Hindawi
Complexity
Volume 2019, Article ID 8545364, 14 pages
https://doi.org/10.1155/2019/8545364

Research Article

An Optimal Composition Strategy for Knowledge Service Component Based on Flexible Tracking Particle Swarm Algorithm

Yan-chao Yin ,[1] Fu-zhao Chen,[1] Wei-zhi Liao ,[2] and Cui-yin Liu[3]

[1]*Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, China*
[2]*School of Mechanical and Electrical Engineering, University of Electronic Science and Technology, Chengdu, China*
[3]*Yunnan Province Key Laboratory of Computer Technology Applications, Kunming University of Science and Technology, Kunming, China*

Correspondence should be addressed to Yan-chao Yin; yinyc@163.com

Received 16 August 2019; Revised 5 November 2019; Accepted 14 November 2019; Published 12 December 2019

Guest Editor: Raúl Baños

Copyright © 2019 Yan-chao Yin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is urgent to combine knowledge resources with manufacturing business processes to form a knowledge service in the cloud mode, so as to provide intelligent support for business activities in product development process. The main challenge of knowledge resource service, however, is how to rapidly construct the complex resource service system and respond promptly to the changeable service requirements in the business process, which is similar to the software system modeling using a component in software engineering. This paper is concerned with an optimal composition framework (OCF) of knowledge resource service, including service decomposition, component encapsulation, and optimal composition. Firstly, the typical business processes are decomposed into the dynamic knowledge element (DKE), and all kinds of knowledge resources and service behaviors are encapsulated into the reusable resource service components (RSC). Then, a multicomponent optimal composition mathematical model is presented, which transforms the problem of component composition into a multiobjective optimization problem. On this basis, a heuristic algorithm with the adaptive mutation probability is introduced to composite the multigranularity service component dynamically and robustly. Finally, the case of component composition for maintenance resource service is studied and the simulation results are provided to verify the efficacy of the proposed model and algorithms.

1. Introduction

In the cloud mode, enterprises have a large number of knowledge resources such as standards, specifications, methods, and cases. Therefore, how to reuse the existing knowledge resources and experience quickly and effectively for product innovation and development has become the key to improve the competitiveness of enterprises. Although enterprises have widely implemented and promoted application systems such as the product data management and the enterprise resource management, these systems have not yet been upgraded to the knowledge service, but also limited to the traditional data classification storage and retrieval, lack of interaction with business process activities, resulting in low reuse rate of knowledge. So, it is urgent to integrate knowledge with product business process and package them into a knowledge service, forming a standardized guidance mechanism, helping users to use knowledge service correctly, and improving product development efficiency.

How to provide resource services according to the characteristics of enterprise knowledge resources in the cloud mode is an urgent problem to be solved [1, 2]. The purpose of knowledge service is to provide recommended solutions to users' knowledge service needs according to product development process and user identity, which often requires multiple knowledge resources to work together, that is, the combination of knowledge services. Specifically, in the process of new product development, on the one hand, they need to actively recommend various knowledge resources related to business activities according to task requirements in the whole manufacturing life cycle and on the other hand, it is necessary to organize various knowledge resources (including parameters, methods, models, and tools) for a specific function or process in the product development process and provide them to enterprise users in the form of services. This mode of knowledge resource allocation and use on demand urgently needs to combine knowledge with manufacturing business process to form a knowledge service [3]. Consequently, how to combine knowledge resources with manufacturing business processes to form a knowledge service is of great importance for manufacturing industry, which will promote the transformation of industry from "operation mode" to "prediction model" [4, 5]. However, the knowledge service is composed of multiple types of service elements, including service subject, service object, service behavior, knowledge resources, involved organizations and their interaction relationship, which is difficult to respond promptly to the changeable service requirements in business process, that is, on-demand use of knowledge resources requires an effective integration method for resources and business process.

Referring to Figure 1, after analyzing the after-sales maintenance process, it can be divided into five business activities: fault phenomenon analysis, fault cause judgment, fault maintenance guidance, service tool push, and maintenance evaluation. Each service activity organizes knowledge resources such as documents, specifications, patents, cases, methods, empirical parameters, optimization models, and software tools and forms the standardized and curable resources (parameters, models, tools, and methods) into independent functional resource service components through abstraction and encapsulation in these business processes. The knowledge service component encapsulates the intermediate process and only provides the parameter interface, for which a knowledge service activity is a dynamic node of the whole business process, and the output of the previous knowledge service can generally be used as the input of the next knowledge service. When users call the corresponding knowledge services, they only need to combine and optimize different resource service components according to the service requirements to complete the task efficiently.

This paper is concerned with service-oriented encapsulation of knowledge resources and business process, thus transforming knowledge into a resource or product and finally into economic benefits. Following this idea, an optimal composition framework (OCF) including service task decomposition, component encapsulation, and optimal composition was constructed, where the typical business processes of the enterprise are sort out, and knowledge such as tools, methods, and parameters used in typical business activities can be obtained. On this basis, the task of knowledge service is decomposed into a series of subtasks which cannot be subdivided but can be completed by a single knowledge service. For component encapsulation, all kinds of standardized and curable knowledge are encapsulated into different knowledge service components according to their granularity and reuse rate through a business activity. For combinatorial optimization, the candidate set of knowledge services corresponding to each subservice task is searched in the resource pool and an appropriate knowledge service is selected to form the possible combinatorial knowledge services according to the order of tasks. Finally, according to the given service requirements and constraints, the most satisfactory service combination is selected from all possible combinations.

2. Related Work

Knowledge services are transforming from traditional data classification storage and retrieval to knowledge services in the cloud mode. The construction process mainly includes knowledge resource modeling, knowledge service searching, matching, and combination optimization. In this field, knowledge resource modeling, knowledge service search, and matching have been of great interests during the past decades [6]. In particular, the knowledge resources modeling through multidomain ontology, distributed index, and service encapsulate has been well recognized. In the subsequent studies, knowledge semantic retrieval [7], extended algorithm [8], and word segmentation model [9] are used to propose the knowledge retrieval model that can meet the goal of innovative design, and the prototype system is built [10].

In the cloud mode, combination complexity and uncertainty of the knowledge resources may be unavoidable due to the complex resource types, the various combination, or unmeasured quality evaluation that affects the combination performance. Consequently, how to build and choose the best service composition scheme is of great importance for knowledge service design. To address this issue, some efforts have been advanced to exploit the strategy and algorithm for combination of manufacturing resources and computing resources. In particular, various well-known approaches, for example, process oriented service composition, semantic oriented service composition, and service composition based on service quality optimization, are reviewed and compared in terms of performance. It is shown in [11] that a flexible and light weight workflow framework based on the object modeling system to deploy and execute data-centric workflow in a decentralized manner across multiple distinct cloud resources, avoiding limitations of all data passing through a centralized engine. The author of [12] introduced fundamental guidelines for the automated composition of activities, where the propositions further encompass means to objectively derive these properties based on structure data-flow relation in a PDM. The presented method overcomes the impediments of time consumption and extensive domain knowledge. In [13], ontology of service functional logic and its subtyping rules were constructed to support precise relationship determining among services in cloud computing environment, for which the uncertainty stabilization approached to improve the practical performance of service composition.

