Received: 2 January 2025 | Revised: 28 February 2025 | Accepted: 27 March 2025 | Published online: 26 May 2025

RESEARCH ARTICLE

Journal of Comprehensive Business Administration Research
2025, Vol. 00(00) 1–12
DOI: 10.47852/bonviewJCBAR52025143

# Optimizing Runtime Business Processes with Fair Workload Distribution

**Hiroki Horita**[1,*]

[1] *Graduate School of Science and Engineering, Ibaraki University, Japan*

**Abstract:** Predictive process monitoring, which utilizes historical event log data from previously executed business processes to provide support for processes currently in progress, has emerged as a prominent area of research in recent years. By leveraging this technology, organizations can improve work efficiency by allocating optimal human resources to tasks based on predictions. However, most of these studies primarily focus on minimizing task completion time, which often results in an imbalance of workload among the human resources executing the business processes. This imbalance can lead to overburdened employees and decreased overall productivity in the long term. In this study, a predictive model generated from event logs is used to forecast activities to be executed in the future and estimate their execution times. Based on these predictions, we propose an online human resource allocation strategy that prioritizes workload leveling. Validation using simulation-generated hospital event logs demonstrates that the proposed method successfully equalizes human resource workloads, albeit with a slight reduction in immediate work efficiency.

**Keywords:** process mining, predictive process monitoring, resource allocation, machine learning

## 1. Introduction

In recent years, data analysis has been used as an effective method for organizations to improve their business processes. Process mining is one such data analysis method that analyzes event logs, which are records of business activities recorded by information systems, and uses them for process improvement. Many process mining methods are offline methods that use event logs of past business processes that have already been completed. On the other hand, online process mining for running business processes has recently been attracting attention. Among them, predictive process monitoring [1, 2] is an approach that enables effective process execution by predicting and dealing with the future state of an executing business process [3, 4]. In addition to event logs, predictive process monitoring handles data called event streams, which are data from running business processes. By using machine learning to build predictive models from event logs and analyzing event streams, predictions can be provided for the completion time of an event being executed, the next activity, and so on.

The information provided by process mining can be applied to staffing, in which workers are assigned to tasks as human resources. Appropriate staffing using the results of process mining analysis can improve the work efficiency of business processes [5]. In the no-clairvoyant online job shop scheduling problem, where future activities are unknown, predictions obtained from predictive process monitoring can be used to assign personnel to efficiently complete tasks [6].

However, these methods prioritize completing tasks in the shortest possible time, which can result in resource allocation strategies that disproportionately burden specific human resources. Since uneven work hours among human resources increase the risk of work inefficiency due to overload, existing human resource allocation methods may have a negative impact on business processes in the long run.

In the field of resource allocation, workload balancing methods have been studied to reduce the risk of resource overload by leveling workloads [7]. Silaban and Margaretha [8] demonstrated in their study conducted in Indonesia that work–life balance has a positive impact on job satisfaction and employee retention. Furthermore, Herawaty et al. [9] showed through their research in the banking sector that work–life balance positively influences employee retention. Based on their research findings, it is suggested that improving work–life balance through workload leveling may contribute to enhanced job satisfaction and employee retention. However, in the field of predictive process monitoring, there is little research that considers workload leveling.

Based on a two-step method combining predictive process monitoring and human resource allocation proposed by a previous study [6], this study proposes a method for fair allocation of personnel with distributed workload during business process execution. A predictive model is built by analyzing the event logs generated by the simulation in a hospital to predict future activity and work hours of a running business process. It is believed that human resource allocation that takes workload leveling into account can reduce the risk of overload and promote work proficiency. Furthermore, it provides human resource allocation that improves both work hours and workload leveling.

The objective of this study is to propose a method for fair human resource allocation by leveraging predictive process monitoring to balance workloads during business process execution. By doing so, we aim to enhance process efficiency while mitigating the risk of resource overload.

***Corresponding author:** Hiroki Horita, Graduate School of Science and Engineering, Ibaraki University, Japan. Email: hiroki.horita.is@vc.ibaraki.ac.jp

The remainder of this paper is organized as follows: Section 2 describes event logs; Section 3 introduces research related to this paper; Section 4 describes workload leveling; Section 5 describes our proposed method; Section 6 describes the experiments, results, and discussion; and Section 7 concludes this paper.

## 2. Preliminaries

The event log is data that records the history of business processes executed through the information system. An example of an event log is shown in Table 1. The event log in Table 1 is the history of patient care and examinations in a hospital and contains case IDs, activities, resources, and timestamps of the start and end times of events. A case with the same case ID is called a trace or a process instance. In the event log shown in Table 1, each case in a business process corresponds to an individual patient and is distinguished by a case ID. The process, such as an examination, that each patient undergoes is associated with the patient's case as an event, and the work content of the event is recorded in the event log as an activity. Each event has attributes such as the resources required to perform the activity and the timestamps of the event start and end times, which are also recorded in the event log. In the event log shown in Table 1, the resources recorded are the human resources that executed the activity.

## 3. Related Works

In this section, we present related studies to this research.

### 3.1. Predictive process monitoring

Predictive process monitoring is defined as a method that takes as input the event logs of currently running and previously executed traces, with the goal of generating predictive models that can be used to predict specific values of process instances [10, 11]. Predictive process monitoring builds predictive models from the event logs of previously executed business processes as a first step to make predictions about the running business process. This is done using statistical methods such as stochastic queueing models used by Ha et al. [7] and machine learning methods used by Pika and Wynn [3]. Information about human resources in a business process can also be extracted from event logs, for example, by using a method that encodes human resource work experience as a feature [12]. In the next step, the predictive model is used to make predictions about the running business process. Tax et al. [13] used Long Short-Term Memory (LSTM) networks to predict both time and next events. The resulting forecasts can be used to support the execution of business processes, such as considering preventive measures against risks.

Predictive process monitoring can be applied to various domains in different countries. While this paper focuses on artificial logs in hospitals, there are cases where it has been applied to diverse domains and countries, including an Italian bank [14], a Dutch bank [13], a Belgian airport [15], and a multinational company in the coatings and paints industry in the Netherlands [16].

### 3.2. Resource allocation

Resource allocation in a business process aims to execute each task of a certain process instance at the right time and with the right resources [17]. Figure 1 shows the relationship between human resources and tasks in a business process. In Figure 1, human resources in a business process are associated with tasks that can be executed by each resource. The relationship between resources and executable tasks is not always one-to-one. For example, task $wi_5$ can only be executed by human resource $r_3$, but human resource $r_3$ can execute two different tasks $wi_4$ and $wi_5$. Since resources in a business process are usually limited, it is necessary to allocate resources efficiently to execute a given task under resource constraints.

