IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 22, 2021, accepted March 7, 2021, date of publication March 10, 2021, date of current version March 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3065360

A Novel Modeling Method of the Crowdsourcing Design Process for Complex Products-Based an Object-Oriented Petri Net

LIJUN SONG[1], YINGYING FU[1], JIAFU SU[2], KANGQU ZHOU[1], AND MEI LONG[3]
[1]College of Mechanical Engineering, Chongqing University of Technology, Chongqing 400054, China
[2]National Research Base of Intelligent Manufacturing Service, Chongqing Technology and Business University, Chongqing 400067, China
[3]Zhubajie Company Ltd., Chongqing 401120, China

Corresponding author: Jiafu Su (jeff.su@cqu.edu.cn)

This work was supported in part by the National Key Research and Development Project of China under Grant 2018YFB1700804, and in part by the Youth Foundation of Ministry of Education of China under Grant 19YJC630141.

ABSTRACT Crowdsourcing design for complex products has open objectives and requires collaboration among different fields. It is difficult to use a single approach to describe the process in which the number and interaction of tasks and steps of tasks are changed dynamically depending on the execution state. Motivated by modeling the collaborative business process of crowdsourcing design in complex products, a crowdsourcing hierarchical process modeling approach based on an object-oriented Petri net is proposed. From the perspective of organization and task, the crowdsourcing object interaction and design process are modeled separately. First, an organization layer modeling approach based on expert review is proposed to describe the information transfer relationship among crowdsourcing objects from the organization perspective. Then, the task layer modeling approach based on the component decomposition idea is proposed. From the perspective of the task, the task is decomposed by using a design structure matrix (DSM) and a product structure tree to model the interaction of the design process. The object-oriented Petri net-is used to realize the representation of the model. Finally, the method is verified by taking the model of the crowdsourcing design of an air cooler as an example. The model subnet is analyzed by means of an overlay tree and incidence matrix, and the reliability and robustness of the model are proven. This method provides a reference for the process modeling of the crowdsourcing design of complex products.

INDEX TERMS Crowdsourcing, process modeling, object-oriented, Petri net, design structure matrix.

I. INTRODUCTION

The digital revolution brought about by Industry 4.0 has had a significant impact on the industrial model of manufacturing. As a manifestation of an enterprise's core competitiveness, innovation has become an important basis for measuring an enterprise's survival and development in the ever-changing market. An increasing number of companies try to acquire people's creativity through open and innovative models to maintain their innovative advantages [1], [2]. Traditional manufacturing is no exception. For the emerging industrial trend of the Internet and intelligent manufacturing, through the business model of cross-processing, cross-organization, and cross-industry, enterprises try to enhance personalized product design to promote complex product design patterns, schemes, and technologies, and improve work efficiency [3]–[5]. Therefore, a way to outsource work and tasks to widely distributed and diverse network experts or enthusiasts through Internet platforms and tools to obtain creative ideas and crowdsourced solutions (crowdsourcing) is considered the most effective innovation model [5]–[9].

Since the concept of "crowdsourcing" was put forward by Jeff [10] in 2006, crowdsourcing has achieved remarkable success in realizing micro-tasks such as information collection, graphic design, and software development. Complex crowdsourcing tasks require professional knowledge to solve practical problems, and they have a certain degree of creativity. Meanwhile, its goals are complex and open-ended, and it requires a long time of collaborative work by technicians in different fields to complete. Therefore, complex product crowdsourcing design tasks usually need to be decomposed and modularized into subtasks of an unknown quantity.

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

Crowdsourcing will lead to unpredictable task decomposition results and a task execution process. The process's key characteristics are as follows: (1) The number of tasks and the execution order are unknown; (2) The collaboration between multiple personnel is a complex dynamic process, limited to modeling these characteristics of the process with a single modeling method.

Modeling the process of complex product crowdsourcing design is the basis and core of designing and analyzing the operating system of a complex product crowdsourcing design platform. To extend crowdsourcing's scope and value, modeling the complex product crowdsourcing design process has become a current focus. At present, the main work in modeling the crowdsourcing design process of complex products is to model it conceptually [11], [12] and to study its model expression methods in data representation and organizational form. There is a relative lack of research on the process operation of complex product crowdsourcing design.

In this paper, based on the idea of combinatorial modeling, we propose an object-oriented crowdsourcing hierarchical model (OOCHM) strategy inspired by the collaboration of large organizations. By introducing the object-oriented Petri net, from the perspective of organization and task, we can flexibly model the dynamic, collaborative business process of complex product crowdsourcing design tasks. Moreover, we emphasize the role of experts in the crowdsourcing design of complex products and strengthen the professionalism and collaboration of crowdsourcing design through expert review. In addition, we decompose the task into multiple design objects, which reduces the task granularity and difficulty of execution, and simplifies the collaborative interaction between objects.

In the rest of this paper, we investigate existing research related to work, analyze the business process of complex product crowdsourcing design, and describe the basic structure and modeling steps of the OOCHM. The feasibility of OOCHM is then illustrated with an example of a crowdsourcing process. Finally, conclusions and future work are presented.

II. RELATED WORK

In this section, we introduce the approach of crowdsourcing design process modeling for complex products through the integration of research literature on crowdsourcing and process modeling methods and compare their advantages and disadvantages.

A. CROWDSOURCING OF COMPLEX OBJECTS

Crowdsourcing is an open-ended and collaborative process, and scholars have studied it in different aspects, including model representation [14], [15], application scenarios [16]–[18], process control [19], and data analysis [20]. However, most studies focus on the macrolevel of crowdsourcing design. A few scholars have rarely been involved in the implementation and operation of the whole crowdsourcing process. Silva *et al.* [19] discussed various crowdsourcing processes and proposed a process framework suitable for most crowdsourcing tasks, but this approach is not suitable for the crowdsourcing process with dynamic changes. A modified crowdsourcing system framework, which can adapt to the increase in the number of crowdsourcing processes and be can applied to complex task objectives, has been proposed [9]. Rouse [25] classified crowdsourcing tasks and defined complex crowdsourcing tasks as tasks that require the use of professional knowledge to solve practical problems, which has a certain degree of creativity. The paper also proposed the modeling idea of decomposing and refining complex tasks into smaller work units.

Recent work has attempted to model complex target crowdsourcing through different model representations. A case-based crowdsourcing conceptual model [26] described tasks as a combination of instances and attributes, which could flexibly carry out single-view modeling of complex crowdsourcing processes. Valentine *et al.* [11] proposed Flash organization, with coded tasks as depersonalized roles and an established hierarchical structure of organizational roles, can adapt to complex crowdsourcing tasks with a different division of labor. However, this method requires a high degree of coordination between computer systems and organizations. which cannot be achieved in the traditional form. Shi *et al.* [12] proposed a business object-oriented workflow modeling method that intuitively models changes in the life cycle of business objects by encapsulating the business data, tasks, and object life cycle of business objects. However, the model is inelastic and cannot be modeled well for crowdsourcing tasks requiring intensive interaction. Embiricos *et al.* [27] proposed expert crowdsourcing significantly improves the design efficiency of the crowdsourcing process through structured communication. Nigam and Caswell [28] proposed a business components that constitute their lifecycle sets based on business data and establish the entire business process model through their interactions.

