A SPREADSHEET BASED SCHEDULING SYSTEM FOR A SMALL PRINTING COMPANY

J. Mendoza, G. Mejia

Department of industrial engineering, Universidad de los Andes, Bogotá, Colombia

Abstract

This work presents the design and implementation of a low cost scheduling tool based on Microsoft Excel, designed for a small printing company in Colombia. The company scheduling problem can be defined as a Flexible Job Shop with highly sequence dependent set up times where makespan is the objective function in a weekly planning horizon. The system architecture includes a product specifications data base, an engine which transforms the product specifications into shop floor operations, a processing times data base and a graphical user interface. The proposed scheduling tool uses Simulated Annealing (SA) for optimization purposes. An initial schedule is generated by a non-delay schedule generation algorithm combined with dispatching rules. Insertion moves on each machine were used for the generation of neighbors. The cooling schedule for the implemented SA algorithm is also well explained. In terms of the operation management the results benefit the company with reduction from hours to seconds on the time required to build a feasible schedule and the availability of new tools to generate reports of shop floor performance metrics. In terms of makespan the results suggest improvements of 40%.

Keywords:

Scheduling; Flexible Job Shop; Simulated Annealing; Decision Support Systems.

1 MOTIVATION

Paper and cardboard industry accounts for the 4.4% of Colombian manufacturing. Annually more than 250 enterprises in this sector are responsible for over 1900 million dollar contribution to the total manufacturing production [1]. Historically manufacturing in Colombia is based on medium and small size enterprises (PYMES for its acronym in Spanish) which represent more than 96% of the total number of companies established in the country [1]. According to the 2005 PYMES survey [2], 65% of these companies are interested on investing on information technology solutions (TICS), but only 13% of Colombian PYMES have done it in the last years. The main reasons for this situation are the high cost of buying and implementing this type of technologies and the constrained investment capacity of PYMES.

This work presents the implementation of the scheduling module of a low cost production planning system based on spreadsheets, developed for a small printing company in Colombia. The main goal of this work is to provide the company with a tool to help planning its operations and to improve its productivity. The socialization of this experience pretends to encourage other researchers and companies to develop similar tools to help PYMES to grow and to keep accomplishing their important role in emerging economies.

1.1 Background

Impresos y Empaques D-Cartón Ltda is a small company, specialized in the production of cardboard packing products. The company serves more than 50 clients and produces over 500 different references. Each product is especially designed for the customer constituting a typical make to order setting [3]. Currently the company uses a locally developed spreadsheet-based planning tool (called the Material Planning Module (MPM)) which checks the inventory for raw materials. If inventory is not enough to cover the order, the manager places an order to the carton supplier. Depending on the order size and the cardboard type, the supplier may or may not perform the first two operations (conversion and sizing) of the production process (see Figure 1) and deliver the cardboard cut in sheets. This decision is made by the manager during the material planning process. Further information about the aforementioned MPM can be found in [4].

The machine park is divided in 7 work centers, as seen on figure 1. Conversion and Sizing work centers are composed by a sheeter machine and an automatic guillotine stack cutter respectively. The function of these two centers is the conversion of cardboard rolls into cardboard sheets. As described earlier, for certain jobs the cardboard can be bought in sheets directly from the supplier, in that case the jobs do not go through the first to operations of the process.

The following operations are printing, varnishing, die-cutting, striping and gluing. These operations are executed in the flow direction shown in figure 1. Even though all jobs follow the same fabrication route, not all of them go through every work center. The route is given by the product design.

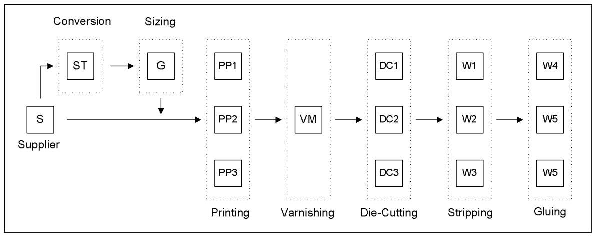


Figure 1: Workflow.

