Solving a Discrete Lot Sizing and Scheduling Problem with Unrelated Parallel Machines and Sequence Dependent Setup Using a Generic Decision Support Tool

Nathalie Klement, Cristóvão Silva, Olivier Gibaru

Solving a Discrete Lot Sizing and Scheduling Problem with Unrelated Parallel Machines and Sequence Dependent Setup Using a Generic Decision Support Tool

Nathalie Klement[1], Cristóvão Silva[2], and Olivier Gibaru[1]

[1] LSIS CNRS UMR 7296 (équipe INSM). École Nationale Supérieure d'Arts et Métiers. 8, boulevard Louis XIV - 59046 Lille Cedex, France
[2] CEMMPRE, Department of Mechanical Engineering, University of Coimbra. Rua Luís Reis Santos, 3030-788 Coimbra, Portugal
nathalie.klement@ensam.eu; cristovao.silva@dem.uc.pt

Abstract. In any manufacturing systems, planning and scheduling are not intuitive. Some dedicate tools may exist to help some specific companies to daily plan and assign their activities. Our purpose is to develop a generic decision support tool to solve any planning or scheduling problems. To do so, we use a hybridization between a metaheuristic and a list algorithm. The metaheuristic is generic to any studied problems. The list algorithm needs to be specific to the considered problem. The use of our tool needs a minimum work development. In this paper, our proposal is illustrated by the case study of a textile company which intends to schedule its production, by assigning it resources and dates. This problem can be seen as a Discrete Lot Sizing and scheduling Problem (DLSP).

Keywords: Lot Sizing and Scheduling Problem; Discrete; Sequence Dependent Setup; Metaheuristic; List algorithm; Hybridization

1 Introduction

One key point of the organization of a manufacturing system is the planning of its production. Each enterprise has several customers, with their specificity and delivery dates. The enterprise has a given quantity of resources (human and/or material, for example machines) that it has to occupy in order to have the best possible return and to satisfy as best as possible customer demand.

When an enterprise is selling only one type of product, planning may be easy. But when the number of product types, customers or resources is increased, planning becomes more complex. Finding an optimal solution may be difficult.

Enterprise Resource Planning (ERP) is an informatic tool which can manage every departments in an enterprise. Information about departments are stored in an unique database, all data are shared, that avoid unnecessary re-entry of information. When a customer makes a new order, ERP is able to check the

stock level, to buy raw material and to launch fabrication order if applicable. But ERP is used at a macro level. According to the demands, it plans month by month or week by week the activity of the company. From our knowledge, it does not exist any generic tool that can interpret ERP fabrication order to make an assignment and scheduling used on the shop floor. There are some software packages specifically dedicated to scheduling, but there are dedicated to a specific situation.

Usually, one experienced worker is making the detailed scheduling of fabrication order every day, and assigns them the required resources. But what could happen if this worker is missing? The aim of our work, is to propose a tool which solves scheduling and assignment problems. This tool has been developed in a generic way, so it should be used to solve many problems. In this paper, this tool is applied to a textile problem, previously studied in 2006 [1].

2 Analysis of the Studied Problem

The considered company is producing acrylic fiber used in the textile industry. These fibers are made following three steps. Dope preparation: a polymer is dissolved in an organic solvent to make the dope. Spinning: dope is going through a spinneret (a tool which is a metal plate with holes from different diameters) to obtain the synthetic filaments. After this step the fiber is called "tow". Cutting and packing: the tow obtained in the previous step can suffer two different types of operation: it can simply be packed before being send to the warehouse or, it can be cut in small segments, originating a new kind of fiber called raw, which is also packed before expedition.

Our study focuses on the spinning area. There are 10 non-identical spinning machines on the shop floor. Machines are dedicated, all products can not be produced on all machines. A compatibility machine-product is defined. All machines do not have the same production rate: some of them can produce more tons of product per hour than others.

Fibers can be made from different diameters, within three colors: shiny, mat and black. There are two types of fiber: tow and raw. In total, the company can make 60 different products. A change of color may induce the stop of the machine for its cleaning. Transition shiny-shiny and shiny-mat/black does not induce a stop, but transition mat/black-shiny does. If there is a setup (change of tool), change of color is made in the meantime. A tool can produce fibers from different diameters by changing the used tensity during the production. Moreover, each tool has a lifetime, from 8 to 45 days. When its lifetime is over, a new tool has to be used. A setup lasts 2 hours. Lifetime of tools depends on the type and the color of the product, and depends on the used machine. Constraints have to be respected:

- All or nothing: if a product is planned, production lasts 24 hours even if the needed quantity is less. Over-quantity is stocked and used next month, it will be deduced from the next orders.
- Possible setup between two products:

- Setup between two products,
- Cleaning of the machine (transition mat-shiny or black-shiny),
- Setup if lifetime is over.