B. PROCESS MODELING APPROACH

Process modeling methods are focused mainly on formal models, such as the IDEF series, UML, and Petri nets. IDEF series modeling methods have excellent computerized ability, which is aimed mainly at functional process modeling. However, those modeling methods do not have multirole modeling functions, so they are not suitable for dynamic modeling. UML adopts object-oriented technology, which is suitable for modeling large-scale complex systems of entity activities. Nevertheless, it also has a weak ability to express processes dynamically; the Petri net has a strong expression ability for dynamic processes, which therefore makes it easy to analyze and improve the model. A survey [29] shows that a Petri net is one of the popular forms of graphical representation of parallel system modeling, and it has obvious effect in the area of design and verification of system logic. However, traditional Petri nets are not good at model organization and roles.

Existing process modeling methods rely mainly on process-oriented and object-oriented methods. Process-oriented modeling methods are process-centric and focus on the running sequence of activities, making it difficult to describe dynamic processes. The object-oriented method abstractly decomposes the process into hierarchical modules, which simplifies the expression of models, reduces model granularity and is suitable for describing large complex systems [30].

For workflow modeling of complex systems, Sarker [13] first proposed the object-oriented Petri net (OOPN) theory and applied this theory in the modeling of real-time systems. The OOPN theory combines the advantages of object-oriented technology, hierarchical technology, and Petri nets and can model the complex, dynamic and nonabstract problems faced by large-scale complex systems. Then, a lightweight Proclet [31] is built by combining object-oriented techniques with Petri nets, but this Proclet cannot be used for cross-organization modeling. Grobelna *et al.* [32] use interpreted Petri net(IPN) modeling systems. They specialize the place or transitions in the model to independent parallel modules to model and analyze the discrete event system (DES). Due to the limitation of a simple Petri net, the model size cannot be too large.

A hierarchically object-oriented Petri Net (HOOPN) [33] combines object-oriented technology, hierarchical design, and the Petri net. A finer-grained structure is obtained through the hierarchical decomposition of places and transitions. This structure has overcome the problems of traditional Petri nets, such as large models and the inability to dynamically model when describing complex systems. The proposal of a reconfigurable Petri net makes its hierarchical model combined with an object-oriented Petri net suitable for the design and modeling of large-scale process systems with dynamic components [34].

Wisniewski *et al.* [35] summarized the key research in recent years and proposed the research direction of discrete time and hybrid system modeling, such as the formalizaed verification problem of reconfigurable time discrete event systems, Petri net approximate analysis and state analysis. Navarro-Gutiérrez *et al*. [36] analyzed the Petri net system behavior through the mesh structure object, and proposed four Petri net reduction rules to improve the understanding of the network.

C. TASK DECOMPOSITION

A product structure tree is a method to decompose product knowledge into its basic components. Zeng and Gu [37] described the relationship between products, parts, and related tasks through an expanded product structure tree. Different types of leaf nodes were introduced to establish the attribution relationship between nodes, and the extended product structure tree was constructed recursively to obtain the basic blocks with mutual relationships.

The design Structure Matrix (DSM) is a widely used design composition tool that expresses the relationship between complex systems and their entities in a visual and straightforward matrix [38]. The dependencies between different entities can be divided into single-mode and dual-mode dependencies [39], where the single-mode represents the relationship between two entities of the same category, and the dual mode represents the relationship between any two entities of different categories. A two-stage matrix decomposition method [40] was proposed by synthesizing different decomposition standards, which could flexibly model complex products. Bonzo *et al.* [41] analyzed the surgical process characteristics, where the design structure matrix and the multidomain matrix were combined as a domain mapping matrix, and the process communication relationship in the operating room process was modeled. The design task was decomposed via component decomposition [42], and the interaction modeling of components through DSM transformation could achieve the minimum cost decomposition of the design task.

III. THE COMPLEX PRODUCT CROWDSOURCING DESIGN PROCESS

This section will introduce the overall process of complex product crowdsourcing design tasks and draw out the process model of the complex product crowdsourcing design process by analyzing the process.

Complex product crowdsourcing design tasks, which are professional and carried out by several participants at the same time, need to solve the actual product design problems. Each participant has the consistency of task objectives and the execution process independence, so the process is distributed and synergistic. The number of subtasks and the uncertainty of the task decomposition execution process make the process dynamic. Task execution of crowdsourcing design depends on the coordination among participants, and its process can be divided into two layers from the perspective of organization and task, with hierarchical characteristics.

The role of experts in the crowdsourcing process is ignored in the existing research on crowdsourcing process modeling. To enhance the professionalism of crowdsourcing design, this paper takes it as the supervision and review of crowdsourcing design and runs it as the flow of crowdsourcing design (Fig. 1), where the packet ignores multiple participants.

The process description is as follows: (1) The contractee submits a design task to the crowdsourcing platform. (2) The platform decomposes and preliminarily evaluates the task and releases the decomposed task. (3) The platform recruits the contractor according to the decomposed subtask. (4) The contractor carries out the work and submits the results to the experts for review. (5) The experts coordinate the tasks with the contractee and evaluate the task results. (6) The experts submit the task review and results to the contractee. (7) After the contractee confirms the results, the remuneration will be sent to the contractor and the crowdsourcing platform experts.

In the above process, the complex product crowdsourcing design process covers multiple stages divided into relatively fixed business processes and variable design processes (task