It is noted that such composition methods are valid for cloud resource and manufacturing resource without considering service performance and thus a combination strategy with global service composition performance was studied based on the perception of service quality. Following this idea, the author [14] proposed a multiobjective service sharing optimization model considering the attributes of

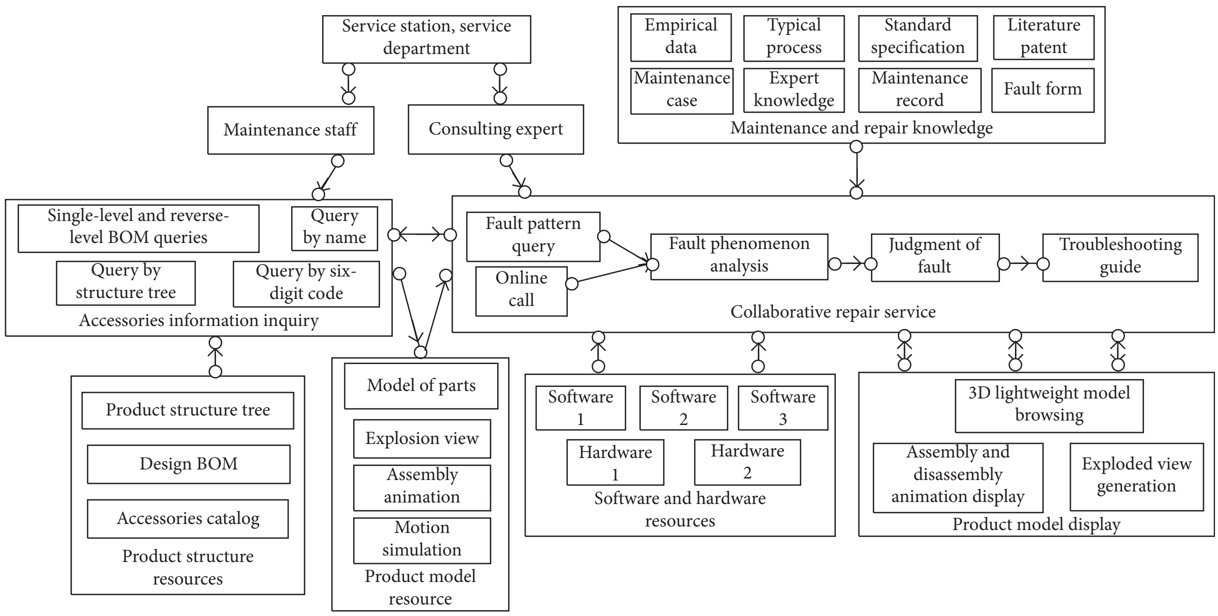


Figure 1: The coupling relationship between knowledge resources and business processes.

throughput, latency, and cost, simultaneously, and a particle swarm optimization-based multiobjective algorithm is proposed with deployment strategy to map a particle position to a resource sharing scheme composition. Although it was proved that the service performance converge to a neighborhood of their true values, the complexity, dynamics, and uncertainty of service composition should be conducted. Besides the swarm intelligence optimization algorithm, the deep reinforcement learning [15], and mining strategy of association rules [16] are also exploited to optimize the service composition. In the latest work, some studies optimized service composition on the basis of task requirements to solve the problem of multitask corresponding multiservice selection [17].

On the other hand, it is noted that the vast majority of knowledge service methods assume that knowledge resource and business process are separated or business process oriented knowledge resource retrieval mainly depend on keywords and semantics matching so that repeated retrieval, analysis, and evaluation must be used to acquire accurate knowledge resources. And most of the aforementioned results rarely study how to encapsulate resources and business process as a service and push the corresponding knowledge resources related to business process according to the requirements of tasks. This may result in lower availability of knowledge resource when multiple knowledge resource elements are needed to collaborative service in the process of new product development.

This paper is concerned with service-oriented encapsulation of knowledge resources and business processes, thus transforming knowledge into a resource or product and finally into economic benefits. Consequently, this study is to abstract, encapsulate, optimize, and compose all kinds of knowledge resources and combine them with business processes to form a knowledge service so as to provide intelligent support for product development process. Following this idea, this paper is organized as follows: the OCF is constructed in Section 2. The problem to be studied and the encapsulation method are proposed in Section 3. Section 4 is devoted to establish the compositional mathematical model and optimization method; simulation results are given in Section 5 and conclusions are made in Section 6.

3. OCF of Knowledge Resource Service

In this paper, we first construct the optimal composition framework (OCF) of knowledge resource service, as shown in Figure 2. The main idea of OCF is all kinds of knowledge resources in business process, such as tools, methods, and parameters, are encapsulated to the reusable service components, and then the knowledge service with specific function are formed by composing such components. Following this idea, the OCF was designed in three parts including service decomposition, component selection, and optimal composition.

Within the task decomposition section, typical business processes of enterprises are sorted out, where knowledge resources such as tools, methods, and parameters used in typical business activities can be obtained. Moreover, the tasks of each stage are different in the business process, which needs knowledge resources in different areas. For example, in the design stage, not only all kinds of documents and design standard resources are needed, but also parametric design templates encapsulated with experience knowledge are needed to improve design efficiency and knowledge reuse. Motivated by this fact, we organize all kinds of knowledge resources through business activity and encapsulate them into a knowledge service, thus business activities completed by a single knowledge service are dynamic node of the whole business process and the output of the former knowledge service is generally the input of the next knowledge service. Consequently, product life cycle

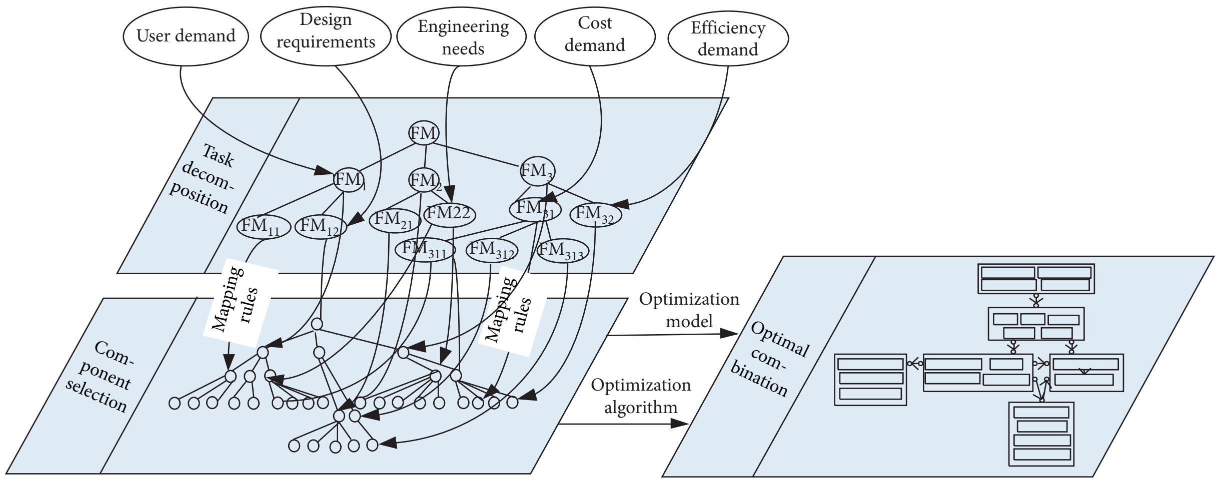


Figure 2: Optimal composition framework.

business process can be formed though dynamic integration of knowledge service.

In the component encapsulation section, normative and solidifying knowledge resources (e.g., empirical parameters, tools, and methods) in the process was encapsulated into knowledge service components with independent functions, which encapsulates the intermediate process and provides input and output parameter interface to the outside. When users call the corresponding knowledge service, they only need to input relevant parameter requirements to complete the task efficiently.

Consequently, knowledge service is the smallest unit in business process. Only through dynamic integration can it constitute the whole life cycle business activities. Following this idea, a mathematical model for multicomponent combination optimization was designed in the optimal composition section, which converts the multicomponent combination problem into a multiobjective optimization problem with constraints, where a heuristic algorithm was constructed and utilized to composite the multigranularity service component dynamically and robustly.

