A Resource Recommendation Approach based on Co-Working History

Nada Mohammed Abdulhameed, Iman M. A. Helal, Ahmed Awad, Ehab Ezat
Information Systems Department
Faculty of Computers and Information
Cairo University, Egypt

***Abstract*—Recommending the right resource to execute the next activity of a running process instance is of utmost importance for the overall performance of the business process, as well as the resource and for the whole organization. Several approaches have recommended a resource based on the task requirements and the resource capabilities. Moreover, the process execution history and the logs have been used to better recommend a resource based on different human-resource recommender criteria like frequency and speed of execution, etc. These approaches considered the recommendation based on the individual's execution history of the task that will be allocated to the resource. In this paper, a novel approach based on the co-working history of resources has been proposed. This approach considers the resources that had executed the previous tasks in the current running process instances. Then, it recommends a resource that has the best harmony with the rest of the resources.**

***Keywords*—Business process; process instance; co-working history; human-resource recommender criteria; harmony**

I. Introduction

Resource allocation is highly relevant to process mining and its applications. This relevance has been an important issue in business process management (BPM) [1]. Some researchers have discussed ways to optimize allocating the resources in an organization, to improve its business process [2], [3]. They have studied the business process structural features and the way to optimize the available resources to reach the perfect fit for the business needs. Other researchers have described the resource patterns and the correlation between different activities and the available resources [4]. In order to allocate resources, a clear set of rules need to be specified at the beginning of the process lifetime, though this can be challenging. In order to better allocate resources, some researchers have provided different resource patterns, e.g. creation, push, pull, detour [4]. Some of these patterns, such as *push* (from system to worker) and *pull* (from worker to system) patterns, do not rank the process performance [3].

A *resource* is an important indicator of a business process performance. Resources can be machines, manpower, money, software, etc. The process of allocating the human resources can be optimized by analyzing their behavior and mining the event logs to find the rules and the different resource patterns. These resources need to be allocated dynamically to improve the efficiency of the process performance in BPM through a resource recommendation approach.

The main contribution is a resource recommendation approach based on the *co-working history* from the event log. This approach considers the resources executed in the previous tasks at the current running process instances. In order to recommend a resource that has the best harmony with the rest of the resources, the proposed approach considers the *frequency* and the *duration* criteria.

The remainder of this paper is organized as follows: an overview of our approach that briefly discusses most of the background ideas, techniques and tools used to cover this paper in Section II. Section III covers a discussion of previous work. The contribution in resource recommendation based on the co-working history is discussed in Section IV. Implementation details and evaluation are discussed in Section V. Finally, the paper concludes with an outlook for the future work in Section VI .

II. Background

This section starts with some basic concepts about business process management, as well as describing some of the basic definitions used in the resource recommendation approach (the proposed approach). It starts with a brief overview about the business process and its components in section II-A. Then, Section II-B introduces the event log concept as the main input to the proposed approach. Finally in section II-C, the raw performance measure [5] is explained to be used later in extracting the co-working history.

A. Business Process Management

Business processes are used to organize the tasks performed in an organization by different resources [6]. The concept of business process has expanded in the domain of BPM [1], where a business process is represented as a set of activities and tasks performed in an organization or cross-organizations. Each business process serves a set of business goals in an organization or in cross-organizations [7].

BPM has several definitions in the community, one of them states that it is composed of a set of concepts, methods and techniques; each of which support the whole business life cycle (i.e. analysis/design, configuration, runtime, and mining) [1]. These methods and techniques manage the execution of business processes in a Business Process Management System.

Business processes are modeled as a set of activities that transit from activity to another using a control flow. Fig. 1 illustrates a simple example for a travel agency [7], [8], where a customer sends a travel request which will be processed for further actions. The request can be either accepted or rejected by a travel agent, (i.e., a resource in the travel

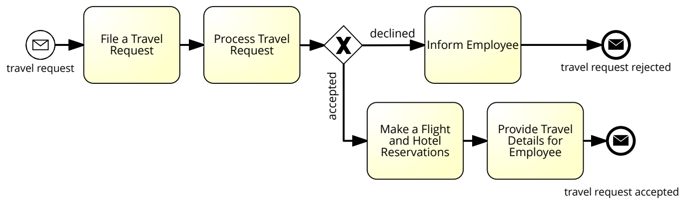

Fig. 1. Travel agency process model [7], [8].

agency). If the request is accepted, the travel agent will reserve both the flight and the hotel for the customer. However, if the travel agent rejected the travel request, the agency will inform the employee about declining the request. This example is graphically presented as a process model using Business Process Model and Notations (BPMN) [**?**].

B. Event Logs

In information systems, an event log contains all saved events that are related to the implemented activities by the specified resources. An event log is composed of a set of events that are correlated to a set of cases. An event is composed of a set of attributes, which includes the activity name (i.e. task), the resource responsible, and the timestamp of event occurrence, etc (cf. Definition 1). The series of registered events in a case is given the term *trace* (cf. Definition 2).

Definition 1 (Event): Let C be the set of all case identifiers, T the set of all task identifiers, R the set of all resource identifiers, S the set of all states, and M the set of all timestamps; So, the event $e \in (C \times T \times R \times S \times M)$ represents an occurrence of a state change in a process instance. We can access properties of an event by the dot notation. $e.case$ refers to the case identifier of e, analogously, $e.task$, $e.resource$, $e.state$, $e.timestamp$.

An event represents an evolution in the execution of a process instance (case), cf. Definition 2. This evolution occurs when one of the task instances (work items) within the case experience has a change of state. For instance, when a work item starts execution, or shows completes, fails, skipped, etc., an event should at least contain information about the case, the task instance, the resource, the type of state change, and the timestamp indicating the time of the event. The resources here refers to human performers involved in the execution of task instances.

Definition 2 (Execution Trace (Case)): An execution trace σ, case, is a sequence of events, $\sigma =<< e_1, e_2, \ldots, e_n >>$, where $e_i, 1 \leq i \leq n$, is an event as per Definition 1. The event can be $e_x < e_y$ if $e_x.timestamp < e_y.timestamp$. If an event e occurs within a trace σ, it is denoted as $e \in \sigma$. Also, the dot notation is used $\sigma.e$ to access event e of the trace. $|\sigma|$ is used to denote the length of the trace. Finally, an event can be accessed by its position in a trace, $\sigma[i],\ where\ 0 \leq i \leq |\sigma|$.

An event log contains different attributes. It is a set of cases each of which contains a set of events (cf. Definition 3). In this paper, an event log contains (task instance, resource, state, and timestamp) attributes. All the event logs provide information about the implementation of a single process by the process model [10].

