Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 7194258, 19 pages
https://doi.org/10.1155/2019/7194258

Research Article

Cloud Manufacturing Service Composition Optimization with Improved Genetic Algorithm

Yongxiang Li,[1,2] Xifan Yao ,[2] and Min Liu[2]

[1] *School of Mechanical Engineering, Guizhou University of Engineering Science, Bijie 551700, Guizhou, China*
[2] *School of Mechanical & Automotive Engineering, South China University of Technology, Guangzhou 510640, Guangdong, China*

Correspondence should be addressed to Xifan Yao; mexfyao@scut.edu.cn

Received 2 November 2018; Revised 2 May 2019; Accepted 11 June 2019; Published 3 July 2019

Academic Editor: Thomas Hanne

Aiming at the problems in which there exists collocation between services and manufacturing tasks, multiobjective cloud manufacturing service composition optimization seldom considers the synergy degree of composite cloud services and the complexity of service composition, so a novel service composition optimization approach, called improved genetic algorithm based on entropy (IGABE), is put forward. First, the mathematical expressions of service collocation degree, composition synergy degree, composition entropy, and their related influence factors of the service composition are analyzed, and their definitions and calculation methods are given. Then, a multiobjective cloud manufacturing service composition optimization mathematical model is established. Moreover, crossover and mutation operators are improved by introducing normal cloud model theory and piecewise function, and improved roulette selection method is used to perform the selection operation. And the fitness function of the proposed IGABE is designed by combining Euclidean deviation with angular deviation. Finally, the manufacturing task of a wheeled cleaning robot is exemplified to verify the correctness of the proposed multiobjective optimization model for cloud manufacturing service composition and the effectiveness of the proposed algorithm, compared with Standard Genetic Algorithm (SGA), Hybrid Genetic Algorithm (HGA), and Cloud-entropy Enhanced Genetic Algorithm (CEGA). The studied results show that IGABE converges faster than SGA and HGA and can analyze and reflect the content difference expressed by the objective functions of service composition scheme and its approximation degree to the corresponding dimensions of the ideal point vector more comprehensively than CEGA. As such, the optimal service composition obtained by IGABE algorithm can better meet the complex needs of users.

1. Introduction

Cloud manufacturing is a new service-oriented manufacturing mode [1]. Through virtualization and servitization of various manufacturing resources and manufacturing capabilities, it provides users with all kinds of manufacturing resources that can be accessed at any time and paid for in the form of cloud services. To meet the complex manufacturing needs of customers, fine-grained simple cloud services are combined into coarse-grained complex cloud services through service composition. Complex cloud services are executed in distributed, heterogeneous, and autonomous environments to fulfill manufacturing tasks, which are highly uncertain and dynamic. Choosing the best cloud service composition to perform manufacturing tasks successfully is one of the key issues to realize resource increment in cloud manufacturing environment [2]. Cloud manufacturing service composition optimization is a typical NP-hard problem, which has the characteristics of multiextremum, nonlinearity, multiobjective, and uncertainty [3]. It has become a hot spot in the field of cloud manufacturing research; many scholars have put efforts on such a NP-hard problem. They have studied the service composition modeling and optimization algorithm to solve the optimization problem of cost, time, profit, and other functional service quality parameters [4–40], but seldom consider the impact of nonfunctional service quality parameters in service composition. The realization of cloud manufacturing is a collaborative process that requires multiple distributed manufacturing resources to participate together. Every cloud manufacturing service execution

agent is in a certain social relationship, not an idealized "rigid body". In the manufacturing process, different data exchange, information transmission, and material transportation are carried out between the cloud manufacturing services. They constrain, collaborate, and compete in the whole manufacturing life cycle. The relationship between cloud manufacturing services and manufacturing tasks and the relationship between every two cloud manufacturing services directly affect the efficiency of service composition in executing manufacturing tasks. In the cloud manufacturing environment, customized product manufacturing to meet the individual needs is a normal situation, which often requires the collaborative participation of customers and cloud manufacturing service providers. Cloud manufacturing service composition needs to meet the requirements of traditional product delivery time and manufacturing cost. In addition, the collocation degree between manufacturing tasks and cloud services, the synergy degree of cloud manufacturing services in the manufacturing process, and the complexity of the state changes of manufacturing resources have a significant impact on the completion of customized product manufacturing tasks. Therefore, it is necessary to optimize the cloud manufacturing service composition by taking service collocation degree, composition synergy degree, and service composition complexity as optimization factors.

The basic rule of service composition in this study is that all the manufacturing tasks generated according to the decomposition of customer manufacturing needs must be assigned to one or more cloud services to be performed. The optimal cloud manufacturing service composition scheme should be as close as possible to the ideal value of the optimization objectives such as service collocation degree, composition synergy degree, and service composition entropy. This study focuses on the improved genetic algorithm that is proposed for cloud manufacturing service composition optimization. The main contributions of this paper are as follows: (1) a novel improved genetic algorithm is proposed for cloud manufacturing service composition optimization; (2) mathematical models of multiple influence factors in cloud manufacturing service composition are set up; and (3) an example is given for verifying the effectiveness of the proposed improved genetic algorithm.

The remainder of this paper is organized as follows: Section 2 gives a comprehensive analysis of the latest researches in cloud manufacturing service composition optimization; Section 3 gives the definitions of service collocation degree, composition synergy degree, and composition entropy, as well as the corresponding calculation methods; Section 4 proposes IGABE algorithm; Section 5 analyzes and verifies the proposed algorithm through application example; and Section 6 summarizes the whole paper.

2. Literature Review

To find relevant literature on cloud manufacturing service composition optimization, we have summarized the methods of previous research and carefully selected the articles that are most relevant to our research. As such, existing such studies can be categorized into three groups.

The first group studied on classification, modeling, and description of manufacturing resource and capabilities in cloud manufacturing. T. Chen and Y. Wang [4] proposed a classified artificial neural network ensemble method to predict the time required for a simulated manufacturing cloud task, in which K-means was used to classify the simulation manufacturing cloud tasks, and for every task category, an artificial neural network was constructed to predict the time required for the manufacturing cloud task. Q. N. Meng and X. Xu [5] proposed a pricing model to describe the pricing mechanism of cloud manufacturing and gave a support vector regression method based on ant colony optimization algorithm to predict the price change of cloud services. Y. Zhang et al. [6] proposed a distributed decision-making mechanism called analytical target cascade method to optimize cloud manufacturing service configuration. Y. Liu et al. [7] studied the impact of task scheduling methods based on different workloads on system performance, such as total completion time and service utilization rate, and proposed a multitask scheduling model for cloud manufacturing, which combined the modeling of task workload, the service efficiency coefficient, and the number of services. F. Li et al. [8] proposed a two-level multitask scheduling model to schedule all subtasks decomposed by multiple heterogeneous tasks in a cloud manufacturing environment for maximum benefit. S. Huang et al. [9] divided the on-demand supply model of manufacturing cloud services into four submodels: on-demand supply, on-demand combination, on-demand design, and on-demand research, and a two-dimensional space-time optimization model describing the optimization mechanism among personalization, cost, and response time was established. N. Liu et al. [10] proposed a multigranularity resource virtualization sharing strategy to bridge the gap between complex manufacturing tasks and underlying resources based on the analysis of three significant factors affecting the gradual decomposition of complex manufacturing tasks: workflow, activities, and resources, a resource aggregation function was constructed, and a resource clustering algorithm that mapped physical resources to virtualized resources was given. P. Yongdong [11] designed a mobile cloud service model of multiobjective wireless optical network considering network delay and energy consumption, in which the multiobjective droplet algorithm was used to find the second-best solution, and NSGA II density design method was used to improve the population diversity of multiobjective droplet algorithm. Y. X. Li and X. F. Yao [12] proposed an intelligent service composition method based on extended process calculus, which constructed cloud manufacturing service description model, interactive scenario model, and composition process model and extended process calculus semantics to describe service quality information. Y. Hu et al. [13] proposed a hybrid algorithm by fusing genetic algorithm, simulated annealing, and fuzzy C-means clustering algorithm to solve the fuzzy classification problem of cloud manufacturing resources.

The second group studied on task decomposition, resource allocation, and service matching. J. Thekinen and J. H. Panchal [14] formulated resource allocation in cloud environment as a bipartite matching problem, and four

bipartite matching mechanisms including Deferred Acceptance (DA), Top Trading Cycle (TTC), Munkres, and First Come First Service (FCFS) were divided from individual rationality, stability, strategy proofness, consistency, monotonicity, and Pareto efficiency. F. Tao et al. [15] designed a hypernetwork-based cloud manufacturing service supply and demand matching simulator, which could compare the service matching results and scheduling algorithm performance. S. Răileanu et al. [16] put forward a design method of high availability production management system based on cloud under the condition of combining energy consumption with product scheduling and resource allocation. A. Brant and M. M. Sundaram [17] studied the micrometal additive manufacturing by electrochemical deposition under cloud condition. The horizontal deposition parameters were optimized based on the deposition resolution, and the manufacturing data were saved in the cloud for users to use on demand. M. R. Namjoo and A. Keramati [18] used resource-based theory and dematel method to solve the problem of causality between dimensionality and attribute of composite service elasticity in cloud manufacturing. G. Zhang et al. [19] decomposed the cloud manufacturing service allocation problem into enhanced Lagrange coordination model, which was solved by loosely coupled and distributed method. M. Zhang et al. [20] studied aggregated resource service allocation with capacity constraints and proposed an improved genetic algorithm to search for optimal matching results in order to achieve the minimum total cost and time. W. Zhang et al. [21] proposed an extended teaching optimization algorithm for parallel optimization of distributed manufacturing resource allocation. W. Y. Zhang et al. [22] proposed a new personalized recommendation method for manufacturing services based on hyperlink inductive topic search algorithm and Bayesian method. B. Sheng et al. [23] studied the matching process diversity, heterogeneity, and multiconstraints between the intelligent matching engine and cloud manufacturing service and established an intelligent searching engine of cloud manufacturing service based on ontology language for service.

The third group studied on service composition, evaluation, optimization algorithm, and cloud service prototype system. Taking sheet metal processing in cloud environment as an example, the minimum cost and maximum profit as the optimization objectives, and considering multiple related subfactors, P. Helo and Y. Hao [24] found a new optimization method for cloud-based production planning and control, proposed a cloud-based dynamic optimization model, and developed an optimization algorithm-based scheduling prototype system. V. B. Souza et al. [25] studied the distributed service placement strategy in the mixed fog-cloud scenarios, proposed a concurrent service execution scheme, and improved the service response time. M. Lang et al. [26] identified service function, legality, contract, geographic location, and flexibility as the highest quality-of-service criteria for cloud service selection through Delphi research on cloud service quality attributes. H. Yan et al. [27] studied the related technologies of cloud robots in intelligent manufacturing environment, including the adaptive adjustment mechanism of cloud robot network service quality, the computing load distribution mechanism of cloud robots, and the group learning based on cloud platform. J. Zhou et al. [28] improved the artificial bee colony algorithm by introducing food source disturbance synergy mechanism, diversity maintenance strategy, and new computing resource allocation scheme to deal with multiobjective service composition and optimal selection problems in complex cloud manufacturing. C. Yang et al. [29] proposed a dynamic service selection method across multiple manufacturing clouds, which utilized the real-time perception ability of the Internet of Things to service execution, the knowledge extraction ability of large data to manufacturing cloud services, and the event-driven dynamic service selection optimization to deal with the interference from users and service markets and continuously adjust service selection. Y. Lu and X. Xu [30] studied knowledge-based service composition and adaptive resource planning in cloud manufacturing environment and proposed a service composition method based on restrictive rule set and resource availability information in manufacturing cloud to rapidly allocate resources for service requests. J. Zhou and X. Yao [31] proposed a multipopulation parallel self-adaptive differential artificial bee colony algorithm for NP-hard optimization of composite cloud manufacturing services, in which every parallel subpopulation evolved according to different mutation strategies borrowed from differential evolution, and generated perturbed food sources for foraging, and the control parameters of every mutation strategy were adjusted independently. F. Chen et al. [32] proposed a QoS-aware multiobjective optimization algorithm for Web service composition, which took QoS performance as the optimization objective and solved the multiobjective optimization model of QoS-aware Web service composition. X. Huang et al. [33] combined the genetic algorithm with the particle swarm optimization and proposed a hybrid genetic particle swarm optimization algorithm based on teaching and learning. A learning mechanism was introduced into the genetic algorithm, which enabled the descendants of the genetic algorithm to learn the characteristics of the elite chromosome from the dual memory learning. The algorithm searched for solutions and exchanges information in the two subpopulations of the genetic algorithm module and the particle swarm optimization module. Y. Li et al. [34] studied service matching degree, composition harmony degree, and cloud entropy and proposed a cloud-entropy enhanced genetic algorithm for solving the multiobjective optimization problem of cloud manufacturing service composition. C. Li et al. [35] studied cloud manufacturing service composition and optimization selection oriented to autonomy and proposed a fuzzy soft decision method based on volatility analysis. Y. Que et al. [36] proposed a new cloud-oriented manufacturer-to-users model to solve the problem of optimal selection of manufacturing service composition, established a comprehensive mathematical evaluation model with four key service quality perception indicators (namely time, reliability, cost, and capability), and used information entropy immune genetic algorithm to solve the model. Y. Wang et al. [37] studied the multiobjective optimization problem of cloud manufacturing service composition quality based on emergency task perception and proposed a

