4th International Conference on Industrial Engineering and Industrial Management
XIV Congreso de Ingeniería de Organización
Donostia- San Sebastián , September 8th -10th 2010

Permutation flowshop problems with initial availability constraints: Characterisation and Analysis*

Paz Pérez-González[1], José Manuel Framiñán Torres[1], Pedro Luis González Rodríguez[1], Rafael Ruiz Usano[1]

[1] Dpto. de Organización Industrial y Gestión de Empresas. Escuela Técnica Superior de Ingenieros. Universidad de Sevilla. Camino de los descubrimientos s/n, 41092 Sevilla. pazperez@esi.us.es, framinan@us.es, pedroluis@esi.us.es, usano@us.es

Abstract

This paper characterises the initial availability constraint problems in a permutation flowshop layout considering different objectives: makespan, total flowtime and idle time. The goal is to analyse the structure of solutions and to discuss the managerial implications of these three problems. Also, we intend to compare the cases with availability constraint with their classical counterparts, i.e. these problems with machines available on the planning horizon. The analysis raises an important conclusion of practical application: Since for most real-life environments scheduling is performed on a periodical basis and this would naturally lead to the unavailability of machines at the starting of the scheduling period, this scheduling decision problem becomes easier than its 'classical' (i.e. without machine unavailability) counterpart.

Keywords: scheduling, problem characterisation, permutation flowshop, machine availability constraint

1. Introduction

Machine availability constraint problems have been widely tackled in scheduling literature since there is a wide range of realistic situations where machines may not be completely available. Machine breakdowns (stochastic unavailability) (Allaoui et al, 2006), and preventive maintenance activities (deterministic unavailability) (Ruiz et al, 2007) are the most studied cases. However, a typical situation of deterministic unavailability is the case when machines are busy by processing jobs belonging to previously scheduled orders, i.e., the so-called initial availability constraint. In this case, machines may not be immediately available for processing the set of jobs to be scheduled, but only from a date a_i that we denote as *availability instant*. This problem is identified in an scenario where jobs must be scheduled at time T in a periodical manner, being H the decision period. In this case, the decision maker should schedule dynamically orders (jobs) that entered the system from T-H to T. Jobs scheduled in the previous period may not be completed at this point, and they can be merged with the new set of jobs and rescheduled, or can be considered as "frozen", which causes the initial availability constraint.

* This work stems from the participation of the authors in the research project "Advanced Systems for Integrated Order Management", grant DPI2007-61345, funded by the Spanish Ministry of Science and Innovation, and "SCOPE", grant P08-TEP-3630, funded by the Andalusian Government.

This paper characterises the initial availability constraint problems in a flowshop layout employing different objectives: makespan, total flowtime and idle time. We focus onto this shop floor setting since it is widely extended in the real-life manufacturing, being often claimed that many job shops are flowshops for most of the jobs (Knolmayer et al, 2002; Storer et al, 1992). The flowshop scheduling problem involves the determination of the order in which jobs with given and fixed processing times are processed in the same machine sequence to meet a desired objective. Here, the permutation case is considered, which assumes that the job sequence is the same on all machines.

Among the objectives studied in the literature about flowshop scheduling, most of the attention has been devoted to minimizing either makespan or flowtime. The practical implications of both criteria are obvious: minimization of makespan leads to the minimization of the total production run, while minimization of flowtime leads to stable or even use of resources, a rapid turn- around of jobs, and the minimisation of in-process inventory. Additionally, minimization of machine idle time yields a high utilization rate for the machines (Framinan et al, 2003).

The goal of our paper is to study these three problems, to analyse their structure of solutions and to discuss the managerial implications. Also, in this study we intend to compare the availability constraint problems with their classical counterparts, i.e. these problems with machines available on the planning horizon.

2. Problem description

Following the notation introduced by Graham et al (1979), our problem is denoted as $Fm|prmu,a_i|\gamma$, where Fm means a flowshop problem with m machines, $prmu$ states that it is a permutation case, a_i specifies the initial availability constraint, and finally, γ may be either C_{max}, (makespan objective), F (total flowtime objective), or IT (idle time).

