Mathematical Biosciences and Engineering

MBE, 17(6): 6838–6872.
DOI: 10.3934/mbe.2020355
Received: 30 June 2020
Accepted: 23 September 2020
Published: 09 October 2020

http://www.aimspress.com/journal/MBE

Research article

A HRGO approach for resilience enhancement service composition and optimal selection in cloud manufacturing

Hao Song[1,2], Xiaonong Lu[1,2,*], Xu Zhang[1,2], Xiaoan Tang[1,2] and Qiang Zhang[1,2]

[1] School of Management, Hefei University of Technology, Hefei 230009, China
[2] Key Laboratory of Process Optimization and Intelligent Decision-making, Ministry of Education, Hefei 230009, China

* **Correspondence:** Email: xnlu@hfut.edu.cn.

Abstract: Cloud manufacturing (CM) establishes a collaborative manufacturing services chain among dispersed producers, which enables the efficient satisfaction of personalized manufacturing requirements. To further strengthen this effect, the manufacturing service composition and optimal selection (SCOS) in CM, as a NP-hard combinatorial problem, is a crucial issue. Quality of service (QoS) attributes of manufacturing services, as the basic criterion of functions and capabilities, are decisive criterions of SCOS. However, most traditional QoS attributes of CM ignore the dynamic equilibrium of manufacturing services and only rely on initial static characterizations such as reliability and availability. In a high uncertainty and dynamicity environment, a major concern is the equilibrium of manufacturing services for recovering their functions after dysfunctional damage. Therefore, this paper proposes a hybrid resilience-aware global optimization (HRGO) approach to address the SCOS problem in CM. This approach helps manufacturing demanders to acquire efficient, resilient, and satisfying manufacturing services. First, the problem description and resilience measurement method on resilience-aware SCOS is modeled. Then, a services filter strategy, based on the fuzzy similarity degree, is introduced to filter redundant and unqualified candidate services. Finally, a modified non-dominated sorting genetic algorithm (MNSGA-III) is proposed, based on diversity judgment and dual-track parallelism, to address combination optimization processing in SCOS. A series of experiments were conducted, the results show the proposed method is more preferable in optimal services searching and more efficient in scalability.

Keywords: cloud manufacturing; service composition and optimal selection; HRGO; quality of service; fuzzy similarity degree

1. Introduction

The current manufacturing global operations asks for more stringent requirements than ever before, such as customized demand, personalized products, strict deadlines, standardization of manufacturing processes, quick response and continuous service, as well as disruptive innovations [1]. To address these challenges, Cloud manufacturing (CM) leverages distributed and heterogeneous manufacturing services to meet the customized manufacturing service requirements from manufacturing enterprises and customers via management and integration. In other words, CM utilizes existing information technology to improve control and collaboration of the manufacturing process, which benefits both the enterprises and customers. Specifically, advances in sensors and communications technology can provide the necessary technology to upload operation data and resources information from the physical facility (i.e., the physical world) to the cyber world of internet applications and software. Then, Cloud manufacturing system (CMS) uses advanced data processing and intelligent management to monitor them in the cyber world. At the same time, the actual operations in the physical world are handled at both the manufacturing process level and system operational level [2]. Manufacturers registered on the CMS collaborate to improve their design, manufacturing, and marketing capabilities by sharing their material resources and technical capabilities, which resulting in high customer satisfaction.

Because of the ubiquitous diversity, heterogeneity, and complexity of manufacturing environment, if a single manufacturer cannot meet the requirements of the manufacturing service demanders (MSD), service composition and optimal selection (SCOS) is considered as one of the pivotal technologies that need to implemented in CM. Moreover, SCOS helps MSD to sort and select each candidate service according to specified criteria, such as business goals, strategic value, and resource constraints [3]. The manufacturing resources and capabilities are virtualized by CMS and encapsulated as candidate services, which are then stored in a resource pool. When the manufacturing task submitted by MSD necessitates the cooperation of multiple manufacturers, CMS divides the manufacturing task into multiple sub-tasks and matches relevant candidate services for each sub-task from the resource pool and optimally combine these to achieve the highest overall quality of service (QoS).

The concept of resilience has been widely studied in manufacturing supply chain, service-oriented computing (SOC), and cloud computing (CC), but not in SCOS. For instance, Tapolcai et al. [4] introduced quality of resilience (QoR) as a new class of QoS attribute to describe both service availability and recovery performance. In addition to the expected events that drive manufacturing operations, at unknown points during manufacturing life cycles, unexpected detrimental simple events, or aggregation of events which detrimental effect QoR of manufacturing, may also occur. The uncertainty and dynamicity can be classified as one of the most important causes of operational disruption [5]. These uncertainties, which are related to services, tasks and service correlations, may disrupt the execution of the composite services and may consequently harm the dynamic equilibrium of the submitted manufacturing task. Therefore, a new and measurable QoS attribute needs to be defined to describe the dynamic equilibrium of manufacturing services.

Although previous research identified preliminary motivations for resilience-aware manufacturing, a systematical SCOS approach that considers resilience awareness in CM still needs to be established. The definition of resilience in SCOS should contain a statement that defines the concept of resilience of a service and a dynamic set. The dynamic set describes the equilibriums of a service throughout its execution. This set includes the abilities of a service to resist various detrimental factors that unfavorably interfere with the regular workflow. The resilience measurement of service should, by definition, consider various equilibriums. The resilience of a service, calculated according to this definition, should be identical to other attributes of QoS, and reflect the QoS. Thereafter, the resilience-aware SCOS process should select appropriate services to compile a list of candidates that are satisfied with the QoS expectation of the-MSD. Then, the selected candidates need to be composited to obtain a service solution that realizes the power of resilience in manufacturing.

In this paper, to cope with the resilience-ware service composition in CM, we focus on developing the resilience-enhanced SCOS framework, and a novel hybrid resilience-aware global optimization (HRGO) approach consists of service filter strategy and modified non-dominated sorting genetic algorithm (MNSGA-III) algorithm is proposed to address the combination optimization problem in SCOS. The experiments of optimizing the cooperation partners selection strategy for automobile manufacturers is illustrated to demonstrate the efficiency of the proposed method.

There are three major contributions in this work. First, by evaluating the equilibrium values of manufacturing services, a resilience measurement method is established which can convert the service resilience to QoS indicator. Second, a service filter strategy based on fuzzy QoS model and fuzzy similarity degree is put forward to acquire the resilience satisfied candidates. Third, the MNSGA-III algorithm which can determine the diversity of population in each generation and choose the corresponding environmental selection mechanism is developed to solve the proposed optimization problem. The numerical results show that our method has better performance and scalability.

The remainder of this paper is organized as follows: The most significant research works relevant to SCOS and resilience-aware CM contexts are reviewed in section 2. The resilience measurement method, fuzzy QoS representation, and the definition of resilience are proposed and then mathematically formulated in section 3. The HRGO approach, including a local filter strategy and a multi-objective optimization algorithm, is introduced in section 4. A case study on the supply chain of automobile company is presented in section 5, the results of which verify the effectiveness of the proposed approach. Finally, section 6 concludes this paper and discusses the future research plans.

2. Related work

With the growing need of manufacturing for customized and personalized designs, the single-function service approach no longer has manufacturing significance because of its associated rigidity and lack of flexibility. In contrast, focusing on providing a more robust combination mechanism can lead to more scalability and flexibility through collaboration across various service providers. Considerable research has addressed the SCOS problem from different methodological perspectives. Huang et al. [6] proposed a genetic algorithm based on uncertainty to optimize the choice of services, and introduced comprehensive performance evaluation indicators to analyze the local and global performance of the algorithm. Zhang et al. [7]summarized the latest research on the new paradigm of manufacturing and service. Yi et al. [8] introduced the basic concepts of harmony search algorithm and presented a survey of the application of harmony search algorithm in the field of intelligent manufacturing. Suresh et al. [9] proposed the BF2VHDR optimization algorithm based on fluid dynamics to solve the optimization selection problem. Rahman et al. [10] proposed a privacy protection service selection framework for cloud-based service system. Cao et al. [11] established a service selection scheduling model that considers time, quality, cost, service and other factors, and combines fuzzy decision theory with ant colony optimization to solve the model. Huang et al. [12] studied the optimization of service combination selection, and proposed an optimization algorithm based on chaos control theory. This algorithm is less time consumption while obtaining the optimal solution. Lin and Chong [13] proposed a genetic algorithm based on resource constraints to complete project scheduling aiming at the problem of resource allocation in CM paradigm. Laili et al. [14] suggested that with the increasing number of alternative services with similar functions, QoS is considered to be the key criterion for selecting the most suitable service. Moreover, a hybrid resource allocation model and an improved niche immune algorithm are proposed. Liu et al. [15] proposed a service combination of 'multi-composition for each task' based on the global method to solve the incompetent composite services. Alrifai and Risse [16] proposed a solution that satisfies both global optimization and local selection to meet the QoS requirements and preferences of end-to-end users. Table 1 lists the main subjects of the studies published during period of 2015–2018 on SCOS. Most of the studies have focused on mathematical modelling or optimization of QoS-aware service composition, and were mainly related to time, cost, reliability, availability. However, the resilience of composite services has not been investigated to date.

Table 1. The main research subjects in CM SCOS since 2015.

Research subject	Optimization algorithm	Algorithms compared	Reference
Cost, time, availability, reliability	IDBA-Pareto	GODSS,NSGA-II,MOPSO	[17]
Time, cost, availability, and reputation	MPsaDABC	DE, ABC,DABC	[18]
Time, cost, and reliability	GLA-Pareto	PSO, GA, Enumeration	[19]
Time, cost, availability, and reliability	FGA	TGA	[20]
Cost, time, reliability, and availability, carbon emission	Improved FPA	GA, DE, FPA	[21]
Cost, time, reputation, availability	DE-caABC	ABC, DE, PSO, GA, ACO	[22]
Time, cost, reliability, availability, maintainability,	ABC	PSO, GA	[23]
Time, cost, reliability	HO-MCGA	MCGA	[24]
Cost, execution time, latency time	EDMOEA	NSGA-II	[11]
Total cost, total time	WPPBA	GA, PSO	[25]
Cost, time, availability, reliability	HGA	TGA, SFOA, DGABC	[26]
Time, reliability, and availability	FBAT	DE, PSO	[27]
Cost, time, availability, reliability	HABC	NSGA-II, AbYSS, MOsaDE,SMPSO	[28]
Cost, time, and reliability	MCGACEP	MCGA	[29]

A resilient manufacturing system model designed to sustain operations under disturbances was proposed by Gu et al. [30]. Furthermore, Zhang and Luttervelt [31] proposed the concept of resilient manufacturing systems as well as the guidelines for the design and management of such systems. Zhang et al. [32] proposed a grid manufacturing service scheduling process and proposed an architectural view of failure management. Tao et al. [33]confirm the identified research gaps and presented a comprehensive literature review on manufacturing service management. Shah and Babiceanu [34] proposed the concept of resilience of interdependent infrastructure systems, and compared the final system-level resilience with component infrastructure system resilience. Bouzary [35] promotes the development of methods for the study of deep uncertainties for resilient assessment, while retaining the same probability to express uncertainties about highly uncertain, unforeseen or unpredictable dangers in design and management activities. All these studies emphasized the lack of research in composite service resilience and fault tolerance. A formal definition of resilience, which addresses the ability to prepare for, absorb, and recover from actually or potentially adverse events, is shown in Table 2. Although several studied in the CC and SOC literature addressed the architectures, frameworks and adaptive approaches for improving fault tolerance of web service, these approaches cannot be directly implemented in manufacturing system because of the special characteristics of the composition.

Table 2. Manufacturing-based definitions of resilience.

Definition of resilience by the reference	Reference
Ability to prepare for actual or potential adverse events, Ability to absorb and recover from actual or potential adverse events	[34]
Robustness, Flexibility and Ability to recover	[35]
Ability to anticipate and Ability to adapt	[31]
Ability to anticipate, Recover rapidly and Ability to efficiently adjust	[30]
Identification of frequent and significant failure patterns, Quantification of interdependency related indicators and Empirically risk analyses	[36]
Ability to return to its equilibrium state and Ability to reach its original state	[37]

As mentioned above, SCOS is one of the core technologies for implementing CM. For complex mechanical product such as automobile, unlike the resilience problem in SOC, CC, and supply chain, resilience mainly considers the reasonable and robust scheduling between local manufacturing, interactive logistics and cloud trading. Specifically, despite the significance of resilience composite services, few studies have been based on conceptual or mathematical definition of resilience. The candidate services and research subjects that needed to be considered are massive, and thus, an efficient selection approach considering that simultaneously considers system resilience and QoS is still worthy of further research.

