IEEE Access Multidisciplinary | Rapid Review | Open Access Journal
SPECIAL SECTION ON INTELLIGENT INFORMATION SERVICES

Received March 7, 2020, accepted March 19, 2020, date of publication March 24, 2020, date of current version April 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983114

A Novel Approach for Business Process Similarity Measure Based on Role Relation Network Mining

QINGTIAN ZENG[1], JING LIU[2], CHANGHONG ZHOU[3], CONG LIU[4], AND HUA DUAN[5]
[1]College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[3]College of Economics and Management, Shandong University of Science and Technology, Qingdao 266590, China
[4]School of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China
[5]College of Mathematics and System Science, Shandong University of Science and Technology, Qingdao 266590, China

Corresponding authors: Changhong Zhou (zhouchanghong@163.com), Cong Liu (liucongchina@163.com), and Hua Duan (huaduan59@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61902222, in part by the Science and Technology Development Fund of Shandong Province of China under Grant ZR2017MF027, in part by the Taishan Scholars Program of Shandong Province under Grant ts2090936 and Grant tsqn201909109, in part by the Education Ministry Humanities and Social Science Project of China under Grant 18YJAZH017, in part by the Science and Technology Support Plan of Youth Innovation Team of Shandong Higher School under Grant 2019KJN024, in part by the SDUST Research Fund under Grant 2015TDJH102, and in part by the Philosophy and Social Science Planning Project of Qingdao under Grant QDSKL1801141.

ABSTRACT Existing business process similarity measure methods mainly focus on the control-flow (e.g., the activity and process model). However, the active factors of business processes, i.e., the roles that execute activities and organizations to which activities belong, have yet received enough attention. Roles are the subject of activities, many changes of process can be summarized as the management of roles and their relationship. In many cases, measuring process similarity from role perspective help find similarities and differences between processes and measure business process similarity more in a comprehensive way. This paper starts from activities and roles, and measures process similarity by constructing two role relation models. One is the Role Relation Network (RRN) that is defined based on the work delivery relation among roles, and the other is the Role Hierarchy Relation Network (RHRN) that is based on the role relation network and the role hierarchy graph. Then, we propose the corresponding business process similarity measure method. The proposed methods extend the applicability of existing process similarity measure methods to a larger application domain. Finally, experiments are designed to demonstrate the effectiveness of the proposed methods.

INDEX TERMS Business processes, process similarity, role relation network, role hierarchy graph.

I. INTRODUCTION

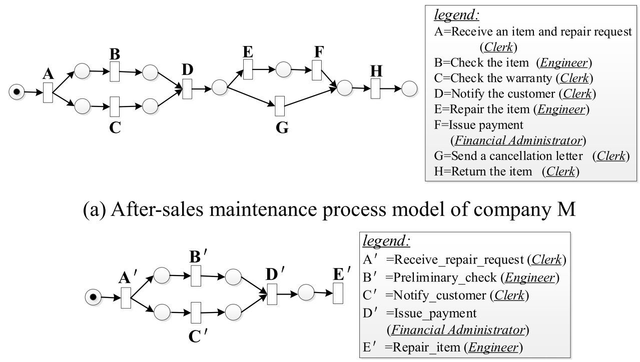
For enterprises, business process is the core of operation. All business management and business activities of enterprises are represented as various business processes. It can be seen that business process is of great significance to the operation and development of enterprises. Under the influence of process theory, more and more attention has been given to the research of Business Process Management (BPM) [1]. Generally speaking, BPM aims to sort out, analyze, improve and monitor the business process of an enterprise, and continuously optimize the business process to reduce the business processing cost. BPM can improve the business processing efficiency, quickly respond to the market and customer needs, and improve the decision-making ability of an enterprise [2].

To complete the construction, optimization and improvement of business processes in a rapid manner, it is important for an enterprise to calculate the similarity between business processes and analyze the similarities and differences between processes more intuitively [3]. Therefore, process similarity measure has become a hot research topic in the field of BPM. Process similarity measure is widely used in different application domain. Consider the process construction as an example, similar processes can be retrieved from existing process repository, and subsequent process activities are recommended for rapid process construction. Analyzing similar business processes can avoid the impact of repeated storage on consolidation efficiency for business process consolidation. For process optimization, it can

The associate editor coordinating the review of this manuscript and approving it for publication was Mengchu Zhou.

help improve the efficiency of existing processes by analyzing differences between them and those efficient ones. In addition, process mining, process integration, process retrieval, process reuse and many other business process management application need to measure the similarity between processes. According to different requirements, the understanding of "process similarity" is different, and the perspective of process similarity measure is also different [4]. For example, some approaches consider two processes similar if the textual labels of the elements in process models are similar [5]. Differently, some approaches measure the similarity by considering the process model topology [6]–[8] or the process model behavior [9]–[11]. In addition, some approaches calculate process similarity from data perspective [12] or a combination of the above perspectives [13]–[15].

However, most existing methods are based on process models and they mainly focus on the activities of the process and the relationship between activities. But they ignore some important factors such as roles and organizations in the process. Roles are executors (or performers) of activities, and the change or adjustment of the relationship between roles has a huge impact on the process models. Sometimes only from the perspective of activities, it may not be able to find the similarities and differences of the change of executive roles. More specifically, for some processes, it is not comprehensive enough to calculate the similarity between processes only from the control-flow perspective. Consider Fig. 1 as an example, the two process models represent the maintenance processes of two different enterprises. The process models in the example are represented by Petri nets. Petri nets as a powerful model, have been used to handle many problems in discrete event systems [16]–[18]. Therefore, in the field of business process management and service computing, many process models are expressed by Petri nets [19]–[22]. The two process models in Fig. 1 are different from each other in activity naming and process structure. This leads to the low process similarity based on traditional similarity measure. However, these two processes do share some similarities. Because both of them are maintenance processes, they are similar in terms of role relationship and role changes. Specially, in Fig. 1, the set of roles in these two process models is similar. The execution role of activities *A, C, D, H* in Fig. 1 (a) and activities *A′, C′* in Fig. 1. (b) is *Clerk*. The executive role of activities *B, E* in Fig. 1. (a) and activities *B′, E′* in Fig. 1 (b) is *Engineer*. And activities *F* in Fig. 1 (a) and *D′* in Fig. 1(b) are all executed by the role of *Financial Administrator*.In addition, for different after-sales service teams in the same enterprise, the business processes may be similar. But the process execution efficiency differ considerably due to different execution roles. Analyzing process similarity from the perspective of roles can enrich the existing work, and adapt to more application scenarios. Process similarity analysis from the perspective of the role can also help adjust the role structure, optimize the process and improve the process efficiency.



