Team Capacity Evaluation and Scheduling Model for Collaborative Product Design

Xiaolei Wang
Beijing Normal University Zhuhai, North China Electric Power University
China
wxlwxl916@126.com

ABSTRACT: *In collaborative product design project, reasonable resource allocation can shorten development cycle and reduce cost. Then, team capacity evaluation and task-team scheduling model are presented. Collaborative team capacity model is constructed, and 2-tuple linguistic method is used to evaluate the capacity of collaborative team. Then, matching degree between design task and collaborative team is defined. A collaborative product design scheduling model considering task-team matching is developed. Combined with simulated annealing operator, based on single-coding strategy, self-adaptive multi-point cross and mutation, an improved genetic algorithm is proposed to solve the model. Finally, a case study is presented to validate the method.*

Keywords: Team Capacity Evaluation, 2-tuple Linguistic, Genetic Algorithm, Collaborative Product Design

Received: 12 September 2017, Revised 11 October 2017, Accepted 18 September 2017

1. Introduction

With the increasing global competition and growing complexity of product, the division of labor is more and more specialized, so that the core firm needs joint with customer, supplier and research institute to overcome these limitations. Through cross-organizational collaborative product design, it can realize the maximization of resource integration and knowledge sharing, as well as the improvement of design efficiency. However, in the process of collaborative product design, the diversity of design agent, and interdependence and mutual restriction between tasks, these make collaborative product design process become more complicated. Therefore, design task and resource should be reasonably allocated to shorten development cycle and reduce cost.

There is a great amount of research work on task and resource allocation of collaborative design project. Some of these researches focused on task identification, task relationship analysis and task scheduling based on Petri Nets[1] and Design Structure Matrix (DSM)[2]. Some of other researches focused on the establishment of task and resource dynamic scheduling optimization model and model solution based on a variety of algorithms, such as heuristic algorithm and intelligent algorithm [3-4]. Pang established design task net and constructed task assignment model from tasks to team members based on the principle of equilibrium-moderation[5]. Yu proposed multi-objective dynamic fuzzy scheduling considering emergency in product collaborative design [6]. Currently, in the study of capacity and matching degree for CPD, Frillman proposed a competency model for engineers functioning in a PLM environment, it emphasized individuals' competencies[7]. Wu proposed a resource capability measuring method and resource capability deployment mechanism by mapping RTCI to RPCI[8]. Based on agent simulation, Zhang and Li simulated the human working behaviors in collaborative product development process, where design agent selected his partner according to the ability and character matching degree[9]. Furthermore, they analyzed static single category resource scheduling problem and multi-category resource static scheduling problem [10]. Based on ontology and service capabilities, He and Hu proposed matching rules and algorithms of manufacturing tasks and services[11]. But these researches did not consider the match between task and collaboration team.

For collaborative product design project, the project is decomposed into tasks firstly and tasks are allocated to collaborative team. Then, tasks are decomposed into sub-tasks or more detailed tasks, these sub-tasks or detailed tasks are assigned to individuals. The previous researches have focused on the matching between task and individual based on task priority or designer preference. However, taking design team as a whole, from the perspective of system engineering, how to realize task-team reasonable matching? Furthermore, for partner selection or task assignment, they need to measure collaborative team comprehensive capacity. It not only refers to individual competency, but also members' cooperation. In addition, for task allocation, it is necessary to evaluate the capability of collaborative team meanwhile consider the cost.

In the sections that follow, capacity model of collaborative team is presented firstly. Then, 2-tuple linguistic evaluation method is adopted to evaluate the capacity of collaborative team. Following this, matching degree (MD) is defined. Furthermore, scheduling model considering matching degree is established, and the improved genetic-annealing algorithm is designed to solve the scheduling mode. An example is solved successfully to illustrate the feasibility and validity of the proposed method and model. Finally, conclusions are presented.

2. Team Capacity Evaluation Based on 2-Tuple Linguistic

2.1 Capacity Model of collaborative team.

Collaborative product design, as a multi-agent involved and knowledge-intensive activity, it emphasizes collaborative work between design teams. Even more, creative customers and suppliers are involved. These innovative design agents have different background knowledge, experience, skill level and interests, which means that every team has its own special abilities and resources. Therefore, collaborative product design not only requires reasonable design task decomposition, but also needs reasonable matching between innovation team and task which has important influence on the efficiency and cost of product design.

Capacity reflects the skill or ability sets necessary for the relevant tasks. Capacity model needs describe the capacity elements for a task. When finding an appropriate team to carry out a design task, team capacity should be considered. For collaborative work, information sharing, goal congruence, decision synchronization, resource sharing, collaborative communication, and joint knowledge creation are significant and interconnecting elements[12-13]. Meanwhile, they are the prerequisite elements.

