

## Research Article

# Two Effective Strategies to Support Cross-Organization Emergency Resource Allocation Optimization

Ying Gao,<sup>1</sup> Cong Liu ,<sup>2</sup> Qingtian Zeng ,<sup>1</sup> and Hua Duan<sup>1</sup>

<sup>1</sup>Department of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China

<sup>2</sup>School of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China

Correspondence should be addressed to Cong Liu; [liucongchina@sdust.edu.cn](mailto:liucongchina@sdust.edu.cn) and Qingtian Zeng; [qtzeng@163.com](mailto:qtzeng@163.com)

Received 20 February 2020; Revised 8 May 2020; Accepted 25 November 2020; Published 4 January 2021

Academic Editor: Carlos T. Calafate

Copyright © 2021 Ying Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cross-organization emergency resource allocation optimization problem is essential to guarantee a successful emergency disposal, and it has become a research focus of modern emergency management. Generally speaking, there are two possible types of resource allocation scenarios: (1) if the emergency resources are overallocated, on the one hand, parallel execution of independent emergency activities can be supported and the emergency disposal time is reduced; on the other hand, too many idle resources may cause low resource utilization rate, high scheduling overhead, and high cost; and (2) if emergency resources are under-allocated, this may lead to resource conflicts and the need for some emergency activities to wait for others to complete, and finally the emergency disposal time may increase. Therefore, reasonable emergency resource allocation strategies are highly desired. To the best of our knowledge, there is no formal approach to support the cross-organization emergency resource allocation issue. To handle this problem, we propose a two-layered framework to facilitate the allocation of limited emergency resources to meet its time constraints with high efficiency. More specifically, a kind of Petri net extended with time, resource, and message information, denoted as CE-net, is presented to model cross-organization emergency response processes. Based on the obtained CE-net, the minimum resource requirements are obtained with corresponding algorithms. Then, Minimum Execution Time (MET) strategy and Minimum Resource Consumption (MRC) strategy with their corresponding estimated execution intervals are introduced to facilitate the stakeholder to determine which strategy is suitable according to the timing requirements. A cross-organization fire emergency case is applied to validate the proposed approaches throughout the whole paper.

## 1. Introduction

The allocation of distributed emergency resources has been a challenging issue in the cross-organization emergency management area. An emergency is defined as a situation that imposes immediate risk to life, property, and environment, which requires urgent disposal and intervention to prevent its worsening [1, 2]. These disposals are normally organized as a group of collaborated emergency response processes charged by an emergency command center and several subordinate emergency organizations that need to work together to accomplish the whole emergency mission [3].

Generally speaking, cross-organization emergency response processes usually exhibit the following features (or

requirements) that differentiate themselves from traditional business processes [1, 4–6]: (1) There are close relations between emergency activities and resources. During the execution of an emergency response process, large quantities of resources, for example, ambulances, fire trucks, medicines, tents, food, and clothing, are badly needed. On the one hand, if the emergency resources are overallocated, parallel execution of independent emergency activities can be enabled to reduce the execution time; however, too many idle resources may lead to low resource utilization rate, high scheduling overhead, and even high cost. On the other hand, if the emergency resources are underallocated, this may lead to resource conflicts and the need for some activities to wait for others to complete, and finally the execution time of the emergency response process may increase. Consequently,

reasonable emergency resource allocation strategies are highly desired. (2) There are a lot of collaborations that need either messages sent by other organizations or resources shared among different organizations. Hence, effective collaboration modeling for this case is required. (3) Cross-organization emergency response processes are a real-time service where timeliness is critical to its mission success. Therefore, time performance evaluation and optimization strategies are needed.

Modeling across-organization emergency response processes is a complicated and time-consuming work. In addition, optimized emergency resource allocation will benefit both successful emergency response process execution with high time performance and high resource utilization rate. Therefore, emergency resources should be optimized and allocated properly at least from the two following perspectives: (1) minimizing emergency resource requirement as much as possible to achieve a high resource utilization rate and (2) shortening emergency response time as much as possible to achieve its crucial timing constraints. Note that this problem is different from conventional workflow resource scheduling [7–12], which aims at minimizing the application’s runtime on a limited resource set. Differently, we try to propose effective strategies to support cross-organization emergency resource allocation optimization. To this end, we introduce a kind of Petri net [13–15] extended with time, resource, and message information, CE-net for short, to model cross-organization emergency response processes. Actually, the CE-net is an extension of our previous E-net model [1] to suit cross-organization scenario. Accurately, a time interval is used to predict the uncertain execution time of an emergency activity. In addition, emergency resources are divided as reusable and consumable ones; that is, the reusable resources can be reused by other emergency activities when released, while the consumable ones can be used only once and cannot be reused any more.

The main contributions of this paper are summarized as follows: (1) a kind of Petri net extended with time, resource, and message information, CE-net for short, is proposed to model cross-organization emergency response processes; (2) a two-layered framework is proposed to support cross-organization emergency resource allocation; and (3) we present two effective resource allocation strategies, that is, the minimum execution time strategy and the minimum resource consumption strategy, to support the efficient emergency resource allocation decision-making based on the resource requirements analysis results.

The remainder of this paper is structured as follows. Section 2 discusses the related work. In Section 3, a cross-organization fire emergency response scenario is given as a case to validate our proposed approaches. Section 4 investigates the modeling approaches for cross-organization emergency response processes using CE-net. Then, minimum emergency resource requirements analysis is investigated based on the CE-net in Section 5. Section 6 introduces two effective resource allocation strategies to support the emergency resource allocation decision-making. Finally, Section 7 draws concluding remarks.

## 2. Related Work

In this section, we mainly review existing work related to (1) workflow resource allocation as well as emergency resource scheduling and (2) modeling of cross-organizational workflow and emergency response processes.

*2.1. Workflow Resource Allocation and Emergency Resource Scheduling.* Workflow scheduling is a well-known problem whose main objective is to minimize the completion time of a workflow using a given (or limited) set of resources. Deelman proposes one of the most popular algorithms, called Pegasus, to schedule workflows on distributed systems [7]. It is built on top of the list scheduling approach and considers both communication and computation cost. Based on experimental evaluation, it is demonstrated that Pegasus achieves relatively good performance for most cases. Besides the list scheduling approach, other techniques, such as DAG-based scheduling [8], greedy randomized adaptive search [9], and critical path first [10], have also been investigated. More recently, Byun proposes a new algorithm named BTS for estimating the minimum number of hosts that are needed to execute a workflow within a given deadline to bridge the gap between workflow management systems and resource provisioning systems [11, 12].

As authoritative emergency management experts, Tufekci and Wallace [16] consider emergency management as a complex multiobjective optimization problem and utilizing limited resources compromisingly is the best way to be adopted. Following their thoughts, emergency resource allocation optimization has been studied by many researchers in the past decades. A dynamic optimization model is used to find the best assignment of available resources to operational areas by Fiedrichet et al. [17]. In [18], by considering multiple disaster places and multiple resource suppliers, an optimized algorithm is proposed to solve resource collision problem for large-scale public emergency response. In [19], the problem of allocating multiple emergency service resources to protect critical transportation infrastructures is studied. Different modeling approaches, including deterministic, stochastic programming and robust optimization, are used to model various risk preferences in decision-making under uncertain service availability and accessibility. To find an optimal solution for resource deployment and dispatching, Kondaveti and Ganz [20] introduce a decision support framework built on rapid information collection and resource tracking functionalities. The equipment control structure presented in [21] enables decentralized and collective decision-making for equipment prioritization and distribution in response to disasters. More recently, by considering time performance, Wang et al. [22] discuss a negotiation strategy and a compromised resources allocation model for emergency response. The negotiation strategy can facilitate requesters to find out the demanded resources efficiently. In addition, a mathematical model is presented to compute the earliest start time of emergency response on condition of continuous requirement for resources.

*2.2. Modeling of Cross-Organizational Workflow and Emergency Response Processes.* Van der Aalst first considers workflows distributed over a number of organizations in [23, 24], where two important questions are addressed: (1) the minimal requirements of interorganizational workflow and (2) how to decide if an interorganizational workflow, modeled with Petri nets, is consistent with an interaction structure specified through a message sequence. In [25], Liu et al. propose a kind of interactive Petri net to model the message channels between different process-oriented systems, and the compatibility preservation of an integrated system with message interaction is revealed. Differently, Schulz and Orłowska focus on three aspects to support the execution of cross-organizational workflows that have been modeled with a process-view approach in [26]: (1) communication between the entities, (2) their impact on an extended workflow engine, and (3) the design of cross-organizational workflow architecture. A Petri net-based state transition approach that binds states of private workflow tasks to their adjacent workflow view is introduced. Its concepts are demonstrated by a scenario involving two workflow management systems. Jiang et al. [27] describe a timed colored Petri net and process-view combined approach to construct cross-organizational workflows, and a three-layered framework is proposed to realize the interoperability of cross-organizational workflows.

