Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

RESCHEDULING OF FLEXIBLE FLOW SHOP WITH SEQUENCE-DEPENDENT SETUP TIMES AND JOB SPLITTING

Jun Kim
Je-Hun Lee
Seong-Lak Choi
Hyun-Jin Jung
Yun-Bae Kim
Hyun-Jung Kim*

Department of Industrial Engineering
Sungkyunkwan University
Seo-bu Street 2066
Suwon, 16419, KOREA

Byung-Hee Kim
Gu-Hwan Chung

VMS Solutions Co., Ltd.
U-Tower Building A
Sinsu Street, 767
Yongin, 16827, KOREA

ABSTRACT

This paper proposes rescheduling algorithms for improving schedules obtained by dispatching rules with a commercial software program, MozArt, developed by VMS Solutions Co., Ltd.. Schedules for flexible flow shops with sequence-dependent setup times and job splitting are analyzed. The objective of the algorithms is to reduce the completion time by decreasing the number of setups and setup times. We first identify four types of problems with badly assigned job sequences and unnecessary idle times in given schedules derived from dispatching rules, and solve the problems by changing the sequence of jobs or splitting jobs. The performance of the proposed algorithms is tested with randomly generated instances based on real data from a factory in Korea.

1 INTRODUCTION

Parallel machine scheduling problems, a special case of flexible flow shops, are NP-hard even when only two machines are considered (Bruno *et al.* 1974). It is therefore hard to obtain an optimal sequence of jobs even using only two parallel machines within a reasonable time. Because of the complexity of the scheduling problems, many studies have developed heuristic algorithms that are intuitive and efficient in practical applications. Dispatching rules, which consist of heuristic algorithms, sequence jobs in order of their priorities and assign them to the machines that become available. The well-known dispatching rules, which are the shortest or longest processing time (SPT or LPT) first rule and earliest due date (EDD) rule, are widely used for complex scheduling, for example, in semiconductor or LCD manufacturing, crude oil, casting, and battery processing. The dispatching rules provide efficient solutions for short times for many practical applications, but are not usually optimal. Hence, workers who are responsible for scheduling jobs modify and revise the schedules based on their own experience in order to reduce the number of setups or setup times and eventually the completion times of jobs. Therefore, in this research, we develop rescheduling algorithms for improving schedules that are derived from dispatching rules by using a software program, MozArt (Manufacturing Operation Zone by Abstract Real Time), developed by VMS Solutions Co., Ltd.. Schedules for flexible flow shops with sequence-dependent setup times are considered. We first identify four types of problems, dealing with badly assigned job sequences and unnecessary idle time that occur in schedules derived from the dispatching rules, and solve the problems by changing a partial sequence of jobs or by splitting jobs. The performance of the proposed algorithms is tested with randomly generated instances based on real data from a factory in Korea.

In the rest of this paper, we review the previous studies in Section 2, and describe the software, MozArt, in Section 3. We then define the four types of problems, and present a problem searching algorithm and a schedule adjusting algorithm in Section 4. The results of numerical experiments will be shown in Section 5.

2 LITERATURE REVIEW

Studies on generating schedules of jobs by means of dispatching rules have been extensively conducted. Pickardt *et al.* (2010) coupled simulation-based genetic programming with dispatching rules for semiconductor scheduling in order to minimize weighted tardiness. Dispatching rules with different combinations of the priority-based rules, such as first-in-first-out (FIFO), SPT, EDD, critical ratio (CR), weighted shortest processing time (WSPT), and weighted modified due-date (WMDD) rules, were automatically generated by the proposed genetic algorithm. Vinod and Sridharan (2011) developed a simulation modeling-based scheduling method using a due date assignment dispatching rule for a dynamic job shop production system. Chen and Matis (2013) proposed a dispatching rule, 'weight-biased modified RR rule', to meet the due dates of high priority jobs. Lin and Lin (2013) presented a mixed-integer programming model for unrelated parallel machine scheduling with release dates, and developed dispatching rules to get a good schedule in a reasonable computational time. Li *et al.* (2013) analyzed complex manufacturing systems, especially semiconductor manufacturing systems, and proposed a data-based scheduling framework that consists of an adaptive dispatching rule (ADR) and a simulation method that estimates the performance of the schedule generated with the ADR. Other studies used metaheuristic algorithms for machine scheduling (Wu *et al.* (2012) and Hao *et al.* (2014)).

