Collaborative Design Implementation on the PN-PDDP Model for the Complex Coupled Rotor Systems

Guoyuan Zhang*, Weigang Zhao,**
Xiutian Yan***

** School of Mechano-Electronic Engineering, Xidian University, Xi'an, CO 710071*
China (Tel: 0086-29-88202470; e-mail: gyzhang@xidian.edu.cn).
*** Xi'an Aerospace Propulsion Institute, China Aerospace Science and Technology Corporation (CASC) ,*
Xi'an, CO 710100,China(e-mail: zgydyx@126.com)
**** Design Manufacture & Engineering Management, University of Strathclyde,*
Glasgow, UK, (e-mail: x.yan@strath.ac.uk)}

Abstract: By considering what is necessary to implement a multi-component collaborative design methodology for the complex product, a PN-PDDP (Petri Nets for Performance Driven Design Process) model based on extended Petri nets is presented. The methodology and model are used to design a complex and multi-component high-speed turbopump rotor system for a liquid rocket engine. The study on the definitions of the features of the coupling design process, such as the sequence of concurrent activities, a reachability tree, a design cycle tree and design conflicts, are presented and the solving methods for these features are proposed. The computer simulation for these features by a self-programming emulating package is shown. The coupling design process for a specific multi-component high-speed turbopump is carried out, and the matrix for the coupling design process and software is developed. The results show that the PN-PDDP model based on the Petri nets can effectively couple existing knowledge resources, solve any conflict arising from different knowledge, and achieve an optimal strategy for the performance driven design process.

Keywords: Design methodology, Petri nets, computer simulation, coupled design, knowledge resources.

1. INTRODUCTION

A past study has presented a PN-PDDP (Petri Nets for Performance Driven Design Process) model based on extended Petri nets so as to enhance design efficiency and quality after analyzing the design process of a performance-driven multi-component coupling rotor system for a liquid rocket engine (Zhang et al., 2016). This defined the performance characteristics library, the characteristics feature library, the traces Token, and firing rules. However, solving methods for the model should be developed, and characteristics parameters should be created for the flow process of varied coupling information and also the performance-driven conflict solving mechanism when adopting the model, so as to meet the requirements of extant rules and feature library. To this end, the present study examines characteristics defining and parameter solving of the model. Computer simulation of the design process is used so as to provide a supporting tool and platform for the digital design and integration of the complex multi-component coupling rotor system (typified by the high-performance high-speed turbopumps).

There have been a number of studies which build Petri nets-based models with varied targets and proposed solving methods. For example, Fang et al. built an object-oriented colored Petri net mode for an automotive electronic system to solve its characteristic parameters (Fang et al., 2011). For a collaborative design system, HUANG H. Z. et al established temporal and resource coordination mechanisms based on Petri Nets, these offer strong support for the performance evaluation of the collaborative design (Huang et al., 2010). Salinas F.S et al proposed a control scheme to regulate the flying capacitor voltages and the output current by using a Petri net methodology (Salinas et al., 2016).

Considerable attention has also been given to the application of Petri nets to the full life cycle for designing and the manufacturing mechanical products, such as in the assembling and disassembling, manufacture, and design stages. For example, HSIEH F.S. investigates the disassembling sequence optimization strategy (Hsieh, 2008). ERDEN Z. developed a virtual prototype for the logical behavior of robot design with Discrete Event System Specification (DEVS) and Petri Net formalism (Erden, 2011). POPESCU C. et al. discussed how to automatically incorporate resources in a Petri-net-derived model of flow that is modifiable at runtime to reflect and influence the routing in a manufacturing line (Popescu et al., 2012). ISSAOUI L. et al. deals with automatic sequence generation for selective disassembly of mechanical product (Issaoui et al., 2015). BASAK O. et al. presented the design and the implementation of a Petri net (PN) model for the control of a flexible manufacturing system (FMS) (Başak and Albayrak, 2015). Luo Y. T. et al. introduced a method for the integration of multi-layer product representation and optimal searching in product selective disassembly planning (Luo et al., 2016).

With respect to modeling the design process with Petri nets, XU Q. L. et al. applied timed colored Petri nets (TCPN) to model various elements of the product variant design process (Xu and Jiao, 2009). KARNIEL A. et al. presented a formal translation of the DSM-based plan to a process-scheme model, the DSM net, which can be executed and simulated (Karniel and Reich, 2011); then, they also developed a multi-level modelling approach for supporting the management of evolving new product development (NPD) processes (Karniel and Reich, 2013). WANG T. R. et al. proposed a process planning framework, established a three-dimensional collaboration model, and analyzed nine collaboration types among activities (Wang et al., 2012). CHENG F. F. et al. presented the information constraint net (ICN) to represent the complex information constraint relations among design activities involved in the building design process (Cheng et al., 2013). ARAZ M. et al. focused on behavioural representation and simulation of design concepts via discrete event system specification formalism and Petri Nets (Araz and Erden, 2014).

These studies demonstrate the effectiveness of Petri nets in mechanical design or product design. Nonetheless, most of them concern particular projects and offer concrete results, and so far no study aims at a solving method for the multi-components collaborative design process. This study, therefore, sets out to obtain such a solving method so as to provide some guidance on conflict decoupling and collaborative design in the multi-source information coupling design process for the rotor system in a liquid rocket engine.

