

# COLLABORATIVE PARADIGM FOR SINGLE-MACHINE SCHEDULING UNDER JUST-IN-TIME PRINCIPLES: TOTAL HOLDING-TARDINESS COST PROBLEM

Maria L.R. Varela<sup>1</sup>, Ana M. Madureira<sup>2</sup>, Joana D. Dantas<sup>1</sup>, André S. Santos<sup>1</sup>, Goran D. Putnik<sup>1</sup>, Justyna Trojanowska<sup>3</sup>, José Machado<sup>4</sup>

<sup>1</sup> University of Minho, School of Engineering, Department of Production and Systems, Portugal

<sup>2</sup> Polytechnic Institute of Porto, School of Engineering, GECAD Research Group, Portugal

<sup>3</sup> Poznan University of Technology, Chair of Management and Production Engineering, Poland

<sup>4</sup> University of Minho, School of Engineering, Department of Mechanical Engineering, Portugal

## Corresponding author:

Maria Leonilde Rocha Varela

University of Minho

Department of Production and Systems

Azurem Campus, 4804-533, Guimaraes, Portugal

phone: +351 253510765

e-mail: leonilde@dps.uminho.pt

---

Received: 6 August 2017

Accepted: 9 March 2018

## ABSTRACT

The problem of sequencing jobs on a single machine to minimize total cost (earliness and tardiness) is nowadays not just important due to traditional concerns but also due to its importance in the context of Collaborative Networked Organizations and Virtual Enterprises, where precision about promptly responses to customers' requests, along with other important requirements, assume a crucial role. In order to provide a contribution in this direction, in this paper the authors contribute with an applied constructive heuristics that tries to find appropriate solutions for single machine scheduling problems under different processing times and due dates, and without preemption allowed. In this paper, two different approaches for single-machine scheduling problems, based on external and internal performance measures are applied to the problem and a comparative analysis is performed. Computational results are presented for the problem under Just-in-Time and agile conditions on which each job has a due date, and the objective is to minimize the sum of holding costs for jobs completed before their due date and tardiness costs for jobs completed after their due date. Additional computational tests were developed based on different customer and enterprise oriented performance criteria, although preference is given to customer-oriented measures, namely the total number of tardy jobs and the maximum tardiness.

## KEYWORDS

single-machine, scheduling, heuristics, just-in-time, agile manufacturing, holding-tardiness cost problem.

---

## Introduction

Manufacturing scheduling (MS) consists on an important production management function, affecting all branches of industrial, commercial and services organizations. In the strict industrial context, a very important objective, which is usually imposed by the clients consists on satisfying due dates for manufacturing orders.

There are many situations on real-world problems involving a single machine resource [1]. This kind of production scheduling problem is also a base problem

usually treated in the context of more complex manufacturing environments, for instance in job shops, for solving bottleneck problems [2–6]. Therefore, single machine scheduling problems are also important while trying to solve more complex ones, as the solution for the whole complex scheduling problem can be individually found through a decomposition that turns it easier to be solved [4, 7–10] and then integrated in the global solution.

Moreover, single machine scheduling problems also remain quite promising and interesting scheduling problems to be explored due to its great potential for

dealing with several distinct performance measures. Performance measures can be generally classified as external or customer-oriented (measures related to the client, such as lead times or due dates) or internal or enterprise-oriented (such as, makespan, total completion time, and earliness among others). Both kind of performance measures are important to be considered, and can be treated jointly [11], under the principles of the just-in-time (JIT) philosophy. In this context, although special attention should be given to the customer-oriented performance measures, internal interests have also to be taken into account.

The applied heuristics consider several performance measures, which intend to be customer and enterprise oriented. Customer oriented performance measures are mainly related to the accomplishment of due dates [12, 13] while enterprise-oriented typically consider time-oriented measures, like makespan.

In this paper, two different approaches for single-machine scheduling problems, based on external and internal performance measures are applied to the problem and a comparative analysis is performed.

In order to clearly refer to the main subjects underlying to this work, this paper is organized as follows. Section 2 presents a summarized overview about studies that have been carried out about single machine scheduling, regarding regular internal and external performance measures. Moreover, a very brief overview about non-regular performance measures is also presented. Section 3 describes the scheduling problem focused in this work. Section 4 briefly describes several different scheduling approaches, including the algorithms used in this work, namely Total holding-tardiness cost for Single-Machine Scheduling Problems (SMSPP). Section 5 presents the results obtained and a comparative analysis based on obtained results from the literature for the same problem. Finally, Sec. 6 presents some conclusions and future work.

## State of the art

Many studies have been carried out over last decades about single machine scheduling. In these problems not only the assumptions and constraints can be different, but also the objective function that is intended to be analyzed, which can led to quite difficult problems to solve [14, 15, 21, 25–28, 47–50].

### Client oriented performance measures

The works described in [14–18] are focused on the client, as the objective of both is to minimizing maximum

lateness, although the assumptions and specificities of the problem considered on each one are different. The first one includes random processing times, deterministic due dates and independent setup times and the problem treated in [7] is more complex as it does include release dates and precedence constraints.

In [14], the authors apply a Branch-and-Bound (B&B) and a Simulated Annealing (SA) meta-heuristic. The last one produces solutions which are close to the optimal solutions found through the branch-and-bound approach. Also in [15] a B&B approach is put forward, based on the use of four different heuristics. Just like [14], the work in [16] applies a meta-heuristic SA to solve a problem with family setups.

The studies in described [17–19] have unusual specificities. In [17] exist a unexpected arrival of new jobs and it is applied a NST to solve the problem of re-scheduling. The problem treated in [18] considers the learning effect. The processing time decreases with the system learning and two heuristics have been proposed to solve the problem. The work described in [19] applies a new mathematical model to minimize the maximum lateness in a case with deteriorating jobs.

The objective of the works [20, 21] is the minimization of total weighted tardiness and both have sequence dependent setup times, however in the first one it is applied a local search heuristic and in [21] is proposed a hybrid algorithm based on SA and a Genetic Algorithm (GA).

### Enterprise oriented performance measures

The performance measures for an enterprise-oriented analysis can be diversified. If the objective is related to the stock reduction, the objective can be related to the earliness and its minimization, like in [22]. In [23, 24], the objective is to minimize the total flow time: in [23] is creating a sequence of job families and of jobs within each family, through a SA-based algorithm, and [24] is based on a problem with batch scheduling. Some heuristics to minimize the mean completion time were proposed, in an environment where different ready times are considered [25]. The heuristics applied with some pre-defined conditions provide optimal sequences in very short execution times.

In [26, 27], the objective is to minimize the makespan and both have in consideration the periodic maintenance. Two problems are studied in [28]. The first one is divided in two steps, production and delivery of one job and the second one consider the periodic maintenance like in [26, 27].

Other contributions use non-regular performance measures, namely the work in [29], which studies the completion time variance and the work in [30] analyzed a variance minimization problem in general. The non-regular performance measures are earning a growing importance in modern industrial systems [29, 30].

### Balanced performance measures

In other studies, we can find the balance of both, internal and external performance measures, like the work presented in [33–46]. They are focused on the Just-in Time philosophy, trying to minimize the earliness and the lateness or tardiness at the same time. In the beginning most of scheduling research are focused on single-criteria problems, being very developed, like we can see in [31] and [32]. However, organizations are concerned with the satisfaction of more than one group of people. It is necessary that clients and enterprise are both pleased.

