Process Modeling of Top-down Collaborative Assembly Design Based on Petri Net

Youdong Yang, Shuting Zhang, Xiang Cheng
State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027, China
{yyoudong, zhangshuting, chengxing}@cad.zju.edu.cn

Abstract

The top-down collaborative assembly design is a high parallel, dynamic and stochastic process. A concise design process model should be explored for effective process control and the system design of the top-down collaborative assembly design. In this paper the activity characteristics of top-down collaborative assembly design are analyzed, the typical sub-process models of collaborative design activities are constructed using Hierarchical Object-oriented Petri Net (HOOPN). The activity process of the top-down collaborative assembly design is modeled and the system architecture framework is built based on the activity process model.

1. Introduction

Complex product design is usually a top-down, high parallel, dynamic and stochastic process which is widely agreed that the design process includes requirements analysis, conceptual design and detailed design. During top-down collaborative assembly design process, the distributed designer can't ascertain the effects of his/her activities to other designers and needs to obtain feedback from others to adjust the design activity according to the dependences and constraints among the design tasks. The information of the whole design process is incomplete and uncertain, and can't be entirely opened to all the designers. So the main collaborative activities of the designers are information share and cooperation. The design process is not uniform during which synchronous and asynchronous interactions and cooperation are both needed for the users to achieve effective collaboration and information sharing.

Many information and distributed computing systems have been built to support the collaborative design process. According to the research survey of Lu and Cai[1], researches in the area of general design process modeling can be differentiated from three aspects of mechanical design: decision making[2], activity manipulation[3] and data support. The research of the design process modeling is one of the key issues of collaboration. PN has many advantages in describing key behaviors of concurrency, conflict and synchronization. Hierarchical Object-oriented Petri-net (HOOPN) is a high level Petri-net which can resolve the disadvantages of PN [4]. The advantage of HOOPN is that the complex process could be decomposed by hierarchical subdividing technology, and the attributes of design activity process state also could be described by object-oriented tools (section 2 in our paper).

In this paper HOOPN is used to construct the design activity process model of top-down collaborative assembly design. This approach can dispose the dynamic and stochastic design process with constraints and conflicts. The activity process model of top-down collaborative assembly design is analyzed using the analysis methods of PN. The main activities of the system are addressed, and the architecture framework of top-down collaborative assembly design system is developed based on its process model.

2. Hierarchical object-oriented Petri-net (HOOPN)

Petri net is noted as formalisms for modeling, simulating and analyzing dynamic systems with concurrent and non-deterministic behavior. An object-oriented Petri-net (OOPN) which is showed in figure 1 combines objects and Petri-net in the form of tokens [5]. This means that a token is also an object that has attributes and methods. A process begins when a sequence of methods in an object is executed by transition firing. This condition will be evaluated by a method invocation which is stimulated only if the attribute of an object is equal to the variable specified in the input arc. Places serve as containers for tokens which carry information of a system state.

978 -1-4244-1579-3/07/$25.00 © 2007 IEEE

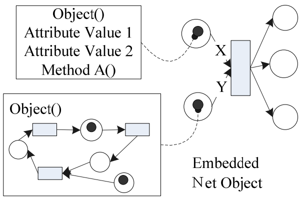

Figure 1. An object-oriented Petri Net

Hierarchical Petri-net (HPN) is commonly used in process modeling of complex dynamic system [6]. In order to implement the substitution of transition sub-net and the canonical structure of sub-net, the sub-net is added into two places, IN and OUT. IN represents the input of the sub-net and OUT represents the output of the sub-net, which is showed Figure 2.

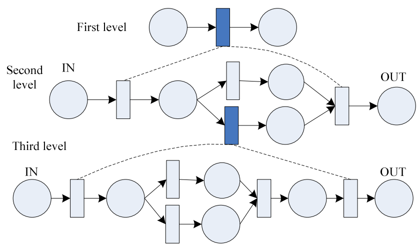

Figure 2. An example of three-level HPN

We bring these two domains together in the new concept of Hierarchical Object-oriented Petri Net (HOOPN). HOOPN is a high-level Petri net that supports object-oriented concepts whose transitions are substituted by OOPN. HOOPN can depict more concrete and detail activities. The idea of hierarchy is similar to that of modularizing design. When modeling a system with HOOPN, we need not consider all the complex and extensive information about the system, because the modeling steps are based on refinement and incremental development. This offers remarkable benefit when modeling complex processes with dynamic and arbitrary abstraction levels such as top-down collaborative assembly design.