2. DESIGN METHODOLOGY AND MODEL FOR THE COMPLEX PRODUCT

2.1 Performance-driven design method

The most important requirement for designing the complex product or system is to meet the performance of product, especially in the actual engineering design field. So the performance-driven design method was proposed (Zhang et al., 2016). The principle purpose of this design methodology is to build the performance characteristic model. This model is a hierarchical model, and the high-level performance characteristics include the overall performance characteristics of the product or system while the low-level performance characteristics specify the high-level performance characteristics. The combination of the low-level performance characteristics will lead to the achievement of the high-level performance characteristics. Meanwhile, in this design methodology, all structural features of the product or system constitute the design structural characteristic model. This structure characteristic model is built based on the hierarchical model of the product's assembly relationship, and the high-level structure feature can be obtained by guaranteeing the low-level or substructure assembly relationship.

But, in the initial stage of the actual engineering design, the performance requirement of product is obtained by analyzing or inquiring the user's demand. Hence, the performance characteristic model is not complete, and reflects only part of the performance characteristics of the product. These performance characteristic is in the high-level layer and is also abstract, and maybe cannot be mapped to the existing structural features model; so the further decomposition of the performance characteristics need to be carried out. The decomposition process is called the performance decomposition process, which will ensure that the performance can be concretized and low-level performance characteristics are developed. The special design activities are divided and are finished in this process. That is, for achieving the goal of meeting the high-level performance, the decomposition of the low-level performance activities are performed. After the implementation of the entire design activities' decomposition, a complete product performance characteristics model is developed.

At the same time, in the initial stage of the design, the structural characteristics model of the product can also only partially meet the initial performance requirements of the product, and it is not complete. The constant modification of structure characteristic model is a key step for building the final model. The goal of each modification is to make the product structure characteristic model further meet the performance characteristic model, and the modification process can be called the design performance-structural mapping process. The structure characteristic model is regarded as the final model only when it can realize the entire mapping relationships between all performance and structural features of the designed product.

As described above, the core of the performance-driven design methodology for the complex product is to construct two models, namely performance characteristic model and structural characteristic model. In the design process, the performance decomposition process and structural design process is not carried out independently, and it is an interactive process. Hence, the design process is a classic distributed discrete system. The decomposition of the performance characteristic model is based on the existing structural model, and the high-level performance characteristics can only be decomposed after determining the special corresponding structure features.

2.2 Design process and model

At present, three kinds of design process model, as information model, organization model and behavior view model have been used to express the design process. The design process for the complex product or system, such as the multi-component coupling system, can be implemented by solving design process model. For the special design process, one of above three models can be selected optimally. The definitions and features of three design models are shown as follows,

The information model is used to express the mutual relations between the various information entities in the design process, and define the information flow characteristics of the different entities.

The organization model can be used to describe the relationship between the design activities and the design roles in the design process, and determine who will perform the various design activities and define the assignment of the design roles and their mutual dependencies.

Behavior view model can also be called as a design logic model, and it is used to study mainly the implementation process of the design, which includes the implementation time of the special design activities, implementation process and form, feedback and iteration form, conditional criterion, the guidelines for the start and end, etc.

The above three types of models can be constructed and solved with the different methods and mathematical tools. At present, there are three types of methods and mathematical tools, namely, object-oriented description method, the Petri net and its extended model description method, and the integrated definition for function modeling (IDEF map) description method.

In this paper, the design process model is developed by replying mainly two following questions,

One is how to ensure that the constructed design process model is sufficient to describe the actual design process in detail.

The other is that what kind of design process model is appropriate, that is, which model can provide a reasonable way to analyze the phenomenon in the design process, and easily get the appropriate solution.

In this paper, the high-speed turbopump rotor system is a classic complex system (or product) which includes the multi-components and their coupling. The design for this system in the actual engineering field is also a typical off-site collaborative design and a classic distributed discrete event system design. The design process involves a lot of the information flow and interaction between the various distributed design entities and design activities. By considering the characteristics of Petri nets, it is a feasible method to model and analyze the design process of this off-site collaborative design system. The design model and methodology are developed by the following two steps,

Firstly, the traditional Petri net is reconstructed and the extended Petri net model which is suitable for describing the cooperative design process is established.

Secondly, the extended Petri net model is solved, and the characteristic parameters of the relevant design process are obtained.

Based on the product design theory for meeting the performance-driven requirement, the design process is described by using the extended Petri net. The advantage of this method is that the method can not only ensure the accuracy of the design model, but also make the Petri net model more lucidity. In addition, the solution for the Petri net with the mathematical tools is more convenient and effective, which will provide an important foundation for modeling and managing the collaborative design process of the complex product or system.

3. PETRI NETS MODEL FOR THE DESIGN PROCESS OF THE MULTI-COMPONENT HIGH-SPEED TURBOPUMP

3.1 General formalism of the design process

The collaborative design process of the complex distributed system is composed of a series of design activities. The model of the design process includes the integral performance characteristic model and product structure characteristic model, and the correlativity between the performance and structure model is effective and accurate. The general formalism for the design process can be expressed as follow,

$$DP:\{\sum_{m}^{i=0} A_i,[D]_0,[P]_0 \mid [D]_0 \to [D]_i,[P]_0 \to [P]_i,[P]_i \to [P]_j\} \quad (1)$$

Where, *DP* represents the design process, *A* is a special design activity, [*P*]and [*D*] are the performance and structure characteristic model respectively. *m* is the total number of design activities; Subscript *i*, *j* represents respectively the *i* and *j*th design activity, *i*, *j* = 0 to *m*.