4. Encapsulation of Resource Service Component

4.1. Business Processes Decomposition. Aiming at knowledge-intensive business processes in enterprises, the typical solidified resource service activities FM_i are sorted out, as shown in Figure 3, where we use the idea of scenario analysis [18] for reference and regard the business process decomposition as the multilevel attribute subdivision of two dimensions: task unit and business personnel who complete the task, e.g., the task unit dimension, needs to consider such attributes as task basic information, task related knowledge, task relationship in the process, task stability, and the set of personnel performing tasks. And the business personnel dimension needs to consider the attributes of user's basic information, knowledge preference, skill level, function preference, task execution set, etc. It is noted that the dimensions should be divided and refined according to the characteristics of the business process in the process of the actual business decomposition. In Figure 3, i is the ith dynamic service activity node in the business process and the process knowledge is classified into four aspects of specialty, method, tool, and task. Then, the dynamic knowledge element DKE_{ij} is constructed, where j is the jth resource service unit corresponding to the ith service activity node. In particular, the specialty is to express the category of specialty belonging to knowledge element and the method means how to implement a process application using the rules, reasoning, and description involved in knowledge elements. In addition, the tools are mainly used to describe basic tools such as software and programs involved in knowledge elements and the task is mainly used to express the business stages where knowledge elements can be applied. Thus, the knowledge resources such as tools, methods, and parameters used in the typical business activities can be obtained through decomposing the business process, which will prepare for the subsequent encapsulation of knowledge service components, and $\mathrm{CS}\_{\mathrm{DKE}_{ij}}$ is the number of alternative components corresponding to the jth resource service unit of the ith service activity node.

4.2. Resource Service Component Encapsulation

4.2.1. Assumption and Definition. For the above purpose, the following assumption and definitions are used.

Assumption 1. DKE_{ij} is the minimum service unit of the resource service process, which contains knowledge resources such as standards, specifications, empirical parameters, tools, and methods and can be encapsulated as knowledge service components with independent functions through templates.

Definition 1. f is the dynamic requirement characteristics of users for knowledge resource, where the variability $V(f)$ of f depends on the number of variable requirement attributes and reuse frequentness. That is, the more

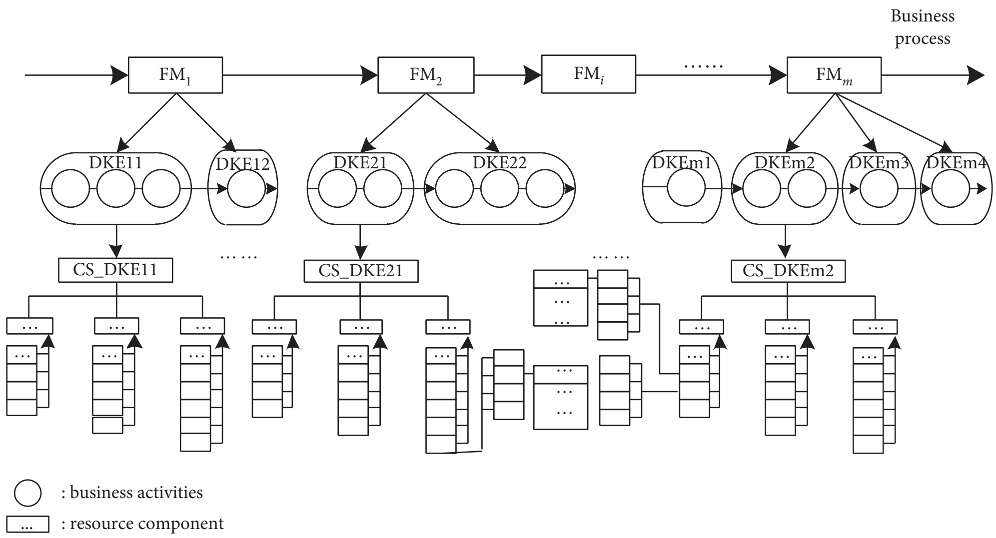


Figure 3: Business process decomposition.

frequently knowledge service elements change, the more unstable their performance is and the lower the reuse frequentness is.

Definition 2. Service stable region is a local space of resource service process, where the space is consisted of different DKE_{ij}, and the maximum service stable region is directly mapped to a component. The variation d_f of requirement attributes is restricted by the trust requirement region h_k in the demand library. For this purpose, the stable region sd is considered as min $sd^{(k)}(d_f) = V(f_k) + p_k d_f + 0.5 d_f Q_k d_f$, where k is the kth stable domain of knowledge service unit, p_k is the changing gradient of service requirements, and Q_k is the requirement trace matrix; thus, the following properties hold:

① $\forall f_i \in sd$, $V(f_i) \le \delta$, and $\|d_{f_i}\| \le h_{ki}$

② $\forall f_j \notin sd$, $V(f_j) > V(f_i)$

4.2.2. Component Encapsulation Algorithm. According to the encapsulation algorithm (Algorithm 1), the number of components F_c can be defined as $F_c = \sum_{i=1}^{m} P_i \sum_{j=1}^{P_i} C_{ij}$, where P_i is the number DKE for the i resource service module and C_{ij} is the component sequence of the jth DKE in the ith resource service module.

5. Optimized Composition for Resource Service Component

5.1. Problem Formulation. In this section, we first address the comprehensive target including performance, cost, and efficiency for composing multiple RSC with user requirements, thus multicomponent composition can be specified as a multiobjective optimization problem. For this purpose, the following mathematic model is considered:

$$\begin{cases} \max P_C = \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} \sum_{k=1}^{K} \lambda_k \mu_{ijl,k}, \\ \min C_g = \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} C_{Cijl} + \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} C_{Rijl} + \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} C_{Dijl}, \\ \max E_g = \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} E_{Cijl} + \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} E_{Rijl} + \sum_{i=1}^{m} \sum_{j=1}^{p_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} E_{Dijl}, \\ C_g \le C_{\max}, \\ \gamma_{ijl} \in \{0,1\}, \\ \sum_{i=1}^{m} \sum_{j=1}^{\rho_i} \sum_{l=}^{C_{ij}} \gamma_{ijl} \le 1, \\ \sum_{k=1}^{K} \lambda_k = 1, \\ E_l \le E_{\max}, \end{cases} \tag{1}$$

where P_c is the comprehensive performance evaluation index of RSC composition, which is evaluated by user satisfaction measurement. The performance includes the ability of composite components to service a business process and the degree of satisfaction to user needs, design needs, and engineering needs. P_c can be expressed as $N = \{N_1, N_2, \ldots N_K\}^T, k = 1, 2, \ldots, K$, where N_k is the kth performance of composite components and the corresponding weight of each performance is $\lambda_N = (\lambda_1, \lambda_2, \ldots, \lambda_K)^T$. In addition, γ_{ijl} corresponds to a binary decision variable, which indicates that the component is selected when it is 1, otherwise it is not selected. $\mu_{ijl,k}$ represents the correlation between the lth component instance of the DKE_j for the FM_i and the kth performance of the RSC, which can be a set of quantified fuzzy comments.

C_g is the total cost of RSC composition evaluated using cost calculation models, which includes customization cost C_c, reuse cost C_R, and development cost C_D, where $C_C =$

Input: the requirement attribute spaces $\Gamma(icbm)$ and resource service spaces $\Psi_i(icbm)$, variability of requirement attributes $V(f)$.
Output: the component sequence $C_{\_\mathrm{DKE}_{ij}}$ of different DKE_{ij}.
Step 1: set $sd = \Phi$, $CS\_DKE_{ij} = \Phi$;
Step 2: $\forall\Psi_i(icbm) \subseteq \Gamma(icbm)$, $\forall f \in \Psi_i(icbm)$;
Step 3: do {
  Searching for the variable threshold δ to meet min $sd^{(k)}(d_f) = V(f_v) + pd_f + 0.5d_f Q d_f \wedge sd \subseteq \Psi_i(icbm)$
    If $(V(f) \le \delta)$
      {
   Incorporate f and its child attributes $child(f)$ into sd;
      For $(\forall f_i \in \{f\} \cup child(f))$
        {
        Set $flag(f_i) = 1$, that is, f_i has been mapped as an independent component or part of a component;
        Go to step 4;
        }
      }
    Else if $(V(f) > \delta)$
      {
      Set $flag(f) = 0$, that is, f_i cannot be directly mapped as a component;
      Go to step 3;
      }
    } until (nonexistent such f)
   Go to step 5;
Step 4: if $(sd \ne \Phi)$
   {
  All the characteristic of *sd could be encapsulted into* the different granularity components $C_{C_{ij}}$ with the same functionality but different performance.
    Join $C_{C_{ij}}$ into $\mathrm{CS}_{\_\mathrm{DKE}_{ij}}$;
    Set $sd = \Phi$, go to step 3;
   }
    Else
      Go to step 3;
Step 5: for $(\forall f \in \Gamma(icbm) \wedge flag(f) = 0 \wedge child(f) = \Phi)$
    {
  Map f into $C_{C_{ij}}$, and add $C_{C_{ij}}$ into $\mathrm{CS}_{\_\mathrm{DKE}_{ij}}$;
      Set $flag(f_i) = 1$;
    }
Step 6: output $\mathrm{CS}_{\_\mathrm{DKE}_{ij}}$;
    If $(\mathrm{CS}_{\_\mathrm{DKE}_{ij}} = \Phi)$
      {
    Develop again;
      }

