applied sciences

MDPI

Article

Reconfiguration Decision-Making of IoT based Reconfigurable Manufacturing Systems

Sumin Han [1], Tai-Woo Chang [2,*], Yoo Suk Hong [1] and Jinwoo Park [1]

[1] Department of Industrial Engineering, Seoul National University, Seoul 08826, Korea; hans8501@snu.ac.kr (S.H.); yhong@snu.ac.kr (Y.S.H.); autofact@snu.ac.kr (J.P.)

[2] Department of Industrial and Management Engineering/Intelligence and Manufacturing Research Center, Kyonggi University, Suwon, Gyeonggi 16227, Korea

* Correspondence: keenbee@kgu.ac.kr; Tel.: +82-31-249-9754

Received: 25 May 2020; Accepted: 10 July 2020; Published: 13 July 2020

Abstract: With the recent diversification of demands, manufacturing systems that can respond to multiple types of goods have become more important. In this circumstance, reconfigurable manufacturing systems (RMSs) that can provide flexible manufacturing with limited machine tools through reconfiguration have gained a lot of attention. As an RMS supports flexibility through layout reconfiguration, reconfiguration decision-making is very important and difficult. The development of IoT technology has made it possible to collect hidden information inside systems. This study focused on the reconfiguration decision-making system with the data acquisition system based on IoT technology. The decision-making system detected a reconfiguration situation and built a reconfiguration plan using the data collected by IoT sensors. The performance of the algorithm proposed in this study was verified in a simulation experiment. It was found that the algorithm had a stable performance under various reconfigurable situations. It is expected that the proposed system will help to improve the performance of RMS.

Keywords: reconfigurable manufacturing system; reconfiguration decision-making; variable neighborhood search algorithm; IoT sensor

1. Introduction

With the development of information and communication technologies or services (Internet, social media, etc.) and with the increased social diversification, a long tail market in which various products are consumed in a small quantity according to diverse preferences has been popular [1]. However, it is hard for a mass-production system to meet small demands efficiently. For this reason, the need for a new manufacturing system capable of responding to the long tail market has emerged.

An intelligent production system that highlights flexibility has been developed, and various relevant studies have been conducted [2]. As a result, various manufacturing systems such as flexible manufacturing systems (FMSs) and reconfigurable manufacturing systems (RMSs) have been suggested. In addition, various emerging technologies such as the Internet of Things (IoT), cloud system, and 3D printing have tried to be applied [3].

This study focused on RMSs with the IoT sensor system. An RMS is a production system that implements flexible responses to external and internal needs through the reconfiguration of modules [4]. With the concept of reconfiguration, RMS can implement flexibility with less cost when compared to a full flexibility manufacturing system. However, reconfiguration requires a complex decision-making process including the status of modules, line conditions, and surrounding environments. To make a proper reconfiguration decision, it is necessary to obtain a variety of information including detailed information in modules.

Recently, with the development of IoT technology, it is possible to collect the hidden data of a manufacturing system, which had never been collected previously. This study proposes a decision-making algorithm for reconfiguration that uses information obtained by the IoT sensors installed in RMS modules. The proposed system is expected to detect reconfiguration situations quickly to create appropriate reconfiguration planning and achieve fast stabilization after reconfiguration by using the IoT information.

The structure of this study is as follows. Section 2 describes the related studies and their differences with this study. Section 3 presents the features of the RMS and the application of the proposed system. We explain the application of IoT technology to the RMS in Section 4 and present the proposed reconfiguration decision-making process in Section 5. Section 6 describes the evaluation of the proposed system in a simulation experiment. Section 7 presents the achievements of this study and the follow-up research.

2. Related Works

A RMS can provide flexible manufacturing through reconfiguration. However, for a RMS to reach its potential, it is necessary to address various additional issues such as reconfiguration decision making and maintenance. Various studies have been conducted on these issues.

Bi et al. [1] described the definitions of RMS and technical situations by reviewing various studies on RMS. The research is meaningful to the point that it includes diverse definitions of RMS and provides an integrated view of RMS. Koren et al. [4] reviewed various studies on RMS and described a current research direction and the applicability of the latest technologies from many different perspectives. This study suggested various applications of emerging technologies (e.g., IoT, cloud computing, and artificial intelligence) to RMS. Minhas et al. [5] suggested reconfigurable strategies for RMS in various levels of the manufacturing system. The suggested strategies were applied to inter-cell levels, machining cell levels, and process monitoring system levels.

Xia et al. [6] studied an algorithm to generate a plan for the reconfiguration and maintenance of RMS. They proposed an algorithm for establishing a plan in consideration of each machine's reliability and failure, and the costs for replacement and failure by applying a reconfigurable maintenance time window (RMTW). Koren et al. [7] proposed a genetic algorithm-based algorithm for reconfiguration decision-making in RMS, and suggested a reconfiguration decision-making method according to a certain productivity level. The method is for minimizing changes and maximizing productivity under restrictions of different available processes according to machines and of different required processes according to goods. Kumar et al. [8] focused on the reconfiguration of the assembly system of RMS and discussed the production planning algorithm for the assembly scheduled at a minimized reconfiguration cost and with maximized reconfiguration gains. They proposed an algorithm that had various objective functions considering cost and delay time for reconfiguration, work balance, due date, etc. Huang et al. [9] suggested a reconfiguration-points decision algorithm based on complexity. Information-based entropy theory was applied to calculate complexity. Based on complexity, the proposed algorithm calculated a machine state probability to make reconfiguration decisions.

Hsieh and Lin [10] suggested a negotiation-based planning algorithm for RMS. They assumed a manufacturing system with self-aware machines and applied the Petri-net to model the behavior of the system and suggested a negotiation policy for machine selection. Hsieh [11] suggested a meta-heuristic approach for dynamic process planning in RMS. He modeled the behavior of their self-aware manufacturing system with the Petri-net method, which applied a discrete particle swarm algorithm to solve the planning problem. The suggested algorithm performed well in a simulation experiment. Tang et al. [12] suggested a reconfiguration method for the holonic manufacturing system and proposed an agent-based approach, in which each machine negotiates for job allocation and connection with another machine. Their method is based on the assumption that entries of every demand trigger reconfiguration. Han et al. [13] researched the pallet-fixture allocation of reconfigurable modular cells and studied an algorithm for pallet and fixture allocation in an RMS. The proposed algorithm was verified in simulation experiments and the results showed that the algorithm was applicable to the reconfiguration decision-making of pallets and fixtures.

With improvements in sensor technologies and telecommunication technologies, it is possible to acquire information rapidly from IoT sensors. To improve the performance of RMS, many studies have been conducted on the application of IoT sensors to RMSs.

Kurniadi and Ryu [14] discussed the problem of reconfiguration when IoT was applied to a RMS and suggested a reconfiguration planning algorithm to set a certain productivity level at a minimal cost. They also proposed a specialized simulation method to evaluate various reconfiguration plans. Bi et al. [15] reviewed the various application of IoT to the modern manufacturing system. They enlisted various features that were enabled by IoT applications. They set reconfigurability as the major goal of IoT application. Scholz et al. [16] suggested a modular reconfigure system based on 3D printing and introduced the concept of SMARTLAM, which is a reconfigurable system with image processing and IoT sensors.

