da Rocha, Rony Peterson; da Rocha, Fabrício Wesley; S. S. Ravagnani, Mauro Antônio;
Paraíso, Paulo Roberto; Gonçalves Andrade, Cid Marcos
USE OF EXCEL WORKSHEETS WITH USER-FRIENDLY INTERFACE IN BATCH
PROCESS (PSBP) TO MINIMIZE THE MAKESPAN
Independent Journal of Management & Production, vol. 5, núm. 1, enero, 2014, pp. 142-
165
Instituto Federal de Educação, Ciência e Tecnologia de São Paulo
Avaré, Brasil

USE OF EXCEL WORKSHEETS WITH USER-FRIENDLY INTERFACE IN BATCH PROCESS (PSBP) TO MINIMIZE THE MAKESPAN

Rony Peterson da Rocha
Universidade Estadual do Paraná (UEP), Brazil
E-mail: petersonccbpr@hotmail.com

Fabrício Wesley da Rocha
Universidade Estadual de Campinas (UNICAMP), Brazil
E-mail:phabriciowesley@hotmail.com

Mauro Antônio S. S. Ravagnani
Universidade Estadual de Maringá (UEM), Brazil
E-mail:ravage@dep.uem.br

Paulo Roberto Paraíso
Universidade Estadual de Maringá (UEM), Brazil
E-mail: paulo@dep.uem.br

Cid Marcos Gonçalves Andrade
Universidade Estadual de Maringá (UEM), Brazil
E-mail: cid@dep.uem.br

ABSTRACT

In the chemical industry, the necessity for scheduling is becoming more pronounced, especially in batch production mode. Nowadays, planning industrial activities is a necessity for survival. Intense competition requires diversified products and delivery in accordance with the requirements of consumers. These activities require quick decision making and the lowest possible cost, through an efficient Production Scheduling. So, this work addresses the Permutation Flow Shop scheduling problem, characterized as Production Scheduling in Batch Process (PSBP), with the objective of minimizing the total time to complete the schedule (Make span). A method to approach the problem of production scheduling is to turn it into Mixed Integer Linear Programming- MILP, and to solve it using commercial mathematical

programming packages. In this study an electronic spreadsheet with user-friendly interface (ESUFI) was developed in Microsoft Excel. The ease of manipulation of the ESUFI is quite evident, as with the use of VBA language a user-friendly interface could be created between the user and the spread sheet itself. The results showed that it is possible to use the ESUFI for small problems.

Keywords: Excel; Permutation Flow Shop; Mixed Integer Linear Programming – MILP

1.   INTRODUCTION

In the chemical industry, the necessity for scheduling is becoming more pronounced, especially in batch production mode. According to Raklaitis (1995), batch production can be described as income-oriented production, with equipment network connectivity and resource limitations. Mendez *et al.* (2006) classifies this type of operation: Single stage, which is subdivided into single or parallel resources, and multiple stages, which are subdivided into multipurpose (Job Shop) and multiproduct (Flow Shop).

The scheduling problem addressed in this work is of batch type - multiproduct - Permutation Flow Shop. According to Polon (2010), Batch-Multiproduct plants are in general employed for a set of products whose income structure is the same and production lines are also referred to as Flow Shop.

According to MacCarthy and Liu (1993) and Baker (1974), Flow Shop is a type of process where all tasks have a similar flow pattern, that is, they have the same processing schedule on all resources and the number of resources in each stage of production equals one.

The environment of the permutation Flow Shop production scheduling problem is described by MacCarthy and Liu (1993), Baker (1974), Taillard (1993), and Moccellin (1995) as a type of Flow Shop in which the processing order of tasks is the same in all resources. Severo (2007) states that many chemical process industries such as oil and paint, pharmaceutical and fine chemistry industries fit into this category.

The diversity and complexity of a batch production industry recommend, as stated by Severo (2007), the application of production planning techniques in order to meet consumer demands. Thus, scientific development in the area of Production Scheduling is increasing, especially concerning batch processes.

This work presents the development of a spreadsheet, created using Visual Basic (VB) with interface with the user, whose goal is to minimize the task completion time (Makespan) of a Permutation Flow Shop Scheduling environment.

Therefore, this paper is divided into six sections. The first section presents the research problem. The second section will be described in a literature review on the topic in question. The third section will be described the methodology used in the work. On Wednesday we will present a description of each of the methodological steps, as well as the formulation of the research problem. The fifth section will be held to present and discuss research results. Finally, the sixth section will be held the final considerations.

2.   STATE OF THE ART ABOUT SCHEDULING PROBLEM

Since Johnson's work edition in 1954, about flow shop scheduling , as described by Gupta and Stafford JR (2006), there was an important progress in this knowledge area, derived from it, there are endless other researches published all over the world, about this subject. Gupta and Stafford JR (2006), describes a review about flow shop scheduling, based on five (5) decades of publications, so the author leaves it pretty clear that the results of those publications are the many different ways released in literature about the heuristic solutions proceeding and of available optimization to solve a huge variety of problems, flow shop scheduling. So the Gupta and Stafford JR's work, presents a summary about the evolution of these problems with the propose of showing a redirection, of this area, to the future. Frequently, as said by Gupta and Stafford JR (2006), problems about the proceeding of a certain number of job in a number of machines, are characterized by different researchers of the area as a problem of scheduling, dispatching, sequencing, or combinations thereof. The first term: "scheduling", has to do not only with the programming, but also with the sequencing of the production. In this context, even if decades of studies pass, the scheduling problem is still discussed a lot on its specific literature, because it involves the accomplishment of certain objectives (function objective), related with the numberless restrictions of the different study environments, which leads to another variety of programming problems; as, among others, a simple scheduling problem, a programming of multiple machines problem, or a problem of labor programming.

The initial studies about the flow shop scheduling were published by Johnson (1954), Bellmane Gross (1954), Bellman (1956) and Bellman et al (1982). Being JOHNSON's work (1954), about the flow shop scheduling problem, directed to an environment with two machines and its forerunner. Therefore, JOHNSON's study emerged from a method (algorithm) to optimize the sequencing of a group of a certain number of jobs to be processed in a production system with only two machines.

