Bilateral Collaborative Optimization for Cloud Manufacturing Service

Bin Xu[1, 2], Yong Tang[1], Yi Zhu[1], Wenqing Yan[1], Cheng He[3] and Jin Qi[1, *]

Abstract: Manufacturing service composition of the supply side and scheduling of the demand side are two important components of Cloud Manufacturing, which directly affect the quality of Cloud Manufacturing services. However, the previous studies on the two components are carried out independently and thus ignoring the internal relations and mutual constraints. Considering the two components on both sides of the supply and the demand of Cloud Manufacturing services at the same time, a Bilateral Collaborative Optimization Model of Cloud Manufacturing (BCOM-CMfg) is constructed in this paper. In BCOM-CMfg, to solve the manufacturing service scheduling problem on the supply side, a new efficient manufacturing service scheduling strategy is proposed. Then, as the input of the service composition problem on the demand side, the scheduling strategy is used to build the BCOM-CMfg. Furthermore, the Cooperation Level (CPL) between services is added as an evaluation index in BCOM-CMfg, which reveals the importance of the relationship between services. To improve the quality of manufacturing services more comprehensively. Finally, a Self-adaptive Multi-objective Pigeon-inspired Optimization algorithm (S-MOPIO) is proposed to solve the BCOM-CMfg. Simulation results show that the BCOM-CMfg model has advantages in reliability and cost and S-MOPIO can solve BCOM-CMfg effectively.

Keywords: Service composition, service scheduling, bilateral collaborative optimization, evolutionary computation, PIO.

1 Introduction

Cloud Manufacturing is a new service-oriented manufacturing model. Intricate manufacturing tasks are decomposed into subtasks, which composes various candidate services. Due to the increasing user demand and the expansion of manufacturing service, the manufacturing network is very complex [Zhao and Peng (2019)], it is becoming more and more difficult for researchers to choose the right services from the massive manufacturing service cloud pool to form the best service composition. It has become a research focus both in the academic and industrial fields. Recently, research on the

[1] Nanjing University of Posts and Telecommunications, Nanjing, 21000, China.
[2] Nanjing pharmaceutical Co., Ltd., Nanjing, 21000, China.
[3] University of New South Wales, Sydney, 2000, Australia.
[*] Corresponding Author: Jin Qi. Email: qijin@njupt.edu.cn.
Received: 22 April 2020; Accepted: 14 May 2020.

service composition on the demand side of Cloud Manufacturing has been carried out and preliminary results have been achieved. A method for optimal transmission of hierarchical network for heterogeneous service in Cloud Scenarios was presented [Huang, Bai, Liu et al. (2018)]. It provides technical support for the construction of the Cloud Manufacturing platform. A method based on the combination of fuzzy hierarchical analysis and grey relational analysis is proposed to deal with the service composition [Peng and Meng (2016)]. A new Quality of Service evaluation model of manufacturing service composition is proposed according to the networked collaboration model in Cluster Supply Chains [Xue, Wang and Lu (2016)]. After considering the evaluation and selection of manufacturing resources in networked manufacturing, a manufacturing resources evaluation system including time, quality, cost, service, green and other indexes are established, and a manufacturing resources evaluation model is put forward combined with Analytic Hierarchy Process, comparative judgment and fuzzy theory [Chen, Huang, Lin et al. (2014)]. A cross-tenant role-based access control model for collaborative cloud services is proposed, which is beneficial to the utilization and popularization of cloud services [Liu and Xia (2019)]. An API management system is designed to facilitate efficient service composition [Wang, Sun, Zhang et al. (2019)].

However, the above studies do not take into account the relationship between manufacturing services, especially the cooperative relationship between services while it is valuable. During the industrial manufacturing process, the environment of Cloud Manufacturing services is dynamic and uncertain. There is a potential relationship between different Cloud Manufacturing services, which is critical to the quality of Cloud Manufacturing services. If Cloud Manufacturing tasks are accomplished by services that cooperate frequently, product quality, manufacturing cycle and reliability will have more advantages. Therefore, it is necessary to study the CPL between services in the manufacturing service composition.

