Task Decomposition Method of R&D Project Based on Product Structure Tree

Peng Jia
School of Mechanical Engineering, Shandong University, Jinan, China
Key Laboratory of High Efficiency and Clean Mechanical Manufacture (Shandong University), Ministry of Education, Jinan, China
Email: meiki100@163.com

Qi Gao, Xue Ji and Ting Xu
School of Mechanical Engineering, Shandong University, Jinan, China
Key Laboratory of High Efficiency and Clean Mechanical Manufacture (Shandong University), Ministry of Education, Jinan, China
Email: {gaoqi@sdu.edu.cn, jisnow1991@163.com, xutingup@126.com}

***Abstract*—To improve the accuracy and efficiency of task decomposition of R&D (Research and Development) projects, a new decomposition method based on product structure tree was proposed. The relevance between product structure tree and work breakdown structure (WBS) was analyzed through their natural quality and association relations. The mapping rules between product components and project tasks were defined. Mathematical models about the mapping from components and parts to tasks and combining tasks were established. Based on the above mechanism and models, the automatic mapping transformation from product components to project tasks was achieved. A prototype system was developed based on the task decomposition method, and the system had been applied in R&D projects of a diesel engine factory, which verified its effectiveness and feasibility.**

***Index Terms*—product structure tree, work breakdown structure, mapping transformation**

I. Introduction

The first step of scheduling after approving a project is decomposing project tasks. The purpose of decomposing tasks is to subdivide the whole project into smaller, more manageable sub-tasks or work-modules, and then the work breakdown structure was be built [1]. WBS plays an important role in project management, as it is the basis of developing project scheduling, allocating resources, estimating costs, and managing risks and so on. At present, the WBS is mainly decomposed by project managers manually according to their experiences. However, this kind of task decomposition method is inefficient, and easy to miss tasks. The method also cannot guarantee the accuracy and integrity of WBS.

The main objective of product R&D project tasks is to design and develop the components of a product. However, in the R&D process the product structure tree organizes the product, the components and parts, and the interdependent relations between them. Therefore, product structure tree is the important basis of decomposing R&D project tasks. Arnold [2] developed a generic 100% product-oriented WBS decomposition model. He Heng and Deng Jiati [3] subdivided the WBS of complex product R&D project into product WBS (PWBS), delivery WBS (DWBS) and activity WBS (AWBS). Radpul [4] studied specific work breakdown structure for specific project. Zhang Jie, Li Yuan and others [5] studied the task decomposition method based on process planning bill of material (PBOM) for aviation manufacturing projects, and proposed a WBS generation algorithm for aircraft assembly scheduling based on assembly order. He Miao and others [6] presented a view mapping method of product breakdown structure (PBS) and WBS for the aerospace product lifecycle, and completed the data transformation from PBS to WBS. At present, the scholars had made a beneficial study on the decomposition method of WBS, the relation between WBS and product structure tree and so on. Nevertheless, these studies are just in ideological aspect, lacking of technology implementation means, and the automatic decomposition of WBS is difficult to achieve.

Therefore, the objective of this paper is to develop a R&D project task decomposition method based on product structure tree. The paper first analyzes the corresponding relation between product structure tree and work breakdown structure, and provides the mapping rules that need to conform to in the mapping process from components to tasks. Then a mapping mathematical model is built to realize the mapping decomposition from components to tasks and the merging and reorganization of tasks automatically. Finally, the reasonable decomposition of WBS is accomplished.

Corresponding author: Qi Gao; Email: gaoqi@sdu.edu.cn.

doi:10.4304/jsw.9.7.1894-1902

TABLE I.
THE INFORMATION OF BOM AND WBS

	BOM	WBS
Natural Properties	the name of products and components, ID, size, material, weight, and origin	the name of tasks, ID, period, start/end time, resources, and responsible person
Relations	the hierarchical relation between the components (father-child relation)	the hierarchical relation between the tasks (father-child relation)
	the constraint relation between the components	predecessor/successor dependency between the tasks

II. Analysis of the Corresponding Relation between Product Structure Tree and Wbs

In the information management system, the bill of material (BOM) expresses the components and parts as a product structure tree, and the WBS expresses the task composition of project schedule. Both of them cover the nature properties of nodes and the relation between the nodes, as shown in Table I.

The nature properties of each component in BOM describe the basic information such as its name, ID, size, material, weight, origin and so on [7]. The nature properties of each task in WBS describe the basic information such as its name, ID, period, start/end time, resources, responsible person and so on. Therefore, the component nodes in BOM and the task nodes in WBS can be correspondingly mapped with the same properties such as name and ID.

The WBS of R&D project schedule is a result of hierarchical decomposition. There are hierarchical relations between the tasks (i.e. father-child relation), which fully reflect the guiding ideology of “top-down, from coarse to fine, stepwise refinement” [8]. The guiding ideology is reflected hierarchical relations between the components (i.e. father-child relation) in BOM. Therefore, the hierarchical relation of components decides the hierarchical relation of tasks, and the father-child relation of tasks can be deduced and solved according to the father-child relation of components.

The BOM also implies many constraint relations of components, such as assembly constraints, geometric constraints, logic constraints, etc. These constraints cause the information interaction between corresponding tasks and the interdependent relations between tasks. However, the interdependent relations don’t influence task decompositions. Therefore, the constraint relations can’t be taken into consideration in the process of WBS decomposition.

III. The Process of Task Decomposition based on BOM

After the properties of components obtained from BOM, the decomposition of R&D project can begin according to the corresponding relation between the BOM and WBS. The steps are as follows: Firstly, R&D project is initially decomposed according to information such as different sources of components, and then the initial WBS (IWBS) with the smallest size of task decomposition is obtained. Secondly, the IWBS is planned and tasks are combined based on design structure matrix (DSM), whose size of task decomposition is smaller, in order to obtain reasonable planned WBS (PWBS).

