

4-15-2024

Intelligent Optimization Method of Cloud Manufacturing Swarm Based on Incomplete Information Game

Kunpeng Zhang

School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China, 2041203565@qq.com

Yan Wang

School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China, wangyan88@jiangnan.edu.cn

Zhicheng Ji

School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation. For more information, please contact xtfzxb@126.com.

Intelligent Optimization Method of Cloud Manufacturing Swarm Based on Incomplete Information Game

Abstract

Abstract: In the process of cloud manufacturing, the incomplete information status and the mutual competition and restriction relationship between cloud platform operator and demander lead to the difficult choice of manufacturing services. A cloud manufacturing swarm intelligent optimization method based on incomplete information game model is proposed. A static game model based on incomplete information is established for the interest competition between demand-side and cloud platform, with the goal of rationally pursuing the maximization of their own revenue function. The competition rules between demand-side and cloud platform are proposed, which are introduced into nature through Harsanyi transformation and converted into a dynamic game under complete information to obtain Bayesian extended formula, and the existence and uniqueness of Bayesian Nash equilibrium are proved. A particle swarm optimization algorithm based on the update of Gaussian function and perturbation strategy is proposed to solve the above model. The simulation shows that the improved algorithm has faster convergence rate and higher total revenue of cloud manufacturing system compared with the other algorithms, and the incomplete information game model can take into account different types of demand side to improve the total revenue of cloud manufacturing system.

Keywords

cloud manufacturing, the game, incomplete information, Bayesian Nash equilibrium, income function, PSO

Recommended Citation

Zhang Kunpeng, Wang Yan, Ji Zhicheng. Intelligent Optimization Method of Cloud Manufacturing Swarm Based on Incomplete Information Game[J]. Journal of System Simulation, 2024, 36(4): 915-928.

基于不完全信息博弈的云制造群智能优化方法

张坤鹏, 王艳*, 纪志成

(江南大学 物联网工程学院, 江苏 无锡 214122)

摘要: 为解决云制造过程中云平台经营方与需求方之间的不完全信息以及相互竞争制约的关系导致制造服务难以抉择的问题, 提出了一种基于不完全信息博弈模型的云制造群智能优化方法。以各自理性追求自身收益函数最大化为目标, 针对需求方与云平台之间的利益竞争关系建立了基于不完全信息的静态博弈模型, 并提出了需求方与云平台之间的竞争规则, 通过海萨尼转换引入自然, 将其转换为完全信息下的动态博弈得到贝叶斯扩展式, 并证明了贝叶斯纳什均衡的存在性和唯一性。提出了一种基于高斯函数与扰动策略更新的粒子群算法对上述模型进行求解。仿真结果表明: 改进算法相对其他算法有较快的收敛速度与较高的云制造系统总收益, 不完全信息博弈模型能够兼顾不同类型的需求方提高云制造系统的总收益。

关键词: 云制造; 博弈; 不完全信息; 贝叶斯纳什均衡; 收益函数; 粒子群算法

中图分类号: TP391.9 文献标志码: A 文章编号: 1004-731X(2024)04-0915-14

DOI: 10.16182/j.issn1004731x.joss.22-1500

引用格式: 张坤鹏, 王艳, 纪志成. 基于不完全信息博弈的云制造群智能优化方法[J]. 系统仿真学报, 2024, 36(4): 915-928.

Reference format: Zhang Kunpeng, Wang Yan, Ji Zhicheng. Intelligent Optimization Method of Cloud Manufacturing Swarm Based on Incomplete Information Game[J]. Journal of System Simulation, 2024, 36(4): 915-928.

Intelligent Optimization Method of Cloud Manufacturing Swarm Based on Incomplete Information Game

Zhang Kunpeng, Wang Yan*, Ji Zhicheng

(School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China)

Abstract: In the process of cloud manufacturing, the incomplete information status and the mutual competition and restriction relationship between cloud platform operator and demander lead to the difficult choice of manufacturing services. A cloud manufacturing swarm intelligent optimization method based on incomplete information game model is proposed. A static game model based on incomplete information is established for the interest competition between demand-side and cloud platform, with the goal of rationally pursuing the maximization of their own revenue function. The competition rules between demand-side and cloud platform are proposed, which are introduced into nature through Harsanyi transformation and converted into a dynamic game under complete information to obtain Bayesian extended formula, and the existence and uniqueness of Bayesian Nash equilibrium are proved. A particle swarm optimization algorithm based on the update of Gaussian function and perturbation strategy is proposed to solve the above model. The simulation shows that the improved algorithm has faster convergence rate and higher total revenue of cloud manufacturing system compared with the other

收稿日期: 2022-12-14 修回日期: 2023-03-02

基金项目: 国家自然科学基金(61973138)

第一作者: 张坤鹏(1998-), 男, 硕士生, 研究方向为云制造资源配置与决策。E-mail: 2041203565@qq.com

通讯作者: 王艳(1978-), 女, 教授, 博导, 博士, 研究方向为网络化系统、智能产线能耗优化控制。E-mail: wangyan88@jiangnan.edu.cn

algorithms, and the incomplete information game model can take into account different types of demand side to improve the total revenue of cloud manufacturing system.

Keywords: cloud manufacturing; the game; incomplete information; Bayesian Nash equilibrium; income function; PSO

0 引言

云制造是一种面向服务、高效低耗和基于知识的网络化智能制造新模式^[1], 融合了信息化制造、云计算、物联网、语义 Web、高性能计算等先进信息技术^[2-3], 通过对现有网络化制造与服务技术进行延伸和变革, 将各类制造资源和制造能力虚拟化、服务化^[4], 并进行统一、集中的智能化管理和经营, 实现智能化、普适化和高效的共享和协同, 通过网络为制造全生命周期过程提供可随时获取、优质廉价、安全可靠的服务^[5-6]。

针对云制造资源组合优化问题, 许多学者进行了相关研究, 文献[7]在云制造调度优化问题上针对需求方建立了时间、成本、可靠性的多目标优化模型, 采用遗传算法对其进行求解。文献[8]建立了 QoS 感知数学模型, 扩展了花授粉算法, 更有效获得了制造服务组合。文献[9]为了决策最佳服务组合, 针对服务提供方与需求方建立了深度神经网络, 选择云制造服务的最佳方案。文献[10]将 PSO 算法与遗传算法结合, 利用父种群生成子种群, 替换较差个体, 利用遗传算法稳态进化的性质, 保证了改进 PSO 算法的多样性。

上述研究在参与方选择方面, 仅考虑单方独立优化, 而现实中参与方不止需求方, 考虑多个参与方之间的协调与竞争关系, 能够进一步提高云制造系统的可靠性。考虑到需求方与云平台间相互协调、相互制约, 博弈论常被用来解决此类问题。文献[11]提出了面向云制造的博弈模型, 支持云制造合作之间的资源有效分配与决策。文献[12]针对云制造服务问题, 建立了基于需求方合作理念的博弈模型, 优化了成本指标。文献[13]根据云制造复杂特征的资源调度问题, 提出了一种资源竞价机制, 保证了云制造市场的公平性。

上述研究的博弈大多为完全信息, 实际情况不完全信息更为常见且合理。为此, 本文讨论需求方与云平台方的不完全信息, 基于贝叶斯博弈方法对云制造组合优化问题进行深入研究。针对不完全信息场景问题, 贝叶斯博弈^[14]是最有效的解决办法。

由于在实际云制造过程中, 需求方与云平台存在竞争利益问题, 且两者的目标都是追求各自利益最大化, 同时需求方存在不同的类型。针对上述问题, 本文提出不完全信息下的云制造场景, 建立了贝叶斯博弈模型, 利用海萨尼转换将不完全信息静态博弈转换成完全信息动态博弈, 合理抉择竞争策略, 提高云制造系统总收益。为提高粒子多样性, 提出引入高斯项分布特性改进算法, 配合速度扰动更新策略提高收敛速度和解的质量, 利用该算法对上述模型进行求解。