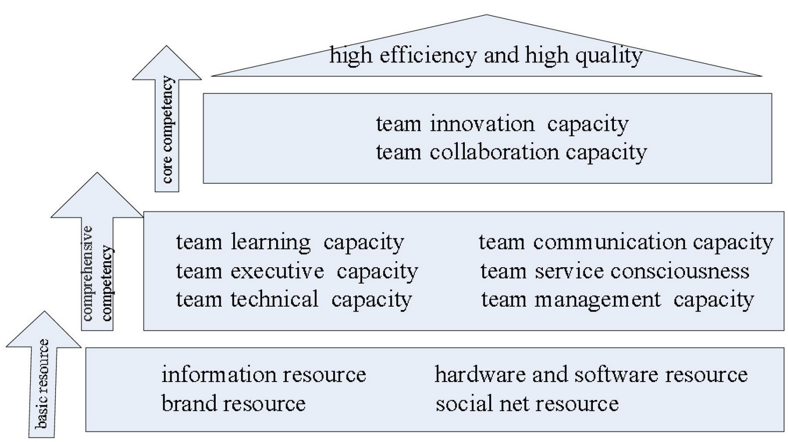

Figure 1. Capacity model of collaborative product design

Thus, the capacity model of collaborative team is constructed as shown in Figure 1.

In the model, collaborative product design team resources contain hardware and software resource, brand resource and social net resource. These are basic resources for collaborative work. Where, information resource includes available technical information and industry information. Important customers, government, and partners in other industries constitute team's social net resource. The comprehensive capacity consist of team learning capability, communication capability, team executive capability, technical capability, service consciousness, management capability. Learning capability and communication capability are more important than the others at this level. The core capacities are team innovation capability and collaboration capability. Team collaboration requires good communication and executive ability as well as excellent team management. Learning capability and technical capability are important prerequisite and foundation for innovation. Finally, high efficiency and high quality are the ultimate goals.

2.2 Team Capacity Evaluation based on 2-Tuple Linguistic

For capacity evaluation, the common methods are based on fuzzy mathematics theory, such as AHP, triangular fuzzy numbers. However, in these methods, fuzzy operation based on extension principle increases the fuzziness of the results and results in information loss or distortion. In addition, evaluation experts often adopt natural language to express their preference, such as they use "high", "average" and "low" to evaluate the team capacity. Maybe, they use "very high", "high", "average", "low" and "very low" to express their evaluation results. That is, different experts maybe express their evaluation information with different granularity. 2-tuple linguistic method can effectively aggregate natural language evaluation information of different granularity to avoid information loss and make the result more precise [14-15]. Thus, 2-tuple linguistic method is adopted to evaluate the competencies of collaborative team.

2-tuple linguistic method represents the linguistic evaluation information by means of a two-tuple (s_i, α_i), where s_i is a linguistic label from predefined linguistic term set $S = \{s_0, s_1, \ldots, s_g\}$ and α_i is the value of symbolic translation, $\alpha_i \in [-0.5, 0.5)$, and $g + 1$ is the granularity of the set S. For example, a set $S = \{s_1, s_2, s_3, s_4, s_5\}$ represents the evaluation information set. The meanings of linguistic terms s_1, s_2, s_3, s_4, s_5 are "very high", "high", "average", "low" and "very low", respectively.

Definition 1: A real number $\beta \in [0, g]$ is a number value representing the aggregation result of linguistic symbolic. The function Δ used to obtain the 2-tuple linguistic information equivalent to β is defined as:

$$\Delta\text{: } [0, g] \rightarrow S \times [-0.5, 0.5), \quad \Delta(\beta) = \begin{cases} s_k, k = round(\beta) \\ \alpha_k = \beta - k, \alpha_k \in [-0.5, 0.5) \end{cases} \tag{1}$$

Where round( ) is the rounding operator, S_k has the closest index label to β, α_k is the value of the symbolic translation.

On the contrary, the 2-tuple linguistic variable can be converted into the crisp value β by the inverse function Δ^{-1}:

$$\Delta^{-1}\text{: } S \times [-0.5, 0.5) \rightarrow [0, g], \quad \Delta^{-1}(s_k, \alpha_k) = k + \alpha_k = \beta \tag{2}$$

Definition 2: Let $S = \{(s_1, \alpha_1), (s_2, \alpha_2), \ldots, (s_m, \alpha_m)\}$ is a 2-tuple linguistic variable set with same granularity, the arithmetic average operator of the set is computed as follow:

$$(\bar{s}, \bar{\alpha}) = \Delta\left[\frac{1}{m}\sum_{j=1}^{m}\Delta^{-1}(s_j, \alpha_j)\right], \quad \bar{s} \in S, \bar{\alpha} \in [-0.5, 0.5) \tag{3}$$

