A method to evaluate process performance by integrating time and resources

Yu Wang¹, Qingjie Wei², Shuang Jin¹
¹College of Software Engineering, Chong Qing University of Posts and Telecommunications, Chongqing, 400065, China
²College of Computer Science and Technology, Chong Qing University of Posts and Telecommunications, Chongqing, 400065, China

Abstract. The purpose of process mining is to improve the existing process of the enterprise, so how to measure the performance of the process is particularly important. However, the current research on the performance evaluation method is still insufficient. The main methods of evaluation are mainly using time or resource. These basic statistics cannot evaluate process performance very well. In this paper, a method of evaluating the performance of the process based on time dimension and resource dimension is proposed. This method can be used to measure the utilization and redundancy of resources in the process. This paper will introduce the design principle and formula of the evaluation algorithm. Then, the design and the implementation of the evaluation method will be introduced. Finally, we will use the evaluating method to analyse the event log from a telephone maintenance process and propose an optimization plan.

1 Introduction
Process mining [1] is used to optimize the business process. Process performance analysis is used to evaluate the optimization result. The process performance can be analysed through the related attributes in the event log. These attributes mainly include performance, resources and quality [2]. At present, the research on process performance evaluation method is relatively scarce. Van Beest used the process throughput as a performance evaluation indicator when analysed the process of a gas company in the Netherlands [3]. A. Rozinat used the process execution time and process waiting time as the evaluation indicators of process performance in the study of the public complaint handling process in the municipal service of the Netherlands [4]. In the study of Van der Aalst. The time-related indicators such as synchronization time, process time and waiting time are proposed to measure the performance of the process [5]. The existing evaluation methods mainly use basic statistical information to evaluate, but these basic statistics is inaccurate to evaluate process performance. Therefore, this paper presents a method to evaluate the process performance of time and resources. This is a more comprehensive and accurate evaluation method to evaluate process resource utilization.

2 Resource Utilization
In the current study of resource utilization, the computational formula of resource utilization is as Equation (1) [6].

\[ UR(\text{Role}) = \frac{RBT(\text{Role})}{RWT(\text{Role})} \] (1)
Role means a kind of resource. RBT(Role) means the role busy time. It indicates the total busy time of the role. RWT(ROLE) means the role work time. It indicates the total work time of the role. The advantage of this calculation is simple to understand and calculate. However, the algorithm of RWT(Role) in Equation (2) is not accurate, because the actual role is not working all the time. For make up the defect, Haiyang Hu presented an improved calculate method of RWT(Role) as Equation (2) \[ ^7 \].

\[ RWT(Role) = BT(Role) + FT(Role) \] (2)

This method considered the free time of roles, so the result of RWT(Role) will be more accurate. But this method did not consider the different resource of a same role may do the work at the same time, this will lead the error of calculation of free time. In this paper, we will introduce a new evaluation method of resource utilization.

3 An Evaluating Method Integrating Time and Resource

3.1 Introduction of RBTR

The evaluation indicator proposed in this paper is called RBTR(Role Busy Time Rate). It is used to evaluate the resource utilization by using attributes of time and resource in event log. The equation of RBTR shows as Equation (3).

\[ RBTR(Role) = \frac{RBT(Role)}{RWT(Role)} \] (3)

In Equation (3), RBT(Role) means the total service time of the role in all instances of the process. RWT(Role) refers to the total working time of a role. The calculate method of RBT(Role) is shown as Equation (4). The calculate method of RWT(Role) is shown as Equation (5).

\[ RBT(Role) = \sum_{i=1}^{n} BusyTime(ProcInst_i) \] (4)

\[ RWT(Role) = Num(RoleInst) \cdot \sum_{i=1}^{n} RBT(Block_i) \] (5)

In Equation (4), RBT(Role) is the sum of the time the role participates in a process. In Equation (5), Num(RoleInst) is the number of role instance of this role. RBT(Block,) is a continuous busy time. The calculation of RBT(Block,) is shown as Equation (6).

\[ RBT(Block) = (TE_{block} - TS_{block}) \] (6)

TE\(_{block}\) is the end time of this block and the TS\(_{block}\) is the start of the block.

