Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Sequence learning-based schedule prediction for flexible manufacturing systems under uncertainties

To cite this article: Prita Meilanitasari and Seung-Jun Shin 2021 *J. Phys.: Conf. Ser.* **1996** 012010

Sequence learning-based schedule prediction for flexible manufacturing systems under uncertainties

Prita Meilanitasari[1], Seung-Jun Shin[2*]

[1.] Graduate School of Technology & Innovation Management, Hanyang University
[2.*]Division of Interdisciplinary Industrial Studies, Hanyang University

email: sjshin@hanyang.ac.kr[2]

Abstract. This study presents a method of production schedule prediction for flexible manufacturing systems with consideration of the uncertainty factors including limited machine capacity, diverse processing time and unplanned waiting time. The proposed method can predict product-level schedules using sequence learning, which derives data-learned models to predict production sequence proactively and granularly at the product-level. A decision tree technique is applied to derive such predictive models to pre-trace the locations of individual products allocated to each workstation. A deterministic technique is also applied to estimate waiting and production time per product as well as total production time consumed to fabricate all products assigned by work orders.

1. Introduction

Flexible Manufacturing Systems (FMS) are the manufacturing systems that can fabricate a variety of different product types simultaneously under program control at the various workstations **[1]**. In FMS, production scheduling is critical because production with multiple orders needs to be precisely planned and efficiently operated in such flexible and complex environment so that production per order should be complete within delivery dates. As production sequencing means the allocation of product sequences at individual workstations for deciding production schedules, First-Come-First-Serve (FCFS), Earliest Due Date (EDD) and Shortest Processing Time (SPT) are commonly used as production sequencing rules **[2]**. However, such static sequencing rules do not work well in real situations because frequent and rapid changes occur in production sequence. On the other hand, uncertainty indicates any unpredictable events that disturb operations and productions due to limited machine capacity, diverse processing time, sudden order, machine failure, deadlock, demand change and unknown reasons **[3]**. Uncertainties tend to make grower differences between predicted and actual deployment of manufacturing operations along with time **[4]**. Due to the uncertainties over time, the original and planned production sequence increasingly mismatches with its corresponding actual sequence. These problems can result in poor visibility and on-time delivery failure in FMS. Thus, it is important to predict production sequences accurately in the FMS that get affected by uncertainty factors.

Previous literature has contributed to providing mathematical and/or heuristic algorithms for production sequencing and scheduling with consideration of major uncertainty factors [5] [6] [7] [8]. However, they are limited to provide the predictability of production sequences and schedules at the product-level because their algorithms are mostly focused on sequence and schedule optimization at the production or order level. Also, they are limited to predict the locations of individual products in the environment where production sequences frequently change.

This study proposes a data-driven method to predict production sequence and schedule at the product-level with consideration of major uncertainty factors including limited machine capacities, diverse processing time and unplanned waiting time. For such purpose, we develop a sequence learning-based model that predicts production sequence at the product-level. This model uses a decision tree technique to pre-trace the locations of individual products allocated to each workstation. We also develop a deterministic model to calculate production time at the product-level and lastly output the predicted total production time needed to fabricate all the products ordered. We demonstrate the feasibility of the proposed method in a discrete-event simulation environment.

This article is structured as follows. Section 2 describes the problem definition, Section 3 proposes our research methodology and Section 4 explains our case studies. Section 5 concludes this article.

2. Problem definition

FMS comprise several workstations and these workstations contain at least more than one machines. To conceptualize our problem, we separate each machine into two types - Machine Type A $\left(\mathrm{M}_{\mathrm{j}}^{\mathrm{A}}\right)$ and Machine Type B $\left(\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}\right)$, depending on whether uncertainty factors are applied in a certain machine. $\mathrm{M}_{\mathrm{j}}^{\mathrm{A}}$ is a single machine where uncertainty is not applied and thus products are sequentially fabricated through the FCFS rule. As shown in Figure 1, M1 and M2 correspond to $\mathrm{M}_{\mathrm{j}}^{\mathrm{A}}$ and no change in product sequence would occur in their Gantt chart because both machines continuously fabricate products with their unlimited capacity. Accordingly, arrival time (Aij) of a product in each machine equals to the sum of travel time (djj'), waiting time (wij), and setup time (δ_{ij}).

