Simulation Modelling Practice and Theory 40 (2014) 12–27

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

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

Production system-based simulation for backward on-line job change scheduling

Taedong Kim [a], Byoung K. Choi [a,b,*]

[a] VMS Lab., Department of Industrial and Systems Engineering, KAIST, 335 Gwahak-ro, Yuseong-gu, Daejeon, 305-701, Republic of Korea
[b] King Abdulaziz University, Jeddah, Saudi Arabia

ARTICLE INFO

Article history:
Received 26 March 2012
Received in revised form 30 July 2013
Accepted 7 August 2013
Available online 6 September 2013

Keywords:
On-line job change scheduling
Production system
Simulation

ABSTRACT

Backward on-line job change scheduling, referring to the on-line job change scheduling of a current processing step to satisfy the job change schedule of the subsequent processing step, is a common problem in modern Fabs. In this research, the production system-based simulation methodology is proposed to solve the backward on-line job change scheduling problem. This simulation is processed by the state change that is caused by an execution of the operator, and it finds the schedule with the best handle values considering the current status. Several simulation runs with diverse handle values were required to find the best values because the status of the shop floor can change dynamically. To validate the simulation, this production system-based simulation is applied to the on-line job change scheduling of a tire belt processing step as part of the tire manufacturing process.

1. Introduction

The backward on-line job change scheduling (BJCS) problem is an on-line job change scheduling problem between two processing steps, involving a number of machines with various processing times separated by a buffer. On-line job change scheduling implies job change scheduling for a real-world operation, requiring the application of current statuses of the shop floor, such as the current loading status of machines and the current WIP information, and to make the job change scheduling no longer than 10 min in order to apply the current status rapidly or to handle disturbance situations immediately. The job change schedule determines the start time of the job change, the production quantity, and the operating machines for each job-type. A job change is equivalent to the operation required when the loading job-type is switched and the job is the unit of production in the processing step and has a job-type attribute. The BJCS problem is defined as follows: given a job change schedule of the following processing step, the task is to find the job change schedule of the current processing step that minimizes the starving time of the following processing step and the job change loss time of the current processing step and holds the WIP level between the current and the following processing step steady. The starving time is sum of the idle time during the processing step caused by the late arrival of a job in the previous processing step. The details of the starving time are described in the Appendix A. The job change loss time is defined as the sum of the job change time in the processing step. The BJCS problem uses the job change schedule of the following processing step as its input. The output of the problem is the job change schedule of the current processing step. The target WIP, the initial WIP for each job-type and the loading status of the machines are also part of the input information. Reference information such as the processing time, the job change time and the bill of processes are used, and several constraints and requirements should be satisfied. The constraints and require-

---

[*] Corresponding author at: VMS Lab., Department of Industrial and Systems Engineering, KAIST, 335 Gwahak-ro, Yuseong-gu, Daejeon, 305-701, Republic of Korea. Tel.: +82 423503115; fax: +82 423503110.

E-mail addresses: taedong.kim@vmslab.kaist.ac.kr (T. Kim), bkchoi@kaist.ac.kr (B.K. Choi).

1569-190X/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.simpat.2013.08.003

ments can be added or removed according to the features of the target domain. In addition, this BJCS problem focuses on the shop floor which has the feature of a static job-shop. Therefore, all materials for the current processing step already exist in the stock. The objective of the BJCS is for the performance evaluation measures to guarantee a certain level, as determined by the manager. The starving time of the following processing step, the job change loss time of the current processing step and the WIP level between the current and the following processing step are the performance evaluation measures of the BJCS problem. A summary of the BJCS problem is depicted in Fig. 1.