Rescheduling jobs, in which a partial or whole sequence of jobs is adjusted when there are unexpected problems, such as machine breakdown and jobs arriving late, have been extensively studied. The previous studies on rescheduling can be categorized into three parts: (1) defining unexpected cases that require rescheduling, (2) rescheduling methods reacting to those cases, and (3) evaluating the rescheduling methods (Pinedo 2015). Hall and Potts (2010) proposed a rescheduling method for handling unexpected delays in starting time. Dong and Jang (2012) developed two heuristic algorithms with a tabu search heuristic and a simulated annealing approach in order to minimize the impact of machine breakdown in job shop schedules. Katragjini *et al.* (2013) used a greedy algorithm to propose a rescheduling method to achieve a good trade-off between schedule quality and stability. Gurel and Cincioglu (2014) first introduced the number of delayed jobs as an evaluation measure for the stability of rescheduling and proposed a mixed-integer second-order cone-programming model to minimize the measure. Most of these previous studies assumed that rescheduling is performed only when initial schedules are affected by exceptional events. In this study, we analyze the given schedules and improve them by adjusting sequences of jobs and by splitting jobs.

3 PROBLEM DESCRIPTION WITH MOZART

In this section, the commercial software program, MozArt, is introduced and specific problems identified in given schedules from dispatching rules are described. MozArt is an integrated development and operations solution, which is implemented for production planning and scheduling applications, with a virtual model created by abstraction from real manufacturing. Figure 1 shows the coverage of MozArt in the processes of planning and scheduling. MozArt mainly covers five processes: master planning (MP), lot pegging, factory planning (FP), FAB simulation, and scheduling (Ko *et al.* 2013). Production target for a week is set in MP, and daily plans to meet the weekly target are decided in FP. Given the weekly and daily targets, the MozArt RTF (return to forecast) module carries out the process of lot pegging, which maps specific lots to demand. Considering a due date, production target quantity, lead time, and WIP, MozArt RTF calculates which lot should be pegged into a certain demand with a backward stepwise simulation. Assignment of the lots to machines, operating times of each machine, and changes of WIP level are estimated by the MozArt LSE (loading simulation engine) module with a forward stepwise what-if simulator. Using the what-if simulator, MozArt LSE drives a fab simulation by testing various scenarios for production before real production begins. The MozArt APS (advanced planning and scheduling) module generates lot-in-and-out plans and machine schedules to be executed. For machine scheduling, each machine chooses the dispatching rule that selects the next lot to be produced. The dispatching rules include the rules to meet the demand, reduce cycle time, keep the WIP balance, and minimize the number of job changes; these include rules such as EDD, FIFO, and minimum/maximum lot size constraints. With a weighted-sum approach, MozArt APS calculates which dispatching rule should be used for a machine.

The problem we consider is LCD manufacturing with flexible flow shop lines in which parallel machines are arranged to do several processes in series. Processes we examine intensively are photolithography processes that are regarded as bottlenecks, because improving the bottlenecks can greatly enhance the KPIs (key performance indicators), such as the completion time or setup related measures. There are 10 different photolithography processes, and 17 parallel facilities which can carry out all of the photolithography processes with sequence-dependent setup times. For this scheduling problem, a job is defined as a set of lots that are processed consecutively on a machine. Since a job is composed of lots, a job can be split up into several lots, or some lots can be combined into a job. The objective of the problem is to reduce completion times by decreasing setup times and the number of setups.

4 ALGORITHM FOR SCHEDULE IMPROVEMENT

4.1 Problem Classification

We first define the four types of problems with badly assigned job sequences and unnecessary idle time in schedules, which can be improved later. The problems are derived by analyzing the schedules of a real LCD manufacturing facility using MozArt.

