공학석사 학위논문

A Study on RFID User Memory Applications for Production Management and Scheduling

RFID User Memory를 통한 생산 관리 및 일정 계획 수립 활용 방안에 대한 연구

2016년 8월

서울대학교 대학원

산업공학 전공

김 경 민

A Study on RFID User Memory Applications for Production Management and Scheduling

지도교수 박 진 우

이 논문을 공학석사 학위논문으로 제출함
2016년 5 월

서울대학교 대학원
산업공학과 전공
김 경 민

김경민의 석사 학위논문을 인준함
2016년 6 월

위 원 장 ____________________ (인)

부위원장 ____________________ (인)

위 원 ____________________ (인)

Abstract

A Study on RFID User Memory Applications for Production Management and Scheduling

Kyoungmin Kim
Dept. of Industrial Engineering
The Graduate School
Seoul National University

Industrial products nowadays consist of tens of thousands of parts, and the corresponding supply chains for these products are constituted by large numbers of component suppliers. In these complex supply chains, the use of radio-frequency identification (RFID) can be a viable method to collect real-time shop floor information, including that pertaining to inventory or work progress. However, little to no research to date has been conducted on the use of RFID-based real-time information systems in traditional scheduling problems, although such technologies offer considerable benefit. The RFID tag data standard has recently been developed to set guidelines for the user memory bank for storing or sharing business data on tag user memory.

In this study, the thesis propose a real-time response scheduling system that utilizes this new RFID technology. Proposed system include tag based data structure, pre-processing of data and production scheduling algorithm. This research show that quicker response scheduling solutions are available, and verify the efficiency of the proposed solution through a simulation.

keywords : RFID, Tag user memory bank, Manufacturing system, Production scheduling, NEH algorithm
Student Number : 2014 - 22635

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Background and motivation

Many products in the current manufacturing industry have a complex structure composed of tens of thousands of parts, where this structure is accompanied by a similarly complex supply chain (K. Kim & Park, 2015a, 2015b; Na & Park, 2014). Two representative examples are the automobile and the heavy machinery industries. The automobile industry in particular faces many problems while managing products received from different vendors, along with the complex supply chain for numerous types of parts (Jo & Cho, 2012).

Therefore, it has always been important to maintain a minimum inventory and a high utilization rate in any given manufacturing plant. Accordingly, many companies have developed or adopted production systems, such as JIT (Just in Time) and JIS (Just in Sequence), in order to build effective production systems that deliver at the appropriate time (Duplaga, Hahn, & Hur, 1996; Wagner & Silveira-Camargos, 2011). Since the unified PULL and PUSH system began to take root on production sites, the importance of production planning and sequencing has increased (Åhlström, 1998). MRP(Material Requirement Planning) is used for the calculation and ordering of components required for poduction in many companies. MRP accounts for a large part of the total cost of the enterprise. It means that production planning and sequencing is highly related with production cost of enterprise.

However, it is difficult to create a production schedule that reflects and satisfies the many constraints of a manufacturing system containing multiple parts and modules. Moreover, the amount of data generated nowadays at the enterprise and the plant is ever increasing due to advancements in technology and the increasing complexity of production systems. However, it is difficult to immediately use this data for problem solving or improvement (Zhong, et al., 2015). Consequently, the previously used function for the collection and analysis of the data on the production site needs to be transformed into a system that can promptly aid the manager's decision for production-line control through the collected data.

Therefore, this study propose a production management system for prompt decision making on the production site. Proposed scheduling algorithm uses onsite real-time information from RFID (radio-frequency identification) in the permutation-flow shop environment and improves computational efficiency to enhance scheduling performance.

1.2 Research objective and scope

In this paper, we discuss the production system through the synchronization between shop floor and information system to gather information. The production site is composed of 4M (Man, Machine, Material, Method) and information. An RFID contains a unique identifier and information of a part or product. Accordingly, information such as the location and quantity can be found in the information system in real-time. The information system is a system such as ERP(Enterprise Resource Planning), MES(Manufacturing Execution System) that is used primarily to take advantage at the enterprises.

In traditional researches, collecting real-time shop floor production data was used to identify the future production plan or predict the status of the current point in time or a future time. In other words, it was mainly the conducting of numerous studies for real-time monitoring. In this study, the target system will provide a more extended solution to provide a solution for real-time monitoring, for real-time production control. In particular, this study establishes the production schedule in a short time and performs the research on the algorithm to generate a sequence of operations.

Previous studies have found a scheduling solution by rule based techniques or optimization techniques. Heuristic methods and mathematical programming is the typical methodology. There are heuristic methods such as the genetic algorithm, tabu search, simulated annealing, et al. In this paper, it is necessary to obtain an answer within a limited time, and there is a need for consideration of a fact that possesses the consistency of the obtained solution. For these reasons, the rule-based algorithm is applicable to this thesis. The problem is organized by mathematical modeling and this paper compares this model with the obtained optimal solution of the problem in a small size of problem set. Finally, the performance and possibility was conformed with an application to the real case by a simulation experiment of the case study.

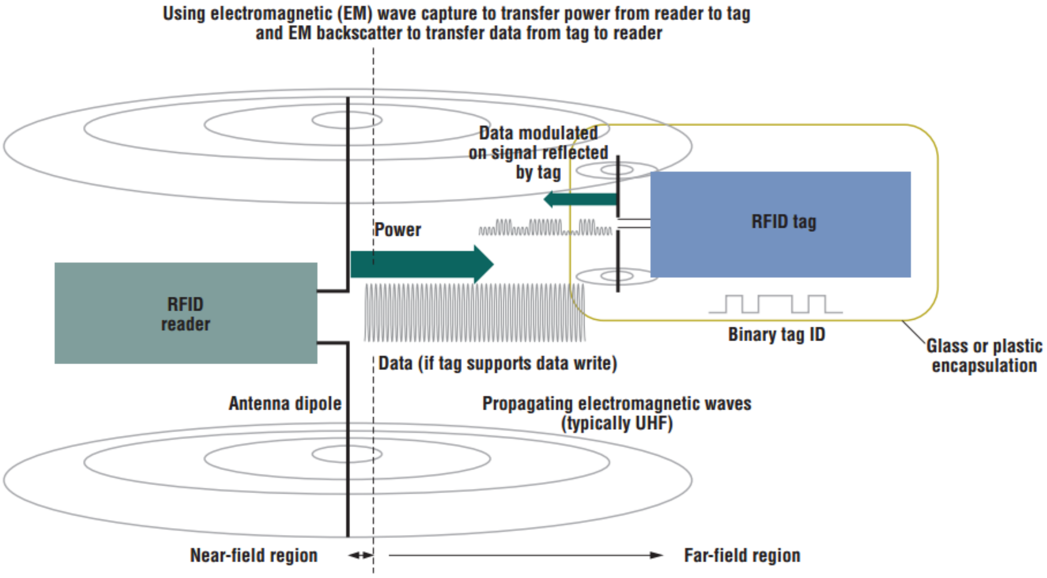
1.3 Structure of the thesis

This thesis is organized as follows: Chapter 2 consists of a review of related work in the area, whereas Chapter 3 is dedicated to a discussion of the problem at hand. In Chapter 4, the thesis discuss the proposed RFID-enabled data architecture and scheduling method. Chapter 5 describe the process and results of a simulation experiment to examine the improvement effected by the proposed system through a case study involving a moderately sized factory. Finally, the conclusion of the thesis is presented in Chapter 6.

2. Related work

2.1 RFID

Radio frequency identification(RFID) is a technology that has the ability to recognize objects from a distance. RFID is combined idea of visual recognition and machine-readable form including tag, reader, middle ware system. RFID tag is consisting of an integrated circuit and an antenna. After the received energy and commands from a Reader, RFID tag transmits the data and reader extracts the data that stored in tag as shown in [Figure 2-1].



[Figure 2-1] Operating mechanism for RFID tags operating at greater than 100 MHz (Want, R. 2006)

Tag can be attached to metal, wood, plastic. because of these reason, the possibility of application is more extended. RFID has the features according to the frequency. Following the frequency of tag, There are low frequency(LF) and high frequency(HF), ultra high frequency(UHF), Microwave. Advantages and disadvantages of each frequency is shown in the [Table 2-1].

[Table 2-1] Advantages and disadvantages of frequency

Frequency of tag	Advantage	Disadvantage	Purpose of use
Low Frequency (125 kHz or 134 kHz) tag	Work well in any weather condition	Short reading distance (1mm~1m)	Livestock tracking
High Frequency (13.56 MHz) tag	Recognizable in liquid or metallic objects around	Short reading distance (30cm~1m)	Transport card
Ultra High Frequency (860~960 MHz) tag	Long reading distance	Difficulty in near the liquid, metal	Manufacturing, Supply chain area

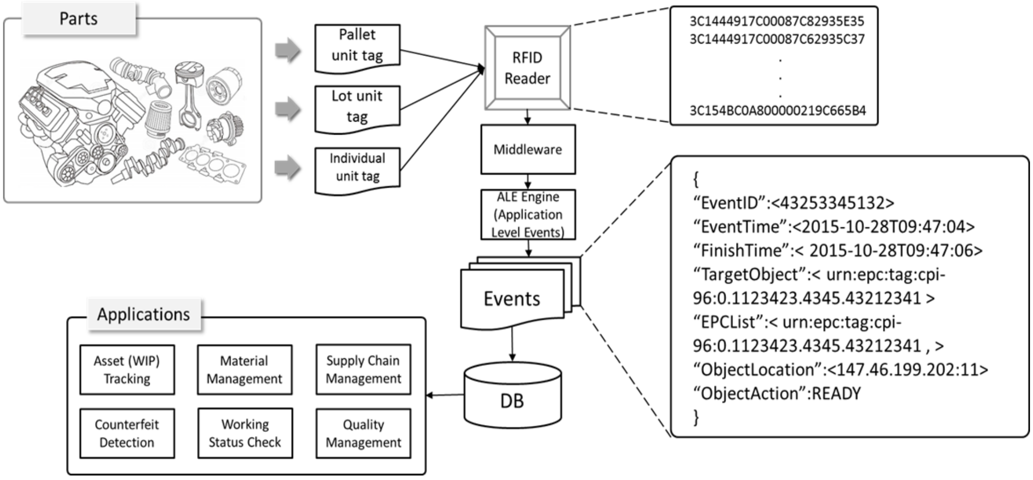
2.1.1 RFID in industries

Following its invention in the 1920s, RFID has been used to carry out the function of part or product identification, originally the function of the bar code. Nowadays, many manufacturing, agricultural, and distribution areas utilize this identification system (Landt, 2005).

RFID is intensively used at all levels of industry from production to distribution (Liukkonen, 2015).

According to recent surveys, the price and size of RFID tags have decreased over time, and it is predicted that future use of RFID will increase in range. In the current supply field, RFID is used to check parts’ stock and tracking records (D. Lee & Park, 2008; Stankovski, Lazarević, Ostojić, Ćosić, & Puric, 2009), and the subsequent productivity and improvement of these systems has been recorded in many studies and examples (Cui, Wang, & Deng, 2014). Recently, in contrast to showing the serial number through RFID and Electronic Product Code (EPC), a new concept called the IoT (Internet of Things) has been proposed (Xu, He, & Li, 2014), which connects everything through a network. With the introduction of IoT-enabled cyber physical system-based connected smart factories (Brettel, Friederichsen, Keller, & Rosenberg, 2014; Jiafu, Hehua, Hui, & Fang, 2011), the possible directions of use of RFID in production have increased as shown in [Figure 2-1].