Definition 3: Let $S = \{(s_1, \alpha_1), (s_2, \alpha_2), \ldots, (s_t, \alpha_t)\}$ be a set of 2-tuple and $C = \{(c_1, \beta_1), (c_2, \beta_2), \ldots, (c_t, \beta_t)\}$ be the linguistic weighting vector of 2-tuple (s_k, α_k) $(k = 1, 2, \ldots, t)$, The extended 2-tuple weighted geometric (ET-WG) operator is defined as follows [16]:

$$(\tilde{s}, \tilde{\alpha}) = ET\_WG_C((s_1, \alpha_1), (s_2, \alpha_2), \ldots, (s_t, \alpha_t)) = \Delta\left(\prod_{k=1}^{t}(\Delta^{-1}(s_k, \alpha_k))^{\frac{\Delta^{-1}(c_k, \beta_k)}{\sum_{k=1}^{t}\Delta^{-1}(c_k, \beta_k)}}\right) \tag{4}$$

Definition: 4 Let $(\tilde{s}_1, \tilde{\alpha}_1)$, $(\tilde{s}_2, \tilde{\alpha}_2)$, …, $(\tilde{s}_u, \tilde{\alpha}_u)$ are the two-tuple linguistic information with different granularities that will be aggregated. u is the number of groups. The improved EOWA operator is defined as:

$$(s^*, \alpha^*) = IEOWA((\tilde{s}_1, \tilde{\alpha}_1), (\tilde{s}_2, \tilde{\alpha}_2), \ldots, (\tilde{s}_u, \tilde{\alpha}_u)) = \Delta(\lambda_1'(\Delta^{-1}(\tilde{s}_i, \tilde{\alpha}_i))) \quad (5)$$

Where $(\tilde{s}_i, \tilde{\alpha}_i)$ is the evaluation information with the ith maximum granularity, λ_i' is the ith maximum number in array λ. $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_u)$ is the weight of EOWA operator, it is quantified by the fuzzy operator $E(r)$:

$$\lambda_i = E(i/u) - E((i-1)/u),\ i = 1, 2, \ldots, u$$

$$E(r) = \begin{cases} 0 & r < a \\ (r-a)/(b-a) & a \le r \le b \\ 1 & r > b \end{cases} \quad (6)$$

Where a, b, $r \in [0, 1]$, corresponding to the fuzzy linguistic quantitative principle of "half", "most" and "as much as possible", the parameters (a, b) take values (0, 0.5), (0.3, 0.8), (0.5, 1), respectively.

The specific evaluation steps are as follows:

Step 1: The experts with the same granularity are divided into a group. The weight evaluation result of expert k (k= 1, 2,…, t) for capacity y is denoted as (c_k^y, β_k^y). The evaluation result of team j for task i in capacity y given by expert k is denoted as $(c_{kij}^y, \beta_{kij}^y)$. According to the Equation (4), the integrated information of group with the same granularity, denoted as $(\tilde{s}_{ij}^y, \tilde{\alpha}_{ij}^y)$, is got.

Step 2: Obtaining the weight vector $\lambda' = (\lambda_1', \lambda_2', \ldots, \lambda_u')$ according to the improved EOWA operator, then, aggregating the integrated information $(\tilde{s}_{ij}^y, \tilde{\alpha}_{ij}^y)$ according to the Equation (5), the comprehensive evaluation information of team j for task i in capacity y, denoted as $(s_{ij}^y, \alpha_{ij}^y)$ is got. Then, it is converted into a crisp value g_{ij}^y.

3. Scheduling Model for CPD

3.1 Matching Degree Between Task and Team

Matching degree refers to measure the fitness between elements. For example, when matching project task with collaborative team, if the matching degree is too low, it manifests that collaborative team's capabilities and resources are not enough to support them to complete the task. Higher matching degree ensures team can accomplish the tasks high-efficiency and high-quality, but it also means cost rise. Thus, this paper constructs a task-team matching degree model of collaborative product design project.

The task-team matching degree model is constructed in two ways, one is personnel capability matching degree of collaborative team, and another one is the available resources matching degree.

The matching degree between task i and team j at the dimension of personnel capabilities, denoted as TC_{ij}, is defined as follows:

$$TC_{ij} = \sum_{p=1}^{8} \alpha_i^p \left(1 \pm \sqrt{\frac{\left|(g_{ij}^p)^2 - (e_i^p)^2\right|}{(e_i^p)^2}}\right) \quad (7)$$

Where p denotes the pth personnel capability; α_i^p is the weight of the pth personnel capability for task i; g_{ij}^p is the evaluation value of the p th personnel capability of team j for task i, e_i^p is the required value of the p th personnel capability for task i. In Equation (7), if $g_{ij}^p > e_i^p$, takes "+"; else take "-".

Some available resources can be quantified, such as hardware and software. Thus, the matching degree calculation model between project task i and collaborative team j at the dimension of available resource, denoted as TR_{ij}, is defined as follows:

$$TR_{ij} = \sum_{r=1}^{4} \beta_i^r * \frac{g_{ij}^r}{e_i^r} \tag{8}$$

Where, r denotes the rth resource; β_i^r is the weight of the rth resource for task i, g_{ij}^r is the available amount of the rth resource of team j for task i, e_i^r is the required amount of the rth resource for task i.

