A SCHEDULING MODEL FOR A KNITTING PLANNING PROBLEM

Carina Pimentel[1], Filipe Alvelos,[1,2], António Duarte[1,3], J. M. Valério de Carvalho[1,2]

[1]Algoritmi Research Center – University of Minho, Braga, Portugal
[2]Production and Systems Department – University of Minho, Braga, Portugal
[3]Industrial Management Department – Polytechnic Institute of Bragança, Bragança, Portugal

ABSTRACT

In this paper we present two planning and scheduling models for a real problem of a textile industry that produces fine knitted goods. In both of them we develop plans to assist the knitting planning of one of three knitting subsections. In this problem we intend to assign and sequence, within a set of available and identical parallel machines, the demand associated with each component or garment part. This demand can be split in lots of smaller quantities and these lots can be independently produced at any time in one or more of the available machines. In the first model we develop a mixed integer programming (MIP) formulation and in the second one we develop network flow based models and a scheduling heuristic. The main advantage of the second model, in opposition to the first one, is the small computational resources needed to solve this huge and complex problem. We solve an instance generated in accordance with the characteristics of the real problem by the second model and present some performance measures.

Keywords: planning and scheduling, mixed integer programming, network flows

INTRODUCTION AND MOTIVATION

The aim of this study is to develop a planning tool to assist the knitting production planning of a multinational fully fashion textile industry. This tool will solve in an integrated way two problems: (1) the lot sizing problem (or lot splitting problem); and (2) the assignment and scheduling problem in which the lots determined in (1) are assigned and scheduled over sets of machines in parallel.

Our motivation for studying this problem arose from the interest of this company to improve their actual scheduling plans and to increase its efficiency, and to automate the knitting production planning.

The work we present in this paper is related with one of the three knitting subsections, which we will refer as cotton subsection this point forward. The planning of the other two subsections will be done based on the schedule obtained for the cotton subsection.

We developed a mixed integer programming model for the cotton subsection problem. This is a complex model for which exact solutions cannot be easily obtained. It is worth mention that this is a huge real-life problem, where solution times are a very important concern for two main reasons: (1) the need for updated production schedules as the available information changes; (2) the need for obtaining schedules in real-time. We also devised a heuristic approach, based on iteratively solving network flow problems associated with scheduling of the different sets of items. As demonstrated by the computational tests this alternative method is very fast.

LITERAURE REVIEW

In the literature there seems to exist two streams dedicated to the study of the division of the demand in smaller quantities: (1) the lot sizing stream, in which the demands are divided in smaller lots with the aim of minimizing a sum of costs, namely storage and setup costs; and (2) the scheduling stream. In this case usually the lot sizing is known as lot splitting and typically the aims of the problem are not related with cost based objectives, but with time based objectives. (Drexl and Kimms, 1997) surveyed the main mathematical programming models that integrate lot sizing and scheduling, in a single level and multi-level basis. They emphasize models where the planning horizon is separated into several discrete-time periods of equal length and cost based objective functions. Two review papers more related with the scheduling stream are the ones of (Allahverdi et al., 1999) and (Zhu and Wilhelm, 2006). Both of them consider research involving setups and present the literature categorized by the shop environment/machine configuration. In (Zhu and Wilhelm, 2006) both optimization and heuristic solution methods are reviewed and a section of the paper is dedicated to the works that integrate lot sizing and scheduling problems. (Zhu and Wilhelm, 2006) argue that the literature for the combined lot sizing and scheduling problem for parallel machine configuration is sparse. They also mention that there are fertile opportunities available for research addressing due date related objectives. We also noted that fact and, as far as we know, there are not published papers addressing the minimization of the deviation between the conclusion times of all the components belonging to the same final item, which we study in here. (Allahverdi et al., 1999) also argue that due date related objectives need to be emphasized specially in parallel machines and multi-stage scheduling configurations.

