\mathbf{RP}}$ but with lowered reconfiguring cost. To find such alternative solutions, a neighborhood search algorithm is designed of which pseudo code is depicted in <Figure 5.6>. The search is conducted until no improvement can be made.

Algorithm NeighborhoodSearch

$D = \sum D_c(L_c^*, M_c^*)$; //initial reconfiguring cost. Note $L_c^*, M_c^* \forall c$ are given

do

  foreach cell c

    if $L_c^0 < L_c^*$

      $L_c' \leftarrow L_c^* - 1$;

      if $C_{\max}^c(L_c', M_c^*) + D_c(L_c', M_c^*) \le H$ **or**

          $C_{\max}^c(L_c', M_c^*) + D_c(L_c', M_c^*) \le C_{\max}^c(L_c^*, M_c^*) + D_c(L_c^*, M_c^*)$

      //the condition above indicates $z^{*\mathbf{RP}}$ does not decreased with L_c'

        find k such that $L_k^0 > L_k^*$ //at least one exists

        $L_c^* \leftarrow L_c^* - 1$; //this reduces reconfiguring cost, not decreasing $z^{*\mathbf{RP}}$

        $L_k^* \leftarrow L_k^* + 1$; //this also reduces reconfiguring cost, not decreasing $z^{*\mathbf{RP}}$

      endif

    endif

    if $M_c^0 < M_c^*$ //the same for L_c^*

      $M_c' \leftarrow M_c^* - 1$;

      if $C_{\max}^c(L_c^*, M_c') + D_c(L_c^*, M_c') \le H$ **or**

          $C_{\max}^c(L_c^*, M_c') + D_c(L_c^*, M_c') \le C_{\max}^c(L_c^*, M_c^*) + D_c(L_c^*, M_c^*)$

        find k such that $M_k^0 > M_k^*$

        $M_c^* \leftarrow M_c^* - 1$;

        $M_k^* \leftarrow M_k^* + 1$;

      endif

    endif

  endforeach

while $D \ne \sum D_c(L_c^*, M_c^*)$ //until no improvement is made

end do while

<Figure 5.6> Pseudo code of the neighborhood search algorithm

After improved solution is obtained, the value of the pallet-fixture assignment

$\underline{p_{ci}}$ is derived from L_c^*, M_c^*. Remember the value is the minimal number such that the corresponding part type does not cause pallet-bottleneck. Underlying assumption in this approach is that the total number of pallets of the whole factory is enough to satisfy minimal requirement and that it is possible to resolve a temporary pallet-bottleneck case via adjusting pallets.

According to the derived pallet-fixture assignment, it is determined whether it is possible to adjust the current pallet-fixture assignment with pallets inside the cell or additional pallets should be provided. In the latter case, additional pallets are delivered from the cells which have excess pallets above the minimal number, to the cell in lacking.

5.3. Experiment

To verify the performance of the proposed algorithm, an experiment was conducted.

To experiment various shop conditions, three experimental factors were considered: the number of RMCs, total number of L/U stations and total number of NC machines. For an experimental condition, the initial configuration was generated randomly.

Moreover, the unit reconfiguring time was set 480 min. for L/U station ($\alpha = 480$), and 720 min. for NC machine ($\beta = 720$).

The demand period or the available time (H) was set to 200 hour or 12000 min.(8 hour/day × 25days × 60min./hour). The number of parts per each cell was set to 10. The production volume for each part type (q_{ci}) was generated randomly but the initial configuration was reflected so that the following equation holds:

$$\frac{\sum m_i \times q_{ci}}{M_c^0} \cong H \times (1 - \mathrm{UNIF}(-\delta, \delta))$$

where δ is the demand fluctuation factor, set to 0.3 as default in the experiment. $\mathrm{UNIF}(-\delta, \delta)$ is a random variable generated from the uniform distribution with the minimum value of $-\delta$ and the maximum value of δ. Weights of cells were set the same for all cell ($w_c = 1 \forall\, c$). The rest of parameters were generated as in chapter 4.

For benchmarking, the general B&B algorithm was considered which does not utilize the relaxation models to obtain the lower and upper bounds, but uses a simple lower bound scheme.

As performance measures, the number of searched nodes, the search time to reach the optimum and the number of calls of scheduling procedure (CoS) were recorded for both the proposed and general B&B algorithms. Note that the scheduling procedure requires relatively large amount of time and it is desired to reduce the number of calls of it [69].

Finally for each experimental condition, the average performance from 10 replications was recorded.

5.4. Result and Analysis

<Table 5.1> indicates the experimental result. The result showed that through the proposed B&B algorithm, the search effort is significantly reduced compared to the general B&B algorithm. The proposed B&B showed improved performance for all measures - the number of searched nodes and the CoS measure as well as the search time. It was also observed that the proposed B&B appeared to work more efficiently as the problem size gets larger.

Experimental result verifies the efficacy of the relaxation model (**RPX**) and the proposed B&B algorithm based on the relaxation model.

Among performance measures, improvement in search time was relatively smaller than that in the other measures. It was because it takes time to solve the relaxation problem itself. Remember that the relaxation problem is solved almost every time a chosen node (a partial solution) is examined. If this redundancy is eliminated in solving the relaxation problem, more improvement would be made.

<Table 5.1> The experimental result

Cell	*L/U*	*NC*	***Suggested B&B***			***General B&B***			***%Improvement***		
			Searched nodes	Search time(sec.)	CoS	Searched nodes	Search time(sec.)	CoS	Search time	Searched nodes	CoS
4	8	12	706	54	19	2099	70	33	22	66	42
4	8	16	2442	75	36	6153	81	49	8	60	27
4	8	20	5704	156	67	18056	159	75	2	68	10
4	12	12	1105	189	29	7301	268	42	30	85	31
4	12	16	8874	466	54	31363	522	64	11	72	16
4	12	20	4556	270	59	17859	312	83	14	74	29
4	16	12	3545	602	45	24437	746	58	19	85	21
4	16	16	4022	579	46	38036	1043	75	45	89	38
4	16	20	3193	556	62	13100	670	81	17	76	23
6	12	18	1659	53	27	19036	79	59	33	91	55
6	12	24	15884	85	55	434409	132	92	36	96	40
6	12	30	67222	178	92	2266121	297	137	40	97	32

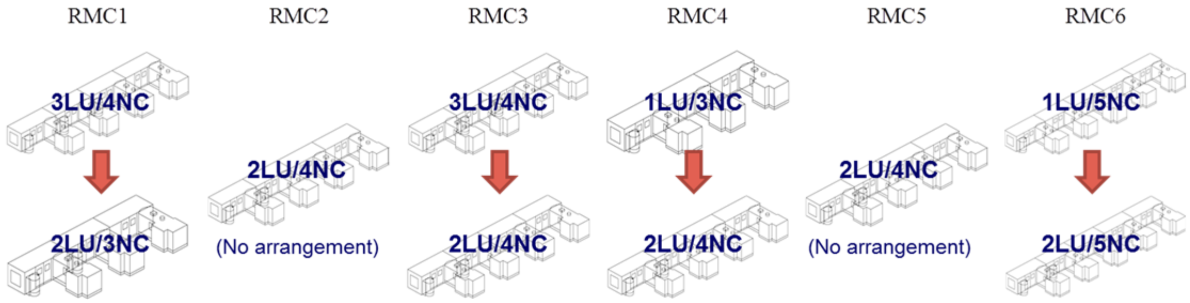
6	18	18	3017	93	24	143870	296	61	69	98	62
6	18	24	4781	175	49	129001	295	95	41	96	49
6	18	30	11547	221	77	1339292	376	147	41	99	48
6	24	18	24882	555	44	1302117	883	67	37	98	34
6	24	24	3282	224	29	81657	930	91	76	96	68
6	24	30	33582	1444	116	7748436	1827	142	21	100	18
10	20	30	10407656	1662	43	98187443	3704	111	55	89	61
								Average(%):	**32**	**86**	**37**