Set up times are sequence dependent on the conversion, sizing, printing, varnishing and die-cutting workstations. In the case of the printing work center set up times vary from 32 to 70 minutes depending on the chosen sequence of operations; the case of the die-cutting center is even more critical, set up times can be zero or 75 minutes depending on the resulting production schedule. Set up times are measured continuously by the personnel and registered. Set up times are also assumed to be deterministic due the small variance observed in the data set.

As presented on figure 1, printing and die-cutting centers have three printing presses and three die cutter machines each with different specifications. The printing quality levels, the maximum and minimum size of the sheet and the maximum caliber of the cardboard that can be cut, arise as constraints for the machines of these two centers. Because of its design, not all printable products can be processed on every press and not all the products that require cutting can be assigned to every die cutter machine.

More than 85% of the products registered on the data base include a printing operation, between those; more than 70% required printing of at least two different colors. The three printing presses installed at the shop floor of the company are single color presses. The setup of these presses is one of the most time consuming activities and considerable savings can be achieved if jobs requiring the same color are processed consecutively.

Summarizing, the scheduling problem to be solved can be characterized as a “flexible job shop framework” (See definition below).The main constraints are: sequence dependent set up times, machine eligibility constraints and recirculation. Due to managerial decisions preemptions are not allowed. From the optimization point of view this is a very hard problem to solve [5].

1.2 Problem description

From the scheduling theory point of view, the problem of sequencing operations in this production system can be framed as a Flexible Job Shop Scheduling Problem (FJSSP). Formally the FJSSP consists of a set *J* of *n* jobs with known operation times that are processed on a finite set *I* of *m* workstations; all operations for each job must be processed on a predefined order (routing). There may be one or more non-identical parallel machines at each work center. The sequence dependent setup time, s_{kj} is defined as the time needed to prepare a machine to execute job *j* when it is set to process job *k*. Finally, the set M_{ij} is defined as the set of machines at workcenter *i* that can process job *j* (Machine eligibility).

1.3 Current scheduling practices

Information about processing and set up times was not taken into account in the scheduling process before the implementation of this production planning system. The production manager based on his experience constructs a schedule using a combined Fist in First out (FIFO) – Earliest Due Date (EDD) dispatching rule [6]. This process is done at the beginning of every week and takes between one and two hours depending on the number of orders to be scheduled.

The main objective of the current scheduling practice is to meet the due dates for the most important customers. This objective is not measured in terms of a mathematical expression as the total weighted tardiness, or the total weighted number of tardy jobs or any due-date related objective describe on the literature, but rather with common service level indicators.

Usually a high percentage of jobs are delivered on time (above 90% for 2006), but a large amount of overtime is usually needed to complete the production schedule. This occurs even though resource utilization rates are not as high as the manager may wish (around 70%).

Literature FJSSP has addressed makespan as the classical objective function of this category of problems. Makespan can be defined as the completion time of the last job to leave the system and its minimization is commonly associated with high resource utilization [7] and maximization of throughput [8].

Currently due dates are being met at an acceptable rate, but some personnel believes that this is because the loose due dates quoted to the customers; it is coherent to think that this objective would still be accomplished at least at acceptable levels even if the objective function is not due date related. On other hand, maximizing utilization of resources by minimizing the makespan might reduce the cost of overtime. In fact, after brainstorming with the staff of the company, makespan was decided to be the objective function for the scheduling system.

2 DC-PRODUCTION PLANNING SYSTEM

DC-Production planning system (DC-PPS) is a modular decision support system based on MS-Excel spreadsheets developed for the company. The core of the system consists of four databases (DB), two decision support (DS) modules and an ABC cost application as shown in figure 2.