Besides, manufacturing service scheduling also has an important impact on service composition. The service scheduling scheme determines whether the results of service composition can be effectively implemented. The research on service scheduling in the field of Cloud Manufacturing mainly focuses on resource scheduling in the Job Shop layer of Cloud Manufacturing, the resource search strategy in the Cloud Manufacturing environment and the intelligent algorithm based on scheduling model. Lee et al. [Lee and Katz (2011)] investigate the Job Shop scheduling under scheduling tasks, resource and process model and scheduling algorithm, and study scheduling optimization algorithms under different tasks in consideration of workflow and resource heterogeneity. An energy consumption model is proposed to compute the energy consumption for a machine in different states [Wu and Sun (2018)]. A new Job Shop scheduling method based on a digital twin is proposed to reduce the scheduling deviation and a prototype system is designed to verify it [Fang, Peng, Ping et al. (2019)]. Sun et al. [Sun, Lin, Gen et al. (2019)] propose an effective hybrid cooperative coevolution algorithm for the minimization of fuzzy make span. Kim et al. [Kim, Jin and Sun (2011)] focus on process manufacturing in discrete manufacturing Job Shop and elaborate the applicable conditions and characteristics of different dynamic scheduling methods. However, the research on manufacturing service scheduling has not been considered together with Cloud Manufacturing service composition optimization. The manufacturing service scheduling scheme in the Cloud Manufacturing environment directly affects whether the manufacturing scheme can be effectively implemented, it is necessary to make a scheduling strategy in real-time according to the service provider's production resources.

The purpose of this paper is to construct BCOM-CMfg by considering the collaborative optimization of Cloud Manufacturing service composition and manufacturing service scheduling. Furthermore, an algorithm to solve BCOM-CMfg is proposed, which is called S-MOPIO. The main contributions of this paper are as follows: 1) The CPL between manufacturing services is proposed to evaluate the relationship between services, which is added to BCOM-CMfg to reveal the importance of the relationship between services. 2) The S-MOPIO algorithm is proposed based on an adaptive learning mechanism, which is used to solve the Efficient Service Scheduling Model and BCOM-CMfg. 3) The simulation results show that the effect of the S-MOPIO algorithm for solving BCOM-CMfg is better than MOPIO, MOPSO [Saremi and Mirjalili (2020)] and CPSMOEA [Zhang and Zhou (2015)]. Besides, BCOM-CMfg is beneficial to reduce manufacturing costs and improve the reliability of Cloud Manufacturing.

2 System model and problem formulation

2.1 Manufacturing service composition optimization model

In Cloud Manufacturing environment, a complex manufacturing task I , $I = \{1,2,3, \dots i, \dots k\}$, where subtask i contains multiple candidate services. Manufacturing task I can be completed with different service composition X=$\{x_1, x_2, x_3, \dots\ x_i, \dots\ x_k\}$, where x_i is the service selected when subtask i is completed, Ω is a collection of all possible service compositions of task I. In general, there are many different evaluation indexes for service x_i, such as time, cost, reliability of service, etc. Therefore, we need to evaluate the different indexes of X and choose the best one after comparison. T_i, C_i and R_i are used to represent the time, cost, and reliability of the scheme of service x_i respectively. In general, T_i is dynamic. Except for the time, cost and reliability of Cloud Manufacturing services, we propose Cooperation Level, a new evaluation index, L_{ik} is used to describe the CPL between candidate services x_i and x_k. A detailed description of each index are as follows.

2.1.1 Time of service

In this paper, the time of service is dynamic while the time of completing subtasks fluctuates with the idle condition of the machine, so the time of service is evaluated in real-time according to the Efficient Service Scheduling Model in Section 2.2.

2.1.2 Cost of service

The service cost includes the cost of labor, the cost of raw materials, the cost of resource scheduling and other costs.

$$C_i = \mu_1 C_i^{labor} + \mu_2 C_i^{materials} + \mu_3 C_i^{scheduling} + \mu_4 C_i^{other} \tag{1}$$

where C_i^{labor} is the cost of labor, $C_i^{materials}$ is the cost of raw materials, $C_i^{scheduling}$ is the cost of service scheduling, C_i^{other} is other cost. μ_i is the weight of each element of costs. $\mu_i \in (0,1)$, it depends on the actual situation of the service provider, For example, the main costs come from materials and labor to most small manufacturing enterprises, but the cost of resource scheduling in some large manufacturing enterprises occupy some proportion. In the experiment, we will generate μ_i randomly in a range.