(a) After-sales maintenance process model of company M

(b) After-sales maintenance process model of company N

FIGURE 1. Maintenance process models of M and N.

In this paper, the basic concept is introduced in Section II. Two methods for calculating business process similarity are proposed in Section III. The similarity measurement methods based on the two models are described in Section IV. In SectionV, experiments are designed to compare and analyze the two methods based on the role relation models and traditional measurement methods, and then the effectiveness is demonstrated. Finally, the paper is concluded in Section VI.

II. BASIC CONCEPT

This paper mainly mines the role relationship from the process event log, and then calculates the process similarity.

A. ROLE

Role is a set of common features in the structure, nature, behavior and function of a certain kind of object. It is a comprehensive reflection of many essential characteristics, such as goals, abilities, responsibilities, and so on. It can be used as a reasonable criterion for classification of things [23], [24]. In an enterprise's business process, each activity has a specific executor. Role is the abstraction of these executors with the same function and similar behavior, whose goal is to complete a certain kind of task of the enterprise [25]. From the control-flow perspective, processes are essentially a collection of activities and their relationships. In fact, roles are the subject of activities, and many changes in processes can be summarized as the management of roles and their relationships. The activities and their relationships are only external manifestations of roles and their synergies. Role-oriented business process modeling builds a role relationship view, which regards the process as a process of multi-role collaboration [26]–[29] by analyzing the effects and interaction of roles in the process.

B. EVENT LOG

An event log is composed of a set of cases where each case refers to an independent execution of a business process. A case consists of a sequence of events. For each event, it may have different attributes, e.g., activity name, timestamp, organization, resource, executor, role, and etc. Note

TABLE 1. A fragment of an event log L.

CaseID	TaskID	Timestamp	ActivityName	Executor	Role
1	e_1	2014-09-22 15:23:00	Receive_repair_request	John	Clerk
1	e_2	2014-09-23 09:10:00	Preliminary_check	Mike	EngineerManager
1	e_3	2014-09-23 14:12:12	Warranty_check	John	Clerk
1	e_4	2014-09-24 12:21:12	Notify_customer	Sue	Clerk
1	e_5	2014-09-25 10:08:00	Issue_Payment	Clare	Financial
1	e_6	2014-09-25 10:58:00	Repair_Item	Pete	Engineer
2	e_7	2014-09-26 15:23:00	Receive_repair_request	John	Clerk
2	e_8	2014-09-2712:10:00	Warranty_check	John	Clerk
2	e_9	2014-09-27 14:12:12	Preliminary_check	Mike	EngineerManager
2	e_{10}	2014-09-27 12:21:12	Notify_customer	Sue	Clerk
2	e_{11}	2014-09-28 10:08:00	Send_cancellation_letter	Fred	Clerk
3	e_{12}	2014-09-28 15:23:00	Receive_repair_request	John	Clerk
3	e_{13}	2014-09-29 09:10:00	Preliminary_check	Mike	EngineerManager
3	e_{14}	2014-09-29 14:12:12	Warranty_check	John	Clerk
3	e_{15}	2014-09-29 12:21:12	Notify_customer	Sue	Clerk
3	e_{16}	2014-09-30 10:08:00	Issue_Payment	Clare	Financial
3	e_{17}	2014-09-30 10:58:00	Repair_Item	Pete	Engineer

that we only consider the attributes of the activity name, the executor and the role in this paper. Suppose L as an event log, A as the set of activities in the process, E as the set of events appearing in the log and $Attr$ as the set of event attributes. For any $e \in E$ and $attr \in Attr$, $\#_{attr}(e)$ represents the value of an attribute associated with event e. For instance, as shown in Table 1, $E = \{e_1, e_2, \ldots, e_{17}\}$, $Attr = \{Timestamp, ActivityName, Executor, Role\}$, $\#_{Executor}(e_1) =$ John, $\#_{Role}(e_1) =$ Clerk.

If for any two events $e_1 \in E$ and $e_2 \in E$ and $\#_{Timestamp}(e_1) < \#_{Timestamp}(e_2)$ and $\#_{Role}(e_1) \neq \#_{Role}(e_2)$, there is no event $e \in E$, which satisfies $\#_{Role}(e_1) \neq \#_{Role}(e) \neq \#_{Role}(e_2)$ and $\#_{Timestamp}(e_1) < \#_{Timestamp}(e) < \#_{Timestamp}(e_2)$. Then we think there is a work delivery relationship between $\#_{Role}(e_1)$ and $\#_{Role}(e_2)$, denoted as $\#_{Role}(e_1) \rightarrow \#_{Role}(e_2)$. For example, as shown in Table 1, $\#_{Role}(e_1) = Clerk$, $\#_{Role}(e_2) = EngineerManager$ and $Clerk \rightarrow EngineerManager$.

C. ORGANIZATION MINING

Process mining analyzes the execution of real processes based on event logs, and studies business processes from the perspective of control-flow, performance, and organization. Most of the early process mining studies were about the discovery of control flow. The business process was usually modeled by Petri net, BPMN and EPC by analyzing the execution sequence of activities in the process. With the development of related researches on social networks, more and more attention has been paid to process mining from the perspective of organization [30]. In such studies, organizational structure and social network among business participants of a process are mined, to discover the ways of information flow and work coordination within and between organizations.

One of the research emphasis of organization mining is to mine the role interaction model. The interaction between participants can be analyzed based on the social network model to discover the importance of different roles. In the social network analysis, role-based social network models can be established by different measurement methods [31]. When analyzing business processes focusing on roles or organizations, different methods of social network analysis can be adopted to depict the collaborative relationship between roles or organizations, business circulation relationship and the importance of individuals in business processes from multiple perspectives [32]–[34]. The existing research on organization mining is also based on social network analysis [35].

D. ROLE HIERARCHICAL GRAPH

A role hierarchy graph describes hierarchical relationships among roles in a business process. It is a tree structure with branches that represent the hierarchical relationships between roles, from superiors to subordinates. The role management relationship of the business process can be shown clearly in the role hierarchy graph.

Definition 1: A *role hierarchy graph* is defined as a 3-tuple $CG = (N_{root}, N_R, E_R)$, in which:

(1) N_{root} is the root node which only serves as the summarized connection node;

(2) N_R is the set of nodes which represent the roles in the process; and

(3) $E_R \subseteq (N_{root} \cup N_R) \times (N_{root} \cup N_R)$ is the set of edges which represent the relationship of roles.