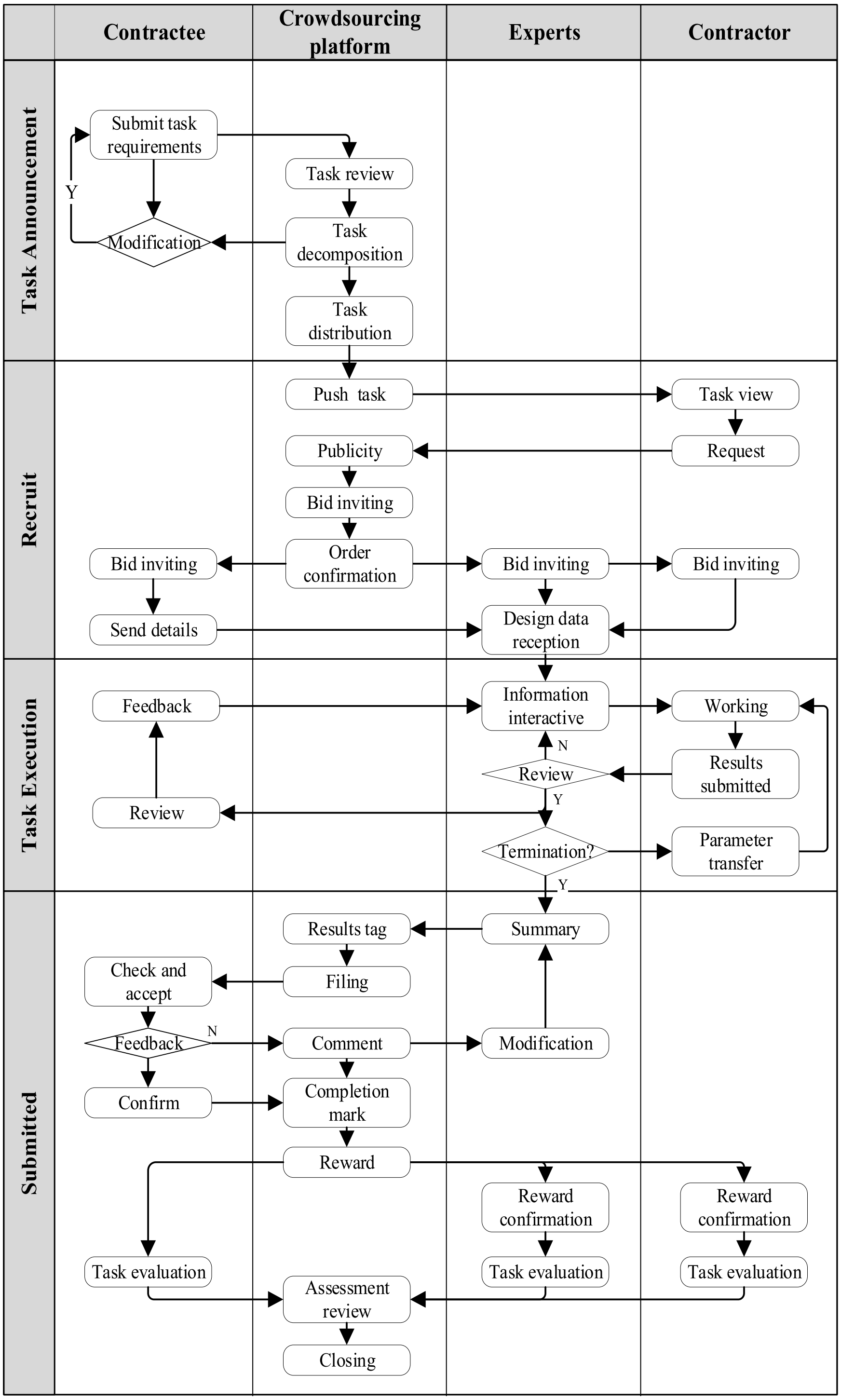


FIGURE 1. Flow chart of crowdsourcing design of complex products.

execution phase). The design process is formed by task decomposition and parameter interaction between subtasks, without considering the internal execution process of the contractor.

In this paper, we model the process as a two-layer model from the perspective of organization and task, according to hierarchical, distributed, and dynamic characteristics. The organizational layer controls the interaction and process relationships among the participants. The task layer models the interaction flow of task information within participants, and this flow is affected by the organization layer constraints. Effective decomposition of complex crowdsourcing design tasks, which are flexible, iterative, and optional process execution paths, is the key to modeling the task layer.

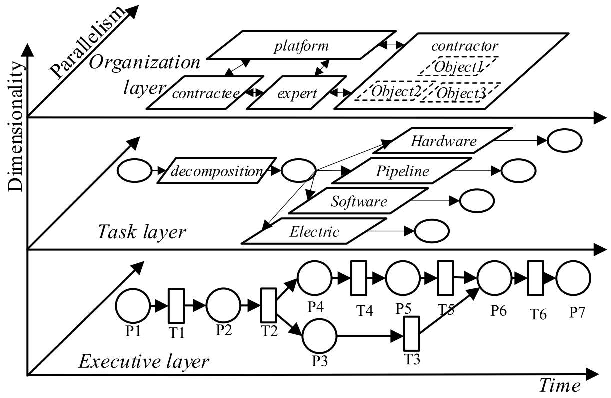


FIGURE 2. Hierarchical process integration model of complex product crowdsourcing design.

Object-oriented Petri nets have obvious advantages in object encapsulation and dynamic concurrent representation of processes in common modeling methods. Considering the modeling complexity, we realized hierarchical modeling of complex product crowdsourcing design through an object-oriented Petri net. The object abstraction function of the object-oriented Petri net is used to represent the interaction between participants in the crowdsourcing design of complex products from an organizational point of view. From the perspective of a task, the crowdsourcing design task was modularized by combining the product structure tree and DSM from the perspective of component decomposition, representing the contractor's internal interaction relationship and facilitating the understanding of the execution process of crowdsourcing design of the complex product at the beginning of modeling.

For the needs of process simulation analysis, the decomposition process of the task layer is mapped to the object-oriented Petri net model to form the object-oriented Petri net layered model of complex product crowdsourcing design.

IV. MODELING METHOD OF COMPLEX PRODUCT CROWDSOURCING DESIGN PROCESS

A. COMPLEX PRODUCT CROWDSOURCING DESIGN PROCESS HIERARCHICAL MODEL

Based on crowdsourcing process analysis, this paper proposes an object-oriented, complex product crowdsourcing design layered process model, as shown in Fig. 2. The model is divided into three layers, including the organization layer, task layer, and execution layer.

Definition 1: The object-oriented crowdsourcing hierarchical model (OOCHM) is defined as:

$$OOCHM = < P, T, E, R_e, R_t > \tag{1}$$

where P is the organization layer model of complex product crowdsourcing design; T is the task layer model of complex product crowdsourcing design; E is the execution layer model of the complex product crowdsourcing design task; $R_e \subseteq P \times E$ is the relationship between organization layer P and executive layer E; and $R_t \subseteq T \times E$ is the relationship between task layer T and executive layer E.

From the perspective of role relationships, organization layer P represents the interaction between the participants and the process progress of executive layer E. Task layer T describes the execution process of execution layer E from the perspective of the task. The subtasks in the executive layer E feed the task results back to the expert of the organization layer for review, making the organization layer and the executive layer interconnect.

Object-oriented crowdsourcing hierarchical modeling is divided into two steps. First, the organization layer system net is established, consisting of the object subnet and its interaction, according to the complex product crowdsourcing design flow. The task is decomposed and modularized from the point of view of product component decomposition, the subtask information transmission process is obtained, and the task layer model is established. Finally, the process is transformed into OOPN to realize the connection with the task layer through the interactive transition between objects.

B. ORGANIZATIONAL LAYER MODELING METHOD

The organization layer describes the organizational relationship and information interaction of complex product crowdsourcing design tasks, modeling task-related personnel as object classes, which are described by a unified representation.

Definition 2 (Object lasses): The object class represents a set of objects with the same attributes in a process, expressed as a four-tuple:

$$O_i = (C, A, \varphi, n) \tag{2}$$

where C represents the name of the object class; A represents the set of tasks r contained in the object class; φ represents the process stages covered by the object class; and $n(n > 0)$ represents the number of objects in the object class.