Gupta and Stafford JR's (2006) show in their article that a flow shop is characterized by an unbroken flow of jobs through a variety of serial machines. The job flow is unidirectional whereas all jobs obey the same technological route through the machines (considered in the chemical engineering as a multiproduct plant). So some chances and theorems related to the problem of scheduling flow shop are represented on Gupta and Stafford JR's (2006) article from the Conway et al. work (1967) and Tanaev et al. work (1994). However Gupta and Stafford JR say that the cases of scheduling flow shop show restricted structures when it comes to a practical point of view (real cases), with mathematical formulations that ignored multiple important factors that interrupt the job flow on the system of production (SP), causing unexpected delays, are discussed on Ashour's (1972) literature, such as: lack of consistence on the performance and absence of the operators; delays on the material, tools or equipment supply; variability on the processing time; changes on the job specifications; pressure coming from the clients to a quick conclusion of the work, etc.

Gupta et al (1979), present various assumptions to the flow shops, put together by the job characteristics, machines and policies of the process functioning (operational policies). Simplifying hypothesis are also imposed to the flow shop case, although these can increase the generality of the developed models, so the possibility of finding the ideal scheme, sometimes is lower, because the models get too far from the real situations. Thereafter, during the years it have been added lots of modifications to the original model of Johnson (1954), by researchers, to attend more and more the needs imposed by the system of production (SP) of the real life, such as the terms: without intermediate queues; time of preparation dependent of the sequence; flow shops hybrids, and so on. It has been showing an evolution of this subject facing the reality, for example: Gupta and Stafford JR's (2006), present on

145

their article a general description of the evolution of this area from 1955 to 1964 (first decade), from 1965 to 1974 (second decade), from 1975 to 1984 (third decade), from 1985 to 1994 (fourth decade), from 1995 to 2004 (fifth decade) and since 2004 (actual times).

On the first decade, that came after the publication of the Johnson's classic work in 1954 about flow shop scheduling to the production programming in two machines, this area has increased to an environment of production programming with more machines with the goal of minimizing the total time of conclusion of the tasks (makespan).

In this time the researchers studied the environment of production programming with m-machines, based on a completely theoretical view, using optimization techniques (mathematical programming) and the simulation technique and Monte-Carlo, as shown in Wagner's work (1959), Sisson's (1959), Manne's (1960), Muth and Thompson (1963), Dudek and Teutão (1964), and also published by Gupta and Stafford JR (2006). In this period of time, although just a few techniques of solution were developed, still some processing configuration and scenarios were identified. The difficulty to solve a lot of problems considered possible to solve, existed because of the lack of computer efficient programs and because of a huge part of the different variant of the flow shop problem with two machines are considered NP-hard, in other words, problems there are difficult to solve.

The "second decade" is known by its wide range of solution techniques and by the appearance of different functions objective. In this decade the works of Lomnicki (1965), Ignall e Schrage (1965), Brown and Lomnicki (1966), Mcmahon and Burton (1967), Smith and Dudek (1967), Mcmahon (1969), Gupta (1969B), Gupta (1970), Gupta (1971), Szwarc (1971), Gupta (1972) and Szwarc (1973) were cited. A technique used in that time was the "Branch and bound". The difficulty found on the development of algorithms of optimization to programming problems in flow shop environments, resulted on the development and usage of heuristics to find good or almost great solutions.

On the "third decade", the theory of NP-Completude was born; it presented an important impact to the evolution of the production programming area in flow shop. A lot of heuristic approaches were developed in this phase, besides in this decade has occurred a proliferation of a variety of considerations about programming problems in

flow shop, that violated some of the restrictive assumptions listed at the beginning of this theory.

The fourth decade is marked by the appearance of flow shops hybrids where each of its phase can present various parallel machines. In this period of time it emerged the studies and development of metaheuristics, such as: Tabu Search, Genetic algorithms and "simulated annealing". This decade has also witnessed the usage of techniques based on artificial intelligence. In this phase are cited authors like: Byrd and Moore (1983); Zweben and Fox (1994); Brown et al. (1995); Osman and Kelly (1996); Rayward-Smith et al. (1996); Aarts and Lenstra (1997); Allahverdi et al. (1999) and Cheng et al. (2000).

In the fifth decade there was the continuity of the proliferation of a variety of programming problems flow shop, function objective and solution approaches. An important part of this decade was the analysis and solution of (multi-criteria) that have become really popular. It has also been important in this period of time the studies about lot sizing and programming of machines of lot processing, efficiency of diverse metaheuristics to the minimization of the makespan into flow shops with setup times dependents of the sequence, significant progress on the development of solution process to robotic and equipped with automatic vehicles flow shops (AGV). It´s also important to say that because of the progress on the computer area, there was an increasing interest on the job for mathematical programming approaches of flow shop problems. There are cited here the Works of: Potts and Van Wassenhove (1992), Tkindt and Billaut (1993), Potts and Kovalyov (2000), Ruiz, Maroto and Alcaraz (2005), Geismar et al. (2004), Tseng et al. (2004).

Nowadays studies are developed considering the problem of preparation time of the machines, in other words, differently of the initial study of Johnson (1954), that considered the time of preparation together with the processing time, recent studies show that if these times are separated, the configuration leads to better performance. Although the approach of mathematical programming to the production programming in environments flow shop, for a certain time, was rejected, because of its excessive computer luggage to solve the mathematical model that shows the reality, so heuristics proceedings would have to be used all the time. But recently huge advances on the solution of problems of "mathematical programming" and the availability of proceedings with approximate solution shows that the approach of the

mathematical programming can be used to find real answers and with less computer effort. This, as said by Gupta and Stafford JR (2006), shows that the techniques of mathematical programming and other programming proceedings in flow shop (heuristic based on mathematical programming approaches) indicate a trend of search redirected to the future.

Gothe-Lundgren and Persson (2002) have studied the problem of scheduling in an oil refinery with the intention of choosing the best way of operation, in order to attend the demand and the capacity of storage, reducing the cost of production. The role model was elaborated in the Mixed Integer Linear Programming. Thereby, it was showed that the model of optimization can be used as a applicable tool to support the planning of production and programming at the refinery, and that the model can bear the shipment planning and strategic decisions about new products and investments in storage capacity.

Sundaramoorthy and Karimi (2005) reported on the production scheduling problem in the short term for plants in batches multipurpose. This study was presented one formulation based on mixed integer linear programming (MILP), using a continuous representation of time slots synchronous and a new idea of multiple balances (time, mass, resources). The model uses without big-M constraints, and is equally effective for both maximize profit and minimize the makespan.

Burkard and Hatzl (2006) researched the problem of production scheduling in the chemical industry. The objective of the study was implement an objective function to minimize the makespan, using heuristics. In the study proposed an iterative algorithm construction that alternates between phases of construction and deconstruction. It was also proposed strategies for diversification and intensification in order to obtain optimal solutions in good moderate running times. Computational results show the power of this algorithm.