Given $e \in E_R$ as a branch in CG and $e = < N_{Ri}, N_{Rj} >$. The edge e is defined as the leadership relation edge if and only if N_{Ri} is the parent node of N_{Rj} and $N_{Ri} \neq N_{root}$. The set of leadership relation edges is denoted as $E_{s\text{ìč}}$ And the set of non-leadership relation edges is denoted as E_d, that is, $E_d = E_R - E_s$.

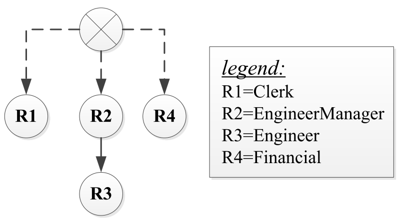


FIGURE 2. Role hierarchy graph for the process of event log L in Fig. 1.

Taking the process shown in the event log L as an example. The corresponding role hierarchy graph is shown in Fig. 2. The circle represents the role, and the circle with a fork inside represents the root node. Solid arrows denote the leadership relation and dotted arrows denote the non-leadership relation.

III. ROLE RELATIONSHIP NETWORK

A. ROLE RELATIONSHIP NETWORK

Definition 2 A *role relationship network graph* is represented as a 4-tuples $RG=(N, E, f_n, f_{trans})$, in which:

(1) N is the set of role nodes;

(2) $E \subseteq N \times N$ is a set of edges;

(3) f_n: $N \to R$ is a function that maps each node to a value. R is the set of real numbers which represent the role node's participation degree in the activity execution.

(4) f_{trans} : $E \to R$ is a function that maps edges to real numbers which denote the weight.

Take the process event log L in Table1 as an example, the role relation network is obtained as shown in the Fig. 3.

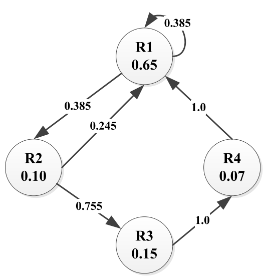


FIGURE 3. Role relation network RG_1 for the process of event log L in Fig 1.

B. ROLE HIERARCHY RELATION NETWORK

Based on the role hierarchy graph, we can obtain the set of leadership relation edges in the process. On the basis of the role relation network and the role hierarchy graph, we can distinguish whether the edges have a leadership relation, and then get the role hierarchy relation network.

Algorithm 1 Construction of Role Relation Network

Input: Event log L

Output: Role relation network $RG=(N, E, f_n, f_{trans})$

1. Suppose R is a set of all roles and $|R| = k$;
2. Initialize a k*k metric *R_edge* to record the number of appearances of each edge, a k*k metric *R_freq* to record the weights of each edge and an array *Rf* of length k to record the number of appearances of each role node;
3. $i = 0$;
4. for each $l_1, l_2, \ldots l_n$ in the L // $l_1, l_2, \ldots l_n$ denotes the sequence of activities that occur sequentially by timestamp in a case
5. Flag(l_1) =Flag(l_2) $= \cdots =$ Flag(l_n) $= 0$; //Flag array records whether events have been traversed
6. $i++$;
7. for j=1 to length(l_i) // Traverse the activities in the activity sequence
8. if the executor role of the activity l_j is R_m, next executor is R_n
9. then *R_edge[m−1, n−1]++*; *Rf[m-1]++*;
10. for i = *0* to k
11. for $j = 0$ to k // traverse *R_edge* and *Rf* to compute the frequency of each edge in RG
12. *R_freq[i, j] = R_edge[i, j]/Rf[i];*
13. End
14. return RG

Definition 3: A *role hierarchy relation network* is defined as a 6-tuple $RCG=(N', E', E_s', E_d', f_n', f'_{trans})$ where

Taking the process event log L in Table 1 as an example, the role hierarchy relation network is as shown in the Fig. 4.

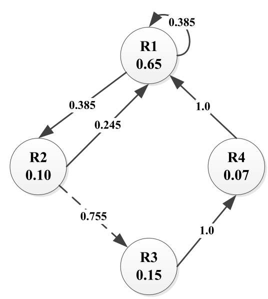


FIGURE 4. Role hierarchy relation network for the process of event log L in Fig 1.

IV. PROCESS SIMILARITY MEASURE BASED ON ROLE RELATIONSHIP

A. PROCESS SIMILARITY MEASURE BASED ON ROLE RELATION NETWORK

To compare two RGs, we extend the definition of graph edit distance to a new metric. The graph edit distance between two graphs is defined as the minimal cost of transforming one graph into the other. Transformations are captured as

Algorithm 2 Construction of Role Hierarchy Relation Network

Input: Role relation network $RG=(N, E, f_n, f_{trans})$, role hierarchy graph $CG=(N_{root}, N_R, E_R)$

Output: Role hierarchy relation network $RCG(N', E', E_s', E_d', f_n', f'_{trans})$

1. Suppose $|N| = k$, $N'=N$, $E'=E$, $f_n' = f_n$, $f'_{trans} = f_{trans}$
2. Initialize $E_s' = null$, $E_d' = null$
3. Initialize a $k*k$ metric R_cate to record whether the edge in E' *is* leadership relation edges or not;
4. Traverse the CG, if $E_R(i, j)$ *is a* leadership relation edge
5. $R\_cate[i, j] = 1$;
6. $i = 0, j = 0$;
7. for each $E(i, j) \in RG$, $i<k$, $j < k$ //Traverse the RG
8. if $R\_cate[i, j] = 1$
9. $E_s += E(i, j)$;
10. else
11. $E_d += E(i, j)$;
12. End
13. return RCG

sequences of elementary transformation operations including node substitution, node insertion/deletion and edge insertion/deletion. Each elementary operation has a cost, which is given by a cost function. The more similar two graphs are, the smaller the graph edit distance they have, i.e. the smaller the transformation cost is. The minimum edit distance can be obtained by A-star algorithm [36].

Definition 4: Let $RG_1 = (N_1, E_1, f_{n1}, f_{trans1})$ and $RG_2 = (N_2, E_2, f_{n2}, f_{trans2})$ be two role relation network graphs. Let $M: N_1 \rightarrow N_2$ be a partial injective mapping that maps nodes in RG_1 to nodes in RG_2. Let $dom(M) = \{n_1|(n_1, n_2) \in M\}$ be the domain of M and $cod(M) = \{n_2|(n_1, n_2) \in M\}$ be the codomain of M. We define the following basic operations:

(1) Given a node $n \in N_1 \cup N_2$, n is substituted if and only if $n \in dom(M)$ *or* $n \in cod(M)$. sn represents the set of all substituted nodes.