Algorithm 1

$\{C_{C111}, \ldots, C_{Cijl}, C_{CmP_iC_{ij}}\}^T$ is the customization cost matrix, $C_R = \{C_{R111}, \ldots, C_{Rijl}, C_{RmP_iC_{ij}}\}^T$ is the reuse cost matrix, and $C_D = \{C_{D111}, \ldots, C_{Dijl}, C_{DmP_iC_{ij}}\}^T$ is the development cost matrix, C_{Cijl}, C_{Rijl}, and C_{Dijl} are the customization cost, reuse cost, and development cost of the lth component instance of the DKE_j for the FM_i, respectively.

E_g is the total efficiency of RSC composition evaluated through service platform, which includes customization efficiency, reuse efficiency, and development efficiency, where $E_C = \{C_{C111}, \ldots, C_{Cijl}, C_{CmP_iC_{ij}}\}^T$ is the customization efficiency matrix, $E_R = \{C_{R111}, \ldots, C_{Rijl}, C_{RmP_iC_{ij}}\}^T$ is the reuse efficiency matrix, and $E_D = \{C_{D111}, \ldots, C_{Dijl}, C_{DmP_iC_{ij}}\}^T$ is the development efficiency matrix, E_{Cijl}, E_{Rijl}, and E_{Dijl} are the customization efficiency, reuse efficiency, and development efficiency of the lth component instance of the DKE_j for the FM_i, respectively.

$C_g \le C_{\max}$ is the cost constraint, which cannot exceed the specified cost threshold. $\gamma_{ijl} \in \{0, 1\}$ is the decision variable constraints, and its value is only 0 or 1. $\sum_{i=1}^{m}\sum_{j=1}^{\rho_i}\sum_{l=}^{C_{ij}}\gamma_{ijl} \le 1$ is a single constraint, that is, only one can be selected for component sequences of each DKE. $\sum_{k=1}^{K}\lambda_k = 1$ is the weight constraint, that is, the sum of weight vectors corresponding to different performances is equal to 1. $E_l \le E_{\max}$ is the efficiency constraint, and the total efficiency should not be lower than the prescribed minimum efficiency.

5.2. Optimization Algorithm

5.2.1. The FMEA Model of Component Composition. The multicomponent composition presented in this paper is formulated as a large-scale nonlinear multiple objective optimization problem with a considerable number of equality and inequality constraints. Motivated by this fact, the analysis of a fuzzy matter-element [19, 20] is considered, which transformed the multiobjective optimization into a single-objective optimization. That is, the fuzzy matter-element model of the multi component composition in equation (1) can be written as follows:

$$\tilde{CR}_k = \begin{bmatrix} & P \\ p_1 & \mu(f_1(X)) \\ p_2 & \mu(f_2(X)) \\ \cdots & \cdots \\ p_k & \mu(f_k(X)) \end{bmatrix}, \quad (2)$$

where $\tilde{CR}_k$ is a component composition matter-element of k dimensions; P is the name of the composition scheme; p_i is the name of the ith subobjective; Xi is the ith composition variable; and $\mu(f_i(X))$ is the excellence of degree for the composition scheme P ith subobjective p_i corresponding to $f_i(X)$, $\mu(f_i(X)) \in [0,1]$.

If $f_i(X)$ is an ordinary mathematical function expression, one way to compute the excellent dependent degree function $\mu(f_i(X(t))) \in [0,1]$ is by using the model as follows:

$$\mu(f_i(X)) = \begin{cases} 1, & f_i(X) \le f_{i\min}, \\ \dfrac{f_{i\max} - f_i(X)}{f_{i\max} - f_{i\min}}, & f_{i\min} \le f_i(X) \le f_{i\max}, \\ 0, & f_i(X) \ge f_{i\max}. \end{cases} \quad (3)$$

It is well known that the weight of respective subobjective may be different for the component composition, so the weight R_λ corresponding to these objectives form a weight set as the following formula:

$$R_\lambda = \begin{bmatrix} & p_1 & p_2 & \cdots & p_k \\ \lambda_i & \lambda_1 & \lambda_2 & \cdots & \lambda_k \end{bmatrix}, \quad (4)$$

where R_λ is a weight compound matter-element of the composition scheme subobjective and $\lambda_i (i = 1, 2, \ldots, k)$ is the ith objective weight.

Consequently, the performance of component composition can be evaluated using degree of correlation, and the larger the degree of correlation, the better the composition scheme. The correlation degree function K is designed as follows:

$$K(X(t)) = \sum_{i=1}^{k} \mu(f_i(X(t)))\lambda_i. \quad (5)$$

Then, it is possible to transform the nonlinear multiple objective optimization problem in equation (1) into the single-objective fuzzy matter-element optimization problem. Consequently, it satisfies

$$\begin{cases} \text{find } X(t) = (x_1(t), x_2(t), \ldots, x_m(t))^T \\ \max \quad k(X(t)) \\ \text{s.t.} \quad g_j(X(t)) \le 0, \quad j = 1, 2, \ldots, J. \end{cases} \quad (6)$$

5.2.2. FTPSO Algorithm. One of the major drawbacks of the PSO proposed by Kennedy and Eberhart [21] is the lack of diversity of the swarm, which results in the convergence of the swarm to local optima. Then, the main reasons are that the personal best vector, the global best vector, and their fitness values are all stored in the memory of each particle, and such memory will be not updated until a new vector location with higher fitness was found. That is, it is difficult for particles to detect the change of the latest extremum in a convergence process. Motivated by this fact, we presented a flexible tracking particle swarm optimization (FTPSO), which improves the population's ability to perceive and respond to changes in the external environment. For this purpose, the following rules and definitions are considered.

Rule 1. Each particle detects its individual extreme value before velocity updates to perceive environmental changes.

Rule 2. By comparing with the global optimal fitness and the average fitness of the population, the leading particles and the eliminated particles are judged and different mutation probabilities are given to judge their fitness to the changing environment. Thus, the dynamic update response mode is introduced to update the particles gradually according to their fitness.

Definition 3. In evolutionary process, the particle is regarded as a leading particle with high judgment if the number of times will reach a certain value, where the distance between the individual extreme value ($f_{P\text{new}}$) of the original position of any particle and the global extreme value ($f_{X\text{new}\max}$) of the new position for the population is less than the given threshold.

Definition 4. In evolutionary process, the particle is regarded as an eliminated particle if the number of times will reach a certain value, where the individual extreme value ($f_{P\text{new}}$) of the original position of any particle is less than the average extreme value ($\overline{f}_{X\text{new}}$) of the new position for the population.

Consequently, mutation operation is carried out according to the flexible mutation probability given in formula (7). Thus, the historical optimum position of the particle can be initialized by the new position generated by the particle mutation:

$$P_{fm} = \begin{cases} \tau_1 \cdot \dfrac{\left| f_{X\text{new max}} - f_{P\text{new}} \right|}{\varepsilon}, & \left[\left| f_{X\text{new max}} - f_{P\text{new}} \right| \le \varepsilon \right]^{\text{iterations}} \xrightarrow{\text{upto}} \alpha, \\ \tau_2 \cdot \dfrac{\left| f_{X\text{new max}} - f_{P\text{new}} \right|}{f_{X\text{new max}} - \overline{f}_{X\text{new}}}, & \overline{f}_{X\text{new}} < f_{P\text{new}} \le f_{X\text{new max}}, \\ \tau_3, & \left[f_{P\text{new}} \le \overline{f}_{X\text{new}} \right]^{\text{iteration times}} \xrightarrow{\text{upto}} \beta, \end{cases} \quad (7)$$

where τ_1, τ_2, and τ_3 is a constant less than 0.5, which is used to constrain the flexible mutation probability P_{fm} to 0.0–0.5. ε is the given threshold, α and β are the given number of times, respectively. $f_{X\text{new max}}$ is the new global optimal fitness, $f_{P\text{new}}$ is the particle fitness value for mutation according to mutation probability, $\overline{f}_{X\text{new}}$ is the average fitness of the population, and p_{fm} is a flexible mutation probability. In addition, the values of ε, α, and β can be adjusted appropriately according to the actual demand, so that the convergence rate of the population can be controlled more flexibly.