<Table 5.2> and <Figure 5.7>, and <Figure 5.8> show an exemplary result of **RP.** In <Table 5.2>, arrangement of equipment and required time for such arrangement are presented for each RMC, together with change of manufacturing performance (i.e. production delay) before and after hardware reconfiguration. Note that the values about pallets are the minimal number not causing bottleneck. <Figure 5.7> indicates graphical representation of hardware reconfiguration. Change of the schedule with respect to arrange of equipment is depicted in <Figure 5.8>. (Only parts of the schedule are depicted because the length of the entire schedules is too large)

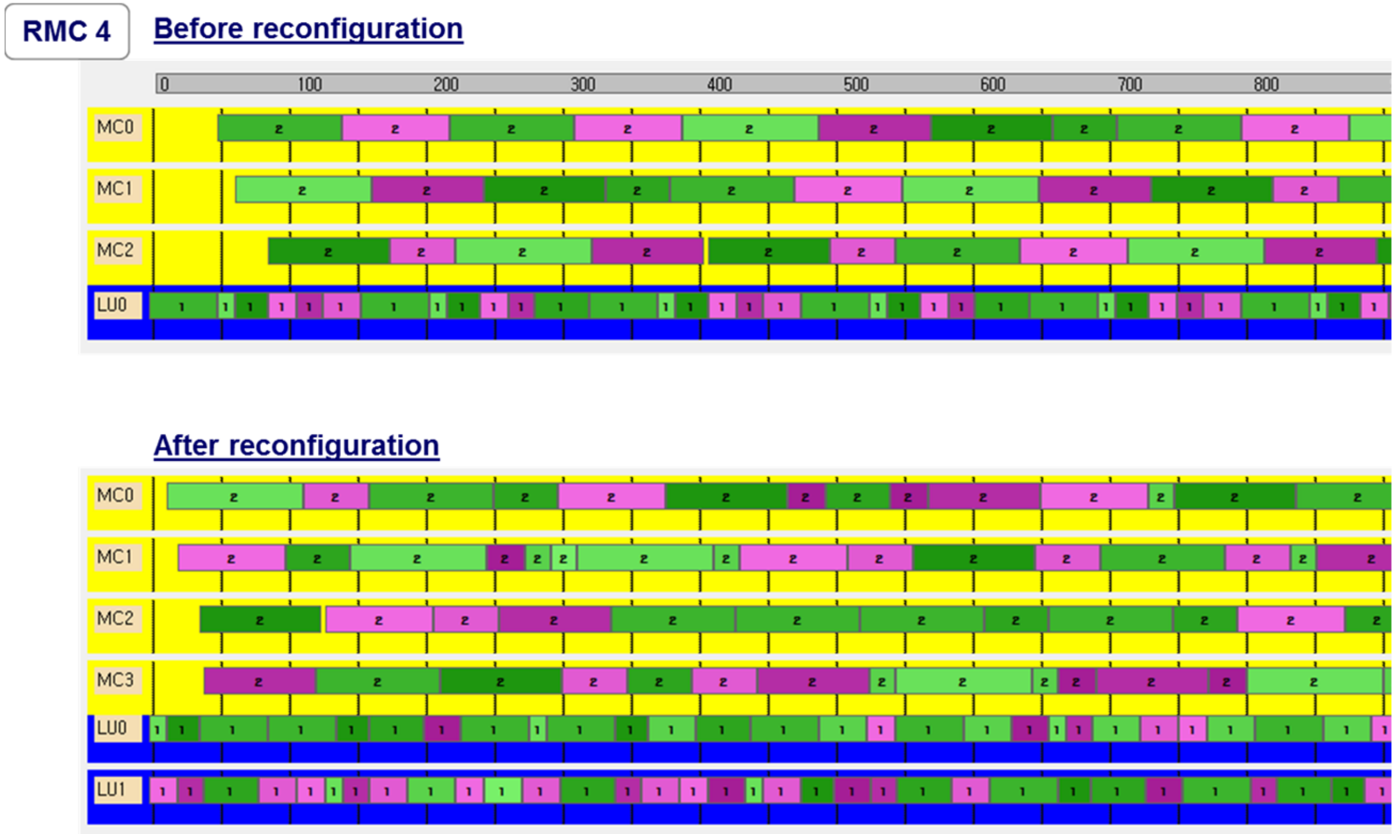
With the proposed algorithm, and derived result, managers may obtain information for hardware reconfiguration and be able to make decisions for response to demand.

<Table 5.2> The result of hardware reconfiguration problem (example)

	RMC1	*RMC2*	*RMC3*	*RMC4*	*RMC5*	*RMC 6*
L/U	3→2	No arrangement	3→2	1→2	No arrangement	1→2
NC	4→3		4→4	3→4		5
PA(min#)	>10	>10	>10	>11	>10	>11
Reconf. time(hour)	20	-	8	20	-	8
Prod. delay(hour)	0→26	0	0→0	124→8	0	240→28



<Figure 5.7> The result of hardware reconfiguration problem in graphical representation (an example)



<Figure 5.8> The result of hardware reconfiguration problem in change of schedules (an example, partially presented)

6. RMC Equipment Purchase Planning

In long term, production managers are concerned with whether the current system may catch up with the demand pattern. If they judge the current system is not enough to cover future demand, then they may consider providing additional capacity by purchasing equipment or outsourcing demand in part. Our research considers the former.

The **reconfigurable equipment purchase problem** (REPP) is defined as: given demand forecast, to find the best purchase plan which specifies whether, which and how many equipment would be brought and when, to catch up with future demand.

REPP is often referred in previous studies as the capacity scalability problem [13][14][15], or the scalability planning problem [91]. Unlike the traditional FMS design problem [83], the capacity scalability problem is concerned with dynamic changes of production capacity along demand periods, including both capacity expansion and reduction. Here capacity reduction indicates decreasing capacity of some parts of a production system while capacity of other parts is increased. This —dynamic changes of production capacity — is enabled via reconfigurable capability of reconfigurable manufacturing systems (RMSs), therefore hardware reconfiguration should be considered together in the resolution of the capacity scalability problem.

Traditionally equipment purchasing decisions were usually made in strategic-level of detail [12][76][89]. However, in RMSs, hardware reconfiguration happens frequently in short term and reconfiguration decisions are made based on operational-level information (i.e. schedule information in our study). Therefore the equipment purchasing in RMS needs to be considered in operational level in relation with hardware reconfiguration. This is more practical as reconfiguring capability of RMS is evolved and time required for hardware reconfiguration is reduced. (The final objective for system-setup (i.e. hardware reconfiguration of existing and purchased equipment) time in the governmental development project for RMCs [42] is one day.)

In this background, this chapter aims to define and resolve the reconfigurable capacity planning problem in operational level of detail. In the next section extended production environment is explained and in the following section a mathematical model is introduced. After that the solution algorithm to find the best purchase plan is suggested. Then an experiment is conducted to verify the suggested algorithm together with analysis of result.

6.1. Mathematical Model

Production environment is basically the same as in chapter 3, but a little bit extended.

