International Journal of Industrial Engineering, 31(2), 413-428, 2024

MULTIPLE-OBJECTIVE SCHEDULING FOR BATCH PROCESS SYSTEMS USING STOCHASTIC UTILITY EVALUATION

Hongsuk Park[1], Youngchul Shin[2], Ilkyeong Moon[3, 4, *]

[1]Industry-Academic Cooperation Foundation
Kumoh National Institute of Technology
Gyeongsangbuk-do, South Korea

[2]Department of Industrial Engineering
Ajou University
Gyeonggi-do, South Korea

[3]Department of Industrial Engineering
Seoul National University
Seoul, South Korea

[4]Institute of Engineering Research
Seoul National University
Seoul, South Korea

Most research studies in the batch process control problem are focused on optimizing system performance. The methods address the problem by minimizing a single criterion, such as cycle time and tardiness, or bi-criteria such as cycle time and tardiness, and earliness and tardiness. We demonstrate the utilization of the Stochastic Utility Evaluation (SUE) function approach to the performance of batch process systems using multiple criteria. Tri-criteria problem is used as an example to illustrate the use of the SUE function. That is, we explore how SUE function with stochastic information can be implemented to improve existing approaches to the batch process control problem. The simulation results demonstrate that strategies based on the SUE function perform better than existing strategies based on static utility values.

Keywords: Multiple-objective scheduling, Batch process system, Stochastic utility evaluation function, Mean cycle time, Earliness, Tardiness

(Received on December 19, 2023; Accepted on March 17, 2024)

1. INTRODUCTION

In the last few years, as the semiconductor market has experienced steady growth, there has been a steady increase in the research of integrated circuits fabrication technology and methodologies (Yugma *et al*., 2015; Espadinha-Cruz *et al*., 2021; Fang *et al*., 2023). Wafer production involves a complicated process characterized by a lengthy cycle time (Lima *et al*., 2021), with much of this complexity arising from issues related to batch size, time, and sequences at various stages of the production cycle (Lee and Lee, 2022). The predominant focus of problem-solving approaches in this domain is to minimize production attributes, including cycle time, earliness, and tardiness (Mathirajan and Sivakumar, 2006). The batch process often becomes the bottleneck, especially as there are competing considerations between a partial batch and a full batch.

Numerous studies in the past have focused on using heuristics to minimize overall cycle time (or makespan), earliness, and tardiness (Adamu and Adewumi, 2016; Elmi and Topaloglu, 2017; Kaplano˘glu, 2016; Ogun and Alabas-Uslu, 2018; Hung *et al*., 2017; Pan *et al*., 2017; Liu *et al*., 2023; Uruk and Yal¸cıner, 2024). With a long production cycle time, when simultaneously considering cycle time, tardiness and earliness criteria, the weight of commodities for earliness and tardiness can change depending on the time. Due date and inventory cost are critical factors: when tardiness is considered as a criterion, it can adversely affect contractual obligations in the form of penalties or dissatisfied customers switching to competitors (Panwalkar *et al*., 1982; Wu and Wang, 1999) and when earliness is considered as a criterion, it can affect storage cost. For example, the price of commodities is fixed for all processing periods; however, after the due date, additional cost or value must be considered along with the loss of customer goodwill and other factors associated with

DOI: 10.23055/ijietap.2024.31.2.9889 **ISSN 1943-670X**

breaking a contract. On the other hand, additional costs for storage must be considered when jobs are completed too soon. This paper considers the changing weight for earliness and tardiness as “utility” and theorizes that the weighted values are useful when choosing the optimal production strategy within a system characterized by long production cycle times with changeable weights for earliness and tardiness.

A stochastic utility evaluation (SUE) function is introduced to incorporate the weight for earliness and tardiness. The SUE function is generated by the information at a decision epoch from the arrival distribution of commodities, due date, and remaining orders. As the due date nears, the weight assigned to the tardiness parameter increases. Conversely, the weight for the earliness parameter drops since the period for keeping in storage shortens. Among the commonly used optimal production strategies, Minimum Batch Size (MBS) (Van De Rzee *et al*., 1997) and Next Arrival Control Heuristic (NACH) (Fowler *et al*., 2000) have been used extensively. The SUE function has been adapted with MBS and NACH for the tri-criteria control problems. The approaches proposed for MBS and NACH using the SUE function are referred to as MBS-SUE and NACH-SUE, respectively. The MBS-SUE approach searches for the best MBS to attain maximum weighted value, and the NACH-SUE approach explores the best batch processing point among arrivals using near-future arrival information. The main contribution of our study can be summarized as follows:

- Unlike previous studies, we consider the SUE function for solving tri-criteria problems. Considering that this method has not previously been investigated, it can be considered as playing a pioneering role in the field of academics.
- We demonstrate how the SUE function can be combined with existing approaches. Various simulation experiments were conducted to verify this, and insights were derived from the results of the experiments.
- We show how the SUE function operates and performs in the existing function by comparing the performances of full batch policy, no-idle policy, MBS-SUE and NACH-SUE. Simulations for cycle time, earliness, and tardiness are performed.

The remainder of this paper is organized as follows. Section 2 presents an overview of previous research related to the dynamic control of batch process systems. Section 3 introduces the SUE function and the procedures to derive it for tardiness, earliness, and both tardiness and earliness simultaneously. In Section 4, a discussion about the methodology for applying the SUE function to existing approaches such as no idle, full batch, MBS, and NACH is presented. Furthermore, the benchmark approaches for the tri-criteria problem are modified in this section. The results of the benchmark approaches are discussed and analyzed in Section 5. In Section 6, research contributions and future directions for the solution of batch process control problems are discussed.

2. BACKGROUND AND LITERATURE REVIEW

Although there is limited research on employing utility functions to address strategic decision-making in batch processing systems within dynamic environments, a significant amount of analysis has been conducted on their performance, utilizing metrics like cycle time, tardiness, earliness, and various other factors. Mathirajan and Sivakumar (2006), who reviewed the literature related to batch process control, have grouped papers into stochastic and deterministic problems. Based on the nature of the product flow and availability of future information, Cerekci (2008) has created three subgroups of problems based on the availability of arrival information from an upstream process: no future arrival information, full knowledge on future arrivals, and near-future arrival information. On the other hand, this paper represents the subgroup of the literature by solution methodology: mathematical programming, heuristic and simulation. In most research studies, a single criterion, such as cycle time, tardiness, WIP, or makespan, is used to solve batch process control problems. However, in practice, many other factors should be included even though they are difficult to incorporate in the problem due to differences of metrics, the trade-off in relationships, etc. There is a limited amount of literature available on the use of multiple criteria. Table 1 groups the existing literature for multiple criteria approaches, which can be categorized by cycle time and due date, and earliness and tardiness according to the solutions methodology. Please refer to the review paper for detailed information regarding the scheduling with parallel batch processing (Fowler and M¨onch, 2022).

Table 1. List of literature on approach for multiple criteria

	Considered criteria	
Solution Methodology	**Cycle time and Due date**	**Earliness and Tardiness**
Mathematical Programming	No report	Wu and Wang (1999)
Heuristic (Genetic)	Monch *et al*. (2005), Reichelt and Monch (2006), Mason *et al*. (2007)	Gupta and Sivakumar (2007)

	Considered criteria	
Solution Methodology	**Cycle time and Due date**	**Earliness and Tardiness**
Heuristic (Pareto optimal)	Ganesan *et al*. (2004), Gupta and Sivakumar (2005)	Monch *et al*. (2006), Jeong and Kim (2008)

2.1 Cycle time and due date-related approach

Taking into account the criteria of cycle time and due dates, Monch *et al*. (2005) proposed two decomposition algorithms for tackling the NP-hard problem, aiming to minimize total tardiness on parallel machines and jobs with different ready times. The first method fixes the batches, allocates them to machines using a Genetic Algorithm (GA), and then determines the batches. The second method, in contrast, assigns jobs with machines with the GA, determines the batches for each machine based on the assigned jobs, and sequences these batches.