Availability instants a_i define the time from which machine i is available, thus $a_i \geq 0$ for $i=1,\ldots m$. Without loss of generality we can assume that $a_i \leq a_{i+1}$ for $i=1,\ldots m$. If $a_i \geq a_{i+1}$ for some i, then a_{i+1} would not influence the problem. In fact, a given a_{i+1} may have influence if it is greater than $a_i+\min_j\{p_{ij}\}$ for $i=1,\ldots m$, $j=1,\ldots n$ being p_{ij} the processing time of job j in machine i. In addition, we can assume that $a_1=0$. If this is not the case, the reference change $a_i' = a_i - a_1$ may be done to guarantee that the first machine is available from the beginning of the decision period.

2.1. Makespan objective

Makespan is computed as the completion time of the last job in the last machine, i.e. $C_{max} = C_{mn}$. The classical permutation flowshop problem with makespan objective, $Fm|prmu|C_{max}$, denoted in the following as CP_{mak}, can be optimally solved by the Johnson rule for m=2. This problem is NP-hard in the strong sense for m>2. The distribution of the solutions of this problem was studied by Taillard (1990). The distribution of the solutions of the constrained problem with makespan objective ($Fm|prmu,a_i|C_{max}$ and denoted by AP_{mak} in the following) is analysed, and compared to that of CP_{mak}, in Perez-Gonzalez and Framinan (2009), concluding that the former is easier that the latter.

2.2. Flowtime objective

Flowtime is defined as the sum of the completion times of each job in the last machine, i.e. $F=\Sigma_j C_{mj}$. The classical permutation flowshop problem with the total flowtime objective, $Fm|prmu|F$, denoted as CP_{flw}, is NP-complete even in the two machine case (Garey et al, 1976). $Fm|prmu,a_i|F$ is denoted as AP_{flw}in the following.

2.3. Idle time objective

Finally, the idle time is defined as the sum of the idle times of each machine. This objective has been scarcely studied in the literature on flowshop scheduling. Only Ho and Chang (1991), Sridhar and Rajendran (1996) and Framinan et al (2003) consider the classical problem $Fm|prmu|IT$, denoted CP_{idle}. All of these references compare the three objectives considered in this work for the classical versions of the problem. The distribution of the classical problem has not been studied in the literature. The availability constraint version of this problem, $Fm|prmu,a_i|IT$, is denoted as AP_{idle}.

3. Analysis of the problems

To analyse the problems presented in the previous section, we build a high number of problem instances and obtain all possible schedules together with the corresponding solution values (for the three objectives). A similar approach has been carried out by Taillard (1990) for CP_{mak}, and Armentano and Ronconi (1999) for $Fm|prmu|\Sigma T_j$, with ΣT_j the total tardiness. The latter reference considers different scenarios for the generation of the due dates. We adapt the method considering different scenarios for the availability vector, and controlling the size of the vector by a factor k.

3.1. Instances generation

The parameters for the problem instances are the number of jobs, the number of machines and the availability vector a.

Regarding the number of jobs and the number of machines, they should be restricted to small values in order to obtain all possible schedules and objective function values in a reasonable time. Therefore, n={5, 10} and m={5, 10}, generating problems with the following sizes $n\times m$: 5×5, 5×10, 10×5 and 10×10.

Finally, we calculate different availability vectors with several sizes by employing $C_i(S_{ini})$, the completion time of sequence S_{ini} =[1,...,n], verifying that $C_i(S_{ini}) < C_j(S_{ini})$ for $i < j$, i=1,...,m. An initial vector is computed from these values doing a reference change where $a_j' = C_j(S_{ini}) - C_1(S_{ini})$ for j=1,...,m, i.e. $a' = [a_1',...,a_m'] = [0, C_2(S_{ini}) - C_1(S_{ini}),..., C_m(S_{ini}) - C_1(S_{ini})]$. To control the size of the availability vector, we consider different values of k, and use $a = k*a'$ as the availability vector. The selected values are k=0 (obtaining the problems CP_l without availability constraint and the corresponding objective function) and k=0.5, 1 and 2 (obtaining some cases of AP_l with l=mak,flw,idle for different sizes of a).100 instances have been generated for each combination of the values of n, m and k, so 100×2×2×4 = 1600 instances of the problem are exactly solved by complete enumeration. These results are then summarized in order to extract conclusions on the distribution of the space of solutions.

3.2. Distribution of the space of Solutions

The distribution of the space of solution is given in relative terms to the optimal solution, i.e. we calculate the relative objective function value $f^r(S)$ to obtain the approximation percentage to the optimal solution S^* of each sequence S regarding to its objective value. Then $f^r(S) = f(S)/f(S^*) - 1$.