1 问题描述与模型建立

1.1 问题描述

云制造服务环境主要由需求方(demand side, DS)、制造资源提供方(manufacturing resource suppliers, MRS)和云制平台运营方(cloud platform operators, CPO)组成^[15]。制造资源提供方提供的各种制造资源通过虚拟化、服务化后, 形成虚拟的制造资源池。云制造环境下的制造资源优化配置指根据制造服务需求方提交的不同制造任务, 云制造服务平台为其配置合理的制造资源服务, 并交由制造资源提供方完成制造任务的过程, 如图1所示。此过程中需求方与云平台经营方存在利益竞争问题, 各自选择策略, 追求自己的利益最大化且存在个人信息, 需讨论不完全信息下的模型建立。

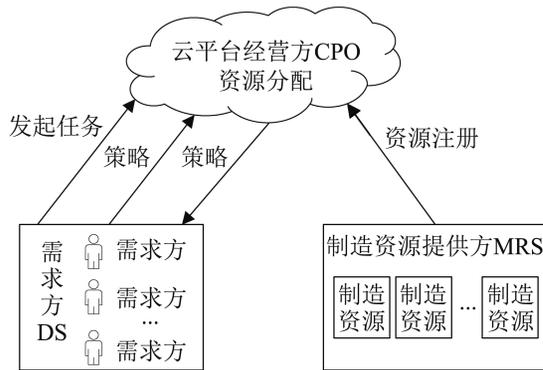


图1 云制造过程

Fig. 1 Cloud manufacturing process

云平台根据一定规则将需求方任务分解为单一资源所能完成的制造子任务。假设任务被分解为 m 个子任务 $O = \{O_1, O_2, \dots, O_m\}$, 云制造平台对 m 个子任务寻找相应的候选资源集合 $R = \{R_1, R_2, \dots, R_n\}$ 。

云制造生产模式如图2所示, 制造资源分散在各地, 云平台通过注册来虚拟化资源将各地的制造资源存储在资源池中, 等接收到需求方的任务时, 开始对资源池中的资源进行配置, 完成用户的任务, 提高资源利用率。

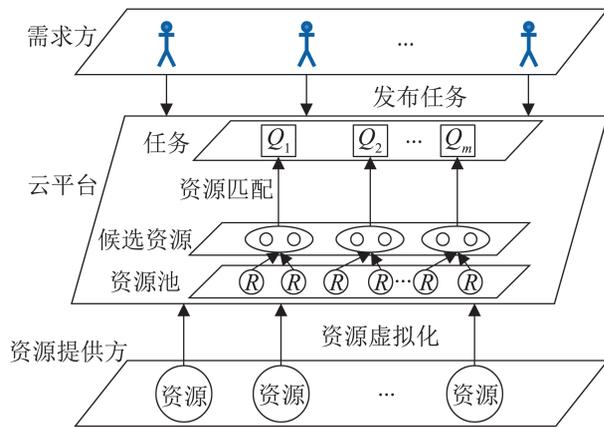


图2 云制造生产模式

Fig. 2 Cloud manufacturing production mode

1.2 模型假设

为了便于模型的建立与求解, 做出以下假设:

(1) 假设需求方与云平台都是完全理性的, 在做出策略时均以自己的收益函数最大化优先。

(2) 假设需求方存在多种类型, 主要有时间型需求方和成本型需求方2种类型, 对任务的需求不同, 决定了各自的收益不同。

(3) 假设需求方与云平台只存在竞争关系, 为了提高各自的收益函数都会选取对自己有利的策略。

1.3 建立模型

为了提供优质的云服务给需求方, 同时还要兼顾云平台经营方自身的利益, 需要解决需求方与云平台经营方之间的竞争利益问题, 为此建立双层规划模型。不同类型的需求方, 任务需求不同, 兼顾不同类型下需求方的需求去设计目标函数。云制造环境下资源配置优化服务组合, 需要确定优化指标, 为了筛选出最优资源服务, 针对需求方采用交付时间、交付成本、服务质量3个指标作为评价标准。为了提高云平台自身利益, 针对云平台采用云平台成本与云平台服务评分这2个指标, 构建各自的数学模型。

(1) 交付时间

交付时间为从云制造平台上接受需求方订单到交付给需求方所需要的时间, 主要为制造时间, 为了提高生产效率, 需要尽可能去减少交付时间。需求方等待时间

$$t_{\text{all}} = \sum_{i=1}^m \sum_{j=1}^{n_i} x_{ij} t(R_{ij}) \quad (1)$$

式中: m 为子任务数量; n_i 为第 i 个子任务对应的候选资源数量; x_{ij} 为资源选择变量; R_{ij} 为第 i 的任务对应的第 j 个资源。

(2) 交付成本

交付成本为从云制造平台上接受需求方订单到完工交付到需求方所需要的成本, 主要包括生产成本、运输成本和云平台成本等, 在资源配置优化过程中, 需降低成本。

需求方消耗成本:

$$C_{\text{ma}} = \sum_{i=1}^m \sum_{j=1}^{n_i} x_{ij} C(R_{ij}) \quad (2)$$

需求方消耗成本:

$$C_{\text{all}} = C_{\text{ma}} + C_{\text{CPO}} \quad (3)$$

式中: C_{CPO} 为云平台成本。

(3) 服务质量

云制造资源复杂多变, 为了使需求方能够得到高质量的服务, 需求方在做出策略时, 需要考虑制造服务质量, 云服务的服务质量为整个制造服务过程中任务合格率的重要指标。

需求方消耗成本:

$$Q_{\text{all}} = \sum_{i=1}^m \sum_{j=1}^{n_i} x_{ij} Q(R_{ij}) \quad (4)$$

(4) 云平台成本

云平台经营方在参与过程中调动资源需要消耗成本, 最后和提供商的价格合并反馈给需求方, 云平台要尽可能优化所需要的成本, 这里定义与需求方提交的任务复杂情况有关。

$$C_{\text{CPO}} = \alpha C_{\text{ma}} + C_{\text{base}} \quad (5)$$

式中: α 为成本影响因子; C_{ma} 为制造成本和运输成本; C_{base} 为基础成本。

(5) 云平台服务评分

为了综合反映制造过程中云平台服务质量, 本文选择云平台服务评分作为评价指标, 定义服务评分为需求方对此次服务后的总评分。

$$S_{\text{CPO}} = S_{\text{base}} + \Delta S \quad (6)$$

式中: ΔS 为博弈后的评分差; S_{base} 为云平台基础评分。

根据上述指标, 定义目标函数:

$$f_1 = \max \left(w_1 \frac{C_{\text{max}} - C_{\text{all}}}{C_{\text{max}}} + w_2 \frac{t_{\text{max}} - t_{\text{all}}}{t_{\text{max}}} + w_3 (Q_{\text{all}} - Q_{\text{min}}) \right) \quad (7)$$

$$\sum_{j=1}^{n_i} x_{ij} = 1, 1 \leq i \leq n_i, 1 \leq j \leq m \quad (8)$$

$$\alpha \in [0, 1] \quad (9)$$

$$C_{\text{all}} \in [C_{\text{min}}, C_{\text{max}}] \quad (10)$$

$$t_{\text{all}} \in [t_{\text{min}}, t_{\text{max}}] \quad (11)$$

$$Q_{\text{all}} \in [Q_{\text{min}}, 1] \quad (12)$$

$$S_{\text{CPO}} \in [S_{\text{base}}, 1] \quad (13)$$

云平台的目标函数:

$$f_2 = \max \left(v_1 \frac{C_{\text{max}} - C_{\text{CPO}}}{C_{\text{max}}} + v_2 S_{\text{CPO}} \right) \quad (14)$$

$$f_{\text{总}} = \max(f_1 + f_2) \quad (15)$$

式中: w_1 、 w_2 、 w_3 为需求方权重系数; v_1 、 v_2 为云平台权重系数。

在需求方与云平台经营方的目标函数优化中, 受到各自策略的影响, 同时还要保证整个方案的总效益最优, 但从个体角度来说, 也要追求自身利益最大化, 这使得需求方与云平台相互制约、相互竞争, 故两者之间的云制造组合优化问题可归为一类复杂系统的多主体优化决策问题。能够考虑多决策主体相互作用、协调平衡的博弈论是解决上述组合优化难题的有力工具, 但由于完全信息下的博弈方法太过于理想化, 在需求方与云平台经营方追求各自收益最大化的过程中, 实际上博弈双方的很多信息属于个人私密信息, 双方无法清楚知道彼此的信息, 因此, 本文针对云制造过程, 提出一种不完全信息的贝叶斯博弈方法, 对云制造组合优化问题进行研究。