A. The IWBS Mapping Decomposition of R&D Project

According to the sources, the components are judged whether the mapping from components to tasks is needed. The task nodes identification (i.e. name and ID) are produced for the components that need to be mapped, and the father-child relation in the WBS are built according to the father-child relation of components in the BOM. This process is repeated until all the mapping decomposition of R&D project tasks is finished. The following rules should be obeyed in the process of mapping from BOM to IWBS.

Rule 1: the task decomposition rule based on the sources of components

From the aspect of enterprise design, the sources of components can be divided into self-made components, cooperation components and purchased components. The self-made components are designed and manufactured within the enterprise. Cooperation components are manufactured by other cooperation factories with the raw materials provided by the enterprise according to its requirements. Purchased components are purchased from external enterprises, and can be put into production directly.

According to the sources of components, the task decomposition rules in the product R&D project are as follows.

1) *Self-made components.* The components are judged whether to be newly made or borrowed from old ones according to their property of first used model. If the first used model belongs to this product, such components are newly made, and otherwise they are borrowed. The newly made components need to be mapped to the corresponding task nodes, while the borrowed ones needn’t.

2) *Cooperation components.* The cooperation components are subdivided into pure cooperation manufacturing components and cooperation design & manufacturing components according to whether their design are outsourced to other cooperation companies. The former ones need to be mapped to the corresponding task nodes, while the later ones needn’t.

3) *Purchased components.* Needn’t be mapped to the corresponding task nodes.

Rule 2: the task decomposition rule based on the design activities of components

In the process of product development, the design activities include computer-aided design (CAD), computer-aided engineering (CAE), computer-aided process planning (CAPP), and computer-aided fixture design (CAFD) [9-11]. As the importance and complexity of components are different, the needed design activities are not the same. For example, some components just need simple model design and process planning activities, but some key components also need the CAE analysis activities. In other words, one component may be involved in one or more design activities in the product development process, and accordingly there are one or more design tasks in R&D project.

Based on the design activities related to components, the task decomposition rules are described as follows: map the design tasks with the same number and corresponding content according to the required number and content of component design activities. For example, when one component needs CAD and CAE activities, two design tasks (the model design and CAE analysis) are gained through mapping.

Rule 3: the generation rule of task node identification

The name and ID of task nodes are generated according to the components' name and ID. The name of task is generated from the name of component using the equation "task name = component name + verb". The verb in this equation is verb phrase that can express the work content or work purpose of task such as model design, CAE analysis etc. The ID of task is generated through certain encoding rules like adding suffix.

Rule 4: the mapping rule of father-child relation

The father-child relation of tasks in project WBS is the same as that of the components in BOM. If the relation of two components is father-child in BOM, the relation of corresponding tasks will be father-child too.

B. Mapping from IWBS to PWBS

The task decomposed based on BOM will be the smallest, and the size of them is difficult to control. The smaller the size is, the easier the task planning, management and controlling will be. However, much too small decomposition size will result in invalid management, low efficiency in resource use and task implement. Therefore, it is necessary to merge and combine the smaller task in the IWBS to get PWBS with proper decomposition size.

There are many information interactions during R&D project, which is one of the projects' main characters. Thus, it is difficult to recognize the relation between tasks and to ensure how to combine the tasks with small size reasonably. DSM, as a powerful tool to plan project tasks [12, 13], is able to recognize the information interactions between serial, parallel and coupled tasks. By combining the tasks with smaller size and more information interactions, DSM not only can adjust the size of tasks in WBS, but also can reduce the information interactions among tasks. Finally, the PWBS with reasonable decomposition size can be obtained. Many scholars [14, 15] have done research about the task planning based on DSM, and come up with many effective solutions. In this paper, the concrete process of task planning based on DSM will not be discussed anymore.

IV. The Mapping Models of WBS Decomposition

A. Definition

In order to define the mapping function between BOM and WBS accurately, a mathematical description method of father-child relation of product components and that of tasks is given as follows.

Definition 1: product components set $C=\{a_1^C,......,a_n^C\}$; product R&D project tasks set $T=\{A_1^T,......,A_m^T\}$.

Components set C consists of the identification of all components in the father-child relation of BOM, not including the natural properties of components, such as size, material, weight and so on. Tasks set T consists of the identification of all tasks in WBS of product R&D project, either not including the natural properties of tasks, such as start time, finish time, responsible person and so on.

The identification of components is expressed by lowercase letters. The superscript expresses the range of component set, and the subscript is used to distinguish the identifications of different components, for example, a_n^C. The identification of tasks is expressed by capital letters that correspond with the lowercase letters of the component. The superscript expresses the range of task set, and the subscript is used to distinguish the identifications of different tasks, for example, A_n^T.

Definition 2: components relations dualistic group $S_{Cm/n}=\langle a_m^C,a_n^C\rangle, a_m^C\in C, a_n^C\in C$; tasks relations dualistic group $S_{Tr/t}=\langle A_r^T,A_t^T\rangle, A_r^T\in T, A_t^T\in T$.

Components relations dualistic group $S_{Cm/n}$ shows the direct father-child relation between component a_m^C and a_n^C. Tasks relations dualistic group $S_{Tr/t}$ shows the direct father-child relation between task A_r^T and A_t^T.

In addition, empty components relations/empty tasks relations dualistic group is defined as ϕ, $\langle a_m^C,a_m^C\rangle=\phi$, $\langle A_r^T,A_r^T\rangle=\phi$. It means that if the father components/tasks and children in the dualistic group are the same, components relations/tasks relations do not exist actually. In the power set of all father-child component relations/task relations dualistic group, two operations are defined as follows.