In [33, 34] the objective is to minimize the total weighted earliness and de number of tardy jobs and the work in [35, 36] have the same objective (minimize the total weighted earliness) but subject to no tardy jobs.

In [37–42] the balance between earliness and tardiness are made by a single performance measure criterion. In [37] the objective is to minimize the sum of the weighted earliness and tardiness with different weights and [38] is similar to the last one, being the objective minimizing the sum of early and tardy penalties. The study of [39] try to minimize the costs associated to early and tardy jobs, that is similar to [40], that minimize earliness and tardiness penalties. In [41] the objective is to minimize the deviation of jobs completion times around their due dates and [42] is exposed an objective function that includes earliness, tardiness, due dates, makespan and total resource consume costs.

In [43] is proposed a different approach that take the bi-criteria situation that we found in [33–36], with the minimizing of the tardy jobs and the integration of earliness and tardiness in one irregular performance measure like in [37–42], minimizing the number of early/tardy jobs.

However, the relation between clients and enterprises may not always be balanced by the earliness and tardiness performance measures, like we can see in [44–46]. In [44] are analyzed two performance measures to satisfy both criteria. The minimization of the number of tardy jobs is used to satisfy the clients and the minimization of the maximum weighted tardiness is to minimize the penalizations that the enterprise suffers with that. In [45], the objective is to minimize

the total tardiness (client interest) and minimize the WIP (work-in-progress) costs (enterprise interest).

The work reported in [46] is intended to follow the JIT principles with a more complete approach. It had earliness/tardiness penalties, interruption penalties and holding costs of jobs.

In this paper, a special attention will be given to the work presented in [33], as we will consider the resolution of the same base problem and will carry out a comparative analysis, based on their results, although they just explore two performance measures and in our work we intend to further analyze some other closely related ones, as we are going to describe further in Sec. 3.

### Case study

The scheduling problem considered in this study consists on a single machine scheduling problem with  $n$  jobs to be processed in order to follow, as close as possible, a JIT strategy. This strategy intends to minimize the number of tardy jobs, while also being concerned about the minimization of the maximum earliness and the maximum tardiness, among other more secondary performance measures, which are also very important in scheduling approaches, in general, and in JIT-based approaches in particular. Such additional performance measures are related to the maximum completion time of jobs or makespan ( $C_{\max}$ ) and the total completion time of jobs ( $\sum C_i$ ), which although being internal or enterprise-oriented performance measures, as stated before, are also very important ones.

Therefore, we still have to pay attention to them too, as the existence of a balanced approach, which intends to reach solutions that try to obtain equilibrated solutions in terms of external and internal objectives is of utmost importance, and special attention should be given to this balance, which is even more important due to the current economic crisis that we are still experiencing.

### Problem definition

The single machine scheduling problem studied will follow some assumptions:

- all the jobs are available at the same time zero,
- the machine can process at most one job at a time,
- no preemption of jobs is allowed,
- associated to each job  $j$  ( $j = 1, 2, \dots, n$ ) there is a processing time  $p_j$  and a due date  $d_j$ .

Therefore, the problem can be classified as  $1|p_j, d_j|Nt, T_{\max}, C_{\max},$  and  $E_{\max}$ , and also  $(\sum T_i$  and  $T_{\text{mean}}, \sum C_i$  and  $C_{\text{mean}}, \sum E_i$  and  $E_{\text{mean}})$ , according to  $\alpha|\beta|\gamma$  Graham nomenclature [53].

### Dominance properties/objectives

In this paper, we will present a comparative analysis based on the different performance measures referred above, which are either associated to external or internal objectives:

- $N_t$  – number of tardy jobs,
- $T_{\max}$  – maximum tardiness,
- $C_{\max}$  – maximum completion time,
- $E_{\max}$  – maximum earliness,
- $\sum T_j, T_{\text{mean}}$  – total and mean tardiness,
- $\sum C_j, C_{\text{mean}}$  – total and mean completion time,
- $\sum E_j, E_{\text{mean}}$  – total and mean earliness.

We intend to analyze all this different performance measures trying to reach a balance between them and come across a good or at least a satisfactory solution to the analyzed problem under the scope of the proposed JIT principles.

### Total holding-tardiness cost for SMSP

A diversity of problems involving sequencing and scheduling of jobs are naturally formulated as integer programs with variables indexed by  $(i, t)$  where  $i$  denotes a job to be processed and  $t$  is a time period. The problem of sequencing jobs on a single machine to minimize total cost is considered. Machine capacity constraints require that, at any time, at most one job is processed. Also, no machine idle-time between processing jobs is allowed.

Total holding-tardiness cost (HT):  $\min \sum g_j(t_j)$ , with  $g_j(t_j) = h_j * \max \{d_j - t_j, 0\} + w_j * \max \{t - d_j, 0\}$ , where  $h_j$  is a holding cost per time unit incurred if the job is completed before  $d_j$  and  $w_j$  is a tardiness cost per time unit in late (see Abdul-Razaq and Potts [47]).

When the cost of storage  $h_j$  is null for all tasks, the problem “holding-tardiness” is reduced to the problem of minimization of weighted tardiness [47].

This class of scheduling problem implies the philosophy of “Just-in-Time” (JIT), and in a simplified form can be expressed as follows: “to produce only what is needed when it is necessary”. The decrease in stocks at all stages of production (raw materials, products in work in progress and finished products), and the reduction of delivery times of products are some of the JIT objectives. On the one hand, the reduction in storage costs is achieved by reducing stock levels, on the other, the reduction of delivery times is achieved by the reduction and/or elimination of delays.

These problems are NP-hard (see Madureira [49]) and are sometimes solved to optimality, through Branch and Bound algorithms or Dynamic Programming techniques. In Abdul-Razaq and Potts [47] a formulation for dynamic programming (the relax-

ation method the state space) for this problem is presented. In this case are two methods to improve lower limit: the application of penalties in the objective function or use of the relaxation of state space. The algorithm of Branch and Bound resulting advantageous in great trouble finding the optimal solution. In Sydney [48] it had been already presented an algorithm to solve this class of scheduling problem and consists of two phases: the ordination of tasks (sequence tasks simultaneously in order not decreasing the launch dates and delivery dates) and definition of optimum start time. Additional and more recent references contributions to this class of problems could be found in [49].

### Scheduling optimization approaches and algorithms

Most scheduling problems can be solved optimally, provided that enough processing time is available. Often, they can be formulated as a (mixed) integer programming problem. Due to the NP-completeness nature that these kind of problems usually present, complete enumeration of all possibilities quickly becomes prohibitively time consuming [50, 54].

However, in some situations, efficient B&B methods, Dynamic Programming techniques, and zero one binary integer programming may provide optimal solutions for reasonable (and usable) problem sizes in reasonable time [55]. In [30] are presented several of these and other more complex scheduling approaches and systems, namely for real-time, distributed and dynamic scheduling, which also can be even more complex scenarios when some process for dealing with uncertainty is added [3, 4, 8, 10, 44–61].

If the processing time required for obtaining optimal solutions is beyond reasonable bounds, it is necessary to use heuristic solutions, like dispatching procedures (which usually do not yield optimal solutions). Dispatching procedures with priority rules is the best known heuristic for scheduling and is surveyed in detail in [55]. Using dispatching rules based schedules can be produced almost instantaneously. Because of their simplicity, they are also used quite often in industry and in flexible manufacturing and assembly systems. Dispatching rules, however tend to yield a low and unpredictable performance.