Meanwhile, $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$ is a single machine that gets affected by the uncertainty factors including limited machine capacity and diverse processing time and thus requires the interaction with a buffer station. Figure 2 presents an example. M3 corresponds to $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$ with its maximum capacity (e.g., K=2) while M1 is $\mathrm{M}_{\mathrm{j}}^{\mathrm{A}}$. M3 forces P3 to enter into a buffer station when it exceeds the limited capacity. M3 also spends different processing time, depending on product types. Suppose that the product sequence in M1 is given by $\mathrm{S}_1 = \{\mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_3, \mathrm{P}_4\}$. The product sequence in M3 can be changed to $\mathrm{S}_3 = \{\mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_4, \mathrm{P}_3\}$ because M3 deals with P4 prior to P3 while P3 stands-by in the buffer station due to K=2. This sequence change causes an unintended consequence and successively results in the change of production schedule. Production schedule can be incalculable or unpredictable because the uncertainty applied to $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$ transforms the planned and ordered sequence to an unplanned and randomized sequence. As a chain reaction, production time for each product also becomes unpredictable. This comes from that some products need to be entered into the buffer station and their waiting time can be unknown until they are retrieved from the buffer station. These phenomena would increasingly occur as the number of $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$and the severity of such uncertainty increase. Therefore, it is necessary to re-transform such unpredictable product sequence and production time into calculable and predictable ones for filling the gap between reality and foresight.

Resultantly, the problem in this study is how we can predict product sequence and its corresponding production time accurately at the product-level in the FMS environment where $\mathrm{M}_{\mathrm{j}}^{\mathrm{A}}$ and $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$ coexist. Especially, $\mathrm{M}_{\mathrm{j}}^{\mathrm{B}}$ gets affected by the uncertainty factors including limited machine capacity, diverse processing time and unplanned waiting time. The following items are the assumptions and constraints for problem simplification.

a. There are n-jobs in each machine with different processing time for each job.

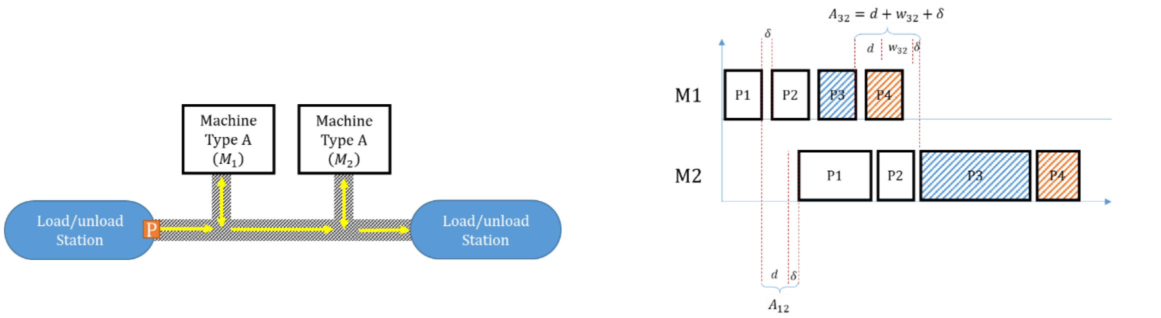
b. A machine processes a job for one product at a time and the type of the product determines processing time on the job.

c. Machines are continuously available without breakdown (machine breakdown is out-of-scope of this study).

d. When an operation in a machine begins, it proceeds without interruption.

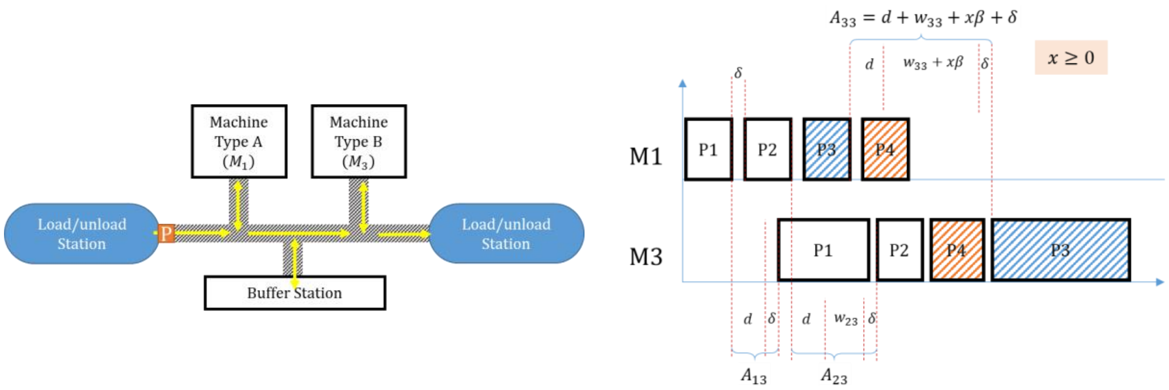
e. The capacity of a buffer station is unlimited.

f. Setup time on each machine is constant for each product.