Based on the above analysis, this paper proposes a HRGO approach to assist CM to obtain excellent manufacturing services. Firstly, the problem description on resilience-aware SCOS is summarized. Second, based on the fuzzy similarity degree, a services pool filter strategy is presented in detail to identify appropriate candidate services. Third, a global Optimization Algorithms, modified by NSGA-III, which performs in high dimension objects, is used to conduct the combinatorial optimization process.

3. Problem description and mathematical modeling

3.1. System overview

As a service-oriented manufacturing model, CM originated from pre-existing current manufacturing models and corporation information technologies, supported by manufacturing ability virtualization, CC, Internet of Things (IoT), service-oriented technologies, and advanced computing technologies. To accomplish a complex manufacturing task, the requirements by MSD are decomposed into several sub-tasks, each of which can be completed by a candidate set. More specifically, this technology needs to sequentially solve the three following sub-problems, as show in Figure 1. Table 3 summarizes the notations used in problem representation.

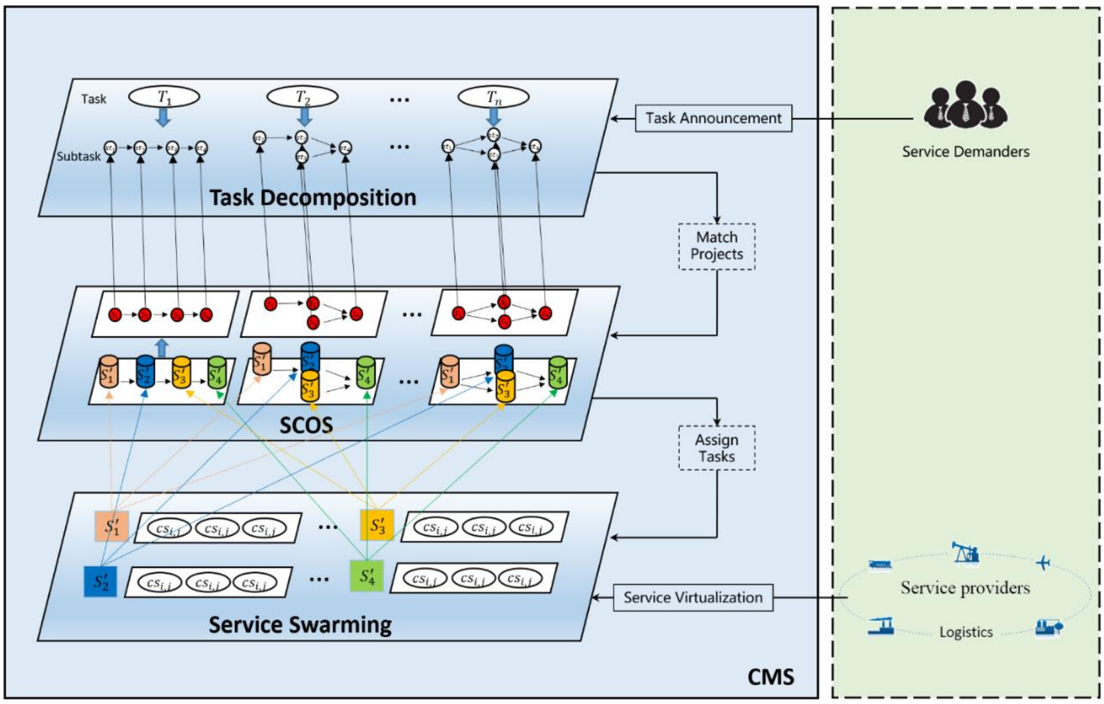

Figure 1. System architecture of CM.

(1) Task Decomposition: breaking down each tasks published by MSD into sub-tasks, equivalent to $T=\{st_1,st_2,\ldots st_j,\ldots,st_J\}$ in which T is the task and st_j is the jth sub-task of T, and J represents the total number of sub-tasks. Each sub-task has certain QoS requirement determined by the MSD, and only those manufacturing services that meet the QoS requirements are eligible to be candidate services for the final optimization step.

(2) Service Swarming: swarming of alternative manufacturing candidate services in services

pools based on functional requirements. Here, $S=\{cs_1,cs_2,\ldots,cs_i,\ldots,cs_I\}$, in which *S* represents the services pool, cs_i represents the *i*th candidate services of *S*, and *I* is the total number of candidate services.

(3) SCOS: Finally, orchestrating the selected manufacturing candidate service from services pools to form a SCOS plan [38]. Here, a candidate service is selected from each corresponding candidate set to composite services while ensuring that the overall service QoS is optimal.

Table 3. Summary of notations.

Symbol	Description
SCOS	Service composition and optimal selection
MS	Manufacturing service
NS	Abstract network structure in SCOS
MSD	Manufacturing service demanders
T	Task published by MSD
S	Services pool
st_j	The *j*th candidate service in NS, $j=1,2,\ldots,J$
cs_i	The *i*th sub-task in NS, $i=1,2,\ldots,I$
$Q_1(\psi_{i,j})$	Cost factor in QoS attribute of manufacturing services
$Q_2(\psi_{i,j})$	Time factor in QoS attribute of manufacturing services
$Q_3(\psi_{i,j})$	Reliability factor in QoS attribute of manufacturing services
$Q_4(\psi_{i,j})$	Resilience factor in QoS attribute of manufacturing services
$S(a,b)$	Similarity degree between two fuzzy number a and b
in	Interval number, $in=[in^-,in^+]=[in \mid in^- \le in \le in^+]$
fn	Triangular fuzzy number, $fn=[fn^L,fn^M,fn^U]$
$rc_{i,j}$	Resource cost of cs_i, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$sc_{i,j}$	Setup cost of cs_i, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$ed_{i,j}$	Service execution duration, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$sd_{i,j}$	Setup duration, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$

Continue next page

Symbol	Description
$qs_{i,j}$	Average quality of service, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$otd_{i,j}$	On-time delivery rate, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$dsr_{i,j}$	Demand satisfied rate, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
λ	The number of relevant variables to study reliability
$en_{i,j,m}$	Endogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$, $m=1,2,\ldots,M$
$ex_{i,j,l}$	Exogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$, $l=1,2,\ldots,L$
m	The mth endogenous attribute
l	The lth exogenous attribute
$\sigma_{i,j,m}$	The weight of each endogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$, $m=1,2,\ldots,M$
$\rho_{i,j,l}$	The weight of each exogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$, $l=1,2,\ldots,L$
$\mu_{i.j}$	Weight of total endogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
$\nu_{i,j}$	Weight of total exogenous attributes, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
M	Number of endogenous attributes
N	Number of exogenous attributes
$\psi_{i,j}$	Decision variable, if service is chosen, then $\psi_{i,j}=1$, else $\psi_{i,j}=0$, $i=1,2,\ldots,I$, $j=1,2,\ldots,J$
C_{exp}	The expected cost from service demander
T_{exp}	The deadline from service demander
RL_{exp}	The expected reliability from service demander
RS_{exp}	The expected resilience from service demander

3.2. The definition of resilience in CM and the measurement of equilibriums

Resilience has often been defined as the ability of a partially disrupted manufacturing services to recover its key functions and components after, and even in the event of, a destructive event [5]. Industry experts evaluate the resilience of a service by measuring their equilibriums, which reflect their abilities to resist different, unexpected, and detrimental factors. When the manufacturing service is in operation, unexpected and detrimental factors, that can cause breakdown of manufacturing services in the manufacturing life cycle, include task failure rate, workflow failure, resource failure rate, cybersecurity penetration, partner credit-default, and logistics delay. Intrinsically, resilience and two types of resilience attributes in SCOS were identified as follows:

Definition: In CM, resilience-aware manufacturing service refers to a service process in which the service leverages the exogenous and/or endogenous equilibriums to return from the disruptive perturbation deviating state to the state of equilibrium.

Exogenous attributes:

- The ability to withstand communication blocking and cybersecurity penetration;
- The ability to withstand behaviors of breaching reputation and trust;
- The ability of composite services to anticipate irresistible factors such as logistics delays

Endogenous attributes:

- The ability to successfully predict work flow level failure;
- The ability of the composite services to operate within pre-defined parameters and condition;
- The ability of composite services to reconfigure and adapt to changes at runtime

According to the above, the resilience measurement of a service is composed of its equilibriums to resist various detrimental factors (manner of composition in section 3.4). The equilibrium of manufacturing services to resist certain detrimental factors is measured by the fluctuation of the corresponding measured value during the observation period. For example, calculating the equilibrium to withstand communication blocking and cybersecurity penetration requires the data transmission speed as input. Calculating the equilibrium to withstand breaching reputation is to measure the delay in delivery time when partners break faith. The equilibrium to resist a certain detrimental factors is equal to the integral of the absolute value of the disturbance value, divided by the maximum amount of change of measured value (that is, the maximum performance degradation due to the detrimental factors).

$$E_A = \frac{\int_{t_a}^{t_b} \left| V_A - V_A^E \right| dt}{\max\left(V_A\right) - \min\left(V_A\right)} \tag{1}$$

Where R_A refers to the measured value corresponding to equilibrium A, V_A represents real-time measured value of equilibrium A, V_A^E represents measured value of equilibrium A in consistent operation, t_a and t_b represent the establishment time and final time of the inspection. The denominator indicates the value of the maximum change of measured value during the inspection period. The larger the change of this value, the worse the performance of the service when it is attacked by detrimental factors and the worse the resilience.

3.3. Mathematical definition: fuzzy QoS representation

QoS attributes provide a standard for measuring the QoS stem from CC and was later applied to CM. QoS is usually used to represent the utility value of web services based on customers' QoS demands. However, QoS is not suitable for estimating manufacturing services because it is not suitable for indicating manufacturing capabilities such as precision, matching relationship, and working life. These receive more attention in the manufacturing industry. Moreover, because of uncertain manufacturing environment, manufacturing industries tend to use fuzzy number to describe the capabilities of manufacturing companies. As a result, this paper uses fuzzy theory to extend the QoS model of CC into a fuzzy QoS model to describe the QoS.

The following are a number of the representations of manufacturing QoS attributes.

$$MS = \{q_{cost}, q_{time}, q_{availbility}, q_{reliability}, q_{reputation}, \ldots\} \tag{2}$$

Where *MS* refers to manufacturing service, q_{cost} is used to describe the service cost such as processing cost and logistics cost, q_{time} represents the total service time from demand response to the final end, $q_{availbility}$ represents available time ratio of composite service, $q_{reliability}$ represents the satisfaction of the manufacturing service, $q_{reputation}$ is applied to describe the authenticity of the service.

To represent the fuzzy attributes of QoS, triangular fuzzy numbers and interval numbers are used to express the linguistic variables and value constraints respectively. The interval number can be defined as $in = [in^-, in^+] = [in \mid in^- \le in \le in^+]$. The triangular fuzzy number can be expressed as $fn = [fn^L, fn^M, fn^U]$. Furthermore, an example of the mapping relation between linguistic variables and triangular fuzzy number is shown in Table 4.

Table 4. Linguistic variables and fuzzy evaluation.

Linguistic variable	Triangular fuzzy number
Very bad	(0, 0.1, 0.2)
Bad	(0.2, 0.3, 0.4)
Medium	(0.4, 0.5, 0.6)
Good	(0.6, 0.7, 0.8)
Very Good	(0.8, 0.9, 1.0)

3.4. Mathematical definition: representation of optimization objectives

The optimization objectives of SCOS are always high-dimensional and correlated. Depending on the manufacturing background, several of the multiple-objective optimization can be converted to single-objective optimization using simple additive weighting (SAW). This paper adopts multi-objective optimization algorithms and mainly takes cost, time, reliability and resilience into account for the following optimization process. Since the QoS representations of different composition models are different (e.g., sequence model, parallel model, selective model, and circular model), and considering the case application of this paper, the sequence composition model was used. Considering the dynamic operating range of manufacturing decentralized network architecture leads to considering the QoS as a dynamic function, depending on node-in and node-out, respectively. The requested sets of inputs and outputs from a given service such (i.e., $QoS^{Nin,Nout}$), *Nin* and *Nout* represent node-in and node-out, respectively. A resilience-aware basis for the SCOS scenario including the following QoS attributes.

Cost: The cost $Q_1(\psi_{i,j})$ attribute includes all the inherent costs for the service execution as well as the associated preparations and adjustments. The price is function of the service implied and the manufacturing demand (i.e., inputs and outputs) and defined beforehand by its related service provider. For the reach of the optimal solution, cost must be minimized and can be calculated from the maximum price that the service demander is willing to invest independently of the currency.

$$\begin{cases} \min Q_1(\psi_{i,j}) = \sum_{j=1}^{N} \psi_{i,j} \cdot (rc_{i,j}^{Nin,Nout} + sc_{i,j}^{Nin,Nout} + p_{i,j} \cdot tr_{i,j}^{Nin,Nout}) \\ s.t.\ \ Q_1(\psi_{i,j}) \le C_{exp} \end{cases} \tag{3}$$

Where $Q_1(\psi_{i,j})$ represents the cost of the ith candidate of the jth sub-task, $rc_{i,j}(Nin, Nout)$ represents manufacturing resource cost , $sc_{i,j}(Nin, Nout)$ represents manufacturing setup cost; $tr_{i,j}^{Nin,Nout}$ represents logistics transport distance incurred by the ith candidate service; $p_{i,j}$ represents the unit transport cost incurred by the ith candidate service, which is related to the mode of transport ;$\psi_{i,j}$ is a decision variable, where, if service is chosen, then $\psi_{i,j} = 1$, else $\psi_{i,j} = 0$; C_{exp} represents the expected cost from service demander.