Figures 1, 2 and 3 show the empirical distributions representing all possible values of each relative objective function obtained by complete enumeration of 100 problems with 10 jobs and 10 machines for all levels of the parameter k, for makespan, flowtime and idle time. The horizontal axis represents the percentage of approximation to optimal value and the vertical axis shows the empirical frequency (%), i.e. the percentage of solutions at each percentage of approximation to optimal value.

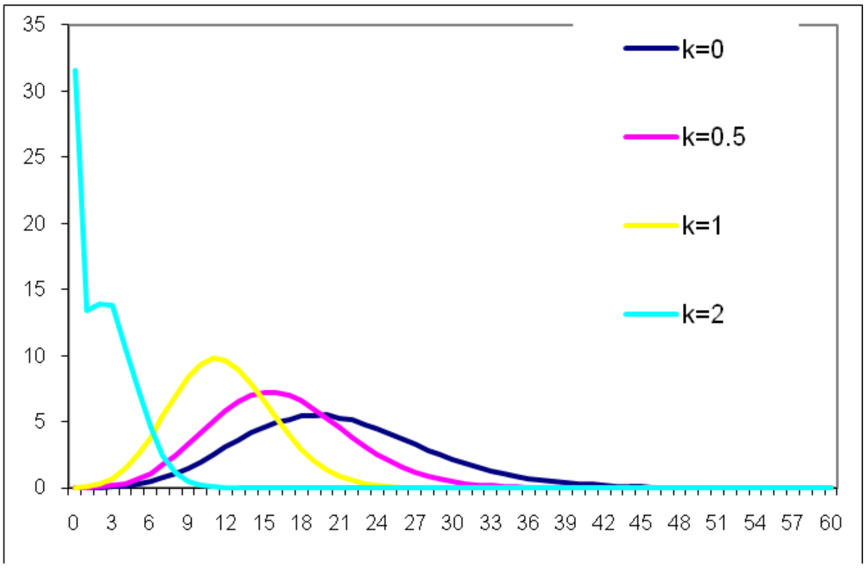

Figure 1. Distribution of solutions for small problems: makespan objective

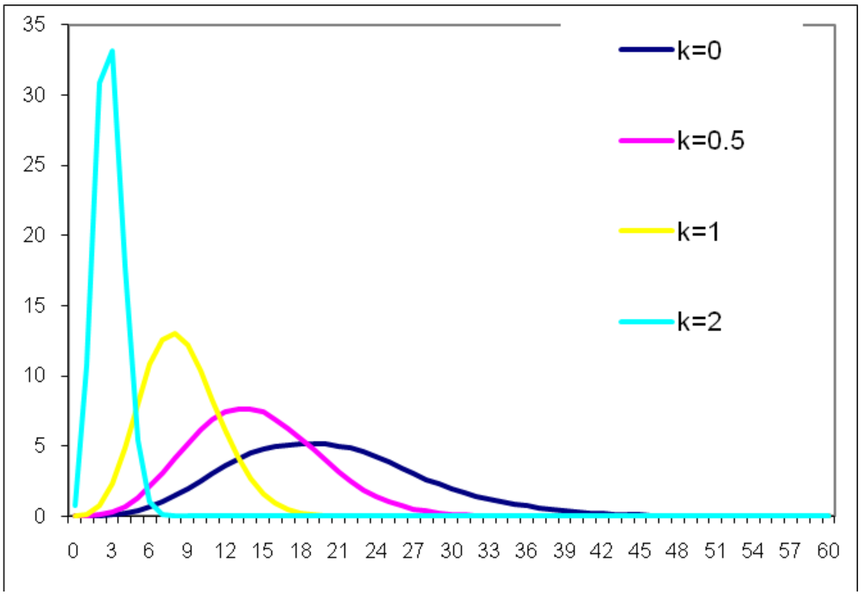

Figure 2. Distribution of solutions for small problems: flowtime objective

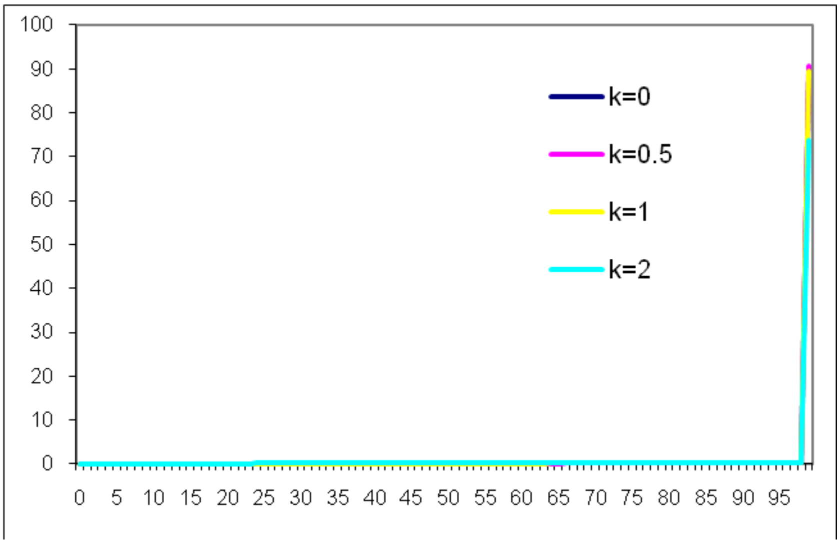

Figure 3. Distribution of solutions for small problems: idle time objective

From Figures 1 and 2 it can be observed that as *k* increases, the frequency of solutions with a small approximation percentage is larger. Therefore, the problem becomes easier (in statistical terms) as *k* increases, being the case *k*=0, i.e. the classical version of the problems CP_{mak} and CP_{flw}, the most difficult problem for both objectives. Figure 3 indicates that the problem with idle time as objective is the most difficult, increasing the difficulty as *k* increases since a larger percentage of solutions are at 99% or more from the optimal solution. Similarities between the distribution of the solutions for the cases makespan and flowtime objectives can be observed as well.

Table 1 presents the mean of the approximation percentage to the optimal solution for each problem, and the upper bound of the approximation percentage to the optimuml for 95% of the solutions. These results are classified by the objective function used, for all values of *k* and for all combinations of number of jobs and number of machines ($n \times m$).

Table 1. Mean and 95% of approximation to the optimal values for each objective.

k	Objective	5×5		5×10		10×5		10×10	
		Mean	95%	Mean	95%	Mean	95%	Mean	95%
0	makespan	17.362	36	13.108	27	23.421	38	20.893	33
	flowtime	18.715	41	14.567	30	26.193	43	20.268	33
	idle time	87.971	99	86.285	99	96.396	99	95.957	99