(a) Basic layout loop type (b) Production schedule without uncertainties

Figure 1. Process flow and Gantt chart of Machine Type A $\left(M_j^A\right)$



(a) Layout loop type with buffer (b) Production schedule with uncertainties

Figure 2. Process flow and Gantt chart of Machine Type B $\left(M_j^B\right)$

3. Research Methodology

This section describes the proposed method to solve the problem defined in Section 2. Our basic idea is to apply learning algorithms. Learning algorithms can be effectively and efficiently applied because the learning from data is specialized to manage complexity, change and uncertainty in dynamic manufacturing environments [9]. Especially, machine learning algorithms have shown the superiority of finding the best dispatching rule for specific states in dynamic FMS through acquiring the knowledge needed to make future decisions from training data [10].

Figure 3 presents the conceptual model of our proposed method. The input data contain machine data, product data and buffer data. The resultant output values include predicted product sequence, predicted arrival time and predicted production time at both of product and production levels. The proposed method consists of the three modules: (1) product clustering, (2) product classification and sequence prediction, and (3) production time determination. In the product clustering, each product is clustered based on a specific range of processing time. It comes from that the prediction of product sequence tends to be more difficult as processing time becomes more diverse because this diversity affects the time when a product enters to M_j^B. By sorting individual products into some clusters, we can reduce the fluctuation of processing time within a certain threshold. In the product classification and sequence prediction, each product is classified to decide whether it should enter to M_j^B immediately or not once product clustering has been made. If not, the product would move into a buffer station to stand-by until the machine is available. Then, product sequence is predicted to pre-trace the locations of individual products allocated to each machine. In the production time determination, production time for individual products can be obtained after the predicted product sequence is decided. Finally, total production time for orders can be obtained by using mathematical formula. The following sub-sections explain the details of the three modules.

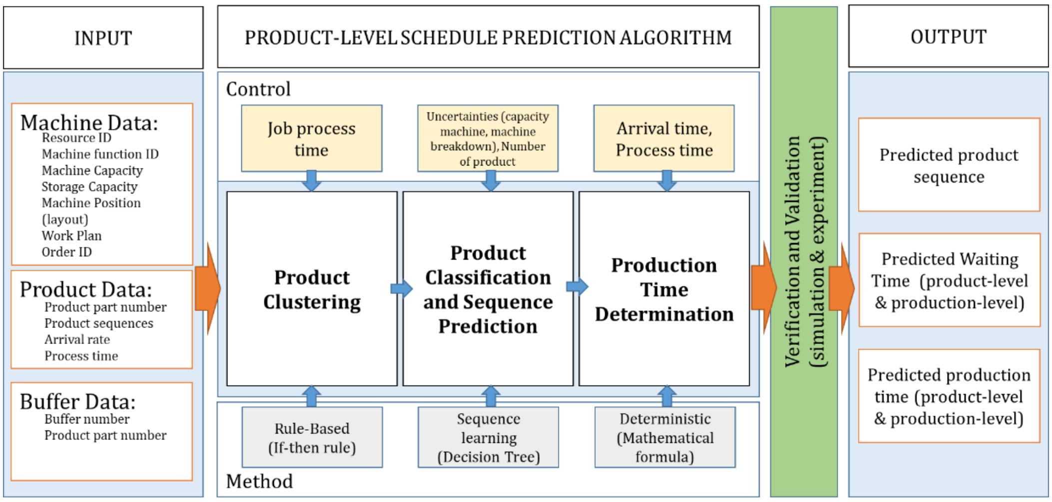


Figure 3. Conceptual model of the proposed method

3.1 Product Clustering

The product clustering sorts each product into some clusters, depending on the lower and upper bounds of processing time. Each cluster has a tendency of making the similar sequence pattern; meanwhile, different clusters make different sequence patterns due to the duration of processing time. Thus, this module is necessary to alleviate the significant influence of the uncertainty indicator - diverse processing time - on product sequence prediction. For this purpose, the lower and upper bounds of each cluster need to be decided by manual and this decision can be reasonably derived from observing the distribution of processing time in a training dataset. Processing time (p_{ijk}) is obtained from product data and then compared with average buffer station time (b_{ib}) to determine the ranges of individual clusters. For example, if b_{ib} is 120 second and p_{ijk} ranges from 0 to 120 second, we can make 3 clusters with the use of 40 and 80 second thresholds, as follows: (1) product with low processing time with range 0 to 40, (2) product with medium processing time with range 41 to 80, and (3) product with high processing time with range 81 to 120. It is also possible to make uneven distributions of clusters depending on the distribution of processing time values. We use If-Then rule for this module. This rule can be simply used when the distributions of processing time and buffer station time are even.