Time: The time $Q_2(\psi_{i,j})$ attribute refers to service execution duration and the preparation time. The duration is a function of the precision and quality of the equipment and a feature of the request. Identical to the price, duration is a parameter that is minimized and evaluated from the requested deadline.

$$\begin{cases} \min Q_2(\psi_{i,j}) = \sum_{j=1}^{N} \psi_{i,j} \cdot (ed_{i,j}^{Nin,Nout} + sd_{i,j}^{Nin,Nout} + t_{i,j} \cdot tr_{i,j}^{Nin,Nout}) \\ s.t.\ \ Q_2(\psi_{i,j}) \le T_{exp} \end{cases} \tag{4}$$

where $Q_2(\psi_{i,j})$ represents the time of the ith candidate of the jth sub-task, $ed_{i,j}(Nin, Nout)$ represents the duration of manufacturing service execution, $sd_{i,j}(Nin, Nout)$ represents the setup duration, $tr_{i,j}^{Nin,Nout}$ represents the unit transport duration incurred by the ith candidate service, which is also related to the mode of transport $t_{i,j}$, and T_{exp} represents the deadline imposed by the service demander.

Reliability: The reliability $Q_3(\psi_{i,j})$ attribute of a service reflects the ability to successfully complete manufacturing tasks at a given time and condition. The reliability is primarily driven by the global reliability of the service provider. However, the need for specifications and the use of a particular resource can also impact the ultimate reliability. The reliability is expressed as a percentage value and is measured against the minimum level of reliability accepted by the service demander.

$$\begin{cases} \max Q_3(\psi_{i,j}) = \prod_{j=1}^{N} \psi_{i,j} \cdot \dfrac{mut_{i,j}^{Nin,Nout}}{mut_{i,j}^{Nin,Nout} + mdt_{i,j}^{Nin,Nout}} \\ s.t.\ \ Q_3(\psi_{i,j}) \ge RL_{exp} \end{cases} \tag{5}$$

where $Q_3(\psi_{i,j})$ represents the manufacturing service reliability of the ith candidate of the jth sub-task, $mut_{i,j}(Nin, Nout)$ represents the mean up time of the service, $mdt_{i,j}(Nin, Nout)$ represents the mean down time of the service, and RL_{exp} represents the expected reliability from MSD.

Resilience: The resilience $Q_4(\psi_{i,j})$ attribute of this paper mainly consider the exogenous equilibriums of withstanding communication blocking, partner credit-default, and logistics delay, and the endogenous equilibriums of withstand facility failure prognostication error, equipment life-span prediction error, and employee work failure. The difference between reliability and resilience is that reliability is the property the service uses to measure the life of the system (i.e., the continued success of the system), while resilience is an attribute of the service, which is used to measure the insensitivity of the system to disturbances [31]. Furthermore, the proposed resilience QoS mathematical model can also be used to solve any other combination problem. Only slight adjustment is needed according to the user's own application environment, and the proposed model is expected to obtain more competitive or superior results than existing QoS structures.

The mathematical morphology of resilience is expressed in percentage as follows:

$$\begin{cases} \max Q_4(\psi_{i,j}) = \prod_{j=1}^{N} \psi_{i,j} \cdot (\frac{\mu_{i.j}}{M} \sum_{o=1}^{M} \sigma_{i,j,m} \cdot en_{i,j,m}^{Nin,Nout} + \frac{\nu_{i,j}}{N} \sum_{p=1}^{L} \rho_{i,j,l} ex_{i,j,l}^{Nin,Nout}) \\ s.t.\ \ Q_4(\psi_{i,j}) \geq RS_{exp} \end{cases} \tag{6}$$

where $Q_4(\psi_{i,j})$ means the resilience of manufacturing service of the ith candidate of the jth sub-task, $en_{i,j,m}(Nin, Nout)$ represents endogenous equilibriums and $ex_{i,j,l}(Nin, Nout)$ represents exogenous equilibriums, $en_{i,j,m}(Nin, Nout)$ and $ex_{i,j,l}(Nin, Nout)$ are calculated by Eq (1). M and L represent the number of endogenous attributes and the number of exogenous attributes. respectively; $\sigma_{i,j,m}$ and $\rho_{i,j,l}$ represent the weight of each endogenous attributes and the weight of each exogenous attributes, respectively; $\mu_{i,j}$ and $\nu_{i,j}$ represent the weight of total endogenous attributes and the weight of total exogenous attributes, respectively; RS_{exp} represents the resilience, as expected from the service demander.

Based on the principles discussed above, the mathematical formulations of the resilience-aware optimization problem in CM can be represented as follows.

$$\begin{cases} \min Q_1(\psi_{i,j}) = \sum_{j=1}^{N} \psi_{i,j} \cdot (rc_{i,j}^{Nin,Nout} + sc_{i,j}^{Nin,Nout} + p_{i,j} \cdot tr_{i,j}^{Nin,Nout}) \\ \min Q_2(\psi_{i,j}) = \sum_{j=1}^{N} \psi_{i,j} \cdot (ed_{i,j}^{Nin,Nout} + sd_{i,j}^{Nin,Nout} + t_{i,j} \cdot tr_{i,j}^{Nin,Nout}) \\ \max Q_3(\psi_{i,j}) = \prod_{j=1}^{N} \psi_{i,j} \cdot \frac{mut_{i,j}^{Nin,Nout}}{mut_{i,j}^{Nin,Nout} + mdt_{i,j}^{Nin,Nout}} \\ \max Q_4(\psi_{i,j}) = \prod_{j=1}^{N} \psi_{i,j} \cdot (\frac{\mu_{i.j}}{M} \sum_{o=1}^{M} \sigma_{i,j,m} \cdot en_{i,j,m}^{Nin,Nout} + \frac{\nu_{i,j}}{N} \sum_{p=1}^{L} \rho_{i,j,l} ex_{i,j,l}^{Nin,Nout}) \\ s.t.\ \ Q_1(\psi_{i,j}) \leq C_{exp}, Q_2(\psi_{i,j}) \leq T_{exp}, Q_3(\psi_{i,j}) \geq RL_{exp}, Q_4(\psi_{i,j}) \geq RS_{exp} \end{cases} \tag{7}$$

4. Hybrid resilience-aware global optimization approach

Virtual service pools represent the primary causal and most correlated attribute of resilience-aware SCOS [39]. To increase the resilience of a manufacturing system and enhance the operating efficiency of manufacturing SCOS, the service filter strategy is proposed prior to the composition process to filter the redundant candidate services and prevent uncertain and dynamic hazards. This step of filter can also improve the efficiency of the subsequent optimization process.

The resilience-aware SCOS is a typical multi-objective optimization problem. The focus of this process is to acquire optimally composited services, which provide the resilience-efficient advantage and better satisfy MSD. Therefore, in this section, based on services filter strategy and MNSGA-III, a HRGO algorithm is proposed. The framework of HRGO approach is shown in Figure 2.

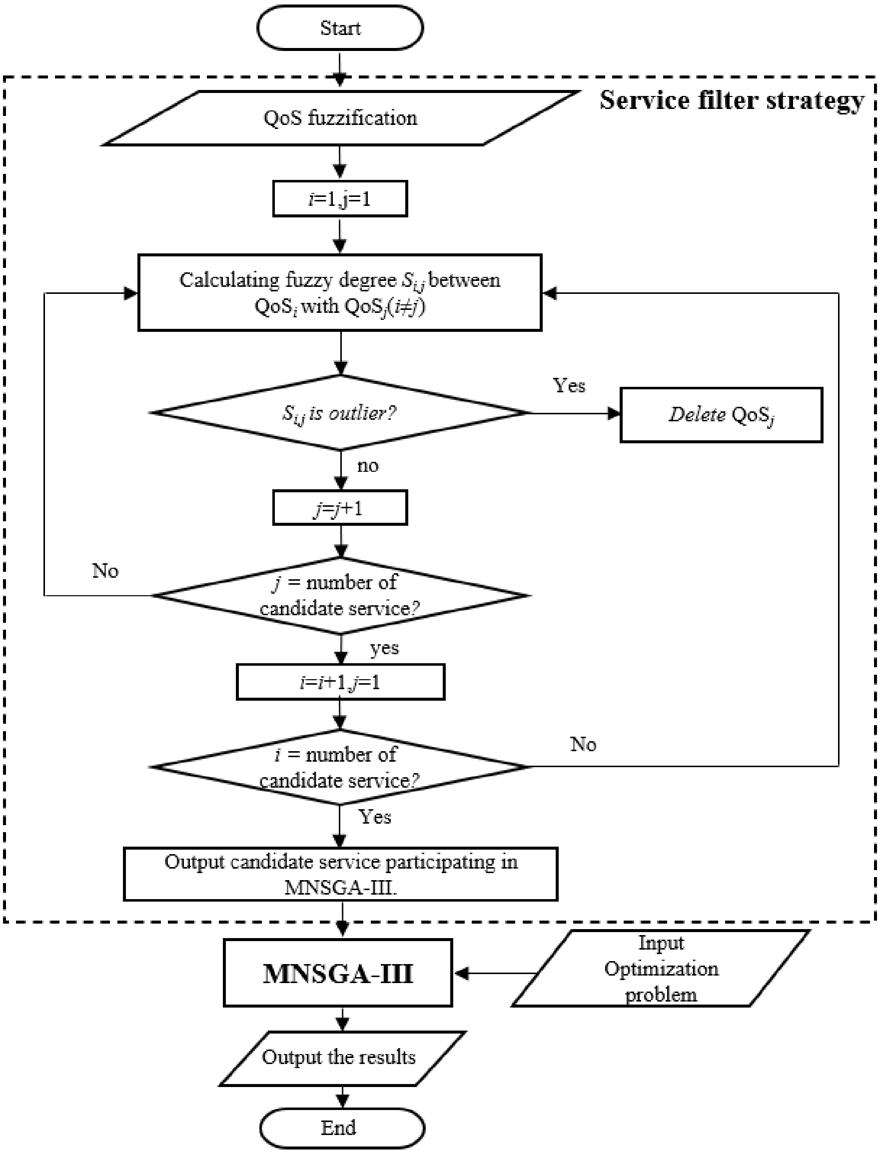

Figure 2. Framework of proposed HRGO.

4.1. Services filter strategy based on fuzzy similarity degree

The CM needs to select the manufacturing services with matching functions and constraints from numerous manufacturing services according to the requirements specified by the service demander. Therefore, a fuzzy similarity degree theory is introduced to filter out redundant and inappropriate service. The literature [40] summarizes important concepts of fuzzy numbers, among which, "dominant" and "no difference" are two important concepts of fuzzy numbers. These concepts constitute a new method which is introduced as follows:

Let *a* and *b* be two fuzzy numbers, originating from different QoS values. If there is overlap between *a* and *b*, these values can be considered as no different in the overlapping part; if there is a non-overlapping part between *a* and *b*, then, for each overlapping part, not *a* control *b* or *b* control *a*. The direction of domination is defined as Table 5.

Table 5. The direction of fuzzy numbers domination.

Case 1 In case of nonoverlap between fuzzy number *a* and *b* are the following
1. The area representing that *a* dominates *b* is defined by ➢ Fuzzy number *a* is in the right-hand side of the fuzzy number *b*, ➢ Fuzzy number *b* is in the left-hand side of the fuzzy number *a*.
2. The area representing that *b* dominates *a* is defined by ➢ Fuzzy number *b* is in the right-hand side of the fuzzy number *a*, ➢ Fuzzy number *a* is in the left-hand side of the fuzzy number *b*
Case 2 In case of overlap between fuzzy number *a* and *b* are the following
1. The area representing that *a* dominates *b* is defined by ➢ The area belongs to *a* and is in the right-hand side of the overlap area, or ➢ The area belongs to *b* and is in the left-hand side of the overlap area
2. The area representing that *b* dominates *a* is defined by ➢ The area belongs to *a* and is in the left-hand side of the overlap area, or ➢ The area belongs to *b* and is in the right-hand side of the overlap area.