Shakib and Logendran (2007) studied the problem of production scheduling model as a mixed integer linear programming (binary) developed for scheduling tasks in multitasking environments, for which the number of completed tasks is not a good measure. Fixed issues for small, medium and large, with the aid of a tabu search algorithm. The solution obtained from the algorithm is compared with that of the optimal solution or the upper bound found from using the Lagrangian relaxation.

Pan, Qian and Li (2008) describe a study on for short-term scheduling of multipurpose batch plants using Mixed Integer Linear Programming. This work presented the network states and tasks (STN) to eliminate the inconvenience of precedence based formulations which typically include a large number of batches. Rules have been proposed in the study heuristics for solving the problem. The results were effective to find the best solution to problems.

Privitera, Day, Deshi and Long (2011), presented a paper about the application of the Mixed Integer Linear Programming on the technological school of renewable energy, with the goal of reducing the CO2 emission. On that paper the authors implemented the objective function based on the cost reduction. the implementation was done by using the solver: "lp_solver", that is a free-font Mixed Integer that was incorporated in a Microsoft Excel app.

 Grossmann (2012) presented a study about optimization, that is an important subject when talking about process industries, because of the competitively on the global market. The author presented a general view in terms of mathematical programming structure such as techniques of mathematical programming as the Mixed Integer Linear Programming. In this paper, it is concluding that it is necessary to put the control with the planning and the programming all together; however, the attention has been dedicated mainly to the integration of the planning and programming.

Lin and Liao (2012) presented a paper about a scheduling problem for a two-stage assembly shop in a machinery factory. The objective is to minimize the weighted sum of makespan, total completion time and total tardiness. A Mixed Integer Programming (MIP) model is developed for solving small-size problems, and three heuristics are proposed for solving medium- and large- size problems.

Catalá, Durand, Blanco and Alberto (2013) Developed a research in a farm about the Mixed Integer Linear Programming usage, to plan the restructuring of an orchard of pears and apples. The result shows the fruits replacing policy during 20 years. On the model, the maximization of the business current liquid value was taken into account, and also it's planting structure during a certain time under different funding conditions. The model can be used not only by government agencies to counsel private sectors and develop strategic economic policies, but also by companies to optimize their profit.

Koné, Artigues, Lopez and Mongeau (2013) have studied the problem of scheduling problem that takes into account storage resources which may be produced or consumed by activities. The role model was elaborated in the Mixed Integer Linear Programming.

Fumero, Corsano and Montagna (2013) also conducted a study on Mixed Integer Linear Programming. In this study was conducted a model for both design and scheduling of flow shop batch plants considering mixed product campaign and parallel unit duplication. The results showed that a realistic formulation is attained, where industrial and commercial aspects are jointly taken into account.

Solimanpur and Elmi (2013) proposed a model of mixed integer linear programming for the scheduling problem of cells, as well as an application of tabu search approach to solve the problem heuristically.

Thus, this study sought to investigate the problem of production scheduling for batch process (Flow - Shop) through a mathematical formulation implemented in Excel and VBA, adopting as a principle the state tasks network (STN) formulation and Mixed Integer Linear Programming to the problem of reducing the total time of completion of tasks (MAKESPAN).

3.   METHODOLOGY

For the creation of a user-friendly interface to solve a production scheduling model for the permutation Flow Shop system, it was necessary to divide it into five parts:

3.1 Model Development

The problem proposed by Polon et al. (2006), of a multiproduct batch plant, characterized as a permutation Flow Shop environment, was analyzed for the model development.

3.2 Model Formulation

At this stage, the objective function was defined, considering the case of production scheduling as a mixed integer linear programming problem (MILP).

In this permutation Flow Shop programming environment, one research scenario were modeling, all of them with the aim of minimizing the total task completion time (Makespan).

150

The model constraints are: a) the storage of intermediate products should not be available between the processing resources, that is, if a product is processed at resource j and resource j+1 is not available at the time of completion, the finished product should be kept at resource j, until resource j+1 is ready; b) after finishing the processing of a product at the last resource (equipment), this product is immediately sent to the stock of finished products; c) all resources are initially empty at time zero and the manufacture of any product may be delayed an arbitrary amount of time to keep it in the previous resource; d) the ordering of tasks on each resource is the same.

3.3 Model Resolution in Excel Spreadsheets

After finishing the modeling, the mathematical equations of all one scenario were manually transcribed into Excel spreadsheets, and solved by the add-in Solver.

3.4 Creation of Interface Between User and Worksheet in Visual Basic Language (VB), and Finally

The VB language was used to create an interface between the user and the Excel spreadsheet. This way, all the mathematical equations that had been modeled in the one scenario, along with the add-in Solver, were developed in order to create a generalized programming model for this environment and with the restrictions imposed by the proposed model.

3.5 Evaluation of the Consistency of the Interface

The evaluation of the consistency of the spreadsheet interface with the user was done by comparing the results that were manually obtained with Excel spreadsheets with those obtained using the VBA language.

4. USER-FRIENDLY INTERFACE FOR THE RESOLUTION OF A MODEL OF PRODUCTION SCHEDULING IN A PERMUTATION FLOW SHOP SYSTEM WITH MAKESPAN REDUCTION

4.1 Model Development

To develop the research was analyzed a research scenario, referring to the programming environment of three tasks (t1, t2, t3) and five resources, with the respective processing times shown in Table 1.

Table 1 shows the respective processing times for each of the tasks on each machine.

151

Table 1 - Processing time (h)

Resources	Tasks		
	t1	**t2**	**t3**
1	Pt 11	Pt 21	Pt 31
2	Pt 12	Pt 22	Pt 32
3	Pt 13	Pt 23	Pt 33
4	Pt 14	Pt 24	Pt 34
5	Pt 15	Pt 25	Pt 35

4.2 Model Formulation

The three scenarios were modeled. However, as all of them are part of the same programming environment (permutation Flow Shop) and meet the same constraints, only the modeling of scenario 1 will be presented. In the mathematical modeling of this scenario, Equation 1 represents the main goal of the problem to be solved.

$$Minimize\ C_{NM} \tag{1}$$

Variable C in Equation 1 represents the Makespan, while N and M represent the number of tasks and the number of resources, respectively. Thus, the objective function expressed by Equation 1, is about finding, among the various combinations of tasks and resources, the NM sequence that provides the lowest Makespan. For the scenario 1, N = 4 and M = 3 in the objective function. This objective function is subject to a number of constraints, as described below. The first constraints, shown in Equations 2 and 3, are binary.

$$\sum_i X_{ik} = 1 \quad \forall k \tag{2}$$
$$\sum_k X_{ik} = 1 \quad \forall k \tag{3}$$

Variable *i* represents the task to be processed at the resource and *k* represents the position of this task in the order of sequencing. Equations 2 and 3 thus represent a discrete optimization problem involving the decision between two alternatives, that is, if a task is processed at a given resource, the other tasks cannot be processed concurrently at the same resource. Therefore, X_{ik} is a binary variable defined as: X_{ik} = 1 if task *i* is in position *k*, and X_{ik} = 0, otherwise.