Because of the importance and complexity of the resource allocation problem in business processes, various approaches have been developed in recent years to support it semi-automatically; the work of Pufahl et al. has systematically summarized approaches to resource allocation in business processes [5]. Most approaches to resource allocation in business processes are based on a process orientation that attempts to identify the optimal combination

**Figure 1**
**Relationship between resources and tasks**

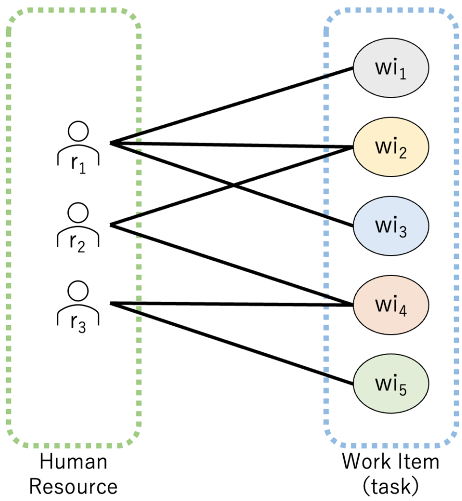



**Table 1**
**Example of event log**

| Case ID | Activity | Resource | Start timestamp | Complete timestamp |
|---|---|---|---|---|
| 1 | Registration | 1 | 25:00 | 26:00 |
| 1 | Triage and Assessment | 3 | 27:00 | 28:00 |
| 1 | Intravenous | 6 | 28:00 | 29:00 |
| 1 | X-ray | 8 | 30:00 | 31:00 |
| 1 | Evaluation | 14 | 31:00 | 32:00 |
| 1 | Admission | 20 | 33:00 | 34:00 |
| 1 | Discharge | 23 | 35:00 | 36:00 |

between tasks and resources based on time constraints, such as the capabilities and deadlines required to execute the tasks, and characteristics of the resources, such as their capabilities and experience. Research aiming to minimize the execution time of a process [18] and research aiming to minimize the cost of executing a process [19] fall under the process-oriented approach. On the other hand, there is also research on resource orientation that focuses more on resources in business processes. Research aiming to optimize human resources considering resource availability constraints [20, 21] and research aiming to optimize worklists that list tasks to be performed by each resource [22] fall into this category. Various algorithmic approaches have been proposed to achieve these goals. These include genetic algorithms [19], machine learning approaches [23, 24], and answer set programming [25]. Li et al. [26] conducted a study of resource allocation in emergency management. Häfke and van Zelst [27] proposed a method for checking the feasibility of resource-focused processes. Yeon et al. [28] proposed a resource allocation method focusing on social networks. Our research differs from these studies in that it is an approach that balances the efficiency of business process execution and fairness among human resources at the time of business process execution.

## 4. Workload Leveling

This section describes and defines workload leveling.

### 4.1. Workload leveling in business processes

In business processes, the workload of workers executing tasks is a factor that has a significant impact on the work efficiency of individuals and organizations. If the workload is excessive, the individual or organization will experience stress, which will reduce work efficiency [29]. On the other hand, appropriate assignment of tasks can develop workers' skills and provide opportunities for task mastery [30]. Insufficient workload may result in loss of human resource proficiency opportunities and motivation and thus lower work efficiency. Workload leveling is one of the methods that can avoid overload and underload. Ha et al. [7] proposed a process execution rule for workload leveling using individual worklists of human resources.

### 4.2. Definition of workload leveling

In a business process in which multiple human resources execute multiple activities, each human resource can be classified into groups according to the type of activity it executes. In this study, it is assumed that each human resource performs one or two types of activities and does not belong to more than one group. In this situation, it is not realistic to equalize the workload of all human resources equally if there are differences in the frequency and duration of each activity. Therefore, the workload leveling of human resources is performed within each group, which is divided by the type of activity to be performed. To achieve workload leveling within each group, this study adopts an approach that minimizes the variation in workload among human resources belonging to a group. The workload variance is defined as the difference between the largest workload and the smallest workload among the human resources in a group. The workload is defined as the cumulative work time that each human resource has spent performing an activity.

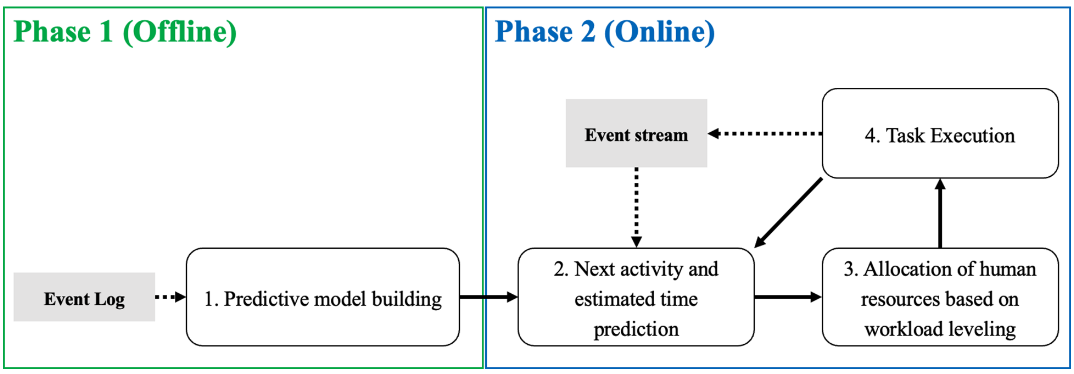
### 4.3. Workload leveling in process mining

Most resource allocation studies using process mining or predictive process monitoring do not consider workload leveling. This is due to the fact that many studies aim to complete the task at hand in the least amount of time. For example, if two human resources are responsible for the same type of task, and one is able to perform the task faster than the other, the better one can minimize the time to complete the work by completing more tasks [7]. Such a human resource allocation method can be said to be an allocation method that emphasizes short-term work efficiency because it minimizes the completion time of the current task at the expense of increasing the risk of overload by biasing the work toward superior human resources. Conversely, the human resource allocation proposed in this study reduces the current work efficiency instead of equalizing the task allocation. The improvement of work efficiency through the prevention of overload and the enhancement of learning opportunities and motivation may contribute to the improvement of business processes over a long period of time, instead of being difficult to achieve in a short period of time. For example, reducing overload situations may prevent worker absences and turnover, thereby limiting the loss of human resources. Therefore, such an approach is one that emphasizes long-term efficiency.

## 5. Proposed Method

A schematic diagram of the proposed method is shown in Figure 2. The method consists of two phases. Phase 1, which is performed offline, builds a prediction model from event logs using an LSTM neural network. Phase 2, which is executed online in parallel with the running business process, has three steps: predicting the

**Figure 2**
**Overview of the proposed method**



next task and the time required, allocating human resources based on workload leveling, and processing the execution. The three steps are repeatedly executed using event streams, which are data from the running business process.

## 5.1. Predictive model building

LSTM neural networks are used to build a prediction model that predicts the activities that will be executed in the future and the work time of the process instances in progress. This part is based on the work of Park and Song [6]. The flow of constructing the prediction model is shown in Figure 3 [7]. An LSTM neural network is constructed using feature vectors extracted from event logs as training data. The architecture proposed by Tax et al. [13] is used to construct the neural network. With this architecture, a shared LSTM layer for both prediction goals is followed by an LSTM layer specialized for predicting the next activity (red box in the figure) and an LSTM layer specialized for predicting the time until the next event (green box in the figure). Each LSTM layer is composed of consecutive LSTM cells, and the arrows connecting LSTM layers of the same color in Figure 3 indicate that multiple consecutive LSTM cells constitute an LSTM layer.