A factory consists of multiple production lines and each has an RMC as a part of it. Each RMC is defined by the cell configuration, by the number of L/U stations, NC machines and pallets.

Demand forecast is provided in advance by the marketing department. Demand forecast contains expected demand for multi periods. For each period, production volume is specified along product types. By this way the correlation inside and among product groups would be considered implicitly.

According to demand forecast, equipment purchase is considered. Purchasable components are: L/U stations, NC machines, and pallets. Performance and purchasing cost is assumed identical for components of the same type. And purchased equipment is assumed to be used for production just after introduction without the testing period.

Managerial concern is to cover future demand for entire periods with minimized purchasing cost. Most of studies on the FMS design or the RMS capacity scalability problem set demand satisfaction as a hard constraint and the capital cost as the objective function [15][17][76][89], and the capital cost consists of the investment cost(i.e. purchasing cost) and the operation cost. In this study, the operation cost is assumed relatively small compared to the investment cost, therefore is neglected. This assumption is originated from the fact that target RMCs can rapidly change hardware configurations with small reconfiguration cost which is one of major cost in the operation of an RMS. Besides, information about operations of RMCs is described in chapter 4.

Based on the description of production environment, the following mathematical model for the reconfigurable equipment purchase problem (**REPP**) is built:

REPP

· Input parameters (New or redefined variables only)

t : Index of production period

q_{cit}: Demand forecast of part i of cell c for production period t

L_{c0}^{0} : The initial number of L/U stations of cell c at the starting period

M_{c0}^{0}: The initial number of NC machines of cell c at the starting period

E^{LU}: Unit purchasing cost of L/U station

E^{NC}: Unit purchasing cost of NC machine

ε: The risk-free interest rate

· Decision variables

L_t^{+} : The number of L/U stations which is to be bought at time t

M_t^{+}: The number of NC machines which is to be bought at time t

· Dependent variables

L_{ct}^{0}: The input value of L_{c}^{0} for the problem $\mathbf{RP}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit})$

M_{ct}^{*}: The input value of M_{c}^{0} for the problem $\mathbf{RP}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit})$

L_{ct}^{*}: The optimal value of L_{c} of the problem $\mathbf{RP}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit})$

M_{ct}^{*}: The optimal value of M_{c} of the problem $\mathbf{RP}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit})$

$\mathbf{RP}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit})$ indicates an instance of the reconfiguring problem with input parameters of $L_{c}^{0} = L_{ct}^{0}, M_{c}^{0} = M_{ct}^{0}, q_{ci} = q_{cit}, \sum_{c} L_{c}^{0} = \sum_{c} L_{ct}^{0} + L_{t}^{+}, \sum_{c} M_{c}^{0} = \sum_{c} M_{ct}^{0} + M_{t}^{+}$. Note that the last two conditions are to modify the constraint (1) in **RP** to reflect introduction of purchased equipment.

· Constraints

$\mathrm{z}^{*\mathbf{RP}}(L_{ct}^{0}, M_{ct}^{0}, L_{t}^{+}, M_{t}^{+}, q_{cit}) \leq 0 \forall\, t$ (1)

$L_{\mathrm{ct}}^{0} = L_{\mathrm{c(t-1)}}^{*}, M_{\mathrm{ct}}^{0} = M_{\mathrm{c(t-1)}}^{*} \forall \mathrm{t}$ (2)

$L_{t}^{+}, M_{t}^{+} \in \mathrm{Z}^{+} \cup \{0\} \forall t$ (3)

· Objective function

$$z^{\mathbf{REPP}} = \min \sum_{t} \left(E^{LU} \times \frac{L_{t}^{+}}{(1+\varepsilon)^{t-1}} + E^{NC} \times \frac{M_{t}^{+}}{(1+\varepsilon)^{t-1}} \right) \qquad (4)$$

(1) indicates the production capacity should cover demand forecast of every production periods. (2) indicates the result of hardware reconfiguration of the former period becomes the initial configuration of the following period. (3) indicates the integrality condition. Finally (5) represents the net-present-value

(NPV) of total purchasing cost considering the risk-free rate.

Note that although the above model seems not considering purchase of pallets, the purchase amount of pallets is also derived through **REPP**. Remember the minimal number of pallets are determined through **RP** whenever the number of equipment ($L_{\mathrm{ct}}^{*}, M_{\mathrm{ct}}^{*}$) is decided. Additional pallets need to be purchased when the sum of the minimal pallets exceeds the total pallets of the factory. On the other hand, purchasing cost of pallets is not considered in the objective function, because the unit cost of pallets is considered small compared to that for L/U stations or NC machines.

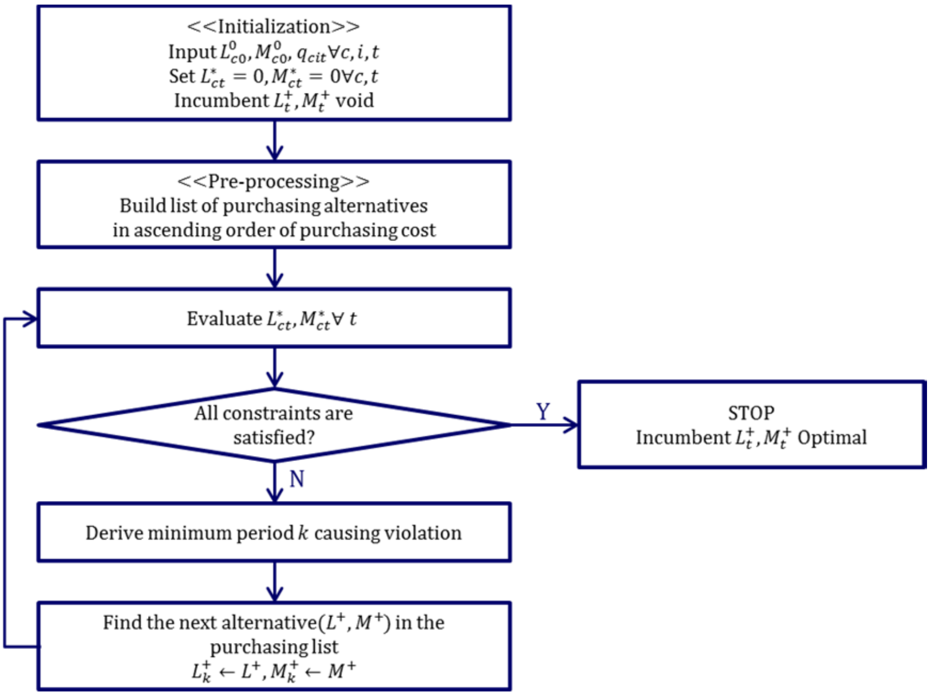
6.2. Development of Algorithm

In this section the algorithm for **REPP** is discussed.

Considering that the objective is cost-minimization, the following facts are derived:

i) It would be better to purchase minimum number of equipment
ii) If it needs to purchase the same number of equipment, it would be better to purchase the cheaper equipment as many as possible
iii) If the number and the type of equipment is the same, It would be better to purchase as late as possible

Considering the facts above, the following algorithm is designed. The flow chart of the algorithm is depicted in <Figure 6.1>.



<Figure 6.1> The flow chart of REPP algorithm

Description of the algorithm is like below:

First, as pre-processing, build the list of purchasing alternatives.

The best purchasing alternative is no purchasing. The next would be to purchase cheaper equipment by one. Like this, the list may be built for all purchasing alternatives, considering the unit purchasing cost of L/U station and NC machine. <Table 6.1> shows the example of such list.