Definition 3 (Design Objects): The subobject of the contractor is defined as a design object, expressed as $A_i(i > 0)$.

Definition 4 (Task): The task is expressed as $r = \{m, r_{in}, r_{on}, t\}$. where m is the task name, r_{in} is the task input information, r_{on} is the task output information, and t is the expected completion time of the task.

The objects are represented by OOPN. The system network of the organization layer connects object subnets that encapsulate multiple inputs and outputs place through gate transition.

Definition 5 (Organizational Layer Model): Organizational layer model P is expressed as:

$$P = < O, S, G, M_0 > \tag{3}$$

where O is the collection of object classes in the process of product crowdsourcing design; S is the interaction network between objects; G is the set of input and output relationships between various objects; $G = \{G_{ij} | i > 0, j > 0, i \neq j\}$; and M_0 is the initial state of the organization layer model.

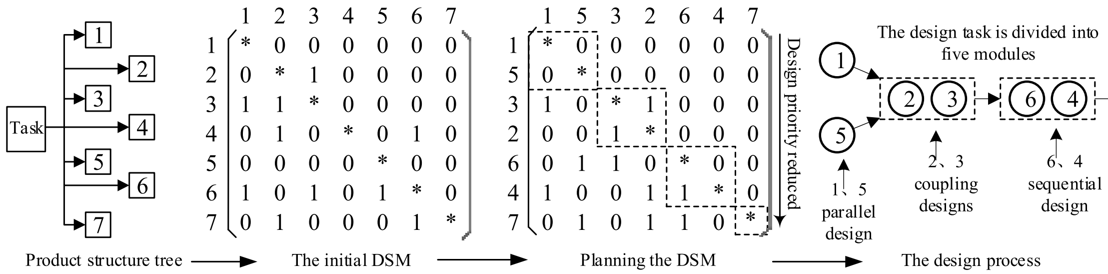


FIGURE 3. Task-level modeling method.

	A	*B*
A	*	0
B	1	*

(a) serial relationship

	A	*B*
A	*	0
B	0	*

(b) parallel relationship

	A	*B*
A	*	1
B	1	*

(c) coupling relationship

FIGURE 4. Matrix representation of the model of the design unit.

C. FILE FORMATS FOR GRAPHICS

The subtasks will be different depending on the products during the execution of the complex product crowdsourcing design task. Therefore, it is difficult to model a variable instance process using a fixed process template. However, the structure and design mode of the product and its components are relatively fixed, so we can dynamically customize the task flow through the product structure. The task layer model decomposes tasks for complex products by way of product component decomposition, which combines the product structure tree and DSM. We will obtain the crowdsourcing subtasks and their relationships of information transfer and establish the design object mode of the task layer through the mapping between DSM and OOPN. The specific modeling method (Fig. 3) steps are as follows:

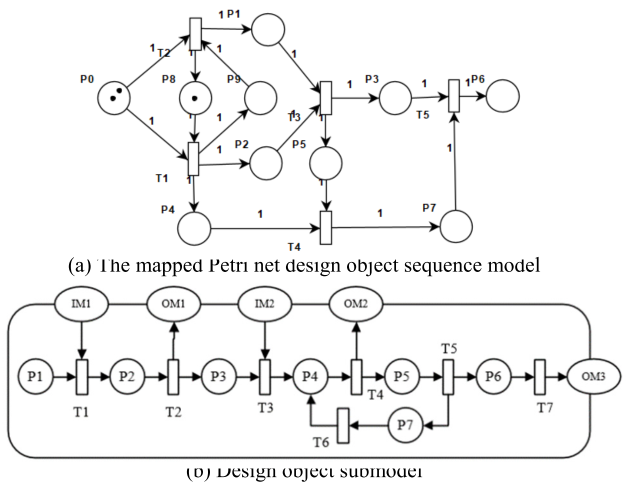
(1) The product is divided into parts according to the product structure tree, and the initial DSM is established through the parameter transfer relationship of the parts;

(2) Plan the initial DSM, identify the fuzzy relationship between tasks, and obtain the design module and its parameter transfer relationships;

(3) According to the dependencies between the design modules divided after planning, the design process of design subtasks is dynamically generated;

(4) The decomposed module and its parameter transfer relationships are transformed by the Petri net to obtain the task layer model.

Design units are obtained by referring to the product structure tree. According to the parameter associations between the design units, there are three associations between the design tasks [43]: sequence, parallel, and coupling relationships. Fig. 4 describes the representation of these three relationships in the design structure matrix. In Fig. 4, the diagonal element "∗" represents the task itself. The value "1" indicates that there is information interaction between the two tasks. The value "0" indicates that there is no information interaction between tasks.



(a) The mapped Petri net design object sequence model

(b) Design object submodel

FIGURE 5. Petri net model of design structure matrix mapping.

According to the design units obtained, the n^{th}-order product design structure matrix is established as follows:

$$DSM_{n\times n} = \begin{matrix} & T_1 & T_2 & T_3 & \cdots & T_n \\ T_1 \\ T_2 \\ T_3 \\ \vdots \\ T_n \end{matrix} \begin{pmatrix} T_1 & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & T_2 & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & T_3 & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & T_n \end{pmatrix} \quad (4)$$

If the matrix element $a_{ij}(i \neq j)$ with a value of 1 indicates that task T_i requires a design parameter from task T_j, there is no interaction when the value is 0.

Through the design structure matrix, the timing and control relationship between the design subtasks are obtained, and the design units are reclassified according to the following transformation principles:

Principle 1 (Minimum Coupling Principle): The interaction between design subtasks is as low as possible;

Principle 2 (Lowest Cost Principle): The design and manufacturing similarity of the design units within each design subtask is as high as possible;

Principle 3 (Minimum Time Principle): Design subtasks without information input should be executed as early as possible, and design subtasks without information output should be executed as late as possible.