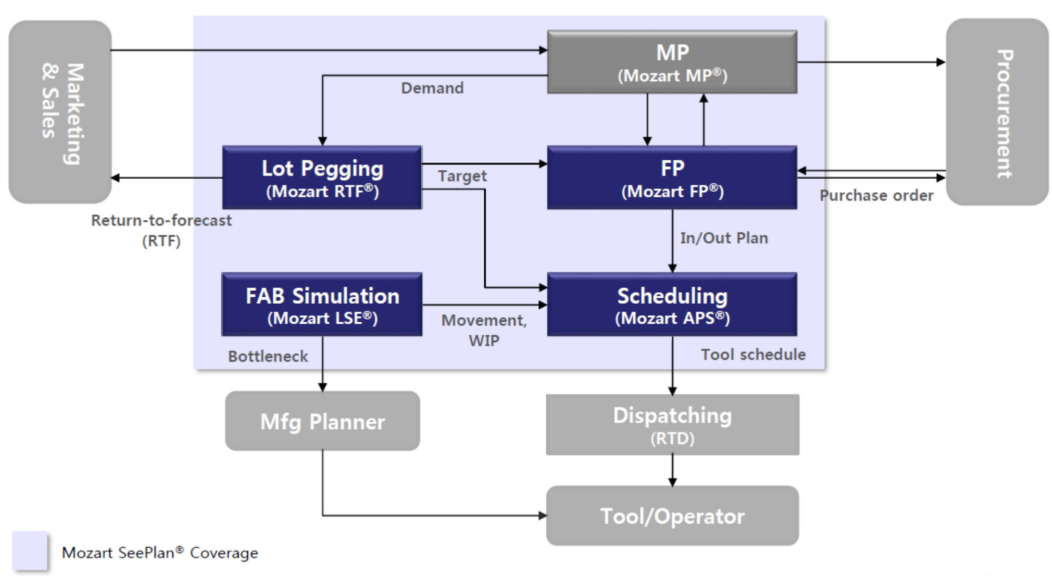

Figure 1: Outline of MOZART (Ko *et al.* 2013).

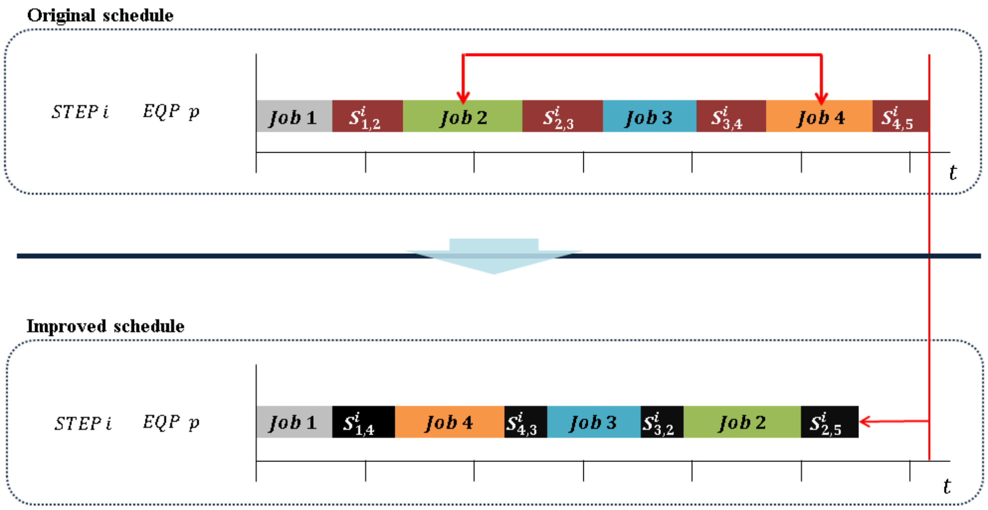

Figure 2: An example of *Problem 1*.

4.1.1 *Problem 1*. Job Sequence Change on a Machine

The first problem is one in which setup times of jobs in a schedule can be reduced by changing the job sequences on a machine. Figure 2 illustrates an example of *Problem 1* in *EQP p* of *STEP i* where *EQP p* and *STEP i* indicate the *p*th machine and process step *i*, respectively. We denote $S^i_{n,k}$ to indicate the setup time between jobs n and k in step i, where i is a set of photolithography processes. As you can see in Figure 2, when the positions of jobs 2 and 4 on a machine are switched, the completion time of the machine decreases because of the setup time reduction. If jobs 1 and 4 consist of lots of the same product, even the number of setups can be reduced. To switch the sequence of jobs n and k, where the position of job n is ahead of job k, in order to improve the original schedule, several conditions should be satisfied. When R^i_n, ST^i_n, and C^i_n denote the ready time, starting time, and completion time of job n in step i, respectively, and $R^{i}_n{'}$ stands for the ready time of job n in step i in the newly adjusted schedule, the conditions that can reduce the completion time of a machine are as follows:

$$S^i_{n-1,k} + S^i_{k,n+1} + S^i_{k-1,n} + S^i_{n,k+1} \le S^i_{n-1,n} + S^i_{n,n+1} + S^i_{k-1,k} + S^i_{k,k+1} \quad \forall k,n \tag{1}$$

$$R^{i+1}_n{'} \le ST^{i+1}_n \quad \forall n \tag{2}$$

$$R^i_k \le C^i_{n-1} + S^i_{n-1,k} \quad \forall k,n \tag{3}$$

The condition for which the sum of setup times should be reduced when the job positions are switched is presented in (1). When the sequence of jobs n and k is switched, setup times regarding the switched jobs are changed. The left-hand side stands for changed setup times and the right-hand side stands for the original setup times associated with the switched jobs. Conditions for which the switching of the job positions should not affect the schedule of the next and previous steps are presented as (2) and (3), respectively.