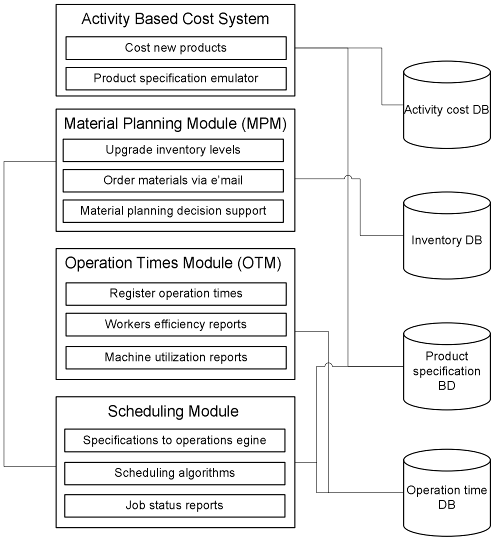

Activity Based Cost System
Cost new products
Product specification emulator
Material Planning Module (MPM)
Upgrade inventory levels
Order materials via e'mail
Material planning decision support
Operation Times Module (OTM)
Register operation times
Workers efficiency reports
Machine utilization reports
Scheduling Module
Specifications to operations egine
Scheduling algorithms
Job status reports
Activity cost DB
Inventory DB
Product specification BD
Operation time DB

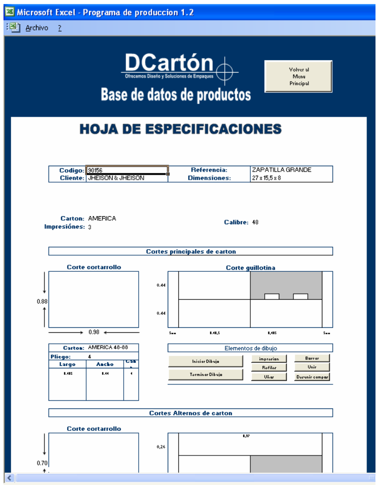
Figure 2: DC-PPS Core

2.1 Relevant system components

Both databases and DS modules are independent Excel workbooks linked by a main manager application. Two components are relevant for the proposed scheduling system, the Product Specification Database (PSDB) and the Operation Time Database (OPDB).

Product Specification Database

The PSDB contains the design specifications of each product. Information includes data such as type and caliber of the cardboard, sheet dimensions, number of colors to print, glue type to be used and other design specifications. Each product record on the PSDB also contains the reference of the printing plates and dies and the machines where the product can be processed. This information permits the scheduling system to build the operation route and the machine eligibility restrictions for each job. The view of the product registration from is shown on figure 3.



HOJA DE ESPECIFICACIONES

Codigo: 90156
Cliente: JHEISON & JHEISON
Referencia: ZAPATILLA GRANDE
Dimensiones: 27 x 15,5 x 8

Carton: AMERICA
Impresiónes: 3
Calibre: 48

Carton: AMERICA 48-88
Pliego: 4

Largo	Ancho	Can
0.485	0.44	4

Figure 3: DC-PPS Product specification input GUI

Operation Time Database

The OPDB contains the information of the processing and setup times for each operation to be executed on the shop floor. Using a computer terminal installed on the shop area, a worker executing an operation registers through bar code tags the time actually spent on an operation. A record is automatically inserted on the Operation Time table (Figure 4). A materialized view of the table containing all the average process and set up times in each machine is updated every month. This view is the source of information used by the scheduling system.