two-stage (namely, synthesis and reorganization) optimization algorithm. S. Zhang et al. [38] studied a fuzzy QoS-aware manufacturing service composition method based on extended pollination algorithm. B. Xu and Y. Cai [39] proposed an efficient global optimization algorithm based on multidata for automotive body design. The general computing technology based on graphics processing unit and hybrid parallel computing method was used to improve the efficiency of the solution. F. Zhao et al. [40] proposed a called SPA2 algorithm based on self-adaptive selection evolutionary operator to solve multiobjective optimization problems. In the evolutionary process, the algorithm adaptively selected simulated binary crossover operator, polynomial mutation operator, and differential evolution operator according to the operator contributions.

However, the above researches rarely consider the impact of nonfunctional service quality parameters in service composition and the social relation in cloud manufacturing.

3. Influence Factors Modeling

3.1. Problem Description. Suppose that a complex manufacturing task $\boldsymbol{J}$ can be decomposed into m manufacturing subtasks, namely, $\boldsymbol{J}=\{J_1, J_2, \ldots, J_m\}$. In cloud resource pool, several appropriate cloud services are combined to collaboratively complete the manufacturing tasks. Each manufacturing subtask corresponds to a cloud service set, and is expressed as $\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots, \boldsymbol{S}_m$. The numbers of cloud services contained in the service sets are $b_1, b_2, \ldots, b_m$, respectively. The total number of cloud services contained in m sets is $N = \sum_{i=1}^{m} b_i$. The i-th cloud service set is represented as $\boldsymbol{S}_i=\{S_{i,1}, S_{i,2}, \ldots, S_{i,b_i}\}$. The m manufacturing tasks are allocated to the most suitable services in N cloud services for collaborative completion, and the allocation relationship is shown in Figure 1 [41].

3.2. Service Collocation Degree Modeling. Service collocation degree (CD) is a quantitative measure of suitability between cloud manufacturing services and allocated manufacturing tasks. Many factors can affect the collocation degree of cloud manufacturing services for manufacturing tasks, such as the total number of similar manufacturing tasks performed by the cloud manufacturing service, the cloud manufacturing service activity level for a period of time, the distance between the manufacturing resources mapped by the cloud manufacturing service and the objects served, equipment status, comprehensive manufacturing capacity, service quality, the idle degree of manufacturing resources, and so on. According to the characteristics of influence factors, the factors affecting service collocation can be summarized as technical factor, hunger factor, and distance factor.

Technical factor (TF) is used to describe the technical capability of N cloud services to complete the manufacturing tasks. It is expressed by the technical matrix $\boldsymbol{TF}=(TF_{ij})_{N\times m}$, where TF_{ij} ($1\le i\le N$, $1\le j\le m$) represents the measurement of technical capability of the i-th cloud service to complete the j-th manufacturing task. The value of TF_{ij} is described by a rank vector $[0, 0.1, \ldots, 1]$, which is mainly evaluated by referring to the cumulative number of similar manufacturing tasks completed over a period of time, service execution rate, quality of service, equipment performance, and so on.

Hunger factor (HF) is used to describe the desire level of the cloud manufacturing service to undertake and complete the j-th manufacturing task. It is represented as HF_{ij}. The vacancy rate of manufacturing resources refers to the ratio of the number of available manufacturing resources that are not undertaken manufacturing tasks to the total number of manufacturing resources. The hunger degree is closely related to the vacancy rate of manufacturing resources mapped by cloud manufacturing services. The higher the manufacturing resource vacancy rate is, the hungrier the demand for manufacturing tasks is, and the greater the hunger factor of cloud manufacturing services. The value of hunger factor HF_{ij} can be calculated by manufacturing resource vacancy rate. Therefore, the hunger factor matrix can be constructed, namely, $\boldsymbol{HF}=(HF_{ij})_{N\times m}$, where HF_{ij} ($1\le i\le N$, $1\le j\le m$), which represents the hunger degree of the i-th cloud service to complete the j-th manufacturing task. The value range of $HF_{ij} \in [0, 1]$, i.e., $0 \le HF_{ij} \le 1$.

Distance factor (DF) is used to describe the impact of relative distance between customer and manufacturing resource mapped by cloud manufacturing service on service collocation degree. It is represented as DF_{ij}. For example, if cloud manufacturing resources are not in the customer's city, DF_{ij}=0.4; if cloud manufacturing resource is in the customer's city, DF_{ij}=0.8; if cloud manufacturing resource is in the customer's enterprise, DF_{ij}=1. The larger the relative distance between the customer and the manufacturing resource is, the smaller the distance factor is. Otherwise, the larger the distance factor is.

Service collocation degree matrix can be calculated as follows according to technical factor TF_{ij}, hunger factor HF_{ij} and distance factor DF_{ij}.

$$\boldsymbol{CD} = \begin{pmatrix} CD_{11} & CD_{12} & \cdots & CD_{1,m-1} & CD_{1m} \\ CD_{21} & \cdots & \cdots & \cdots & CD_{2,m} \\ \cdots & \cdots & CD_{ij} & \cdots & \cdots \\ CD_{N-1,1} & \cdots & \cdots & \cdots & CD_{N-1,m} \\ CD_{N,1} & CD_{N,2} & \cdots & CD_{N,m-1} & CD_{N,m} \end{pmatrix}, \quad (1)$$

where $CD_{ij}=\alpha\times TF_{ij} +\beta\times HF_{ij}+\gamma\times DF_{ij}$; CD_{ij} ($1\le i\le N$, $1\le j\le m$), which represents service collocation degree of the i-th cloud manufacturing service to complete the j-th manufacturing task; α, β, and γ are the weight coefficients of the corresponding factors, and $\alpha+\beta+\gamma=1$.

3.3. Composition Synergy Degree Modeling. Composition synergy degree (SD) is a quantitative measure of collaboration level of multiple cloud manufacturing services that are combined to complete a complex manufacturing task. The high synergy degree between two cloud manufacturing services is conducive to handy information interaction and smooth material transportation. It helps to complete the manufacturing task smoothly and shorten the execution time. On the contrary, the low synergy degree hinders the

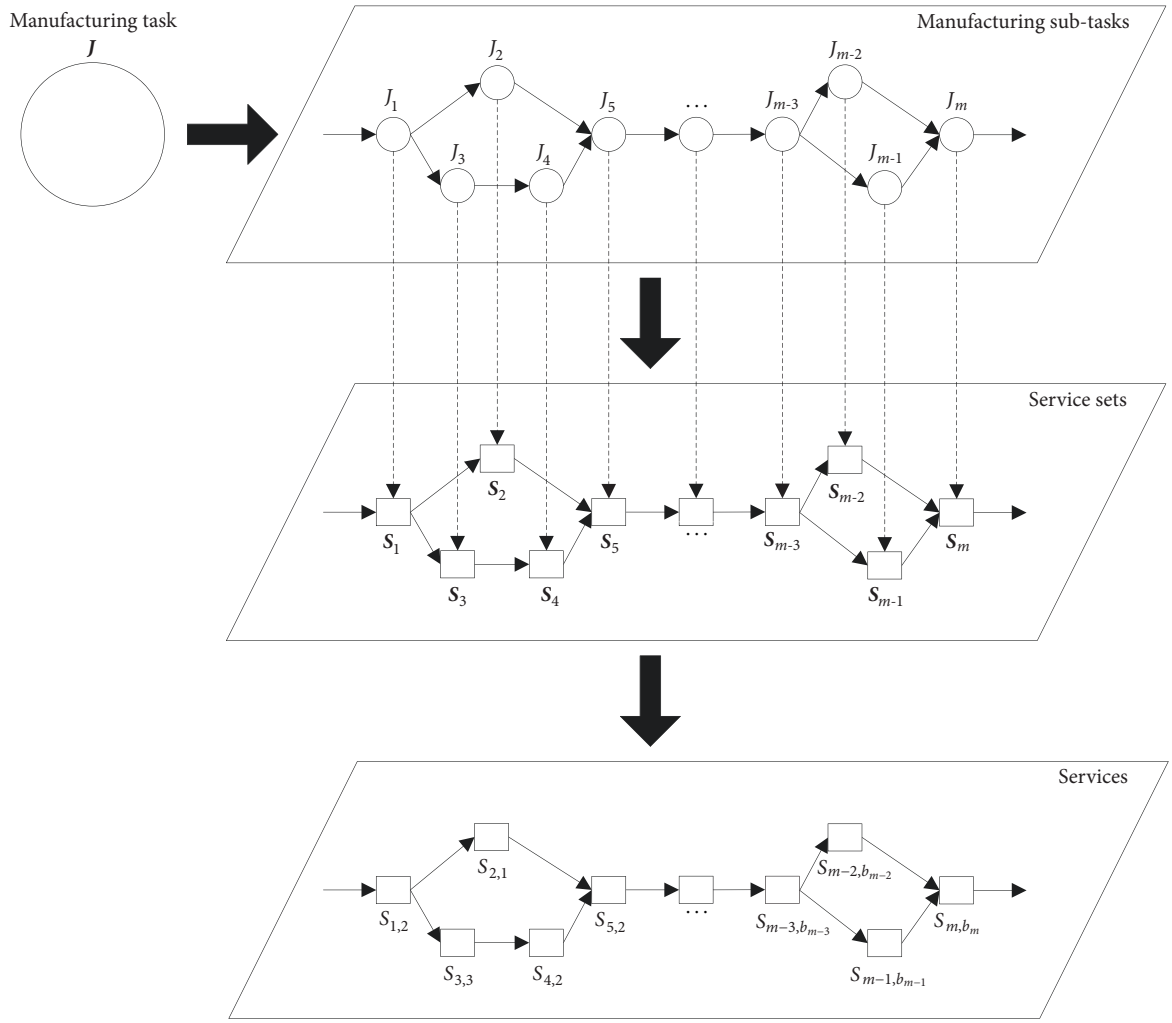


Figure 1: Mapping relationship between cloud services and manufacturing tasks [41].

smooth information exchange and material delivery between two cloud manufacturing services, resulting in the delay of product delivery and the increase of manufacturing cost. The composition synergy degree can be measured by the time that cloud manufacturing services complete the manufacturing tasks. For example, the composition synergy degree between the cloud manufacturing service S_i for the manufacturing task J_i and the service S_j for the task J_i is calculated as follows:

$$SD_{ij} = \frac{T_i + T_j}{T_{ij}}, \tag{2}$$

where T_{ij} is the total time taken by two cloud manufacturing services S_i and S_j to collaboratively complete two manufacturing tasks J_i and J_j; T_i is the time taken by the cloud manufacturing service S_i for independently completing the manufacturing task J_i; and T_j is the time taken by the service S_j for the task J_j.