1) *Addition and subtraction operation.* When the elements of father-child components relations/tasks relations dualistic group are the same in the operation, the groups' addition result is itself, the subtraction operation gets empty relations. When the elements are different in the operation, the groups' addition operation is treated as the addition operation of the elements, and there is no subtraction operation. It can be expressed by the following equations.

① $\langle a_1^C,a_2^C\rangle+\langle a_1^C,a_2^C\rangle=\langle a_1^C,a_2^C\rangle$,

$\langle A_1^T,A_2^T\rangle+\langle A_1^T,A_2^T\rangle=\langle A_1^T,A_2^T\rangle$

② $\left\langle a_1^C,a_2^C\right\rangle+\left\langle a_3^C,a_4^C\right\rangle=\left\{\left\langle a_1^C,a_2^C\right\rangle,\left\langle a_3^C,a_4^C\right\rangle\right\}$,
$\left\langle A_1^T,A_2^T\right\rangle+\left\langle A_3^T,A_4^T\right\rangle=\left\{\left\langle A_1^T,A_2^T\right\rangle,\left\langle A_3^T,A_4^T\right\rangle\right\}$

③ $\left\langle a_1^C,a_2^C\right\rangle$-$\left\langle a_1^C,a_2^C\right\rangle=\phi$, $\left\langle A_1^T,A_2^T\right\rangle$-$\left\langle A_1^T,A_2^T\right\rangle=\phi$

④ $\left\langle a_1^C,a_2^C\right\rangle\pm\phi=\left\langle a_1^C,a_2^C\right\rangle$, $\left\langle A_1^T,A_2^T\right\rangle\pm\phi=\left\langle A_1^T,A_2^T\right\rangle$

2) *Substitution operation.*

① $\overline{a_0^C,a_1^C}\left\langle a_1^C,a_2^C\right\rangle=\left\langle a_0^C,a_2^C\right\rangle$, $\overline{A_0^T,A_1^T}\left\langle A_1^T,A_2^T\right\rangle=\left\langle A_0^T,A_2^T\right\rangle$

② $\overline{a_0^C,a_2^C}\left\langle a_1^C,a_2^C\right\rangle=\left\langle a_1^C,a_0^C\right\rangle$, $\overline{A_0^T,A_2^T}\left\langle A_1^T,A_2^T\right\rangle=\left\langle A_1^T,A_0^T\right\rangle$

In the above operations, the priority of substitution operation is higher than addition and subtraction operation.

Definition 3: If the father-component of a_m^C is a_s^C, and its sub-components are $a_{m.1}^C,......,a_{m.n}^C$, then all the component relations of a_m^C are defined as $R_C\left(a_m^C\right)=\left\{S_{Cs/m},S_{Cm/m.1},......,S_{Cm/m.n}\right\},a_s^C,a_m^C,a_{m.1}^C,......,a_{m.n}^C\in C$; if the father-task of $A_r^T, r\in N$ is A_k^T, and its sub-tasks are $A_{r.1}^T,......,A_{r.t}^T$, then all the tasks relations of A_r^T are defined as $R_T\left(A_r^T\right)=\left\{S_{Tk/r},S_{Tr/r.1},......,S_{Tr/r.t}\right\},A_k^T,A_r^T,A_{r.1}^T,......,A_{r.t}^T\in T$.

The components relations/ tasks relations of a component/task can represent all the father-child components relations/ tasks relations dualistic group of it. It not only indicates all the father components/ tasks of the component/task, but also indicates all the child components/ tasks. Two operators about the components relations of "father" are defined as follows.

1) Operator ∇ is applied to extract "its father relation". $\nabla R_C\left(a_m^C\right)$ represents the components relations dualistic group comprised by the component a_m^C and its father component. Similarly, $\nabla R_T\left(A_r^T\right)$ represents the tasks relations dualistic group comprised by the task A_r^T and its father task.

2) Operator $|\,|$ is applied to extract "father" identification. $\left|\nabla R_C\left(a_m^C\right)\right|$ represents the father-component of a_m^C,and $\left|\nabla R_T\left(A_r^T\right)\right|$ represents the father-task of A_r^T.

In addition, ∇^n is defined to express order n of ∇. $\left|\nabla^2 R_C\left(a_m^C\right)\right|=\left|\nabla R_C\left(\left|\nabla R_C\left(a_m^C\right)\right|\right)\right|$ represents the father-component's father-component of a_m^C, $\left|\nabla^2 R_T\left(A_r^T\right)\right|=\left|\nabla R_T\left(\left|\nabla R_T\left(A_r^T\right)\right|\right)\right|$ represents the father-task's father-task of A_r^T. ∇^0 represents itself. For example, $\left|\nabla^0 R_C\left(a_m^C\right)\right|=a_m^C$ represents component a_m^C, $\left|\nabla^0 R_T\left(A_r^T\right)\right|=A_r^T$ represents task A_r^T.

B. The Mapping Models

The mapping from components to tasks includes 1:1 and 1: n. It means that one component corresponds to one task or is decomposed into several tasks. Each mapping includes node identification mapping and decomposition relation mapping. The mathematical description is as follows.

1) *The 1:1 mapping from components to tasks*

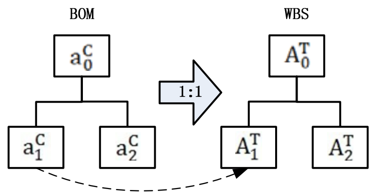

Figure 1. The example of 1:1 mapping.