2 不完全信息下的云制造贝叶斯博弈方法

博弈论可根据收益函数是否为共同知识来区分, 分为完全信息博弈和不完全信息博弈^[16]。如果博弈的参与者保留仅有自己了解的私有信息, 那么该博弈为不完全信息博弈, 反之则为完全信息博弈。云平台与需求方之间存在的信息大多是不完全信息, 即各自的收益函数双方都不清楚, 不完全信息博弈还需考虑博弈方的类型, 所以, 能兼顾各种情况的发生, 因此, 考虑双方之间的信息不完全情况, 使云制造资源优化结果更符合实际情况。

常规的博弈模型组成部分分别是参与方、策略空间、收益函数, 而对于不完全信息的贝叶斯博弈还需要类型空间与先验概率分布, 根据不同类型的博弈方建立不同的收益函数。

2.1 不完全信息贝叶斯博弈模型

建立需求方与云平台的贝叶斯博弈模型, 用 $G = \{N_1, N_2, \dots, N_n; S_1, S_2, \dots, S_n; \theta_1, \theta_2, \dots, \theta_n; p_1, p_2, \dots, p_n; U_1, U_2, \dots, U_n\}$ 来表示一个贝叶斯博弈^[16]。

(1) 博弈方集合: $N = \{N_1 = \text{DS}, N_2 = \text{CPO}\}$ 。

(2) 类型空间: 在该博弈中, 用类型定义博弈方的个人信息, $\theta = \{\theta_0 = 0, \theta_1 = 1\}$ 为需求方的类型空间, θ_0 代表需求方是成本类型, θ_1 代表需求方是时间类型。

(3) 信念: 信念表示类型空间上的先验概率分布 $p(\theta_i, \theta_{-i})$, 其他博弈方概率的推断是 $p(\theta_{-i}|\theta_i)$ 。

该概率分布是公有信息, 它遵循贝叶斯公式:

$$p(\theta_{-i}|\theta_i) = \frac{p(\theta_{-i}, \theta_i)}{p(\theta_i)} \quad (16)$$

式中: θ_i 为博弈方 i 的类型; θ_{-i} 为除博弈方 i 之外的类型组合; $\theta_{-i} \in \Theta_{-i}$, Θ_{-i} 为 θ_{-i} 组成的集合。

(4) 策略空间: 需求方 i 对于每一个可能的类型 $\theta_i \in \Theta_i$ 都选择一个策略 $s_i \in S_i$, 那么策略组合为 $\Omega_i = (S_i, S_{-i}(\theta_{-i}))$ (17)

式中: S_i 为博弈方 i 各类型下的策略空间集合; $S_{-i}(\theta_{-i})$ 为除博弈方 i 之外的全部博弈方所有类型的策略集。该不完全信息博弈中博弈方的策略集为 $S_1 = \{s_{11} = \text{Compete}, s_{12} = \text{No compete}\}$, $S_2 = \{s_{21} = \text{No accept}, s_{22} = \text{Accept}\}$ 。

根据本文假设, 云平台与需求方之间只存在竞争关系, 提高收益函数而选取各自最有利的策略。为了更好地模拟需求方与云平台之间利益竞争情况, 设计一种竞争规则:

$$\begin{cases} C_{\text{all}} = 0.9C_{\text{all}}, C_{\text{CPO}} = 1.1C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.2, \\ S = (s_{11}, s_{21}) \\ C_{\text{all}} = 0.95C_{\text{all}}, C_{\text{CPO}} = 1.05C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.1, \\ S = (s_{11}, s_{22}) \\ C_{\text{all}} = C_{\text{all}}, C_{\text{CPO}} = C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}}, \\ S = (s_{12}, s_{21}) \\ C_{\text{all}} = C_{\text{all}}, C_{\text{CPO}} = C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.1, \\ S = (s_{12}, s_{22}) \end{cases} \quad (18)$$

若 $S = (s_{11}, s_{21})$, 则需求方成本降低, 云平台成

本增加, 服务评分增加 0.2; 若 $S = (s_{11}, s_{22})$, 则需求方成本降低, 云平台成本增加, 服务评分增加 0.1; 若 $S = (s_{12}, s_{21})$, 则需求方成本不变, 云平台成本不变, 服务评分不变; 若 $S = (s_{12}, s_{22})$, 则需求方成本不变, 云平台成本不变, 服务评分增加 0.1。

同理, 对于时间类型的需求方, 规则为

$$\begin{cases} t_{\text{all}} = 0.8t_{\text{all}}, C_{\text{CPO}} = 1.1C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.2, \\ S = (s_{11}, s_{21}) \\ t_{\text{all}} = 0.9t_{\text{all}}, C_{\text{CPO}} = 1.05C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.1, \\ S = (s_{11}, s_{22}) \\ t_{\text{all}} = t_{\text{all}}, C_{\text{CPO}} = C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}}, \\ S = (s_{12}, s_{21}) \\ t_{\text{all}} = t_{\text{all}}, C_{\text{CPO}} = C_{\text{CPO}}, S_{\text{CPO}} = S_{\text{CPO}} + 0.1, \\ S = (s_{12}, s_{22}) \end{cases} \quad (19)$$

(5) 收益函数: 需求方与云平台以各自的收益函数最大化进行博弈, 需求方在不同类型下的博弈收益函数与云平台经营方的收益函数分别为

$$\begin{cases} u_1(\theta = 0) = w_1 \frac{\Delta C_{\text{all}}}{C_{\text{max}}} \\ u_1(\theta = 1) = w_2 \frac{\Delta T_{\text{all}}}{T_{\text{max}}} \end{cases} \quad (20)$$

$$u_2 = v_1 \frac{\Delta C_{\text{CPO}}}{C_{\text{CPOmax}}} + v_2 \Delta S \quad (21)$$

2.2 贝叶斯纳什均衡分析

由于云平台不明确需求方类型, 属于不完全信息场景, 须要引用海萨尼转换^[17], 将不完全信息博弈转换为完全信息动态博弈进行求解。云平台根据需求方类型的分布概率可以进行分析, 根据本文提出竞争规则, 可以得到收益矩阵:

$$\begin{cases} (a, b) = \left(0.1w_1 \frac{C_{\text{all}}}{C_{\text{max}}}, -0.1v_1 \frac{C_{\text{all}}}{C_{\text{max}}} + 0.2v_2 \right) \\ (c, d) = \left(0.05w_1 \frac{C_{\text{all}}}{C_{\text{max}}}, -0.05v_1 \frac{C_{\text{all}}}{C_{\text{max}}} + 0.1v_2 \right) \\ (e, f) = \left(0.2w_2 \frac{t_{\text{all}}}{t_{\text{max}}}, -0.1v_1 \frac{C_{\text{CPO}}}{C_{\text{CPOmax}}} + 0.2v_2 \right) \\ (g, h) = \left(0.1w_2 \frac{t_{\text{all}}}{t_{\text{max}}}, -0.05v_1 \frac{C_{\text{CPO}}}{C_{\text{CPOmax}}} + 0.1v_2 \right) \end{cases} \quad (22)$$

需求方的类型包括成本型和时

型策略为 $S_1 = \{\text{Compete}, \text{No compete}\}$, 云平台的策略为 $S_2 = \{\text{No accept}, \text{Accept}\}$ 。根据海萨尼转换, 引入自然节点 Nature, 根据表1~2的收益矩阵, 得到贝叶斯博弈扩展式, 云平台由概率 μ 得知需求方类型为成本型, 如图3所示。

表1 成本型收益矩阵
Table 1 Cost-benefit matrix

关系	拒绝	接受
竞争	(a, b)	(c, d)
不竞争	(0, 0)	(0, 0.1v ₂)

表2 时间型收益矩阵
Table 2 Time-type return matrix

关系	拒绝	接受
竞争	(e, f)	(g, h)
不竞争	(0, 0)	(0, 0.1v ₂)