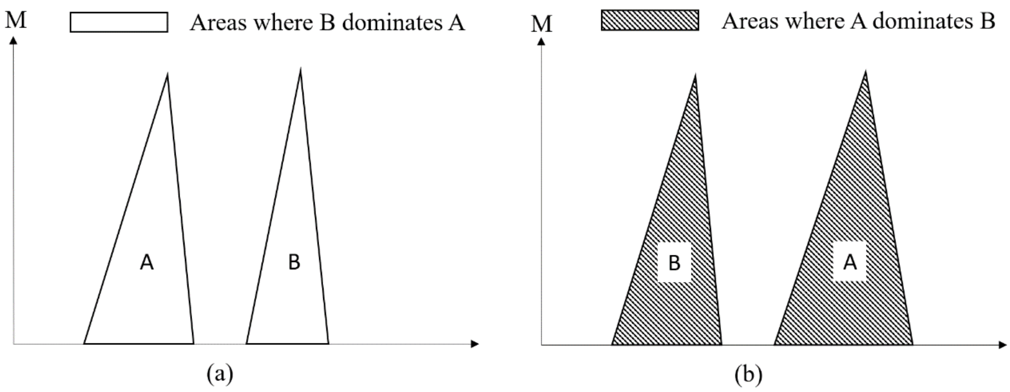

Figure 3. Two dominance situations in the nonoverlap case.

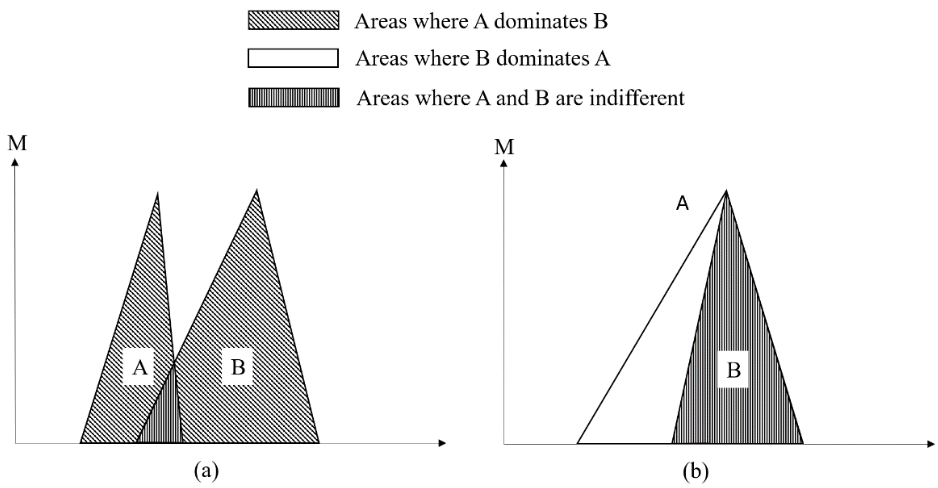

Figure 4. Two possible situations of dominance and no difference in overlap case.

Figures 3 and 4 illustrate the notions of the dominance and the no difference in the nonoverlap and the overlap cases, respectively.

Suppose there are the *k*th QoS parameter of *i*th candidate service of the *j*th sub-task in the form of triangular fuzzy number, $a_{j,k}$ and $b_{j,k}$ in the Eq (8) refer to a same QoS attributes of different candidate services in the same sub-task. The similarity degree between two candidate services *a* and *b* can be calculated by:

$$S(a,b)=\frac{O(a_{j,k}\cap b_{j,k})}{O(a_{j,k}\cap b_{j,k})+\omega_k D(a_{j,k},b_{j,k})+\zeta_k D(b_{j,k},a_{j,k})} \tag{8}$$

where $O(a\cap b)$ represents the area of indifference, $D(a\cap b)$ represents the area where *a* dominates *b*, $D(b\cap a)$ represents the area where *b* dominates *a*, ω_k and ζ_k represent weight parameters that can be determined by decision makers, and the values of ω_k and ζ_k depend on the QoS parameter preference of decision makers and can be computed by analytic hierarchy process (AHP). The fuzzy similarity degree is calculated on the *k*th QoS parameter value of the *i*th candidate service of the *j*th sub-task. The candidate service with outlier is filtered out, and the remaining candidate services participate in global optimization.

4.2. A modified NSGA-III based on diversity judgment and dual-track parallelism

In this section, the details of MNSGA-III are introduced to address the optimization problem as mentioned is section 3.4. The candidate service, filtered by the service filtering strategy, is used as input of MNSGA-III. Then, the optimal service composition is obtained. NSGA-III substitutes the reference-point-based environmental selection mechanism for the crowding distance operator in NSGA-II. This paper adds an environment selection mechanism based on clustering, while modifying the environmental selection mechanism based on reference points. By assessing the diversity of the offspring, one of the above two environmental selection mechanisms are applied for the selection of individuals from the offspring. The above steps can better retain the distribution characteristics of the population, and are more suitable to solve the problem of how QoS attributes interactively affect the final composition in SCOS. The following briefly presents the framework of MNSGA-III, which is followed by a description of the main components of the proposed algorithm. The framework of the existing NSGA-III and proposed MNSGA-III is shown in Figure 5.

4.2.1. Existing Algorithms

Each generation *t* of the NSGA-III procedure is briefly described as follows: first, the offspring population Q_t is generated by crossover and mutation of the parent population P_t; then, the parent population and offspring population are merged into S_t. The population S_t is classified into non-dominated levels F_i and the next generation is formed as $P_{i+1}=\bigcup_{i=1}^{l-1}F_i$. NSGA-III guarantees the diversity of optimal solutions through predefined reference points $H=\binom{M+P-1}{P}$ on a hyper-plane as shown in Figure 6. To achieve this, population members S_t is normalized by identifying the minimum value so that the translated ideal point becomes a zero vector. The definition of reference lines $\boldsymbol{w}$ corresponds to each reference point on the hyper-plane; then, each population member is associated with a reference point by calculating the minimum perpendicular distance $d^{\perp}(\mathbf{s},\mathbf{w}) = \left\|\left(\mathbf{s} - \mathbf{w}^{T}\mathbf{s}\mathbf{w}/\|\mathbf{w}\|^{2}\right)\right\|$ from each population member to each reference line. By applying a niche-preservation operation, an appropriate number of members of the population is connected to a reference point. At last, K members are chosen, one at a time, from F_l to construct P_{t+1}.

Figure 5. Framework of the existing NSGA-III and proposed MNSGA-III.

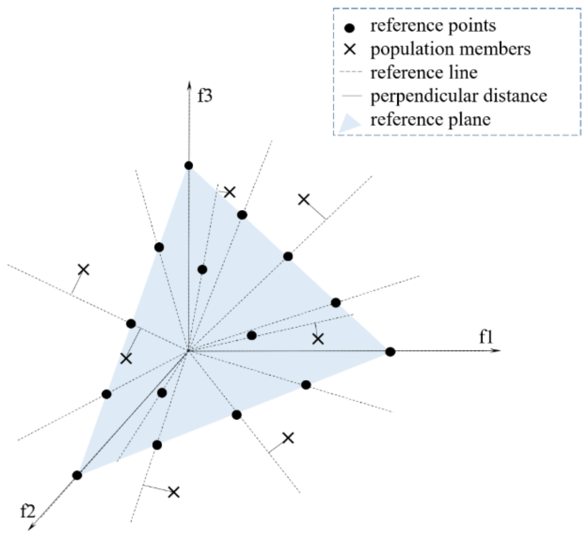

Figure 6. Environment selection mechanism of existing algorithms (three-objective problem with $p = 4$).

4.2.2. Framework of the proposed algorithm

For completeness, a brief description of the MNSGA-III is first presented in algorithm 1. The initial population P_0 with N members is coded by filtered candidate services. The offspring population Q_t is obtained by using a recombination and mutation operator. Then, P_t and Q_t are combined to form S_t with a size of $2N$. Non-dominant sorting is used to sort S_t into different non-domination levels, denoted by (F_1, F_2, …), thus forming S_t in the order of non-dominant levels, starting from F_1 and ending with $F_{l\text{-}1}$. The following steps present the implemented improvements to NSGA-III. If $|S_t| = N$, S_t is directly used as P_{t+1}. If not, whether F_l is diverse needs to be calculated, and the corresponding environmental selection mechanism to select the next generation is selected. To achieve this, fuzzy c-means (FCM) was used to divide F_l into K clusters. Then, the diversity of F_l was evaluated using the cluster quality index [41]. If F_l has a poor diversity (i.e., $R < R_{flr}$), by calculating the perpendicular distance from a reference point to cluster center, a stray reference point can be found. Then, based on stray reference points, an environmental selection mechanism was applied to construct P_{t+1}. If F_l already has good diversity (i.e., $R \geq R_{flr}$), an environmental selection mechanism [42] based on cluster centers was applied to construct P_{t+1}.

4.2.3. Two-layered reference points structure

To increase the diversity of the obtained solutions, NSGA-III generates a set of N reference points to help create offspring populations. Here, a two-layered reference points structure is adopted as previously suggested [43]. The two-layers refer to boundary layer H_1 and internal layer H_2, respectively. The population size can be calculated as:

$$N = \binom{H_1 + m - 1}{m - 1} + \binom{H_2 + m - 1}{m - 1} \tag{9}$$

For instance, if in a three-objective problem (m = 3), two divisions ($H_1 = 2$) are chosen for each objective axis in the boundary layers and one division ($H_2 = 1$) are chosen for each objective axis in the inside layers, then, a total of $\binom{2+3-1}{3-1} + \binom{1+3-1}{3-1} = 6 + 3 = 9$ reference points are created. For clarity, these reference points are shown in Figure 7. Since these reference points are widely distributed over the whole normalized hyperplane, the obtained solutions may also be widely distributed in or near the Pareto-optimal frontier (POF), i.e., the corresponding approximate Pareto-optimal solution, thus, the method avoids falling into the local optimal solution.

4.2.4. Crossover, mutation, and elite-preserving mechanism

Crossover and mutation are conducted with two selected paternal individuals P_t to obtain offspring individual Q_t. Crossover implies that two individuals each contribute a portion of their chromosomes to generate two new individuals. Mutation refers to a random change in a number of genes in a new individual. Specifically, the simulated binary crossover (SBX) and polynomial mutation are used for inheritance in this paper. Note that the use of differential evolution (DE) instead of SBX and polynomial mutation operators, as used in a previous study [44], has also achieved good performance. However, this study follows the advice presented with the original algorithm and uses larger values of distribution index for the SBX operator.

To avoid the loss of well-behaved chromosomes within the population, the elite-preserving mechanism merges the parent population P_t and the offspring population Q_t into a mixed population ($R_t \leftarrow P_t \cup Q_t$). With this mechanism, all paternal individuals compete with offspring individuals, and elite chromosomes will be preserved. Specifically, the dominant relationship between two individuals in the population is based on their performance with regard to specified objectives. The dominating principles are defined as follows: if $\forall i,\ f_i(x_1) \geq f_i(x_2)$ and $\exists i,\ f_i(x_1) > f_i(x_2)$, then, individual x_1 dominates x_2, of which, $i = 1, 2, \ldots, M$, where M represents the object dimensions. Non-dominant sorting was performed for individuals in the mixed population R_t. Individuals within the population who are not dominated by other individuals are classified as non-dominant F_1. The level F_1 is then removed to identify new non-dominant individuals in level F_2. This process is repeated until all individuals in R_t are divided into several non-dominant levels F_1, F_2, …, F_n. Individuals at the former level performed better than individuals at the latter level with regard to each objective. Therefore, individuals at the former level are preferentially preserved in the next generation.

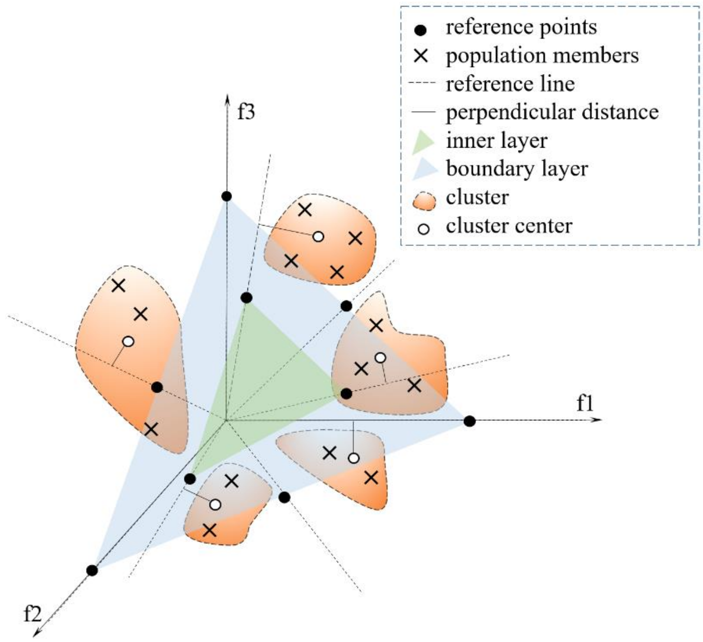

Figure 7. Environment selection mechanism of the proposed algorithm (three-objective problem with $H1 = 2$ and $H2 = 1$).