Suppose component a_1^C maps to task A_1^T, as Fig. 1 shows. Equation (1) represents the correspondence of the identifications by defining the two identifications as equal. The decomposition of task A_1^T is equal to the decomposition of component a_1^C, as in (2).

$$A_1^T=a_1^C. \tag{1}$$

$$R_T\left(A_1^T\right)=R_C\left(a_1^C\right). \tag{2}$$

2) *The 1:n mapping from components to tasks*

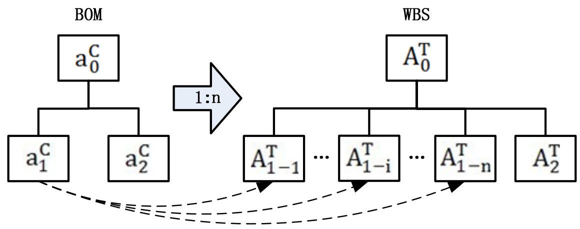

Figure 2. The example of 1:n mapping.

Suppose component a_1^C maps to tasks $A_{1-1}^T,...,A_{1-i}^T,...A_{1-n}^T$, which belong to the same father-task A_0^T. The subscript of task identification adds "-sequence code" to express component decomposition mapping to several tasks, as Fig. 2 shows. Equation (3) defines that the father-task of tasks $A_{1-1}^T,...,A_{1-i}^T,...A_{1-n}^T$ is the same one. In WBS, the decomposition relations of father-task A_0^T need to add some information of tasks relations. The father-child relation of task A_0^T in WBS can be deduced and solved automatically according to the father-child relation of component in BOM using (4).

$$\left|\nabla R_T\left(A_{1-1}^T\right)\right|=...=\left|\nabla R_T\left(A_{1-i}^T\right)\right|=...=\left|\nabla R_T\left(A_{1-n}^T\right)\right|. \tag{3}$$

$$R_T\left(\left|\nabla R_T\left(A_{1-i}^T\right)\right|\right)=\sum_{i=1}^{n}\overline{A_{1-i}^T,a_1^C}R_C\left(\left|\nabla R_C\left(a_1^C\right)\right|\right). \tag{4}$$

The Fig. 3 is the algorithm flow of mapping components to tasks.

The mapping between tasks include 1:1 and n:1 during the mapping planning process from IWBS to PWBS.

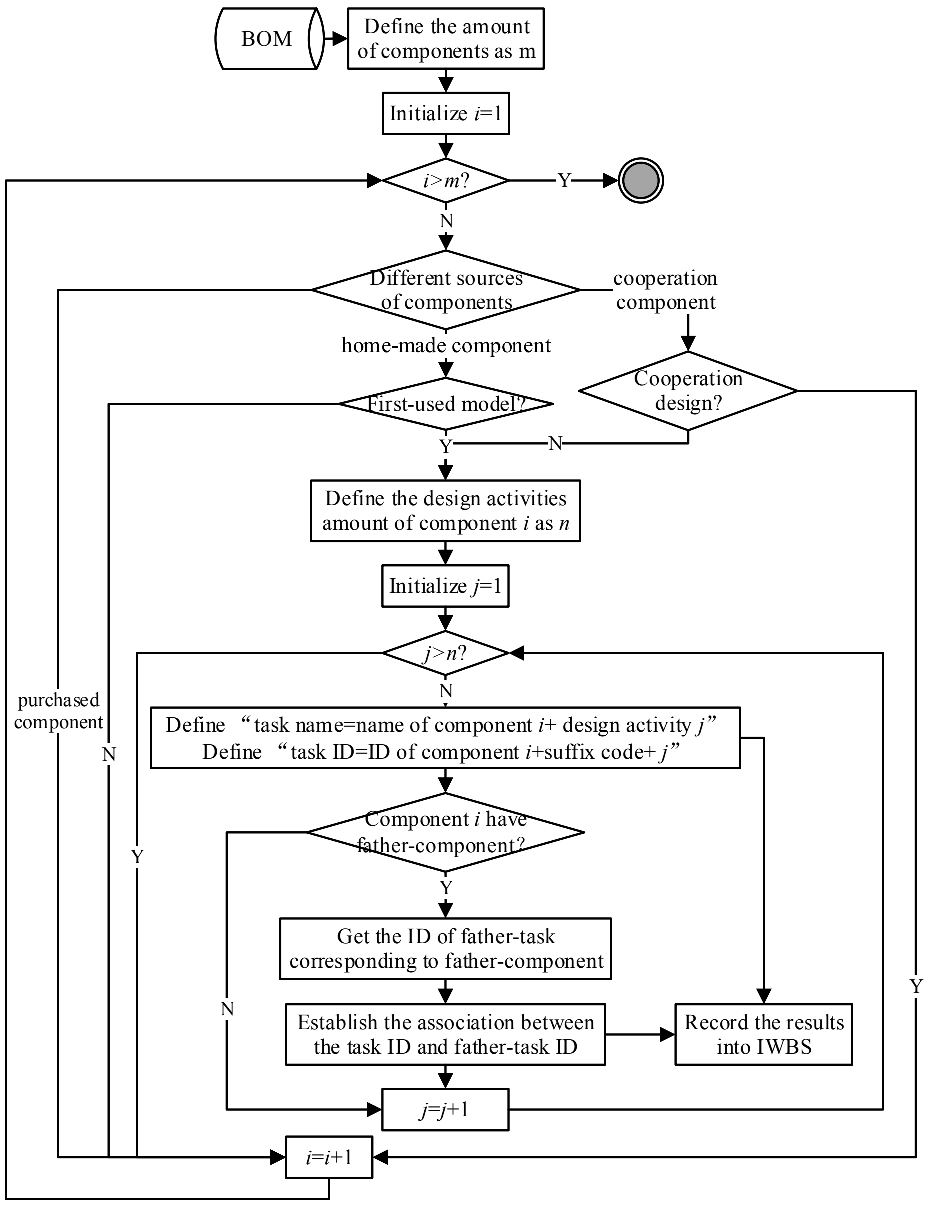

Figure 3. The algorithm flow of mapping components to tasks.