In [28], we investigate the application of process mining for workflow integration using a type of Petri nets extended with resource and message factors. A process integration-based approach is presented to obtain the model for a cross-organization workflow based on the model mined for each organization and the coordination patterns among different organizations. More recently, we formally define several collaboration patterns, including message interaction pattern, resource interaction pattern, task collaboration pattern, and service outsourcing pattern, in [29]. Modeling and analysis of a cross-department medical workflow are effectively supported, and the correctness is verified by building the reachability graph. To cope with the cross-organizational characteristics of emergency response, we conduct the formal modeling and analysis of cross-organizational emergency response system in [4]. This work first introduces the formal model of emergency activities and identifies five kinds of interactivity relations during the cross-organizational coordination process. Then, OTRM-Net, a kind of Petri nets extended with time, resource message, and organization information, is presented to model the emergency response processes. Also, we further study its timing performance, resource conflict detection, and the reduction or concealment of inner emergency activities. More recently, we propose a top-down approach for model construction and correctness verification of cross-organization emergency response processes in [3]. For the resource conflict and resolution issue during the execution of cross-organization emergency response process, a novel controller design method is given in [5]. In addition, the privacy issue during the modeling of cross-organization emergency response processes is handled by a package reduction-based privacy protection approach in [6].

*2.3. Summary.* Based on the above literature review, we can see that researches into (1) modeling and analysis of cross-organizational emergency response processes and (2) emergency resource allocation and optimization issue have drawn much public attention. However, the existing works have at least the following limitations: (1) emergency resources are not fully quantified (e.g., [2–6]); that is, emergency resource quantity is not investigated during the modeling and allocation process, which will inevitably lead to an inaccurate resource allocation; and (2) there is a lack of time description for emergency activities, based on which we can estimate the time performance of cross-organization emergency response processes when using a specific resource allocation strategy. To deal with these limitations, this work provides a two-layered framework to support optimized resource allocation for cross-organizational emergency response processes.

### 3. A Two-Layered Framework to Support Emergency Resource Allocation

In this section, we introduce a cross-organization fire emergency scenario that will be used as an illustrating example for the paper. Then, we propose our emergency resource allocation mechanism.

*3.1. A Cross-Organization Fire Emergency Scenario.* In this paper, a fire emergency scenario in [3] is adopted as a typical scenario. It involves the following organizations: police station, emergency command center (ECC), explosive ordnance disposal (EOD) team, fire brigade, and hospital. Some of the critical missions in this scenario are rescue of victims and disposal of the moving fire. This scenario includes the following steps:

- (1) The police station first receives the fire emergency call and then reports the emergency information to ECC.
- (2) The police rushes to the emergency site to perform its detailed disposal missions and then reports the site conditions to ECC.
- (3) After receiving the emergency information, ECC first establishes a temporary emergency command group and then makes and issues emergency plans to its collaborative organizations, that is, medical rescue instruction to a hospital, search EOD instruction to an EOD team, and fire rescue instruction to the fire brigade.
- (4) The EOD team rushes to the site upon receiving the search EOD instruction from ECC and conducts its specific disposal activities according to its emergency handling requirements and finally reports the EOD search results to ECC.
- (5) The fire brigade rushes to the site upon receiving the fire rescue instruction from ECC and conducts its specific disposal activities according to its emergency handling requirements and finally reports the fire rescue results to ECC.

- (6) The hospital personnel rush to the site upon receiving the medical rescue instruction from ECC and conduct their specific disposal activities according to their emergency handling requirements and finally report the medical rescue results to ECC.
- (7) After receiving all the feedback information from the hospital, EOD team, and fire brigade, ECC makes emergency summary and evaluation and finally does the file archive.
- (8) ECC arranges the media coverage for the whole emergency response, and, finally, ECC, fire brigade, and hospital do the media coverage together.

**3.2. Cross-Organization Emergency Resource Allocation Mechanism.** To support the effective emergency resource allocation of cross-organization emergency response processes, a two-layered framework is presented as shown in Figure 1.

**3.2.1. Layer 1.** Each emergency organization establishes its respective emergency response process based on the emergency requirements. Then, each emergency organization submits its emergency response process to the ECC for further integration and evaluation. Next, the ECC integrates the response process received from its subordinate emergency organizations.

**3.2.2. Layer 2.** The ECC first conducts the emergency resource requirement analysis in terms of minimum consumable resource requirement, minimum reusable resource requirement, and reliable reusable resource requirement. According to the resource requirement analysis results, two strategies, that is, minimum execution time strategy and minimum resource consumption strategy, are obtained to guide the resource allocation. In this way, stakeholders can determine which strategy is more suitable according to the amount of available resources and expected finishing time.

The scope of this paper is to provide decision-making information to support emergency resource allocation in a cross-organization scenario. In this case, emergency resources are like the public transportation vehicles, public communication devices, emergency personnels, and so forth.

#### 4. CE-Net-Based Modeling of Cross-Organization Emergency Response Processes

In this section, we propose the formal definition of a Petri net-based model for the cross-organization emergency response processes, named CE-net, which is an extension of our previous E-net [1]. The CE-net is different from our previous models, such as RM\_WF\_Net in [29] and OTRM\_Net in [4], as it emphasizes (1) the uncertainty execution time of emergency activities and (2) effective resource allocation strategy of cross-organization emergency response processes.

**4.1. CE-Net.** Our work is based on Petri nets (or WF-net). Some of the essential terminologies and notations are reviewed following [13–15, 23, 24, 30–37].

Let  $Z = \{0, 1, 2, \dots\}$  and  $Z_n = \{1, 2, \dots, n\}$ , where  $n$  is a positive integer, and let  $\mathbf{R}$  be the set of nonnegative real numbers.

**Definition 1** (see [13]). A Petri net is a 4-tuple  $\Sigma = (P, T, F, M_0)$ , where (1)  $P = \{p_1, p_2, \dots, p_m\}$  is a finite set of places; (2)  $T = \{t_1, t_2, \dots, t_n\}$  is a finite set of transitions; (3)  $F \subseteq (P \times T) \cup (T \times P)$  is a finite set of arcs (flow relation); (4)  $M_0: P \rightarrow Z$  is the initial marking; and (5)  $P \cap T = \emptyset$  and  $P \cup T \neq \emptyset$ .

For all  $x \in P \cup T$ , the set  $\bullet x = \{y \mid y \in P \cup T \wedge (y, x) \in F\}$  is the preset of  $x$ , and  $x^\bullet = \{y \mid y \in P \cup T \wedge (x, y) \in F\}$  is the postset of  $x$ .  $p$  is marked by  $M$  if  $M(p) > 0$ . A transition  $t \in T$  is enabled under  $M$ , if and only if,  $\forall p \in \bullet t: M(p) > 0$ , denoted as  $M[t > .$  If  $M[t > .$  holds,  $t$  may fire, resulting in a new marking  $M'$ , denoted as  $M[t > M'$ , such that  $M'(p) = M(p) - 1$  if,  $\forall p \in \bullet t \setminus t^\bullet$ ,  $M'(p) = M(p) + 1$  if  $\forall p \in t^\bullet \setminus \bullet t$ , and otherwise  $M'(p) = M(p)$ . An initial marking is denoted by  $M_0$  and  $R(M_0)$  is defined as the set of all reachable marking sets of  $\Sigma$ , where  $\forall M_i \in R(M_0)$  such that  $M_0[\delta > M$ .

**Definition 2** (see [13]). A weighted Petri net is a 5-tuple  $\Sigma_W = (P, T, F, M_0, W)$ , where (1)  $\Sigma = (P, T, F, M_0)$  is a Petri net and (2)  $W: F \rightarrow Z_n$  is a weight function that maps an arc to a positive integer.

**Definition 3** (see [23]). A Petri net  $\Sigma = (P, T, F, M_0)$  is a WF-net if (1) there is one source place  $p_s \in P$  such that  $\bullet p_s = \emptyset$ ; (2) there is one sink place  $p_e \in P$  such that  $p_e^\bullet = \emptyset$ ; (3) each node  $x \in P \cup T$  is on a path from  $p_s$  to  $p_e$ ; and (4)  $\forall p \in P, M_0(p) = 1$  if  $p = p_s$ , and otherwise  $M_0(p) = 0$ .

In a WF-net, the transition set  $T$  is used to represent the normal activities, the place set  $P$  is used to represent logic connection relation of activities, and source place and sink place especially represent the start and end of the process. Based on the classical WF-net, we propose CE-net by extending it with uncertain execution time, resource, and message information to suit the cross-organization emergency response process modeling demand. It differs from our previous RM\_WF\_Net in [29] as it involves the emergency activity execution interval.