Reichelt and Monch (2006) considered the minimization of makespan and tardiness across multiple batches, employing the three-stage algorithm, including batching, assignment, and sequencing. During the batch assignment phase, they adapted the process by implementing the Non-dominated Sorting Genetic Algorithm II (NSGA-II) to identify Pareto Optimum, followed by a local search to refine these solutions. Meanwhile, Mason *et al*. (2007) utilized NSGA-II to decrease variations in cycle time and lower the incidence of timer violations to minimize the recirculation. Similarly, Ganesan *et al*. (2004) considered a scheduling framework to minimize the average cycle time and the maximum tardiness. They emphasized the significance of making each decision by considering the near future information and evaluating the results based on set criteria to offer a Pareto-optimal choice. Tackling the complex issue of scheduling jobs, Gupta and Sivakumar (2005) implemented a Pareto-optimum method aiming to reduce the mean cycle time, tardiness, and improve production utilization. This approach began with a discrete event simulation and was complemented by a compromise programming technique at various points throughout the simulated timeline.

2.2 Earliness and tardiness related approach

Due to the complexity of the problem for multiple criteria, there is limited use of mathematical programming approach to solve the problem. However, the problem for earliness and tardiness is comparatively simple to solve with a mathematical programming approach. Wu and Wang (1999) have considered the problem for optimal due dates and optimal sequence to a set of jobs on a single machine. Five lemmas and a polynomial-time algorithm minimize the earliness and tardiness penalties and additional penalties such as due date penalties and completion time penalties. Solving the earliness and tardiness problem is important in Just-In-Time (JIT) systems. In JIT systems, completion of jobs before the due date affects inventory cost, while completion of jobs after the due date affects the contract penalty cost. Monch *et al*. (2006) have proposed several two-phase heuristic approaches based on GA and dominance properties to minimize the sum of the deviation from due date for earliness and tardiness. The first phase uses the condition of no-maximum allowable tardiness constraint, and the second phase changes the schedule to meet the maximum allowable tardiness constraint. Gupta and Sivakumar (2007) have studied how to minimize earliness and tardiness on a batch processor. A look-ahead batching method evaluates different batch scenarios, and compromise programming is used to find the Pareto-optimal boundary. Jeong and Kim (2008) have considered *n* jobs with different release times, due dates, and space limits on parallel machines. Their heuristic approach consists of job selection, job assignment and sequencing, and solution improvement. This three-module heuristic has been compared to GA, hybrid GA, and Tabu Search. Verma *et al*. (2021) and Tian *et al*. (2021) provide a comprehensive overview of the popular multi-objective optimization approach.

3. STOCHASTIC UTILITY EVALUATION FUNCTION

As mentioned in the introduction, the utility value of lots in a system with long production cycles can fluctuate due to various factors. Among time-sensitive factors, due dates and inventory costs are especially crucial. This is because the utility value can vary significantly in response to penalties for not meeting due dates and the costs associated with storing products that are completed ahead of schedule. In addition, particularly with semiconductor manufacturing, there is a risk of contamination and associated costs during the storage of completed products. A basic form of the time-utility function, which is based on the due date, has been identified by Clark (1990). In the foundational work conducted by Park and Banerjee (2011), a novel approach was undertaken through the integration of the stochastic utility evaluation function within the scheduling framework for batch process systems. This marked a significant shift from traditional methods, introducing a more detailed perspective on handling the uncertainties and variabilities inherent in batch processing operations. Our study seeks to build upon this foundation, exploring further into the details and implications of their methodology.

By revisiting the contributions of Park and Banerjee (2011), we aim not only to acknowledge the groundwork but also to expand the scope of their findings. It is important to note that our research has prompted us to investigate and refine several crucial aspects that were not completely explored in the experiments conducted by Park and Banerjee (2011). These refinements have led to additional insights and outcomes in our study. In summary, by adjusting various factors, we have achieved more significant results compared to the original simulation experiments conducted by Park and Banerjee (2011). The three different utility evaluation functions were used by Park and Banerjee (2010) as follows:

1. Hard deadline utility function, which has a utility value of 1 before the due date, otherwise a value of 0.
2. Step time utility function which has different values on both sides of the due date.
3. Linear time utility function with a stable slope over time, and adopted these to MBS-U and DBH-U models.

Accordingly, the following assumptions are made for the purpose of explaining the utility evaluation function:

- The utility evaluation function features a linear model.
- The due date corresponds to the current time, not the expected completion time.

The current time should be used to determine utility value when the current time has not yet exceeded the due date, but the anticipated completion time is beyond it. Nevertheless, their research was still limited as they considered linear utility evaluation functions in their work. It is important to note that depending on the assumptions for the utility function, the result can change. There was no consideration of the due date or inventory cost prior to occurrence. In the case of a bi-criteria problem, an algorithm would concentrate on minimizing cycle time if tardiness is not considered before the due date. However, the importance of tardiness increases as the deadline approaches; thus, there must be a value associated with the utility function before the due date in order to incorporate its significance prior to the due date.

Much of the existing research employs stochastic processes for arrival times, often utilizing an exponential distribution (Cerekci, 2008; Fowler *et al*., 1992; ZEE *et al*., 2001), and to address the bi-criteria issue, the decision maker assigns subjective weights to earliness and tardiness. Nevertheless, within a stochastic framework, it is possible to obtain a probability function that expresses the relationship values between two variables, one being random and the other independent or dependent variable. Consequently, when the need arises to account for weights for earliness and tardiness that vary with time, the values derived from the probability function can be utilized as such weights. These values, grounded in probability, can be seen as more objective compared to those determined by a decision-maker, which tends to be subjective. In the traditional approach to the bi-criteria problem, the decision maker still sets the tardiness weight, which remains unchanged over time. Utilizing the weight derived from a probability function, we adopt the SUE function, which adjusts the weight for earliness and tardiness in accordance with temporal changes.

In this study, we used the following notations for the SUE function:

N_j:	Number of orders for product j
D_j:	Due date for product j
$C_{t,j}$:	Completed jobs for product j at time t
a_j:	Number of remaining orders for product j at time t
$W_j(t)$:	Weight function for product j at time t
$F_j(t, a_j)$:	Probability function for product j at time t given number of remaining orders for product j

We define the SUE function for batch process control problems as follows:

$$U_j(t) = W_j(t) = f\left(F_j(t, a_j), D_j, N_j, C_{t,j}\right)$$

The SUE function aims to determine the optimal weight for earliness, tardiness, and both earliness and tardiness, simultaneously. By using the SUE function $U_j(t)$, the probability that the job will be completed by the due date is combined with other deterministic parameters such as the due date, the number of orders, completed jobs, and remaining orders. Suppose a specific job arrives at an exponential distribution, with a due date T, and n number of orders. There are two possible scenarios that can occur. In the first case, the probability of completion of the remaining jobs within the due date is estimated at a decision point. When the probability is low, it indicates that arrivals are not sufficient to finish all jobs within the due date. Therefore, it is necessary to give high weight to tardiness in order to finish the jobs on time. In the second case, the ratio of the number of orders that remain to be completed and the number of possible jobs to complete them is calculated. In cases where the ratio features high, fewer lots than orders are expected to arrive. As a result, the SUE function is derived in the first case.

3.1 SUE function for tardiness

It has been found that various approaches and algorithms have been proposed for solving the batch process control problem, taking into account cycle time and tardiness, as well as both cycle time and tardiness. Because of the trade-off between cycle time and tardiness, the bi-criteria objective problem is generally more challenging than a single-objective problem.