2.1.3 Reliability of service

The reliability of service is evaluated from the perspectives of Cloud Manufacturing products and Cloud Manufacturing service providers. To improve the reliability of products, the production process needs to be upgraded, but it will not be improved greatly in a short time. However, the reliability of Cloud Manufacturing service providers can be improved, which affected by excessive load, unreasonable production resource scheduling and many external factors, so we should pay more attention to Cloud Manufacturing service providers.

$$R_i = \varphi_1 R_i^{product} + \varphi_2 R_i^{manufacturer} + \varphi_3 R_i^{other} \tag{2}$$

As shown in Eq. (2), $R_i^{product}$ is the reliability of the product, $R_i^{manufacturer}$ is the reliability of service providers. $\varphi_i \in (0,1)$, it depends on the actual situation of the service providers.

2.1.4 CPL between services

In this paper, we use the contract value, quantity and duration between service providers to assess the Cooperation Level of service. The calculation method is shown in Eq. (3). α is the sum of contract price between the services, β is the total number of phase cooperation between services, γ is the duration of cooperation between the services. These data are available periodically and will be updated continuously.

$$L_{ik} = \frac{1}{1 + e^{-(\omega_1 \alpha^{ik} + \omega_2 \beta^{ik} + \omega_3 \gamma^{ik})}} - 0.5 \tag{3}$$

where α^{ik} is the total contract price between x_i and x_k, β^{ik} is total contract number, γ^{ik} is total contract duration, ω_1, ω_2, ω_3 are the weights.

Based on the analysis of service indexes, a new service composition model in Cloud Manufacturing is described in Fig. 1, First of all, the data of Cloud Manufacturing services are collected and preprocessed. Secondly, the processed data is calculated by CPL mode, the CPL value is obtained, it will be stored in the CPL database, which will be updated by optimization result. The detail of the Dynamic Evaluation Model will be described in Section 2.2. The outputs of the Efficient Scheduling Model and the CPL Dynamic Evaluation Model are used as the time and CPL of manufacturing service respectively, it is inputted into the multi-objective optimization model together with the reliability and cost. Finally, the optimization results are output. It is worth noting that the Efficient Scheduling Model can also output scheduling solutions while outputting the time of manufacturing service.

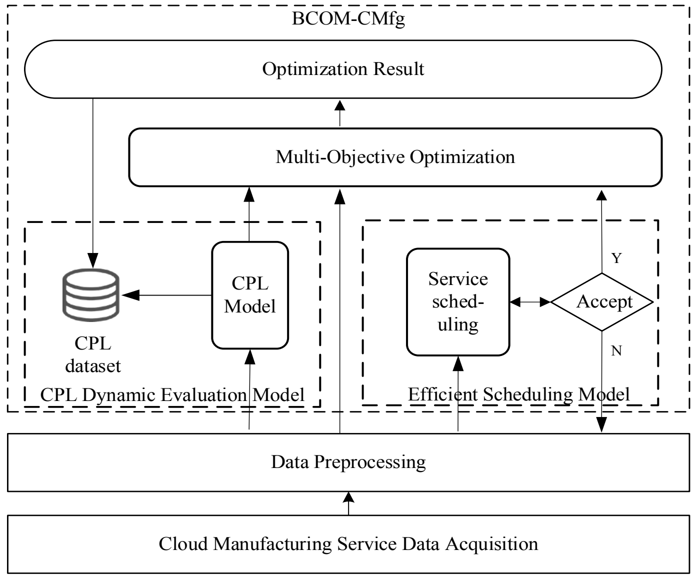

Figure 1: Bilateral collaborative optimization model of Cloud Manufacturing

In Eq. (4), as shown in Eq. (4), BCOM-CMfg is established:

$$\min_{X\in\Omega} f(X) = \Big(\sum_{x_i\in X} T_i, \sum_{x_i\in X} C_i, -\prod_{x_i\in X} R_i\,, -\sum_{x_i,x_k\in X, i\neq k} L_{ik}\Big) \tag{4}$$

$$s.t.\quad \begin{cases} \sum_{x_i\in X} T_i \leq T^{max} \\ \sum_{x_i\in X} C_i \leq C^{max} \\ \sum_{x_i,x_k\in X, i\neq k} L_{ik} \geq L^{min} \\ \sum_{x_i\in X} R_i \geq R^{min} \end{cases}$$

where T^{max} is the upper limit of time to complete a complex manufacturing task, C^{max} is the upper limit of cost, L^{min} is the minimum CPL of *X*, R^{min} is the minimum reliability of *X*. BCOM-CMfg can not only provide the best combination of manufacturing services for service demanders, but also provide timely production scheduling solutions for service providers. It can improve the reliability of Cloud Manufacturing services and reduce the investment of service providers in manufacturing service scheduling.

