G Model
CIRPJ-406; No. of Pages 10

ARTICLE IN PRESS

CIRP Journal of Manufacturing Science and Technology xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

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

# Manufacturing systems: Using agents with local intelligence to maximize factory profit

Miri Weiss-Cohen [a], Michael Mitnovizky [b], Moshe Shpitalni (1) [b,*]

[a] Department of Software Engineering, Braude College of Engineering, Karmiel, Israel
[b] Department of Mechanical Engineering, Technion – Israel Institute of Technology, Haifa, Israel

ARTICLE INFO

*Article history:*
Available online xxx

*Keywords:*
Stochastic manufacturing systems
Dynamic scheduling
Multi-agent systems

ABSTRACT

Many of the problems of manufacturing systems, including scheduling, are NP-hard and therefore cannot be solved and optimized in real time as required in the real world. This paper investigates the possibility of controlling manufacturing systems while maximizing the enterprise profit in a stochastic and dynamically varying environment. In this research, we developed a multi-agents system that operates on the factory floor level and directly controls the manufacturing system. The paper examines the proposed multi-agents system, demonstrates its feasibility by simulation under various experimental settings, presents important findings and excellent results, and draws conclusions.



## Introduction

The problem of profit maximization in a limited resource environment is a main concern for any enterprise or service provider. In this paper, we consider a productive enterprise whose main objective is to maximize profits by maximizing production while using the enterprise's own resources in the best possible way and minimizing penalties for tardiness.

Only in the simplest case when all the production system parameters are constant and not expected to change during the process can global optimization be obtained by advanced and off-line computation of the optimal schedule for resource allocation throughout the production process. Such an optimal schedule describes what every machine (resource) should do at any given time. As noted, this approach is not adequate for the general case, and various approaches have been suggested.

The job shop problem (JSP) has the well-earned reputation of being one of the most computationally complex combinatorial problems considered to date. In computational complexity theory, the JSP is classified as an NP-hard problem [1–3]. The JSP has several variations. The classic job shop problem can be formulated as follows: Given are a finite set of $n$ jobs, where each job consists of a chain of operations and a finite set of $m$ machines. Each job should be processed through the machines in a particular order, while each machine can handle at most one operation at a time. In addition, each operation needs to be processed during an uninterrupted period of a given length of time on a given machine [4]. In the dynamic job shop problem, however, jobs can arrive at some known (deterministic) or unknown (stochastic) future time. The goal is to find the sequence of $j$ jobs to be completed on $m$ machines such that a given objective function is minimized. A number of different objectives can be minimized, among them the total elapsed time to complete all jobs, the weighted mean completion time of a job, the weighted mean tardiness (i.e., the system's ability to meet given deadlines) and the weighted mean idle time of the factory's resources. The flexible job shop problem (FJSP) [5], an extension of the classical job-shop scheduling problem, allows an operation to be processed by any machine in a given set. The problem is to assign each operation to a machine and to order the operations on the machines to minimize the maximal completion time (make span) of all operations. The flexible job-shop scheduling problem can be formulated as follows: $n$ jobs are to be scheduled on $m$ machines. Each job $i$ represents $n_i$ ordered operations. Each operation $k$ of job $i$ (noted as $O_{ik}$) can be executed on one machine $j$, selected from a set of available machines that can process $O_{ik}$, called $A_{ik}$ [6]. The execution will occupy the chosen machine for $t_{ikj}$ time units until the operation is completed. The objective of the FJSP problem is to schedule the operations to be executed by the machines in a way that minimizes the make span, subject to the following constraints: the operation sequence for each job is prescribed and each machine can process only one operation at a time.

Manufacturing systems are often subject to random events that may disturb their working process, among them machine failure, operator unavailability, out-of-stock condition, change in availability date and uncertain processing time. The stochastic job shop

* Corresponding author.
*E-mail address:* shefi@tx.technion.ac.il (M. Shpitalni).

http://dx.doi.org/10.1016/j.cirpj.2016.11.005
1755-5817/© 2016 CIRP.

problem (SJSP) considers such events and allows one or more of the problem parameters to be random [7,8]. The literature has considered the job shop problem in its different variations, with countless papers proposing numerous approaches. This paper proposes a dynamic [9,10], flexible and stochastic variant of the job shop problem that offers better quality and cost effective manufacturing systems.

In recent years, researchers have examined the Multi Agent Systems (MAS) approach for manufacturing and job shop handling. Agents can be defined as autonomous control software systems that communicate with each other [11]. The MAS approach is one of the fastest growing domains in agent-oriented technology that deals with autonomous decision modeling [12–15]. Shen [16], Leitão [6,17] and Hees [18] provide extended and detailed state of the art reviews. Barbati et al. [19] survey and classify MAS according to their contribution and means of coordination while analyzing different system layouts and architectures. The authors supply a detailed segregation of the protocols and their effectiveness in productivity.