(2) A node $n_1 \in N_1$ is deleted from RG_1(or inserted to RG_2) if and only if $n_1 \notin sn$. A node that is deleted from RG_2(or inserted to RG_1) is defined in the same way. The set of all inserted or deleted nodes is represented as idn.

(3) Let $(n_1, m_1) \in E_1$ be an edge in RG_1. (n_1, m_1) is deleted from RG_1 (or inserted in RG_2) if and only if there does not exist a mapping M such that $(n_1, n_2) \in M$ and $(m_1, m_2) \in M$ and $(n_2, m_2) \in E_2$. Edges that are deleted from RG_2(or inserted to RG_1) are defined similarly. The set of all inserted or deleted edges is represented as ide.

(4) An edge is substituted if it is not inserted or deleted. se represents the set of all substituted edges, i.e., $se = (E_1 \cup E_2) - ide$.

Based on the above basic operations, we can extend the definition of graph edit distance based on the role relation network graph.

Definition 5: Let $RG_1 = (N_1, E_1, f_{n1}, f_{trans1})$ and $RG_2 = (N_2, E_2, f_{n2}, f_{trans2})$ be two role relation network graphs. Let $M : N_1 \rightarrow N_2$ be a partial injective mapping that maps nodes in RG_1 to nodes in RG_2. Let $dom(M)=\{n_1|(n_1, n_2) \in M\}$ be the domain of M and $cod(M)=\{n_2|(n_1, n_2) \in M\}$ be the co-domain of M. The edit distance of the role relationship network graphs based on the mapping M is computed as follows:

$$RGED_M(RG_1, RG_2) = \| sn \| + \| idn \| + \| se \| + \| ide \|$$

where:

$\|sn\|$ is the operational cost of node substitution. It is defined as the sum of the absolute values of the difference between the participation degrees of the corresponding substituted nodes, i.e., $\|sn\| = \sum_{n \in sn} |f_{n1}(n) - f_{n2}(n)|$;

$\|idn\|$ is the operational cost of node insertion and node deletion. It is defined as the sum of the participation degree of the inserted and deleted nodes, i.e., $\|idn\| = \sum_{n \in idn \wedge n \in N1} f_{n1}(n) + \sum_{n \in idn \wedge n \in N2} f_{n2}(n)$;

$\|se\|$ is the operational cost of edge substitution. It is defined as the sum of the absolute values of the difference between the weights of the corresponding substituted edges, i.e., $\|se\| = \sum_{e \in se} |f_{trans1}(e) - f_{trans2}(e)|$; and

$\|ide\|$ is the operational cost of edge insertion and edge deletion. It is defined as the sum of the weight of the inserted and deleted edges, i.e.,

$\|ide\| = \sum_{n \in ide \wedge e \in E1} f_{trans1}(e) \sum_{n \in ide \wedge n \in E2} f_{trans2}(e)$.

The $RGED$ of the two RGs can be computed as the minimal possible distance based on mapping M:

$$RGED(RG_1, RG_2) = \min_{M} RGED_M(RG_1, RG_2)$$

Let sn, idn, ide and se be the sets of substituted nodes, inserted/deleted nodes, inserted/deleted edges, and substituted edges. And wsn, $widn$, $wide$, wse are the weights that we assign to substituted nodes, inserted or deleted nodes, substituted edges and inserted or deleted edges, and $0 \leq wsn, widn, wide, wse \leq 1$. We define $fidn$, fsn, $fide$ and fse as follows:

$$fidn = \frac{\sum_{n \in idn \wedge n \in N_1} f_{n1}(n) + \sum_{n \in idn \wedge n \in N_2} f_{n2}(n)}{\sum_{n \in N_1} f_{n1}(n) + \sum_{n \in N_2} f_{n2}(n)};$$

$$fsn = \frac{\sum_{n \in sn} |f_{n1}(n) - f_{n2}(n)|}{\sum_{n \in sn} max(f_{n1}(n), f_{n2}(n))};$$

$$fide = \frac{\sum_{e \in ide \wedge e \in E_1} f_{trans1}(e) + \sum_{e \in ide \wedge e \in E_2} f_{trans2}(e)}{\sum_{e \in E_1} f_{trans1}(e) + \sum_{e \in E_2} f_{trans2}(e)};$$

$$fse = \frac{\sum_{e \in se} |f_{trans1}(e) - f_{trans2}(e)|}{\sum_{e \in se} max(f_{trans1}(e), f_{trans2}(e))}$$

where $fidn$ represents the fraction of inserted/deleted nodes, $fide$ represents the fraction of inserted/deleted edges, fsn represents the average distance of substituted nodes, and fse represents the average changes in weight value of substituted edges.

The graph edit similarity of RG_1 and RG_2 is defined as:

$$sim(RG_1, RG_2) = 1.0 - \frac{widn \times fidn + wide \times fide + wsn \times fsn + wse \times fse}{widn + wide + wsn + wse}$$

Considering for example the role relation network of the process event log L' is shown in Fig. 5. According to Figs. 3 and 5, we have R4∈ *idn*, <R2, R1>∈ *ide*, <R3, R4>∈ *ide*, <R4, R5>∈ *ide* and the other edges are the substituted edges. Therefore, using the weights $wsn = wse = wide = widn = 1$, the similarity is computed as shown at the bottom of this page.

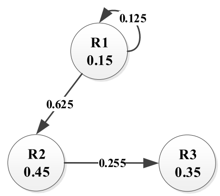


FIGURE 5. Role relation network RG_2 for the other process of event log *L'*.

B. PROCESS SIMILARITY MEASURE BASED ON ROLE HIERARCHY RELATION NETWORK

Let $RCG_1 = (N_1, E_1, E_{s1}, E_{d1}, f_{n1}, f_{trans1})$ and $RCG_2 = (N_2, E_2, E_{s2}, E_{d2}, f_{n2}, f_{trans2})$ be two role hierarchy relation network graphs. Let $M : N_1 \rightarrow N_2$ be a partial injective mapping that maps nodes in RG_1 to nodes in RG_2. Let $dom(M)=\{n_1|(n_1, n_2) \in M\}$ be the domain of M and $cod(M)=\{n_2|(n_1, n_2) \in M\}$ be the co-domain of M. The similarity measure of the role hierarchy relation network is very similar to that of the role relationship network, except that the leadership relation edges and the non-leadership relation edges should be distinguished in the operation of edges. Specially, the difference is as follows:

(1) Let $(n_1, m_1) \in E_{s1}$ be an leadership relation edge in RCG_1. (n_1, m_1) is substituted if and only if there exists a mapping M such that $(n_1, n_2) \in M$ and $(m_1, m_2) \in M$ and $(n_2, m_2) \in E_{s2}$.