0.5	makespan	13.615	28	11.003	22	15.013	25	16.392	25
	flowtime	12.751	27	11.807	24	15.703	25	14.478	22
	idle time	88.004	99	87.461	99	95.103	99	96.491	99
1	makespan	10.609	22	9.395	19	9.273	16	11.792	18
	flowtime	9.233	19	9.262	19	8.983	14	8.522	13
	idle time	87.279	99	87.937	99	94.187	99	96.277	99
2	makespan	5.294	13	5.157	11	2.269	6	2.260	5
	flowtime	4.079	8	2.973	6	2.778	4	2.778	4
	idle time	84.208	99	83.244	99	87.845	99	89.473	99
Total	makespan	11.720	24.75	9.666	19.75	12.494	21.25	12.834	20.25
	flowtime	11.195	23.75	9.652	19.75	13.414	21.5	11.512	18
	idle time	86.866	99	86.232	99	93.383	99	94.550	99

For example, for makespan, problems with five jobs and five machines in the case of *k* = 2 the mean of approximation percentage to the optimal solution is 5.294, i.e. any solution is (on average) below 5.294% of the optimal makespan. Furthermore, 95% of solutions are below

13% of the optimal makespan. However, the mean for $k = 0$ is 17.372 and the upper bound for the 95% of solutions is 36%, showing in a clear way that the problems are easier while k increases. Means and 95% for makespan and flowtime are similar for all sizes $n \times m$ and all values of k. However, the performance of the idle time is different, since the means are around 90% for all sizes and cases of k, and the 95% of solutions are at 99% of approximation to the optimal idle time.

The hardness of the problems for idle time, regardless of the size, increases with k, following a different pattern than that of the other objective functions. However, a possible correlation between the results for makespan and flowtime could exist according to the similarities observed in the previous results. For both objectives the difficulty decreases with k and the problem size.

3.3. Correlation makespan-flowtime

We would like to determine the similarities between the distribution of the solutions for the problems with makespan and flowtime objectives. Correlations give us the statistical relationship between two variables. First, we represent the scatter diagrams for each value of k=0,0.5,1 and 2 in Figures 4, 5, 6 and 7 respectively. As it can be observed in the figures the correlation between both variables decreases ask increases. .

