Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 62 (2017) 147 – 152

www.elsevier.com/locate/procedia

10th CIRP Conference on Intelligent Computation in Manufacturing Engineering - CIRP ICME '16

Hybrid flow shop management: multi objective optimisation

Lorenzo Lassig, Fabio Mazzer, Marino Nicolich*, Carlo Poloni

Department of Engineering and Architecture, University of Trieste, via A.Valerio 10, 34127 Trieste, Italy.

* Corresponding author. Tel.: +39 040 558 3825; fax: +0-000-000-0000. *E-mail address:* nicolich@units.it

Abstract

The objectives of the optimisation management that considers both assembly and production Hybrid Flow-Shops lines are several: optimal numbers of machines, shifts, product priority and time-period scheduling. The approach uses a combination of DES software with an MDO (Multidisciplinary Design Optimization) software.. The DES model is optimized following the process: 1) A Multi-objective Genetic Algorithm cycle generates a set of optimal solutions; 2) A Clustering cycle groups the solutions into different sets; 3) The selection of the preferred solution via post-processing; 4) A mono-objective optimisation of each cluster; and 5) creation of weekly scheduling with optimal results. Finally, application and results are described and discussed.

Peer-review under responsibility of the scientific committee of the 10th CIRP Conference on Intelligent Computation in Manufacturing Engineering

Keywords: Optimisation; simulation; management; production; flow;

1. Introduction

Currently, management and investigation of manufacturing systems is a very important aspect for any industry. The inaccurate or defective definition of manufacturing system can affect the real capacity of a company to stay in the market. For this reason in the last 20 years, several techniques to support the decision making have been developed in order to evaluate the manufacturing production strategies and the system itself.

The main purpose of production management is generally the scheduling of operations to maximize/minimize one or more performances. The literature shows several methods to solve problems of this type such as enumerative, heuristic and meta-heuristics methods.

In the first two cases, i.e. enumerative and heuristics, scientific papers examine a wide scenarios' spectrum [1], [2], [3]. In recent years, researchers have started to unite the heuristic method with iterative methods to find the optimal solution. This method is called metaheuristic method. The basic idea is to insert a random component inside a heuristic method. The repetition of such processes gives a better result than a purely heuristic one. The Multi-Objective Flow shop Scheduling Problem (MOFSP) [4] has been studied by researches who considered two or more objectives, such as makespan, flowtime, earliness, tardiness and idle time. Many types of optimisation algorithm are used, for example: *Simulated Annealing-SA, Tabu Search-TS, Genetic Algorithms-GA, Ant Colony Optimisation-ACO, Neural Networks-NN, etc*. [3], [5], [6], [7], [8], [9]. The latter methods lend themselves to connect with production systems simulation software, the Discrete Event Simulation tools (DES) that can simulate a wide range of activities and situations that may occur within a production system [9], even if significant computational time is needed [10]. Present work considers both assembly and process Hybrid Flow-Shop lines as case study. There are many definitions of HFS [11]. In general, HFS is defined as a set of n jobs to be processed in m stages. In our definition of HFS, the following assumptions are made:

- The number of processing stages m is at least 2,
- Each stage has $k \geq 1$ machines in parallel and in at least one of the stages has $k > 1$,
- All jobs are processed following the same production flow: stage 1, stage 2,. . . , stage m. A job might skip any number of stages, assumed it is processed in at least one of them.
- Some jobs can have loops in the production system

The HFS problem is considerably more complex and intractable than conventional flow shop; it belongs to an N-P hard case. Due to these difficulties a precise method has not been found that solves every case in a reasonable time. In [12] the authors define the minimization of *Earliness* and maximization of *Tardiness* as objectives in order to meet the due date. On the other hand, [11] considers various methods

Peer-review under responsibility of the scientific committee of the 10th CIRP Conference on Intelligent Computation in Manufacturing Engineering

doi:10.1016/j.procir.2016.06.007

and objectives that are used for HFS: not only max *Earliness* and min *Tardiness*, but also minimize *makespan*, flow time, inventory cost, etc. Even if 60% of the cases consider minimizing the *makespan*, other objectives may be more important depending on the situation. Considering also the constraints and assumptions of real life cases, it is difficult that real HFS problems exactly match a model studied in literature. In many cases, for the best results the objective is not simply a mono-objective but multi-objective. Production scheduling problems are multi-objective by nature, and in most of the cases, these objectives are in conflict amongst themselves. The authors [13] consider solving a HFS scheduling problem by minimizing the *makespan* and inventory cost through a multi-objective-hybrid-metaheuristic approach, which combine genetic algorithm and variable neighbourhood search. They declare that the approach is robust, fast and simply structured.