3. Process modeling of top-down collaborative assembly design

During the top-down collaborative assembly the design process, designers need to share design information one another and the conflicts and interferences of them should be resolved through real-time design cooperation. We know that the activity of the designer is very complex, stochastic and dynamic through the case study in top-down collaborative assembly design [7]. So HOOPN adopted in the paper can reduce the model complexity and clearly express the stochastic and dynamic activity.

3.1. General activity sub-process model for collaboration

The foundations of high effective top-down collaborative assembly design are efficient communication and cooperation. Well disposed cooperation mechanism is one of the key issues of top-down collaborative assembly design. The general activities of top-down collaborative assembly design are message receiving, sending and disposing. In order to reduce the complexity of the system model and implement modularization modeling, the basic activity processes are modeled as sub-process model.

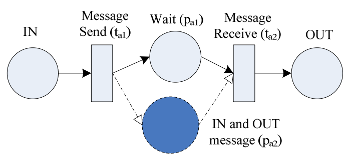

Figure 3. Message receiving and sending process model

The message receiving and sending process indicates that the message sent by someone needs to be disposed by the receiver, and the disposed result would be returned. Figure 3 shows that after the message is sent, the sender should wait for the returned message which is disposed by the receiver. So the transition t_{a1} should generate two tokens, one lets the sender in waiting state (in p_{a1}), and the other is an object which includes sending message (in p_{a2}) and fires the relative information disposing transition.

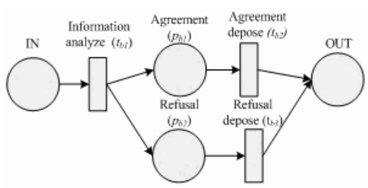

Figure 4. Application disposal process mode

During top-down collaborative assembly design process, the main activities are application of sharing information and cooperation. While the object attributes of application are different, the method of disposal is similar, which is served as a sub-process model as showed in figure 4. This model depicts the analysis transition t_{b1} fired by the object token of the receiving message， and the message of analyzed result returned to the sender. The token with agreement message is output by firing the transition t_{b2} if the result of analysis is agreement. Otherwise, the refusing message is output by the transition t_{b3}. While the received message is the refusal, the designer may bring forward an appeal.

3.2. The activity process model for the top-down collaborative assembly design

The design process model of top-down collaborative assembly design is represented in HOOPN as shown in figure 5.

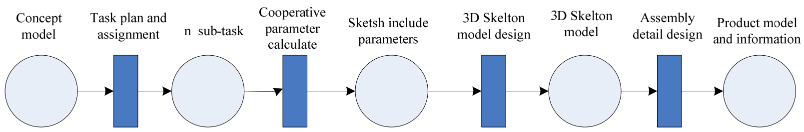

Figure 5. Design process model of top-down collaborative assembly design system

During the process of top-down collaborative assembly design, the parameter calculation, 3D skeleton model construction and the assembly detail design are achieved collaboratively by the designers. There are corresponding design (calculation or modeling) tools for each design process. But the participators have similar activity types. The design process is the random switch among the three activities.

The participators of the top-down collaborative assembly design are the designers and the manager. The manager needs to supervise the whole design process and mediate the appeal of the designers. The manager has higher weight than the general designers and is responsible for setting weight and relationships of the designers. The appeal activities of the designers are generated due to the refusing message received. The reasons of the appeal are brought forward by the designer and mediated by the manager through cooperation. There are no essential differences of the activity process between the manager and designer if the manager serves as a designer. So we don't distinguish the designer and the manager during research of the system activity process.

The system activity process model for the top-down collaborative assembly design is showed in figure 6. The chart in the dashed frame represents the exterior disposing process of the IN and OUT information, which denotes that the designer needs to interact and share information with the exterior. This figure only describes the design process of one designer because other designers have the same design activity processes. The activity process model of the system is composed of two main transitions t_{s1} and t_{s2}, namely the collaborative design process transition t_{s1} of designer and the message disposing activity process transition t_{s2} of the exterior, which have sub-process models. The top-down collaborative assembly design is a refinement and iterative process. The refinement result of the design is examined, and the design issues are found and modified during the design process. Finally, the design task result is output until the design requirements are satisfied.