Mourtzis et al. [17] suggested an IoT based monitoring system for shop floor control. They also suggested data acquisition modules and monitoring logics based on OPC-UA (Open Platform Communication - Unified Architecture) protocol. Mourtzis et al. [18] suggested an OPC-UA model for monitoring and data gathering in a general machine and included performance indicators, an evaluation process, a data management model, and data acquisition process management manufacturing systems based on a RMS.

Pellicciari et al. [19] tried to make a digital twin of an actual manufacturing site. To make a realistic digital environment, various inputs from IoT sensors were applied. They tried to enhance the changeability of a reconfigurable system with the digital twin. Ferreira et al. [20] suggested a web-based integration procedure for reconfigurable robotic cells through software structures and management logic with web-based control. Shin et al. [21] proposed a dynamic reconfiguration algorithm based on the cyber-physical system (CPS) environment. The algorithm was based on the support vector machine (SVM) method. The results of the algorithm selected the processing machine and route for each entering demand.

Most of the related studies focused on the management and installation of a manufacturing system based on a RMS, and only a few studies have focused on reconfiguration algorithms and strategies. Other studies that focused on reconfiguration and IoT applications have different assumptions such as self-awareness, reconfiguration without delay, job shop environment, or FMS environment, which are not popular in the current manufacturing systems. Therefore, this study supplemented existing researches by concentrating on reconfiguration decision-making algorithms for RMS using the advantages of IoT sensor application.

3. Concepts of Reconfigurable Manufacturing Systems (RMSs)

3.1. Definition of Reconfigurable Manufacturing System (RMS)

A concept of RMS was suggested in the 1990s when a small quantity batch production system emerged, which came to be widely known in the late 1990s by Koren. He defined RMS as a production system capable of reconfiguring hardware and software [22]. Since then, an RMS has been defined in the way of developing its application to the operation level, which is higher than the level of the manufacturing site, and recent studies have even considered reconfiguration on the rapid adoption of new manufacturing technologies and competitiveness [3,23]. This study defines an RMS as a manufacturing system that responds to external and internal changes through fast reconfiguration.

3.2. Factors in RMS

RMS considers four reconfigurable factors as shown in Figure 1 [1]. A reconfigurable machining system is a part that mainly includes machine tools and their assisting parts. It is key to a manufacturing process and a major reconfiguration target in RMS. A reconfigurable material-handling system serves the role to deliver materials and processed objects to each machine tool and to connect individual machine tools. A reconfigurable fixturing system is related to the fixtures used for transferring and fixing processed objects. A reconfigurable assembly system manages the reconfiguration of an assembly machine to finish a product. A reconfigurable software system manages the whole system and reconfigurable factors.

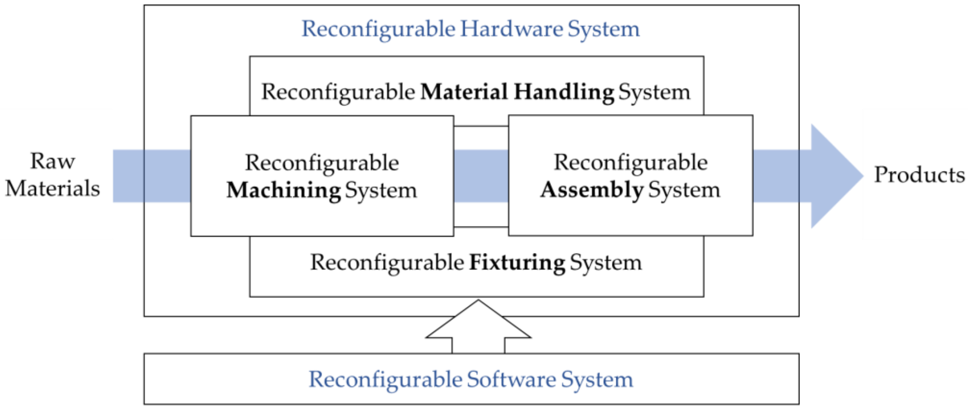

Figure 1. Reconfigurable factors in a reconfigurable manufacturing system (RMS).

In this study, reconfiguration was limited to machine tools in a factory of discrete manufacturing. The system suggested in the study builds a reconfiguration plan to respond to changes in diverse production situations.

3.3. Characteristics of RMS

A mass-production system is usually established based on a dedicated manufacturing system (DMS) that concentrates on certain work. The system has high productivity, but it is hard to manufacture different products. Although FMS and RMS have flexible responses, their productivity is lower than that of DMS.

Both FMS and RMS can provide flexible responses, but their levels are different. FMS concentrates on flexibility and supports flexible production without reconfiguration. Due to the cost of its excellent flexibility, the system has the lowest productivity. A RMS provides flexibility through reconfiguration. Due to the reconfiguration, RMS has lower productivity than DMS, but is better than FMS.

To offer the flexibility and productivity of RMS, it is essential to make reconfiguration decision-making efficient. If reconfiguration is made inefficient, it is possible to waste time due to frequent reconfiguration and low productivity. This study proposed a decision-making system with efficient reconfiguration using the data collected by IoT sensors.

4. Internet of Things (IoT) Based Data Collection System for RMS

4.1. Necessity of Internet of Things (IoT) Application

Reconfiguration in RMS is triggered by various situations including machine problems, low productivity, a change in management strategy, and a change in regulation. To recognize this situation and make a fast and proper reconfiguration decision, it is necessary to gather various real-time data from the inside and outside of machines.

The data acquisition from the inside of a machine has very high costs. However, with the development of IoT sensor technologies, data acquisition costs are on the decrease, and it is possible to gather the hidden data in machines at a low cost. In this study, the suggested system tries to utilize IoT sensor data in reconfiguration decision-making. With the data from IoT sensors, it is possible to detect the reconfiguration needs from the inside.

After reconfiguration, the RMS has to spend time doing ramp-up and fine-adjustment between the modules [24]. Data acquired from machines and machine tools through IoT sensors can be utilized in this process. The IoT sensor data can accelerate data acquisition and reduce wasted time in the ramp-up process.

4.2. Data Acquisition and Flow in RMS

Data acquisition tools are installed in at various points of the RMS, as shown in Figure 2. The data are acquired through IoT sensors in various levels of the RMS and are transferred to a factory management system. The sensors on machine tools collect the wear and crack data of tools, and the sensors on machines collect information on the inside of machines and the relationship with other modules. The collected data are gathered in a software system, transferred to the factory management system, and applied to reconfiguration decision-making.

