Study on the task scheduling problem of complicated products' design

XiaoBo Cao*, ChengDong Xu, GuoChao Fan

Key Laboratory of Dynamics and Control of Flight Vehicle, Ministry of Education, Beijing

School of Aerospace Engineering, Beijing Institute of Technology, Beijing, China

**Corresponding author's e-mail: caoxiaobo198610@163.com*

Received 1 December 2014, www.cmnt.lv

Abstract

The process of complicated products design has the characters of high complexity, long period and various requirement of resource. So a reasonable resource scheduling scheme has great significance to the design of complicated product, so as to shortening product development cycles and reducing the cost of product. Firstly, design task collaboration and scheduling features in three complicate product design process modes are discussed in terms of research results of collaboration design and Integration Design Environment (IDE). Secondly, Virtual Design Unit (VDU) is adopted to be taken as the basic task execute unit, VDU design is a complex product design, Furthermore, Design Task Scheduling Approach based Design Ability (DTSADA) is detailed stated by combining ACO and GA.. Finally, a design task scheduling case is demonstrated to validate the proposed approach.

*Keywords: c*omplicated product, virtual design unit (VDU), IDE, task scheduling

1 Introduction

With the increase of structure complexity and function requirement, design process of complicate product involves in more and more disciplines and design resources. Customer-oriented product design with low cost and high efficiency has become a tendency. How to realize design resources share and optimization allocation in a high efficiency is the key to short the design time and cost. Aiming to the drawbacks of traditional design modes, some distributed network collaboration design technologies such as Concurrent Engineering (CE) [1], Cloud Computing (CC) [2] and Cloud Manufacturing (CMfg) [3] try to address above problems in design process by design resources virtualization, knowledge reuse, reasonable design process model and high efficiency design task scheduling. It has become a widely attention problem to facilitating product design by scheduling reasonable design resources to finish complicate product design task with low cost and short time.

In the design process of complicate product, design task scheduling problem mainly refers to design task modeling, design executor selection, design ability evaluation and task scheduling approach construction. 1) Design task model must be clearly abstracted from different types of design tasks. It is important to specifically describe task basic information, design constrains, design requirement, task interaction relation and design process association in the design task model. Only if design task targets are determined, can we select suitable design resources to finish it; 2) Design resources must be organized in certain form of high-efficiency executor unit to play full role in the design activity. There are so many kinds of design resources involved in the design process such as design people, equipment, tool, software and network. Design people are the core in the design activity. So it is necessary to construct high-efficiency design unit by matching reasonable resources with designer so as to bring into play maximum initiative of designer; 3) Design abilities of design executor must be accurately evaluated by comprehensively considering design success ratio, design skilled degree and design robust. Design ability of executor will dynamically change following the change of design experience. A design ability evaluation model that can reflect the effect of success ratio, design skilled degree and design robust is indispensable; 4) Optimization mathematics function is supposed to be built, and it is as important as a high-efficiency scheduling approach of design task.

From what we discuss above, our research aims to develop a new design task scheduling approach with accurate design ability model, and it makes efficient utilization of distributed design resources in such a way as to minimize design cost. Remainder sections of this paper are organized as follows.

2 The algorithm of complex task scheduling problem

With the development of distributed network collaboration modes in the domain of computing, design and manufacturing, design task scheduling become an important frontier problem. Some heuristics intelligence algorithms are prevailed to solve the distributed task scheduling optimization problems such as Simulated Annealing (SA), Genetic Algorithm (GA), Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). Based on the above methods, there are a lot of researches oriented to complicated product design task scheduling at home and board.

Yu, XB [4] puts forward a novel adaptive hybrid algorithm based on PSO and DE (HPSO-DE) by developing a balanced parameter between PSO and DE. Adaptive mutation is carried out on current population when the population clusters around local optima. The HPSO-DE enjoys the advantages of PSO and DE and maintains diversity of the population. S.J. Shyu [5] proposes an application of the ACO to a two-machine flow-shop scheduling problem. In the flow-shop, no intermediate storage is available between two machines and each operation demands a setup time on the machines. The transformation of the scheduling problem is translated into a graph-based model. The method seeks to compose a schedule that minimizes the total completion time. A.Y.Abdelaziz [6] introduces the Ant Colony Optimization algorithm (ACO) implemented in the Hyper-Cube (HC) framework to solve the distribution network minimum loss reconfiguration problem. The HC framework limits the pheromone values by introducing changes in the pheromone updating rules resulting in a more robust and easier to implement version of the ACO procedure. ZHANG Yu [7] brought up an algorithm combining Genetic algorithm (GA) and Ant Colony algorithm (ACO) for the programming framework of cloud computing. In the algorithm, the GA adopts task-worker coding method, every chromosome representing a specific scheduling scheme, and chooses the average completing time of all tasks as its fitness function. Timur Keskinturk [8] proposes Ant Colony Optimization (ACO) method outperforms heuristics and genetic algorithm, and it is used to solve the problem of minimizing average relative percentage of imbalance (ARPI) with sequence-dependent setup times in a parallel-machine environment. A mathematical model that minimizes ARPI is proposed. As papers limited, the other researches are not detailed stated.

