Available online at www.sciencedirect.com

SCIENCE DIRECT®

SIMULATION MODELLING PRACTICE AND THEORY

ELSEVIER

Simulation Modelling Practice and Theory 13 (2005) 389–406

www.elsevier.com/locate/simpat

A multi-agent reinforcement learning approach to obtaining dynamic control policies for stochastic lot scheduling problem

Carlos D. Paternina-Arboleda [a,*], Tapas K. Das [b]

[a] *Department of Industrial Engineering, Universidad del Norte, Km 5 via Puerto Colombia, Barranquilla, Colombia*

[b] *Department of Industrial and Management Systems Engineering, University of South Florida, Tampa, FL 33620, USA*

Received 26 March 2003; received in revised form 15 October 2004; accepted 10 December 2004
Available online 22 January 2005

Abstract

This paper presents a methodology that, for the problem of scheduling of a single server on multiple products, finds a dynamic control policy via intelligent agents. The dynamic (state dependent) policy optimizes a cost function based on the WIP inventory, the backorder penalty costs and the setup costs, while meeting the productivity constraints for the products. The methodology uses a simulation optimization technique called Reinforcement Learning (RL) and was tested on a stochastic lot-scheduling problem (SELSP) having a state–action space of size 1.8×10^7. The dynamic policies obtained through the RL-based approach outperformed various cyclic policies. The RL approach was implemented via a multi-agent control architecture where a decision agent was assigned to each of the products. A Neural Network based approach (least mean square (LMS) algorithm) was used to approximate the reinforcement value function during the implementation of the RL-based methodology. Finally, the dynamic control policy over the large state space was extracted from the reinforcement values using a commercially available tree classifier tool.

* Corresponding author. Tel.: +57 300 800 1327; fax: +57 359 8852.
E-mail addresses: cpaterni@uninorte.edu.co (C.D. Paternina-Arboleda), das@eng.usf.edu (T.K. Das).

1569-190X/$ - see front matter
doi:10.1016/j.simpat.2004.12.003

Keywords: SELSP; Scheduling; Reinforcement learning; Simulation optimization

1. Introduction

The problem of economic production scheduling of multiple items on a single facility under a random production environment is generally referred to as the stochastic lot scheduling problem (SELSP). Other specific features of this problem include random sequence dependent production change-over times and costs, random asymmetric demand arrival patterns for the items, inventory holding costs, backordering/lost customer costs, and, perhaps, specified service level constraint for the items. Industrial applications where SELSP can be found are auto manufacturing (stamping, forging, painting, etc.), paper mills, and bottling operations.

The important question associated with controlling a SELSP is "how to dynamically switch production from one item to another depending on the buffer levels and demand characteristics of the current item being produced as well as the others so as to minimize the overall cost per unit time while, perhaps, satisfying the service level constraints for this items?" An excellent review of the SELSP literature can be found in Sox et al. [26]. The SELSP literature can be primarily classified into two classes: cyclic scheduling and dynamic scheduling. As far as the solution approach is concerned, there are two primary classifications in the literature: discrete time control [3,9,16,19,25] and queuing-based control [1,8,21]. The latter approach is again classified into three categories: the pooling systems approach, heavy traffic approximation, and simulation-based approach. A good application using simulation agents can be seen in Gelenbe et al. [10]. The approach adopted in this paper is a combination of discrete simulation and reinforcement learning, of which the latter is derived from dynamic programming and stochastic approximation. Our approach essentially consists of the following steps:

(1) Develop a mathematical model for the SELSP problem.
(2) Develop a multi-agent RL algorithm.
(3) Develop a simulation model of the system.
(4) Run the system simulation (with service level constraints built into it) and apply the RL algorithm during the simulation to arrive at a dynamic scheduling policy.

The multi-agent system uses reinforcement learning algorithms to perform unsupervised learning. An excellent review of reinforcement learning agents can be seen in [18,22,27]. We give a brief introduction to reinforcement learning in the next section.

2. Reinforcement learning

Reinforcement learning (RL) is a way of teaching agents (decision-makers) near-optimal control policies. This is accomplished by assigning rewards and punishments