2.2 Efficient manufacturing service scheduling model

The theoretical research of the scheduling mainly focuses on the scheduling problem based on the minimum scheduling time represented by the manufacturing service scheduling. As we know, a complex manufacturing system involves massive manufacturing service scheduling processes, which makes the solution to the scheduling problem very complex. Because of the large solution space and the nesting nature of the scheduling problem, the graphical method becomes impractical. However, the optimization algorithm has a good performance in solving high-dimensional multi-objective optimization problems. The mapping of the solution vector to the scheduling scheme is the primary problem of using the optimization algorithm to solve the scheduling problem. In this paper, the two-dimensional vector is used to implement the mapping of the solution vector to the scheduling scheme. Assume that a manufacturing subtask i is decomposed into w artifacts in the Cloud Manufacturing environment, and u machines are required to complete the subtask i, we treat processing machines or production lines as manufacturing services. The subtasks are sequentially executed on certain machines according to the requirements, and each machine can only perform one manufacturing task at a time. $N=\{1, 2, 3,... n, ... w\}$ denotes the set of artifacts needed to complete subtask i, $M=\{1, 2, 3, ... m ... u\}$ is the set, that the element is production equipment required by subtask i, $J=\{1, 2, 3, ... j, ... v\}$ is the steps of completing the workpiece n. $X=\{x_1, x_2, x_3, x_j, ... x_{w*v}\}$ is the scheduling result of the subtask i. x_j is a two-dimensional vector, the first dimension of x_j is called the process vector, which can represent all the processes required to complete the workpiece. If the workpiece No. 3 to completed need 5 processes, the process vector should contain five Roman numerals 3 as elements. The second dimension of x_j is called the solution vector, which contains the priority of each process. The solution vector is iterated according to the optimization algorithm. The two-dimensional vector is sorted according to the evolution result while the process vector also changes in sequence with the ordering of the solution vector. A sequence of different process vectors can represent a set of scheduling schemes.

To accomplish efficient manufacturing, we establish a multi-objective manufacturing service scheduling model considering manufacturing time and equipment utilization rate to manage the scheduling and distribution of manufacturing resources. t_{nj}^m is the j-th process of the workpiece n in the processing time of the machine m. If all the machines start at the same time and all the machines close right after completing the last workpiece of the subtask while the time of the workpiece transferring between different machines is not taken into account, the total time taken by machine m from startup to shutdown is ST^m. For a single machine m, the total machining time of the machine is $\sum_{n\in N, j\in J} t_{nj}^m$, $t_{spare}^m(X)$ is the idle time of the machine m by X. As shown in Eq. (5), $T(X)$ is the total completion time of subtask X.

$$T(X) = \max_{m\in M}\{ST^m\} = \max_{m\in M}\left\{t_{spare}^m(X) + \sum_{n\in N, j\in J} t_{nj}^m\right\} \tag{5}$$

where $E(X)$ is the device utilization rate to complete a subtask, u is the number of machines to complete the subtask.

$$E(X) = \frac{1}{u}\sum_{m\in M}\frac{\sum_{n\in N, j\in J} t_{nj}^m}{T(X)} \tag{6}$$

Eq. (7) is the objective function of the multi-objective manufacturing service scheduling model, where E^{min} is the minimum efficiency allowed for subtask completion, T^{max} is the maximum time allowed for subtask completion. Also, there is a time relationship between the processes, which is the latter process needs to be processed after the completion of the previous process.

$$\min f(X) = (T(X), -E(X))$$
$$s.t. \quad \begin{cases} E(X) \geq E^{min} \\ T(X) \leq T^{max} \end{cases} \tag{7}$$

3 Method

Both Eqs. (4) and (7) are the equations of complex multi-objective optimization problems, so we propose S-MOPIO to solve them. S-MOPIO is improved based on the Pigeon-Inspired Optimization (PIO) algorithm [Dou and Duan (2016)], which is a group intelligence optimization algorithm emerged in recent years. In PIO, pigeons use different navigation tools in different stages of finding targets, the map and compass operator and landmark operator are proposed to update the position and speed of individuals. In the first stage, the map and the compass operator is introduced. The pigeons can sketch the homing in the brain through their induction of the earth's magnetic field, then clarify the direction of flight according to the direction of the sun. In the D-dimensional search space, the speed and position of the pigeons are updated according to Eqs. (8) and (9) in each iteration.