In a past study by (Zhang, Huang, Sun, & Yang, 2014), RFID was used with a multi-agent-based decision making model to propose the idea of a ubiquitous factory. Moreover, (Yang, Xu, Wong, & Wang, 2015) conducted a study where a model was created to reflect dynamic situations in a production environment for production-related decision making, and included a connectivity structure between machine parts. In many studies, information systems such as ERP (Enterprise Resource Planning) and MES (Manufacturing Execution System) have been used to collect in-time information in order to connect the situation on site.



```
3C1444917C00087C82935E35
3C1444917C00087C62935C37
.
.
.
3C154BC0A800000219C665B4
```

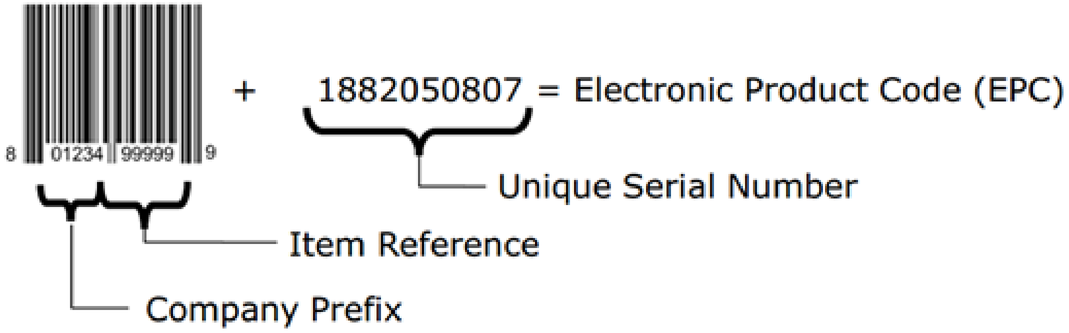
```
{
"EventID":<43253345132>
"EventTime":<2015-10-28T09:47:04>
"FinishTime":< 2015-10-28T09:47:06>
"TargetObject":< urn:epc:tag:cpi-
96:0.1123423.4345.43212341 >
"EPCList":< urn:epc:tag:cpi-
96:0.1123423.4345.43212341 , >
"ObjectLocation":<147.46.199.202:11>
"ObjectAction":READY
}
```

[Figure 2-2] Data transaction process of RFID

A past study used this information to transform the situation on the production floor into an RFID cuboid for easier visualization (Gonzalez, Han, Li, & Klabjan, 2006). Certain studies have conducted simulations and mined tag event data of the parts and machines on site to create a key performance indicators(KPI)(Wang, Liu, & Wang, 2008). RFID has been used to obtain ideas and information from different parts of production in several studies.

2.1.2 Electronic Product Code (EPC)

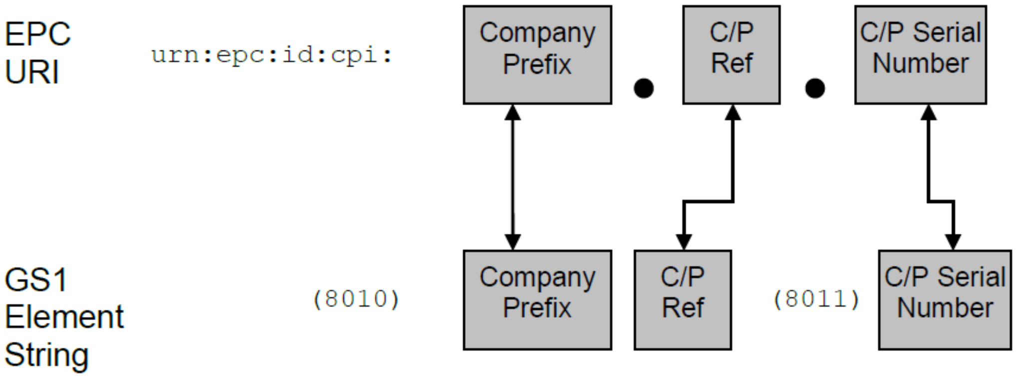
EPC(Electronic Product Code), an international standard has been used as a universal identifier of all items. [Figure 2-2] shows the frame of Electronic Product Code. EPC includes the recognition through the RFID Tag and can be also represented as existing in the form of a bar code, information data. RFID is attached to things that actually exist for a purpose recognized by enclosed identifier assigned to the item. EPC is used for identifying and tracking the exist or flow in the system.



[Figure 2-3] Description of Electronic Product Code (Jürgen Müller, 2013)

The reason for this unique identifier is found in the management of individual objects. Each object or item can be managed the product life cycle with a unique identifier by manager. The international standards organization GS1 offers EPC categories of object through the GS1 identification keys. There is correspondent relationship between EPC and GS1 keys. Through the GS1 key and EPC, it can easily identify the things that belong to any category. GS1 keys are a namespace of identifier for a particular category of real-world product.

Table 2-2 shows the standard of EPC scheme. Each EPC scheme can be used for appropriate purpose of use. Specially, CPI is the option for manufacturing environment with complex part structure. Each part is managed by information system and CPI scheme is assigned to every parts as shown in [Figure 2-3].



[Figure 2-4] Correspondence between CPI EPC URI and GS1 Element String (GS1 EPC Tag Data Standard Version 1.9, 2014)

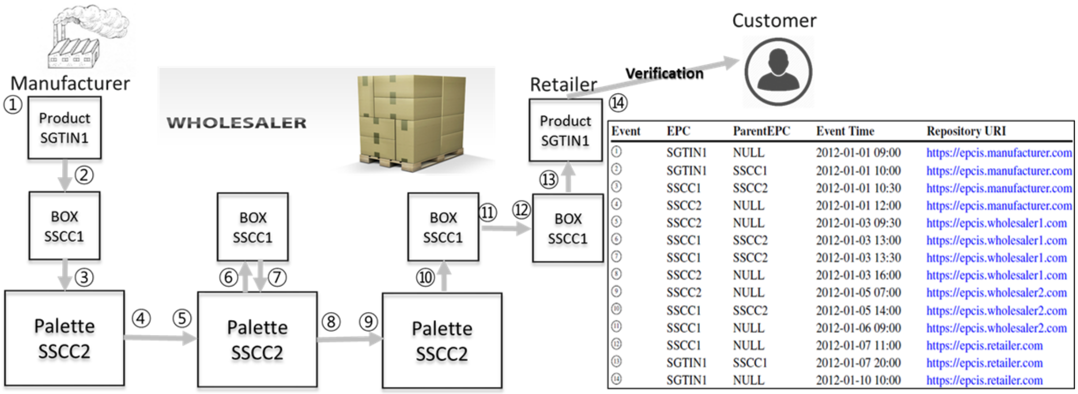
[Table 2-2] EPC scheme (GS1 EPC Tag Data Standard Version 1.9, 2014)

EPC Scheme	Tag Encoding Format	GS1 Key	Purpose of Use
sgtin	sgtin-96 sgtin-198	GTIN key	Item
sscc	sscc-96	SSCC	Pallet or logistics unit
sgln	sgln-96 sgln-195	GLN key	Location
grai	grai-96 grai-170	GRAI	Returnable/reusable asset
giai	giai-96 giai-202	GIA	Fixed asset
gsrn	gsrn-96	GSRN	Hospital or club membership
gsrnp	gsrnp-96	GSRN	Medical or loyalty club
gdti	gdti-96 gdti-112 gdti-174	GDTI	Document
sgcn	sgcn-96	GCN	Coupon
cpi	cpi-96 cpi-var		Manufacturing (Automotive, Electronic, et.al) components/parts
gid	gid-96		Unspecified
usdod	usdod-96		US Dept of Defense supply chain
adi	adi-var		Aerospace and defense - aircraft and other parts and items

2.1.3 Adoption and application of RFID user memory bank

Standardization is one of the major problem for implementing RFID to industry. Following the OECD, EPC standard and RFID tag standard is the ongoing discussion. For the adopting the RFID amd EPC, The RFID EPC a memory bank that can hold part identification information in the tag memory following the GS1 standard (GS1, 2014). The currently used system involves reading the RFID tag filtering by middleware, and saving and recalling these event data and the EPC in and from a database. At the same time, user memory can store business data (item reference, work instructions) obtained from the UHF RFID Tag (Pais & Symonds, 2011).

An advantage of user memory is that it enables reading information tagged at the shop floor without sending a query to the database. Data collected from, say, IoT devices and PLCs (programmable logic controllers) can be stored immediately on the tag user memory, hence increasing the visibility and traceability of the item in the PLM (Product Lifecycle Management) field (Schmidt, Ziemba, & Volkswagen, 2010). [Figure 2-4] shows the supply chain stream of item which attached with RFID tag and EPC. Each process information is stored at the main database of enterprise. Counterfeit detection is easily done with high traceability.



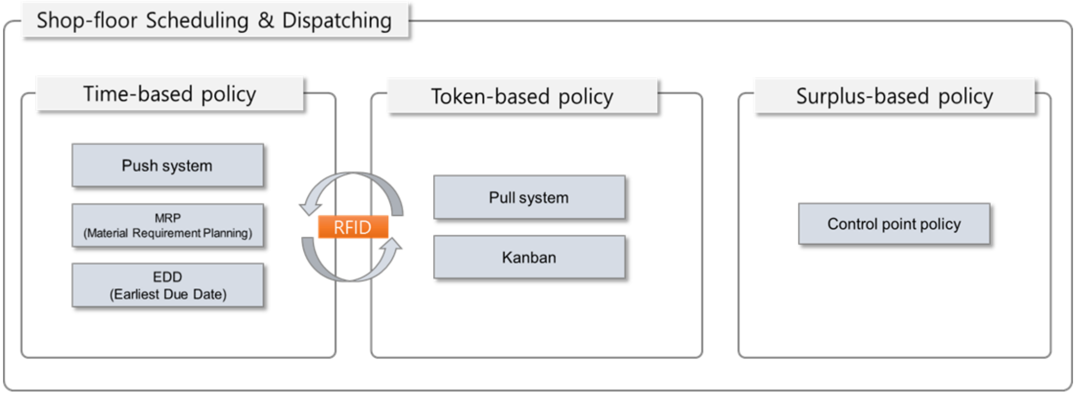
Event	EPC	ParentEPC	Event Time	Repository URI
①	SGTIN1	NULL	2012-01-01 09:00	https://epcis.manufacturer.com
②	SGTIN1	SSCC1	2012-01-01 10:00	https://epcis.manufacturer.com
③	SSCC1	SSCC2	2012-01-01 10:30	https://epcis.manufacturer.com
④	SSCC2	NULL	2012-01-01 12:00	https://epcis.manufacturer.com
⑤	SSCC2	NULL	2012-01-03 09:30	https://epcis.wholesaler1.com
⑥	SSCC1	SSCC2	2012-01-03 13:00	https://epcis.wholesaler1.com
⑦	SSCC1	SSCC2	2012-01-03 13:30	https://epcis.wholesaler1.com
⑧	SSCC2	NULL	2012-01-03 16:00	https://epcis.wholesaler1.com
⑨	SSCC2	NULL	2012-01-05 07:00	https://epcis.wholesaler2.com
⑩	SSCC1	SSCC2	2012-01-05 14:00	https://epcis.wholesaler2.com
⑪	SSCC1	NULL	2012-01-06 09:00	https://epcis.wholesaler2.com
⑫	SSCC1	NULL	2012-01-07 11:00	https://epcis.retailer.com
⑬	SGTIN1	SSCC1	2012-01-07 20:00	https://epcis.retailer.com
⑭	SGTIN1	NULL	2012-01-10 10:00	https://epcis.retailer.com

[Figure 2-5] RFID user memory based product life cycle management application (Jürgen Müller, 2013)

Furthermore, user memory is helpful in improving the visibility of parts for applications such as recycling and counterfeit detection. An information exchange model for close-looped supply chains using RFID tag was recently proposed by (Y. W. Kim & Park, 2014). For the interchange of business data between supply chain participants, GS1 proposes a user memory bank data standard for the RFID UHF tag (GS1, 2014). The change from company-specific standards for the user memory bank to a GS1 standard suggests that an increasing number of fields will make use of user memory bank.