Definition 3 (Event Log): An event log W is a set of traces. $W = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$, where $\sigma_i, 1 \leq i \leq k$, is a case as per Definition 2.

C. Co-working History

In order to determine the significance of the co-working history of a set of resources in an event log, these resources must have clear measures for their performance. ***Raw performance measure*** (RW) is concerned with deciding different performance measures for each resource in the execution log [5]. It is stated as a tuple for which the measure is calculated. It contains *process model instance*, *case instance*, *task instance* with the *resource* responsible, number of *occurrence* for the task performed by this resource within a *start* and *end* timestamps, and finally the *value* of the performance measure. RW can measure a resource with respect to its effective time, waiting time, service time, etc. In this paper, only the effective time is considered from RW as presented in [5].

Definition 4 (Trace History): For a work item $t \in T$, which has an event e, i.e. $e.task = t$, in a case σ, we define $\sigma_{<e} = \ll e_i | e_i < e \gg$ to be a sub-sequence of σ including all events that occurred before e.

The co-working history is based mainly on the event logs. It is one of the key notations that is defined as a task over an event log W (cf. Definition 5).

Definition 5 (Co-working History): For a work item $t \in T$ and an event log W, we define $W_{<t} \subseteq \mathcal{P}(W) = \{\{\sigma_v | \exists e \in \sigma_v \wedge e.task = t \wedge \sigma_{v<e} is\ a\ trace\ history\ for\ e\}\}$. Moreover, $\forall \sigma_x, \sigma_y \in W_{<t} : \sigma_x[i].task = \sigma_y[i].task \wedge \sigma_x[i].resource = \sigma_y[i].resource,\ where\ 0 \leq i \leq |\sigma_x|$.

Definition 5 finds $t \in T$ the different sets of trace histories that have common tasks and common resources performing them to recommend the resource who will execute the task.

Many studies have identified and classified the main criteria used in resource allocation approaches. These studies aimed at improving tasks performance within the process which are related to the properties of human resources (a taxonomy of resource allocation criteria) [11]. These specified criteria are *Amount, Experience, Expertise, Preference, Previous performance, Role, Social context, Trustworthiness,* and *workload.*

This paper presents a criteria for resources recommendation based on the co-working history, which considers the resources that performed the previous tasks in the current running process instances. This criteria is used to recommend a resource having the best harmony with the rest of the resources. This aspect of co-working history, *frequency* and *duration*, as a *harmony*-based aspect can be inserted among standards of social context, which includes *Collaboration, Compatibility, Influence* and *Social position*. This consideration works for the improvement of tasks performance in the process. Fig. 2 illustrates the proposed criteria and their inclusion in the classifications for resource allocation [11].

III. Related Work

In [4], the authors proposed 43 workflow resource patterns, classified into six categories. These categories cover, among

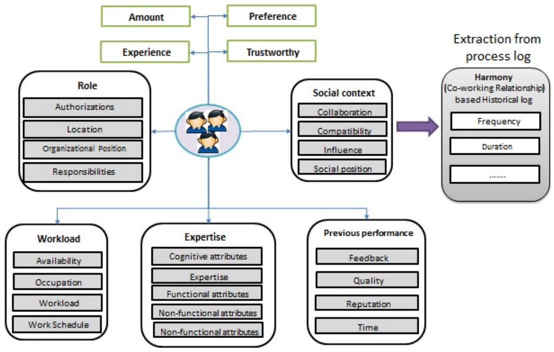

Fig. 2. Taxonomy of resource allocation criteria [11].

other aspects, how the resources can be assigned at the process-design time and how they can be allocated at runtime. Those patterns provide different approaches to identify the eligible resources, as in creation patterns, but they do not provide any means to recommend one candidate over the others. In [12], the authors provide a life-cycle support for the rules of staff assignment based on an organizational model and an event log to discover the allocation rules by learning the decision tree.

In [13], the problem of resource allocation optimization is modeled as Markov decision processes and solved using reinforcement learning. The optimization was fixed to either cost or flow-time based queuing for ordering the queue of work items allocated to each resource. A similar approach is presented in [14]. It is worth noting that both approaches considered role assignment at process *design time* in contrast to resource allocation at *runtime* which this research aims for.

In [15], the authors applied machine learning techniques on the process logs and process models to extract classifiers about the resource who can execute the activity. These classification models are then used to recommend resources for running instances. In [16], the authors used data mining techniques to support and identify resource allocation decisions by extracting information about the process context and process performance from past process executions histories.

In [17], [18], data mining techniques are used to extract resource allocation rules from process logs. The approach recognizes the so-called dependent resource assignment, e.g., if activity a_1 is executed by resource r_1, activity a_2 is executed by resource r_2, then activity a_3 should be executed by resource r_3. However, it is unclear how the approach would deal with other runtime aspects like workload or the unavailability of the resource. The proposed approach in this research recommends the resource based on the co-working history among other aspects, cf. [19]. Then, it adapts to the actual allocation that will take place on an instance level for the recommendation of the next task. Thus, the aim in this research is to introduce flexibility at runtime compared to the rigidness of the extracted association rules.

In [20], the authors have specified the preferences for different resources using expressions based on a Resource Assignment Language (RAL). These preferences are then used at runtime to rank the potential performers of an activity. The concept of providing a list of performers along with their rank is interesting and is of practical relevance. At runtime, it is not helpful to recommend just one resource as he might be engaged in other work or not present. Thus, it is important to provide several alternatives to do not block the process instance waiting for a free resource. In [19], a framework for recommending resource allocation based on process mining is defined. It introduces six dimensions to compare between potential resources and the user who can change the weight of each dimension to control the final recommendation. This approach can be seen as an extension of the work in [20]. Compared to what this research aims to achieve, this proposed approach targets going beyond the one-to-one relation between a task to allocate for a resource by studying the $n-ary$ relationships between groups of tasks and their respective potential performers.

Cooperation correlation among pairs of resources have been introduced and measured in [21]. Compared to the proposed approach perspective, the cooperation is applied only on pairs of resources whereas $n-ary$ sets of resources are considered based on the completed tasks within a case. Moreover, the authors have identified resource recommendation as a use case for calculated measures. However, for that specific use case they do not consider cooperation correlation as a criterion for resource recommendation. Rather, they consider resource preferences and competency. A similar approach about resource cooperation is presented in [22].

In [23], the authors have provided the resource allocation method under constraints of preference, availability and the total cost constraints. Then they analyzed the influence of collaboration between resources on process performances. In [24], the authors introduced a method to compute the social relation between two resources; then, they computed the influence of the previous resources on the candidate resources by using a Q-learning algorithm for dynamic task allocation. In [25], the authors presented a model which measures the compatibility among resources when assigning work items to the collaborative groups by using compatibility matrix. They have also developed an allocation algorithm to maximize team cooperation, the needs for inquiring the effect of cooperation on throughput and other process results.

