Expert Systems With Applications 45 (2016) 71–84

Contents lists available at ScienceDirect

Expert Systems With Applications

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

An object-oriented approach for multi-objective flexible job-shop scheduling problem

Vahit Kaplanoğlu*

University of Gaziantep, Department of Industrial Engineering, Gaziantep, Turkey

ARTICLE INFO

Keywords:
Object-oriented manufacturing control
Object-oriented design
Multi-objective flexible job-shop scheduling
Simulated annealing algorithm

ABSTRACT

Flexible manufacturing systems are very complex to control and it is difficult to generate controlling systems for this problem domain. Flexible job-shop scheduling problem (FJSP) is one of the instances in this domain. It is a problem which inherits the job-shop scheduling problem (JSP) characteristics. FJSP has additional routing sub-problem in addition to JSP. In routing sub-problem each operation is assigned to a machine out of a set of capable machines. In scheduling sub-problem the sequence of assigned operations is obtained while optimizing the objective function(s). In this paper an object-oriented (OO) approach is presented for multi-objective FJSP along with simulated annealing optimization algorithm. Solution approaches in the literature generally use two-string encoding scheme to represent this problem. However, OO analysis, design and programming methodology help to present this problem on a single encoding scheme effectively which result in a practical integration of the problem solution to manufacturing control systems where OO paradigm is frequently used. OO design of FJSP is achieved by using UML class diagram and this design reduces the problem encoding to a single data structure where operation object of FJSP could hold its data about alternative machines in its own data structure hierarchically. Many-to-many associations between operations and machines are transformed into two one-to-many associations by inserting a new class between them. Minimization of the following three objective functions are considered in this paper: maximum completion time, workload of the most loaded machine and total workload of all machines. Some benchmark sets are run in order to show the effectiveness of the proposed approach. It is proved that using OO approach for multi-objective FJSP contributes to not only building effective manufacturing control systems but also achieving effective solutions.

1. Introduction

Scheduling problems play a very important role in many industrial systems. It attracts the attention of many researchers (Baykasoğlu, Özbakir, & Sönmez, 2004; Brucker & Schlie, 1990; Dai, Tang, Giret, Salido, & Li, 2013; Garey, Johnson, & Sethi, 1976; Liu, Yin, & Gu, 2014; Lu, Wang, & Wang, 2014; Wang & Liu, 2014; Xia & Wu, 2005; Xingong & Yong, 2015; Zhao, Hsu, Cheng, Yin, & Wu, 2014). Most of the scheduling problems are complex combinatorial optimization problems and very difficult to solve (Xia & Wu, 2005; Zeballos, 2010). One of the complex scheduling problems is JSP. In a job shop, each job can have a different processing route through the system, and hence the scheduling problem becomes highly complex (Fahmy, Balakrishnan, & ElMekkawy, 2011). It is a branch of production scheduling. It is a well-known NP-hard problem (Garey et al., 1976). FJSP is a problem which inherits the JSP characteristics. Due to its importance in the industry, FJSP has attracted the attention of many researchers. FJSP has additional routing sub-problem in addition to JSP. In routing sub-problem each operation is assigned to a machine out of a set of capable machines. In scheduling sub-problem the sequence of assigned operations is obtained while optimizing the objective function(s). Therefore, FJSP presents two difficulties; operation assignment and finding optimal operation schedule (Xia & Wu, 2005).

Although an optimal solution algorithm for the classical JSP has not been developed yet, there is a trend in the research community to model and solve much more complex version of the JSP. The need to model and solve FJSP has mainly emerged due to the fact that modern machine tools have considerable amount of overlapping capabilities (Baykasoğlu et al., 2004). The number of studies for FJSP seems more than the ones for JSP. Brucker and Schlie (1990) were among the first to address FJSP. They developed a polynomial algorithm for solving the FJSP with two jobs. Brandimarte (1993) used a hierarchical algorithm for the FJSP based on the tabu-search meta-heuristic. Dauzere-Peres and Paulli (1997) introduced an extended version of the disjunctive graph model that is able to take into account the fact that operations have to be assigned to machines for FJSP. Mastrolilli and Gambardella (2002) introduced two

* Tel: +90 3423172610; fax: +90 3423604383.
E-mail address: kaplanoglu@gantep.edu.tr, vahitkaplanoglu@hotmail.com

http://dx.doi.org/10.1016/j.eswa.2015.09.050
0957-4174/

neighborhood functions along with a tabu search procedure for FJSP.

FJSP is also researched when there is more than one objective function. Baykasoğlu et al. (2004) proposed a linguistic-based meta-heuristic modeling and solution approach for multi-objective FJSP. Xia and Wu (2005) proposed an effective hybrid optimization approach for multi-objective FJSP. Zhang, Shao, Li, and Gao (2009) proposed an effective hybrid particle swarm optimization algorithm for multi-objective FJSP. Wang, Zhou, Xu, and Liu (2012) proposed an enhanced pareto-based artificial bee colony algorithm to solve the multi-objective FJSP effectively. Li, Pan, and Xie (2012) proposed a hybrid shuffled frog-leaping algorithm for solution of the problem. Pour and Ghasemishabankareh (2013) proposed a hybrid genetic and simulated annealing (SA) algorithms for multi-objective FJSP. Wang, Wang, and Liu (2013) proposed an effective pareto-based distribution algorithm for the problem.