(2) Let $(n_1, m_1) \in E_1$ be an edge in RG_1. (n_1, m_1) is deleted from RG_1 (or inserted in RG_2) if there does not exist a mapping M such that $(n_1, n_2) \in M$ and $(m_1, m_2) \in M$ and $(n_2, m_2) \in E_2$.

(3) Let $(n_1, m_1) \in E_{s1}$ be an leadership relation edge in RCG_1. (n_1, m_1) is deleted from RG_1 (or inserted in RG_2) if there exists a mapping M such that $(n_1, n_2) \in M$ and $(m_1, m_2) \in M$ and $(n_2, m_2) \in E_{d2}$.

(4) Let $(n_1, m_1) \in E_{d1}$ be an leadership relation edge in RCG_1. (n_1, m_1) is deleted from RG_1 (or inserted in RG_2) if there exists a mapping M such that $(n_1, n_2) \in M$ and $(m_1, m_2) \in M$ and $(n_2, m_2) \in E_{s2}$.

As long as the operation is determined, the remaining measurement methods are the same as the similarity measure of role relation network mentioned in A part of IV section.

TABLE 2. Activities and roles in L_0.

Role	Activity
R1	t1, t2 , t14
R2	t3, t4, t5,t7
R3	t6, t8,t9
R4	t10, t11
R5	t12, t13

V. EXPERIMENTS

This section performs a comprehensive set of experiments to evaluate the proposed approaches.

A. EXPERIMENTAL DATA

The data set used in this paper is an event log, denoted as L_0, of an enterprise procurement license approval process, containing the evaluation, allocation, procurement and other activities in the procurement process. The whole data set contains 608 cases, 9119 events and 5 roles. To facilitate the verification and comparison of the methods described in this

$$\begin{aligned}
\|sn\| &= |0.65-0.15|+|0.10-0.45| \\
&\quad +|0.15-0.35| = 1.05; \\
\|idn\| &= 0.07; \\
\|ide\| &= 0.245+1+1 = 2.245; \\
\|se\| &= |0.385-0.125|+|0.385-0.625| \\
&\quad +|0.755-0.255| = 1; \\
fsn &= \frac{1.05}{0.65+0.45+0.35} \approx 0.724; \\
fidn &= \frac{0.07}{0.65+0.10+0.15+0.15+0.45} \approx 0.053; \\
fide &= \frac{2.245}{0.385+0.385+0.245+0.755+1+1+0.125+0.625+0.255} \approx 0.47 \\
fse &= \frac{1}{0.385+0.625+0.755} \approx 0.567. \\
sim(RG_1, RG_2) &= 1.0 - \frac{1\times 0.724+1\times 0.053+1\times 0.47+1\times 0.567}{1+1+1+1} \approx 0.5465
\end{aligned}$$

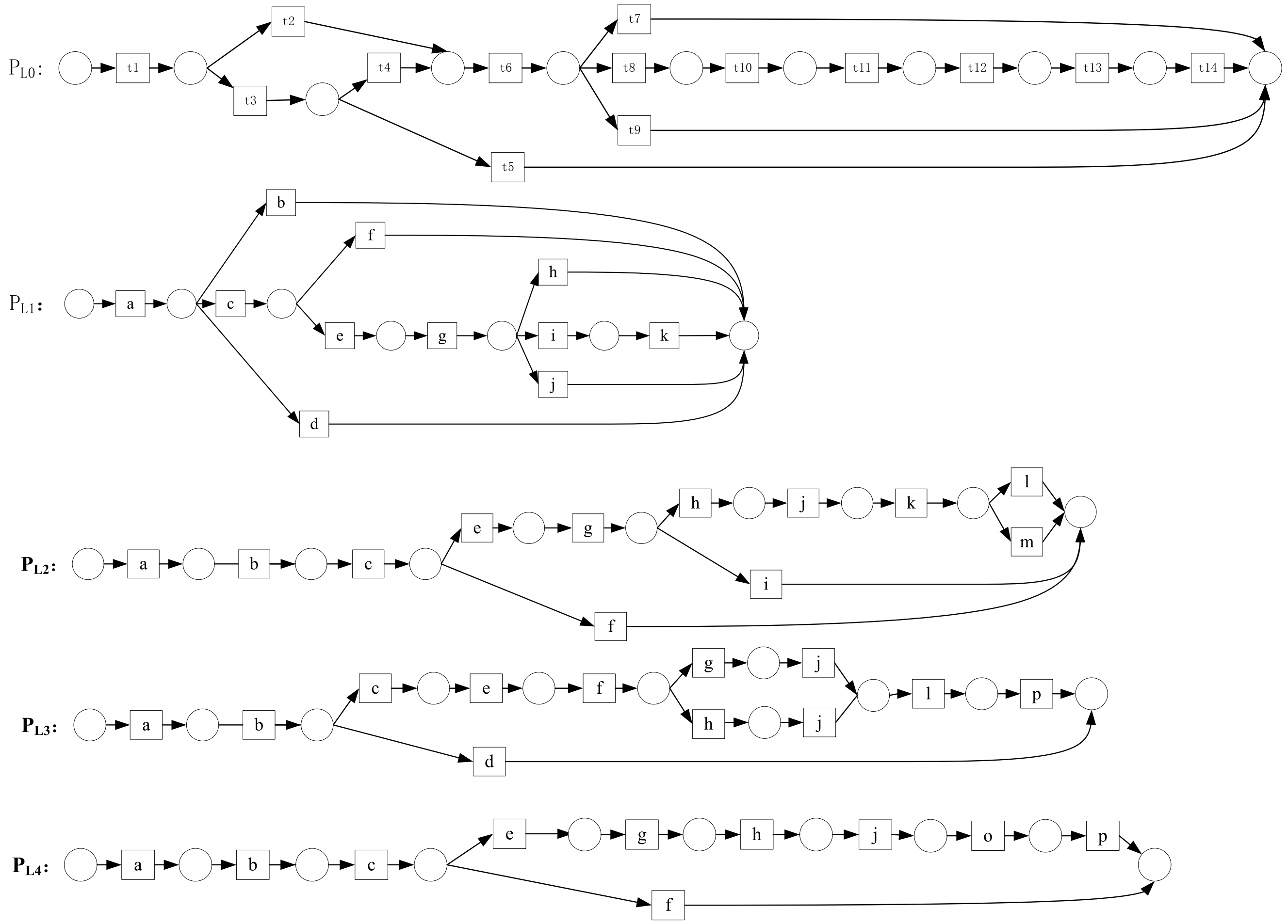


FIGURE 6. Process models mined from event logs $L_0 - L_4$.