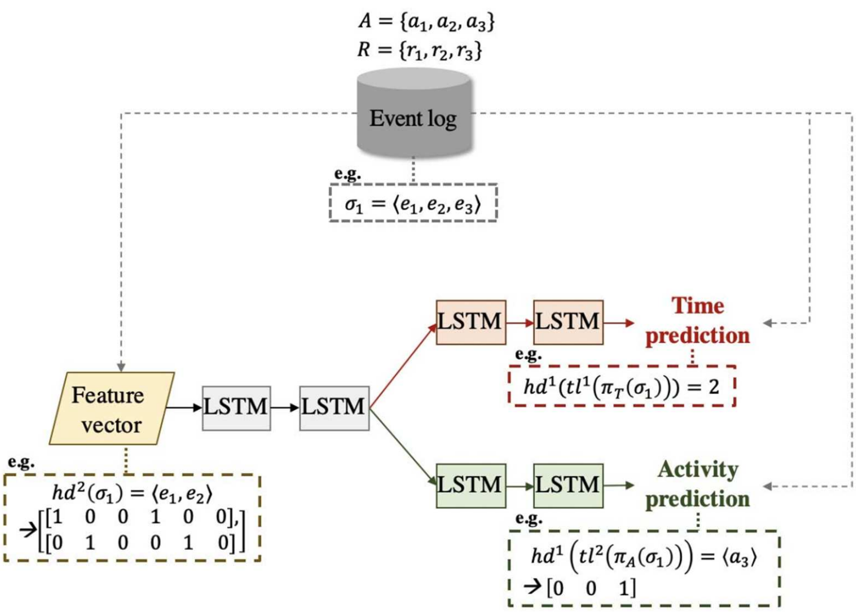
The prediction model to be built is trained to be able to predict the execution time $hd^1(tl^{k-1}(\pi T(\sigma)))$ of a running activity and the next activity $hd^1(tl^k(\pi A(\sigma)))$ with the event stream $hd^k(\sigma)$ as input. In the training example in Figure 3, the event stream $hd^2(\sigma_1) = <e_1, e_2>$ is used as input to learn the execution time of event $e_2$ (2) and the next activity ($a_3$). All previous events in the event log are vectorized by one-hot encoding. The length of the vector varies depending on the type of activity and the number of human resources involved, and the features corresponding to the activities performed and the human resources involved in each event are encoded as 1, while other values are encoded as 0. In the example in Figure 3, we consider the activities $A = \{a_1, a_2, a_3\}$ executed in the event log and the human resources $R = \{r_1, r_2, r_3\}$ that executed each activity. The input event stream $hd^2(\sigma_1) = <e_1, e_2>$ considers three types of activities and three human resources, so the length of the one-hot encoding vector is 6. If $\pi_A(e_1) = a_1$, $\pi_R(e_1) = r_1$, that is, event $e_1$ is associated with $a_1$, $r_1$, then $e_1$ is encoded as [1,0,0,1,0,0]. Similarly, $e_2$ is encoded as [0,1,0,0,1,0]. In this vector, the three elements on the left represent the execution of activities $a_1$, $a_2$, and $a_3$ in sequence, and the three elements on the right represent the human resources $r_1$, $r_2$, and $r_3$ that executed the events in sequence. The next activity, $hd^1(tl^k(\pi_A(\sigma)))$, which is the target for prediction, is also vectorized in a similar manner from the event log. In the example shown in Figure 3, activity $a_3$ is encoded as [0,0,1]. Another prediction target, the execution time of the ongoing activity $hd^1(tl^{k-1}(\pi_T(\sigma)))$, utilizes the numerical value of the working time recorded in the event log directly for learning. The generated vectors are used to train the model with the Adam algorithm to minimize the mean absolute error between the actual working time and the predicted working time, as well as the cross-entropy between the next activity and the predicted activity. Xavier initialization is used for weight initialization, and dropout and batch normalization are employed for regularization.

## 5.2. Next activity and estimated time prediction

Using the constructed prediction model and the event stream, predictions for the next activity and its execution time are generated. In practice, these predictions are created through two consecutive steps. As the first step, the data from the event stream is used as input to generate a prediction for the next activity. For example, the first prediction is generated by using the one-hot encoded vectors $\{e_1, e_2\}$ as input. The prediction generated at this step includes the execution time of event $e_2$ (3) and the next activity ($a_1$).

Next, artificial events are created by combining the generated predictions with past events, and these are used as the second input to the prediction model. An artificial event $\hat{e}_3$ corresponding to the predicted next activity $a_1$ is created. In this example, the predicted next activity ($a_1$) can be one-hot encoded in the same way as past events by considering the human resource ($r_1$) capable of executing it. By combining this with past events $\{e_1, e_2\}$, a one-hot encoded vector $\{e_1, e_2, \hat{e}_3\}$ is created and used as the input for the second prediction. Based on the input vector, the prediction generated is the execution time (2) of the next activity, which is the target of the prediction.

**Figure 3**
**Overview of the predictive model**



### 5.3. Allocation of human resources based on workload leveling

Based on the generated predictions, the optimal allocation of human resources is determined using the minimum-cost maximum-flow algorithm [31]. Algorithm 1 presents the algorithm for scheduling human resources. Additionally, Table 2 provides explanations of the symbols used in Algorithm 1. The algorithm takes as input a set of work items, which are the activities scheduled for execution, and the available human resources and outputs a resource allocation based on the leveling of working hours. In lines 1–14, a bipartite graph is constructed, consisting of two types of nodes: scheduled activities and human resources. An example of the constructed bipartite graph is shown in Figure 4. According to lines 2–4 of Algorithm 1, the leftmost source node is connected to the work item nodes via edges. The work item nodes may also include activities predicted for future execution, represented as dotted circles. Next, as specified in lines 5–7, the nodes representing available human resources are connected to the rightmost sink node via edges. Subsequently, in lines 8–14, the work item nodes are connected to their corresponding human resource nodes via edges. The edges linking work items and human resources are assigned costs, representing the variance in working hours among human resources in each group. These costs are calculated based on the cumulative workload of the human resources and the working hours of the activities.

**Algorithm 1**

Resource scheduling algorithm for workload balancing

**Input**: $\widehat{WI}$, $\hat{R}$

**Output**: Pseudo-Assignment $\hat{M}$

1: *Produce source node s, sink node t;*
2: **for** *node* $wi_{i,k} \in \widehat{WI}$ **do**
3: *add edge* $(s,\ wi_{i,k},\ (0,\ 1))$
4: **end for**
5: **for** *node* $r_j\ \in\ \hat{R}$ **do**
6: *add edge*$(r_j,\ t,\ (0,\ 1))$
7: **end for**
8: **for** *node* $wi_{i,k} \in \widehat{WI}$ **do**
9: **for** *node* $r_j\ \in\ \hat{R}$ **do**
10: $wl_j = wl_j + p_{i,k,j}$
11: $c \leftarrow \max(WL) - \min(WL)$
12: *add edge* $(wi_{i,k},\ r_j,\ (c,\ 1))$
13: **end for**
14: **end for**
15: $\hat{M} \leftarrow$ *MinCostMaxFlow*$(s, t)$
16: **return** $\hat{M}$

The method for calculating the cost of edges is illustrated in Figure 5. The bar graph in the figure represents the cumulative workload of human resources A, B, and C, with the shaded portions indicating the workload from newly assigned activities. Since A, B, and C belong to the same group of human resources, they can execute the same types of activities. When assigning a new activity, three cost patterns are calculated under the assumption that the activity is assigned to each of A, B, and C. Figure 5 specifically illustrates the pattern where the activity is assigned to B. Using the predicted execution time required for B to perform the new activity, the workload of B at the point when the execution of the new activity is completed can be calculated. The variance in workload among the human resources at that point is determined by the difference between the maximum workload (B's workload) and the minimum workload (A's workload). The calculated variance in workload among the human resources is then set as the cost for the edge connecting the corresponding work item and human resource nodes in the bipartite graph.

**Table 2**
**Explanation of symbols in Algorithm 1**

| Notation | Description |
|---|---|
| *I, R, A* | Set of instances, resources, and activities |
| *WI* | Set of work items |
| *WL* | Set of resources workload in group |
| $wi_{i,k}$ | $k_{th}$ work instance $I_i$ |
| $wl_j$ | Total workload of $R_j$ |
| $p_{i,j,k}$ | Processing time of work item $wi_{i,k}$ by $R_j$ |
| *c* | Cost of edge |
| *s, t* | Source node, sink node |

