Computers & Industrial Engineering 117 (2018) 152–163

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

Bicriterion total flowtime and maximum tardiness minimization for an order scheduling problem

Chin-Chia Wu[a], Shang-Chia Liu[b], Tzu-Yun Lin[a], Tzu-Hsuan Yang[a], I-Hong Chung[a], Win-Chin Lin[a,*]

[a] Department of Statistics, Feng Chia University, Taichung 40724, Taiwan
[b] Business Administration Department, Fujen Catholic University, Hsinpei City, Taiwan

ARTICLE INFO

Keywords:
Order scheduling
Iterated greedy algorithm
Particle swarm optimization algorithm
Total completion time
Maximum tardiness

ABSTRACT

An order scheduling problem arises in numerous production scheduling environments. Makespan, mean flow time, and mean tardiness are the most commonly discussed and studied measurable criteria in the research community. Although the order scheduling model with a single objective has been widely studied, it is at odds with real-life scheduling practices. In practice, a typical manager must optimize multiple objectives. A search of the literature revealed that no articles had addressed the issue of optimizing an order scheduling problem with multiple objectives. Therefore, an order scheduling model to minimize the linear sum of the total flowtime and the maximum tardiness is introduced in this study. Specifically, several dominance relations and a lower bound are derived to expedite the search for the optimal solution. Three modified heuristics are proposed for finding near-optimal solutions. A hybrid iterated greedy algorithm and a particle swarm colony algorithm are proposed to solve this problem. Finally, a computational experiment is conducted to evaluate the performances of all proposed algorithms.

1. Introduction

In many manufacturing environments, the producer meets customer deadlines and develops efficient production items at the same time because the demand for production items increases sharply. In light of encouraging the demand on asking for on-time delivery of high quality product, producers go through technologies based on dispatching and scheduling to reduce the production time.

Recently, customer order scheduling has become a popular field of research. The applications of customer order scheduling can be found in manufacturing environments: a converting operation in a process industry by Leung, Li, and Pinedo (2005, 2006b) and a car repair shop by Yang (2005). In these applications, a planning developer independently creates one of a set of separate productive parts, each of which is considered to be finished when the full set of production parts has been completed. Ahmadi, Bagchi, and Roemer (2005) presented another example of customer order scheduling in the manufacturing of semi-finished lenses.

Numerous researchers have studied order scheduling. Regarding the total completion time criterion, Leung et al. (2005), Leung, Li, and Pinedo (2006a) and Wagneur and Sriskandarajah (1993) have discussed the complexity of problems on two or more machines; Ahmadi et al. (2005), Wang and Cheng (2003), and Sung and Yoon (1998) have developed approximate algorithms to solve such problems. For the criterion of total weighted order completion time, Sung and Yoon (1998) and Ahmadi and Bagchi (1993) have shown that order scheduling is NP-hard or strongly NP-hard for the two-machine case. Ahmadi et al. (2005), Leung, Li, and Pinedo (2007a, 2008), Leung, Li, Pinedo, and Zhang (2007b), Leung, Lee, Ng, and Young (2008), Wang and Cheng (2003), and Chen and Hall (2001) have separately proposed heuristics to find near-optimal solutions. As to the order scheduling models involving due dates, readers may refer to Blocher, Chhajed, and Leung (1998), Erel and Ghosh (2007), Hsu and Liu (2009), Lee (2013), Leung et al. (2006b), Yang (2005), Yang and Posner (2005), Xu et al. (2016), Lin et al. (2017), and Wu, Liu, Zhao, Wang, and Lin (2017).

The aforementioned works have focused on minimizing a single objective. However, in competitive markets, producers must focus on minimizing the production periods and the manufacturing costs. Production managers must determine how to minimize the tradeoff costs between short production periods and on-time delivery of customer orders. Motivated by the lack of research on the issue of optimizing an order scheduling problem with multiple objectives, this study addresses an order scheduling problem to minimize a linear sum of the total flow time and the maximum tardiness.

* Corresponding author.
E-mail address: linwc@fcu.edu.tw (W.-C. Lin).

https://doi.org/10.1016/j.cie.2018.01.011
Received 27 March 2017; Received in revised form 11 January 2018; Accepted 15 January 2018
Available online 31 January 2018
0360-8352/ © 2018 Elsevier Ltd. All rights reserved.

The main contribution of this paper is providing an objective that is a weighted average of two performance measures and developing a lower bound for the branch-and-bound procedure. This paper also uses three heuristics and two metaheuristics (iterated greedy algorithm and particle swarm optimization algorithm) reported in the literature to generate an upper bound for the branch-and-bound procedure. The remainder of this paper is organized as follows. Section 2 introduces the notation and problem definition. Section 3 presents some propositions and a lower bound for the branch-and-bound algorithm. Section 4 provides the iterative greedy (IG) algorithm and several heuristics. Section 5 reports the experimental simulations of all the proposed algorithms, and Section 6 provides conclusions and suggestions.