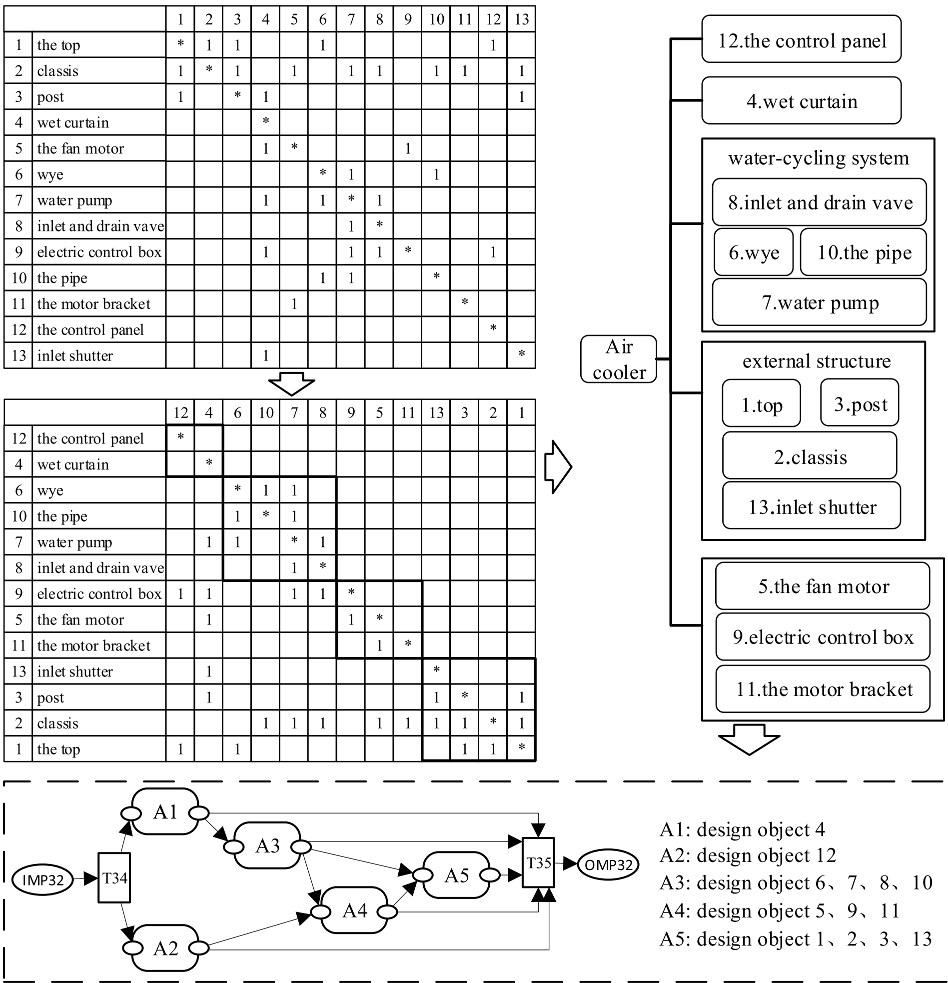


FIGURE 6. Crowdsourcing design task layer model of air cooler.

The organization layer model is the system net of object-oriented Petri nets, represented mainly by the information transmission between the contractee, platform, contractor, and expert. The Petri net mapped by the task layer is shown as the task transmission network of the organization layer. To establish a complete execution layer model, tasks are divided into preset and postset tasks according to the complete order, corresponding to the former and subsequent set in the Petri net. If tasks a and b are sequential executions, i.e., there is parameter transfer. Then task a is the preset task of task b, and task b is the postset of task a. If two tasks are coupled, they are the preset and postset of each other. The Petri net conversion process follows:

(1) The parameter transfer and module division relationships between tasks can be obtained from the task layer DSM (the design flow of Fig. 3);

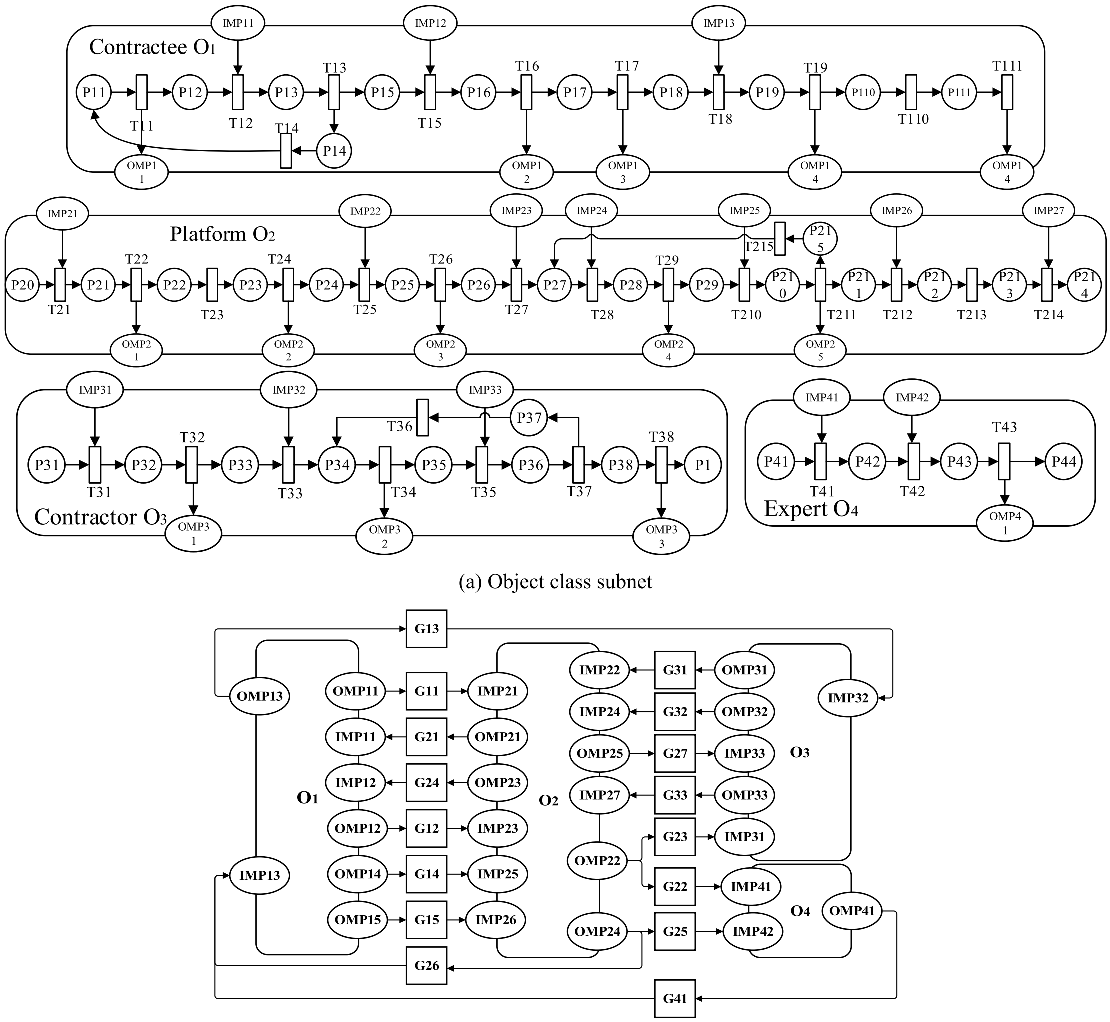
(2) The design task is divided into the execution status and execution action. They are, respectively represented by place and transition in the Petri net. The place represents the state of task execution, and the transition represents the action of the task execution. As shown in Fig. 3, for example, five design modules were obtained after task decomposition, corresponding to five places and five transitions, $P = \{P1, P2, P3, P4, P5\}$ $T = \{T1, T2, T3, T4, T5\}$. A directed arc connects the places and transitions to form a basic Petri net model.

(3) The input place and output place are added to the basic Petri net, a directed arc connecting the input and output place to form a complete Petri net of the design object.