Tasks with 1:1 mapping don't change. Instead, the identification of tasks with n:1 mapping will be cumulatively recorded and combined. The father-child relation of combined task in WBS will be adjusted. The n:1 mapping can be classified into two cases as whether the father task of the tasks needing combined is the same. The mathematical description is as follows.

1) *The combination of sub-tasks that belong to the same father-task in IWBS*

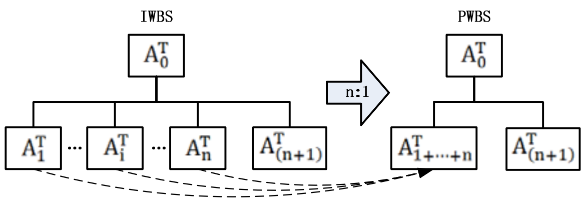

Figure 4. The example of n:1 mapping (belong to the same father-task).

Suppose sub-tasks $A_1^T,...,A_i^T,...,A_n^T$ that belong to the same father-task A_0^T are combined into task $A_{1+...+n}^T$. The subscript of combined task identification represent the cumulative record of subscripts of sub-tasks using the symbol "+", as Fig. 4 shows. Equation (5) defines that the father-task of tasks $A_1^T,...,A_i^T,...,A_n^T$ is the same one. The father-child relation of father-task A_0^T in PWBS can be deduced and solved automatically according to the father-child relation of sub-tasks in IWBS using (6).

$$\left|\nabla R_T\left(A_1^T\right)_{IWBS}\right| = ... = \left|\nabla R_T\left(A_i^T\right)_{IWBS}\right| = ... = \left|\nabla R_T\left(A_n^T\right)_{IWBS}\right| . \quad (5)$$

$$R_T\left(\left|\nabla R_T\left(A^T_{1+\ldots+n}\right)_{PWBS}\right|\right)_{PWBS} = R_T\left(\left|\nabla R_T\left(A^T_i\right)_{IWBS}\right|\right)_{IWBS} - \sum_{i=1}^{n}\nabla R_T\left(A^T_i\right)_{IWBS} + \sum_{i=1}^{n}\overline{A^T_{1+\ldots+n}, A^T_i}\nabla R_T\left(A^T_i\right)_{IWBS}. \quad (6)$$

2) *The combination of sub-tasks that belong to different father-tasks in IWBS*

Suppose sub-tasks $A^T_1,\ldots,A^T_j,\ldots,A^T_m$ that belong to the same father-task A^T_0 and sub-tasks $A^T_{i.1},\ldots,A^T_{i.k},\ldots,A^T_{i.s}$ that belong to the same father-task A^T_i are combined into task $A^T_{1+\ldots+m+i.1+\ldots+i.s}$. The subscript of combined task identification represent the cumulative record of subscripts of sub-tasks using the symbol "+", The position of combined task in WBS is the next hierarchy of the common father-task of the sub-tasks that need to be combined, such as the example of Fig. 5, in which the task $A^T_{1+\ldots+m+i.1+\ldots+i.s}$ is the child-task of task A^T_0. Therefore, the task decomposition relation of all father-tasks of the sub-tasks that need to be combined will be changed in WBS.

Equation (7) defines that the father-task of sub-tasks $A^T_1,\ldots,A^T_j,\ldots,A^T_m$ is the same one. Equation (8) defines that the father-task of sub-tasks $A^T_{i.1},\ldots,A^T_{i.k},\ldots,A^T_{i.s}$ is the same one. Equation (9) illustrates that the father task of sub-tasks $A^T_{i.1},\ldots,A^T_{i.k},\ldots,A^T_{i.s}$ and the father task of sub-tasks $A^T_{i.1},\ldots,A^T_{i.k},\ldots,A^T_{i.s}$ are different. Equation (10) shows that the m-order father-task of tasks $A^T_1,\ldots,A^T_j,\ldots,A^T_m$ and the n-order father-task of tasks $A^T_{i.1},\ldots,A^T_{i.k},\ldots,A^T_{i.s}$ is the same task. In the example of Fig. 5, m=1 and n=2. Using two steps of (11) and (12), the father-child relation of all the tasks will be deduced and solved automatically when the combination is finished.

$$\left|\nabla^m R_T\left(A^T_j\right)_{IWBS}\right| = \left|\nabla^n R_T\left(A^T_{i.k}\right)_{IWBS}\right|. \quad (10)$$

$$R_T\left(\left|\nabla R_T\left(A^T_{1+\ldots+m+i.1+\ldots+i.s}\right)_{PWBS}\right|\right)_{PWBS} = R_T\left(\left|\nabla^m R_T\left(A^T_j\right)_{IWBS}\right|\right)_{IWBS} + \overline{A^T_{1+\ldots+m+i.1+\ldots+i.s}, \left|\nabla^{m-1} R_T\left(A^T_j\right)_{IWBS}\right|}\nabla^m R_T\left(A^T_j\right)_{IWBS} = R_T\left(\left|\nabla^n R_T\left(A^T_{i.k}\right)_{IWBS}\right|\right)_{IWBS} + \overline{A^T_{1+\ldots+m+i.1+\ldots+i.s}, \left|\nabla^{n-1} R_T\left(A^T_{i.k}\right)_{IWBS}\right|}\nabla^n R_T\left(A^T_{i.k}\right)_{IWBS}. \quad (11)$$

$$R_T\left(\left|\nabla R_T\left(A^T_r\right)_{IWBS}\right|\right)_{PWBS} = R_T\left(\left|\nabla R_T\left(A^T_r\right)_{IWBS}\right|\right)_{IWBS} - \sum_r \nabla R_T\left(A^T_r\right)_{IWBS}, \quad (12)$$

A^T_r comes from tasks that need to be combined.