**Figure 4**
**Example of a constructed bipartite graph**

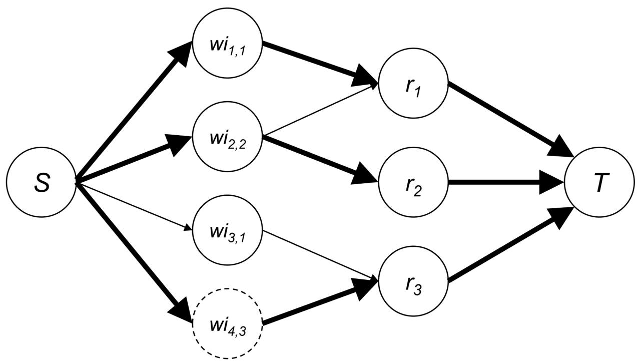
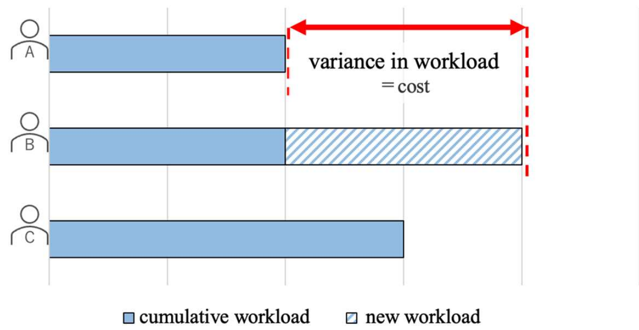



**Figure 5**
**How to calculate costs**




Next, in line 15 of Algorithm 1, the allocation of human resources is determined using the minimum-cost maximum-flow algorithm based on the network simplex method [31]. This algorithm selects paths on the bipartite graph that minimize the costs assigned to the edges. In the example shown in Figure 4, the selected paths are represented by bold lines. Among the four work items, three are assigned to the corresponding human resources connected by the bold lines. The selected paths correspond to a human resource allocation based on workload leveling, minimizing the variance in workload among the human resources.

In this study, we aim to optimize resource allocation in a non-clairvoyant online environment and adopt an approach that integrates

LSTM-based prediction with the minimum-cost and maximum-flow algorithm, rather than using reinforcement learning or heuristic-based methods. Heuristic approaches assume that processing times and job sequences are known in advance, making them unsuitable for our target environment, where such information is unavailable beforehand. Additionally, reinforcement learning requires extensive trial-and-error learning, which is impractical in business environments due to the difficulty of collecting sufficient training data and the necessity for immediate optimal resource allocation. Therefore, we employ LSTM to predict processing times and the next activity, leveraging these predictions within a minimum-cost and maximum-flow algorithm to achieve efficient scheduling.

## 5.4. Task execution

Tasks are executed based on the human resource allocation generated in the previous steps. Since the human resource allocation generated in the previous steps includes pseudo-assignments that involve tasks not currently executable, filtering is applied to remove such tasks. This ensures that the resulting human resource allocation only includes tasks that can be executed at the current time, where both the corresponding instances and the human resources are available at the present moment. For example, predicted future activities are executed when they become executable, unlike other activities already executable at the current time. Additionally, if human resources are unavailable due to delays or other reasons, the task execution will wait until the resources become available. The data from executed instances is used as a new event stream for generating predictions. In other words, the three steps within Phase 2 are repeatedly executed online in parallel with the ongoing business process.

## 5.5. Multi-objective optimization

By combining the workload leveling optimization method from Section 5.4 with an existing method [6] that optimizes the weighted total task completion time, a multi-objective optimization approach can be provided. In this study, a simple goal programming approach [32] is employed to achieve multi-objective optimization, addressing the allocation of human resources in ongoing business processes. In goal programming-based multi-objective optimization, multiple objectives are optimized by minimizing the distance between the target values and the actual values for each objective. When future activities and their working times are unknown, it is not possible to reverse-calculate the target working times for human resources from the total working time of the final activities. To address this issue, this study does not adopt a specific working time as the target value. Instead, it adopts the variance in workload among human resources within a group at the given point in time as the target value. The workload variance is calculated as the difference between the maximum and minimum workloads of the human resources in each group and is set as the cost for the edges in the bipartite graph.

Algorithm 2 illustrates the scheduling of human resources using the multi-objective optimization approach. Table 3 provides explanations of the symbols used in Algorithm 2. In line 11 of Algorithm 2, the costs for the multi-objective optimization approach, which are assigned to the edges of the bipartite graph, are calculated. Among the expressions assigned to the cost c, the left term is the cost $(p_{i,k,j} + \max(ri_i, rr_j, 0))/w_i$ defined in previous research, multiplied by a weight coefficient $\lambda$ representing its importance. In the existing method [6], this cost was designed to minimize the planned working time of human resources, taking into account factors such as the weight of each instance, to optimize the weighted total task completion time. On the other hand, the right term is the cost (max (WL)−min (WL)), proposed in this study, multiplied by a weight coefficient $1-\lambda$, which also represents its importance. Since both terms are calculated based on the time it takes for human resources to perform an activity in a single instance, they can be directly compared. The multi-objective optimization approach allocates human resources by considering both the proposed cost and the cost of the existing method. By comparing the results of resource allocation under different importance settings in the multi-objective optimization approach, organizational decision-makers can select the human resource allocation that is most beneficial to their objectives.

**Algorithm 2** **Resource scheduling algorithm for multi-optimization**

**Input**: $\widehat{WI}$, $\hat{R}$
**Output**: Pseudo-Assignment $\widehat{M}$
1: *Produce source node s, sink node t;*
2: **for** *node* $wi_{i,k} \in \widehat{WI}$ **do**
3: *add edge* $(s,\ wi_{i,k},\ (0,\ 1))$
4: **end for**
5: **for** *node* $r_j \in \hat{R}$ **do**
6: *add edge*$(r_j,\ t,\ (0,\ 1))$
7: **end for**
8: **for** *node* $wi_{i,k} \in \widehat{WI}$ **do**
9: **for** *node* $r_j \in \hat{R}$ **do**
10: $wl_j = wl_j + p_{i,k,j}$
11: $c \leftarrow \lambda(p_{i,k,j} + \max(\mathrm{ri}_i,\ \mathrm{rr}_j,\ 0))/w_i + (1-\lambda)(\max(WL) - \min(WL))$
12: *add edge* $(wi_{i,k},\ r_j,\ (c,\ 1))$
13: **end for**
14: **end for**
15: $\widehat{M} \leftarrow$ *MinCostMaxFlow*$(s, t)$
16: **return** $\widehat{M}$

**Table 3**
**Explanation on symbols in Algorithm 2**

| Notation | Description |
|---|---|
| $\lambda$ | Priority level of goal |
| $w_i$ | Weight of instance $I_i$ |
| $ri_i$ | Remaining time for $I_i$ |
| $rr_j$ | $R_j$ to be ready |

## 6. Experiments

Two experiments were conducted to verify the effectiveness of the proposed method. First, we evaluate whether the proposed workload leveling method can allocate human resources and eliminate workload concentration. Second, we verify whether a multi-objective optimization method that considers both total work completion time and workload leveling can provide useful alternatives.

## 6.1. Experimental setting

In the experiment, an artificial event log was generated from a 7-day simulation of a hospital emergency department operating 24 hours a day, handling 1000 patients with 25 human resources. The event log records the case ID, activity, human resource responsible for the activity, the start and end time of the event on the simulation, timestamps of the start and end time, weights, and the time taken to execute the event. It also generates an event stream of 90 patient visits at regular intervals as data for the running business process to generate human resource assignments. Each patient is assigned a weight from 1 to 10, assuming that the weights follow a uniform distribution. Each human resource can perform one or two of a total of 11 different activities. These human resources are classified into eight groups according to the types of activities they can perform. Table 4 shows the classification of human resources. The human resources in Group 4 can execute the activity “X-ray” in common. However, Resources 11, 12, and 13 can additionally perform the activity “MRI.” Therefore, Resources 11, 12, and 13 are considered to be in a different group from Resources 8, 9, and 10 when the allocation of personnel to the activity “MRI” is considered. The execution time of the activity corresponding to each resource depends on the proficiency level of each resource, which is set assuming that it follows a Gaussian distribution. In the experiments, human resources are assigned to the generated event streams using the proposed method. The proposed method allocates human resources in such a way that the variation of workload is minimized in each of the eight human resource groups. The human resource allocation is performed in parallel with the event stream running in the simulation, and the human resource responsible for a task in the simulation is determined. Each unit of time that elapses in the simulation, a task is executed by the determined human resource, and the human resource allocation is performed again. This process is repeated until all process instances have been executed. The program used in the experiments is based on the one published on GitHub by Park and Song [6].