The relationship between every two manufacturing subtasks in a complex manufacturing task has a great influence on the total execution time. The total execution time of complex manufacturing task consisting of several independent subtasks is relatively easy. If the tasks J_i and J_j are independent and in parallel, the total execution time is $T_{ij} = \max[T_i, T_j]$. If the tasks J_i and J_j are independent and in sequence, the total execution time is $T_{ij} = T_i + T_j$. The computation of the total execution time of a complex manufacturing task including interactive coupling relation is more difficult. If T_i is the time taken by the cloud manufacturing service S_i for independently completing the manufacturing task J_i, and T_j is the time taken by the service S_j for the task J_j, when there is an interactive coupling relationship between them, the total execution time of the complex manufacturing task can be computed as follows [42]:

$$T_{ij} = T_i + T_j + 2\zeta_{ij}\sqrt{T_i \cdot T_j}, \tag{3}$$

where ζ_{ij} is the interactive coupling coefficient between the manufacturing tasks J_i and J_j, whose value range is $[-1, 1]$. ζ_{ij}

is influenced by the frequency and level of previous cooperation, service interaction, material transportation, and so on. The higher the frequency and level of previous cooperation are, the better the smoothness of service interaction and material transportation is, and the smaller the coefficient ζ_{ij} is. Otherwise, the bigger the coefficient ζ_{ij} is.

Based on the above analysis, cloud manufacturing service composition synergy degree matrix can be established as follows:

$$\boldsymbol{SD} = \begin{pmatrix} SD_{11} & SD_{12} & \cdots & SD_{1,N-1} & SD_{1,N} \\ SD_{21} & \cdots & \cdots & \cdots & SD_{2,N} \\ \cdots & \cdots & SD_{ij} & \cdots & \cdots \\ SD_{N-1,1} & \cdots & \cdots & \cdots & SD_{N-1,N} \\ SD_{N,1} & SD_{N,2} & \cdots & SD_{N,N-1} & SD_{N,N} \end{pmatrix}, \quad (4)$$

where SD_{ij} represents the composition synergy degree between the cloud manufacturing services S_i and S_j.

3.4. Composition Entropy Modeling. Composition Entropy (CE) is the quantitative measure of the complexity and orderliness of cloud manufacturing service composition. The composition entropy of an ordered and simple cloud manufacturing service composition is smaller than that of a disordered and complex cloud manufacturing service composition. The simple and orderly cloud manufacturing service composition has greater certainty in the successful completion of manufacturing tasks. According to the manufacturing system entropy calculation model established in [43], the cloud manufacturing service composition entropy is computed as follows [43]:

$$CE_i = -\sum_{j=1}^{Q_i} \frac{ST_{ij}}{TT_i} \ln \frac{ST_{ij}}{TT_i}, \quad (5)$$

$$CE_{\text{sum}} = \sum_{i=1}^{N} CE_i, \quad (6)$$

where ST_{ij} is the time when the i-th cloud manufacturing service is in the j-th state. TT_i is the total time for the i-th cloud manufacturing service to complete the corresponding manufacturing task. N is the total number of cloud manufacturing services in the service composition scheme. Q_i is the total state number of the i-th cloud manufacturing service. CE_i is the composition entropy of the i-th cloud manufacturing service. CE_{sum} is the total composition entropy of the service composition scheme, which can be used to measure the complexity of service composition scheme. In cloud manufacturing activities, the starting work node time of the first working procedure of a part is regarded as the starting time of the part processing, and the finishing work node time of the last working procedure of a part is regarded as the ending time of the part processing. The starting work node time of the first operation of a service is regarded as the starting time of the service execution, and the finishing work node time of the last operation of a service is regarded as the ending time of the service execution. The composition entropy of a cloud manufacturing service is calculated by (5), and the total composition entropy of the cloud manufacturing service composition scheme is computed by (6), so as to evaluate the complexity of the service composition scheme. The bigger the composition entropy is, the more complex and unreliable the cloud manufacturing service composition scheme is, and the lower the probability of the service composition successfully completing the manufacturing task is.

3.5. Multiobjective Service Composition Optimization Modeling. The purpose of cloud manufacturing service composition is to select and compose the most suitable cloud services to complete all the manufacturing tasks. The service composition scheme should meet the constraints of manufacturing time and cost and make the composition synergy degree of the cloud manufacturing services be the highest and the composition entropy be the smallest. The service composition optimization model is as follows:

$$\max \quad Y_1 = \sum_{i=1}^{N} \sum_{j=1}^{m} \left(CD_{ij} \cdot \xi_{ij} \right), \quad (7)$$

$$\max \quad Y_2 = \sum_{j=1}^{m} \sum_{k=1}^{j} \sum_{p=1}^{N} \sum_{q=1}^{N} \left(SD_{pq} \cdot \xi_{pj} \cdot \xi_{qk} \right), \quad (8)$$

$$\max \quad Y_3 = -\sum_{i=1}^{N} CE_i, \quad (9)$$

$$\text{s.t.} \quad \max\left(T_1, T_2, T_3, \ldots, T_{m-1}, T_m\right) \le ET_0, \quad (10)$$

$$\sum_{j=1}^{m} EC_j = \sum_{j=1}^{m} \sum_{i=1}^{N} \left(T_i \cdot w_i \cdot \xi_{ij} \right) \le EC_0, \quad (11)$$

$$\sum_{i=1}^{N} \xi_{ij} \ge 1, \quad (12)$$

where ξ_{ij} is a switching variable:

$$\xi_{ij} = \begin{cases} 1, & \text{if the } j\text{-th manufaturing task is allocated to the } i\text{-th cloud service;} \\ 0, & \text{otherwise.} \end{cases}, \quad (13)$$

ξ_{pj} and ξ_{qk} are the switch variables of the j-th and k-th manufacturing tasks, respectively, whose values can be obtained by (13); w_i represents the service price per unit time; and EC_j represents the execution cost of the j-th manufacturing task.

Equations (7)-(9) are objective functions. Equation (7) calculates the maximum value of the total service collocation degree. Equation (8) calculates the maximum value of the total composition synergy degree. Equation (9) calculates the maximum value of negative total composition entropy of cloud manufacturing service composition scheme. Equations (10)-(12) are constraints. Equation (10) indicates that the maximum execution time of manufacturing tasks must not exceed the threshold time ET_0. Equation (11) indicates that the total execution cost of manufacturing tasks must not exceed the threshold cost EC_0. Equation (12) indicates that each manufacturing task must be allocated to one or more cloud services to be performed.

4. Improved Genetic Algorithm

Cloud manufacturing service composition is a multiobjective optimization NP-hard problem. Traditional genetic algorithm is prone to premature and easy to fall into local extremum, so its optimization quality needs to be improved. In the following study, the normal cloud model [44] and piecewise function are used to improve the selection operator, crossover operator, and mutation operator of the traditional genetic algorithm, and an improved genetic algorithm based on entropy (IGABE) is designed to solve the multiobjective optimization model of cloud manufacturing service composition. The calculation process is shown in Figure 2.

Normal cloud model is a random number set which follows normal distribution law and has stable tendency. It is described by four main parameters: expected value *ExpVal*, cloud entropy *CloEnt*, standard deviation *StaDev*, and hyperentropy *HypEnt*. The cloud particles in the normal cloud model are generated randomly and their distribution tends to be stable, so IGABE algorithm improves the mutation and crossover operators according to the special attributes of cloud particles in the normal cloud model. At the beginning of the IGABE algorithm, bigger mutation probability and crossover probability are used to improve the emergence rate of the superior individuals; at the end of the IGABE algorithm, smaller mutation probability and crossover probability are used to keep the superior individuals in the population as much as possible to accelerate the global convergence rate.

The formulas for mutation probability and crossover probability are given as follows [44]:

Crossover probability P_c is

$$P_c = \begin{cases} \eta_1 e^{-(\Phi - ExpVal)^2/2(StaDev)^2}, & \Phi \geq \overline{\Phi}; \\ \eta_2 \left(\dfrac{\Phi_{\max} - \Phi}{\Phi_{\max} - \Phi_{\min}} P_{\mathrm{cmax}} + \dfrac{\Phi - \Phi_{\min}}{\Phi_{\max} - \Phi_{\min}} P_{\mathrm{cmin}} \right), & \Phi < \overline{\Phi}. \end{cases}, \quad (14)$$

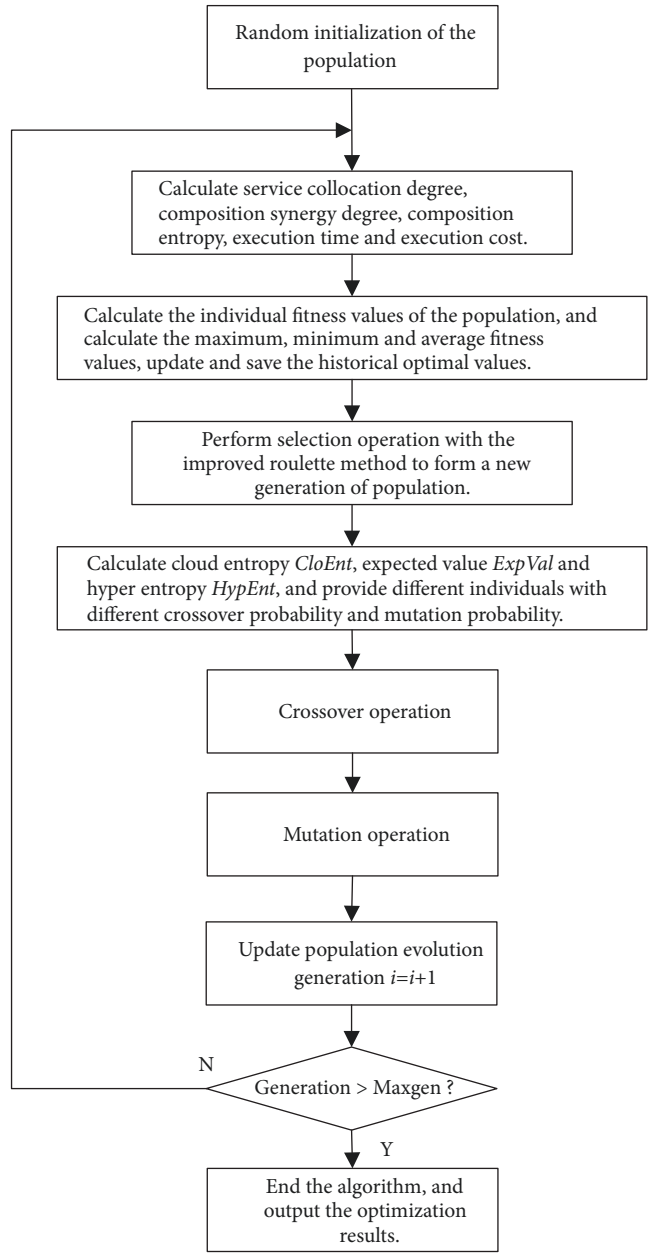


Figure 2: IGABE flowchart.

where $StaDev = \mathrm{rand}(n) \times HypEnt + CloEnt$; $CloEnt = (\Phi_{\max} - \overline{\Phi})/\tau_1$; τ_1 is the cloud entropy control coefficient in crossover operator; $ExpVal = \overline{\Phi}$, where $\overline{\Phi}$ is the average fitness value of the population; and $HypEnt = CloEnt/\tau_2$, where τ_2 is the hyperentropy control coefficient in crossover operator.