For instance, the Earliest Due Date rule (EDD) is often used for Job-Shop Scheduling (JSS) Artificial Intelligence (AI) Approaches [2, 4, 5, 8–10, 56–61].

For some simple scheduling problems, efficient algorithms exist to calculate the optimal solutions [2, 14, 15, 23, 33, 49].



For instance, the EDD dispatching rule provides an optimal solution for the minimization of maximum tardiness for the single machine scheduling problem. However, this is not the only one, the Shortest Processing Time dispatching rule (SPT), in a single machine problem, lead to a minimal mean completion time and mean lateness. This rule sequenced the jobs in a increasing order of processing times [32].

Usually, real-life problems are more complex than these academic problems, but sometimes, by using a simplified model and applying the results to the real situation, the scheduling problem can be successfully solved (e.g. scheduling a factory with a single bottleneck as a single processor scheduling problem [2, 3, 8]).

AI-based approaches to scheduling, sometimes known as just knowledge-based scheduling, use specific knowledge about the scheduling problem in their decision process [4, 5, 9, 10, 57]. For instance, rule-based approaches store scheduling rules (entered by humans) into a knowledge base and some web-based systems for supporting manufacturing scheduling decision making exist, which enable to solve quite different scheduling problems, varying from single-machine environments up to more complex ones, like job shops, manufacturing cells or other kind of flexible manufacturing systems [59].

Usually near-optimal approaches enable to obtain a higher performance than simple dispatching rules, but they also use a considerable amount of processing time [31]. Therefore, the selection of a scheduling algorithm depends on a trade-off between processing time and schedule performance.

In this paper, was used a software to obtain the results set out in Sec. 5. The used software includes two different approaches, a Hodgson’s algorithm adaptation (HA) and a Neighbourhood Search Technique (NST) which will be described in the next sections.

**Hodgson algorithm adaptation (HA)**

The Hodgson algorithm determines the sequence of tasks whose number of tardy jobs is minimum [29, 30, 33, 37, 44]. Let  $E$  be the set that contains all the jobs that must be processed and  $L$  the set of late jobs, which starts being empty. Thus, the algorithm integrates the sequence of steps described on Table 1.

There are several priority measures, which can be used for establishing an initial solution as a starting point for the Hodgson’s algorithm, which will act as a rule that specifies the priority on how the jobs present in the waiting queue of a machine are processed. The Earliest Due Date (EDD) rule speci-

fies that when a machine is free, is selected the entity or job that has the earliest due date to be processed first.

Table 1  
Hodgson algorithm.

Step 1
Sort the jobs that belong to the $E$ set and sort them by increasing delivery date (earliest due date rule, EDD).
Step 2
If none of the jobs is delayed the sequence is optimal. If the opposite happens, the delayed job at its $k$ position (i.e. $[k]$ ) is identified.
Step 3
Identify the entity or job with the bigger duration or processing time ( $p_j$ ) in the set of the first $k$ jobs. Remove it from the set $E$ and put it into the $L$ set. Establish the new time for the conclusion of the remaining jobs on $E$ and return to step 2.

In this work was used a Hodgson’s algorithm Adaptation (HA) implementation, using JAVA 5 language [59].

As described before, the Hodgson’s algorithm (and HA) ensures the sequence with the least number of delays or tardy jobs in a single machine environment [31].

**Neighbourhood Search Techniques (NST)**

The Neighbourhood Search Techniques is a searching technique included in the scope of heuristic approaches that perform a local search for searching for some local optimal solution (or local optimum); and they usually are fast, simple to implement and flexible [31, 59]. Associated to this kind of methods, it turns out to be necessary to specify a mechanism, to create an initial solution or seed, which consists upon a neighbourhood generation mechanism, and also a criterion for selecting the next seed and a stopping criterion [31, 59]. For some problems, efficient NST-based algorithms do exist to find the optimal solution [31, 59]. For most problems, optimal algorithms exist, but may require excessive processing time [31]. If the optimal solution cannot be found in reasonable time, heuristics or meta-heuristics should be used [31, 59]. The simplest heuristics are based on simple dispatching rules [31, 59]. Several other heuristics are based on techniques developed in AI [2, 4, 5, 9, 10, 31, 56–60]. Neighbourhood search techniques, and particularly meta-heuristics, have also provided some powerful algorithms for near-optimal schedules [31].

The NST procedure used in this work includes the steps described on Table 2 [59].

Table 2  
NST procedure.

Step 1
Select a seed solution for evaluating its performance.
Step 2
Generate and evaluate solutions in the neighbourhood of the seed. If none of the solutions produces better performance than the seed, then the search is concluded. Otherwise continue.
Step 3
Select the solution from the neighbourhood with better performance for being the new seed.

Moreover, it is necessary to specify [57]:

1. A method for obtaining a seed.
2. A certain mechanism for generating the neighbourhood.
3. A method for selecting the solution that will be the next seed.

The HA and NST algorithms implementations used in this work enable to obtain two different sequences of jobs, for the problem under consideration, that correspond to quite good solutions, which are going to be described further in the next section.

### Computational study

The comparison to be carried out is based on the results obtained through the use of the dispatching rule SPT and HA and NST algorithms mentioned above, and by the ones obtained from the sample's source paper [33]. Obtained results are summarized in Table 3.

Table 3  
Problem data.

<i>j</i>	1	2	3	4	5
<i>p<sub>j</sub></i>	3	7	5	6	10
<i>d<sub>j</sub></i>	6	8	10	12	25

The work presented in this paper is based on a previous work [48] with an instance of five jobs for the problem presented in Sec. 3. In this paper we intend to generalize that work. To generalize the problem analysis, we apply the same two methods, NST and HA and the dispatching rule SPT, to fifty-six dimensions increasing the number of jobs, being from five to ninety nine.

The evolution of problems results is a linear growth, for most of the performance measures. The results obtained for Hodgson Algorithm application appear to be the more effective for some of the performance measures, like minimizing the Total Tardiness and the Earliness costs, along with the total and Mean Earliness. However the Mean Earliness is on-

ly better for large problems with more than twenty jobs.

The NST and SPT application to these problems reached better results for the Total and Mean Tardiness and Completion Time and the Mean Earliness did perform better for small problems with less than twenty jobs.

For all methods, the maximum completion time is the same, being the minimum for the respective size of problem. This is because none of them introduce idle times between jobs for a minimized earliness, producing a sequence of jobs without any forced interruption.

### Problem data

The problem instance considered includes the jobs' data stated in Table 3, based on the example 1 defined in [33].

### Computational results

In Table 4, it is possible to verify that the SPT rule obtain better results in most of the performance measures but did not maintain the minimum number of tardy jobs. The SPT rule compared to the two heuristics applied, this last two did not obtain much higher values to the other performance measures and managed to maintain the minimum two tardy jobs.

Table 4  
Problem results.

Performance measures	Methods				
	Paper solution 1	Paper solution 2	SPT	HA	NST
$N_t$	2	2	3	2	2
$T_{max}$	27	29	13	19	23
$C_{max}$	37	37	31	31	31
$E_{max}$	0	0	3	3	3
$\sum T_j$	51	49	21	36	25
$\sum C_j$	112	110	77	85	92
$\sum E_j$	0	0	5	20	6
$T_{mean}$	10.2	9.8	4.20	4.8	3.8
$C_{mean}$	22.4	22	15.40	17	16
$E_{mean}$	0	0	1	4	2

If we compared the methods applied in this work with those used in the work [33], we can obtain better results in Tardiness with a little increase of the Earliness.