There are few papers considering continuous time based MIP models for the combined lot sizing and scheduling problem. (Chen and Ji, 2007) present a continuous time based MIP model that integrates MRP and scheduling, thus considering all the product structure. The main characteristics of their model are: lot-for-lot strategy used in the lot sizing, setup times are not considered, non-preemptive tasks, parallel machines, capacities, operation sequences, lead times and due dates. Their goal is to minimize the production idle-time plus the earliness and tardiness. The authors present results for an instance of two final items, with four levels, five machines and five orders, solved exactly by the Cplex software (ILOG, 2008). Another recent paper from (Doganis and Sarimveis, 2007) presents a continuous time based MIP model to a yogurt production line that integrates lot sizing and scheduling. They consider sequence dependent setup times and their goal is to minimize the setup, the inventory and the labor costs. They use software Cplex (ILOG, 2008) to solve the MIP model.

(Sheen and Liao, 2007) present a network flow technique to solve a preemptive scheduling problem with identical parallel machines that have availability constraints associated. Their goal is to minimize the maximum lateness. In their problem, each job is only allowed to be processed on specific machines. They solve this problem using a series of maximum flow problems. They propose a polynomial time two-phase binary search algorithm to verify the feasibility of the problem and to solve the scheduling problem optimally if a feasible schedule exists. In their paper the authors argue that this problem is seldom studied in the literature. This problem is close to the one we study, although they have a major difference. In our problem we allow lot splitting while in (Sheen and Liao, 2007) this is not possible. Though they allow preemption, they do not allow the same item to be produced at the same instant in different machines. (Yalaoui and Chu, 2003) and (Tahar et al., 2006) propose a heuristic algorithm to solve the identical parallel machine scheduling problem with sequence dependent setup times and job splitting to minimize makespan. They solve the problem in two phases. In phase 1 the problem is reduced into a single machine scheduling problem with sequence dependent setup times. They transform this problem into a TSP and solve it using Little´s method. In phase 2, (Yalaoui and Chu, 2003) try to improve the solution obtained in phase 1 in a step by step manner, taking into account the setup times and the job splitting. (Tahar et al., 2006) solve phase 2 by a linear program to determine the dimension of the lots. (Yalaoui and Chu, 2003) emphasize that no work has been yet published for this problem. The main difference between our problem and this one is associated with the objectives. Besides, the proposed solution methods are different.

Papers concerning lot streaming are seldom studied in the literature. There are few papers considering this feature (for example (Serafini, 1996) and (Dauzère-Pérès and Lasserre, 1997)) but they do not consider setup times.

PROBLEM STATEMENT

We begin this section by giving a brief description of the company. It produces about 1.300.000 final items per year that are splitted among an average of 4.300 different references. The company is divided in four productive sections: knitting, linking, dyeing and finishing. Our focus is on the knitting section. This section is dedicated to the production of the main components of the final item. Some types of components produced in this section are: garment parts (sleeves, back bodies, front bodies, etc.), ribs, cuffs, pockets, collars, etc.

In the knitting section there are different types of machines used in the production of those components. The garment parts are produced in cotton machines, the ribs, cuffs and others usually are done in Protti machines and the collars, pockets, and others in Protti, Shima and Dubied machines. All the machines have a gauge associated which is determinant to the production planning phase, as the planning is done by gauge. As an example we mention the cotton machines where there are three different gauges: 21, 24 and 27. This will force the existence of three productions plans, one for each gauge.

Between the ribs and cuffs and the garment parts there are precedence relations because the ribs and cuffs are embodied in the garment parts.

The knitting section can be further divided into three subsections according to the type of components produced (garment parts; ribs, cuffs and others; and collars, pockets and others) and to the type of machines used.

The company works only with ordered work, having four types of orders, namely prototype orders, assortment orders, pre-serie orders and manufacturing orders. An order can include several different final items, in several sizes and colors. In the knitting phase the colors are not important since usually the items are produced using raw yarns. An exception is the production of items with stripes. A production order will state that a given machine will produce a component X in size Y of the final item Z. Thus a given order will have associated several production orders.

In order to avoid misunderstanding we distinguish final items from items. An item has associated a final item/component/size while a final item is associated to an end product without considering sizes or components.

In the present, the assignment and the sequencing decisions at the knitting subsection are taken manually by a scheduler, based on a production priorities document that contains the sequence in which the customer orders should be done. This document considers all the accepted manufacturing orders and is sent every Friday to the scheduler. The prototype, assortment and pre-serie orders are not included in the production planning and are carried on by a full dedicated operator. Usually those orders are done in machines dedicated to their production or in machines dedicated to the manufacturing orders. In this case, the full dedicated operator asks the operator that is working in the selected machine to stop the manufacturing production order and to do the prototype, assortment or pre-serie order.