IV. A Resource Recommendation Approach

This section presents the proposed approach for resources recommendation based on co-working history by specifying and applying frequency and duration criteria.

A. Recommendation Criteria based on Co-working History

As in Section II, the proposed approach is based on the co-working history for resource recommendation. It determines the criteria and metrics from the event log and uses them as a new dimension for resource recommendation which has been termed as *co-working* history. These criteria are as follows:

- Frequency Criterion (FC): It recommends the appropriate resource to perform the target task based on the number of times in which the resource works with the previous resources in the same cases at the event log. It is suitable to recommend a resource that works more times with the previous resources in the same cases for the event log to perform the target task; i.e.

the number of times during which the resources work with each other in the same cases in the event log.

- Duration Criterion (DC): It recommends the appropriate resource with less average time to perform the target task based on the previous resources. It is suitable to recommend a resource to perform the target task based on previous resources that has less average time in the execution of the tasks.

Based on these criteria, a resource recommendation approach is needed to find the most appropriate resources to work with previous resources based on co-working history.

B. Calculating Co-working History

The resource allocation based on co-working history approach mainly includes two parts:

Part 1, The *preprocessing steps*: The raw performance measures (RW) are generated from the implementation of the approach presented in [5]. Both of the event log and the raw performance measures (RW) are inputs to a set of preprocessing steps to obtain the co-working relationships. After extracting the event log and RW from [5], data preprocessing has been conducted as follows: (1) filter out cases in which sum METRIC-VALUE are less than 0, and cases that contain less than three activities, (2) choose the effective time as a measure for the proposed approach, (3) find the latest resource who performed each activity in each case within the event log and RW, and finally (4) detect whether the resource executes the same activity more than once in the same case. In the last step, the average is calculated as the effective time for the activity. As an output, the event log is ready to be used as an input to the proposed approach. Fig. 3 illustrates how to obtain and calculate *co-working* relationships.

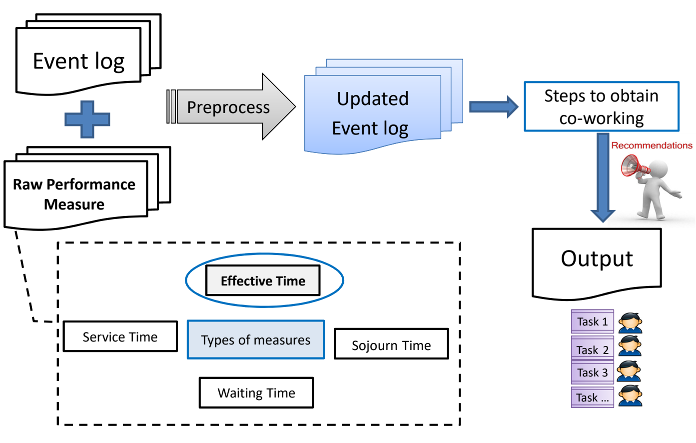

Fig. 3. An overview of the proposed approach.

Part 2, *Recommending a resource*: There are two steps to recommend a suitable resource for working with the previous resources for each activity based on the co-working history. They are as follows: (1) divide the event log into training and testing sets, (2) recommend resources for each activity according to the proposed criteria (*Frequency* and *Duration*).

V. Evaluation

The evaluation of the proposed approach is applied on two logs:

1) Synthesized Log: For the evaluation, a synthesized Log that had been generated from the ProM [26] plugin "Perform a simple simulation of (stochastic) Petri net" [27] was used. This log was taken from [5]. It contains 100 cases with a total of 4677 events, 10 activities, and 9 resources. For further references clarification in this paper, this log is referred to as $W1$.
2) Real Log: The approach has been applied on a real log from the Business Process Intelligence (BPI) Challenges. The log was taken from a Dutch Financial institute (referred to as $W2$), and it contains data that that represent the process of personal loans applications [1]. the log contains 13087 cases with a total of 262200 events, 25 activities and 69 resources.

The proposed approach has been implemented using Java and a relational database. And it has been tested on Windows 8 with 4G RAM and a Core i5 processor.

A. Co-working Effect on Process Performance

The aim of this section is to statistically prove the significance of the co-working relationships (team harmony) on resource recommendation and process performance. Definition 5: The co-working history works on finding, for a given task, $t \in T$, the different sets of trace histories that have common tasks and common resources performing them. Suppose that $W = \{\sigma_1 = \ll e_1(t_1, r_1, complete, tm_1), e_2(t_2, r_2, complete, tm_2), e_3(t_3, r_3, complete, tm_3), \cdots \gg, \sigma_2 = \ll e_{10}(t_1, r_1, complete, tm_{10}), e_{11}(t_2, r_2, complete, tm_{11}), e_{12}(t_3, r_3, complete, tm_{12}), \cdots \gg, \sigma_3 = \ll e_{100}(t_1, r_7, complete, tm_{100}), e_{101}(t_2, r_5, complete, tm_{101}), e_{102}(t_3, r_3, complete, tm_{102}), \cdots \gg\}$. If we consider task t_3, then the resulting co-working history will be $W_{<t_3} = \{\{\ll e_1(t_1, r_1, complete, tm_1), e_2(t_2, r_2, complete, tm_2) \gg, \ll e_{10}(t_1, r_1, complete, tm_{10}), e_{11}(t_2, r_2, complete, tm_{11}) \gg\}, \{\ll e_{100}(t_1, r_7, complete, tm_{100}), e_{101}(t_2, r_5, complete, tm_{101}) \gg\}\}$.

Here, traces σ_1 and σ_2 are grouped together because they have a common trace history for task t_3, the same tasks executed before t_3 with the same human resources. Trace σ_3 is in another set because it deviates from the other two traces with respect to the human resources.

In order to check whether the co-working history affects human resource performance of the task-in-hand, a statistical test was formulated where the statistical significance of such hypothesis was tested. The null hypothesis H_0 is that the harmony (common co-working history) is ineffective and has no influence on the performance of human resources. The alternative hypothesis states otherwise. That is, the co-working history has an influence on human performance. To test the hypothesis, a paired T-Test using unequal variance was applied. For this test, there is need to prepare two sets. The first set contains the time taken by the different human resources who executed the target task t with all cases (traces) in which t was executed. The second set contains human resources who executed t but within cases that have common co-working history.