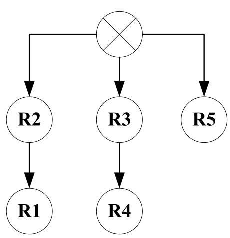


FIGURE 7. Role hierarchy graph of L_0.

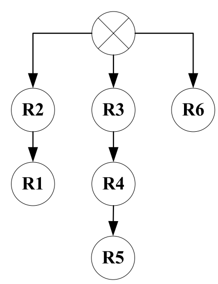


FIGURE 8. Role hierarchy graph of $L_1 - L_4$.

paper, the event logs which have a role structure similar to log L_0 are selected. More concretely, data sets $L_1 - L_4$ represent the building permit approval processes of four different cities. They have similar role structures and different process structures compared to L_0. After some pre-processing, it can be used as a comparative data set.The activities contained in the log L_0 and the corresponding execution roles are shown in Table 2. The activities contained in the log $L_1 - L_4$ and the corresponding execution roles are shown in Table 3.

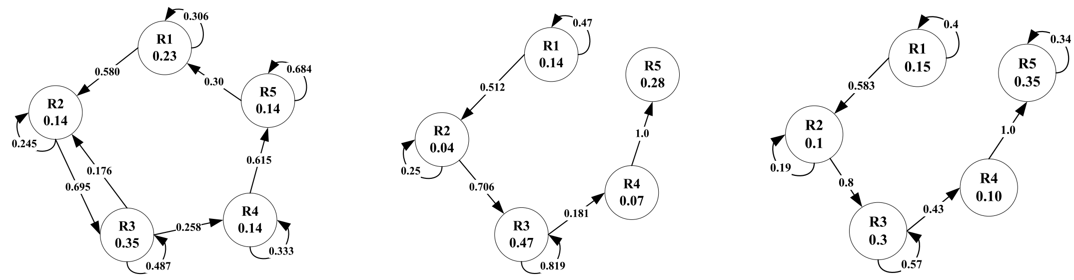
B. EXPERIMENTAL METHOD

In the experiment, we compare four methods as shown in Table 4.

It should be noted that the Graph Edit Distance (GED) method in M1 and M2 methods comes from the method in literature [30]. The main difference is that M1 converts the process model into a directed graph and then calculates the GED and M2 calculates the GED based on the role hierarchy graph. M3 and M4 are proposed in the part IV of this paper. And the weights of all graph operations in the experiment are set to 1.

C. EXPERIMENTAL MODEL

Different methods use different models for calculation. The models mainly include process model, role relationship network and role hierarchical relationship network.



(a) Role relation network of L_0

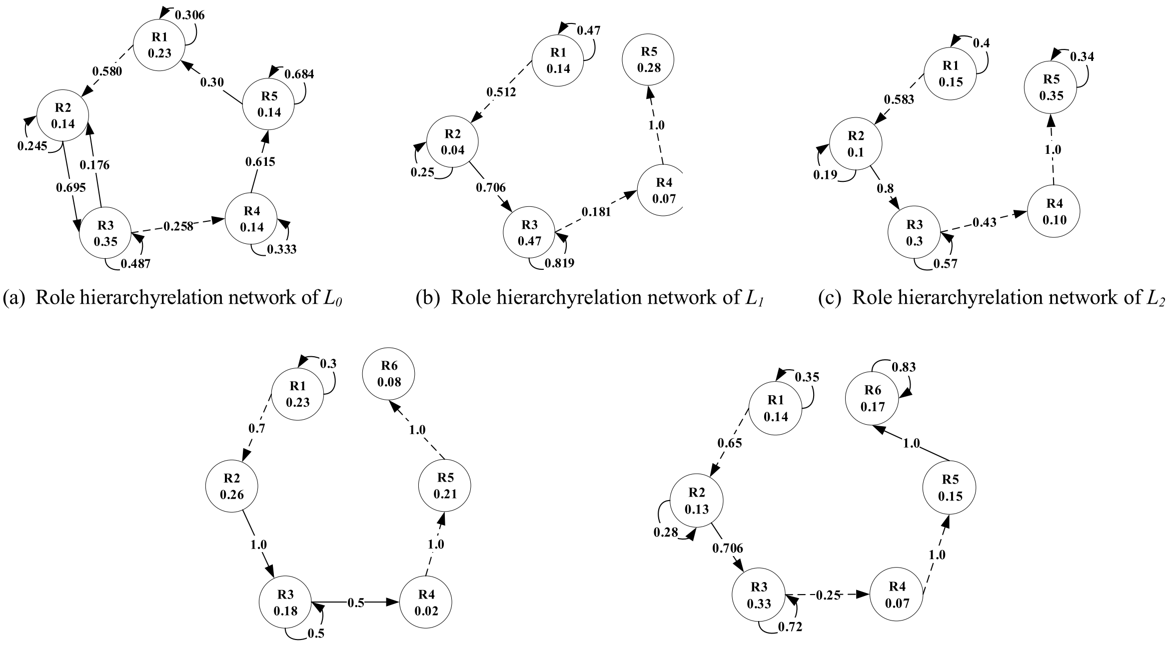
(b) Role relation network of L_1

(c) Role relation network of L_2

(d) Role relation network of L_3

(e) Role relation network of L_4

FIGURE 9. The role relation networks of $L_0 - L_4$.



(a) Role hierarchyrelation network of L_0

(b) Role hierarchyrelation network of L_1

(c) Role hierarchyrelation network of L_2

(d) Role hierarchy relation network of L_3

(e) Role hierarchy relation network of L_4

FIGURE 10. The role hierarchy relation networks of $L_0 - L_4$.

The process models of event logs $L_0 - L_4$ are obtained by Prom 6.6 tools and the Alpha algorithm. The corresponding mined process models are shown in Fig. 6. The role hierarchy graph of L_0 is shown in Fig. 7, and the role hierarchy graph of $L_1 - L_4$ is shown in Fig. 8.

The role relation networks mined from $L_0 - L_4$ by using Algorithm 1 are shown in Fig. 9 (a)-(e) below. And the role hierarchy relation networks are shown in Fig. 10 (a)-(e) below.

D. RESULT AND DISCUSSION

1) EXPERIMENTAL RESULTS

The above models of event logs $L_1 - L_4$ and L_0 are used to calculate the similarity by four methods *M1, M2, M3*, *M4*. The results are shown in Table 5:

TABLE 3. Actvities and roles in $L_1 - L_4$.

Role	Activity
R1	a, b
R2	c, d, f
R3	e, g, h,
R4	i, j,
R5	k, l, m
R6	o, p

TABLE 4. Methods for comparison.