When the manufacturing production orders, which contain the quantities to produce of a given component of a given final item in a given size, are sent to the section, they have already been splitted in lots. In the present the lots are created by final item and have a dimension in the range of 800 to 999 pieces. These lots contain the different clothe sizes. The lot sizing and the scheduling decisions are taken in different phases of the process and by different entities. The lot sizing decisions are taken by the planning department and the sequencing decisions are taken by the knitting manager.

The tool to develop will solve in an integrated way the lot sizing and scheduling problem. In a solution to the problem, the quantities to produce by item are splitted among smaller lots of variable size, the machines in which those lots will be produced are determined, as well as the order among which they will be produced. Each lot can be independently produced at any time in one or more of the suitable machines. The lot sizing decisions are taken at the same level and in coordination with the sequencing decisions, thus increasing the level of accuracy of the solutions.

As the work developed in this text is related with the subsection where the garment parts are produced, which we denote by cotton subsection, from now on we will concentrate us in this subsection. The problem of the cotton subsection considers: sets of identical parallel machines, demands and due dates associated with each final item, unit production times, a compatibility matrix between machines and items, release dates of machines, sequence dependent setup times, lot splitting and a weekly planning horizon.

The solution objectives to reach are: (1) minimize total tardiness and (2) minimize the deviation between the conclusion time of an item and the conclusion time of all the other items that belong to the same final item. This second aim is considered by the company of major importance since it will allow a reduction in the work in progress and an increase in the process flow.

MIXED INTEGER PROGRAMMING MODEL

The mixed integer programming model that is described in this section is associated with the cotton subsection. The solution of the other two subsections will be determined based on the solution of the cotton subsection. The reason for beginning by this subsection is because it presents the tightest capacity and requires a very good planning.

The MIP model we present below considers all the aspects mentioned during the presentation of the real problem in the above section.

Consider the sets, parameters and decision variables presented below:

Sets:
J - set of products in raw (or final items/sizes)
J_1 - set of cotton items
M_1 - set of cotton machines
K - set of runs
I - set of days
$S(j)$ - set of cotton items of product in raw j, in a lower level of the bill of materials

Parameters:
D_j - demand of product in raw j
r_m - ready time of machine m
Υ_m - item to which machine m is prepared at the beginning of the planning horizon
a_j - production unit time of item j
s_{jl} - setup time spent in a changeover from item j to item l
f_{jl} - number of units of item j required to produce one unit of product in raw l ($j \in S(l)$)

β_j - cost of lateness of product in raw j per unit time
d_j - due date of product in raw j
h_{mi} - clock time when day i finishes on machine m
θ_j - cost of a production run of item j
α_j - per unit time cost resulted from the deviation between the production completion time of product in raw j and each of its childs
b_{jm} - is equal to 1 if item j can be processed on machine m and is equal to 0 otherwise
M - a big number

Decision Variables:
X_{jmki} - quantity to produce of item j during the k^{th} production run of day i on machine m
T_{jmki} - finish time of item j on the k^{th} production run of day i on machine m
C_j - production completion time of product in raw j
L_j - lateness of product in raw j
Z_{jmki} - is equal to 1 if item j is produced on the k^{th} production run of day i on machine m and is equal to 0 otherwise.
U_{jmki} - is equal to T_{jmki} if item j is produced on the k^{th} production run of day i on machine m ($Z_{jmki}=1$) and is equal to C_j if item j is not produced on the k^{th} production run of day i on machine m ($Z_{jmki}=0$).
$W1_{jmki}$ - is equal to 1 if $Z_{jmki}=0$ and is equal to 0 if $U_{jmki} \le T_{jmki}$.
$W2_{jmki}$ - is equal to 1 if $Z_{jmki}=1$ and is equal to 0 if $U_{jmki} \le C_l \mid j \in S(l)$
Q_{jlmki} - is equal to 1 if a changeover from item j to item l occurs on the beginning of the k^{th} production run of day i on machine m and is equal to 0 otherwise.

Note: the index $j \in J_l$ is constituted by: product in raw reference + component reference + size + order number. For example a sleeve (S) of a product in raw A in size 1 of order x will be represented by: SA1x.