**Definition 4.** A 5-tuple  $\Sigma_{CE} = (P, T, F, \alpha, \beta, W, M_0)$  is a CE-net if

- (1)  $P = P_L \cup P_M \cup P_R$ , where  $P_L \cap P_R = \emptyset$ ,  $P_R \cap P_M = \emptyset$ ,  $P_L \cap P_M = \emptyset$ ,  $P_L$  represents the logic place set,  $P_M$  represents the message place set, and  $P_R$  represents the resource place set in a  $\Sigma_{CE}$ , and  $P_R = P_{RR} \cup P_{CR}$ , where  $P_{RR}$  is the reusable resource place set and is the consumable resource place set;
- (2)  $T$  is the transition set of a CE-net, and it represents an emergency activity set;
- (3)  $F = F_L \cup F_M \cup F_R$ , where

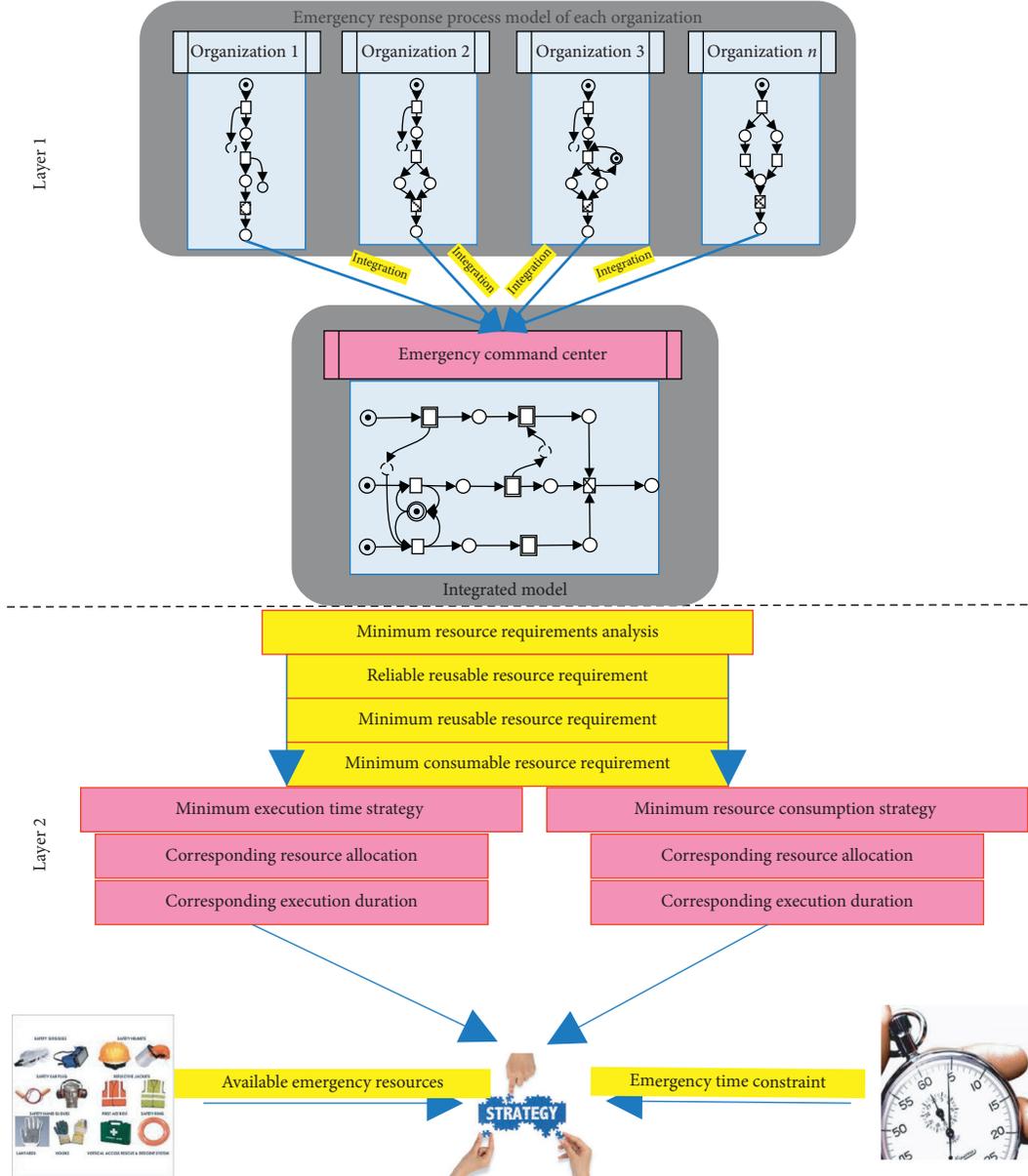


FIGURE 1: A two-layered framework to support emergency resource allocation.

- (3.1)  $F_L = (P_L \times T) \cup (T \times P_L)$  represents the logical structure of a  $\Sigma_{CE}$ ;
- (3.2)  $F_M = (P_M \times T) \cup (T \times P_M)$  represents the required and sent messages of a  $\Sigma_{CE}$ ;
- (3.3)  $F_R = (P_R \times T) \cup (T \times P_R)$  represents the required and released resources of a  $\Sigma_{CE}$ ;
- (4)  $\alpha: T \cup P_R \longrightarrow R, \forall t \in T \cup P_R, \alpha(t) \geq 0$  is the minimum time to execute (or prepare) an emergency activity (or resource)  $t$ ;
- (5)  $\beta: T \cup P_R \longrightarrow R, \forall t \in T \cup P_R, \beta(t) \geq 0$  is the maximum time to execute (or prepare) an emergency activity (or resource)  $t$ , such that  $\alpha(t) \leq \beta(t)$ ;
- (6)  $W: F \longrightarrow Z_n$  is the weighted function, such that,  $\forall f \in F, W(f) = 1$  if  $f \in F_L \cup F_M$ ,  $W(f) = \#req(r, t)$  if  $(f \in (P_{RR} \cup P_{CR}) \times T) \wedge (f = (r, t))$ , and  $W(f) = \#sent(t,$

$r)$  if  $(f \in T \times P_{RR}) \wedge (f = (t, r))$ .  $\#req(t, r)$  represents the required amount of resource  $r$  to execute activity  $t$ , and  $\#sent(t, r)$  represents the sent amount of resource  $r$  when activity  $t$  finishes (only reusable resources have this flow); (7)  $\forall p \in P, M_0(p) = 1$  if  $p \in P_L \wedge \bullet p = \emptyset, M_0(p) = \#num(p)$  if  $p \in P_R$ , and otherwise  $M_0(p) = 0$ .  $\#num(p)$  represents the initial available amount of resource  $p$ .

The firing rule of a CE-net is the same as that of a traditional Petri net. Given a marking  $M, \forall t \in T, t$  is enabled under  $M$  if,  $\forall p \in \bullet t, M(p) \geq 1$ , where  $p \in P_L \cup P_M \cup P_R$ . Firing an enabled  $t$  removes a token from each place in  $\bullet t$  and deposits one to each place in  $t^\bullet$ . All properties, such as reachability and boundedness, can be defined similarly. The main differences between an CE-net and a Petri net are the following: (1) the CE-net is a special type of Petri net

extended with message place set ( $P_M$ ) and resource place set ( $P_R$ ); (2) a transition in CE-net is associated with two time functions to represent its minimum and maximum execution time; and (3) a weighted function is introduced to represent the amount of required or sent resources when an emergency activity starts or ends.

**4.2. Modeling of Cross-Organization Emergency Response Processes Using CE-Net.** Modeling of cross-organization emergency response processes involves the following steps: (1) modeling emergency activities with CE-net; (2) modeling control structure with CE-net; and (3) integration of control structure model with those activity models.

In a CE-net, an emergency activity model is shown in Figure 2, where  $p_{ready}$  is the ready place,  $p_{end}$  is the end place,  $P_{resourceReq}$  and  $P_{resourceSent}$  are resource places which mean the required and sent resources,  $p_{messageReq}$  and  $p_{messageSent}$  are required and sent message places, and  $\alpha(t)$  and  $\beta(t)$  are the minimum execution time and maximum execution time, respectively. To distinguish the logic, resource, and message places properly, a logic place is drawn with a normal circle, a message place is drawn as a circle with dashed line, and a resource place is drawn as a two concentric circles with full line.

It is worth noting that the control structure of a CE-net, denoted as  $(P_L, T, F_L, M_0|_L)$ , is a standard WF-net, where  $M_0|_L$  is the projection of  $M_0$  on  $P_L$ . Therefore, its basic control structures, including sequence structure, concurrent structure, choice structure, and loop structure, can be modeled in the same way as that demonstrated in [29]. According to the emergency response descriptions in Section 3, emergency activity information, including activity ID, activity name, required resources/messages, sent resources/messages, of the Police state, ECC, EOD team, fire brigade, and hospital, is shown in Tables 1–3, respectively.

It is worth mentioning that we use multisets to represent the required resource set and the sent resource set of an emergency activity.  $M(R)$  is the set of all multisets over a set  $R$ . For some multiset  $m \in M(R)$ ,  $m(r)$  denotes the number of times element  $r \in R$  appears in  $m$ . For example,  $m = \{a^2, b^3\}$  is a multiset over  $\{a, b\}$ . It represents that element  $a$  appears two times and element  $b$  appears three times in  $m$ . The ordering of elements is irrelevant for multiset.

Using the detailed modeling approaches in [29], a CE-net is obtained as shown in Figure 3, which represents the overall execution process of the cross-organization fire emergency response processes in Section 3.

It is worth noting that there are five resource places that correspond to five kind resources. For graphic simplicity, some resources may have more than one resource place.

## 5. Reduction Rules of CE-Net

Based on the modeling approaches in last subsection, a CE-net can be constructed. Unfortunately, the CE-net may contain an excessive number of places and transitions, which will lead to an inefficient analysis and troublesome understanding. Hence, two reduction rules are introduced to

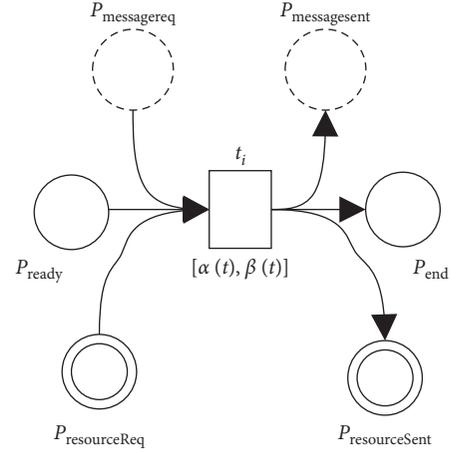


FIGURE 2: CE-net model of an emergency activity.

reduce the model scale while maintaining the structure, time, resource, and message invariant.