4.2.5. Diversity discrimination based on clustering

After normalization, at the last non-dominated level F_l, NSGA-III uses reference points based on environmental selection mechanism to enhance offspring diversity. This paper considers the distribution characteristics of the population in each generation, and then applies the corresponding environmental selection mechanism to select the offspring. Therefore, populations of the non-dominant level F_l are clustered. It is worth noting that the number of clusters after clustering must neither be too large nor too small. This is because the environmental selection mechanism is affected by the number of clusters and will ultimately affect the distribution of Pareto-optimal solutions. When the number of clusters is too small, the individuals in each cluster occupy a wide solution space; thus, Pareto-optimal solutions cannot be subdivided or traversed along the entire POF. When the number of clusters is too large, although the solution can be subdivided and traverse the entire POF, it will cause high computational complexity.

Therefore, this study adopts the FCM algorithm, which continuously updates the membership matrix U and the clustering center V through an iterative scheme to achieve convergence of the objective function $J_m(U,V)$:

$$J_m(U,V) = \sum_{k=1}^{K}\sum_{i=1}^{I}(u_{ik})^m d_{ik}^2(x_i, v_k) \tag{10}$$

where, $U = \{u_{ik}\}$ represents the membership matrix, which satisfies $u_{ik} \in [0,1], \forall i,k; 0 < \sum_k u_{ik} < n, \forall i; \sum_i u_{ik} = 1, \forall k$, M represents the fuzzification parameter, $V = \{v_1, v_2, \ldots, v_k\}$ represents the cluster centers set, $X = \{x_1, x_2, \ldots, x_I\}$ represents the exemplars set, and $d_{ik}^2(x_i, v_k)$ represents the distance from the ith individual to the center of the kth cluster:

$$d_{ik}^2(x_i, v_k) = \|x_i - v_k\|_A^2 = (x_i - v_k)^T A (x_i - v_k) \tag{11}$$

For $A = I_{s\times s}$, d_{ik}^2 represents the Euclidean norm.

During iteration, the membership function is updated as follows:

$$u_{ik} = \left[\sum_{k=1}^{K}\left[\frac{d_{ik}(x_i, v_k)}{d_{ik}(xi, v_k)}\right]^{2/(m-1)}\right]^{-1} \tag{12}$$

The cluster center is updated as follows:

$$v_k = \frac{\sum_{i=1}^{I}(u_{ik})^m x_i}{\sum_{i=1}^{I}(u_{ik})^m} \qquad i = 1, 2, \ldots, c \tag{13}$$

Repeat Eqs (11) and (12) until $J_m(U,V)$ converges.

Index Q is introduced to determine the diversity of the population. If the value of the clustering quality index Q is small, the individuals in F_l are concentrated and form a single cluster. When the value of Q is large, this suggests that population F_l has a widespread distribution. Only when the population F_l has a widespread distribution, a widely distributed Pareto-optimal solution can be obtained that represents the POF. The obtained Pareto-optimal solution can represent the POF of different regions.

$$Q = \sum_{i=1}^{K}\frac{1}{|p_i^k|}\sum_{i\in k} d_{ik}(p_i^k, v_k) \tag{14}$$

Here, ξ is defined as the criterion to determine the diversity of the current population:

$$\xi_{t+\Delta} = \theta \times Q_t \tag{15}$$

Where, Q_t represents the Q value of the population in the tth generation, θ is a penalty factor that prevents undue oscillation of Q_t as the population approaches POF, $\xi_{t+\Delta}$ means that ξ is calculated at the tth generation and is used as threshold for the next Δ generations. If Q_t is lower than $\xi_{t+\Delta}$, from $t+1$ to $t+\Delta$, the population is considered to have poor diversity; otherwise, the population is considered to have good diversity.

4.2.6. Environment selection mechanism

In this section, different environmental selection mechanisms are chosen according to whether the population is diverse. The difference is that if the current generation is diverse, the selection method of the offspring will be based on the distribution characteristics (clusters) of the population; if the current generation is not diverse, the selection method of the offspring will likely be based on the reference point.

If $Q_t < \xi_{t+\Delta}$ (i.e., the population in the current generation has poor diversity), first, the distance from the *n*th reference point to the *k*th cluster center is calculated and denoted by matrix Φ_{kn}. Then, the sum of the distances from the *n*th reference point to all center points is calculated and denoted by π_n. Reference points are sorted according to π_n and individuals near stray reference points are selected (i.e., the top β points with the largest π_n). Individuals closest to the stray reference point are placed in $Slot_1$. The cluster whose center is closest to the stray reference point is called the stray cluster, and the individual closest to the center of this stray cluster is placed in $Slot_2$. Finally, K/β points are randomly selected in the stray cluster and placed in $Slot_3$. Finally, K individuals are selected to fill S_t according to their order of priority $Slot_1 > Slot_2 > Slot_3$.

In the event of $Q_t \geq \xi_{t+\Delta}$ (i.e., that the population in the current generation has good diversity), first, all individuals in F_l are assigned to each cluster center based on their Euclidean distance. Then, for all cluster centers, if the number of assigned individuals is greater than or equal to 3, the individual closest to the cluster center is placed in T1, and the other individuals are placed in T2. If the number of assigned individuals is less than 3, the individual closest to the cluster center is placed in T1, and the other individuals are placed in T3. Similarly, K individuals are selected to fill S_t according to their order of priority $Slot_1 > Slot_2 > Slot_3$.

Algorithm 1 Framework of the Proposed Modified NSGA-III
Input: *H_1, H_2, two-layered reference points divisions; P_0, initial parent population with size N;*
m, dimension of objective space
Output: *P_{MaxGen}, population meet the max generations*

Z← Generate Reference Points (H_1, H_2, m)
S_t←Ø
t ← 0 %The generations iteration count
while not met the maximum number of generations(MaxGen) **do**
    Generate offspring: Q_t ←Crossover and Mutation (P_t)
    Combine parent and offspring populations : $R_t \leftarrow Pt \cup Q_t$
    Get Pareto non-domination levels: (F_1, F_2, …) ←Pareto non-dominated sorting (R_t)
    while $|S_t| < N$ **do**
        $S_t = S_t \cup F_i$
        $i \leftarrow i+1$
    end while
    Last front to be included: $F_l = F_i$
    if $|S_t| = N$ **then**
        $P_{t+1} = S_t$, ***break***
    else

Points to be chosen from F_l: $K = N - |P_{t+1}|$
Normalize objectives (S_t, Z)
if $t = 0$ **then**
Initialize cluster center set $V=\{v_1, v_2,\ldots,v_k\} \in S_t$
end if
$r \leftarrow 0$ %The FCM iteration count
repeat
$U^{r+1} \leftarrow$Compute membership matrix (S_t, V^r)
$V^{r+1} \leftarrow$Calculate K cluster centers (S_t, U^{r+1})
$J^{r+1} \leftarrow$Calculate Objective functions (S_t, V^{r+1}, U^{r+1})
until Objective functions J^{r+1} convergence
Last iteration to be included: $p(k)$, V, $d_{ik}(p_i^k, v_k)$
% $p_i^k \leftarrow$all individuals in cluster k
% $V \leftarrow$cluster centers set:$\{ v_k \}$
% $d_{ik}(p_i^k, v_k)$ $\leftarrow$distance from individuals p_i^k to cluster centers v_k
$R \leftarrow$ Evaluate the diversity of current generation [41]($p(v_k)$, V)
if $R < R_{flr}$ **then** % population in current generation have a bad diversity
$\Phi_{kn} \leftarrow$ Calculate distance between reference line and cluster center ($p(v_k)$, λ_n)
Calculate the sum of the distances of each reference line: $\pi_n = \sum_{k=1}^{K} \Phi_{kn}$
$\pi'_\beta\{\lambda_n\}$ $\leftarrow$Sort the reference points in descending order according to π_n
$\pi'_\beta\left\{\lambda_\beta\right\}$ $\leftarrow$Extract the first β reference points: $\pi'_\beta\{\lambda_n\}$
for $j \leftarrow 1$ to β **do**
$Slot_1 \leftarrow Slot_1 \cup \left\{Pop^{f_l}_{\min\lambda} \mid \arg\min(dis(Pop^{f_l}, \lambda_j))\right\}$
$Slot_2 \leftarrow Slot_2 \cup \left\{Pop^{f_l}_{\min v} \mid \arg\min(dis(Pop^{f_l}, v_k \text{ closest to } \lambda_j))\right\}$
$Slot_2 \leftarrow Slot_2 \cup \left\{Pop^{f_l}_{rndv} \mid \text{randomly select } K/\beta \text{ populations from the kth cluster}\right\}$
end for
If $|Slot_1| == K$ **then**
$S_t \leftarrow Slot_1$
else if $|Slot_2| \geq K - |Slot_1|$ **then**
$S_t \leftarrow Slot_1 \cup$ randomly select $(K - |Slot_1|)$ population from $Slot_2$
else
$S_t \leftarrow Slot_1 \cup Slot_2 \cup$ randomly select $(K - |Slot_1| - |Slot_2|)$ population from $Slot_3$
end if
else % population in current generation have a good diversity
$S_t \leftarrow$K individuals selected by cluster centers[42] ($p(v_k)$, $d_{ik}(x_i, v_k)$, V)
end if
$P_{t+1} \leftarrow S_t$
end if
end while

5. Case study

A series of experiments were designed to test the performance of the proposed algorithm in section 5.1, experiment background is briefly described. Then, in section 5.2, the parameter setting of seven algorithms are listed. Finally, in section 5.3, three experimental methods which are metric of reference points, metric of Pareto-optimal points, and impact of number of candidate service were used to state the performance of comparison results. Among them, the aim of section 5.3.1 and section 5.3.2 is to evaluate the effectiveness of improvement strategies of multi-objective optimization algorithm for MNSGA-III. The aim of section 5.3.3 is to evaluate the effectiveness of the performance of SCOS capability for HRGO.

5.1. Order task chain model

To demonstrate the effectiveness of the proposed method, this paper investigates the SCOS in a group manufacturing enterprise (http://www.cheryinternational.com/) named Chery Automobile Co., Ltd. (Chery). The products manufactured by Chery are exported to more than 80 countries and regions across the world. The products of Chery are mainly composed of systems for automotive engine parts, automotive chassis parts, automotive body and accessory parts, and automotive electronic and electrical parts. Each component system consists of many sub-components that need to be selected for the best manufacturing service.

To meet the production requirements, this paper considers SCOS of Chery for the abstract network structure (NS) of automotive electronic and electrical parts systems. These include the manufacturing process of automotive body electronic control components and on-board electronics such as engine storage battery and control system (ECS). The information not only includes QoS attributes, but also processing data and after-sales data. At the initial stage of service optimal selection, information about customer demand should be analyzed to identify QoS constraints. Table 6 shows an example, with the collected QoS constraints of auto parts service of Chery's demand. Applying the services filter strategy as proposed in Section 4, the candidate services can be identified, as shown in Table 7.

Table 6. Example of collected QoS constraints.

Attributes	Cost	Time	Reliability	Resilience	Reputation
Constraint	< 500	< 5	> 65	> 70	good

Table 7. Example of candidate service attributes.

Class	No.	Cost（CNY）	Time(hours）	Reliability（percentage）	Maintainability（percentage）
Storage battery	1	[289,298,299]	[1.6,1.9,2.2]	[60,90,95]	[63,70,79]
	2	[319,328,399]	[2.2,2.8,2.9]	[65,70,79]	[55,60,66]
	3	[299,310,321]	[3.1,3.3,3.6]	[69,70,80]	[58,70,73]
	4	[359,370,381]	[3.5,3.7,3.9]	[55,70,85]	[85,90,93]
	…				
Alternator	1	[109,130,141]	[11.1,12.4,13.9]	[76,80,89]	[66,70,74]
	2	[169,190,199]	[25.4,26.1,27.1]	[45,50,59]	[55,70,78]
	3	[218,239,259]	[13.3,14.7,15.8]	[53,60,67]	[68,80,88]
	4	[259,269,289]	[16.2,17.6,18.9]	[87,90,94]	[67,80,91]
	…				
Starter Motor	1	[259,269,279]	[0.9,1.3,1.7]	[35,50,55]	[57,60,63]
	2	[310,340,360]	[0.4,0.5,0.7]	[67,80,89]	[85,90,92]
	3	[299,310,330]	[0.7,0.9,1.1]	[53,60,71]	[41,50,57]
	4	[285,295,310]	[0.9,1.2,1.6]	[89,90,96]	[76,80,83]
	…				

Note: Manufacturing time is calculated according to the annual yield. For example, in lead-acid battery, the forming step of plate polarization takes 100 hours, which is obviously not suitable as a manufacturing time.