3.2 Petri Nets for Performance-Driven Design Process (PN-PDDP)

A PN-PDDP is composed of four elements as follow,

$$PN\cdot PDDP=(S,T;F,M_0) \quad (2)$$

Where, PN-PDDP is the Petri Nets for Performance Driven Design Process, *S* is the set of PN-PDDP performance and structure characteristics parameters, *T* is the set of design activities, *F* is the set of the relationships between the performance and structure characteristic parameters of the design activities, that is, $F\subseteq(S\times T)\bigcup(T\times S)$, and M_0 describes the initial state of PN-PDDP.

Define that the set $X=U\bigcup T$, then *X* represents the element set of PN-PDDP. If $x\in X$, then the new sets are defined as follows,

$^*x$ is the before set of *x*, and $^*x=\{x\mid(x,y)\in F\}$;

x^* is the after set of *x*, and $x^*=\{y\mid(x,y)\in F\}$.

The necessary and sufficient conditions for PN-PDDP are as follows:

$$\begin{cases} S\bigcap T=\varphi \\ F\subseteq(S\times T)\bigcup(T\times S) \\ S\bigcup T\neq\varphi \\ dom(F)\bigcup cod(F)=S\bigcup T \end{cases} \quad (3)$$

Where, $dom(F)=\{x\mid\exists y:(x,y)\in F\}$; $cod(F)=\{x\mid \exists y:(y,x)\in F\}$.

The libraries of PN-PDDP are classified into two categories: performance and structure libraries (S_P and S_D), and the former describes the performance characteristics while the latter involves the product structure characteristics.

3.3 PN-PDDP model of the multi-component high-speed turbopump

Fig.1 presents the PN-PDDP model for the multi-component coupling turbopump rotor system in line with the definition of PN-PDDP. The whole design process for this model is introduced in our previous study (Zhang et al., 2016). Each symbol in Fig.1 is shown in Table 1 of the reference.

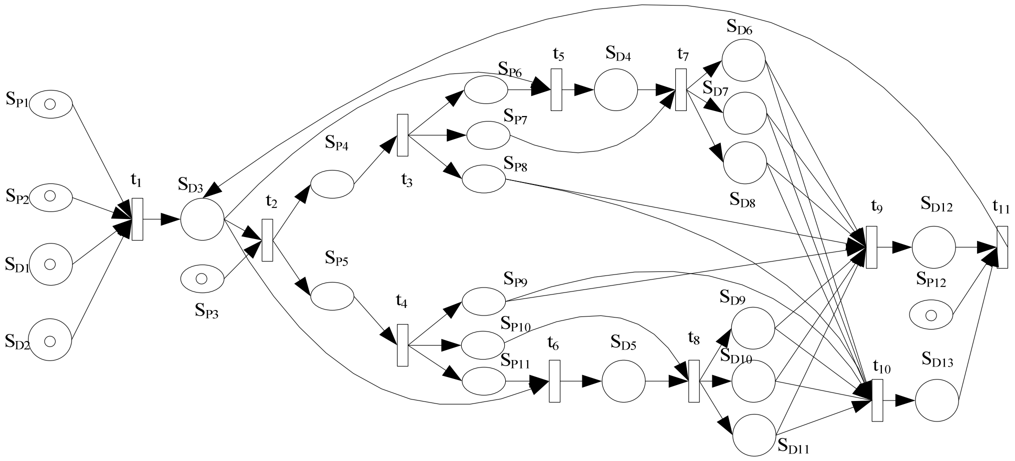


Fig. 1. PN-PDDP for the multi-component coupling turbopump rotor system (Zhang et al., 2016).

Fig. 1 shows the complex PN-PDDP design process where the design starts with the rotor's structure addressing a holistic requirements on the characteristic parameters of the rotor positioning and transmitting performance, the rotor system's components structure and the turbopump's shell structure parameters. This generates the rotor structure characteristics parameters. The bearing assembly is designed based on the load carrying needs of the rotational support, producing the radial and axial load carrying characteristics. Next, performance decomposition is achieved on radial multi-component coupling, thrust multi-component coupling, load capacity, and dynamics performance. The rotor and stator is then designed based on all these performance requirements. What follows is the design of the power amplifier controller and sensor according to the rotor and stator's structural characteristics and performance requirements. Finally, a dynamic check is made. If it does not meet the requirements, the process would iterate through revising the rotor's structural characteristics.

4. DETERMINATION OF CHARACTERISTICS PARAMETERS IN THE PN-PDDP

This section elaborates on the procedure for determining characteristics parameters and obtaining solving methods with the help of a simplified multi-component coupling rotor system design PN-PDDP model, as shown in Fig. 2. The symbol and shape in Fig.2 are introduced in Table 1 of the reference (Zhang et al., 2016).

4.1 Incidence matrix

It is determined that the PN-PDDP includes the library set $S=\{s_1,s_2\cdots s_n\}$ and the activity set $T=\{t_1,t_2\cdots t_m\}$, and the matrix $C=\left\|c_{ij}\right\|(1\le i\le n,1\le j\le m)$ be the incidence matrix of the PN-PDDP, where,

$$c_{ij}=\begin{cases}1 \quad when, s_i \in t_j^{*}\\ -1 \quad when, s_i \in {}^{*}t_j \\ 0 \quad when, s_i \notin {}^{*}t_j \cup t_j^{*}\end{cases} \tag{4}$$