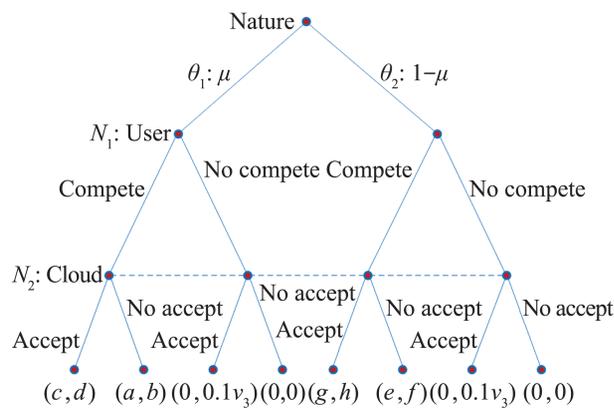


图3 贝叶斯博弈扩展式
Fig. 3 Bayesian game extension

定义1. 博弈占优策略^[17]

分别用 s_{i1} 和 s_{i2} 表示博弈方 i 的2个策略, 如果其他博弈方存在可行策略 s_{-i} , 并且博弈方选择策略 s_{i1} 的收益是大于等于策略 s_{i2} 的收益, $U_{i1}(s_{i1}, s_{-i}) \geq U_{i2}(s_{i2}, s_{-i})$, 那么则称 s_{i1} 是相对于 s_{i2} 的占优策略。

定义2. 贝叶斯纳什均衡^[17]

$G = \{N_1, N_2, \dots, N_n; S_1, S_2, \dots, S_n; \theta_1, \theta_2, \dots, \theta_n; p_1, p_2, \dots, p_n; U_1, U_2, \dots, U_n\}$ 代表 n 个不完全信息静态博弈, 纯策略贝叶斯纳什均衡是一个类型策略组合解 $\{s_i^*(\theta_i), i = 1, 2, \dots, n\}$, 在每个博弈方 i 确定自己

类型 θ_i 与其他博弈方的策略组合 $s_{-i}^*(\theta_{-i})$ 的情况下, 使自身的期望收益函数 U_i 最大化。若对于所有的博弈方 i , $s_i \in S_i(\theta_i)$, $s_i^*(\theta_i) \in \arg \max_{s_i} \sum p(\theta_{-i} | \theta_i) \times U_i(s_i, s_{-i}^*(\theta_{-i}); \theta_i; \theta_{-i})$, 策略组合 $s^* = (s_1^*(\theta_1), s_2^*(\theta_2), \dots, s_n^*(\theta_n))$ 是一个纯策略的贝叶斯纳什均衡解。若博弈方 i 的策略空间是 $\{s_{i1}, s_{i2}, \dots, s_{ik}\}$, 则策略相对于的概率分布 $p_i = \{p_{i1}, p_{i2}, \dots, p_{ik}\}$ 是 i 的一个混合策略, $p_{ik} = p(s_{ik})$ 表示博弈方 i 选择策略 s_{ik} 的概率, 约束:

$$\begin{cases} 0 \leq p_{ij} \leq 1, & j = 1, 2, \dots, k \\ p_{i1} + p_{i2} + \dots + p_{ik} = 1 \end{cases} \quad (23)$$

如果对于所有的博弈方 i 的期望收益函数 $U_i(p_i^*, p_{-i}^*) \geq U_i(p_i, p_{-i}^*)$, 则称 $p^* = \{p_1^*, p_2^*, \dots, p_n^*\}$ 是该博弈混合策略的一个贝叶斯纳什均衡解。

定理1. 纳什均衡存在性定理^[18]

在 $G = \{S_1, S_2, \dots, S_n; \theta_1, \theta_2, \dots, \theta_n; p_1, p_2, \dots, p_n; U_1, U_2, \dots, U_n\}$ 博弈中, 如果 n 是有限的, 并且 n 个博弈方的策略空间 S_i 也是有限的, 那么对于此博弈, 至少存在一个纳什均衡, 该纳什均衡可能是纯策略纳什均衡或混合策略纳什均衡。

证明如下:

当 $\theta = \theta_1$, 即 N_1 的类型为成本型时, $S_1 = \{\text{Compete}, \text{No compete}\}$, $S_2 = \{\text{No accept}, \text{Accept}\}$ 根据存在性定理1可得, 该博弈至少存在一个混合纳什均衡, 有可能是纯策略纳什均衡。根据定义2假设 N_1 与 N_2 混合纳什均衡策略组合为 $P = (P_1, P_2)$, 其中, $P_1 = (p, 1-p)$, $P_2 = (q, 1-q)$ 且 $p, q \in [0, 1]$ 。

(1) 当 N_2 以 q 的概率选择拒绝, 以 $1-q$ 的概率选择接受时: 若 N_1 选择竞争时, 根据贝叶斯概率可得期望收益为 $E_1 = aq + c(1-q) = (a-c)q + c$ 。若 N_1 选择不竞争时, 根据贝叶斯概率可得期望收益为 $E_2 = 0$ 。根据式(22)可得, $c = 2a$, $E_1 - E_2 = (2-q)a > 0$, 所以, N_1 的最优反应策略为竞争, 即 $p = 1$ 。

(2) 当 N_1 以 p 的概率选择竞争, 以 $1-p$ 的概率选择不竞争时: 若 N_2 选择拒绝时, 根据贝叶斯概

率公式可以计算出 N_2 自己的期望收益 $E_3 = pb$ 。若 N_2 选择接受时, 可得 $E_4 = pd + 0.1v_2(1 - p)$ 。根据式 (22) 可得, $E_3 - E_4 = (b + 0.1v_2 - d)p + 0.1v_2 > 0$, 以 N_2 的最优反应策略为拒绝, 即 $q = 1$ 。

综上所述得到博弈最优反应图如图 4 所示, 对参与方 N_2 来说, 最佳策略 $P_2 = (1, 0)$, 对参与方 N_1 的最佳策略 $P_1 = (1, 0)$, 那么该博弈有且仅有一个纳什均衡即 $P = ((1, 0), (1, 0))$, 同时也是纯策略纳什均衡。同理, 当 N_1 类型为时间型时, 可证存在纳什均衡且唯一。

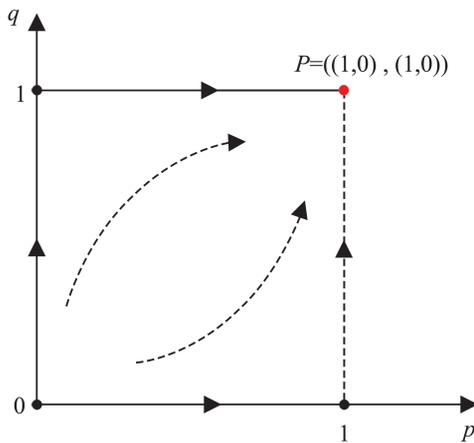


图 4 博弈最优反应图
Fig. 4 Game optimal reaction diagram

2.3 算法求解

针对不完全信息下云制造资源分配的最优策略选择是独立优化各自目标的耦合问题, 处理这一类问题经常使用粒子群算法、遗传算法等, 粒子群算法相对其他智能算法的效率较高, 因此, 采用粒子群算法求解。

2.3.1 传统粒子群算法

传统粒子群算法^[19]适用性强、收敛速度快, 在求解纳什均衡问题上被广泛应用。粒子位置和速度更新的公式为

$$x_j(t+1) = x_j(t) + v_j(t+1), \quad j = 1, 2, \dots, d \quad (24)$$

$$v_j(k+1) = wv_j(k) + c_1r_1[p^* - p_j(k)] + c_2r_2[g^* - p_j(k)] \quad (25)$$

$$w = w_{\max} - \frac{k(w_{\max} - w_{\min})}{N} \quad (26)$$

式中: k 为迭代次数; d 为粒子维数; $v_j(k)$ 为粒子 j 在第 k 次迭代的速度; w 为惯性权值; c_1 和 c_2 为学习因子; p 为粒子 j 个体最优解; p_g 为群粒子最优解; N 为总迭代次数; R_1 与 R_2 为 0~1 之间的随机数, 如果在每次迭代中加入随机性, 则会提高种群的多样性, 提高算法优化效果。

2.3.2 种群编码

如图 5 所示, $O_1 \sim O_6$ 为子任务集合, R 为候选资源集, N 为资源序号, 例如, N_1 就表示对应候选资源的第 1 个, 最底层为相应的服务方案, $R_{1,2}$ 代表第 1 个任务选中了相应候选资源集的第 2 个资源。