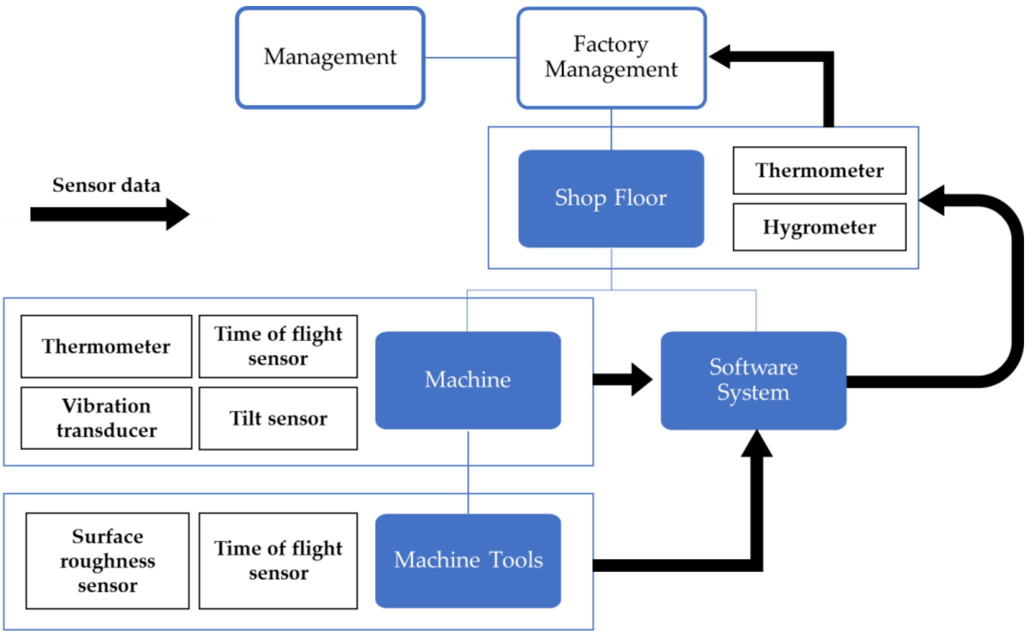

Figure 2. Data acquisition and flow in the RMS.

In this study, the status data of RMS were acquired by a small module with wireless communication with the structure shown in Figure 3. In the data acquisition module, various IoT sensors are installed to detect the status of the attached module, the processing module to convert the analog signal to digits, and wireless and wired communication modules. As a low power technology was applied to the sensors and the data acquisition module, they work in a low power or small battery environment.

Table 1 describes a list of sensors in the data acquisition module. With these sensors, tool abrasion, expansion, and crack and machine temperature, vibration, tilting, and connection between modules can be detected. Collected data are then applied to reconfiguration situation detection and decision-making, reconfiguration plan building, and reconfiguration completion check.

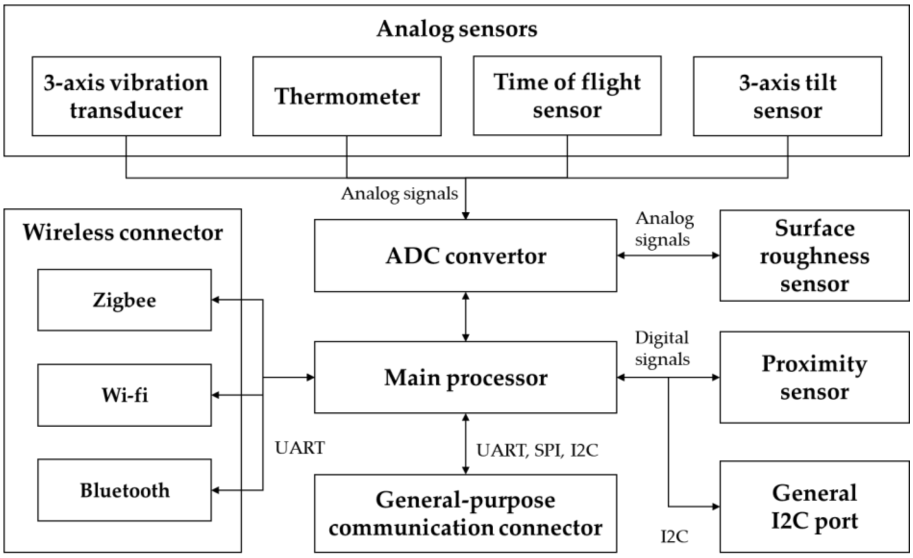

Figure 3. Structure of the data acquisition module.

Table 1. Example of Internet of Things (IoT) sensors in the RMS.

Category	Sensor	Collecting Item
Machine status	Thermometer 3-Axis vibration transducer Time of flight sensor 3-Axis tilt sensor	Internal temperature of a machine Vibration during machining Assembly status of machining tools Tilted angle of an installed machine
Machine Tools and Parts status	Surface roughness sensor	Surface wear of tools Cracks of tools
	Time of flight sensor	Expansion of tools or parts Whether tools and parts are in place
Shop floor	Thermometer Hygrometer	Temperature of factory Humidity of factory

5. Decision-Making System

5.1. Decision-Making Process

The detailed decision-making process from data collection to reconfiguration execution is illustrated in Figure 4. In this process, the RMS continuously reconfigures its structure to adjust the ability of the manufacturing system to meet the detected reconfiguring needs.

5.1.1. Reconfiguration Situations

Based on the data collected by the IoT sensors, current factory information, external information, and factory status were identified, and reconfiguration decisions were made. Without the data obtained by the IoT sensors, it would take longer to recognize a situation.

The situations that need reconfiguration decision-making are as follows: The first situation is changed in external factors such as changed sales strategies, revised government regulations, changed ratios of products, new products, and the introduction of new processes. In this case, it is required to make a reconfiguration decision to meet a given external goal.

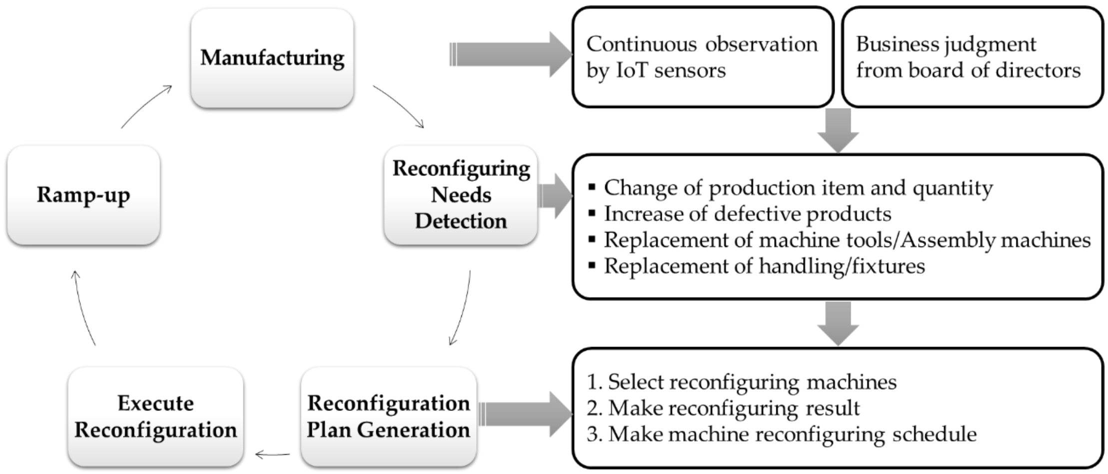

Figure 4. Reconfiguration process of the RMS.

The second situation is changed in internal factors such as failures of productivity goals, machine failures, and predicted failures of productivity. In this case, it is necessary to execute reconfiguration to deal with the situation. Table 2 presents the causes of situations that need reconfiguration decision-making.

Table 2. An example of reconfiguration situations.