Following the above steps, transform the design process (Fig. 7) into the Petri net process (Fig. 8(a)). The transition $T1 \sim T5$ represents the five design modules in Fig. 7. The object subnet modeling of the organization layer is established, as shown in Fig. 8(b). The contractor contains multiple object subnets of design objects, and each object subnet is executed as a transition in the contractor object class.

V. CASE STUDY

The object-oriented crowdsourcing hierarchical modeling approach is shown in the previous section. In this section, we illustrate the feasibility of the object-oriented crowdsourcing design modeling approach in complex products and



(a) Object class subnet

(b) Organizational System Network

FIGURE 7. **Crowdsourcing design organization model of air cooler.**

illustrate the modeling method of object-oriented crowdsourcing design modeling dynamic, uncertain processes with an air cooler example.

A. THE HIERARCHICAL MODEL OF CROWDSOURCING DESIGN FOR AIR COOLERS

The goal of this task is to decompose the tasks of the air cooler into smaller tasks through the network, obtain task modules with minimal information interaction, solve these task modules, and then merge from the bottom up to obtain the design structure of the air cooler.

This design process is dynamic, including an uncertain number of design modules. Therefore, first, we decompose the air cooler task through the combination of a product structure tree and DSM. The air cooler's crowdsourcing design task was decomposed through product component decomposition, as shown in Fig. 6.

The specific modeling process is as follows: (1) Establish a directed graph of the design structure based on the product structure tree; (2) Establish an initial DSM for the product based on the directed graph; (3) Plan the initial DSM to obtain the component types and the affiliation and timing relationships between them, remapping the product structure tree; and (4) Dynamically iterate the product design process based on the product structure tree until the design process includes all the design components and obtains the product design flow. (5) Convert the design process to an object-oriented Petri net. After task decomposition, the air cooler's crowdsourcing task can be divided into five design subtasks A1 $\sim$ A5 with their information transmission relationship. The five design subtasks are executed simultaneously, and the k-layer model of the air cooler is established (Fig. 6).

Define the main agents of the crowdsourcing design process of air coolers: contractee, platform, contractor, and experts are object classes. According to the crowdsourced design flow (Fig. 1), we establish the object subnet of the air cooler, the object including contractee, platform, expert, and contractor, as shown in Fig. 7(a). The gate transition is used

TABLE 1. The meaning of gate changes.

Gate transition	Meaning
$G11$	The contractee sends the task requirements to the platform
$G12$	The contractee sends the result of selecting the contractor to the platform
$G13$	The contractee sends the design data to the contractor
$G14$	The contractee submits the modification of task result to the platform
$G15$	The contractee sends task confirmation to the platform
$G21$	The platform sends the feedback results of the requirements to the contractee
$G22$	The platform pushes the task information to the contractor
$G23$	The platform pushes the contractor's request to expert and contractee
$G24$	The platform pushes task results to experts for review
$G25$	The result modification request is fed back by the platform to the contractor
$G31$	The contractor sends a task request to the platform
$G32$	The contractor submits the task results to the platform
$G33$	Contractor confirms the receipt of remuneration
$G41$	The expert submits the result review report to the contractee

as the interface to connect the input and output repositories of the object subnet to obtain the air cooler organization layer OOPN model (Fig. 6(b)). The design object of contractor O3 interacts through the information transfer process of the design object shown in Fig. 6.

The meaning of the gate transition in Fig. 7 is shown in Table 1. From the above, we obtain the layered model of the air cooler model crowdsourcing design.

As shown in Fig. 6, the resulting design flow shows that all design modules run in parallel, regardless of execution time. Its flow represents the information transmission relationship between design modules. Each design module is running separately when the output information means that the module design phase is complete, and sends the information to the platform and expert examination (Fig. 7(b)). If it passes, the information is merged and passed to the next task until all tasks are completed. The tasks are finally merged on the platform, and the employer obtains the design result of the air cooler. Through OOCHM, the information transfer relationship of the complex product crowdsourcing design process can be displayed intuitively, different design processes can be generated dynamically-according to different products, and flexible modeling of the process can be realized.

B. MODEL ANALYSIS

The structure and operation of the process can be simulated by the properties of a Petri net by modeling the crowdsourcing design process of the cooling fan. Therefore, after the establishment of the Petri net model, the feasibility of the model needs to be verified through the analysis of the accessibility, safety, boundedness, activity and conservation of Petri nets through reachable identification graphs, coverage trees, correlation matrices, etc.

In this part, the state space analysis and incidence matrix are used to analyze the properties of the packet square subnet

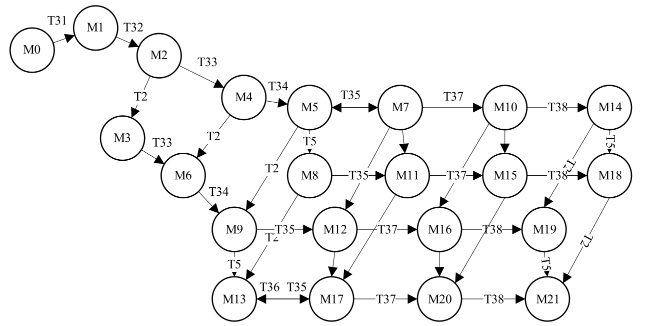


FIGURE 8. The state space diagram of O4.

TABLE 2. Identity in the status diagram.

	I1	I2	I3	P1	P2	P3	P4	P5	P6	P7	P9	Pe1	Pe2	Pe3
M0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
M1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
M2	0	1	1	0	1	1	0	0	0	0	0	0	0	0
M3	0	0	1	0	1	0	1	0	0	0	0	0	0	0
M4	0	1	1	0	0	1	0	0	0	0	0	1	0	0
M5	0	0	1	0	1	0	0	1	1	0	0	0	0	0
M6	0	0	1	0	0	0	1	0	0	0	0	1	0	0
M7	0	0	0	0	1	0	0	1	0	1	0	0	0	0
M8	0	0	1	0	1	0	0	0	1	0	0	0	1	0
M9	0	0	1	0	0	0	0	1	1	0	0	1	0	0
M10	0	0	0	0	1	0	0	1	0	0	1	0	0	0
M11	0	0	0	0	1	0	0	0	0	1	0	0	1	0
M12	0	0	0	0	0	0	0	1	0	1	0	1	0	0
M13	0	0	1	0	0	0	0	0	1	0	0	1	1	0
M14	0	0	0	0	1	0	0	1	0	0	0	0	0	1
M15	0	0	0	0	1	0	0	0	0	0	1	0	1	0
M16	0	0	0	0	0	0	0	1	0	0	1	1	0	0
M17	0	0	0	0	0	0	0	0	0	1	0	1	1	0
M18	0	0	0	0	1	0	0	0	0	0	0	0	1	1
M19	0	0	0	0	0	0	0	1	0	0	0	1	0	1
M20	0	0	0	0	0	0	0	0	0	0	1	1	1	0
M21	0	0	0	0	0	0	0	0	0	0	0	1	1	1

in the fan crowdsourcing design process model established above. In contractor subnetwork O_3 shown in Figure 7, the message input place is I_i ($I1, I2, I3$) and the message output place is Pei ($Pe1, Pe2, Pe3$), so the set of places is M. According to the analysis method proposed in the literature [44], the state space accessibility diagram (Figure. 8) is constructed, where the state data are shown in Table 2.