$$V_i^g = V_i^{g-1} * e^{-R*g} + r * (X_{gbest} - X_i^{g-1}) \tag{8}$$

$$X_i^g = X_i^{g-1} + V_i^g \tag{9}$$

where X_i and V_i are the position and speed of the pigeons, g is the number of iterations, r is a random number between 0 and 1, R is the map and compass factor, $R \in (0,1)$. X_{gbest} is the global optimal position, found by Grid Search.

In the second stage, the landmark operator is introduced. After the pigeons approach the destination, they will navigate according to the familiar landmarks. If there are familiar landmarks near the individuals, they will fly directly to the destination. Otherwise, they will fly with other pigeons, which are familiar with the landmarks. During each iteration, the number of pigeons is halved according to Eq. (10), while the first half of the population with better fitness is selected as the current population. X_c is the central position of the current pigeons, the flight reference direction is calculated according to Eq. (11), f is fitness function, ε is a constant close to 0, the pigeon position is updated according to Eq. (12).

$$N^g = \frac{N^{g-1}}{2} \tag{10}$$

$$X_c^{g-1} = \frac{\sum_{i=1}^{N^{g-1}} X_i^{g-1}\left(f\left(X_i^{g-1}\right) + \varepsilon\right)^{-1}}{N^{g-1} \sum_{i=1}^{N^{g-1}}\left(f\left(X_i^{g-1}\right) + \varepsilon\right)^{-1}} \tag{11}$$

$$X_i = X_i^{g-1} + r * \left(X_c^{g-1} - X_i^{g-1}\right) \tag{12}$$

where N^g is the population size of $g\text{-}th$ iteration, X_i is the position of the pigeon. Based on investigating the existing academic achievements and fully considering the characteristics of different operators, the Pigeon Swarm Optimization algorithm, Particle Swarm Optimization Algorithm and Differential Evolution algorithm are introduced into the strategy pool.

In Eqs. (13) and (14), the reward of each operator is calculated, then the roulette method is used to select the operator according to the probability selection algorithm. The operator to be selected for the next evolution is determined according to the historical performance of the operator.

$$V_m^g = a * IGD_m^g - b * HV_m^g \tag{13}$$

$$Q_m^g = \sum_{g=3}^{n} \eta^{-g}\left(V_m^{g\text{-}1} - V_m^{g\text{-}2}\right) \tag{14}$$

where g is the number of iteration, IGD_m^g is the Inverted Generation Distance (IGD) [Leonardo, Manuel and Thomas (2017)] value of population when evolve by the $m\text{-}th$ operator at $g\text{-}th$ iteration. HV_m^g is the Hypervolume (HV) [Yang, Emmerich, Deutz et al. (2019)] value of population when evolve by the $m\text{-}th$ operator at $g\text{-}th$ iteration. a is the weight of the IGD_m^g, b is the weight of HV_m^g. V_m^g is used to evaluate the effect of population evolution, it will be added into V_{EH} (Evolutionary history). As shown in Eq. (14), Q_m^g is the reward that $m\text{-}th$ operator is selected at $g\text{-}th$ iteration, η is the learning rate. The probability that the $m\text{-}th$ operator is selected to evolve the $g\text{-}th$ iteration population is proportional to Q_m^g, it will be added into M_P (Policy Matrix). The pseudo-code of S-MOPIO is shown in Tab. 1.

Table 1: The pseudo-code of S-MOPIO

S-MOPIO

Input: N (Population Size), $\overline{N}$(Archive Size), G(The maximum number of iterations), Mesh_div(Mesh dimension).