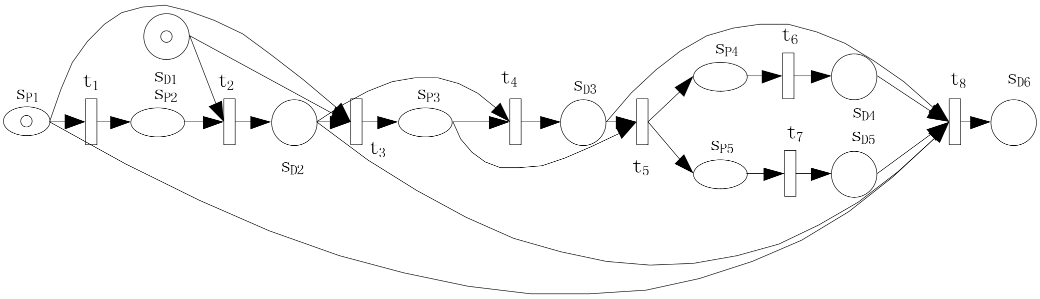


Fig. 2. A simplified PN-PDDP model for a design process.

The incidence matrix C of the aforementioned PN-PDDP model is shown as,

$$C = \begin{array}{c|cccccccc} & t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 \\ \hline S_{P1} & -1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \\ S_{P2} & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ S_{P3} & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\ S_{P4} & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ S_{P5} & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ S_{D1} & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ S_{D2} & 0 & 1 & -1 & -1 & 0 & 0 & 0 & -1 \\ S_{D3} & 0 & 0 & 0 & 1 & -1 & 0 & 0 & -1 \\ S_{D4} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ S_{D5} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ S_{D6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$

The incidence matrix describes the structure of the PN-PDDP, and obtains the characteristics of the design process.

4.2 *Activity sequence*

The state equation of the design process is as follows:

$$M_{Z+1} = M_Z + C'U_Z \tag{5}$$

Where, M_{Z+1} and M_Z are the two state labels in the design process, C' is the transformation matrix of the incidence matrix, U_z is the control vector and subscript z represents the Zth design activity stage. Due to the fact that it does not consider the resource consumption in the PN-PDDP, so all of the consumption elements (the elements with a minus sign) in the incidence matrix C can be replaced by 0, and a new transformation incidence matrix C' are obtained. The control vector U_z corresponds to the activity transition set T, and it is the process sequence of the Zth design activity stage.

The transformation incidence matrix C' of the above PN-PDDP in Fig.2 is in the following.

$$C' = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Define $\sigma = M_0 t_1 M_1 t_2 \cdots t_n M_n$ as the activity sequence of the PN-PDDP, when it meets the condition of,

$$\forall i, 1 \le i \le n \Rightarrow M_{i-1}\left[t_i\right\rangle M_i$$

Where, $M_0, M_1 \cdots M_n$ representing varied intermediate stages (0,1,2,…,n). $M_{i-1}\left[t_i\right\rangle M_i$ indicates that when the library is in the state of M_{i-1}, activity t_i fires such that the library enters a new state M_i.

From Fig.4, the whole design process from its beginning to its end goes through the activity sequence of $\sigma = M_0 t_1 M_1 t_2 M_3 t_3 M_4 t_4 M_5 t_5 M_6 t_6 M_7 t_7 M_8 t_8 M_9$, where M_0 and M_9 signal the initial and the final state, respectively and $M_0 = \{1,0,0,0,0,1,0,0,0,0,0\}$, $M_9 = \{0,0,0,0,0,0,0,0,0,0,1\}$.

The activity sequence σ can be simplified as $\sigma = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$ if the activity sequence M is ignored.

Because each activity occurs just once, the control vector U_Z is the identity matrix I. As a result, the state equation of the entire design process can be captured as follows:

$$M_9 = M_0 + C'I \tag{6}$$

5. PN-PDDP CHARACTERISTICS

Regarding the multi-component collaborative design, the design process model manifests all characteristics of the design process, which are closely related to the successful execution and execution efficiency of the design process. The present study defines the major characteristics of PN-PDDP (concurrent design activities, design reachability, design cycle, design conflicts and so on), and solves the design process characteristics through the PN-PDDP characteristics parameters illustrated below.

5.1 *Concurrent activity sequence*

If two activities sequences σ_1 and σ_2 coincide with each other (signified as $\sigma_1 co \sigma_2$), then the following relationship exists:

$$\sigma_1 co \sigma_2 :\Leftrightarrow \sigma_1 \cap \sigma_2 = \Phi \wedge \neg(\sigma_1 < \sigma_2) \wedge \neg(\sigma_2 < \sigma_1)$$

Where, < signals the interdependent relationship and the attendant sequential relationship between σ_1 and σ_2, that is, σ_1 occurs before σ_2. Fig. 3 shows the concurrent activity sequence diagram of the model described in Fig. 2.

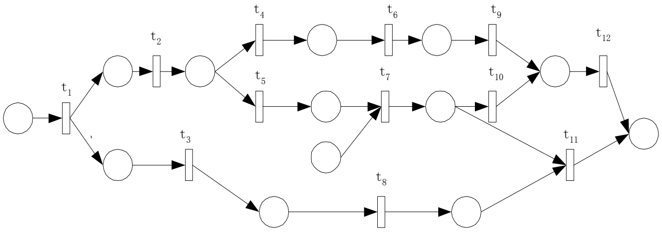


Fig. 3. Concurrent activity sequence diagram.

As shown in the above collaborative design PN-PDDP model, the activity sequences $\sigma_1 = \{t_4, t_6, t_9, t_{12}\}$ and

$\sigma_2 = \{t_5, t_7, t_{10}, t_{12}\}$ meet the concurrent conditions such that they are called concurrent activity sequences.