In the example of Fig. 5, (12) has two expressions as follows.

$$R_T\left(\left|\nabla R_T\left(A^T_j\right)_{IWBS}\right|\right)_{PWBS} = R_T\left(\left|\nabla R_T\left(A^T_j\right)_{IWBS}\right|\right)_{IWBS} - \sum_{j=1}^{m}\nabla R_T\left(A^T_j\right)_{IWBS}.$$

$$R_T\left(\left|\nabla R_T\left(A^T_{i.k}\right)_{IWBS}\right|\right)_{PWBS} = R_T\left(\left|\nabla R_T\left(A^T_{i.k}\right)_{IWBS}\right|\right)_{IWBS} - \sum_{k=1}^{s}\nabla R_T\left(A^T_{i.k}\right)_{IWBS}.$$

The Fig. 6 is the algorithm flow of n:1 mapping IWBS to PWBS.

V. System Implementations and Application Example

According to above researches, this paper expand and develop the WBS decomposition and task planning function of project management module on the Shanda Hoteamsoft product lifecycle management (PLM) system platform (as showed in Fig. 7), realizing the WBS auto-decomposition of R&D project based on BOM. Moreover, the system has been put into practice in a diesel engines factory successfully.

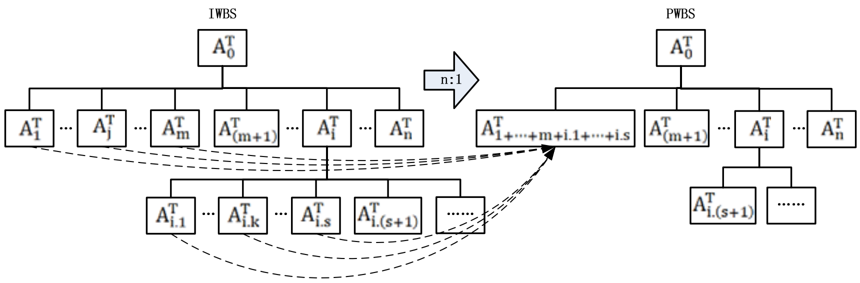

Figure 5. The example of n:1 mapping (belong to different father-tasks).

$$\left|\nabla R_T\left(A^T_1\right)_{IWBS}\right| = \ldots = \left|\nabla R_T\left(A^T_j\right)_{IWBS}\right| = \ldots = \left|\nabla R_T\left(A^T_m\right)_{IWBS}\right|. \quad (7)$$

$$\left|\nabla R_T\left(A^T_{i.1}\right)_{IWBS}\right| = \ldots = \left|\nabla R_T\left(A^T_{i.k}\right)_{IWBS}\right| = \ldots = \left|\nabla R_T\left(A^T_{i.s}\right)_{IWBS}\right|. \quad (8)$$

$$\left|\nabla R_T\left(A^T_j\right)_{IWBS}\right| \neq \left|\nabla R_T\left(A^T_{i.k}\right)_{IWBS}\right|. \quad (9)$$