<Table 6.1> An example of the purchasing alternative list

L^+	0	1	0	1	2	0	3	…
M^+	0	0	1	1	0	2	0	…
Purchasing cost ($E^{LU} \times L^+ + E^{NC} \times M^+$)	0	150	200	350	300	400	450	…

Then, stating with the current solution of no purchasing (i.e. $L_t^+ = M_t^+ = 0 \forall t$), evaluate constraints with the current solution.

Constraints would be evaluated by solving **RP** in sequence from earlier periods to later periods. If the current solution does not violate any constraint, then the current solution would be the best solution, since the solution is evaluated ascending order of the objective function.

If some of constraints are violated, then the minimum period t can be derived where the violation happens. To resolve the violation, the current solution L_t^+, M_t^+ should be updated. The values are chosen the first pair from the alternative list excluding selected values.

Through the suggested algorithm, the optimal purchase plan can be derived.

6.3. Experiment

To verify the performance of the **REPP** algorithm, an experiment was conducted.

The experiment was conducted varying the number of cells, total number of L/U stations and total number of NC machines. For given an experimental condition, the initial configuration (L_{c0}^{0} M_{c0}^{0}) of the starting period was generated randomly.

Demand forecast (q_{cit}) is generated for upcoming 12 periods by weighted moving average method from past 6 periods. Each period's demand indicates the expected monthly demand, and past demand is generated randomly.

Moreover, the unit purchasing cost set 100 million KRW for L/U station ($E^{LU} = 100$), and 300 million KRW for NC machine ($E^{NC} = 300$).

The demand period or the available time for each period (H) was set to 200 hour (1 month) in accordance with demand forecast. The number of part types per each cell was set to 10. The rest of parameters were generated as in chapter 4.

As a performance measure, the search time is gathered.

Finally for each experimental condition, the average performance from 5 replications was recorded.

6.4. Result and Analysis

<Table 6.2> shows the experimental result. It took a little longer to solve the equipment purchasing problem than to solve the reconfiguring problem, since it should solve the reconfiguration problem for multi periods and in an iterative manner. But the search time is still acceptable. Note that the result of the problem includes not only the purchase plan which specifies equipment to be purchased, proper time of purchasing, and total purchasing cost, but also reconfiguration plan with the purchased equipment for entire periods, and the corresponding schedules.

<Table 6.2> The experimental result

Cell	***L/U***	***NC***	***Search time (sec.)***
4	8	12	58
4	8	16	105
4	12	12	121
4	12	16	287
6	12	18	104
6	12	24	125
6	18	18	237
6	18	24	181
10	20	30	196
10	20	40	550
10	30	30	1066
10	30	40	910

<Table 6.3>-<Table 6.5> and <Figure 6.2> show an exemplary result of **REPP**. In <Table 6.3>, recommendation for purchase of equipment and hardware reconfiguration with purchased equipment are presented for each production period, and its graphical representation for a single period is depicted in <Figure 6.2>. The reason why it does not require hardware reconfiguration in later periods is that demand forecast is generated by moving average method, so demand is less fluctuated in later periods. Information about purchasing cost is depicted in <Table 6.4>. Benefit of equipment purchase is of improved manufacturing performance (i.e. reduced production delay) which is depicted in <Table 6.5>. Each value in <Table 6.5> indicates estimated production delay (hour) of each RMC. For comparison, manufacturing performance in the case without equipment purchase is also presented. From difference of production delay between two cases (with and without equipment purchase), return-on-investment (ROI) of purchased equipment can be calculated like below:

$$\mathrm{ROI} = \frac{\sum_t \frac{\mathrm{OC}}{(1+\varepsilon)^{t-1}} \times \left(\mathrm{z}^{*\mathbf{RP}}(L_{ct}^0, M_{ct}^0, L_t^+, M_t^+, q_{cit}) - \mathrm{z}^{*\mathbf{RP}}(L_{ct}^0, M_{ct}^0, 0, 0, q_{cit})\right)}{\mathrm{z}^{*\mathbf{REPP}}}$$

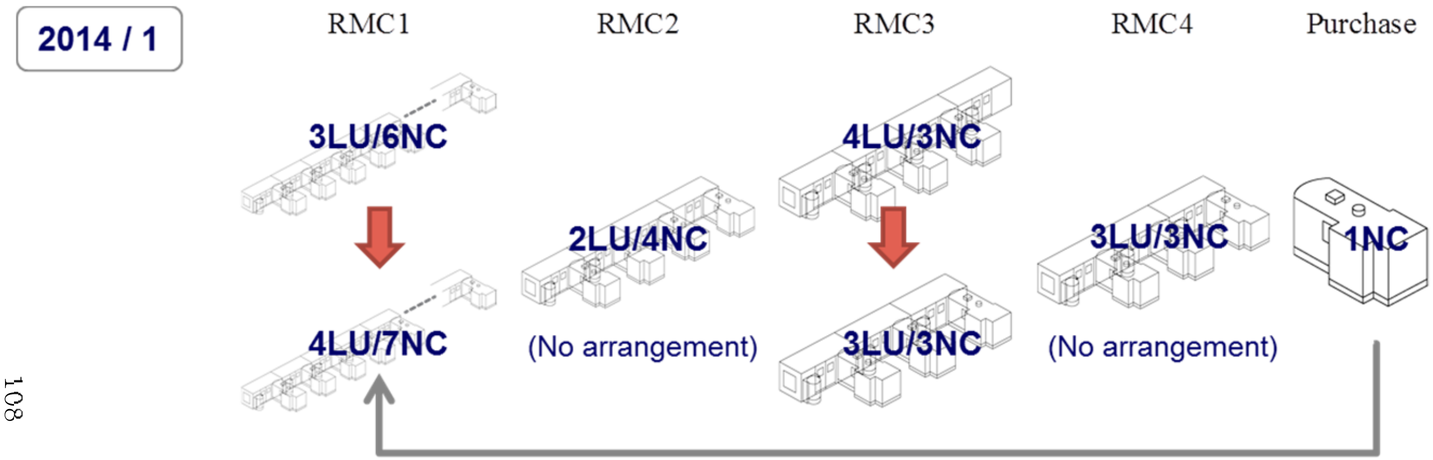
, where OC is the operation cost of an RMC per an hour.

With the proposed algorithm, managers may obtain information for equipment purchase in long-term aspect.

<Table 6.3> The result of reconfigurable equipment purchase problem (an example)

Period (Month)	*Resource type*	*RMC1*	*RMC2*	*RMC3*	*RMC4*	*Purchase*
2014 / 1	L/U	3→4	No arrangement	4→3	No Arr.	0
	NC	6→7		3→3		1
2	L/U	4→3	2→2	3→4	No Arr.	0
	NC	7→7	4→5	3→3		1
3	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
4	L/U	No Arr.	No Arr.	No Arr.	3→3	0
	NC				3→4	1
5	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
6	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
7	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
8	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0
9	L/U	No Arr.	No Arr.	No Arr.	No Arr.	0
	NC					0

10	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0
11	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0
12	L/U NC	No Arr.	No Arr.	No Arr.	No Arr.	0 0



<Figure 6.2> The result of reconfigurable equipment purchase problem in graphical representation (example)

<Table 6.4> Cost of equipment purchase (an example)

Period (Month)	***Purchase - L/U station***	***Purchase - NC machine***	***Purchasing cost (million KRW)***
2014/ 1	-	1	300
2	-	1	299
3	-	-	-
4	-	1	296
5	-	-	-
6	-	-	-
7	-	-	-
8	-	-	-
9	-	-	-
10	-	-	-
11	-	-	-
12	-	-	-
Total	**0**	**3**	**895**