[1] https://tinyurl.com/FinacialLog

To explain how the testing works to prove the hypothesis assumption, a set of steps on the event log has been applied: (1) Obtain the process model for the event log, (2) Select cases or traces that contain at least three activities and more, (3) Select the target activity; the chosen target activity is not the first one in the trace but must be preceded by a number of activities to have a co-working history, (4) Find all the resources who execute the target activity, (5) Find the effective time for all resources who perform the target activity for each case in the event log (situation 1), (6) Identify specific resources for the target activity, and finding all possibilities for co-working history to all activities that precede the target activity, c.f. Definition 5, (7) Compile similar groups in co-working history of all activities that precede the target activity (situation 2), (8) Run the paired T-Test with un-equal variance.

In this paper, the approach in [5] has been used to extract the event log and the performance indicators. To give an example about how the test works, trace from event log for process model was chosen as shown in Fig. 4. Table I illustrates a sample from the event log.

TABLE I. A SAMPLE EVENT LOG WITH SEQUENCES W-C,W-N,W-V

EVENTID	CASEID	RESOURCES	ACTIVITY	Effective-Time
1	173688	Dummy	W-C	8
2	173688	11049	W-N	0
3	173688	10629	W-V	32
4	173844	11201	W-C	6
5	173844	11049	W-N	0
6	173844	10629	W-V	15
.	.	.	.	.
511	173691	Dummy	W-C	6
512	173691	11049	W-N	1
513	173691	10809	W-V	7.33333
514	173913	Dummy	W-C	16
515	173913	10899	W-N	0
516	173913	10809	W-V	19.5
517	174511	10909	W-C	5
518	174511	11259	W-N	0
519	174511	10809	W-V	17
.	.	.	.	.
1468	173715	10912	W-C	11
1469	173715	10899	W-N	0
1470	173715	10138	W-V	4
1471	173751	Dummy	W-C	11
1472	173751	10899	W-N	0
1473	173751	10138	W-V	10
.	.	.	.	.

After choosing trace, the target activity $W-V$ was selected from the trace shown in Table I. Then, all the resources that perform this activity along with their effective time (18 resources) were provided. This step is referred to as (Situation 1). Table II shows all resources and their effective time in performing the target activity in all cases of the log. Then, $W_{<W-V}$,cf. Definition 5 was constructed, which contains the sets of activities that precede the target activity to get the co-working history of the target activity for each resource. This step is referred to as (Situation 2). Table III shows the special groups for each resource in each case according to the co-working history.

The paired T-Test used contains several tests for data analysis. Two tests were chosen. These two tests are Equal-Variance-Test and Unequal-Variance T-Test. The focus was on Unequal-Variance T-Test to prove the assumption, as it is the most common type of T-tests and the most used tests that cover large part in statistical test or hypothetical tests.

TABLE II. SAMPLE SITUATION 1 FROM TABLE I AND ACTIVITY W-V

EVENTID	CASEID	RESOURCES	ACTIVITY	Effective-Time
516	173688	10629	W-V	32
517	173844	10629	W-V	15
518	174105	10629	W-V	7
519	174141	10629	W-V	23
.	.	.	.	.
362	173694	10609	W-V	40
363	173790	10609	W-V	17
364	173817	10609	W-V	14
365	173868	10609	W-V	24
366	174009	10609	W-V	20
.	.	.	.	.
686	173691	10809	W-V	7.33333
687	173913	10809	W-V	19.5
688	174511	10809	W-V	17
689	174707	10809	W-V	16
.	.	.	.	.

TABLE III. SAMPLE SITUATION-2 FROM TABLE I AND ACTIVITY W-V

CASEID	RESOURCE T1	RESOURCE T2	RESOURCE Target	ACTIVITY Target	Effective Time
173688	Dummy	11049	10629	W-V	32
175798	Dummy	11049	10629	W-V	29
176000	Dummy	11049	10629	W-V	6
177317	Dummy	11049	10629	W-V	35
182101	Dummy	11049	10629	W-V	32
198107	Dummy	11049	10629	W-V	10.5
203648	Dummy	11049	10629	W-V	25
203702	Dummy	11049	10629	W-V	32
.	.	.	.	.	.
173844	11201	11049	10629	W-V	15
179456	11201	11049	10629	W-V	26
183910	11201	11049	10629	W-V	11
.	.	.	.	.	.
180187	11169	11259	10809	W-V	14
185771	11169	11259	10809	W-V	12.5
188317	11169	11259	10809	W-V	19.5
196228	11169	11259	10809	W-V	2.66667
201710	11169	11259	10809	W-V	6
205803	11169	11259	10809	W-V	11.25
.	.	.	.	.	.

To give an example, in situation 1, without co-working history for the resource 10629 the effective time according to the target activity $W.V$ in all cases is 32 min in case 173688, 15 min in case 173844, etc., cf. Table II. In situation 2, with co-working history for resources $Dummy, 11049\ and\ 10629$, the effective time in this group for resource 10629 is $32, 29, 6, 35, 32, 10.5, 25, 32$ min in all cases respectively. For the other group in situation 2, The effective time for the resource 10629 in co-working history with $11201\ and\ 11049$ is $15, 26, 11$; cf. Table III. Each group from situation 2 will be tested against situation 1 individually. Note that the size of situation 1 is not equal to the size of situation 2. The size of situation 1 is larger than the size of situation 2.

B. Process Performance Results

The results stated that there is a certain percentage of each resource confirming the assumption of this research which says that "the harmony among resources with co-working history has an influence on the human resource performance". The percentage, which confirmed the assumption for each resource that has performed the target activity, is calculated using the following equation:

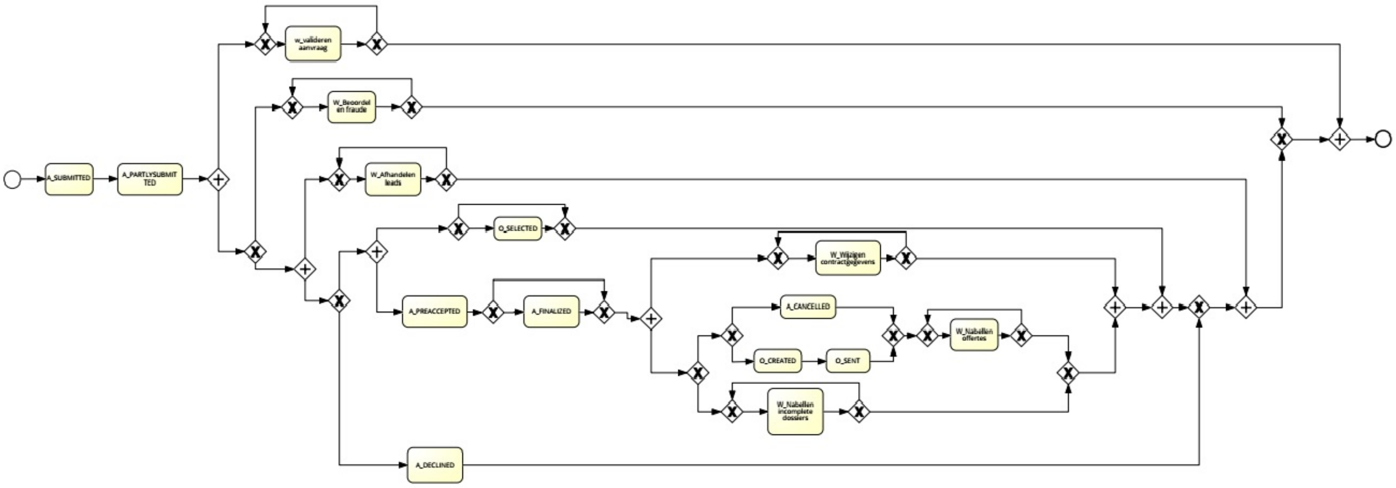

Fig. 4. Personal loans business process.

Co-working-Hypothesis =

$$\frac{\sum_{i=1}^{n} \text{ Cases When Reject } H_0\text{=Yes}}{\text{Count of Cases for all groups}}, \ \forall\ R \qquad (1)$$

n is the number of cases where H_0 is rejected and the count of cases for all groups is the summation of the cases for each resource with co-working history (situation 2). And the confidence level (CL) was 95 % when was the default value α = 0.05. Table IV shows the results of the statistical tests for real log and the percentages obtained by each resource in the provided example. These results prove that the test result for all the groups have proven the hypothesis for each resource.

TABLE IV. THE RESULTS OF THE STATISTICAL TESTS FOR REAL LOG BY UNEQUAL VARIANCE T-TEST

#	RESOURCES	Count of Group(cases)	Count Group Reject H_0 =Yes(cases)	Significant Different(%)
1	10629	80 (170 case)	45 (70 Casa)	0.41
2	10809	82 (165 case)	50 (58 case)	0.35
3	10609	81 (154 case)	51 (57 case)	0.37
4	10138	120 (361 case)	49 (63 case)	0.17
.	.	. .	.	.

As an example, when resource 10629 performed the target activity $W-V$, 80 groups with 170 cases which have common co-working history were formed. There are 45 groups among the 80 groups that confirmed the hypothesis (i.e., count of groups where H_0 was rejected is 45 groups out of the 80 groups). The number of cases in the 45 groups which confirmed the hypothesis is 70 cases. When (1) was applied on resources, the result of resource 10629, was $70/170 = 41\%$, 35% for resource 10809, 37% for resource 10609, etc.

C. Implementation and Experimental Evaluation

In this section, the proposed approach is described and applied along with details in order to verify the influence and the effectiveness of the harmony among resources. In order to obtain co-working relationships, the event log is preprocessed and split into training and testing sets (80% for training set, 20% for test set). In this part, the real-life event log (i.e. $W2$) was used to test the proposed approach. The approach was implemented in [5] which calculates the RW out of the input event log (Table VI). Table V shows a sample of the event log and Table VI shows a sample of the raw performance measures for cases 205715 and 205721 as an example.

TABLE V. A SAMPLE OF THE EVENT LOG

EVENTID	CASEID	RESOURCES	ACTIVITY	EVENT TYPE	TIMESTAMP
.	.	.	.	.	.
209686	205715	112	A-SUBMITTED	complete	2/1/2012 20:04
209687	205715	112	A-PARTLYSUBMITTED	complete	2/1/2012 20:04
209688	205715	112	W-Afhandelen leads	allocate	2/1/2012 20:04
209689	205715	10933	W-Afhandelen leads	start	2/1/2012 20:06
209690	205715	10933	A-PREACCEPTED	complete	2/1/2012 20:10
209691	205715	10933	W-Completeren aanvraag	allocate	2/1/2012 20:10
209692	205715	10933	W-Afhandelen leads	complete	2/1/2012 20:10
209693	205715	10933	W-Completeren aanvraag	start	2/1/2012 20:10
209694	205715	10933	A-ACCEPTED	complete	2/1/2012 20:17
.	.	.	.	.	.
209737	205721	10933	W-Completeren aanvraag	start	2/1/2012 20:26
209738	205721	10933	W-Completeren aanvraag	complete	2/1/2012 20:26
209739	205721	11119	W-Completeren aanvraag	start	2/1/2012 20:27
209740	205721	11119	W-Completeren aanvraag	complete	2/1/2012 20:27
.	.	.	.	.	.

The proposed approach needs preprocessing for the current event log in Table V. The event log is filtered using Table VI to remove all cases where the sum metric value for all resources is less than or equal to zero, and using Table V to remove all cases that contain less than three activities. Hence, only the cases that contain three or more activities are considered. Moreover, effective time is used as the main performance metric which is one of the measures extracted from [5].

After filtering both the event log and the raw performance measure (RW) tables, the event log was scanned to find the latest resource who performed each activity in each case within the event log. Then, these resources are linked with the performance measures when the event type = $complete$. For example, in Table V, the resource 10933 has *allocated* activity "$W - Completeren\ aanvraag$" for case 205721 at time tc = $2 - 1 - 201220 : 26$, and the resource 11119 has *allocated*

TABLE VI. A SAMPLE OF RAW PERFORMANCE MEASURE(RW)

CASEID	ACTIVITY	RESOURCES	OCCURRENCE	MEASURE TYPE	METRIC VALUE
.	.	.	.	.	.
205715	W-Afhandelen leads	10933	1	Effective	3
205715	W-Completeren aanvraag	10933	1	Effective	8
205715	W-Nabellen offertes	10933	1	Effective	1
205715	W-Nabellen offertes	11179	2	Effective	2
205715	W-Nabellen offertes	11181	3	Effective	0
205715	W-Nabellen offertes	10629	4	Effective	0
205715	W-Valideren aanvraag	10629	1	Effective	29
205715	W-Nabellen incomplete dossiers	10982	1	Effective	4
205715	W-Nabellen incomplete dossiers	11003	2	Effective	20
205715	W-Nabellen incomplete dossiers	11003	3	Effective	0
205715	W-Nabellen incomplete dossiers	11169	4	Effective	4
205715	W-Nabellen incomplete dossiers	10889	5	Effective	3
.	.	.	.	.	.
205721	W-Completeren aanvraag	10933	1	Effective	0
205721	W-Completeren aanvraag	11119	2	Effective	0
205721	W-Completeren aanvraag	11119	3	Effective	19
205721	W-Nabellen offertes	11119	1	Effective	2
205721	W-Nabellen offertes	11119	2	Effective	0
205721	W-Nabellen offertes	Dummy	3	Effective	1
205721	W-Nabellen offertes	11259	4	Effective	0
205721	W-Valideren aanvraag	10629	1	Effective	15
.	.	.	.	.	.

activity “$W - Completeren\ aanvraag$” for case 205721 at time $tc = 2 - 1 - 201220 : 27$, in this scenario, the resource 11119 is chosen as the latest one. This strategy is applied to all cases in the event log. Then, these recent resources are connected with the performance measures from Table VI.