Therefore, integrating the weights for each criterion into a unified metric has been employed to address the bi-criteria issue. Nevertheless, this approach has limitations in that it does not account for varying weights for each criterion. Using the SUE function, this research modifies the method for bi-criteria problems to account for the varying weight. Several approaches to bi-criteria objective optimization have been developed, including genetic algorithms and Pareto-optimal algorithms. The latter approach combines multiple criteria into a single objective. That is, all weights are assigned to all of the criteria determined by the decision-maker. There can be a lack of objectivity in the weights chosen by a decision-maker. In contrast, the SUE function can be used to achieve the desired objective weight based on stochastic information, such as expected value.

Let's us assume that lots arrive at a batch processor with an exponential distribution with rate λ, for N orders and a due date D, the SUE function is obtained by evaluating the probability of job completion within the remaining time before the due date. When considering the number of orders N and the due date D at time t for the arrival of a single product, the function $F(\cdot)$ with the random variable X can be formulated as follows:

$$F(t, a) = 1 - P\{X \leq a \mid t\}$$

Using the function, the probability could be estimated where the random variable X takes on a value less than or equal to the number of orders remaining by the due date. When a lower probability is estimated, the probability of not completing jobs by the due date is greater, indicating that tardiness increases. Thus, when the SUE function is incorporated into the cycle time-based approach, the property determines weights of tardiness at time t. Considering the probability of not completing jobs by the due date is very low, the bi-criteria problem should minimize cycle time. That is, the weight of tardiness could be set to 1, making the algorithm equivalent to a cycle time-based approach. When the time nears to the due date, the probability of not completing jobs increases. Thus, the tardiness weight increases as the probability of tardiness increases. Let us assume that lots are being received at a batch processor with an exponential distribution with rate λ. Figure 1 illustrates an example where, as time approaches the due date (100-time units), the probability of completing jobs by the due date decreases.

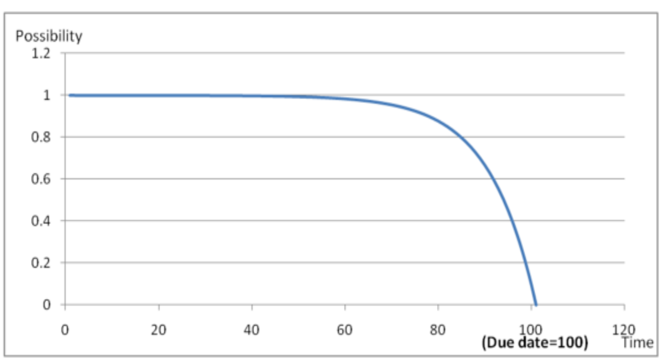


Figure 1. An example of the time-probability function when the time nears the due date (100-time units)

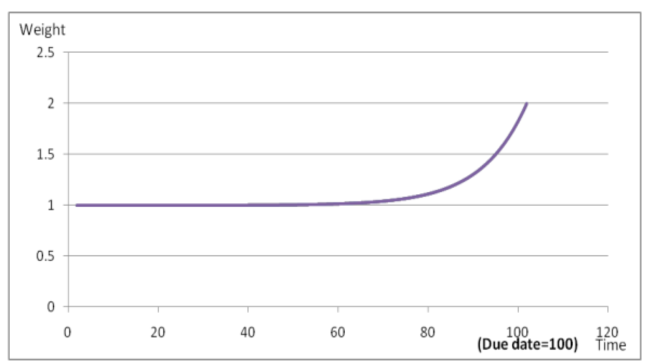


Figure 2. An example of time-weight function for tardiness

In contrast, as time approaches the due date, tardiness becomes more significant since it is necessary to consider more than just cycle time to complete jobs by the due time. Figure 2 illustrates a specific instance of the weight varying over time and assumes the following formula:

$$W(t) = 1, \ \ where\ F(t, a) = 1$$
$$W(t) = 2 - F(t, a), \qquad where\ F(t, a) < 1$$

For solving the bi-criteria objective (cycle time and tardiness), the tardiness value can be applied to existing methods for minimizing cycle time.

3.2 SUE function for earliness and tardiness

If we follow the procedure for tardiness described above, we can also derive the SUE function for earliness. In this case, the earliness arises prior to the due date. If the earliness is factored into the probability function, the JIT scheduling becomes critical. To obtain the JIT schedule, the minimum period during which the number of arrivals is equal to the number of orders needs to be considered. When the due date has been reached, the earliness is not required to be considered since products are moved from one system to another once the job has been completed. Hence, in this case, earliness is assigned a weight of 1, which is its base value. Practically, it is possible to determine the tardiness costs, such as contract penalties, and the earliness costs, such as inventory costs. Hence, the ratios of earliness and tardiness weights can be utilized to derive the probability and SUE function. Let us assume that lots arrive at a batch processor according to an exponential distribution with a rate of λ. As the due date approaches, the probability of completing jobs on time decreases (see Figure 1). In contrast, as the time approaches the due date, the weight for earliness decreases as the inventory cost decreases. Refer to Figure 3 as an example of the weight of earliness varying over time. We have assumed the following formula:

$$r = \frac{cost\ for\ earliness}{cost\ for\ tardiness}$$
$$W(t) = 1,\ \ where\ F(t,a) = 0$$
$$W(t) = 1 + r \cdot F(D - t, a), \qquad where\ 0 < F(t,a) < 1$$
$$W(t) = 1 + r,\ \ where\ F(t,a) = 1$$

Refer to Figure 4 for an illustration of the SUE function used to solve a tri-criteria objective problem:

$$W_{ET}(t) = W_E(t) + W_T(t) - 1$$

where

$$W_E(t) = Weight\ for\ earliness$$
$$W_T(t) = Weight\ for\ tardiness$$

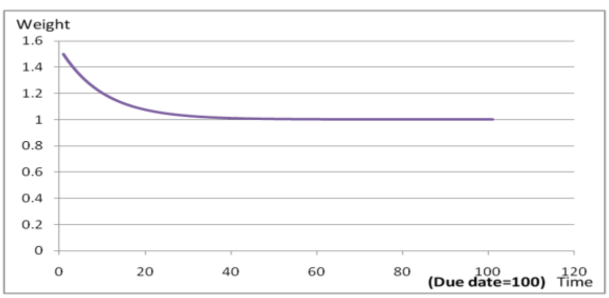


Figure 3. An example of time-weight function for earliness

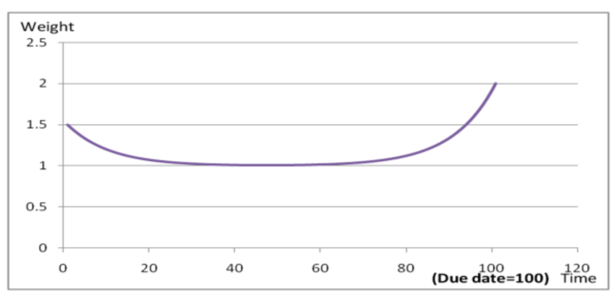


Figure 4. An example of time-weight function for earliness and tardiness

It is important to note that there is little research addressing the tri-criteria objective problem of cycle time, earliness, and tardiness. An SUE function (for both earliness and tardiness) may, however, be incorporated into existing approaches aimed at minimizing cycle time alone.

4. APPLICATION OF SUE FUNCTION TO THE BATCH PROCESS CONTROL PROBLEM

Practically, it is common for a decision-maker to consider multiple criteria. However, because of the high complexity, a limited amount of research has been conducted on tricriteria, including cycle time, earliness, and tardiness.

4.1 MBS-SUE approach

When the SUE function is used to measure earliness and tardiness, the weighted value remains one when the job is expected on time, meaning that earliness and tardiness will not occur since commodity weights will not change. In this scenario, the MBS-SUE model behaves in the exact same way as the MBS model does. In the case of jobs that are expected to arrive late or, the results of the MBS-SUE method will depend on stochastic information related to arrivals, such as distribution and rate of arrivals and traffic density. In the case of multiple products, the MBS-SUE is similar to the MBSX approach for triple criteria. When the queue contains multiple product types, each with an MBS value exceeding a certain threshold, the product with the highest weighted waiting time is selected according to the SUE function, which takes into account both earliness and tardiness. When multiple nominees are present, the candidate with the shortest weighted processing time, which takes into account earliness and tardiness, is selected.