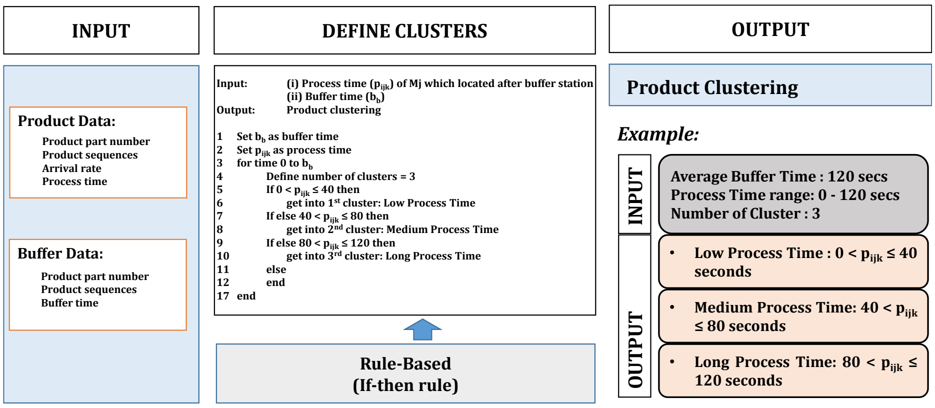


Figure 4. Algorithm for product clustering based on product and buffer data

3.2 Product Classification and Sequence Prediction

First, the product classification decides whether a product will enter to a buffer station or not, depending on finish time at the previous machine and arrival time at the current machine, as shown in Figure 5. This function is important to simultaneously resolve the two uncertainty indicators – limited machine capacity and unplanned waiting time. Due to the limited machine capacity, some products should move to the buffer station and wait without any specific timing until the current machine is not occupied by other products.

Here, we apply sequence learning. Particularly, we apply sequence prediction using decision tree. The sequence prediction is a part of sequence learning and enables to predict elements of a sequence based on preceding elements [11]. In this study, if we have one series of sequence input $S_i = \{P_{[i]}, P_{[i+1]}, .. P_{[n]}\}$ and we want to predict the $P_{[n+1]}$, we can make this prediction based on all the previously preceding elements in S_i. The decision tree sorts a product into two classes: (1) the product that enters to a machine immediately, and (2) the product that enters to a buffer station at least once. In the decision tree, the output 0 means the first class; whereas, N>0 does the second class. N stands for the number of entering to the buffer station. For example, 2 means the product needs to go around on a conveyor belt and be stocked in the buffer station in two times because the machine is still unavailable.

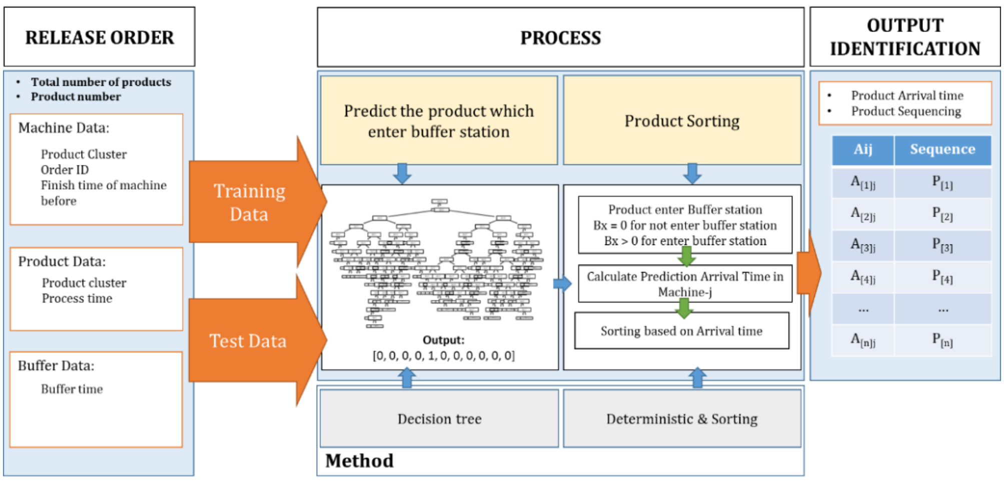


Figure 5. Algorithm for product classification and sequence prediction

Second, the sequence prediction derives a predicted sequence of products through estimating arrival time of individual products on a machine, as shown in Figure 5. In other words, this function derives the product positions regarding when the product will arrive in the designated machine and where it is located at a certain time. Once arrival time per product is decided, each product is allocated serially along with its arrival time and thus the product sequence can be constructed on individual machines. Here, we use the result of the decision tree generated at the product classification and use a deterministic method for calculating arrival time. Equation (1) expresses the formulae to calculate arrival time for the products entered to the machine without waiting; meanwhile, Equation (2) expresses the formulae for the products that stand-by in the buffer station with waiting.