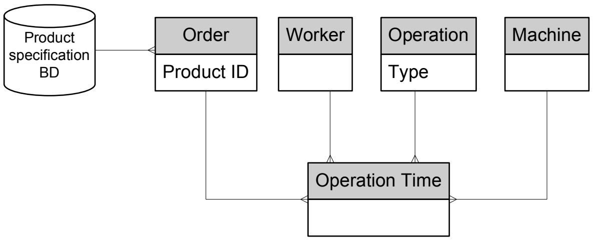


Figure 4: DC-PPS Operation Time Database structure

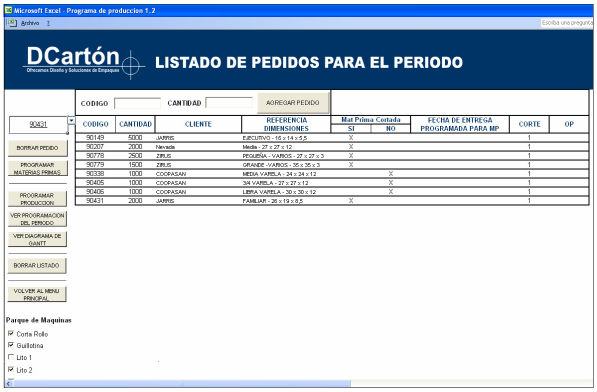
3 SCHEDULING MODULE

3.1 Input data

Orders are introduced on the system using the graphical user interface (GUI) shown on figure 5. To schedule an order, its status must have been set on “schedulable” on the MPM; the “schedulable” status is reached only when raw materials for an order are delivered by the supplier or confirmation of internal inventory availability is given by the MPM, this guarantees that all jobs have a release day of zero. The list of jobs to be schedule is built by selecting all jobs from the schedulable job pool in the MPM.

When the scheduling process begins, the system identifies the product code of each order and searches the corresponding database for the information of operations and machine eligibility to build the data structures to be used by the scheduling algorithm. Set up times are calculated in real time by the scheduling algorithm depending on the solution being evaluated.

It is also important to note that through the GUI on figure 5 the user may indicate to the scheduling algorithm which of the machines will be available for processing by checking the corresponding boxes. Jobs that can only be processed on not available machines are either dropped or considered for the next planning week.



LISTADO DE PEDIDOS PARA EL PERIODO

CODIGO CANTIDAD

90431

CODIGO	CANTIDAD	CLIENTE	REFERENCIA DIMENSIONES	Mat Prima Cortada		FECHA DE ENTREGA PROGRAMADA PARA MP	CORTE	OP
				SI	NO			
90149	5000	JARRIS	EJECUTIVO - 16 x 14 x 5,5	X			1	
90207	2000	Nevada	Media - 27 x 27 x 12	X			1	
90778	2500	ZIRUS	PEQUEÑA - VARIOS - 27 x 27 x 3	X			1	
90779	1500	ZIRUS	GRANDE -VARIOS - 35 x 35 x 3	X			1	
90338	1000	COOPASAN	MEDIA VARELA - 24 x 24 x 12		X		1	
90405	1000	COOPASAN	3/4 VARELA - 27 x 27 x 12		X		1	
90406	1000	COOPASAN	LIBRA VARELA - 30 x 30 x 12		X		1	
90431	2000	JARRIS	FAMILIAR - 26 x 19 x 8,5	X			1	

Parque de Maquinas

☑ Corta Rollo
☑ Guillotina
☐ Lito 1
☑ Lito 2

Figure 5: DC-PPS Scheduler GUI

3.2 Scheduling Algorithm

The implemented scheduling method is based on the Simulated Annealing algorithm. Simulated Annealing (SA) is a local search meta-heuristic for combinatorial optimization problems [9] presented by Kirkpatrick in 1983 [10]. The name is derived from its analogy with the annealing process in metallurgic. The annealing process consists of heating up a solid and then cooling it down slowly [11]. As the solid cools down it goes through stages of quasi-equilibrium in which the temperature remains approximately constant. As the temperature decreases, the solid reaches states with lower energy levels. However there is a small probability of reaching states of higher energy. In terms of optimization a “state” corresponds to a solution. The energy level is analog to the value of the objective function. The algorithm proceeds by moving from higher energy states to states with lower energy levels. In order to avoid local optima the algorithm may choose (according to a probability) to move to states with higher energy levels. Such a probability increases with the temperature (or control factor) and decreases with the energy difference of any two states.

