Privacy-preserving Behavioral Correctness Verification of Cross-organizational Workflow with Task Synchronization Patterns

Cong Liu, QingTian Zeng, Long Cheng, Hua Duan, MengChu Zhou, *Fellow, IEEE*, and JiuJun Cheng

***Abstract*—Workflow management technology has become a key means to improve enterprise productivity. More and more workflow systems are crossing organizational boundaries and may involve multiple interacting organizations. This paper focuses on a type of loosely-coupled workflow architecture with collaborative tasks, i.e., each business partner owns its private business process and is able to operate independently, and all involved organizations need to be synchronized at certain point to complete certain public tasks. Because of each organization's privacy consideration, they are unwilling to share the business details with others. In this way, traditional correctness verification approaches via reachability analysis is not practical as a global business process model is unavailable for privacy preservation. To ensure its globally correct execution, this work establishes a correctness verification approach for cross-organizational workflow with task synchronization patterns. Its core idea is to use local correctness of each sub-organizational workflow process to guarantee its global correctness. We proved that the proposed approach can be used to investigate the behavioral property preservation when synthesizing sub-organizational workflows via collaborative tasks. A medical diagnosis running case is used to illustrate the applicability of the proposed approaches.**

***Note to Practitioners*— Cross-organizational workflow verification techniques play an increasingly important role for ensuring correct execution of collaborative enterprise businesses. This work addresses the issue of correctness verification for loosely-coupled interactive workflows with collaborative tasks. To ensure the globally correct execution, a behavioral correctness verification approach is established. All proposed concepts and techniques are supported by open-source tools and evaluation over a medical diagnosis process case has shown their applicability. The proposed methodology is readily applicable to industrial-size workflow correctness verification problems.**

***Index Terms*—Cross-organizational Workflow; Petri Nets; Behavioral Correctness Verification; Task Synchronization Pattern; Business Privacy Preservation**

I. Introduction

Workflow management has become a mature technology to guarantee the effectiveness and efficiency of enterprise business processes. However, with the development of information system environment, modern business processes are crossing organizational boundaries and typical involve a group of interacting organizations or partners [18]. For instance, a multi-modal transportation business scenario may involve customers, suppliers, and consigners. To support the interoperability of cross-organizational workflow, several conceptual architectures, e.g., chained execution, capacity sharing, and sub-contracting, are introduced in [38]. This work concentrates on a loosely coupled scenario where each business partner has its private business process and is able to operate independently. To ensure the correct execution of global cross-organizational workflow, these loosely-coupled processes need to be synchronized at certain points. Synchronization among interacting processes is the potential source of errors (e.g., deadlock and livelock) [39]. Consequently, it is of vital importance to perform the correctness verification for a cross-organizational workflow.

In general, there are two types of interaction mechanisms to communicate among different process partners, i.e., asynchronous and synchronous interaction patterns. The former consists of a message interaction pattern and a resource one, and the latter refers to a task synchronization pattern (different processes collaborate through some specific tasks) and a task out-sourcing pattern. Following [25], [48], [49], we have:

- Given two tasks that belong to two different organizations, one sends a message, and the other reads this message before executing. In this case, there exists a message interaction between them. A typical message interaction model between two processes is schematized with a Petri net in Fig. 1(a);
- Given two tasks that belong to two different organizations, if their required resource sets intersect completely or partially, then they satisfy a resource interaction pattern. A typical resource interaction model

This work was supported in part by the National Natural Science Foundation of China under Grant 61902222 and 61872271, in part by the Science and Technology Development Fund of Shandong Province under Grant ZR2017MF027, in part by the Taishan Scholars Program of Shandong Province under Grant ts20190936 and Grant tsqn201909109, and in part by the Shandong University of Science and Technology Research Fund under Grant 2015TDJH102. (Corresponding Authors: Q.T. Zeng and L. Cheng)

C. Liu is with the School of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China. (e-mail: liucongchina@sdust.edu.cn)

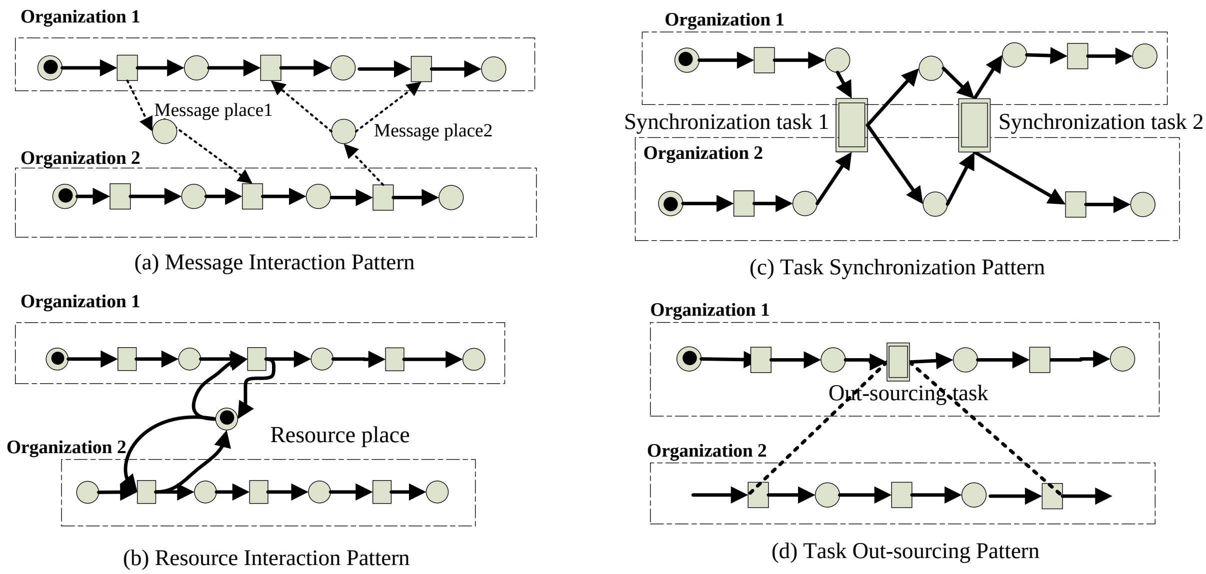
Q.T. Zeng and H. Duan are with the Shandong University of Science and Technology, Qingdao 266590, China. (e-mail: {qtzeng, huaduan59}@163.com)

L. Cheng is with Insight Centre for Data Analytics, School of Computing, Dublin City University, Dublin 9, Ireland. (e-mail: long.cheng@dcu.ie)

M.C. Zhou is with the Institute of Systems Engineering, Macau University of Science and Technology, Macau 999078, China and with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102. (e-mail: zhou@njit.edu)

J.J. Cheng is with Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, 200092, China (e-mail: chengjj@tongji.edu.cn)

Manuscript received February XX, 2019; revised XX, XX.



(a) Message Interaction Pattern

(b) Resource Interaction Pattern

(c) Task Synchronization Pattern

(d) Task Out-sourcing Pattern

Fig. 1. Four Kinds of Interaction Patterns of Loosely-Coupled Cross-organizational Workflow

between two processes is schematized with a Petri net in Fig. 1(b);

- Given two tasks that belong to two different organizations, if they need to collaborate to finish a certain mission, i.e., each organization shoulders parts of the mission, then there exists a task synchronization pattern. An example of task synchronization pattern is given in Fig. 1(c); and
- If one task in one organization is decomposed into (sub-contracted to) several sub-tasks in the other organization to be executed, then this task is an abstraction of the corresponding sub-process, and there is a task out-sourcing pattern between them. Such a model between two processes is shown in Fig. 1(d).

In the traditional research of discrete event systems, Petri-net based analysis of message and resource interactions among different sub-systems received much attention in the past two decades. As one of the most typical kind of resource interaction (allocation) systems [32], modeling and correctness verification of flexible manufacturing systems concerning resource allocation problems [12], [45], [50]–[52] have been extensively studied. Message interactions in Web service composition and cross-organizational workflow management have also received much concern [27], [38], [50]. When considering a task out-sourcing pattern, one may examine the research on the hierarchical analysis of control and manufacturing systems [1], [4], [8] and Petri net-based workflow integration [5], [7], [16], [41], [46]. According to Table I, most of the existing correctness verification work focuses on message interaction, resource allocation and task out-sourcing patterns among different workflow processes, while a task synchronization pattern has not received its deserved attention.

Process synchronization between two organizations using synchronization tasks can be classified into the following three cases: 1) Synchronization tasks exist in both processes. Graphically, synchronization tasks are drawn by concentric rectangles, as shown in Fig. 2 (a). 2) A synchronization task exists in only one process, and the other organization is asked to participate. We use a virtual task that is drawn by a black rectangle to synchronize with a real task, as shown in Fig. 2 (b). 3) There is no real synchronization task in both processes. We use a virtual task for each process to realize the synchronization, as shown in Fig. 2 (c). Note that the virtual tasks do not have any real-life meaning, and they are used to control process routing. In this paper, we limit our work to the first case, i.e., when synchronization tasks exist in both processes. This case is more common and useful in real-life settings, e.g., emergency management [7], healthcare [48], and logistics [49] than the other two cases.