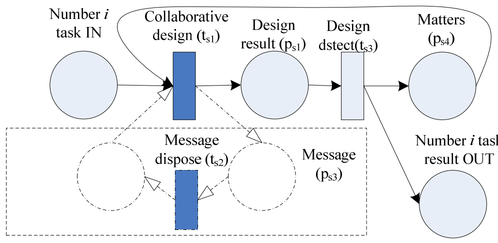

Figure 6. System activity process model

According to the actual design state, the designer needs to share information and cooperate with other designers during the independent design process. The sharing information denotes the sending and receiving information or the application of relative design information. The activity process of the designer is the stochastic switch process of the three upwards activities. The collaborative design process model of the designer is showed in figure 7 which is the sub-process model of figure 6 (t_{s1}).

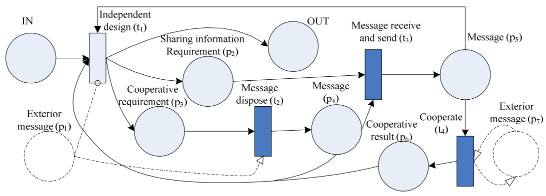

Figure 7. Collaborative design process model

The reasons of cooperative requirements mainly include the active cooperation and the passive cooperation due to the interferences, the conflicts and the cooperative invite of the exterior. The active cooperation is mainly due to the design matters and designer appeal. The interference and conflicts are due to the design modeling, the modification of the assembly interface and the key parameters. While the interference or conflict is detected, it should be eliminated and the cooperative requirement should be auto-generated by the disposing mechanism. The generated token is loaded into the place (p_3) which fires the information disposal transition t_2. Then the designer decides whether or not to cooperate with the relative designers, and the result token is loaded into the place (p_4). Furthermore, figure 7 shows that the transition t_1 may be interrupted by the exterior message token (in place p_1 depicted by constraint arc which has a small circle at one end), and the t_2 will be fired.

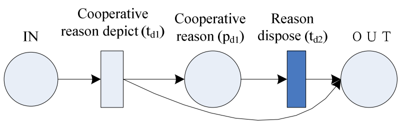

Figure 8. Message disposal model (t_2)

The cooperative requirement disposing transition t_2 shown in figure 8 is the sub-process model of t_{s1}. There are three types of object token attributes. The first is the cooperative requirement application actively initiated by the designer which will fire the transition (t_{d1}). The second is the cooperative requirement derived from interferences and conflicts which fires the transition t_{d2}. If the attribute of object token is refusal, t_{d2} is fired, namely the design activity returns to independent design; otherwise the object token is generated like the active cooperation. The third is the token comes from p_1, which fires the transition t_2. The output message of t_2 is the agreement information or the refusing reason. If the cooperative invite is refused, two tokens are output to p_4, one token fires the transition t_3 and the other fires t_1. The token of p_2 which is disposed by the exterior is sent and received by t_3. The result is output to the place p_5. The attribute types of token (in p_5) are sharing message, refusing message and cooperating space message and so on. According to the different attributes of token, t_1 or t_4 would be fired. During the process of cooperation, the designer continually interacts with the other designers. Finally, the token of cooperative result fires t_1, which means the design process returns to the independent design activity.

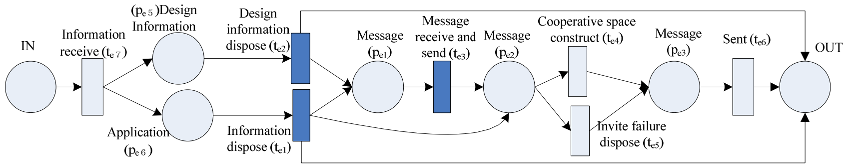

Figure 9. Message disposing activity process model of the exterior

The message disposing activity process model of the exterior is showed in figure 9, which is the sub-process model of figure 6 (t_{s2}). There are four types of token attribute of the exterior, which is showed as the following:

The first type of token attribute is the requirement of sharing information which causes the transition t_{e1} to be fired. The following token of t_{e1} has the message to be output directly to OUT place including sharing information or refusing reason.

The second type of token attribute is the cooperative application which causes the transition t_{e1} to be fired. If the following token of t_{e1} has the message of cooperative invite (in p_{e1}), the transition t_{e3} is fired and all collaborative design activity process model of inviting designer receive a token. The tokens which have the disposing results of the cooperative invite are returned to the exterior (p_{e2}). The transition t_{e5} would be fired if only one inviting failure token was output to p_{e2}. If all messages of the returned token are agreement, the cooperative application is success, and the transition t_{e4} is fired.