The simulation conditions in this study were designed based on prior research [6], following similar assumptions used in previous studies. While we acknowledge that real-world validation would provide stronger empirical support, obtaining actual hospital event logs is challenging due to privacy concerns and data availability constraints. Moreover, even if such data were accessible, implementing the proposed method for real-world workforce allocation in hospitals presents additional challenges due to operational constraints and the complexity of human resource management in healthcare settings. Although simulation-based approaches have limitations, they are often used in healthcare workflow research as a practical alternative when direct implementation is not feasible.

## 6.2. Experiments in workload leveling

The experiments in this section compare the total workload and total work completion time of each resource at the time all work is completed with the existing method [6]. By doing so, we verify whether the human resource allocation generated by the proposed method can achieve workload leveling.

**Table 4**
**Human resource groups and activities for which they are responsible**

| Group | Resources | Activity |
|---|---|---|
| Group1 | Resources 1,2 | Registration |
| Group2 | Resources 3,4,5 | Triage and Assessment |
| Group3 | Resources 6,7 | Intravenous |
| Group4-a | Resources 8,9,10 | X-ray |
| Group4-b | Resources 11,12,13 | X-ray, MRI |
| Group5 | Resources 14,15,16 | Evaluation, Diagnosis |
| Group6 | Resources 17,18,19 | Blood Test, Urine Test |
| Group7 | Resources 20,21,22 | Admission |
| Group8 | Resources 23,24,25 | Discharge |

### *6.2.1. Results*

The total workload for each human resource when all instances of human resource allocation have been completed using the existing method [6] is shown in Figure 6. Three out of 8 groups of 25 human resources are shown, with each group color-coded. In the experiment, the workload was concentrated on the best human resources (Resources 6, 14, and 23 in Figure 6) with the shortest activity work time in all eight human resource groups. On the other hand, the final workload was smaller for the beginner human resources. In some groups, there were human resources that were never assigned a task, resulting in a total workload of zero. Similarly, Figure 7 shows the total workload of each human resource when all instances have been executed using the proposed method for human resource allocation. The results for the same three groups as in Figure 6 are shown in different colors. In all groups, the variation in work time among human resources is smaller than in the existing method. In the existing method, the workload was concentrated on the most talented person in the group, but in the proposed method, the workload is evenly distributed to each human resource. In addition, the human resource who has never been assigned a task in the existing method is given the same workload as other human resources in the same group.

Table 5 compares the resource allocation results between the existing method and the proposed method. The workload variation decreased from 60.250 to 1.625 (97% reduction), the average workload of the best human resource decreased from 70.875 to 44.875 (37% reduction), the average workload of beginner human resources increased from 10.625 to 44.50 (319% increase), the weighted completion time increased from 4769 to 6165 (29% increase), and the total work time increased from 805 to 1073 (33% increase).

### *6.2.2. Discussion*

A comparison of the experimental results between the existing and proposed methods is shown in Table 5. The variation of total workload is calculated by averaging the variation of workload among human resources in each group, based on the definition introduced in Section 4.2. The proposed method reduces the variation of the total workload by 97% compared to the existing method. Experiments showed that the proposed method can achieve workload leveling by eliminating the workload imbalance that has been a problem with the existing method. In the experiment, the proposed method achieves perfect leveling in which the work time of all resources is equal in all four groups, including the group to which Resources 23, 24, and 25 belong, as shown in Figure 7. In practice, whether perfect leveling is achieved depends on the number of times the task is performed and the size of the difference in proficiency between the human resources. For example, if a group that has achieved perfect leveling is given a new task, perfect leveling is no longer achieved when that task is executed. In other words, nonzero workload variation among human resources within a group is an acceptable event in the method.

The average workload of the best human resource in each group with the shortest activity work time was reduced by 37% by the proposed method. This result indicates that the overload situation caused by the concentration of tasks on a particular human resource can be avoided by using the proposed method for human resource allocation. Since human resources’ work efficiency decreases when they become overloaded, reducing the risk of overloading is considered to maintain the work efficiency of