In the real world manufacturing systems, the analysis of the scheduling problem and developing algorithms and procedures are only part of the story. The procedure has to be embedded in a system that enables the decision maker to actually use it. The system has to be integrated into the information system of the organization, which can be a formidable task (Pinedo, 2012). Integration of design model with process and scheduling information in real-time is necessary in order to increase product quality, reduce the cost, and shorten the product manufacturing cycle (Šormaz, Arumugam, Harihara, Patel, & Neerukonda, 2010). System designers need orientation of the developed scheduling algorithms and procedures to their real manufacturing information system because scheduling system have many links with other different systems in an organization. Sometimes it takes much more time to adapt the scheduling algorithm to manufacturing information system than developing the algorithm or procedure. The integration of the scheduling systems to the ongoing information system is not a trivial task for the computer programmers.

The design of software systems is a very complex activity as it is strongly influenced by the quality required in the final products (Vazquez, Andres Diaz Pace, & Campo, 2014). OO analysis, design and programming has been developing for 30 years so far, however, it is in the last decade that it received fast growing and developing methodologies by using speed and processing capability advances in the development of high complexity software (Moreno Reyna et al., 2012). OO programming is a method of programming which is highly structured. It includes modularity and rational order of software structures (Shixin & Mengguang, 2000). OO technologies are most commonly used computer programming technology in today's manufacturing controlling information technologies (Fernandez de Canete, Garcia-Cerezo, Garcia-Moral, Del Saz, & Ochoa, 2013; Kwon, 2011; Liao & Hu, 2011; Lin, Tseng, & Tsai, 2003). Therefore the researchers in machine scheduling area should consider the appropriateness of their solution to modern information technologies.

The application of information technology (IT) is a significant enabler of opportunities (Ngai, Peng, Alexander, & Moon, 2014). Today's database management systems (DBMS) which are used in manufacturing control systems are generally built on OO programming principles. They use structured query language (SQL) while doing data recording and retrieval. Oracle, MySql, Sybase, MS SQL Server are the examples DBMS's which are commonly used in production control systems. DBMSs are used in connection with scheduling algorithms. Therefore there is a very close relationship among OO programming, machine scheduling and DBMS.

Job and machine related data could be processed and recorded to a DBMS based on OO programming methodology. These data could be encapsulated in objects and could be used on scheduling and data record/retrieval purposes. Fig. 1 represents some basic attributes and behaviors of typical job and machine class in a manufacturing information system.

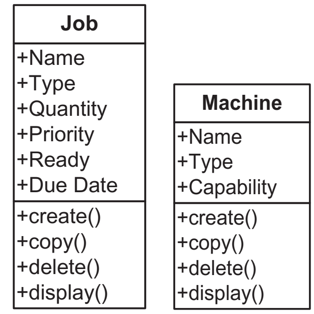

Fig. 1. Job and machine class definition (Adapted from Pinedo, 2012).

Static data about a job could be held on a job object. For example *operation list, associated weight, material handling time, shipping cost* and *due date* are some static data about a particular job object. Some basic behaviors for a job could be *create, copy, delete* and *display*. Static data about machine could also be encapsulated on machine object such as; *list of capabilities, speed, scheduled maintenance time* etc. On the other hand there are some static data about both jobs and machines. For example *sequence dependent set up time* could be considered as static data for both machines and jobs. These data types are classified as static because they do not depend on the generated schedule.

In FJSP, the operations of jobs have alternative machines to be processed. The data about alternative machine is depending on both machine and job operations data. Therefore FJSP includes an additional machine-job association. The researches in the literature generally divide this association into two sub-problems. The first sub-problem is the assignment of operations to a machine from several available machines. The second sub-problem is sequencing of all operations on each machine (Shao, Liu, Liu, & Zhang, 2013).

Although OO programming methodology and FJSP problem are not new for the researchers, the design of FJSP by OO paradigm seems novel for both information science practitioners and algorithm developers. It seems there is a gap between software developers of manufacturing control systems and algorithm developers of scheduling theory which was previously stated as formidable task to fill it (Pinedo, 2012). This research is supposed to fill this gap and be a bridge between the software developers and algorithm developers.

In this paper, the association between machine and operations is represented by defining a new class. By doing so multi-objective FJSP two-phase solution representation could be reduced to one which result in a smart and effective solution structure. Presenting the system entities and their associations on software objects is supposed to be reducing the system complexity without losing the solution quality. SA algorithm is used for the proposed OO approach.

The rest of the paper is organized as follows: Section 2 presents problem definition for multi-objective FJSP. Section 3 presents the OO design of the FJSP. Section 4 presents the application of SA algorithm to the proposed approach. Section 5 presents some test-bed problems and their solutions on the proposed approach and Section 6 summarizes the paper.

2. Problem definition

FJSP is one of the most commonly encountered production scheduling problem in manufacturing enterprises. FJSP is a generalization of the classical JSP where operations are allowed to be processed on any among a set of available machines. Then, FJSP is more difficult than the classical JSP, since it introduces a further decision level beside the sequencing one, i.e., the job routes. To determine the job routes means to decide, for each operation, what machine must