4.2 NACH-SUE approach

In this approach, the gain or loss within the NACH is calculated by utilizing the weights derived from the SUE function for earliness and tardiness. Accordingly, the optimal epoch is chosen for achieving the positive value. As part of the NACH-SUE, the NACH utilizes the SUE function for earliness and tardiness. Consequently, the additional loss for NACH-SUE from the future arrival at t_1 can be determined by the following equation:

$$WArea_1(t) = W_{ET}(t) \times q(t_1 - t_0)$$

It represents the total additional loss for the q lots in the queue at t_0, which incorporates the weighted value calculated from the SUE function for earliness and tardiness. Based on the following formula, we can calculate the gain resulting from waiting for the future arrival at t_1:

$$WArea_2(t) = W_{ET}(t) \times \left(t_0 + T_j - t_1\right)$$

Therefore, the net gain is given by

$$WNet(t) = WArea_2(t) - WArea_1(t)$$

Similar to the single product case, the NACH-SUE for multiple product types addressing earliness and tardiness is modeled after the NACH-SUE procedure for tardiness, as shown in Figure 5 and Algorithm 1.

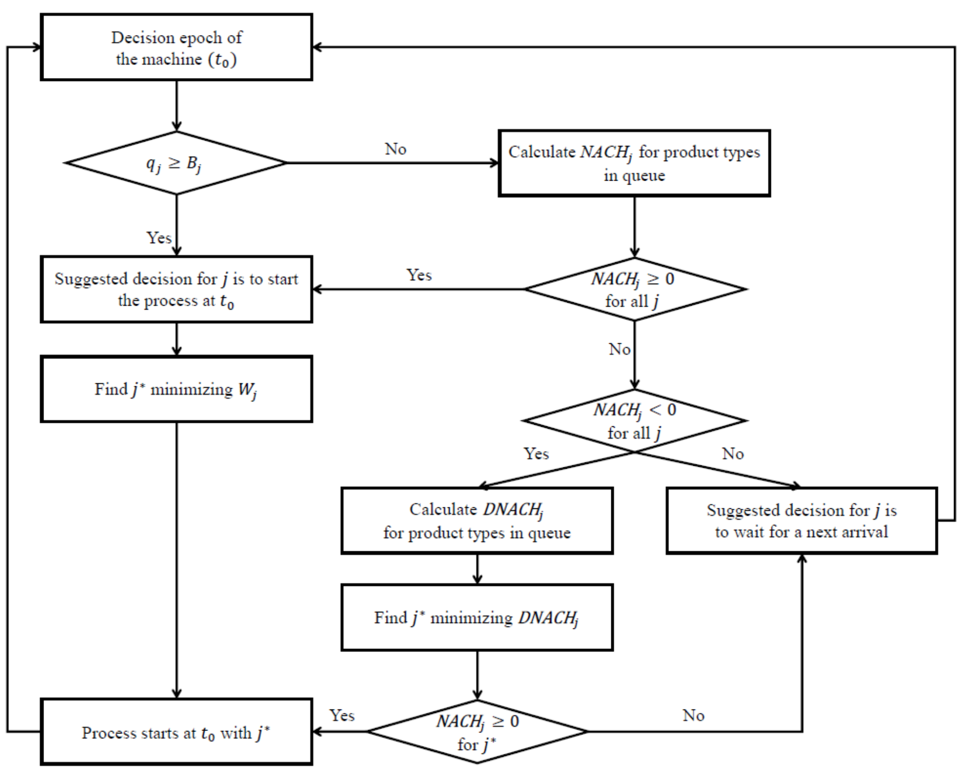


Figure 5. Flow chart of the NACH-SUE algorithm

Algorithm 1 Pseudocode of the simulation

if *the machine is idle* **then**
    if *full batches are available* **then**
        choose a product a product $j^* = arg\,min_{q_i \geq C_j} W_j$ $(W_j = T_j \sum_{i \neq j} W_{ET_i}(t) \times q_i)$
        and batch is processed.
        Here, W_j is the total delay for the other products when the product j is processed.
        j^* means that among product types, the product j^* has minimum waiting time metric value.
    end
    else
        evaluate $NACH_j$ for all j $(NACH_j = \sum_{i=1, i \neq j}^{N} [W_{ET_i}(t) \times (q_i(t_{1,i} - t_0) - (t_0 + T_j - t_{1,i}))])$
        if $j \in SY$, **then**
            wait *($NACH_j < 0$, for all j = 1, …, N)*.
        end
        else
            if $j \in SN$ *($NACH_j \geq 0$, for all j = 1, …, N)*,
            Here, SY is the set of j, which needs to wait for a next arrival.
            On the other hand, SN is the set of j, which does not need to wait for a next arrival.
            then
                choose $j^* = arg\,min_{j \in J} W_j$
            end
            else
                $D_j = W_j + \sum_{i=1}^{N} max\,(0, W_{ET_i}(t) \times (t_0 + T_j - t_{1,i}))$ $j \in SN$
                $D_j = \sum_{i=1}^{N} (W_{ET_i}(t) \times q_i(t_{1,i} - t_0)) + W_j + \sum_{i=1}^{N} max\,(0, W_{ET_i}(t) \times (t_0 + T_j - t_{1,i}))$ $j \in SY$
                Where D_j is the total delay when product j loaded.
                Choose a product $j^* = arg\,min_{j \in J}\, D_j$
                if $j^* \in SN$, **then**
                    batch is processed.
                end
                else
                    wait.
                end
            end
        end
    end
end
if *the machine is idle and a product j arrives,* **then**
    proceed as indicated by $NACH_j$.
end

4.3 Full Batch Policy

At the decision point t_0, if the queue contains only one full batch, that batch is then processed as a full batch. If multiple product types with full batches are waiting in the queue, the product with the longest weighted waiting time will be chosen. Otherwise, the one with the shortest weighted processing time will be chosen based on the SUE function for earliness and tardiness.

4.4 No-idling Policy

The purpose of this policy is to keep a processor operating as long as there are available products in the queue. There is no difference between the no-idling policy and the full batch policy in conditions where there are several full batches available at the decision epoch t_0. When only partial batches are present, the one with the longest weighted waiting time is chosen; if multiple candidates exist, the shortest weighted processing time is selected.

5. SIMULATION RESULT

We conducted simulation studies by modifying the benchmark strategy under the following conditions. To satisfy a reasonable utilization level and steady-state queue length, the value of arrival distribution, machine capacity, processing time, due date, and number of orders have been chosen. The test for each control was conducted for every scenario. The settings for the simulation are presented in Table 2.

i) Control strategy: NACH-SUE, MBS-SUE, Full batch, No idle
ii) Simulation run length: 100,000 times
iii) Number of replications: 10 times
iv) Warm-up period: 5,000 times

Table 2. Configuration of the simulation experiments

No.	Factors	Settings		
1	Number of Products (NP)	2		
	Number of Products	5		
2	Product Mix (PM)	Equal (E)	2 products	(0.5,0.5)
	Product Mix	Equal	5 products	(0.2,0.2,0.2,0.2,0.2)
	Product Mix	Different (D)	2 products	(0.2,0.8)
	Product Mix	Different	5 products	(0.1,0.1,0.1,0.35,0.35)
3	Machine Capacity by Product (MC)	Equal	2 products	(5,5)
	Machine Capacity by Product	Equal	5 products	(5,5,5,5,5)
	Machine Capacity by Product	Different	2 products	(3,7)
	Machine Capacity by Product	Different	5 products	(3,4,5,6,7)
4	Processing Time by Product (PT)	Equal	2 products	(25,25)
	Processing Time by Product	Equal	5 products	(25,25,25,25,25)
	Processing Time by Product	Different	2 products	(10,40)
	Processing Time by Product	Different	5 products	(10,20,25,30,40)
5	Traffic Intensity (TI)	0.2		
	Traffic Intensity	0.5		
	Traffic Intensity	0.8		