**Figure 6**
**Total workload of resources in existing method**

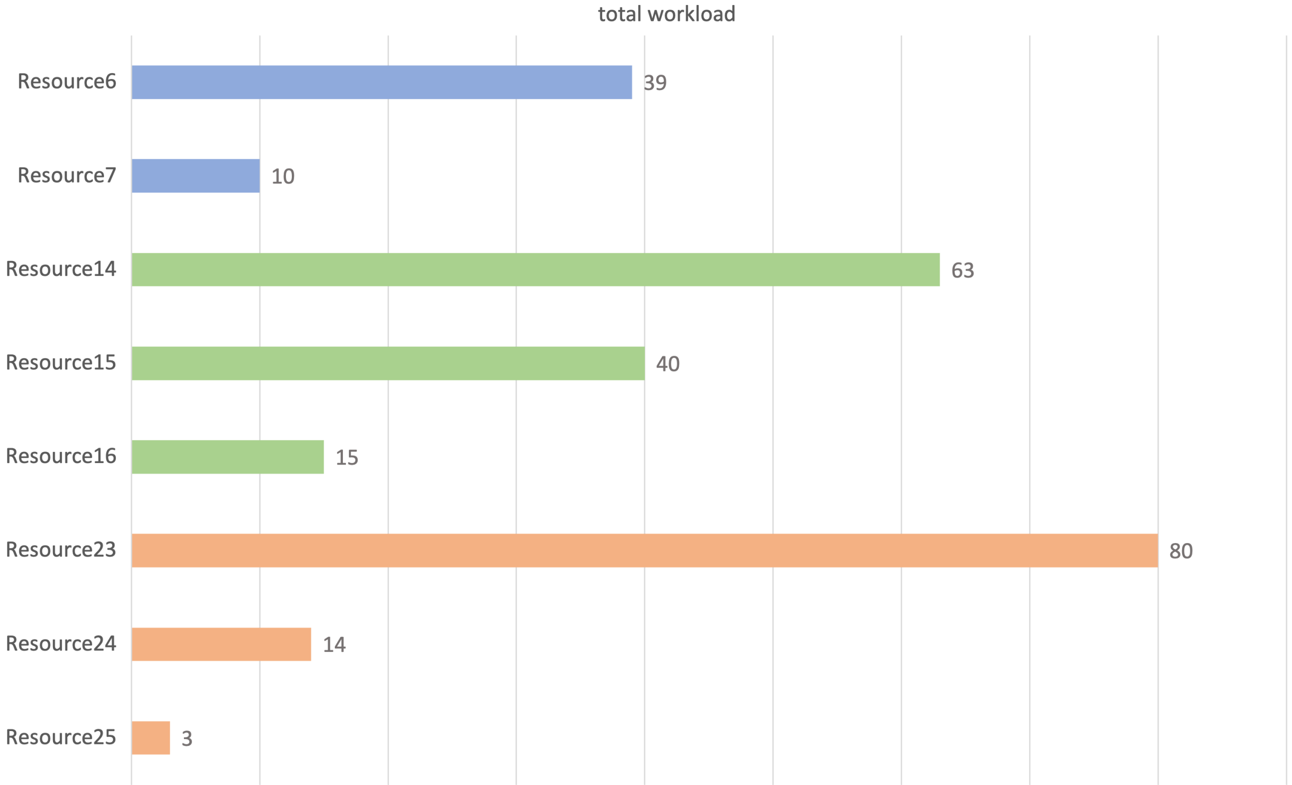



**Figure 7**
**Total workload of resources in the proposed method**

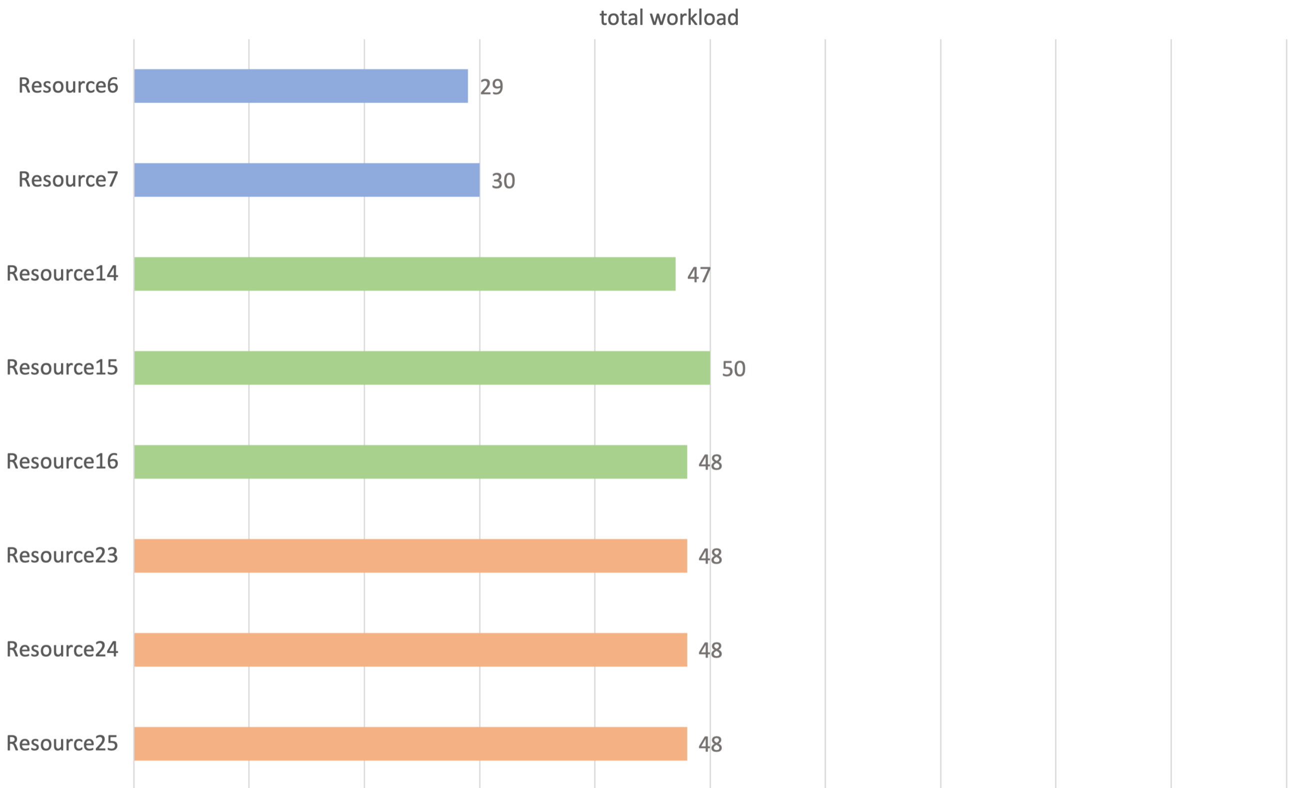


**Table 5**
**Comparison of resource allocation between existing method and proposed method**

| | Existing method | Proposed method | Change |
|---|---|---|---|
| Variation of total workload | 60.250 | 1.625 | –97% |
| Average workload of best human resource | 70.875 | 44.875 | –37% |
| Average workload of beginner human resource | 10.625 | 44.50 | +319% |
| Weighted completion time | 4769 | 6165 | +29% |
| Total work time | 805 | 1073 | +33% |

human resources from a long-term perspective. Avoiding situations in which human resources become overloaded is also considered to prevent the loss of human resources due to leave of absence or retirement caused by overwork. In existing methods, workload is concentrated on excellent human resources whose time required per task execution is short. The loss of excellent human resources not only greatly affects the execution of business processes but also results in significant losses for the organization. The proposed method, which can prevent the loss of excellent human resources by improving the execution environment of business processes, is considered to be useful for organizations.

The average workload of the beginner human resource with the longest activity work time in each group increased by 319% with the proposed method. This result shows that the proposed method increases the learning opportunities for beginner human resources compared to the existing methods by providing them with the same workload as other human resources. It is important to ensure that beginner human resources have the opportunity to become proficient under the assumption that human resources improve their work efficiency when they perform tasks and become proficient, such as in on-the-job training. Beginner human resources correspond to new employees or human resources with limited task experience in the organization that executes the business process. If beginner human resources are not allowed to experience tasks as in the case of existing methods, they will not be able to learn, and experienced, superior human resources will be in charge of many tasks. When this type of human resource allocation is repeated, the gap in proficiency between beginner human resources and superior human resources will continue to widen. In such a situation, if an excellent human resource is lost due to retirement or other reasons, it becomes difficult to execute a task efficiently because there is no human resource who can execute the task with sufficient experience. The proposed method of assigning tasks to beginner human resources so that the workload is equalized within the group will ensure that beginner human resources have opportunities to become proficient and will meet the need for skill development of human resources in an organization.

The weighted completion time is the sum of the time required to execute all instances, calculated by considering the importance of each instance. Compared to the existing method, in which human resources are allocated to minimize the weighted completion time, the proposed method increases the weighted completion time by 29%. The total work time, which is the sum of the work time of all activities performed by each resource, increased by 33%. These results indicate that the proposed method decreases the current work efficiency due to workload leveling. The proposed method also assigns tasks to beginner human resources in order to achieve workload leveling. Therefore, when a task is executed the same number of times, the total time required to execute these tasks is longer than that of existing methods.

## 6.3. Experiments in multi-objective optimization

In the experiments in this section, we generate human resource assignments using a multi-objective optimization method that combines existing methods [6] and our proposed method and verify the relationship between the weighted total work completion time and the variation of the total workload. This confirms that the multi-objective optimization method is a useful option as a human resource allocation method. In the experiment, the importance levels of the existing and proposed multi-objective optimization methods varied. If the importance of the existing method is $\lambda$ and that of the proposed method is 1-$\lambda$, the total importance is 1 (100%). Five sets of multi-objective optimization ($\lambda$ = 80%, 60%, 50%, 40%, 20%) are provided, as well as two sets of single-objective optimization experiments with the existing method ($\lambda$ = 100%) and the proposed method ($\lambda$ = 0%).