<Table 6.5> Benefit of equipment purchase: Reduced production delay (an example)

Period ***(Month)***	***Equipmenddt purchase***				***No purchase***				***Time saving (hour)***
	RMC1	*RMC2*	*RMC3*	*RMC4*	*RMC1*	*RMC2*	*RMC3*	*RMC4*	
2014/1	0	0	0	0	25	0	0	0	25
2	0	0	0	0	21	1	0	0	22
3	0	0	0	0	22	1	0	0	23
4	0	0	0	0	22	4	0	1	27
5	0	0	0	0	21	8	0	2	31
6	0	0	0	0	21	3	0	0	24
7	0	0	0	0	21	3	0	0	23
8	0	0	0	0	21	3	0	0	24
9	0	0	0	0	21	3	0	0	24
10	0	0	0	0	21	3	0	0	24
11	0	0	0	0	21	3	0	0	24
12	0	0	0	0	21	3	0	0	24
					Average(hour/month):				**25**

7. Conclusion

This study has aimed to develop an integrative decision-making system for RMC environment, a manufacturing system with hardware reconfiguring capability. For reconfiguring capability of RMCs, 1-2 days of reconfiguring time has been considered. The capability is still advancing.

Managerial problems encountered in the operation of RMCs have been defined in chapter 3. In chapter 4, a scheduling algorithm appropriate for those problems has been developed as a basic component of the decision-making system. Using the developed scheduling algorithm, in chapter 5, a mathematical model is suggested to maximize demand fulfillment for the reconfiguring problem. In the model, impact of reconfiguring time on manufacturing performance is considered. For resolution of the reconfiguring problem, a relaxation based branch-and-bound algorithm is proposed, and properties for reduction of the problem dimension and reduction in the domain of variables of the relaxation problem have been investigated. A local (neighborhood) search is also included in the branch-and-bound algorithm to find solutions with reduced reconfiguring cost while maintaining the optimum level of demand fulfillment. Finally, in chapter 6, equipment purchasing is considered to respond to long-term demand forecast. A multi-period model where each period represents a reconfiguring problem is proposed and an exact algorithm for that model is suggested.

For the development of a scheduling algorithm, the lower bound prediction scheme based on work-in-process projection works consistently and effectively for the measure of makespan, with small computational cost. The proposed approach is somewhat compensating the limitation of the greedy (dispatching) approach by looking ahead of the future status of manufacturing systems. Consideration of the lower bound is related to bottleneck (i.e. constrained resource) control.

For the reconfiguring problem, difficulty has come from numerous configurations that should be considered. Some constraints such as the reconfiguring time were obstacles to simplification of the problem. To overcome such a problem structure, the relaxation model has helped eliminate unpromising or redundant alternatives and enabled efficient search. Through the alternative formulation, the benefit has been expressed apparently.

The reconfigurable equipment purchase problem, though it is the extended one of the reconfiguring problem, could been solved plainly (relative to the other problems) relying on the resolution technique of the reconfiguring problem. The reconfiguring problem itself has taken advantage from the scheduling algorithm which is consistent and fast.

Investigation on impact of the pallet-fixture assignment change is remained for the future study. This may be valuable when short-term (e.g. a week) response of production system is concerned, after the hardware configuration is decided. Moreover, in this study, a single type of machining equipment is assumed, since the RMC in development consists of a single type of machines (i.e. horizontal machining centers). To cover more of the real problems, hardware reconfiguration with more than one type of machines such as vertical machines needs to be considered. And this study prohibited alternative material flows among RMCs in different production lines. In real situation, however, some of RMCs may be assigned to process product groups in different lines, while still others may not be used to do so. Finally in this study the required production time (i.e. makespan) is used as the performance measure to evaluate a given hardware configuration. Though the makespan may be a practical measure for production systems where utilization and productivity are important, there are other production systems where due date fulfillment of individual jobs are more important than productivity. Therefore evaluation of manufacturing performance with due date-related measures may help improving the suggested decision-making system.

There exists another challenge. Throughout the study, deterministic demand or demand forecast is considered. In reality, uncertainty exists. Nevertheless, managers should make decisions. Information gathering to resolve uncertainty should be aligned with hardware reconfiguration, toward the objective.

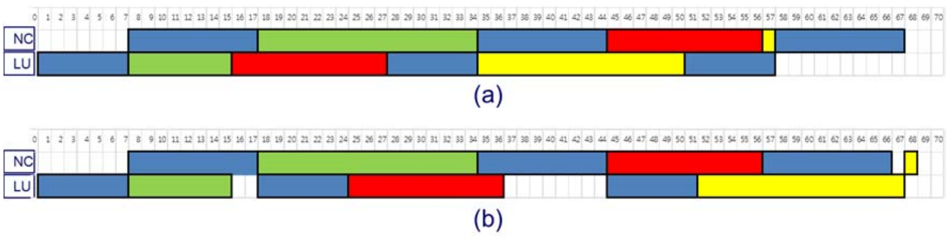
Appendix

Proof for property in section 4.1

Consider a case of 1 L/U station and 1 NC machine and 4 part types with (loading time(l_i), machining time(m_i), production quantity(q_i))-(12,12,1),(7,10,3),(8,17,1),(16,1,1), and that there is only one pallet for each part type (i.e. $p_i = 1 \ \forall \ i$).

For non-permutation schedules is the optimal schedule with the makespan of 67, indicated in <Figure A.0.1> (a).

For permutation schedules is the optimal schedule with the makespan of 68, indicated in <Figure A.0.1> (b). ■



<Figure A.0.1> The optimal schedules:

the non-permutation schedule (a) and the permutation schedule (b)

References

[1] M. R. Abdi and A. W. Labib, “A design strategy for reconfigurable manufacturing systems (RMSs) using analytical hierarchical process (AHP): a case study,” *Int. J. Prod. Res.*, vol. 41, no. 10, pp. 2273-2299, 2003.

[2] M. R. Abdi and A. W. Labib, “Grouping and selecting products: the design key of reconfigurable manufacturing systems (RMSs),” *Int. J. Prod. Res.*, vol. 42, no. 3, pp. 521-546, 2004.

[3] M. R. Abdi, “Selection of a layout configuration for reconfigurable manufacturing systems using the AHP,” in *Int. Symp. Analytic Hierarchical Process*, Honolulu, HI, 2005.

[4] A. Adlemo et al., “Towards a truly flexible manufacturing system,” *Control Eng. Practice*, vol. 3, no. 4, pp. 545-554, 1995.

[5] F. M. Asl and A. G. Ulsoy, “Capacity management via feedback control in reconfigurable manufacturing systems,” in *Proc. Japan-USA Symp. Flexible Automation*, Hiroshima, Japan, 2002.

[6] M. Azizoglu et al., “A flexible flowshop problem with total flow time minimization,” *Euro. J. Opl. Res.*, vol. 132, pp. 528-538, 2001.

[7] K. R. Baker, *Introduction to sequencing and scheduling.* New York, NY: John Wiley and Sons, 1974.

[8] K. R. Baker, *Principles of sequencing and scheduling.* New York, NY: John Wiley and Sons, 2009.

[9] A. Bensmaine et al., “Design of reconfigurable manufacturing systems: optimal machines selection using non dominated sorting genetic algorithm (NSGA-II),” in *Proc. 41st Int. Conf. Computers and Industrial Engineering*, Los Angeles, CA, 2011, pp. 110-115.