Each task is sequentially processed by resources (1), (2), and (3), respecting the permutation Flow Shop programming environment.

Equation 4 represents another problem constraint, which also involves Makespan. In this Equation, $C_{k,j}$ is the time of end of processing, by resource j, of the task occupying position k in the sequence; N is the number of tasks; $X_{s,k}$ is the binary variable, and $TP_{s,j}$ is the time for processing task i at resource j.

$$C_{k,j} \geq C_{k-1,j} + \sum_{s=1}^{N} X_{s,k} TP_{sj} \quad \forall j; k = 2, \dots, N \tag{4}$$

Generally, the resolution of the various alternatives of Equation 4 shows that for a scheduled task to be processed in resource j, from the second order of sequencing, it must present a Makespan greater than or equal to the Makespan of a task placed earlier on the same resource j, plus the processing time of the same task on resource j.

Another important constraint is Equation 5, which also has to do with the Makespan of the tasks in the resources. Equation 5 shows that the Makespan of the task of order k on resource j is greater than or equal to the Makespan of the same task on resource j-1 plus the sum of the products of the binary variable $X_{s,k}$ and the processing times $TP_{s,j}$, starting with resource (2) and finishing with resource (3).

$$C_{k,j} \geq C_{k,j-1} + \sum_{s=1}^{N} X_{s,k} TP_{s,j} \quad \forall j; j = 2, \dots, M \tag{5}$$

The Makespan of the task placed in the sequence 1 of the production scheduling, at resource j, is expressed in Equation 6. This Equation presents the constraint that the Makespan of such task is greater than or equal to the Makespan of the task placed in the sequence 1 of the production scheduling, at resource j-1, plus the sum of the products of the binary variable $X_{s,1}$ and the processing times $TP_{s,j}$. In this Equation, resource j must start with resource 2 and finish with resource 3, according to the model initially proposed.

$$C_{1,j} \geq C_{1,j-1} + \sum_{s=1}^{N} X_{s,1} TP_{s,j} \quad j = 2, \dots, M \tag{6}$$

Another important constraint is the Makespan of the task sequenced with order 1 on resource 1, which is represented by Equation 7. This Equation shows that the Makespan of this task must be greater than or equal to the sum of the products of the binary variable ($X_{1,1}$, $X_{2,1}$, $X_{3,1}$, $X_{4,1}$) and the processing times ($TP_{1,1}$, $TP_{2,1}$, $TP_{3,1}$, $TP_{4,1}$).

$$C_{1,1} \geq \sum_{s=1}^{N} X_{s,1} TP_{s,j} \tag{7}$$

The constraint presented in Equation 8 shows that the Makespan sequenced in the order k on resource j must be greater than or equal to the Makespan sequenced in the order k-1 on resource j +1, starting from the order of sequence 1 on resource 1.

$$C_{k,j} \geq C_{k-1,j+1} \qquad k = 1, \dots, N \qquad j = 1, \dots, M-1 \tag{8}$$

Finally, the last constraint of the problem concerns the question of non-negativity of the model, which is represented by the general Equation 9.

$$C_{k,j} \geq 0 \quad \forall k, \forall j \tag{9}$$

4.3 Model Resolution in Excel Spreadsheets

Equations (1) through (9) of the three tasks and five resources model were inserted in the worksheet, in different cells. After that, the add-in Solver, available within the Microsoft Excel electronic spreadsheet, was used to optimize the scheduling of the tasks on the resources.

In general, for the development of the manual resolution of the processing orders scheduling model proposed in this study, all available data was first organized in an Excel spreadsheet, separating the cells representing the decision variables and the objective function. For each constraint of the problem, a formula was created in a separate cell of the spreadsheet, corresponding to the left-hand side (LHS) and the right-hand side (RHS) of the restriction.

LHS and RHS correspond to the transformation of the set of constraints into a set of equivalent equations, by introducing variables that will represent the gap between the left (LHS) and right (RHS) sides of the inequalities.

After inserting all the equations of the PSBP model, representing the objective function, decision variables and restrictions, in the Microsoft Excel electronic spreadsheet shown, the add-in Solver, available in the tool bar of this spreadsheet, was used to find the optimal solution for the problem.

4.4 Creation of the Friendly Interface Between the User and the Excel Spreadsheet in the Language *Visual Basic for Applications* (VBA)

154

For the creation of the user-friendly interface with the Excel spreadsheet (ESUFI) used to solve the model, in order to reduce the Maskespan following the constraints set forth in the methodology, it was necessary to first define a standard spreadsheet. The standard spreadsheet refers to a Microsoft Excel spreadsheet, composed of all the equations described in the study, organized in a standard model, which means to put each of the equations in an array of rows and columns, available to perform the generalization of the model. The generalization of this model was performed using the VBA programming language.

The implementation of the VBA code in the spreadsheet made possible the generalization of the model of three tasks and five resources for the configuration of (n) tasks and (m) resources. With this implementation, a friendly interface was created between the user and the Microsoft Excel spreadsheet. At this friendly interface the user should only report the number of tasks (n) and resources (m), and the processing time of each task at the respective resource, and then click on the button "Optimize Scheduling Order" to obtain the best production sequence that minimizes the Makespan. Figure 1 exemplifies the model of the Microsoft Excel spreadsheet with user-friendly interface.
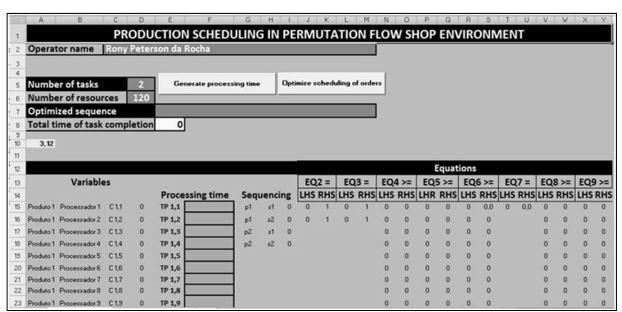

Figure 1 - Microsoft Excel spreadsheet model with user-friendly interface

To build the user-friendly interface created for the ESUFI presented in Figure 1, it was necessary to carry out some validation tests for each of the model equations. These tests serve to validate the model, that is, to show the veracity of each of the equations from the expansion of the model of three tasks and five resources for different situations of (N) tasks and (M) resources. The tests also

served to find a number of combinations of (N) and (M) possible to be applied with the help of the Solver tool.