Most academics place different emphasis on resources selection and task scheduling. But several items are neglected as follows:

1) Pay much attention to task scheduling in single design stage and ignore the relation between task and design process, interaction among design tasks. Design task in design process is not modeled form the angle of multi-granularity and multi-stage. Design task decomposition and combination are also not reasonable.

2) Design ability models are simple or partial. Average completing time and cost are used to calculate the design consumption of one design task while ignore or pay little attention to the effect of task success ratio, design skilled degree and design robust in the design ability quantity evaluation. Experience growth and ability dynamically promotion of manpower in design activity are not considered in the former models.

3) Design task executor is usually limited in single design resource. There are few reasonable design resources organization unit and resources selection mechanism.

4) Most of design task scheduling methods are limited to solve scheduling problem of same task type. In the former scheduling methods, a design executor is constrained to execute one kind of task in the former methods, which cannot play full role of design resources.

Accordingly, in our work we seek to address above problems and contribute to the works towards design resources structure optimization, quantity design ability modeling and high-efficiency design task scheduling.

3 Design task collaboration and scheduling modes

Complicated product design is a complicated collaboration process refers to reasonable tasks decomposition and combination, resources organization, task scheduling and result assessment. And it involves in multiple stages, multiple design entities, multiple iteration processes, multiple disciplinary and multiple resources. From the macroscopic angle, the keys to optimization design cost of complicate product is to reduce the interconnection among design tasks, reduce the iteration times in process and increase the control of design flow. In the design process:

1) Data interaction and design tasks are closed coupled together. Data input and output are finished inner the design task. Data interaction is transparency outside of design module. Only setting the execution sequence of design executor, a design process will be done. The interfaces of design executors is only task driven interface, it is not associated with design behavior, and data interaction is independence with interface.

2) Data interaction and design tasks are loose coupled. Design task executors only take responsibility for the execution of task, and data interaction is independent from design tasks. It made a request that design parameters pattern must match with the interfaces of design task execution modules.

Traditional design systems have used a sequential model for design generation, which breaks the design task into subtasks that are serially executed in a predefined pattern. This type of design process model is easy to realize and modeling. But sequential design is not extensible. Any mistake of one design node will affect the overall design process. Downstream information flow of sequential design makes it difficult to iterative design and feedback from low-level design activity to high-level activity. In this paper, VDU is taken as the example of complicated product, and VDU design process mainly includes design task decomposition, task agglomeration, resources and task scheduling and result evaluation. Refer to complex product design processes of VDU, this paper summarizes design task collaboration modes in three typical VDU design process:

1) Subsystem Paralleled Collaboration Design Pattern.

Subsystem paralleled design pattern is mainly oriented to mature VDU products. In the mode, design process is relatively matured, so it generally does not need to undertake large-scale iterative design across the system. Design object structure is already decided, and subsystems design tasks are separated in the light of production structure. Each subsystem design node is independent designed. When each subsystem attains some periodic result, design results of the subsystem are comprehensively collected and evaluated.

Design confliction are checked out, and modify opinions are also put forward. Design data of the upstream subsystem are delivered to downstream subsystem. Consequently, when some design conflictions between two subsystem design nodes occur, local optimization of subsystem can solve the own existence conflictions. In the level of part design task, confliction between two tasks inside one subsystem is eliminated by iterative amendments of all part design tasks. Conflictions of two part design tasks do not affect the other subsystem design tasks. When mission conversion of two task nodes occurs, design result data, next design requirements and tolerance specifications information are delivered to the execution of downstream process unit.

2) Global Iterative Design Pattern.

Global iterative design pattern is mainly oriented to the study and development of new types of VDU. In the mode, each subsystem is associated design with other subsystems. When each subsystem reaches a stage of node, all subsystem of design result are put together to carry out comprehensive evaluation. Modify opinions is put forward to solve the conflicts by all participants. And modification schemes are used in the process of the next iteration of amendments to eliminate conflict. The synergy mode is called global iterative design pattern. Subsystem is taken as design and verification nodes. Firstly, local optimization design of subsystems is realized. And then overall design optimization is achieved according to optimize and combine the various subsystems index. When problems are founded in the subsystem, it is need to modify each subsystem to iteration to achieve. "Multiple input, multiple output" is the main input and output characteristics of task nodes relation.

3) Assisted Association Design Pattern.

Assisted association design pattern is oriented to component coordinated design, and design tasks are collaborated with several units. As shown in Figure 1.

	Subsystem Paralleled Collaboration Design Pattern	Global Iterative Collaboration Design Pattern	Assisted Association Collaboration Design Pattern
Application object	Mature type missile or component design	New type missile or component design	Component or subsystem design
Design entity/ Design task	Multi-entity/Single object	Multi-entity/Multi-object	Multi-entity/Single object
Executor coupling degree	○	◒	●
Design process complexity	○	●	◒
Concurrent design degree	●	◒	○
Iterative Number	⊗	●	●

NOTE: ⊗ Low ○ Weak ◒ Moderate ● Strong

FIGURE 1 Features of three collaboration design modes

Design changes of a subsystem design is closely subjected by the other subsystem. Every subsystem design node carry forward according to the overall design index. As a design collaborator, modifications of other subsystem design nodes will cause large-scale design to design tasks affiliated to assisted association design pattern. "Multiple input, multiple output" is also the input and output characteristics of task nodes relation.