Mutation probability P_m is

$$P_m = \begin{cases} \eta_3 e^{-(\Phi' - ExpVal)^2/2(StaDev)^2}, & \Phi' \geq \overline{\Phi}; \\ \eta_4 \left(\dfrac{\Phi_{\max} - \Phi'}{\Phi_{\max} - \Phi_{\min}} P_{\mathrm{mmax}} + \dfrac{\Phi' - \Phi_{\min}}{\Phi_{\max} - \Phi_{\min}} P_{\mathrm{mmin}} \right), & \Phi' < \overline{\Phi}. \end{cases}, \quad (15)$$

where *StaDev*= rand(n) × *HypEnt* + *CloEnt*; *ExpVal*= $\overline{\Phi}$; *CloEnt*= $(\Phi_{\max} - \overline{\Phi})/\tau_3$, where τ_3 is the cloud entropy control coefficient in mutation operator; and *HypEnt*= *CloEnt*$/\tau_4$, where τ_4 is the hyperentropy control coefficient in mutation operator.

In (14) and (15), P_{cmax} is the maximum crossover probability of the population; P_{cmin} is the minimum crossover probability; P_{mmax} is the maximum mutation probability; P_{mmin} is the minimum mutation probability; Φ' is the fitness value of the mutation individual; $\Phi_{\max}$ is the maximum fitness value of the population; $\Phi_{\min}$ is the minimum fitness value of the population; $\overline{\Phi}$ is the average fitness value of the population; and Φ is the bigger one between the fitness values of two crossover individuals. $\eta_1, \ldots, \eta_4 \in [0, 1]$. Given $\eta_1 = \eta_3 = 0.4$, $\eta_2 = \eta_4 = 0.8$. From the analysis of (14) and (15), both crossover probability P_c and mutation probability P_m have bigger values at the beginning of the algorithm, and their values gradually decrease with the increase of iterations. On the one hand, IGABE algorithm improves the chance of low fitness individuals participating in mutation and crossover activities, so that the individuals in the population are diverse and representative. On the other hand, IGABE algorithm protects individuals with high fitness values, and the superior individuals are given smaller mutation probability and crossover probability at the later stage of the algorithm.

The normal cloud model is mainly controlled by several main parameters such as *ExpVal*, *CloEnt*, *StaEnt,* and *HypEnt*, and the standard deviation *StaEnt* is mainly influenced by cloud entropy *CloEnt* and hyperentropy *HypEnt*. The perpendicularity of the normal cloud model is controlled by *CloEnt*. The average value of the normal cloud model is reflected in *ExpVal.* The cloud particles in the model fluctuate up and down around *ExpVal*. The fluctuating state of cloud particles is a manifestation of the discreteness. The discrete degree of cloud particles is mainly determined by *HypEnt*. The randomness of the normal cloud model increases with the increase of *HypEnt*, and its stability increases with the decrease of *HypEnt*. τ_1 is the cloud entropy control coefficient in crossover operator; τ_2 is the hyperentropy control coefficient in crossover operator; τ_3 is the cloud entropy control coefficient in mutation operator; and τ_4 is the hyperentropy control coefficient in mutation operator. $\tau_1, \ldots, \tau_4 \in [0.1, 10]$. Set $\tau_1 = \tau_3 = 0.8$, and $\tau_2 = \tau_4 = 0.4$. τ_1 and τ_3 affect the perpendicularity of the normal cloud model. The larger τ_1 and τ_3, the larger the perpendicularity and the smaller the search range in crossover and mutation operations; the smaller τ_1 and τ_3, the smaller the perpendicularity and the larger the search range in crossover and mutation operations. τ_2 and τ_4 affect the randomness of the normal cloud model. The larger τ_2 and τ_4, the smaller the randomness of the normal cloud model; the smaller τ_2 and τ_4, the larger the randomness of the normal cloud model. With the help of normal cloud model, IGABE algorithm improves the randomness of the initial stage and restrains the prematurity of the algorithm to facilitate the formation of a more comprehensive solution space. It focuses on the protection of outstanding individuals in the later stage of the algorithm.

The implementation steps of the IGABE algorithm are as follows.

4.1. Encoding Method. Assume that there are b_i candidate cloud manufacturing services that can fulfill the i-th manufacturing task, and the b_i candidate cloud manufacturing services are queued according to the cloud service set number. IGABE algorithm chooses binary encoding method. A gene segment represents a manufacturing subtask. A gene in the gene segment represents a candidate cloud manufacturing service that can complete the manufacturing task. m gene segments are combined into a chromosome. In IGABE encoding, a gene with a value of 1 represents that the manufacturing task mapped by the gene segment is performed by the cloud service mapped by the gene; a gene with a value of 0 represents that the manufacturing task mapped by the gene segment is not performed by the cloud service mapped by the gene. Gene decoding method can be obtained by gene encoding method, since they are mutually inverse processes.

The encoding scheme shown in Figure 3 can be obtained from the above encoding method. The scheme in the graph represents that there are m manufacturing subtasks, each of which corresponds to a set including b_i candidate cloud manufacturing services. The manufacturing task J_1 is accomplished by the second cloud service in the cloud service set S_1. Its corresponding gene value is 1, and the others in the same gene segment are 0. Similarly, the manufacturing task J_m is completed by the first cloud service in the cloud service set S_m.

4.2. Fitness Function Design. The multiple objectives of cloud manufacturing service composition interact with each other and are difficult to be solved directly by ordinary mathematical methods. Cloud service users often have clear objectives and expectations for cloud manufacturing service composition. From the mathematical point of view, it is easy to solve the single objective optimization problem. The purpose of cloud manufacturing service composition can be expressed as the solution of cloud manufacturing service composition that is as close as possible to the cloud service user's expectation under the condition of limited time and so on. The cloud service user expectation is defined as the ideal point of the objective function of cloud manufacturing service composition.

IGABE algorithm uses the ideal point method to design the fitness function. There are two main methods to determine the ideal point. One is to calculate the optimal value of every single objective function through the single objective optimization algorithm to form the ideal value. Another is to specify the ideal value by cloud service user. The standard to judge the effect of service composition scheme is Euclidean deviation (ED) and Angular Deviation (AD). ED refers to the Euclidean distance between the ideal point and the objective function value. AD refers to the angle between the objective function value vector $\boldsymbol{Y}$ and the ideal point vector $\boldsymbol{Y}^*$. The smaller the ED and AD, the better the service composition scheme; the larger the ED and AD, the worse the service composition scheme.

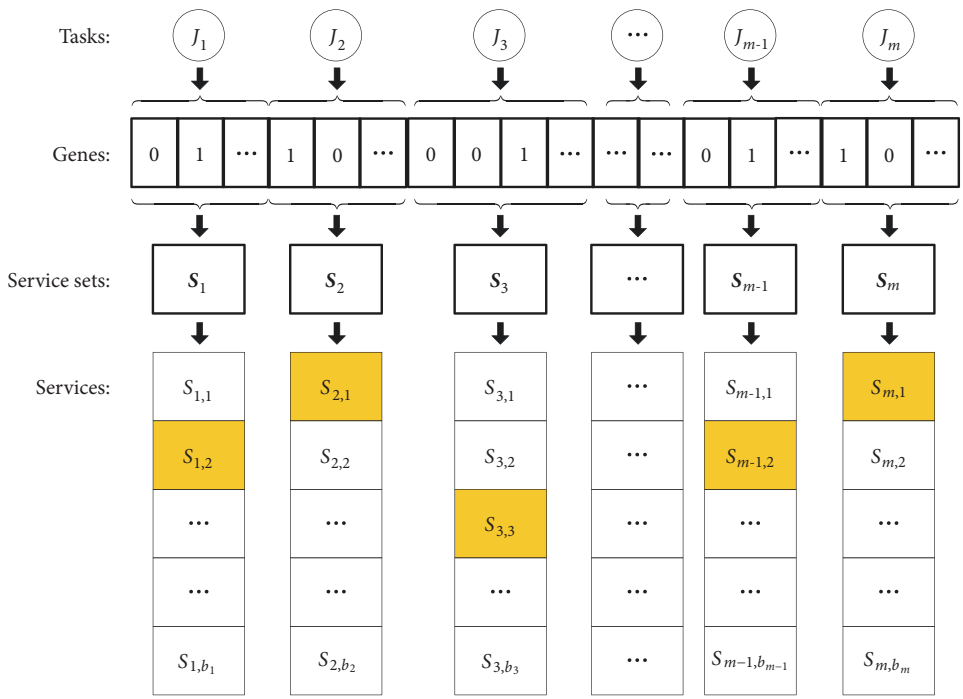


Figure 3: Encoding method example.

Euclidean deviation can be calculated as follows:

$$\min ED = \sqrt{\sum_{j=1}^{k} \left(Y_j - Y_j^*\right)^2} \tag{16}$$

where Y_j^* is the j-th objective function value of the ideal point; Y_j is the j-th objective function value of cloud manufacturing service composition scheme; k denotes that there are k objective functions in the cloud manufacturing service composition scheme; and ED is the Euclidean deviation of cloud manufacturing service composition scheme. If $k = 3$, there are three objective functions in cloud manufacturing service composition scheme, and the Euclidean deviation formula is as follows:

$$\min ED' = \sqrt{\left(Y_1 - Y_1^*\right)^2 + \left(Y_2 - Y_2^*\right)^2 + \left(Y_3 - Y_3^*\right)^2} \tag{17}$$

The formula for calculating the cosine value of the angular deviation is as follows:

$$\min CSD = \frac{\boldsymbol{Y} \cdot \boldsymbol{Y}^*}{\|\boldsymbol{Y}\| \, \|\boldsymbol{Y}^*\|} = \frac{\sum_{j=1}^{k} Y_j \times Y_j^*}{\sqrt{\sum_{j=1}^{k} \left(Y_j\right)^2} \times \sqrt{\sum_{j=1}^{k} \left(Y_j^*\right)^2}} \tag{18}$$

where CSD is the cosine value of the angular deviation of cloud manufacturing service composition scheme, and its unit is radian; $\boldsymbol{Y}$ is the objective function value vector of cloud manufacturing service composition scheme; $\boldsymbol{Y}^*$ is the ideal point vector; Y_j and Y_j^* represent the components of vectors $\boldsymbol{Y}$ and $\boldsymbol{Y}^*$, respectively; and $\boldsymbol{Y} \cdot \boldsymbol{Y}^*$ represents the point product of vectors $\boldsymbol{Y}$ and $\boldsymbol{Y}^*$.

The angular deviation formula is as follows:

$$\min AD = \arccos(CSD) = \arccos\left(\frac{\sum_{j=1}^{k} Y_j \times Y_j^*}{\sqrt{\sum_{j=1}^{k} \left(Y_j\right)^2} \times \sqrt{\sum_{j=1}^{k} \left(Y_j^*\right)^2}}\right) \tag{19}$$

where AD is the angular deviation of cloud manufacturing service composition scheme. If $k = 3$, the angular deviation formula is as follows:

$$\min AD' = \arccos\left(\frac{Y_1 \times Y_1^* + Y_2 \times Y_2^* + Y_3 \times Y_3^*}{\sqrt{\left(Y_1\right)^2 + \left(Y_2\right)^2 + \left(Y_3\right)^2} \times \sqrt{\left(Y_1^*\right)^2 + \left(Y_2^*\right)^2 + \left(Y_3^*\right)^2}}\right) \tag{20}$$

The weight coefficients are given according to the different importance of Y_1, Y_2, Y_3, ED, and AD. The fitness function of IGABE algorithm is designed as follows:

$$\Phi_i = \Gamma - \left(\varepsilon_1 \times \sqrt{\sum_{j=1}^{k} \delta_j \left(\frac{Y_j^i - Y_j^*}{Y_j^*} \right)^2} + \varepsilon_2 \times \frac{\sum_{i=1}^{k} \left(\delta_j \times Y_j^i \times Y_j^* \right)}{\sqrt{\sum_{j=1}^{k} \left(Y_j^i \right)^2} \times \sqrt{\sum_{j=1}^{k} \left(Y_j^* \right)^2}} \right) \tag{21}$$