From Chaudhry and Templeton (1983), the batch traffic intensity (ρ) is defined as the average arrival rate of each product divided by the maximum batch processing rate when the machine operates at full capacity. Traffic intensity can be calculated using the following equation:

$$\lambda_{\mathrm{j}} = \rho / \sum_{j=1}^{N} \frac{P_j T_j}{B_j}$$

where

λ_j: Mean arrival rate for product j
ρ: Batch processor traffic intensity
P_j: Product mix for product j
T_j: Batch process time for product j
B_j: Batch processor capacity for product j

To satisfy a reasonable utilization level and steady-state queue length, the values for the number of order and due dates have been selected for each scenario as follows:

$$N_j = S/\rho \times \sum_{j=1}^{N} \frac{P_j B_j}{T_j}$$

where

S = Standard number of orders when all factors (NP, PM, MC, PT, TI) have equal values

$$D_j = \lambda_j \times N_j$$

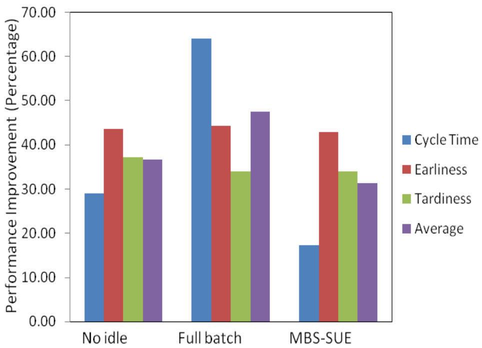
In order to create all settings of the simulation scenario, the experiment of control strategies occurring on the set-ups' compilation provides 48 (2 · 24) scenarios. Along with a run-time of 100,000-time units, a warm-up time of 5,000-time units and 10 replications of every scenario are set up.

This section describes the performance of NACH-SUE in comparison with the three benchmark strategies. Tables 3, 4 and 5 summarize the simulation outcomes. The results in the tables are averaged over different settings of the product, machine and process characteristics. Experimental results in Tables 3 and 4 show that the best performance rule is NACH-SUE. The MBS-SUE performs most similarly to NACH-SUE because it is the only benchmark that appraises decision options in collaboration with the effects on all product types. Conversely, the no idle rule and full batch rule have an inferior performance because the former does not allow waiting for the next arrival, while the latter necessitates waiting for the next arrival to form a full batch. Table 5 presents the trend of performance improvements when the SUE function is used in the strategies. Dynamic control strategy, of which utility value varies over time features significant improvement than static control strategy.

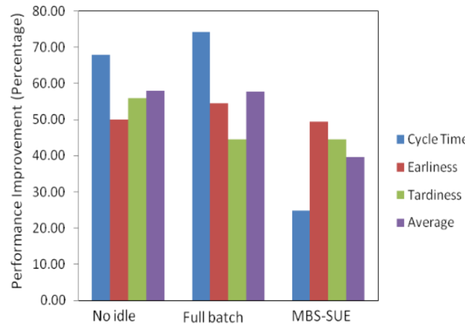
Figure 6 indicates performance improvement gained by NACH-SUE with an increasing number of products. Although improvement percentages for cycle time, earliness and tardiness do not have a robust trend, the actual improvement values are significantly large. There is a significant performance improvement with a larger number of products (number of products of 5 compared to number of products of 2). This is due to the fact that NACH-SUE can consider more alternatives (process or wait) at each decision epoch, and there are more decision epochs for the number of products of 5. It is evident from Figure 7 that performance improves steadily with increasing traffic intensity. It can be seen that the performance improvement between no idle and MBS-SUE is similar when the traffic intensity is low because the optimal MBS at low traffic intensity features almost equal to the MBS of 1, which is the same as the no idle strategy. In contrast, performance improvement over the full batch is comparable to MBS-SUE at high traffic intensity because the highest performing MBS at low traffic intensity is an extremely low value comparable to full batch MBS.

There are no significant trends and changes in performance improvement at different machine capacities (see Figure 8). The results show that performance is better when the machine capacity is equal, this is due to the fact that equal capacity leads to a balanced batching and processing. Improvements over MBS-SUE are less than others. This is due to the fact that the MBS strategy chooses the best MBS among available MBSs. Available MBSs include MBS of 1, which is equal to no idle strategy, and MBS of machine capacity, which is equal to full batch strategy. The performance improvement obtained by NACH-SUE does not present any significant trend and change with equal and different product mixes (see Figure 9). This is due to the SUE function effect. Even though a steady trend is expected when the product mix is unbalanced as the product mix is dominated by a few product types, the SUE function affects to stabilizes the performance improvement for all criteria (cycle time, earliness and tardiness). The dynamic weight is given to each product at the decision epoch to optimize based on multi-objective criteria. In this step, all attempts by different weights overtime on each product affect both equal product mix and different mix. This effect soothes the performance improvement gap between equal and different product mixes.

Similar analysis with product mix can be performed to equal and different processing times. The performance improvement obtained by NACH-SUE does not show any significant trend and change with equal and different processing times (see Figure 10). Even though processing time is unbalanced between product types, the SUE function plays a role in balancing the two attributes. The performance obtained by dynamic strategy using the SUE function is compared to the performance gained by static strategy using fixed utility value over time (see Table 5). Figure 11 illustrates that cycle time decreases or almost remains the same for all strategies. The interpretation of this result is due to the fact that the existing strategies (NACH, MBS, etc.) use fixed utility values over time to minimize the cycle time only. On the other hand, even though cycle time increases when the SUE function is used in the existing strategies, the strategies using the SUE function attempt to optimize a multi-objective problem with attributes of cycle time, earliness and tardiness. In an overview of earliness and tardiness, the results obtained from dynamic strategies are better than the results from static strategies. As a result, there is an improvement in total averaged performance with cycle time, earliness and tardiness using a dynamic strategy as compared to a static strategy.

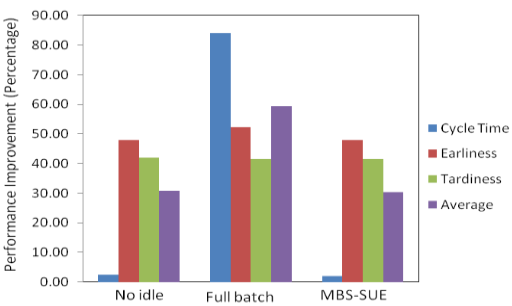


(a) Number of products: 2

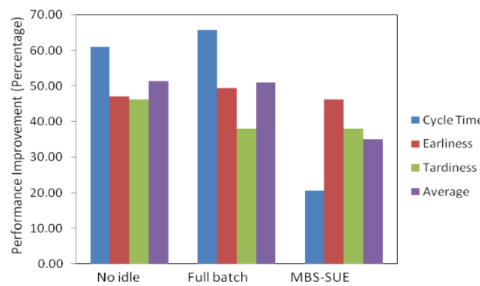


(b) Number of products: 5

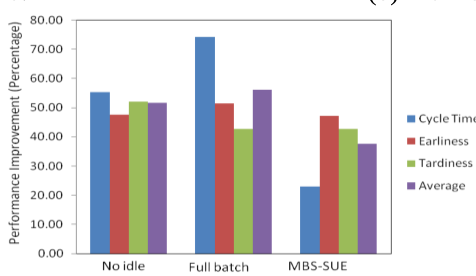
Figure 6. Performance improvements for NACH-SUE with the different number of products