With comparing features of three modes, in the reality, it is rare to purely use single collaboration mode to complete VDU design. Mostly, three collaboration patterns that we discussed above are appropriate combined on appropriate occasions to meet different types of VDU.

4 VDU model

VDU is defined as a collection of some virtual design resources. VDU is taken as the basic execute unit of design task in this paper. The model of VDU is described in the form of five-tuple.VDU={BasicInfo, Resource, DesignActivity, DesignAbility, Constrain}. Moreover, each tuple is explicated as following:

BasicInfo={ID, Name, BuidTime, VmuType, Position, Workshop, Status} denotes the basic information of VDU such ID, name, construction time, type, position, affiliated unit and status.

Resource={ResouceStructure, ResouceRelation} denotes the design resource and resource structure in the VDU. ResourceStructure={M,E,T}.M={m1, m2, m3,…}.∀m∈M, m denotes man resource, M≠0; E={C,Eq,S…} denote the equipment resources include computers, experiment apparatus ,simulation platform.∀c∈C, c denotes computer resource, C∩E≠0;

DesignActivity={Da1, Da2,…, Dam} denotes the design activities that can be executed in the VDU. Dam={DesignObject, DesignMethod, DesignActivity, DesignInput, DesignOuput} denotes the design activities attributes set.

DesignAbility={DA1, DA2,…, DAn} denotes the design abilities that VDU takes engage in different tasks. DAn={DesignActivityID, SuccessRatio, SkilledDegree, Robust, DesignQuality, DesignCost, DesignTime} denotes the design ability attributes.

Constrain={Con1, Con1,…, Conx} denotes the constrains restrain the design activity of VDU.

As the basic design task execute unit, each VDU can engage in at least 1type of design task. In the design activity, designer, computer equipment, simulation apparatus, software, model, tool and knowledge are the mainly factors that can affect quality of design activity. Inside, all kinds of designers are the dominant body. Designers engage in design, management and maintenance by operating all kinds of hardware and software. So designer is the indispensable element in VDU. In addition, main design computer is as important as designer. Design resources in the VDU connect together by main design computer with network. In the level of atom-resources, main design computer is used to register resources virtual information, accept design task, communicate among atom-resources and collect task information. In the level of VDU, it takes charge the communication and collaboration among different VDUs. Structure diagram of VDU is shown as Figure 2.

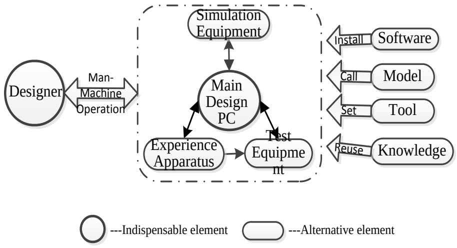

FIGURE 2 VDU structure diagram

5 Design task optimization model and its parameters

Building reasonable design task model is prerequisite of design task scheduling. According to the activity type in complicate product such as air-vehicle, design tasks are classified into product investigation, programming, design indicate determination, geometric modeling, mesh generation, mathematic formulation modeling, flight weight design, work flow design, structure design, aerodynamic analysis, intensity analysis, engine internal trajectory calculation, thermal simulation, control loop design ,control function analysis, warhead explosive simulation, flight kinetic simulation and result evaluation. Aim to abstract a general model for above design task types, design task model is built based on the study results, and it describes the basic information, design instance, design object, design activity, executor information and execution conditions as shown in Figure 3.

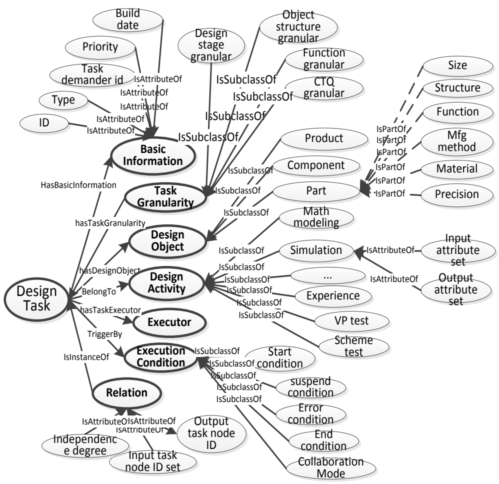

FIGURE 3 Design task optimization model

DT is used to denote design task model, and it is expressed with 7-tuple as follows: DT=(B, G, O, A, E, C, R).

Among them, B denotes basic design information including task ID, task name, independence degree, priority and task demander ID.

G denotes granularity information of design task.

G=(g1, g2, g3, g4). g1=(StageType, PhaseLevel) denotes stage granular dimension information.

g2=(Object-Type, Structure level) denotes design object structure granular dimension information.

g3=(Function Type, FunctionLevel) denote function granular dimension information.

g4=(TCQlevel, Time, Cost, Quality) denotes Time&Cost&Quality granular dimension information.