Category	Situation	
External	Change of product mix	Introduction of new products changes in product combinations
	Introduction of additional processes	Introduction of additional processes due to environmental regulation or additional features
Internal	Lack of productivity	Reconfiguration for enhancing the productivity of a production line
	Breakdown	Replacement due to sudden failure of the machine and reconfiguration
	Preventive maintenance	Preventive maintenance and reconfiguration according to tool life and internal condition

5.1.2. Decision-Making Factors

When a situation for reconfiguration occurs, it is necessary to build a reconfiguration plan that includes the outcome and timing of reconfiguration. First, it is necessary to decide which machine is to be reconfigured and the outcome of reconfiguration. Then, the time schedule of each machine's reconfiguration needs to be decided. This is scheduled in consideration of a reconfiguration time, the unavailable time of machines, and the current production plan. During reconfiguration, connected machines have to stop. Therefore, it greatly influences a production plan. In this study, a reconfiguration time was set based on the assumption that reconfiguration is possible after stopping related machines in the entire manufacturing line.

Through reconfiguration, a manufacturing system is modified to meet given goals such as a productivity level and a regulation change. As the current manufacturing system has a change through reconfiguration, it is natural to apply existing production policies such as a scheduling rule and an objective function to the reconfigured system. The result drawn by reconfiguration is operated and evaluated according to these policies. The dispatching rules for production scheduling generally applied include FIFO (first in first out), SPT (shortest processing time), and EDD (earliest due date). There are various target objective functions such as a due date and a level of productivity.

Before reconfiguration decision-making, it is required to determine the reconfiguration targets. For efficient reconfiguration, reconfiguration targets have to include both reconfigured modules and their related modules. The algorithm proposed in this study considers all modules as reconfiguration targets in reconfiguration planning and takes into account every available reconfiguration starting time.

5.2. Definition of Reconfiguration Problem

The problem of reconfiguration was modeled with integer programming. For the modeling, the parameters and decision variables were defined as shown in Table 3.

Table 3. Parameters and decision variables of integer programming.

Category		**Detail**
Machine	l	Index of line (1~|L|)
	r	Index of room for machines in each line (1~R)
	i	Index of machine type (1~I)
	m_r^l	Initial type of r^{th} machine in l^{th} line
Cost	c_i^l	Installation cost of machine with machine type i in l^{th} line (without IoT)
	$c_{penalty}$	Penalty of an unfulfilled demand
	it_i^l	Installation time of machine with machine type i in l^{th} line
Demands	t	Index of demand type (1~|T|)
	d_t	Index of demand with type t before change (1~D_t)
	$due_{d_t}^t$	Due of demand d_t with type t
	$p\prime_i^{t,l}$	Whether demand type t is available in machine type i in l^{th} line (0 or 1)
	$p_i^{t,l}$	Process time of demand type t in machine type i in l^{th} line.
	$C_{d_t}^t$	Whether demand d_t with demand type t is canceled (0-cancelled or 1-not cancelled)
	$R_{d_t}^t$	Whether change of demand d_t with demand type t is recognized in planning type (0-not recognized or 1-recognized)
	DC	Time when the demand change is recognized
Decision variables	$m\prime_r^l$	Changed type of r^{th} machine in l^{th} line (Integer)
	m_r^{*l}	Whether machine type of r^{th} machine in l^{th} line is changed (0-not changed or 1-changed)
	rt_r^l	Reconfiguration time of r^{th} machine in l^{th} line (Integer)
	$x_{d_t,\,r}^{t,l}$	Starting time of demand d_t with type t in r^{th} machine in l^{th} line (Integer)
	$y_{d_t,r}^{t,l}$	Whether demand d_t with type t is processed by r^{th} machine l^{th} line before reconfiguration (0-not processed or 1-processed)
	$y\prime_{d_t,r}^{t,l}$	Whether demand d_t with type t is processed by r^{th} machine l^{th} line after reconfiguration (0-not processed or 1-processed)
	$s_{d_t}^t$	Whether due of demand d_t with type t is satisfied (0-not satisfied or 1-satisfied)

The objective function in Equation (1) considers four elements: the minimum number of the demands unfulfilled within due date, the minimum completion time of the whole production, the replacement cost for reconfiguration, and the minimum penalty of the demands unfulfilled within the due date. α, β, γ, or δ is the weight value of each factor, respectively.

$$min\left\{\alpha\sum_t^T\sum_{d_t}^{|D_t|}s_{d_t}^tC_{d_t}^t+\beta max_{t,d_t,r}\left(x_{d_t,r}^{t,L}+p_{m_t^l}^{t,L}\right)+\gamma\sum_l^L\sum_r^R m_r^{*l}c_r^l+\delta\sum_t^T\sum_{d_t}^{|D_t|}C_{d_t}^t\left(1-s_{d_t}^t\right)c_{penalty}\right\} \quad (1)$$

The constraints are presented as follows:

$$x_{dt,r}^{t,l}\leq M\left(y_{dt,r}^{t,l}+y\prime_{dt,r}^{t,l}\right)\ (\forall t,\ d_t,l,r). \quad (2)$$

$$x_{dt,r}^{t,l}>\sum_r^R x_{dt,r}^{t,l-1}+\sum_i^I p_{dt,\ l-1,\ i}^t(\forall t,\ d_t,l,\ r) \quad (3)$$

$$0 + DC \times C^{t}_{d_t} \le \sum_{r}^{R} x^{t,l}_{d_t,r} (\forall t,\ d_t, l), \tag{4}$$

$$\sum_{r}^{R} x^{t,L}_{d_t,r} < due^{t}_{d_t} + \left(1 - s^{t}_{d_t}\right) M \,(\forall t,\ d_t), \tag{5}$$

$$\sum_{r}^{R} y^{t,l}_{d_t,r} \le\ 1\ (\forall t,\ d_t, l), \tag{6}$$

$$\sum_{r}^{R} y'^{t,l}_{d_t,r} \le\ 1\ (\forall t,\ d_t, l), \tag{7}$$

$$\sum_{r}^{R} y'^{t,l-1}_{d_t,r} \le \sum_{r}^{R} y'^{t,l}_{d_t,r} (\forall t,\ d_t, l = 2 \sim L), \tag{8}$$

$$\sum_{r}^{R} y^{t,l-1}_{d_t,r}\ \ge\ \sum_{r}^{R} y^{t,l}_{d_t,r} (\forall t,\ d_t, l = 2 \sim L), \tag{9}$$

$$\sum_{r}^{R} y'^{t,l}_{d_t,r} + \sum_{r}^{R} y^{t,l}_{d_t,r}\ \le\ 1\ (\forall t,\ d_t, l), \tag{10}$$

$$\left(x^{t,l}_{d_t,r} + y^{t,l}_{d_t,r} p^{t,l}_{m^l_r}\right) \le x^{t^*,l}_{d_{t^*},r}\ or\ x^{t*,l}_{d_t*,r} + y^{t*,l}_{d_t*,r} p^{t*,l}_{m^l_r} \le M\left(1 - y^{t,l}_{d_t,r}\right) + x^{t,l}_{d_t,r}\quad (\forall t^* \in T, d_{t^*}, l, r, d_t \in D_t / d_{t^*}) \tag{11}$$

$$\left(x^{t,l}_{d_t,r} + y'^{t,l}_{d_t,r} p^{t,l}_{m^l_r}\right) \le x^{t^*,l}_{d_{t^*},r}\ or\ x^{t*,l}_{d_t*,r} + y'^{t*,l}_{d_t*,r} p^{t*,l}_{m^l_r} \le M\left(1 - y'^{t,l}_{d_t,r}\right) + x^{t,l}_{d_t,r} (\forall t^* \in T, d_{t^*}, l, r, d_t \in D_t / d_{t^*}) \tag{12}$$