Moreover, through the algorithms used in this work was also possible to reach better results in terms of the other internal performance measures analyzed, associated to the completion time ( $C_{max}$  and  $\sum C_j$ ).

As we can see, through the first Gantt chart (Fig. 1), based on the results presented in [33], the total completion time is higher, due to the existence of gaps between jobs, which the authors did introduce in order to obtain a better solution in terms of minimization of the earliness ( $E_{max}$ ).

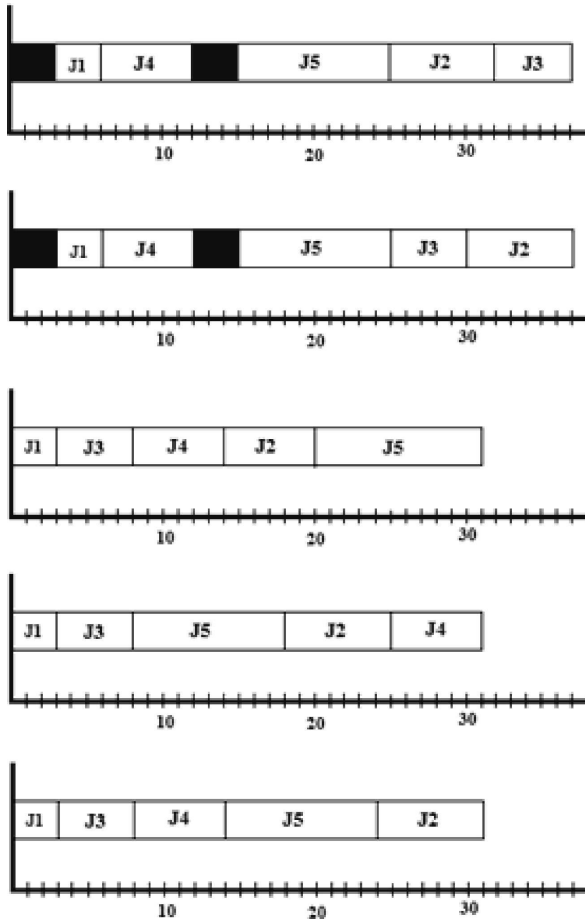


Fig. 1. Gantt charts for each sequence.

These gaps mean that the machine will be unavailable during that time intervals, which, therefore, will imply that the jobs ahead in the sequence will be delivered after the respective due date and, thus, will contribute for a higher  $T_{max}$  value.

Table 5 shows the difference between the  $T_{max}$  and the  $E_{max}$ , and through this difference value, in time units ( $t.u.$ ), we can better analyze the variation between the maximum tardiness and the maximum earliness, for each one of the solutions provided through the algorithms used in this work and the results (1 and 2) presented by the authors in [33].

Moreover, through these results we can also analyze how much each result presented derivate from de ideal solution, which is zero. As we can see none of the results is zero or close to zero, however the

results obtained in this work are visibly better, and specially through the use of the SPT, which is the more effective ( $10t.u.$ ).

Table 5  
Comparative analysis results.

Values in time units ( $t.u.$ )	Methods				
	Paper [10] solution 1	Paper [10] solution 2	SPT	HA	NST
$T_{max} - E_{max}$	27	29	10	22	26
$(C_{mean} - C_{mean}^*)$	7.2	6.8	0	1.8	0.2
$(C_{med} - C_{med}^*) + (T_{max} - E_{max})$	34.2	35.8	10.0	23.8	26.8

Another and even more interesting comparative analysis, in the context of the JIT principles behind this work, which enables also to consider the influence of an internal-oriented performance measure ( $C_{mean}$ ) can be performed based on ( $C_{mean} - C_{mean}^*$ ), which is also presented in Table 5, where  $C_{mean}^*$  is the optimal value for the Mean Completion Time ( $15.2t.u.$ ). And the sum of this two can show in general, and analyzing three different performance measures, how much distant the results can be from the optimal solution. The values presented do also clearly show that the solutions obtained in this work are considerably better than the ones obtained in [33], and considering this comparative analysis measure the best solution was reached by the SPT dispatching rule used in this work ( $10t.u.$ ). This happens because the SPT rule reach the minimum value of mean completion time and has a low maximum tardiness.

### Generalization

To generalize the problem analysis, we apply the same two methods to thirty dimensions, increasing the number of jobs, being from five to ninety nine.

The evolution of problems results has a linear growth, for most of the performance measures.

The results obtained for Hodgson Algorithm application appear to be the most effective to some of the performance measures, like minimizing the Tardiness and Earliness Maximum and Total and Mean Earliness. However the mean earliness is only more effective for large problems with more than twenty jobs.

The NST and SPT application to these problems had better results for the Total and Mean Tardiness and Completion Time and the Mean Earliness for small problems with less than twenty jobs.

We could realize that through all the analyzed methods, the Maximum Completion Time is the

same, being the minimum for the respective problem dimension. This is because none of them apply gaps between jobs to minimize Earliness, having a sequence of jobs without any interruptions.

**Irregular performance measures analysis**

Analysing the three already referred irregular performance measures, we obtain the graphics depicted in Figs. 2, 3 and 4. In the performance measure that includes the Maximum Tardiness and the Maximum Lateness, it is possible to conclude that the methods performance is similar and with the increase of the number of jobs the difference between the Maximum Tardiness and the Maximum Lateness increases.

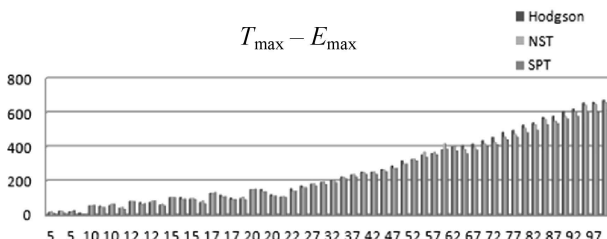


Fig. 2. Graph for the performance measure  $T_{max} - E_{max}$ .

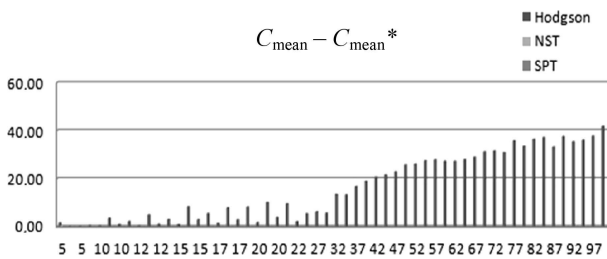


Fig. 3. Graph for the performance measure  $C_{mean} - C_{mean}^*$ .

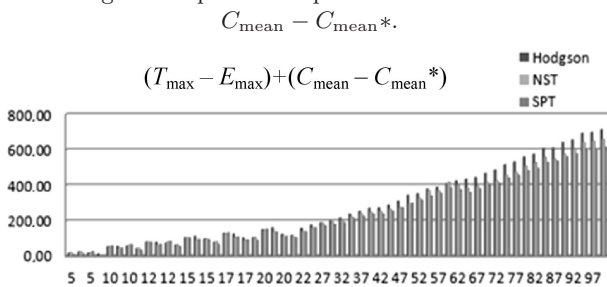


Fig. 4. Graph for the performance measure  $(T_{max} - E_{max}) + (C_{mean} - C_{mean}^*)$ .

In this performance measure, the Hodgson solutions obtain a similar behavior to the previous performance measure, but the SPT performance is null, because this dispatching rule guarantee a minimal mean completion time, therefore the difference between the mean completion time found and its optimal value is null.

