# A methodology for task decomposition and conflict resolution to support product design

SHAO Weiping[1,a], FAN Gangqiang[2,b],HAO Yongping[3,c]

[1,2,3]R&D Center of CAD/CAM Technology, Shenyang Ligong University, Shenyang, China;

[a, b]shaoweiping3008@sohu.com, [c] yphsit@126.com

**Keywords:** task planning algorithm; coupling task set; conflict resolution; task decomposing

**Abstract.** In view of the complex product collaborative design task planning, an algorithm of task planning is put forward based on Design Structure Matrix (DSM) and the detail operation steps of algorithm is given. A coupling task set is obtained using reachable matrix to participate in calculation. Then use tearing algorithm to decouple or reorganization for coupling tasks. At the same time, a decision framework is introduced to reveal the design conflict problems in the design process planning and its mathematical model for describing conflict problems is also established. It provided the theoretical basis for better seeking the optimal solution.

## Introduction

In order to organize design and manufacture activities in collaborative design process availably, mass information, data and resources which related to product development process must be effectively managed. So it's very important to layout design tasks and task decomposition in the initial design stage.

Aiming at task decomposition for large scale system, Sobieski[1] has put forward a method which separate large scale system into linear structure sub-problem. Product development time can be shortening by parallel processing sub-task. But it ignore the relativities among these sub-systems. Kuisiak[2] has proposed that using matrix show the relations between design tasks and design variables. But coordination problem between sub-tasks in design process has not solved. Xie Y.B.et.[3] have optimized and configured the dispersive design resources by using existing computer technologies and network platforms. But they didn't think about the problem of task partition.

In this paper, we have discussed a design task planning algorithm based on Design Structure Matrix (DSM) and a design task conflict resolution method using mathematics model is brought forward. When tasks are planed, first define the tasks according to product database and general task planning activities. Then decompose the tasks and seek the solution of the question.

## Task planning algorithm in the cooperative design

The aim of task planning in the product cooperative design is making out design and development strategy, determining the input information of product design, concluding the set of feasible solution which satisfy the product performance and structure design via the designers collaborative work and finding out the optimal solution in the set of feasible solution using scheme evaluation system and criterion.

Reiteration in product development process is one of the important reasons which effect product quality and design time. So reducing the iteration is a very important aim in studying product task planning problem. An appropriate and effective process planning method is very necessary to reduce the task iteration and improve the efficiency of product task planning [4]. In this paper, design task planning algorithm based on DSM is given as fellow:

*Step 1*: Decompose process into some design tasks according to analysis the specific characteristic and functional requirements of product design process. Then analyze and evaluate the data requirements and dependency relationship among tasks and finally build DSM matrix.

*Step 2*: Solve the reachability matrix P by using DSM matrix and recognize the independent task and the coupling task set in DSM matrix.

$$X = \begin{bmatrix} X_{11} & X_{12} & \Lambda & X_{1n} \\ X_{21} & X_{22} & \Lambda & X_{2n} \\ \Lambda & \Lambda & \Lambda & \Lambda \\ X_{n1} & X_{n2} & \Lambda & X_{nn} \end{bmatrix} \tag{1}$$

The reachability matrix P is shown as fellow:

$$P = X^{(1)} \vee X^{(2)} \vee ... \vee X^{(n)} \tag{2}$$

Where, $X^{(i)} = X \wedge \Lambda_i \wedge X$ Suppose, there exist two Boolean DSM matrixes: $A = (a_{ij})_{n\times n}$ , $B = (b_{ij})_{n\times n}$ , then,

$$A \wedge B = Q = (q_{ij})_{n\times n}\,,\ A \vee B = U = (u_{ij})_{n\times n}\,,\ u_{ij} = \bigvee_{k=1}^{n} (a_{ij} \wedge b_{ij})\,,\ u_{ij} = a_{ij} \vee b_{ij} \tag{3}$$

Calculate the coupling matrix of reachability matrix P as fellow:

$$P \cdot P^T = \begin{bmatrix} p_{11} & p_{12} & \Lambda & p_{1n} \\ p_{21} & p_{22} & \Lambda & p_{2n} \\ \Lambda & \Lambda & \Lambda & \Lambda \\ p_{n1} & p_{n2} & \Lambda & p_{nn} \end{bmatrix} \cdot \begin{bmatrix} p_{11} & p_{21} & \Lambda & p_{n1} \\ p_{12} & p_{22} & \Lambda & p_{n2} \\ \Lambda & \Lambda & \Lambda & \Lambda \\ p_{1n} & p_{2n} & \Lambda & p_{nn} \end{bmatrix} = \begin{bmatrix} p_{11}^2 & p_{12} \cdot p_{21} & \Lambda & p_{1n} \cdot p_{n1} \\ p_{21} \cdot p_{12} & p_{22}^2 & \Lambda & p_{2n} \cdot p_{n2} \\ \Lambda & \Lambda & \Lambda & \Lambda \\ p_{n1} \cdot p_{1n} & p_{n2} \cdot p_{n2} & \Lambda & p_{nn}^2 \end{bmatrix} \tag{4}$$

Step 3: Tasks are interdependence in coupling task set. It's very difficult to gain the execution logical relation of tasks only via information association relationship between tasks. So, decoupling operation for the coupling task set is necessary. Arrange the task order in DSM matrix element. Take each coupling task set as an independent aggregation (namely, a task block) and regroup the DSM matrix which contains these task blocks. Put the independent task blocks in the most front or end of the matrix. Then, a new DSM matrix without coupling task set is gained [5].