3.2 Design of RBTR Algorithm

The input of RBTR calculation is the event log of process. And the output is the RBTR value of every role. The flow chart of the 0\(^{th}\) layer of the RBTR algorithm is shown in Figure (1).

![Figure 1: The 0\(^{th}\) layer flow chart of the RBTR algorithm](image)

We calculate the RBTR for 5 steps.

1. Read the event log of process. We will use DOM4J to resolve the event log which is in XML format.
2. Parse role information. This step will parse the information of the map of resources and roles from the event log.
(3) Process the process. This step is the core of our algorithm. Only process the one process for one time. The details of the process shown in Figure (2).
(4) Calculate RBTR. The final step, we will use Equation (3)(4)(5) to calculate the RBTR of all roles.

Figure 2: The flow chart of the process “3. Process the Process”

As shown at Figure 2, we process step 3 in flow chart of Figure 1 for 6 steps.
(1) Get Process. Using DOM4J to get a process object.
(2) Parse Event. Parse an event from the process.
(3) Put the Event into Already Start Event List. Put the event into already start event list If this event is not a complete event.
(4) Update Block List. Parse the event to a block and update the block list. The detail of this step is shown in Figure 3.
(5) Delete the Event from Already Start Event List. Delete it after use it.

Figure 3: The flow chart of the process “3.5 Update Block List”

As shown in Figure 3, the process step 3.5 is 5 steps.
(1) Create Block. Using the input event information to create a new block.
(2) Get Block List. Get the block list of this role by using the resource information in this event object.
(3) Find the Position of Block in Block List. Find the position of the new block create at the step 3.5.1 in the block list.
Add the Block into Block List. After we get the position of the new block, we will add the block into the block list.

Delete Covered Block. The new block added into the block list may cover some of the old block in the block list, we need to delete these old block from the block list.

3.3 Performance Analysis of RTBR Algorithm

We will analyse the performance of the RBTR algorithm from the time complexity and spatial complexity.

(1) Time Complexity

Through the analysis of the algorithm code, we can get the time complexity of this method. The results of the following Table 1 obtained by analysing the four processes of the flow chart in Figure 1.

| Process Name                | Time Complexity |
|-----------------------------|-----------------|
| Read Event Log              | O(1)            |
| Process Role Information    | O(R)            |
| Process the Process         | O(E*B)          |
| Calculate RBTR              | O(G)            |

Table 1: The time complexity of processes in RBTR algorithm flow chart.

The time complexity of process “Read Event Log” is O(1). Because this process is a IO operation, and IO operation is considered as a constant operation. The time complexity of process “Process Role Information” is O(R). The “R” means the number of resources. This process will iterator every resource, so that the result is O(R). The time complexity of process “Process the Process” is O(E*B). “E” is the total number of events in this event log, and “B” is the total number of blocks. The time complexity of process “Calculate RBTR” is O(G), and “G” is the number of groups of resource. Because of the number of events is always over 10^3 while the number of resources and resource groups are below 10^2. Then, the time complexity of RBTR algorithm is O(E*B).

(2) Spatial Complexity

As the same way, we can get the spatial complexity of the four processes of the flow chart in Figure 1.

| Process Name                | Spatial Complexity |
|-----------------------------|--------------------|
| Read Event Log              | O(F)               |
| Process Role Information    | O(R)               |
| Process the Process         | O(Max(G, B))       |
| Calculate RBTR              | O(1)               |

Table 2: The spatial complexity of processes in RBTR algorithm flow chart.