Method number	Experiment Method
M1	Graph edit distance based on process model
M2	Graph edit distance based on role hierarchy graph
M3	Extended Graph edit distance based on role relation network
M4	Extended Graph edit distance based on role hierarchy relation network

TABLE 5. Experiment results.

Method / Data	*M1*	*M2*	*M3*	*M4*
L1&L0	0.4667	0.8333	0.7795	0.801
L2&L0	0.4937	0.8333	0.825	0.762
L3&L0	0.5542	0.8333	0.7393	0.710
L4&L0	0.6097	0.8333	0.8135	0.758

The results can be shown as line chart in Fig. 11.

2) DISCUSSION

(a) From the experimental results, the result of *M1* is relatively low, while the result of *M2* is much higher. By analyzing the specific reasons, we can find that the four models and the original model are quite different in activities and environment. But the purpose of the processes is similar. It has similar role hierarchy structure so that the similarity based on role hierarchy is relatively higher and the similarity based on process model is lower. Although the results of *M2* is relatively high, this approach cannot differentiate processes properly, i.e., the same values for all processes. This is because *M2* considers the interaction of roles without taking the specific level of interactions into consideration. Therefore, compared with *M3* and *M4*, *M2* does not make much sense for role interaction comparison.

(b) The result of *M3* proposed in this paper also has relatively high similarity calculation results because the four models and the original model are similar in process purpose, role setting and role interaction.

(c) The method *M4* proposed in this paper considers the role hierarchical relationship compared to *M3*. Because the role hierarchy relationship of the four models is different from

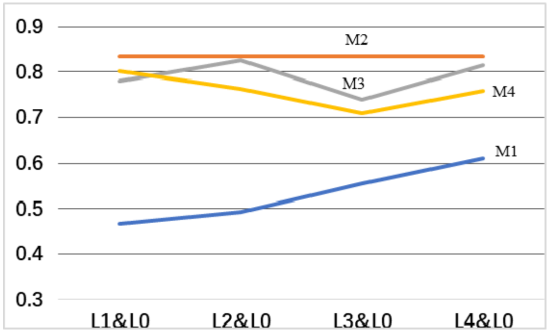


FIGURE 11. Experiment result.

that of the original model, the results of the method *M4* is less different from that of *M3*. The results are almost reduced, but the impact is not significant. This is in line with expectations.

(d) From the applicability perspective, process similarity calculation based on role relationship can be used in various scenarios. For example, two processes are similar in terms of traditional structure and behavior, but their execution efficiency differ greatly. This can be attributed to the different execution roles of these two processes. Then, we can find the differences between two processes based on the role hierarchy relation network, and modify the role structure of the low efficiency process according to the role level relationship of the high efficiency process. In addition, when the similarity of the two processes at the activity level is very small, activities can also be modified based on the role similarity.

VI. CONCLUSION

With the rapid development of social network, more and more attention has been paid to the organization, role and other aspects of business processes. A role is the executor of an activity, and the adjustment of the role affects the whole process. This paper measures process similarity by constructing two models. One is the Role relation network based on work delivery between roles, and the other is the Role hierarchy relation network based on role relation network and role hierarchy graph. The proposed methods can be used in scenarios where the process activities are quite different but the execution roles are similar. In addition, when the process activities are similar but the execution roles are different, which leads to low process efficiency, the proposed method can also be used to find the differences in the role relationship to help adjust the role structure and optimize the process configuration.

Based on the experimental evaluation, we can see that the process similarity measure based on role relationship can handle some limitations of existing process similarity measure, and measure process similarity from a different perspective. In the future, we plan to evaluate, adjust and optimize the processes from the perspectives of activities and roles by calculating the similarity of business processes.

REFERENCES

[1] W. M. P. van der AalstArthur H. M. ter Hofstede, and M. Weske, "Business process management: A survey," in *Business Process Management*. Berlin, Germany: Springer, 2003.

[2] Z. Weidong and L. I. U. Haitao, "Application of Process mining in process optimization," *Comput. Integr. Manuf. Syst.*, vol. 20, no. 10, pp. 2632–2642, 2014.

[3] G. Antunes, M. Bakhshandeh, and J. Borbinha, "The process model matching contest 2015," *GI-Edition, Lecture Notes Inform.*, vol. 248, pp. 127–155, Jun. 2015.

[4] A. Schoknecht, T. Thaler, and P. Fettke, "Similarity of business process models-a state-of-the-art analysis," *ACM Comput. Surv.*, vol. 50, no. 4, pp. 1–33, 2017.

[5] R. Dijkman, M. Dumas, B. van Dongen, R. Käärik, and J. Mendling, "Similarity of business process models: Metrics and evaluation," *Inf. Syst.*, vol. 36, no. 2, pp. 498–516, Apr. 2011.

[6] V. Gacitua-Decar and C. Pahl, "Structural process pattern matching based on graph morphism detection," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 27, no. 02, pp. 153–189, Mar. 2017.

[7] J. Li, L. J. Wen, and J. M. Wang, "Process model storage mechanism based on Petri net edit distance," *Comput. Integr. Manuf. Syst.*, vol. 19, no. 8, pp. 1832–1841, 2013.

[8] M. L. Sebu and H. Ciocarlie, "Similarity of business process models in a modular design," in *Proc. IEEE 11th Int. Symp. Appl. Comput. Intell. Informat. (SACI)*, May 2016, pp. 31–36.

[9] J. Song, L. Wen, and J. Wang, "A similarity measure for process models based on task occurrence relations," *J. Comput. Res. Develop.*, vol. 54, no. 4, pp. 832–843, 2017.

[10] Z. Dong, L. Wen, and H. Huang, "CFS: A behavioral similarity algorithm for process models based on complete firing sequences," in *Proc. OTM Conf. Int. Conf. Move Meaningful Internet Syst.* Berlin, Germany: Springer, 2014, pp. 202–219.

[11] Z. X. Wang, L. J. Wen, S. H. Wang, and J. M. Wang, "Similarity measurement for process models based on transition-labeled graph edit distance," *Comput. Integr. Manuf. Syst.*, vol. 22, no. 2, pp. 343–352, 2016.

[12] M. J. Amiri and M. Koupaee, "Data-driven business process similarity," *IET Softw.*, vol. 11, no. 6, pp. 309–318, Dec. 2017.

[13] C. Zhou, Q. Zeng, C. Liu, H. Duan, and G. Yuan, "Business process similarity computing method based on process model structure and log behavior," *Comput. Integr. Manuf. Syst.*, vol. 24, no. 7, pp. 1793–1805, 2018.