Rajabinasab and Mansour [20] propose an agent-based system that deals with a dynamic, flexible job shop problem with the possibility of some stochastic occurrences, such as processing time uncertainty and machine breakdowns. The proposed method performs well in the given scenarios, but the paper does not examine the method's performance in a variety of process-time uncertainty levels. Sudo [21] proposes MAS for assembly. Subramaniam et al. [22] examine dynamic and flexible manufacturing systems, but they do not consider the effects of uncertainties (except machine breakdowns) and the factory's global utilization on the proposed algorithm's performance. Wei et al. [23] propose a method that deals with the dynamic, flexible job shop problem. The method is a CNP-based negotiation that takes place between the jobs and the machines. The quality of the bids is evaluated based on various dispatching rules. Bruccoleri [24], Yeung [25] and Kalihara [7] study the negotiation rules that affect the quality of the agents' decision-making while detailing simulations and experiments of a networked environment.

Bio-inspired methodologies for solving MS with MAS have been researched in [17,26–30], contributing to understanding the communication methods used by ant colonies (agents). Putnic [7,31] examines the use of virtual environments by teams to handle dynamic reconfigurabilty.

The use of Genetic Algorithms for handling dynamic job shop is discussed in [32–35], while [36] discusses selection of a candidate partner/agent for conveying autonomous decisions, handling unexpected events and machine selection. MAS coalition [30] strategy for agent cooperation in a virtual enterprise was demonstrated in [37,38].

Valckenaers et al. [39] introduce holonic manufacturing systems. ADACOR [40] and ADACOR2 [41] are examples of systems developed using MAS and used for adaptive and reactive production control. The system integrated particle swarm or genetic algorithm for unexpected resource allocation.

After reviewing the work done in the field so far, this work proposes a MAAS (multiple autonomous agent system) approach to the decision-making process in a dynamic, flexible and stochastic job shop problem. The approach yields qualitative results both in system performance and in the computational complexity of the system. In addition, we recognize the need to analyze over-capacitated factories where more jobs arrive than the factory is able to process. Since such a contingency is a viable option in a real life factory, the approach proposed in this research can efficiently handle either over-capacitated or under-capacitated factories.

The current paper (a comprehensive extension of our previous paper [42]) investigates the possibility of controlling manufacturing systems while maximizing the enterprise profit in a stochastic environment. We developed a multi-agents system that operates on the factory floor level and directly controls the manufacturing system. In the paper, we describe and examine the proposed system and demonstrate its feasibility by simulation under various experimental settings. We then analyze the results and draw conclusions.

## Aim, approach and system definition

The aim of this study is to investigate the possibility of using agents with local intelligence to maximize factory profit by on-line scheduling of stochastic manufacturing systems/job shops with limited resource capabilities and under different operating conditions and control parameters. Note that on-line scheduling implies that when a new job arrives, a decision about that job must be made before the next job arrives. To implement this investigation by simulation we constructed a simulation system (shown in Fig. 1) with the following specifications and assumptions:

- Jobs arrive stochastically (Poisson distribution). We chose the Poisson distribution as it is the most common distribution for describing stochastic event occurrences defined by an average number of occurrences per time unit.
- The system can handle a limited number (3) of job types at a time. The factory has the capacity to perform five different tasks.

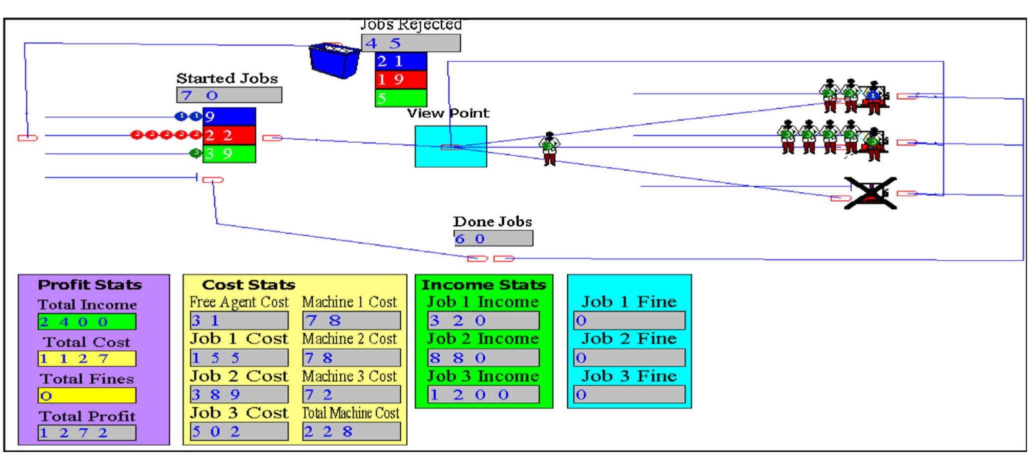


**Fig. 1.** A snapshot of the simulation system [43].