Furthermore, the matching degree (MD_{ij}) between task i and team j is defined as:

$$\begin{aligned} MD_{ij} &= \omega_{i1} * TC_{ij} + \omega_{i2} * TR_{ij} \\ &= \omega_{i1} (\sum_{p=1}^{8} \alpha_i^p (1 \pm \sqrt{\frac{\left|(g_{ij}^p)^2 - (e_i^p)^2\right|}{(e_i^p)^2}})) + \omega_{i2} (\sum_{r=1}^{4} \beta_i^r * \frac{g_{ij}^r}{e_i^r}) \end{aligned} \tag{9}$$

Where w_{i1} and w_{i2} are the weights of personnel capability and available resource for task i, respectively.

3.2 Scheduling Model

In collaborative innovation project, through rational resource select and configure according to project tasks' requirement, duration and cost optimal are achieved.

Parameters:

PT : The project duration;

C: The project cost;

T: The set of project tasks, $T=\{T_1, T_2, \ldots, T_m\}$;

G: The set of collaborative teams, $G=\{G_1, G_2, \ldots, G_n\}$, n is the number of collaborative team;

$S=\{s_{t1}, s_{t2}, \ldots, s_{ti} \ldots, s_{tm}, s_{tm+1}\}$, s_{ti} denotes the start time of task i, task m+1 is a virtual task;

MD_{ij}: The matching degree between task i and team j;

t_{Ni} : The standard expected execution time of task i;

Δt_i : The maximum shorten amplitude of execution time for task i;

t_{ij} : The expected time of collaborative team j to execute task i.

For collaborative product design, the shortened duration often leads to increased costs. Chen et al proposed that there was linear relation between activity time reduction and cost increases, the time-cost trade-off problem can be transferred into a linear programming problem [17]. Thus, the optimization objective is as follows:

$$\min f(x) = a_1 * PT + a_2 * C = a_1 * S_{tm+1} + a_2 * C \tag{10}$$

Constraints:

$$x_{ij} = \begin{cases} 1, & \text{team j complete task i} \\ 0 & \text{else} \end{cases} \tag{11}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \tag{12}$$

$$\sum_{i \in A_t} x_{ij} = 1 \tag{13}$$

$$e^i_{rmin} \le e^i_r \le e^i_{rmax} \tag{14}$$

$$S_{tq} = \max\min(S_{ti} + t_{ij}), T_i \in B_{(q)} \quad (15)$$

$$t_{ij} = \begin{cases} \frac{t_{Ni}}{\mathrm{MD_{ij}}}, & \mathrm{MD_{ij}} \le 1.0 \\ \mathrm{Max}\{\frac{t_{Ni}}{\mathrm{MD_{ij}}}, (t_{Ni} - \Delta t_i)\}, & \mathrm{MD_{ij}} > 1.0 \end{cases} \quad (16)$$

In the objective function $f(x)$, a_1 and a_2 are the weights of project duration and cost respectively. Constraint (12) expresses resource constraint. Constraint (13) ensures that the task i just only be done by one collaborative team. Constraint (14) ensures that one collaborative team just only can perform one task at a period, A_t denotes collection of tasks that are carrying out at time t. Constraint (15) denotes time constraint, $B(q)$ is precedence activities set of task q. Equation (16) denotes the time will be taken for collaborative team j to finish task i while considering the matching degree.

4. The Improved GA

The issue proposed in this paper is combinatorial optimization problem, but it is different from traditional combinatorial optimization problems that the encoding cannot be repeated. A collaborative team can execute several tasks as long as the tasks do not overlap in one period. To solve the problem, genetic algorithm is improved, where genetic operators are used to represent the individual of feasible solution in the encoding process. Single-coding in the solution space not only eliminates the decoding process between gene space and solution space, but also can enhance the accuracy and reduce the complexity of computation process.

The steps of improved genetic algorithm are as follow:

(1) Coding
Adopting decimal single coding, each gene locus represents the task code and the number on the gene locus represents the corresponding matching collaborative team, as shown in Figure 2.

Chromosome	$Gene_1$	$Gene_2$	$Gene_3$	$Gene_4$	$Gene_5$	$Gene_6$	……	$Gene_m$
Collaborative Team	G_5	G_1	G_3	G_8	G_2	G_4	……	G_7

Figure 2. Coding

(2) Fitness Function
The fitness function of GA is known as evaluation function, it is used to determine the quality of individual. In this article, objective function is set as fitness function F(x).

$$F(x) = f(x)$$

(3) Selecting the Initial Population
Randomly generate a certain number of individuals. Then, remove the repeated individuals and the individuals that do not meet the constraints, choose the best individual into the initial population and select a-1 individual from the remaining individuals randomly. These individual compose initial population with number of a. The probability (p_i) that can be selected is set as follow.

$$p_i = \frac{F_i}{\sum F_i} \quad (17)$$

(4) Crossover Operator
Multi-point crossover is adopted. In the process of evolution, if the current individual fitness is lower than the average fitness, the individual evolution is invalid. In order to improve the search speed, it is necessary to improve individual crossover

probability. Therefore, adaptive crossover probability strategy is adopted. The crossover probability (p_c) is defined as

$$p_c = \begin{cases} p_{c1} - \dfrac{(p_{c1} - p_{c2})(F_i - F_{av})}{(F_{\max} - F_{av})} & F_i \geq F_{av} \\ p_{c1} & F_i < F_{av} \end{cases} \tag{18}$$

Where, F_{av} and F_{max} are the average fitness and the largest fitness, respectively.