O=(ObjID,ObjName, ObjClass, Size, Structure, Function, Mat, Mfgmethod) denotes object information of design task, it includes object Id, object name, object category, object size, object structure, function, material and Mfg method.

A=(ActID, Name, Type, Method, InputIDSet, OutputIDSet) denotes design activity information of design task. It includes activity id, name, type, method, activity input attribute ID set and activity output attribute ID set.

E=(ExecutorID, Name, Structurelevel, Function, Affiliation, DesignAbility) denotes design task executor information. It includes executor id, name, structure level, function, affiliation and design ability.

C=(Execuation, Transfer, Collaboration, Start, Suspend, ErrorDeal, End) denotes condition and constrain of design task in different status. It includes task execution condition, transfer condition, collaboration condition, start condition, suspend condition, error dealing condition and end condition.

R=(Independence Degree, InputTaskIDSet, OutputTaskIDSet) denotes the interaction relation with other tasks. It includes task independence degree, Input task Id Set and Output task Id.

From the angle of design task [10], product design process can be defined as a sequential task set that executed with certain interaction and execution constrains. To realize high-efficiency product design, design process modeling must adhere to the following items.

a) Design process model should reflect the dependent degree among design tasks.

b) It is necessary to build input and output relation among different tasks.

c) It can contribute to improve design efficiency and short design cost.

In the design process of complicate product, each subsystem involves many components and parts. Design process of subsystem include design goal dissociation, design executor selection, tasks scheduling, cooperative decision, task execution, and schedule progress monitoring, etc. Thus, design process of complicate product can be described as a set with five elements: DP=<P, E, S, C, M>.

Moreover, each element is explicated as following:

P={Subpro1, Subpro2,…,subpron}: goal dissociation.

The general project can be decomposed into n subprocess that demand interoperation with one another. Each sub-process is composed of many sequential tasks.

E={E1, E2,…, Eq}: task executor selection.

Referring to the rules such as design priority, the lowest price and the shortest producing time, the process management server selects the candidate executors to participate in the cooperation team, and determines qualified total q executors to implement the total n subtasks together. Usually there is q>n.

S={S1, S2,…, Sm}: tasks Schedule.

Each sub-process can be dissolved into several activities or tasks that can be carried out in terms of design cost, design load and design time. Reasonable design task number must be allocate to the candidate executor to finish all the

C={C1, C2,…,Co}: task coordination and negotiation between the different executors.

It devotes to the cooperation style by means of network dialogues and sharing information among all participants. Each Co is described as a set of three tuples Co={Dinterdepend, Rel, Cor}. Design Task (DT) and Design Collaboration (Co) are the important links in design process. DT place emphasis on functional improvement and promotion, Co pay much attention to information interaction, nodes result check and scheme modification. Cost and time of design task are determined by design ability of executors. More design collaboration times can rectify design scheme, reduce iteration of wrong design error and add possibility to attain optimization design scheme. But more Co means more consumption of cost and time. Reasonable evaluation equation is needed to build So that it can determine rational number of Co.

M = {M1, M2,…, Mt}: schedule-progress monitor.

The dominant fractal-agent needs to monitor the schedule progress of the key subtasks. If necessary, the corresponding executive team may demand increased new partners.

6 Algorithm in Complex product development design process

Genetic algorithm (GA) and Ant Colony algorithm (ACO) are both efficient and powerful heuristics intelligence algorithms for solving optimization problems, which have been widely applied in many scientific and engineering fields. According to the comparison between GA and ACO, GA is less efficient than ACO, but it maintain the diversity of solutions. ACO is prevailed in convergence, but it can easily fly into local optima and lack the ability of jumping out of local optima. So it is a good idea to combine them together to realize the efficient researching of global solution. Refer to the hybrid methods combine GA and ACO.

7 Algorithm design

1) Decompose task and generate tasks list.

a) Dealing with multiple kinds of tasks, different design tasks are classified and agglomerated. Purpose of task allocation is to select reasonable number of every kind of task.

b) (2) Select a candidate VDU set according to the requirements of design task type and design characteristic.

c) Relevant degree between task and VDU is proposed to describe the matching degree between one task and one VDU. The value of relevance degree between task and VDU decides whether one VDU is suitable for executor the task. In general, there exists correlation between tasks and VDU. Let T be the correlation matrix of task and VDU, the element Tij denotes the correlation factor between the i-th task and j-th VDU.P=(p1, p2,…, pi)denotes the attributes set of task, and Q=(q1, q2,…, qj) denotes ability attribute set of VDU. Euclidean distance is used to judge the relevance degree between task and VDU as follows

$$d_2(p,q)=\sqrt{\sum_{i=1}^{m}\left|p_i-q_i\right|^2} \quad (1)$$

As shown in Equation (1), if there is strong relevance between the i-th task and j-th VDU, d2(pi,qj)=0. If there is little or no relevance between the i-th task and j-th VDU, d2(pi,qj)≠0. According to the distance of design task and VDU, VDUs are selected as the candidate executors.