The MIP model developed for the cotton subsection is:

$$Min \sum_{l \in J} \sum_{j \in J_l | j \in S(l)} \sum_{m \in M_l | b_{jm}=1} \sum_{k \in K} \sum_{i \in I} \alpha_l (C_l - U_{jmki}) + \sum_{j \in J_l} \sum_{m \in M_l | b_{jm}=1} \sum_{k \in K} \sum_{i \in I} \theta_j Z_{jmki} + \sum_{l \in J} \beta_l L_l$$

subject to:

$$\sum_{m \in M_l | b_{jm}=1} \sum_{k \in K} \sum_{i \in I} X_{jmki} = f_{jl} D_l, \ \forall l \in J, \forall j \in J_l \mid j \in S(l) \quad (1)$$

$$T_{jm11} \ge r_m + a_j X_{jm11} + st_{\gamma_m j} Q_{\gamma_m jm11} - M(1 - Q_{\gamma_m jm11}), \ \forall j \in J_l \mid b_{jm} = 1, \forall m \in M_l \quad (2)$$

$$T_{jmki} \ge T_{lm,k-1,i} + a_j X_{jmki} + s_{lj} Q_{ljmki} - M(1 - Q_{ljmki}), \ \forall j,l \in J_l \mid b_{jm} = 1 \text{ and } b_{lm} = 1, \forall m \in M_l, k = 2,\ldots,K_{max}, \forall i \in I \quad (3)$$

$$T_{jm1i} \ge T_{lm,k_{max},i-1} + a_j X_{jm1i} + s_{lj} Q_{ljm1i} - M(1 - Q_{ljm1i}), \ \forall j,l \in J_l \mid b_{jm} = 1 \text{ and } b_{lm} = 1, \forall m \in M_l, i \ge 2 \quad (4)$$

$$C_l \ge T_{jmki} - M(1 - Z_{jmki}), \ \forall l \in J, \ \forall j \in J_l \mid b_{jm} = 1 \text{ and } j \in S(l), \forall m \in M_l, \forall k \in K, \forall i \in I \quad (5)$$

$$T_{jmki} \le h_{mi}, \ \forall j \in J_l \mid b_{jm} = 1, \forall m \in M_l, \forall k \in K, \forall i \in I \quad (6)$$

$$T_{jm1i} - a_j X_{jm1i} - s_{lj} Q_{ljm1i} \geq h_{m,i-1}(1-(1-\sum_{l \in J_1 | b_{lm}=1} Q_{ljm1i})),\ \forall j,l \in J_1 \mid b_{jm} = 1 \text{ and } b_{lm} = 1, \forall m \in M_1, i \geq 2 \tag{7}$$

$$L_l \geq C_l - d_l,\ \forall l \in J \tag{8}$$

$$Z_{jmki} + W1_{jmki} = 1,\ \forall j \in J_1 \mid b_{jm} = 1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{9}$$

$$U_{jmki} - T_{jmki} \leq MW1_{jmki},\ \forall j \in J_1 \mid b_{jm} = 1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{10}$$

$$Z_{jmki} = W2_{jmki},\ \forall j \in J_1 \mid b_{jm} = 1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{11}$$

$$U_{jmki} - C_l \leq MW2_{jmki},\ \forall l \in J,\ \forall j \in J_1 \mid b_{jm} = 1 \text{ and } j \in S(l), \forall m \in M_1, \forall k \in K, \forall i \in I \tag{12}$$

$$\sum_{l \in J_1 | b_{lm}=1} Q_{\gamma_m lm11} = 1,\ \forall m \in M_1 \tag{13}$$

$$\sum_{j \in J_1 | b_{jm}=1} Q_{jlmki} = \sum_{r \in J_1 | b_{rm}=1} Q_{lrm,k+1,i},\ \forall l \in J_1 \mid b_{lm} = 1, \forall m \in M_1, k = 1,\ldots,K_{max} - 1, \forall i \in I \tag{14}$$

$$\sum_{j \in J_1 | b_{jm}=1} Q_{jlmK_{max},i-1} = \sum_{r \in J_1 | b_{rm}=1} Q_{lrm1i},\ \forall l \in J_1 \mid b_{lm} = 1, \forall m \in M_1, i \geq 2 \tag{15}$$