According to the adaptability of particles to changes in external environment, the self-renewal ability of particles is gradually improved as follows:

$$P_i(t+1) = \begin{cases} P_i(t)\left(1 - P_{fm}\right), & f_{X\text{new}} \le f_{P\text{new}}\left(1 - P_{fm}\right), \\ X_i(t+1), & f_{X\text{new}} > f_{P\text{new}}\left(1 - P_{fm}\right). \end{cases} \quad (8)$$

According to the above improvement ideas, the FTPSO algorithm based on FMEA can be expressed as follows:

Step 1 (initialize the particle population).

Initialize the population space of the component composition scheme P_i $(i = 1, 2, \ldots, m)$ by choosing from the composition problem space with k subobjectives randomly. Within this space, $T_{\max}$ is the maximum generation, t is the number of current iterations, and c_1 and c_2 are the acceleration coefficients. Then, the space of component composition scheme can be represented as the compound matter-element form of $m \times k$:

$$\widetilde{R}_{mk} = \begin{bmatrix} & P_1 & P_2 & \cdots & P_m \\ p_1 & \mu\left(f_{11}(X)\right) & \mu\left(f_{21}(X)\right) & \cdots & \mu\left(f_{m1}(X)\right) \\ p_2 & \mu\left(f_{12}(X)\right) & \mu\left(f_{22}(X)\right) & \cdots & \mu\left(f_{m2}(X)\right) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ p_k & \mu\left(f_{1k}(X)\right) & \mu\left(f_{2k}(X)\right) & \cdots & \mu\left(f_{mk}(X)\right) \end{bmatrix}, \quad (9)$$

where $\mu(f_{ij}(X))$ $(i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, k)$ is the jth object optimal membership degree for the ith scheme, P_i is the name of the ith composition scheme, and p_j is the jth composition subobject.

Step 2 (calculating fitness function).

It is well known that fuzzy matter-element correlation function can be served as the fitness function [22]. According to (5), fitness function can be derived as

$$f_i(X) = k_i(X) \cdot \text{pun}(X), \quad i = 1, 2, \ldots, m, \quad (10)$$

where $k_i(X)$ is the fuzzy matter-element correlation function and $\text{pun}(X)$ is a penalty function.

Step 3. In the following, $f_{X\text{new max}}$ and $\overline{f}_{X\text{new}}$ can be calculated.

Step 4 (assessment of particle adaptability).

Then, we assess the adaptability of each particle using flexible mutation probability represented in (7), where the $f_{P\text{new}}$ will be updated in each iteration.

Step 5. It is now ready to update the P according to formula (8).

Step 6. Finally, to contrast the current fitness value with the population previous optimal, we set *gbest* to the current particle's array value if it is better than *gbest*.

Step 7. Calculate and update particle's velocity and position with each particle.

Step 8. Continue to circulate from the second step until the maximum iteration criterions algebra is satisfied.

6. A Case Study: Maintenance Resources Service

In this section, a resources service case for auto parts are given to validate the optimization model and method proposed in Sections 4 and 5. Firstly, according to the idea of business processes decomposition in Section 4.1, the maintenance process of auto parts are decomposed into five business feature spaces, that is, $\Gamma(\text{icbm}) = \{\Psi_i(\text{icbm}) \mid i = 1, 2, \ldots, 5\}$ = {maintenance case training, maintenance knowledge reasoning and query, maintenance tool pushing, maintenance process guidance, maintenance collaborative interaction}. Then, the dynamic knowledge element DKE_{ij} is constructed, as shown in Table 1, where the component sequences can be encapsulated using the selection method in Section 4.2, and the components for different sequences are given in Table 2. Moreover, the reuse efficiency and cost of different component sequences can be obtained through the performance analysis.

Table 1: Component sequences for $CS_{\_DKE_{ij}}$.

Ψ_i (icbm)	Name	DKE_{ij}	$CS_{\_DKE_{ij}}$	Included component	Reuse efficiency	Reuse cost
Ψ_1 (icbm)	Maintenance training	Maintenance knowledge training	1	C_{11} C_{12} C_{13} C_{21} C_{22}	0.04	23.54
			2	C_{11} C_{12} C_{14} C_{21} C_{24}	0.05	27.93
			3	C_{35} C_{36} C_{37} C_{311}	0.06	32.46
		Maintenance case training	1	C_{14} C_{15} C_{16} C_{24} C_{25}	0.04	24.57
			2	C_{14} C_{15} C_{16} C_{21} C_{24}	0.05	28.32
			3	C_{310}	0.07	33.65
Ψ_2 (icbm)	Maintenance knowledge, reasoning, and query	Maintenance knowledge and tool reasoning	1	C_{17} C_{27} C_{28}	0.05	23.44
			2	C_{18} C_{28} C_{29}	0.06	25.26
			3	C_{18} $C_{211}C_{212}$	0.06	24.53
		Maintenance knowledge and technology query	1	C_{111} C_{112} C_{213} C_{214}	0.05	26.21
			2	C_{111} C_{22} C_{23}	0.07	31.13
Ψ_3 (icbm)	Maintenance tool pushing	Maintenance tool information pushing	1	C_{211} C_{222}	0.05	25.33
			2	C_{18} C_{19} C_{110} C_{219}	0.06	26.45
		Maintenance tool information customization	1	C_{112} C_{213} C_{221} C_{212}	0.06	19.78
			2	C_{111} C_{213} C_{214} C_{223}	0.05	21.34
Ψ_4 (icbm)	Maintenance process guidance	Maintenance knowledge and case query	1	C_{116} C_{14} C_{16} C_{24} C_{25} C_2 C_{111} C_{111}	0.04	27.45
			2	C_{116} C_{14} C_{15} C_{23} C_{24}	0.06	28.25
			3	C_{34} C_{36} C_{38} C_{39}	0.08	34.16
		Maintenance posture and tool pushing	1	C_{14} C_{15} C_{16} C_{113}	0.05	25.67
			2	C_{21} C_{22} C_{23} C_{28} C_{29}	0.05	26.23
			3	C_{14} C_{15} C_{16} C_{111}	0.07	32.61
Ψ_5 (icbm)	Maintenance collaborative interaction	Synergetic annotation	1	C_{213} C_{214} C_{225} C_{226}	0.05	27.18
			2	C_{31}	0.07	33.24
		Failure consultation	1	C_{15} C_{16} C_{111} C_{213}	0.05	27.54
			2	C_{15} C_{16} C_{31} C_{33} C_{38}	0.07	31.52

According to the composition algorithm in Section 3, the RSC for maintenance process were composed, and the composition process is presented below:

(1) The performance requirements of the RSC for maintenance can be expressed using the natural language, that is, $P = \{P_1, P_2, \ldots, P_8\}$ = {stability, maintainability, security, reliability, scalability, reusability, flexibility, adaptability}, using AHP to determine the weights of these performance for $\lambda_1 = 0.154$, $\lambda_2 = 0.120$, $\lambda_3 = 0.178$, $\lambda_4 = 112$, $\lambda_5 = 0.177$, $\lambda_6 = 0.089$, $\lambda_7 = 0.141$, and $\lambda_8 = 0.029$. Moreover, the 5 levels' quantized values are expressed as strong, less strong, medium, weak, and irrelevant and the measured degree of relative value is defined as 9, 7, 5, 3, and 1. As shown in Table 3, the correlation matrix between component instance and composition performance in $CS_{\_DKE_{ij}}$ can be constructed according to formula (5).