The competitive business environment forces enterprises and organizations to be operationally innovative to outperform their competitors. Business process design innovations typically stand a significant portion of the total development cost. Therefore, process designers hope to keep their sensitive information confidential. For instance, detailed information of how certain business process fragments are organized or the group of process fragments used by an organization are proprietary. Unfortunately, the traditional reachability analysis-based correctness verification method is not practical to deal with the cross-organizational workflow as its global business process is unavailable when the business privacy-preserving issue of each organizations is considered. In addition, constructing a global reachability graph for some large-scale cross-organizational workflow may require prohibitively high computation. Therefore, we need to resort to other privacy-preserving verification techniques.

Note that local correctness of all sub-organization process models and their interaction patterns are available. Therefore, we can use them to imply its global correctness. We focus on the correctness verification of cross-organizational workflow with task synchronization patterns by taking business process

TABLE I
BRIEF SUMMARY OF EXISTING WORK ON VERIFICATION

Interaction Pattern	Research Background	Existing work
Message Interaction	Web service composition and Cross-organizational workflow	[27], [38], [50]
Resource Interaction	Resource allocation system, flexible manufacturing systems	[32], [47], [50]–[52]
Task out-sourcing	Control system, manufacturing systems and workflow integration	[1], [4], [5], [7], [8], [41]
Task synchronization	–	–

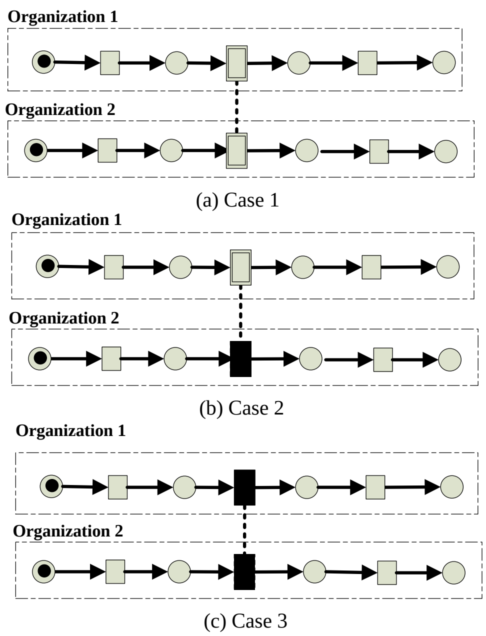


Fig. 2. Three Kinds of Task Synchronization Cases

privacy into consideration. Petri nets [24], [30], as a well-developed process language, are widely used to model and analyze workflows [36], [37]. In this paper, we lay our research on the basis of Petri nets. More precisely, we investigate correctness of a cross-organizational workflow from its behavioral perspective by using a Petri net language which is a set of transition firing sequences.

The contribution of this work is twofold:

- We propose a behavioral characterization approach to a sound WF-net using Petri net language theory;
- We propose a behavioral correctness verification approach for cross-organizational workflow with task synchronization pattern, which is able to preserve organization privacy; and
- The effectiveness of the proposed approach is demonstrated via a medical diagnosis case study.

The rest of this paper is structured as follows. Section II discusses the related work. Section III introduces Petri nets and their language theory. In Section IV, the behavior of a sound WF-net is characterized. Section V discusses the proposed correctness verification approaches. Section VI investigates a cross-organizational medical diagnosis business to show the applicability. Finally, Section VII concludes the paper.

II. RELATED WORK

This section summarizes the work related to the correctness verification of workflow of single organization and cross-organization workflow, and business process privacy preservation issue.

A. *Workflow of Single Organization*

Correctness verification aims at checking the rationality of a workflow model. It is usually performed on a kind of formal method, for instance, process algebra or Petri nets. [33] uses directed graphs to specify workflows, and therefore, the correctness of a workflow model is checked by analyzing a directed graph. As one of the most dominant models, WF-nets [37] as a special kind of Petri nets are widely used for workflow modeling and correctness verification. For WF-nets, the so-called soundness is defined, which can be verified by checking the liveness and boundedness, also known as reachability analysis. Woflan tool [42] that is based on the state-of-the-art Petri net theory is developed to verify standard WF-net. However, this method is based on structural analysis of a WF-net, i.e., performing the structural characterization of WF-nets, and no attention is paid on the behavioral characterization of a sound WF-net. In addition, the complexity for structure analysis may be high for a large-scale cross-organizational workflow and the global view of a cross-organizational workflow may be unavailable when considering the business privacy of individual organization.

B. *Cross-organizational Workflow*

In [38], Van der Aalst summarizes workflow processes that are hosted on a number of distributed organizations, based on which two main questions are addressed: (1) how to decide the minimal requirements to deploy these distributed processes, and (2) how to decide whether an inter-organizational workflow is consistent with an interaction structure. In [27], Liu et al. introduce interactive Petri nets to model message interaction behaviors between different processes, and the compatibility preservation conditions are proved. To support cross-organization interaction, van der Aalst introduces the use of lightweight workflow processes in [40]. It is proven that the flexibility and expressive power are improved compared to exsiting workflow modelling language. More recently, we formalize various cross-organizational coordination patterns, including message interaction pattern, task collaboration pattern, and resource interaction pattern, in [48]. The correctness is checked by generating the reachability graph. However, when considering process privacy for individual organization a global business process is usually not available, let alone its reachability graph. More recently, Reisig defined a general framework to support business process composition in [31]. The framework is instantiated by representing processes using different classes of Petri nets. Some property preservation conditions under composition are discussed with details.

C. Business Process Privacy Preservation

A privacy-aware framework is proposed by Labda et al. in [10] for mobile applications. The aim is to set up a set of constraints and based on which the BPMN is extended with privacy notations to guarantee privacy security from a process point of view. In [11], the framework is extended to support constraints reasoning and enforcement. Its applicability is validated by an airport emergency system scenario. All these existing works emphasize the security of personal data.

Schulz and Orlowska [34] propose to use process views for cooperation and communication among individual processes in a cross-organization setting. An organization owns its private process that guaranteed to be visible only for himself. Different partners participate in the shared process through a special view which is essentially an abstraction of the private process. This work introduces two approaches to construct views and the correctness criteria for different views are also investigated. Following this work, Eshuis and Grefen [9] provide a two-phase approach on the basis of inheritance for constructing optimized process views. Considering the preservation of business logic, each participating organization needs to reveal part of its business information to support cooperation while they should try to keep their sensitive information unseen by the others due to lack of mutual trust or other reasons. Therefore, it has been a great challenge to protect the business process privacy on condition that the whole cooperation is accomplished. Business process view provides an effective solution to this problem [2]. Following this idea, Tahamtan and Johann [35] give a new approach to facilitate the construction of process views for protecting process privacy. In this case, changes in a private process can be kept local and therefore the interactions with other partners are not affected. In our previous, we proposed a three-layered framework to model cross-organization emergency response processes by taking into account their privacy protection and temporal performance evaluation in [6]. The correctness is verified based on Petri nets reachability analysis.

D. Summary

From this brief literature review, we can see that research on cross-organizational workflow [34], [38], [48] have drawn a great deal of attention yet they have two limitations: (1) all existing studies on cross-organization workflow correctness verification are based on its structural analysis, i.e., performing structural characterization of a WF-net [37] but pays no attention on its behavioral characterization; and (2) most existing verification methods on cross-organizational workflow are based on constructing the reachability graph of its entire model whose computational complexity is prohibitively high for a large-scale workflow.

Because of each organization's privacy requirements, they are unwilling to share the business details with others. In this way, traditional correctness verification approaches via reachability analysis are not practical as a global business process model is unavailable when considering their respective business privacy [2], [9], [34], [35]. Therefore, we have to resort to use local correctness of each organization process and their interaction patterns to obtain its global correctness. Existing works on this idea only handle message interaction, resource interaction and task out-sourcing patterns [1], [4], [5], [8], [27], [38], [41], and none has addressed the correctness verification issues of cross-organizational workflow with task synchronization pattern by fully considering of its business privacy preservation. To our best knowledge, this is the first try towards this challenge.

III. Preliminaries

A sequence over set S of length n is a function $\sigma : \{1, 2, ..., n\} \rightarrow S$. If $\sigma(1) = a_1, \sigma(2) = a_2, ...\sigma(n) = a_n$, we write $\sigma = \langle a_1, a_2, ...a_n\rangle$. $|\sigma| = n$ represents the length of sequence σ is n. The set of all finite sequences over S is denoted as S^*. Given a sequence σ and an element e, we have $e \in \sigma$ if $\exists i : 1 \leq i \leq |\sigma| \wedge \sigma(i) = e$. Let $u, v \in S^*$ be two sequences, the *concatenation* operation denoted by $\sigma = u \circ v$ is defined as σ: $\{1, 2, ...|u|+|v|\} \rightarrow S$, such that $\sigma(i) = u(i)$ for $1 \leq i \leq |u|$, and $\sigma(i) = v(i - |u|)$ for $|u| + 1 \leq i \leq |u| + |v|$.

Workflow modeling techniques have been investigated for several decades, and many formal models, e.g., workflow net (WF-net), XML Process Definition Language, and Business Process Model and Notation (BPMN), are widely used. This work is based on Petri nets and their language. Some terminologies and notations of Petri nets [3], [13], [15], [17], [19], [20], [22]–[24], [26], [30], [43], [49] and WF-net [36], [37] are reviewed as follows.

Definition 1: (**Petri nets** [30]) A *Petri net* is defined as a 4-tuple $\Sigma = (P, T, F, M_0)$, such that

- $P \cap T = \emptyset$, $P \cup T \neq \emptyset$ where P is a finite set of places and T is a finite set of transitions;
- $F \subseteq (P \times T) \cup (T \times P)$ is a finite set of arcs; and
- $M_0 : P \rightarrow \{0, 1, 2, 3...n\}$ is the initial marking.