**5.1. Rule 4.1.** Given a CE-net  $\Sigma_{EC} = (P, T, F, \alpha, \beta, W, M_0)$ , if  $t_i, t_j \in T$  are two sequential activities such that  $t_i^* = \bullet t_j$ ,  $\bullet t_i \cap P_R = \emptyset$ ,  $t_i^* \cap (P_R \cup P_M) = \emptyset$ ,  $\bullet t_j \cap (P_R \cup P_M) = \emptyset$ , and  $t_j^* \cap P_R = \emptyset$ , and their timing constraints are  $[\alpha(t_i), \beta(t_i)]$  and  $[\alpha(t_j), \beta(t_j)]$ , respectively. Then, transitions  $t_i$  and  $t_j$  can be merged to a new one, denoted as  $t_{ij}$ , such that  $\bullet t_{ij} = \bullet t_i$ ,  $t_{ij}^* = t_j^*$ ,  $\alpha(t_{ij}) = \alpha(t_i) + \alpha(t_j)$ , and  $\beta(t_{ij}) = \beta(t_i) + \beta(t_j)$ .

Rule 1 shows the reduction rule for sequential activities that do not require any messages and resources, and an example of Rule 4.1 is shown in Figure 4, where  $t_i$  and  $t_j$  are merged to  $t_{ij}$ .

**5.2. Rule 4.2.** Given a CE-net  $\Sigma_{EC} = (P, T, F, \alpha, \beta, W, M_0)$ , if  $t_i, t_j \in T$  are two concurrent activities such that  $\bullet t_i \cap (P_R \cup P_M) = \emptyset$ ,  $t_i^* \cap (P_R \cup P_M) = \emptyset$ ,  $\bullet t_j \cap (P_R \cup P_M) = \emptyset$ , and  $t_j^* \cap (P_R \cup P_M) = \emptyset$ , and their timing constraints are  $[\alpha(t_i), \beta(t_i)]$  and  $[\alpha(t_j), \beta(t_j)]$ , respectively. Then, transitions  $t_i$  and  $t_j$  can be merged to a new transition, denoted as  $t_{ij}$ , such that  $\alpha(t_{ij}) = \max\{\alpha(t_i), \alpha(t_j)\}$  and  $\beta(t_{ij}) = \max\{\beta(t_i), \beta(t_j)\}$ .

Rule 2 shows the reduction for two concurrent activities that do not require any messages and resources, and an example of Rule 4.2 is illustrated in Figure 5, where  $t_i$  and  $t_j$  are replaced by  $t_{ij}$ .

We have the following explanations for reduction rules: (1) we only introduce the atomic reduction rules, and some advanced composite rules can be realized on top of these basic ones; (2) we only introduce rules that suit the sequence and concurrent structures, while choice and loop structures are not considered because their reduction results are usually not deterministic and can cause property changes compared with the original model; and (3) the firing rule of a transition obtained by reduction is the same as that of a traditional one; that is, the firing rule of a reduced CE-net is the same as that of a Petri net.

In the following, we show how to use our reduction rules to reduce the CE-net in Figure 3.  $t_{11-12-13}$  is obtained by merging transitions  $t_{11}$ ,  $t_{12}$ , and  $t_{13}$  using reduction Rule 4.1

TABLE 1: Emergency activity information.

Activity ID	Minimum execution time	Maximum execution time	Required messages	Sent messages	Required resources	Sent resources	Organizations	Preactivities
$t_1$	2	4	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Police station}	$\emptyset$
$t_2$	3	5	$\emptyset$	$\{p_{m1}\}$	$\{p_{r2}\}$	$\{p_{r2}\}$	{Police station}	$\{t_1\}$
$t_3$	4	7	$\emptyset$	$\emptyset$	$\{p_{r1}\}$	$\{p_{r1}\}$	{Police station}	$\{t_2\}$
$t_4$	5	10	$\emptyset$	$\emptyset$	$\{p_{r3}^2, p_{r5}^2\}$	$\{p_{r3}^2\}$	{Police station}	$\{t_3\}$
$t_5$	6	12	$\emptyset$	$\emptyset$	$\{p_{r3}^2, p_{r5}^2\}$	$\{p_{r3}^2\}$	{Police station}	$\{t_3\}$
$t_6$	8	14	$\emptyset$	$\emptyset$	$\{p_{r3}^2, p_{r5}^2\}$	$\{p_{r3}^2\}$	{Police station}	$\{t_3\}$
$t_7$	5	8	$\emptyset$	$\{p_{m2}\}$	$\emptyset$	$\emptyset$	{Police station}	$\{t_4, t_5, t_6\}$
$t_8$	6	10	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Police station}	$\{t_7\}$
$t_9$	2	3	$\{p_{m1}\}$	$\emptyset$	$\emptyset$	$\emptyset$	{ECC}	$\emptyset$
$t_{10}$	3	5	$\{p_{m2}\}$	$\{p_{m3}, p_{m4}, p_{m5}\}$	$\{p_{r2}\}$	$\{p_{r2}\}$	{ECC}	$\{t_9\}$
$t_{11}$	5	8	$\{p_{m6}, p_{m7}\}$	$\emptyset$	$\emptyset$	$\emptyset$	{ECC}	$\{t_{10}\}$
$t_{12}$	8	15	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{ECC}	$\{t_{11}\}$
$t_{13}$	2	4	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{ECC}	$\{t_{12}\}$
$t_{14}$	3	5	$\emptyset$	$\emptyset$	$\{p_{r3}^2, p_{r4}\}$	$\{p_{r3}^2, p_{r4}\}$	{ECC, fire brigade, hospital}	$\{t_{13}\}$
$t_{15}$	4	8	$\{p_{m3}\}$	$\emptyset$	$\emptyset$	$\emptyset$	{EOD team}	$\emptyset$
$t_{16}$	14	16	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{EOD team}	$\{t_{15}\}$
$t_{17}$	4	5	$\emptyset$	$\{p_{m6}\}$	$\{p_{r2}\}$	$\{p_{r2}\}$	{EOD team}	$\{t_{16}\}$
$t_{18}$	4	6	$\{p_{m4}\}$	$\emptyset$	$\{p_{r1}\}$	$\{p_{r1}\}$	{Fire brigade}	$\{\emptyset\}$
$t_{19}$	8	12	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Fire brigade}	$\{t_{18}\}$
$t_{20}$	6	8	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Fire brigade}	$\{t_{19}\}$
$t_{21}$	6	8	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Fire brigade}	$\{t_{19}\}$
$t_{22}$	2	3	$\emptyset$	$\{p_{m7}\}$	$\{p_{r2}\}$	$\{p_{r2}\}$	{Fire brigade}	$\{t_{20}, t_{21}\}$
$t_{23}$	3	5	$\{p_{m5}\}$	$\emptyset$	$\{p_{r1}\}$	$\{p_{r1}\}$	{Hospital}	$\{\emptyset\}$
$t_{24}$	4	6	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Hospital}	$\{t_{23}\}$
$t_{25}$	8	12	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Hospital}	$\{t_{24}\}$
$t_{26}$	6	8	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Hospital}	$\{t_{24}\}$
$t_{27}$	4	6	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	{Hospital}	$\{t_{25}, t_{26}\}$
$t_{28}$	2	3	$\emptyset$	$\{p_{m8}\}$	$\{p_{r2}\}$	$\{p_{r2}\}$	{Hospital}	$\{t_{27}\}$

such that  $\alpha(t_{11-12-13}) = \alpha(t_{11}) + \alpha(t_{12}) + \alpha(t_{13}) = 5 + 8 + 2 = 15$  and  $\beta(t_{11-12-13}) = \beta(t_{11}) + \beta(t_{12}) + \beta(t_{13}) = 8 + 15 + 4 = 27$ .  $t_{19-20-21}$  is obtained by (1) merging transitions  $t_{20}$  and  $t_{21}$  using reduction Rule 4.2 to obtain  $t_{20-21}$  such that  $\alpha(t_{20-21}) = \max\{\alpha(t_{20}), \alpha(t_{21})\} = \max\{6, 6\} = 6$ , and  $\beta(t_{20-21}) = \max\{\beta(t_{20}), \beta(t_{21})\} = \max\{8, 8\} = 8$  and (2) merging transitions  $t_{19}$  and  $t_{20-21}$  with reduction Rule 4.1 such that  $\alpha(t_{19-20-21}) = \alpha(t_{19}) + \alpha(t_{20-21}) = 8 + 6 = 14$  and  $\beta(t_{19-20-21}) = \beta(t_{19}) + \beta(t_{20-21}) = 12 + 8 = 20$ .  $t_{24-25-26-27-28}$  is obtained by (1) merging transitions  $t_{25}$  and  $t_{26}$  using reduction Rule 4.2 to obtain  $t_{25-26}$  such that  $\alpha(t_{25-26}) = \max\{\alpha(t_{25}), \alpha(t_{26})\} = \max\{8, 6\} = 8$ , and  $\beta(t_{25-26}) = \max\{\beta(t_{25}), \beta(t_{26})\} = \max\{12, 8\} = 12$  and (2) merging transitions  $t_{24}$ ,  $t_{25-26}$ ,  $t_{27}$ , and  $t_{28}$  to obtain  $t_{24-25-26-27-28}$  with reduction Rule 4.1 such that  $\alpha(t_{24-25-26-27}) = \alpha(t_{24}) + \alpha(t_{25-26}) + \alpha(t_{27}) + \alpha(t_{28}) = 4 + 8 + 4 + 2 = 18$  and  $\beta(t_{24-25-26-27}) = \beta(t_{24}) + \beta(t_{25-26}) + \beta(t_{27}) + \beta(t_{28}) = 6 + 12 + 6 + 3 = 27$ .