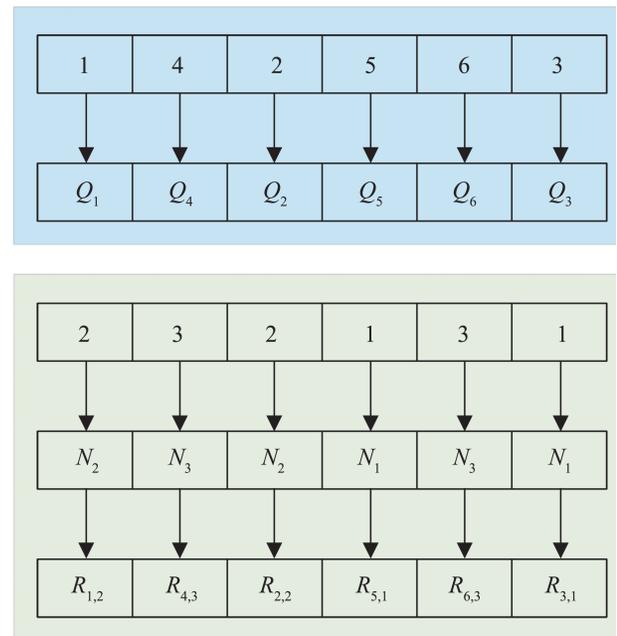


图 5 编码示意图
Fig. 5 Coding diagram

2.3.3 高斯项引入

由于 PSO 易陷入局部最优提前收敛以及种群多样性少, 本文将对粒子群算法进行优化。文献 [20] 提出了线性变化的惯性权重, 但是在粒子搜索过程中 w 是非线性变化较为复杂的, 当 w 变化符合正态分布时, 算法的搜索能力会增强^[21]。本文

采用高斯函数式(27)的分布特性将惯性权值设置为动态变化,以便在迭代前期扩大搜索范围,迭代后期便于局部搜索,并且为了丰富粒子的多样性,本文同时提出新的速度扰动更新策略式(28),由此弥补粒子种群减少的多样性趋势, v 与 w 更新方式如式(29)和(30)所示。

$$g(t) = ae^{-(x-b)^2/c^2} \quad (27)$$

$$d(t) = (t/N)^2 + t/N \quad (28)$$

$$v_j(k+1) = wv_j(k) + c_1r_1[p_j^* - p_j(k)] + c_2r_2[g^* - p_j(k)] + d(t) \quad (29)$$

$$w = \frac{(w_{\max} + 3w_{\min})}{2} + \frac{(w_{\max} - w_{\min})}{2} \times e^{-\left(\frac{t}{kt_{\max}}\right)^2} \quad (30)$$

式中: k 为系数, $k=0.2$ 时 PSO 有较好的优化效果^[22],本文设 $k=0.2$ 。

在本文问题中,需要同时优化需求方与云平台的利益,但会制造需要调和的矛盾,根据本文的贝叶斯博弈方法为每个粒子创建个体博弈粒子群集和整体博弈粒子群集,根据传统粒子群的速度更新式(27),将个体最优解和群体最优解替换为个体博弈最优 p_{bo}^* 与群体博弈最优 g_{bo}^* 得到式(31),通过上述模型的竞争策略,每次迭代寻找博弈最优解。

$$v_j(k+1) = wv_j(k) + c_1r_1[p_{bo}^* - p_j(k)] + c_2r_2[g_{bo}^* - p_j(k)] \quad (31)$$

竞争策略更新:每次粒子迭代过程中,分别计算需求方与云平台方的收益函数,根据定义2选取各自的最优策略追求自身利益最大化选取策略 $s_i^*(\theta_i)$ 。如果需求方与云平台可以通过选择其他策略增加自身的收益,则选择最优策略进行博弈,并将个体与群体最优博弈解更新,算法流程如下。

算法流程:粒子群算法

输入: $G = \{N_1, N_2; S_1, S_2; \theta_1, \theta_2; p_1, p_2; U_1, U_2\}$

可选策略 $\Omega_i = (S_i, S_{-i}(\theta_{-i}))$

输出:最佳方案

初始化种群,初始化策略 s_0 ,最优值 p_{best} 初

始化

while $t < T$ do

$$w = \frac{(w_{\max} + 3w_{\min})}{2} + \frac{(w_{\max} - w_{\min})}{2} \times e^{-\left(\frac{t}{kt_{\max}}\right)^2}$$

根据式(31)进行粒子速度更新

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1)$$

For each N_i in $\{N_1, N_2\}$

For each (s_i, s_{-i}) in $\Omega_i = (S_i, S_{-i}(\theta_{-i}))$

If $u_i(s_i^*, s_{-i}(\theta)) > u_i(s_0, s_{-i}(\theta))$

$s_0 = s_i^*$

end if

end for each

end for each

update p_{bo}^* , update g_{bo}^*

end while

output: 最佳方案

3 实例验证与分析

为了验证本文贝叶斯博弈优化方法求解云制造资源配置问题的性能,对其进行实验测试,实验平台为 Windows 10, Intel(R) Core(TM) i5-10500 CPU 3.1 GHz 处理器 8 G 内存;编程语言为 MATLAB2016a。为了验证本文改进算法的求解结果,对比算法为 NSGA-II^[7]和融合遗传算法的粒子群算法^[10]。根据文献[23]的主观赋权法,即 AHP 法可以得到各指标对应的权重分别为 $w_1=0.3$ 、 $w_2=0.33$ 、 $w_3=0.37$, $v_1=0.57$ 、 $v_2=0.43$,共有 5 个需求方发起任务,先验概率 $\mu=0.6$,根据文献[24]的参数配置,本文设置粒子群参数为 $w_{\max}=0.9$ 、 $w_{\min}=0.4$,种群规模大小为 30,个体学习因子 $c_1=1.7$ 、 $c_2=1.5$,最大迭代次数为 100。本文方法机理图如图 6 所示,需求方发起任务,云平台通过注册的资源进行资源分配,根据不完全信息贝叶斯博弈模型,抉择出贝叶斯纳什均衡,在通过改进 PSO 算法进行求解,提高双方收益,并使云制造系统总收益最大化,合理进行资源分配。