For any $x \in P \cup T$, ${}^\bullet x = \{y|(y, x) \in F\}$ is the preset of x and $x^\bullet = \{y|(x, y) \in F\}$ is its postset. M_0 denotes the initial marking and $R(M_0)$ is the set of reachable markings of Σ. For any $t \in T$, t is *enabled* under M, denoted as $(\Sigma, M)[t >$, if $\forall p \in^\bullet t : M(p) \geq 1$. If $(\Sigma, M)[t >$ holds, t may fire, resulting in a new marking M', denoted as $(\Sigma, M)[t > (\Sigma, M')$ such that $M'(p) = M(p) - 1$ if $p \in^\bullet t \backslash t^\bullet$, $M'(p) = M(p) + 1$ if $p \in t^\bullet \backslash^\bullet t$, and otherwise $M'(p) = M(p)$.

Definition 2: (**Boundedness** [30]) Let $\Sigma = (P, T, F, M_0)$ be a Petri net. $p \in P$ is *bounded* if there is an non-negative integer n, such that $\forall M_i \in R(M_0) : M_i(p) \leq n$. Σ is *bounded* if all its places are bounded.

This work focuses on bounded Petri nets, whose language is regular language [28]. The language operators: *connection* operator (denoted as $\circ$), *choice* operator (denoted as $+$), parallel operator (denoted as $\|$) , and *Kleen-closure* operator (denoted as $*$), [14], [30] can be used.

As the number of transitions is finite, transitions in a Petri net can be labeled with a set of not necessarily distinct symbols. All possible firing sequences of transitions constitute a formal language called a Petri net language [28]. According to [29], four types of languages, i.e., *L-type*, *G-type*, *T-type* and *P-type*, are defined by distinguishing their final markings. Each language is then divided into three classes based on the choice of transition labeling (*free*, *λ-free*, and *arbitrary*). In the following, we use the free L-type language as example by setting the finite final marking set as the subset of its reachable marking set. T^* denotes all possible transition sequences in T including the empty one.

A Petri net that describes a workflow process is named a workflow net (or WF-net). Its definition is reviewed based on [37].

Definition 3: (**Workflow nets** [37]) A Petri net $\Sigma=(P,T,F,M_0)$ is a WF-net if the following conditions hold:

- there exists a source place $i \in P$ such that ${}^\bullet i = \emptyset$;
- there exists a sink place $o \in P$ such that $o^\bullet = \emptyset$;
- each node is on a path from i to o; and
- for each $p \in P$, $M_0(p) = 1$ if $p = i$, and otherwise $M_0(p) = 0$.

Because a process is created once it enters the workflow engine and destroyed when completed, we use a source place and a sink place to explicitly denote the initial and final states. Transitions are used to represent tasks or activities in a process.

IV. Correctness Verification of Workflow within Single Organization

In this section, we verify the correctness of workflow within one organization using Petri net language. Before rendering our methodology, we first introduce the classical correctness notion of a WF-net, which is called soundness. A necessary and sufficient condition to determine its soundness is due to [37]. Note that the requirements for soundness in Definition 4 are built on top of the requirements for being a WF-net in Definition 3.

Definition 4: (**Soundness** [37]) Let $\Sigma = (P,T,F,M_0)$ be a WF-net. Σ is *sound* if and only if:

- for any M_i reachable from the initial marking M_0, there exists a firing sequence leading from M_i to the final marking M_e, i.e., for any $M_i \in R(M_0)$, we have $M_e \in R(M_i)$;
- M_e is the only marking reachable from M_0 with at least one token in the sink place o; and
- there are no dead transitions in Σ.

The first condition states that starting from M_0, it is always possible to reach the final marking M_e. Then, the second condition implies that if place o contains one token, other places are empty. Finally, the last one shows that there is no dead transition during its execution. This correctness requirement is verified by checking the *boundedness* and *liveness* of an extended short-circuit WF-net. For more discussion, readers can refer to [37]. Differently, we investigate the correctness of a WF-net based on its language characteristic.

According to Definition 3, a WF-net has a pre-defined final marking, denoted as M_e, such that $M_e(o) = 1$ for sink places o; and $M_e(p) = 0$ for other places ($p \in P \wedge p \neq o$). Therefore, we first define two special types of WF-net languages. Note that we use the term correct and sound synonymously for the remaining discussion.

Definition 5: (**Sink language**) Let $\Sigma = (P,T,F,M_0)$ be a WF-net, $L(\Sigma, M_e) = \{\sigma \in T^* | M_0[\sigma > M_e\}$ is defined as the *sink language* of Σ.

Definition 6: (**Full language**) Let $\Sigma = (P,T,F,M_0)$ be a WF-net, $L(\Sigma, R(M_0)) = \{\sigma \in T^* | M_0[\sigma > M_i \wedge M_i \in R(M_0)\}$ is defined as the *full language* of Σ.

Obviously, the *sink language* consists of transition sequences whose firing leads from M_0 to M_e while the *full language* takes all possible transition sequences whose firing leads M_0 to any marking in $R(M_0)$.

Definition 7: (**Prefix language**) Let $\Sigma = (P,T,F,M_0)$ be a WF-net and $L(\Sigma, R(M_0)) \subseteq T^*$ be its *full language*. $Pref(L(\Sigma, R(M_0))) = \{\sigma | \exists \sigma' \in T^* : \sigma \circ \sigma' \in L(\Sigma, R(M_0))\}$ is defined as the *prefix language* of $L(\Sigma, R(M_0))$

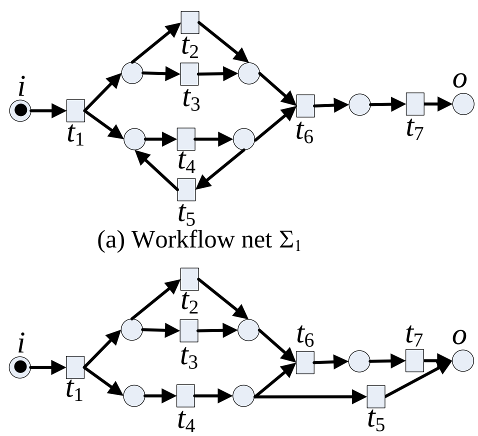
Then, we prove the soundness of a workflow net from the language perspective.

Theorem 1: Let $\Sigma = (P,T,F,M_0)$ be a WF-net, Σ is sound if and only if $L(\Sigma, R(M_0)) = Pref(L(\Sigma, M_e))$.

Proof: (Sufficiency) For any $\sigma \in L(\Sigma, R(M_0)$, $\exists M_i \in R(M_0)$ such that $M_0[\sigma > M_i$. Because Σ is sound, there exists $M_i \in R(M_0)$ such that $M_0[\sigma' > M_i$ and $M_i[\sigma'' > M_e$ such that $\sigma = \sigma' \circ \sigma''$, i.e., $\sigma \in Pref(L(\Sigma, M_e))$, therefore, we have $Pref(L(\Sigma, M_e)) \subseteq L(\Sigma, R(M_0))$. On the other hand, it is obvious that we have $L(\Sigma, R(M_0)) \subseteq Pref(L(\Sigma, M_e))$. Finally, we prove that $L(\Sigma, R(M_0)) = Pref(L(\Sigma, M_e))$.

(Necessity) (1) For any $M_i \in R(M_0)$, we have a firing sequence $\sigma \in L(\Sigma, R(M_0))$ satisfying $M_0[\sigma > M_i$. If $M_i = M_e$, it is obvious we have $\sigma \in Pref(L(\Sigma, M_e))$; If $M_i \neq M_e$, we assume that $\nexists \sigma'$ such that $M_i[\sigma' > M_e$, i.e., $\sigma \notin Pref(L(\Sigma, M_e))$. Hence, we conclude that $L(\Sigma, R(M_0)) \neq Pref(L(\Sigma, M_e))$ which is a contradiction. Therefore, our assumption is invalid and there exists σ' such that $M_i[\sigma' > M_e$, i.e. $M_0[\sigma > M_i$ and $M_i[\sigma' > M_e$. In a word, for every marking M_i reachable from M_0, there exists a firing sequence leading from M_i to M_e if Σ is sound. (2) Assume that M_e is not the only marking reachable from M_i with at least one token in the sink place o, i.e., $\exists M_e' \in R(M_0)$ such that $M_e'(o) = 1$ and $M_e' \neq M_e$, so we also conclude that $L(\Sigma, R(M_0)) \neq Pref(L(\Sigma, M_e))$ which is a contradiction. As a result, we prove that M_e is the only marking reachable from M_0 with at least one token in o if Σ is sound. (3) Assume that there exists a dead transition $t' \in T$, i.e., $M_i \in R(M_0)$ such that t' is not enabled and $M_i \neq M_e$. We can also conclude that $L(\Sigma, R(M_0)) \neq Pref(L(\Sigma, M_e))$ which is a contradiction. Thus, there is no dead transition in Σ if it is sound. ■

According to Theorem 1, to determine the correctness of a WF-net, we need to test if its whole language and the prefix language of its sink language are equal. Based on the language theory, we need to judge the equivalence of its corresponding automata to test the equality of these two languages. For our case, the whole language can be generated by the automata corresponding with the original Petri net and the prefix language of sink language can be generated by the automata corresponds with the Petri net that can produce the sink language.