$$\mathrm{A_{ij}} = \mathrm{pf_{i(j-1)k}} + \mathrm{d_{jj}} \tag{1}$$

$$A^{*}_{ij} = \mathrm{pf_{ib}} + \mathrm{d_{bj}} \tag{2}$$

where, Aij: arrival time of Pi in Mj, A^{*}_{ij}: arrival time of Pi in Mj after buffer station (Bb), pfi(j-1)k: production time of Pi in M(j-1), pfib: production time of Pi after the buffer station, djj': distance time between M(j-1) and Mj, dbj': distance time between Bb and Mj.

3.3 Production Time Determination

The production time determination calculates the time domain values at the product level as well as the production level. The time domain values involve arrival time, waiting time and production time. We create a deterministic algorithm for such time domain values, as shown in Figure 6. Here, Aij represents the arrival time per product, wijk does waiting time, pfijk does production time.

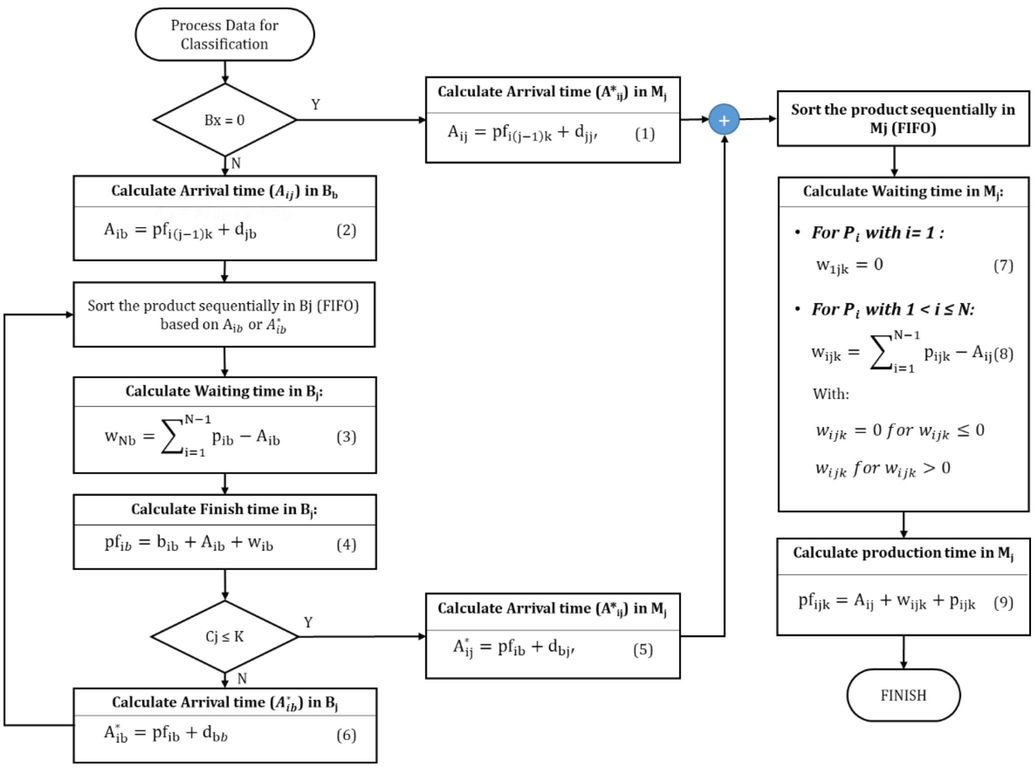


Figure 6. Algorithm for production time determination

4. Case Study

This section describes an experimental case study to check the feasibility and performance of the proposed method. We perform this case study in a simulation environment. We use Arena, which is a discrete-event simulation and automation software (https://www.arenasimulation.com).