Step 4: Using tearing algorithm based on DSM matrix to decouple or regroup the coupling tasks in the coupling task set. And finally, reasonable process order can be gained via task planning.

The aim of tearing algorithm is to reduce the design process iteration and depress the degree of coupling between tasks. It must obey the follow regulation when tear the coupling task set.

① When tear the design process coupling task set, First tear the design task which input information amount and intensity is lowest. For, input information amount and intensity of task can indicate the degree of association between this task and the others. The less the input information of task is, the less influence the other tasks can be get.

② Ensure that the number of dissevered task nodes is as less as possible. In this way, the degree of design process information damage produced owing to task tear minimize to the lowest level. The flow of the task tearing algorithm based on DSM is shown as Fig. 1.

## Design task decomposing

The intention of task decomposing is to reduce the complexity of design process. Only subdivide the whole product design process to operable task, can the total design tasks be complete perfectly. It is possible to bring on unexpected conflict in the product development process and result in the other correlation tasks not work normally if task decomposing method is inappropriate. So task decomposing plays a vital role in the product collaborative design process.

**Design task decomposing theory.**When task is decomposed, we must think about the product's functions and assign reasonable subtasks according to coordination design rule between parts of product. The theory of task decomposing is described in detail in Fig.2. Express design tasks and its subtasks as hierarchical structure.

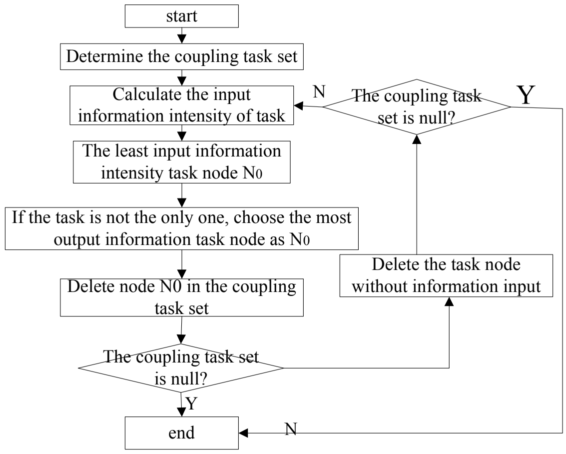


Fig.1. The flow of the task tearing algorithm based on DSM

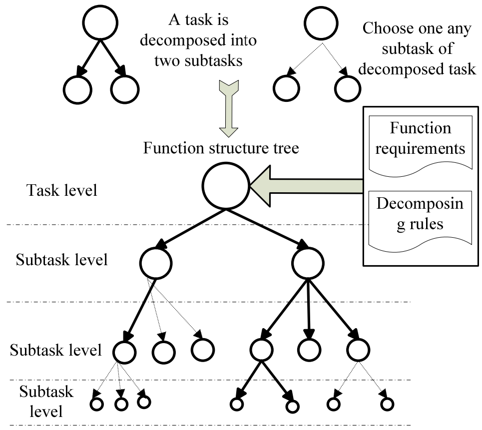


Fig.2. The theory of task decomposing

In order to complete product design successfully, decomposing regulation must be as constraint to proper decompose the task. The decomposed task set of collaborative design product can be described as follow: $TS=\{TS_i \mid i=1,2,\ldots,n,\ n\in N\}$, where n is the total design task count. And the decomposed subtask set can be described as: $STS=\{STS_{ij} \mid j=1,2,\ldots,m,\ m\in N\}$, where m is the total subtask set count of the $i^{th}$ task.

**Design task decomposing principle**.Design tasks are closely related to product feature, organizing structure and resource capacity. In addition, there exist a great many indeterminacy and complexity in design process itself. So it’s difficult to use the uniform mode and method for task decomposing. In this paper, we obey the following design task decomposing principles: ① task independence principle; ② task solvability principle; ③ granularity modest principle; ④ hierarchical decomposing principle; ⑤ time consistency principle. According to upwards design task decomposing principles, task decomposing flow is shown as in Fig.3.