4.1.2 *Problem 2*. Job Position Switching Between Machines

The second problem is similar to *Problem 1*, in that the sum of setup times is reduced by changing job positions, but is different because jobs in different machines are switched. Figure 3 shows an example of *Problem 2*. When jobs 2 and 5 switch positions, the sum of setup times in both machines can be reduced, as can the completion time of the two machines. To search for whether switching the positions of jobs n and k assigned to different machines can improve the schedule, the following conditions should be satisfied:

$$S^i_{n-1,k} + S^i_{k,n+1} \le S^i_{n-1,n} + S^i_{n,n+1} \quad \forall k,n \tag{4}$$
$$S^i_{k-1,n} + S^i_{n,k+1} \le S^i_{k-1,k} + S^i_{k,k+1} \quad \forall k,n \tag{5}$$
$$R^{i+1'}_n \le ST^{i+1}_n \quad \forall n \tag{6}$$
$$R^{i+1'}_k \le ST^{i+1}_k \quad \forall k \tag{7}$$
$$R^i_k \le C^i_{n-1} + S^i_{n-1,k} \quad \forall k,n \tag{8}$$
$$R^i_n \le C^i_{k-1} + S^i_{k-1,n} \quad \forall k,n \tag{9}$$

Conditions (4) and (5) are related to the reduction of the sum of setup times on the two machines. Conditions (6) and (7), and (8) and (9) ensure that the altered job positions should not affect the schedules of the next and previous steps, respectively.

4.1.3 *Problem 3*. Job Position Switch Between Machines with Job Splitting

In the third problem, job splitting is considered in addition to that in *Problem 2*. An example of *Problem 3* is presented in Figure 4. When job 2 is split into two parts (or subjobs), and the processing time of one part is exactly the same as that of job 4, the total setup time can be reduced by switching the positions of jobs 2 and 4, even though the number of setups increases. To investigate *Problem 3*, we need several conditions. When the processing time of job n is longer than that of job k, the conditions for *Problem 3* are as follows:

$$S^i_{n-1,k} + S^i_{k,n} \le S^i_{n-1,n} \quad \forall k,n \tag{10}$$
$$S^i_{k-1,n} + S^i_{n,k+1} \le S^i_{k-1,k} + S^i_{k,k+1} \quad \forall k,n \tag{11}$$
$$R^{i+1'}_n \le ST^{i+1}_n \quad \forall n \tag{12}$$
$$R^{i+1'}_k \le ST^{i+1}_k \quad \forall k \tag{13}$$
$$R^i_k \le C^i_{n-1} + S^i_{n-1,k} \quad \forall k,n \tag{14}$$
$$R^i_n \le C^i_{k-1} + S^i_{k-1,n} \quad \forall k,n \tag{15}$$