[10] D. Chung., “A study on job shop scheduling problems considering production capacity adjustment,” Ph.D.'s Dissertation, Dept. Industrial Eng., Seoul Natl. Univ., Korea, 1999.

[11] D. Chung et al., “Developing a shop floor scheduling and control software for an FMS,” *Comput. Industrial Eng.*, vol. 30, no. 3, pp. 557-568, 1996.

[12] Y. Dallery and Y. Frein, “An efficient method to determine the optimal configuration of a flexible manufectudng system,” *Ann. Ops. Res.*, vol. 15, pp. 207-225, 1988.

[13] A. M. Deif and H. A. ElMaraghy, “Assessing capacity scalability policies in RMS using system dynamics,” *Int. J. Flex. Manuf. Syst.*, vol. 19, pp. 128-150, 2007.

[14] A. M. Deif and W. H. EIMaraghy, “A control approach to explore the dynamics of capacity scalability in reconfigurable manufacturing systems,” *J. Manuf. Syst.*, vol. 25, no. 1, pp. 12-24, 2006.

[15] A. M. Deif and W. H. ElMaraghy, “Investigating optimal capacity scalability scheduling in a reconfigurable manufacturing system,” *Int. J. Adv. Manuf. Technol.*, vol. 32, pp. 557-562, 2007.

[16] D. R. Denzler et al., “An experimental investigation of FMS scheduling rules under uncertainty,” *J. Ops. Manage.*, vol. 7, no. 1-2, pp. 139-151,

1987.

[17] J. Dou et al., “A GA-based approach for optimizing single-part flow-line configurations of RMS,” *J. Intell. Manuf.* vol. 22, pp. 301-317, 2011.

[18] H. A. ElMaraghy, “Flexible and reconfigurable manufacturing systems paradigms,” *Int. J. Flex. Manuf. Syst.*, vol. 17, pp. 261-276, 2006.

[19] R. Galan et al., “A systematic approach for product families formation in reconfigurable manufacturing systems,” *Robotics Computer-Integrated Manuf.*, vol. 23, pp. 489–502, 2007.

[20] P. Gilmore and R. Gomory, “Sequencing a one state-variable machine: a solvable case of the traveling salesman problem,” *Ops. Res.*, vol. 12, no. 5, pp. 655-679, 1964.

[21] A. Guinet et al., “A computational study of heuristics for two-stage flexible flowshops,” *Int. J. Prod. Res.*, vol. 34, no. 5, pp. 1399-1415, 1996.

[22] H. Gultekin, et al., “Scheduling in a three-machine robotic flexible manufacturing cell,” *Comput. Ops. Res.*, vol. 34, no. 3, pp. 2463-2477, 2007.

[23] J. N. D. Gupta, “Hybrid flowshop scheduling problem,” *J. Opl. Res. Soc.*, vol. 39, no. 4, pp. 359-364, 1988.

[24] Y. Gupta et al., “A genetic algorithm-based approach to cell composition and layout design problems,” *Int. J. Prod. Res.*, vol. 34, no. 2, pp. 447-482, 1996.

[25] S. Han et al., “Pallet-fixture allocation in reconfigurable manufacturing cells: an integrative approach,” *J. Korean Soc. Precision Eng.*, vol. 20, pp.

357-366, 2012.

[26] F. Hasan et al., “Optimum configuration selection in reconfigurable manufacturing system involving multiple part families,” *OPSEARCH*, DOI 10.1007/s12597-013-0146-1, 2013.

[27] L. Huang et al., “Configuration selection for reconfigurable manufacturing systems by means of characteristic state space,” *Chinese J. Mech. Eng.*, vol. 23, pp. 1-10, 2010.

[28] IBM Corp.(2009) *IBM ILOG OPL Language User's Manual* [Online]. Available: http://pic.dhe.ibm.com/infocenter/odmeinfo/v3r4/index.jsp?topic=%2Filog.odms.ide.odme.help%2FContent%2FOptimization%2FDocumentation%2FODME%2F_pubskel%2FODME_pubskels%2Fstartall_ODME34_Eclipse991.html.

[29] S. Jang et al., “An integrated decision support system for FMS production planning and scheduling problems,” *Int. J. Manuf. Technol.*, vol. 11, pp. 101-110, 1996.

[30] H. Jeong et al., “A study about integrated supply chain planning model with considering upstream and down-stream negotiation for maximizing profit in open business environment,” *Entrue J. Inform. Technol.*, vol. 11, no. 2, pp. 135-153, 2012.

[31] Y. Koren and M. Shpitalni, “Design of reconfigurable manufacturing systems,” *J. Manuf. Syst.*, vol. 29, pp. 130–141, 2010.

[32] Y. Koren and A. G. Ulsoy. “Reconfigurable manufacturing system having a production capacity method for designing same and method for

changing its production capacity," U.S. Patent 6 349 237, Feb 19, 2002.

[33] Y. Koren and A. G. Ulsoy, "Vision, principles and impact of reconfigurable manufacturing systems," *Powertrain Int.*, vol. 5, no. 3, pp. 14-21, 2002.

[34] Y. Koren et al., "Impact of manufacturing system configuration on performance". *Ann. CIRP*, vol. 47, no. 1, pp. 369-372, 1998.

[35] Y. Koren et al., "Reconfigurable manufacturing systems," *Ann. CIRP*, vol. 48, pp 1-14., 1999.

[36] K. Kumar et al., "Optimal configuration selection for reconfigurable manufacturing system using NSGA II and TOPSIS," *Int. J. Prod. Res.*, vol. 50, no. 15, pp. 4175-4191, 2012.

[37] C. Lee and G. L. Vairaktarakis, "Minimizing makespan in hybrid flowshops," *Ops. Res. Letters*, vol. 16, pp. 149-158, 1994.

[38] D. Lee and Y. Kim, "Part-mix allocation in a hybrid manufacturing system with a flexible manufacturing cell and a conventional jobshop," *Int. J. Prod. Res.*, vol. 34, no. 5, pp. 1347-1360, 1996.

[39] D. Lee and I. Ryo, "A study on scheduling by mixed dispatching rule in flexible manufacturing systems," *J. Soc. Korea Industrial Syst. Eng.*, vol. 47, pp. 35-45, 1998.

[40] D. Lee et al., "A study on integrated supply chain planning in open business environment," *Entrue J. Info. Technol.*, vol. 9, no.2, pp. 123-141, 2010.

[41] G. H. Lee, "Reconfigurability consideration design of components and manufacturing systems," *Int. J. Adv. Manuf. Tech.*, vol. 13, pp. 376-386,

1997.

[42] S. Lee et al., “Development of integrated operation technology for autonomous reconfigurable production system,” in *Proc. 2010 Fall Korean Society of Machine Tool Engineers Conf.*, Korea, 2010, pp. 167-169.

[43] J. Lin and C. Lee, “A petri net-based integrated control and scheduling scheme for flexible manufacturing cells,” *Comput. Integrated Manuf. Syst.*, vol. 10 pp. 143-160, 1997.

[44] J. Liu and B. L. MacCarthy, “The classification of FMS scheduling problems,” *Int. J. Prod. Res.*, vol. 34, no. 3, pp. 647-656, 1996.

[45] P. B Luh et al., “Schedule generation and reconfiguration for parallel machines,” *IEEE Trans. Robot. Autom.,* vol. 6, no. 6, pp. 687-696, 1990.

[46] B. L. MacCarthy and J. Liu, “A new classification scheme for flexible manufacturing systems,” *Int. J. Prod. Res.*, vol. 31, no. 2, pp. 229-309, 1993.