A reasonable composition coding problem is essential for an optimization process based on the HRGO algorithm. This paper applies an integer array to represent a chromosome, as shown in Figure 8. The length of the array represents the number of combined sub-tasks, and the element in the array expresses the index of corresponding candidate services.

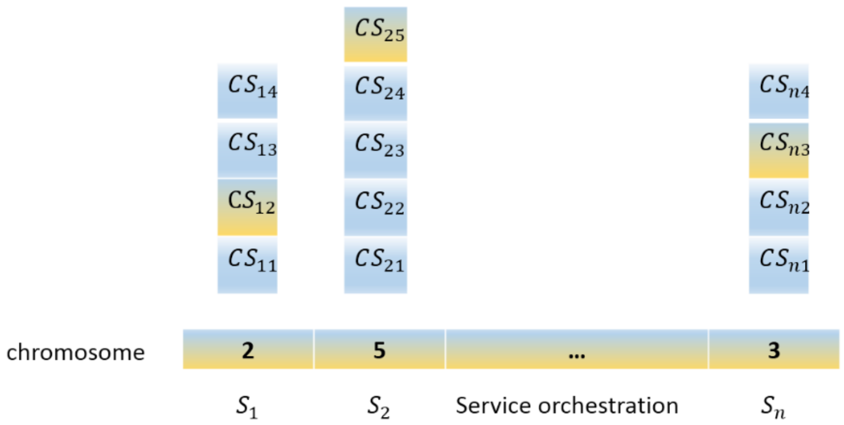

Figure 8. An example of encoding for SCOS problem.

The hardware and software platform parameters are listed in the following: Windows 7 64-bit system, CPU Intel(R) Core (TM) i7-4790K 4.0 GHz frequency, 16G RAM, MATLAB R2014b and Python 3.7.0.

5.2. Experimental setting

To fully evaluate the performance of the proposed algorithms, three classic multi-objective optimization algorithms (MOEA/D, NSGA-III, and NSGA-II) were selected to evaluate the convergence and diversity of MNSGA-III. In addition, three heuristic algorithms (CGA, HGAFOA, and FPSO) were also applied to SCOS are used to evaluate the effectiveness of HRGO.

- MOEA/D [46]:

MOEA/D decomposes a multi-objective optimization problem into several scalar optimization sub-problems, and optimize them simultaneously.

- NSGA-III [43]:

NSGA-III follows the NSGA-II framework and emphasizes multi-objective evolutionary algorithms based on reference points. NSGA-III specialties in handling many-objective optimization problems.

- NSGA-II [47]:

For most problems, NSGA-II can identify the better solution distribution and better convergence algorithms near the true Pareto-optimal frontier.

- CGA [48]:

Cluster-based genetic algorithm (CGA) use FCM algorithm and genetic algorithm to solve the problem of manufacturing resource combination optimization.

- HGAFOA [26]:

Hybrid genetic algorithm and fruit fly optimization (HGAFOA) combines genetic algorithm and fruit fly algorithm for evolutionary search process.

- FPSO [49]:

The fuzzy particle swarm optimization algorithm (FPSO) proposes a QoS calculation model, based on fuzzy theory, and uses particle swarm optimization algorithm to identify the optimal service combination.

The experimental parameter settings are listed in the following:

Number of runs: Each algorithm is independently run 20 times for each test instance.

Termination criterion: The termination criterion of an algorithm for each run is specified in the form of the maximum number of generations (MaxGen). Since the used test problems have varying computational complexity, different MaxGen are used for different problems.

Population size: The population size N for MOEA/D, NSGA-II, and NSGA-III cannot be arbitrarily specified, since N is controlled by parameter H. For other algorithms, the population size can be set to any positive integer; however, to ensure a fair comparison, the same population size is adopted.

Parameters for crossover and mutation: SBX and polynomial mutation [45] are used in all considered algorithms. For MOEA/D, NSGA-II, NSGA-III, and HRGO, the settings are only slightly different, according to [43], where η_c is set to 30.

Table 8 lists the definition of algorithm parameters used in this paper for the problem with different numbers of objectives. Moreover, to verify the proposed algorithm and compare it to other state-of-the-art algorithms, the following four performance indexes were used:

- IGD-metric [46]:

Let A be a set of optimal solutions obtained by the algorithm. Let P^* be the true Pareto front. The distance P^* from to A is defined as following:

$$\mathrm{IGD}(A,P^*)=\frac{\sum_{v\in P^*} d(v,A)}{|P^*|} \tag{16}$$

Where $d(v,A)$ represents the minimum Euclidean distance between v and the points in A.

- Set coverage (C-metric) [50]:

Let A and B be the two optimal solutions obtained by two algorithms. $C(A,B)$ refers to the percentage of optimal solution in B that are dominated by at least one optimal solution in A.

$$C(A,B)=\frac{|\{u\in B \mid \exists v\in A: v\ \textbf{dominates}\ u\}|}{|B|} \tag{17}$$

- Optimality of the composition [51]:

The metric represents the ratio between the value of optimization objective of the algorithm V_{OC} and the value of optimization objective of composite service obtained with an exhaustive search algorithm V_{OC}^{ESA}.

$$\textit{optimality of the composition}=\frac{V_{OC}}{V_{OC}^{ESA}} \tag{18}$$

- Elapsed time:

The metric refers to the convergence time of the algorithm or the time when the number of iterations to reach 400. The unit is seconds.

Table 8. Definitions of algorithm parameters, where n represents the number of variables.

Algorithm	Parameters
MOEA/D	Probability of crossover $P_c=0.75$, Probability of mutation $P_m=1/n$, Distribution index for crossover $\eta_c=30$, Distribution index for Mutation $\eta_m=20$.
NSGA-III	Probability of crossover $P_c=0.75$, Probability of mutation $P_m=1/n$, Distribution index for crossover $\eta_c=30$, Distribution index for Mutation $\eta_m=20$.
NSGA-II	Probability of crossover $P_c=0.9$, Probability of mutation $P_m=1/n$, Distribution index for crossover $\eta_c=30$, Distribution index for Mutation $\eta_m=20$.
CGA	Probability of crossover $P_c=0.75$, Probability of mutation $P_{m1}=0.1$, $P_{m2}=0.04$, population size is 40, maximum iterations is 400, Cluster Number = 3.
HGAFOA	Probability of crossover $P_c=0.9$, Probability of mutation $P_m=0.9$, neighbors size $SN=1$ and the number of the selected positions $L=1$.
FPSO	The inertia weight coefficient $w=0.8$, and the acceleration factor c_1 varies from 1.5 to 2.5 linearly, while c_2 varies from 1.5 to 2.5.
HRGO	Probability of crossover $P_c=0.5$, Probability of mutation $P_m=0.1$, Distribution index for crossover $\eta_c=30$, Distribution index for Mutation $\eta_m=20$.

5.3. Comparisons and results

5.3.1. Performance metric of reference points

Recently, IGD is a way to compute multi-objective evolutionary algorithms (MOEAs), in which the reference points or reference directions are supplied. This paper uses IGD to perform a comparative experiment with algorithms containing reference points (MOEA and NSGA-III). With regard to the four optimization objectives detailed in section 3.4, Q1 and Q2 are the basic subjects of the SCOS, and Q3 and Q4 are advanced subjects, which need to be considered in conjunction with Q1 and Q2. Therefore, Q3 and Q4 are not only studied separately in the experiments.

The experiments are conducted on two-objective optimization problems: Q1&Q2, Q1&Q3, Q1&Q4, Q2&Q3, and Q2&Q4; three-objective optimization problems: Q1&Q2&Q3, Q1&Q2&Q4, Q1&Q3&Q4, and Q2&Q3&Q4; and the four-objective optimization problem: Q1&Q2&Q3&Q4. Twenty independent runs were conducted with each of the compared algorithms. Each run produced a set of non-dominant solutions.

Table 9 gives the comparisons between MOEA/D, NSGA III and MNSGA-III in terms of IGD-metric. The better performance is marked in bold. Table 9 shows that for the NS problem MNSGA-III and NSGA-III are tied in terms of the IGD metric, followed by MOEA/D. For the two-objective problem, MNSGA-III slightly performs better than NSGA-III. However, in three-objective problems, NSGA-III performs better, we suspect that the reason is that MNSGA-III sometimes fails to capture the associate reference point in 3-dimensional objective space. MOEA/D consistently does not perform well in all dimensional versions of the problem. This observation is similar to that concluded in the original MOEA/D study [50] based on two-objective and three-objective problems.

Table 9. Comparison of IGD-metric of MOEA/D, NSGA-II and, MNSGA-III.

IGD-metric	MOEA/D	NSGA-III	MNSGA-III
Q1&Q2	5.617×10^{-3}	$\mathbf{2.876\times10^{-3}}$	5.453×10^{-3}
Q1&Q3	2.115×10^{-2}	6.768×10^{-3}	$\mathbf{6.278\times10^{-3}}$
Q1&Q4	6.966×10^{-2}	4.966×10^{-3}	$\mathbf{3.262\times10^{-3}}$
Q2&Q3	1.563×10^{-2}	6.271×10^{-3}	$\mathbf{2.801\times10^{-3}}$
Q2&Q4	4.274×10^{-2}	$\mathbf{3.019\times10^{-3}}$	7.617×10^{-3}
Q1&Q2&Q3	9.902×10^{-2}	$\mathbf{7.682\times10^{-3}}$	8.124×10^{-2}
Q1&Q2&Q4	8.251×10^{-2}	$\mathbf{5.511\times10^{-2}}$	7.896×10^{-2}
Q1&Q2&Q4	5.083×10^{-2}	4.257×10^{-2}	$\mathbf{4.139\times10^{-2}}$
Q2&Q3&Q4	3.284×10^{-1}	5.041×10^{-2}	$\mathbf{4.652\times10^{-2}}$
Q1&Q2&Q3&Q4	7.351×10^{-1}	$\mathbf{8.041\times10^{-2}}$	5.665×10^{-1}

5.3.2. Performance metric of Pareto-optimal points

However, such an IGD is not applicable to MOEAs without using reference points or directions, e.g., NSGA-II. For these algorithms, the many-objective optimization task is to identify sparsely distributed Pareto-optimal points over the entire POF [52]. In this scenario, a further commonly used indicator, i.e., the C-metric [50], is adopted to evaluate the performance.

Tables 10 shows that the final result, obtained by MNSGA-III, is better than that obtained by MOEA/D in terms of the C-metric. This is true for all test instances except for Q2&Q4 in which MNSGA-III has a slightly worse C-metric than MOEA/D. Taking instance Q1&Q2&Q4 as example, on average of 61.5% of the final solutions generated by MOEA/D are dominated by those generated by MNSGA-III, while in only 20.2%, it is the other way around.

Table 10. Average set coverage between MNSGA-III and MOEA/D.

	C-metric (MNSGA-III, MOEA/D)	C-metric (MOEA/D, MNSGA-III)
Q1&Q2	**79.3**	13.3
Q1&Q3	**60.6**	20.7
Q1&Q4	**53.4**	19.3
Q2&Q3	**34.2**	18.7
Q2&Q4	29.1	**31.6**
Q1&Q2&Q3	**76.1**	15.4
Q1&Q2&Q4	**61.5**	20.2
Q1&Q3&Q4	**57.3**	1.98
Q2&Q3&Q4	**46.2**	13.1
Q1&Q2&Q3&Q4	**31.6**	9.46

Table 11 shows that the final result, obtained by MNSGA-III is better than that obtained by NSGA-II in terms of the C-metric. The reason is that for all test instances, except for the three two-objective instance Q1&Q2, Q1&Q4, and Q2&Q3, MNSGA-III is slightly worse than NSGA-II. This phenomenon is likely caused by the better performance of NSGA-III compared with NSGA-II with regard to high dimensional objects, and the proposed algorithm is improved according to NSGA-III. Taking instance Q1&Q2&Q3&Q4 as example, an average of 41.1% of the final solutions generated by NSGA-II are dominated by those generated by MNSGA-III, while in only 0.83%, it is the other way around.

Table 11. Average set coverage between MNSGA-III and NSGA-II.

	C-metric (MNSGA-III, NSGA-II)	C-metric (NSGA-II, MNSGA-III)
Q1&Q2	30.5	**63.8**
Q1&Q3	**64.2**	18.3
Q1&Q4	36.4	**54.7**
Q2&Q3	17.6	**65.2**
Q2&Q4	**57.2**	27.6
Q1&Q2&Q3	**72.8**	21.0
Q1&Q2&Q4	**55.2**	18.7
Q1&Q3&Q4	**51.7**	16.0
Q2&Q3&Q4	**68.9**	22.3
Q1&Q2&Q3&Q4	**41.1**	0.83

Table 12 shows that for the test instances Q2&Q3, Q2&Q4, Q1&Q2&Q3, and Q1&Q3&Q4, MNSGA-III has slightly worse C-metric than NSGA-III. Similarly, the reason is likely that the centroid of the cluster in MNSGA-III sometimes fails to capture the reference point. However, the final result obtained by MNSGA-III is slightly better than that obtained by NSGA-III with regard to the C-metric. Taking Q1&Q2 as example, an average of 63.1% of the final solutions generated by NSGA-III are dominated by those generated by MNSGA-III, while in only 21.3%, it is the other way around.