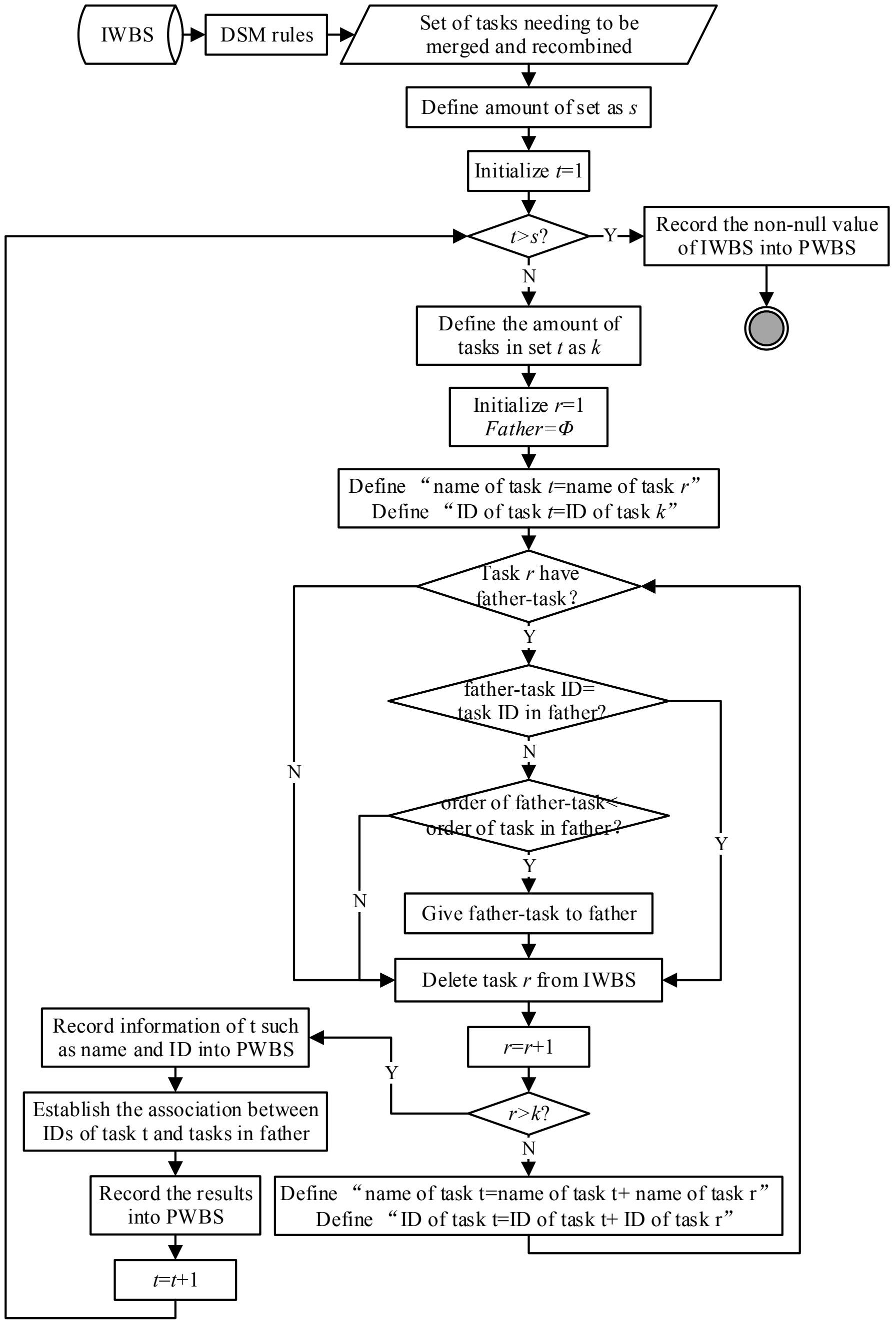

Figure 6. The algorithm flow of mapping from IWBS to PWBS.

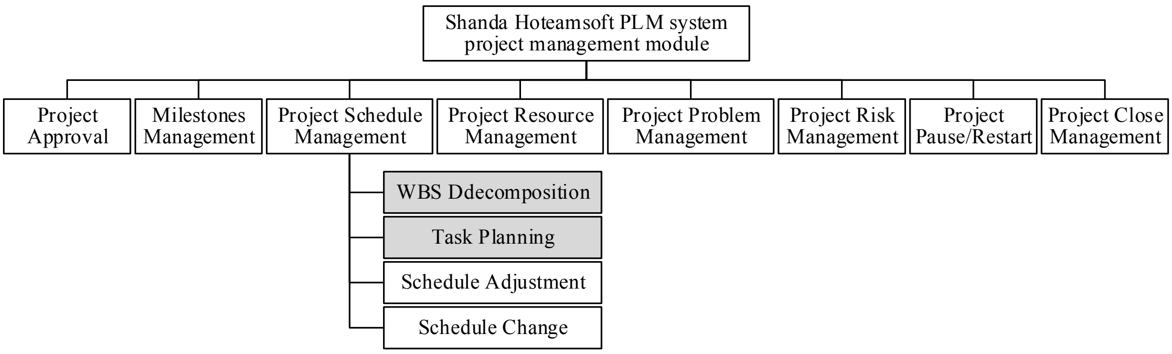

Figure 7. The function of project management module in Shanda Hoteamsoft PLM system.

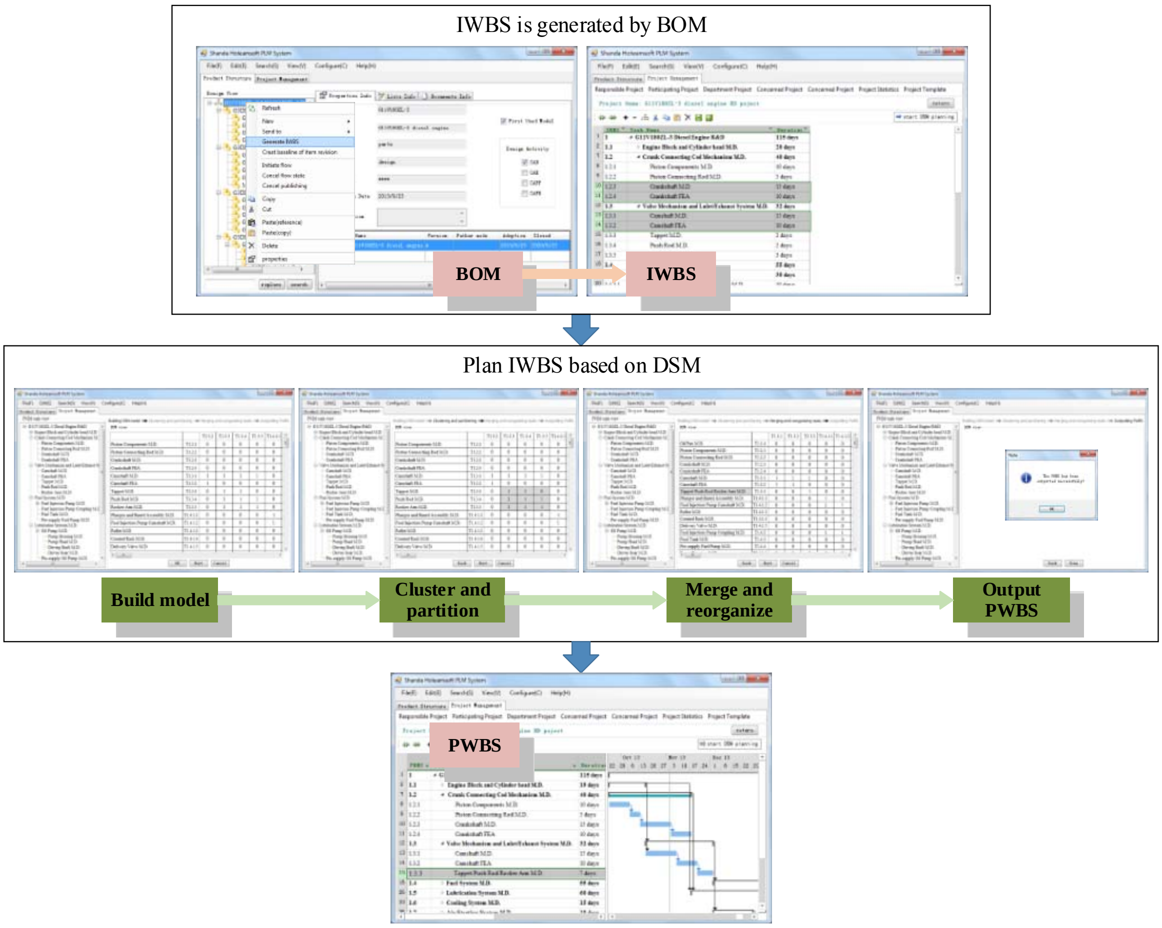

Figure 8. The task decomposition of G11V180ZL-3 diesel engine R&D project.