The third type of token attribute is the agreement or refusal message of cooperative invite which causes the transition t_{e1} to be fired. The following token of t_{e1} is output to p_{e2}. The following activity process is same as the second type.

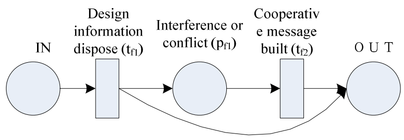

Figure 10. Disposal transition (t_{e2})

The fourth type of token attribute is the design information which is uploaded by the designer which causes the t_{e2} to be fired. The information of the design model needs not to be uploaded in real time due to the characteristics of top-down collaborative assembly design. The whole assembly model of the designer doesn't reflect the actual model of each designer. There are no interferences and conflicts during independent design, but some can be detected by the exterior after the design information is uploaded. The sub-process model of transition t_{e2} is showed in figure 10. If the message of the follower token of t_{e2} is interference or conflict in place p_{e1}, the following activity process is the same as the disposal of the cooperative invite.

4. Analysis of the activity process model

The structure of the upwards sub-process model can be simplified as it is composed of the sequential structures, concurrent structures and condition selection structures. The reachability of all sub-process models can be proved by the rules of PN simplification [8]. The whole system process model is composed of the collaborative design process model of all designer and the message disposal of the exterior activity process model. According to upwards analysis, we only need to analyze the validity between the single activity process model of the designer and the exterior message disposal process model.

4.1. Process model analysis

Through the analysis of activity process model of the designer, the design activities directly relate to the exterior except the independent design activity. The main activities of a designer are the independent design, sharing information disposal and cooperation. The cooperative activities and the sharing information disposal directly generate results which affect the other designer activities while the independent design doesn't. So we only need to analyze the validity of the system process model at the state of cooperative activity and the sharing information disposal.

In figure 7, the transition t_2 is fired by two tokens: token of p_3 is the cooperative application; token of p_1 is the cooperative invite from the exterior. The marking M_x (p_1, p_2, p_3, p_4, p_5) may have five states, $M_{x1}(1,0,0,0,0)$, $M_{x2}(0,1,0,0,0)$, $M_{x3}(0,0,1,0,0,0,0)$, $M_{x4}(1,0,1,0,0)$, $M_{x5}(1,1,0,0,0)$. At the states of M_{x1}, M_{x2}, M_{x3}, the system process model is reachable, and has *S*-invariant because each state only has one token. At the state of $M_{x4}(1,0,1,0,0)$, two tokens of p_1 and p_3 fire the transition t_1 simultaneously, which leads to conflict. At the state of $M_{x5}(1,1,0,0,0)$, two transitions t_2 and t_3 are fired concurrently and the follower mark of M_{x5} is $(0,0,0,1,1)$. The concurrency of t_2 and t_3 doesn't affect the validity of the system due to the asynchronism of concurrency. The transition t_3 sends the information to the exterior. Due to the parallel of the designer activities, the message disposing activity process model of the exterior (figure 8) may receive several message disposing requirements simultaneously, namely the transition t_{e7} may be fired by several tokens. But t_{e7} only can dispose one token of them, and then conflicts happen.

Furthermore, during the activity process of the designer and the exterior, while the transition t_3 and t_{e7} are fired (the sub-process model of t_3 and t_{e7} depicted in figure 3), the number of the received tokens has stated requirement. The follower transition could not be fired until all disposing messages of the receivers are returned; otherwise, the deadlock arises and the liveness of the system loses.

4.2. Process model improvement

The three upwards states affect the liveness of the system, generate conflicts and deadlock. In order to eliminate the conflicts and deadlock, the activity process model should be improved.

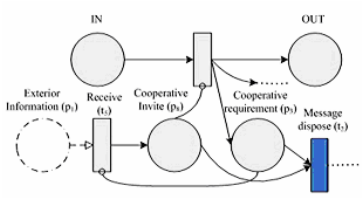

Figure 11. Improvement of designer activity process model

To find a solution to the first problem, we improve process model of the designer which is showed in figure 11. Because the designer can't simultaneously dispose two cooperative messages, the cooperative invite can not be received while the place p_3 has token. In figure 11, the transition t_5 is added, the constraint arc restrains the active of t_1 when the place p_8 has token, and the exterior message is disposed. While the place p_3 has token, the transition t_5 is restrained also by the constraint arc, the exterior message can't be received, and the conflict is diminished.