(5) Mutation Operator

Execute mutation operation for each individual, the gene changes at a certain probability and varies from 1 to n (n is the total number of collaborative team). In the process of mutation, single point mutation is used the first half of the individual, and multi-point mutation is adopted in the second part.

(6) Selection Operator

The previous generation population, population after crossover and population after mutation constitute the selection set. Remove the ones that do not meet the constraints. Then, the best individual of preceding generation population, crossover population and mutation population are retained respectively. For the remaining individuals, two individuals are selected randomly and one of them is chosen using simulated annealing operator with probability exp(-Äc/è) to bring into the next generation, the other is taken back.

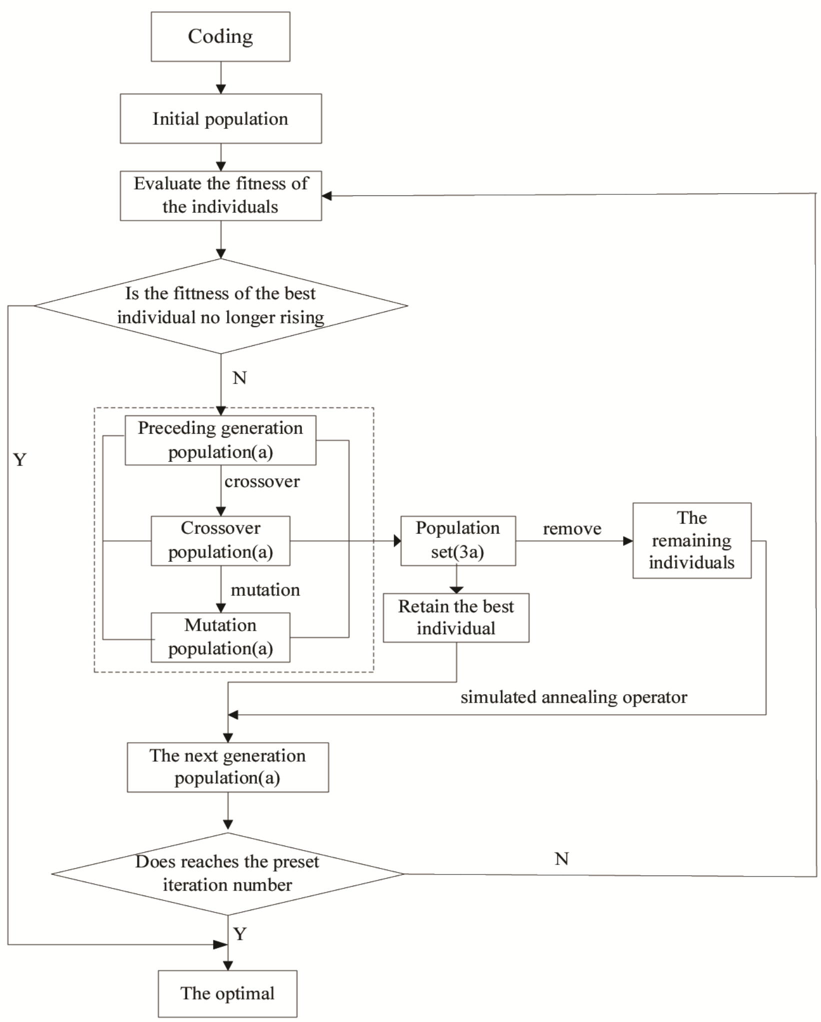

Figure 3. The procedure of improved genetic algorithm

Repeat the above procedure until the amount of next generation reached number a, then turn into the next round.

(7) Termination Condition, Output the Optimal
When meet one of the conditions, the iteration is stopped:

(1) Fitness of the best individual and the group are no longer rising;

(2) The number of iterations reaches the preset number.

The procedure of improved genetic algorithm is shown in Figure 3.

5. Case Study

First of all, we conducted an experiment on our scheduling optimization algorithm of mobile phone collaborative product design. The relationship of design task was shown in Figure 4. Totally 15 tasks were included in the project, and 20 collaborative teams were available.