The company wants to plan products within a planning horizon of one month, by periods of 24 hours. Customers orders are analyzed to make a production plan. Our objective is to schedule these fabrication orders on the different machines and the different periods during the considered month. The result will be a schedule over 28 days by periods of 24 hours. Two criteria have to be minimized:

– The quantity of products not produced during the considered month. This quantity would be produced next month,
– The number of setups.

3 State of the Art

This problem can be seen as a Discrete Lot Sizing and scheduling Problem (DLSP). The fundamental assumption of the DLSP is the so-called "all-or-nothing" production: only one item may be produced per period, and, if so, production uses the full capacity [2]. Some papers refer DLSP as a Dynamic Lot Sizing Problem, which is not the problem considered in this paper.

The first resolution of this kind of problem seems to be done in 1971 [3]. The authors developed an efficient algorithm to solve a multi-item scheduling problem with identical parallel machines. This paper is referenced in a chronology of lot sizing and scheduling problems [4], used as an example of multiproduct small bucket problem. A problem is said with small bucket if only one item can be done per period, in opposition to large bucket problems, where several items can be done per period. DLSP is a small bucket period. This accronym is quite recent. Its first apparition is made in 1990: after having defined the problem, [5] solved it with a branch and bound used with Lagrangian Relaxation. In 1991, a classification of DLSP extensions were proposed and complexity was discussed [6]. According to this classification, our problem can be denoted: P (parallel unrelated machines) / M (positive integer number of machines) / N (positive integer number of items). Without setup consideration, this kind of context has a NP-Complete feasibility problem and a NP-Hard optimization problem [6]. In 1994, the two-stages consideration is introduced [7]. All these historical articles consider problems with an unique resource, or several identical resources. Most of them propose some algorithms inspired by exact method such as branch and bound or Lagrangian decomposition.

More recently, some papers propose new methods to solve the DLSP. In 2011, an exact solution approach is proposed to solve medium-size instances, in the case of identical parallel machines [8]. Mixted Interger Programming (MIP) formulation are proposed and used with a commercial solver. This method solves instances up to 20 products, 100 periods and 5 resources. The Proportional Lot-sizing and Scheduling Problem allows processing of two products within a single period, one before and another after the setup operation, while for the

DLSP, the setup is made at the beginning of the period. It has mainly been solved with identical resources, for example using MIP procedures [9]. Instances considering 10 products and 5 machines are not solved in less than 30 minutes. As a perspective, the author proposed to use these procedures to unrelated parallel machines.

In 2017, a classification and review about simultaneous lot sizing and scheduling problems, including DLSP, has been done [10]. While trying to find analogies with our current problem in that recent review, it was noticed that three terms can be used to define when all machines can not treat all products. Machines can be called as not-identical, heterogeneous or unrelated. Among the 175 referenced articles, 59 are one of the kind, in all scheduling and lot-sizing problem. According to the DLSP, 34 papers are referenced, 7 of them dealing with non-identical machines. Only 3 of them take into account the setup: 2 with sequence independent setup and 1 with sequence dependent setup [11] using Lagrangian Relaxation.

Our problem is a DLSP with unrelated parallel machines due to compatibility restrictions, considering sequence dependent setup. [1] used a heuristic approach to solve that specific problem: the list of products is sorted, and then products are assigned to resources. In the following, our generic method, which will be used to solve our problem, is presented.

4 Proposed Method

The used method to solve this problem has previously been applied. First the method has been developed in a hospital context: exams planning and resources assignment [12]. Then it has been used to solve a lot sizing and scheduling problem for the injection industry, identified as a Capacitated Lot Sizing Problem (CLSP) [13]. Our main objective is to show that our tool is sufficiently generic. It is used with a minimum work development to solve the current problem, identified as a DLSP.

4.1 General Description

The method can be summarized as Fig. 1. It is a hybridization of a metaheuristic and a list algorithm. The list algorithm L has to be specified for any considered problem. It is applied on a list $Y \in \Omega$, to schedule and assign all jobs $Y_i,\ i \in \{1,$ number of products$\}$, respecting all constraints of the problem. This builds the solution $X \in S$. The list algorithm must consider the list order while building solution X, it must not sort the list in any way. The order of the list is determinated by the metaheuristic, by applying a neighborhood system. An objective function H evaluates the quality of the solution X, by computing the number of remaining quantity at the end of the month or the number of setup. The whole method is illustrated by Equation (1), with Ω the set of all lists of products and S the set of all admissible solutions. A solution is admissible if it is a schedule and assignment which respects all constraints of the system.