The CE-net of the cross-organization fire emergency response processes after reduction is shown in Figure 6. To give a comparison, Table 4 gives the number of transitions, message places, and resource places of the CE-net before and after reduction. Based on the comparison, we can clearly see that (1) its scale (in terms of the number of transitions and logic places) is much smaller and (2) the collaboration elements (resources and messages) stay invariant.

## 6. CE-Net-Based Minimum Resource Requirement Analysis

In this section, we discuss the minimum resource requirements of cross-organization emergency response processes based on the CE-net. During this procedure, we analyze reusable and consumable resources separately. For simplicity, we first redefine two vector operators, “ $<$ ” and “ $\geq$ .” Let  $X = \langle x_1, x_2, \dots, x_n \rangle$  and  $Y = \langle y_1, y_2, \dots, y_n \rangle$  be two  $n$ -dimension vectors. If  $\forall X.x_i \geq Y.y_i$ , for  $1 \leq i \leq n$ , we have  $X \geq Y$ . If  $\exists X.x_i < Y.y_i$ , for  $1 \leq i \leq n$ , we have  $X < Y$ . Assume that the available consumable resource vector and available reusable resource vector of cross-organization emergency response processes are denoted as  $R_{AC}$  and  $R_{AR}$ , respectively.

**6.1. Minimum Consumable Resource Requirement.** The minimum consumable resource vector is denoted as  $R_{MC} = \langle q(r_1), q(r_2), \dots, q(r_n) \rangle$ , where (1)  $r_i$  ( $i = 1, 2, \dots, n$ ) refers to a kind of consumable resource and it is modeled as a resource place  $p_{ri}$  in  $P_{CR}$  and (2)  $q(r_i)$  represents the number of  $r_i$ .  $V_{MC}$  can be computed by taking as input a CE-net based on the following algorithm.

The complexity of Algorithm 1 is  $O(|P_{CR}| \times |T|)$ , where  $|P_{CR}|$  is the number of consumable resource types and  $|T|$  is

TABLE 2: Emergency activity and message information.

Activity/message	Meaning
$t_1$	Alarm receipt
$t_2$	Report emergency information
$t_3$	Rush to the site (police)
$t_4$	Deal with the scene
$t_5$	Evacuation
$t_6$	Rescue the wounded
$t_7$	Alarm receipt
$t_8$	Search the suspect
$t_9$	Establish temporary emergency center
$t_{10}$	Make and issue emergency plans
$t_{11}$	Emergency summary and evaluation
$t_{12}$	File archive
$t_{13}$	Arranges media coverage
$t_{14}$	Do media coverage (together)
$t_{15}$	Rush to the site (EOD team)
$t_{16}$	Search the EOD
$t_{17}$	Report the EOD search results
$t_{18}$	Rush to the site (fire bridges)
$t_{19}$	Fight the fire
$t_{20}$	Recovery the site
$t_{21}$	Conduct mitigation operations
$t_{22}$	Report the fire rescue
$t_{23}$	Rush to the site (hospital)
$t_{24}$	Shunt the wounded
$t_{25}$	Treat the severely injured people
$t_{26}$	Treat the slightly injured people
$t_{27}$	Environmental quality detection
$t_{28}$	Report the medical rescue
$p_{m1}$	Emergency information
$p_{m2}$	Site conditions
$p_{m3}$	Medical rescue instruction
$p_{m4}$	Fire rescue instruction
$p_{m5}$	Search EOD instruction
$p_{m6}$	EOD search results
$p_{m7}$	Fire rescue results
$p_{m8}$	Medical rescue results

the number of emergency activities. For real-life emergency resource management, the cross-organization emergency response processes will break down because of shortage of consumable resources if  $R_{AC} < R_{MC}$ .

**6.2. Minimum Reusable Resource Requirement.** The minimum reusable resource vector is denoted as  $R_{MR} = \langle q(r_1), q(r_2), \dots, q(r_m) \rangle$ , where (1)  $r_i$  ( $i = 1, 2, \dots, n$ ) refers to a kind of reusable resource and it is modeled as a resource place  $p_{ri}$  in  $P_{RR}$  and (2)  $q(r_i)$  represents the number of  $r_i$ . Algorithm 2 shows how to compute  $V_{MR}$  by taking as input a CE-net.

The complexity of Algorithm 2 is  $O(|P_{RR}| \times |T|)$ , where  $|P_{RR}|$  is the number of reusable resource types and  $|T|$  is the number of activities. For real-life emergency resource management, the cross-organization emergency response processes will break down because of shortage of reusable resources if  $R_{AR} < R_{MR}$ .

**6.3. Reliable Reusable Resource Requirement.** Even though we have  $R_{AR} \geq R_{MR}$ , resource conflicts may still exist during

TABLE 3: Emergency resource information.

Resource ID	Meaning	Property
$P_{r1}$	Public transportation vehicle	Reusable
$P_{r2}$	Public communication device	Reusable
$P_{r3}$	Emergency personnel	Reusable
$P_{r4}$	Media personnel and equipment	Reusable
$P_{r5}$	Smoke masks	Consumable

the execution of cross-organization emergency response processes because of resource dependency and limited available resources. This kind of conflicts may delay the execution time of the emergency process. Therefore, the reliable reusable resource vector of a CE-net, denoted as  $R_{RR} = \langle q(r_1), q(r_2), \dots, q(r_m) \rangle$ , is introduced, where  $r_i$  ( $i = 1, 2, \dots, n$ ) refers to a kind of reusable resource and is represented as a resource place  $p_{ri}$  in  $P_{RR}$ . Algorithm 3 shows how to compute  $R_{RR}$  by taking as input a CE-net.

The complexity of Algorithm 3 is  $O(|P_{RR}| \times |T|)$ , where  $|P_{RR}|$  is the number of reusable resource types and  $|T|$  is the number of emergency activities. If  $R_{AR} \geq R_{RR}$ , there is no resource conflict during process execution; that is, potential resource conflicts are avoided as sufficient resources are provided to support parallel execution of activities in resource dependency.

**Theorem 1.** *The proposed approaches to analyze resource requirements of cross-organization emergency response processes on the basis of CE-net have  $O(m \times n)$  time complexity, where  $m$  is the number of resource types and  $n$  is the number of emergency activities.*

*Proof.* To analyze the resource requirement, three kinds of resource metrics, that is, the minimum consumable resource vector, the minimum reusable resource vector, and there liable reusable resource vector, are needed. Their calculation algorithms are implemented in Algorithms 1–3, the complexity of which is no bigger than  $O(m \times n)$ , where  $m$  is the number of resource types and  $n$  is the number of emergency activities. Therefore, the proposed method to analyze the resource requirement has its  $O(m \times n)$  time complexity.

Taking the cross-organization fire emergency response processes as an example, we have  $R_{MC} = \langle 6 \rangle$ ,  $R_{MR} = \langle 1, 1, 2, 1 \rangle$ , and  $R_{RR} = \langle 3, 2, 6, 1 \rangle$  by executing Algorithms 1–3.  $\square$

## 7. Two Effective Resource Allocation Strategies

In this section, we propose two optimized resource allocation strategies, that is, the minimum execution time strategy and the minimum resource consumption strategy. The former pursues time efficiency of cross-organization emergency response processes, and the latter focuses on high resource utilization rate. For each strategy, we estimate its corresponding time performance, based on which the stakeholder can determine which strategy is suitable according to the timing constraints. Before rendering the detailed strategies, we first introduce the time performance estimation and resource conflict detection approaches.