The BJCS problem is a production scheduling problem. In general, to solve a production scheduling problem, there are two major approaches – a mathematical approach and a simulation-based approach. In the mathematical approach, there is a wealth of research about scheduling using Linear Programming, Integer Programming, Mixed Integer Programming and other methods, but they solve the problem under certain assumptions of problem environments, making them difficult to apply as a solution in the real-world. The BJCS problem can be represented by the Integer Programming formulation, but it cannot apply the current status and it takes a long time to make a schedule of the proper size for use in the real-world. The details of the Integer Programming formulation are explained in Appendix B. On the other hand, a simulation is free from those limits. With simulation-based scheduling, discrete event simulation methodologies are mainly used. There are three simulation modeling approaches for discrete event simulations: the event, activity and state approaches [13]. Among these approaches, event-based modeling [21] and simulation are largely used for production scheduling. In previous research on backward scheduling using event-based simulation, the release plan was initially created by a backward simulation. A forward simulation was then executed using the release plan [1–4]. However, this approach is limited by its inability to follow the sequence of operations as predicted by the backward simulation [2]. The event-based simulation is executed by means of a forward time-advance in the forward simulation and by a backward time-advance in the backward simulation. Hence, the backward scheduling problem cannot be solved at once and is therefore divided into release planning and production scheduling problems. This lowers the capability level. This is identical to an activity-based simulation. A state-based simulation is executed by a state change rather than a time-advance. In research on production scheduling with a state-based simulation, the modeling of a job shop problem with Discrete Event System Specification (DEVS) [15] deals only with the concept of a simple job shop, and Finite State Automata is used in the scheduling of cluster tools [16]. However, the backward scheduling problem is not considered to be serious.

The production system, one of the state-based simulation approaches, is a problem-solving methodology that uses the productions (i.e., condition-action rules) [5]. Its concept is explained in terms of states and operators. The production system can easily represent the thinking process of a human [7]. Therefore, it has been widely researched in the Artificial Intelligence (AI) domain. It is applied to expert systems in various domains, such as agriculture [8], finance [9] and mechanical engineering [10]. It is also used in the manufacturing domain. Expert systems for Flexible Manufacturing System (FMS) scheduling and planning are developed with the concept of the production system [14,17]. However, this system is specific to the FMS domain, which is focused on operation sequencing and does not consider job changes. This makes it difficult to apply to BJCS problems.

In this study, a simulation using a production system is proposed to solve the BJCS problem. The BJCS problem is described in terms of the state and operator. The state is defined in terms of the job and the machine state and the operator is described as executing the loading of the job to the machine. Solving this problem is identical to the selection and execution of the operator to change the state so as to achieve the goal state. That is, solving the BJCS problem involves a transition

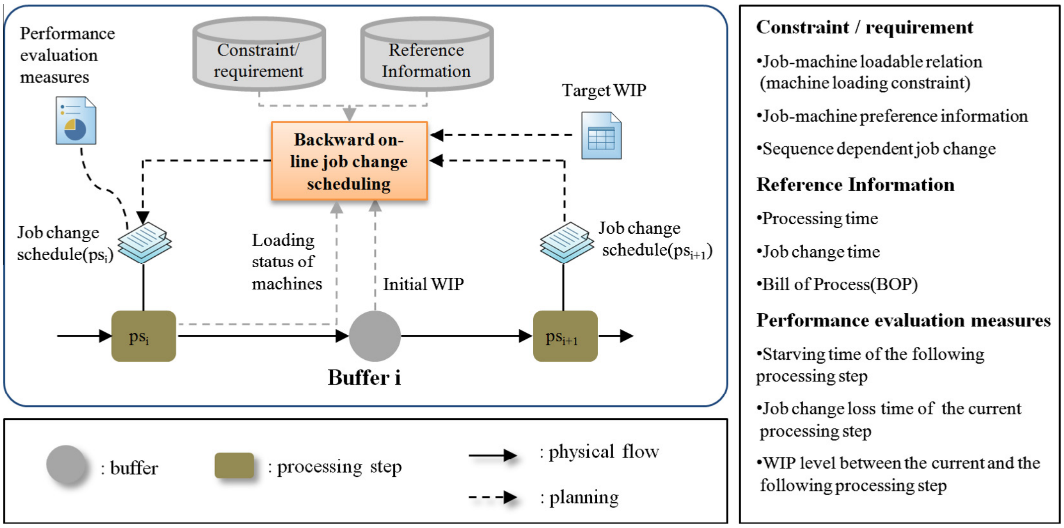

Fig. 1. Backward on-line job change scheduling problem.