Fig. 8 shows the process in which a diesel engines factory decomposes the R&D project WBS of G11V180ZL-3 diesel engine with the Shanda Hoteamsoft PLM system, and the relative interface. Firstly, the system generates the corresponding IWBS of G11V180ZL-3 diesel engine R&D project by analyzing and calculating automatically based on BOM and its mapping rules to WBS. Then the DSM model of IWBS is established. The system determines the task sets needing to be merged and recombined by the clustering and dividing operation of DSM. Finally, PWBS is output, and the R&D project can be scheduled based on the PWBS.

VI. CONCLUSION

This paper puts forward a R&D decomposition method based on BOM, to help project manager decompose the project task rapidly and to improve the accuracy and integrity of R&D project WBS. This paper analyzes the relations between BOM and WBS according to their information composition. Based on these relations, this paper comes up with a series of mapping rules from components to tasks, including generative rules of task node identification, task decomposition rules based on the sources and design activities of components, mapping rules of father-child relations, etc. This paper realizes the mapping from components to tasks, from tasks to tasks, establishes the mathematical model of mapping node identification and that of mapping father-child relations , and constructs three kind of mapping transformation functions including 1:1,1:n, and n:1. Finally, a prototype system is established and put into practice in a diesel engines factory successfully. It proves that the WBS decomposition method can not only reduce the task decomposition workload of project R&D manager and improve the efficiency, but also ensure the accuracy and integrity of project WBS.

ACKNOWLEDGMENT

The research is supported by the National High Technology Research and Development Program of China (No. 2012AA040910) and the Natural Science Foundation of Shandong Province (No. ZR2012GM015).

REFERENCES

[1] Project Management Institute, Inc, "A guide to the project management body of knowledge (PMBOK guide) fourth edition", 2008.

[2] M. R. Arnold, "100% product-oriented work breakdown structures and their importance to system engineering," *IEEE Aerospace Conference Proceedings*, vol. 6, pp. 3984-3993, 2004.

[3] H. Heng, D. Jiati, "WBS supporting complex product development", *Aeronautical Manufacturing Technology*, pp. 74-78, 2009.

[4] Radpul, “Deliverable-oriented Work Breakdown Structure”, *AACE International Transactions of the Annual Meeting*, 1999.

[5] Z. Jie, L. Yuan, Y. Jianfeng, Z. Kaifu, “A new method for generating work breakdown structure on aircraft assembly scheduling”, *2008 International Conference on Wireless Communications*, 2008.

[6] H. Miao, Y. Haichen, J. Shikai, “Decomposition method of WBS based on PBS for complex products”, *China Mechanical Engineering*, vol. 22, pp. 1960-1964, 2011.

[7] L. Jiangang, “Research on the key technique of integrated management of product structure and development process in concurrent engineering”, *Nanjing University of Aeronautics and Astronautics*, *Nanjing,* 2006.

[8] L. Guangrui, “Research of methods and application on mutil-project management of network collaborative design with generalized resource constraint”, *Chongqing University, Chongqing,* 2011.

[9] W. Tunhua, S. Qinqin, Z. Changle, W. Ping, “Design and implementation of a general purpose 2D CAD system”, *Journal of Computers*, vol. 7, pp. 666-671, 2012.

[10] W. Jinfeng, D. Biqing, D. Haimin, “A modified genetic algorithm(GA) for optimization of process planning”, *Journal of Computers*, vol. 6, pp. 1430-1437, 2011.

[11] Z. Yi, M. Sihui, “Study of computer-aided design process and methods based on welding fixture”, *Journal of Computers*, vol. 7, pp. 1421-1428, 2012.

[12] M. Danilovic, T.R. Browning, Managing Complex Product Development Projects with Design Structure Matrices and Domain Mapping Matrices”, *International Journal of Project Management*, vol. 25, pp. 300-314, 2007.

[13] D.B. Tang, R.M. Zhu, J.C. Tang, et al. “Product Design Knowledge Management Based on Design Structure Matrix”, *Advanced Engineering Informatics*, vol. 24, pp. 159-166, 2010.

[14] D. B. Luh, Y. T. Ko, C. H. Ma, “A Dynamic Planning Approach for New Product Development”, *Concurrent Engineering: Research and Applications*, vol. 17, pp. 43-59, 2009.

[15] L. Y. Liang, “Grouping Decomposition under Constraints for Design/Build Life Cycle in Project Delivery System”, *International Journal Technology Management*, vol. 48, pp. 168-187, 2009.

Peng Jia was born in Jinan in 1986, Ph.D. candidate in School of Mechanical Engineering, Shandong University. She received her bachelor degree from Linyi University in 2008. Her current research interests include R&D project WBS decomposition and scheduling model.

Qi Gao was born in Dongying in 1970, professor and Ph.D. supervisor in School of Mechanical Engineering, Shandong University. She received her Ph.D. degree from Zhejiang University in 1998. Her current research interests include product lifecycle management, project management, intelligent design, and knowledge engineering.

Xue Ji was born in Jinan in 1986, master candidate in School of Mechanical Engineering, Shandong University. She received her bachelor degree from Shandong University in 2012. Her current research interests include product lifecycle management and knowledge engineering.

Ting Xu was born in Jining in 1991, master candidate in School of Mechanical Engineering, Shandong University. She received her bachelor degree from Shandong University in 2013. Her current research interests include product lifecycle management and scheduling model.