We can build the sub-process model of the transition t_{e7}, which can resolve the second question. FIFO Queue can be used in the sub-process model [8].

The solution to the third question is showed in figure 12. If the wait place p_{a1} has token, the time transition is fired. The transition t_{a2} is restrained by the constraint arc when the time terminate place p_{a3} has token, and the transition t_{a3} is fired and generates the ending message token. The t_{a4} will be restrained while the place OUT has token. The deadlock of the message receiving and sending process mode is resolved.

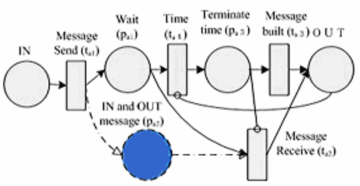

Figure 12. Improvement of the message receiving and sending process model

The conflicts and deadlock is diminished, and the validity of system model is guaranteed through the upwards improvements.

5. The architecture framework of top-down collaborative assembly design system

According to the analysis of the activity process model, the system of top-down collaborative assembly design has two type cooperators, namely manager and designer. We employ a three-tier architecture that partitions the overall system into the server-side, client-side, and database, showed in figure 13. The client-side is composed of two type clients (manager and designer). The supervision and management of the whole project is charged by manager, and the product design is achieved by the manager and the designers collaboratively.

The prototype of the top-down collaborative assembly design system has been developed and the relative paper is presented in the paper [9].

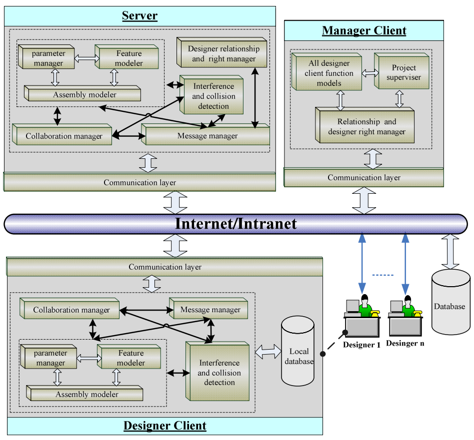

Figure 13. Architecture framework of top-down collaborative assembly design

6. Conclusion and future work

This paper presents the activity process model of the top-down collaborative assembly design using HOOPN to clearly understand the top-down collaborative assembly complex design process, and the design activity model is explored to ascertain it's fluent. It can explicitly and realistically represents concurrent operations, synchronous activities, resource sharing and so on. It overcomes the difficulties for the requirements analysis and design evaluation.

The process of top-down collaborative assembly design is an iterative refinement and creative process. It is one of the key issues of the Computer Supported Cooperative Work in Design (CSCWD). There are many issues need to be further studied, such as the security of design information, the representation of assembly design, the management and schedule of collaborative design activities and so on.

References

[1] S. CY Lu, J. Cai, "A Collaborative Design Process Model in the Socio-technical Engineering Design Framework", *Source Artificial Intelligence for Engineering Design, Analysis and Manufacturing Archive*, 2001, 15 (1), pp.3-20.
[2] D. Steward, "System analysis and Management Structure, Strategy and Design", *Petrocelli Books*, NY, 1991.
[3] A. Kusiak, and J.R Wang, "Decomposition of the Design Process", *Journal of Mechanical Design*, 1993, 115(44), pp. 687-696.
[4] JE Hong, DH Bae, "Software Modeling and Analysis Using a Hierarchical Object-oriented Petri Net", *Information* Sciences, 2000, 130(2), pp.133-164
[5] R Valk., "Petri-net as Token Objects: An Introduction to Elementary Object Nets", *In: Proc of the 19th Int'l Conf on Application and Theory of Petri-net,* Berlin: Springer-Verlag, 1998, pp.1- 25.
[6] W M Zuberek, I. Bluemke, "Hierarchies of Place/transition Refinements in Petri-net", *The 5th IEEE Int'l Conf on Emerging Technologies and Factory Automation (ETFA'96)*, Kauai, 1996
[7] Y D. Yang, Z H. Li, et al., "A Case Study in Top-down Collaborative Assembly Design", *International Technology &Innovation Conference 2006 (ITIC2006)，* Hangzhou (China), 2006, pp.179-186.
[8] C Y. Yuan, "Petri-net Theory and Application", *publishing house of electronics Industry*, Beijing, 2005.
[9] S T. Zhang, X. Chen, et al., "A Framework for Collaborative Top-down Assembly Design", *Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Las Vegas, Nevada, USA, 2007 (accept).