Mathematical model of workforce scheduling problem in flow shop with makespan minimization

Santiago Sánchez H.
Universidad de La Sabana
Chía (Cundinamarca), Colombia
davidsahe@unisabana.edu.co

Jairo R. Montoya Torres
Universidad de La Sabana
Chía (Cundinamarca), Colombia
jairo.montoya@unisabana.edu.co

Abstract
We study a hand-intensive production flow-shop with variable workers' productivity. Actual processing times of jobs are assumed to vary depending on their position on the schedule. A mathematical model is proposed. Computational experiments are run using different datasets in order to evaluate the power of the model in terms of efficiency and efficacy to solve the problem.

Keywords: Workforce scheduling, flow-shop, position-dependent processing times

INTRODUCTION

Despite the industrial development and the technological resources that nowadays are available for industries around the world (Boudreau et al. 2002), the fact is that persons are a key element in operations systems, either as decision makers or as systems operators (Neumann and Dul 2010). Decades ago, industry managers considered that workers were resources that could be eventually replaced by automation (Baines and Kay 2002). Nevertheless, there are tasks and labors that necessarily require precision, intelligence, analysis and logic own only of human beings. Operations experts recognize, indirectly and implicitly, the importance of people as resources, and thus, operations management textbooks have its respective human resources chapters and sections (Heizer and Render 2007). Nevertheless, the topic is rarely mentioned in operations management scientific journals. Alternatives for integrating both elements were proposed more than a decade ago, however; those early initiatives never elucidated a strong mathematical basis to incorporate human behavior within operations management paradigm (Lodree et al. 2009). Therefore, there is a significant gap between the human factor and operations management. Human behavior influence in operations systems has been underestimated (Bentefouet and Nembhard 2013). Companies are involved in a dominant logic in which it is believed that the human factor does not have a significant impact in the achievement of its strategic objectives (Prahalad 2004).

The gap mentioned in the last paragraph, trigger in a very complex situation: generally, when a human resources analysis is developed in a productive environment, many assumptions and critical simplifications are done about characteristics and behavior of people: workers have a predictable behavior or their performance is constant and there is no chance for them to feel fatigue or tired (Boudreau et al. 2002). In such a context, the aim of the current paper is to address the problem of production scheduling in a hand-intensive manufacturing system. The configuration under study is a flow line (or flow shop) in which workers productivity may be affected by different situations such as: heavy loads lifting, repetitive tasks, hard weather conditions, sentimental or psychological issues, etc. As explained in detail in the next section, the problem can be modeled as processing times deterioration scheduling problem.

In formal terms, we consider the problem of having m workers $i = 1, \dots, m$ with 100% production rate at initial time of scheduling (production horizon). Those workers must process n independent jobs $j = 1, \dots, n$. Each job must be processed by all workers, it means the workers are organized sequentially. The sequence in which the jobs will be processed must be the same for all workers (permutation flow shop). Each worker can process one job at a given time and interruption during processing of a job is not allowed. As traditionally defined (Pinedo 2012), the processing time p_{ij} represents the basic time the job j needs to be processed by worker i. As previously explained, the processing time of job j will increase in function of its position on the sequence: processing time of job j scheduled in the k^{th} position is computed as $p_{i,k} = p_{i,j}k^{\alpha}$, where α is the deteriorating rate. The objective is to minimize the completion of all jobs or makespan (Cmax). Following the standard notation in Scheduling Theory, the problem under study is denoted as $Fm|p_{i,j,r} = p_{i,j}r^{\alpha}|Cmax$.

Our objective is to formulate a mathematical model to test the efficiency and efficacy for different problem sizes. The rest of this paper is organized as follows. We first present a review of related literature. Afterwards, the proposed mathematical model is explained in detail, followed by the results of the computational experiments. The paper ends by presenting some concluding remarks and opportunities for future research.