(a) Workflow net Σ_1

(b) Workflow net Σ_2

Fig. 3. A Simple WF-net Example and its Variant

To illustrate this verification process, we give an example WF-net (denoted as Σ_1) and one of its variants (denoted as Σ_2) as shown in Fig. 3. For Fig. 3 (a), we have $L(\Sigma_1, R(M_0)) = Pref(t1((t2 + t3) \parallel (t4t5)^{*}t4)t6t7)$ and $L(\Sigma_1, M_e) = t1((t2 + t3) \parallel (t4t5)^{*}t4)t6t7$, therefore $L(\Sigma_1, R(M_0)) = Pref(L(\Sigma_1, M_e))$, which means that Σ_1 is sound. Similarly for Fig. 3 (b), we can obtain $L(\Sigma_2, R(M_0)) = Pref(t1((t2 + t3) \parallel t4)(t6t7 + t5))$ and $L(\Sigma_2, M_e) = t1((t2 + t3) \parallel t4)t6t7$, therefore $L(\Sigma_2, R(M_0)) \neq Pref(L(\Sigma_2, M_e))$. It means that Σ_2 is not sound.

V. Cross-organizational Workflow with Task Synchronization Patterns

When designing a loosely-coupled workflow architecture, different types of interaction patterns, e.g., asynchronous and synchronous patterns, are used. Based on these patterns, each organization can interact with each other at certain points. To ensure the correct execution of a global cross-organizational workflow, we investigate its correctness verification in this section. It is worth noting that each organization has full control of their respective private workflow process and our verification is laid on the fact that each involved organization is correct, i.e., using local correctness to imply global correctness. In addition, the interaction patterns like message interaction pattern, resource interaction pattern and task out-sourcing pattern received much attention in Web service composition area and flexible manufacturing systems [1], [4], [5], [8], [27], [38], [41], [44], [50] in the past two decades as summarized in Table I. Therefore, we focus on another type of synchronization interaction pattern, named task synchronization pattern.

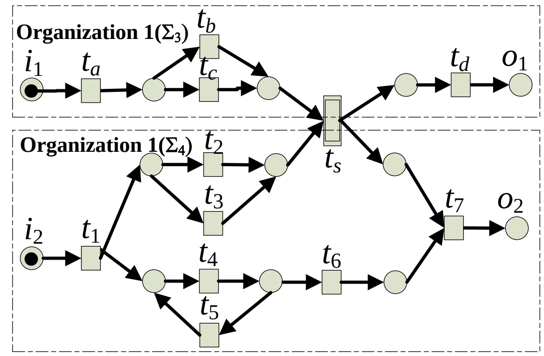


Fig. 4. A CWF-net with Task Synchronization Pattern

If two organizations need to synchronize to finish a shared task, i.e., each undertakes parts of the task, then task synchronization pattern exists between them. Let $\Sigma_1 = (P_1, T_1, F_1, M_{01})$ and $\Sigma_2 = (P_2, T_2, F_2, M_{02})$ be the WF-nets. There exists a task synchronization pattern if (1) $P_1 \cap P_2 = \emptyset$, and (2) $T_1 \cap T_2 \neq \emptyset$.

Definition 8: (**CWF-nets**) Let $\Sigma_1 = (P_1, T_1, F_1, M_{01})$ and $\Sigma_2 = (P_2, T_2, F_2, M_{02})$ be two WF-nets and $\Sigma_{TS} = (P_{TS}, T_{TS}, F_{TS}, M_{TS0})$ be the cross-organization workflow net with a task synchronization pattern, denoted as *CWF-net*, such that:

- $P_1 \cap P_2 = \emptyset$ and $P_{TS} = P_1 \cup P_2$;
- $T_1 \cap T_2 \neq \emptyset$ and $T_{TS} = T_1 \cup T_2$;
- $F_{TS} = F_1 \cup F_2$; and
- $\forall p \in P$, $M_{TS0}(p) = 1$ if $\exists i \in \{1, 2\}$ such that $M_{0i}(p) = 1$; otherwise $M_{TS0}(p) = 0$.

Fig. 4 gives a CWF-net where two organizations (denoted as Σ_3 and Σ_4) are involved and the synchronization task (transition) is denoted as t_s. These two organizations interact with each other via a task synchronization pattern.

To verify the correctness of a CWF-net, we first introduce the unfolding of an arbitrary CWF-net to a standard WF-net. In fact, this idea is motivated by the work in [39] which gives a detailed discussion on the correctness verification of cross-organizational workflows.

Definition 9: (**UWF-nets**) Let $\Sigma_{TS} = (P_{TS}, T_{TS}, F_{TS}, M_{TS0})$ be a CWF-net, and $\Sigma_1 = (P_1, T_1, F_1, M_{01})$ and $\Sigma_2 = (P_2, T_2, F_2, M_{02})$ be its corresponding WF-net. $\Sigma_U = (P_U, T_U, F_U, M_{U0})$ is the unfolding net of Σ_{TS}, denoted as UWF-net, if:

- $P_U = P_{TS} \cup \{i, o\}$;
- $T_U = T_{TS} \cup \{t_i, t_o\}$;
- $F_U = F_{TS} \cup \{(i, t_i), (t_i, i_1), (t_i, i_2), (t_o, o), (o_1, t_o), (o_2, t_o)\}$; and
- for any $p \in P_U$, $M_{U0}(p) = 1$ if $p = i$; otherwise $M_{U0}(p) = 0$.

In a UWF-net, local WF-nets are connected by a source transition t_i and a sink one t_o. In addition, source place i is added to initialize the UWF-net and sink place o is used to represent the termination of its execution. The synchronization tasks are regarded as ordinary transitions

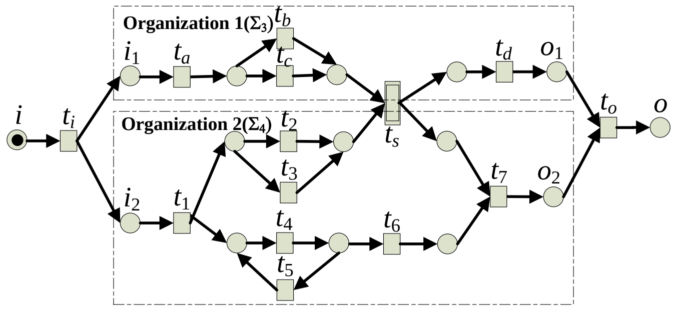


Fig. 5. A UWF-net Example

in the newly-constructed UWF-net. The UWF-net of the CWF-net in Fig. 4 is shown in Fig. 5.

Theorem 2: Let $\Sigma_U = (P_U, T_U, F_U, M_{U0})$ be a UWF-net. Then Σ_U is a WF-net.

Proof: Obviously, the four criteria defined in Definition 3 are satisfied, i.e., (1) there exists a source place $i \in P_U$ such that $^\bullet i = \emptyset$; (2) there exists a sink place $o \in P_U$ such that $o^\bullet = \emptyset$; (3) every node $x \in P_U \cup T_U$ is on a path from i to o; and (4) for any $p \in P_U$, $M_{U0}(p) = 1$ if $p = i$; otherwise $M_{U0}(p) = 0$. ■

Next, we define the behavioral correctness of a CWF-net based on UWF-net.

Definition 10: (**Behavioral correctness of CWF-nets**) A CWF-net is behaviorally correct if it is both locally correct and globally correct. Moreover, it is locally correct if all its local WF-nets are sound, and it is globally correct if its unfolding net UWF-net is correct.

According to Definition 10, to verify the behavioral correctness of a cross-organization workflow with n organizations $(\Sigma_1, \Sigma_2, \ldots, \Sigma_n)$, we need to check the correctness of $n + 1$ WF-nets, i.e., Σ_1, Σ_2, ..., Σ_n and its UWF-net which has been proved to be a WF-net in Theorem 2. Theoretically, this is easy as we have fully discussed the correctness verification approach of a WF-net from its language aspect in Section IV. Unfortunately, in a loosely-coupled environment, it is unrealistic and even impossible to obtain the global cross-organizational workflow process model when considering privacy preservation of each organization [18], [49]. All we have are the information about the correctness of each organization and interactions among them. Next, we investigate the correctness of the whole cross-organizational workflow based on the correctness of its sub-organizations, i.e., investigating the global correctness via local correctness.

Definition 11: (**Projection**) Let X be a set and $Q \subseteq X$ be its subset. $\Gamma_{X \to Q} \in X^* \to Q^*$ is a projection function and is defined recursively: $\Gamma_{X\to Q}(\langle\rangle) = \langle\rangle$; and for $\sigma \in X^*$ and $x \in X$:

$$\Gamma_{X\to Q}(\langle x\rangle \circ \sigma) = \begin{cases} \Gamma_{X\to Q}(\sigma) & if \quad x \notin Q \\ \langle x\rangle \circ (\Gamma_{X\to Q}(\sigma) & if \quad x \in Q \end{cases} \quad (1)$$

Specially, for $L \subseteq X^*$, $\Gamma_{X\to Q}(L) = \{\Gamma_{X\to Q}(\sigma) | \sigma \in L\}$.

For example, assume that $X = \{a, b, c, d\}$ and $Y = \{a, b\}$, $L = \{abcd, aabbccdd, bbcdd\}$, we have $\Gamma_{X\to Y}(L) = \{ab, aabb, bb\}$.

Let $\Sigma_i = (P_i, T_i, F_i, M_{0i}) (i \in \{1, 2\})$ be two WF-nets. $\Sigma_{TS} = (P, T, F, M_0)$ is the CWF-net of Σ_1 and Σ_2 with task synchronization pattern.

