Computers & Industrial Engineering 121 (2018) 27–35

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

A minimax linear programming model for dispatching rule selection

Gholam R. Amin[a,*], Ahmed El-Bouri[b]

[a] Faculty of Business, University of New Brunswick at Saint John, Saint John, NB E2L 4L5, Canada
[b] Department of Operations Management and Business Statistics, College of Economics and Political Science, Sultan Qaboos University, Oman

ARTICLE INFO

Keywords:
Dispatching rules
Minimax linear programming
Aggregation
Multi-criteria decision making
Data envelopment analysis

ABSTRACT

Dispatching rule selection is an important problem in production scheduling. This paper introduces a minimax linear programming (LP) model for dispatching rule selection in the presence of multiple criteria. The multi-criteria dispatching rule selection problem is first converted into a preference voting system, and a minimax LP model is then introduced for solving the corresponding problem. The advantage of this conversion is that it provides a way to identify dispatching rules that are moderately good in all criteria, rather than selecting dispatching rules that are good respect to only a few variables. An experimental study considering two different production priority settings is used to show the applicability of the proposed method.

1. Introduction

Scheduling of operations in a job shop environment is a widely researched topic in production control, and optimal solutions are generally difficult to arrive at, because the problem is known to be NP-hard (Lawler, Lenstra, Rinooy Kan, & Shmoys, 1993). Therefore, numerous different methods have been proposed for obtaining efficient, if not near-optimal, schedules for various scheduling objectives. A common approach that has been favoured for its simplicity is the application of dispatching rules for deciding which job to load next on machines as they become free (Ouelhadj & Petrovic, 2009).

In using dispatching rules, a production schedule evolves as a result of the dispatching decisions made at the machines, and no a priori schedule needs to be constructed. This real-time applicability of dispatching rules makes them an appealing alternative in dynamic job shops, where continuous arrival of new jobs, or machine breakdowns create new conditions that typically render an existing schedule immediately obsolete.

There are a large number of different dispatching rules that may be used in scheduling job shops, but there is no single dispatching rule that dominates the others for a given performance criterion under all conditions. This creates the problem of deciding which dispatching rule to use for a job shop, given a particular environment and performance criteria. This problem becomes even more complex when multiple criteria are considered (Adibi, Zandieh, & Amiri, 2010; Cheng, Chiang, & Fu, 2011; Vázquez-Rodríguez & Petrovic, 2010).

There exist various measures of performance, and these are typically functions of completion times and job due dates. In the present study, the scheduling objective is to maximize performance with respect to a series of multiple criteria. Scheduling under multiple performance criteria is a challenging task because one schedule which may be excellent according to one criterion, might be rather poor with respect to another (Adibi et al., 2010). Typically, a production schedule for multiple criteria may be evaluated based on how well it satisfies the different criteria in aggregate (El-Bouri & Amin, 2015).

This paper proposes a novel method for dispatching rule selection with multiple criteria by introducing a minimax linear programming (LP) model. The Multi-criteria decision making problem is converted into a preference voting model, where the candidates are the set of alternative dispatching rules. The next section reviews the research on multi-criteria scheduling, and this is followed in Section 3 by a presentation of the proposed minimax LP approach. Section 4 covers a computational analysis for evaluating the proposed approach in a job shop with more than 25 dispatching rule candidates. The paper concludes in Section 5 with a discussion of the results.

2. Literature review

A major difference in scheduling for dynamic production systems, as opposed to static systems, is that in static systems all jobs are available at the start of production, and once a schedule has been constructed there is no need to revise it afterwards. Dynamic production systems are characterized by uncertainty, exhibited in the form of unexpected events, such as arrival of new job orders during production, and machine breakdowns or re-work of jobs. Research in multi-objective scheduling for production systems has tended to focus on static

* Corresponding author.
E-mail address: gamin@unb.ca (G.R. Amin).

https://doi.org/10.1016/j.cie.2018.05.021
Received 11 December 2016; Received in revised form 29 December 2017; Accepted 14 May 2018
Available online 16 May 2018

production systems (see Nagar, Haddock, & Heragu, 1995; Lei, 2009 for research surveys).

With regard to dynamic production environments, a number of different scheduling approaches exist. These include dispatching rules, heuristics and meta-heuristics (Zhou, Nee, & Lee, 2009), multi-agent systems (Lou, Liu, Zhou, Wang, & Sun, 2012), knowledge-based systems and artificial intelligence (Calis & Bulkan, 2015). Proposed scheduling methods are frequently hybrids of the preceding approaches (see Adibi et al., 2010 as example). A general review of dynamic scheduling approaches is available in Ouelhadj and Petrovic (2009), while a survey of research specifically in the area of dynamic scheduling with multiple objectives was recently published by Shen, Zhang, and Fu (2014).

Ouelhadj and Petrovic (2009) refer to the definition of dynamic scheduling under three categories: completely reactive, predictive-reactive, and robust pro-active scheduling. Priority dispatching rules fall under the first category, completely reactive, in that scheduling decisions are done in real time based on current machine, job and system attributes, without a pre-constructed schedule to follow. There exists a large array of dispatching rules that can be applied for various shop systems (job shops, flow shops, parallel machine systems, flexible manufacturing systems, etc.) and different scheduling objectives (Blackstone, Phillips, & Hogg, 1982; Haupt, 1989; Ramasesh, 1990). Additional studies on the performance of dispatching rules have been presented in Rajendran and Holthaus (1999) and Sabuncuoglu (1998).