$$\sum_{j \in J_1 | b_{jm}=1} \sum_{l \in J_1 | b_{lm}=1} Q_{jlmk1} = 1,\ \forall m \in M_1, k \geq 2 \tag{16}$$

$$\sum_{j \in J_1 | b_{jm}=1} \sum_{l \in J_1 | b_{lm}=1} Q_{jlmki} = 1,\ \forall m \in M_1, \forall k, i \geq 2 \tag{17}$$

$$\sum_{j \in J_1 | b_{jm}=1} Z_{jmki} \leq 1,\ \forall m \in M_1, \forall k \in K, \forall i \in I \tag{18}$$

$$X_{jmki} \leq MZ_{jmki},\ \forall j \in J_1 \mid b_{jm} = 1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{19}$$

$$X_{jmki} \leq M \sum_{l \in J_1 | b_{lm}=1} Q_{ljmki},\ \forall j \in J_1 \mid b_{jm} = 1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{20}$$

$$Y_l, X_{jmki}, T_{jmki}, C_l, L_l \text{ and } U_{jmki} \geq 0\ \ \forall l \in J, \forall j \in J_1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{21}$$

$$Z_{jmki}, W1_{jmki}, W2_{jmki} \text{ and } Q_{jlmki} \in \{0,1\},\ \forall j \in J_1, \forall m \in M_1, \forall k \in K, \forall i \in I \tag{22}$$

In the objective function the deviations between the conclusion time of a product in raw and the conclusion time of all the items that belong to this product in raw are minimized, as well as the lateness. The second term of the objective function is a penalty cost for having a production run, that can be considered a dummy cost. This cost is only considered because of constraints *(5)*. Constraints

(1) are the demand constraints. These constraints associate the demand of a product in raw with the demand of its childs. The quantities to produce of each child (X_{jmki}) can be sized in lots of smaller quantities and can be produced in several machines at the same time or at different times. Constraints *(2)*, *(3)* and *(4)* give the conclusion time of each item in each machine, run and day. In constraints *(5)*, the completion time of a given product in raw is determined. Constraints *(6)* and *(7)* are the capacity constraints. They associate the conclusion time of each item in each machine, run and day with the finish time of a day. In constraints *(6)* the conclusion time of each item in each machine and run of day i must be less than or equal to the clock time when day i finishes, and in constraints *(7)* the start time of each item in each machine in run *1* of day i must be greater than or equal to the clock time when day $i-1$ finishes. Constraints *(8)* relate the due date of a product in raw with its completion time and lateness. The sets *(9)* to *(12)* are related with the aim of minimizing the deviations between the conclusion time of a product in raw and the conclusion time of all the items that belong to this product in raw. In these constraints the values of the U_{jmki} presented in the objective function are determined. Constraints *(13)* to *(17)* are the changeover constraints. Constraints *(13)* to *(15)* determine the items sequence and constraints *(13)*, *(16)* and *(17)* force one changeover per machine, run and day, even if it is a dummy changeover from an item to itself. In constraints *(18)* it is stated that at most one production per machine, run and day can occur. Constraints *(19)* establishes the relation between the quantity to produce of a given item in a given machine, run and day (X_{jmki}) and the Z_{jmki} variables, forcing the X_{jmki} to be equal to zero if the Z_{jmki} variables are equal to zero too. Finally, in constraints *(20)* the relation between the quantities to produce and the changeovers are established, forcing a changeover whenever it is decided to produce. Constraints *(21)* are the non-negativity constraints and constraints *(22)* force the Z_{jmki} , $W1_{jmki}$, $W2_{jmki}$ and Q_{jlmki} decision variables to take binary values.

This MIP model considers all the aspects of the real problem, but has the disadvantage of taking prohibitive times to solve, even for small instances. Nevertheless it allowed us to get a deep insight of the problem at hands.

NETWORK MODELS AND SCHEDULING HEURISTIC DEFINITION

In this section we present an alternative solution method to solve the lot sizing and scheduling problem of the cotton knitting subsection. In this case, the problem is solved in two phases, iteratively, using network flow models and a scheduling heuristic. We solve the problem in two phases with the aim of minimizing the deviation between the conclusion times of the several items belonging to the same final item, while taking into account that a large number of setups is not desirable. Later on, this issue will be analyzed in more detail.