RELATED LITERATURE

According to the academic literature, flow shop scheduling problems can be classified as either "permutation flow-shop" or "non-permutation flow shops". In the former environment, the execution sequence of all jobs is the same at all manufacturing resources, while in the latter environment, the execution sequence of jobs is not necessarily the same. When considering workers, the scheduling problem has been traditionally classified in three main problems (Ernst et al. 2004):

- Day-off Scheduling: The objective is to determine, for each worker, work and off-work days depending on the time scheduling horizon (Alfares 2002)
- Shift Scheduling: The aim is to select the best set of shifts between a known number of workers in a single day or in a time horizon previously defined (Aykin 2000)
- Tour Scheduling: Day-off scheduling and shift scheduling problems are integrated, allowing to determine off-days for workers and the best shifts for their working days (Goodale and Thompson, 2004)

A full description of worker scheduling and allocation issues can be found in (Ernst et al. 2004). An extensive literature review about the Tour Scheduling Problem was developed in (Alfares 2004). The importance of integrating human factors in personnel scheduling and allocation (Operations Management) has been studied in the works of Neumann and Dul (2010) and Boudreau et al. (2002).

When dealing with worker scheduling problems, skills and qualifications act as restrictive filters on who can perform a task (Moreno-Camacho and Montoya-Torres 2015). We must describe two cases mainly. In the first case, all workers have the same abilities, which means that anyone can execute a task. In the second case, workers have different abilities, then, the challenge is to match workers abilities with available tasks, reaching the lower cost and/or the higher productivity levels (Ernst et al. 2004).

Commonly, academic literature on workforce scheduling considers that processing times are deterministic (i.e., do not vary over time) (Brucker 2007). Nonetheless, for nowadays, industrial environments, processing times of tasks can change with ease. Psychological and sentimental issues, fatigue, and other elements previously mentioned may, and very often do, bring on the increase in processing times of jobs. According to the well-known scheduling models studied in the literature, the problem we are analyzing can be modeled as a scheduling problem with deteriorating processing times of jobs.

Time- and position-dependent processing times problems have been studied significantly. (Gawiejnowicz 2008) developed an extensive literature review, which includes various scheduling models and problems which take in to account deteriorating job processing times. Recent publications have considered the deterioration of processing times for the two-machine problem (e.g., Zhao and Tang 2012, Wang and Liu 2009), for the three-machine environment (e.g., Wang and Wang 2013, Wang et al. 2010) and for m-machines problem (e.g., Wang et al. 2011, Lee et al. 2009). Few of the analyzed publications formulate mathematical models or resolution algorithms, the most of those works are focused on theoretical analysis (e.g., types of deteriorating functions for processing times (Cheng et al. 2004) and problems complexity demonstrations (Thörnblad and Patriksson 2011)). The efficiency of mathematical models is relatively unexplored (Moreno-Camacho and Montoya-Torres 2015).

MIXED-INTEGER PROGRAMMING MODEL

This section presents the detailed algebraic formulation of the mathematical model proposed to address the problem under study. Our model is based on the MIP model proposed in (Guéret et al. 2000) which is a corrected version of the model formulated in (Pinedo 2012). We adapted the model to include position-dependent processing times. It means that processing times of jobs are assumed to vary depending on their position on the schedule, through a deterioration factor, noted as α, which can vary between 0 and 1.

Three decision variables are defined. Binary variable $x_{jk} = 1$ if job j is processed in the k^{th} position, and 0 otherwise. The other variables required to run the model are defined as integer variables: e_{ik} represents the idle time in the worker i between the processing of jobs in the k^{th} and the $(k + 1)^{\text{th}}$ positions; and a_{ik} representing the waiting time of the job in the position k between workers i and $i + 1$.