Table 12. Average set coverage between MNSGA-III and NSGA-III.

	C-metric (MNSGA-III, NSGA-III)	C-metric (NSGA-III, MNSGA-III)
Q1&Q2	**63.1**	21.3
Q1&Q3	**47.9**	23.0
Q1&Q4	**66.1**	12.8
Q2&Q3	11.6	**75.5**
Q2&Q4	33.1	**58.1**
Q1&Q2&Q3	27.5	**49.2**
Q1&Q2&Q4	**33.7**	15.8
Q1&Q3&Q4	0.91	**27.8**
Q2&Q3&Q4	**44.3**	30.7
Q1&Q2&Q3&Q4	**31.6**	11.5

5.3.3. Impact of number of candidate service

To evaluate the resilience related to redundant candidate services and operating efficiency of this proposed algorithm, two scenarios are considered. In the first scenarios, the numbers of sub tasks are set to 10, 15, and 20, and the number of related candidate services varies from 30 to 120 with an increment of 30. In the second scenarios, the number of sub tasks is set to 10 and the number of related candidate services varies from 30 to 180 with an increment of 15. The stopping criterion for all compared algorithms is that the best fitness value has to remain unchanged over the last 20 iterations, or the maximum number of iterations is reached.

In the first scenarios, the box plots of optimality of the composition obtained with CGA, HGAFOA, FPSO and HRGO are plotted and compared by varying the number of candidate services. As observed in Figure 9, the sub-task is set to 10 and the candidate services is varies from 30 to 120 with an increment of 30, when the number of candidate services is low, the optimality of the composition obtained with HRGO is slightly lower than the optimal value obtained in the case of the HFAFOA algorithm, but with the increase of candidate services the HRGO remains more efficient than the HFAFOA and CGA algorithms and especially more efficient than FPSO algorithms. Indeed, the HFAFOA, CGA and HRGO uses a selection phase that is based on the QoS level to filter out candidate services that have a positive effect on the optimality of the composition. In Figure 10 with the sub-task is set to 15 and the candidate services is still varies from 30 to 120 and in Figure 11 with the sub-task is change to 20 and the candidate services is still varies from 30 to 120, we observe that the optimality of the composition for the HRGO and HGAFOA algorithms still close to 90%, when the number of concrete services increases. This discovery can be interpreted as that as the number of candidate services increases, the probability of high-quality candidate services being discovered also increases, and the probability of obtaining the close-to-optimal solution also increases. In addition, an advantage of the HRGO is that it provides a composition that is very close to the optimum, with a much more reduced elapsed time in comparison to the HFAFOA, CGA and FPSO algorithms.

In the second scenarios, the elapsed time of above four algorithms are evaluated. Figure 12 shows that the elapsed time of almost all four algorithms increases with the number of candidate services. This result can be interpreted to the time needed to find the suitable candidate services which belong to a feasible composition increases with the number of candidate services. Figure 8 also shows that the elapsed time of the CGA, HGAFOA and HRGO increases slightly with the increase of number of candidate services. The reason of reduction in execution time is the candidate services filter based on the fuzzy similarity degree used in the HRGO drastically reduces the number of candidate services before optimization process, namely, reduce the search space of compositions. The filtering process used in the HRGO is very efficient when the number of candidate services is abundance because of the significant reduction of the search space of the compositions.

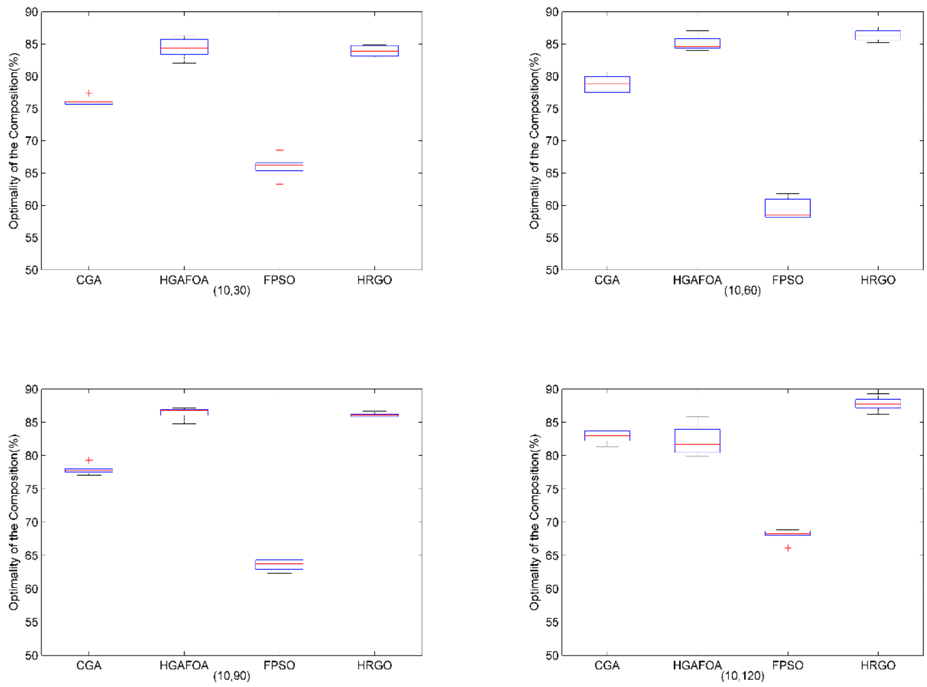

Figure 9. Optimality of the composition versus the number of sub-task is 10 and candidate services varies from 30 to 120.

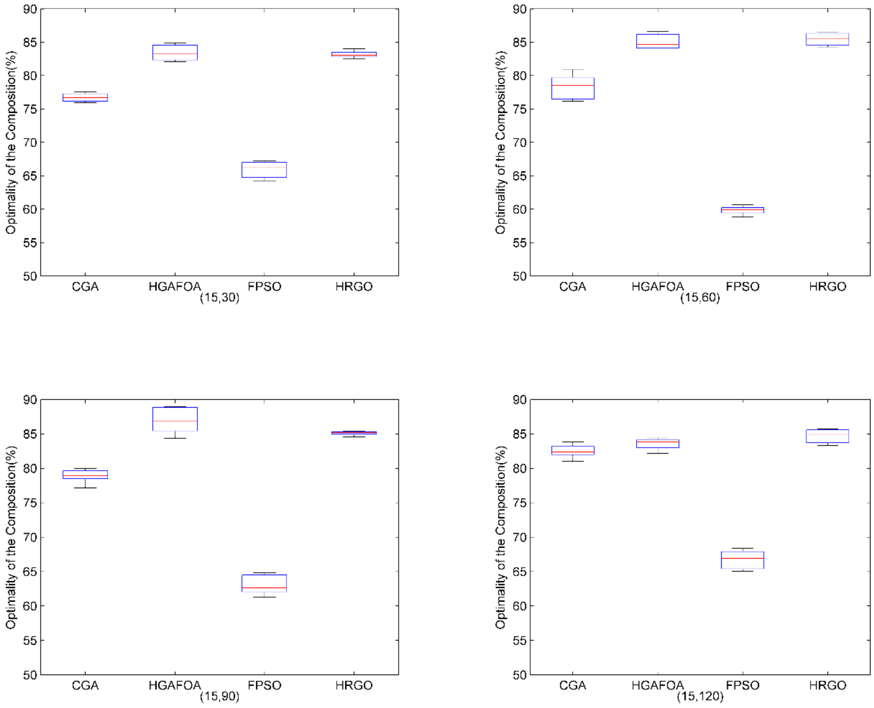

Figure 10. Optimality of the composition versus the number of sub-task is 15 and candidate services varies from 30 to 120.

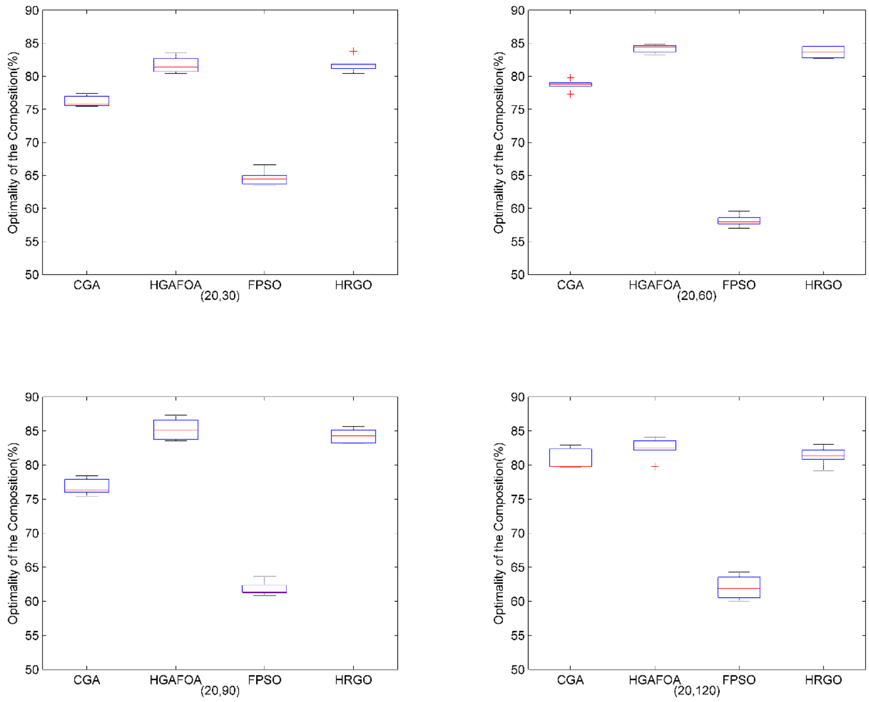

Figure 11. Optimality of the composition versus the number of sub-task is 20 and candidate services varies from 30 to 120.

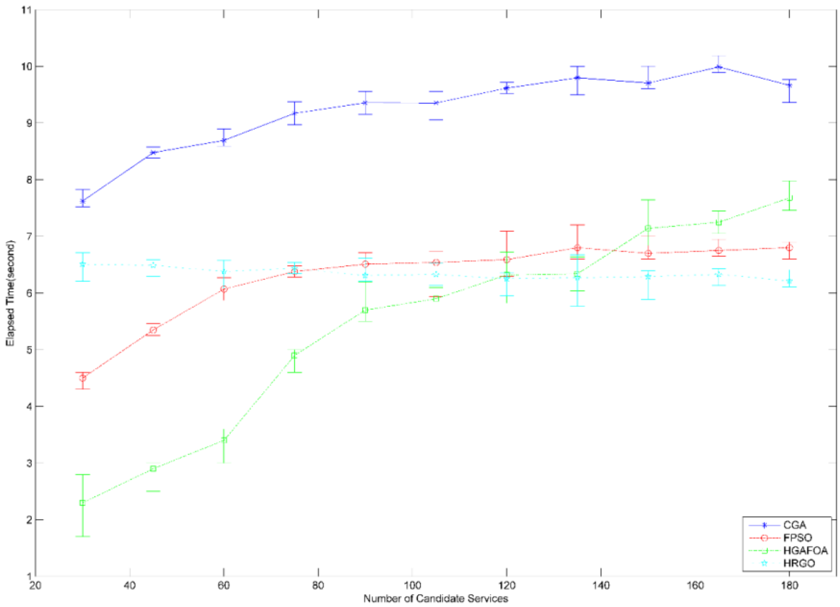

Figure 12. Elapsed time versus the number of candidate services.

6. Conclusions

SCOS is one of the most pivotal problems in the field of CM. To increase the ability of manufacturing enterprises to resist the interference of uncertainty, a hybrid resilience-aware SCOS is introduced. In general, three strategies are proposed to ensure consistency in the manufacturing process:

The QoS attributes that refer to a system with the capacity to return to its state of equilibrium after disruptive perturbation are described and modeled mathematically. A candidate services filtering strategy, based on the fuzzy similarity degree, is proposed to filter services and enhance the resilience of CM. Based on diversity judgment and dual-track parallelism, the MNSGA-III is proposed to solve the combinatorial optimization problem. To validate the efficiency of the proposed HRGO algorithm for solving SCOS problems, a series of experiments were designed and conducted. The results showed that compared with MOEA/D and NSGA-III, in general, MNSGA-III performs well in terms of the diversity of the non-dominated solution. Compared with MOEA/D, NSGA-II, and NSGA-III, MNSGA-III performs well in the majority of experimental cases in terms of the quality of POF. Compared with CGA, FPSO, and HGAFOA, HRGO remains superior in terms of the optimality of the composition, at the expense of a negligible increase of the average elapsed time.