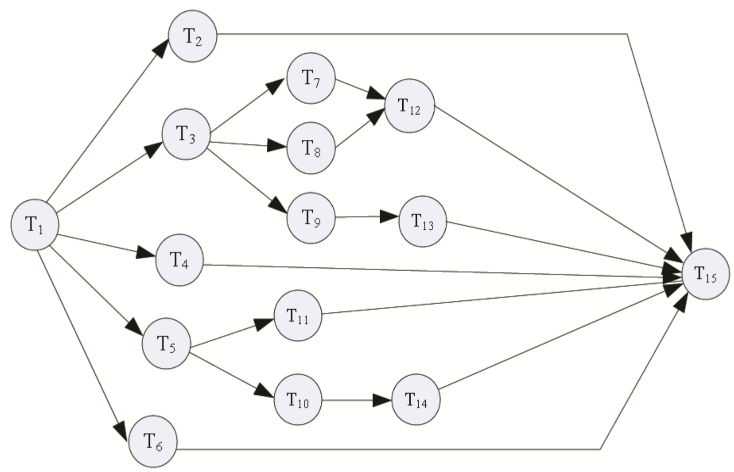

Figure 4. Task relationship

Standard time and the maximum shorten time of the tasks were shown in Table 1.

	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}(Days)
t_{Ni}	4	5	5	6	30	30	25	7	5	15	5	1	1	1	4
Δt_i	1	2	3	2	2	3	2	1	2	3	2	0.5	0.2	0.5	2

Table 1. Standard execution time and the maximum shorten time of design tasks

The matching degrees between collaborative teams and tasks were shown in Table 2.

	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}	G_{16}	G_{17}	G_{18}	G_{19}	G_{20}
T_1	1.859	0.514	1.358	1.608	1.149	1.446	0.468	1.465	1.022	1.259	1.427	2.267	1.906	1.487	1.039	1.300	1.973	1.814	0.502	2.296
T_2	1.604	0.911	0.725	1.422	1.719	1.209	0.665	0.570	0.541	1.131	2.244	1.852	1.338	2.211	1.360	1.882	1.902	0.879	0.852	1.745
T_3	1.054	1.698	0.902	0.774	0.780	1.563	0.522	0.819	1.178	0.758	1.743	1.511	1.229	1.784	0.608	1.020	1.623	1.853	1.492	1.370

T_4	1.595	0.918	0.490	1.173	1.905	1.128	1.469	1.667	1.202	0.677	1.638	1.467	0.590	1.364	0.860	2.319	1.666	1.882	0.933	1.378
T_5	1.469	1.224	0.612	1.688	1.891	1.605	1.168	1.230	0.730	0.790	1.594	1.311	0.813	2.023	0.569	1.442	2.475	1.980	1.362	1.490
T_6	0.862	0.630	1.155	0.680	1.472	1.150	0.920	1.589	1.688	0.829	1.291	1.004	0.930	1.351	0.555	2.383	1.208	0.748	0.644	1.674
T_7	0.957	1.701	1.953	0.714	1.477	1.743	1.722	1.870	1.298	0.915	1.112	1.824	1.710	1.151	0.962	2.191	1.374	0.536	1.259	1.864
T_8	1.371	0.289	0.709	1.401	1.476	0.798	1.322	0.798	0.609	0.513	2.366	1.244	0.786	1.924	0.906	1.454	1.231	1.169	1.150	1.673
T_9	1.139	1.245	0.863	1.858	1.892	0.946	0.708	0.848	0.951	0.540	2.486	1.448	1.475	1.023	1.540	1.340	1.066	1.543	1.074	1.419
T_{10}	1.822	1.995	0.660	0.860	1.039	0.918	0.694	0.980	1.501	1.213	1.499	1.385	1.805	1.150	1.168	1.919	2.175	1.746	1.387	1.364
T_{11}	1.589	1.035	1.780	0.860	1.393	1.608	1.062	1.495	1.060	0.895	1.076	2.077	1.149	1.065	1.102	2.326	1.166	1.861	0.583	1.203
T_{12}	0.358	0.913	0.660	0.977	0.796	0.654	1.912	0.641	1.910	0.833	1.561	1.803	1.531	1.336	1.377	1.456	2.295	1.004	0.974	1.868
T_{13}	1.845	0.743	1.063	1.892	1.254	1.076	1.083	1.301	1.437	1.492	2.346	1.815	1.227	2.234	0.933	1.945	2.286	1.117	0.534	1.125
T_{14}	1.049	1.282	1.588	1.251	0.759	1.644	0.556	0.411	1.160	0.919	1.568	1.638	1.658	1.367	0.644	2.193	1.052	1.502	1.528	1.285
T_{15}	0.483	1.864	1.982	1.816	1.561	1.409	0.960	0.747	1.176	0.922	2.017	2.001	0.582	1.534	0.972	1.219	2.128	1.346	0.775	1.129

Table 2. Matching degree between collaborative team and task

The task costs were listed in Table 3.