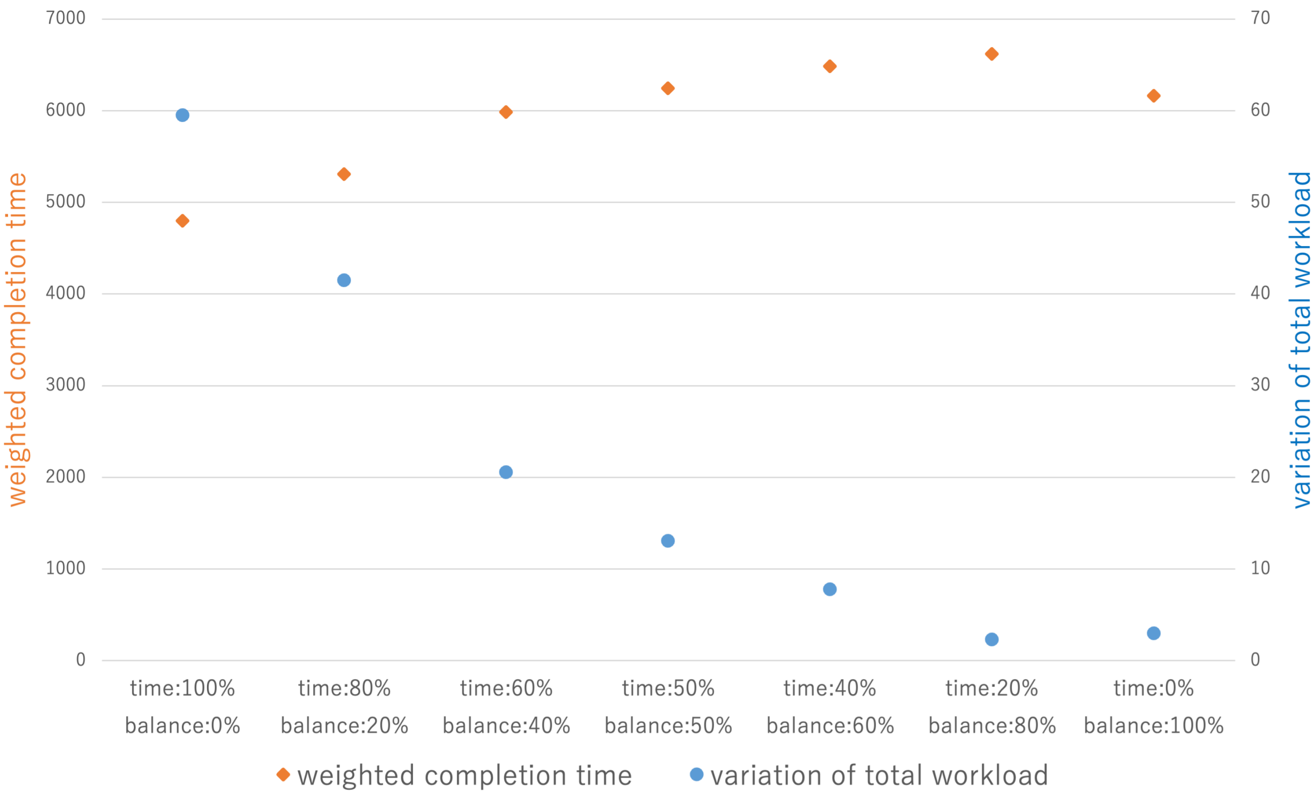
### *6.3.1. Results*

Figure 8 presents the experimental results comparing the existing and proposed methods. The orange dots indicate the weighted total work completion time (left vertical axis), while the blue dots represent the workload variation among human resources (right vertical axis). The horizontal axis reflects the trade-off between completion time and workload balance. The leftmost side (time: 100%, balance: 0%) represents the existing method, which focuses on minimizing completion time, resulting in high workload imbalance. The rightmost side (time: 0%, balance: 100%) corresponds to the proposed method, which prioritizes workload balance at the cost of increased completion time. The results show that as the importance of workload balance increases, the workload variation decreases significantly, while the completion time increases moderately, demonstrating the trade-off inherent in the optimization strategy.

### *6.3.2. Discussion*

From left to right, the weighted total work completion time increases, while the variability of the total workload among human resources decreases. This result indicates that there is a trade-off between the weighted total work completion time and the variability of the total workload among human resources. That is, the variability of both cannot be minimized simultaneously. An organization can choose any combination of these alternatives when allocating human resources. How to choose depends on various factors such as the organization and business process. For example, if an organization wants to minimize the variation of total workload among human resources under the constraint that the weighted total work completion time must be less than 6000 in terms of delivery and operation time, it can choose the third human resource allocation method from the left (time: 60%, balance: 40%) in Figure 8. In this way, the multi-objective optimization method can provide a choice

**Figure 8**
**Trade-offs with multi-objective optimization methods**




of useful human resource allocation methods that can be selected arbitrarily according to various situations in an organization.

## 7. Conclusion

It is important to consider the human resources that actually perform the tasks in a business process. Allocating human resources to appropriate tasks can improve work efficiency and reduce risk. In this study, based on the analysis results of predictive process monitoring, we proposed a method for allocating human resources that considers the leveling of the workload of each resource. We generated a forecasting model using predictive process monitoring and LSTM neural networks and minimized the variation of workload among human resources using the obtained forecasts, thereby providing a human resource allocation method based on workload leveling in business processes where future tasks are unknown. This method not only reduces the risk of overloading human resources but also facilitates the learning of human resources who lack experience working on a task. Compared to the existing method, the proposed method is expected to support business process improvement from a longer-term perspective.

Beyond business process efficiency, workload leveling has important social and organizational implications. Prior studies have shown that improving work–life balance positively impacts job satisfaction and employee retention. For example, Silaban et al. demonstrated that work–life balance contributes to higher job satisfaction and employee retention in Indonesia [8], while Herawaty et al. [9] found similar positive effects in the banking sector. Their findings suggest that enhancing work–life balance through workload leveling may lead to improved job satisfaction and employee retention. By reducing the burden on individual workers and promoting a more sustainable work environment, our approach has the potential to contribute not only to operational efficiency but also to long-term human resource stability.

Although real-world validation remains a challenge, our approach has potential applications as a decision-support tool in hospital workflow management. By integrating our workload leveling model into hospital simulation tools, healthcare administrators can test different resource allocation strategies in a controlled environment before implementing them in practice. This enables hospitals to explore workload balancing strategies without requiring direct access to sensitive patient data.

One of the challenges in this study is verifying the impact of workload leveling on business process improvement. The experiments conducted in this study did not account for the effects on completion time resulting from decreased work efficiency due to overload or increased efficiency driven by motivation and proficiency. Simulations incorporating the impact of workload leveling could provide insights into its potential for long-term business process improvement. The other challenge is to use explainable machine learning to provide resource-focused explanations. In recent years, the explanatory nature of predictive process monitoring has been required [33–35]. High explanatory power makes resource allocation more convincing. Furthermore, there remain challenges when applying this method in dynamic and unpredictable environments. In real business processes, unexpected tasks and sudden changes in resource availability may occur. To address such situations, it is necessary to build a mechanism that dynamically updates the predictive model and reallocates resources in real time. In the future, introducing online learning techniques that can adapt to environmental changes should be considered to develop a more practical and flexible human resource allocation method.

## Ethical Statement

This study does not contain any studies with human or animal subjects performed by the author.

## Conflicts of Interest

The author declares that he has no conflicts of interest to this work.

## Data Availability Statement

Data available on request from the corresponding author upon reasonable request.

## Author Contribution Statement

**Hiroki Horita**: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Visualization, Writing – review & editing, Supervision, Project administration.

## References

[1] Ceravolo, P., Comuzzi, M., de Weerdt, J., Francescomarino, Di., C, & Maggi, F. M. (2024). Predictive process monitoring: Concepts, challenges, and future research directions. *Process Science*, *1*(1), 2. https://doi.org/10.1007/s44311-024-00002-4

[2] Kubrak, K., Milani, F., Nolte, A., & Dumas, M. (2022). Prescriptive process monitoring: Quo vadis? *PeerJ Computer Science*, *8*, e1097. https://doi.org/10.7717/peerj-cs.1097

[3] Pika, A., & Wynn, M. T. (2021). A machine learning based approach for recommending unfamiliar process activities. *IEEE Access*, *9*, 104969–104979. https://doi.org/10.1109/ACCESS.2021.3096513

[4] Rama-Maneiro, E., Vidal, J. C., & Lama, M. (2021). Deep learning for predictive business process monitoring: Review and benchmark. *IEEE Transactions on Services Computing*, *16*(1), 739–756. https://doi.org/10.1109/TSC.2021.3139807

[5] Pufahl, L., Ihde, S., Stiehle, F., Weske, M., & Weber, I. (2021). Automatic resource allocation in business processes: A systematic literature survey. *arXiv Preprint:2107.07264*.