(2) In the process of component composition, the weight of each subobjective including composition performance $P_C(X)$, composition efficiency $E_C(X), E_R(X), E_D(X)$, and composition cost $C_C(X), C_R(X), C_D(X)$ may be different, and then weight R_λ corresponding to these objectives form a weight set as follows:

Table 2: List of resource service component.

No.	Name
C_{11}	Maintenance of common sense
C_{12}	Assembly scheduled maintenance
C_{13}	Assembly regular inspection
C_{14}	Maintenance records
C_{15}	Fault form
C_{16}	Maintenance case
C_{17}	Product structure tree
C_{18}	BOM table
C_{19}	Inventory information
C_{110}	Supplier information
C_{111}	Three-dimensional model of product
C_{112}	Exploded view
C_{113}	Assembly animation
C_{114}	Motion simulation
C_{115}	The parts catalogue
C_{116}	Process knowledge of products
C_{21}	General maintenance inquiries
C_{22}	Maintaining common sense queries
C_{23}	Fault shape query
C_{24}	Maintenance record query
C_{25}	Maintenance case analysis
C_{26}	Troubleshooting program query
C_{27}	Query by structure tree
C_{28}	Number by name
C_{29}	Single-level BOM check
C_{211}	Single-level countercheck

Table 2: Continued.

No.	Name
C_{212}	Multilevel BOM check
C_{213}	Multilevel countercheck
C_{214}	Three-dimensional model browsing
C_{221}	Supply information query
C_{222}	Inventory information query
C_{223}	Confirm order
C_{224}	Generating order
C_{225}	Online call
C_{226}	Synergetic annotation
C_{31}	Collaborative calling
C_{32}	Product structure
C_{33}	Product 3D model browsing
C_{34}	Vehicle spectrum
C_{35}	Vehicle configuration
C_{36}	Maintenance history
C_{37}	Assembly maintenance
C_{38}	Troubleshooting
C_{39}	Diagnosing trouble
C_{310}	Illustrating product
C_{311}	Installation and dismantling tutorials
C_{312}	Training course
C_4	Staff
C_5	Organization
C_{61}	Software resources
C_{62}	Hardware resources
C_{63}	Environment of application service
C_{71}	Training of maintenance

$$R_\lambda = \begin{bmatrix} & h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 \\ \tau_I & 0.192 & 0.121 & 0.165 & 0.093 & 0.154 & 0.142 & 0.133 \end{bmatrix}. \tag{11}$$

(3) The composition scheme space like as the compound matter-element form of 50×7 was considered as the initial population of particle:

$$R_{50\times 7} = \begin{bmatrix} & P_1 & P_2 & \cdots & P_{50} \\ P_C & \mu(f_{11}(X)) & \mu(f_{21}(X)) & \cdots & \mu(f_{50,1}(X)) \\ E_C & \mu(f_{12}(X)) & \mu(f_{22}(X)) & \cdots & \mu(f_{50,2}(X)) \\ E_R & \mu(f_{13}(X)) & \mu(f_{23}(X)) & \cdots & \mu(f_{50,3}(X)) \\ E_D & \mu(f_{14}(X)) & \mu(f_{24}(X)) & \cdots & \mu(f_{50,4}(X)) \\ C_C & \mu(f_{15}(X)) & \mu(f_{25}(X)) & \cdots & \mu(f_{50,5}(X)) \\ C_R & \mu(f_{16}(X)) & \mu(f_{26}(X)) & \cdots & \mu(f_{50,6}(X)) \\ C_D & \mu(f_{17}(X)) & \mu(f_{27}(X)) & \cdots & \mu(f_{50,7}(X)) \end{bmatrix}. \tag{12}$$

We now calculate the fitness function $f_i(X)$ as follows:

$$f_i(X) = K_i(X) = \sum_{j=1}^{7} \nu\left(f_j(X)\right)\tau_j, \quad i = 1, 2, \ldots, 50, j = 1, 2, \ldots, 7. \tag{13}$$

In order to find the optimal composition scheme, we simulated the classical experiment after Angeline [23] and Carlisle and Dozier [24], where three kinds of goal movements such as linear, circular, and random were implemented. As to the linear motion, a constant offset for each dimension is added in the update frequency intervals; for circular motion, a circular path with a cycle of 25 updates is added to the offset in each dimension; and as to the random motion, the Gaussian random variable for each dimension is added in the update frequency interval.

Each experiment with different movement goal, step size, and update frequency has been implemented 100 times, where the APSO and FTPSO were performed with same configuration and dynamic environment. Like in Angeline's and Carlisle's experiments, the motion step size was set as 0.1 and 0.5, updating frequency for simulation was 10 iterations and the maximum generation is set to 500.

Additionally, the fundamental parameters for FTPSO are set as follows: $w = 1.1$, $c_1 = c_2 = 2$; the initial inertia weight is 0.9 descending to 0.4 linearly, acceleration coefficients is $c_1 = c_2 = 1.49$, and the threshold value is $\varepsilon = 0.01$ in calculating the mutation probability p_m of FTPSO; the given iteration times are $\alpha = 15$, $\beta = 20$, and other APSO parameters were specified as reference.

Statistical results with different movement pattern are provided in Table 4, where the proposed method in this paper can achieve better performance for different movement pattern. Note that the performance of component sequence composition with FTPSO has many potential advantages in random movement pattern.

Comparison results of APSO and FTPSO were provided in Figures 4–7.

Figures 4–7 provide the simulation results, where the composition performance and tracking process are estimated for each working condition. From these figures, one can observe that the adaptive mutation probability and the response mode of dynamic updating were introduced to improve the adaptability of particles for FTPSO, where the proposed algorithm can track the latest change of system extreme in the process of component composition. Otherwise, the APSO can track the moving solutions in more than one hundred iterations with low accuracy due to lack of an adaptive updating mechanism for particle's *pbest*.

From the aforementioned results, one can conclude that our work specifies the service component composition problem as a multiobjective optimization problem. Thus, the relevance of the composition performance and the component instance can be estimated quantitatively. Moreover, the mathematical model of component composition can be provided using the fuzzy matter-element model, where the service component composition problem can be solved in the form of combination of qualitative and quantitative, and the multiobject composition optimal problems can be transformed into single-object optimal problems. In particular, FTPSO introduces the adaptive mutation probability and the response mode of dynamic updating, where the adaptability of particles were improved in the initial iterations.

Finally, the presented model and algorithm have been applied to the construction of the maintenance service

Table 3: Correlation matrix between component instance and composition performance for $CS_{\_DKE_{ij}}$.

	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
L_{111}	9	9	5	1	5	7	5	3
L_{112}	5	7	3	3	5	7	5	7
L_{113}	9	9	3	9	7	7	9	7
L_{121}	7	7	3	9	5	5	4	3
L_{122}	7	5	1	7	5	3	5	7
L_{123}	7	5	7	3	5	5	7	9
L_{211}	9	7	7	5	7	9	3	7
L_{212}	5	7	5	1	3	7	5	3
L_{213}	7	3	1	9	9	5	3	5
L_{221}	9	7	5	9	3	3	5	5
L_{222}	7	3	5	3	5	3	5	9
L_{223}	7	5	3	1	5	5	1	5
L_{311}	7	3	9	7	5	3	5	7
L_{312}	5	7	7	9	5	9	7	7
L_{321}	5	3	7	9	7	9	5	5
L_{322}	9	3	5	7	5	7	9	5
L_{411}	3	5	7	1	3	5	7	5
L_{412}	7	5	3	3	7	7	5	5
L_{413}	9	7	7	5	5	3	5	7
L_{421}	7	9	3	1	7	5	3	7
L_{422}	7	9	5	5	3	5	7	7
L_{423}	9	3	1	7	5	1	7	7
L_{511}	7	5	3	7	7	9	9	5
L_{512}	7	7	9	5	3	1	7	5
L_{521}	9	9	5	7	5	9	3	3
L_{522}	7	3	9	7	7	3	9	7

Table 4: Comparison of composition program.