In this study, the notion of concurrent rate is introduced to indicate the degree of concurrent activities in a design activity sequence, and is defined as the rate of the number of concurrent activities to the total number of activities in a design activity sequence. If R_c signals concurrent rate, the following formula obtains,

$$R_c(\sigma) = \frac{N_p}{N_\sigma} \tag{7}$$

Where, N_P is the number of concurrent activities and N_σ is the total number of design activities in the design activity sequence σ .

5.2 Design reachability

The design reachability in the PN-PDDP is meant to indicate that the set $[M_0 \ \rangle$ is the minimal set meeting the following conditions:

(1) $M_0 \in [M_0 \ \rangle$;

(2) If $M' \in [M_0 \ \rangle, t \in T$, then $M'[t \ \rangle M$, resulting in $M \in M_0$.

The design reachability is used to verify whether the system could enter a certain state. In the PN-PDDP, reachability analysis is conducted to determine whether design performance and design data could be gained. Reachability analysis can be achieved through reachability trees, whose nodes indicate varied reachable states, and whose arcs represent events. The methods to create reachability trees can be referred to Reference(Chongyi, 2005). Fig. 4 shows the reachability tree of the PN-PDDP in Fig. 2.

As can be seen in Fig. 4, the name of the library is used to indicate that it contains ordinary *Tokens*, and the name of the library together with the sign of ′ indicates that it contains trace *Tokens*. The reachability tree captures all the possible states represented by all the possible activities in the design process. Each activity and its node set contain all the possible design data which are gradually defined and added during the design process. If an activity in the design process is not included in the activity set, this activity is unlikely to occur; and if a data library is not included in the node set, the data is not liable to arise from the design process.

5.3 Design cycle

If the first and the last activities are the same in the activity sequence σ of the PN-PDDP, then this sequence is referred the design cycle. Fig.5 shows one example of a design cycle, in which the first and last activities of the activity sequence $\sigma_1 = (t_1, t_2, t_3, t_6, t_1)$ are the same. The design process is often too iterative. It is important to reduce number of interactions of the design activity cycle by identifying all design activities in the design cycle using the PN-PDDP and subsequently providing all possible design information to ensure high quality and first time right decision making.

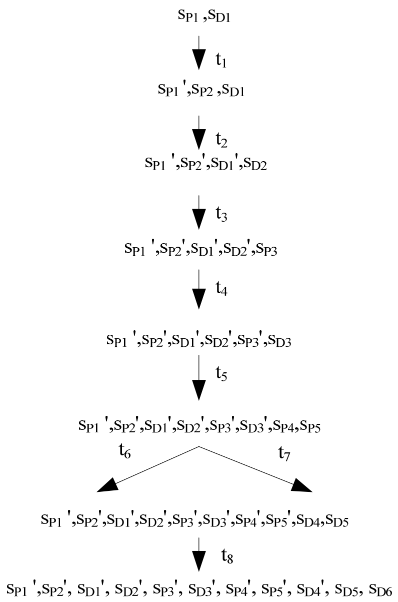

SP1 ,SD1
t1
SP1 ',SP2 ,SD1
t2
SP1 ',SP2',SD1',SD2
t3
SP1 ',SP2',SD1',SD2',SP3
t4
SP1 ',SP2',SD1',SD2',SP3',SD3
t5
SP1 ',SP2',SD1',SD2',SP3',SD3',SP4,SP5
t6
t7
SP1 ',SP2',SD1',SD2',SP3',SD3',SP4',SP5',SD4,SD5
t8
SP1 ',SP2', SD1', SD2', SP3', SD3', SP4', SP5', SD4', SD5, SD6

Fig. 4. Reachability tree.

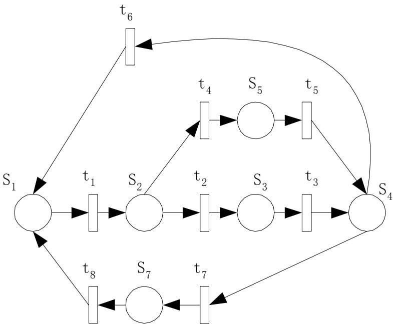


Fig. 5. Diagram for the design cycle.

5.4 Design conflicts

If two design activities t_i and t_j in the PN-PDDP have the relationship of $t_i^\bullet \cap t_j^\bullet \neq \varphi$, the two activities are said to have conflicts. That is, if two design activities in the design process attempt to modify either structure or performance characteristics data simultaneously, they are considered to have conflicts. For example, two design activities t_1 and t_2 modify the design data s_1 simultaneously, respectively such that they have access conflicts.

In multi-components collaborative design, conflicts are classified into two types according to which library is involved or where conflicts exist: namely performance conflicts and structure conflicts.

Performance conflicts occur in the performance decomposition process. If two preceding performance decomposition activities demand the same subsequent performance activity to ensure their respective performance, conflicts are said to have occurred in this subsequent performance activity. For instance, in the design of the complex multi-component coupling high-speed turbopump rotor system, maintaining the high-speed pumping performance of the pump system demands the rotor to achieve a certain rotation performance, and meanwhile, maintaining the dynamics performance of the rotor system also calls for the rotor to achieve a certain rotation performance. Consequently, the two aforementioned performance decomposition activities induce conflicts in the rotation performance activity. When performance conflicts occur, priority should be given to the issue of whether intersection exists between these performance characteristics. If it does, the intersection should be assigned as the value of performance characteristics so as to dissolve the conflicts. By contrast, if it does not, performance decomposition activities should be redone so as to dissolve conflicts either by maintaining the same scheme but generating an intersection between the performance characteristics or by choosing a different scheme so as to preempt the conflicts of the two preceding performance decomposition activities in the subsequence performance activity.