(2) Calculate and evaluate task’s execution time and cost of VDU.

d) Execution time, design cost and shipping time of each working procedure are identified, when VDU carry out some design activities. Refer to paper [19], cost calculation Equation of VDU is showed as follows:

$$C_{\mathrm{ri}}=\frac{\sum_{k=1}^{k}(n_{ek}\bullet c_{ek})+\sum_{s=1}^{s}(n_{ms}\bullet c_{ms})+\sum_{t=1}^{t}(n_{tt}\bullet c_{tt})+c_{aux}}{L_{\mathrm{ri}}}. \quad (2)$$

In the Equation (2), Cri denotes the design activity cost when r-th VDU carry out the ith task. nek and cek denote the average work hour number and average hour cost of the k-th kind of equipment. nms and cms denote the number and hour cost of the s-th kind of man.ntt and ctt denote the number and hour cost of the t-th kind of auxiliary tools. caux denotes the sum of other affix cost. Lri denotes the hourly workload of rth VDU when rthVDU take in ith design task. After calculating the average design time and design cost, DAC is used to evaluate design time and cost of next time design task followed Equation (2).

e) (4) Initialize Ant colony.

f) m is assumed as the number of ant. n denotes the number of ant iteration. Every design task type is task as the city node for ant visiting. When m-th ant visits the i-th type of design tasks, ant allocates task number of all design task types to execute by all VDUs. Every ant start randomly from one of i design types. And then, ant select visit sequence and design task number schem until all design task types are executed. All design tasks must be arranged when ant finish the whole journey. When ant completely visits all task types, design task number matrix in the path is generated as one solution. Minimum delivery time is used to judge whether one solution meet the charge time. If Equation (1) is not satisfied，it demonstrates that current task number schemes fail to meet the deliver time limitation. Under such situation, deliver time limitation or outsourcing processing is supposed to adjusted and executed.

g) (5) Initialize solution variable.

h) Candidate VDU could not execute all kind of task, so relevance correlation matrix is introduced as shown in Equation (3). If r-th VDU is suitable for i-th task, eri=0; else eri=1. xri denotes the number that rth VDU carry out i-th type design task. xri is taken as the solution variable. And then, initial value of xri is selected randomly from the value range [0,Ni]. Meanwhile xri value must meet the Equation (4). All the xri construct the design task number matrix of task allocation optimization as Equation (5).

$$E=\begin{bmatrix} e_{11} & e_{12} & \dots & e_{1i} \\ \dots & \dots & \dots & \dots \\ e_{r1} & e_{r2} & \dots & e_{ri} \end{bmatrix} \quad (3)$$

$$x_{ri}=N_i-\sum_{r=1}^{r-1} x_{ri}\bullet e_{ri} \quad (4)$$

$$X=\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1i} \\ \dots & \dots & \dots & \dots \\ x_{r1} & x_{r2} & \dots & x_{ri} \end{bmatrix} \quad (5)$$

i) (6) Select next task type of mth ant at the t time.

j) When mth ant moves from j-th type of design task to k-th type of design task in t time. Taboo table is built to exclude task types that ant has visited. The line between j-th types and k-th types is denoted with (j, k), and pheromone is denoted with τjk. Initial value of τjk is set as τjk(0)=1. Motion transition probability that ant select the j-th design types is formulated as Equation (6).

$$P_{jk}^{m}(\mathrm{t})=\begin{cases} \dfrac{\tau_{jk}^{\alpha}\bullet(\eta_{jk})^{\beta}}{\sum_{l\notin Tabu}[\tau_{jl}(t)\bullet(\eta_{jl})^{\beta}]} \\ 0 \end{cases} \quad (6)$$

k) (7) Update local pheromone. When ant moves from j-th type of design task to k-th type of design task, it leaves pheromone in the line(j,k).The more ants move from line(j, k),the more pheromone leave. When one ant move from line(j,k), local pheromone is updated by the Equation (7) and Equation (8).

$$\Delta\tau_{jk}^{m}=\begin{cases} \dfrac{Q}{d_{all}}, & \text{when ant m vist jk node} \\ 0 & \text{else} \end{cases} \quad (7)$$

$$\tau_{jk}(t+1)=\rho\cdot\tau_{jk}(t)+\Delta\tau_{jk}^{m} \quad (8)$$

When ant visits from start design task type to the current j-th design task type, j-1lines are visited. The total cost Cj and total time Tj of j-1 lines are calculated by path calculation Equation (9) and Equation (10). Walking path length of ant dall is calculated by weighted function Equation (11). To consider effect from both financial cost and time, weight factor δ is introduce to calculate the comprehensive overall cost of the path.

$$C_j=\sum_{j=1}^{\mathrm{j}}(e_{ji}\bullet(\sum_{i=1} x_{ji}\bullet(C_{\mathrm{ji}}+C_{\mathrm{ji}\to i+1}))\,, \quad (9)$$

$$T_j=\sum_{j=1}^{j}\sum_{i=1}^{i}(e_{ji}\bullet(x_{ji}\bullet T_{\mathrm{ji}}+T_{\mathrm{ji}\to i+1}))\,, \quad (10)$$

$$d_{all}=\delta\cdot C_j+(1-\delta)\cdot T_j,(\delta\in[0,1])\,. \quad (11)$$

(8) Generate new solutions by adaptive crossover and mutation operation.

l) In order to avoid the local optimal, crossover and mutation operation have to be carried out to the components of solution after pheromone updating to achieve the global optimal.