Movement pattern\composition program		Composition program	Composite performance
Linear motion with step size 0.1	APSO	L_{111}, L_{122}, L_{211}, L_{223}, L_{312}, L_{311}, L_{411}, L_{511}	2.13
	FTPSO	L_{112}, L_{121}, L_{212}, L_{222}, L_{312}, L_{322}, L_{412}, L_{422}, L_{511}	3.85
Linear motion with step size 0.5	APSO	L_{112}, L_{123}, L_{212}, L_{223}, L_{312}, L_{412}, L_{423}, L_{521}	2.67
	FTPSO	L_{111}, L_{122}, L_{213}, L_{223}, L_{311}, L_{423}, L_{512}, L_{522}	4.12
Circular motion with step size 0.1	APSO	L_{112}, L_{121}, L_{213}, L_{221}, L_{321}, L_{411}, L_{521}	2.08
	FTPSO	L_{111}, L_{123}, L_{212}, L_{222}, L_{323}, L_{422}, L_{512}, L_{522}	4.73
Random motion with step size 0.5	APSO	L_{113}, L_{123}, L_{212}, L_{222}, L_{312}, L_{413}, L_{423}, L_{512},	1.78
	FTPSO	L_{111}, L_{122}, L_{211}, L_{221}, L_{311}, L_{322}, L_{412}, L_{421}, L_{512}, L_{521}	5.41

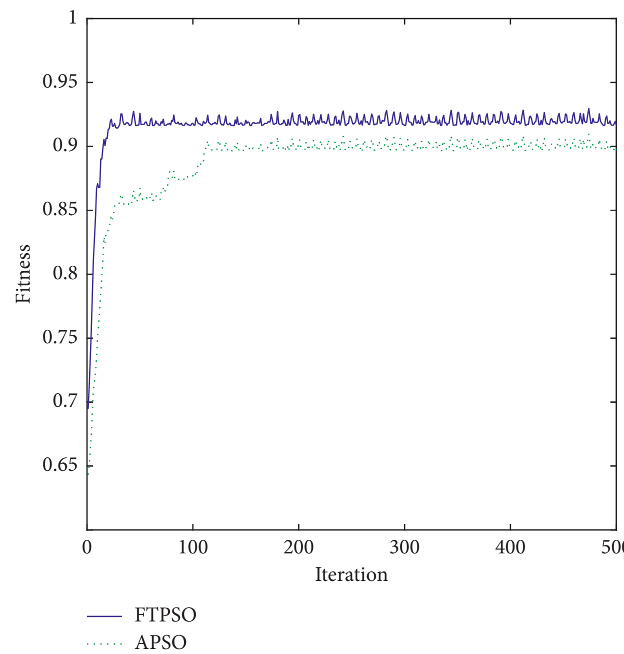


Figure 4: Performance comparison with linear motion step size 0.1 and updating frequency 10.

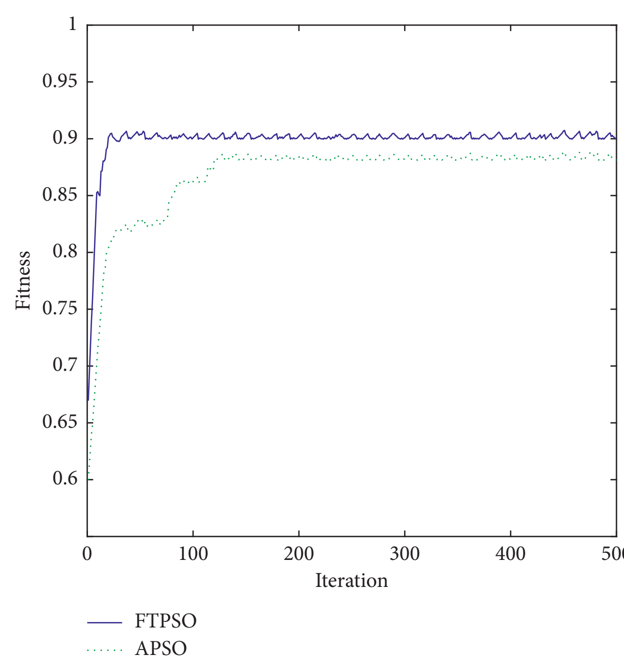


Figure 5: Performance comparison with linear motion step size 0.5 and updating frequency 10.

system for auto parts in Sichuan Truck Manufacture Plant Co., Ltd., where the architecture is divided into 5 layers including service interface layer, web service layer, business logic, data access layer, and building method layer. In addition, the information transformation and data exchange of component composition algorithm were obtained by using XML. Nowadays, the maintenance service system has been implemented in the process of auto parts' maintenance, where the user first sends the task package of "complete the maintenance of engine rocker shaft" to the platform, and the system will automatically analyze the task and obtain the information related to the task, such as service task description, specialty, undertaker, and other attribute information. Then, the system transforms "complete the

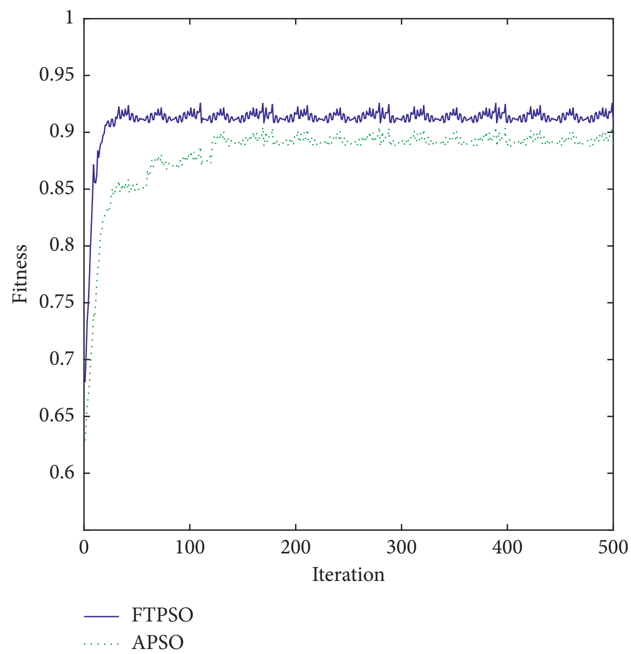


Figure 6: Performance comparison with circular motion of 25 updates.

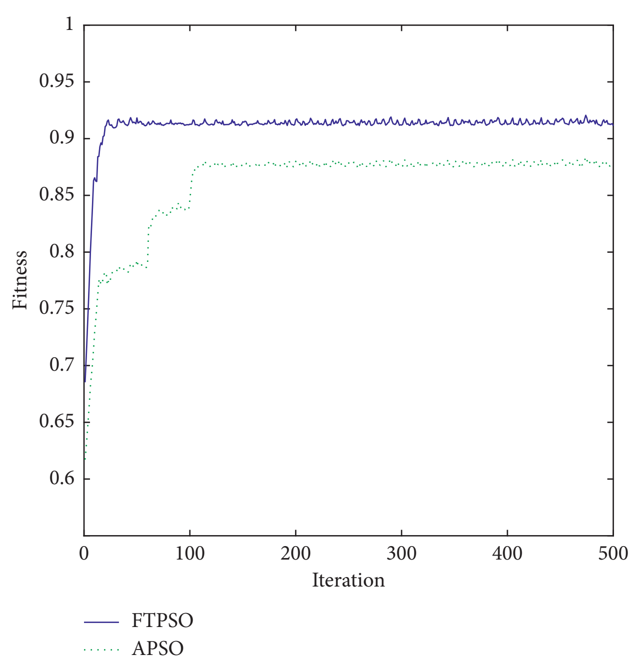
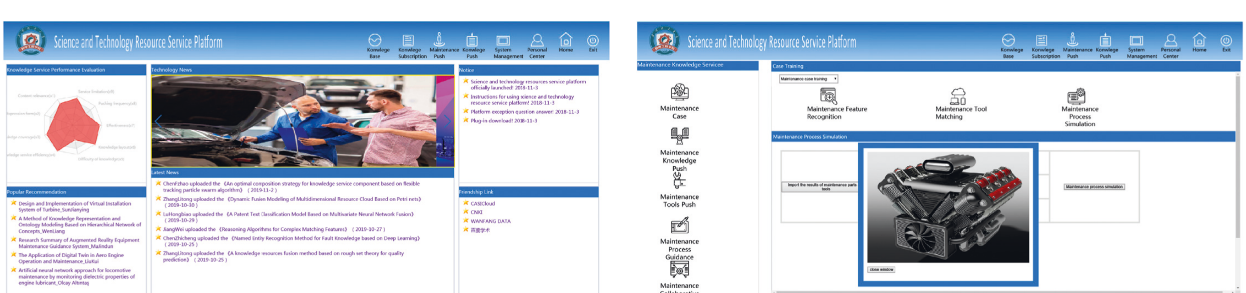


Figure 7: Performance comparison with random motion step size 0.5 and updating frequency 10.