Let Z be the objective function. The aim is to minimize the makespan or total completion time of all jobs, noted as Cmax. The Mixed-Integer Programming (MIP) model is defined as follows:

$$\min Z = \sum_{i=1}^{m-1} \sum_{j=1}^{n} p_{ij} * x_{j1} + \sum_{j=1}^{n} \sum_{k=1}^{n} p_{mj} * x_{jk} * k^{\alpha} + \sum_{j=1}^{n-1} e_{mj} \quad (1)$$

Subject to:

$$\sum_{k} x_{jk} = 1 \qquad \forall\, j = 1, \dots, n \qquad (2)$$

$$\sum_{j} x_{jk} = 1 \qquad \forall\, k = 1, \dots, n \qquad (3)$$

$$e_{ik} \geq 0 \qquad \begin{matrix} \forall\, i = 1, \dots, m \\ \forall\, k = 1, \dots, n-1 \end{matrix} \qquad (4)$$

$$a_{ik} \geq 0 \qquad \begin{matrix} \forall\, i = 1, \dots, m-1 \\ \forall\, k = 1, \dots, n \end{matrix} \qquad (5)$$

$$e_{1k} = 0 \qquad \forall\, k = 1, \dots, n-1 \qquad (6)$$

$$a_{i1} = 0 \qquad \forall\, i = 1, \dots, m-1 \qquad (7)$$

$$\begin{matrix} e_{ik} + \sum_{j=1}^{n} p_{ij} * x_{j,k+1} + a_{i,k+1} = \\ a_{ik} + \sum_{j=1}^{n} p_{i+1,j} * x_{jk} + e_{i+1,k} \end{matrix} \qquad \begin{matrix} \forall\, i = 1, \dots, m-1 \\ \forall\, k = 1, \dots, n-1 \end{matrix} \qquad (8)$$

$$x_{jk} \in \{0,1\} \qquad \begin{matrix} \forall\, j = 1, \dots, n \\ \forall\, k = 1, \dots, n \end{matrix} \qquad (9)$$

$$0 \leq \propto \leq 1 \qquad (10)$$

Constraint (1) represents the objective function. It is composed by three mathematical expressions. The first is the total time that the job j selected in the position $k = 1$, elapses arriving to the worker m; the second expression is the sum of processing times of j jobs executed by worker m; and the third expression is the sum of idle times between the j jobs processed by worker m. Constraints (2) specify that for each job only one position must be assigned. Constraints (3) ensure that, for each position, only one job must be assigned. Constraints (4) and (5) guarantee that e_{ik} and a_{ik} variables are non-negative. Constraints (6) forces to be zero any idle time e for the first worker. Constraints (7) force to be zero any waiting time a for the first position in the schedule ($i = 1$). Constraints (8) describe the relation between jobs in position k and $k + 1$ in function of: idle times between those jobs, waiting times between $m - 1$ workers and processing times. Constraints (9) define decision variable x_{jk} as binary. Finally, Constraints (10) define upper and lower boundaries for the deteriorating factor α.

COMPUTATIONAL EXPERIMENTS

Description of Datasets

In order to test and analyze performance of the proposed model, computational experiments were conducted on a PC AMD Phenom N-640 2.9 Hz and 4,00 GB of RAM memory. The model was programmed using GAMS and solved using MIP solver ILOG CPLEX. To prevent excessive computational time, the solver's computation time limit was restricted to 1000 seconds. All datasets giving the number of jobs, their processing times, and the number of workers were taken from the literature, available at: http://soa.iti.es/instancias-problemas and named as "Benchmarks for flow shops and due dates". Problems with 2, 3, 5 and 10 workers were considered; to process a total of 25, 50, 75, 100, 150, 200, 250, 300 and 350 jobs. Three values for the deteriorating factor (α) were defined: 0 (there is no change in initial processing times), 0.2 and 0.8. Overall, a total of 108 instances were employed in the computational experiment. We next explain the results obtained as shown in Table 1.