This work faces also the inverse problem related to the definition of the production plant layout together with the production scheduling needed to meet prescribed production capabilities.

A typical approach to the design of the production system is therefore divided into 2 phases:

- 1. setting the general features of the production system
- 2. verifying if the production system can achieve a particular level of performances

Repeat phase 1 and phase 2 until the production system achieves the established performance avoiding undesired behaviours in the production system such as bottlenecks. DES tools can be used in the second phase to better understand the behaviour of the production system.

The method discussed here focuses directly on the performances and defines automatically the general features of the production system so that the performances can be achieved; in this method DES tool turns from a validation tool to a design tool.

In this case, the purpose of the optimisation is to find out the optimal general features: the right set of input variables so that the output achieves our goals. Thus, implying the idea that the performances are our objective and using optimisation as a research tool, the basis of the here discussed method is fixed. Therefore, this way of designing a production system, in comparison with the first one, is faster and gives the user more information.

The systems discussed here are two Hybrid Flow Shop with different objectives and constraints. Both of them use the same approach (chapter 2) to achieve their objectives. One is an assembly and testing plant of large mechanical products, which is described with its results in chapter 3, whereas the second is a processing production plant, described with its results in chapter 4. Finally, a conclusion is made in chapter 5.

2. Computing

In general, the mathematical behaviours of a production system are strongly non-linear and governed by numerous variables, whereas the performances, typically, are two or more. These two facts provide us with guidelines to design the computing phase that is directly linked with the choice of optimisation algorithm and the optimisation cycles.

The approach is to connect an optimisation cycle to a model that simulates the production using a discrete event simulation (DES) with WITNESS 14. The concept of the approach used to solve the problem is inserting variable inputs into a model that simulates the process flow. This model gives an output, which is compared with the main objective. This information is sent to the Genetic Algorithm (Esteco MODE-FRONTIER). Then, through the genetic algorithm, the inputs are changed accordingly (trying to improve the output) and inserted back into the model. The model will give another output and the loop is repeated. The number of iterations (loops) is defined at the beginning by the user. It is important to define the right number of iterations to have a sufficient number (although not too many) of optimal solution sets (fig.1.) The task of the genetic algorithm is changing inputs producing output from DES simulator that will converge to the objectives, not giving a single solution but a set of optimal solutions [14].

To increase the robustness of the genetic algorithm, it needs information on how the system works, for example, most critical variables and regions of optimal solution. The optimization algorithm is initialized by a set of configurations obtained by a DOE (Design of Experiment) technique.

A non-linear model can be well managed by a stochastic algorithm while gradient based methods cannot be used both because of the multi-objective natter of the problem as well as because the DES output cannot be derived. After the first cycle of multi-objective optimisation, with the help of the Pareto Front a set of optimal solutions are given. Each of these solutions are characterized by the fact that one objective cannot improve without worsening other objectives [15]. Since the optimal solutions can be very different among themselves as well as numerous, it is advisable to apply a clustering method to better understand the different features of the optimal solutions. Pareto Front is rearranged in clusters through a clustering method. For each cluster a second round of optimisation that maximizes the main objective is done. In the second round of optimisation, a mono-objective Simplex algorithm (greater accuracy), that needs a narrow set of optimal solutions (because less robust), is used to obtain a small set of optimal solutions (fig. 2) [16].

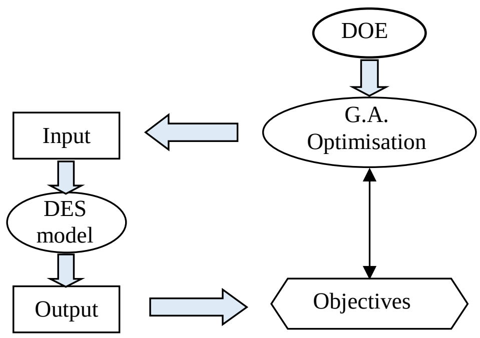

Fig. 1. Optimisation cycle

In the production test case (chapter 4), the same method of optimisation was used, but, in addition two decision support methods are used: Self-Organizing Map (SOM) [17] to project the high-dimensional data onto a two-dimensional map used to identify interest clusters and MCDM (Multi Criteria Decision Making) to choose the best solution considering the attributes and their preferences [13].