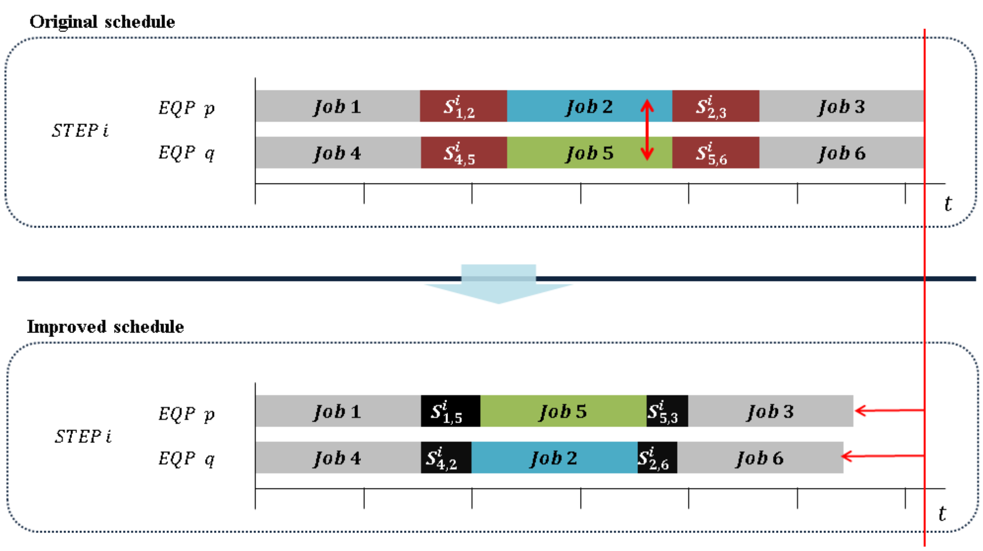

Figure 3: An example of *Problem 2*.

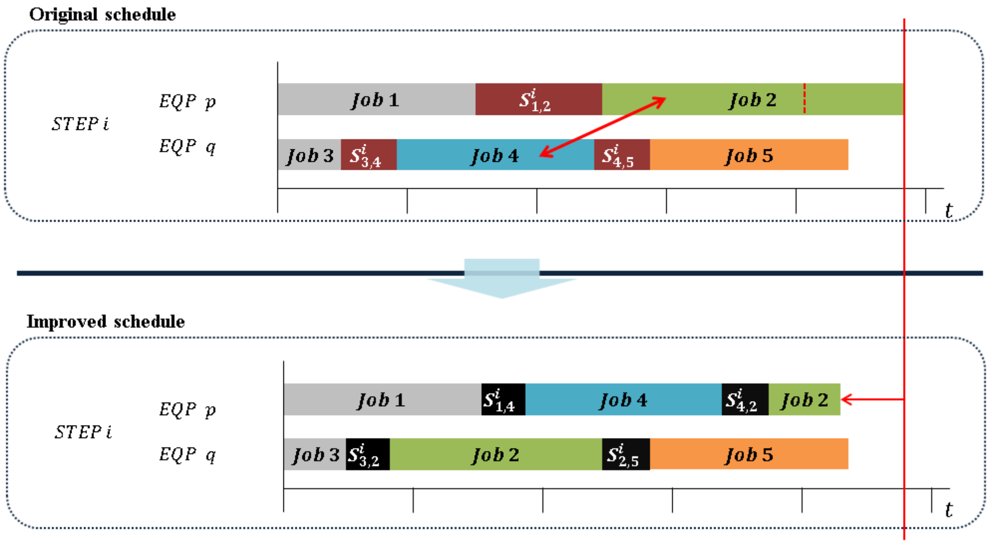

Figure 4: An example of *Problem 3*.

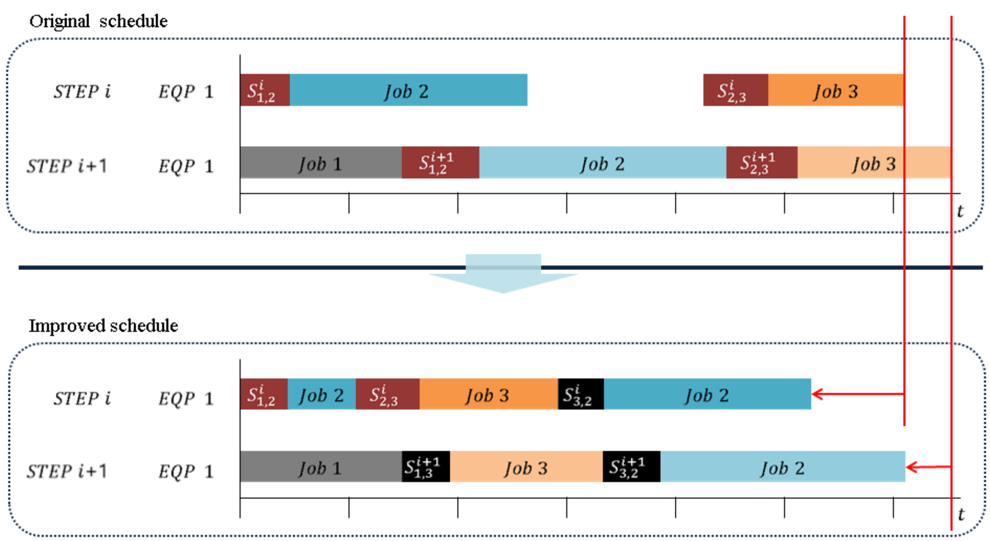

Figure 5: An example of *Problem 4*.