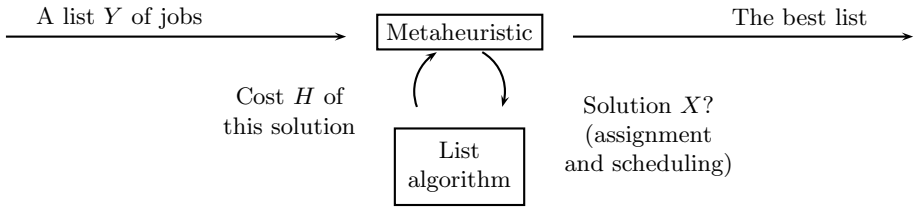

Fig. 1. Hybridization metaheuristic - List algorithm

$$Y \in \Omega \underset{Heuristic\ L}{\longrightarrow} L(Y) = X \in S \underset{Criterion\ H}{\longrightarrow} H(X) \tag{1}$$

The metaheuristic is the generic part. Algorithm 1 illustrates the method, using the stochastic descent as a metaheuristic. Any metaheuristics can be used. The stochastic descent is working on a list Y of jobs. Metaheuristic browses the set Ω of lists Y by applying a neighborhood system V. V is the permutation between two products in the list Y. Other neighborhood system may be used. Solution are compared thanks to the objective function H. According to the value of H, the best solution becomes the current one.

Algorithm 1: Principle algorithm of stochastic descent

```
   Data: Initial solution Y
1  X = L(Y)
2  while necessary do
3  |  Choose uniformly and randomly Y' ∈ V(Y)
4  |  X' = L(Y')
5  |  if H(X') ≤ H(X) then
6  |  |  Y := Y'
```

4.2 List Algorithm L

When solving such a problem using our method, the main thing to do is to develop a list algorithm which schedules the products assigning them to machines, respecting the constraints previously described in Section 2. Because it must be a list algorithm, the algorithm is considering jobs one by one respecting the order defined by the list. The used list algorithm is summarized in Algorithm 2. The list algorithm needs to be efficient to find a solution (good or not) in a small computation time. It does not need to be effective because a good solution will be found after having tried some neighbors thanks to the metaheuristic.

The hypothesis is made that at the beginning of the month, all machines are empty, there is no tool on them, no raw material. This hypothesis is not restrictive. Maybe, the needed tool at the beginning of the month was already on the machine at the end of the previous month, so a setup is avoided.

Algorithm 2: Principle algorithm of the list algorithm

```
   Data: Initialize all variables at zero
 1 forall the JOB in the list do
 2     while all quantity of JOB has not been scheduled do
 3         MACHINE = first machine
 4         while Next machines can be considered do
 5             DAY = first day
 6             while Next days can be considered do
 7                 if MACHINE is compatible with JOB then
 8                     if MACHINE is available then
 9                         Assign JOB, the needed tool, and the needed color to
                           MACHINE during DAY
10                         Update the lifetime of the tool, change the tool if needed
                           with a new tool (lifetime = 100%)
11                         Update the number of setups if needed (change of tool,
                           transition from black to shiny, transition from mat to
                           shiny)
12                         Update the remaining quantity of JOB to produce
13                 DAY = Next day
14             MACHINE = Next machine
```

4.3 Objective Function H

The objective function is used to compare solutions in the metaheurisitic. The main used criterion is the remaining quantity of all products at the end of the month, written C_q. To provide all customer demands, our system must produce as many quantities as possible in a month. Another used criterion is the number of setup (caused by a change of tool, a change of color, or an exceeded lifetime), written C_s. A hierarchy of both criteria is used.

5 Experiments and Results

To make our experiments, instances have been created, representing the data used by the company. Each instance is made by:

- A list of jobs to be done, with the needed quantity, the type and color and the used tool.
- A matrix representing the effectiveness of the machines to produce the jobs, which can be null if the machine and the job are not compatible.
- The lifetime of each tool on each machine, depending on the type and color of made product.

Our experiments are made on a computer powered by an i7 CPU running at 2.6 GHz with 16 Go RAM. Table 1 summarizes the results of our method

giving the remaining quantity C_q and the number of setups C_s, with the needed computational time. The number of products is assigned to 10 machines over 28 days. Results are compared to the ones found by using the method presented by [1]. The results show that our method gives better results than the ones from the previous method. Indeed, the previous method finds a solution given by a constructive heuristic while our method browses a set of solutions and gives the best one among all tested solutions.