2. Notation definition and problem description

The notations used throughout the paper are defined as follows.

n: denotes the number of orders;
m: denotes the number of machine;
M_k: denotes the machine codes k, $k = 1, 2,\dots, m$;
O_i: denotes the order codes i, $i = 1, 2, \dots, n$.
σ, σ_1, σ_2: denote the schedules of the given n orders;
π_1, π_2: denote two partial schedules of the given n orders;
t_{ik}: denotes the processing time of order O_i to be operated on machine k, $k = 1, 2, \dots, m$;
s_k: denotes the starting time of an order on machine k, $k = 1, 2, \dots, m$;
: denotes the due date of order O_i $i = 1, 2, \dots, n$;
$C_i(\sigma_1)$, $C_j(\sigma_1)$: denote the completion times of orders O_i and O_j in σ_1;
$C_j(\sigma_2)$,$C_i(\sigma_2)$: denote the completion times of orders O_j and O_i in σ_2;
$[r]$: denotes the r^{th} position of orders in a sequence;
$T_i(\sigma_1)$,$T_j(\sigma_1)$: denote the tardiness values of orders O_i and O_j in σ_1;
$T_j(\sigma_2)$,$T_i(\sigma_2)$: denotes the tardiness values of orders O_j and O_i in σ_2; where

$$T_i(\sigma) = \max\{0,C_i(\sigma)-d_i\}$$

.
$\sum_{i=1}^{n} C_i(\sigma)$: denotes the total completion time of n orders in σ.
$T_{\max}(\sigma)$: denotes the maximum tardiness of n orders in σ, or $T_{\max}(\sigma) = \max_{1\leqslant i\leqslant n}\{T_i(\sigma)\}$.
$v_1\sum_{i=1}^{n} C_i(\sigma) + (1-v_1)T_{\max}(\sigma)$: denotes the objective function of this study, where $0 < v_1 < 1$.

The problem under study is as follows. A set of n orders (placed by n different clients) must be processed on m machines, which are arranged in parallel. Each item can be executed on one dedicated machine. Preemption, machine breakdown, and interruption are not allowed. The ready times for the n orders are not included in this problem. The objective function of this problem is to determine a schedule that optimizes a linear sum of the total flowtime (or total completion time) and maximum tardiness of n orders. Ahmadi et al. (2005) confirmed that the problem of minimizing the total flowtime is NP-hard, and thus this problem is NP-hard as well. Therefore, this paper introduces several dominance relations and a lower bound to be used in a branch-and-bound method for the optimal solution. Following that, some heuristics are proposed, and an iterated greedy (IG) algorithm and a particle swarm optimization algorithm are developed to search for approximate solutions.

3. Properties and a lower bound

This section derives some properties and a lower bound to speed up the branch-and-bound search for an approximate solution. Let $\sigma_1 = (\pi_1,O_i,O_j,\pi_2)$ and $\sigma_2 = (\pi_1,O_j,O_i,\pi_2)$ be two order schedules in which π_1 and π_2, respectively, are partial order sequences. To show that σ_1 dominates σ_2, it suffices to show the following:

$$v_1(C_i(\sigma_1) + C_j(\sigma_1)) + (1-v_1)\max\{T_i(\sigma_1),T_j(\sigma_1)\} \leqslant v_1(C_j(\sigma_2) + C_i(\sigma_2)) + (1-v_1)\max\{T_j(\sigma_2),T_i(\sigma_2)\}$$

and $C_j(\sigma_1) < C_i(\sigma_2)$, for all v_1 such that $0 < v_1 < 1$.

Property 1. Consider two adjacent orders O_i and O_j in which $s_k + t_{ik} + t_{jk} \leqslant d_j\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} < \max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$ for all $k = 1, 2, \dots, m$, then σ_1 dominates σ_2.

Proof. The completion times of orders i and j in schedules σ_1 and σ_2 are respectively,

$$C_i(\sigma_1) = \max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\}C_j(\sigma_1) = \max_{1\leqslant k\leqslant m}\{s_k + t_{ik} + t_{jk}\}$$

$$C_j(\sigma_2) = \max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$$

and

$$C_i(\sigma_2) = \max_{1\leqslant k\leqslant m}\{s_k + t_{jk} + t_{ik}\} = C_j(\sigma_1).$$

It follows from $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} < \max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$ for all $k = 1, 2, \dots, m$, that $C_i(\sigma_1) \leqslant C_j(\sigma_2)$, and thus $C_i(\sigma_1) + C_j(\sigma_1) \leqslant C_j(\sigma_2) + C_i(\sigma_2)$.