Conditions (10) and (11) are used to search for possible setup time reductions that can result from switching the job positions. Conditions (12) - (15) analyze the possibility of job changes that do not affect the other steps.

4.1.4 *Problem 4*. Job Sequence Change to Eliminate Idle Time

The last problem tries to reduce idle times of a schedule by job sequence changes. As you can see in the original schedule in Figure 5, there are sometimes idle times that are caused by dispatching jobs with inappropriate rules. The original schedule in Figure 5 can be improved by switching the positions of jobs 2 and 3 in step $i + 1$, and by moving the starting time of job 3 in step i forward by splitting job 2. As a result, the idle time on the machine can be eliminated, and its completion time can also be reduced. When idle times exist between jobs n and k $(n < k)$ in step i, the following conditions are needed to search for *Problem 4*:

$$S_{n,k}^{i} + S_{k,n}^{i} \le C_k^i - C_n^i \quad \forall k,n \tag{16}$$

$$S_{n-1,k}^{i+1} + S_{k,n}^{i+1} \le S_{n-1,n}^{i+1} + S_{n,k}^{i+1} \quad \forall k,n \tag{17}$$

$$R_n^{i+2'} \le ST_n^{i+2} \ \ \forall n \tag{18}$$

$$R_k^i \le C_{n-1}^i + S_{n-1,k}^i \quad \forall k,n \tag{19}$$

Condition (16) ensures that the setup time caused by splitting a job in step i should not exceed the idle time of the machine. Condition (17) indicates that the setup time should be reduced in step $i+1$ when the sequence of jobs n and k is switched. Conditions (18) and (19) ensure that the job position switch should not affect the next and previous steps, respectively.

4.2 Problem Searching Algorithm

In this section we present the problem searching algorithm. All of the possible pairs of jobs allocated in photolithography processes are searched to find out whether they belong to any one of the proposed four problems by using the conditions derived in Section 4.1. The problem searching algorithm is as follows:

- STEP 1. Generate a schedule using MozArt.
- STEP 2. Select a photolithography process to be searched.
- STEP 3. Generate all possible pairs of jobs in the selected process for searching problems.
- STEP 4. Examine whether the pairs of jobs belong to the four types of problems by using the problem searching conditions.
- STEP 5. Repeat STEP 2 through STEP 4 for the rest of the photolithography processes.

4.3 Schedule Adjusting Algorithm

When the problems are detected with the problem searching algorithm, the schedule is improved by using the schedule adjusting algorithm. When multiple problems are found for a job, the problem that can reduce the completion time the most is selected. The schedule adjusting algorithm has the following steps:

- STEP 1. Select a pair of jobs that are found to be a problem.
- STEP 2. Check whether a job of a selected pair is duplicated in other pairs of jobs. Go to STEP 3 if duplication is found; otherwise, go to STEP 4.
- STEP 3. Compute improved completion time of all the job-related pairs, and keep only the pair that lowers the completion time the most when the schedule is adjusted.
- STEP 4. Adjust the schedule of the pair of jobs.
- STEP 5. Iterate STEP 1 through STEP 4 for the rest of the pairs of jobs that have been identified as problems.

5 NUMERICAL EXPERIMENTS

Using the real process parameters of LCD manufacturing in Korea, we generated five sample datasets. A schedule based on the sample data with MozArt produces 11 products in 3 days. The five sample datasets have same total quantity of demands. A difference between the sample datasets is demands for each products. Sample 1 has the same demands for all the products, and the demands for the number of one, two, four, and six products are set to be 20% higher than other products in samples 2, 3, 4, and 5, respectively. As mentioned in Section 3, there are 10 photolithography processes and 17 parallel machines which can carry out all of the 10 processes with sequence-dependent setup times. About 60 jobs are carried out by photolithography processes in the given schedules. Using the datasets, we applied the problem searching and schedule adjusting algorithms, and investigated the number of problems searched for each dataset and improvement of KPIs. Sequence-dependent setup times for layer change that makes a facility can conduct a different photolithography process are set to be 10% of processing time for a photolithography process which is to be changed. The number of searched problems, which is defined in Section 4, for each dataset is summarized in Table 1. A total of 11 problems were found on average. An average number of 3.8, 1, 1.4, and 4.8 problems were detected for each type of problem, respectively. The number of problems found in sample 3 is the largest whereas sample 5 has the smallest number of problems. We can also recognize that the schedules from samples 2 and 3 have more idle times than others in Table 1 since problem 4 is mostly searched in sample 2 and 3. Tables 2 to 4 show how the KPIs are improved when the schedule adjusting algorithm is applied. As you can see in Table 2, the completion times of 3.4 facilities were reduced on average by up to 111.2 seconds per machine. The improvement of setup times is presented in Table 3. For sample 3, the sum of setup times was reduced by 829 seconds with the largest deviation, and the sum of setup times of sample 2 was decreased by 122 seconds. In the number of setups, samples 1 and 2 showed no difference after the schedule adjustment. Sample 3 had two setup reductions, and one setup was removed in samples 4 and 5.