(a) (b)

Figure 8: Interface of system implementation.

maintenance of engine rocker shaft" into "engine" and "rocker maintenance" semantic terms as the subject words of knowledge service. Users only need to input maintenance parameters step by step according to the guidance of the system, and the system automatically combines service components to generate failure causes, maintenance methods, or cases. The main service interfaces can be seen in Figure 8.

7. Conclusion

This paper addresses a combinatorial optimal strategy for knowledge resource service-oriented business process. Business processes are decomposed into dynamic knowledge elements, and all kinds of knowledge resources through business activity such as standards, specifications, empirical parameters, tools, and methods are encapsulated into knowledge service components with independent functions. In addition, a mathematical model for multicomponent combination optimization is designed, which converts the multicomponent combination problem into a multiobjective optimization problem with constraints. On this basis, we presented a combinatorial heuristic algorithm with the adaptive mutation probability and the response mode of dynamic updating so that the adaptability of particles was improved in the initial iterations to composite the multi-granularity service component dynamically and robustly. Finally, the case of component composition for maintenance resource service is studied and the experimental results show that our model and algorithm improve the efficiency and stability of the resource service system.

Planned future works and improvements include a comprehensive application for knowledge resource service of regional industrial cluster and an extension of the model and algorithm.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also form part of an ongoing study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This paper was supported by the National Key R&D Program of China (2017YFB1400301).

References

[1] F. Vendrell-Herrero, E. Gomes, O. F. Bustinza, and K. Mellahi, "Uncovering the role of cross-border strategic alliances and expertise decision centralization in enhancing product-service innovation in MMNEs," *International Business Review*, vol. 27, no. 4, pp. 814–825, 2018.

[2] A. S. A. Shukor, M. F. Mohammad, R. Mahbub, and F. Halil, "Towards improving integration of supply chain in IBS construction project environment," *Procedia—Social and Behavioral Sciences*, vol. 222, pp. 36–45, 2016.

[3] Y. Chao, Z. Yun, and T. Zhong, "Optimization model of cloud manufacturing services resource combination for new product development," *Computer Integrated Manufacturing Systems*, vol. 18, no. 7, pp. 1368–1378, 2012.

[4] D. Xue, Y.-L. Yu, G. Chen, and J. Yao, "Formation control of networked multi-agent systems," *IET Control Theory & Applications*, vol. 4, no. 10, pp. 2168–2176, 2010.

[5] L. Mu and C. K. Kwong, "A multi-objective optimization model of component selection in enterprise information system integration," *Computers & Industrial Engineering*, vol. 115, pp. 278–289, 2018.

[6] 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.

[7] D. Kyriazis, "Modelling and prediction of resources and services state evolvement for efficient runtime adaptations," *Future Generation Computer Systems*, vol. 94, pp. 1–10, 2019.

[8] X. Q. Li, H. C. Yang, S. K. Jing et al., "Knowledge service modeling approach for group enterprise cloud manufacturing," *Computer Integrated Manufacturing Systems*, vol. 18, no. 8, pp. 1869–1880, 2012.

[9] Y. C. Yin, W. G. Ding, and L. Wu, "Knowledge service method of cloud manufacturing based on uncertain rule reasoning," *Computer Integrated Manufacturing Systems*, vol. 21, no. 4, pp. 1114–1123, 2015.

[10] X. F. Xu, J. Hao, Y. R. Deng, and Y. Wang, "Design optimization of resource combination for collaborative logistics network under uncertainty," *Applied Soft Computing*, vol. 56, pp. 684–691, 2017.

[11] D.-H. Shin, K.-H. Lee, and T. Suda, "Automated generation of composite web services based on functional semantics," *Journal of Web Semantics*, vol. 7, pp. 332–343, 2009.

[12] H. van der Aa, H. A. Reijers, and I. Vanderfeesten, "Designing like a Pro: the automated composition of workflow activities," *Computers in Industry*, vol. 75, pp. 162–177, 2016.

[13] C. Huang, X. Wang, and D. Wang, "Type theory based semantic verification for service composition in cloud computing environments," *Information Sciences*, vol. 469, pp. 101–118, 2018.

[14] C. Zhang, J. Ning, J. Wu, and B. Zhang, "A multi-objective optimization method for service composition problem with sharing property," *Swarm and Evolutionary Computation*, vol. 49, pp. 266–276, 2019.

[15] J.-W. Liu, L.-Q. Hu, Z.-Q. Cai, L.-N. Xing, and T. Xu, "Large-scale and adaptive service composition based on deep reinforcement learning," *Journal of Visual Communication and Image Representation*, vol. 65, Article ID 102687, 2019.

[16] C. Jatoth, G. R. Gangadharan, and R. Buyya, "Optimal fitness aware cloud service composition using an adaptive genotypes evolution based genetic algorithm," *Future Generation Computer Systems*, vol. 94, pp. 185–198, 2019.

[17] M. Yuan, Z. Zhou, and X. Cai, "Service composition model and method in cloud manufacturing," *Robotics and Computer Integrated Manufacturing*, vol. 61, pp. 1–13, 2019.

[18] F. Zhang and L. Li, "Research on knowledge push method for business process based on multidimensional hierarchical context model," *Journal of Computer-Aided Design & Computer Graphics*, vol. 29, no. 4, pp. 751–758, 2017.

[19] W. Cai, "Extension theory and its application," *Chinese Science Bulletin*, vol. 44, no. 17, pp. 1538–1548, 1999.

[20] Y. W. Zhao and G. X. Zhang, “A new integrated design method based on fuzzy matter-element optimization,” *Journal of Materials Processing Technology*, vol. 129, no. 1–3, pp. 612–618, 2002.

[21] J. Kennedy and R. C. Eberhart, “Particle swarm optimization,” in *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.

[22] T. Loukopoulos and I. Ahmad, “Static and adaptive distributed data replication using genetic algorithms,” *Journal of Parallel and Distributed Computing*, vol. 64, no. 11, pp. 1270–1285, 2004.

[23] P. J. Angeline, “Tracking extrema in dynamic environments,” in *International Conference on Evolutionary Programming VI*, pp. 335–345, Springer-Verlag, Berlin, Germany, 1997.

[24] A. Carlisle and G. Dozier, *Tracking Changing Extrema with Adaptive Particle Swarm Optimizer*, World Automation Congress, Orlando, FL, USA, 2002.


Advances in
Operations Research
Advances in
Decision Sciences
Journal of
Applied Mathematics
The Scientific
World Journal
Journal of
Probability and Statistics
International
Journal of
Mathematics and
Mathematical
Sciences
Hindawi
Submit your manuscripts at
www.hindawi.com
Journal of
Optimization
International Journal of
Engineering
Mathematics
International Journal of
Analysis
Journal of
Complex Analysis
Advances in
Numerical Analysis
Mathematical Problems
in Engineering
International Journal of
Differential Equations
Discrete Dynamics in
Nature and Society
International Journal of
Stochastic Analysis
Journal of
Mathematics
Journal of
Function Spaces
Abstract and
Applied Analysis
Advances in
Mathematical Physics