The NST performance obtain a null or a very low value to this performance measure which means that the NST applied obtain a optimal value for the Mean

Completion Time or a very near to the optimum, in a very similar way that the application of SPT rule.

For the last performance measures, the methods performance is similar, despite the SPT has been more effective considering the performance to the mean completion time.

**Correlation with the number of jobs in the problem**

We did analyze, with the SPSS software, the correlation between these three performance measures and the number of jobs in the problem for the solutions of both methods, by the Pearson Correlation. For the first performance measure analyzed, we obtain the scatterplots in Fig. 5 to Fig. 7, for the solutions obtain by SPT, HA and NST respectively. In Fig. 5 we see the correlation of the SPT results for the difference between the maximum tardiness and

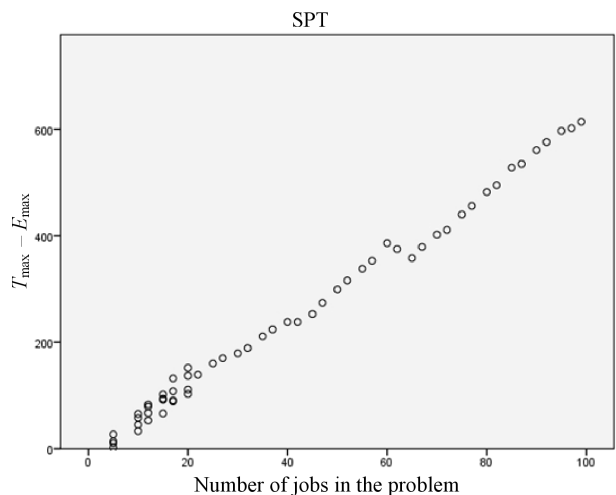


Fig. 5. Scatterplot  $T_{max} - E_{max}$  for SPT solutions.

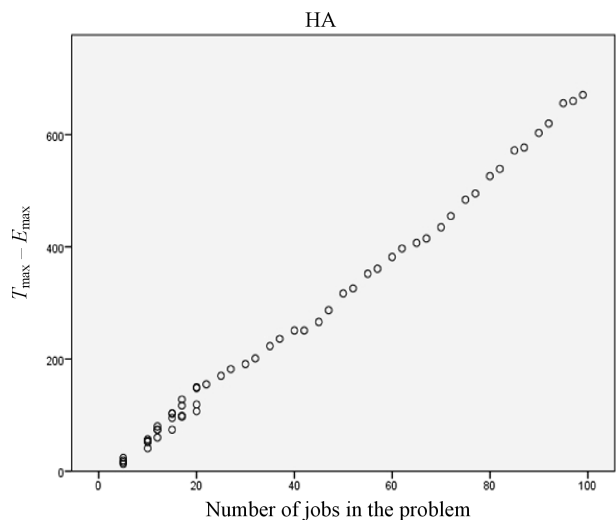


Fig. 6. Scatterplot  $T_{max} - E_{max}$  for HA solutions.



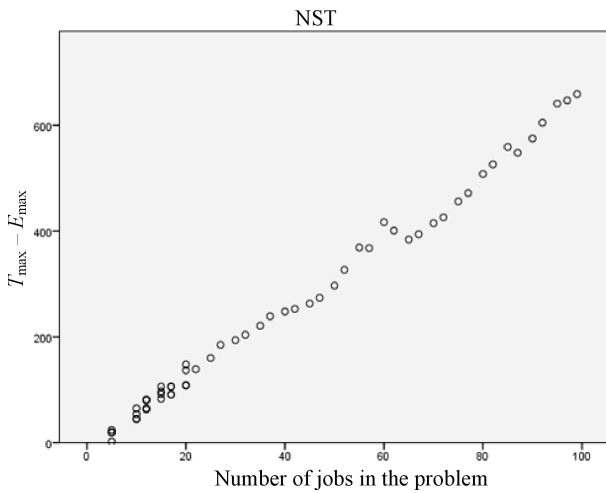


Fig. 7. Scatterplot  $T_{\max} - E_{\max}$  for NST solutions.

maximum earliness and the number of jobs in the problem.

In Fig. 6, we can notice that there is a correlation between the number of jobs and this performance measure for the solutions by Hodgson Algorithms.

For the NST solutions, in Fig. 7, we can see this correlation too.

For the next performance measure, the difference between the methods results mean completion time and the optimal mean completion time, and through this evaluation, we can confirm that the number of jobs in the problem does not affect the mean completion time obtain by SPT and NST application, like we can see in the next two Fig. 8 and 9.

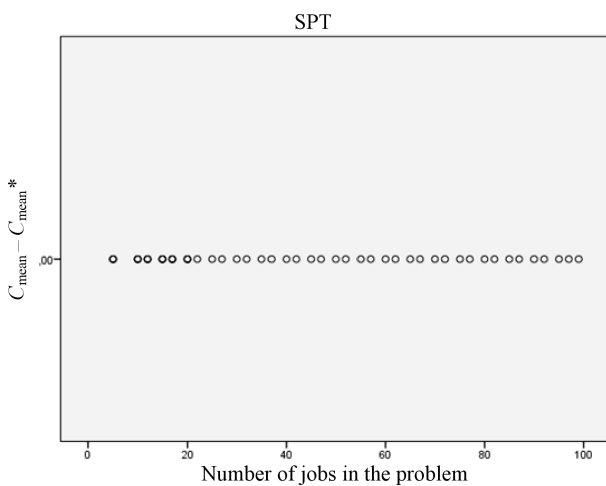


Fig. 8. Scatterplot  $C_{\text{mean}} - C_{\text{mean}}^*$  for SPT solutions.

In Fig. 9, it is evident that there is no correlation between the number of jobs in the problem and the performance measure analyzed.

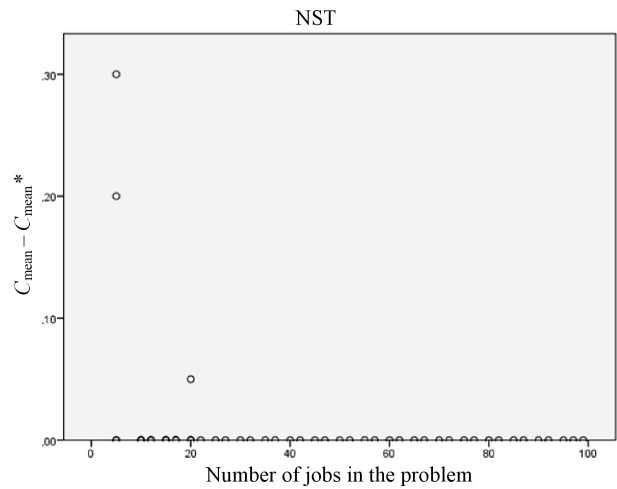


Fig. 9. Scatterplot  $C_{\text{mean}} - C_{\text{mean}}^*$  for NST solutions.

The performance measures of the solutions that result of HA application are presented in Fig. 10, and the correlation is confirmed by the Pearson Correlation test.