4.5   Evaluation of the Consistency of the Interface

The assessment of the consistency of the generalized equations in the Excel spreadsheet with user-friendly interface was done by comparing the results that were manually obtained with Excel spreadsheets with those obtained with the user-friendly interface, using the VBA language.

To validate the automated spreadsheet regarding the accuracy of the equations generated by the expansion of the model, two scenarios were tested in each one of the classes N $\in$ {2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13} and M $\in$ {2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 40, 60}, with time intervals of [1,99], totalling 360 problems tested (2 scenarios x 12 classes of tasks x 15 classes of resources).

The classes of problems N $\in$ {2, 3, 4, 5, 6, 7} and M $\in$ {2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25}, according to the classification of Boiko (2008), represent the small sized production scheduling problems. However, for the same author, classes N $\in$ {8, 9, 10, 11, 12, 13} and M $\in$ {30, 40, 60} correspond to medium-sized production scheduling problems.

In assessing the consistency of generalized equations in the Excel worksheet with user-friendly interface, tests were performed in cases of production scheduling in small and medium businesses. Besides verifying the veracity of equations (1) through (9), expanded on the Excel worksheet with user-friendly interface, the tests sought to find the limits of adjustable cells on the spreadsheet. Adjustable cells mean the largest number of tasks (N) and resources (M) of the subclasses that have the possibility of solving the objective function (minimizing the Makespan) in the Excel worksheet with user-friendly interface, regardless of the final solution.

The correlation coefficient (r) given by Equation 10 was used to assess the relationship between the number of tasks (N) and resources (M) of the limiting subclasses of the Excel spreadsheet model with user-friendly interface (MOREIRA, 2010, p 327).

$$r = \frac{n.(\sum X.Y)-(\sum X).(\sum Y)}{\sqrt{n.(\sum X^2)-(\sum X)^2}.\sqrt{n.(\sum Y^2)-(\sum Y)^2}} \tag{10}$$

Where: r = correlation coefficient; n = number of limiting subclasses; X = value of the number of the task subclass (n); Y = value of the number of the resource subclass (m).

Table 2 was used to interpret the values of r, in the analysis of the relationship between the number of tasks (n) and resources (m) of the limiting subclasses of the PSBP model in the Excel spreadsheet with friendly interface.

Table 2 - Interpretation of r values

(r)	correlation
0 to 0,2	very low
0,2 to 0,4	low
0,4 to 0,6	average
0,6 to 0,8	high
0,8 to 1,0	towering

According to data presented in Table 2, if it is found a value between 0 and 0.2 can be said that the correlation is very low and amounts to between 0.8 and 1.0, there is a high correlation.

5.   RESULTS AND DISCUSSION

It was verified in the tests of each problem of subclasses {2,2}, {2,3}, {2,4}, {2,5}, {2,6}, {2,7}, {2,8}, {2,9}, {2,10}, {2,15}, {2,20}, {2,25}, {2,30}, {2,40}, and {2,60} that all equations were consistent with the generic model equations. However, one realized that there was still scope to increase the number of resources, since the expansion of the equations was not at the limit allowed in the Excel spreadsheet. Therefore, a new set of problems with N $\in$ {2} and M > 60 was tested, and the limit of {2,98} was found.

The same behaviour was observed with the subclasses {3,2}, {3,3}, {3,4}, {3,5}, {3,6}, {3,7}, {3,8}, {3,9}, {3,10}, {3,15}, {3,20}, {3,25}, {3,30}, {3,40}, and {3,60}. This way, a new set of problems with N $\in$ {3} and M > 60 was tested, and the limit of {3,62} was found.

All equations of the problems of subclasses {4,2}, {4,3}, {4,4}, {4,5}, {4,6}, {4,7}, {4,8}, {4,9}, {4,10}, {4,15}, {4,20}, {4,25}, {4,30}, {4,40}, and {4,60} were consistent with the generic model. However, subclass {4,60} could not be verified, since it went beyond the limit of Solver adjustable cells present in the ESUFI. Thus, after conducting further tests, the limit of {4, 46} was found.

157

The same behaviour was observed for the subclasses {5,2}, {5,3}, {5,4}, {5,5}, {5,6}, {5,7}, {5,8}, {5,9}, {5,10}, {5,15}, {5,20}, {5,25}, {5,30}, {5,40}, and {5,60}, and it was found that subclasses {5,40} and {5,60} went beyond the limit. New tests led to the determination of the limit {5, 35}.

Further tests with the subclasses {6,2}, {6,3}, {6,4}, {6,5}, {6,6}, {6,7}, {6,8}, {6,9}, {6,10}, {6,15}, {6,20}, {6,25}, {6,30}, {6,40}, and {6,60}, pointed to the limit of {6,27}, after it was found that subclasses {6,30}, {6,40}, and {6,60} went beyond the limit of Solver adjustable cells present in the ESUFI.

Similarly, in the problems of subclasses {7,2}, {7,3}, {7,4}, {7,5}, {7,6}, {7,7}, {7,8}, {7,9}, {7,10}, {7,15}, {7,20}, {7,25}, {7,30}, {7,40}, and {7,60}, all equations were consistent with the generic model, although subclasses {7,25}, {7,30}, {7,40}, and {7,60} exceeded the limit of adjustable cells in the ESUFI. Thus, after performing new tests, the limit of {7, 21} was determined.

For the same reason, subclasses {8,20}, {8,25}, {8,30}, {8,40}, and {8,60} could not be verified, when analysing the subclasses {8,2}, {8,3}, {8,4}, {8,5}, {8,6}, {8,7}, {8,8}, {8,9}, {8,10}, {8,15}, {8,20}, {8,25}, {8,30}, {8,40}, and {8,60}, but all equations were consistent with the model up to the limit of {8,17}.

During the tests with the problems of subclasses {9,2}, {9,3}, {9,4}, {9,5}, {9,6}, {9,7}, {9,8}, {9,9}, {9,10}, {9,15}, {9,20}, {9,25}, {9,30}, {9,40}, and {9,60}, all equations were consistent with the generic model, but the last six subclasses could not be verified, as they went beyond the limit of Solver adjustable cells present in the ESUFI. New tests were performed and the limit of {9, 13} was found.