Output: $\overline{P}_t$(nondominated set)

Initialization: N=100; $\overline{N} = 100$; $G = 1000$; M_P=[1/3,1/3,1/3]; the initial value of V_{EH} is null matrix; P_t is randomly generated, Mesh_div=10.

while $g < G$ **do**

  Calculate fitness of population;

  Add non-dominated solutions to $\overline{P}_t$;

  Find $gbest$ by Mesh method in $\overline{P}_t$;

  Calculate V_m^g by Eq. (13) and update V_{EH};

  Calculate Q_m^g by Eq. (14) to update M_P;

  Select evolution operator to update the population according to M_P;

  $g = g + 1$;

end

4 Simulation and analysis

The purpose of this section of the experiment is to verify the effectiveness of S-MOPIO in solving the BCOM-CMfg. The QWS 2.0 dataset is used in the experiment, which was collected by Guelph University [Al-Masri and Mahmoud (2007)]. We randomly expand the dataset to 4000 within the reasonable range of the original data. At the same time, due to the lack of CPL in the dataset, we simulated the relevant data of the total contract price, the total number of contracts and the total length of the contracts, which are used to calculate the value of CPL. The real Pareto Front is formed by combining the Non-dominant solutions of the algorithms involved in this paper and Non-dominant sorting them based on the Pareto principle.

To compare the performance of the algorithm, IGD and HV are used to evaluate the performance of the algorithm. The smaller the IGD is, the closer the non-dominant dissociation True Pareto Front is. A larger HV, a more uniform distribution of the non-dominated solution. To verify the effectiveness of the algorithm, we conducted comparative experiments under different dimensions to compare some multi-objective optimization algorithms, including MOPSO, MOPIO, S-MOPIO, and CPSMOEA.

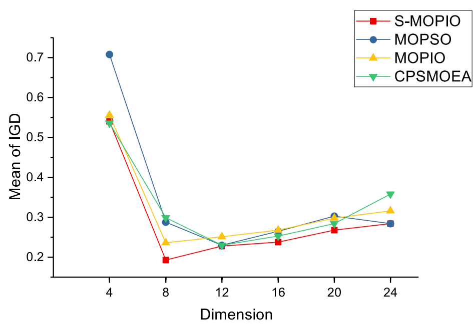

Figure 2: Mean of IGD

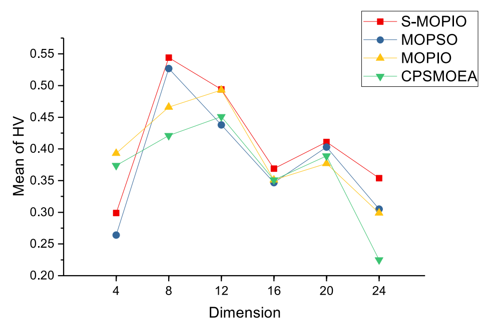

Figure 3: Mean of HV

We calculate the IGD according to the Non-dominant solution of each algorithm, Four Multi-objective optimization algorithms are used to solve the BCOM-CMfg in different dimensions. The solutions to these multi-objective optimization algorithms under different dimensions are displayed in Fig. 2, when dimension=8 or 16 or 20, it is obvious that the IGD of solution calculated by the S-MOPIO is the smallest. In other dimensions, it's also closer than the smallest one. It can be inferred that the solution of S-MOPIO is very close to True Pareto Front than the other three algorithms. The change of the IGD is observed. Under different dimensions, the S-MOPIO algorithm shows a better performance in approaching the Pareto Front of BCOM-CMfg.

The trend of the HV is shown in Fig. 3. When the dimension is four, the Pareto Front which calculated by the MOPIO is the smallest. However, when the dimension is greater than or equal to 8, the Pareto Front which calculated by the S-MOPIO is the smallest. It can be inferred that the S-MOPIO algorithm is good at solving models with high dimensions.

To compare the difference between BCOM-CMfg and Crisis Management Supply Chain (CMSC) [Xu, Tang, Wang et al. (2018)], we calculate the index data of manufacturing services in the two models. The average and median values of service time, service cost and service reliability of 4000 candidate services in BCOM-CMfg and CMSC are shown in Figs. 4 and 5. The advantage of BCOM-CMfg in the cost and reliability of service is obvious.