Table 1. Results of Computational Experiments

		Deterioration Rate (α)								
		0			0.2			0.8		
Number of Workers	**Number of Jobs**	**Cmax**	**Relative Gap**	**CPU time (s)**	**Cmax**	**Relative Gap**	**CPU time (s)**	**Cmax**	**Relative Gap**	**CPU time (s)**
2	25	1133	0	0.17	1635	0	0.31	5726	0	0.39
	50	2648	0	0.56	4142	0	0.89	22997	0	0.88
	75	3947	0	1.66	7244	0	0.72	51686	0	0.58
	100	5327	0	1.13	10350	0	1.13	88302	0	1.19
	150	8126	0	6.38	17134	0	3.02	188444	0	2.65
	200	10256	0	8.76	20442	0	5.93	235614	0	5.86
	250	13102	0	54.71	27254	0	9.54	359351	0	12.53
	300	15526	0	84.32	34261	0	24.67	521242	0	22.31
	350	17718	0	49.14	41296	0	36.19	699125	0	31.39
3	25	1216	0	0.38	1812	0	0.13	6827	0	0.13
	50	2476	0	1.27	4203	0	3	23753	0	4.19
	75	3948	0	2.34	6737	0	1.34	43518	0	1.05
	100	5328	0	6.63	9692	0	14.19	75522	0	213.5
	150	8127	0	36.7	15916	0	15.06	159578	0	1000*
	200	10266	0	139.23	22366	0	24.19	278903	0	144.87
	250	13112	0	526.37	29558	0	37.14	418975	0	61.05
	300	16105	0.04	1000*	36496	0	113.21	579183	0	40.46
	350	19013	0.0631	1000*	44039	0	171.14	779959	0	80.73
5	25	1288	0	276.78	1915	0	137.62	6467	0	11.31
	50	2547	0.0079	1000	4208	0.009	1000*	21561	0.0016	1000*
	75	3985	0	131.18	7128	0.0002	1000*	49125	0.0007	1000*
	100	5365	0	26.73	10179	0	35.7	84407	0.0003	1000*
	150	8164	0	103.9	16472	0	36.56	172540	0.0002	1000*
	200	11238	0.0835	1000*	22714	0.0042	1000*	284995	0.0003	1000*
	250	13634	0.0358	1000*	29314	0.0019	1000*	412141	0.0002	1000*
	300	16378	0.0493	1000*	36366	0.0064	1000*	565854	0.0002	1000*
	350	-	1	1000*	43010	0.0088	1000*	724027	0.0002	1000*
10	25	1761	0.0368	1000*	2482	0.0317	1000*	8109	0.0032	1000*
	50	3116	0.0183	1000*	4778	0.0217	1000*	22504	0.0043	1000*
	75	4538	0.0134	1000*	7958	0.0369	1000*	50376	0.0042	1000*
	100	5899	0.0175	1000*	11001	0.034	1000*	84892	0.0034	1000*
	150	-	1	1000*	16907	0.0367	1000*	162462	0.0024	1000*
	200	12285	0.1056	1000*	23518	0.0174	1000*	-	1	1000*
	250	-	1	1000*	-	1	1000*	419041	0.0006	1000*
	300	-	1	1000*	-	1	1000*	579867	0.0004	1000*
	350	-	1	1000*	-	1	1000*	-	1	1000*

Results

The analysis of results is carried out over the combinations of workers and jobs, starting with two machines and increasing until ten. All cases are presented in Table 1. The first two columns indicate the number of workers and the number of jobs. The upper row represents the three deteriorating rates considered. The column "Cmax" indicates the value of the objective function (makespan). Column "Relative gap" corresponds to the percentage deviation between the best integer solution found by GAMS solver at the end of the running time of 1000 seconds and the value of the bound defined by the software (if relative gap is zero, it means, the solver found the optimal solution). Finally, computation times are reported in seconds in "CPU time (s)" columns.

For the two-worker instances, the model obtained solutions within the time limit and found the optimal solution for 100% of datasets evaluated. For the instances with three workers, the model obtained solutions within the time limit for all datasets. However, optimal solution was found in 25 (out of 27) instances evaluated (this corresponds to the 92.5% of datasets). For the remaining instances, no optimal solution was reached but very high quality solutions were achieved (average relative gap lower than 0.055).