Definition 12: (**Behavioral invariance**) Σ_1 and Σ_2 are behaviorally invariant if:

- $\Gamma_{T\to T_i}(L(\Sigma_{TS}, R(M_0))) = L(\Sigma_i, R(M_{0i}))$; and
- $\Gamma_{T\to T_i}(L(\Sigma_{TS}, M_e) = L(\Sigma_i, M_{ei}))$, $i \in \{1, 2\}$.

Theorem 3: Σ_{TS} is behaviorally correct if Σ_1 and Σ_2 are behaviorally invariant and correct.

Proof: According to Definition 12, if Σ_1 and Σ_2 are behaviorally invariant, then we have $\Gamma_{T\to T_i}(L(\Sigma_{TS}, R(M_0))) = L(\Sigma_i, R(M_{0i}))$ and $\Gamma_{T\to T_i}(L(\Sigma_{TS}, M_e) = L(\Sigma_i, M_{ei}))$ $(i \in \{1, 2\})$. According to Theorem 1, if Σ_i is sound then we have $L(\Sigma_i, R(M_{0i})) = Pref(L(\Sigma_i, M_{ei}))$. Based on the pre-mentioned conditions, we have $\Gamma_{T\to T_i}(L(\Sigma_{TS}, R(M_0))) = Pref(\Gamma_{T\to T_i}(L(\Sigma_{TS}, M_e))$, i.e. $\Gamma_{T\to T_i}(L(\Sigma_{TS}, R(M_0))) = \Gamma_{T\to T_i}(Pref(L(\Sigma_{TS}, M_e)))$. Therefore, $L(\Sigma_{TS}, R(M_0)) = Pref(L(\Sigma_{TS}, M_e))$. In this way, we prove that Σ_{TS} is correct. In a word, we conclude that Σ_{TS} is correct if and only if Σ_1 and Σ_2 are behaviorally invariant and correct. ■

According to Theorem 3, to determine the behavioral correctness of a CWF-net, we need to judge the following two conditions: (1) if all its sub-organization processes are correct; and (2) if all its sub-organization processes are behaviorally invariant. We have addressed the correctness verification approaches of each WF-net in last section, and therefore, we only need to investigate the behaviorally invariant property among those sub-organizations.

To determine whether two arbitrary workflow processes are behaviorally invariant or not, we need to judge if their language projections on a public synchronization transition set are identical. The following theorem presents the details.

Theorem 4: Σ_1 and Σ_2 are behaviorally invariant iff: (1) $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1, R(M_{01}))) = \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2, R(M_{02})))$; and (2) $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1, M_{e1})) = \Gamma_{T_2\to T_1\cap T_2}L(\Sigma_2, M_{e2})$.

Proof: As $L(\Sigma_i, R(M_{0i}))$ and $L(\Sigma_i, M_{ei})$ represent two kinds of languages differed by their terminal markings, without the loss of generality, we unify these two notions as $L(\Sigma_i)$ for convenience. In other word, we only need to prove that Σ_1 and Σ_2 are behaviorally invariant if and only if $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1)) = \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2))$.

(Sufficiency) According to Definition 12, if Σ_1 and Σ_2 are behavioral invariant, then $\Gamma_{T\to T_i}(L(\Sigma_{TS})) = L(\Sigma_i)(i \in \{1, 2\})$. First, we prove that $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1)) \subseteq \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2))$. We denote $T_1 \cap T_2 = T_\Delta$ and, obviously, we have $T_\Delta = \emptyset$ as Σ_1 and Σ_2 satisfy a task synchronization pattern. For $\sigma_{\Delta i} \in \Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1))$ then $\sigma_i \in L(\Sigma_1)$ such that $\Gamma_{T_1\to T_1\cap T_2}(\sigma_i) = \sigma_{\Delta i}$. As we have $\Gamma_{T\to T_1}(L(\Sigma_{TS})) = L(\Sigma_1)$ and $\sigma_i \in L(\Sigma_1)$, $\exists\sigma \in L(\Sigma_{TS})$ such that $\Gamma_{T\to T_1}(\sigma) = \sigma_i$. Thus, $\sigma_{\Delta i} = \Gamma_{T_1\to T_1\cap T_2}(\Gamma_{T\to T_1}(\sigma)) = \Gamma_{T\to T_1\cap T_2} \in \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2))$. Therefore, we have $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1)) \subseteq \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2))$. Similarly, we can prove that $\Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2)) \subseteq \Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1))$. As a result, $\Gamma_{T_1\to T_1\cap T_2}(L(\Sigma_1)) \subseteq \Gamma_{T_2\to T_1\cap T_2}(L(\Sigma_2))$.

(Necessity) $\forall \sigma_i \in L(\Sigma_1)$, we denote $\sigma_{\Delta i} = \Gamma_{T_1 \to T_1 \cap T_2}(\sigma_i)$. Because $\Gamma_{T_1 \to T_1 \cap T_2}(L(\Sigma_1)) = \Gamma_{T_2 \to T_1 \cap T_2}(L(\Sigma_2))$, we have $\sigma_{\Delta i} \in \Gamma_{T_2 \to T_1 \cap T_2}(L(\Sigma_2))$, i.e., $\exists \sigma_j \in L(\Sigma_2)$ such that $\Gamma_{T_2 \to T_1 \cap T_2}(\sigma_j) = \sigma_{\Delta i}$. Let $\sigma = \{\omega | \omega \in T^* \wedge (\Gamma_{T \to T_1} = \sigma_i) \wedge (\Gamma_{T \to T_2} = \sigma_j)\}$, and we have $\Gamma_{T \to T_1}(\sigma) = \sigma_i \in L(\Sigma_1)$ and $\Gamma_{T \to T_2}(\sigma) = \sigma_j \in L(\Sigma_2)$, and thus $L(\Sigma_1) \subseteq \Gamma_{T \to T_1(L(\Sigma_{TS}))}$. Similarly, we can prove that $L(\Sigma_2) \subseteq \Gamma_{T \to T_2(L(\Sigma_{TS}))}$. Therefore, $L(\Sigma_i) \subseteq \Gamma_{T \to T_i}(L(\Sigma_{TS}))(i \in \{1, 2\})$. $\Gamma_{T \to T_i}(L(\Sigma_{TS})) \subseteq L(\Sigma_i)(i \in \{1, 2\})$ can be proved in the same way. As a result, $\Gamma_{T \to T_i}(L(\Sigma_{TS})) = L(\Sigma_i)(i \in \{1, 2\})$. ■

Theorem 4 reveals how to determine the behavioral invariance between two workflow processes. Specially, it holds when the synchronization transition set contains only one transition.

Theorem 5: Σ_1 and Σ_2 are behaviorally invariant iff $|T_1 \cap T_2| = 1$.

Proof: As $|T_1 \cap T_2| = 1$, let us assume that $|T_1 \cap T_2| = \{\alpha\}$, then we have $\Gamma_{T_1 \to T_1 \cap T_2}(L(\Sigma_1, R(M_{01}))) = \Gamma_{T_2 \to T_1 \cap T_2}(L(\Sigma_2, R(M_{02}))) = \alpha$. According to Theorem 4, Σ_1 and Σ_2 are behaviorally invariant. As a result, we prove that Σ_1 and Σ_2 are behaviorally invariant when $|T_1 \cap T_2| = 1$.

Generally speaking, to determine the correctness of a CWF-net, we need to check if all its sub-organization processes are correct and if all its sub-organizations are behaviorally invariant. To verify the correctness of each organization, we can resort to both the reachability graph-based analysis approach and the behavioral characterization-based one. Moreover, Theorem 4 can be used to check if all sub-organizations are behaviorally invariant. It is worth noting that the reachability graph-based approach is not applicable to check the correctness of the global cross-organizational workflow when considering the business process privacy issue.

Consider the CWF-net in Fig. 4 as an example. We have the full language of Σ_3 as $L(\Sigma_3, R(M_0)) = Pref(t_a(t_b + t_c)t_s t_d)$, the sink language of Σ_3 is $L(\Sigma_3, M_e) = t_a(t_b + t_c)t_s t_d$, the full language of Σ_4 is $L(\Sigma_4, R(M_0)) = Pref(t_1(((t_2 + t_3)t_s) \| ((t_4 t_5)^* t_6)t_7)$ and the sink language of Σ_4 is $L(\Sigma_4, M_e) = t_1(((t_2 + t_3)t_s) \| ((t_4 t_5)^* t_6)t_7)$. Therefore, $L(\Sigma_3, R(M_0)) = Pref(L(\Sigma_3, M_e))$ and $L(\Sigma_4, R(M_0)) = Pref(L(\Sigma_4, M_e))$. In this way, Σ_3 and Σ_4 are proved to be correct. In addition, $\Gamma_{T_3 \to T_1 \cap T_2}(L(\Sigma_1, M_e)) = t_s$ and $\Gamma_{T_4 \to T_3 \cap T_4}(L(\Sigma_4, M_e)) = t_s$. Thus, we have $\Gamma_{T_3 \to T_3 \cap T_4}(L(\Sigma_3)) = \Gamma_{T_4 \to T_3 \cap T_4}(L(\Sigma_4))$. As a result, Σ_3 and Σ_3 are proved to be behaviorally invariant. According to Theorem 4, we can conclude that CWF-net in Fig. 4 is correct.

To simplify this verification process, we have presented Theorem 5 for some special scenario such that $|T_3 \cap T_3| = 1$. As $|T_3 \cap T_3| = \{t_s\}$, for the CWF-net in Fig. 4, we can easily prove that it is correct by using Theorem 5.

