Robotics and Computer Integrated Manufacturing 56 (2019) 127–139

Contents lists available at ScienceDirect

Robotics and Computer Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcim

Two-level multi-task scheduling in a cloud manufacturing environment

Feng Li[a], T.W. Liao[b], Lin Zhang[a,*]

[a] School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China
[b] Department of Mechanical & Industrial Engineering, Louisiana State University, Baton Rouge, LA 70803, USA

ARTICLE INFO

Keywords:
Multi-task scheduling
Two-level scheduling model
Ant colony optimization
Cloud manufacturing

ABSTRACT

Cloud manufacturing is a new paradigm of global manufacturing networks. In a cloud manufacturing environment, the problem of scheduling multiple heterogeneous tasks is very challenging. Tasks submitted to the cloud manufacturing platform possess different functional requirements and various complexities. They are decomposed into fine-grained subtasks with their precedence relationship to be processed by the aggregated distributed resources. From the perspective of each individual task, it is desirable to be completed in the shortest time possible without due date constraint. On the other hand, from the perspective of the cloud platform, all tasks being scheduled should also be completed as quickly as possible. How to schedule all the subtasks decomposed from multiple heterogeneous tasks to maximize the benefits not only to individual tasks themselves but also to the platform as a whole is an intractable but important issue in cloud manufacturing. In this paper, a two-level multi-task scheduling model is proposed. Two scheduling strategies based on the two-level scheduling model are presented and evaluated. The test results obtained from computational experiments suggest that the proposed two-level scheduling model is more effective than the traditional one-level model. The two-level two-optimization scheme in particular can find a better schedule beneficial not only to individual task but also to the cloud platform as a whole.

1. Introduction

With the development of global manufacturing, the mode of cloud manufacturing has attracted much attention. A cloud manufacturing platform, as one important part of cloud manufacturing system architecture, can be set up to aggregate a collection of distributed resources offered by different providers [1]. In cloud manufacturing, distributed resources are encapsulated into cloud services [2,3]. To provide best services to clients, the cloud manufacturing platform shall implement centralized planning and management. In most cases, these tasks submitted by the clients are complicated which have to be decomposed into several subtasks to be processed by the aggregated distributed resources. Precedence constraints which show a complicated relationship often exist in these subtasks. Meanwhile, not every resource registered in the platform can meet all kinds of task requests because of the limitation in ability. Hence, the eligibility of a resource to fulfill a subtask as part of a requested manufacturing service should also be taken into account. How to schedule multiple tasks and how to schedule all necessary subtasks in consideration of precedence constraints as well as resource eligibility are difficult and important issues.

To date, a number of researchers have studied scheduling issues in cloud manufacturing. However, so far only a few focus on the scheduling of multiple complicated tasks. Section 2 provides a more detailed literature review of past works. Multiple tasks are generally divided into two kinds: homogeneous and heterogeneous. The homogeneous tasks possess the same characteristics and are handled by identical production processes whereas heterogeneous tasks are dissimilar and have to be processed by a variety of different processes. As a result of mass customization, requirements are expected to be diverse. Hence, it is more practical and sensible to focus on scheduling of heterogeneous multiple tasks. Most of previous works on multi-task scheduling treat all subtasks decomposed from different tasks the same and schedule these subtasks indistinguishably. In doing so, an individual task might take a long time to be completed. In fact, each task hopes to be completed as soon as possible if there is no due date constraint. However, there is more than one task to be completed and benefit conflicts could exist among different tasks. Hence, instead of minimizing the makespan of a certain task, the average of the makespan of individual tasks should be minimized. Meanwhile, from the perspective of the cloud manufacturing platform, all requests should also be completed as soon as possible so that the involved resource services can be utilized in a high efficient way and have a chance to quickly response to the new coming task sets. Hence, the total makespan of multiple tasks should also be minimized.

* Corresponding author.
E-mail address: zhanglin@buaa.edu.cn (L. Zhang).

https://doi.org/10.1016/j.rcim.2018.09.002
Received 18 February 2018; Received in revised form 3 September 2018; Accepted 3 September 2018

Based on the above-mentioned ideas, a new two-level scheduling model is proposed to deal with multiple tasks scheduling requests submitted to the platform. In this model, the upper level is designed to generate a schedule of multiple tasks. The lower level, on the other hand, focuses on the scheduling of subtasks decomposed from requests, with the objective of minimizing makespan of every individual complicated request in consideration of not only precedence constraint but also resource eligibility. In other words, the upper level deals with task-level scheduling whereas the lower level handles subtask-level scheduling. For the solution of the proposed two-level scheduling model, two different schemes, i.e. two-level-one-optimization scheduling (TLOOS) and two-level-two-optimization scheduling (TLTOS), are developed for comparison. The two-level scheduling schemes differ in how tasks are scheduled (i.e., task-level scheduling). Specifically, the TLOOS method sorts multiple tasks according to the “first in first out” rule whereas the TLTOS strategy finds a near-optimal sequence of tasks with the objective of minimizing the total makespan of all requests. For both schemes, the effective improved ACO algorithm which was proposed by Liao et al [4] is chosen and adapted for the scheduling of subtasks. To evaluate the goodness of the proposed model and solution methods, randomly generated data with varying number of requests, varying number of subtasks in each request and varying number of registered resources are tested. For comparison, a one-level scheduling scheme is also implemented. Nonparametric statistical tests are performed to determine the statistical significance of their performance differences.