2.2 Production scheduling problem

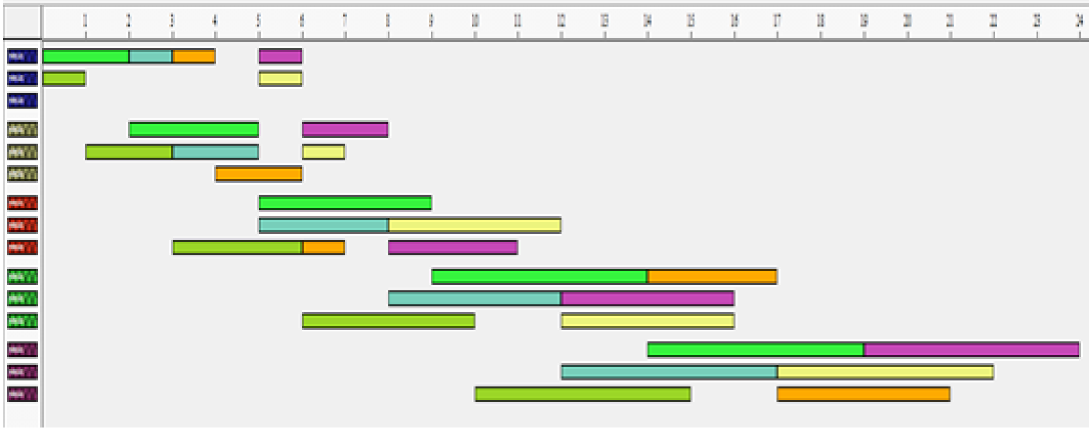
2.2.1 Shop floor production scheduling



[Figure 2-6] Decision policies for production scheduling (Kim, 1983)

Some studies have been conducted about planning the production schedule based on the typical production environment. At the shop floor of a real factory, the material requirements and due date is calculated by the MRP (Material Requirement Planning), and Production schedules are made on the APS (Advanced Planning and Scheduling) module or ERP (Enterprise Resource Planning). Schedules created by this information system are faced with the problem of have not being able to reflect the real-time information of the production shop floor.

Because the real-time scheduling problem is defined in rapidly changing environment, it contains many variables such as WIP (Work in Process) and operation time. Possibilities of failure on the shop floor and uncertainty of inventory stock is also a huge part of the shop floor problem. Therefore, developing a relatively workable schedule, even if it is not the optimal schedule, is important in order for the manager to execute the schedule quickly on the shop floor.



[Figure 2-7] Gantt chart of flow shop scheduling problem

There are three main real-time scheduling methodology suggested for the production site (token-based, time-based, surplus-based policy) as shown in [Figure 2-6]. Time-based methods are mainly divided into two types- MRP, a push system, and dispatching rule- and the most widely-used rules of the latter include SPT (Shortest Processing Time), EDD (Earliest Due Date), and LS (Least Slack). There is Kanban which is a token based policy. However, a token based policies such as Kanban does not follow these time-based criteria when applied to current manufacturing systems.

Many companies use both the Kanban and MRP systems together. Therefore, it is important to proceed with the synchronization between the pull and push system in order to make scheduling and dispatching decisions, representatively units per hour, job allocations for machinery, and sequence of operations.

2.2.2 Real-time production scheduling problem

Many studies have proposed real-time shop floor planning and scheduling systems. Various heuristic rules have been used to solve scheduling problems defined by constraints and production site environments, such as the job shop and the flow shop (Baker & Trietsch, 2013). In real situations, the dispatching rule is used to make a decision regarding sequencing and scheduling by real enterprise information systems (ERP, MES), but the outcome does not perform well in comparison with other methods. Many researches have hence studied heuristics-based production scheduling with RFID. In one such study (Luo, Fang, & Huang, 2015), an RFID-based multi-period hierarchical scheduling model was developed. In another, a genetic algorithm (GA) was used to calculate the schedule based on real-time shop floor information obtained through RFID (Zhang, et al., 2014). Similarly, simulated annealing and tabu search have been used in several studies (Ben-Daya & Al-Fawzan, 1998; Chiang, Cheng, & Fu, 2011; Li & Yin, 2013; Rahimi-Vahed & Mirghorbani, 2007).

Unfortunately, many realistic constraints limit the heuristic method. In the case of the genetic algorithm, the initial population and settings (i.e., mutation) influence the solution (Jun & Park, 2015). Moreover, the random search approach, while reducing the probability that a solution will be neglected in the local optimum, renders the solution unstable. Estimating total completion time is also difficult. Consequently, it is not likely that we would find a solution in real time in an appropriately sized factory due to the risk from the use of the method that interrelates the length of the calculation time and the efficiency of the solution. Because of this, a realistic solution to this problem involves efficiently using real-time information obtained by

RFID.

Therefore, we propose an RFID-based real-time production management and scheduling system, which contains an RFID user memory data architecture to reflect real-time data to production management from the shop floor. Further, we discuss an algorithm that is responsive to RFID tag event data. In the following section, we provide the problem description and discuss our proposed model.

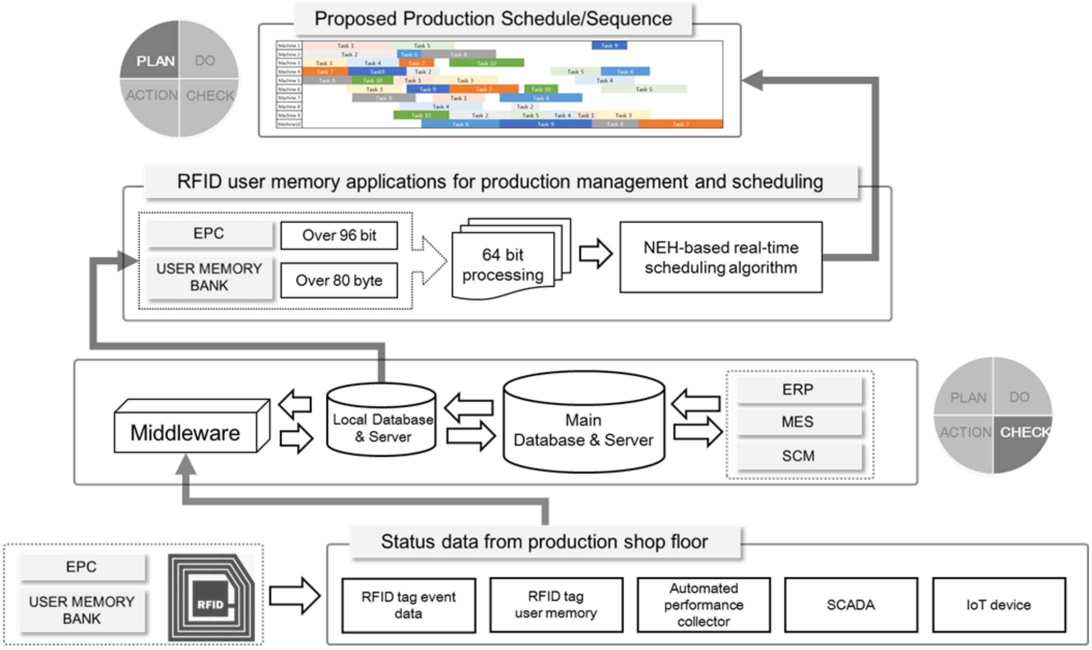
3. Problem description

In the manufacturing industry, due to the large number of subdivisions, it is common for each company to own a customized production scheduling system, rather than adopt a general system or software that can satisfy their needs. Thus, in order to develop such a generalized system, it is necessary to have a flexible manufacturing planning system that can deal with fluctuating factors, such as the manufacturing line environment, supply chain constraints, and the priorities of the established plans (inventory optimization, sales maximization, yield optimization, etc.)

In the case of production lines that have adopted the demand flow manufacturing system covered in this paper, it can be classified as a scheduling problem, including or permutation flow shop scheduling problems. In the same line, it produces a number of products with no large specification differences. In this respect, the production line can be seen as a batch production scheduling problem. Therefore, it needs to determine the sequence of jobs, according to the daily production plan. The quantity and the input time of the parts is determined by a predetermined sequence.

In a real situation, the problem is caused by parts (LOT defects/ supply delays / inventory inspection error) and unpredictable events (mechanical failure), and it will be necessary to change the predetermined plan. If the above situation occurs, it is necessary to establish a re-planning to change the production sequence plan in a short time.

3.1 Model framework



[Figure 3-1] Framework of proposed system

The overall architecture of the proposed system is shown in Figure 3-1. We propose the use of event data and RFID tag user memory as input data for creating the data architecture of production schedules. Through the proposed data architecture and algorithm operations, the shop floor production scheduling solution can be swiftly calculated. This can help the execution monitoring processes at the shop floor level, and aid the production manager's decision making. The framework of the proposed model in this study is a system that can execute the plan and check functions of Deming's PDCA (Plan-Do-Check-Action) cycle to better control and continually improve the production process and the product.

3.2 Assumptions

The following assumptions are made in this paper (Pinedo, 2012):
(1) All jobs are released simultaneously at the start time.
(2) All machines are available at the start time.
(3) Each machine can only process one operation at a time.
(4) Each job is follows deterministic processing time and routing.
(5) All information concerning the machines and the jobs is known in advance.
(6) There are no pre-emptions.
(7) Setup times are sequence independent, and included in the processing time. There are no sequence-dependent setup times.

3.3 Mathematical modeling

3.3.1 Parameter and variable

The following parameters and decision variables are used:

I	number of unscheduled jobs
K	number of stages
i	index of jobs(i=1, 2, 3,..., I)
j	index of stages (j=1, 2, 3..., K)
Π	set of all permutations(schedule)
π	job permutation(schedule), $\pi=\pi_1,\pi_2,\ldots,\pi_n$
j_i	job i
$O_{i,j}$	operation of job i in stage j
$\Phi_{i,j}$	previous operation of job i in stage j
d_i	due date of job i
$p_{i,j}$	processing time of job i in stage j
$s_{i,j}$	start time of job i in stage j
$f_{i,j}$	finish time of job i in stage j
$ent_{i,j}$	RFID tag time stamp of job i on entrance point of stage j
$ext_{i,j}$	RFID tag time stamp of job i on exit point of stage j
$X_{i,j,k}$	1, if operation $O_{i,j}$ is allocated in k-th sequence 0, otherwise
$MS_{j,t}$	1, if stage j is busy on time t 0, otherwise

3.3.2 Objective function

The objective function for $F_m/prmu/C_{\max}$ problem is given by
$Min\,C_{\max}(\pi)$

$$= Min\,(\max_{1 \le t_1 \le t_2 \le n}(\sum_{i=1}^{t_1} p_{\pi(i),1} + \sum_{i=1}^{t_2} p_{\pi(i),2} + \dots + \sum_{i=1}^{n} p_{\pi(i),K}))$$