VI. Running Case Analysis

For the purpose of illustrating the effectiveness and efficiency of our proposed approaches, a cross-organizational medical diagnosis scenario is used as a case study. This section is structured as follows. Basic information of the medical diagnosis scenario is first introduced in Section VI-A. Without considering the business privacy of each organization, the correctness of this case is first verified by the reachability-based techniques in Section VI-B. Finally, Section VI-C checks the correctness based on the proposed privacy-preserving techniques.

A. A Cross-department Medical Diagnosis Business Process

With the progress of medical informatization, medical diagnosis businesses are crossing department boundaries to meet the diagnosis information integration requirements. A typical cross-department medical diagnosis business process scenario may involve following departments: the surgical department, cardiovascular department, X-ray department, charge office, and pharmacy. To save space, we only describe the businesses relevant to the surgical department and the cardiovascular department in detail. More details can be founded in [48].

This scenario is described in the following. When a patient arrives at a hospital, an outpatient medical staff helps with the pre-examination triage, and then the patient information is generated according to the pre-examination results and sent to the surgical medical staff for further use. Next, a surgical medical staff takes admissions, and presents the reservation application, and then generates the reservation form. After, the surgeon diagnoses the patient and determines if a consultation with the cardiovascular internists is needed. In case a consultation is required, the surgeon applies consultation by sending a consultation form to the cardiovascular internists. Otherwise, the surgeon gives a prescription on the basis of the patients symptom. After that, the internists receive the consultation request and start the consultation. Following, the surgeon and internists conduct the consultation, and give a prescription together. Finally, the internists make the consultation summary and the patient takes the medicine before leaving.

Based on the medical business process descriptions, we have the following observations: (1) the medical diagnosis scenario mainly involves two departments where each department has its respective medical process and task set. More specifically, the surgical department contains tasks, such as admission, and reservation application, diagnosis, applying for consultation and giving prescription. The cardiovascular department mainly consists of tasks like receiving a consultation request, consultation arrangement, and consultation summary; and (2) these two departments need to collaborate to finish certain tasks, i.e., surgical and cardiovascular departments need to collaborate via tasks consultation and giving prescription. Thus, there exists task synchronization patterns between them.

According to the medical business description, we obtain its WF-nets of the surgical department (Σ_S) and cardiovascular

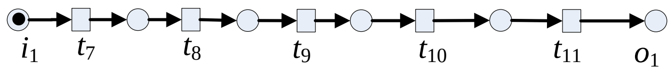


Fig. 6. WF-net of Cardiovascular Department (Σ_C)

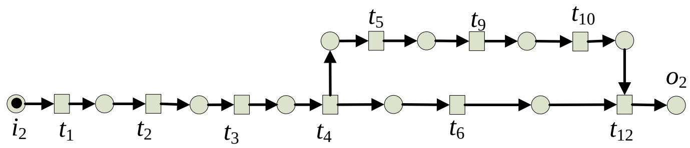


Fig. 7. WF-net of Surgical Department (Σ_S)

TABLE II
MEANING OF TRANSITIONS OF Σ_S AND Σ_C

Transition Name	Meaning
t_1	Pre-examination triage
t_2	Admissions
t_3	Reservation application
t_4	Diagnosis
t_5	Consultation application
t_6	Give prescription
t_7	Receive consultation request
t_8	Consultation arrangement
t_9	Consultation
t_{10}	Give prescription together
t_{11}	Consultation summary
t_{12}	Take medicine

department (Σ_C) which is shown in Figs. 6-7 and the specific meaning of each transition is depicted in Table II.

B. Reachability-based Correctness Verification

Without considering the business privacy of each organization, we check the correctness of the medical process by the reachability-based techniques for workflow nets. According to Section VI-A, the surgical department and the cardiovascular department need to collaborate to finish tasks t_9 and t_{10}, i.e., there exists task synchronization patterns between them.

Fig. 8 illustrates the CWF-net denoted as Σ_{CS} where Σ_C and Σ_S are connected by their synchronization tasks (transitions) t_9 and t_{10}.

To perform the correctness verification of a Σ_{CS} based on reachability-based technique, we construct its corresponding

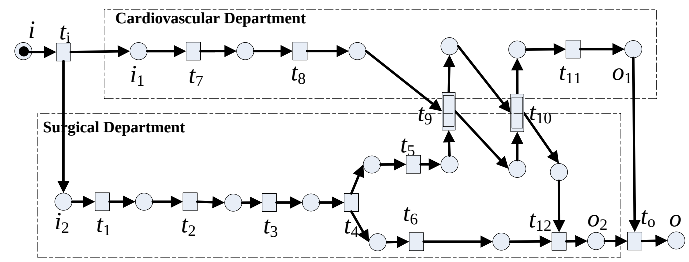


Fig. 9. UWF-net of the Cross-department Medical Diagnosis Process(Σ_{UCS})

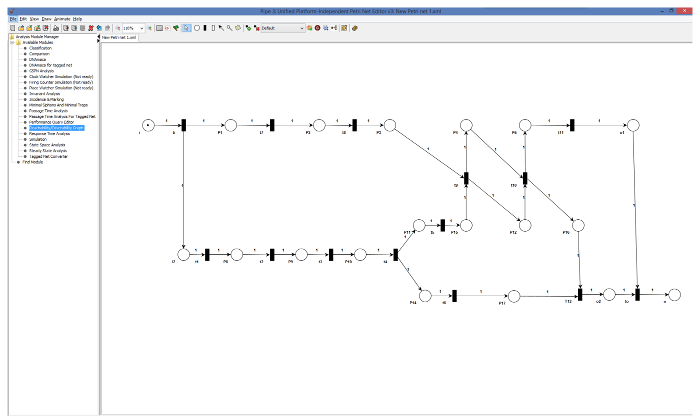


Fig. 10. Enacted UWF-net on the PIPE 3.0 Platform

UWF-net, denoted as Σ_{UCS} as shown in Fig. 9. In Σ_{UCS}, Σ_C and Σ_S are connected by a source transition t_i and a sink transition t_o. In addition, source place i is added to initialize Σ_{UCS} and sink place o is used to represent the termination of Σ_{UCS}. The synchronization tasks, i.e. t_9 and t_{10}, are regarded as ordinary transitions. It is proved by Theorem 1 that Σ_{UCS} is also a standard WF-net.

According to Definition 10, to verify the correctness of the cross-department medical diagnosis process, we need to check the correctness of Σ_{UCS}, Σ_C and Σ_S via reachability graph-based analysis approach. Specifically, we used PIPE 3.0[1] to perform the verification process. The corresponding Σ_{UCS} in PIPE 3.0 is enacted as illustrated in Fig. 10. Based on simulation and verification analysis, a reachability graph

[1]http://sourceforge.net/projects/pipe3/

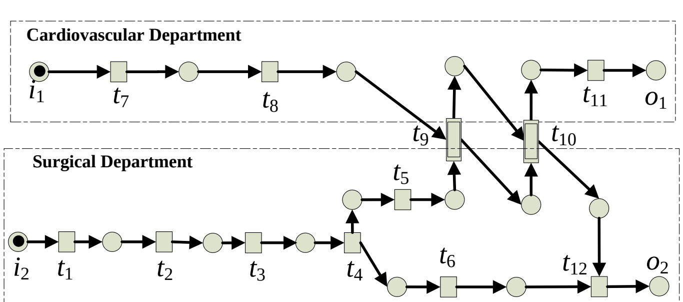


Fig. 8. CWF-net of the Cross-department Medical Diagnosis Process (Σ_{CS})

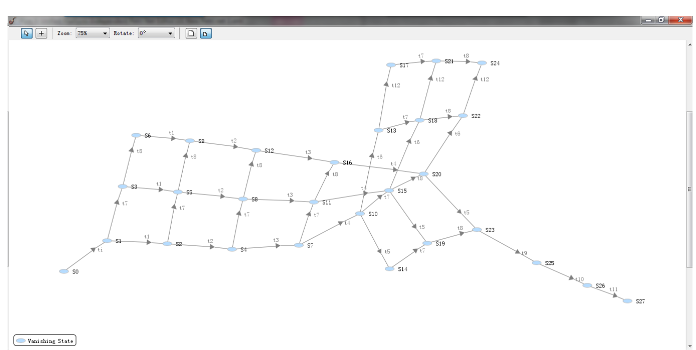


Fig. 11. Reachability Graph of the UWF-net in Fig. 10

by running the PIPE 3.0 tool is generated and shown in Fig. 11. By taking as input the reachability graph, we check the three conditions in Definition 4. Therefore, we prove that the model in Fig. 9 is correct.

Unfortunately, traditional reachability analysis based correctness verification method is not practical to deal with the cross-organizational workflow as its global business process (i.e., Σ_{UCS}) is unavailable when considering the business privacy of each organizations. Therefore, we have to resort to other privacy-preserving correctness verification approaches.

C. Behavioral Correctness Verification

Because each organization is not willing to share their inner business details with other (competitive) organizations or a third-party organization, all we have are the information about the correctness of each organization and interactions among them. Therefore, the correctness of the global cross-organization business processes cannot be verified by constructing their reachability graphs directly. In this way, we verify the correctness using our proposed approaches.