Because $C_i(\sigma_1) \leqslant C_i(\sigma_2)$,

$$T_i(\sigma_2) = \max\{C_i(\sigma_2)-d_i,0\} \geqslant T_i(\sigma_1) = \max\{C_i(\sigma_1)-d_i,0\}.$$

In addition, by $s_k + t_{ik} + t_{jk} \leqslant d_j$ for all $k = 1, 2, \dots, m$, we have $d_j \geqslant C_j(\sigma_1)$. Thus, $T_i(\sigma_2) = \max\{C_i(\sigma_2)-d_i,0\} \geqslant T_j(\sigma_1) = \max\{C_j(\sigma_1)-d_j,0\} = 0$. Consequently $\max\{T_j(\sigma_2),T_i(\sigma_2)\} \geqslant T_i(\sigma_2) \geqslant \max\{T_i(\sigma_1),T_j(\sigma_1)\}$, as required. □

Property 2. Consider two adjacent orders O_i and O_j in which $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} < \max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$ and $d_i \leqslant d_j$, then σ_1 dominates σ_2.

Proof. By the proof of Property 1, it suffices to show that $T_i(\sigma_2) \geqslant T_j(\sigma_1)$. Indeed, with $C_j(\sigma_1) = C_i(\sigma_2)$ and $d_i \leqslant d_j$, we have $C_i(\sigma_2)-d_i \geqslant C_j(\sigma_1)-d_j$, implying that $T_i(\sigma_2) = \max\{C_i(\sigma_2)-d_i,0\} \geqslant T_j(\sigma_1) = \max\{C_j(\sigma_1)-d_j,0\}$, as required. □

Property 3. Consider two adjacent orders O_i and O_j in which $s_k + t_{ik} + t_{jk} \leqslant \min\{d_i,d_j\}$ for all $k = 1, 2, \dots, m$, and $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} < \max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$, then σ_1 dominates σ_2.

Proof. The condition $s_k + t_{ik} + t_{jk} \leqslant \min\{d_i,d_j\}$ implies that $T_i(\sigma_2) = T_j(\sigma_1) = 0$. Thus, by the proof of Properties 1 and 2, the result holds. □

Property 4. Consider two adjacent orders O_i and O_j in which $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\}-d_i \geqslant 0$, $\max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\} \geqslant d_j$, $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\}-d_i \geqslant \max_{1\leqslant k\leqslant m}\{s_k + t_{ik} + t_{jk}\}-d_j$, $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} \leqslant (1-v_1)\max_{1\leqslant k\leqslant m}\{s_k + t_{ik} + t_{jk}\} + v_1\max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$, then σ_1 dominates σ_2.

Proof. By the proof of Property 1, $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} \geqslant d_i$ and $\max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\} \geqslant d_j$ give that $T_i(\sigma_1) = C_i(\sigma_1)-d_i,T_j(\sigma_1) = C_j(\sigma_1)-d_j,T_j(\sigma_2) = C_j(\sigma_2)-d_j$,and $T_i(\sigma_2) = C_i(\sigma_2)-d_i$. Moreover, $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\}-d_i \geqslant \max_{1\leqslant k\leqslant m}\{s_k + t_{ik} + t_{jk}\}-d_j$ gives $C_i(\sigma_1)-d_i \geqslant C_j(\sigma_1)-d_j$ and $C_i(\sigma_2)-d_i \geqslant C_j(\sigma_2)-d_j$, implying that $\max\{T_i(\sigma_1),T_j(\sigma_1)\} = T_i(\sigma_1)$ and $\max\{T_i(\sigma_2),T_j(\sigma_2)\} = T_i(\sigma_2)$. Thus, by $\max_{1\leqslant k\leqslant m}\{s_k + t_{ik}\} \leqslant (1-v_1)\max_{1\leqslant k\leqslant m}\{s_k + t_{ik} + t_{jk}\} + v_1\max_{1\leqslant k\leqslant m}\{s_k + t_{jk}\}$, we have

$$\begin{aligned}&v_1(C_j(\sigma_2) + C_i(\sigma_2)) + (1-v_1)\max\{T_j(\sigma_2),T_i(\sigma_2)\}-v_1(C_i(\sigma_1) + C_j(\sigma_1))\\&-(1-v_1)\max\{T_i(\sigma_1),T_j(\sigma_1)\} = v_1C_j(\sigma_2) + (1-v_1)(C_i(\sigma_2)-d_i)-v_1C_i(\sigma_1)\\&-(1-v_1)(C_i(\sigma_1)-d_i) = (1-v_1)C_i(\sigma_2)\\&+ v_1C_j(\sigma_2)-C_i(\sigma_1) \geqslant 0\end{aligned}$$

as required. □

Property 5. Consider two adjacent orders O_i and O_j in which

Download English Version:

https://daneshyari.com/en/article/7541408

Download Persian Version:

https://daneshyari.com/article/7541408

Daneshyari.com