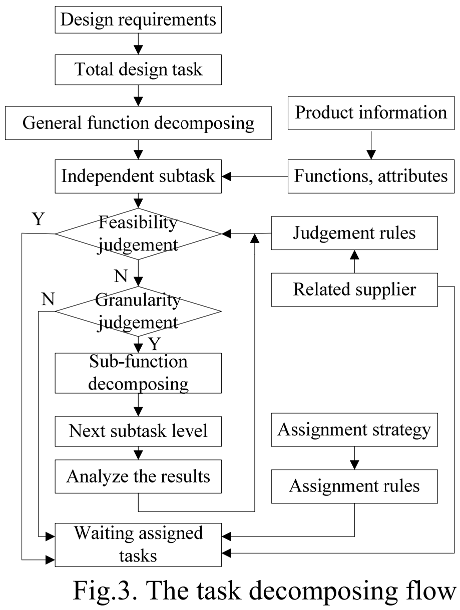


Fig.3. The task decomposing flow

## Conflict resolutions

Use math model to express the conflict problem in the design process. Design problem itself is an objective constraint solving problem which contains multi-design variables. Suppose, there exist n design tasks and then have n design variables in the design process, described as $Z_i=\{z_{i1},z_{i2},\ldots,z_{in}\}$. There are two distributed design tasks A and B. They are non-independent relationship and transfer information each other. Suppose task A has a variables and task B has b variables, then their conflict model can be expressed as follow:

$$\begin{bmatrix} \min f_A = f_A(Z_A, Z_{BA}) \\ st: G_A(Z_A, Z_{BA}) \le 0 \\ H_A(Z_A, Z_{BA}) = 0 \\ Z_{BA} \subseteq Z_B \end{bmatrix} \underset{Z_{BA}}{\overset{Z_{AB}}{\Leftrightarrow}} \begin{bmatrix} \min f_B = f_B(Z_B, Z_{AB}) \\ st: G_B(Z_B, Z_{AB}) \le 0 \\ H_B(Z_B, Z_{AB}) = 0 \\ Z_{AB} \subseteq Z_A \end{bmatrix}$$

Where, $f_i$--objective function of design task; $G$--inequality constraint set of design task; $H$--equality constraint set of design task; $Z_A$,$Z_B$ -- the design variable set of task $A$ and $B$.

If task A and B are non-independent relationship, their design variables influence each other. Suppose, $Z_{AB}$ is the variable set in task $A$ which influence the design results of task B, and $Z_{BA}$ is the variable set in task B which influence the design results of task $A$. $Z_\mu = Z_{AB} \bigcup Z_{BA}$. $Z_\mu \ne \phi$, then $Z_\mu$ is the coupling design variable set of task $A$ and $B$. The evaluation criterion $f_A = f_A(Z_A, Z_{BA})$ of task A is not in accordance with the evaluation criterion $f_B = f_B(Z_B, Z_{AB})$ of task $B$. If we want to get the optimum solutions in both two tasks, it's possible to come forth design constraint conflict and design objective conflict. The problem solving of task A and B can be expressed:

$$\textit{find } Z_A^*: \begin{Bmatrix} \min f_A = f_A(Z_A, Z_{BA}) \\ st: G_A(Z_A, Z_{BA}) \le 0 \\ H_A(Z_A, Z_{BA}) = 0 \\ Z_{BA} \subseteq Z_B \\ Z_A^* = Z_A \bigcup Z_{BA} \end{Bmatrix}, \text{ find } Z_B^*: \begin{Bmatrix} \min f_B = f_B(Z_B, Z_{AB}) \\ st: G_B(Z_B, Z_{AB}) \le 0 \\ H_B(Z_B, Z_{AB}) = 0 \\ Z_{AB} \subseteq Z_A \\ Z_B^* = Z_B \bigcup Z_{AB} \end{Bmatrix}$$

$Z_\lambda = Z_A^* \cap Z_B^*$

Where, $Z_A^*$,$Z_B^*$-- decoupled variable set of task A and B respectively;

$Z_\lambda$--the public design variable set of task A and B.

## Summary

In this paper, task planning model in the collaborative design process has been discussed. The key of process planning is task planning and its objective is in order to build the proper design process model. Design task planning algorithm based on DSM has been put forward and the detail operation steps of this algorithm have also been given. Regard each coupling task set as a task block and regroup the DSM matrix which contains task block. Adopt tearing algorithm for coupling task to decouple or regroup the tasks. The general rules of design task decomposing have been proposed according to the objective and theory of design task decomposing. Furthermore, design conflict problem in the design process planning has been studied in detail. The mathematics model of conflict problem has been established and a conflict resolution method based on integration theory has been introduced too.

## Acknowledgements

This work was financially supported by the National Natural Science Foundation of China with No. 50975183 in 2010-1012.

## References

[1] H. Verjus, F.A. Pourraz. Washington,D.C.,USA: IEEE Computer Society, (2007), p. 245.
[2] A.Kusiak, J.Wang.ASME Journal of Mechanic Design, Vol.115, No. 4, (1993).
[3] Y. B.Xie. Clem: submitted to China Mechanical Engineering, Vol.13, No. 4,(2002), p. 290.
[4] H. Q. Liu, G. N. Qi, submitted to Journal of Computer-Aided Design &Computer Graphic, Vol. 21, No. 11, (2009), p. 1638.
[5] W.D. Huang. Nanjing, China: PhD, (2009), p.55.