The spatial complexity of process “Read Event Log” is O(F). Because this process will use DOM4J to read a XML file and this process will parse the whole XML file to an object. The spatial complexity of process “Process Role Information” is O(R). The “R” means the number of resources. This process need to save all resources in this event log. The spatial complexity of process “Process the Process” is O(Max(G, B)). “G” is the number of resource group, and “B” is the number of Block. This process need to save the RBT of every group, and save all blocks in a block list. Finally, process “Calculate RBTR” do not need any other space, so its spatial complexity is O(1). All operations only need read a part of event log file and parse them to objects. Thus, the spatial complexity of all operations are impossible to over O(F). So, the spatial complexity of RBTR algorithm is O(F).

4 Using RBTR to Analyse a Process

The experimental data is an event log of a telephone company's aftermarket maintenance process. The business process flow is shown in Figure 4. The event log includes 1000 cases, 10827 events in total. A simple example of event log is shown following as Figure 5. The label of “trace” save a process information. The label of “event” save an event information. The attributes of event include resource,
time stamp and other flow information. The data contains group information of the resource. This is an event log which is suitable to analyze its resource utilization by using RBTR.

![Diagram of the business process of “Repair Telephones”](image)

Figure 4: The business process of “Repair Telephones”.

![Example of event log](image)

Figure 5: An example of event log.

After analyzing this event log by using RBTR, we can obtain the results shown as Table 3.

| Department Name | RBTR value |
|-----------------|------------|
| Detection       | 0.3716     |
| Repair          | 0.2036     |
| QA              | 0.3754     |
| System          | 1.0        |

Table 3: The RBTR calculation result of “Repair Telephones” Process.

We can obtain from Table 3, the resource utilization of Detection Dept. and QA Dept. is relatively high, while the Repair Dept. is relatively low. The System has only one resource of computer, so its RBTR is always 1. The higher RBTR means the staff in Repair Dept. has a high degree of redundancy.
After analysing the resource information. It was found that there were seven employees in Repair Dept., while the Detection Dept. and QA Dept. had only three employees for each. And the amount of tasks is equal between Repair Dept. and other two departments. Therefore, we recommend streamlining the Repair Dept. to improve the resource utilization of this process.

5 Conclusion
This paper proposed a process performance evaluation method integrating time and resource attributes. This method is mainly used to evaluate the redundancy of resources in a process, that is, when a role has a task, to find the rate of a person either in a busy or an idle state. This value can be used as an evaluation indicator of resource utilization in process performance. This paper also gives the detailed implementation of the method. The experimental part used the evaluation indicator proposed in this paper to analyse an event log and proposed a suggestion for the streamlining department. This is a practice example of RBTR.

Acknowledgements
This work was supported by The Natural Science Foundation of China (61171060) and The Project of Software Elite Training and Public Service Platform Construction of Chongqing Economic and Information Commission (Yu Economic and Information Investment [2011]167).

References
[1] Van Der Aalst, Wil. "Process mining: Overview and opportunities." ACM Transactions on Management Information Systems (TMIS) 3.2 (2012): 7.
[2] W.M.P. van der Aalst. Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer-Verlag, Berlin, 2011.
[3] Van Beest, N. R. T. P., and Laura Maruster. "A Process Mining Approach to Redesign Business Processes-a case study in gas industry." Symbolic and Numeric Algorithms for Scientific Computing, 2007. SYNASC. International Symposium on. IEEE, 2007.
[4] Rozinat, Anne, et al. "Discovering simulation models." Information systems 34.3 (2009): 305-327.
[5] Van der Aalst, Wil MP, and Boudewijn F. van Dongen. "Discovering workflow performance models from timed logs." Engineering and Deployment of Cooperative Information Systems. Springer Berlin Heidelberg, 2002. 45-63.
[6] Van der Aalst, Wil MP. "Business alignment: using process mining as a tool for delta analysis and conformance testing." Requirements Engineering 10.3 (2005): 198-211.
[7] Haiyang Hu, et al. "Decision support method based on process mining." Computer Integrated Manufacturing System 19.08 (2013): 1755-1770.