4.1 Experimental setup

Figure 7 illustrates the configuration of the target FMS, which consists of 1-Working Center (WC). The WC comprises one buffer station and two machines: M_1 with Machine Type A ($K_1 = \infty$) and M_2 with Machine Type B. Due to the maximum capacity ($K_2 = 3$) in M_2, this WC can make two different product flows: Flow 1 (A-B-C-D) when M_2 is available and Flow 2 (A-B-E-F-C-D) when M_2 is unavailable. The number of orders is 15, and each order contains one of the numbers of products from 5 to 20. The products assigned in each order belong to the same cluster, as described in Section3.1. We repeat 200 times per order to acquire 3000 data samples under the simulation setting above. These datasets are separated into 80% as training datasets and 20% as testing ones. The minimum traveling time on a conveyor belt is 120 seconds, and djj' and djb are constant 12 seconds. Figure 8 presents the process flow implemented in Arena.

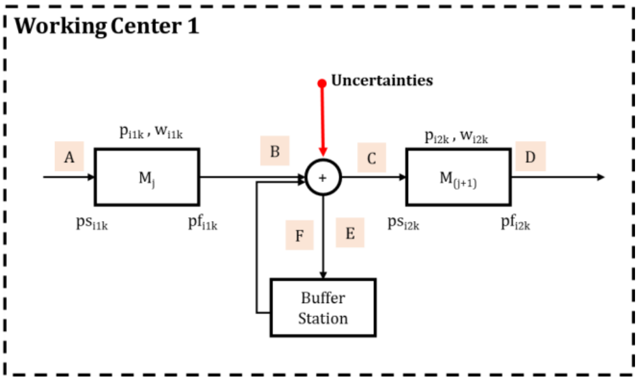


Figure 7. Product flow plan for ARENA simulation

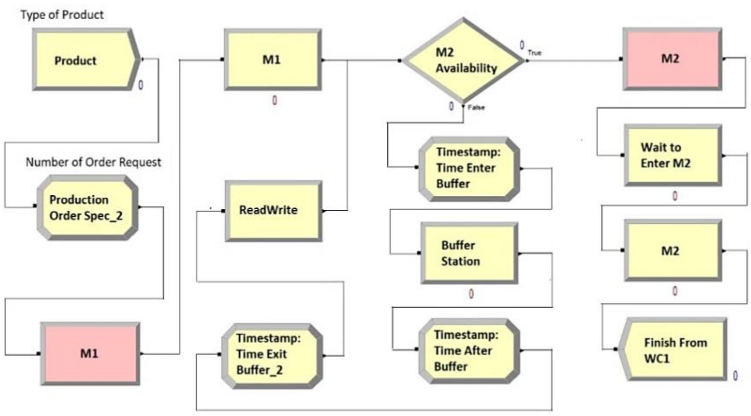


Figure 8. ARENA simulation in one working center

4.2 Modeling

We build the sequence prediction model based on the proposed method, as explained in Section 3. In the product clustering, we apply the If-Then rule to divide processing time in (pi2k) into three clusters: (1) product with low processing time (sec) with $0 \leq p_{ijk} \leq 40$, (2) product with medium processing time with $40 < p_{ijk} \leq 80$, and (3) product with high processing time with $80 < p_{ijk} \leq 120$. In the product classification and sequence prediction, we create a decision tree model to decide which products enter to the buffer station. The parameters for this decision tree are set as: criterion = entropy, maximum depth = 2, splitter = best. Figure 9 shows the decision tree result. In the production time determination, we calculate arrival time, waiting time and production time at the product and production levels. Table 1 presents our prediction results, compared with the simulated results.