The probability of moving to a higher energy level follows the Boltzmann and it is given by equation (1).

$$P(E) = e^{\frac{-\Delta E}{cT}} \qquad (1)$$

Where E is the energy of the system, T is the current temperature and c the Boltzmann constant.

As iterations of the simulated annealing algorithm are completed, the temperature decreases following a cooling schedule. When the temperature is “high” the search is diversified as more “bad” (solutions with higher energy levels) moves are allowed. On the other hand, when the temperature is low, fewer “bad” moves occur and the search is intensified.

When designing a simulated annealing algorithm, four important factors must be taken into account [10]: (i) The objective function, (ii) the solution representation, (iii) the neighborhood design and (iv) the cooling schedule. Several schemes using different combinations of the above factors were tested; the ones selected and implemented on the SA on DC-PPS scheduler are presented next.

Objective function

As proposed in section 1.2 the objective function evaluated by the scheduler is makespan. Let *J* be the set of jobs, *I* the set of machines, *N* the set of all (*i,j*) operations, $p_{i,j}$ the processing time of job *j* on machine *I* and $y_{i,j}$ the starting time of job *j* on machine *i*. Makespan (denoted by C_{max}) can be calculated as follows [7].

$$C_{Max} \geq y_{i,j} + p_{i,j} \quad \forall (i,j) \in N \qquad (2)$$

Every time a new solution is generated, the scheduler evaluates the objective function by calculating the completion time of each operation of all jobs in set *J* and storing the maximum value found.

Schedule representation

A Multi Permutation Representation (MPR) based on Genetic Vehicle Representation (GVR) [12] was selected. In the implemented MPR, *c* dynamic arrays contain the sequence in which jobs are to be processed on each of the *c* work centers of the shop. To evaluate the objective function, the MPR must be parsed into a schedule; this parsing is done by a special procedure named “Schedule

Builder". The Schedule Builder checks each position (from first to last) of a permutation, sequencing the jobs on the corresponding work center on the same order. As explained on section 1.1, four out of seven work centers on the shop have more than one machine. In this case, the job that is being scheduled is assigned to the first available machine or worker (if the machine eligibility constraint is met). In the printing and Die-Cutting work centers when more than one machine is available, the job is assigned to the one with lowest machine load [13]. If ties persist the job is assigned by lexicography.

Table 1 shows the route of four actual products with modified processing times. Table 2 shows the machine eligibility constraint for the four products in Printing and Die-Cutting work centers. Figure 6 presents the MPR of a feasible solution for example 1. Figure 7 presents the solution after been parsed by the schedule builder. For simplicity all set up times in this example are assumed as zero.

Table 1: Example 1, Processing Times

Job	C1	C2	C3	C4	C5	C6	C7
1	5	5		10	10	5	
2			10/10		10	5	15
3			10				
4	10	10	15		15	10	

Where C1: Conversion, C2: Sizing, C3: Printing, C4: Varnishing, C5: Die-Cutting, C6: Stripping and C7: Gluing work centers respectively.

Table 2: Example 1, Machine Constraint

Job	C3: Printing			C5: Die-Cutting		
	PP1	PP2	PP3	DC1	DC2	DC3
1				x	x	x
2		x		x		
3	x					
4		x			x	x

Neighborhood design

Three different neighborhood designs based on two different types of moves [14] (Insert and swap) were implemented and tested.

Insert exchanges consist in randomly selecting two jobs *j* and *k* on a permutation, job *k* is then inserted on the position of job *j*. When *k* > *j* jobs from *j* to *k-1* are moved one position forward on the permutation, otherwise jobs from *j* to *k+1* are moved one position backwards.

Swap exchanges require a pair of jobs *j* and *k* to be selected in a random fashion; In the exchange, job *j* takes the position occupied by job *k* on the permutation and job *k* is then sequenced on *j* position. Figure 7 shows both schemes applied to a single permutation selected also randomly.

A third neighborhood design consists on making swap exchanges on all the permutation of a solution simultaneously.