[47] V. Maier-Speredelozzi et al., “Convertibility measures for manufacturing systems,” *Ann CIRP*, vol. 52, no. 1, pp. 367–370, 2003.

[48] H. Makino and T. Trai, “New developments in assembly systems,” *Ann. CIRP*, vol. 43, pp. 501-522, 1994.

[49] M. Mashaei and B. Lennartson, “Energy reduction in a pallet-constrained flow shop through on–off control of idle machines,” *IEEE Trans. Automation Sci. Eng*, vol. 10, no. 1, pp. 45-56, 2013.

[50] M. Mashaei et al., “Optimal number of pallets for reconfigurable cyclic manufacturing plants,” in 14th *IEEE Conf. Emerging Technologies and*

Factory Automation, Mallorca, Spain, 2009, pp. 1-8.

[51] D. T. Matt and E. Rauch, "Design of a network of scalable modular manufacturing systems to support geographically distributed production of mass customized goods," *Procedia CIRP*, vol. 12, pp. 438-443, 2013.

[52] M. G. Mehrabi et al., "Reconfigurable manufacturing systems: key to future manufacturing," *J. Intell. Manuf.*, vol. 11 pp. 403-419, 2000.

[53] M. G. Mehrabi et al., "Reconfigurable manufacturing systems and their enabling technologies," *Int. J. Manuf. Tech. Manage*, vol. 1, no. 1, pp. 114–131, 2000.

[54] M. G. Mehrabi et al., "Trends and perspectives in flexible and reconfigurable manufacturing systems," *J. Intell. Manuf.*, vol. 13, no.2, pp. 135-146, 2002.

[55] X. Meng, "Modeling of reconfigurable manufacturing systems based on colored timed object-oriented petri nets," *J. Manuf. Syst.*, vol. 29, pp. 81-90, 2010.

[56] J. Mun, K. Ryu, and M. Jung, "Self-reconfigurable software architecture: design and implementation," *Comput. Industrial Eng.*, vol. 1, pp. 163–173, 2006.

[57] H. Na et al., "A study on scheduling and rescheduling problem of the FMC during transient disturbance period with pallet constraints," in *Proc. 21st Int. Conf. Production Research*, Stuttgart, Germany, 2011.

[58] X. Nan et al., "Product scheduling and manufacturing line reconfiguration using petri nets and heuristic search," in *Proc. 2007 IEEE Int. Conf. Robotics and Biomimetics*, Sanya, China, 2007, pp.1721-1726.

[59] S. L. Narasimhan and P. M. Mangiameli, "A comparison of sequencing rules for a two-stage hybrid flow shop," *Decision Sci.*, vol. 18, pp. 250-265, 1987.

[60] S. L. Narasimhan and S. S. Panwalkar, "Scheduling in a two-stage manufacturing process," *Int. J. Prod. Res.*, vol. 22, no. 4, pp. 555-564, 1984.

[61] W. E. Newman et al., "Examining the use of dedicated and general purpose pallets in a dedicated flexible manufacturing system," *Int. J. Prod. Res.*, vol. 29, no. 10, pp. 2117-2133, 1991.

[62] J. Park et al., "RMC scheduling considering setup and pallet constraint," in *Proc. 2010 Joint Conf. Korean Institute of Industrial Engineering and Operations Research and Management Science Society*, Korea, 2010, pp. 1087-1093.

[63] J. Park and S. Woo, "The robustness of queuing network models in FMS production plans," *J. Korea Soc. Simulation*, vol. 12, pp. 48-54, 1992.

[64] M. Penedo, *Scheduling - theory, algorithms, and systems,* 4th ed. NJ: Prentice Hall, 1995.

[65] S. Rahimifard and S. T. Newman, "The application of information systems for the design and operation of flexible machining cells," *J. Intell. Manuf.,* vol. 10, pp. 21-27, 1999.

[66] C. Rajendran and D. Chaudhuri, "A multi-stage parallel-processor flowshop," *Euro. J. Opl. Res.*, vol. 57, pp. 111-122, 1992.

[67] D. L. Santos et al., "Global lower bounds for flow shops with multiple processors," *Euro. J. Opl. Res.*, vol. 80, pp. 112-120, 1995.

[68] J. Seo and J. Park, “Developing a practical machine scheduler for worker-involved systems,” in *Proc. Asia Simulation Conf.* Korea, 2011, pp. 316-325.

[69] J. Seo and J. Park, “An integrative scheduling framework for reconfigurable manufacturing cell,” in *Proc. 2012 Joint Conf. Korean Institute of Industrial Engineering and Operations Research and Management Science Society*, Korea, 2012, pp. 1661-1665.

[70] J. Seo and J. Park, “The effect of enhanced flexibility in the reconfigurable manufacturing cell,” in *Proc. 22nd Int. Conf. Production Research*, Iguassu, Brazil, 2013.

[71] J. Seo et al., “Development of capacity simulator for reconfigurable manufacturing system,” in *Proc. 2010 Spring Conf. Korean Society of Precision Eng.*, Korea, 2010, pp. 251-252.

[72] J. Seo et al., “A solution procedure for integrated supply chain planning problem in open business environment using genetic algorithm,” *Int. J. Adv. Manuf. Technol.*, vol. 62, no .9-12, pp. 1115-1133, 2012.

[73] S. P. Sethi et al., “Minimizing makespan in a pallet-constrained flowshop,” *J. Scheduling*, vol. 2, pp. 115-133, 1999.

[74] S. S. Shah et al., “Reconfigurable logic control using modular FSMs: design, verification, implementation, and integrated error handling,” in *Proc. American Control Conf.*, Anchorage, AK, 2002, vol.5, pp.4153-4158.

[75] S. Shalev-Oren et al., “Analysis of flexible manufacturing systems with priority scheduling: PMVA,” *Ann. Ops. Res.*, vol. 3, pp. 115-139, 1985.

[76] H. Shin et al., “A decision support model for the initial design of FMS,” *Comput. Industrial Eng.*, vol. 33, no. 3-4, pp. 549-552, 1997

[77] A. Slocum, *Precision machine design.* NJ: Prentice Hall, 1992.

[78] P. Solot, “A heuristic method to determine the number of pallets in a flexible manufacturing system with several pallet types,” *Int. J. Flex. Manuf. Syst.*, vol. 2, pp. 191-216, 1990.

[79] P. Solot and J. M. Bastos, “MULTIQ: a queueing model for FMSs with several pallet types,” *J. Opl. Res. Soc.*, vol. 39, no. 9, pp. 811-821, 1988.

[80] P. Spicer and H. J. Carlo, “Integrating reconfiguration cost into the design of multi-period scalable reconfigurable manufacturing systems,” *J. Manuf. Sci. Eng.*, vol. 129, pp. 202-210, 2007.

[81] P. Spicer et al., “Design principles for machining system configurations,” *Ann. CIRP*, vol. 51, no. 1, pp. 275–280, 2002.

[82] C. Sriskandarajah and S.P. Sethi, “Scheduling algorithms for flexible flowshops: worst and average case performance,” *Euro. J. Opl. Res.*, vol. 43, pp. 143-160, 1989.

[83] K. E. Stecke, “Design, planning, scheduling, and control problems of flexible manufacturing systems,” *Ann. Ops. Res.*, vol. 3, pp. 3-12, 1985.

[84] K. E. Stecke and I. Kim, “A study of FMS part type selection approaches for short-term production planning,” *Int. J. Flex. Manuf. Syst.*, vol. 1, pp. 7-29, 1988.