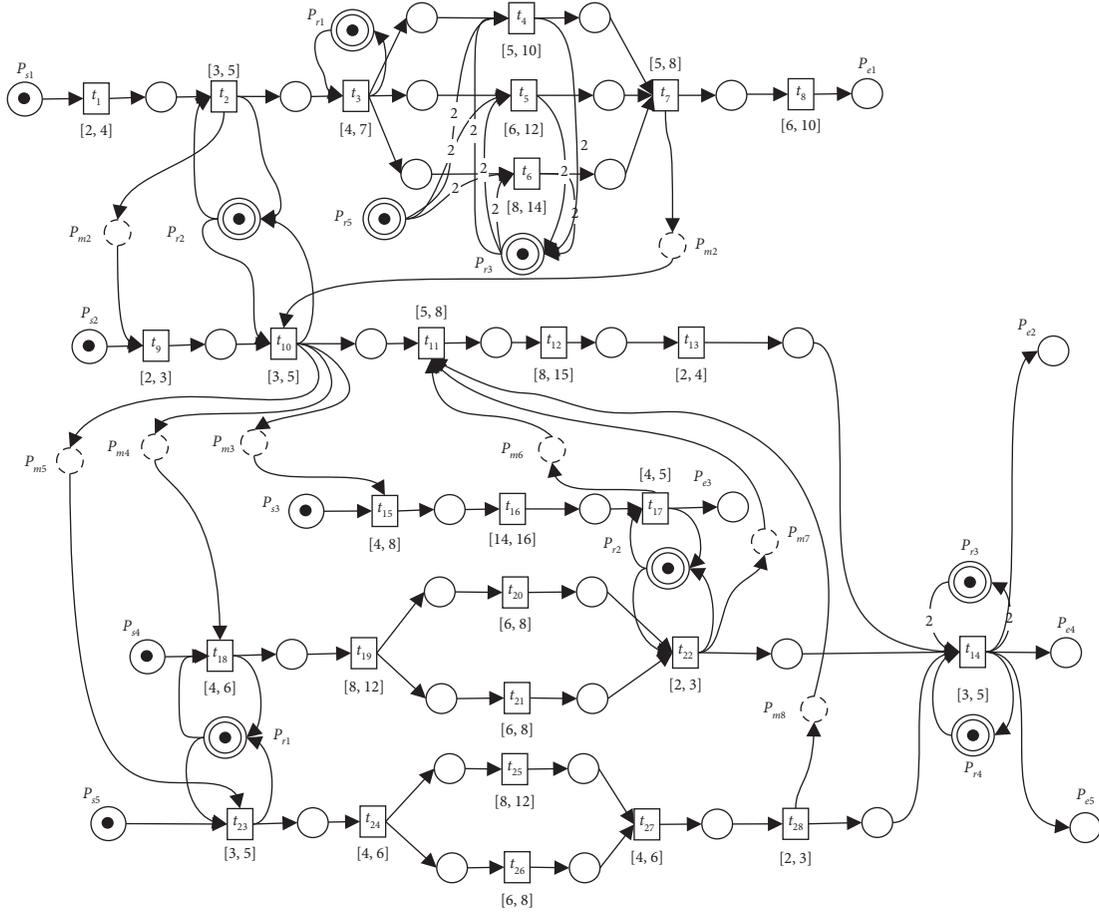


FIGURE 3: CE-net of cross-organization fire emergency response processes.

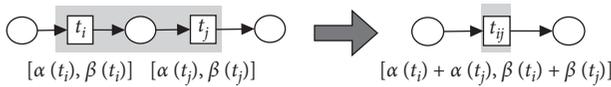


FIGURE 4: Reduction example of rule 1.

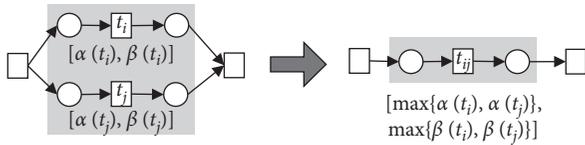


FIGURE 5: Reduction example of rule 2.

**7.1. Time Performance Estimation and Potential Resource Conflict Detection.** In a cross-organization emergency response processes scenario, an emergency activity starts to be executed only (1) after the termination of all its preactivities, (2) after receiving the required messages sent by other organizations, and (3) under the condition that all of its required resources are available and sufficient. In this subsection, the minimum execution time of a cross-organization emergency response process can be obtained.

Without considering resource factor, if each emergency activity is completed in its minimum execution time, the earliest time to start activity  $t$ , denoted by  $T_{e1}(t)$ , is as follows:

$$T_{e1}(t) = \begin{cases} 0, & \bullet(\bullet t) \\ = \emptyset, \max\{T_{e1}(t') + \alpha(t') | t' \in \bullet(\bullet t)\}, & \text{otherwise.} \end{cases} \quad (1)$$

Without considering resource factor, if each activity is completed in its maximum execution time, the earliest time to start activity  $t$ , denoted by  $T_{e2}(t)$ , is as follows:

$$T_{e2}(t) = \begin{cases} 0, & \bullet(\bullet t) \\ = \emptyset, \max\{T_{e2}(t') + \beta(t') | t' \in \bullet(\bullet t)\}, & \text{otherwise.} \end{cases} \quad (2)$$

The minimum execution time of a cross-organization emergency response process can be found if no resource conflict occurs. However, resource conflicts are inevitable during the execution of a cross-organization emergency response process. Therefore, resource conflict checking approach is investigated here.

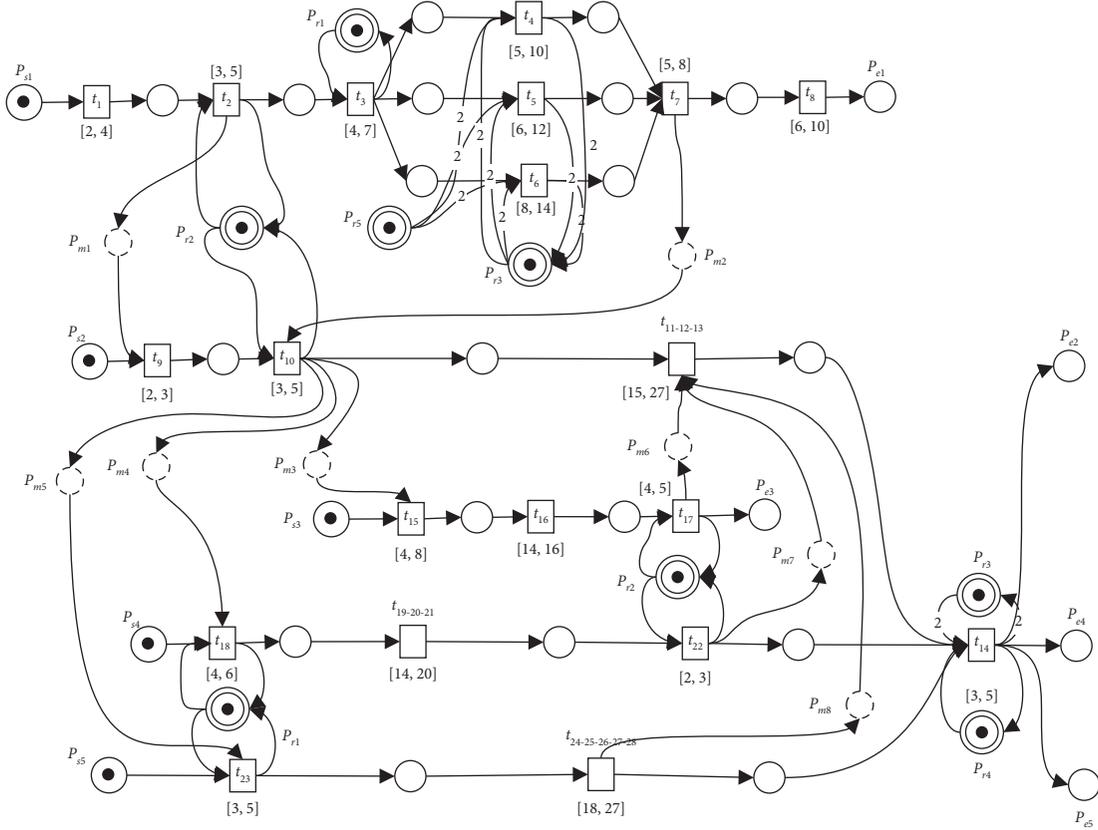


FIGURE 6: Reduced CE-net of the cross-organization fire emergency response process.

TABLE 4: Scale comparison of the CE-net before and after reduction.

Item	Before reduction	After reduction
Number of transitions	28	20
Message places	8	8
Resource places	5	5

In the following discussion, we use  $T_{\text{start}}(t)$  and  $T_{\text{end}}(t)$  to represent the real start and end time of an emergency activity  $t$ . We have two types of resources, reusable and consumable ones. The former one becomes available and can be reused after being released, while the latter is consumed during execution and cannot be reused. For this reason, resource conflict mentioned in our work is essentially caused by the reusable resources. If a conflict is caused by consumable resources, the only way to solve it is to add more such type resources.

**Definition 5.**  $\forall t_i, t_j \in T (t_i \neq t_j)$ ,  $t_i$  and  $t_j$  have *resource dependency*, denoted as  $t_i \Theta t_j$ , if  $\bullet(t_i) \cap \bullet(t_j) \subseteq P_R$ .

According to Definition 5, an algorithm to obtain the resource dependency in a cross-organization emergency response process is presented as follows.

In Algorithm 4, the complexity of Step 2 is  $O(|T|^2)$ . Therefore, the complexity of Algorithm 3 is  $O(|T|^2)$ . By executing Algorithm 4, the resource dependency set of the emergency response process scenario is obtained:  $\text{DependencySet} = \{(t_2, t_{10}), (t_4, t_5), (t_4, t_6), (t_5, t_6), (t_{18}, t_{23}),$

$(t_{17}, t_{22})\}$ . Here, if two activities have resource dependency; that is, they share same emergency resources. However, this does not necessarily mean that resource conflicts will occur between them. Only when these two activities need to be executed simultaneously will resource conflicts take place. In the following, the resource conflict between two emergency activities is formulized.

**Definition 6.**  $\forall t_i, t_j \in T (t_i \neq t_j)$ ,  $t_i$  and  $t_j$  are in *potential resource conflict*, denoted as  $t_i \otimes t_j$ , if (1)  $t_i \Theta t_j$  and (2)  $[T_{\text{start}}(t_i), T_{\text{end}}(t_i)]$  and  $[T_{\text{start}}(t_j), T_{\text{end}}(t_j)]$  are overlapping.

Definition 6 defines the potential resource conflicts, based on which we present an algorithm to check resource conflicts.