Note that local correctness of each sub-organization process model and their interaction patterns are available, therefore, we can use them to imply its global correctness. To determine the correctness of a CWF-net, we need to check if all its sub-organization processes are correct and if all its sub-organizations are behaviorally invariant. To verify the correctness verification of each organization, we can resort to both the reachability graph based analysis approach and the behavioral characterization based one. Moreover, Theorem 4 can be used to check if all sub-organizations are behaviorally invariant. The detailed behavior-based verification process is discussed in the following.

Based on Definition 10, we need to check the correctness of Σ_C, Σ_S and Σ_{UCS} to accomplish the correctness verification of Σ_{CS}. According to Theorem 1, to determine the correctness of Σ_C and Σ_S, we need to test if their whole language and the prefix language of their sink language are equal. According to Figs. 5-6, we obtain that the full language of Σ_C is $L(\Sigma_C, R(M_0)) = Pref(t_7 t_8 t_9 t_{10} t_{11})$ and the sink language of Σ_C is $L(\Sigma_C, M_e) = t_7 t_8 t_9 t_{10} t_{11}$, thereby $L(\Sigma_C, R(M_0)) = Pref(L(\Sigma_C, M_e))$. Similarly, we obtain the whole language of Σ_S is $L(\Sigma_S, R(M_0)) = Pref(t_1 t_2 t_3 t_4 ((t_6 \| (t_5 t_9 t_{10})) t_{12})$ and the sink language of Σ_S is $L(\Sigma_S, M_e) = t_1 t_2 t_3 t_4 ((t_6 \| (t_5 t_9 t_{10})) t_{12}$, thereby $L(\Sigma_S, R(M_0)) = Pref(L(\Sigma_S, M_e))$. Therefore, we prove that Σ_C and Σ_S are correct via behavioral characterization approach. In addition, we have $\Gamma_{TC \to TC \cap TS}(L(\Sigma_C, M_e)) = t_9 t_{10}$ and $\Gamma_{TS \to TC \cap TS}(L(\Sigma_S, M_e)) = t_9 t_{10}$, i.e. $\Gamma_{TC \to TC \cap TS}(L(\Sigma_C)) = \Gamma_{TS \to TC \cap TS}(L(\Sigma_S))$. As a result, Σ_C and Σ_S are proved to be behaviorally invariant according to Theorem 4. Finally, Σ_{CS} is proved to be behaviorally correct. In summary, the result using our proposed behavioral correctness verification approach is consistent with the one using reachability graph-based analysis. In addition, our proposed approach is capable of preserving the business privacy of each organization.

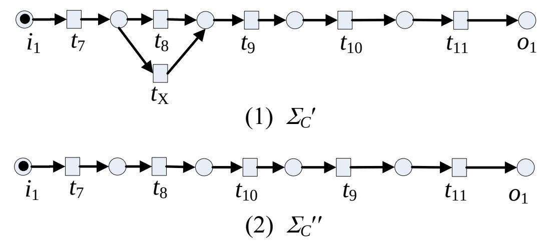


Fig. 12. Two Variants of Σ_C

To show the generality of our approach, we discuss another two cases. In the first case, we give Σ'_C (in Fig. 12 (a)) that is a behavioral equivalent variant of Σ_C, i.e., we have $\Gamma_{TC' \to TC' \cap TS}(L(\Sigma'_C, M_e)) = t_9 t_{10}$ and $\Gamma_{TC' \to TC' \cap TS}(L(\Sigma'_C)) = \Gamma_{TS \to TC \cap TS}(L(\Sigma_S))$. Therefore, the CWF-net that connects Σ'_C and Σ_S by synchronization transitions t_9 and t_{10} is behaviorally correct. In the second case, we give Σ''_C that is a correct but not behavioral equivalent variant of Σ_C, i.e., we have $\Gamma_{TC'' \to TC'' \cap TS}(L(\Sigma''_C, M_e)) = t_{10} t_9$ and $\Gamma_{TC'' \to TC'' \cap TS}(L(\Sigma'_C)) \neq \Gamma_{TS \to TC \cap TS}(L(\Sigma_S))$. Therefore, the CWF-net that connects Σ''_C and Σ_S by transitions t_9 and t_{10} is not behaviorally correct.

VII. Conclusion

In this paper, we established the correctness verification approach for cross-organizational workflows with task synchronization patterns. This paper has two main contributions: (1) behavioral characterization of a sound WF-net is given based on Petri net languages; and (2) behaviorally correctness verification of cross-organizational workflow with task synchronization patterns performed via an example. Our proposed approach can be used as a key technique for investigating coordination mechanisms among cross-organizational worflows. For instance, it is applicable to develop a set of practical guidelines for synthesizing sub-organizational workflows to perform collaborative tasks, which contributes to the characterization of a class of cross-organizational workflows whose local soundness implies global soundness.

This paper is limited to control-flow perspective of processes, other aspects, e.g., resources and time, are not covered. To support collaborative processes, we use task synchronization patterns to connect loosely-coupled processes. Other interaction patterns, e.g., message exchange and sub-process sub-contracting patterns, are currently not supported. Our approaches towards behavioral correctness verification are focused on task synchronization patterns. In the future, we plan to extend the results for other interaction patterns, e.g., message interaction pattern, resource sharing pattern, and sub-process sub-contracting patterns [48] and [25]. In addition, we would like to incorporate resource, time, and data constraints in our approach to check the behavioral correctness from a broad perspective. Considering for example the data perspective, given two organizations that synchronize initially but have the (optional) second synchronization this can be sound iff both organizations are aware of when the second synchronization is required. This requires an understanding of data-aware workflow [20] and [21] where data elements imply the condition used.

REFERENCES

[1] C. Betous-Almeida and K. Kanoun, "Construction and stepwise refinement of dependability models," *Performance Evaluation*, vol. 56, no. 1, pp. 277–306, 2004.

[2] I. Chebbi, S. Dustdar, and S. Tata, "The view-based approach to dynamic inter-organizational workflow cooperation," *Data & Knowledge Engineering*, vol. 56, no. 2, pp. 139–173, 2006.

[3] J. Cheng, C. Liu, M. Zhou, Q. Zeng, and A. Ylä-Jääski, "Automatic composition of semantic web services based on fuzzy predicate Petri nets," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 2, pp. 680–689, 2015.

[4] Z. Ding, C. Jiang, M. Zhou, and Y. Zhang, "Preserving languages and properties in stepwise refinement-based synthesis of Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 38, no. 4, pp. 791–801, 2008.

[5] Z. Ding, Y. Zhang, C. Jiang, and Z. Zhang, "Refinement of Petri nets in workflow integration," in *International Conference on Computer Supported Cooperative Work in Design*. Springer, 2006, pp. 667–678.

[6] 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. 55 573–55 585, 2018.

[7] ——, "Refinement-based hierarchical modeling and correctness verification of cross-organization collaborative emergency response processes," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, no. 99, pp. 1–15, 2018.

[8] T. Ertay, D. Ruan, and U. R. Tuzkaya, "Integrating data envelopment analysis and analytic hierarchy for the facility layout design in manufacturing systems," *Information Sciences*, vol. 176, no. 3, pp. 237–262, 2006.

[9] R. Eshuis and P. Grefen, "Constructing customized process views," *Data & Knowledge Engineering*, vol. 64, no. 2, pp. 419–438, 2008.

[10] W. Labda, N. Mehandjiev, and P. Sampaio, "Privacy-aware business processes modeling notation (prvbpmn) in the context of distributed mobile applications," in *Trends in Mobile Web Information Systems*. Springer, 2013, pp. 120–134.

[11] ——, "Modeling of privacy-aware business processes in bpmn to protect personal data," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. ACM, 2014, pp. 1399–1405.

[12] J. Li, X. Meng, and X. Dai, "Collision-free scheduling of multi-bridge machining systems: a colored traveling salesman problem-based approach," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 139–147, 2018.

[13] Q. Li, Y. Deng, C. Liu, Q. Zeng, and Y. Lu, "Modeling and analysis of subway fire emergency response: an empirical study," *Safety science*, vol. 84, pp. 171–180, 2016.

[14] P. Linz, *An introduction to formal languages and automata*. Jones & Bartlett Publishers, 2011.

[15] C. Liu, "Automatic discovery of behavioral models from software execution data," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1897–1908, 2018.

[16] ——, "Hierarchical business process discovery: Identifying sub-processes using lifecycle information," in *International Conference on Web Services*. IEEE, 2020, pp. 1–5.

[17] C. Liu, J. Cheng, Y. Wang, and S. Gao, "Time performance optimization and resource conflicts resolution for multiple project management," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 3, pp. 650–660, 2016.

[18] C. Liu, H. Duan, Z. Qingtian, M. Zhou, F. Lu, and J. Cheng, "Towards comprehensive support for privacy preservation cross-organization business process mining," *IEEE Transactions on Services Computing*, vol. 12, no. 4, pp. 639–653, 2019.

[19] C. Liu, B. van Dongen, N. Assy, and W. M. van der Aalst, "Component behavior discovery from software execution data," in *Computational Intelligence (SSCI), 2016 IEEE Symposium Series on*. IEEE, 2016, pp. 1–8.

[20] C. Liu, Q. Zeng, H. Duan, and F. Lu, "Petri net based behavior description of cross-organization workflow with synchronous interaction pattern," in *International Workshop on Process-Aware Systems*. Springer, 2014, pp. 1–10.

[21] C. Liu, Q. Zeng, H. Duan, L. Wang, J. Tan, C. Ren, and W. Yu, "Petri net based data-flow error detection and correction strategy for business processes," *IEEE Access*, vol. 8, no. 1, pp. 43 265 – 43 276, 2020.