Structure conflicts occur in the detailed design process. If the same structure characteristic is required to meet two performance characteristics, conflicts can occur. For instance, in the design of the multi-component coupling rotor system, achieving the rotor's transmission performance demands action on the shaft's structure characteristics; meanwhile, achieving the shaft's static and dynamic performance characteristics also demands action on the shaft's structure characteristics. As a result, these three design activities (shaft's structure design activity, rotor's static design activity, and rotor's dynamic design activity) will have conflicts in the shaft's structure performance activity. The proposed structure conflicts resolution follows the following steps:

First, examine whether an intersection solution exists within the possible solution spaces of the conflicted structure requirements. If it does exist, the intersection should be assigned as the value of the structure characteristics. If it does not exist, the detailed design activity should be repeated with a view to create intersection between the solution spaces of the conflicted structure characteristics. If intersection still could not be obtained, the design performance requirements should be related and then decomposed again so as to redefine the performance characteristics of the design activity, which will allow design activities to proceed. In this way, structure conflicts are resolved.

From the above analysis of the PN-PDDP, finding conflicts in the design process and then resolving them could considerably enhance the design efficiency.

6. COMPUTER SIMULATION OF THE DESIGN PROCESS AND SOLVING METHODS FOR CHARACTERISTICS PARAMETERS

6.1 Solving method of concurrent activity sequence analysis

The following solving method could be worked out in terms of the definition of concurrent activity sequence. If library *s* is the before set of activity t_i and the after set of activity t_j, t_j is considered as the depending activity of activity t_i. All depending activities of t_i constitute its depending set.

(1) Search the activities which share the same depending set, and make them the concurrent activity set.

(2) Create the complete depending set of concurrent activities. If t_j is the depending activity of concurrent activities, t_j together with all its depending activities should be included in the complete set. That is, all the depending activities of concurrent activities coupled with their own depending activities should be included in the final complete depending set.

(3) Search the activities which take the concurrent activity set and the complete depending set as their depending set, and accordingly creates their own concurrent activity set.

(4) Once all such activities have been included into concurrent activity sets, the activities of each concurrent activity set constitute an activity chain, called a concurrent activity chain.

6.2 Solving method of design cycle analysis

According to the Petri nets theory (Lohmann et al., 2009), the design cycle can be gained by solving the nonnegative constant $T\_$ of the PN-PDDP, which could be calculated by the following equation:

$$C \cdot J = \theta_s \tag{8}$$

Where, J is the constant $T\_$ of the PN-PDDP, C is the incidence matrix of the PN-PDDP, θ_s is the vector $S\_$ in which all elements have the value of 0.

6.3 Solving method of design conflicts analysis

The following procedure could be followed to search possible conflicts and relevant performance characteristics in terms of the above definition of design conflicts. The total number of all the libraries is defined as *m*.

(1) Create the incidence matrix C.

(2) Set $i = 1$.

(3) Find the row in which library s_i in the incidence matrix C stays. Then find the columns in which elements in that row are of -1. Next find the activities that correspond to those columns, and those activities (the total number is defined as k) are the design activity set where conflicts are likely to occur in the library s_i, marked as T_i, $T_i = \{t_{i1}, t_{i2} \cdots t_{ik}\}$.

(4) Set $j = 1$.

(5) Find the column where t_{ij} lies in. Then find the performance libraries where corresponding elements in that column are of 1. Next, put those performance libraries into the conflict performance characteristics set P_i.

(6) If $j < k$, then jump to step (4).

(7) If $i < m$, then jump to step (2). Otherwise, the whole procedure is over.

7. APPLICATION AND IMPLEMENTATION FOR THE PN-PDDP MODEL

7.1 Concurrent activity sequence

The above solving method will be exemplified with the design of the multi-component coupling high-speed turbopump rotor system. Fig.1 shows the PN-PDDP of the multi-component coupling high-speed turbopump rotor system. Based on Eq. (4), its incidence matrix is shown as following,

$$C^* = \begin{array}{c|ccccccccccc}
 & t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 & t_9 & t_{10} & t_{11} \\
\hline
S_{P1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{P2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{P3} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
S_{P4} & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{P5} & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{P6} & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{P7} & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
S_{P8} & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
S_{P9} & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
S_{P10} & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
S_{P11} & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
S_{P12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
S_{D1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{D2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
S_{D3} & -1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
S_{D4} & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\
S_{D5} & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
S_{D6} & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 1 & 0 \\
S_{D7} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 \\
S_{D8} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \\
S_{D9} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

(1) The depending sets of varied activities can be obtained through the relevant matrix, as shown in the following.

$D_{t1} = \{\varphi\}$; $D_{t2} = \{t_1, t_{11}\}$; $D_{t3} = \{t_2\}$; $D_{t4} = \{t_2\}$; $D_{t5} = \{t_3, t_1\}$; $D_{t6} = \{t_3, t_1\}$; $D_{t7} = \{t_3, t_5\}$; $D_{t8} = \{t_4, t_6\}$; $D_{t9} = \{t_3, t_4, t_7, t_8\}$; $D_{t10} = \{t_3, t_4, t_7, t_8\}$; $D_{t11} = \{t_9, t_{10}\}$;

(2) Find the activities with the same depending set, which in turn form the concurrent activity sets. As shown in the above depending set, D_{t3} and D_{t4} have the same depending set such that they constitute a pair of concurrent activity sets C_{11} and C_{12}, which all depend on the complete depending set of events DA_1. That is, $C_{11} = \{t_3\}$, $C_{12} = \{t_4\}$, $DA_1 = \{t_1, t_2\}$.