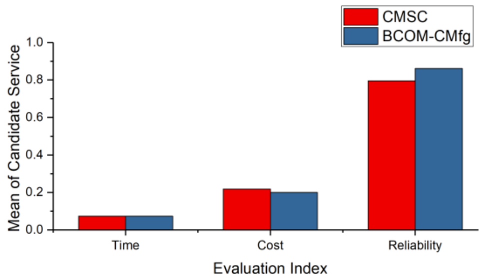

Figure 4: Mean of metrics for Candidate Service

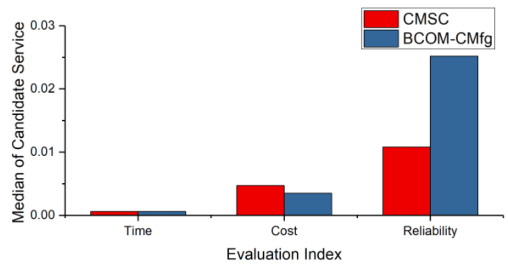

Figure 5: Median of metrics for Candidate Service

Besides, S-MOPIO is used to solve the CMSC and BCOM-CMfg in different dimensions for 30 times. The mean values of reliability and cost are displayed in Figs. 6 and 7.

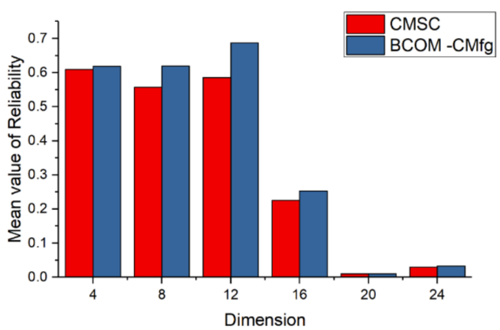

Figure 6: Comparison of reliability

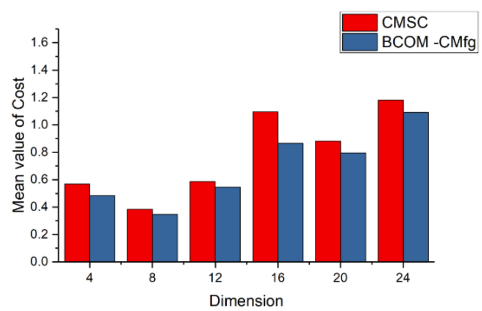

Figure 7: Cost of reliability

In different dimensions, the mean of service composition cost in BCOM-CMfg is lower than the CMSC model. The mean of service composition reliability in BCOM-CMfg is higher than the CMSC model. It shows that BCOM-CMfg can reduce Cloud Manufacturing cost and improve the reliability of Cloud Manufacturing.

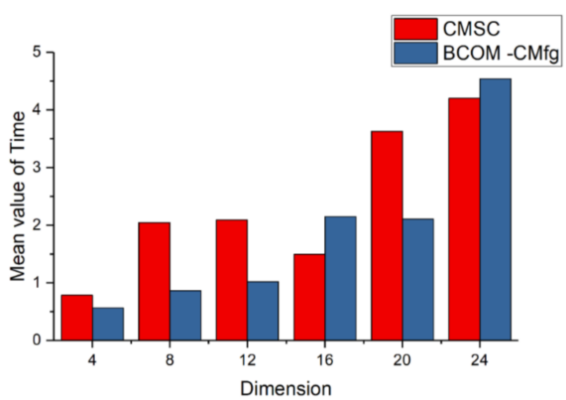

Figure 8: Comparison of time

In Fig. 8, the performance of the two models have their advantages and disadvantages in the time index. The reason is that the algorithm also needs to consider other indexes while selecting candidate services, and there is no significant difference between the two models in calculating the time of a single candidate service, so it is easy to be affected by other indexes.

5 Conclusion

Considering the service composition and service scheduling of both sides on the Cloud Manufacturing platform, a multi-objective efficient scheduling strategy and CPL evaluation mechanism are proposed to construct the BCOM-CMfg model. The S-MOPIO algorithm is proposed to solve the cooperative optimization problem. The experimental results show that BCOM-CMfg is beneficial to the reliability of Cloud Manufacturing and the realization of Cloud Manufacturing solution. However, we lack the research on the structure of manufacturing tasks, so we can fully consider the correlation between different tasks in the future, and find effective model solving algorithms according to different business backgrounds while building a more accurate service portfolio optimization model.

Funding Statement: This paper was supported in part by Natural Science Foundation of Jiangsu Province of China under Grant BK20191381, in part by Jiangsu Planned Projects for Postdoctoral Research Funds under Grant 2019K223, in part by the National Natural Science Foundation of China under Grant 61802208, Grant 61772286, Grant 61771258, and Grant 61701252, in part by Project funded by China Postdoctoral Science Foundation Grant 2019M651923, in part by Primary Research & Development Plan of Jiangsu Province under Grant BE2019742, and in part by NUPTSF under Grant NY220060, NY218035.