Table 1: Number of problems found in sample data.

Sample data	Problem 1	Problem 2	Problem 3	Problem 4	Total
Sample 1	4	1	2	4	11
Sample 2	4	1	1	7	13
Sample 3	3	2	1	8	14
Sample 4	4	0	2	3	9
Sample 5	4	1	1	2	8
Average	3.8	1	1.4	4.8	11

Table 2: Improvement of completion times.

Sample data	Number of improved facilities	Reduction of completion time (sec)
Sample 1	4	-32
Sample 2	2	-63
Sample 3	5	-165
Sample 4	3	-130
Sample 5	3	-166
Average	3.4	-111.2

Table 3: Improvement of setup times.

Sample data	Setup time (sec)		
	Before	**After**	**Difference**
Sample 1	121,771	121,630	-141
Sample 2	114,906	114,784	-122
Sample 3	102,205	101,376	-829
Sample 4	129,146	128,754	-392
Sample 5	132,730	132,231	-499

Table 4: Improvement of the number of setups.

Sample data	Number of setups		
	Before	**After**	**Difference**
Sample 1	161	161	0
Sample 2	152	152	0
Sample 3	129	127	-2
Sample 4	147	146	-1
Sample 5	156	155	-1

6 CONCLUSION

This research has proposed rescheduling algorithms for improving given schedules derived from dispatching rules. We have first identified the four types of problems by analyzing sequence-dependent setup times of jobs and idle times of machines, and then revised the schedules by switching positions of jobs and splitting jobs. For the numerical experiments, the real data from a factory in Korea were taken, and five sample datasets were generated. We were able to find an average number of 11 problems for each sample dataset with the proposed algorithms. The completion times, the number of setups, and the setup times were reduced accordingly. Future research might include identifying more problem types, conducting numerical experiments with extensive demand scenarios and extending the results to job shop schedules.

ACKNOWLEDGMENTS

This work was partially supported by the ICT R&D program of MSIP/IITP [B0364-15-1008, Development of Open FaaS IoT Service Platform for Mass Personalization], by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education [2016R1D1A1B03930952], by a National Research Foundation grant [NRF-2015R1A2A2A01005871] funded by the Ministry of Education, Science and Technology, Korea, by Brain Korea 21 Plus, by the Technology Innovation Program [10065741, Development of Factory Automation and Manufacturing Equipment for the Parts Manufacturing Industries Smart-factory] and [10080296, Development of Advanced Operation Management Systems for Smart Factory Based on Clean Energy] funded by the Ministry of Trade, Industry &Energy (MI, Korea), and by the ICT R&D program of MSIP/IITP [R-20150505-000691], IoT-based CPS platform technology for the integration of virtual-real manufacturing facilities.

REFERENCES

Bruno, J., Coffman Jr, E. G., and Sethi, R. 1974. "Scheduling independent tasks to reduce mean finishing time". *Communications of the ACM*, Vol. 17, No. 7, 382-387.

Chen, B., and Matis, T. I. 2013. "A flexible dispatching rule for minimizing tardiness in job shop scheduling". *International Journal of Production Economics*, Vol. 141 No. 1, 360-365.

Dong, Y. H., and Jang, J. 2012. "Production rescheduling for machine breakdown at a job shop". *International Journal of Production Research*, Vol. 50, No. 10, 2681-2691.

Gürel, S., and Cincioğlu, D. 2015. "Rescheduling with controllable processing times for number of disrupted jobs and manufacturing cost objectives". *International Journal of Production Research*, Vol. 53, No. 9, 2751-2770.

Hall, N. G., and Potts, C. N. 2010. "rescheduling for job unavailability". *Operations Research*, Vol. 58, No. 3, 746-755.