	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}	G_{16}	G_{17}	G_{18}	G_{19}	$G_{20}(10^4)$
T_1	6	8	7	8	7	10	7	7	8	8	10	7	10	8	9	10	10	8	6	10
T_2	8	6	7	6	9	9	6	5	8	7	10	8	6	9	6	9	10	6	5	8
T_3	6	7	7	5	7	6	7	5	6	6	9	5	6	5	7	7	9	7	5	8
T_4	8	9	10	9	11	11	10	8	10	9	11	10	11	12	9	11	11	9	7	12
T_5	18	19	17	17	20	18	16	23	21	17	30	18	21	19	23	24	35	20	16	24
T_6	23	26	27	27	24	25	22	20	25	24	35	27	22	25	22	26	32	23	20	26
T_7	15	18	16	16	18	17	18	19	16	18	16	10	16	15	18	16	16	17	19	17
T_8	12	10	10	13	13	11	14	15	14	12	15	15	13	14	11	15	13	13	11	13
T_9	6	7	8	7	7	5	7	9	9	7	10	8	8	9	8	6	10	8	5	7
T_{10}	18	17	16	21	19	18	20	19	18	17	26	19	15	20	18	19	27	15	14	17
T_{11}	5	4	5	4	6	7	5	5	7	6	6	6	5	7	5	7	6	5	4	5
T_{12}	2	3	4	5	2	3	5	6	4	3	5	3	4	6	5	5	5	3	2	7
T_{13}	7	4	5	7	4	4	5	7	6	5	5	7	7	4	7	7	6	7	5	6
T_{14}	3	5	3	4	5	6	5	6	5	4	8	3	5	5	3	4	7	3	3	7
T_{15}	10	7	10	9	8	9	8	12	9	9	15	11	11	9	10	12	17	10	8	9

Table 3. The cost that collaborative team complete the task

Parameter configurations of the improved GA were as follows: the initial population size was 20, P_{c1} was 0.85, P_{c2} was 0.65, mutation probability was 0.9, the maximum number of iteration was 800. a_1 was 0.6 and a_2 was 0.4. Based on the data above, the procedures of the improved genetic algorithm were written by Matlab, and the optimal program was shown in Table 4.

Task Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Collaborative team	1	19	14	19	7	8	12	3	19	19	2	5	5	12	2

Table 4. Tasks - team matching program

Under this matching program, the objective optimal value is 74.10, while duration is 45.7 days and cost is 1,180,000 RMB. The solution got by GA is {1 19 4 1 7 8 12 2 19 13 2 1 5 12 2}. The fitness curve of the improved GA and traditional GA was shown in Figure 5. The optimal was got at the 458th iteration by the improved GA while got at the 622th iteration by traditional genetic algorithm. The result of the comparison displayed the advantage of the improved algorithm in finding the optimal and convergence speed, shown in Table 5.

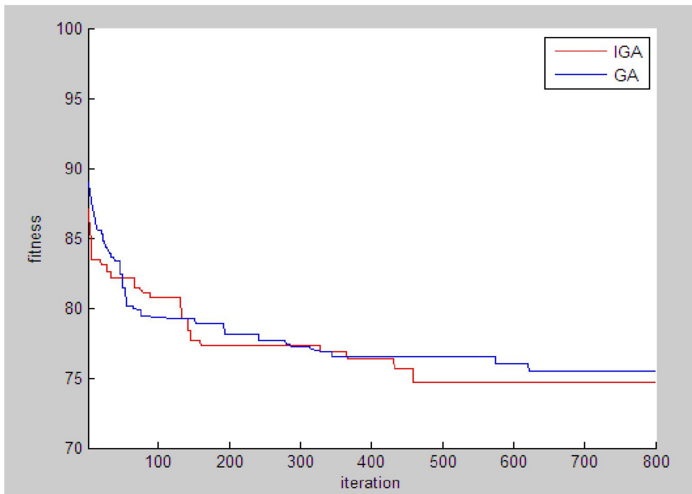

Figure 5. Fitness curves of the improved GA and GA

algorithm	fitness	run time(s)	iteration
GA	75.45	32.6	622
Improved GA	74.65	20.4	458

Table 5. Comparison of improved GA and GA

The project task allocation and schedule plan was shown in Figure 6.

6. Conclusions

In this paper, competence evaluation and scheduling model of collaborative product design are studied based on matching degree. In the competence model, Collaborative team capacity is composed of core competency, basic competency and basic resource. Variable competencies or resources have different effect on the matching degree**.** Then, the 2-tuple linguistic method is used to avoid information loss and make the evaluation result more precise. The scheduling model considering matching degree is established considering matching degree, project duration and cost. In the improved algorithm, single-coding strategy, multi-point mutation and crossover are adopted.

Although the case study, it is shown that the proposed approach can really be a useful tool to obtain the reasonable program.