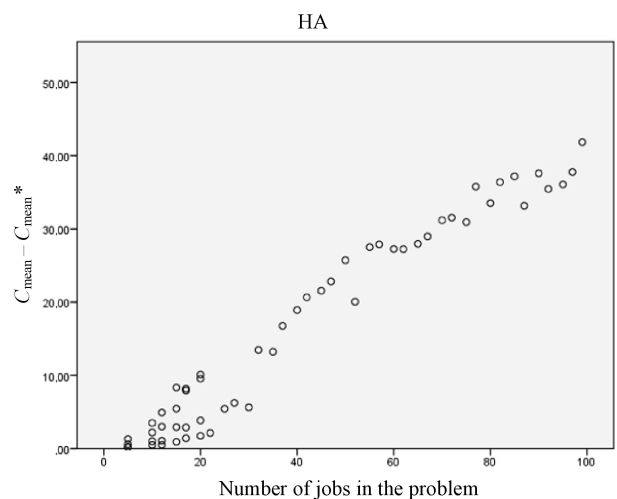


Fig. 10. Scatterplot  $C_{\text{mean}} - C_{\text{mean}}^*$  for HA solutions.

For the last performance measure analyzed, the correlation is proved for both of the methods and it is visible in the next three Figs. 11, 12, and 13.

In all of them we it is possible to analyse the linear growth between the number of jobs and the formula results.

We analyzed, through the use of SPSS, the correlation between these three performance measures and the number of jobs in the problem, by using the Pearson Correlation method.

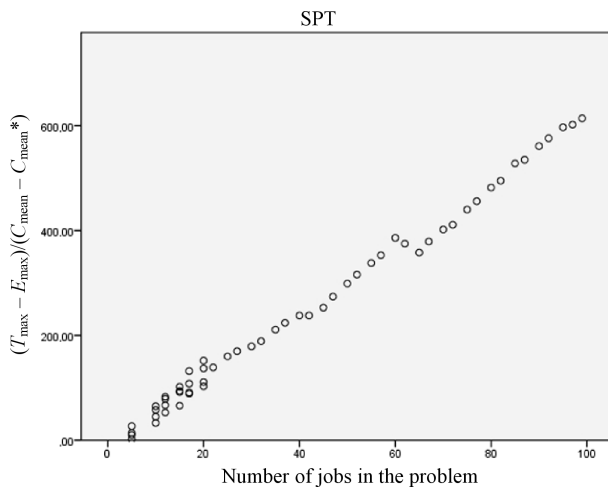


Fig. 11. Scatterplot to formula for SPT solutions.

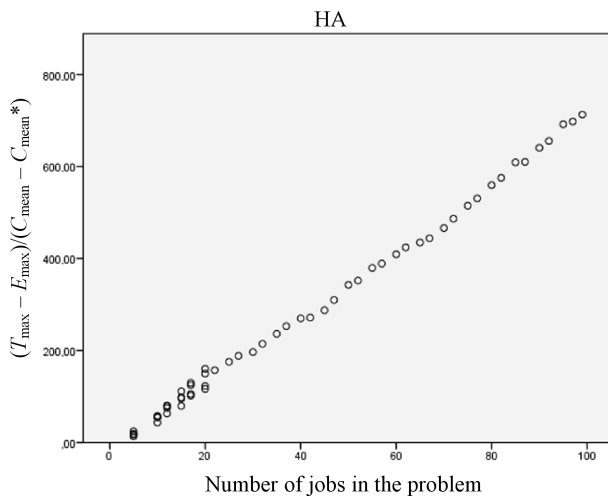


Fig. 12. Scatterplot to formula for HA solutions.

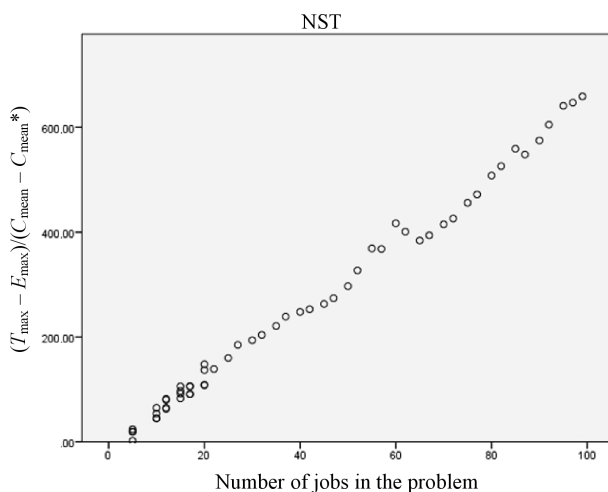


Fig. 13. Scatterplot to formula for NST solutions.

In the Table 6 it is possible to analyze the correlation results obtained through the SPSS software. It is possible to conclude, with 95% of confidence, that

there is a correlation, except for the  $C_{\text{mean}} - C_{\text{mean}^*}$  results through the use of the SPT and the NST algorithms. Regarding the case of the SPT results it is impossible to realize the correlation since the performance measure evaluated is a constant (zero). For the case of the NST results the correlation did not exist, as the results are near-optimal for all problem sizes.

Table 6  
Results for correlation analysis for both methods.

Performance measures	SPT	HA	NST
$T_{\text{max}} - E_{\text{max}}$	0.997	0.998	0.997
$C_{\text{mean}} - C_{\text{mean}^*}$	-	0.978	-0.238
Formula	0.997	0.998	0.997

### Evaluation of the methods performance

The performance of each method was evaluated through the ANOVA methodology (Table 7).

Table 7  
Results of Anova.

Performance measures	Sig.
$T_{\text{max}} - E_{\text{max}}$	0.890
$C_{\text{mean}} - C_{\text{mean}^*}$	0.00
Formula	0.662

The obtained performance is similar, with the same exception already referred about the mean completion time performance measure ( $C_{\text{mean}} - C_{\text{mean}^*}$ ). For the  $T_{\text{max}} - E_{\text{max}}$ , the ANOVA does indicate that there is a variance equality, which confirms an existence of similar results for the three methods (SPT, HAA and NST). The other performance measure (Table 7), continues to verify the exception case, which seems natural, considering the performance of the SPT and the NST. So the ANOVA does indicate that it is possible to reject the hypothesis of the variances equality of the methods results. For the formula that encloses the previous two performance measures there is variance equality, so we can accept the hypothesis about a similar methods performance.

### Evaluation of the methods performance with size differentiation

To differentiate the behavior between the small problems and the big ones, we considered the following categorization.

As there are no more than thirty instances for the problems with a small dimension to be able to assume its normal distribution, it is necessary to perform a test that proves it. So, a Shapiro-Wilk test was performed, and obtained the results presented on Ta-

bles 8 and 9. As we can realize, it is not possible to reject null hypothesis with a 95% confidence level, that problem data follows a normal distribution.

Table 8  
Different dimensions of problems.

Dimension	Size of problems
Small	$\leq 20$
Big	$> 20$

Table 9  
Results for Shapiro Wilk test.

	Sig.
SPT	0.856
HA	0.534
NST	0.609

For one performance measure one of the methods reveals a better performance than the other. With the same evaluation of the previous point but taking into consideration the size differentiation for the three performance measures, it is possible to conclude that the methods performances are similar.

The obtained results are systematized on Tables 10 and 11 for smaller and bigger problems.

Table 10  
Results of Anova for small problems.

Performance measures	Sig.
$T_{\max} - E_{\max}$	0.942
$C_{\text{mean}} - C_{\text{mean}^*}$	0.000
Formula	0.813

Table 11  
Results of Anova for big problems.

Performance measures	Sig.
$T_{\max} - E_{\max}$	0.766
$C_{\text{mean}} - C_{\text{mean}^*}$	0.000
Formula	0.393

We can only reject, with 95% confidence, that all methods performances, for the  $C_{\text{mean}} - C_{\text{mean}^*}$ , are not identical. In the other performance criteria we assume that all three methods achieved similar results.