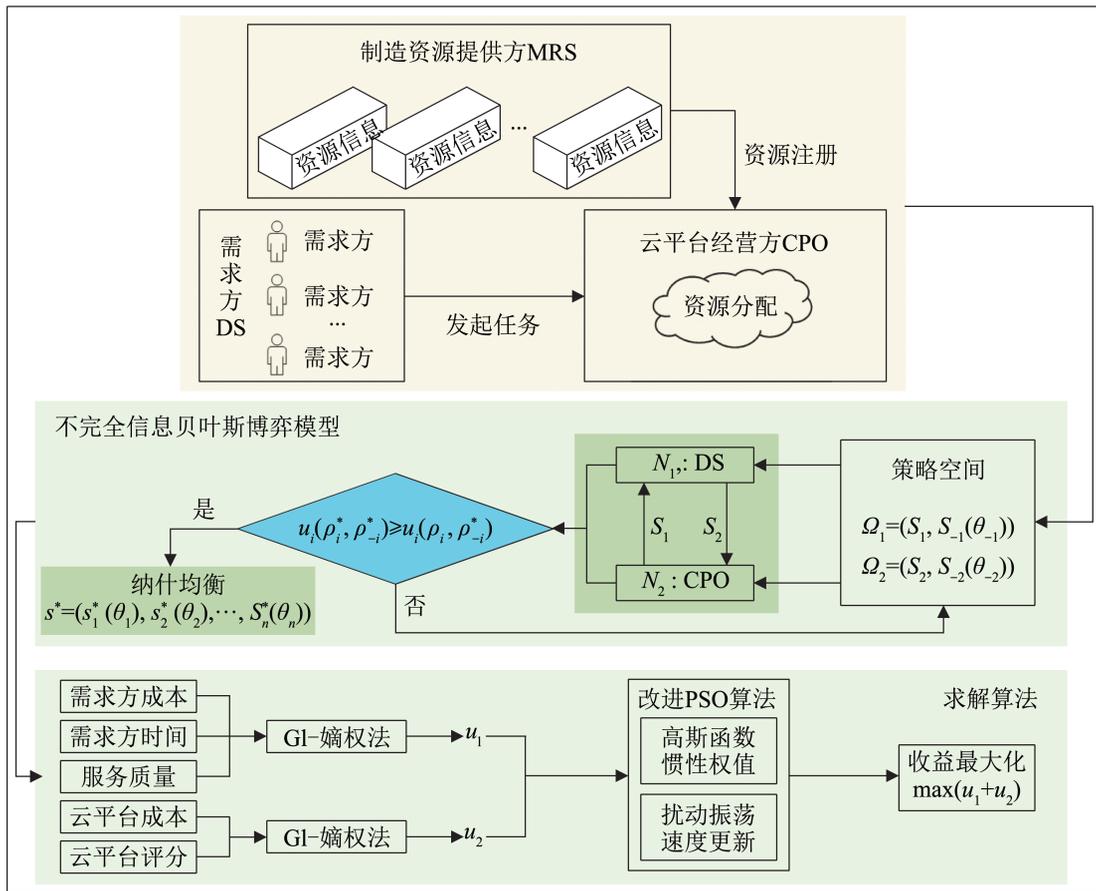


图 6 不完全信息云制造群智能优化流程

Fig. 6 Incomplete information cloud manufacturing swarm intelligent optimization process

3.1 实例验证

为了验证本文模型和算法在解决云制造资源配置问题上的可行性, 以升装配机器人生产的主要加工工程制造为背景, 需求方提交了升装配机器人的制造任务需求给云平台, 云制造平台将任务分解为车体生产、零件生产、驱动装置生产、举升装置生产、电控系统生产、涂装生产生产 6 个子任务 $O_1 \sim O_6$, 如图 7 所示。

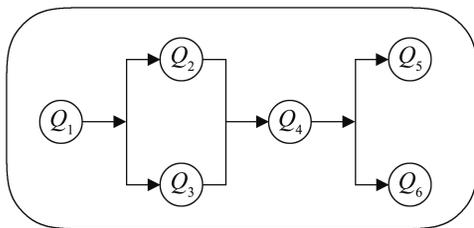


图 7 任务结构图

Fig. 7 Task structure diagram

云平台进行搜索闲置机器人制造资源, 优化云制造服务方案, 给每个子任务找到相应的候选资源集, 相关信息如表 3 所示。云平台基本成本 $C_{base} = 100$, 基础评分为 $S_{base} = 0.7$, 成本影响因子 $\alpha = 0.1$, 个体需求方接受的成本范围为 $[600, 800]$, 时间可接受范围为 $[100, 180]$, 服务质量可接受范围为 $[4, 5]$, 云平台的成本可接受范围为 $[100, 200]$ 。

图 8 为云制造总收益函数收敛曲线图, 最终收益函数均趋于稳定, 达到平衡状态。本文方法在求解过程中的收敛速度与稳定状态最优, 这是由于需求方不止单一类型, 其他文献方法并没有考虑其他类型, 只能针对特定情况, 整体效果不如本文方法。

表3 任务信息加工
Table 3 Task information processing

候选资源	O_1	O_2	O_3	O_4	O_5	O_6
R_1	20/125/0.75	—	30/100/0.70	—	25/115/0.85	—
R_2	—	35/100/0.70	25/100/0.60	20/110/0.50	—	30/130/0.80
R_3	30/110/0.60	—	30/120/0.85	—	30/110/0.80	40/120/0.85
R_4	—	20/130/0.70	—	60/120/0.90	—	—
R_5	20/120/0.70	—	—	50/100/0.70	—	40/110/0.90
R_6	—	30/130/0.80	40/120/0.85	—	40/110/0.90	—
R_7	—	25/110/0.60	—	—	—	35/100/0.70
R_8	—	—	25/100/0.60	—	50/120/0.95	—
R_9	—	30/110/0.60	—	40/130/0.80	—	—
R_{10}	35/100/0.80	—	20/130/0.80	—	40/120/0.95	35/120/0.90
R_{11}	30/100/0.60	35/120/0.80	—	45/130/0.85	—	—
R_{12}	—	—	35/120/0.80	—	30/100/0.75	40/130/0.95
R_{13}	40/90/0.70	—	—	35/110/0.65	—	—
R_{14}	20/120/0.70	—	30/110/0.60	—	20/120/0.60	—
R_{15}	—	—	—	35/120/0.70	—	30/120/0.85
R_{16}	—	25/110/0.65	—	50/110/0.80	—	25/100/0.60

注：数值分别代表加工时间(h)/加工成本(元)/服务质量。

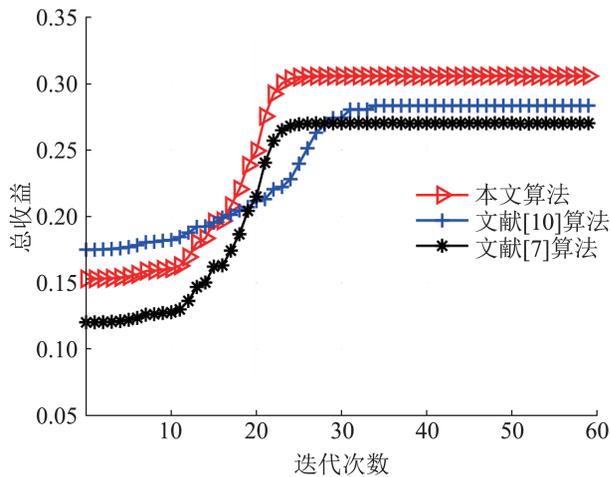


图8 不同算法收敛对比图

Fig. 8 Comparison diagram of different algorithm convergence

表4~5是需要方和云平台不同算法得到的结果。本文方法得到的总收益为0.305，CPU运行时间相对于文献[7, 10]均得到了提高，云平台的成本得到了优化，需求方的成本、时间与服务质量也均得到了优化， Q 代表服务质量， C 代表成本， t 代表时间， S 代表云平台评分。

表4 需求方求解对比分析

Table 4 Demand-side algorithm to solve comparative analysis

算法	收益	C /元	t /min	Q	CPU/s
文献[7]	0.156	4 005	700	20.5	42.8
文献[10]	0.162	3 980	725	20.1	31.4
本文	0.177	3 935	675	20.5	30.6

表5 云平台求解对比分析

Table 5 Cloud platform algorithm to solve comparative analysis table

算法	收益	C /元	S	CPU/s
文献[7]	0.116	852.5	0.8	42.8
文献[10]	0.115	877.5	0.8	31.4
本文	0.128	812.5	0.8	30.6

表6为不同算法求解的资源配置方案，表7为不同算法求解20次算例所得最优解平均值以及算法独立运行20次平均计算时间对比。结果表明：本文算法目标 Q 与文献[10]相近，其余4个指标目标值均优于文献[7, 10]。综上所述，本文算法得到的总收益最高，拥有良好的求解效率，保证了云制造系统的稳定运行，因此，能够提供有效的资源配置方案，提高资源利用率。

表6 最优解方案
Table 6 Optimal solution

算法	最优解方案
文献[7]	$R_5 \rightarrow R_4 \rightarrow R_{10} \rightarrow R_2 \rightarrow R_1 \rightarrow R_{16}$
文献[10]	$R_5 \rightarrow R_3 \rightarrow R_6 \rightarrow R_2 \rightarrow R_8 \rightarrow R_{16}$
本文	$R_5 \rightarrow R_4 \rightarrow R_{10} \rightarrow R_2 \rightarrow R_1 \rightarrow R_{16}$

表7 算法运行20次最优解与CPU平均值对比
Table 7 Algorithm runs 20 times to compare optimal solution with CPU average value

算法	总收益	C/元	t/min	Q	CPU/s
文献[7]	0.276	4 015	720	20.4	44.5
文献[10]	0.279	3 995	715	20.5	32.1
本文	0.314	3 945	680	20.4	31.5