(a) Traffic intensity: 0.2

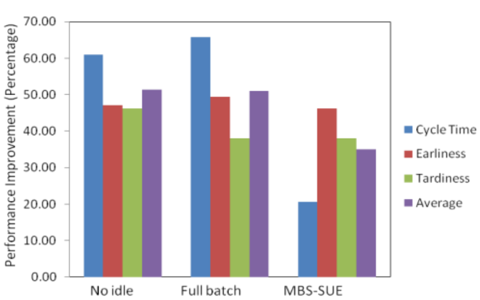


(b) Traffic intensity: 0.5

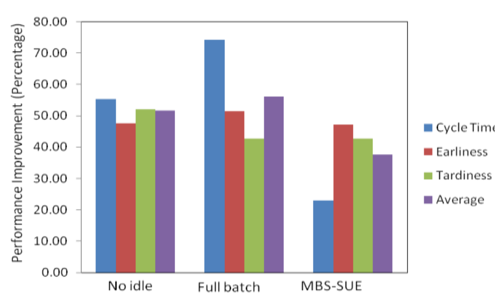


(c) Traffic intensity: 0.8

Figure 7. Performance improvements for NACH-SUE with different traffic intensities

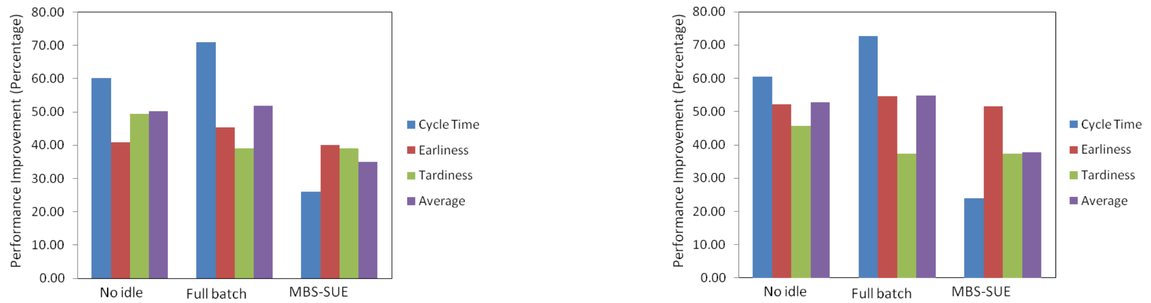


(a) Machine capacity: Equal



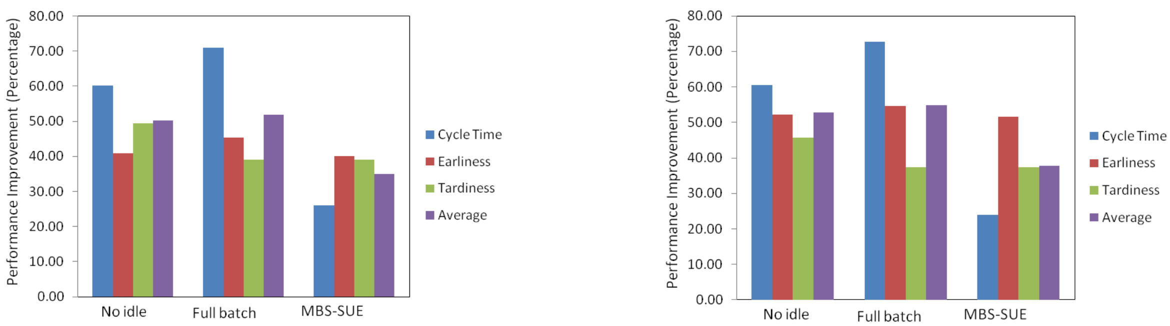
(b) Machine capacity: Different

Figure 8. Performance improvements for NACH-SUE with different machine capacities



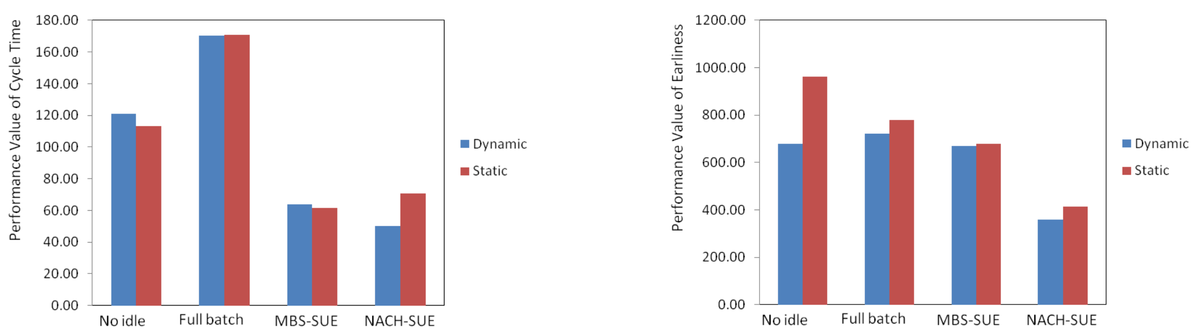
(a) Product mix: Equal (b) Product mix: Different

Figure 9. Performance improvements for NACH-SUE with different product mixes

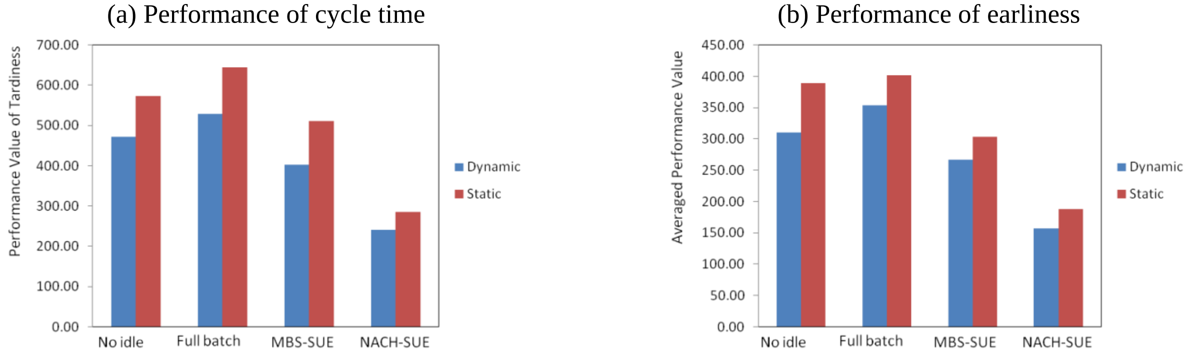


(a) Processing time: Equal (b) Processing time: Different

Figure 10. Performance improvements for NACH-SUE with different processing times



(a) Performance of cycle time (b) Performance of earliness



(c) Performance of tardiness (d) Averaged performance

Figure 11. Comparison of performances obtained by dynamic strategy and static strategy with cycle time, earliness, tardiness and averaged all criteria