$$m^{*l}_r\left(rt^l_r + it^l_r\right) \le y'^{t,l}_{d_t,r} x^{t,l}_{d_t,r}\ or\ x^{t,l}_{d_t,r} + y^{t,l}_{d_t,r} p^{t,l}_{m^l_r} \le M\left(1 - m^{*l}_r\right) + rt^l_r\quad (\forall t, d_t,\ l, r), \tag{13}$$

$$rt^l_r \le M y^{t,l}_{d_t,r} + y'^{t,l}_{d_t,r} x^{t,l}_{d_t,r}\ \ (\forall t, d_t,\ l, r)\ , \tag{14}$$

$$- M y'^{t,l}_{d_t,r} + y^{t,l}_{d_t,r} x^{t,l}_{d_t,r}\ \le rt^l_r\ (\forall t, d_t,\ l, r), \tag{15}$$

$$x^{t,l}_{d_t,r} \le \begin{cases} M\left(y^{t,l}_{d_t,r} + 1 - \sum_{i}^{I} p^{t,l}_{i}\right) & \left(x^{t,l}_{d_t,r} > t^l_r\right) \\ M\left(y'^{t,l}_{d_t,r} + 1 - \sum_{i}^{I} p^{t,l}_{i}\right) & \left(x^{t,l}_{d_t,r} < t^l_r\right) \end{cases}\quad (\forall t,\ d_t, l, r), \tag{16}$$

$$y^{t,l}_{d_t,r}\ \le p^{t,l}_{d_t,m^l_r}\ (\forall t,\ d_t, l, r), \tag{17}$$

$$y'^{t,l}_{d_t,r}\ \le p^{t,l}_{d_t,m'^l_r}\ (\forall t,\ d_t, l, r), \tag{18}$$

$$0 \le m'^l_r \le I \tag{19}$$

$$m^{*l}_r\ =\ \begin{cases} 1 \to & y^{t,l}_{d_t,r}\ \ne\ y'^{t,l}_{d_t,r} \\ 0 \to & y^{t,l}_{d_t,r}\ =\ y'^{t,l}_{d_t,r} \end{cases}, \tag{20}$$

$$x^{t,l}_{d_t,r} \ge 0\ ,\quad t^l_r \ge 0\ (\forall\ t,\ d_t,\ l,\ r), \tag{21}$$

Equation (2) represents the constraint that a process can start only in the position with proper machine allocation. Equation (3) represents the constraint that the starting time of each process is set after a completion time of its previous process. Equation (4) represents the constraint that the starting time of work should be set after its recognition time. Equation (5) represents the constraint that it is necessary to check whether the final process is complete within the due date. Equations (6) and (7) represent the constraints that one process can be allocated to only one machine. Equations

(8) and (9) represent the constraints of demand allocation to the original and reconfigured machines. Equation (10) represents the constraint that the process allocation should not be duplicated before and after reconfiguration.

Equations (11) and (12) represent the constraint that the starting time of a process should not interfere with the execution time of a different process in the same line. Equation (13) represents the constraint that the reconfiguration time of a machine should not overlap the execution time of the processes. Equations (14) and (15) represent the constraint that the allocation and start time of the process have to be fitted with an original or reconfigured machine. Equation (16) represents the constraint that the starting time of a process can be changed when the machine is skipped. Equations (17) and (18) represent the constraint that a process should be allocated in line with the type of machine installed. Equation (19) means that the type of reconfigured machine should be an available type. Equation (20) represents the constraint that it is necessary to check whether a type of machine is changed through reconfiguration. Equation (21) represents the positive condition of the decision variable related to time.

In this problem, there is a multiplication of variables, and constraints are changed according to reconfiguration conditions. Therefore, this problem fails to meet the PSD (positive semi-definitive), which is the condition to solve an easily solved linear programming problem. In addition, it is impossible to find the optimal solution in a short time. Therefore, this study suggests a meta-heuristic algorithm based on a variable neighborhood search (VNS) algorithm and adopts a simulation approach in order to find a good solution within a short time.

5.3. Algorithm for Reconfiguration Planning

The neighborhood search-based algorithm reflects the neighborhood search algorithm for finding a local optimal solution. The VNS algorithm randomly shakes a solution to fend off convergence to a local optimal solution, as shown in Algorithm 1. A solution in the proposed VNS algorithm is the list of installed modules after reconfiguration, and the schedule of reconfiguration is confirmed in the evaluation process.

Algorithm 1: Process of VNS algorithm.

1. Generate neighbor structure $N_k(k = 1, \cdots, k_{max})$
2. Generate initial solution x
3. $k=1$
4. Repeat until $k = k_{max}$
   4.1. Shaking: Randomly select one of the neighbor structure $N_k(x)$ of solution x to randomly select x' from $N_k(x)$
   4.2. Local search by VNS
      4.2.1. $l=1$
      4.2.2. Repeat until $l = l_{max}$
         4.2.2.1. Search best neighbor x' from $N_l(x')$ and set as x''
         4.2.2.2. if x'' is better than x', replace x' to x'' and set $l = 1$
         4.2.2.3. if not, set $l = 1 + 1$
   4.3. Local search: Compare x and x''
      4.3.1. if x'' is better than x, replace x as x'' and set $k = 1$
      4.3.2. if not, set $k = k + 1$

5.3.1. Generation of Initial Solution

In this study, an initial solution was generated randomly. To generate the initial solution, the algorithm randomly changes the initial type of an installed module with a 20% probability. If there are modules that must be changed, the algorithm changes them in the initial solution generation step.

After that, if the process, which is essential in the given demand combination, is not available in a modified solution, one of the machines is randomly selected and is changed into the machine executing the missed process. In this way, it is possible to build a solution line that can respond to the purposes of reconfiguration.

5.3.2. Factory Layout and Neighbor Structure

The setting of a neighboring structure depends on the structure of a manufacturing system. The flow shop with gantry type structure is assumed in this study, as shown in Figure 5. In this assumption, a gantry manages products in each process step, and multiple machines work simultaneously.

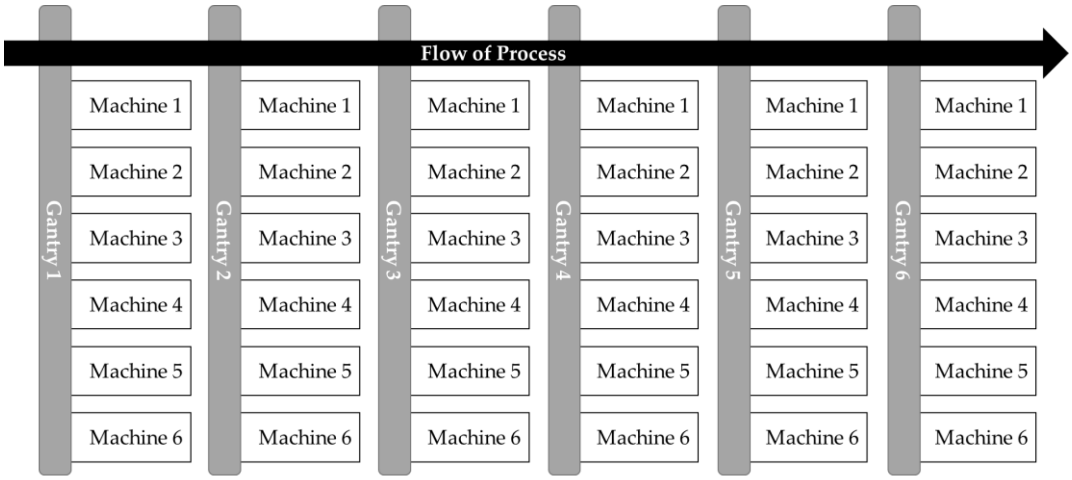

Figure 5. An example of a gantry type system.