TABLE VII. A SAMPLE OF THE HISTORY FOR EVENT LOG EXTRACTED FROM TABLE V

EVENTID	CASEID	RESOURCES	ACTIVITY	EVENT TYPE	TIMESTAMP
.	.	.	.	.	.
209686	205715	112	A-SUBMITTED	complete	2/1/2012 20:04
209687	205715	112	A-PARTLYSUBMITTED	complete	2/1/2012 20:04
209690	205715	10933	A-PREACCEPTED	complete	2/1/2012 20:10
209692	205715	10933	W-Afhandelen leads	complete	2/1/2012 20:10
209694	205715	10933	A-ACCEPTED	complete	2/1/2012 20:17
209695	205715	10933	A-FINALIZED	complete	2/1/2012 20:19
209696	205715	10933	O-SELECTED	complete	2/1/2012 20:19
209697	205715	10933	O-CREATED	complete	2/1/2012 20:19
209698	205715	10933	O-SENT	complete	2/1/2012 20:19
209700	205715	10933	W-Completeren aanvraag	complete	2/1/2012 20:19
209708	205715	10629	O-SENT BACK	complete	2/16/2012 15:36
209710	205715	10629	W-Nabellen offertes	complete	2/16/2012 15:36
209713	205715	10629	W-Valideren aanvraag	complete	2/16/2012 16:10
209723	205715	10889	O-DECLINED	complete	2/18/2012 13:26
209724	205715	10889	A-DECLINED	complete	2/18/2012 13:26
.	.	.	.	.	.
209743	205721	11119	A-FINALIZED	complete	2/1/2012 20:47
209748	205721	11119	W-Completeren aanvraag	complete	2/1/2012 20:47
209754	205721	11202	O-SELECTED	complete	2/6/2012 12:27
209755	205721	11202	O-CANCELLED	complete	2/6/2012 12:27
209756	205721	11202	O-CREATED	complete	2/6/2012 12:27
209757	205721	11202	O-SENT	complete	2/6/2012 12:27
.	.	.	.	.	.

Next, if any activity which is executed by the same resource more than once is found in the same case, the average is calculated as the effective time for the activity. For example, in Table VIII, activity “$W - Completeren\ aanvraag$” is executed by the resource 11119 more than once in the case 205721, and the effective time for the activity “$W - Completeren\ aanvraag$” by the resource 11119 is $(0, 19)$ respectively. The average time is $(0 + 19/2 = 9.5)$. The same is applied for all the cases in the event log. Finally, Table IX illustrates the final result of the preprocessing steps.

TABLE VIII. A SAMPLE OF THE RESULT OF JOINING TABLES VI AND VII

EVENTID	CASEID	RESOURCES	ACTIVITY	EVENT TYPE	MEASURE	METRIC VALUE
.	.	.	.	.	.	.
15591	205715	10933	W-Afhandelen leads	complete	Effective	3
15592	205715	10933	W-Completeren aanvraag	complete	Effective	8
15593	205715	10629	W-Nabellen offertes	complete	Effective	0
15594	205715	10629	W-Valideren aanvraag	complete	Effective	29
15595	205715	10889	W-Nabellen incomplete dossiers	complete	Effective	3
15596	205721	11119	W-Completeren aanvraag	complete	Effective	0
15597	205721	11119	W-Completeren aanvraag	complete	Effective	19
15598	205721	11259	W-Nabellen offertes	complete	Effective	0
15599	205721	10629	W-Valideren aanvraag	complete	Effective	15
.	.	.	.	.	.	.

After the preprocessing steps, the new event log is used as input for the proposed approach. This event log contains information about 3718 cases, 13704 events, 58 resources and 9 activities. The attributes for each case include EVENTID, CASEID, RESOURCE, ACTIVITY, and METRIC VALUE (Effective Time), cf. Table IX. This table is used to calculate the *co-working* relationships based on applying (Frequency and Duration Criteria) for recommending the resources based on co-working history. This co-working history verifies the influence of the harmony among resources on the performance of resources, where a significant difference has emerged.

TABLE IX. SAMPLE OF THE FINAL EVENT LOG FOR OUR APPROACH

EVENTID	CASEID	RESOURCE	ACTIVITY	METRIC-VALUE
.	.	.	.	.
10966	205715	10933	W-Afhandelen leads	3
10967	205715	10933	W-Completeren aanvraag	8
10968	205715	10629	W-Nabellen offertes	0
10969	205715	10629	W-Valideren aanvraag	29
10970	205715	10889	W-Nabellen incomplete dossiers	3
.	.	.	.	.
10971	205721	11119	W-Completeren aanvraag	9.5
10972	205721	11259	W-Nabellen offertes	0
10973	205721	10629	W-Valideren aanvraag	15
.	.	.	.	.

The event log data (cf. Table IX) is split into training and testing sets to obtain ***co-working*** relationships. The *training* set is used to extract the co-working relationships using SQL queries, which generate a *co-working relationship* table. This table is used to recommend the resource based on both (Frequency and Duration Criteria) after applying some SQL queries. On the other hand, the *test* set is used to compare the results before and after applying the proposed approach.

Table X presents some comparative examples before and after implementing the approach. It compares the original log (i.e., test set) and the output of the proposed resource rec-

TABLE X. SOME COMPARATIVE EXAMPLES BEFORE AND AFTER IMPLEMENTING OF OUR APPROACH

		original log		Co-working Relationships			
				Frequency Criterion		Duration Criterion	
CASEID	ACTIVITY	RESOURCES	METRIC VALUE	RESOURCES	METRIC VALUE	RESOURCES	METRIC VALUE
205733	W-C	10932	20.00	10932	9.80	10932	9.80
205733	W-N	10789	0.00	11259	0.00	10138	0.00
205733	W-V	10138	8.00	10138	14.86	10629	8.84
.	.	.	.	.	.	.	.
205745	W-Afhandelen	11169	2.00	11169	4.53	11169	4.53
205745	W-C	11119	19.00	11189	14.00	11203	8.03
205745	W-N	10629	0.00	11259	0.28	10899	0.00
205745	W-V	10629	20.00	10138	13.98	10138	13.50
.	.	.	.	.	.	.	.
205766	W-C	11201	11.00	11201	9.07	11201	9.07
205766	W-N	11259	0.00	11049	0.67	10609	0.00
205766	W-V	11289	50.50	10138	15.99	10629	8.05
205766	W-N incomplete dossiers	11289	0.00	10899	0.27	10899	0.15
.	.	.	.	.	.	.	.

ommendation approach after applying frequency and duration criteria. For example, there are cases where each case records the resource which performs the task. In the case 205733 of the original log, resource10932 executes task $W-C$, resource 10789 executes task $W-N$, resource 10138 executes task $W-V$, and so on. Each resource has an effective time for its corresponding activity $(20.00, 0.00, 8.00$ min), respectively.