As shown in Figure 8, the relevant properties of O_3 are analyzed.

(1) In the theory of Petri nets, a system is bounded if and only if there is unbounded quantity ω in its identification. From the analysis of the marker diagram in Figure 8, there is no unbounded quantity ω in the marker vector of all nodes, so O_3 is bounded.

(2) Since O_3 is bounded, the bounds of each place can be $B(p_i) = 1$, so place P_i is safe. Obviously, $B(O_3) = 1$, so net O_3 is safe.

(3) Calculating the coverability tree shows that for any $M \in R(M_0)$. $R(M) \subseteq R(M_0)$. Therefore, net O3 is reachable.

(4) Since O_3 is reachable, under any reachable mark, every transition $T_i (i = 1, 2, \ldots, 9)$ in the network is likely to obtain

the right of occurrence, which indicates that every transition T_i in O_3 is alive, so O_3 is also alive.

VI. CONCLUSION

In this paper, we propose a crowdsourcing modeling approach based on object-oriented Petri nets. The approach can model not only the control flow of a complex system but also the dynamic, uncertain flow. Meanwhile, the hierarchical model based on different perspectives also provides help for the refinement of the model.

We present an object-oriented Petri net organization layer modeling approach based on expert review to emphasize the importance of experts. The model relies on expert decisions to ensure the complete quality of crowdsourcing design tasks and realizes the control of the crowdsourcing process through the input-output place. We present a task layer modeling approach based on product component decomposition, which decomposes complex products into design subtasks through a product structure tree and a design structure matrix and maps it to Petri nets to obtain the information transfer relationship model between design objects. Finally, we take the crowdsourcing design of the air cooler as an example to realize the crowdsourcing design process modeling of complex products.

This modeling method makes full use of the advantages of OOPN, describes the interaction between objects of complex products from different perspectives, and realizes process encapsulation and connection. This method provides a reference for the design of a complex product crowdsourcing design system. While modeling the process, it is useful to realize the automation of process modeling. We expect future work to include the processing of process data flows to support processes.

REFERENCES

[1] K. Laursen and A. Salter, "Open for innovation: The role of openness in explaining innovation performance among U.K. Manufacturing firms," *Strategic Manage. J.*, vol. 27, no. 2, pp. 131–150, Feb. 2006.

[2] S. Jiafu, Y. Yu, and Y. Tao, "Measuring knowledge diffusion efficiency in R&D networks," *Knowl. Manage. Res. Pract.*, vol. 16, no. 2, pp. 208–219, Apr. 2018.

[3] J. Füller, "Refining virtual co-creation from a consumer perspective," *California Manage. Rev.*, vol. 52, no. 2, pp. 98–122, Feb. 2010.

[4] B. L. Bayus, "Crowdsourcing new product ideas over time: An analysis of the dell idea storm community," *Manage. Sci.*, vol. 59, no. 1, pp. 226–244, 2013.

[5] S. Qin, D. Van Der Velde, E. Chatzakis, T. McStea, and N. Smith, "Exploring barriers and opportunities in adopting crowdsourcing based new product development in manufacturing SMEs," *Chin. J. Mech. Eng.*, vol. 29, no. 6, pp. 1052–1066, Nov. 2016.

[6] H. Simula and T. Ahola, "A network perspective on idea and innovation crowdsourcing in industrial firms," *Ind. Marketing Manage.*, vol. 43, no. 3, pp. 400–408, Apr. 2014.

[7] A. Afuah and C. L. Tucci, "Crowdsourcing as a solution to distant search," *Acad. Manage. Rev.*, vol. 37, no. 3, pp. 355–375, Jul. 2012.

[8] J. Su, F. Zhang, S. Chen, N. Zhang, H. Wang, and J. Jian, "Member selection for the collaborative new product innovation teams integrating individual and collaborative attributions," *Complexity*, vol. 2021, pp. 1–14, Feb. 2021.

[9] S. S. Bhatti, X. Gao, and G. Chen, "General framework, opportunities and challenges for crowdsourcing techniques: A comprehensive survey," *J. Syst. Softw.*, vol. 167, Sep. 2020, Art. no. 110611.

[10] H. Jeff, "The rise of crowdsourcing," *Wird Mag.*, vol. 14, no. 14, pp. 1–5, 2006.

[11] M. A. Valentine, D. Retelny, A. To, N. Rahmati, T. Doshi, and M. S. Bernstein, "Flash organizations: Crowdsourcing complex work by structuring crowds as organizations," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2017, pp. 3523–3537.

[12] X. Shi, J. Lin, G. Hu, and Y. Yu, *Business Objects—A New Business Process Modeling Approach*. Cham, Switzerland: Springer, 2018, pp. 3–8.

[13] M. Sarker, *An Overview of Object Oriented Design Metrics*. Umeå, Sweden: Umeå Univ., 2005.

[14] D. C. Brabham, "Crowdsourcing the public participation process for planning projects," *Planning Theory*, vol. 8, no. 3, pp. 242–262, 2009.

[15] A. I. Chittilappilly, L. Chen, and S. Amer-Yahia, "A survey of general-purpose crowdsourcing techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 9, pp. 2246–2266, Sep. 2016.

[16] H. Kärkkäinen, J. Jussila, and J. Multasuo, "Can crowdsourcing really be used in B2B innovation?" in *Proc. 16th Int. Academic MindTrek Conf. MindTrek*, 2012, pp. 134–141.

[17] Y. Xiao, C. Li, L. Song, J. Yang, and J. Su, "A multidimensional information fusion-based matching decision method for manufacturing service resource," 2021, doi: 10.1109/ACCESS.2021.3063277.

[18] O. A. Acar, "Motivations and solution appropriateness in crowdsourcing challenges for innovation," *Res. Policy*, vol. 48, no. 8, Oct. 2019, Art. no. 103716.

[19] V. Silva, D. Schneider, L. Oliveira, and J. de Souza, "Towards a framework for supporting community modeling in crowdsourcing systems," in *Proc. Comput. Supported Cooperat. Work Design*, 2017, pp. 36–41.

[20] M.-C. Yuen, I. King, and K.-S. Leung, "A survey of crowdsourcing systems," in *Proc. Privacy Secur. Risk Trust*, 2011, pp. 766–773.

[21] E. Estellés-Arolas and F. González-Ladrón-de-Guevara, "Towards an integrated crowdsourcing definition," *J. Inf. Sci.*, vol. 38, no. 2, pp. 189–200, Apr. 2012.