To generate a neighbor structure, it is necessary to define the structure of neighbors. In this study, L neighbor structures were generated in the way of selecting one machine from line l (1~$|L|$) and removing it or changing its type. Accordingly, the generated neighbor structure N_l is the neighbor structure in which the machine in the line l is changed.

In addition, a different neighbor structure was generated in the way of selecting machines one by one in two different lines and exchanging their types. Through this process, additional $L(L\text{-}1)/2$ neighbor structures were introduced.

5.3.3. Evaluation of Solution

With the reconfiguration result in a solution, the solution is evaluated. A production schedule contains a reconfiguration schedule that is created according to the existing scheduling rules. It is assumed that reconfiguration is executed without any suspension of jobs in the process and only in between jobs.

In this step, every available reconfiguration time is generated, and a reconfiguration schedule with the best performance is selected as the final reconfiguration plan. The flow of the process is illustrated in Figure 6.

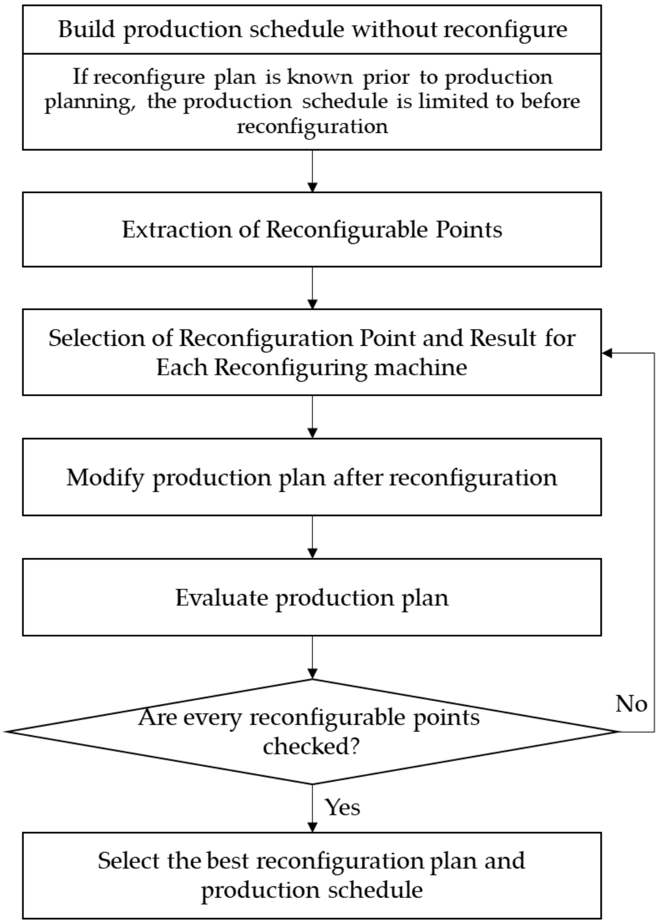

Figure 6. The reconfiguration planning process.

For the evaluation of a solution, Equation (22), which is the objective function of integer programming, calculates the adjusted values of coefficients α, β, γ, and δ. The number of satisfying demands and finishing time are first considered in the evaluation of a solution. Replacement and delay costs are divided by their maximum cost and then added.

$$\sum_{t}^{T}\sum_{d_t}^{|D_t|} s_{d_t}^{t} C_{d_t}^{t} + max_{t,d_t,l,r}\left(x_{d_t,r}^{t,l} + p_{m_t^l}^{t,l}\right) + \frac{\sum_l^L \sum_r^R m_r^{*l} c_r^l}{\sum_l^L \sum_r^R c_r^l} + \frac{\sum_t^T \sum_{d_t}^{|D_t|} C_{d_t}^t \left(1 - s_{d_t}^t\right) c_{penalty}}{\sum_t^T \sum_{d_t}^{|D_t|} c_{penalty}} \quad (22)$$

6. Experimental Verification

6.1. Experimental Setting

In this study, the performance of the proposed system was verified in a simulation experiment. The experiment was conducted on the assumption that reconfiguration was executed in the gantry-type system as shown in Figure 6.

The simulation program was written with C#. In the solution evaluation step, Arena simulator was applied to the simulation model presented in Figure 7. This simulation model reflects a gantry-type manufacturing system with six processes with rooms for five parallel machines where the decision modules to assign jobs to a machine are positioned before each stage.

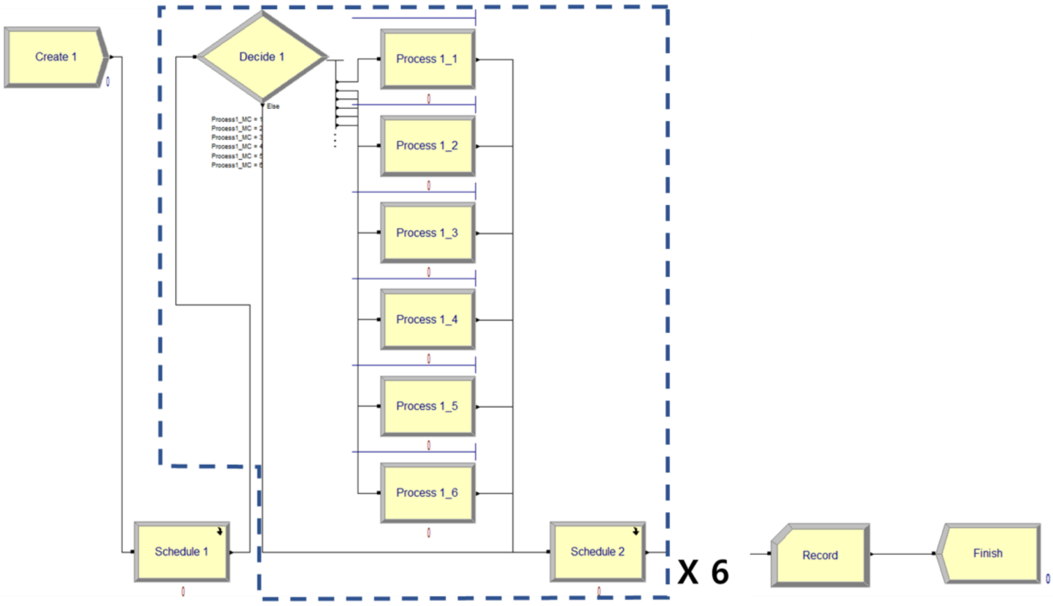

Figure 7. A simulation model with Arena.

The computer used for the experiment had an Intel Core i5 4690 CPU (Central Processing Unit) and 12 GB RAM (Random Access Memory). The production system in the simulation experiment had the attributes shown in Table 4, and the time unit of a schedule was a day.

Table 4. Setting of simulation factory.