3.2 完全信息与不完全信息对比结果分析

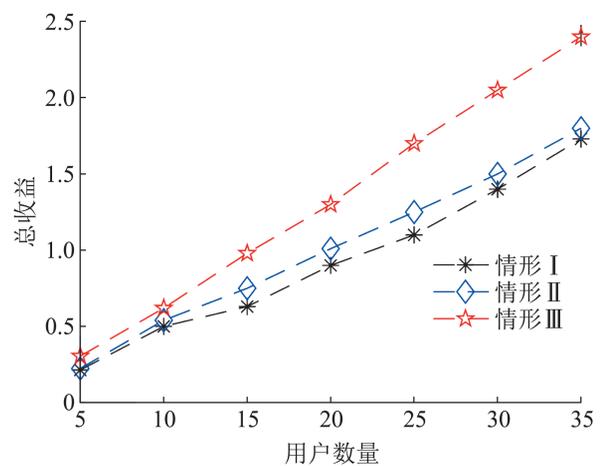
为了验证本文所提贝叶斯博弈的必要性和优点, 可以设置一组事后分析实验。由于信息壁垒, 云平台无法获知需求方的类型信息, 只知道对方类型的先验概率, 而需求方的类型组合可以有3种情形: 时间型(情形I)、成本型(情形II)、时间型与成本型(情形III)。因此, 在传统博弈优化方案下只能猜测并选择情形I或情形II类型的其中一种, 此做法存在侥幸可能, 而不完全信息贝叶斯静态博弈可以兼顾所有可能的类型。参数与3.1节设置相同, 对上述3种情形进行对比分析, 如表8和图9所示。

表8 不同情形对比分析
Table 8 Comparative analysis of different situations

方案	参与方	收益	C/元	t/min	Q	S
情形I	需求方	0.129	4 105.0	690	20.5	—
	云平台	0.085	860.5	—	—	0.8
情形II	需求方	0.025	4 150.0	695	20.5	—
	云平台	0.199	845.5	—	—	0.8
情形III	需求方	0.177	3 935.0	675	20.5	—
	云平台	0.128	812.5	—	—	0.8

由表8可知, 当云平台无法获知需求方具体类型的信息时, 采用传统的完全信息博弈方法, 此时共有2种情形分别为情形I与情形II, 若采用贝叶斯博弈优化方法, 情形III下的云制造系统总收益高于情形I与情形II两种特定类型, 相比完全

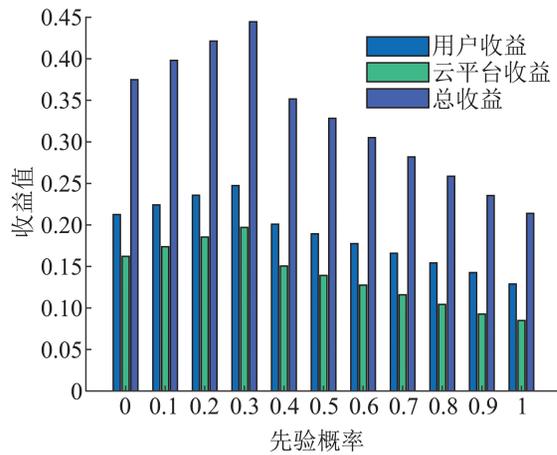
信息博弈方法收益均得到提升。因此, 当博弈过程中存在不完全信息时, 采用贝叶斯博弈组合优化方法能兼顾非完全信息下所有可能出现的情形, 最大化利益。此时, 云平台调度策略都是对自己和其他需求方所有类型组合的总体最优响应, 而不是对某一特定的类型组合的最优响应。根据图9可以发现, 随着需求方数量的不断增加, 情形III的优势越来越明显, 这是因为需求方数量增加更能兼顾所有类型, 不完全信息下的云平台与需求方的总收益明显优于完全信息下的结果, 证明本文方法更加合理可靠。

图9 不同情形下需求方数量的影响
Fig. 9 Influence of demand-side quantity in different situations

3.3 先验概率对结果的影响分析

先验概率的变化对需求方与云平台的收益有着直接的影响, 为分析先验概率 μ 对结果的影响, 分别将先 μ 设定为0、0.1、0.2、0.3、0.4、0.5、0.6、0.7、0.8、0.9和1进行求解, 如图10所示。

由图10可以看出, 随着概率的变化, 需求方与云平台的收益以及总收益同时在变化, 当先验概率 $\mu=0.3$ 时各方收益以及总收益达到最大值, 兼顾类型符合贝叶斯分布情况下的总收益, 使利益最大化, 当 $\mu=0$ 或者 $\mu=1$ 时, 只能满足特定需求方类型的利益即完全信息, 并不能兼顾各种类型。

图10 先验概率 μ 对各收益的影响Fig. 10 Influence of prior probability μ on each return

3.4 云制造分布式环境扰动实验

为了验证本文方法在实际云制造分布环境中的合理性与可行性，根据云制造分布式的特点，制造资源分布在各地，不受地理位置限制，资源种类较多，同时具有资源扰动性的问题，需要考虑到部分制造资源因人为退出或是受到云安全隐患而导致无法使用的问题，动态扰动更加符合实际情况，同样云制造平台中存在用户提交的订单任务，由于分布式云制造资源具有资源取消以及安全性等相关动态扰动，假设在制造过程中取消了 R_2 和 R_4 资源的制造能力，在该情况下去完成订单任务。根据上述表述可以得到相关候选资源集，相关参数设置与3.1节相同，通过对资源池中分布式制造资源的初筛可得子任务与相应候选资源集如表9所示。

任务	候选资源集
O_1	$\{R_1, R_3, R_5, R_{10}, R_{11}, R_{13}, R_{14}\}$
O_2	$\{R_6, R_7, R_9, R_{11}, R_{16}\}$
O_3	$\{R_1, R_3, R_6, R_8, R_{10}, R_{12}, R_{13}\}$
O_4	$\{R_5, R_9, R_{11}, R_{13}, R_{15}, R_{16}\}$
O_5	$\{R_1, R_3, R_6, R_8, R_{10}, R_{12}, R_{14}\}$
O_6	$\{R_3, R_5, R_7, R_{10}, R_{12}, R_{15}, R_{16}\}$

在分布式云制造环境中， R_2 和 R_4 资源在制造过程中由于扰动原因而丧失了制造能力，那么对于 O_2 、 O_3 、 O_4 、 O_6 在资源种类上失去了优势，为

了提高收益，只能寻找其他优势资源。表10为各算法求解的资源选择方案。如图11所示，在受到资源扰动时，在30代时资源扰动影响总收益下滑，后续重新资源配置趋于稳定，本文算法相对于文献[7, 10]总收益更高，更加可靠。

表10 最优解方案
Table 10 Optimal solution

算法	资源选择方案
文献[7]	$R_5 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{13} \rightarrow R_1 \rightarrow R_{16}$
文献[10]	$R_5 \rightarrow R_3 \rightarrow R_6 \rightarrow R_2 \rightarrow R_{15} \rightarrow R_{16}$
本文	$R_5 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{13} \rightarrow R_1 \rightarrow R_{16}$

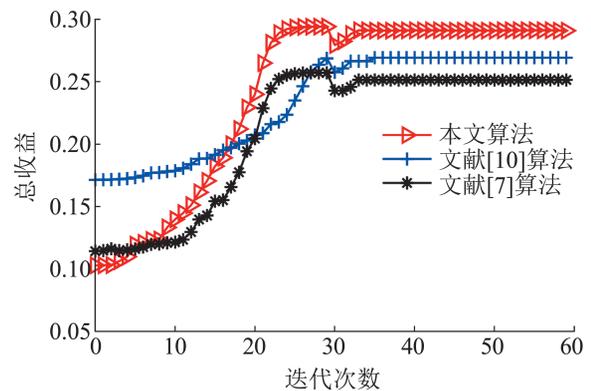


图11 不同算法收敛对比图

Fig. 11 Comparison diagram of algorithm convergence

由表11~12可以看出，在分布式环境资源扰动下，本文算法依然表现出良好的搜索性能，更适合解决本文的分布式制造资源优化配置问题，体现出算法的可行性与合理性。

表11 需求方算法求解对比分析

Table 11 Demand-side algorithm to solve comparative analysis

算法	收益	$C/\text{元}$	t/min	Q	CPU/s
文献[7]	0.148	4 095	720	19.8	38.6
文献[10]	0.158	4 020	735	19.5	35.4
本文	0.170	4 005	690	19.8	32.8