[85] R. Suri and R. R. Hildebrant, “Modelling flexible manufacturing systems using mean-value analysis,” *J. Manuf. Syst.*, vol. 3, no. 1, pp. 27-38, 1984.

[86] L. Tang et al., “Concurrent line-balancing, equipment selection and

throughput analysis for multi-part optimal line design," *Int. J. Manuf. Sci. Prod.*, vol. 6, no. 1-2, pp. 71-81, 2004.

[87] L. Tang et al., "Selection principles on manufacturing system for part family," in *Proc. CIRP 3rd Int. Conf. Reconfigurable Manufacturing*, Ann Arbor, MI, 2005.

[88] L. Tang et al., "Computer-aided reconfiguration planning: an artificial intelligence-based approach," *Trans. ASME*, vol. 6, pp. 230-240, 2006.

[89] U. A. W. Tetzlaff, "A model for the minimum cost configuration problem in flexible manufacturing systems," *Int. J. Flex. Manuf. Sys.*, vol. 7, pp. 127-146,1995.

[90] D. M. Tilbury and S. Kota, "Integrated machine and control design for reconfigurable machine tools," in *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics*, Atlanta, GA, 1999, pp. 629-634.

[91] W. Wang and Y. Koren, "Scalability planning for reconfigurable manufacturing systems," *J. Manuf. Syst.*, vol. 31, pp. 83-91, 2012.

[92] L. A. Wolsey, *Integer Programming.* New York, NY: John Wiley and Sons, 1998.

[93] S. Woo et al., "An integrated approach for loading and scheduling of a flexible manufacturing system," *J. Korean Inst. Industrial Engineers*, vol. 25, no. 3, pp. 298-309, 1999.

[94] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system part1: a framework," *Int. J. Prod. Res.*, vol. 38, no. 10, pp. 2273-2285, 2000.

[95] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing

system part 2: optimal configurations," *Int. J. Prod. Res.*, vol. 38, no. 12, pp. 2829-2842, 2000.

[96] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system part 3: optimal selection policy," *Int. J. Prod. Res.*, vol. 39, no. 4, pp. 747-758, 2001.

[97] Z. Xiaobo et al., "A stochastic model of a reconfigurable manufacturing system - part 4: performance measure," *Int. J. Prod. Res.*, vol. 39, no. 6, pp. 1113-1126, 2001.

[98] Y. Yamada et al., "Layout optimization of manufacturing cells and allocation optimization of transport robots in reconfigurable manufacturing systems using particle swarm optimization," in *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, NV, 2003, vol. 2, pp. 2049-2054.

[99] S. Yang et al., "Modeling and analysis of multi-stage transfer lines with unreliable machines and finite buffers," *Ann. Oper. Res*, vol. 93, pp. 405-421, 2000.

[100] H. Ye and M. Liang, "Simultaneous modular product scheduling and manufacturing cell reconfiguration using a genetic algorithm," *Trans. ASME*, vol. 128, pp. 984-995, 2006.

[101] A. S. Yigit et al., "Optimizing modular product design for reconfigurable manufacturing," *J. Intell. Manuf.* vol. 13, no. 4. pp. 309-16, 2002.

[102] A. M. A. Youssef, "Optimal configuration selection for reconfigurable manufacturing systems," Ph. D. dissertation, Univ. of Windsor, Canada, 2006.

[103] A. M. A. Youssef and H. A. ElMaraghy, “Modelling and optimization of multiple-aspect RMS configurations,” *Int. J. Prod. Res.*, vol. 44, no. 22, pp. 4929-4958, 2006.

[104] A. M. A. Youssef and H. A. ElMaraghy, “Assessment of manufacturing systems reconfiguration smoothness,” *Int. J. Adv. Manuf. Technol.*, vol. 30, pp. 174-193, 2006.

[105] A. M. A. Youssef and H. A. ElMaraghy, “Availability assessment of multi-state manufacturing systems using universal generating function,” *Ann. CIRP*, vol. 55, no. 1, pp. 445–448, 2006.

[106] A. M. A. Youssef and H. A. ElMaraghy, “Optimal configuration selection for reconfigurable manufacturing systems,” *Int. J. Flex. Manuf. Syst.*, vol. 19, pp. 67-106, 2007.

[107] A. M. A. Youssef and H. A. ElMaraghy, “Availability consideration in the optimal selection of multiple-aspect RMS configurations,” *Int. J. Prod. Res.*, vol. 46, no. 21, pp. 5849-5882, 2008.

[108] J. Yu, et al., “Iterative algorithms for part grouping and loading in cellular reconfigurable manufacturing system,” *J. Opl. Res. Soc.*, vol. 63, pp. 1635-1644, 2012.

[109] B. Zheng et al., “Configuration optimization of manufacturing system based resource reconfiguration,” in *Proc. 2nd Int. Conf. Mechanical and Electronics Engineering*, London, UK, 2010, vol. 1, pp. 193-195.

[110] W Zhong et al., “Performance analysis of machining systems with different configurations,” in *Proc. Japan-USA Symp. Flex. Manuf.* Ann Arbor, MI, 2000, pp. 1-8.

초 록

재구성가능 생산셀(Reconfigurable Manufacturing Cell, RMC)은 시장 수요의 변화에 대응하기 위해 하드웨어 구성을 빠르게 변화시킬 수 있는 생산시스템이다. 현재 이러한 재구성에 걸리는 시간은 1-2일 정도이다. RMC의 운영에 있어 관리자는 제조시스템의 재구성 능력을 어떻게 활용할 것인지에 관련하여 두 가지 의사결정문제에 직면하게 된다.

첫 번째는 하드웨어 구성을 변경할 것인지 또 변경한다면 어떻게 변경할 것인지에 관한 문제이다. 하드웨어 구성을 변경하면 수요에 대한 생산 시스템의 성능이 향상될 수 있으나, 의사결정의 신뢰도를 높이기 위해서는 상세한 자원 할당과 재구성 기간 동안 발생하는 생산 능력의 손실에 대한 정확한 평가가 이루어져야 한다. 그러나 하드웨어 재구성에 관한 기존 연구들은 재구성 문제를 운영적 문제가 아닌 전략적 문제로 다루었고 제조시스템의 정확한 성능 평가를 제공하지 못하였다.

두 번째 문제는 장비 구매에 관한 것이다. 관리자는 현재 시스템이 미래의 수요를 충족시킬 수 있을 것인지, 아니면 추가적인 장비 구매가 필요한지 알기 원한다. 이 때 수요를 충족시키면서 구매

비용을 최소화하기 위해서는 제조시스템의 재구성 능력이 함께 고려되어야 한다.

본 연구는 이상의 두 가지 문제를 위한 통합적인 의사결정시스템을 개발하고자 한다. 하드웨어 재구성 문제를 스케줄링 수준에서 해결하기 위해, RMC 환경에 알맞은 스케줄링 알고리즘을 개발한다. 그리고 개발한 스케줄링 알고리즘을 이용하고 재구성 기간 동안의 생산 능력의 손실을 고려하여, 하드웨어 재구성을 위한 수리 모델을 제시한다. 제시한 모델을 풀기 위하여, 완화에 기초한 분기한정 알고리즘을 제안한다. 마지막으로 장비 구매 문제를 해결하기 위하여 구매 비용의 현재가치를 최소화하는 알고리즘을 제안한다.

주요어: Reconfigurable Manufacturing Cell, Scheduling, Hardware Reconfiguration, Algorithm
학 번: 2008-21225