[6] Park, G., & Song, M. (2019). Prediction-based resource allocation using LSTM and minimum cost and maximum flow algorithm. In *International Conference on Process Mining*, 121–128. https://doi.org/10.1109/ICPM.2019.00027

[7] Ha, B. H., Bae, J., Park, Y. T., & Kang, S. H. (2006). Development of process execution rules for workload balancing on agents. *Data & Knowledge Engineering*, *56*(1), 64–84. https://doi.org/10.1016/j.datak.2005.02.007

[8] Silaban, H., & Margaretha, M. (2021). The impact work-life balance toward job satisfaction and employee retention: Study of millennial employees in Bandung City, Indonesia. *International Journal of Innovation and Economic Development*, *7*(3), 18–26. https://doi.org/10.18775/ijied.1849-7551-7020.2015.73.2002

[9] Herawaty, M. T., Asmadi, I., Rahayu, E. I. H., & Zahra, Z. (2024). Impact of work-life balance, job satisfaction, and stress levels on employee retention in banking sector companies. *International Journal of Business, Law, and Education*, *5*(2), 2405–2417. https://doi.org/10.56442/ijble.v5i2.875

[10] Lu, K., Fang, X., & Fang, X. (2022). Business process outcome prediction based on deep latent factor model. *Electronics*, *11*(9), 1509. https://doi.org/10.3390/electronics11091509

[11] Weinzierl, S., Dunzer, S., Zilker, S., & Matzner, M. (2020). Prescriptive business process monitoring for recommending next best actions. In D. Fahland, C. Ghidini, J. Becker, & M. Dumas (Eds.), *Business process management forum* (pp. 193–209). https://doi.org/10.1007/978-3-030-58638-6_12

[12] Kim, J., Comuzzi, M., Dumas, M., Maggi, F. M., & Teinemaa, I. (2022). Encoding resource experience for predictive process monitoring. *Decision Support Systems*, *153*, 113669. https://doi.org/10.1016/j.dss.2021.113669

[13] Tax, N., Verenich, I., la, Rosa., M, M., & Dumas. (2017). Predictive business process monitoring with LSTM neural networks. In *Advanced Information Systems Engineering: 29th International Conference*, 477–492. https://doi.org/10.1007/978-3-319-59536-8_30

[14] Galanti, R., Coma-Puig, B., de Leoni, M., Carmona, J., & Navarin, N. (2020). Explainable predictive process monitoring. In *2nd International Conference on Process Mining*, 1–8. https://doi.org/10.1109/ICPM49681.2020.00012

[15] Gunnarsson, B. R., van den Broucke, S. K., & de Weerdt, J. (2019). Predictive process monitoring in operational logistics: A case study in aviation. In *Business Process Management Workshops: BPM 2019 International Workshops*, 250–262. https://doi.org/10.1007/978-3-030-37453-2_21

[16] Brunk, J. (2020). Structuring business process context information for process monitoring and prediction. In *IEEE 22nd Conference on Business Informatics*, 39–48. https://doi.org/10.1109/CBI49978.2020.00012

[17] Kumar, A., van der Aalst, W. M., & Verbeek, E. M. (2002). Dynamic work distribution in workflow management systems: How to balance quality and performance. *Journal of Management Information Systems*, *18*(3), 157–193. https://doi.org/10.1080/07421222.2002.11045693

[18] Middelhuis, J., Bianco, R. L., Sherzer, E., Bukhsh, Z., Adan, I., & Dijkman, R. (2025). Learning policies for resource allocation in business processes. *Information Systems*, *128*, 102492. https://doi.org/10.1016/j.is.2024.102492

[19] Huang, Z., Lu, X., & Duan, H. (2012). A task operation model for resource allocation optimization in business process management. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *42*(5), 1256–1270. https://doi.org/10.1109/TSMCA.2012.2187889

[20] Xu, J., Liu, C., Zhao, X., Yongchareon, S., & Ding, Z. (2016). Resource management for business process scheduling in the presence of availability constraints. *ACM Transactions on Management Information Systems*, *7*(3), 9. https://doi.org/10.1145/2990197

[21] Kunkler, M., & Rinderle-Ma, S. (2024). Online resource allocation to process tasks under uncertain resource availabilities. In *6th International Conference on Process Mining*, 137–144. https://doi.org/10.1109/ICPM63005.2024.10723280

[22] Yaghoubi, M., & Zahedi, M. (2016). Resource allocation using task similarity distance in business process management systems. In *2nd International Conference of Signal Processing and Intelligent Systems*, 1–5. https://doi.org/10.1109/ICSPIS.2016.7869851

[23] Huang, Z., van der Aalst, W. M., Lu, X., & Duan, H. (2010). An adaptive work distribution mechanism based on reinforcement learning. *Expert Systems with Applications*, *37*(12), 7533–7541. https://doi.org/10.1016/j.eswa.2010.04.091

[24] Durán, F., Pozas, N., & Rocha, C. (2024). Business processes resource management using rewriting logic and deep-learning-based predictive monitoring. *Journal of Logical and Algebraic Methods in Programming*, *136*, 100928. https://doi.org/10.1016/j.jlamp.2023.100928

[25] Havur, G., Cabanillas, C., & Polleres, A. (2022). Benchmarking answer set programming systems for resource allocation in business processes. *Expert Systems with Applications*, *205*, 117599. https://doi.org/10.1016/j.eswa.2022.117599

[26] Li, H., Liu, C., Zeng, Q., He, H., Ren, C., Wang, L., & Cheng, F. (2021). Mining emergency event logs to support resource allocation. *IEICE Transactions on Information and Systems*, *104*(10), 1651–1660. https://doi.org/10.1587/transinf.2021EDP7029

[27] Häfke, H., & van Zelst, S. J. (2023). Event data-driven feasibility checking of process schedules. In *Advanced Information Systems Engineering: 35th International Conference*, 210–225. https://doi.org/10.1007/978-3-031-34560-9_13

[28] Yeon, M. S., Lee, Y. K., Pham, D. L., & Kim, K. P. (2022). Experimental verification on human-centric network-based resource allocation approaches for process-aware information systems. *IEEE Access*, *10*, 23342–23354. https://doi.org/10.1109/ACCESS.2022.3152778

[29] Hockey, G. R. J. (1997). Compensatory control in the regulation of human performance under stress and high workload: A cognitive-energetical framework. *Biological Psychology*, *45*(1–3), 73–93. https://doi.org/10.1016/S0301-0511(96)05223-4

[30] Dymock, D., & McCarthy, C. (2006). Towards a learning organization? Employee perceptions. *The Learning Organization*, *13*(5), 525–537. https://doi.org/10.1108/09696470610680017

[31] Cunningham, W. H. (1976). A network simplex method. *Mathematical Programming*, *11*, 105–116. https://doi.org/10.1007/BF01580379

[32] Tamiz, M., Jones, D. F., & El-Darzi, E. (1995). A review of goal programming and its applications. *Annals of Operations Research*, *58*, 39–53. https://doi.org/10.1007/BF02032309

[33] Galanti, R., de Leoni, M., Monaro, M., Navarin, N., Marazzi, A., Di Stasi, B., & Maldera, S. (2023). An explainable decision support system for predictive process analytics. *Engineering Applications of Artificial Intelligence*, *120*, 105904. https://doi.org/10.1016/j.engappai.2023.105904

[34] Rizzi, W., Di Francescomarino, C., & Maggi, F. M. (2020). Explainability in predictive process monitoring: When understanding helps improving. In D. Fahland, C. Ghidini, J. Becker, & M. Dumas (Eds.), *Business process management forum* (pp. 141–158). https://doi.org/10.1007/978-3-030-58638-6_9

[35] El-Khawaga, G., Abu-Elkheir, M., & Reichert, M. (2022). XAI in the context of predictive process monitoring: An empirical analysis framework. *Algorithms*, *15*(6), 199. https://doi.org/10.3390/a15060199

**How to Cite:** Horita, H. (2025). Optimizing Runtime Business Processes with Fair Workload Distribution. *Journal of Comprehensive Business Administration Research*. https://doi.org/10.47852/bonviewJCBAR52025143