The contributions of this paper are listed as follows: (1) A multiple complicated tasks model with presence relationship and a multiple services model with service eligibility are both taken into account. (2) A novel two-level scheduling model which can benefit not only individual task but also the cloud platform is proposed and an adjustment algorithm is designed in the process of scheduling. (3) Based on the model, two new optimization schemes are presented which can focus on individual performance as well as the whole performance. (4) Comparative studies between one-level scheduling scheme and two-level-scheduling methods is carried out. (5) Nonparametric statistical tests are carried out among different methods and the advantages of one method over others are identified.

The rest of the paper is organized as follows. Section 2 presents the related works about scheduling in a cloud manufacturing environment. In Section 3, problem description is presented. In Section 4, the proposed scheduling models and solution methods are described in detail. The results obtained from computational experiments are given in Section 5. At last, conclusions are drawn and topics for future works are identified in Section 6.

2. Literature review

Scheduling related issues in the cloud manufacturing environment have been studied recently. Tao et al. [5] proposed a supply-demand matching simulator which included the process of matching and scheduling algorithms selection and design. Many researches dealt with the scheduling issues from different perspectives. Table 1 summarizes past works on task scheduling in the field of cloud manufacturing. Each work is distinguished with the following features: task quantity, task characteristics, scheduling approach, scheduling objectives and optimization method.

Tasks submitted to a cloud manufacturing platform can be simple or complicated but most often complicated tasks. A simple task refers to a single-service requirement task that requires only one service for its completion, whereas a complicated task is a multi-service requirement task that consists of a series of subtasks and requires multiple services to be completed. The number of tasks can be single or multiple. Of course, multiple tasks are more realistic because a cloud manufacturing platform cannot survive without taking requests from multiple clients. Therefore, four combinations of task types are possible: single simple, single complicated, multiple simple, and multiple complicated. Multiple tasks can be further distinguished into two kinds: multiple homogeneous tasks and multiple heterogeneous tasks.

Scheduling of single simple task is trivial; hence no paper addresses this topic. Scheduling of other three kinds of tasks has been studied for many years. In order to solve cloud design resources scheduling problem, Wei and Tian [6] formulated a scheduling model and solved it by a genetic algorithm (GA) with the fitness function defined by linearly weighted multiple objectives to minimize cost and time and to maximize quality of service (LWCTQ). Cao et al. [7] formulated service selection and scheduling model as a multi-objective optimization problem in consideration of time, quality, cost and service rating (TQCS). They combined multiple objectives into one by weighting with weights determined by analytic hierarchical process (AHP). Hence, the objective considered in this paper is linearly weighted TQCS (LWTQCS). Then, they proposed an ant colony optimization algorithm called ACOS to find the near-optimal solution that is valid in terms of no time conflict in selecting the same service for different subtasks at the same time. Lin and Chong [8] developed an improved GA to schedule multiple simple task in cloud manufacturing in consideration of task precedence constraint and resource limitation with the objective to minimize the maximum completion time, i.e. makespan.

Cheng et al. [9] established comprehensive utility models which consider energy consumption, cost and risk for different sides (i.e. provider, consumer and operator). Four scheduling models were proposed to maximize the utility of each participant, and the system as a whole; these models are named provider centered, consumer centered, operator centered and system centered, respectively. The experimental results showed that the system-centered cooperative scheduling method has the highest potential for realizing the aim of cloud manufacturing to produce higher maximal utilities not only for the whole system but also for the three key participants. Cheng et al. [10] presented a multitask oriented virtual resource integration and scheduling problem in consideration of resource time-sharing and correlation among resources. A genetic algorithm based on real number matrix coding was proposed to solve the scheduling problem with the objective to maximize the linearly weighted profit and number of tasks being executed (LWPN). Jian and Wang [11] addressed the problem of scheduling a batch of tasks with the same characteristics and the same production processes using an improved particle swarm optimization algorithm (IPSO) with the objective to minimize the total cost.

In order to efficiently exploit distributed resources, Li et al. [12] proposed a cloud manufacturing scheduling model so that industrial robots in different enterprises could cooperatively handle a batch of tasks. Specifically, they considered four robot deployment methods including random deployment, robot-balanced deployment, function-balanced deployment, and location-aware deployment and proposed three subtask-scheduling strategies for three optimization objectives, including load-balance of robots (LB), minimizing overall cost, and minimizing overall processing time, respectively. These strategies were implemented by genetic algorithm and simulation results demonstrated that each strategy could achieve the respective optimization objective. In addition, the results also showed that the physical distance between two enterprises could influence the overall cost, and location-aware deployment lead to smaller transportation cost.

The optimization methods of all the above-mentioned papers are based on single objective or turn multi-objective into single objective. Some other researches pay their attention to multi-objective optimization. Jiang et al. [13] formulated a mathematical model to describe a cloud-based disassembly system, which provides disassembly services for multiple requests, and applied a multi-objective genetic algorithm, specifically NSGA-II, to solve the model with two optimization objectives, i.e., minimizing the total makespan and total cost. Focusing on reconfigurable assembly line (RAL) in cloud manufacturing, Yuan et al. [14] proposed a multi-objective optimization scheduling model to improve the production efficiency of a RAL and solved the model with an