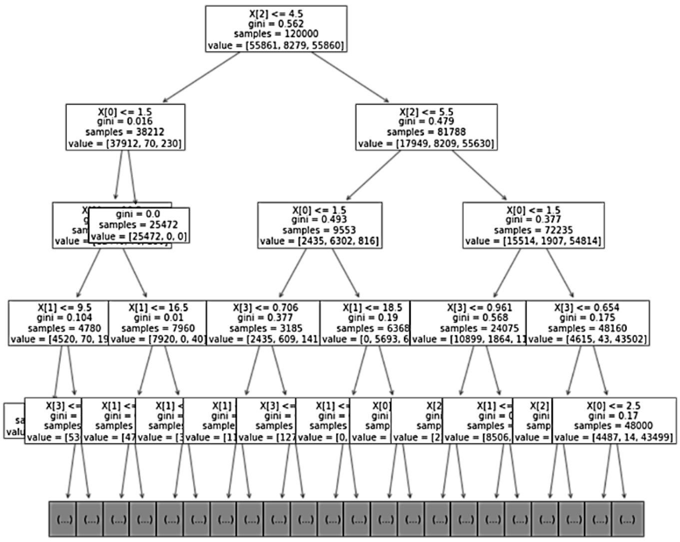


Figure 9. Decision tree result

Table 1. Prediction results compare with simulated results

$S_{i(j+1)}$	Simulated Data						Predicted Data					
	P.No	$A_{i(j+1)}$	$p_{i(j+1)k}$	$w_{i(j+1)k}$	$pf_{i(j+1)k}$	X_B	**P.No**	$A_{i(j+1)}$	$p_{i(j+1)k}$	$w_{i(j+1)k}$	$pf_{i(j+1)k}$	X_B
1	1	0.36	1.91	0.00	2.27	0	1	0.28	1.91	0.00	2.19	0
2	2	0.50	1.78	1.77	4.05	0	2	0.42	1.78	3.27	5.46	0
3	3	0.66	1.67	3.39	5.72	0	3	0.58	1.67	4.78	7.02	0
4	4	0.76	1.43	4.96	7.15	0	4	0.68	1.43	6.11	8.22	0
5	16	2.30	1.57	0.72	8.72	0	5	2.78	1.38	5.39	9.55	1
6	6	4.86	1.87	3.86	10.59	1	6	6.89	1.87	3.15	11.91	2
7	7	6.86	1.76	3.73	12.35	1	7	9.04	1.76	2.76	13.56	2
8	8	8.86	1.50	3.49	13.85	1	8	11.20	1.50	2.10	14.80	2
9	9	10.86	1.44	2.99	15.29	1	9	13.34	1.44	1.40	16.18	2
10	10	12.86	1.99	2.43	17.28	1	10	15.46	1.99	1.27	18.72	2
11	11	14.86	1.49	2.42	18.77	1	11	16.58	1.49	1.65	19.71	2
12	12	16.86	1.74	1.91	20.51	1	12	17.70	1.74	2.25	21.69	2
13	13	18.86	1.58	1.65	22.09	1	13	19.86	1.58	1.68	23.12	2
14	14	20.86	1.93	1.23	24.02	1	14	21.97	1.93	1.50	25.41	2
15	15	22.86	1.56	1.16	25.58	1	15	24.11	1.56	0.93	26.59	2
16	17	24.86	1.76	0.48	27.34	1	16	26.22	1.57	0.38	28.17	2

$S_{i(j+1)}$	Simulated Data						Predicted Data					
	P.No	$A_{i(j+1)}$	$p_{i(j+1)k}$	$w_{i(j+1)k}$	$pf_{i(j+1)k}$	X_B	P.No	$A_{i(j+1)}$	$p_{i(j+1)k}$	$w_{i(j+1)k}$	$pf_{i(j+1)k}$	X_B
17	18	26.86	1.98	0.47	29.33	1	17	28.37	1.76	0.00	30.13	2
18	19	28.86	1.76	0.23	31.09	1	18	30.49	1.98	0.00	32.47	2
19	20	30.86	1.86	0.09	32.96	1	19	32.60	1.76	0.00	34.36	2
20	5	32.86	1.38	4.85	34.34	2	20	34.74	1.86	0.00	36.60	2

4.3 Performance measurement

We measure the performance of our results using Root Mean Square Error (RMSE) and Total Relative Error (TRE) by comparing the simulated output (yreal) with the predicted output (ypred). Equations (3) and (4) express the equations for RMSE and TRE, respectively. Table 2 presents the values of RMSE and TRE in terms of waiting time and production time per product. Figure 10 shows the comparison between the simulated and the predicted output.

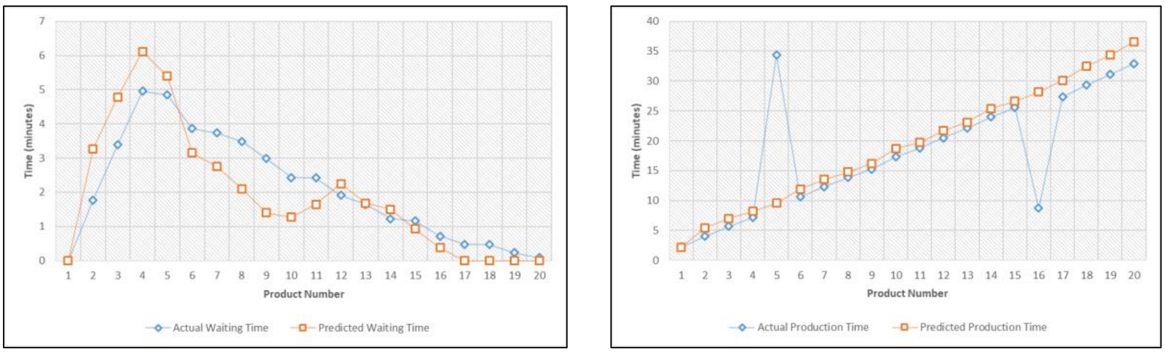
We conclude that our prediction results marginally fit with the simulated results. The predicted production time quite matches with the simulated one. However, it is observable that our model for product sequence prediction does not detect well the interruption of the products (P.No: 5 and 16) that suddenly cut in the serial product sequence. The interruption will affect the total production time which can be seen in Figure 10(b). This result comes from that our model is limited to predict waiting time accurately, as shown in Figure 10(a).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\left(y_{pred}-y_{real}\right)^2}{N}} \tag{3}$$