As it is not easy to include sequence dependent setups using network based models, in this case we do not take into account the sequence dependent setups. Instead, we try to minimize the number of setups between time intervals. In fact, the major number of changes of items involves setup times that can be neglected. Only the changes that involve change of yarns can be of major impact.

Phase I

In this subsection we define the network model of phase I and the scheduling heuristic used to schedule the flow solution obtained when solving the network model.

Network flow model

In the first phase we consider only one item of each final item/size: the one with the highest production time. These items are associated with supply nodes, and the supply of each node is defined by the production time of the corresponding item.

The demand nodes are associated with time intervals and machines. The information used in the definition of the time intervals consists in: the release times of the machines, the due dates of the items considered in the supply nodes and the planning horizon. After having defined the time intervals, each one of them is decoupled per machine. The definition of the intervals is illustrated using the following example:

Example 1:

Consider a problem with five final items that must be scheduled, at most in five machines, in the next 36 hours. The data associated with this example is presented in Table 1 and in Table 2.

Table 1 - Example 1 data

Final item	Item	Gauge	Quantity	Unit time (minutes)	Due date (hours)	Compatible machines	Production time (hours)
A1	CA1	27	1000	1	20	1,2,3,4,5	16.67
A1	FA1	27	1000	1	20	1,2,3,4,5	16.67
A2	CA2	27	1500	1	22	1,4,5	25
A2	FA2	27	1500	1	22	1,4,5	25
B2	CB2	27	500	1	30	1,2,3,4	8.33
B2	FB2	27	500	1	30	1,2,3,4	8.33
B2	MB2	27	1000	1	30	1,2,3,4,5	16.66
C3	CC3	27	200	1	21	1,3,4	3.33
C3	FC3	27	200	1	21	1,3,4	3.33
C3	MC3	27	400	1	21	1,2,3,4,5	6.66
C4	CC4	27	300	1	24	1,2,3,4,5	5
C4	FC4	27	300	1	24	1,2,3,4,5	5
C4	MC4	27	600	1	24	1,2,3,4,5	10

Table 2 - Release times of machines

Machine	Release time (hours)
1	0
2	1
3	0
4	1
5	2

The items defining the supply nodes are: CA1, CA2, MB2, MC3 and MC4. In Figure 1 the time intervals are defined, based on the release times of the machines presented in Table 2, the due dates of the items considered in the supply nodes and the planning horizon. The total number of demand nodes for example 1 is 36 (8 associated with machine 1, 7 with machine 2, 8 with machine 3, 7 with machine 4 and 6 with machine 5). The demand associated with each demand node is determined by the duration of the time interval. For example, the demand associated with time interval I3-M1 is 18.