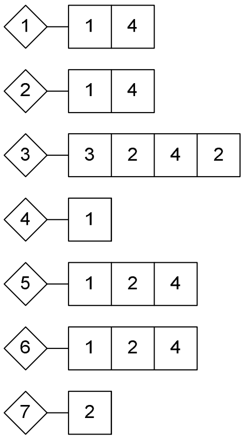


Figure 6: MPR of a feasible solution for example 1

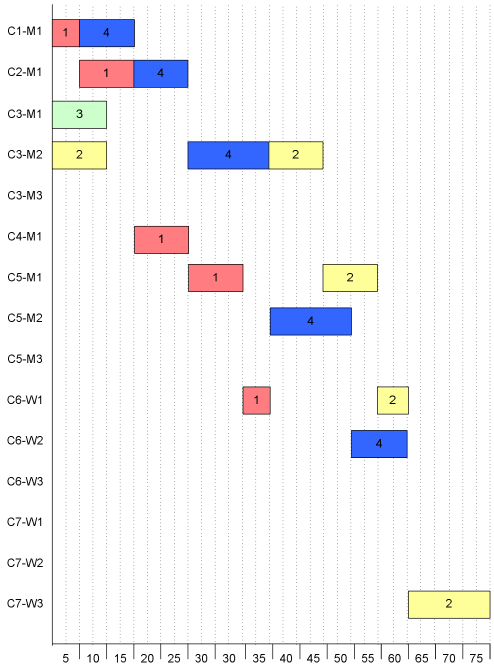


Figure 7: Feasible solution for example 1 parsed into a schedule

Because of the nature of the process operations at any work center for a given job can not start until operations of the same job are finished in the predecessor work center, this condition guarantees that none of the neighborhood designs proposed can produce unfeasible solutions for any instance of the problem even if some of the products to be scheduled involve recirculation on the printing work center (i e, Job 2 on Example 1).

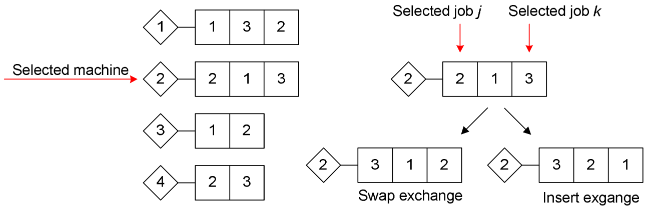


Figure 7: Neighborhood design

Initial solution

Three different generation schemes for the initial solution were tested, random generation and the well known Giffler and Thompson (G&T) enumerative algorithms for active and non-delay schedules [15]. To break ties on the G&T, two suggested [16] priority dispatching rules are used. In this system Shortest Processing Time (SPT) and Most Work Remaining (MWR) were used.

Cooling schedule

The cooling schedule includes the initial temperature, a formula for changing the temperature and the stopping criterion. To set initial temperature, the scheduler uses the value of the objective function obtained of the initial solution and the value of the objective function of a randomly generated solution. The difference between these two values is taken as ΔE in (1), from where the value of T can be obtained by given $P(E)$ a desired value. As the intention of SA is to perform a diversified search on the first iterations, the given value of $P(E)$ for the initial temperature calculations should be high [9]. For the DC-PPS scheduler this parameter was fixed on 95. The chosen formula to reduce the temperature is a stepwise reduction scheme known as the geometric cooling rule [11]. Under this rule, K iterations are made at each temperature level, when such iterations are completed the temperature is reduced by a factor α ($0 < \alpha < 1$). Let T_i be the current temperature, T_{i+1} the updated value of temperature and α the cooling factor: The new temperature level is given by (3).

$$T_{i+1} = \alpha T_i \qquad (3)$$

The stopping criterion is a predefined number of changes n_t on the temperature level without improving the best solution found. Values for α, K, and n_t have been fixed on 0.95, 10 and 1000 respectively. These values were set after a number of trial tests.