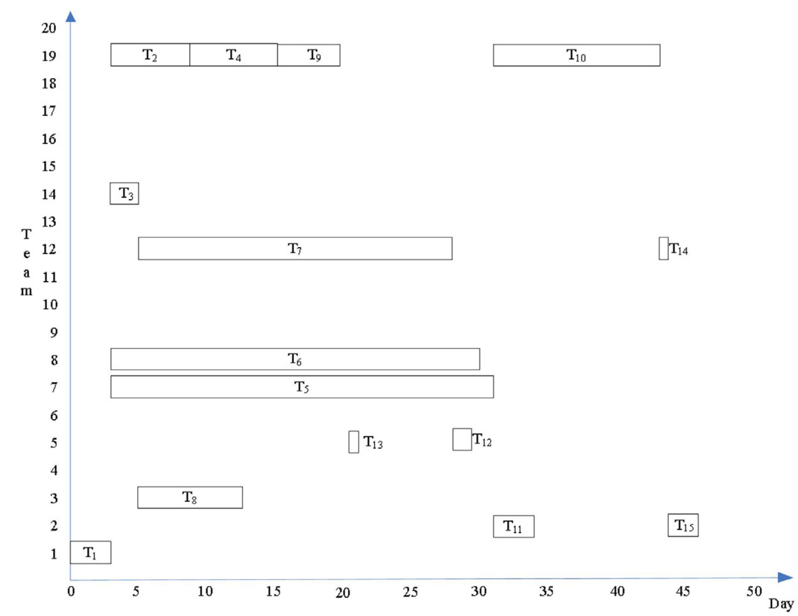

Figure 6. Project task allocation and timing chart

But there are still limitations in the approach, such as the subjectivity of evaluation and the precision of resource quantization. Furthermore, during the process of collaborative product design, there may be resource conflict and partner selection conflict. In the future, more work can be done such as encouragement and collaboration mechanism for collaborative design.

Acknowledgements

This research is supported by Beijing Natural Science Foundation (No.9174044) and the Science Foundation of Beijing Normal University, Zhuhai (No.201771010)

References

[1] Lee Y. H., Chang C. T., Wong D. S., Jang S. S. (2011). Petri-net based scheduling strategy for semiconductor manufacturing processes. *Chemical Engineering Research and Des*ign, 89 (3) 291-300.

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

[3] Bruni, M. E., Beraldi, P., Guerriero, F., Pinto, E. (2012). A heuristic approach for resource constrained project scheduling with uncertain activity durations. *Computers & Operations Research,* 38 (9) 1305-1318.

[4] Tahooneh A., K. Ziarati. (2011). Using artificial bee colony to solve stochastic resource constrained project scheduling problem. *Advances in Swarm Intelligence,* 293-302, Springer Berlin.

[5] Pang, H., Fang, Z. D., Guo, H., Zhao, Y. (2008). Research on task scheduling method of cooperative design. *Systems Engineering and Electronics*, 30 (10) 1899-1903.

[6] Yu, G. D., Yang, Y., Zhao, X., Liu, A. J. (2015). Multi-objective dynamic fuzzy scheduling and its algorithm in product collaborative design considering emergency. *Journal of Intelligent & Fuzzy Systems*, 29, 1355-1365.

[7] Frillman, S. A., Wilde, K. L., Kochert, J. F. (2010). Entry-level engineering professionals and Product Lifecycle Management: A competency model. *International Journal of Manufacturing Technology and Management*, 19 (3-4) 306-311.

[8] Wu, Y. H., Qin, X. S. (2011). Study on Resource Capability Measure and Its Optimal Deployment in Product Development Process. *Industrial Engineering Journal*, 14 (2) 80-84.

[9] Zhang, S., Li, Y. Z. (2014). Multi-agent simulation of partner selection behavior based on matching degree in collaborative

product development process. *Computer Modelling and New Technologies*, 18 (10) 25-30.

[10] Li, Y. Z., Zhang, S. (2015). Static scheduling in consideration of resource categories in collaborative product development project. *Advances in Mechanical Engineering*, 7 (10) 1-9.

[11] He, Y., Hu, D. C. (2012). A Matching Approach of Manufacturing Tasks and Services based on Ontology in Manufacturing Grid. *International Journal of Digital Content Technology and its Applications*, 6 (11) 46-53.

[12] Du, B., Li, J. F. (2012). The construction and empirical study on team meta-competence model. *Science Research Management*, 33 (11) 40-48.

[13] Cao, M., Vonderembse, M. A., Zhang, Q. Y., Ragu-Nathan, T.S. (2010). Supply chain collaboration: Conceptualisation and instrument development. *International Journal of Production Research*, 48 (22) 6613-6635.

[14] Herrera, F., Martinez, L. (2000). A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems*, 8 (6) 746-752.

[15] Herrera, F., Martinez, L. (2005). Managing non-homogeneous information in group decision making. *European Journal of Operational Research*, 166 (11) 115-132 .

[16] Wei, G. W. (2010). A method for multiple attribute group decision making based on the ET-WG and ET-OWG operators with 2-tuple linguistic information. *Expert Systems with Applications*, 37 (12) 7895-7900.

[17] Chen, S. P., Tsai, M. J. (2011). Time-cost trade-off analysis of project networks in fuzzy environments. *European Journal of Operational Research*, 212 (2) 386-397.