3.3.3 Constraints

This paper assume the following constraints (Naderi, 2010; Gao. 2011):

$$\forall \pi \in \Pi, C_{\max}(\pi^*) \leq C_{\max}(\pi) = f_{\pi_{n,K}} - s_{\pi_{n,K}} \tag{1}$$

$$C_{\max}(\pi^*) \leq ext_{\pi_{n,K}} - ent_{\pi_{n,K}} \tag{2}$$

$$ent_{i,j} \leq s_{i,j}, \quad \forall\, i, j \tag{3}$$

$$f_{i,j} \leq ext_{i,j}, \quad \forall\, i, j \tag{4}$$

$$s_{i,j} + p_{i,j} \leq f_{i,j}, \quad \forall\, i, j \tag{5}$$

$$f_{i,j} \leq s_{i,j+1}, \quad \forall\, i, j \tag{6}$$

$$f_{i,j} \leq s_{i+1,j}, \quad \forall\, i, j \tag{7}$$

$$\sum_{k=1}^{n} X_{ijk} = 1, \quad \forall\, i, j \tag{8}$$

$$\sum_{t=s_{i,j}}^{f_{i,j}} MS_{jt} = p_{i,j}, \quad \forall\, i, j \tag{9}$$

$$p_{i,j} \geq 0, \quad \forall\, i, j \tag{10}$$

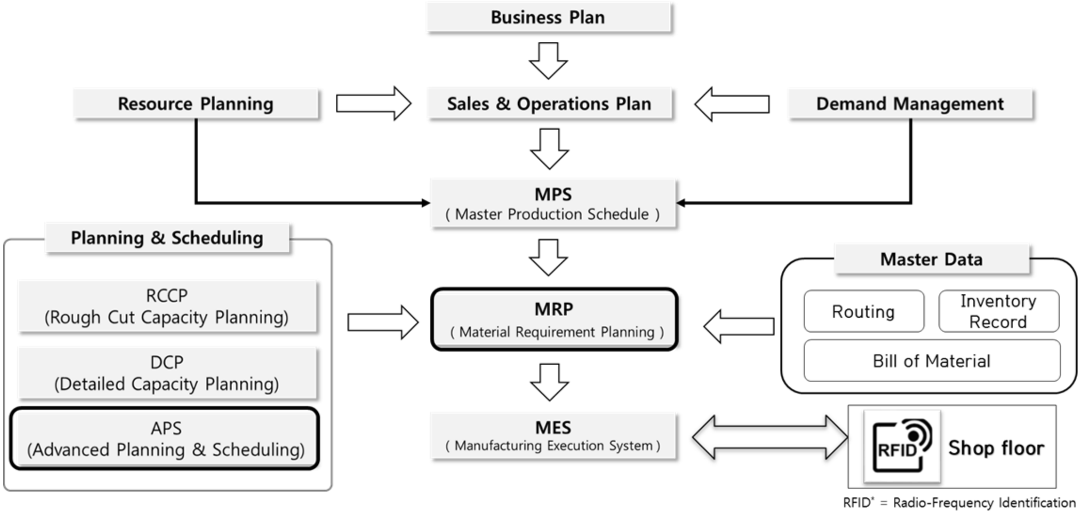
$$s_{i,j} \geq 0, \quad \forall\, i, j \tag{11}$$

$$f_{i,j} \geq 0, \quad \forall\, i, j \tag{12}$$

$$d_i \geq 0, \quad \forall\, i \tag{13}$$

Description of constraints shown in the model are as follows. Constraint (1) and (2) mean the description of the optimal solution. Constraint (3) represents the time for the input of the operation. The time for the exit of the operation is presented in constraint (4) and (5). The Constraint (6) and (7) refer to the constraints that descendent work can start after the preceding job is completed on the stage. Equation (8) and (9) are the assignment problem as discussed in assumption (3). Constraint (10), (11), (12), (13) were restricted to have a processing time, start time, end time, the due date value of negative.

3.4 Problem and model description



[Figure 3-2] Manufacturing planning and control process (Berry, 2005)

The production system handled in this paper is a permutation flow shop problem (PFSP) where demand flow manufacturing occurs. The permutation flow shop scheduling problem with three or more machines has been proven to be NP-hard in many case studies relevant to this problem (Framinan, Gupta, & Leisten, 2004; Nowicki, 1993).

The production line of the automobile industry is a representative example (Ying & Liao, 2004). In this case, the product goes through numerous stages of machining and assembling. In PFSP, each stage involves a machine, and every job follows identical process routes (Kurz & Askin, 2004). Therefore, in the case of n jobs, there are (n!) possible routes.

In the case of a small n, all possible sequences can be calculated. However as the size of n grows, the calculation time expands rapidly, because of which the use of a simple principle, the dispatching rule, or a heuristic method becomes necessary.

Past studies have proved that for a case involving two jobs, Johnson's rule can be applied, and when more than three machines are involved, the Nawaz - Enscore - Ham (NEH) algorithm has proven to be effective (Nawaz, Enscore, & Ham, 1983). It is possible to lower the complexity of computation from the original (n!) cases to (n{n+1))/2-1 cases. Our study is based on the NEH algorithm in order to create a schedule that reflects shop floor data. Results obtained through the proposed algorithm are subject to the comparison process with rule-based methods and algorithms that were used in previous studies. However, meta-heuristic methods, such as the genetic algorithm and ant colony algorithms, were excluded due to time constraints because they were not suitable in covering the current issues.

RFID provides current status data of the part constituting the job in real time on the production site. When immediate resource reallocation or decision making is required, RFID information collected earlier in the necessary decision conditions are used as shown in Figure 3-2. it can be seen as being similar to the process in which the problem is recognized and then reacted to, such as in the dynamic scheduling problem. This research is focused on the fact that information collected by the RFID is being used in the production scheduling problem, in which immediate response is necessary.

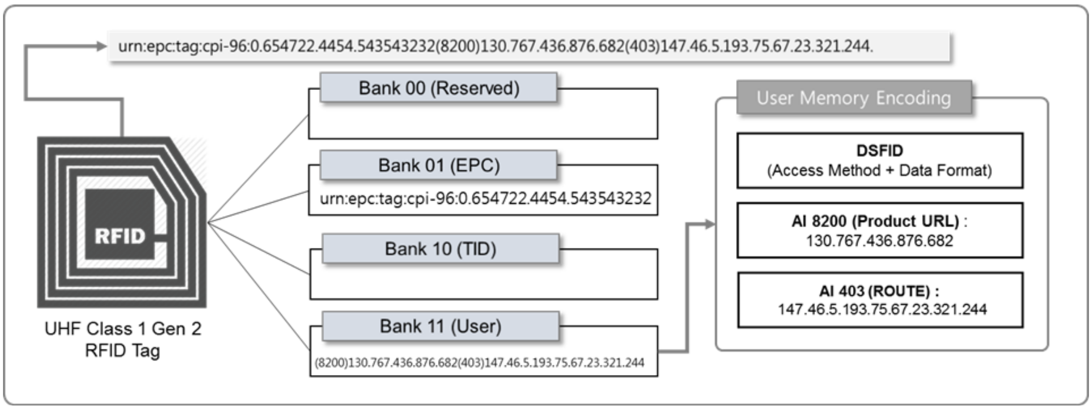
With the data structure integrated in the RFID tag, it is possible to reduce the information query process in the enterprise information system.

4. RFID-based real-time production management and scheduling system

4.1 RFID data architecture for production management and scheduling

Electronic Product Code (EPC), an RFID tag memory bank, is used for scheme and tag encoding according to the intended use in enterprises and industries. The component-part identifier (CPI) was established for manufacturing environments involving many companies and numerous parts and products, such as the aerospace or automobile industry. The CPI can record the company to which a part or a product belongs, the part reference, and its serial number. A tag memory bank, the user memory bank, has the ability to store production master data, and product traceability and business data that are generated during the manufacturing or the distribution process. Due to this ability, it is possible to record the entire product life cycle in successive distribution and maintenance steps. The RFID tag data storage and applications stated above are based on theorems discussed in the following.

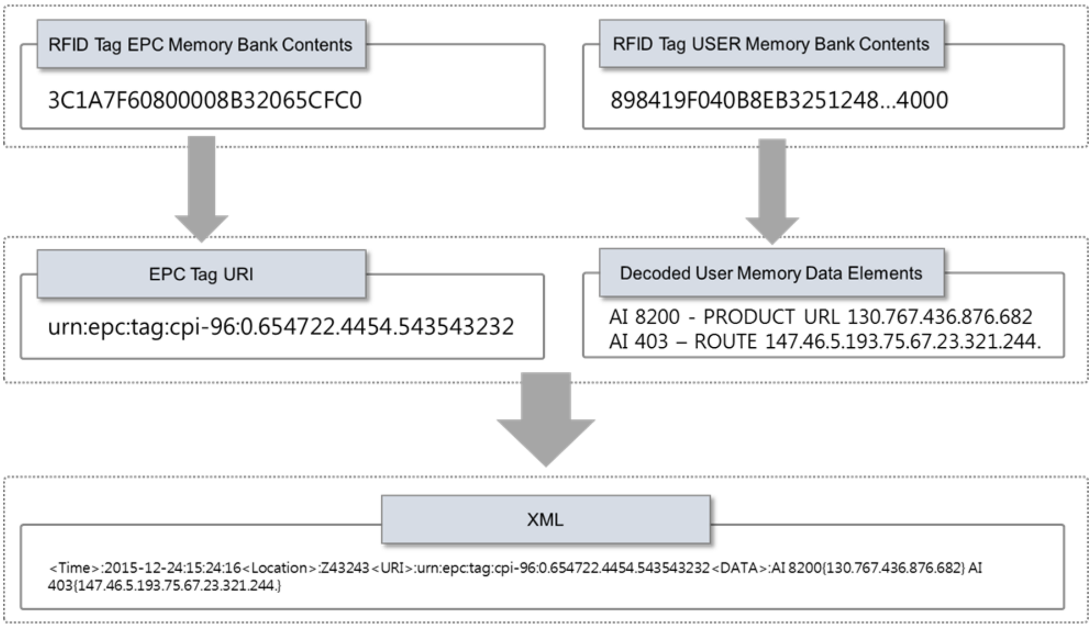
Theorem 1. The RFID tag discussed in our paper is based on UHF Class 1 Gen 2 Tag (ISO 18000-6C Tag), and type of tag is passive. The composition of this tag's memory bank is as shown in [Figure 4-1]. Bank 11 (User memory) consists of the Data Storage Format Identifier (DSFID) in the first eight bits, and the DSFID is composed of logical areas—access method, extended syntax Indicator, and data format (GS1, 2014). The remaining area decodes and codes business data on the basis of the AI (Application Identifier) standard stated in the GS1 Tag Data Standard Version 1.9 or ANSI MH10.8.2 DI (J. S. Lee & Kim, 2006).



[Figure 4-1] RFID UHF Class 1 Gen2 tag memory bank structure

Theorem 2. The tag of the part sensed through the reader on the shop floor has its own unique EPC, and is allocated the according job based on the generated schedule.

Theorem 3. The tag is sensed by the reader, and the tag event data filtered by the middleware is composed of a reading location, tagged time, EPC, and user memory data, as shown in [Figure 4-2].