[22] C. Liu, Q. Zeng, H. Duan, M. Zhou, F. Lu, and J. Cheng, "E-net modeling and analysis of emergency response processes constrained by resources and uncertain durations," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 1, pp. 84–96, 2015.

[23] C. Liu, Q. Zeng, D. Hua *et al.*, "Formulating the data-flow modeling and verification for workflow: A Petri net based approach," *International Journal of Science and Engineering Applications*, vol. 3, pp. 107–112, 2014.

[24] C. Liu, Q. Zeng, J. Zou, F. Lu, and Q. Wu, "Invariant decomposition conditions for Petri nets based on the index of transitions," *Information Technology Journal*, vol. 11, no. 7, pp. 768–774, 2012.

[25] C. Liu and F. Zhang, "Petri net based modeling and correctness verification of collaborative emergency response processes," *Cybernetics and Information Technologies*, vol. 16, no. 3, pp. 122–136, 2016.

[26] C. Liu, J. Zhang, G. Li, S. Gao, and Q. Zeng, "A two-layered framework for the discovery of software behavior: A case study," *IEICE Transactions on Information and Systems*, 2017.

[27] G. Liu, C. Jiang, M. Zhou, and P. Xiong, "Interactive Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 2, pp. 291–302, 2013.

[28] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[29] J. L. Peterson, "Petri nets," *ACM Computing Surveys (CSUR)*, vol. 9, no. 3, pp. 223–252, 1977.

[30] W. Reisig, *Petri nets: an introduction*. Springer Science & Business Media, 2012, vol. 4.

[31] ——, "Associative composition of components with double-sided interfaces," *Acta Informatica*, vol. 56, no. 3, pp. 229–253, 2019.

[32] S. A. Reveliotis, *Real-time management of resource allocation systems: A discrete event systems approach*. Springer Science & Business Media, 2006, vol. 79.

[33] W. Sadiq and M. E. Orlowska, "Analyzing process models using graph reduction techniques," *Information systems*, vol. 25, no. 2, pp. 117–134, 2000.

[34] K. A. Schulz and M. E. Orlowska, "Facilitating cross-organisational workflows with a workflow view approach," *Data & Knowledge Engineering*, vol. 51, no. 1, pp. 109–147, 2004.

[35] A. Tahamtan and J. Eder, "Privacy preservation through process views," in *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on*. IEEE, 2010, pp. 1100–1107.

[36] W. M. van der Aalst, "Verification of workflow nets," in *International Conference on Application and Theory of Petri Nets*. Springer, 1997, pp. 407–426.

[37] ——, "The application of Petri nets to workflow management," *Journal of circuits, systems, and computers*, vol. 8, no. 01, pp. 21–66, 1998.

[38] ——, "Process-oriented architectures for electronic commerce and interorganizational workflow," *Information systems*, vol. 24, no. 8, pp. 639–671, 1999.

[39] ——, "Loosely coupled interorganizational workflows:: modeling and analyzing workflows crossing organizational boundaries," *Information & Management*, vol. 37, no. 2, pp. 67–75, 2000.

[40] W. M. van der Aalst, P. Barthelmess, C. A. Ellis, and J. Wainer, "Proclets: A framework for lightweight interacting workflow processes," *International Journal of Cooperative Information Systems*, vol. 10, no. 04, pp. 443–481, 2001.

[41] W. M. van der Aalst, J. Desel, and A. Oberweis, "Business process management models, techniques and empirical studies." 2000.

[42] H. Verbeek, W. M. van der Aalst, and A. Kumar, "Xrl/woflan: Verification and extensibility of an xml/Petri-net-based language for inter-organizational workflows," *Information Technology and Management*, vol. 5, no. 1, pp. 65–110, 2004.

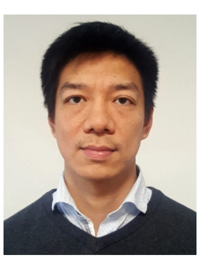
[43] D. Xiang, G. Liu, C. Yan, and C. Jiang, "Detecting data inconsistency based on the unfolding technique of petri nets," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 2995–3005, Dec 2017.

[44] P. Xiong, Y. Fan, and M. Zhou, "A Petri net approach to analysis and composition of web services," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 2, pp. 376–387, 2010.

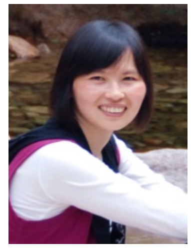
[45] F. Yang, N. Wu, Y. Qiao, and R. Su, "Polynomial approach to optimal one-wafer cyclic scheduling of treelike hybrid multi-cluster tools via Petri nets," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 270–280, Jan 2018.
[46] Q. Zeng, H. Duan, and C. Liu, "Top-down process mining from multi-source running logs based on refinement of Petri nets," *IEEE Access*, vol. 8, pp. 61 355–61 369, 2020.
[47] Q. Zeng, C. Liu, and H. Duan, "Resource conflict detection and removal strategy for nondeterministic emergency response processes using Petri nets," *Enterprise Information Systems*, vol. 10, no. 7, pp. 729–750, 2016.
[48] Q. Zeng, F. Lu, C. Liu, H. Duan, and C. Zhou, "Modeling and verification for cross-department collaborative business processes using extended Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 2, pp. 349–362, 2015.
[49] Q. Zeng, S. X. Sun, H. Duan, C. Liu, and H. Wang, "Cross-organizational collaborative workflow mining from a multi-source log," *Decision Support Systems*, vol. 54, no. 3, pp. 1280–1301, 2013.
[50] M. C. Zhou, F. DiCesare, and D. L. Rudolph, "Design and implementation of a Petri net based supervisor for a flexible manufacturing system," *Automatica*, vol. 28, no. 6, pp. 1199–1208, 1992.
[51] M. Zhou, F. DiCesare, and A. A. Desrochers, "A hybrid methodology for synthesis of Petri net models for manufacturing systems," *IEEE transactions on robotics and automation*, vol. 8, no. 3, pp. 350–361, 1992.
[52] M. Zhou and K. Venkatesh, *Modeling, simulation, and control of flexible manufacturing systems: a Petri net approach*. World Scientific, 1999, vol. 6.

Cong Liu received the BS and the MS degree in computer software and theory from Shandong University of Science and Technology, Qingdao, China, in 2013 and 2015 respectively. He received the PhD degree in the Section of Information Systems (IS) in the Department of Mathematics and Computer Science, Eindhoven University of Technology. He is a full Professor in Shandong University of Technology, in 2019. His research interests are in the areas of business process management, process mining, Petri nets, and software process mining.

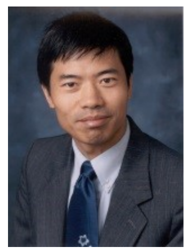
Qingtian Zeng received the BS degree and the MS degree in computer science from Shandong University of Science and Technology, Taian, China, in 1998 and 2001 respectively, and the PhD degree in computer software and theory from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005. He is currently a Professor with Shandong University of Science and Technology, Qingdao, China. His research interests are in the areas of Petri nets, process mining, and knowledge management.



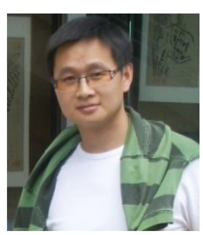
Long Cheng is an Assistant Professor in the School of Computing at Dublin City University, Ireland. He received the B.E. from Harbin Institute of Technology, China in 2007, M.Sc from University of Duisburg-Essen, Germany in 2010 and Ph.D from National University of Ireland Maynooth in 2014. He was a Marie Curie Fellow. He has worked at organizations such as Huawei Technologies, IBM Research, TU Dresden, TU Eindhoven and UCD. His research focuses on high-performance data analytics, distributed systems, and process mining.



Hua Duan received the BS degree and the MS degree in applied mathematics from Shandong University of Science and Technology, Taian, China, in 1999 and 2002, and the PhD degree in applied mathematics from Shanghai Jiaotong University, in 2008. She is a professor at Shandong University of Science and Technology. Her research interests are in the areas of Petri nets, business process management and machine learning.



MengChu Zhou (S88-M90-SM93-F03) received the BS degree in control engineering from Nanjing University of Science and Technology, Nanjing, China, in 1983, the MS degree in automatic control from Beijing Institute of Technology, Beijing, China, in 1986, and the PhD degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, New York, in 1990. He joined New Jersey Institute of Technology (NJIT), Newark, NJ, in 1990, and is now a distinguished professor of electrical and computer engineering. His research interests include Petri nets, internet of things, big data, web service, semiconductor manufacturing, and transportation systems. He has more than 800 publications including 12 books, 500+ journal papers (400+ in IEEE Transactions), and 29 bookchapters. He is the founding editor of IEEE Press Book Series on Systems Science and Engineering. He was the recipient of Humboldt Research Award for US Senior Scientists, Franklin V. Taylor Memorial Award and the Norbert Wiener Award from IEEE Systems, Man and Cybernetics Society, and Distinguished Service Award from IEEE Robotics and Automation Society. He is a life member of Chinese Association for Science and Technology-USA and served as its President in 1999. He is a fellow of International Federation of Automatic Control (IFAC) and American Association for the Advancement of Science.



Jiujun Cheng received the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2006. He is currently a Professor with Tongji University, Shanghai, China. In 2009, he was a Visiting Professor with Aalto University, Espoo, 1239 Finland. He has over 50 publications including conference and journal papers. His current research interests include mobile computing, complex networks with a focus on mobile/Internet interworking, service computing, and Internet of Vehicles.