According to *frequency* criterion for resources recommendation, when the resource 10932 executes task $W-C$, the appropriate resource to execute task $W-N$ is 11259 with average time (0.00 minute). Hence, when the resource 10932 executes task $W-C$ and the resource 11259 executes task $W-N$, then the appropriate resource to execute task $W-V$ is 10138 with average time (14.86 min). While according to *duration* criterion, different resource recommendations are as follows: when the resource 10932 executes task $W-C$, the appropriate resource to execute task $W-N$ is 10138 with average time (0.00 min). Moreover, when the resource 10932 executes task $W-C$ and the resource 10138 executes task $W-N$, the appropriate resource to execute task $W-V$ is 10629 with average time (8.84 min).

Another example, in the case 205745 of the original log, the resource 11169 executes task $W-Afhandelen$, the resource 11119 executes task $W-C$, the resource 10629 executes task $W-N$, and the resource 10629 executes task $W-V$. Each resource takes an effective time for an activity $(2.00, 19.00, 0.00, 20.00$ min) respectively. Table X shows the different variations on both frequency and duration criteria after applying the proposed approach.

D. Evaluation Results

The evaluation of the results is based on synthesized and real life logs. In order to investigate whether the proposed approach contributes to get better results and improve the performance of tasks, (2) was used to calculate the overall result for applying the approach for the duration criterion.

$$\text{Overall}=\Sigma_{i=1}^{n}\text{ETB approach} - \text{ETA approach} \quad (2)$$

where n is the number of test case, and the overall represents sum of the total difference between before and after the proposed approach application according to the criterion of duration. The Effective Time Before (ETB) applying the approach represents the effective times of activities that resources have performed in the original log. While, the Effective Time After (ETA) applying the approach represents the effective times of activities that resources have performed after applying the criteria. Table XI summarizes the results of applying (2) on synthesized and real life logs. It shows resources recommendation based on the average time, the minimum time, and the maximum time to execute each activity in each case over all the log.

TABLE XI. OVERALL CO-WORKING RELATIONSHIPS

	Co-working Relationships (Duration Criterion)		
	Overall		
Logs	Avg	Min	Max
W_1	23279.16	62893.99	25998.2
W_2	10496.77366	6123.24469	-2091.40666

The total of the effective time after and before applying the proposed recommendation approach is computed using the following equation:

$$\text{Total Effective Time}=\Sigma_{i=1}^{n}\text{Effective time(A} \setminus \text{B)} \quad (3)$$

where n is the number of test cases, and the effective time for each test case (after(A) and before(B)) the recommendation is the summation of the effective time (after and before) applying the proposed recommendation approach.

In (4), the average of the effective time for test set (20%) before applying the proposed recommendation approach was computed.

$$\textbf{Avg}_{BR} = \frac{\sum_{i=1}^{n}\text{Effective time (BR)}}{n} \quad (4)$$

where n is the number of test case, and the effective time for each test case before the recommendation (BR) is the summation of the effective time before applying the recommendation approach.

In (5), the average of the effective time for test set (20%) after applying the recommendation approach is computed.

$$\textbf{Avg}_{AR} = \frac{\sum_{i=1}^{n}\text{Effective time(AR)}}{n} \quad (5)$$

TABLE XII. RESULTS OF APPLYING THE PROPOSED APPROACH ON REAL AND SYNTHESIZED EVENT LOGS

Logs	Co-working Relationships			
	Frequency Criterion		Duration Criterion	
	W_1	W_2	W_1	W_2
Sum Effective Time (min)	187384.85	17555.064	159901.054	12804.2593
Sum Effective Time (min) original	183180	22488.88615	183180	22488.88615
$\mathbf{Avg}_{BR}$	9641.052632	33.02332767	9641.052632	33.02332767
$\mathbf{Avg}_{AR}$	9862.360526	25.778361	8415.844947	18.8021429
Improvement Rate	-11.64778393 (min)	0.2193894 (min)	0.127082356 (min)	0.43064057 (min)

where n is the number of test case, and the effective time for each test case after the recommendation (AR) is the summation of the effective time after applying the proposed recommendation approach.

Table XII shows the results of applying the proposed approach on real and synthesized event logs. It uses (3), (4) and (5) on test set (20% of the event log). For the order fulfillment log ($W1$), the total of the effective time after applying the approach based on the criteria (Frequency, Duration) is 187384.85 min, 159901.054 min, respectively. The total effective time of the original log before applying the approach is 9641.052632 min. The $\boldsymbol{Avg}_{AR}$ after applying the approach recommendation based on the criteria (Frequency, Duration) is 9862.360526 min, 8415.844947 min respectively. On the other hand, $\boldsymbol{Avg}_{BR}$ of original log before applying the recommendation approach is 9641.052632 min.

For the Financial log ($W2$), the total of the effective time after applying the recommendation approach based on the criteria (Frequency, Duration) is 17555.064 min, 12804.2593 min respectively. The total effective time of original log before applying the approach is 22488.88615 min. The $\boldsymbol{Avg}_{AR}$ after applying the proposed recommendation approach based on the criteria (Frequency, Duration) is 25.778361 min, 18.8021429 min, respectively. On the other hand, $\boldsymbol{Avg}_{BR}$ of original log before applying the recommendation approach is 33.02332767 min.

The improvement rate of the proposed approach was calculated and evaluated by using the following equation:

$$\text{Improvement Rate} = (\mathbf{Avg}_{BR} - \mathbf{Avg}_{AR})/\mathbf{Avg}_{BR} \qquad (6)$$

The results of the proposed approach have an improvement of the real data set and synthesized data set. The results show that the time is minimized to 0.2350476 min with frequency criterion and 0.43064057 min with duration criterion of the real data set. For synthesized logs, the results show that the time is minimized to 0.127082356 min with duration criterion, while the results state that the time is maximized to -11.64778393 min with frequency criterion. The negative value implies that the resources recommendation approach gives bad results.