Table 3. Summary of the simulation results

No.	Average at:	No idle			Full batch			MBS-SUE			NACH-SUE		
		CT	**earliness**	**tardiness**	**CT**	**earliness**	**tardiness**	**CT**	**earliness**	**tardiness**	**CT**	**earliness**	**tardiness**
1	Number of products: 2	59.07	561.20	358.61	116.54	567.92	433.68	50.67	553.31	341.28	41.89	316.29	225.13
2	Number of products: 5	180.37	808.76	589.80	224.06	888.52	630.24	77.09	798.50	468.82	57.95	404.78	260.06
3	Traffic intensity: 0.2	44.95	864.15	413.32	274.24	942.13	555.00	44.73	862.53	408.98	43.83	449.68	239.44
4	Traffic intensity: 0.5	90.24	645.97	502.94	132.09	678.93	576.84	67.35	640.96	429.71	57.20	328.57	240.82
5	Traffic intensity: 0.8	235.50	467.88	487.48	105.09	483.47	429.89	81.43	446.51	349.22	48.85	290.26	221.41
6	Product mix: equal	124.95	669.01	517.85	142.59	699.79	607.80	61.35	658.63	449.79	48.73	354.22	278.96
7	Product mix: different	114.49	700.96	430.56	198.01	756.65	456.11	66.41	693.18	360.31	51.11	366.85	206.23
8	Machine capacity: equal	111.66	531.89	504.63	155.56	575.96	527.33	61.25	526.41	424.93	49.31	321.19	244.24
9	Machine capacity: different	127.77	838.08	443.79	185.04	880.48	536.58	66.51	825.40	385.17	50.53	399.89	240.94
10	Processing Time: equal	123.64	475.15	491.60	169.82	513.91	507.67	66.54	468.09	408.14	49.20	280.67	248.47
11	Processing Time: different	115.79	894.81	456.82	170.77	942.53	556.24	61.23	883.72	401.96	50.64	440.40	236.71
12	Overall average	120.77	677.99	472.49	170.35	720.94	528.85	64.05	668.84	402.57	49.93	359.34	240.22

Table 4. Summary of the simulation results: Comparison of the NACH-SUE versus benchmark control strategies

No.	Average at:	Cycle Time			Earliness			Tardiness			Average		
		Δ1	**Δ 2**	**Δ 3**	**Δ 1**	**Δ 2**	**Δ 3**	**Δ 1**	**Δ 2**	**Δ 3**	**Δ 1**	**Δ 2**	**Δ 3**
1	Number of products: 2	29.10	64.06	17.34	43.64	44.31	42.84	37.22	34.03	34.03	29.38	40.08	24.26
2	Number of products: 5	67.87	74.14	24.83	49.95	54.44	49.31	55.91	44.53	44.53	49.58	48.63	31.34
3	Traffic intensity: 0.2	2.48	84.02	2.01	47.96	52.27	47.87	42.07	41.45	41.45	22.84	50.54	22.46
4	Traffic intensity: 0.5	36.62	56.70	15.07	49.14	51.60	48.74	52.12	43.96	43.96	37.77	42.15	27.80
5	Traffic intensity: 0.8	79.25	53.51	40.00	37.96	39.96	34.99	54.58	36.60	36.60	50.94	36.70	31.37
6	Product mix: equal	61.00	65.83	20.58	47.05	49.38	46.22	46.13	37.98	37.98	43.55	42.83	27.22
7	Product mix: different	55.36	74.19	23.04	47.66	51.52	47.08	52.10	42.76	42.76	43.76	47.57	29.78
8	Machine capacity: equal	55.84	68.30	19.50	39.61	44.23	38.99	51.60	42.52	42.52	42.42	44.31	27.17
9	Machine capacity: different	60.46	72.69	24.04	52.29	54.58	51.55	45.71	37.44	37.44	44.10	45.81	29.09
10	Processing Time: equal	60.21	71.03	26.06	40.93	45.39	40.04	49.46	39.12	39.12	43.38	44.28	28.40
11	Processing Time: different	56.27	70.35	17.30	50.78	53.27	50.17	48.18	41.11	41.11	43.28	46.03	27.83
12	Overall average	58.66	70.69	22.05	47.00	50.16	46.27	49.16	40.33	40.33	43.77	45.37	28.50

Δ1 = 100* (No idle – NACH-SUE)/No idle
Δ2 = 100* (Full batch – NACH-SUE)/Full batch
Δ3 = 100* (MBS-SUE – NACH-SUE)/MBS-SUE

Table 5. Summary of the simulation results: Comparison of the dynamic weight strategy versus the static weight strategy

No.	Average at:	No idle			Full batch			MBS-SUE			NACH-SUE		
		Static	**Dynamic**	**Δ**	**Static**	**Dynamic**	**Δ**	**Static**	**Dynamic**	**Δ**	**Static**	**Dynamic**	**Δ**
1	Number of products: 2	338.03	232.76	31.14	319.71	278.06	13.03	242.76	222.87	8.19	171.38	141.72	17.31
2	Number of products: 5	443.31	391.52	11.68	488.48	432.85	11.39	369.26	315.05	14.68	205.76	173.47	15.69
3	Traffic intensity: 0.2	484.49	296.78	38.74	487.12	433.43	11.02	346.17	294.99	14.78	224.67	169.37	24.61
4	Traffic intensity: 0.5	320.20	305.39	4.63	394.32	349.46	11.37	291.37	272.51	6.47	175.07	154.10	11.98
5	Traffic intensity: 0.8	349.50	318.97	8.73	304.83	258.91	15.06	257.23	217.97	15.26	153.74	138.46	9.93
6	Product mix: equal	467.94	325.77	30.38	415.87	366.76	11.81	333.64	280.15	16.03	203.34	168.27	17.25
7	Product mix: different	313.40	298.51	4.75	392.32	344.15	12.28	278.38	257.77	7.40	173.80	146.92	15.47
8	Machine capacity: equal	419.91	294.08	29.97	367.48	323.63	11.93	293.13	249.80	14.78	183.90	151.38	17.68
9	Machine capacity: different	361.44	330.20	8.64	440.71	387.29	12.12	318.88	288.13	9.64	193.24	163.80	15.23
10	Processing Time: equal	328.70	284.27	13.52	365.24	311.48	14.72	285.70	236.24	17.31	177.57	146.00	17.78
11	Processing Time: different	452.64	340.01	24.88	442.94	399.43	9.82	326.31	301.68	7.55	199.57	169.18	15.23
12	Overall average	389.05	310.75	20.13	401.73	353.22	12.07	303.89	267.02	12.14	187.46	156.61	16.46

Δ = 100*(Static-Dynamic)/Static

6. CONCLUDING REMARKS

The main contribution of this research is the introduction of an SUE function approach to provide a decision-maker with more information about the conditions of the existing batch process control program. Using this SUE function, the modified benchmark strategies, NACH-SUE and MBSSUE, are developed to optimize a tri-criteria (cycle time, earliness and tardiness) problem. Detailed experimental results show that the overall performance of the strategies with the SUE function is improved compared to the strategies which use a static utility value. Even more crucially, we discovered that the NACH-SUE strategy exhibits the most significant performance improvement.

In Section 5, the results have been analyzed with the trend to present the tendency relation for each strategy and the change to show the performance difference. The performance enhancement observed with the NACH-SUE strategy does not significantly vary in relation to the number of products, machine capacity, and processing time. This phenomenon occurs from the SUE function effect to balance the trend and change. The observed performance enhancements for NACH-SUE range between 28.5% and 45.4%, while the performance differences between the dynamic and static strategies vary from 12.1% to 20.1%. Listed below are the supplementary contributions and potential applications to the industry. First, this research recognizes that the weights for earliness and tardiness change over time and develops a method to incorporate this in existing methods with the help of the SUE function. Second, when a change in earliness and tardiness factors occurs, only the SUE function is needed to account for the change. This makes the SUE function easy to use and does not affect the entire model. Finally, the SUE function can be applied easily to existing methods as demonstrated in this paper with MBS-SUE and NACH-SUE. The SUE function provides the mechanism to solve multi-criteria decision-making problems. A tri-criteria problem is used in this paper as an example to demonstrate its applicability.

There are several lines of research that could be carried out in the future. It is possible to adopt the SUE function to other existing dynamic control systems for serial or batching processing. Furthermore, there is still much to be explored regarding the estimation of the SUE function.

ACKNOWLEDGMENTS

The authors would like to thank the associate editor and three anonymous reviewers for the valuable comments. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(No.RS-2023-00218913).

REFERENCES

Adamu, M. O. and Adewumi, A. O. (2016). Minimizing The Weighted Number of Tardy Jobs on Multiple Machines: A Review. *Journal of Industrial & Management Optimization*, 12(4):1465.