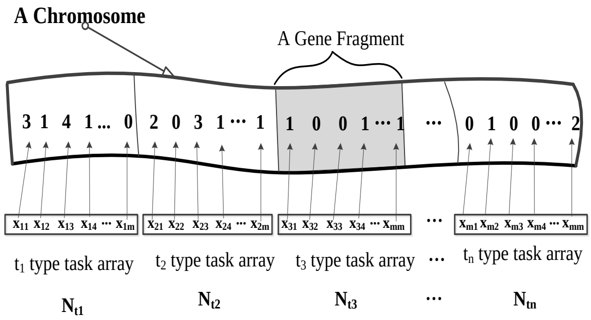

FIGURE 4 A chromosome model of task number

Every ant completely finishes path visiting, a design task number matrix is generated and it is taken as one solution. Each solution is taken as a chromosome of task number as shown in Figure 4. According to path length Equation (11), two solutions are selected to carry out crossover by certain probability. Suppose that m-th ant path length is denoted by dover, m and its probability of being chosen is formulated as Equation (12). In the traversal paths of m ants, the smaller the path length ant travel, the choose probability is smaller. Smaller choose probability means better solution.

$$\mathrm{P}_m=d_{over,m}\,/\sum_{d=1}^{m} d_{over,d} \quad (12)$$

In the crossover operation, supposed x(1)=(x11, x21,…, xr1)Tand x(2)=(x12, x22,…, xr2)T are 1th and 2th column elements set of selected solution(X). Crossover is executed between column elements set x(1) and x(2). Real crossover probability is calculated by Equation (13) and Equation (14).

$$\mathrm{P}_{cross}=P_{sysem}\bullet p_c\,, \quad (13)$$

$$\mathrm{P}_c=d_i\,/\sum_{h=1}^{m} d_h\,, \quad (14)$$

$$d_i=\sum_{r=0}^{r}[e_{ri}\bullet x_{ri}\bullet(C_{\mathrm{ri}}+C_{\mathrm{ri}\to i+1})]\,. \quad (15)$$

psystem is generated by system. Comparison probability p∈[0,1] is also randomly generated. If P>psystem, crossover operation is executed. α∈[0,1] is generated randomly. x(1)' and xr(2)' are used to replace x(1) and x(2). x(1)'=α·x(1)+(1-α)·x(2); x(1)'=α·x(2)+(1-α)·x(1).

After executed crossover operation, mutation operation is randomly executed between two elements of column elements set x(1) and x(2) by certain system probability. Real mutation probability is calculated by Equation (16)

$$\mathrm{P}_{mutation}=P_M\bullet p_m \quad (16)$$

PM is generated by system, and Comparison probability p∈[0,1] is also randomly generated. If p>pM, mutation operation is executed. ε∈[0,1] is generated randomly, x11'=ε·x11+(1-ε)·x12 ; x12'=ε·x12+(1-ε)·x11.

(9) Select best solution and carry on iterate.

m) After above crossover and mutation, two new solutions are generated and walking path length is calculated. Design task number matrix of the shortest walking path in m solutions is selected as the initial solution of next iteration.

(10) Update Global pheromone.

n) For the shortest line of path, pheromone is updated by Equation (17) and Equation (18).

$$\tau_{jk}^{new} = \alpha \bullet \tau_{jk}^{old} + \Delta\tau_{jk}^{m} \quad (17)$$

$$\Delta\tau_{jk}^{m} = \begin{cases} \frac{Q}{d_{shortest}}, & \text{when (j,k)is the shortes line of path} \\ 0 & \text{else} \end{cases} \quad (18)$$

(11) Finish Iteration and output the best task allocation scheme.

The experimental results and the parameter analysis.

Task scheduling as discussed above is an assignment of the task type and task number to the suitable task executors [9]. Refer to the DAC model and DTSADA. To demonstrate the validation of DTSADA, traditional GA algorithm and ACO algorithm are also integrated into the prototype system. To compare the efficiency of GA, ACO and DTSADA, a design task schedule case is used to discuss the iteration efficiency and solution diversity. Design task attributes and task number of every task type are listed in Figure 5. There are 6 candidate VDUs that can execute the above design tasks. Each VDU has different design abilities to deal with different types of design tasks. Design ability attributes of 6 candidate VDUs are listed in Figure 6. Budget overall design cost is ￥42759, and budget overall design time is 2723 work hours. What the problem we should solve is try to figure out a design task scheduling scheme with lowest overall design cost by following constrains of budget design cost and budget design time.