Table 1. Results of our method

Instance	[1]: $(C_q; C_s)$	Our method: $(C_q; C_s;$ Time)
22 products	$(552; 17)$	$(522.6; 17;$ 13 seconds)
25 products	$(232; 16)$	$(68.0; 21;$ 8 seconds)
33 products	$(387; 17)$	$(196.0; 19;$ 11 seconds)

6 Conclusion and Further Work

This paper presented an application of our generic decision support tool to an application of lot sizing and scheduling in a textile industry. The results given by our method are better than the previous known results. Because our tool is generic, it is easy to use with a minimum work development. The only things to do are to build a list algorithm, to compute an objective function, and to transform the real data in usable instances. Many problems can be solved as long as a list algorithm can be used to provide a solution to the considered problem.

As a first perspective, our tool could be used connected with the Information System of the company. Internet of Things may collect data on the shop-floor to better describe the system. Scheduling and assignment solution could be transformed to Fabrication Orders directly transmitted to the shop-floor via the Manufacturing Enterprise System. Thanks to a connection to the Enterprise Resource Planning, when a customer gives a new order, our tool could simulate when this customer will be delivered, thanks to our criteria.

Our tool could be improved by using other metaheuristics such as population based metaheuristics, for example particle swarm optimization. Using this, our tool could be performed using parallel computation on GPUs. Then, we could implement a database of our industrial problems we already solved, to better solve the next ones.

To assess the quality of our result, we should find some benchmarck in the literature. By solving the available data, we will be able to position the quality of our tool among all other existing methods. Existing methods are dedicated, so the fact that our tool is generic is already an originality.

All problems we already solved are about tactical and operational levels. They deal with planning and assignment problems, lot-sizing and scheduling problems.

As future work, we intent to solve problems at a strategic level, about sizing of the system, sizing of resources, according to the future demand. Nowadays, because of the way of consuming, products lifecycle becomes shorter and shorter, so demands are unpredictable. Reconfigurable manufacturing systems should now be considered. Manufacturing systems need to be agile and flexible to be adaptable to the variety of product and quantity. Our next step is to size and maintain such a system with our generic decision support tool.

References

1. Silva, C., Magalhaes, J.M.: Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry. Computers & Industrial Engineering **50**(1) (2006) 76–89
2. Drexl, A., Kimms, A.: Lot sizing and schedulingsurvey and extensions. European Journal of Operational Research **99**(2) (1997) 221–235
3. Lasdon, L.S., Terjung, R.: An efficient algorithm for multi-item scheduling. Operations research **19**(4) (1971) 946–969
4. Eppen, G.D., Martin, R.K.: Solving multi-item capacitated lot-sizing problems using variable redefinition. Operations Research **35**(6) (1987) 832–848
5. Fleischmann, B.: The discrete lot-sizing and scheduling problem. European Journal of Operational Research **44**(3) (1990) 337–348
6. Salomon, M., Kroon, L.G., Kuik, R., Van Wassenhove, L.N.: Some extensions of the discrete lotsizing and scheduling problem. Management Science **37**(7) (1991) 801–812
7. Brüggemann, W., Jahnke, H.: Dlsp for two-stage multi-item batch production. The International Journal of Production Research **32**(4) (1994) 755–768
8. Gicquel, C., Minoux, M., Dallery, Y.: Exact solution approaches for the discrete lot-sizing and scheduling problem with parallel resources. International Journal of Production Research **49**(9) (2011) 2587–2603
9. Kaczmarczyk, W.: Proportional lot-sizing and scheduling problem with identical parallel machines. International Journal of Production Research **49**(9) (2011) 2605–2623
10. Copil, K., Wörbelauer, M., Meyr, H., Tempelmeier, H.: Simultaneous lotsizing and scheduling problems: a classification and review of models. OR Spectrum **39**(1) (2017) 1–64
11. De Matta, R., Guignard, M.: Studying the effects of production loss due to setup in dynamic production scheduling. European Journal of Operational Research **72**(1) (1994) 62–73
12. Klement, N., Gourgand, M., Grangeon, N.: Medical imaging: Exams planning and resource assignment: Hybridization of a metaheuristic and a list algorithm. In: 10th International Conference on Health Informatics, Porto, Portugal. (2017)
13. Silva, C., Klement, N., Gibaru, O.: A generic decision support tool for lot-sizing and scheduling problems with setup and due dates. In: International Joint Conference - CIO-ICIEOM-IIE-AIM (IJC 2016), San Sebastian, Spain, ICIEOM (2016)