Analogously, the last six subclasses of the set {10,2}, {10,3}, {10,4}, {10,5}, {10,6}, {10,7}, {10,8}, {10,9}, {10,10}, {10,15}, {10,20}, {10,25}, {10,30}, {10,40}, and {10,60} were not verified, and the limit of {10,10} was found after performing new tests.

Following the same pattern, the limit observed for the subclasses {11,2}, {11,3}, {11,4}, {11,5}, {11,6}, {11,7}, {11,8}, {11,9}, {11,10}, {11,15}, {11,20}, {11,25}, {11,30}, {11,40}, and {11,60} was {11,7}, while for the subclasses {12,2}, {12,3}, {12,4}, {12,5}, {12,6}, {12,7}, {12,8}, {12,9}, {12,10}, {12,15}, {12,20}, {12,25}, {12,30}, {12,40}, and {12,60} the limit was {12,4}.

As for the results of the tests with the problems of subclasses {13,2}, {13,3}, {13,4}, {13,5}, {13,6}, {13,7}, {13,8}, {13,9}, {13,10}, {13,15}, {13,20}, {13,25}, {13,30}, {13,40}, and {13,60}, all equations were found to be consistent with the generic model, but only the subclass {13,2} could be generated on the spreadsheet, as the others went beyond the limit of Solver adjustable cells present in the ESUFI. Table 3 summarizes the evaluated classes and subclasses of problems, and the last subclass of each problem corresponds to the ESUFI limit for the application of the Solver tool to optimize the Makespan.

Table 3 - Summarizes the evaluated classes and subclasses of problems

CLASS	SUBCLASS
2	{2,2};{2,3};{2,4};{2,5};{2,6};{2,7};{2,8};{2,9};{2,10};{2,15}; {2,20}; {2,25};{2,30};{2,40};{2,60};{2,98}
3	{3,2};{3,3};{3,4};{3,5};{3,6};{3,7};{3,8};{3,9};{3,10};{3,15}; {3,20}; {3,25};{3,30};{3,40};{3,62};
4	{4,2};{4,3};{4,4};{4,5};{4,6};{4,7};{4,8};{4,9};{4,10};{4,15}; {4,20}; {4,25};{4,30};{4,40};{4,46};
5	{5,2};{5,3};{5,4};{5,5};{5,6};{5,7};{5,8};{5,9};{5,10};{5,15}; {5,20}; {5,25};{5,30};{5,35};
6	{6,2};{6,3};{6,4};{6,5};{6,6};{6,7};{6,8};{6,9};{6,10};{6,15}; {6,20};{6,25};{6,27};
7	{7,2};{7,3};{7,4};{7,5};{7,6};{7,7};{7,8};{7,9};{7,10};{7,15}; {7,20}; {7,21};
8	{8,2};{8,3};{8,4};{8,5};{8,6};{8,7};{8,8};{8,9};{8,10};{8,15}; {8,17};
9	{9,2};{9,3};{9,4};{9,5};{9,6};{9,7};{9,8};{9,9};{9,10};{9,13};
10	{10,2};{10,3};{10,4};{10,5};{10,6};{10,7};{10,8};{10,9};{10,10};
11	{11,2};{11,3};{11,4};{11,5};{11,6};{11,7};
12	{12,2};{12,3};{12,4};
13	{13,2};

Therefore, after performing all the tests in the classes of problems N ∈ {2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13} and M ∈ {2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 40, 60}, with time intervals of [1, 99], for the validation of the automated spreadsheet regarding the accuracy of the equations generated by the model expansion, and establishment of the limits of classes N x M of the PSBP model in the ESUFI, it was possible to conclude that equations (1) through (9) were correctly expanded in all 360 analyzed problems, thus ensuring the accuracy of the results obtained with the ESUFI.

Observing the limits found for the classes of problems N $\in$ {2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13} and M $\in$ {2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 40, 60}, with time intervals of [1,99], there is a clear reduction in the number of resources (M), as the number of tasks (t) increases. This reduction in the number of resources, concerning the limitation of adjustable cells in the ESUFI for the use of the Solver tool to solve the objective function of the PSBP problem, can be assessed by the application of the correlation coefficient (r), given by Equation 10. The value of -0.89 shows a very high inverse correlation between the number of tasks (n) and the number of resources (m), that is, for the adjustable cells of the ESUFI, as the number of tasks is increased, the number of resources decreases, resulting in a reduction in the number of resources to be programmed in the ESUFI.

In analyzing the performance of the model PSBP ESUFI held for class 3, were tested percentage of success ($\%S_{\mathrm{ESUFI}}$) of each of the subclasses listed in Table 3.

Table 3 - Class 3

CLASS	SUBCLASSES	INTERVAL OF TIME
3	{3,2};{3,3};{3,4};{3,5};{3,6};{3,7};{3,8};{3,9};{3,10};{3,15};{3,20};{3,25};{3,30};{3,40};{3,62};	[1,8] [1,12] [1,24] [1,45] [1,99]

In the time intervals [1, 8] and [1, 12], the subclasses {3, 2}, {3, 3}, {3, 4}, {3, 5}, {3, 6}, {3 , 7}, {3, 8}, {3, 9}, {3, 15}, {3, 20}, {3, 25}, {3, 30}, {3, 40} and {3, 62}, showed that 100% of the scenarios tested in the ESUFI (10 scenarios / subclasses) had the objective function of optimization.

In the time interval [1, 24] subclasses {3, 2}, {3, 3}, {3, 4}, {3, 5}, {3, 6}, {3, 7}, {3 , 8}, {3, 9}, {3, 15}, {3, 20} {3, 25} {3, 30} {3, 40} and showed that 100% of the scenarios tested in to ESUFI (10 scenarios / subclasses) had the objective function of optimization, since the subclass {3, 10} presented 90%.

In the time interval [1.45] subclasses {3, 2}, {3, 3}, {3, 4}, {3, 5}, {3, 6}, {3, 7}, {3 , 8}, {3, 9}, {3, 15} {3, 20} {3, 25} {3, 30} and {3, 40}, showed that 100% of the scenarios tested in the ESUFI (10 scenarios / subclasses) had the objective function of optimization, since the subclasses {3, 10}, {3, 62} showed 90%.