Category	Item	Setting
Characteristics of factory	Length of the whole line	6
	Number of installable machines per process	5
	Type of machines (per each process)	3
	Reconfiguration cost	5000
	Penalty of an unfulfilled demand	1000
IoT Senor	Delay in recognizing reconfiguration situation when IoT sensor is not used	5 days
	Delay in finishing reconfiguration when IoT sensor is not used	5 days
	Installation cost of IoT sensors	1000

Based on the data in Table 4, the simulation experiment was conducted in various factory environments. Dispatching rules of FIFO, SPT, and EDD were applied and compared. In the case of factory demands, two types were applied according to the number of production processes, and their values were randomly set as shown in Table 5.

Table 5. Parameters of demands.

Category	Item	Setting
Processing Time	Minimum process time per process	1 day
	Maximum process time per process	7 days
Process	Number of production processes	3 processes/6 processes
	Number of types per each process	3 types per process
Due	Minimum due	Total process time + 1 day
	Maximum due	Total process time + 20 days
Quantity	Total number of demands	50
	Type of demands	10

In the experiment, the reconfiguration situations shown in Table 6 were taken into consideration. The experiment considered three types of scheduling rules, four reconfiguration situations, and two types of manufacturing process length based on the manufacturing system in Table 4.

Table 6. Reconfiguration situations in simulation experiment.

No.	Situation		Prediction	Remarks
1	Change of production ratio	Unpredicted ratio changes one week after starting	Unpredictable	
2	Introduction of additional processes	Insert additional processes one week after starting	Predictable	
3	Breakdown	Sudden breakdown of the machine and subsequent reconfiguration	Unpredictable	Delay in recognition when IoT is not used
4	Preventive maintenance	Preventive maintenance and reconfiguration are planned after one week	Predictable	

6.2. Result Analysis

The system suggested in this study provided its efficiency in various reconfiguration situations. The experimental results according to IoT application and reconfiguration situations are presented in Table 7. The 'performance ratio compared without reconfiguration' means a value when the number of the demands fulfilled without reconfiguration within the due date is compared with the number of the fulfilled demands with reconfiguration. The mean value of the satisfied demands without reconfiguration was 35.62 and was used as a reference value for performance comparison.

Table 7. Experimental result according to IoT application and reconfigurable situations.

IoT	Situation	Number of Satisfied Demands	Performance Ratio Compared without Reconfiguration (%)	Total Cost	Calculation Time (sec)
Applied	1	29.53	82.92	2047.3	92.02
	2	30.69	86.15	2057.1	89.25
	3	28.19	79.14	3972.4	97.6
	4	29.97	84.15	4020	93.68
Not applied	1	27.37	76.85	2116.9	97.81
	2	29.91	83.96	1975.5	87.17
	3	22.77	63.92	1884.8	89.38
	4	28.12	78.94	2025.3	86.91
Average		28.32	79.50	2,512.41	91.73

The IoT based system in four reconfiguration situations was expected to spend an additional cost. However, there was no big difference in cost, however, in performance comparison, the system performance was significantly different before reconfiguration and after reconfiguration. Situations 1 and 2 (changes in demands) had lower average performance than situations 3 and 4 (changes in machines), which seems to be because of delays in the schedule made by the forced reconfiguration of machines, and more machine replacement caused more cost. When IoT was not applied, situations 2 and 4 (easy prediction and the prior application to a plan) were less different from situations 1 and 3.

There were more delayed demands and lower cost gains when IoT was not applied than when IoT was applied. This is because the delay made was due to a situational awareness delay and the prolonged time for stabilization after reconfiguration often caused the demand to exceed the deadline in reconfiguration situation 3.

The different performance of the model according to the number of processes for production is presented in Table 8. When there were more processes, it took more time to devise a reconfiguration plan. The greater the number of processes, the lower the reconfiguration performance. This is because, in a greater number of processes, individual demands are highly likely to be influenced by reconfiguration delay, which in turn leads to a higher cost.

Table 8. Experimental result according to the number of processes and reconfigurable situations.

Number of Processes	Situation	Number of Satisfied Demands	Performance Ratio Compared without Reconfiguration (%)	Total Cost	Calculation Time (sec)
3	1	28.85	80.99	1781.2	82.14
	2	32.31	90.71	1143.4	86.59
	3	27.71	77.80	2897.9	74.90
	4	31.13	87.40	2917.7	81.27
6	1	28.06	78.77	2383.0	107.48
	2	28.28	79.41	2889.2	100.60
	3	23.24	65.25	2959.3	108.03
	4	26.96	75.69	3127.6	92.81

The experimental results, according to scheduling rules, are presented in Table 9 where it can be observed that the results followed the characteristics of the scheduling rules. According to the experiment, the rule that showed the best performance in terms of the number of satisfied demands was EDD, which was directly related to the due date. FIFO and SPT had similar performances, though SPT was a little better. Regarding the comparison between with and without reconfiguration, the three rules had a similar value in terms of the performance ratio. This reveals that a reconfiguration plan is not greatly influenced by the type of scheduling rules and indicates that the proposed system can be applied to various scheduling rules and situations.

Table 9. Experimental result according to scheduling rule and reconfigurable situations.

Rule	Situation	Number of Satisfied Demands	Performance Ratio Compared without Reconfiguration (%)	Total Cost	Calculation Time (sec)
FIFO	1	27.33	83.35	2082.70	93.56
	2	24.58	74.95	2000.37	95.41
	3	23.58	71.91	2928.02	90.73
	4	29.02	88.49	3022.61	85.06
SPT	1	27.71	82.13	2083.00	92.93
	2	26.48	78.51	1992.40	96.31
	3	24.21	71.78	2927.75	90.36
	4	29.03	86.04	3022.59	84.07
EDD	1	30.32	75.18	2080.55	97.94
	2	39.83	98.75	2056.09	89.06
	3	28.64	71.01	2929.95	93.30
	4	29.09	72.13	3022.80	91.99

According to the experimental results, the IoT based reconfiguration system was able to respond to various reconfiguration situations, which the dedicated production system failed to respond to. The reconfigured plan, which was created through the proposed algorithm, showed 79% performance improvement compared to the case where reconfiguration was not applied. Compared to the FMS that does not require reconfiguration, the expected performance degradation was not significant. Therefore, if the system proposed in this study is applied to RMS, it is expected to help to produce various goods through reconfiguration without a large delay.

7. Conclusions

Various types of manufacturing systems have been tried to realize flexible manufacturing. RMS is one of the solutions to enable flexibility. This study proposed an algorithm for reconfiguration decision-making of RMS with IoT sensors. Generally, machines in RMS can change their available tasks through reconfiguration.

A RMS requires reconfiguration to offer flexibility. Therefore, managers of RMS need to make reconfiguration decisions efficiently. To make a good decision, it is required to collect various information in and out of machines. This study proposed a variable neighborhood search (VNS) based reconfiguration decision-making algorithm to create a reconfiguration plan with the use of an IoT sensor-based data acquisition system for RMS. We used a simulation approach to evaluate a reconfiguration plan and VNS to find an improved solution. Through the simulation experiment in various environments, the performance of the proposed algorithm was verified. In the experiment, the algorithm proposed in this study delivered excellent performance in reconfiguration situations.

As a result of the simulation experiments, the advantages of IoT sensor applications outweighed the cost of the sensors. IoT sensors make it possible to respond quickly to an unpredicted situation and rapid ramp-up. The simulation experiment revealed that the suggested system suppressed the performance degradation made after reconfiguration at a stable level in various scheduling rules.