In the structure of the MNSGA-III, by judging the diversity of a population, the two-track parallel environment selection mechanism is used to select individuals from the offspring. When the population has good diversity, selecting individuals from different clusters can obtain the Pareto-optimal solutions in the distributed regions of POF. In contrast, when the population has poor diversity, it tends to use the reference point to select individuals, and Pareto-optimal solutions in distributed regions of POF can also be obtained. The advantage of this proposed algorithm is that it makes the best use of the statistical distribution characteristics of the population in each generation, rather than forcing the use of reference points to select specific offspring. This is appropriate for the addressed problem characteristics. The disadvantage of the proposed method is that even with its library functions, FCM consumes considerable computing resources. For MNSGA-III alone, the use of FCM is connected with the risk to slow down convergence. Fortunately, this service fuzzy filtering strategy decreases this risk. In addition, setting the value of the penalty factor θ directly affects the degree of oscillation of the population close to the POF. This setting was based on the empirical value of 0.25, as recommended in the literature; however, for large-scale combinatorial optimization problems, fixed values are difficult to adapt. Therefore, future research should reduce the computational complexity of the clustering step and design an adaptive discriminant factor. In general, the algorithm structure proposed in this paper can be easily modified and applied to other combinatorial optimization problems.

Acknowledgements

This work is supported by 'the National Natural Science Foundation of China' (No. 71901086, No.72071060, No.71690230, No.71690235), 'the Natural Science Foundation of Anhui Province' (No. 1808085QG220).

Conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

1. S. Mejjaouli, R. F. Babiceanu, T. Borangiu, D. Trentesaux, *Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics*, Springer International Publishing, (2014), 31–46.
2. R. F. Babiceanu, R. Seker in T. Borangiu, A. Thomas, *Service Orientation in Holonic and Multi-Agent Manufacturing*, Springer International Publishing, (2015), 165–173.
3. Y. Wang, A Formal Model of QoS-Aware Web Service Orchestration Engine, *IEEE Trans. Network Serv. Manage.*, **13** (2016), 113–125.
4. J. Tapolcai, P. Cholda, T. Cinkler, K. Wajda, A. Jajszczyk, A. Autenrieth, et al., *Quality of resilience (QoR): NOBEL approach to the multi-service resilience characterization*, 2nd International Conference on Broadband Networks, 2005. Available from: https://ieeexplore.ieee.org/abstract/document/1589762.
5. S. K. Pradhan, S. Routroy, Development of supply chain risk mitigation strategy: a case study, *Int. J. Procurement Manage.*, **7** (2014), 359–375.
6. S. Huang, S. Zeng, Y. Fan, G. Q. Huang, Optimal service selection and composition for service-oriented manufacturing network, *Int. J. Comput. Integr. Manuf.*, **24** (2011), 416–430.
7. Q. Zhang, X. Lu, Z. Peng, M. Ren, Perspective: a review of lifecycle management research on complex products in smart-connected environments, *Int. J. Prod. Res.*, **57** (2019), 6758–6779.
8. J. Yi, C. Lu, G. Li, A literature review on latest developments of Harmony Search and its applications to intelligent manufacturing, *Math. Biosci. Eng.*, **16** (2019), 2086–2117.

9. Y. Suresh, T. Kalaivani, J. Senthilkumar, V. Mohanraj, BF2 VHDR based dynamic routing with hydrodynamics for QoS development in WSN, *Math. Biosci. Eng.*, **17** (2020), 930–947.
10. M. S. Rahman, I. Khalil, A. Alabdulatif, X. Yi, Privacy preserving service selection using fully homomorphic encryption scheme on untrusted cloud service platform, *Knowl. Based Syst.*, **180** (2019), 104–115.
11. Y. Cao, S. Wang, L. Kang, Y. Gao, A TQCS-based service selection and scheduling strategy in cloud manufacturing, *Int. J. Adv. Manuf. Technol.*, **82** (2016), 235–251.
12. B. Huang, C. Li, F. Tao, A chaos control optimal algorithm for QoS-based service composition selection in cloud manufacturing system, *Enterp. Inf. Syst.*, **8** (2014), 445–463.
13. Y. K. Lin, C. S. Chong, Fast GA-based project scheduling for computing resources allocation in a cloud manufacturing system, *J. Intell. Manuf.*, **28** (2017), 1189–1201.
14. Y. Laili, F. Tao, L, Zhang, B. R. Sarker, A study of optimal allocation of computing resources in cloud manufacturing systems, *Int. J. Adv. Manuf. Technol.*, **63** (2012), 671–690.
15. W. Liu, B. Liu, D. Sun, Y. Li, G. Ma, Study on multi-task oriented services composition and optimisation with the 'Multi-Composition for Each Task' pattern in cloud manufacturing systems, *Int. J. Comput. Integr. Manuf.*, **26** (2013), 786–805.
16. M. Alrifai, T. Risse, A Hybrid Approach for Efficient Web Service Composition with End-to-End QoS Constraints, *ACM Trans. Web*, **2** (2012), 7.
17. W. Xu, S. Tian, Q. Liu, Y. Xie, Z. Zhou, D. T, Pham, An improved discrete bees algorithm for correlation-aware service aggregation optimization in cloud manufacturing, *Int. J. Adv. Manuf. Technol.*, **84** (2016), 17–28.
18. J. Zhou, X. Yao, Multi-population parallel self-adaptive differential artificial bee colony algorithm with application in large-scale service composition for cloud manufacturing, *Appl. Soft Comput.*,**56** (2017), 379–397.
19. F. Xiang, Y. Hu, Y. Yu, H. Wu, QoS and energy consumption aware service composition and optimal-selection based on Pareto group leader algorithm in cloud manufacturing system, *Cent. Eur. J. Oper. Res.*, **22** (2014), 663–685.
20. H. Jin, X. Yao, Y. Chen, Correlation-aware QoS modeling and manufacturing cloud service composition, J. Intell. Manuf., **28** (2017), 1947–1960.
21. W. Zhang, Y. Yang, S. Zhang, D. Yu, Y. Xu, A New Manufacturing Service Selection and Composition Method Using Improved Flower Pollination Algorithm, *Math. Probl. Eng.*, **2016** (2016), 1–12.
22. J. Zhou, X. Yao, DE-caABC: differential evolution enhanced context-aware artificial bee colony algorithm for service composition and optimal selection in cloud manufacturing, *Int. J. Adv. Manuf. Technol.*, **90** (2017), 1085–1103.
23. J. Lartigau, X. Xu, L. Nie, D. Zhan, Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm, *Int. J. Prod. Res.*, **53** (2015), 4380–4404.
24. J. Zhou, X. Yao, Hybrid teaching–learning-based optimization of correlation-aware service composition in cloud manufacturing, *Int. J. Adv. Manuf. Technol.*, **91** (2017), 3515–3533.
25. Y. Cao, S. Wang, L. Kang, C. Li, L. Guo, Study on machining service modes and resource selection strategies in cloud manufacturing, *Int. J. Adv. Manuf. Technol.*, **81** (2015), 597–613.

26. F. Seghir, A. Khababa, A hybrid approach using genetic and fruit fly optimization algorithms for QoS-aware cloud service composition, *J. Intell. Manuf.*, **29** (2018), 1773–1792.
27. B. Xu, Z. Sun, A fuzzy operator based bat algorithm for cloud service composition, *Int. J. Wireless Mobile Comput.*, **11** (2016), 42–46.
28. J. Zhou, X. Yao, Multi-objective hybrid artificial bee colony algorithm enhanced with Lévy flight and self-adaption for cloud manufacturing service composition, *Appl. Intell.*, **47** (2017), 721–742.
29. B. Liu, Z. Zhang, QoS-aware service composition for cloud manufacturing based on the optimal construction of synergistic elementary service groups, *Int. J. Adv. Manuf. Technol.*, **88** (2017), 2757–2771.
30. X. Gu, X. Jin, J. Ni, Y. Koren, Manufacturing System Design for Resilience, *Procedia CIRP*, **36** (2015), 135–140.
31. W. J. Zhang, C. A, van Luttervelt, Toward a resilient manufacturing system, *CIRP Ann.*, **60** (2011), 469–472.
32. L. Zhang, H. Guo, F. Tao, Y. L. Luo, Flexible management of resource service composition in cloud manufacturing, *2010 IEEE International Conference on Industrial Engineering and Engineering Management*, 2010. Available from: https://ieeexplore.ieee.org/abstract/document/5674175/.
33. F. Tao, L. Zhang, Y. Liu, Y. Cheng, L. Wang, X. Xu, Manufacturing Service Management in Cloud Manufacturing: Overview and Future Research Directions, *J. Manuf. Sci. Eng. Trans.*, **137** (2015).
34. S. S. Shah, R. F. Babiceanu, Resilience Modeling and Analysis of Interdependent Infrastructure Systems, *Syst. Inf. Eng. Des. Symp.*, **2015** (2015), 154–158.
35. R. Francis, B. Bekera, A metric and frameworks for resilience analysis of engineered and infrastructure systems, *Reliab. Eng. Syst. Saf.*, **121** (2014), 90–103.
36. M. Ouyang, Review on modeling and simulation of interdependent critical infrastructure systems, *Reliab. Eng. Syst. Saf.*, **121** (2014), 43–60.
37. P. Érdi, *Complexity Explained*, Springer Science & Business Media, (2007).
38. W. Y. Zhang, S. Zhang, M. Cai, J. X. Huang, A new manufacturing resource allocation method for supply chain optimization using extended genetic algorithm, *Int. J. Adv. Manuf. Technol.*, **53** (2011), 1247–1260.
39. M. R. Namjoo, A. Keramati, Analysing Causal dependencies of composite service resilience in cloud manufacturing using resource-based theory and DEMATEL method, *Int. J. Comput. Integr. Manuf.*, **31** (2018), 942–960.
40. T. Y. Tseng, C. M. Klein, New algorithm for the ranking procedure in fuzzy decision-making, *IEEE Trans. Syst. Man Cyber.*, **19** (1989), 1289–1296.
41. K. Sindhya, K. Miettinen, K. Deb, A Hybrid Framework for Evolutionary Multi-Objective Optimization, *IEEE Trans. Evol. Comput.*, **17** (2013), 495–511.
42. Y. Hua, Y. Jin, K. Hao, A Clustering-Based Adaptive Evolutionary Algorithm for Multi-objective Optimization with Irregular Pareto Fronts, *IEEE Trans. Cyber.*, **49** (2019), 2758–2770.
43. K. Deb, H. Jain, An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints, *IEEE Trans. Evol. Comput.*, **18** (2014), 577–601.

44. Hui. Li, Q. F. Zhang, Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II, *IEEE Trans. Evol. Comput.*, **13** (2009), 284–302.
45. K. Deb, R. B. Agrawal, Simulated Binary Crossover for Continuous Search Space, *Complex Syst.*, **9** (1995), 115-148.
46. P. M. Pardalos, I. Steponavičė, A. Žilinskas, Pareto set approximation by the method of adjustable weights and successive lexicographic goal programming, *Optim. Lett.*, **6** (2012), 665–678.
47. K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, **6** (2002), 182–197.
48. S. Guo, B. Du, Z. Peng, X. Huang, Y. Li, Manufacturing resource combinatorial optimization for large complex equipment in group manufacturing: A cluster-based genetic algorithm, *Mechatronics*, **31** (2015), 101–115.
49. H. Zheng, Y. Feng, J. Tan, A fuzzy QoS-aware resource service selection considering design preference in cloud manufacturing system, *Int. J. Adv. Manuf. Technol.*, **84** (2016), 371–379.
50. Q. F. Zhang, H. Li, MOEA/D: A Multi-objective Evolutionary Algorithm Based on Decomposition, *IEEE Trans. Evol. Comput.*, **11** (2007), 712–731.
51. M. E. Khanouche, F. Attal, Y. Amirat, A. Chibani, M. Kerkar, Clustering-based and QoS-aware services composition algorithm for ambient intelligence, *Infor. Sci.*, **482** (2019), 419–439.
52. H. Ishibuchi, N. Akedo, Y. Nojima, Behavior of Multiobjective Evolutionary Algorithms on Many-Objective Knapsack Problems, *IEEE Trans. Evol. Comput.*, **19** (2015), 264–283.
53. M. R. Namjoo, A. Keramati, S. A. Torabi, F. Jolai, Quantifying the Resilience of Cloud-Based Manufacturing Composite Services, *Int. J. Cloud Appl. Comput.*, **8** (2018), 88–117.