[Figure 4-2] Example of RFID tag data transaction process

[Table 4-1] Data architecture for production management and scheduling using RFID UHF tag

No.	Data Item	Standard /Unique Data Format (GS1 AI & ANSI MH10.8.2 DI)	Data Transaction Flow
1	Job ID	GS1 Component/Part Identifier (CPI) - Serial	ERP → EPC (RFID tag)
2	Operation information	ANSI DI - 3W (Combined Work Order Number and the Operation Sequence Number)	ERP → RFID tag user memory
2	Part identifier (number)	GS1 Component / Part Identifier (CPI)- Component Part Reference	ERP → RFID tag EPC
3	Setup time (Sequence dependent)	ANSI DI - 3W + Unique format from the company	ERP
4	Processing time		ERP → RFID tag user memory
5	Routing (Job operation data)	AI 403 - ROUTE + Last tagged location	ERP, shop floor tag reader → RFID tag user memory
6	Due date	AI 7003 - Expiration Date and Time (YYMMDDHHMM)	ERP → RFID tag user memory
7	Operation start time Operation finish time	ANSI DI - 18D, 19D (Tag activation time, Tag deactivation time: YYYYMMDDHHMM) AI 8008 - Date and Time of Production	RFID tag user memory → ERP → Shop floor
8	Assigned machine info (Parallel machine case)	ANSI DI - 4W (Status Code)	ERP → RFID tag user memory & PLC

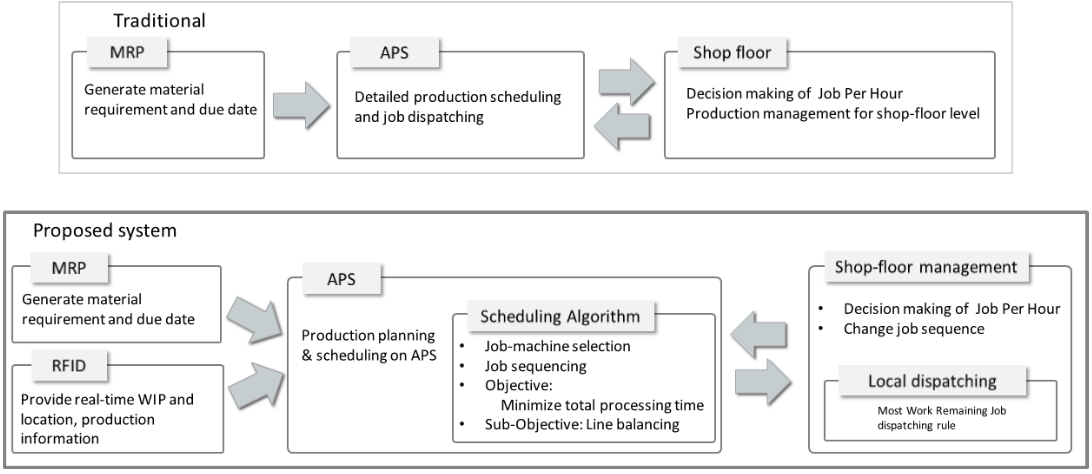
This thesis propose an RFID tag data architecture consisting of the electronic product code and tag user memory (K. Kim & Park, 2015b), as shown in [Table 4-1]. The contents of the table are based on information for real-time production management and scheduling using RFID technology. The table contains the data item name, data format, and the data transaction flow. It consists of existing production data and measurement value at the manufacturing company. Real-time measured data, such as the information of the proceeding job on a specified machine, are not commonly used in traditional production research.

By contrast, this study can to reflect the shop floor status with greater visibility in production scheduling. Moreover, quicker real-time scheduling is possible due to minimized database access and the formatted structure of the data. These two improvements are especially useful while evaluating the feasibility and performance of the proposed algorithm, and results in saved CPU time.

4.2 RFID-based real-time production scheduling algorithm

4.2.1 Data pre-processing for scheduling algorithm

Production planning and scheduling in a manufacturing company is a problem where the global optimal solution cannot be found in limited CPU time. Sometimes, for a small-sized problem or environment, the global optimal solution is searched using a precision method (i.e., an enumeration method, mathematical programing, etc.). In the case of a general manufacturing environment, the constructive method or meta-heuristic algorithms (i.e., Simulated Annealing, Genetic Algorithms, etc.) are popular techniques. However, when we consider constraints such as limited CPU time and other variables of significance in a plant, meta-heuristic algorithms encounter several problems, which have been described in the foregoing section. Therefore, a constructive method is suitable for solving real-time dynamic and complex problems in the manufacturing industry.



[Figure 4-3] Flow chart of the proposed system

Although the constructive method has a shorter computation time than the precision method, the evaluation process has the same duration for both methods, which is the function that takes up the most time in the algorithm. When applied to a real plant, the data transaction process occurs simultaneously in real time with the sending of the query for the stored data of the database. The proposed RFID-enabled data architecture-based data pre-processing procedure for improving the data transaction and evaluation functions is described in the following.

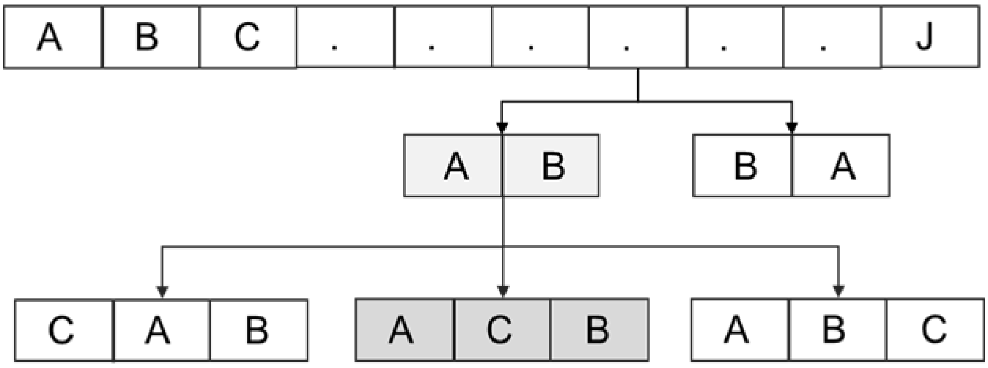
First, in order to immediately reflect the dozen-to-hundreds of jobs and tens of thousands of parts involved in scheduling, it is necessary to use sensed tag data to reformat the generated extensible markup language file into a pre-defined data format. The 64-bit integer format-based data architecture is composed of parts' specification and master data for production, such as the standard processing time of the product and its bill-of-material (BOM) structure.

At the warehouse level, the RFID tag attachment task is conducted, and different jobs are assigned to each part on the production site. The progression of the allocation of part-jobs begins at the point where production seems to be needed. The data belonging to each job is gathered in a job set data, and the pre-processed data is used in the evaluation function and the site monitoring function. In contrast to the conventional route through the database, pre-processed data is used in the evaluation function by being sent directly to the main memory in the workstation, which is in charge of computation tasks. Through this process, although there is a large load incurred in sending the query to the database, it is possible to improve the performance of the evaluation function by using a pre-defined dataset.

4.2.2 Modified NEH algorithm

Our proposed scheduling algorithm is based on the Nawaz - Enscore - Ham (NEH) algorithm, an algorithm that has been shown to be effective (Baskar, 2016; Jin, Song, & Wu, 2007; Liu, Song, & Wu, 2012; Metlicka, Davendra, Hermann, Meier, & Amann, 2014) in searching for initial solutions and finding a permutation sequence that meets the constraints on the permutation flow shop production site.

The modified NEH algorithm shown in [Figure 4-5] reflects the knowledge of possible changes in performance due to the insertion of a job at specific locations that modifies the job insert part of the algorithm in comparison with the traditional NEH algorithm.



[Figure 4-4] Partial scheduling process

Step 1. Align, in descending order, the n confirmed jobs at specific position L during time △t after calculating the total processing time of each job.

Step 2. Take the first two jobs, and schedule them to minimize the makespan.

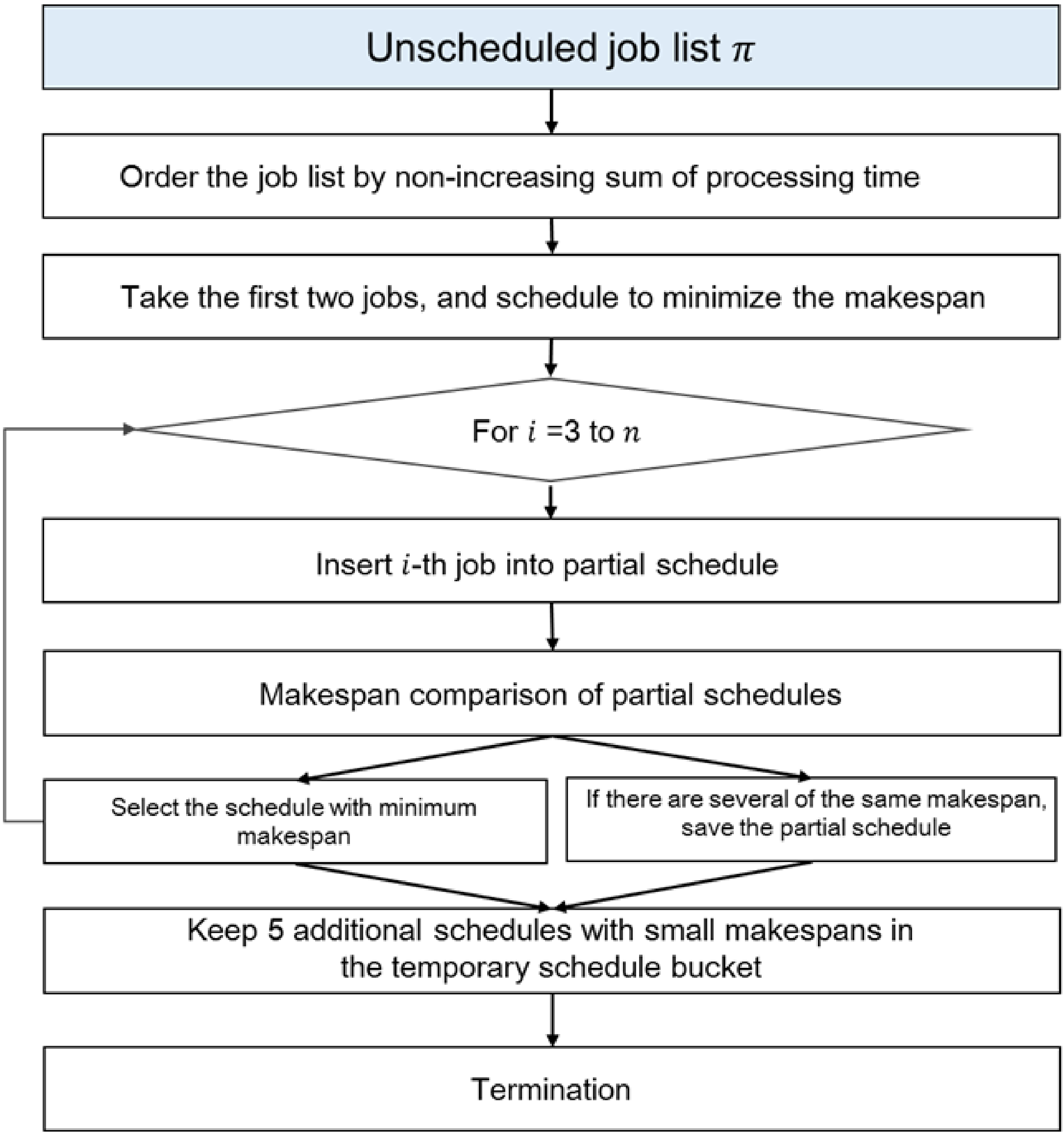
Step 3. For k = 3 to n, execute Step 4.

Step 4. Embed the k-th job into the partial schedule to minimize the makespan among the k possible jobs, as shown in [Figure 4-4]. Unlike in the traditional NEH algorithm, if there are several identical makespans in the partial schedules, save them on the waiting list for the branch.

Step 5. When k = n, choose the schedule with the smallest makespan

When all steps have been completed, in order to explore schedules that satisfy all constraints, we maintain five additional schedules with small makespans in the temporary schedule bucket following Step 5 of the traditional NEH algorithm.

If there is not a single machine per one stage in the flow shop environment, job is assigned to the machine which an empty or considering the start time and end time of the machine using the tag data structure. This machine allocation process can proceed as an additional process in the current job sequencing.