$$TRE = \frac{y_{pred}-y_{real}}{y_{real}} \tag{4}$$

Table 2. RMSE and TRE values

	Waiting time	Production Time
RMSE	1.72 min	1.21 min
TRE	0.09%	0.08%



(a) waiting time (b) production time

Figure 10. The comparison between the simulated (blue) and the predicted (orange) output

5. Conclusion

This study proposed a data-driven method to predict production sequence and schedule at the product-level with the uncertainty factors including limited machine capacities, diverse processing time and unplanned waiting time. We developed learning- and deterministic- based models to predict product sequence and calculate waiting time and production time at the product and production levels. The case study demonstrated the feasibility and performance of the proposed method in a simulated FMS environment.

The limitations of the study are as follows. In the case study, our sequence prediction model has shown low performance in pre-tracing the locations of individual products. In addition, only one working-center has been implemented and tested using the simulation software. Future works include to: (1) develop more rigid and accurate models to make better performance especially for the product sequence prediction, (2) implement more complex environments by adding more working centers, and (3) use the real data obtained from realistic FMS environments.

Acknowledgments
This work was supported by the Basic Research Program in Science and Engineering through the Ministry of Education of the Republic of Korea and the National Research Foundation (NRF-2018R1D1A1B07047100). We thank Dr. Heejin Seol for technical supports of Smart Factory Laboratory at Hanyang LINC+ Analytical Equipment in Seoul.

References
[1] M. P. Groover , Automation, Production Systems, and Computer-Aided Manufacturing, Englewood Cliffs, New Jersey: Prentice-Hall, 1980, pp. 564-565.
[2] S. P.M, “JOB SEQUENCING RULES: (eds) Encyclopedia of Production and Manufacturing Management,” Kluwer Academic Publishers 2000, Boston, 2000.
[3] K. S. C. L and S. S. M, “Development of a business model for diagnosing uncertainty in ERP environments,” International Journal of Production Research, vol. 40, no. 13, pp. 3015-3039, 2002.
[4] P. N. Pena, T. A. Costa , R. S. Silva and R. H. Takahashi, “Control of Flexible Manufacturing Systems under model uncertainty using Supervisory Control Theory and evolutionary computation schedule synthesis,” Information Sciences, vol. 329, p. 491–502, 2016.
[5] K. Hyun-Jung and L. Jun-Ho, “Three-machine flow shop scheduling with overlapping waiting time constraints,” Computers and Operations Research, vol. 101, p. 93–102, 2019.
[6] S. H., Y. S, L. M. and Q. , “Single Machine Scheduling Based on EDD-SDST-ACO Heuristic Algorithm,” Journal of Intelligent Systems, vol. 26, no. 4, pp. 657-667, 2017.
[7] D. Gyulai, A. Pfeiffer, G. Nick, V. Gallina, W. Sihn and L. Monostori, “Lead time prediction in a flow-shop environment with analytical and machine learning approaches,” IFAC-PapersOnLine, vol. 51, no. 11, pp. 1029-1034, 2018.
[8] M. Nouiri, A. Bekrar, A. Jemai, D. Trentesaux, A. C. Ammari and S. Niar, “Two stage particle swarm optimization to solve the flexible job shop predictive scheduling problem considering possible machine breakdowns,” Computers & Industrial Engineering, vol. 112, pp. 595-606, 2017.
[9] L. Monostori , “AI and machine learning techniques for managing complexity, changes and uncertainties in manufacturing,” Engineering Applications of Artificial Intelligence, vol. 16, p. 277–291, 2003.
[10] P. Priore, D. d. l. Fuente, J. Puente and J. Parreno, “A comparison of machine-learning algorithms for dynamic scheduling of flexible manufacturing systems,” Engineering Applications of Artificial Intelligence, vol. 19, p. 247–255, 2006.
[11] R. Sun, “Introduction to Sequence Learning,” in Sequence Learning Paradigms, Algorithms, and Applications, Berlin, Springer-Verlag, 2001, pp. 1-12.