If $k = 3$, the formula changes to

$$\Phi_i = \Gamma - \left(\varepsilon_1 \times \sqrt{\delta_1 \times \left(\frac{Y_1^i - Y_1^*}{Y_1^*} \right)^2 + \delta_2 \times \left(\frac{Y_2^i - Y_2^*}{Y_2^*} \right)^2 + \delta_3 \times \left(\frac{Y_3^i - Y_3^*}{Y_3^*} \right)^2} + \varepsilon_2 \times \frac{\delta_1 \times Y_1^i \times Y_1^* + \delta_2 \times Y_2^i \times Y_2^* + \delta_3 \times Y_3^i \times Y_3^*}{\sqrt{\left(Y_1^i\right)^2 + \left(Y_2^i\right)^2 + \left(Y_3^i\right)^2} \times \sqrt{\left(Y_1^*\right)^2 + \left(Y_2^*\right)^2 + \left(Y_3^*\right)^2}} \right) \tag{22}$$

where Γ is a sufficiently large positive number; Φ_i is the fitness function value of the i-th service composition scheme; Y_1^i, Y_2^i, and Y_3^i are the objective function values of the i-th service composition scheme; δ_1, δ_2, and δ_3 are the weight coefficients of the objective functions, and $\delta_1 + \delta_2 + \delta_3 = 1$; and ε_1 and ε_2 are the weight coefficients of the Euclidean deviation and the angular deviation respectively, and $\varepsilon_1 + \varepsilon_2 = 1$. The three objective function values affect the calculation of Euclidean deviation and angular deviation and ultimately affect the fitness function value. Each objective function has different importance to the fitness function of IGABE algorithm, and the three objective functions can be given different weight coefficients. The sum of the weight coefficients of the three objective functions is equal to 1, which is helpful for calculating the fitness function and evaluating the importance of the three objective functions. The sum of the weight coefficients of the Euclidean deviation and the angular deviation is equal to 1, which is helpful for evaluating the importance of the two deviations and calculating the fitness function. When the cosine value of the angle between the two service composition objective function value vectors is close to 1, namely, the angular deviation is close to 0 rad, the two service composition schemes are similar, which can be used for service composition classification and characteristic analysis. The smaller the angular deviation, the higher the similarity between the two service composition schemes. The larger the angular deviation, the more irrelevant the two service composition schemes. The angular deviation uses the cosine value of the angle between two service composition objective function value vectors as a measure of the difference between two service composition individuals. Compared with Euclidean deviation, angular deviation pays more attention to the difference of direction between two service composition objective function value vectors. Euclidean deviation measures the absolute distance between two service composition objective function values in space, which is directly related to the location coordinates of the service composition objective function values. When the Euclidean distance of two service composition objective function values equals 0 and the angular deviation equals 0 rad, the two service composition schemes are identical, which helps to delete the duplicate service composition schemes in the process of service composition. Angular deviation measures the angle between two service composition objective function value vectors, which is more reflected in the difference in direction than in position. Euclidean deviation and angular deviation have different calculation methods and measurement characteristics. Euclidean deviation can reflect the absolute difference of individual numerical characteristics of service composition objective functions, so it is more used to analyze the difference from the vector dimension of service composition objective functions. Angular deviation mainly distinguishes differences from the direction, but it is insensitive to the absolute value of the objective function. Therefore, it is more used to analyze the similarity and difference from the type and structure of the vector dimension of the objective function of service composition. Moreover, angular deviation can correct the problem of inconsistent measurement criteria in each dimension of service composition objective function vector.

4.3. Selection Operation. IGABE algorithm uses improved roulette selection method to perform selection operation. Its basic idea is to select an optimal individual by roulette selection; but after an individual is selected, the selection probability of the individual will be reduced accordingly; and then such operations repeat until all the individuals are produced. The specific process is as follows.

Step 1. The selection probability of each individual is calculated by fitness function. In a population composed of *Popsize* individuals, the selection probability of the i-th individual is calculated as follows:

$$P_i = \frac{f(i)}{\sum_{i=1}^{Popsize} f(i)}, \tag{23}$$

where $f(i)$ is the fitness function value of the i-th individual.

Step 2. According to the above selection probability, the individual u is selected by roulette selection method and put into the population. P_u is judged according to (24), and a new probability is given to the individual u.

$$P_u^{(i+1)} = \begin{cases} 0, & \text{if } P_u^{(i)} - \dfrac{1}{Popsize} \le 0; \\ P_u^{(i)} - \dfrac{1}{Popsize}, & \text{if } P_u^{(i)} - \dfrac{1}{Popsize} > 0. \end{cases} \quad (24)$$

where u is an integer and $u \in [1, Popsize]$.

Step 3. If the number of individuals selected reaches the population size, go to Step 4; otherwise, go to Step 1.

Step 4. Save all newly selected individuals, then return.

4.4. Crossover Operation. The crossover probability P_c is calculated according to (14), and *ExpVal*, *CloEnt*, and *HypEnt* of the crossover operator are computed. IGABE algorithm uses double points crossover operation and single point crossover operation. In the early stage of the algorithm, double points crossover operation is used to expand the search space and improve the diversity of population genes. In the later stage of the algorithm, single point crossover operation is used. If the double points crossover operation is selected, two gene segments are randomly selected as the crossover points. The selected gene code in the first parent individual is exchanged with the gene code at the same location in the second parent individual, and the unselected gene codes remain unchanged, resulting in two offspring individuals. The double points crossover operation is shown in Figure 4(a). If the single point crossover operation is selected, a gene segment is randomly selected as the crossover point. Its other operations are the same as double points crossover operation. The single point crossover operation is shown in Figure 4(b).

4.5. Mutation Operation. The mutation probability P_m is calculated according to (15), and *ExpVal*, *CloEnt*, and *HypEnt* of the mutation operator are computed. A gene segment in chromosome is randomly selected, a gene of the selected gene segment is coded at 1, and the rest of the gene segment is coded at 0, so as to generate a new chromosome. The mutation operation is shown in Figure 5.

5. Application Example

The proposed IGABE is applied to the manufacturing task of the MC type wheeled cleaning robot, which can be decomposed into seven subtasks: body production subtask J_1, driving device production subtask J_2, cleaning device production subtask J_3, power supply system subtask J_4, auxiliary control system subtask J_5, main control system subtask J_6, and painting process subtask J_7.

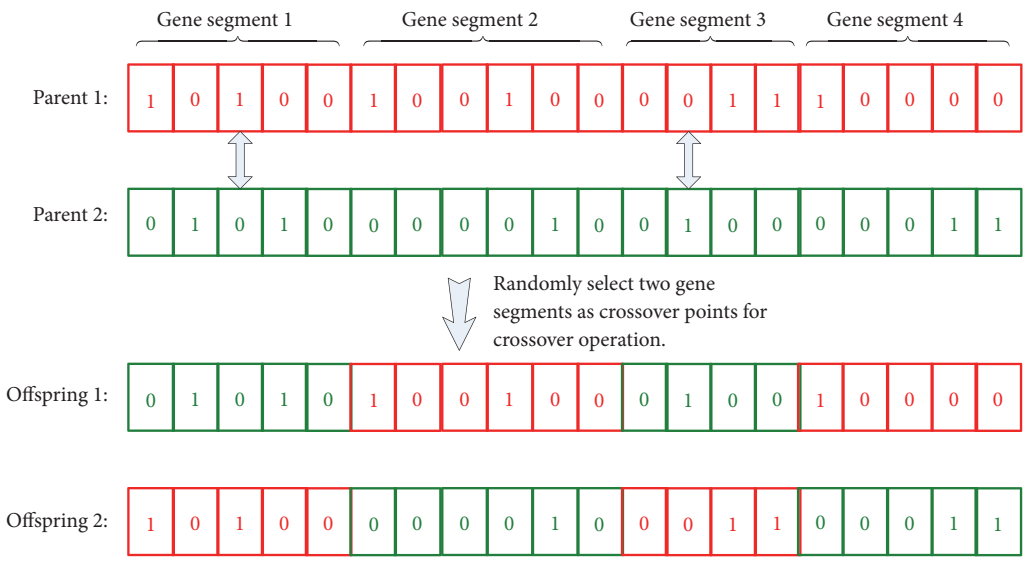
The cloud service sets available for manufacturing tasks J_1, J_2, J_3, J_4, J_5, J_6, and J_7 are S_1, S_2, S_3, S_4, S_5, S_6, and S_7 in turn. The numbers of cloud services contained in the service sets are 2, 3, 4, 2, 3, 2, and 2, respectively. The candidate cloud manufacturing services are arranged into a sequence according to the manufacturing task number, and the influence factor values are obtained according to the definition of service collocation degree. As shown in Table 1, *TF*, *HF* and *DF* represent the three influence factors of service collocation degree, i.e., technical factor, hunger factor and distance factor. T_{exe}, T_{con}, and T_{rep} are used for execution time [hour], longest continuous working time [hour], and maintenance time [hour], respectively. w denotes the unit time cost [dollar/hour]. Set α=0.4, β=0.3, and γ=0.3. The service collocation degree *CD* can be computed according to (1). The service composition entropy *CE* can be calculated according to (6).

According to (4), the composition synergy degree matrix of the manufacturing task is computed as follows:

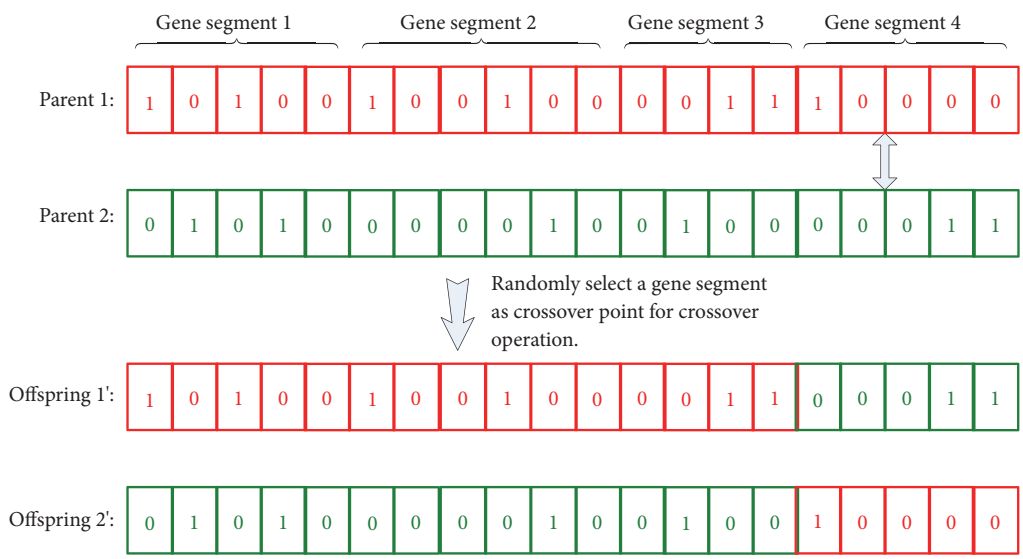
$$\boldsymbol{SD} = \begin{bmatrix} SD_{11} & SD_{12} & SD_{13} & SD_{14} & SD_{15} & SD_{16} & SD_{17} \\ SD_{21} & SD_{22} & SD_{23} & SD_{24} & SD_{25} & SD_{26} & SD_{27} \\ SD_{31} & SD_{32} & SD_{33} & SD_{34} & SD_{35} & SD_{36} & SD_{37} \\ SD_{41} & SD_{42} & SD_{43} & SD_{44} & SD_{45} & SD_{46} & SD_{47} \\ SD_{51} & SD_{52} & SD_{53} & SD_{54} & SD_{55} & SD_{56} & SD_{57} \\ SD_{61} & SD_{62} & SD_{63} & SD_{64} & SD_{65} & SD_{66} & SD_{67} \\ SD_{71} & SD_{72} & SD_{73} & SD_{74} & SD_{75} & SD_{76} & SD_{77} \end{bmatrix} \quad (25)$$

where all of the diagonal elements in the matrix ***SD*** are 1; nondiagonal elements are matrices with different dimensions, for example, SD_{13} is a 2×4 matrix, which is the composition synergy degree of two candidate cloud manufacturing services for the manufacturing task J_1 and four candidate cloud manufacturing services for the manufacturing task J_3, and so are the other elements in the matrix ***SD***. All elements of the matrix ***SD*** are calculated as shown in Table 2.