For the instances with five workers, the model obtained solutions within the time limit in 26 (out of 27) datasets. However, the solver found the optimal solution for only 8 instances evaluated, corresponding to the 30% of datasets. For the remaining 70%, the solver found high quality solutions (relative gap lower or equal than 0.09). Finally, the 10-worker datasets were hard to solve; the solver did not find a solution within the time limit for the 33% of the scenarios. No optimal solutions were found for the remaining 67%, but the solver achieved high quality solutions (equal or lower than 0.04). We observe that for large-sized instances, the percentage of optimal solutions found decrease and the number of solutions found in limit time also decrease.

We observe that the performance of the model varies with the number of workers and the deteriorating job rate (see Figure 1). For instances with two and three workers, the higher the deteriorating job rate, the lower the computation time. Nevertheless, for instances with three workers, computation time increase when deteriorating job rate increase. All this allows concluding that the complexity of the model increases while the size of the instance increases; this statement is supported by the fact that when $m = 10$ workers the computation time exceeded the limit of 1000 seconds.

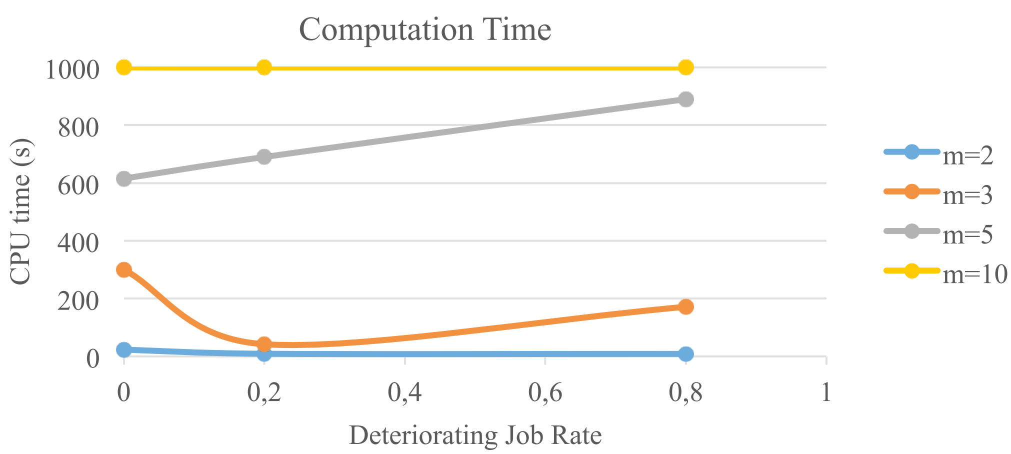

Figure 1 – Relation between computational time and deteriorating job rates for 2, 3, 5, and 10 workers.

CONCLUSIONS AND PERSPECTIVES

This paper considered a permutation flowshop scheduling problem for *m*-workers in which processing times of jobs are dependent on the position in the sequence. A mixed-integer programming (MIP) model was proposed. Computational results showed that the model is able to find optimal solutions in reasonable computation time for small-sized instances. However, for large-sized instances, there is no evidence that the model is able to find optimal values before reaching the upper running time limit; hence the best integer solution is reported. Further research could now be focused on developing cuts to speed-up the resolution of the mathematical model and even on the design of heuristic or meta-heuristic algorithms allowing an efficient solution for large-sized problem datasets.

ACKNOWLEDGEMENT

This paper was developed within the framework of a postgraduate scholarship awarded to the first author by the Master program in Operations Management of the International School of Economics and Management Sciences at Universidad de la Sabana, Chía, Colombia.

REFERENCES

Alfares, H.K. 2002. Optimum workforce scheduling under the (14, 21) days-off timetable. *Journal of Applied Mathematics and Decision Sciences*, **6**(3), 191-199.

Alfares, H.K. 2004. Survey, Categorization, and Comparison of Recent Tour Scheduling Literature. *Annals of Operations Research*, **127**, 145–175.

Aykin, T. (2000). A comparative evaluation of modeling approaches to the labor shift scheduling problem. *European Journal of Operational Research*, **125**(2), 381–397.

Baines, T.S., J.M. Kay. 2002. Human performance modelling as an aid in the process of manufacturing system design: A pilot study. *International Journal of Production Research*, **40**, 2321–2334.