The real data set has a bigger improvement because it contains a greater number of cases, activities and resources. In other words, considering co-working history for task allocation and resource recommendation is efficient. It also reduces process execution time significantly by taking resource harmony into account. Note that, more satisfactory results can be obtained as the number of process instances increases. The reason is that more event logs can generate more accurate harmony measurement which in turn provides more effective allocation recommendation.

VI. CONCLUSION AND FUTURE WORK

This paper has proposed a resource recommendation approach. This approach is built upon the co-working history from an event log. It considers the resources that had performed the previous tasks in the current running process instances, in order to recommend a resource that has the best harmony with the rest of the resources. This proposed approach focuses on the organizational perspective. It depends on a procedure-approach to extract time-related key performance indicators from process execution logs. This procedure-approach supports four measures: effective, service, waiting and sojourn time. The effective time measured was used in the proposed approach.

The proposed approach works to determine the criteria and the metrics from event log for resource recommendation. These criteria are (frequency and duration) based on the *co-working history*. The approach has been implemented and tested on both real and synthesized logs. The results show that it is possible to obtain the appropriate resources recommendation based on the criteria of co-working history. This approach has contributed to reducing the tasks time and to improving both the process and the resources performance.

As a future work, the researcher aims to add the co-working history approach as a new dimension and extend the related approaches for the resource recommendation with other algorithms.

REFERENCES

[1] M. Weske, *Business Process Management - Concepts, Languages, Architectures, 2nd Edition*. Springer, 2012.

[2] J. Xu, C. Liu, and X. Zhao, "Resource allocation vs. business process improvement: How they impact on each other," *Lecture Notes in Computer Science*, vol. 5240 LNCS, pp. 228–243, 2008.

[3] W. Zhao and X. Zhao, *Process Mining from the Organizational Perspective*, pp. 701–708. Springer, 2014.

[4] N. Russell, W. M. P. van der Aalst, A. H. M. ter Hofstede, and D. Edmond, "Workflow Resource Patterns: Identification, Representation and Tool Support," in *17th International Conference, CAiSE 2005, Porto, Portugal, June 13-17, 2005. Proceedings* (O. Pastor and J. and Falcão e Cunha, eds.), vol. 3520, pp. 216–232, Springer, 2005.

[5] N. M. Zaki, A. Awad, and E. Ezat, "Extracting accurate performance indicators from execution logs using process models," in *AICCSA*, pp. 1–8, IEEE, 2015.

[6] Y.-C. Chen, "Business Process Reengineering," *Encyclopedia of library and information science*, vol. 67, no. 1, p. 23, 2001.

[7] I. M. A. Helal, *Towards Monitoring Compliance of Business Processes*. PhD thesis, Cairo University, Giza, 2017.

[8] A. Awad, E. Pascalau, and M. Weske, "Towards Instant Monitoring of Business Process Compliance," *EMISA Forum*, vol. 30, no. 2, pp. 10 – 24, 2010.

[9] OMG, "Business Process Model and Notation (BPMN), Version 2.0," tech. rep., Object Management Group, January 2011.

[10] W. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer, 1st ed., 2011.

[11] M. Arias, J. Munoz-Gama, and M. Sepúlveda, "Introducing a taxonomy of human resource allocation criteria," tech. rep., Pontificia Universidad Católica, Santiago, Chile, 2017.

[12] S. Rinderle-Ma and M. Wil, "Life-cycle support for staff assignment rules in process-aware information systems," tech. rep., TU Eindhoven, 2007.

[13] Z. Huang, W. M. P. van der Aalst, X. Lu, and H. Duan, "Reinforcement learning based resource allocation in business process management," *Data Knowl. Eng.*, vol. 70, no. 1, pp. 127–145, 2011.

[14] A. Koschmider, Y. Liu, and T. Schuster, "Role assignment in business process models," in *BPM Workshops*, vol. 99 of *LNBIP*, pp. 37–49, Springer, 2011.

[15] Y. Liu, J. Wang, Y. Yang, and J. Sun, "A semi-automatic approach for workflow staff assignment," *Computers in Industry*, vol. 59, no. 5, pp. 463–476, 2008.

[16] R. Sindhgatta, A. Ghose, and H. K. Dam, "Context-aware analysis of past process executions to aid resource allocation decisions," in *Int'l Conf. on Adv. Info. Syst. Eng.*, pp. 575–589, Springer, 2016.

[17] Z. Huang, X. Lu, and H. Duan, "Mining association rules to support resource allocation in business process management," *Expert Syst. Appl.*, vol. 38, no. 8, pp. 9483–9490, 2011.

[18] T. Liu, Y. Cheng, and Z. Ni, "Mining event logs to support workflow resource allocation," *Know.-Based Syst.*, vol. 35, pp. 320–331, Nov. 2012.

[19] M. Arias, E. Rojas, J. Munoz-Gama, and M. Sepúlveda, *A Framework for Recommending Resource Allocation Based on Process Mining*, pp. 458–470. Springer, 2016.

[20] C. Cabanillas, J. M. García, M. Resinas, D. Ruiz, J. Mendling, and A. Ruiz-Cortés, *Priority-Based Human Resource Allocation in Business Processes*, pp. 374–388. Springer, 2013.

[21] Z. Huang, X. Lu, and H. Duan, "Resource behavior measure and application in business process management," *Expert Sys. with Apps.*, vol. 39, no. 7, pp. 6458 – 6468, 2012.

[22] W. Zhao, H. Liu, W. Dai, and J. Ma, "An entropy-based clustering ensemble method to support resource allocation in business process management," *Know. Apps. Info. Syst.*, vol. 48, no. 2, pp. 305–330, 2016.

[23] W. Zhao, L. Yang, H. Liu, and R. Wu, "The optimization of resource allocation based on process mining," in *Int'l Conf. on Intel. Comp.*, pp. 341–353, Springer, 2015.

[24] X. Liu, J. Chen, Y. Ji, and Y. Yu, "Q-learning algorithm for task allocation based on social relation," in *Int'l Workshop on Process-Aware Sys.*, pp. 49–58, Springer, 2014.

[25] A. Kumar, R. Dijkman, and M. Song, "Optimal resource assignment in workflows for maximizing cooperation," in *BPM*, pp. 235–250, Springer, 2013.

[26] H. Verbeek, J. Buijs, B. van Dongen, and W. van der Aalst, "ProM6: The process mining toolkit," in *BPM 2010 Demonstration Track*, CEUR Workshop Proceedings, (USA), pp. 34–39, CEUR-WS.org, 2010.

[27] A. Rogge-Solti, W. van der Aalst, and M. Weske, "Discovering stochastic Petri nets with arbitrary delay distributions from event logs," *BPM Workshops*, vol. 171, pp. 15–27, 2013.