(3) Include the activities, which take the concurrent activity set or complete depending set as their depending set, into the concurrent activity set in sequence.

$C_{11} = \{t_3, t_5\}$; $C_{12} = \{t_4, t_6\}$; $C_{11} = \{t_3, t_5, t_7\}$; $C_{12} = \{t_4, t_6, t_8\}$;

Finally, two activity chains, also two concurrent activity sequences, obtain, namely, $\sigma_{11} = \{t_3, t_5, t_7\}$ and $\sigma_{12} = \{t_4, t_6, t_8\}$. As previously shown, in the design of the multi-component coupling rotor system, the activity sequence consisting of the radial support components design t_3, the radial components—rotor coupling structure design t_5, and the radial components—rotor bearing capacity design t_7, and the activity sequence comprised of the axial bearing components performance design t_4, the axial components—rotor coupling structure design t_6, and the axial components—rotor bearing capacity design t_8, can be executed concurrently.

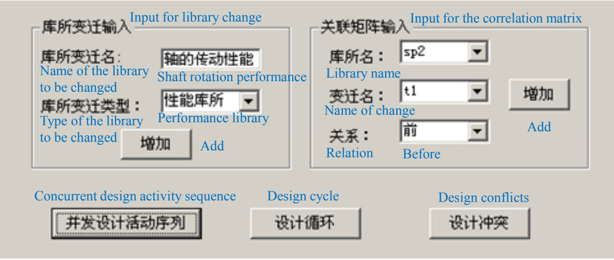
7.2 Design cycle

The analysis of the design cycle shown in Fig.8, results in three $T\_$ constants of the PN-PDDP through Eq. (8): $J_1 = \{1,1,1,0,0,1,0,0\}$, $J_2 = \{1,1,1,0,0,0,1,1\}$, $J_3 = \{1,0,0,1,1,1,0,0\}$. Accordingly, three design cycles obtain: $\sigma_1 = \{t_1, t_2, t_3, t_6, t_1\}$, $\sigma_2 = \{t_1, t_2, t_3, t_7, t_8, t_1\}$, $\sigma_3 = \{t_1, t_4, t_5, t_6, t_1\}$.

7.3 Design conflicts

In the PN-PDDP of Fig.1, S_{D3} is the after set of the two design activities t_1 and t_{11}, which suggest that these two activities have conflicts at S_{D3}. This results in the conflict activity set $T = \{t_1, t_{11}\}$ and the conflict performance set $P = \{S_{p1}, S_{p2}, S_{p12}\}$. That is to say, in the design of the multi-component coupling rotor system, both the rotor structure design t_1 and the turbopump rotor system dynamics design t_{11} would operate on the shaft structure, leading to conflicts. The factors influencing the performance characteristics of the conflict include the rotor's positioning performance characteristics S_{p1}, the rotor's rotation performance characteristics S_{p2}, and the rotor system dynamics performance characteristics S_{p12}. Therefore, conflicts can be resolved through coordinating the demands of these three kinds of performance characteristics.

The solving program of the PN-PDDP characteristic parameters is developed by using VC++ interface, which achieves computer simulation of the design process model. The program's operation interface is shown in Fig. 6.


库所变迁输入 Input for library change
关联矩阵输入 Input for the correlation matrix
增加 Add
增加 Add
并发设计活动序列 Concurrent design activity sequence
设计循环 Design cycle
设计冲突 Design conflicts

库所变迁名: 轴的传动性能 (Name of the library to be changed: Shaft rotation performance)

库所变迁类型: 性能库所 (Type of the library to be changed: Performance library)

库所名: sp2 (Library name)

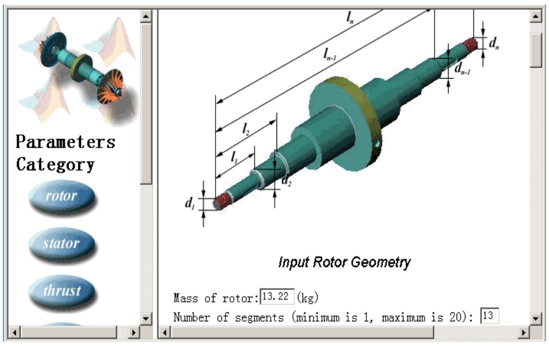
变迁名: t1 (Name of change)

关系: 前 (Relation: Before)

Fig. 6. Interface of solving program for the PN-PDDP characteristic parameters.

A case study of collaborative design has been conducted on the aforementioned multi-component coupling rotor system. Different distributed design parties (alliances) could negotiate with the client about the multi-component design scheme via the real-time interactive system, and further decompose the design task into sub-tasks so as to be assigned to different entities. The component structure design entity uses the AUTOCAD secondary development program to conduct structure design, producing STL files, which are then sent to the processing technology analysis and design entity via e-mail. These two entities conduct relevant analysis, utilizing their respective resources, and then send the results back to the overall structure design entity via e-mail. The overall coupling structure design entity will save the drawings and parameters of different components in the product database when it completes its design. Using the data management services, the monitoring and control system design entity reads the structure data of varied components so as to monitor and control system design, and saves the results in the product design database. The dynamics analysis entity uses these data to conduct dynamics checking, sending the results back to the design alliance.