[14] C. Zhou, C. Liu, Q. Zeng, Z. Lin, and H. Duan, "A comprehensive process similarity measure based on models and logs," *IEEE Access*, vol. 7, pp. 69257–69273, 2019.

[15] C. Liu, Q. Zeng, H. Duan, S. Gao, and C. Zhou, "Towards comprehensive support for business process behavior similarity measure," *IEICE Trans. Inf. Syst.*, vol. E102.D, no. 3, pp. 588–597, Mar. 2019.

[16] X. Guo, S. Wang, D. You, Z. Li, and X. Jiang, "A siphon-based deadlock prevention strategy for S3PR," *IEEE Access*, vol. 7, pp. 86863–86873, 2019.

[17] W. Duo, X. Jiang, O. Karoui, X. Guo, D. You, S. Wang, and Y. Ruan, "A deadlock prevention policy for a class of multithreaded software," *IEEE Access*, vol. 8, pp. 16676–16688, 2020.

[18] S. Wang, D. You, and M. Zhou, "A necessary and sufficient condition for a resource subset to generate a strict minimal siphon in S 4PR," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 4173–4179, Aug. 2017.

[19] J. Sha, Y. Du, and L. Qi, "A user requirement oriented Web service discovery approach based on logic and threshold Petri net," *IEEE/CAA J. Automatica Sinica*, vol. 6, no. 6, pp. 1528–1542, Nov. 2019.

[20] W. Yu, C. Yan, Z. Ding, C. Jiang, and M. Zhou, "Analyzing E-Commerce business process nets via incidence matrix and reduction," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 1, pp. 130–141, Jan. 2018.

[21] W. Liu, P. Wang, Y. Du, M. Zhou, and C. Yan, "Extended logical Petri nets-based modeling and analysis of business processes," *IEEE Access*, vol. 5, pp. 16829–16839, 2017.

[22] L. Wang, Y. Du, and L. Qi, "Efficient deviation detection between a process model and event logs," *IEEE/CAA J. Automatica Sinica*, vol. 6, no. 6, pp. 1352–1364, Nov. 2019.

[23] H. Duan, C. Liu, Q. Zeng, and M. Zhou, "A package reduction approach to modeling and analysis of cross-organization emergency response processes with privacy protected," *IEEE Access*, vol. 6, pp. 55573–55585, 2018.

[24] Q. Zeng, S. X. Sun, H. Duan, C. Liu, and H. Wang, "Cross-organizational collaborative workflow mining from a multi-source log," *Decis. Support Syst.*, vol. 54, no. 3, pp. 1280–1301, Feb. 2013.

[25] X. Fang, "An optimized role identification method," *Comput. Appl. Softw.*, vol. 26, no. 4, pp. 114–117, 2009.

[26] J. Zhao and W. Zhao, "Process roles identification based on workflow logs mining," *Comput. Integr. Manuf. Syst.*, vol. 12, no. 11, pp. 1916–1920, 2006.

[27] W. Zhao, W. Dai, A. Wang, and X. Fang, "Role-activity diagrams modeling based on workflow mining," in *Proc. WRI World Congr. Comput. Sci. Inf. Eng.*, Mar./Apr. 2009, pp. 301–305.

[28] J. Wang, B. Zhu, Y. Wang, and L. Huang, "Mining organizational behaviors in collaborative logistics chain: An empirical study in a port," in *Proc. Int. Conf. Logistics, Informat. Service Sci. (LISS)*, Jul. 2016, pp. 1–5.

[29] M. Shen and D.-R. Liu, "Discovering role-relevant process-views for disseminating process knowledge," *Expert Syst. Appl.*, vol. 26, no. 3, pp. 301–310, Apr. 2004.

[30] X. Gao, B. Xiao, and D. Tao, "A survey of graph edit distance," *Pattern Anal. Appl.*, vol. 13, no. 1, pp. 113–129, 2010.

[31] S. Zhang, X. Liang, and J. Qi, "A review on role identification methods in social networks," *Chin. J. Comput.*, vol. 40, no. 3, pp. 649–673, 2017.

[32] M. Song and W. M. P. van der Aalst, "Towards comprehensive support for organizational mining," *Decis. Support Syst.*, vol. 46, no. 1, pp. 300–317, Dec. 2008.

[33] T. Jin, J. Wang, and L. Wen, "Organizational modeling from event logs," in *Proc. 6th Int. Conf. Grid Cooperat. Comput. (GCC)*, Aug. 2007, pp. 670–675.

[34] C. Alvarez, E. Rojas, M. Arias, J. Munoz-Gama, M. Sepúlveda, V. Herskovic, and D. Capurro, "Discovering role interaction models in the emergency room using process mining," *J. Biomed. Informat.*, vol. 78, pp. 60–77, Feb. 2018.

[35] W. M. P. van der Aalst, H. A. Reijers, and M. Song, "Discovering social networks from event logs," *Comput. Supported Cooperat. Work*, vol. 14, no. 6, pp. 549–593, Dec. 2005.

[36] R. M. Dijkman, M. Dumas, and L. García-Bañuelos, "Graph matching algorithms for business process model similarity search," in *Proc. Int. Conf. Business Process Manage.*, 2009, pp. 48–63.

QINGTIAN ZENG received the B.S. and M.S. degrees in computer science from the Shandong University of Science and Technology, Taian, China, in 1998 and 2001, respectively, and the Ph.D. degree in computer software and theory from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005. He is currently a Professor with the Shandong University of Science and Technology, Qingdao, China. His research interests are in the areas of Petri nets, process mining, and knowledge management.

JING LIU is currently pursuing the M.S. degree in computer software and theory with the Shandong University of Science and Technology, Qingdao, China. Her research interests are in the areas of business process management and process mining.

CHANGHONG ZHOU received the Ph.D. degree in software engineering from the Shandong University of Science and Technology, Qingdao, China, in 2018. She is currently a Lecturer with the Shandong University of Science and Technology. Her current research interests include process recommendation and process management.

CONG LIU received the B.S. and M.S. degrees in computer software and theory from the Shandong University of Science and Technology, Qingdao, China, in 2013 and 2015, respectively, and the Ph.D. degree in computer science and information systems from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2019. He is currently a Professor with the Shandong University of Technology, Zibo, China. His current research interests include business process management, process mining, petri nets, and Big data.

HUA DUAN received the B.S. and M.S. degrees in applied mathematics from the Shandong University of Science and Technology, Taian, China, in 1999 and 2002, respectively, and the Ph.D. degree in applied mathematics from Shanghai Jiaotong University, in 2008. She is currently an Associate Professor with the Shandong University of Science and Technology. Her research interests include petri nets, process mining, and machine learning.