Most times, the difference that turns out a method more effective than another, for one performance measure is not significant, regarding that the performance differences between methods are statistically not significant. Therefore, we have to analyze more than one performance measure, and using more than one approach. However, we can also enclose three performance measures, each one with different perspectives, in one single equation, allowing a more global and efficient analysis.

Moreover, we also analyze the performance measures behavior with the size growth of the problem, which permit to conclude about the methods performance for problems with different dimensions.

With the followed methodology it was possible to obtain a global perspective of the single machine scheduling and the behavior of different performance measures with two different heuristic methods.

## Conclusions

In this paper the problem of sequencing jobs on a single machine to minimize total cost (earliness and tardiness) was considered. Constructive heuristics that try to find good solutions for the single machine scheduling problem were applied.

The obtained results permit to conclude that in most of the times minimizing the earliness may degrade the tardiness results and this does not enable to reach a good solution. Therefore, it is of utmost importance to keep attention to the variation of some other very important performance measures, for instance, external or customer-oriented ones, and realize that it is not possible to give the same importance to performance measures such as the earliness and the tardiness, as the real consequences of each one are totally different in practice.

An implementation of the Hodgson's method and a NST algorithm were used, in this paper, for solving a single machine scheduling problem from the literature and a comparative analysis was performed. The obtained results were compared to the ones referred in the work from the literature and our results proved to be more effective, while enabling to reach a more balanced solution in terms of customer and enterprise oriented performance measures, by means of JIT principles, which are very important in the context of the proposed collaborative paradigm, that implies the existence of an accurate negotiation process for establishing and maintaining updated information about customers due dates for production orders. Therefore, more importance was given to customer-oriented measures, namely to the total number of tardy jobs and the maximum tardiness.

With the generalization of the problem, despite the increase of the value of almost of the performance measures, their growth was proportional to the growth of the number of jobs in the problem. The only exception was the performance of NST for the mean completion time, which was optimal or very near to the optimum. The methods performance has been similar regardless the problem dimension, with the same exception.

In future work, we intend to apply other methods in order to compare with the obtained results for the considered performance measures and even extend the analysis for other important performance measures, also to larger problems, with an increase in the number of jobs and further extending the statistical tests for increasing the analysis of scheduling problems' behavior.

*This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.*

## References

- [1] Sika R., Rogalewicz M., *Methodologies of knowledge discovery from data and Data Mining methods in mechanical engineering*, Management and Production Engineering Review, ISSN (2080-8208), 7, 4, 97–108, 2016, doi: 10.1515/mp-2016-0040.
- [2] Pinto T., Varela L.R., *Comparing extended neighborhood search techniques applied to production scheduling*, The Romanian Review Precision Mechanics, Optics & Mechatronics, 20, 2010.
- [3] Madureira A., Ramos C., Silva S.C., *Resource-oriented scheduling for real world manufacturing systems*, IEEE International Symposium, pp. 140–145, 2003.
- [4] Madureira A., Sousa N., Pereira I., *Self-organization for scheduling in agile manufacturing*, 10th IEEE International Conference on Cybernetic Intelligent Systems 2011, London, United Kingdom, 2011.
- [5] Madureira A., Pereira I., Pereira P., Abraham A., *Negotiation mechanism for self-organized scheduling system with collective intelligence*, Neurocomputing, 132, 97–110, 2014.
- [6] Klos S., Patalas-Maliszewska J., *Using a simulation method for intelligent maintenance management*, [in:] Intelligent Systems in Production Engineering and Maintenance, ISPEM 2017, Burduk A., Mazurkiewicz D. [Eds.], Cham, Springer International Publishing, Advances in Intelligent Systems and Computing, 637, 85–95, 2018.
- [7] Madureira A., Ramos C., do Carmo Silva S.D., *A coordination mechanism for real world scheduling problems using genetic algorithms*, Evaluation Computation, 1, 175–180, 2002.
- [8] Varela M.R.L., Trojanowska J., Carmo-Silva S., Costa N.M.L., Machado J., *Comparative simulation study of production scheduling in the hybrid and the parallel flow*, Management and Production Engineering Review, 8, 2, 69–80, 2017.
- [9] Madureira A., Santos J., *Proposal of multi-agent based model for dynamic scheduling in manufacturing*, WSEAS Transactions on Information Science & Applications, 2, 2005.
- [10] Madureira A., Pereira I., *Intelligent Bio-Inspired system for manufacturing scheduling under uncertainties*, Hybrid Intelligent Systems, pp. 109–112, 2010.
- [11] Koliński A., Śliwczyński B., Golińska-Dawson P., *Evaluation model for production process economic efficiency*, LogForum, 12(2), 129–145, 2016.
- [12] Kujawińska A., Rogalewicz M., Diering M., *Application of expectation maximization method for purchase decision-making support in welding branch*, Management and Production Engineering Review, 7, 2, 29–33, 2016.
- [13] Kawa A., *Simulation of dynamic supply chain configuration based on software agents and graph theory*, International Work-Conference on Artificial Neural Networks, Springer Berlin Heidelberg, pp. 346–349, 2009.
- [14] Hamidreza H., Mohammad Ebrahim A., Keyvan Kamandani P., *A branch and bound for single machine stochastic scheduling to minimize the maximum lateness*, International Journal of Industrial Engineering Computations, 3b, 499, 2012.
- [15] Liu Z., *Single machine scheduling to minimize maximum lateness subject to release dates and precedence constraints*, Computers and Operations Research, 37, 1537–1543, 2010.
- [16] Jin F., Song S., Wu C.m., *A simulated annealing algorithm for single machine scheduling problems with family setups*, Computers and Operations Research, 36, 2133–2138, 2009.
- [17] Liu L., Zhou H., *Applying variable neighborhood search to the single-machine maximum lateness rescheduling problem*, Electronic Notes in Discrete Mathematics, 39, 107–114, 2012.
- [18] Cheng T.C.E., Wang G., *Single machine scheduling with learning effect considerations*, Annals of Operations Research, 98, 273–290, 2000.
- [19] Haddad H., Ghanbari P., Zeraatkar Moghaddam A., *A new mathematical model for single machine batch scheduling problem for minimizing maximum lateness with deteriorating jobs*, International Journal of Industrial Engineering Computations, 3, 253–264, 2012.
- [20] Subramanian A., Battarra M., Potts C.N., *An iterated local search heuristic for the single machine total weighted tardiness scheduling problem with sequence-dependent setup times*, International Journal of Production Research, 52, 2729–2742, 2014.