In the time interval [1, 99] subclasses {3, 2}, {3, 3}, {3, 4}, {3, 5}, {3, 6}, {3, 7}, {3, 8}, {3, 9}, {3, 15} {3, 20} {3, 25} {3, 30} and {3, 40}, showed that 100% of the scenarios tested in the ESUFI (10 scenarios / subclasses) had the objective function of optimization, since the subclasses {3, 10} and {3, 62} showed 80% and 90%, respectively.

In general, the results of tests $\%S_{\mathrm{ESUFI}}$ of Class 3, show that 93,33% of the scenarios tested in the ESUFI showed 100% response (optimization of the objective function), 5,33% had 90% response and 1,33 % showed 80% of responses.

Thus, it appears that although the literature review we highlight the work of Mixed Integer Linear Programming to the problem of programming for Batch Production Systems (FLOW SHOP) with the use of heuristics, this study was focused to solve a mathematical model using Excel and VBA, which presented good results computing, as in all scenarios analyzed were solved and obtained an optimal solution with a good computational time.

6.   CONCLUSION

With the completion of this study, focused on the development of the Microsoft Excel spreadsheet with friendly interface, it was observed that this spreadsheet could be used to minimize the Makespan in the environment of production scheduling in batch processes.

During the research, a primary study was conducted on the production scheduling, where it became clear the vast amount of information involved in this area. Thus, for the specific case of using the ESUFI in the sequencing of production orders for minimization of the Makespan, which is an activity involved in the production scheduling, the ESUFI played a very important role in the production scheduling, since spreadsheets are known by most computer users and are available in almost all Office packages.

The ease of manipulation of the ESUFI is quite evident, as with the use of VBA language a user-friendly interface could be created between the user and the spreadsheet itself. To generate the production scheduling that minimizes the total time of completion of tasks, the only information needed in the worksheet is the number of tasks and resources and the processing times of the tasks in their respective resources. However, the ESUFI had some limitations.

In carrying out the validation tests of the macros used to automate the use of the Microsoft Excel using Visual Basic Application (VBA), a limitation of the spreadsheet was found concerning the amount of adjustable cells, that is, as the number of tasks to be programmed in the spreadsheet increases, the number of resources decreases. Thus, production scheduling is limited to an amount of up to 13 tasks, and therefore this shows that within standard Excel, the present ESUFI can be used for small-scale problems.

As a suggestion, it is important to develop studies applied to real cases, as well to use a professional Excel version to increase the number of adjustable cells, which would extend the limit of variables in the production scheduling and therefore permit working with larger problems.

Although the literature review found many studies have been related to the topic in question by applying heuristics for solving the problem of production scheduling, it appears from this study that the application by means of mathematical formulation is also a viable alternative for small problems size, since in these cases there was obtained a good computational performance.

REFERENCES

ASHOUR, S. (1972). Sequencing Theory. **Springer-Verlag**, Berlin.

BAKER, K. R. (1974). Introduction to Sequencing and Scheduling. **NEW York: John Wiley & Sons, Inc.**

BELLMAN, R. (1956) Mathematical Aspects of Scheduling Theory. **Journal of Society of Industrial and Applied Mathematics** n. 4,p. 168–205.

BELLMAN, R.; ESOGBUE, A.O., NABESHIMA, I. (1982). Mathematical Aspects of Scheduling and Applications. **Pergamon Press**, New York.

BELLMAN, R.; GROSS, O. (1954). Some combinatorial problems arising in the theory of multi-stage processes. **Journal of Society of Industrial and Applied Mathematics** n. 2, p. 175–183.

BOIKO, T. J. P. (2008) Métodos Heurísticos para a Programação em Flow Shop Permutacional com Tempos de Setup Separados dos Tempos de Processamento e Lndependentes da Seqüência de Tarefas. 207 f. **Dissertation (Master Degree in Production Engineering)** - São Carlos Engineering School, São Paulo University - USP, São Carlos.

BROWN, A. P. G.; LOMNICKI, Z. A. (1966). Some Applications of the Branch and Bound Algorithm to the Machine Scheduling Problem. **Operational Research Quarterly** n. 17, p. 173–186.

BROWN, D. E.; SCHERER, W.T. (Eds.). (1995). Intelligent Scheduling Systems. **Kluwer, Boston**.

BURKARD , R. E.; HATZL, J. (2006). A Complex Time Based Construction Heuristic for Batch Scheduling Problems in the Chemical Industry. **European Journal of Operational Research.** n. 174, p. 1162–1183.

BYRD, J., MOORE, T. (1983). Decision Models for Managers. **McGraw Hill**, New York.

CATALÁ, L. P.; DURAND, G. A.; BLANCO, A. M.; ALBERTO B. J. (2013). Mathematical Model for Strategic Planning Optimization in the Pome Fruit Industry. **Agricultural Systems**, v. 115, p. 63-71.

CHENG, T. C. E.; GUPTA, J. N. D.; WANG, G. (2000). A Review of Flow shop Scheduling Research With Setup Times. **Production and Operations Management** n. 9, p. 283–302.

CONWAY, R. L.; MAXWELL, W. L.; MILLER, L. W. (1967). Theory of Scheduling. **Addison Wesley, Reading**, MA.

DUDEK, R. A.; TEUTON Jr., O. F. (1964). Development of M-Stage Decision Rule for Scheduling n Jobs Through M Machines. **Operations Research** n. 12, p. 471–497.

FUMERO, Y.; CORSANO, G. (2013). MONTAGNA, Jorge M. A Mixed Integer Linear Programming Model for Simultaneous Design and Scheduling of Flow shop Plant. **Applied Mathematical Modelling** n. 37, p. 1652–1664.

GROSSMANN, I. E. (2012). Advances in Mathematical Programming Models for Enterprise-Wide Optimization. **Computers and Chemical Engineering**. n. 47, p. 2 – 18.

GUPTA, J. N. D. (1971). An Improved Combinatorial Algorithm for the Flow shop Scheduling Problem. **Operations Research.** n. 19, p. 1753–1758.

GUPTA, J. N. D. (1972). Optimal Scheduling in a Multi-Stage Flow shop. **AIIE Transactions.** n. 4, p. 238–243.

GUPTA, J. N. D.; STAffORD Jr.; Edward F. (2006). Flow shop Scheduling Research After Five Decades. **European Journal of Operational Research.** n. 169, p. 699–711.

IGNALL, E.; SCHRAGE, L. (1965). Application of Branch-and-Bound Technique to Some Flow Shop Problems. **Operations Research**. n. 13, p. 400–412.