The improved genetic algorithm is programmed with MATLAB R2015a. The population size is set to 60, namely, *Popsize*=60, and the largest evolution generation is set to 160, namely, *Maxgen*=160. Set Γ=100. The delivery deadline is 450, and the cost constraint is 19000, namely, $ET_0 \le 450$, and $EC_0 \le 19000$. The weight coefficients of the three objective functions are 0.4, 0.4, and 0.2, namely, δ_1=0.4, δ_2=0.4, and δ_3=0.2. The weight coefficients of Euclidean deviation and angular deviation are set to 0.5, namely, ε_1=0.5 and ε_2=0.5. According to the single objective function optimization algorithm, the ideal point can be solved, namely, 5.15, 19.035, and 7.317. After fifty-seven iterations of the IGABE, the optimal fitness of the population is 99.974. The chromosome code of the optimal scheme is 100010010010101010. The Euclidean deviation between the optimal value of the service composition scheme and the ideal point is 1.170, and the angular deviation is 0.055 rad. The values of CD_{sum}, SD_{sum}, CE_{sum}, ET_{sum}, and EC_{sum} are 4.73, 18.586, 8.313, 415, and 14058, respectively. As shown in Figure 6, the meaning of the chromosome code is that the manufacturing task J_1 is delegated to the first cloud service in the cloud service set S_1; task J_2 to the third cloud service in the cloud service set S_2; task J_3 to the third cloud service in the cloud service set S_3; task J_4 to the second cloud service in the cloud service set S_4;



(a) Double points crossover operation



(b) Single point crossover operation

Figure 4: Crossover operation.

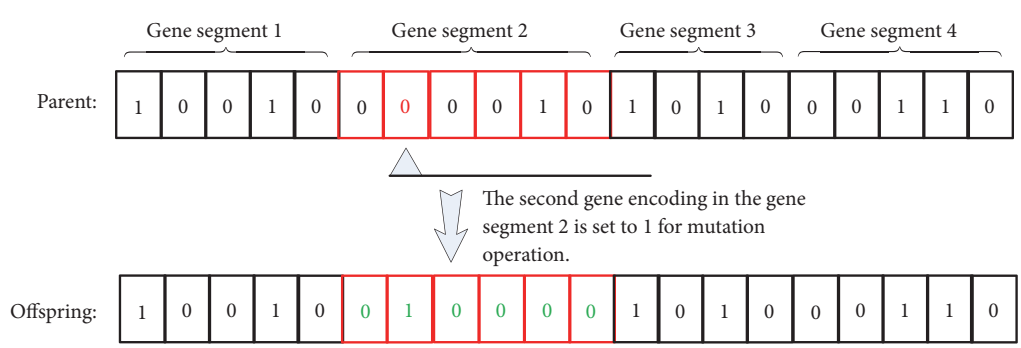


Figure 5: Mutation operation.

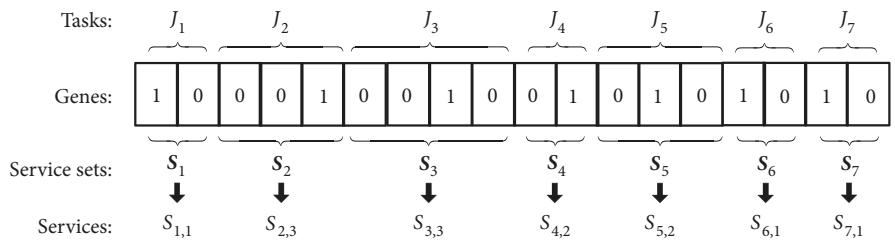


Figure 6: The optimal solution of the MC type wheeled cleaning robot manufacturing task.

Table 1: Collocation table between cloud services and manufacturing tasks.

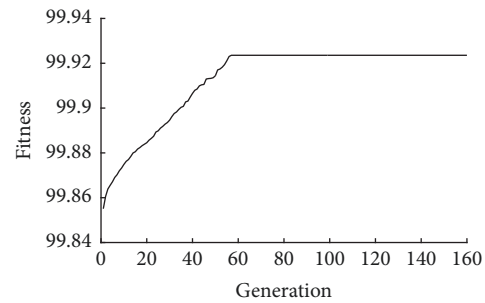
J_j	S_j	$S_{j,i}$	T_{exe}	T_{con}	T_{rep}	w	TF	HF	DF	CD	CE
J_1	S_1	$S_{1,1}$	49	20	4	31	0.4	0.6	0.8	0.58	1.220
		$S_{1,2}$	60	19	2	39	0.2	0.3	1	0.47	1.316
J_2	S_2	$S_{2,1}$	75	24	3	42	0.8	0.5	0.4	0.59	1.343
		$S_{2,2}$	63	20	2	51	0.2	0.4	0.8	0.44	1.309
		$S_{2,3}$	72	16	1	50	0.4	0.8	1	0.7	1.735
J_3	S_3	$S_{3,1}$	77	30	2	35	0.2	0.6	0.4	0.38	1.224
		$S_{3,2}$	68	24	2	35	0.4	0.7	0.8	0.61	1.283
		$S_{3,3}$	62	16	3	30	0.8	0.5	1	0.77	1.691
		$S_{3,4}$	76	20	1	32	0.4	0.9	0.4	0.55	1.527
J_4	S_4	$S_{4,1}$	90	24	2	25	0.6	0.9	0.8	0.75	1.580
		$S_{4,2}$	83	30	1	22	0.8	0.5	1	0.77	1.190
J_5	S_5	$S_{5,1}$	79	30	3	49	0.8	0.9	0.8	0.83	1.281
		$S_{5,2}$	46	24	4	40	0.2	0.5	1	0.53	0.919
		$S_{5,3}$	52	24	1	45	0.2	0.8	0.4	0.44	0.991
J_6	S_6	$S_{6,1}$	48	30	2	31	0.6	0.7	1	0.75	0.792
		$S_{6,2}$	58	40	1	36	0.2	0.4	0.8	0.44	0.689
J_7	S_7	$S_{7,1}$	55	30	1	35	0.6	0.3	1	0.63	0.765
		$S_{7,2}$	62	24	2	40	0.6	0.9	0.8	0.75	1.251

task J_5 to the second cloud service in the cloud service set S_5; task J_6 to the first cloud service in the cloud service set S_6; and task J_7 to the first cloud service in the cloud service set S_7. The average run time of the algorithm is 13.66s, and the evolution curves are shown in Figure 7. Figure 7(a) shows the evolution curve of the optimal individual fitness. Figure 7(b) shows the service collocation degree evolution curve. Figure 7(c) shows the composition synergy degree evolution curve. Figure 7(d) shows the composition entropy evolution curve. Figure 7(e) shows the evolution curve of service composition execution time. Figure 7(f) shows the execution cost evolution curve. Figure 7(g) shows the evolution curve of the Euclidean deviation between the optimal values of the service composition objective functions and the ideal point. Figure 7(h) shows the evolution curve of the angular deviation between the optimal objective function value vector and the ideal point vector. Figure 7(i) shows the three-dimensional scatter plot of the IGABE solution. It tends to be stable when the population evolves to the fifty-seventh generation.

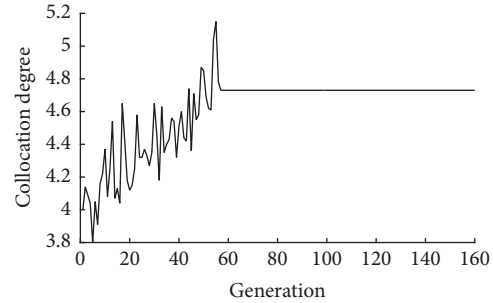
Given the same maximum evolutionary generation and population size, IGABE algorithm, traditional genetic algorithm (SGA), hybrid genetic algorithm (HGA) [45], and cloud-entropy enhanced genetic algorithm (CEGA) [34] are used to solve the same problem. As shown in Figure 8, the IGABE has converged to the optimal solution of the manufacturing task when it evolves to the fifty-seventh generation, the SGA has not converged until the ninety-fourth generation, the HGA converges in the seventy-first generation, and the CEGA converges in the fifty-fourth generation. Experiments were carried out on a portable computer with CPU Intel core i3-3110M, 2.4GHz main frequency, and 4G memory. The times spent by the IGABE, SGA, HGA, and CEGA are 13.66s, 22.78s, 16.79s, and 12.83s, respectively, as shown in Table 3. The IGABE solves the problem faster than the SGA and HGA. The solving speed of the IGABE is close to that of the CEGA. The above case analysis and experimental results show that, for cloud manufacturing service composition multiobjective optimization problems, the IGABE has faster convergence speed and shorter solution time than the SGA and HGA and is close to that of the CEGA. But there are obvious differences between the IGABE and CEGA in analysis effect and content. The CEGA embodies the absolute difference of the total objective function values of different service composition schemes. The differences of service composition schemes are analyzed and reflected from the Euclidean deviation between them and the ideal point. However, the CEGA is insensitive

Table 2: Calculation results of composition synergy degree.

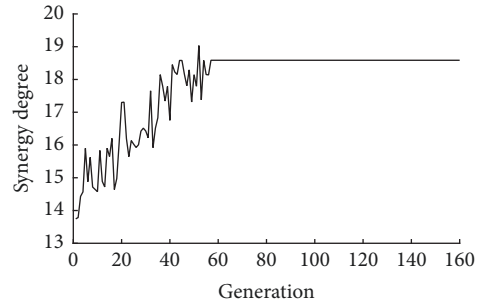
SD	$S_{1,1}$	$S_{1,2}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$	$S_{3,4}$	$S_{4,1}$	$S_{4,2}$	$S_{5,1}$	$S_{5,2}$	$S_{5,3}$	$S_{6,1}$	$S_{6,2}$	$S_{7,1}$	$S_{7,2}$
$S_{1,1}$	1.000	1.000	0.594	0.771	0.836	0.594	0.772	0.834	0.594	0.777	0.838	0.774	0.833	0.588	0.833	0.770	0.834	0.770
$S_{1,2}$	1.000	1.000	0.528	0.833	0.9094	0.528	0.834	0.909	0.528	0.836	0.910	0.835	0.910	0.527	0.910	0.833	0.909	0.833
$S_{2,1}$	0.594	0.528	1.000	1.000	1.000	0.667	0.589	0.527	0.667	0.589	0.527	0.588	0.534	0.670	0.532	0.590	0.529	0.589
$S_{2,2}$	0.771	0.833	1.000	1.000	1.000	0.589	0.769	0.833	0.589	0.772	0.835	0.770	0.835	0.589	0.835	0.769	0.834	0.769
$S_{2,3}$	0.836	0.909	1.000	1.000	1.000	0.526	0.833	0.909	0.526	0.834	0.909	0.833	0.911	0.530	0.911	0.834	0.910	0.834
$S_{3,1}$	0.594	0.528	0.667	0.589	0.526	1.000	1.000	1.000	1.000	0.589	0.526	0.588	0.534	0.671	0.533	0.591	0.530	0.590
$S_{3,2}$	0.772	0.834	0.589	0.769	0.833	1.000	1.000	1.000	1.000	0.771	0.834	0.770	0.836	0.590	0.835	0.770	0.834	0.769
$S_{3,3}$	0.834	0.909	0.527	0.833	0.909	1.000	1.000	1.000	1.000	0.836	0.910	0.834	0.910	0.527	0.834	0.769	0.909	0.833
$S_{3,4}$	0.594	0.528	0.667	0.589	0.526	1.000	1.000	1.000	1.000	0.589	0.527	0.588	0.534	0.671	0.533	0.590	0.530	0.589
$S_{4,1}$	0.777	0.836	0.589	0.772	0.834	0.589	0.771	0.836	0.589	1.000	1.000	0.770	0.841	0.597	0.840	0.773	0.837	0.772
$S_{4,2}$	0.838	0.910	0.527	0.835	0.909	0.526	0.834	0.910	0.527	1.000	1.000	0.833	0.913	0.533	0.912	0.836	0.911	0.835
$S_{5,1}$	0.774	0.835	0.588	0.770	0.833	0.588	0.770	0.834	0.588	0.770	0.833	1.000	1.000	1.000	0.838	0.771	0.836	0.771
$S_{5,2}$	0.833	0.910	0.534	0.835	0.911	0.534	0.836	0.910	0.534	0.841	0.913	1.000	1.000	1.000	0.909	0.834	0.909	0.835
$S_{5,3}$	0.588	0.527	0.670	0.589	0.530	0.671	0.590	0.527	0.671	0.597	0.533	1.000	1.000	1.000	0.527	0.589	0.526	0.589
$S_{6,1}$	0.833	0.910	0.532	0.835	0.911	0.533	0.835	0.834	0.533	0.840	0.912	0.838	0.909	0.527	1.000	1.000	0.909	0.834
$S_{6,2}$	0.770	0.833	0.590	0.769	0.834	0.591	0.770	0.769	0.590	0.773	0.836	0.771	0.834	0.589	1.000	1.000	0.833	0.769
$S_{7,1}$	0.834	0.909	0.529	0.834	0.910	0.530	0.834	0.909	0.530	0.837	0.911	0.836	0.909	0.526	0.909	0.833	1.000	1.000
$S_{7,2}$	0.770	0.833	0.589	0.769	0.834	0.590	0.769	0.833	0.589	0.772	0.835	0.771	0.835	0.589	0.834	0.769	1.000	1.000