表12 云平台算法求解对比分析

Table 12 Cloud platform algorithm to solve comparative analysis

算法	收益	$C/\text{元}$	S	CPU/s
文献[7]	0.104	863.5	0.8	38.6
文献[10]	0.109	885.5	0.8	35.4
本文	0.116	827.5	0.8	32.8

4 结论

本文针对云制造过程中存在的不完全信息问题, 基于贝叶斯博弈提出了不完全信息的云制造群智能优化方法。该方法可以针对不同类型的需求方进行针对性资源配置, 求解出贝叶斯纳什均衡解, 以此提高云制造总收益。同时对粒子群算法提出改进, 引入高斯项分布特性将惯性权值设置为动态变化, 丰富粒子的多样性。本文同时提出新的速度扰动更新策略, 由此弥补粒子种群减少的多样性趋势。仿真结果表明: 改进算法相对于其他算法有较快的收敛速度与更好的解质量。通过算例进行不完全信息博弈和完全信息博弈的优化方案结果进行对比, 结果表明: 贝叶斯博弈优化方案在考虑了需求方与云平台之间的信息不完全情况下, 相对以往研究可以兼顾不同类型, 提高系统总收益, 更加符合实际情况, 然后验证了在云制造分布式环境扰动下的情况下, 本文方法依旧保持着良好的性能, 验证了本文所提方案的合理性和有效性。综上所述, 本文所提方案适用于含多种需求方类型的云制造组合优化, 具有较强的实用价值。本文仅考虑了云制造环境中的不完全信息, 实际上此过程较为复杂还存在其他不完全信息。因此, 考虑云平台与需求方间其他不完全信息的优化值得进一步研究。

参考文献:

- [1] 姚娟, 邢宾, 曾骏, 等. 云制造服务组合研究综述[J]. 计算机科学, 2021, 48(7): 245-255.
Yao Juan, Xing Bin, Zeng Jun, et al. Survey on Cloud Manufacturing Service Composition[J]. Computer Science, 2021, 48(7): 245-255.
- [2] Liu Yongkui, Wang Lihui, Wang Xi, et al. Scheduling in Cloud Manufacturing: State-of-the-art and Research Challenges[J]. International Journal of Production Research, 2019, 57(15/16): 4854-4879.
- [3] Alessandro Simeone, Deng Bin, Alessandra Caggiano. Resource Efficiency Enhancement in Sheet Metal Cutting Industrial Networks Through Cloud Manufacturing[J]. The International Journal of Advanced Manufacturing Technology, 2020, 107(3): 1345-1365.
- [4] Ding Shuhui, Han Jingliang, Meng Xiaojun, et al. Multi-granularity Modeling and Aggregation of Design Resources in Cloud Manufacturing[J]. IEEE Access, 2020, 8: 130797-130819.
- [5] 冯晨微, 王艳. 云制造系统并行任务优化调度[J]. 系统仿真学报, 2019, 31(12): 2626-2635.
Feng Chenwei, Wang Yan. Parallel Tasks Optimization Scheduling in Cloud Manufacturing System[J]. Journal of System Simulation, 2019, 31(12): 2626-2635.
- [6] Yu Chunxia, Zhang Luping, Zhao Wenfan, et al. A Blockchain-based Service Composition Architecture in Cloud Manufacturing[J]. International Journal of Computer Integrated Manufacturing, 2020, 33(7): 701-715.
- [7] Gilseung Ahn, Sun Hur. Multiobjective Real-time Scheduling of Tasks in Cloud Manufacturing with Genetic Algorithm[J]. Mathematical Problems in Engineering, 2021, 2021: 1305849.
- [8] Zhang Shuai, Xu Yangbing, Zhang Wenyu, et al. A New Fuzzy QoS-aware Manufacture Service Composition Method Using Extended Flower Pollination Algorithm[J]. Journal of Intelligent Manufacturing, 2019, 30(5): 2069-2083.
- [9] Alessandro Simeone, Alessandra Caggiano, Deng Bin, et al. A Deep Learning Based-decision Support Tool for Solution Recommendation in Cloud Manufacturing Platforms[J]. Procedia CIRP, 2019, 86: 68-73.
- [10] Si Wen, Qin Bingyang, Li Qingquan, et al. A Novel Adaptive Wavelet Threshold Estimation Based on Hybrid Particle Swarm Optimization for Partial Discharge Signal Denoising[J]. Optik, 2019, 181: 175-184.
- [11] Daniela Carlucci, Paolo Renna, Sergio Materi, et al. Intelligent Decision-making Model Based on Minority Game for Resource Allocation in Cloud Manufacturing[J]. Management Decision, 2020, 58(11): 2305-2325.
- [12] Chen Jian, Huang G Q, Wang Junqiang, et al. A Cooperative Approach to Service Booking and Scheduling in Cloud Manufacturing[J]. European Journal of Operational Research, 2019, 273(3): 861-873.
- [13] Liu Zhaohui, Wang Zhongjie. A Novel Truthful and Fair Resource Bidding Mechanism for Cloud Manufacturing[J]. IEEE Access, 2020, 8: 28888-28901.
- [14] Li Tao, Shahidehpour M. Strategic Bidding of Transmission-constrained GENCOs with Incomplete Information[J]. IEEE Transactions on Power Systems, 2005, 20(1): 437-447.
- [15] Ehsan Aghamohammadzadeh, Omid Fatahi Valilai. A Novel Cloud Manufacturing Service Composition Platform Enabled by Blockchain Technology[J].

- International Journal of Production Research, 2020, 58 (17): 5280-5298.
- [16] 黄宇, 吴思橙, 徐璟, 等. 不完全信息下计及环境成本的多能源集线器博弈优化调度[J]. 电力系统自动化, 2022, 46(20): 109-118.
Huang Yu, Wu Sicheng, Xu Jing, et al. Game Optimal Scheduling Among Multiple Energy Hubs Considering Environmental Cost with Incomplete Information[J]. Automation of Electric Power Systems, 2022, 46(20): 109-118.
- [17] 张维迎. 博弈论与信息经济学[M]. 上海: 格致出版社, 2012.
- [18] 罗云峰. 博弈论教程[M]. 北京: 清华大学出版社, 2007.
Luo Yunfeng. Game Theory[M]. Beijing: Tsinghua University Press, 2007.
- [19] Li Li, Wang Wanliang, Li Weikun, et al. A Novel Ranking-based Optimal Guides Selection Strategy in MOPSO[J]. Procedia Computer Science, 2016, 91: 1001-1010.
- [20] Eberhart R C, Shi Yuhui. Particle Swarm Optimization: Developments, Applications and Resources[C]// Proceedings of the 2001 Congress on Evolutionary Computation. Piscataway, NJ, USA: IEEE, 2001: 81-86.
- [21] 胡建秀, 曾建潮. 具有随机惯性权重的PSO算法[J]. 计算机仿真, 2006, 23(8): 164-167.
Hu Jianxiu, Zeng Jianchao. A Particle Swarm Optimization Model with Stochastic Inertia Weight[J]. Computer Simulation, 2006, 23(8): 164-167.
- [22] 张迅, 王平, 邢建春, 等. 基于高斯函数递减惯性权重的粒子群优化算法[J]. 计算机应用研究, 2012, 29(10): 3710-3712, 3724.
Zhang Xun, Wang Ping, Xing Jianchun, et al. Particle Swarm Optimization Algorithms with Decreasing Inertia Weight Based on Gaussian Function[J]. Application Research of Computers, 2012, 29(10): 3710-3712, 3724.
- [23] Jayadev Gyani, Ahsan Ahmed, Mohd Anul Haq. MCDM and Various Prioritization Methods in AHP for CSS: A Comprehensive Review[J]. IEEE Access, 2022, 10: 33492-33511.
- [24] 石翠翠, 刘媛华, 陈昕. 基于粒子群算法优化支持向量回归的水质预测模型[J]. 信息与控制, 2022, 51(3): 307-317.
Shi Cuicui, Liu Yuanhua, Chen Xin. Water Quality Prediction Model Based on Particle Swarm Optimization Support Vector Regression[J]. Information and Control, 2022, 51(3): 307-317.