Activity Type	Object	ID	Specification	t_{budget}	c_{budget}	N
Geometric Modeling	Part	1	10^1 features	5	52	20
	Part	2	10^2 features	8	130	20
	Component	3	10^2 features	13	275	15
	Component	4	10^3 features	16	340	12
Mesh Modeling	Structure mesh	5	10^4 features	9	120	25
		6	10^5 features	10	160	22
	Non-structure mesh	7	10^4 features	3	60	10
		8	10^5 features	5	80	3
Aerodynamic Analysis	3 dimension	9	10^3mesh number	11	150	18
		10	10^4mesh number	16	210	25
		11	10^5mesh number	17	240	15
Structure Analysis	Statics	12	10^3mesh number	6	60	3
	Dynamics	13	10^4mesh number	10	150	32
		14	10^5mesh number	11	185	28
Flight Dynamics Analysis		15	6 Freedom degree	7	145	9
Internal trajectory Analysis		16	10^1 constrains	4	42	2
		17	10^2 constrains	8	115	2

t_{budget} is denotes the maximum average budget time to finish one single task.

c_{budget} is denotes the maximum average budget cost to finish one single task.

☆(**Budget Overall Design Time = (h)2723**
Budget Overall Design Cost = (￥)42759)

FIGURE 5 Design tasks attributes

Name	Activity Type	Object	Specification	$\bar{t}_i$	c_i	$Coef$	C_{avr}	C_{DAC}
Vdu1	Geometric Modeling	Part	10^1 features	2.5	20.5	0.95	51.3	54
		Part	10^2 features	5.5	22	0.92	121	131
		Component	10^2 features	13	18	0.95	234	246
		Component	10^3 features	15	20	0.96	300	312
	Mesh Modeling	Structure mesh	10^4mesh number	7	16	0.97	112	115.4
			10^5mesh number	9	18	0.92	162	176
Vdu2	Geometric Modeling	Part	10^1 features	4	15	0.98	60	61.2
		Part	10^2 features	9	15	0.97	135	139.1
		Component	10^2 features	12.5	20	0.91	250	274.7
		Component	10^3 features	15.2	20	0.905	304	336
	Mesh Modeling	Structure mesh	10^4mesh number	8	14	0.97	112	115.4
			10^5mesh number	10	16	0.98	160	163
Vdu3	Aerodynamic Analysis	3 dimension	10^3mesh number	10.5	13	0.95	136.5	143
			10^4mesh number	15.5	13	0.92	201.5	219
			10^5mesh number	15	15	0.92	225	239
	Structure Analysis	Dynamics	10^4mesh number	8	18	0.97	144	148
			10^5mesh number	10.5	16	0.92	168	182
Vdu4	Geometric Modeling	Part	10^1 features	3	18.5	0.96	55.5	58
		Part	10^2 features	7	16	0.965	112	116
	Structure Analysis	Statics	10^3mesh number	4.5	12.5	0.95	56.5	59
		Dynamics	10^4mesh number	9	15	0.925	150	162.5
	Flight Dynamics Analysis			7	18	0.91	126	138.6
Vdu5	Geometric Modeling	Part	10^1 features	3.5	18	0.98	63	64
		Part	10^2 features	8	20	0.97	160	165
	Mesh Modeling	Structure mesh	10^4mesh number	5.5	18	0.95	121	127.4
			10^5mesh number	8	20	0.925	160	173
		Non-structure mesh	10^4mesh number	3	18	0.91	54	59
			10^5mesh number	4	18	0.905	72	80
	Aerodynamic Analysis	3 dimension	10^3mesh number	10	14	0.95	140	147
			10^4mesh number	12	16	0.95	192	202
			10^5mesh number	18	12	0.92	216	234
	Structure Analysis	Dynamics	10^4mesh number	9	15	0.95	135	142
			10^5mesh number	11	13	0.97	145	147
Vdu6	Structure Analysis	Dynamics	10^4mesh number	9	17	0.95	153	161
			10^5mesh number	11	15	0.97	165	170
	Internal trajectory Analysis		10^1 constrains	2.5	15.5	0.95	38.8	40.8
			10^2 constrains	5.8	18.5	0.95	107.3	113
	Flight Dynamics Analysis			6.5	20	0.905	130	143.6

☆($\bar{t}_i$ **is average work hours number of complete a task, its unit is hour.**
c_i **is design resources cost /h of complete a task, its unit is ￥/h.**
$C_{avr} = \bar{t}_i \bullet c_i$ **is average cost of complete a task, its unit is Yuan.**)

FIGURE 6 Design ability attributes of candidate VDUs

Traditional GA algorithm and ACO algorithm are used to compare the iteration efficiency and result accuracy with DTSADA. According to the simulation result in Figure 6 and Figure 7, GA prevails in the global solution, but it needs more iteration than ACO when the solutions are equal. Convergence speed of ACO is faster than GA, but it is easy to fall into local optimization. DTSADA assimilates the advantages of GA's solution diversity and ACO's convergence efficiency. Simulation results show that DTSADA is valid. Simulation result curve of GA, ACO and DTSADA is illustrated in Figure 7.