3.3 Output Data

The output data can be presented in two different modalities, the user can chose between a production program for each machine or a Gantt chart with the complete schedule of the plant (figure 8).

To control production process the scheduling module includes a tool that allows the company to know the exact status of an order. A query to the Operation Time table of the OTDB, returns a data set that allows the scheduling system to present to the user the Gantt chart of the completed operations for an order; this is feature that allows to forecast the total the completion time of a job in case a customer wants information about his/her order.

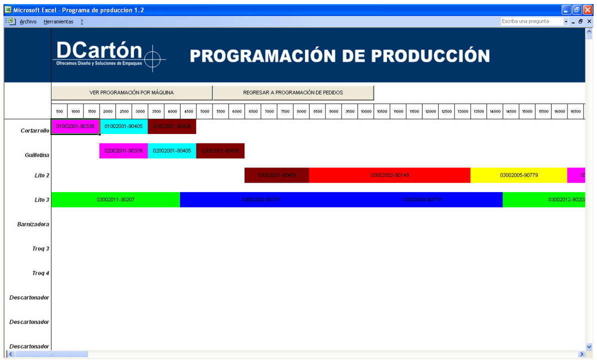


Figure 8: DC-PPS Scheduler output, Gantt chart

3.4 Results

Implementation of the scheduling system has just been finished; databases are still being populated with information of all products and operations and the system has not entered operational mode yet.

Experimentation with real instances shows promising results. An instance is defined as the pool of orders that must be process in a week; to test de system, the orders for the fifty weeks of the last year were considered, among those, ten weeks were randomly selected as benchmark instances.

The best results in terms of the objective function were achieved using the non-delay schedule algorithm in combination with the MWR dispatching rule as the initial solution generation mechanism, and insert exchanges on a single machine as the neighbor design. This combination not only offers the best values for the objective function but also the best computational times. For each instance 10 were run on a Pentium M 1,67GH, 512Mb RAM computer with windows XP operative system. Table 4 summarizes the experiments on instance 1.

Experimentation shows that the system outperforms the current scheduling method by 40% in terms of makespan as the due dates are still met at the current rates. Table 5 presents the results. $\sum U_j$ is the number of tardy jobs[7].

Table 3: Benchmark on test instances

W	Jobs	Current Schecluling Method		Proposed Scheduling System		Impvnt
		$C_{\max}$ (h)	$\sum U_j$	$C_{\max}$ (h)	$\sum U_j$	
1	8	53,294	0	37,009	0	44,00%
2	8	58,192	0	43,105	0	35,00%
3	7	51,936	0	34,856	0	49,00%
4	5	70,247	1	52,423	1	34,00%
5	8	48,104	0	36,721	0	31,00%
6	13	54,403	2	38,044	1	43,00%
7	5	59,013	0	40,145	0	47,00%
8	10	64,391	1	48,053	0	34,00%
9	11	60,47	0	42,287	0	43,00%
10	11	60,55	1	42,049	0	44,00%
		Mean	0,054	-	0,028	40,40%

Table 4: Computational experiments on test instance 1

4 CONCLUSIONS AND FUTURE RESEARCH

A low cost scheduling system for a small printing company was designed and implemented based on spreadsheet databases and VBA macros in MS Excel, to solve a flexible job shop problem with machine eligibility constraints, sequence dependent set up times and recirculation having the minimization of makespan as the objective function. Results are encouraging as the system outperforms the current scheduling method in use by over 40% in very reasonable computational times. This improvement in the utilization of the work centers is expected to reduce the amount of overtime currently hired by the company in order to meet the production program. The system also helps the company to control the execution of the production plan by implementing a computer tool to register the status of an order every time a production operation is complete on the shop floor. Reports on machine utilization and workers productivity can also be generated using this tool.