Bentefouet, F., D.A. Nembhard. 2013. Optimal flow-line conditions with worker variability. *International Journal of Production Economics*, **141**(2), 675–684.

Boudreau, J.W., W. Hopp, J.O. Mcclain, L.J. Thomas. 2002. On the Interface Between Operations and Human Resources Management On the Interface Between Operations and Human Resources. CAHRS Working Papers Series. Available at: http://digitalcommons.ilr.cornell.edu/cahrswp/63/ (accessed: December 27, 2015).

Brucker, P. 2007. Scheduling Algorithms (5th ed.). Springer.

Cheng, T. C., Q. Ding, B.M. Lin. 2004. A concise survey of scheduling with time-dependent processing times. *European Journal of Operational Research*, **152**(1), 1-13.

Ernst, A. T.,H. Jiang, M. Krishnamoorthy, B. Owens, D. & Sier. 2004. An Annotated Bibliography of Personnel Scheduling and Rostering. *Annals of Operations Research*, **127**, 21-144.

Gawiejnowicz, S. 2008. Time-Dependent Scheduling. Springer.

Guéret, C., Prins , C., Seveaux, M. (2000). Programmation linéaire : 65 problèmes d'optimisation modélisés et résolus avec Visual Xpress. Eyrolles Editions.

Heizer, J., &B. Render. 2007. Dirección de la producción y de operaciones. Decisiones estratégicas. Dirección de la producción y de operaciones. Decisiones estratégicas (8th ed.). Prentice Hall.

Lee, W.C., C.C. Wu, Y.H. Chung, H.C. Liu. 2009. Minimizing the total completion time in permutation flow shop with machine-dependent job deterioration rates. *Computers and Operations Research*, **36**(6), 2111–2121.

Lodree, E.J., C.D. Geiger, X. Jiang. 2009. Taxonomy for integrating scheduling theory and human factors: Review and research opportunities. *International Journal of Industrial Ergonomics*, **39**(1), 39–51.

Moreno-Camacho, C.A., J.R. Montoya-Torres. 2015. Workforce Scheduling with Social Responsibility Considerations, Proceedings of the 2015 International Conference on Advanced Logistics and Transportation, pp. 24-29.

Neumann, W.P., J. Dul. 2010. Human factors: spanning the gap between OM and HRM. *International Journal of Operations & Production Management*, **30**(9), 923–950.
Pinedo, M.L. 2012. Scheduling: Theory, Algorithms, and Systems. Springer.
Prahalad, C.K. 2004. The Blinders of Dominant Logic. *Long Range Planning*, **37**(2), 171–179.
Thörnblad, K., M. Patriksson. 2011. A note on the complexity of flow-shop scheduling with deteriorating jobs. *Discrete Applied Mathematics*, **159**(4), 251–253.
Vahedi-Nouri, B., P. Fattahi, R. Ramezanian. 2013. Minimizing total flow time for the non-permutation flow shop scheduling problem with learning effects and availability constraints. *Journal of Manufacturing Systems*, **32**(1), 167-173.
Wang, J.B., L.L. Liu. 2009. Two-machine flow shop problem with effects of deterioration and learning. *Computers and Industrial Engineering*, **57**(3), 1114-1121.
Wang, J.B., M.Z. Wang. 2013. Minimizing makespan in three-machine flow shops with deteriorating jobs. *Computers and Operations Research*, **40**(2), 547–557.
Wang, L., L.-Y. Sun, L.-H. Sun, J.-B. Wang. 2010. On three-machine flow shop scheduling with deteriorating jobs. *International Journal of Production Economics*, **125**(1), 185–189.
Wang, X. Y., M.Z. Wang, J.B. Wang. 2011. Flow shop scheduling to minimize makespan with decreasing time-dependent job processing times. *Computers and Industrial Engineering*, **60**(4), 840–844.
Zhao, C., H. Tang. 2012. Two-machine flow shop scheduling with deteriorating jobs and chain precedence constraints. *International Journal of Production Economics*, **136**(1), 131–136.