[Figure 4-5] Modified NEH algorithm

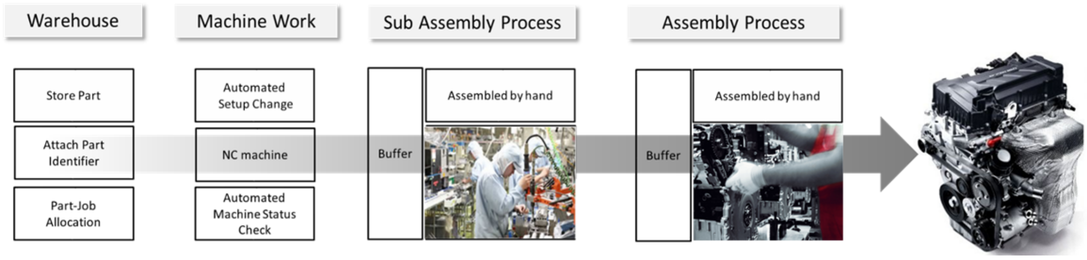
4.2.3 Feasibility check

In this step, the chosen schedule generated by the NEH algorithm is set as the initial schedule, and the sufficiency of the problem constraints is confirmed through a feasibility function. The system searches for schedules satisfying the constraint and, in case the initial solution is unsatisfactory, continues searching the temporary schedule bucket for another solution. If none of the schedules in the temporary schedule bucket satisfies the conditions, the system uses the schedules in the waiting list as source and explores these as options. When a constraint for allocating or prioritizing the sequence of a certain job is demanded, the job is fixed in the sequence, and the remaining jobs are calculated by the NEH algorithm. When a constraint relating to machine suspension (break) time exists, the system can reflect the modified time in the planned schedule by delaying the job.

5. Case Study

In order to verify that the proposed system is useful in solving the shop floor problem, a simulation was conducted involving an automobile engine manufacturing factory. Prior to the experiment, This research conducted a study on the systems used by the company, including the information system, the shop floor-level equipment control and production data collection procedures, and the applications of the factory and the enterprise. Moreover, we completed an analysis of the decision making to establish the production schedule. A real-time shop floor data-based job sequence planning simulation was also conducted and compared to the performance of other algorithms.

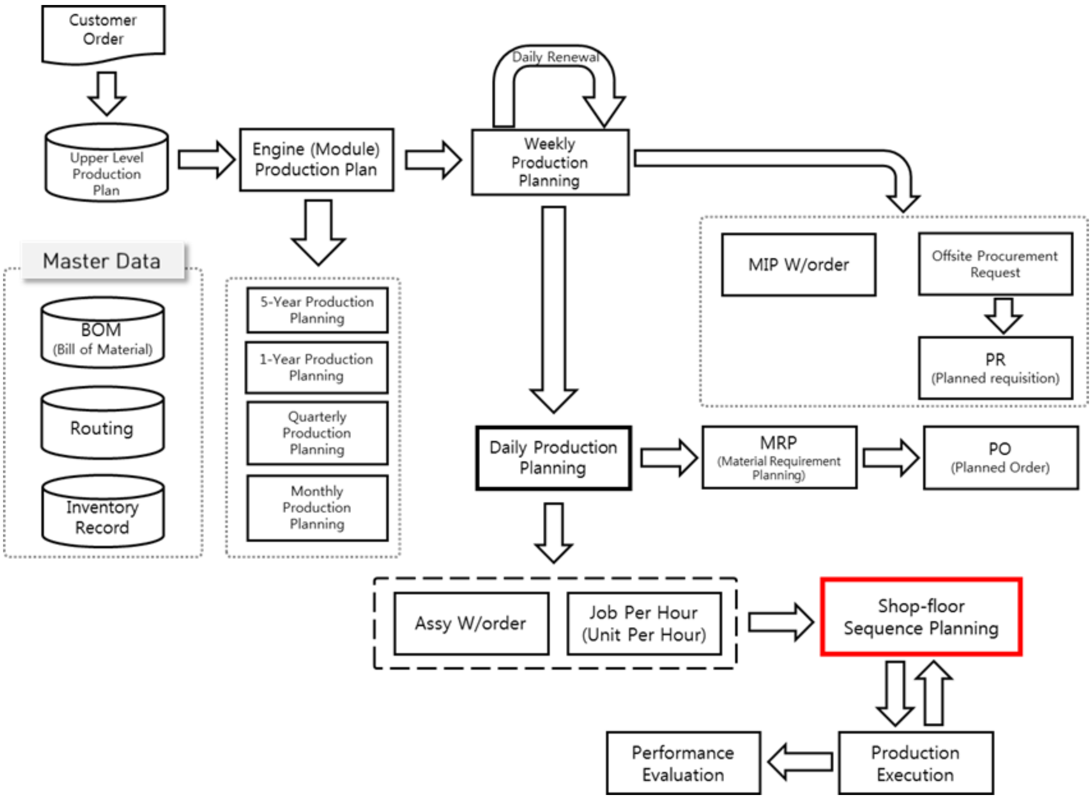
5.1 Introduction to case study



[Figure 5-1] Manufacturing process of automobile engine factory

The factory had adopted a multi-product-type demand flow production system, and operated a multiple production line. At each production line, the machining process and the assembly task were being performed for engine manufacture as shown in Figure 5-1. The factory monitored and controlled the real-time line of parts from start to finish through RFID tags and sensors along the production line. It also recorded the work process, including part picking, receiving, tooling, assembly, and shipping through synchronization with the information system. Furthermore, real-time site monitoring functions, such as machine status and breakdown information, were implementable on the company's ERP software.

Figure 5-2 summarizes the entire decision making process of the factory, from customer order to factory production plan and execution.



[Figure 5-2] Manufacturing planning and control processes in automobile engine factory

In contrast with other functions, such as MRP, shop floor job sequence planning is carried out manually by a human production manager instead of an automated system.

This is because the plant production system was being operated with a combination of the push method of the MRP and the pull method of the demand flow production system. Weekly and daily production planning were carried out at the upper enterprise planning level, and the material requirements and distribution schedules were determined as well. It took approximately 30 minutes to calculate the material requirements for the three subsequent days through MRP, and another daily synchronization was conducted depending on daily fluctuation, including demand or plant status. Accordingly, the MRP decided the planned order and due date. However, following the demand flow production system on the shop floor, the schedule was decided at the time of approval of the production, similar to the process in just-in-time (JIT) systems. As a result, there were no predetermined working schedules or job schedules for the operations of the machines.

Therefore, the problems that might occur during the process in the above figure were as follows: First, the schedule might be being set using a decision based on a mathematical approach. Because of this, the job sequencing problem in the production line might only allocate a task based on the manager's empirical knowledge. Second, there existed a drawback in terms of slow response in case of an unpredictable event requiring immediate response, such as machine breakdown or an unpredicted order.

5.2 Simulation description

A prototype program was developed for the simulation experiment. The program was developed on C# with .NET framework 4.5 and Oracle database 11g, and was run on a PC with Intel i5 (3.20-GHz processor), and a RAM of 8 GB. Moreover, the Lekin scheduler program was used for makespan comparison with other scheduling algorithms.

On the production line, each job sequentially passed through 10 stages. The production process followed the scheduled job sequence and, since it could not simultaneously perform multiple jobs at each stage, the traditional permutation flow shop scheduling problem model was applied. Although the buffer located at the production line was being used for parts of the process to deal with machine breakdown or adjustment of work speed, this buffer did not affect the production planning and scheduling in the system. Moreover, the number of jobs and the due date of the job were decided according to the upper-level plan. The processing time of each stage followed the standard process time measured in advance, and was defined as deterministic time.

In order to verify the performance of our algorithm, an RFID data architecture was created for each part and job and in the program. The method of using the main memory of the workstation by retrieving the required data from the middleware before the evaluation process was used. In order to measure the efficiency of our algorithm, multiple experiments with varying job sizes were conducted. Table 5-1 lists the settings of the experiment. Data from pre-defined set of experiments was conducted based on a randomly selected jobs.

[Table 5-1] Setting of the simulation experiments

Factor	Small size	Medium size	Large size
Number of jobs	10	100	200
Number of stages	10	10	10
Number of composition parts	200	500	1000
Number of experiments	10	10	10

5.3 Evaluation

In the simulation, the schedule for the job set was found in order to effect sequence planning, and a performance evaluation was conducted to compare the effectiveness of our proposed algorithm with other algorithms. Furthermore, in order to compare the solution found by the proposed algorithm with the optimal solution, a feasible solution was tested through existing rule-based dispatching. In rule base dispatching, FIFO(First-in, First-out) has the same meaning as first-come, first-served (FCFS) at the starting point of the line. FIFO is a working policy in which the flow of the manufacturing line is attended to in the order of arrival, without otherbiases or preferences. SPT(Shortest Processing Time) and LPT(Longest Processing Time) are traditional methodologies for line sequencing. They are widely used for comparison with other rules or algorithm in other literatures.

With a small-sized set, the solution obtained through the proposed algorithm was compared with the optimal solution, and was examined through the enumeration method using mathematical programming on C#.

5.4 Results and discussion

The proposed algorithm has provided results that show better speed and ability to search for a feasible solution than the traditional NEH algorithm, as shown in the simulation experiment from [Table 5-2] to [Table 5-9]. Particularly in terms of calculation time, our system generated the production schedule and sequence immediately in the case of small- or medium-sized problems.

[Figure 5-3] shows the CPU times of the algorithm in case of a large-sized problem. Scheduling for the large problem consisting of thousands of jobs was accomplished in a reasonable amount of time. As shown in Table 5-7, the global optimal solution calculated through mathematical programming was similar to the result of the proposed algorithm. This algorithm is useful in situations where obtaining the optimal schedule is not plausible in a limited time due to large problem size. The comparative superiority of the results over the original dispatching rule and its statistical significance were confirmed through multiple simulations.

The main contributions of this paper are as follows: The proposed system contains human-based methods or simple rule-based dispatching on ERP and MES, commonly used in companies today. Our system is a more suitable production management system in generating a realistic schedule for the production site than existing systems. Moreover, re-scheduling by reflecting real-time shop floor information is plausible, and combines the human-based method with the prevalent method for decision making using RFID technology.

It could obtain a production index not only for makespan, but also for WIP and the line utilization rate as a result of the simulation. This part is in accordance with our goal for a real-time production management system. Furthermore, supply chain and production management were improved through the linking of the MRP and shop floor status in a just-in-sequence environment. From the system result, the input time and schedule of the supply part will be obtain for production.