Initial Solution	**Tie Break Criterion**	**Neighborhood**	$C_{\max}$ **(h)**	**Time (S)**
Random Generation		Insert	42,983	16,1
Random Generation		Swap MultipleMachine	41,587	19
Random Generation		Swap SingleMachine	41,035	15,7
Active Schedules		Insert	43,438	16,1
Active Schedules		Swap MultipleMachine	42,301	19,4
Active Schedules		Swap SingleMachine	41,976	15,5
NonDelay Schedules	FIFO	Swap MultipleMachine	41,912	14,1
NonDelay Schedules	MWR	Swap MultipleMachine	37,009	14,9
NonDelay Schedules	FIFO	Insert	41,489	20,6
NonDelay Schedules	MWR	Insert	37,009	12,2
NonDelay Schedules	FIFO	Swap SingleMachine	41,879	13,9
NonDelay Schedules	MWR	Swap SingleMachine	37,009	13,4

Database systems based on spread sheets like the one used on this system, can be a solution to small companies where there is a few data to be stored and there is not need to handle concurrency or transactions; If a system alike this is being consider to solve scheduling problems of a higher scope, it is strongly recommendable to support the databases on the appropriated database technology.

On future research, it is the interest of the authors to use this experience as a start point to design a more general purpose spread sheet based system that can be used by other sector small companies. Also to implement more sophisticated scheduling algorithms that support multiple objective functions.

5 ACKNOWLEDGMENTS

The authors would like to express their thankfulness to the staff of Impresos y Empaques D-Cartón Ltda. in Bucaramanga, Colombia.

6 REFERENCES

[1] National Administrative Department of Statistics. Colombia. Annual survey of manufacturing, 2005 http://www.dane.gov.co/index.php?option=com_content&task=category&sectionid=17&id=43&Itemid=154last accessed on March 15, 2007 (In Spanish)

[2] National Association of Small and Medium Sized Enterprises of Colombia. Annual Balance and Perspectives report, 2005. http://www.acopibogota.org.co/catalogo/docs/BALANCE_PYME_2005.doc last accessed on March 15, 2007. (In Spanish)

[3] Nahmias, S., Production and operation management. McGraw Hill Higher Education, 5Rev Ed edition 2004.

[4] Mendoza, J., Bula G., Design of an integrated architecture for production planning in a printing company, 2004, Industrial Engineering Monograph, Universidad Industrial de Santander. Colombia.

[5] Rinooy, A. H., Machine scheduling problems: classification, complexity and computations, Nijhoff, The Hague.

[6] Panwalkar, S., Iskander, W., A survey of scheduling rules. Operation Research. 25, 1, 1977, 45 – 61.

[7] Pinedo, M., Scheduling. Prentice Hall, 2002.

[8] P. Damodaran and K. Srihari, Mixed integer formulation to minimize makespan in a flow shop with batch processing machines, Mathematical and Computer Modelling, 40, 13, 2004, 1465-1472.

[9] Henderson, D., Jacobson S., Johnson, A., The theory and practice of simulated annealing. Handbook of meta-heuristics, 2003, Kluwer Academic Publishers, 287-319.

[10] Kirkpatrick, S., GELATT, C. D., VECCHI, M. P., Optimization by simulated annealing. Science, 1983, 220, 671-680.

[11] Pham, D.T., Karaboga, D Intelligent Optimization Techniques. Springer, 2000, 11.

[12] Pereira

[13] Koonce, D.A., Tsai,S.C., Using data mining to find patterns in genetic algorithm solutions to a job shop schedule. Computers & Industrial Engineering, 38, 2000 361-374.

[14] Bozejko, W., Grabowski,J., Mieczyslaw, W., Block approach tabu search algorithm for single machine total weighted tardiness problem. Computers & Industrial Engineering, 2006, 1 – 14.

[15] Giffler, B., Thompson G.L., Algorithms for solving production scheduling problems. Operations Research, 1960, 8, 4, 487-503

[16] Chang, Y., Sueyoshi, T., Sullivan, R. Ranking dispatching by data envelopment analysis on job shop environment. IIE Transactions, 1996, 28, 631 – 642.