Hal, X. -C., Wu, J. -Z., Chien, C. -F., and Gen, M. 2014. "The cooperative estimation of distribution algorithm: a novel approach for semiconductor final test scheduling problems". *Journal of Intelligent Manufacturing*, Vol. 25, No. 5, 867-879.

Joo, C. M., and Kim, B. S. 2015. "Hybrid genetic algorithms with dispatching rules for unrelated parallel machine scheduling with setup time and production availability". *Computers and Industrial Engineering*, Vol. 85, No. 1, 102-109.

Katragjini, K., Vallada, E., and Ruiz, R. 2013. "Flow shop rescheduling under different types of disruption". *International Journal of Production Research*, Vol. 51, No. 3, 780-797.

Ko, K., Kim, B. H., and Yoo, S. K. 2013. "Simulation based planning and scheduling system: MozArt®". In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 4103-4104.

Li, L., Zijin, S., Jiacheng, N., and Fei, Q. 2013. "Data-based scheduling framework and adaptive dispatching rule of complex manufacturing systems". *The International Journal of Advanced Manufacturing Technology*, Vol. 66, No. 9, 1891-1905.

Lin, Y. -K., and Lin, C. -W. 2013. "Dispatching rules for unrelated parallel machine scheduling with release dates". *International Journal of Advanced Manufacturing Technology*, Vol. 67, No. 1, 269-279.

Pickardt, C., Branke, J., Hildebrandt, T., Heger, J., and Scholz-Reiter, B. 2010. "Generating dispatching rules for semiconductor manufacturing to minimize weighted tardiness". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yucesan, 2504-2515.

Pinedo, M. 2015. *Scheduling*. Springer, Inc.

Vinod, V., and Sridharan, R. 2011. "Simulation modeling and analysis of due-date assignment methods and scheduling decision rules in a dynamic job shop production system". *International Journal of Production Economics*, Vol. 129, No. 1, 127-146.

Wu, J. -Z., Hao, X. -C., Chien, C. -F., and Gen, M. 2012. "A novel bi-vector encoding genetic algorithm for the simultaneous multiple resources scheduling problem". *Journal of Intelligent Manufacturing*, Vol. 23, No. 6, 2255-2270.

AUTHOR BIOGRAPHIES

JUN KIM is a Ph.D. student in Dept. of Industrial Engineering, Sungkyunkwan University. He received a BS in Industrial Engineering from Sungkyunkwan University. He is interested in scheduling methodologies and applications and operations management. His email address is tomatoes10@skku.edu.

JE-HUN LEE is a MS student in Dept. of Industrial Engineering, Sungkyunkwan University. He is interested in scheduling methodologies and applications and operations management. His email address is swi02050@naver.com.

SEONG-LAK CHOI is a MS student in Dept. of Industrial Engineering, Sungkyunkwan University. He is interested in simulation-based modeling and scheduling. His email address is choi_lak@naver.com.

HYUN-JIN JUNG is a MS student in Dept. of Industrial Engineering, Sungkyunkwan University. She is interested in simulation-based modeling, scheduling and agent based simulation. Her email address is hjjung217@naver.com.

YOON-BAE KIM is a Professor with the Dept. of Industrial Engineering, Sungkyunkwan University. He received the MS degree from the University of Florida, and the Ph.D. degree from Rensselaer Polytechnic Institute. His current research interests are demand forecasting, simulation methodology, simulation based acquisition, simulation output analysis, market analysis and scheduling. His email address is kimyb@skku.edu.

HYUN-JUNG KIM is an Assistant Professor with the Dept. of Industrial Engineering, Sungkyunkwan University. She holds a Ph.D. in industrial and systems engineering from Korea Advanced Institute of Science and Technology. Her research interests include discrete event systems modeling, scheduling and control. Her email address is kim.hj@skku.edu.

BYUNG-HEE KIM is the President of VMS Solutions Co., Ltd.. He received a BS from Sungkyunkwan University, MS and Ph.D. from Korea Advanced Institute of Science and Technology all in industrial engineering. He is interested in simulation-based scheduling and planning, manufacturing information systems, BPMS, and virtual manufacturing. His email address is kbhee@vms-solutions.com.

GU-HWAN CHUNG is a Head Researcher of VMS Solutions Co., Ltd.. He received a MS in industrial engineering from Korea Advanced Institute of Science and Technology. He is interested in simulation-based scheduling and planning, manufacturing information systems, BPMS, and virtual manufacturing. His email address is chunggh@vms-solutions.com.