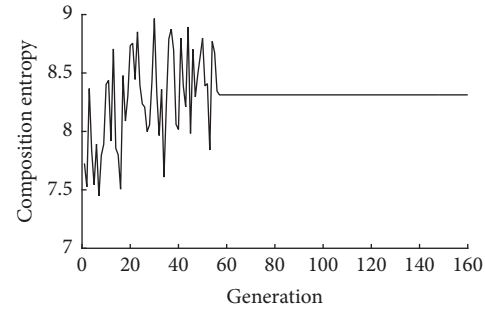
(a) The optimal individual fitness evolution curve [34]



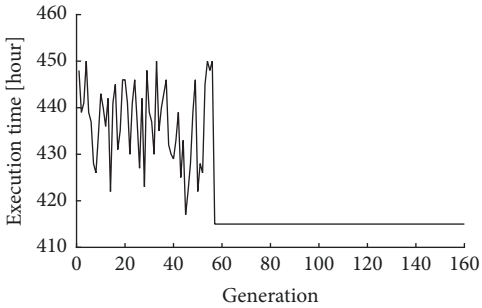
(b) Service collocation degree evolution curve [34]



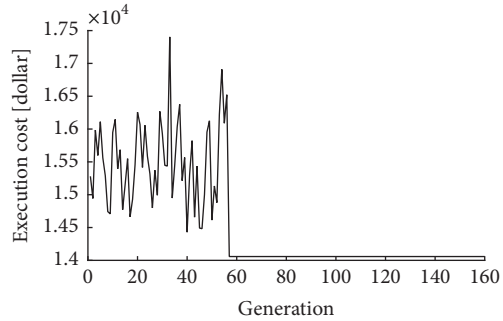
(c) Composition synergy degree evolution curve [34]



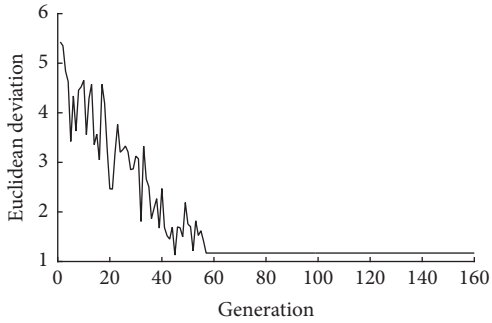
(d) Composition entropy evolution curve [34]



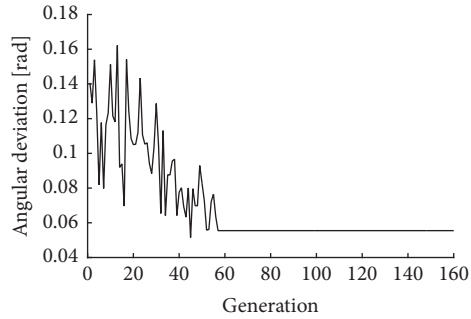
(e) Execution time evolution curve [34]



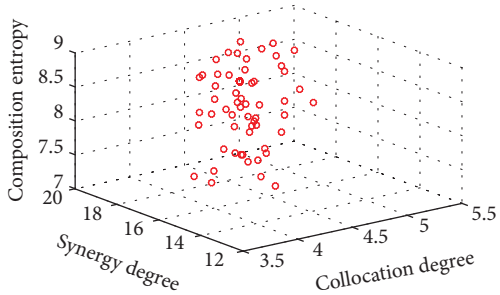
(f) Execution cost evolution curve [34]



(g) Euclidean deviation evolution curve [34]



(h) Angular deviation evolution curve



(i) Three-dimensional scatter diagram of IGABE solution [34]

Figure 7: IGABE evolution curves.

Table 3: Optimization results for the MC type wheeled cleaning robot manufacturing task.

Algorithms	Computing time	Convergence generation	Optimization results	ED	AD
SGA	22.78 s	94	$\{S_{1,1}, S_{2,2}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	1.259	0.058 rad
HGA	16.79 s	71	$\{S_{1,2}, S_{2,2}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	1.129	0.064 rad
CEGA	12.83 s	54	$\{S_{1,2}, S_{2,2}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	1.129	0.064 rad
IGABE	13.66 s	57	$\{S_{1,1}, S_{2,3}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	1.170	0.055 rad

Table 4: Comparison of different factors' influences on service composition.

Algorithms	Optimization results	CD_{sum}	SD_{sum}	CE_{sum}	ET_{sum}	EC_{sum}	ED	AD
IGABE	$\{S_{1,1}, S_{2,3}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	4.73	18.586	8.313	415	14058	1.170	0.055 rad
MTSC	$\{S_{1,1}, S_{2,2}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	4.47	18.142	7.887	406	13671	1.259	0.058 rad
MCSC	$\{S_{1,1}, S_{2,1}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	4.62	16.443	7.921	418	13608	2.714	0.079 rad
MCDSC	$\{S_{1,1}, S_{2,3}, S_{3,3}, S_{4,2}, S_{5,1}, S_{6,1}, S_{7,1}\}$	5.15	18.150	8.675	448	16089	1.621	0.077 rad
MSDSC	$\{S_{1,2}, S_{2,3}, S_{3,3}, S_{4,2}, S_{5,2}, S_{6,1}, S_{7,1}\}$	4.62	19.035	8.409	426	14879	1.214	0.056 rad
MCESC	$\{S_{1,1}, S_{2,2}, S_{3,1}, S_{4,2}, S_{5,2}, S_{6,2}, S_{7,1}\}$	3.77	15.919	7.317	431	15106	3.407	0.072 rad

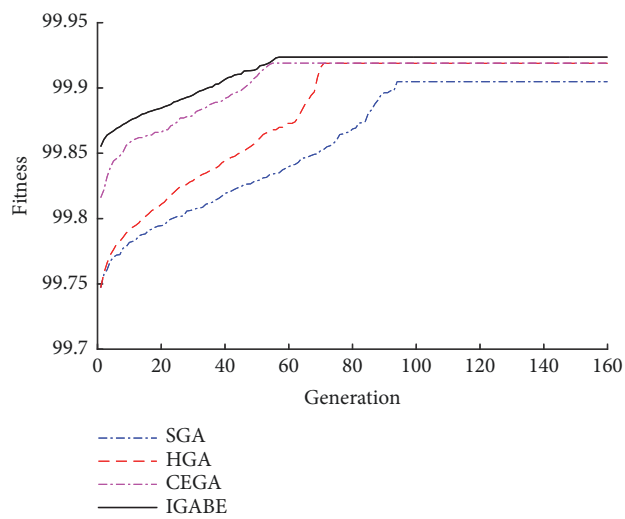


Figure 8: Comparison of IGABE, SGA, HGA, and CEGA evolution curves.

to the different-dimension value difference between the objective function value vector and the ideal point vector in the service composition scheme, and the dimension value difference with small absolute value is easily submerged by the dimension value difference with great absolute value. The IGABE combines Euclidean deviation with angular deviation, which overcomes the problem of nonuniform measurement criteria among the objective function values, and such more comprehensive analyses reflect the differences of the content expressed by the objective functions of service composition schemes and the degree of proximity to the dimension values of ideal point vector. Therefore, the optimal service composition scheme solved by the IGABE can better meet the complex needs of users.

In the process of applying the IGABE to optimize service composition, the effects of technical factor, hunger factor, vacancy rate of manufacturing resources, distance factor, collaboration level of multiple cloud manufacturing services, coupling relationship, complexity, and orderliness of service composition, manufacturing time, and cost are fully considered. Service collocation degree, composition synergy degree, composition entropy, execution time, and execution cost are taken as five variables in the optimization model. The resulting service composition scheme has better comprehensive characteristics than the general methods, as shown in Table 4. Among the six service composition schemes shown in the table, Minimum Time Service Composition (MTSC) scheme has the smallest execution time, but its service collocation degree and composition synergy degree are smaller than the IGABE, and it has bigger Euclidean deviation and angular deviation. Minimum Cost Service Composition (MCSC) scheme has the smallest execution cost, but its service collocation degree and composition synergy degree are smaller than the IGABE, and it has bigger execution time, Euclidean deviation, and angular deviation. Maximum Collocation Degree Service Composition (MCDSC) scheme has the biggest service collocation degree, but its composition synergy degree is smaller than the IGABE, and it has bigger composition entropy, execution time, execution cost, Euclidean deviation, and angular deviation. Maximum Synergy Degree Service Composition (MSDSC) scheme has the biggest composition synergy degree, but its service collocation degree is smaller than the IGABE, and it has bigger composition entropy, execution time, execution cost, Euclidean deviation, and angular deviation. Minimum Composition Entropy Service Composition (MCESC) scheme has smallest composition entropy, but its service collocation degree and composition synergy degree are smaller than the IGABE, and it has bigger execution time, execution cost, Euclidean deviation, and angular deviation. Compared with these five service composition schemes, the IGABE has the smallest Euclidean deviation and angular deviation, and the best comprehensive performance, which help users to make more reasonable decisions from the viewpoint of cloud services. If only the execution cost and execution time are considered in service composition, without consideration of the impacts of service collocation degree, composition synergy degree, and

composition entropy, it may cause user's mistake in decision-making and bring adverse effects on manufacturing, such as the shortcomings of MTSC and MCSC. Because their service collocation degree and composition synergy degree are very small, the manufacturing services selected from the two service composition schemes may have some problems, such as insufficient technical capability, poor service quality, low service reliability, poor information flow, logistics congestion, and so on, which may affect product quality, delivery time, and even the survival of manufacturing enterprises. For example, large multinational manufacturing enterprises like Huawei need to consider service collocation degree, composition synergy degree, composition entropy, execution time, and execution cost comprehensively when purchasing parts globally based on cloud services, because low quality service matching, unstable cooperation, low service reliability, and blocked logistics may hinder the normal manufacturing of products and threaten the survival of enterprises.

6. Conclusion

In order to optimize the allocation of cloud manufacturing resources and improve the execution efficiency of cloud manufacturing service composition, the multiobjective optimization problem of cloud manufacturing service composition has been studied. The main works and contributions include the following aspects.

(1) The main influence factors of cloud manufacturing service composition performance are studied and modeled as service collocation degree, composition synergy degree and composition entropy.

(2) With the constraints of manufacturing task execution time and execution cost, and the objective functions of service collocation degree, composition synergy degree, and composition entropy, the mathematical model of cloud manufacturing service composition optimization is established, which provides a multiobjective optimization solution for cloud manufacturing service composition optimization problems.