In Algorithm 5, because  $O(|\text{ConflictSet}|) \leq O(|T|)$ , the complexity of Step 2 is  $O(|T|)$ . Hence, Algorithm 5 has its computational complexity  $O(|T|)$ . By executing Algorithm 5, we can obtain that  $\text{ConflictSet} = \{(t_4, t_5), (t_4, t_6), (t_5, t_6), (t_{18}, t_{23}), (t_{17}, t_{22})\}$ . However, even if  $t_4$  and  $t_5$  are in potential conflicts,  $t_4$  requires 2 public communication devices, and  $t_5$  also requires 2 public communication devices and if the total number of available investigators in this case is less than 4, conflict occurs because of the competition for public communication devices. If the total number of available investigators is more than 4,  $t_4$  and  $t_5$  will no longer be in conflict. Based on the aforementioned analysis, we can see that whether a cross-organization emergency response process can be accomplished in its ideal execution time

```

Input:  $\Sigma_{CE} = (P, T, F, \alpha, \beta, W, M_0)$ .
Output:  $R_{MC}$ .
(1) /*Step 1: initialization*/
(2) Step 1:  $sum \leftarrow 0, R_{MC} \leftarrow 0$ .
(3) /*Step 2: to obtain the minimum consumable resource vector*/
(4) Step 2: FOR  $\forall p_{ri} \in P_{CR}$  DO
(5)         FOR  $\forall t_i \in T$  DO
(6)              $sum \leftarrow sum + W(p_{ri}, t_i)$ ;
(7)         END DO
(8)          $q(R_{MC}, r_i) \leftarrow sum$ ;
(9)          $sum \leftarrow 0$ ;
(10)        END DO
(11) Step 3: Output  $R_{MC}$ .

```

ALGORITHM 1: Calculate the minimum consumable resource vector  $R_{MC}$ .

```

Input:  $\Sigma_{CE} = (P, T, F, \alpha, \beta, W, M_0)$ .
Output:  $R_{MR}$ .
(1) /*Step 1: initialization*/
(2) Step 1:  $R_{MR} \leftarrow 0$ ;
(3) /*Step 2: to obtain the minimum reusable resource vector*/
(4) Step 2: FOR  $\forall p_{ri} \in P_{RR}$  DO
(5)         FOR  $\forall t_i \in T$  DO
(6)             IF  $W(p_{ri}, t_i) > q(R_{MR}, r_i)$  THEN
(7)                  $q(R_{MR}, r_i) \leftarrow W(p_{ri}, t_i)$ ;
(8)             END IF
(9)         END DO
(10)        END DO
(11) Step 3: Output  $R_{MR}$ .

```

ALGORITHM 2: Calculate the minimum reusable resource vector  $R_{MR}$ .

```

Input:  $\Sigma_{CE} = (P, T, F, \alpha, \beta, W, M_0)$ , DependencySet and  $R_{MR}$ .
Output:  $R_{RR}$ .
(1) /*Step 1: initialization*/
(2) Step 1:  $sum \leftarrow 0, V_{RR} \leftarrow V_{MR}$ ;
(3) /*Step 2: to obtain the reliable reusable resource vector*/
(4) Step 2: FOR  $\forall p_{ri} \in P_{RR}$  DO
(5)         FOR  $\forall (t_i, t_j) \in \text{DependencySet}$  DO
(6)              $sum \leftarrow W(p_{ri}, t_i) + W(p_{ri}, t_j)$ ;
(7)             IF  $sum > q(R_{RR}, r_i)$  THEN
(8)                  $q(R_{RR}, r_i) \leftarrow sum$ ;
(9)             END IF
(10)         $sum \leftarrow 0$ ;
(11)        END DO
(12)        END DO
(13) Step 3: Output  $R_{RR}$ .

```

ALGORITHM 3: Calculate the reliable reusable resource vector  $R_{RR}$ .

```

Input:  $\Sigma_{CE} = (P, T; F, \alpha, \beta, W, M_0)$ 
Output: DependencySet =  $\{(t_i, t_j) | t_i \otimes t_j\}$ .
(1) /*Step 1: initialization*/
(2) Step 1: DependencySet  $\leftarrow \emptyset$ ;
(3) /*Step 2: to detect resource dependency among emergency tasks*/
(4) Step 2: FOR  $\forall t_i, t_j \in T (t_i \neq t_j)$  DO
(5)     IF  $\bullet(t_i) \cap \bullet(t_j) \subseteq P_R$  THEN
(6)         DependencySet  $\leftarrow$  DependencySet  $\cup \{(t_i, t_j)\}$ ;
(7)     END IF
(8) END DO
(9) Step 3: Output DependencySet.

```

ALGORITHM 4: Detecting resource dependency.

```

Input:  $\Sigma_{CE} = (P, T; F, \alpha, \beta, W, M_0)$  and DependencySet.
Output: ConflictSet =  $\{(t_i, t_j) | t_i \otimes t_j\}$ .
(1) /*Step 1: initialization*/
(2) Step 1: ConflictSet  $\leftarrow$  DependencySet,  $T_{e1}(t) \leftarrow 0$ ,  $T_{e2}(t) \leftarrow 0$ ,  $T_{l1}(t) \leftarrow 0$ ,  $T_{l2}(t) \leftarrow 0$ ;
(3) /*Step 2: to detect if time interval of tasks with resource dependency overlaps*/
(4) Step 2: IF ConflictSet  $\neq \emptyset$  THEN
(5)     For  $\forall (t_i, t_j) \in$  ConflictSet DO
(6)         Calculate  $T_{e1}(t_i)$ ,  $T_{e2}(t_i)$ ,  $T_{e1}(t_j)$ ,  $T_{e2}(t_j)$ ;
(7)         IF  $[T_{e1}(t_i), T_{e2}(t_i) + \beta(T_i)] \cap [T_{e1}(t_j), T_{e2}(t_j) + \beta(T_j)] \neq \emptyset$  THEN
(8)             GOTO Step3;
(9)         ConflictSet  $\leftarrow$  ConflictSet  $- \{(t_i, t_j)\}$ ;
(10)    END DO
(11)    END IF
(12) Step 3: Output ConflictSet.

```

ALGORITHM 5: Checking resource conflicts.

depends on the total quantity of available resources. In this way, effective resource allocation strategies are needed to support its successful execution.

**7.2. Minimum Execution Time Strategy.** A cross-organization emergency response process is a real-time service, where timeliness is critical to its mission success. Thus, to maintain a cross-organization emergency response process finish in a high time performance, parallel executions of independent emergency activities are needed to reduce the whole execution time. In the last subsection, ideal execution duration is analyzed for this scenario. To achieve this goal, the consumable and reusable resource vectors,  $V_{AC}$  and  $V_{AR}$ , should be allocated as  $V_{AC} \geq V_{MC}$  and  $V_{AR} \geq V_{RR}$ . In this resource allocation condition, consumable resources can meet the minimum requirement and reusable resources are sufficient enough to support parallel executions and avoid potential resource conflicts. Therefore, the whole process can be finished in its minimum execution interval.

**Definition 7.** For a time interval Interval =  $[T_l, T_u]$ , it is defined as the minimum execution duration of a  $\Sigma_{EC} = (P, T, F, \alpha, \beta, W, M_0)$  if  $T_l = \max\{T_{e1}(t_i) + \alpha(t_i) | t_i \in T\}$  and  $T_u = \max\{T_{e2}(t_i) + \beta(t_i) | t_i \in T\}$ .

According to the above-mentioned formulas (1) and (2),  $T_{e1}(p)$  and  $T_{e2}(p)$  of each activity in Figure 6 can be obtained

and are shown in Table 5. From Table 5, we can obtain the ideal execution time for the cross-organization fire emergency response process, that is, if no resource conflict occurs, the minimum execution interval is [64, 107], where  $T_l = T_{e1}(t_{14}) + \alpha(t_{14}) = 61 + 3 = 64$  time units and  $T_u = T_{e2}(t_{14}) + \beta(t_{14}) = 102 + 5 = 107$  time units.

**7.3. Minimum Resource Consumption Strategy.** To finish an emergency response process with the smallest resource consumption, the optimized consumable and reusable resource vectors,  $V_{AC}$  and  $V_{AR}$ , should be allocated as  $V_{AC} \geq V_{MC}$  and  $V_{AR} \geq V_{MR}$ . In this resource allocation condition, both consumable and reusable resources can meet their minimum requirements. Therefore, the whole process can be finished with the smallest resource consumption. However, resource conflicts may exist during the process execution. Thus, some activities in conflict may be postponed because of waiting for reusable emergency resources which are exclusively occupied by others. As a result, time performance of the emergency response process is affected. Next, Algorithm 6 is proposed to estimate the execution duration for this case.

The complexity of Algorithm 6 is  $O(|T|)$ , where  $|T|$  is the number of emergency activities. By executing Algorithm 6 by taking the timing information of the cross-organization

TABLE 5:  $T_{e1}(p)$  and  $T_{e2}(p)$  of each Task in Figure 6.