[Table 5-2] Summary of Result(Small size)

Case	Difference from optimal solution	Improvement from SPT	Improvement from FIFO
Case 1	0.31%	8.29%	3.31%
Case 2	0.33%	9.20%	7.55%
Case 3	0.32%	6.38%	4.94%
Case 4	0.00%	8.31%	11.21%
Case 5	0.31%	5.07%	7.56%
Case 6	0.97%	5.78%	4.91%
Case 7	0.64%	11.61%	7.69%
Case 8	0.31%	6.07%	8.71%
Case 9	0.33%	7.58%	5.86%
Case 10	2.24%	4.86%	11.58%
Average	0.576%	7.31%	7.33%

[Table 5-3] Summary of Result(Medium size)

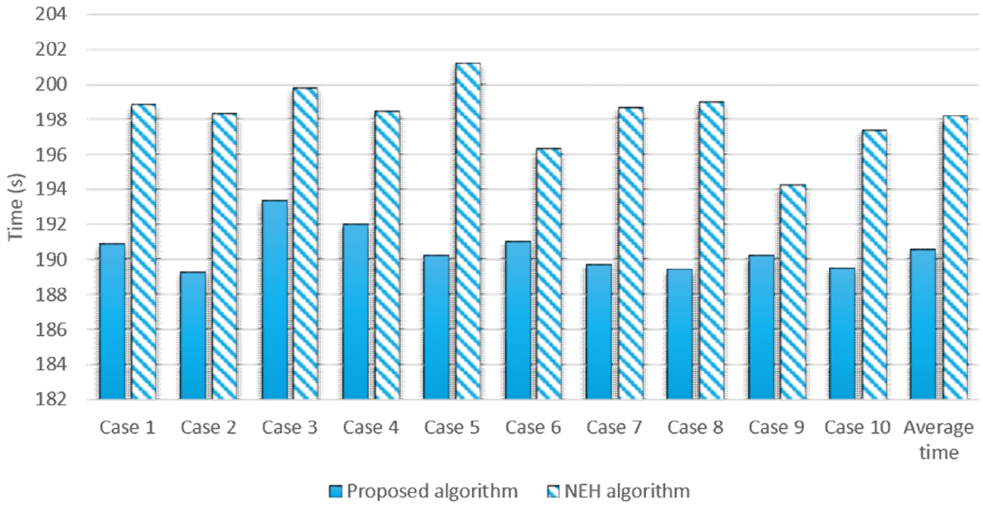
Case	Comparison with NEH algorithm	Improvement from SPT	Improvement from FIFO
Case 1	0.05%	6.39%	6.63%
Case 2	0.11%	5.34%	4.51%
Case 3	0.48%	6.11%	5.44%
Case 4	-0.33%	4.84%	7.05%
Case 5	0.16%	6.48%	4.99%
Case 6	0.00%	5.50%	4.35%
Case 7	0.00%	6.34%	5.43%
Case 8	0.00%	3.40%	4.83%
Case 9	-0.05%	7.01%	4.33%
Case 10	0.22%	6.90%	6.42%
Average	0.06%	5.83%	5.40%

[Table 5-4] Summary of Result(Large size)

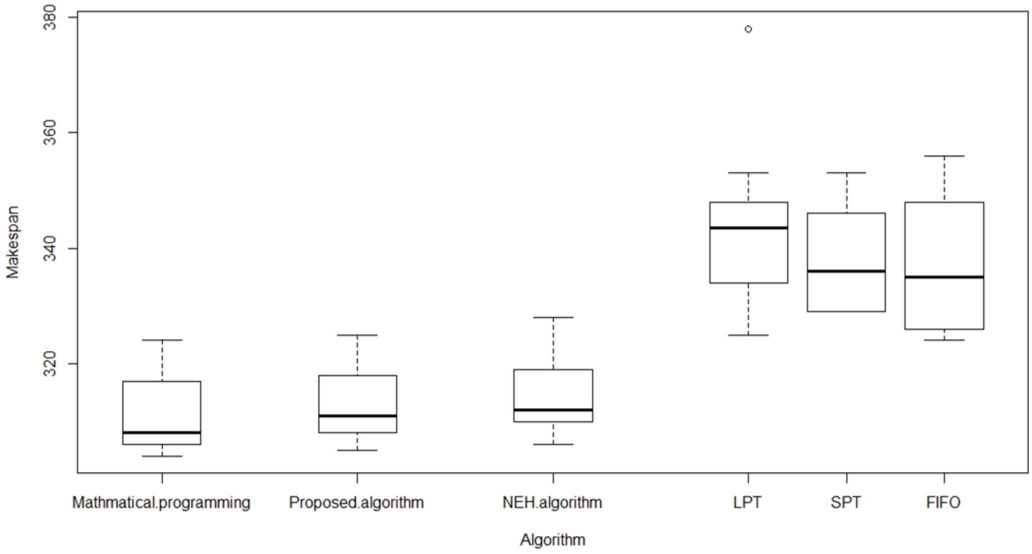
Case	Comparison with NEH algorithm	Improvement from SPT	Improvement from FIFO
Case 1	0.08%	5.02%	4.69%
Case 2	0.36%	5.17%	3.25%
Case 3	-0.03%	5.73%	4.54%
Case 4	0.14%	5.41%	5.16%
Case 5	0.00%	5.43%	2.94%
Case 6	0.79%	6.86%	3.39%
Case 7	0.08%	4.62%	3.11%
Case 8	0.00%	3.98%	3.47%
Case 9	0.19%	5.35%	3.86%
Case 10	0.00%	4.00%	3.27%
Average	0.06%	3.63%	1.55%

[Table 5-5] Comparison of computation time (Large size)

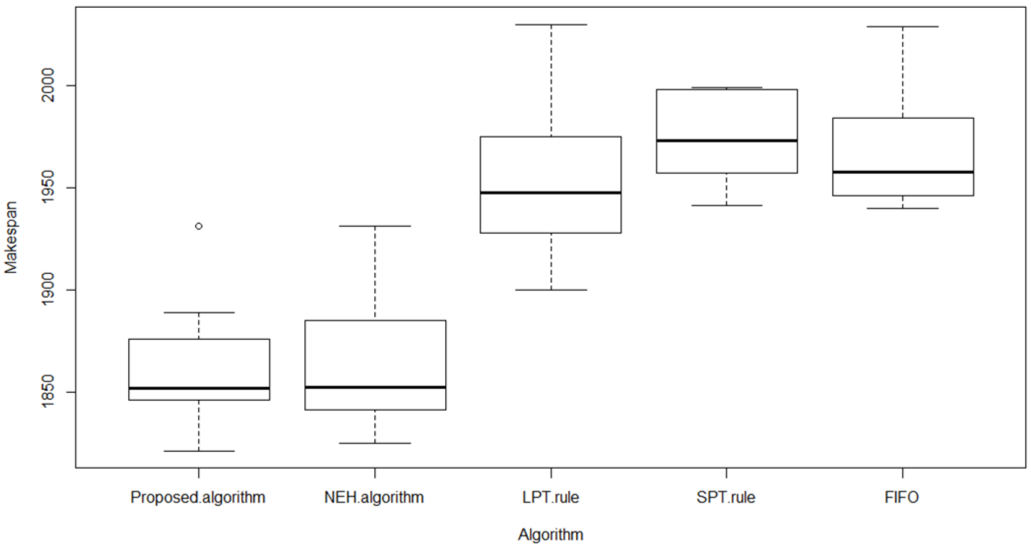
CPU Time(s)	Proposed algorithm (s)	NEH algorithm (s)
Case 1	194.55	205.24
Case 2	201.55	200.16
Case 3	202.61	199.67
Case 4	191.99	198.46
Case 5	197.54	194.2
Case 6	191.06	196.35
Case 7	189.76	198.65
Case 8	189.47	199.01
Case 9	190.29	194.25
Case 10	189.55	197.39
Average time (s)	193.837	198.338



[Figure 5-3] Comparison of CPU time of a 1000 job - 10 stage problem



[Figure 5-4] Interval plot of makespan (Small size)



[Figure 5-5] Interval plot of makespan (Medium size)

[Table 5-6] Computation time for 10 job - 10 machine problem

CPU Time(s)	Enumeration method	Proposed algorithm	NEH algorithm
Case 1	1865.74	0	0
Case 2	1839.49	0	0
Case 3	2641.41	0	0
Case 4	1987.63	0	0
Case 5	2041.28	0	0
Case 6	1841.28	0	0
Case 7	2061.86	0	0
Case 8	2075.6	0	0
Case 9	1974.22	0	0
Case 10	2669.42	0	0
Average time	2099.793 sec	0	0

[Table 5-7] Simulation result of small size problem (10 job - 10 machine)

Case	Optimal solution	Proposed algorithm	NEH algorithm	LPT	SPT	FIFO	Difference from optimal solution	Improvement from SPT	Improvement from FIFO
Case 1	320	321	322	353	350	332	0.31%	8.29%	3.31%
Case 2	305	306	306	337	337	331	0.33%	9.20%	7.55%
Case 3	307	308	310	333	329	324	0.32%	6.38%	4.94%
Case 4	309	309	311	348	337	348	0.00%	8.31%	11.21%
Case 5	317	318	319	348	335	344	0.31%	5.07%	7.56%
Case 6	307	310	310	325	329	326	0.97%	5.78%	4.91%
Case 7	310	312	316	346	353	338	0.64%	11.61%	7.69%
Case 8	324	325	328	378	346	356	0.31%	6.07%	8.71%
Case 9	304	305	306	334	330	324	0.33%	7.58%	5.86%
Case 10	306	313	313	341	329	354	2.24%	4.86%	11.58%
Average	310.9	312.7	314.1	344.3	337.5	337.7	0.576%	7%	7%

[Table 5-8] Simulation result of medium size problem (100 job - 10 machine)

Case	Proposed algorithm	NEH algorithm	LPT rule	SPT rule	FIFO	Comparison with NEH algorithm	Improvement from SPT	Improvement from FIFO
Case 1	1832	1833	1952	1957	1962	0.05%	6.39%	6.63%
Case 2	1863	1865	1928	1968	1951	0.11%	5.34%	4.51%
Case 3	1876	1885	1975	1998	1984	0.48%	6.11%	5.44%
Case 4	1847	1841	1966	1941	1987	-0.33%	4.84%	7.05%
Case 5	1846	1849	1914	1974	1943	0.16%	6.48%	4.99%
Case 6	1889	1889	1975	1999	1975	0.00%	5.50%	4.35%
Case 7	1847	1847	1937	1972	1953	0.00%	6.34%	5.43%
Case 8	1931	1931	2030	1999	2029	0.00%	3.40%	4.83%
Case 9	1856	1855	1943	1996	1940	-0.05%	7.01%	4.33%
Case 10	1821	1825	1900	1956	1946	0.22%	6.90%	6.42%
Average	1860.8	1862	1952	1976	1967	0.06%	5.83%	5.40%

[Table 5-9] Simulation result of large size problem (200 job - 10 machine)

Case	Proposed algorithm	NEH algorithm	LPT	SPT	FIFO	Comparison with NEH algorithm	Improvement from SPT	Improvement from FIFO
Case 1	3539	3542	3719	3726	3713	0.08%	5.02%	4.69%
Case 2	3577	3590	3635	3772	3697	0.36%	5.17%	3.25%
Case 3	3555	3554	3671	3771	3724	-0.03%	5.73%	4.54%
Case 4	3549	3554	3751	3752	3742	0.14%	5.41%	5.16%
Case 5	3538	3538	3667	3741	3645	0.00%	5.43%	2.94%
Case 6	3530	3558	3669	3790	3654	0.79%	6.86%	3.39%
Case 7	3615	3618	3755	3790	3731	0.08%	4.62%	3.11%
Case 8	3642	3642	3731	3793	3773	0.00%	3.98%	3.47%
Case 9	3590	3597	3676	3793	3734	0.19%	5.35%	3.86%
Case 10	3576	3576	3696	3725	3697	0.00%	4.00%	3.27%
Average	3571.1	3576.9	3697	3765	3711	0.16%	5.16%	3.77%

6. Conclusion

6.1 Summary

In this paper, we proposed an RFID-based real-time production management and scheduling system. The proposed system reflects real-time production data generated from RFID and sensors on the shop floor to manage the manufacturing system. It showed that this system is helpful in production planning, execution, and management. Using insights from past studies, we improved the local search method based on the traditional NEH algorithm, which was shown to effective for flow shop. Using this new method, we created a real-time production scheduling method that can quickly find a schedule in the flow shop in many manufacturing sites. Moreover, in order to improve the performance of the scheduling algorithm, we carried out a pre-processing step on the data retrieved from the RFID and the information system to reduce it to a smaller data structure in order to complete the computation process in the main memory of the workstation.

A simulation experiment was conducted through a case study to test the performance of the proposed system. The results showed that the proposed method is superior to rule-based methods being used the existing ERP or MES, and provides sequence planning in a short time. The permutation flow shop targeted by our system is often found in real environments. It is very likely that the proposed system will be useful in other industries or areas that require processing many jobs in a short time.