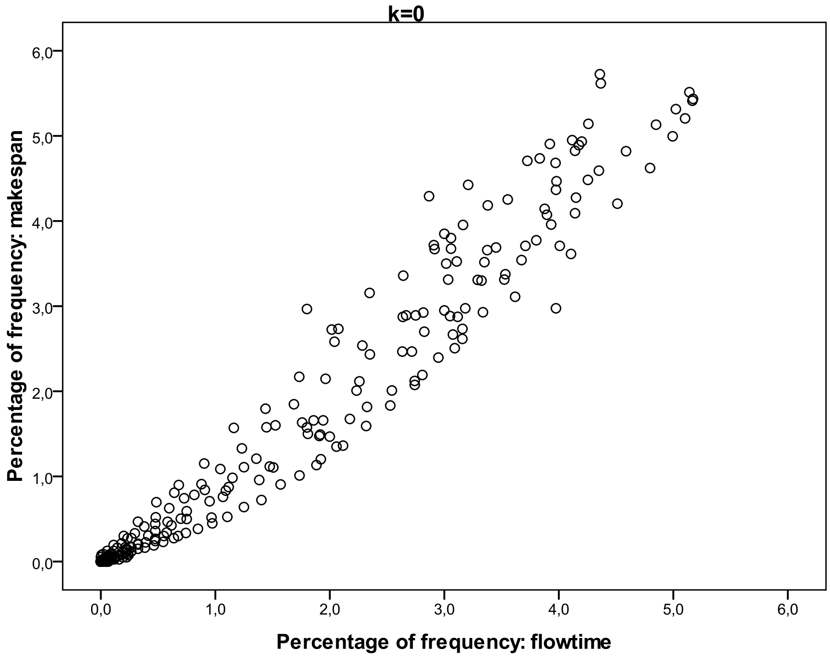

Figure 4. Scatter diagram: case k=0

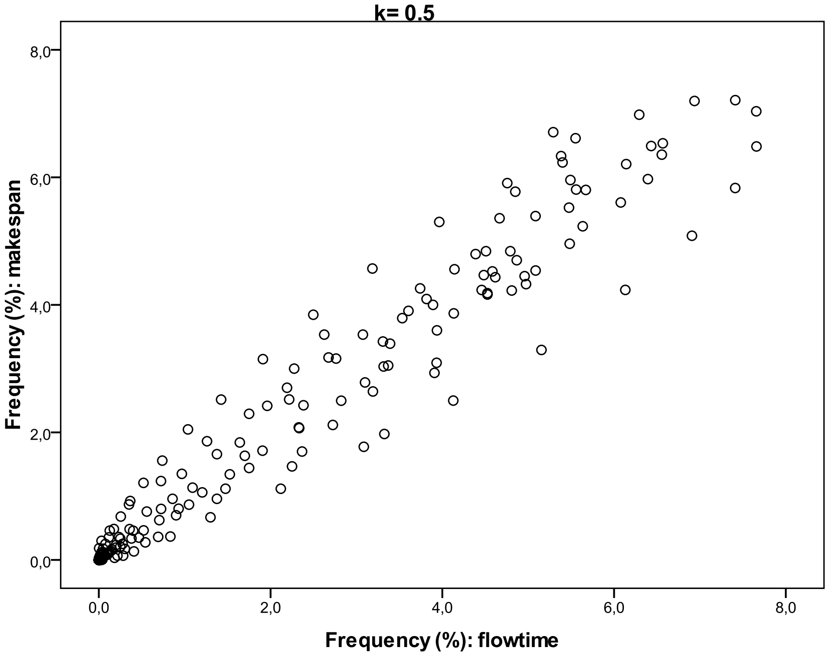

Figure 5. Scatter diagram: case k=0.5

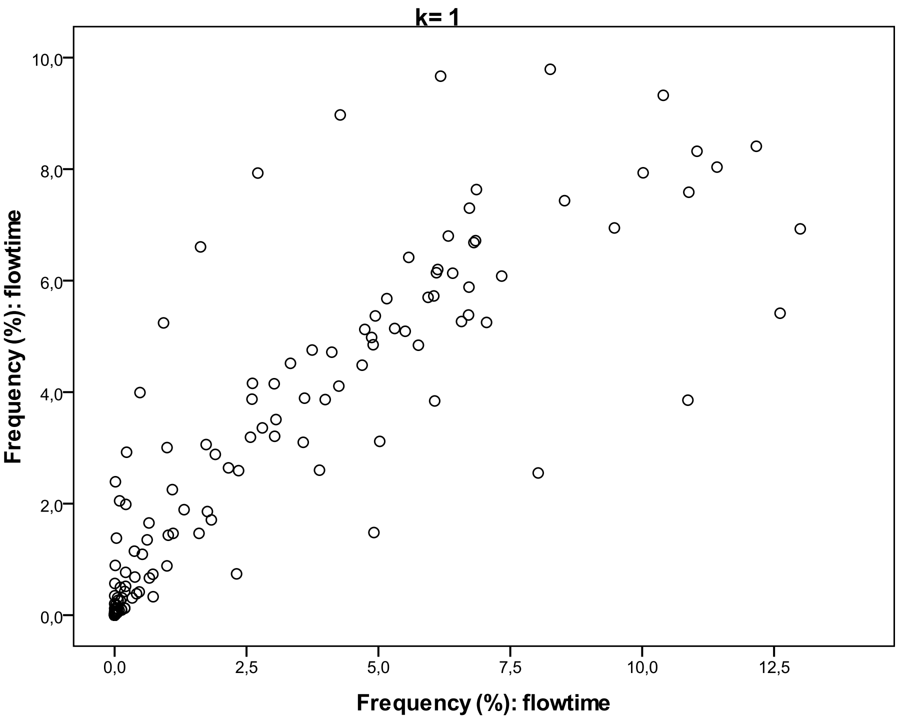

Figure 6. Scatter diagram: case k=1

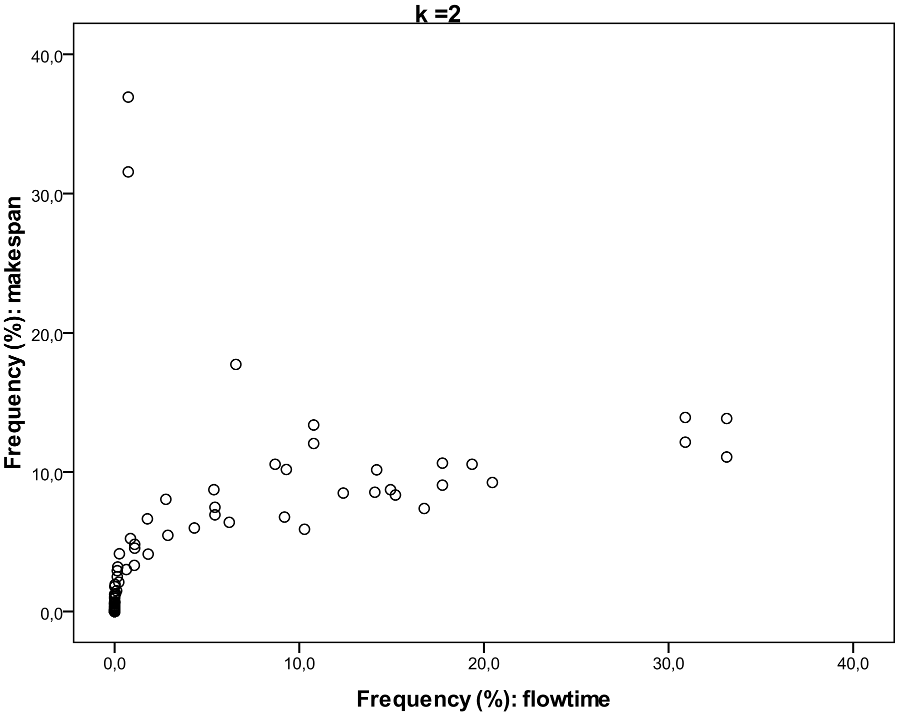

Figure 7. Scatter diagram: case k=2

The conclusions obtained by the scatter diagrams are confirmed by the values of Pearson's correlation coefficient, Kendall's Tau-b coefficient, and Spearman's Rho coefficient showed in Table 2. For all cases the results are significant at level 0.01.