JOHNSON, S. M. (1954). Optimal Two- and Three-Stage Production Schedules With Setup Times Included. **Naval Research logistics Quarterly**. n. 1, p. 61–68.

KONE´, O.; ARTIGUES, C.; LOPEZ, P.; MONGEAU, M. (2013). Comparison of Mixed Integer Linear Programming Models for the Resource-Constrained Project Scheduling Problem With Consumption and Production of Resources. **Flex Serv Manuf J**. n. 25, p. 25–47.

LIN, R.; LIAO, C-J. (2012). A Case Study of Batch Scheduling for an Assembly Shop. **Int. J. Production Economics**. n. 139, p. 473–483.

LOMNICKI, Z. A. (1965). A Branch and Bound Algorithm for the Exact Solution of the Three Machine Scheduling Problem. **Operational Research Quarterly**. n. 16, p. 89–100.

LUNDGREN, M. G. T.; LUNDGREN , J. T.; PERSSON, J. A. (2002). An Optimization Model for Refinery Production Scheduling. **Int. J. Production Economics.** n. 78, p. 255 – 270.

MACCARTHY, B. L.; LIU, J. Y. (1993) Addressing the Gap in Scheduling Research: A Review of Optimization and Heuristic Methods in Production Scheduling. **International Journal of Production Research**, London, n. 31, p. 59-79.

MANNE, A. (1960). On The Job-Shop Scheduling Problem. **Operations Research**. n. 8, p. 219–223.

MCMAHON, G. B., BURTON, P. G. (1967). Flow shop Scheduling With The Branch and Bound Method. **Operations Research**. n. 15, p. 473–481.

MENDÉZ, C. A. et al. (2006) State-of-the-art Review of Optimization Methods for Short-term Scheduling of Batch Processes. **Computers and Chemical Engineering**, n. 30, p. 913-946.

MOCCELLIN, J. V. (1995). A New Heuristic Method for the Permutation Flow Shop Scheduling Problem. **Journal of the Operational Research Society**. Oxford, n. 46, p. 883-886.

MOREIRA, D. A. (2010). Pesquisa Operacional: Curso Introdutório. 2ª Ed. São Paulo: **Cengage Learning**.

MUTH, J., THOMPSON, G. L., (Eds.). (1963). Industrial Scheduling. **Prentice-Hall, Englewood Cliffs**, N.J.

OSMAN, I. H., KELLY, J. P. (1996). Meta-Heuristics: Theory and Applications. **Kluwer Academic Publishers**, Boston.

PAN, M.; QIAN, Y.; LI, X. (2008). A Novel Precedence-Based and Heuristic Approach for Short-Term Scheduling of Multipurpose Batch Plants. **Chemical Engineering Science.** n. 63, p. 4313 – 4332.

POLON, P. E. (2010). Otimização da Produção da Indústria de Embutidos, PEQ/UEM, Maringá- PR (**Thesis**), 115p. (Doctorate in Chemical Engineering) – Chemical Engineering Graduate Program, State University of Maringá.

POLON, P. E.; ANDRADE, C. M. G.; PARAÍSO, P. R.; JORGE, L. M. M. (2006). Utilização de Planilha Eletrônica na Resolução de Problemas de Planejamento e Programação da Produção. **Anais**. XXXIV COBENGE, Congresso Brasileiro de Ensino em Engenharia. Passo Fundo: Ed. Passo Fundo University, September.

POTTS, C. N., KOVALYOV, M. Y. (2000). Scheduling with batching: A review. **European Journal of Operational Research**. n. 120, p. 228–249.

POTTS, C. N., VAN WASSENHOVE, L. N. (1992). Integrating scheduling with batching and lot sizing: A review of algorithms and complexity. **Journal of the Operational Research Society**. n. 43, p. 395–406.

PRIVITERA, G.; DAY, A. R.; DHESIC, G.; LONG, D. (2011). Optimising the Installation Costs of Renewable Energy Technologies in Buildings: A Linear Programming Approach. **Energy and Buildings.** n. 43, p. 838–843.

REKLAITIS, G.V. (1995) Scheduling Approaches for the Batch Process Industries. **ISA Transactions**, n. 34, p. 349 – 358.

RUIZ, R.; MAROTO, C.; ALCARAZ, J. (2005). Solving the Flow shop Scheduling Problem With Sequence Dependent Setup Times Using Advanced Metaheuristics. **European Journal of Operational Research**. n. 165, p. 34–54.

SEVERO, L. S. (2007). Aplicação de Modelo de Programação da Produção na Indústria de Couros, PEQ/UFRGS, Porto Alegre- RS (**Dissertation**), 96p. (Master Degree in Chemical Engineering) – Chemical Engineering Graduate Program, Federal University of Rio Grande do Sul.

SHAKERI, S.; LOGENDRAN, R. (2007). A Mathematical Programming-Based Scheduling Framework for Multitasking Environments. **European Journal of Operational Research**. n. 176, p. 193–209.

SISSON, R. L. (1959). Methods of Sequencing in Job Shops—a Review. **Operations Research** 7.

SOLIMANPUR, M.; ELMI, A. (2013). A Tabu Search Approach for Cell Scheduling Problem With Makespan Criterion. **Int. J. Production Economics.** n. 141, p. 639–645.

SZWARC, W. (1971). Elimination Methods in the m · n Sequencing Problem. **Naval Research Logistics Quarterly**, p. 295–305.

SZWARC, W. (1973). Optimal Elimination Methods in the m · n Flow shop Scheduling Problem. **Operations Research**. n. 21, p. 1250–1259.

T_KINDT, V.; BILLAUT, J-C. (1993). Multi-criteria Scheduling. **Springer-Verlag**, Berlin.

TAILLARD, E. (1993). Benchmarks for Basic Scheduling Problems. **European Journal of Operational Research**, Amsterdam, n. 64, p. 278-285.

TANAEV, V. S.; SOTSKOV, Y. N.; STRUSEVICH, V. A. (1994). Scheduling Theory: Multi-stage Systems. **Kluwer Academic Publishers**, Dordrecht.

TSENG, F. T., STAFFORD, E. F., GUPTA, J. N. D. (2004). An Empirical Analysis of the Integer Programming Formulations for the Permutation Flow shop. OMEGA: **The International Journal of Management Science**. n. 32, p. 285– 293.

WAGNER, H. M. (1959). An Integer Linear-Programming Model for Machine Scheduling. **Naval Research Logistics Quarterly**. n. 6, p. 131–140.

ZWEBEN, M., FOX, M. S. (1994). Intelligent Scheduling. **Morgan Kauffman Publishers**, San Francisco.