6.2 Limits and future work

The limitations of this research are as follows: First, our proposed method takes less CPU time than meta-heuristics, such as Genetic Algorithm and Simulated Annealing, but the trade-off is the inability to attain near-optimal performance. This means that there exists a probability of changes due to other problems. Second, if parallel machines exist at a stage, there is an additional machine allocation problem, and further computation or according development of the algorithm become necessary. There are problems related with adopting the RFID on the shop floor. Problems such as the change of tag recognition rate according to the environment such as metal, liquid, but it is not consider in this paper. It is impossible to attach tag to all parts included in the production process. It has been shown to require a further study of the process which includes the components of small size, such as electronics.

Future work in the area will involve, first, developing our system as a cloud-based production management system founded on Software as a Service (SaaS). With a cloud-based production management system on multiple distributed production sites, we think that quick decision making can be effected through multiple factory controls located all over the world. Because the factory, machines, and their parts are all interconnected through advanced concepts like Cyber Physical System (CPS), the applications of the proposed system can expand into more extended supply chain management in the future. Finally, It is interested in applying the proposed system to solve mass-customization production planning and scheduling problems.

Bibliography

Åhlström, P. (1998). Sequences in the implementation of lean production. *European Management Journal, 16*, 327-334.

Baker, K. R., & Trietsch, D. (2013). *Principles of sequencing and scheduling*: John Wiley & Sons.

Baskar, A. (2016). Revisiting the NEH algorithm-the power of job insertion technique for optimizing the makespan in permutation flow shop scheduling. *International Journal of Industrial Engineering Computations, 7*, 353-366.

Berry, W. L., Whybark, D. C., & Jacobs, F. R. (2005). Manufacturing planning and control for supply chain management. New York: McGraw-Hill/Irwin.

Ben-Daya, M., & Al-Fawzan, M. (1998). A tabu search approach for the flow shop scheduling problem. *European Journal of Operational Research, 109*, 88-95.

Brettel, M., Friederichsen, N., Keller, M., & Rosenberg, M. (2014). How virtualization, decentralization and network building change the manufacturing landscape: An Industry 4.0 Perspective. *International Journal of Science, Engineering and Technology 8 (1), 37, 44.*

Chiang, T.-C., Cheng, H.-C., & Fu, L.-C. (2011). NNMA: An effective memetic algorithm for solving multiobjective permutation flow shop scheduling problems. *Expert Systems with Applications, 38*, 5986-5999.

Cui, L., Wang, L., & Deng, J. (2014). RFID technology investment evaluation model for the stochastic joint replenishment and delivery problem. *Expert Systems with Applications, 41*, 1792-1805.

Duplaga, E. A., Hahn, C. K., & Hur, D. (1996). Mixed-model assembly line sequencing at Hyundai Motor Company. *Production and Inventory Management Journal, 37*, 20-26.

Framinan, J. M., Gupta, J. N., & Leisten, R. (2004). A review and classification of heuristics for permutation flow-shop scheduling with makespan objective. *Journal of the Operational Research Society, 55*, 1243-1255.

Gao, J., & Chen, R. (2011). A hybrid genetic algorithm for the distributed permutation flowshop scheduling problem. International Journal of Computational Intelligence Systems, 4(4), 497-508.

Gonzalez, H., Han, J., Li, X., & Klabjan, D. (2006). Warehousing and analyzing massive RFID data sets. In *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on* (pp. 83-83): IEEE.

GS1. (2014). EPC Tag Data Standard Version 1.9, Ratified, Nov-2014. 200.

Jiafu, W., Hehua, Y., Hui, S., & Fang, L. (2011). Advances in Cyber-Physical Systems Research. *KSII Transactions on Internet and Information Systems(TIIS), 5*, 1891-1908.

Jin, F., Song, S., & Wu, C. (2007). An improved version of the NEH algorithm and its application to large-scale flow-shop scheduling problems. *Iie Transactions, 39*, 229-234.

Jo, H. J., & Cho, J. E. (2012). Does Hyundai Motor Company's Production Management Converge to 'Pull' Production?: A Study on the Evolution of Demand-driven Production Management through Information System. *Korean Journal of Sociology, 46*, 233-257.

Jun, S., & Park, J. (2015). A hybrid genetic algorithm for the hybrid flow shop scheduling problem with nighttime work and simultaneous work constraints: A case study from the transformer industry. *Expert Systems with Applications, 42*, 6196-6204.

Kim, K., & Park, J. (2015a). RFID-based production scheduling data storage system. In *Spring Conference of Korea Institute of Industrial Engineering* (pp. 1259-1267): Korea Institute of Industrial Engineering.

Kim, K., & Park, J. (2015b). A study on RFID user memory applications for production management and scheduling. In *The 13th International Conference on Automation Technology*. Taiwan.

Kim, Y. W., & Park, J. (2014). A lifecycle data management system based on RFID technology of EPC class1 gen2 v2. In *Advances in Production Management Systems. Innovative and Knowledge-Based Production Management in a Global-Local World* (pp. 294-301): Springer.

Kimemia, J., & Gershwin, S. B. (1983). An algorithm for the computer control of a flexible manufacturing system. AIIE Transactions, 15(4), 353-362.

Kurz, M. E., & Askin, R. G. (2004). Scheduling flexible flow lines with sequence-dependent setup times. *European Journal of Operational Research, 159*, 66-82.

Landt, J. (2005). The history of RFID. *Potentials, IEEE, 24*, 8-11.

Lee, D., & Park, J. (2008). RFID based traceability in the supply chainnull. *Industrial Management & Data Systems, 108*, 713-725.

Lee, J. S., & Kim, H. J. (2006). RFID code structure and tag data structure for mobile RFID services in Korea. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference* (Vol. 2, pp. 3 pp.-1055).

Li, X., & Yin, M. (2013). A hybrid cuckoo search via Levy flights for the permutation flow shop scheduling problem. *International Journal of Production Research, 51*, 4732-4754.

Liu, G., Song, S., & Wu, C. (2012). Two Techniques to Improve the NEH Algorithm for Flow-Shop Scheduling Problems. In D.-S. Huang, Y. Gan, P. Gupta & M. M. Gromiha (Eds.), *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence* (Vol. 6839, pp. 41-48): Springer Berlin Heidelberg.

Liukkonen, M. (2015). RFID technology in manufacturing and supply chain. *International Journal of Computer Integrated Manufacturing, 28*, 861-880.

Luo, H., Fang, J., & Huang, G. Q. (2015). Real-time scheduling for hybrid flowshop in ubiquitous manufacturing environment. *Computers & Industrial Engineering, 84*, 12-23.

Metlicka, M., Davendra, D., Hermann, F., Meier, M., & Amann, M. (2014). GPU accelerated NEH algorithm. In *Computational Intelligence in Production and Logistics Systems (CIPLS), 2014 IEEE Symposium on* (pp. 114-119).

Naderi, B., & Ruiz, R. (2010). The distributed permutation flowshop scheduling problem. Computers & Operations Research, 37(4), 754-768.

Na, H., & Park, J. (2014). Multi-level job scheduling in a flexible job shop environment. *International Journal of Production Research, 52*, 3877-3887.

Nawaz, M., Enscore, E. E., & Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega, 11*, 91-95.

Nowicki, E. (1993). AN APPROXIMATION ALGORITHM FOR THE M-MACHINE PERMUTATION FLOW-SHOP SCHEDULING PROBLEM WITH CONTROLLABLE PROCESSING TIMES. *European Journal of Operational Research, 70*, 342-349.

Pais, S., & Symonds, J. (2011). Data storage on a RFID tag for a distributed system. *International Journal of UbiComp, 2*.

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

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

Schmidt, M., Ziemba, H.-W., & Volkswagen, A. (2010). RFID in the Automotive Industry-A Standard Based Approach Towards On-Tag Data Sharing in Cross-Company Logistics. In *MMS* (pp. 167-173).

Stankovski, S., Lazarević, M., Ostojić, G., Ćosić, I., & Puric, R. (2009). RFID technology in product/part tracking during the whole life cycle. *Assembly Automation, 29*, 364-370.

Want, R. (2006). An introduction to RFID technology. Pervasive Computing, IEEE, 5(1), 25-33.

Wagner, S. M., & Silveira-Camargos, V. (2011). Decision model for the application of just-in-sequence. *International Journal of Production Research, 49*, 5713-5736.

Wang, S.-J., Liu, S.-F., & Wang, W.-L. (2008). The simulated impact of RFID-enabled supply chain on pull-based inventory replenishment in TFT-LCD industry. *International Journal of Production Economics, 112*, 570-586.

Xu, L. D., He, W., & Li, S. (2014). Internet of Things in Industries: A Survey. *Ieee Transactions on Industrial Informatics, 10*, 2233-2243.

Yang, Z., Xu, W., Wong, P.-K., & Wang, X. (2015). Modeling of RFID-Enabled Real-Time Manufacturing Execution System in Mixed-Model Assembly Lines. *Mathematical Problems in Engineering, 2015*, 15.

Ying, K.-C., & Liao, C.-J. (2004). An ant colony system for permutation flow-shop sequencing. *Computers & Operations Research, 31*, 791-801.

Zhang, Y., Huang, G. Q., Sun, S., & Yang, T. (2014). Multi-agent based real-time production scheduling method for radio frequency identification enabled ubiquitous shopfloor environment. *Computers & Industrial Engineering, 76*, 89-97.

Zhong, R. Y., Huang, G. Q., Lan, S., Dai, Q. Y., Zhang, T., & Xu, C. (2015). A two-level advanced production planning and scheduling model for RFID-enabled ubiquitous manufacturing. *Advanced Engineering Informatics, 29*, 799-812.

초록

제조업에서 많은 제품들이 수 천, 많게는 수 만개의 부품으로 구성된 복잡한 구조를 가지고 있다. 복잡한 제품 구조는 다수의 공급망과 생산 공정을 거치게 된다는 것을 의미한다. 기업의 입장에서는 최소한의 재고와 높은 생산 설비 가동률을 유지하는 것이 필요하다. 따라서 많은 기업들이 적기납입 방식(Just In Time), 직서열 방식(Just In Sequence) 생산 시스템을 생산 현장과 기업의 시스템에 적용하였다. 그러나 이런 시스템에서는 생산현장과 정보시스템간의 동기화가 필요하다.

RFID 기술은 실제 현장의 정보와 정보시스템을 연결해주는 용도로 생산 현장에서 사용되어왔다. 최근 RFID 기술의 발전에 따라, 기존의 식별 용도에서 추가로 User memory bank 데이터 표준이 제시되었다. RFID user memory를 활용한 생산관리 분야의 새로운 활용방안이 있다.

따라서 본 연구에서는 RFID user memory를 활용하여 실시간 생산관리에 필요한 생산정보 데이터 구조를 연구하며 효율적인 스케줄을 생성하기 위한 알고리즘을 제안하였다. 다양한 조건 아래에서 기존 일정계획 알고리즘과 본 논문에서 제안하는 일정계획 알고리즘의 성능을 평가하였다. 또한 제안하는 모델과 알고리즘의 실현성을 확인하기 위하여 사례 연구를 수행하였다. 사례 연구를 통해 제안한 모델과 알고리즘은 실시간으로 생산 현장의 의사결정 지원을 효과적으로 수행 할 수 있으며, 최신 RFID 기술을 활용한 생산 관리 방안을 확인하였다.

주요어 : RFID, RFID tag user memory bank, 생산 시스템, 생산관리, 생산 스케줄링, NEH 알고리즘
학 번 : 2014 - 22635