[22] Z. A. Bozat, "Crowdsourcing as an open innovation tool for entrepreneurship," *World Acad. Sci., Eng. Technol., Int. J. Social, Behav., Educ., Econ., Bus. Ind. Eng.*, vol. 11, no. 5, pp. 1105–1109, 2017.

[23] A. Ghezzi, D. Gabelloni, A. Martini, and A. Natalicchio, "Crowdsourcing: A review and suggestions for future research," *Int. J. Manage. Rev.*, vol. 20, no. 2, pp. 343–363, Apr. 2018.

[24] A. Tarrell, N. Tahmasbi, D. Kocsis, A. Tripathi, J. Pedersen, J. Xiong, O. Oh, and G.-J. de Vreede, "Crowdsourcing: A snapshot of published research," in *Proc. Amer. Conf. Inf. Syst.*, 2013, pp. 962–975.

[25] A. C. Rouse, "A preliminary taxonomy of crowdsourcing," ACIS, Tech. Rep., 2010.

[26] L. Roman and P. Jeffrey, "Conceptual modeling principles for crowdsourcing," in *Proc. 1st Int. Workshop Multimodal Crowd Sens.*, 2012, pp. 3–6.

[27] A. Embiricos, N. Rahmati, N. Zhu, and M. S. Bernstein, "Structured handoffs in expert crowdsourcing improve communication and work output," in *Proc. User Interface Softw. Technol.*, 2014, pp. 99–100.

[28] A. Nigam and N. S. Caswell, "Business artifacts: An approach to operational specification," *IBM Syst. J.*, vol. 42, no. 3, pp. 428–445, 2003.

[29] D. Gomez-Gutierrez, G. Ramirez-Prado, A. Ramirez-Trevio, and J. Ruiz-Leon, "Observability of switched linear systems," *IEEE Trans. Ind. Informat.*, vol. 6, no. 2, pp. 127–135, May 2010.

[30] G. Booch, R. A. Maksimchuk, and M. W. Engle, "Object-oriented analysis and design with applications," *ACM SIGSOFT Softw. Eng. Notes*, vol. 33, no. 5, p. 29, 2008.

[31] R. Lukyanenko and J. Parsons, "Lightweight conceptual modeling for crowdsourcing," in *Proc. Int. Conf. Conceptual Modeling*, 2013, pp. 508–511.

[32] I. Grobelna, R. Wisniewski, M. Grobelny, and M. Wisniewska, "Design and verification of real-life processes with application of Petri nets," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 47, no. 11, pp. 2856–2869, Nov. 2017.

[33] B. Prüfer, "Hierarchical reconfigurable Petri nets," Hamburg Univ. Appl. Sci., Hamburg, Germany, Tech. Rep., 2018.

[34] J.-E. Hong and D.-H. Bae, "Incremental scenario modeling using hierarchical object-oriented Petri net," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 11, no. 3, pp. 357–386, Jun. 2001.

[35] R. Wisniewski, M. Zhou, L. Gomes, M. P. Fanti, and R. Kumar, "Special issue on recent advances in Petri nets, automata, and discrete-event hybrid systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 10, pp. 3484–3487, Oct. 2020.

[36] M. Navarro-Gutiérrez, J. A. Fraustro-Valdez, A. Ramírez-Treviño, and M. Silva, "Problematic configurations and choice-join pairs on mono-T-semiflow nets: Towards the characterization of behavior-structural properties," *Discrete Event Dyn. Syst.*, pp. 1–35, 2020.
[37] Y. Zeng and P. Gu, "A science-based approach to product design theory part II: Formulation of design requirements and products," *Robot. Comput.-Integr. Manuf.*, vol. 15, no. 4, pp. 341–352, Aug. 1999.
[38] T. R. Browning, "On customer value and improvement in product development processes," *Syst. Eng.*, vol. 6, no. 1, pp. 49–61, 2003.
[39] S. Wasserman and K. Faust, *Social Network Analysis: Method and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
[40] L. Chen, Z. Ding, and S. Li, "A formal two-phase method for decomposition of complex design problems," *J. Mech. Des.*, vol. 127, no. 2, pp. 184–195, Mar. 2005.
[41] S. M. Bonzo, D. McLain, and M. S. Avnet, "Process modeling in the operating room: A socio-technical systems perspective," *Syst. Eng.*, vol. 19, no. 3, pp. 267–277, May 2016.
[42] X. Deng, G. Huet, S. Tan, and C. Fortin, "Product decomposition using design structure matrix for intellectual property protection in supply chain outsourcing," *Comput. Ind.*, vol. 63, no. 6, pp. 632–641, Aug. 2012.
[43] T. R. Browning, "Applying the design structure matrix to system decomposition and integration problems: A review and new directions," *IEEE Trans. Eng. Manag.*, vol. 48, no. 3, pp. 292–306, Aug. 2001.
[44] W. Zhehui, *Introduction to Petri Nets*. Beijing, China: China Machine Press, 2006.

LIJUN SONG was born in Shanxi, China, in 1974. He received the B.S. degree from the College of Materials Engineering, Taiyuan University of Technology, China, in 1998, and the Ph.D. degree from the School of Mechanical Engineering, Chongqing University, Chongqing, China, in 2008. He has been an Assistant Professor with the School of Mechanical Engineering, Chongqing University of Technology, since 2008. He has published over 30 articles in international or domestic journals. His research interests include innovation management, knowledge management, and production scheduling.

YINGYING FU was born in Chongqing, China, in 1996. She received the B.S. degree from the School of Mechanical Engineering, Chongqing University of Technology, in 2018, where she is currently pursuing the master's degree. Her research interest includes process modeling and management.

JIAFU SU was born in Cangzhou, Hebei, China, in 1987. He received the B.S. degree from the College of Mechanical Engineering, North University of China, Taiyuan, China, in 2010, and the Ph.D. degree from the School of Mechanical Engineering, Chongqing University, Chongqing, China, in 2017. He has been an Assistant Professor with the Chongqing Key Laboratory of Electronic Commerce and Supply Chain System, Chongqing Technology and Business University, since 2017. He has published over 20 articles in international or domestic journals, including *Kybernetes*, *Knowledge Management Practice and Research*, *Journal of Simulation* and *Sustainability*. His research interests include innovation management, knowledge management, and supply chain management.

KANGQU ZHOU was born in Sichuan, China, in 1967. She received the B.S. degree from the School of Materials Engineering, Xihua University, Chengdu, China, in 1990, and the Ph.D. degree from the College of Mechanical Engineering, Chongqing University, Chongqing, China, in 2003. She has been a Professor with the School of Mechanical Engineering, Chongqing University of Technology, since 2003. She has published over 40 articles in international or domestic journals.

MEI LONG was born in Chongqing, China, in 1976. She received the M.B.A. degree from Chongqing University. She has presided over and participated in nearly ten key research and development projects of the Ministry of Science and Technology, Chongqing Technology Innovation and Application Demonstration Project, Chongqing Science and Technology Planning Project. Her research interests include lean production, intelligent manufacturing, and intelligent platform.