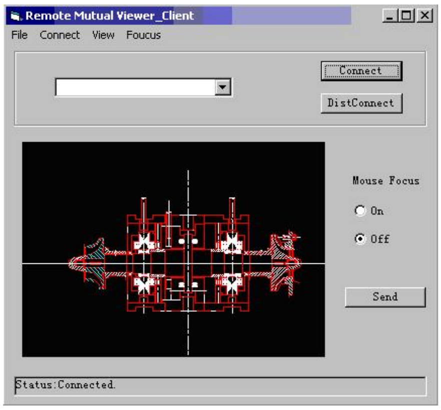
In this paper, a case design study was carried out. Fig. 7 shows the special design drawing of the rotor shaft. Fig. 8 demonstrates the interactive interface of the drawing of the multi-component coupling rotor collaborative design process.



Mass of rotor: 13.22 (kg)

Number of segments (minimum is 1, maximum is 20): 13

Fig. 7. Rotor structure drawing.



○ On

● Off

Status:Connected.

Fig. 8. Interactive interface of the multi-component coupling rotor drawing.

8. CONCLUSION

This paper presents a new design methodology and PN-PDDP model. The study examined the characteristics of the PN-PDDP model coupling design process based on extended Petri nets, including concurrent activity sequence, reachability tree, design cycle, and design conflicts. Based on the definition of the design process spanning reachability tree, concurrent activity sequence, design cycle, and design conflicts, and determines their parameters. The solving method of those characteristic parameters is presented and successfully demonstrates the effectiveness of the model.

The paper also presents the case study design process of a specific multi-component coupling high-speed turbopump. The relevant knowledge matrix is obtained and analysis software of coupling design is demonstrated to be able to handle the design activity support and conflict resolution. The design methodology and process will provide an important reference for the similar engineering design.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China under Grant [No.51575418] and CSC (China Scholarship Council) scholarship.

REFERENCES

Araz, M. & Erden, Z. (2014) Behavioural representation and simulation of design concepts for systematic conceptual design of mechatronic systems using Petri Nets. *International Journal of Production Research*, 52(2), 563-583.

Başak, Ö. & Albayrak, Y. E. (2015) Petri net based decision system modeling in real-time scheduling and control of flexible automotive manufacturing systems. *Computers & Industrial Engineering*, 86, 116-126.

Cheng, F., Li, H., Wang, Y.-W., Skitmore, M. & Forsythe, P. (2013) Modeling resource management in the building design process by information constraint Petri nets. *Automation in Construction*, 29, 92-99.

Chongyi, Y. (2005) Theory and application of Petri net. *Beijing: Publishing House of Electronics Industry*, 1-8.

Erden, Z. (2011) State-based conceptual design in mechatronics via petri nets: A case study for an educational robot. *Journal of Control Engineering and Applied Informatics*, 13(2), 70-75.

Fang, H., Han, J. & Liu, X. (2011) Modeling Method of Automotive Body CAN/LIN Nets Application Protocol Based on Object-oriented Colored Petri Net. *Chinese Journal of Mechanical Engineering*, 24(6), 999-1006.

Hsieh, F.-S. (2008) Robustness analysis of holonic assembly/disassembly processes with Petri nets. *Automatica*, 44(10), 2538-2548.

Huang, H.-Z., Xu, H.-W. & Zu, X. (2010) Petri Net-based Coordination Component for Collaborative Design. *Concurrent Engineering*, 18(3), 199-205.

Issaoui, L., Aifaoui, N. & Benamara, A. (2015) Solution space reduction of disassembly sequences generated automatically via computer aids. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 229(16), 2977-2986.

Karniel, A. & Reich, Y. (2011) Formalizing a workflow-net implementation of design-structure-matrix-based process planning for new product development. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(3), 476-491.

Karniel, A. & Reich, Y. (2013) Multi-level modelling and simulation of new product development processes. *Journal of Engineering Design*, 24(3), 185-210.

Lohmann, N., Verbeek, E. & Dijkman, R. (2009) Petri net transformations for business processes–a survey, *Transactions on petri nets and other models of concurrency II*Springer, 46-63.

Luo, Y., Peng, Q. & Gu, P. (2016) Integrated multi-layer representation and ant colony search for product selective disassembly planning. *Computers in Industry*, 75, 13-26.

Popescu, C., Soto, M. C. & Lastra, J. L. M. (2012) A Petri net-based approach to incremental modelling of flow and resources in service-oriented manufacturing systems. *International Journal of Production Research*, 50(2), 325-343.

Salinas, F., González, M. A., Escalante, M. F. & de León Morales, J. (2016) Control design strategy for flying capacitor multilevel converters based on Petri nets. *IEEE Transactions on Industrial Electronics*, 63(3), 1728-1736.

Wang, T., Guo, S., Sarker, B. R. & Li, Y. (2012) Process planning for collaborative product development with CD-DSM in optoelectronic enterprises. *Advanced Engineering Informatics*, 26(2), 280-291.

Xu, Q. & Jiao, J. R. (2009) Modeling the Design Process of Product Variants With Timed Colored Petri Nets. *Journal of Mechanical Design*, 131(6), 061009.

Zhang, G., Ji, F., Zhao, W. & Li, T. (2016) Study on the Collaborative Design PN-PDDP Model for the Multi-component Coupling Rotor System Based on Petri Nets. *Procedia CIRP*, 56, 67-72.