(3) An improved genetic algorithm IGABE is proposed, whose crossover and mutation operators are improved by introducing normal cloud model and piecewise function. The improved roulette selection method is used to perform the selection operation of the algorithm, and the fitness function is designed by combining Euclidean deviation with angular deviation.

(4) Taking the manufacturing task of MC type wheeled cleaning robot as an example, the correctness of multiobjective optimization mathematical model for cloud manufacturing service composition and the feasibility and effectiveness of the proposed IGABE are verified. The case study results show that for cloud manufacturing service composition multiobjective optimization problem, the IGABE has better solution quality and shorter solution time than the SGA. The HGA gets the same optimization results as the IGABE, but its convergence speed is inferior to the IGABE. The convergence speed of the IGABE is close to that of the CEGA. The IGABE is more sensitive than the CEGA to the different-dimension value differences between the objective function value vector and the ideal point vector in the service composition scheme, and the optimal service composition scheme solved by the IGABE can better meet the complex needs of users.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The project was supported by the National Natural Science Foundation of China (51375168), the GUES Scientific Research Foundation for Advanced Talents (YKHZ G2018009), the Science & Technology Foundation of Guizhou Province (QKHLHZ [2014]7529), and the Science and Technology Youth Talent Growth Foundation of Guizhou Province (QKHKYZ [2018]400).

References

[1] B. H. Li, L. Zhang, S. L. Wang et al., "Cloud manufacturing: a new service-oriented networked manufacturing model," *Computer Integrated Manufacturing Systems*, vol. 16, no. 1, pp. 1–7, 2010.

[2] F. V. Omid and M. Houshmand, "A platform for optimisation in distributed manufacturing enterprises based on cloud manufacturing paradigm," *International Journal of Computer Integrated Manufacturing*, vol. 27, no. 11, pp. 1031–1054, 2014.

[3] J. Lartigau, X. Xu, L. Nie, and D. Zhan, "Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm," *International Journal of Production Research*, vol. 53, no. 14, pp. 4380–4404, 2015.

[4] T. Chen and Y. Wang, "Estimating simulation workload in cloud manufacturing using a classifying artificial neural network ensemble approach," *Robotics and Computer-Integrated Manufacturing*, vol. 38, pp. 42–51, 2016.

[5] Q. N. Meng and X. Xu, "Price forecasting using an ACO-based support vector regression ensemble in cloud manufacturing," *Computers & Industrial Engineering*, vol. 125, no. 11, pp. 171–177, 2018.

[6] Y. Zhang, G. Zhang, T. Qu, Y. Liu, and R. Y. Zhong, "Analytical target cascading for optimal configuration of cloud manufacturing services," *Journal of Cleaner Production*, vol. 151, no. 5, pp. 330–343, 2017.

[7] Y. Liu, X. Xu, L. Zhang, L. Wang, and R. Y. Zhong, "Workload-based multi-task scheduling in cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 45, no. 6, pp. 3–20, 2017.

[8] F. Li, T. Liao, and L. Zhang, "Two-level multi-task scheduling in a cloud manufacturing environment," *Robotics and Computer-Integrated Manufacturing*, vol. 56, no. 4, pp. 127–139, 2019.

[9] S. Huang, X. Gu, H. Zhou, and Y. Chen, "Two-dimensional optimization mechanism and method for on-demand supply of

manufacturing cloud service," *Computers & Industrial Engineering*, vol. 117, no. 3, pp. 47–59, 2018.

[10] N. Liu, X. Li, and W. Shen, "Multi-granularity resource virtualization and sharing strategies in cloud manufacturing," *Journal of Network and Computer Applications*, vol. 46, pp. 72–82, 2014.

[11] P. Yongdong, "Bi-level programming optimization method for cloud manufacturing service composition based on harmony search," *Journal of Computational Science*, vol. 27, no. 7, pp. 462–468, 2018.

[12] Y. X. Li and X. F. Yao, "Cloud manufacturing service composition and formal verification based on extended process calculus," *Advances in Mechanical Engineering*, vol. 10, no. 6, pp. 1–16, 2018.

[13] Y. Hu, X. Chang, Y. Wang, Z. Wang, C. Shi, and L. Wu, "Cloud manufacturing resources fuzzy classification based on genetic simulated annealing algorithm," *Materials and Manufacturing Processes*, vol. 32, no. 10, pp. 1109–1115, 2017.

[14] J. Thekinen and J. H. Panchal, "Resource allocation in cloud-based design and manufacturing: A mechanism design approach," *Journal of Manufacturing Systems*, vol. 43, no. 2, pp. 327–338, 2017.

[15] F. Tao, J. Cheng, Y. Cheng, S. Gu, T. Zheng, and H. Yang, "SDMSim: A manufacturing service supply–demand matching simulator under cloud environment," *Robotics and Computer-Integrated Manufacturing*, vol. 45, no. 6, pp. 34–46, 2017.

[16] S. Răileanu, F. Anton, T. Borangiu, S. Anton, and M. Nicolae, "A cloud-based manufacturing control system with data integration from multiple autonomous agents," *Computers in Industry*, vol. 102, no. 11, pp. 50–61, 2018.

[17] A. Brant and M. M. Sundaram, "A novel system for cloud-based micro additive manufacturing of metal structures," *Journal of Manufacturing Processes*, vol. 20, no. 3, pp. 478–484, 2015.

[18] M. R. Namjoo and A. Keramati, "Analysing causal dependencies of composite service resilience in cloud manufacturing using resource-based theory and dematel method," *International Journal of Computer Integrated Manufacturing*, vol. 31, no. 4, pp. 942–960, 2018.

[19] G. Zhang, Y. Zhang, X. Xu, and R. Y. Zhong, "An augmented Lagrangian coordination method for optimal allocation of cloud manufacturing services," *Journal of Manufacturing Systems*, vol. 48, no. 7, pp. 122–133, 2018.

[20] M. Zhang, C. Li, Y. Shang, and C. Li, "Research on resource service matching in cloud manufacturing," *Manufacturing Letters*, vol. 15, no. 1, pp. 50–54, 2018.

[21] W. Zhang, S. Zhang, S. Guo, Y. Yang, and Y. Chen, "Concurrent optimal allocation of distributed manufacturing resources using extended teaching-learning-based optimization," *International Journal of Production Research*, vol. 55, no. 3, pp. 718–735, 2016.

[22] W. Y. Zhang, S. S. Guo, and S. Zhang, "Combining hyperlink-induced topic search and Bayesian approach for personalised manufacturing service recommendation," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 11, pp. 1–12, 2016.

[23] B. Sheng, C. Zhang, X. Yin et al., "Common intelligent semantic matching engines of cloud manufacturing service based on OWL-S," *The International Journal of Advanced Manufacturing Technology*, vol. 84, no. 1-4, pp. 103–118, 2016.

[24] P. Helo and Y. Hao, "Cloud manufacturing system for sheet metal processing," *Production Planning and Control*, vol. 28, no. 6-8, pp. 524–537, 2017.

[25] V. Souza, X. Masip-Bruin, E. Marín-Tordera et al., "Towards a proper service placement in combined Fog-to-Cloud (F2C) architectures," *Future Generation Computer Systems*, vol. 87, no. 10, pp. 1–15, 2018.

[26] M. Lang, M. Wiesche, and H. Krcmar, "Criteria for selecting cloud service providers: a delphi study of quality-of-service attributes," *Information and Management*, vol. 55, no. 6, pp. 746–758, 2018.

[27] H. Yan, Q. Hua, Y. Wang, W. Wei, and M. Imran, "Cloud robotics in smart manufacturing environments: challenges and countermeasures," *Computers and Electrical Engineering*, vol. 63, no. 10, pp. 56–65, 2017.

[28] J. Zhou, X. Yao, Y. Lin, F. T. Chan, and Y. Li, "An adaptive multi-population differential artificial bee colony algorithm for many-objective service composition in cloud manufacturing," *Information Sciences*, vol. 456, no. 8, pp. 50–82, 2018.

[29] C. Yang, W. Shen, T. Lin, and X. Wang, "IoT-enabled dynamic service selection across multiple manufacturing clouds," *Manufacturing Letters*, vol. 7, no. 1, pp. 22–25, 2016.

[30] Y. Lu and X. Xu, "A semantic web-based framework for service composition in a cloud manufacturing environment," *Journal of Manufacturing Systems*, vol. 42, no. 1, pp. 69–81, 2017.

[31] J. Zhou and X. Yao, "Multi-population parallel self-adaptive differential artificial bee colony algorithm with application in large-scale service composition for cloud manufacturing," *Applied Soft Computing*, vol. 56, no. 7, pp. 379–397, 2017.

[32] F. Chen, R. Dou, M. Li, and H. Wu, "A flexible QoS-aware Web service composition method by multi-objective optimization in cloud manufacturing," *Computers & Industrial Engineering*, vol. 99, no. 9, pp. 423–431, 2016.

[33] X. Huang, Z. Guan, and L. Yang, "An effective hybrid algorithm for multi-objective flexible job-shop scheduling problem," *Advances in Mechanical Engineering*, vol. 10, no. 9, pp. 1–14, 2018.

[34] Y. Li, X. Yao, and J. Zhou, "Multi-objective optimization of cloud manufacturing service composition with cloud-entropy enhanced genetic algorithm," *Strojniski Vestnik: Journal of Mechanical Engineering*, vol. 62, no. 10, pp. 577–590, 2016.

[35] C. Li, J. Guan, T. Liu, and J. Zhang, "An autonomy-oriented method for service composition and optimal selection in cloud manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 96, no. 3, pp. 1–22, 2018.

[36] Y. Que, W. Zhong, H. Chen, X. Chen, and X. Ji, "Improved adaptive immune genetic algorithm for optimal QoS-aware service composition selection in cloud manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 10, pp. 1–11, 2018.

[37] Y. Wang, Z. Dai, W. Zhang, S. Zhang, Y. Xu, and Q. Chen, "Urgent task-aware cloud manufacturing service composition using two-stage biogeography-based optimisation," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 10, pp. 1–14, 2018.

[38] S. Zhang, Y. Xu, W. Zhang, and D. Yu, "A new fuzzy QoS-aware manufacture service composition method using extended flower pollination algorithm," *Journal of Intelligent Manufacturing*, vol. 30, no. 4, pp. 1–15, 2017.

[39] B. Xu and Y. Cai, "A multiple-data-based efficient global optimization algorithm and its parallel implementation for automotive body design," *Advances in Mechanical Engineering*, vol. 10, no. 8, pp. 1–13, 2018.

[40] F. Zhao, W. Lei, W. Ma et al., "An improved SPEA2 algorithm with adaptive selection of evolutionary operators scheme for

multiobjective optimization problems," *Mathematical Problems in Engineering*, vol. 2016, Article ID 8010346, 20 pages, 2016.

[41] W. Liu, Y. Li, and B. Liu, "Service composition in cloud manufacturing based on adaptive mutation particle swarm optimization," *Journal of Computer Applications*, vol. 2018, no. 10, pp. 2869–2874, 2018.

[42] B. F. Bao, Y. Yang, L. T. Li, F. Li, A. J. Liu, and N. Liu, "Multi-objective optimization for task allocation of product customization collaborative development," *Computer Integrated Manufacturing Systems*, vol. 20, no. 4, pp. 739–746, 2014.

[43] Y.-J. Chen, X.-F. Yao, and D.-L. Xu, "Job shop scheduling with profit and entropy as performance measures," *Beijing Gongye Daxue Xuebao/Journal of Beijing University of Technology*, vol. 36, no. 10, pp. 1305–1311, 2010.

[44] C. H. Dai, Y. F. Zhu, W. R. Chen, and J. H. Lin, "Cloud model based genetic algorithm and its application," *Acta Electronica Sinica*, vol. 35, no. 7, pp. 1419–1424, 2007.

[45] Y. J. Huang, X. F. Yao, D. Y. Ge, and Y. X. Li, "Entropy-enhanced genetic algorithm with tabu search for job shop scheduling problems," *Advanced Materials Research*, vol. 590, no. 183, pp. 557–562, 2012.