Cerekci, A. (2008). *Dynamic Control of Serial-Batch Processing Systems*. Texas A&M University.

Chaudhry, M. and Templeton, J. G. (1983). *First Course in Bulk Queues*.

Clark, R. K. (1990). *Scheduling Dependent Real-Time Activities*. Carnegie Mellon University.

Elmi, A. and Topaloglu, S. (2017). Cyclic Job Shop Robotic Cell Scheduling Problem: Ant Colony Optimization. *Computers & Industrial Engineering*, 111:417–432.

Espadinha-Cruz, P., Godina, R., and Rodrigues, E. M. (2021). A Review of Data Mining Applications in Semiconductor Manufacturing. *Processes*, 9(2):305.

Fang, J., Cheang, B., and Lim, A. (2023). Problems and Solution Methods of Machine Scheduling in Semiconductor Manufacturing Operations: A Survey. *Sustainability*, 15(17):13012.

Fowler, J. W., Hogg, G. L., and Phillips, D. T. (2000). Control of Multiproduct Bulk Server Diffusion/Oxidation Processes. Part 2: Multiple Servers. *IIE Transactions*, 32(2):167–176.

Fowler, J. W. and M¨Onch, L. (2022). A Survey of Scheduling with Parallel Batch (P-Batch) Processing. *European Journal of Operational Research*, 298(1):1–24.

Fowler, J. W., Phillips, D. T., and Hogg, G. L. (1992). Real-Time Control of Multiproduct Bulk-Service Semiconductor Manufacturing Processes. *IEEE Transactions on Semiconductor Manufacturing*, 5(2):158–163.

Ganesan, V. K., Gupta, A. K., and Iyer, S. (2004). Bi-Objective Schedule Control of Batch Processes in Semiconductor Manufacturing. in IEEE Conference on Robotics, Automation and Mechatronics, 2004. 2: 1077– 1082. IEEE.

Gupta, A. K. and Sivakumar, A. I. (2005). Single Machine Scheduling with Multiple Objectives in Semiconductor Manufacturing. *The International Journal of Advanced Manufacturing Technology*, 26(9):950–958.

Gupta, A. K. and Sivakumar, A. I. (2007). Controlling Delivery Performance in Semiconductor Manufacturing Using Look Ahead Batching. *International Journal of Production Research*, 45(3):591–613.

Hung, Y.-F., Bao, J.-S., and Cheng, Y.-E. (2017). Minimizing Earliness and Tardiness Costs in Scheduling Jobs with Time Windows. *Computers & Industrial Engineering*, 113:871–890.

Jeong, S. J. and Kim, K. S. (2008). Parallel Machine Scheduling with Earliness-Tardiness Penalties and Space Limits. The International *Journal of Advanced Manufacturing Technology*, 37(7):793–802.

Kaplano˘glu, V. (2016). An Object-Oriented Approach for Multi-Objective Flexible Job-Shop Scheduling Problem. *Expert Systems with Applications*, 45:71–84.

Lee, Y. H. and Lee, S. (2022). Deep Reinforcement Learning Based Scheduling Within Production Plan in Semiconductor Fabrication. *Expert Systems with Applications*, 191:116222.

Lima, A., Borodin, V., Dauz`Ere-P´Er`Es, S., and Vialletelle, P. (2021). A Sampling-Based Approach for Managing Lot Release in Time Constraint Tunnels in Semiconductor Manufacturing. *International Journal of Production Research*, 59(3):860–884.

Liu, Q., Ye, Q., Mei, X., and Zhang, Q. (2023). Inter-Cell Scheduling Based on A Transportation Vehicle Sharing Strategy. *European Journal of Industrial Engineering*, 17(6):939–973.

Mason, S. J., Kurz, M. E., Pfund, M. E., Fowler, J. W., and Pohl, L. M. (2007). Multi-Objective Semiconductor Manufacturing Scheduling: A Random Keys Implementation of Nsga-Ii. in 2007 *IEEE Symposium on Computational Intelligence in Scheduling*, Pages 159–164. IEEE.

Mathirajan, M. and Sivakumar, A. I. (2006). A Literature Review, Classification and Simple Meta-Analysis on Scheduling of Batch Processors in Semiconductor. *The International Journal of Advanced Manufacturing Technology*, 29(9):990–1001.

M¨Onch, L., Balasubramanian, H., Fowler, J. W., and Pfund, M. E. (2005). Heuristic Scheduling of Jobs on Parallel Batch Machines with Incompatible Job Families and Unequal Ready Times. *Computers & Operations Research*, 32(11):2731–2750.

M¨Onch, L., Unbehaun, R., and Choung, Y. I. (2006). Minimizing Earliness–Tardiness on A Single Burn-In Oven with A Common Due Date and Maximum Allowable Tardiness Constraint. *Or Spectrum*, 28(2):177–198.

Ogun, B. and Alabas-Uslu, C¸ . (2018). Mathematical Models for A Batch Scheduling Problem to Minimize Earliness and Tardiness. *Journal of Industrial Engineering and Management (JIEM)*, 11(3):390–405.

Pan, Q.-K., Ruiz, R., and Alfaro-Fern´Andez, P. (2017). Iterated Search Methods for Earliness and Tardiness Minimization in Hybrid Flowshops with Due Windows. *Computers & Operations Research*, 80:50–60.

Panwalkar, S., Smith, M., and Seidmann, A. (1982). Common Due Date Assignment to Minimize Total Penalty for The One Machine Scheduling Problem. *Operations Research*, 30(2):391–399.

Park, H. and Banerjee, A. (2010). Utility-Based Dynamic Control of Batch Processing Systems. in *2010 IEEE International Conference on Automation Science and Engineering*, Pages 439–444. IEEE.

Park, H. and Banerjee, A. (2011). A New Dynamic Scheduling for Batch Processing Systems Using Stochastic Utility Evaluation Function. in *Proceedings of The 2011 Winter Simulation Conference (WSC)*, Pages 2302–2314. IEEE.

Reichelt, D. and M¨Onch, L. (2006). Multiobjective Scheduling of Jobs with Incompatible Families on Parallel Batch Machines. in *European Conference on Evolutionary Computation in Combinatorial Optimization*, Pages 209–221. Springer.

Tian, Y., Si, L., Zhang, X., Cheng, R., He, C., Tan, K. C., and Jin, Y. (2021). Evolutionary Large-Scale Multi-Objective Optimization: A Survey. *ACM Computing Surveys (CSUR)*, 54(8):1–34.

Uruk, Z. and Yal¸Cıner, A. Y. (2024). Two-Machine Flowshop Scheduling with Fuzzy Processing Times and Flexible Operations. *European Journal of Industrial Engineering*, 18(1):100–119.

Van De Rzee, D., Van Harten, A., and Schuur, P. C. (1997). Dynamic Job Assignment Heuristics for Multi-Server Batch Operations-A Cost Based Approach. *International Journal of Production Research*, 35(11):3063–3094.

Verma, S., Pant, M., and Snasel, V. (2021). A Comprehensive Review on NSGA-II for Multi-Objective Combinatorial Optimization Problems. *IEEE Access*, 9:57757–57791.

Wu, Y. and Wang, D. (1999). Optimal Single-Machine Scheduling About A Common Due Window with Earliness/Tardiness and Additional Penalties. *International Journal of Systems Science*, 30(12):1279–1284.

Yugma, C., Blue, J., Dauz`Ere-P´Er`Es, S., and Obeid, A. (2015). Integration of Scheduling and Advanced Process Control in Semiconductor Manufacturing: Review and Outlook. *Journal of Scheduling*, 18(2):195–205.

ZEE, D.-J. V. D., Harten, A. V., and Schuur, P. (2001). On-Line Scheduling of Multi-Server Batch Operations. *IIE Transactions*, 33(7):569–586.