Conflicts of Interest: No conflicts of interest exit in the submission of this manuscript, and manuscript is approved by all authors for publication.

References

Al-Masri, E.; Mahmoud, Q. H. (2007): Discovering the best web service. *Proceedings of the 16th International Conference on World Wide Web*, pp. 1257-1258.

Chen, Y.; Huang, J. W.; Lin, C.; Hu, J. (2014): A partial selection methodology for efficient QOS-aware service composition. *IEEE Transactions on Services Computing*, vol. 8, no. 3, pp. 384-397.

Dou, R.; Duan, H. B. (2016): Pigeon inspired optimization approach to model prediction control for unmanned air vehicles. *Aircraft Engineering and Aerospace Technology: An International Journal*, vol. 88, no. 1, pp. 108-116.

Fang, Y. L.; Peng, C.; Lou, P.; Zhou, Z. D.; Hu, J. M. et al. (2019): Digital-twin based job shop scheduling towards smart manufacturing. *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 384-397.

Huang, D.; Bai, Y.; Liu, J. C.; Liu, Q. L.; Chen, H. T. et al. (2018): Hierarchical optimization of network resource for heterogeneous service in cloud scenarios. *Intelligent Automation and Soft Computing*, vol. 24, no. 4, pp. 883-889.

Kim, S. H.; Jin, H.; Sun, D. K. (2011): Optimizing the persistent scheduling in two-hop relay networks. *IEEE Wireless Communications and Net- Working Conference*, pp. 138-143.

Lee, G.; Katz, R. H. (2011): Heterogeneity-aware resource allocation and scheduling in the cloud. *HotCloud*, pp. 1-5.

Leonardo, C. T. B.; Manuel, L.; Thomas, S. (2017): An empirical assessment of the properties of inverted generational distance on multi-and many-objective optimization. *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 31-45.

Liu, Z. T.; Xia, J. Y. (2019): A cross-tenant RBAC model for collaborative cloud services. *Computers, Materials & Continua*, vol. 60, no.1, pp. 395-408.

Peng, C.; Meng, Y. J. (2016): Empirical study of manufacturing enterprise collaboration network: formation and characteristics. *Robotics and Computer-Integrated Manufacturing*, vol. 42, pp. 49-62.

Saremi, S.; Mirjalili, S. (2020): Evaluating PSO and MOPSO equipped with evolutionary population dynamics. *Optimization Algorithms for Hand Posture Estimation*, pp. 61-88.

Sun, L.; Lin, L.; Gen, M.; Li, H. J. (2019): A hybrid cooperative coevolution algorithm for fuzzy flexible job shop scheduling. *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 5, pp. 1008-1022.

Wang, H.; Sun, G. Y.; Zhang, Q. Y.; Liu, K. M.; Xi, M. et al. (2019): The design and implementation of a service composition system based on a restful API. *Intelligent Automation and Soft Computing*, vol. 25, no. 3, pp. 573-583.

Wu, X. L.; Sun, Y. J. (2018): A green scheduling algorithm for flexible job shop with energy-saving measures. *Journal of Cleaner Production*, vol. 172, pp. 3249-3264.

Xu, B.; Tang, Y.; Wang, Z. S.; Shi, L.; Qi, J. et al. (2018): Cloud manufacturing service composition with service cooperation level evaluation. *International Journal of Wireless and Mobile Computing*, vol. 15, no. 4, pp. 342-350.

Xue, X.; Wang, S. F.; Lu, B. Y. (2016): Manufacturing service composition method based on networked collaboration mode. *Journal of Network and Computer Applications*, vol. 59, pp. 28-38.

Yang, K.; Emmerich, M.; Deutz, A.; Bäck, T. (2019): Multi-objective Bayesian global optimization using expected hypervolume improvement gradient. *Swarm and Evolutionary Computation*, vol. 44, pp. 945-956.

Zhang, J. Y.; Zhou, A. M.; Zhang, G. X. (2015): A classification and pareto domination based multi-objective evolutionary algorithm. *IEEE Congress on Evolutionary Computation*, pp. 2883-2890.

Zhao, T.; Peng, B. (2019): Research on complexity of China's manufacturing networks. *Intelligent Automation and Soft Computing*, vol. 25, no. 4, pp. 725-733.