Table 2. Correlation coefficients

Correlation coefficient	k=0	k=0.5	k=1	k=2
Pearson	0.981	0.980	0.905	0.627
Kendall's Tau-b	0.894	0.916	0.897	0.755
Spearman's Rho	0.965	0.967	0.950	0.790

Pearson's correlation coefficient values are close to 1 for smallest values of *k*, indicating that there is a positive and linear relation between the frequency of approximation to the optimal values for makespan and flowtime objectives. Kendall's Tau-b and Spearman's Rho confirm the results with similar values if the normality assumption cannot be guaranteed.

The information provided by these results shows that there is a relationship between the difficulty of the problems with makespan and flowtime objectives. The number of solutions in a given interval of approximation percentage to the optimal value for makespan is proportional to the number of solutions in the same interval of approximation percentage to the optimal value for flowtime. The levels of difficulty for both problems are very similar for small values of *k*, but the similarity decreases with *k*.

4. Conclusions

In this work we analyse a type of machine availability constraint problems in the permutation flowshop environment, assuming that machines are not available at the beginning of the planning period. We obtain the distribution of the solutions for three objective functions: makespan, flowtime and idle time. The objective is to determine the difficulty degree of these problems. Makespan and flowtime objectives have been widely studied in the permutation flowshop literature in their classical versions, i.e. without availability constraints. However, to the best of our knowledge, only two references tackle the idle time. The analysis reveals that the idle time problem is very difficult, and reflects a relationship between the levels of difficulty for the makespan and flowtime cases.

The analysis carried out in this work raises an important conclusion of practical application: Since for most real-life environments scheduling is performed on a periodical basis and this would naturally lead to the unavailability of machines at the starting of the scheduling period, this scheduling decision problem becomes easier than its 'classical' (i.e. without machine unavailability) counterpart. Moreover, we have proved the relationship between the difficulties of the problems for the most studied objective functions: makespan and flowtime. The probability to find good solutions for problems with makespan as objective is almost the same that for problems with flowtime as objective. This relationship decreases with the size of the availability vector, being most difficult for those problems with makespan as objective.

References

Allaoui, H. et al. (2006) Scheduling of a two-machine flowshop with availability constraints on the first machine. *International Journal of Production Economics*, Vol 99, No.1-2, pp. 16-27.

Armentano, V. A.; Ronconi, D. P. (1999) Tabu search for total tardiness minimization in flowshop scheduling problems. *Computers & Operations Research*, Vol 26, No.3, pp. 219-235.

Framinan, J. M.; Leisten, R.; Rajendran, C. (2003) Different initial sequences for the heuristic of Nawaz, Enscore and Ham to minimize makespan, idletime or flowtime in the static permutation flowshop sequencing problem. *International Journal of Production Research*, Vol 41, No.1, pp. 121- 148.

Garey, M. R.; Johnson, D. S.; Sethi, R. (1976) The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, Vol 1, No.2, pp. 117- 129.

Graham, R. et al. (1979) Optimization and approximation in deterministic sequencing and scheduling: A survey. *Annals of Discrete Mathematics*, Vol 5, pp. 287- 326.

Ho, J. C.; Chang, Y. L. (1991) A new heuristic for the n-job, M-machine flow-shop problem. *European Journal of Operational Research*, Vol 52, No.2, pp. 194- 202.

Knolmayer, G.; Mertens, P.; Zeier, A. (2002). *Supply Chain Management Based on SAP Systems: Order Management in Manufacturing Companies*. Springer, Berlin.

Perez-Gonzalez, P.; Framinan, J. M. (2009) Scheduling permutation flowshops with initial availability constraint: Analysis of solutions and constructive heuristics. *Computers & Operations Research*, Vol 36, No.10, pp. 2866- 2876.

Ruiz, R.; Carlos Garcia-Diaz, J.; Maroto, C. (2007) Considering scheduling and preventive maintenance in the flowshop sequencing problem. *Computers & Operations Research*, Vol 34, No.11, pp. 3314- 3330.

Sridhar, J.; Rajendran, C. (1996) Scheduling in flowshop and cellular manufacturing systems with multiple objectives--a genetic algorithmic approach. *Production Planning & Control*, Vol 7, No.4, pp. 374.

Storer, R. H.; Wu, S. D.; Vaccari, R. (1992) New Search Spaces for Sequencing Problems with Application to Job Shop Scheduling. *Management Science*, Vol 38, No.10, pp. 1495-1509.

Taillard, E. D. (1990) Some efficient heuristic methods for the flow shop sequencing problem. *European Journal of Operational Research*, Vol 47, No.1, pp. 65- 74.