This study proposed a reconfiguration decision-making algorithm for RMS, which reflects the characteristics of the flow shop with reconfiguration functions. It applied realistic assumptions in the ramp-up process and recognition of the situation. It is expected that the proposed algorithm can be applied to an actual factory. However, in this study, reconfiguration was limited to machines. In the future, a reconfiguration decision-making algorithm that covers jigs, fixtures, conveyors, and tools will be researched. Additionally, the following research will be conducted on other different reconfiguration situations than the situations set in this study. In this way, it is expected to develop an applicable reconfiguration decision-making system for a RMS.

Author Contributions: Conceptualization, S.H., T.-W.C., J.P.; Methodology, S.H., T.-W.C; Software, S.H..; Validation, T.-W.C; Y.S.H., and J.P.; Formal analysis, T.-W.C.; Investigation, T.-W.C.; Resources, S.H.; Data curation, S.H..; Writing—original draft preparation, S.H.; Writing—review and editing, T.-W.C.; Visualization, S.H.; Supervision, J.P.; Project administration, T.-W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the GRRC program of Gyeonggi Province ((GRRC KGU 2017-B01), Research on Industrial Data Analytics for Intelligent Manufacturing).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bi, Z.M.; Lang, S.Y.T.; Shen, W.; Wang, L. Reconfigurable manufacturing systems: The state of the art. *Int. J. Prod. Res.* **2008**, *46*, 967–992. [CrossRef]
2. Tao, F.; Cheng, Y.; Zhang, L.; Nee, A.Y. Advanced manufacturing systems: Socialization characteristics and trends. *J. Intell. Manuf.* **2017**, *28*, 1079–1094. [CrossRef]
3. Zhong, R.Y.; Xu, X.; Klotz, E.; Newman, S.T. Intelligent manufacturing in the context of industry 4.0: A review. *Egineering* **2017**, *3*, 616–630. [CrossRef]
4. Koren, Y.; Gu, X.; Guo, W. Reconfigurable manufacturing systems: Principles, design, and future trends. *Front. Mech. Eng.* **2018**, *13*, 121–136. [CrossRef]
5. Minhas, S.H.; Lehmann, C.; Städter, J.P.; Berger, U. Reconfigurable strategies for manufacturing setups to confront mass customization challenges. In Proceedings of the 21st International Conference on Production Research (ICPR 21), Stuttgart, Germany, 31 July–4 August 2011.
6. Xia, T.; Xi, L.; Pan, E.; Ni, J. Reconfiguration-oriented opportunistic maintenance policy for reconfigurable manufacturing systems. *Reliab. Eng. Syst. Saf.* **2017**, *166*, 87–98. [CrossRef]
7. Koren, Y.; Wang, W.; Gu, X. Value creation through design for scalability of reconfigurable manufacturing systems. *Int. J. Prod. Res.* **2017**, *55*, 1227–1242. [CrossRef]
8. Kumar, A.; Pattanaik, L.N.; Agrawal, R. Multi-objective Scheduling Model for Reconfigurable Assembly Systems. In *Innovations in Soft Computing and Information Technology*; Springer: Singapore, 2019; pp. 209–217.
9. Huang, S.; Wang, G.; Shang, X.; Yan, Y. Reconfiguration point decision method based on dynamic complexity for reconfigurable manufacturing system (RMS). *J. Intell. Manuf.* **2018**, *29*, 1031–1043. [CrossRef]
10. Hsieh, F.S.; Lin, J.B. A dynamic scheme for scheduling complex tasks in manufacturing systems based on collaboration of agents. *Appl. Intell.* **2014**, *41*, 366–382. [CrossRef]
11. Hsieh, F.S. A Meta-Heuristic Approach for Dynamic Process Planning in Reconfigurable Manufacturing Systems. In Proceedings of the 2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), Taipei, Taiwan, 18–20 December 2017; pp. 164–169.

12. Tang, H.; Li, D.; Wan, J.; Imran, M.; Shoaib, M. A Reconfigurable Method for Intelligent Manufacturing Based on Industrial Cloud and Edge Intelligence. *IEEE Internet Things J.* **2019**, *7*, 4248–4259. [CrossRef]
13. Han, S.M.; Seo, J.W.; Park, J.W.; Lee, J.G.; Kang, G.C.; Lee, S.H.; Moon, J.S. Pallet-fixture Allocation in Reconfigurable Manufacturing Cells: An Integrative Approach. *J. KSPE* **2012**, *29*, 357–366. [CrossRef]
14. Kurniadi, K.A.; Ryu, K. Development of IoT-based reconfigurable manufacturing system to solve reconfiguration planning problem. *Procedia Manuf.* **2017**, *11*, 965–972. [CrossRef]
15. Bi, Z.; Da Xu, L.; Wang, C. Internet of things for enterprise systems of modern manufacturing. *IEEE Trans. Ind. Inform.* **2014**, *18*, 1537–1546.
16. Scholz, S.; Mueller, T.; Plasch, M.; Limbeck, H.; Adamietz, R.; Iseringhausen, T.; Kimmig, D.; Dickerhof, M.; Woegerer, C. A modular flexible scalable and reconfigurable system for manufacturing of microsystems based on additive manufacturing and e-printing. *Robot. Comput. Integr. Manuf.* **2016**, *40*, 14–23. [CrossRef]
17. Mourtzis, D.; Milas, N.; Vlachou, A. An internet of things-based monitoring system for shop-floor control. *J. Comput. Inf. Sci. Eng.* **2018**, *18*, 1–10. [CrossRef]
18. Mourtzis, D.; Milas, N.; Athinaios, N. Towards Machine Shop 4.0: A General Machine Model for CNC machine-tools through OPC-UA. *Procedia CIRP* **2018**, *78*, 301–306. [CrossRef]
19. Pellicciari, M.; Leali, F.; Andrisano, A.O.; Pini, F. Enhancing changeability of automotive Hybrid Reconfigurable Systems in digital environments. *IJIDeM* **2012**, *6*, 251–263. [CrossRef]
20. Ferreira, P.; Reyes, V.; Mestre, J. A web-based integration procedure for the development of reconfigurable robotic work-cells. *IJARS* **2013**, *10*, 295. [CrossRef]
21. Shin, H.J.; Cho, K.W.; Oh, C.H. SVM-based dynamic reconfiguration CPS for manufacturing system in Industry 4.0. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 5795037. [CrossRef]
22. Koren, Y.; Heisel, U.; Jovane, F.; Moriwaki, T.; Pritschow, G.; Ulsoy, G.; Van Brussel, H. Reconfigurable manufacturing systems. *CIRP Ann.* **1999**, *48*, 527–540. [CrossRef]
23. Sastoque Pinilla, L.; Llorente Rodríguez, R.; Toledo Gandarias, N.; López de Lacalle, L.N.; Ramezani Farokhad, M. TRLs 5–7 advanced manufacturing centres, practical model to boost technology transfer in manufacturing. *Sustainability* **2019**, *11*, 4890. [CrossRef]
24. Andersen, A.; Nielsen, K.; Brunoe, T.D. Prerequisites and barriers for the development of reconfigurable manufacturing systems for high speed ramp-up. *Procedia CIRP* **2016**, *51*, 7–12. [CrossRef]