Most of the published research has tended to consider the performance of dispatching rules mainly with regard to a single performance criterion. Studies of dispatching rules with multiple criteria are less common. Chang, Sueyoshi, and Sullivan (1996) employed data envelopment analysis (DEA) to rank 42 dispatching rules for a static job shop with seven performance measures. The performance measures serve as the DEA inputs, with computation time being adopted as the single output. The conclusion was that the shortest processing time divided by total remaining work rule was the most efficient. Braglia and Petroni (1999) also applied DEA to identify top performing dispatching rules in a job shop where the performance objectives were to minimize in-process waiting times, machine idle times and queue time, all three considered as the outputs of the DEA model. For inputs, the model expressed four other performance criteria in the form of two inputs reflecting resources used, namely dollar days and backlog cost. The outputs in the DEA models discussed in these two previous research studies are based on 'more-is-better' variables, but most of the performance criteria are of the 'less-is-better' (minimization) type. Moreover, specifying inputs and outputs from multiple variables in dispatching rules is not an easy task as discussed in El-Bouri and Amin (2015). As a result, any conclusions based on these types of DEA may be invalid.

Multi-attribute decision-making tools other than DEA have also been used for dispatching rule selection. Petroni and Rizzi (2002) suggested a ranking scheme of dispatching rules based on fuzzy logic, while Kuo, Yang, Cho, and Tseng (2008) applied grey relational analysis for the same problem considered in Braglia and Petroni (1999). Nguyen, Zhang, Johnston, and Tan (2013) employed a hyper-heuristic method that utilizes multi-objective genetic programming to develop a Pareto front of non-dominated dispatching rules which can be used to support a dispatching decision. More recently, Huang and Suer (2015) embedded dispatching rules in chromosomes in applying a genetic algorithm with fuzzy satisfaction levels for deciding which job to load next in a job shop environment, under four different performance criteria.

The present study proposes a minimax LP model for ranking dispatching rules for Multi-criteria scheduling in a conventional job shop environment. As shown in El-Bouri and Amin (2015), the standard DEA method cannot be used directly for dispatching rule selection. This is due to difficulty in selecting inputs and outputs from multiple criteria. The proposed alternative method instead combines the ordered weighted averaging (OWA) operator and DEA methods in order to identify the top performing dispatching rules. In the OWA operator, generating the OWA weights is a crucial step (Amin, 2007; Emrouznejad & Amin, 2010; Wang & Parkan, 2005; Yager, 1988). The minimax disparity model in El-Bouri and Amin (2015) determines the OWA weights by minimizing the distance between the importance weights of any two consecutive components or variables. This assumption may not be realistic especially when there is priority between different variables. In this paper, an alternative dispatching rule selection method using a minimax LP model is introduced. The dispatching rules selection as a Multi-criteria decision making problem is converted into a preference voting system, before the proposed minimax LP model is applied in order to determine the top performing rules. The proposed minimax LP approach is investigated here for two production settings that have different orderings of the scheduling priorities, and it is shown that the model is able to identify top performed dispatching rules to suit either of the two scenarios. Generally, the approach proposed in this paper has the advantage of ability to select dispatching rules that are aggregately good in all criteria, rather than just a few variables.

3. Problem definition

A dynamic job shop operating under multiple performance criteria is considered in this paper. The job shop consists of m machines that are available for performing required operations on arriving job orders. Each job requires m number of operations that have to be performed in a specific sequence through the machines. Therefore, every job follows a designated route through the shop, defined by the order in which the machines must be visited. A machine is visited only once by each job, and arriving jobs usually have different route patterns. An operation on a machine can be performed only if the job's preceding operation has been completed. Furthermore, a machine cannot process operations for two different jobs at the same time. Once a machine completes operations on a job, that job is then transferred immediately to the next machine on its route. If that machine happens to occupied, the arriving job instead joins a queue and waits until it is selected for processing. When a machine completes operations for a job, it becomes 'released', and a decision has to be made with regard to which of the jobs waiting for it, if any, is to be processed next.

The job shop receives job orders in a continuous manner at a prevailing arrival rate, λ. A job's route and processing time requirements on the machines become known only after the job order has arrived. The goal is to schedule the jobs in such a way that a set of performance objectives are fulfilled to the maximum extent possible.

Let

j = index representing job number.
k = index representing machine number.
τ = current time.
N = total number of jobs processed.
a_j = arrival time of job j at the jobshop.
$a_{j,k}$ = arrival time of job j at machine k.
$p_{j,k}$ = processing time required by job j at machine k.
D_j = due date for job j.
C_j = Completion time of job j's final operation.
t_j' = total of the processing times of the operations at the next machine on job j's route.
(t_j' = 0 for job j's final operation).
p_j' = processing time required by job j at the next machine on its route.
(p_j' = 0 for job j's final operation).
$o_{j,k}$ = number of remaining operations for job j after machine k.
R_j = sum of the remaining processing times for job i, from the current machine k up to and including the last machine m.
$\pi_{j,k}$ = priority index of job j at machine k.
Q_k = number of jobs in queue at machine k.