- [21] Mohammad Mahdavi M., Hamidreza H., Mohammad M., Kazem A., *A calibrated hybrid for the single machine scheduling problem with objective of minimizing total weighted tardiness with dependent set-up*, International Journal of Information, Business and Management, 4, p. 55, 2012.
- [22] Mosheiov G., Yovel U., *Minimizing weighted earliness – tardiness and due-date cost with unit processing-time jobs*, European Journal of Operational Research, 172, 528–544, 2006.
- [23] Lu C.-C., Lin S.-W., Ying K.-C., *Robust scheduling on a single machine to minimize total flow time (Report)*, Computers & Operations Research, 39, 1682–1691, 2012.
- [24] Mor B., Mosheiov G., *Single machine batch scheduling with two competing agents to minimize total flowtime*, European Journal of Operational Research, 215, 524–531, 2011.
- [25] Jiyin L., MacCarthy B.L., *Effective heuristics for the single machine sequencing problem with ready times*, International Journal of Production Research, 29, 1521, 1991.
- [26] Low C.Y., Ji M., Hsu C.J., Su C.T., *Minimizing the makespan in a single machine scheduling problems with flexible and periodic maintenance*, Appl. Math. Model., 34, 334–342, 2010.
- [27] Ji M., He Y., Cheng T.C.E., *Single-machine scheduling with periodic maintenance to minimize makespan*, Comput. Oper. Res., 34, 1764–1770, 2007.
- [28] He Y., Zhong W., Gu H., *Improved algorithms for two single machine scheduling problems*, Theoretical Computer Science, 363, 257–265, 2006.
- [29] Chaudhry I., Drake P., *Minimizing flow-time variance in a single-machine system using genetic algorithms*, Int. J. Adv. Manuf. Technol., 39, 355–366, 2008.
- [30] Al-Turki U., Fedjki C., Andijani A., *Tabu search for a class of single-machine scheduling problems*, Computers and Operations Research, 28, 1223–1230, 2001.
- [31] Pinedo M., *Scheduling: theory, algorithms, and systems*, Prentice Hall, 2002.
- [32] Brucker P., *Scheduling algorithms*, Springer, 2007.
- [33] Wan G., Yen B.P.C., *Single machine scheduling to minimize total weighted earliness subject to minimal number of tardy jobs*, European Journal of Operational Research, 195, 89–97, 2009.
- [34] Rahmani K., Mahdavi I., Moradi H., Khorshidian H., Solimanpur M., *A nondominated ranked genetic algorithm for bi-objective single machine preemptive scheduling in just-in-time environment*, International Journal of Advanced Manufacturing Technology, 55, 1135–1147, 2011.
- [35] Pathumnakul S., Egbelu P.J., *Algorithm for minimizing weighted earliness penalty in single-machine problem*, European Journal of Operational Research, 161, 780, 2005.
- [36] Wang X.-R., Huang X., Wang J.-B., *Single-machine scheduling with linear decreasing deterioration to minimize earliness penalties*, Applied Mathematical Modelling, 35, 3509–3515, 2011.
- [37] Chen W.Y., Sheen G.J., *Single-machine scheduling with multiple performance measures: minimizing job-dependent earliness and tardiness subject to the number of tardy jobs*, International Journal of Production Economics, 109, 214–229, 2007.
- [38] Hino C.M., Ronconi D.P., Mendes A.B., *Minimizing earliness and tardiness penalties in a single-machine problem with a common due date*, European Journal of Operational Research, 160, 190–201, 2005.
- [39] Bauman J., Józefowska J., *Minimizing the earliness-tardiness costs on a single machine*, Computers and Operations Research, 33, 3219–3230, 2006.
- [40] Mansour M.A.A.-F., *A genetic algorithm for scheduling n jobs on a single machine with a stochastic controllable processing, tooling cost and earliness-tardiness penalties*, (Report), American Journal of Engineering and Applied Sciences, 4, 341, 2011.
- [41] Wan L., Yuan J., *Single-machine scheduling to minimize the total earliness and tardiness is strongly NP-hard*, Operations Research Letters, 41, 363–365, 2013.
- [42] Shabtay D., Steiner G., *The single-machine earliness-tardiness scheduling problem with due date assignment and resource-dependent processing times*, Ann. Oper. Res., 159, 25–40, 2008.
- [43] Yoo W.S., Martin-Vega L.A., *Scheduling single-machine problems for on-time delivery*, Comput. Ind. Eng., 39, 371–392, 2001.
- [44] Huo Y., Leung J.Y.T., Zhao H., *Bi-criteria scheduling problems: number of tardy jobs and maximum weighted tardiness*, European Journal of Operational Research, 177, 116–134, 2007.
- [45] Mohammad Mahavi M., Farzad Z., Farzad Firouzi J., *A fuzzy modeling for single machine scheduling problem with deteriorating jobs*, International Journal of Industrial Engineering Computations, 1, 147, 2010.
- [46] Mohammad K., Elnaz N., Amin A., Reza K., *Just-in-time preemptive single machine problem with*



- costs of earliness/tardiness, interruption and work-in-process*, International Journal of Industrial Engineering Computations, 3, 321, 2012.
- [47] Abdul-Razaq T.S., Potts C.N., *Dynamic programming state-space relaxation for single-machine scheduling*, J. Opl. Res. Soc., 39(2), 141–152, 1988.
- [48] Sidney J.B., *Optimal single-machine scheduling with earliness and tardiness penalties*, J. Opl. Res. Soc., 25(1), 62–69, 1977.
- [49] Madureira A.M., *Metaheuristics applications in sequencing problems* [in Portuguese], Master Thesis, FEUP, 1995.
- [50] Rao R.V., Vivek P., *An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems*, International Journal of Industrial Engineering Computations, 3, 535, 2012.
- [51] Trojanowska J., Żywicki K., Varela M.L.R., Machado J., *Improving production flexibility in an industrial company by shortening changeover time: a triple helix collaborative project*, [in:] Multiple Helix Ecosystems for Sustainable Competitiveness, Peris-Ortiz M., Ferreira J., Farinha L., Fernandes N. [Eds.], Innovation, Technology and Knowledge Management, Springer, pp. 133–146, 2016.
- [52] Ivanov V., Mital D., Karpus V. et al., *Numerical simulation of the system “fixture–workpiece” for lever machining*, International Journal of Advanced Manufacturing Technology, 91, 79–90, 2017.
- [53] Varela M.L.R., do Carmo Silva S., *An ontology for a model of manufacturing scheduling problems to be solved on the web*, Innovation in Manufacturing Networks, [Ed.] Springer, pp. 197–204, 2008.
- [54] Babaei M., Mohammadi M., Ghomi S.M.T.F., Sobhanallahi M.A., *Two parameter-tuned metaheuristic algorithms for the multi-level lot sizing and scheduling problem*, International Journal of Industrial Engineering Computations, 3, 751, 2012.
- [55] Ouelhadj D., Petrovic S., *A survey of dynamic scheduling in manufacturing systems*, J. Sched., 12, 417–431, 2009.
- [56] Madureira A., *Hybrid meta-heuristics based system for distributed dynamic scheduling*, [in:] Encyclopedia of Artificial Intelligence, Rabuñal J.R., Dorado J., Prazos A. [Eds.], Idea Group Reference, Information Science Reference Ed., 2009.
- [57] Madureira A., Santos J., Pereira I., *Hybrid intelligent system for distributed dynamic scheduling*, Springer-Verlag, Ed., 2009.
- [58] Varela L.R., Ribeiro R.A., *Evaluation of simulated annealing to solve fuzzy optimization problems*, Journal of Intelligent and Fuzzy Systems, 14, 59–71, 2003.
- [59] Varela M.L.R., Putnik G.D., Cruz-Cunha M.M., *Web-based technologies integration for distributed manufacturing scheduling in a virtual enterprise*, International Journal of Web Portals (IJWP), 4, 19–34, 2012.
- [60] Ribeiro R.A., Varela L.R., *Fuzzy optimization using simulated annealing: an example set*, Fuzzy Sets Based Heuristics for Optimization, Ed.: Springer, pp. 159–180, 2003.
- [61] Dantas J.D., Varela L.R., *Scheduling single-machine problem based on just-in-time principles*, Proceedings of the Sixth World Congress on Nature and Biologically Inspired Computing (NaBIC’14), pp. 164–169, 2014.