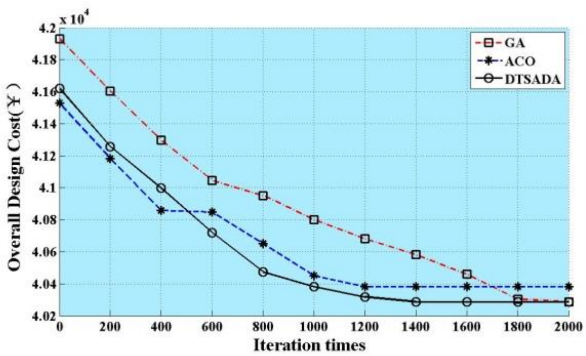

FIGURE 7 Simulation result curve of GA, ACO and DTSADA

TABLE 1 50 times simulation result of three methods

Section	Average value of 50 times simulation	Best Value	Variance
GA	40290	40288	0.386
ACO	40325	40288	1.716
DTSADA	40289.2	40288	0.341

Activity Type	Object	Specification	*N*	*Vdu*1	*Vdu*2	*Vdu*3	*Vdu*4	*Vdu*5	*Vdu*6
Geometric Modeling	Part	10^1 features	20	4	3	0	3	10	0
	Part	10^2 features	20	13	5	0	1	1	0
	Component	10^2 features	15	12	3	0	0	0	0
	Component	10^3 features	12	6	6	0	0	0	0
Mesh Modeling	Structure mesh	10^4 features	25	9	11	0	0	5	0
		10^5 features	22	2	16	0	0	4	0
	Non-structure mesh	10^4 features	10	0	0	0	0	10	0
		10^5 features	3	0	0	0	0	3	0
Aerodynamic Analysis	3 dimension	10^3mesh number	18	0	0	6	0	12	0
		10^4mesh number	25	0	0	5	0	20	0
		10^5mesh number	15	0	0	5	0	10	0
Structure Analysis	Statics	10^3mesh number	3	0	0	0	3	0	0
	Dynamics	10^4mesh number	32	0	0	26	0	4	2
		10^5mesh number	28	0	0	2	0	25	1
Flight Dynamics Analysis			9	0	0	0	1	0	9
Internal trajectory Analysis		10^1 constrains	2	0	0	0	0	0	2
		10^2 constrains	2	0	0	0	0	0	2

☆(**Minimum Overall Design Cost = (￥)40288**
Overall Design Time = (h)2431.4)
Tasks execute sequence is (3 1 5 2 10 7 15 4 9 12 13 8 14 17 16 6 11)

FIGURE 8 Design task schedule result in minimum execute cost

8 Conclusions

Through the study of complex product development present situation at home and abroad and the present situation of the resource-constrained project scheduling, taking the VDU in the design of complex product development part of the process of design process for example. A hybrid design task scheduling approach combined GA and ACO-DTSADA is proposed. A prototype system of design task scheduling based DTSADA is constructed. Simulation demonstrated that DTSADA is valid. Solving the actual complex product development design process has important practical significance. Future research includes: (1) Multi-stage design task allocation method. (2) Multi-granularity design ability modeling.

Acknowledgment

This work is supported by the National High-Tech.R&D Program, China (No. 2011AA120505) and the National Natural Science Foundation, China (No. 61173077).

References

[1] Gardoni M 2005 Concurrent Engineering in Research Projects to Support Information Content Management in a Collective Way *Concurrent Engineering* **13** 135-43 DOI: 10.1177/1063293X05053798

[2] National Institute of Science and Technology. The NIST definition of Cloud Computing [EB/OL] [2011-06-21] http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[3] Li B, Zhang L, Wang S 2010 Cloud manufacturing: a new service-oriented networked manufacturing model *Computer Integrated Manufacturing Systems CIMS* **16** 1-7

[4] Kirkpatrick S, Gelatt C Jr, Vecchi M 1983 Optimization by simulated annealing *Science* (220) 671-80

[5] Goldberg D 1989 *Genetic Algorithms in Search, Optimization, and Machine Learning* Addison-Wesley Publishing Company Inc Reading MA

[6] Dorigo M, Gambardella L 1997 Ant colony system: a cooperative learning approach to the traveling salesman problem *IEEE Transaction on Evolutionary Computation* **1** 53-66

[7] Yu X B, Cao J, Shan H Y 2014 An Adaptive Hybrid Algorithm Based on Particle Swarm Optimization and Differential Evolution for Global Optimization *Scientific World Journal* DOI: 10.1155/2014/215472

[8] Shyu S J, Lin B M T, Yin P Y 2004 Application of ant colony optimization for no-wait flowshop scheduling problem to minimize the total completion time *Computers & Industrial Engineering* **47** 181-93

[9] Abdelaziz A Y, Osama R A, El-Khodary S M 2011 Distribution Systems Reconfiguration Using the Hyper-Cube Ant Colony Optimization Algorithm *SEMCCO Part II LNCS* 7077 257-66

[10] Zang Y, Li F, Zhou T 2012 Task scheduling algorithm based on Genetic Ant colony algorithm in Cloud Computing environment *Computer Engineering and Applications*