Activity	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	$t_7$	$t_8$	$t_9$	$t_{10}$
$T_{e1}(t)$	0	2	5	9	9	9	17	22	5	22
$T_{e2}(t)$	0	4	9	16	16	16	30	38	9	38
Activity	$t_{11-12-13}$	$t_{14}$	$t_{15}$	$t_{16}$	$t_{17}$	$t_{18}$	$t_{19-20-21}$	$t_{22}$	$t_{23}$	$t_{24-25-26-27-28}$
$T_{e1}(t)$	46	61	25	29	23	25	29	43	25	28
$T_{e2}(t)$	75	102	43	52	67	43	49	69	43	48

**Input:**  $\Sigma_{CE} = (P, T; F, \alpha, \beta, W, M_0)$ , ConflictSet, Interval.  
**Output:**  $M\_Interval = [M_l, M_u]$

- (1) /\* Step 1: initialization \*/
- (2) Step 1:  $M_l \leftarrow T_l, M_u \leftarrow T_u$ ;
- (3) /\* Step 2: to obtain the maximum execution time \*/
- (4) Step 2: **FOR**  $\forall (t_i, t_j) \in \text{ConflictSet}$  **DO**
- (5)      $M_l \leftarrow M_l + \min\{\alpha(t_i), \alpha(t_j)\}$ ;
- (6)      $M_u \leftarrow M_u + \min\{\beta(t_i), \beta(t_j)\}$ ;
- (7)     **END DO**
- (8) Step 3: Output  $M\_Interval$ .

ALGORITHM 6: Calculate the execution duration.

emergency response process as an example, we can obtain that its execution duration is [64, 107] using minimum resource consumption strategy.

## 8. Conclusion

Efficient emergency resource allocation decision support is of vital importance for cross-organization emergency management. To some extent, this will influence emergency resource utility rate and the whole emergency mission success. In the work, we first propose CE-net to model a cross-organization emergency response process, based on which emergency resource allocation decision-making strategies, including minimum execution time strategy and minimum resource consumption strategy, are presented. According to the available emergency resources and emergency time constraint, certain strategy is selected by the emergency manager according to their specific requirements. It is proved that the minimum execution time strategy can ensure the global emergency response process execution without resource conflict or block, and therefore the performance is optimal.

The minimum resource consumption strategy can guarantee the whole emergency to be finished with the least emergency consumption, and, in this way, the resource related cost is reduced. However, there may exist some resource conflicts during real-life emergency process execution when we choose the latter. To resolve these conflicts and maintain a success execution, efficient resource conflict resolution strategies are needed. Therefore, resource conflict resolution strategies with corresponding controller design approaches will be our future work. In addition, it would be interesting to discover such emergency response process models from historic emergency response logs by using process discovery techniques and advanced learning

algorithms to realize data-driven emergency decision-making [38].

## Data Availability

The data used in this manuscript come from an emergency management branch, and the authors cannot release all information at the current stage for privacy protection.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 61902222 and U1931207, the Taishan Scholars Program of Shandong Province under Grants ts20190936 and tsqn201909109, Science & Technology Development Fund of Shandong Province of China under Grants 2019KJN024, ZR2019LZH001, and ZR2017MF027, the Humanities and Social Science Research Project of the Ministry of Education under Grant 18YJAZH017, and Shandong University of Science and Technology Research Fund under Grant 2015TDJH102.

## References

- [1] 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.
- [2] 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.
- [3] H. Duan, C. Liu, Q. Zeng, and M. Zhou, “Refinement-based hierarchical modeling and correctness verification of cross-organization collaborative emergency response process,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 8, pp. 2548–2559, 2020.
  - [4] Q. Zeng, F. Lu, C. Liu, and D. Meng, “Modeling and analysis for cross-organizational emergency response systems,” *Journal of Computers (In Chinese)*, vol. 36, no. 11, pp. 2290–2302, 2013.
  - [5] Q. Zeng, C. Liu, H. Duan, and M. Zhou, “Resource conflict checking and resolution controller design for cross-organization emergency response processes,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 10, pp. 3685–3700, 2020.
  - [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, no. 1, pp. 55573–55585, 2018.
  - [7] E. Deelman, G. Singh, M.-H. Su et al., “Pegasus: a framework for mapping complex scientific workflows onto distributed systems,” *Scientific Programming*, vol. 13, no. 3, pp. 219–237, Article ID 128026, 2005.
  - [8] R. Sakellariou and H. Zhao, “A hybrid heuristic for DAG scheduling on heterogeneous systems,” in *Proceedings of the 18th International Parallel and Distributed Processing Symposium*, Santa Fe, NM, USA, January 2004.
  - [9] J. Blythe, S. Jain, E. Deelman et al., “Task scheduling strategies for workflow-based applications in grids,” in *Proceedings of IEEE International Symposium on Cluster Computing on Grid*, vol. 2, pp. 759–767, Cardiff, WA, UK, June 2005.
  - [10] H. Topcuoglu, S. Hariri, and M. Min-You Wu, “Performance-effective and low-complexity task scheduling for heterogeneous computing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 3, pp. 260–274, 2002.
  - [11] E.-K. Byun, Y.-S. Kee, J.-S. Kim, E. Deelman, and S. Maeng, “BTS: resource capacity estimate for time-targeted science workflows,” *Journal of Parallel and Distributed Computing*, vol. 71, no. 6, pp. 848–862, 2011.
  - [12] E. K. Byun, J. S. Kim, Y. S. Kee et al., “Efficient resource capacity estimate of workflow applications for provisioning resources,” in *Proceedings of the 4th IEEE International Conference on e-Science*, pp. 1–20, Indianapolis, IA, USA, January 2008.
  - [13] W. Reisig, *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*, Springer, Berlin, Germany, 2013.
  - [14] 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.
  - [15] C. Liu, Q. Zeng, and H. Duan, “Formulating the data-flow modeling and verification for workflow: a petri net based approach,” *International Journal of Science and Engineering Applications*, vol. 3, no. 4, pp. 107–112, 2014.
  - [16] T. Suleyman and A. W. William, “The emerging area of emergency management and engineering,” *IEEE Transactions on Engineering Management*, vol. 45, no. 2, pp. 103–105, 1998.
  - [17] F. Fiedrich, F. Gehbauer, and U. Rickers, “Optimized resource allocation for emergency response after earthquake disasters,” *Safety Science*, vol. 35, no. 1–3, pp. 41–57, 2000.
  - [18] Su. Wang, Y. Wang, and J. Sun, “An optimized emergency resources allocation algorithm for large-scale public emergency,” in *Proceedings of the International Conference on Machine Learning and Cybernetics*, Hong Kong, China, September 2007.
  - [19] Y. Huang and Y. Fan, “Modeling uncertainties in emergency service resource allocation,” *Journal of Infrastructure Systems*, vol. 17, no. 1, pp. 35–41, 2011.
  - [20] R. Kondaveti and A. Ganz, “Decision support system for resource allocation in disaster management,” *Annual International Conference of the IEEE*, vol. 3425, pp. 3–6, 2009.
  - [21] A. Y. Chen and F. Peña-Mora, “Decentralized approach considering spatial attributes for equipment utilization in civil engineering disaster response,” *Journal of Computing in Civil Engineering*, vol. 25, no. 6, pp. 457–470, 2011.
  - [22] S. Wang, Y. Wang, and J. Sun, “Compromised resources allocation model for emergency response,” in *Proceedings of the Third International Conference on Natural Computation*, Haikou, China, September 2007.
  - [23] W. M. P. Van Der Aalst, “Inter-organizational workflows: an approach based on message sequence charts and petri nets,” *System Analysis and Modeling*, vol. 34, no. 3, pp. 335–367, 1999.
  - [24] 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. E99.D, no. 3, pp. 650–660, 2016.
  - [25] 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.
  - [26] K. A. Schulz and M. E. Orłowska, “Facilitating cross-organizational workflows with a workflow view approach,” *Data & Knowledge Engineering*, vol. 51, no. 1, pp. 109–147, 2004.
  - [27] P. Jiang, X. Shao, H. Qiu, and P. Li, “Interoperability of cross-organizational workflows based on process-view for collaborative product development,” *Concurrent Engineering*, vol. 16, no. 1, pp. 73–87, 2008.
  - [28] 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.
  - [29] Q. Zeng, F. Lu, C. Liu, H. Duan, and C. Zhou, “Modeling and verification for cross-department collaborative medical business processes using extended petri nets,” *IEEE Transaction on System, Man and Cybernetics: Systems*, vol. 46, no. 1, pp. 349–362, 2015.
  - [30] C. Liu, Q. Zeng, H. Duan, and F. Lu, “Petri net based behavior description of cross-organization workflow with synchronous interaction pattern,” *Communications in Computer and Information Science*, vol. 495, pp. 1–10, 2015.
  - [31] C. Liu, H. Duan, Q. Zeng, 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.
  - [32] 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.
  - [33] 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.
  - [34] C. Liu, Q. Zeng, H. Duan et al., “Petri net based data-flow error detection and correction strategy for business processes,” *IEEE Access*, vol. 8, no. 1, pp. 43265–43276, 2020.

- [35] T. Lu, C. Liu, H. Duan, and Q. Zeng, "Mining component-based software behavioral models using dynamic analysis," *IEEE Access*, vol. 8, pp. 68883–68894, 2020.
- [36] C. Liu, Q. Zeng, L. Cheng, H. Duan, M. Zhou, and J. Cheng, "Privacy-preserving behavioral correctness verification of cross-organizational workflow with task synchronization patterns," *IEEE Transactions on Automation Science and Engineering*, vol. 34, pp. 1–12, 2020.
- [37] C. Liu, Y. Pei, L. Cheng, Q. Zeng, and H. Duan, *Sampling Business Process Event Logs Using Graph-Based Ranking Model*, Concurrency and Computation: Practice and Experience, London, UK, 2020.
- [38] C. Liu, H. Li, Q. Zeng, T. Lu, and C. Li, "Cross-organization emergency response process mining: an approach based on petri nets," *Mathematical Problems in Engineering*, vol. 2020, Article ID 8836007, 12 pages, 2020.