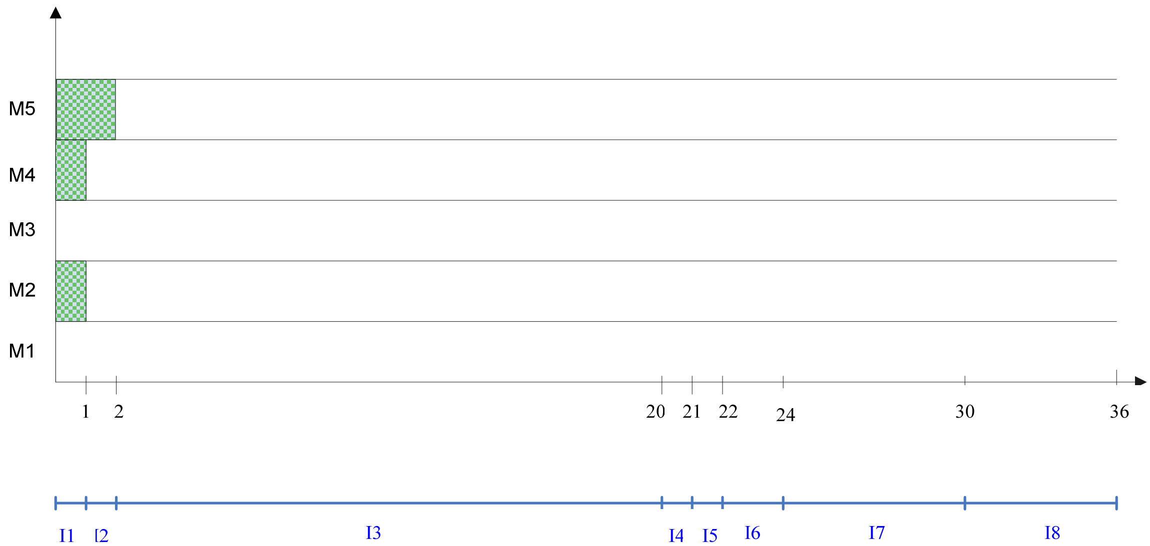

Figure 1 - Initial Gantt chart and time intervals

Each time interval must have a machine associated because of the compatibility matrix between the items and the machines. If the items could be done in all the machines available, this association would not be needed and the demand associated to each time interval would be defined as the product of the time interval duration by the number of machines available in that time interval. If we do not consider this association, when transforming the solution of flows, obtained when solving the above presented model in a network, in a scheduling solution using a Gantt chart, depending on the order in which we select the item, the solution we obtain can become impossible because the machine where a given item should be done was meanwhile occupied by an already scheduled item. We illustrate this using example 2.

Example 2:

Consider a problem with two machines available and that we want to schedule items CA1 and CA2 using these machines. CA2 can be processed in both machines, but CA1 only can be produced in machine 1. Both items have the same due date, which is 20. The flow solution indicates that in interval 3, that has duration of 18 hours in each machine, we must schedule 16 hours of CA1 and 14 hours of CA2. This is a valid solution since in interval 3 we have 18 hours available in machine 1 and another 18 hours available in machine 2. If we do not consider the association between the items and the machines and we begin by scheduling CA2 in machine 1, the scheduling solution become impossible, because we do not have enough time to schedule CA1 in the third interval of machine 1. The only way to prevent this is to consider the association between the intervals and the machines.

Between each supply node and each demand node can only exist an arc if the item of the supply node can be produced in the time interval associated with the demand node and if it can be produced in the machine associated to that time interval.

We consider three types of penalizations (or costs) associated to each arc: lateness, earliness and machines prioritization. We associate weights with the lateness and earliness penalizations, being the lateness weight of 1000 per unit time and the earliness penalization of one per unit time. The lateness penalization only appears in arcs associated with time intervals which end time is greater than the due date of the item being analyzed and is given by the product of the lateness weight by the difference between the end time of the interval and the due date of the item being analyzed. The earliness penalization is associated with arcs linked to time intervals which end time is smaller than the due date of the item being analyzed. Its value is given by the product of the earliness weight by the difference between the due date of the item and the end time of the time interval associated to the arc.

If we do not consider setups at all, their number can potentially be huge when changing between intervals. In order to reduce these setups we prioritize the machines that are capable of processing each item. First we order the machines according to their release times and we use this order in the attribution of the machines to the several arcs associated with the same time interval. Next we associate to each arc a penalization of α multiplied by the arc order. The machine prioritization penalization only appears in arcs which time interval end time is smaller than or equal to the due date of the item being analyzed. We use again example 1 to explain the determination of the penalization associated with the machines prioritization. Considering that the machine order is: M1 – M3 – M2 – M4 – M5 and that the value of α is 100, in the time interval I1-M1 the penalization will be 1×100 (first arc × α) and in time interval I1-M3 will be 2×100 (second arc × α). Depending on the value we attribute to α, it can be better to schedule the same item in earlier periods using always the same machine or it can be better to use another machine with the disadvantage of having a setup. The major drawback of prioritizing the machines is the possible imbalance in the machines utilization. In order to lowering this effect we solve the same problem using several alternative orders of machines and we select the best solution according to some specific objectives.

In the text below we present analytically the network flow model above presented. Consider the sets, parameters and decision variables below defined.

Sets:
N_1 – set of selected items (one for each final item/size)
N_2 – set of interval/machine pairs
A – set of arcs
M_j – set of items that can be produced on machine j.

Parameters:
p_i – total production time of item i
b_j – duration of time interval j
c_{ij} – lateness, earliness and machine priority associated with arc (i,j)
d_i – due date of item i
t_j – finish time of interval j
α – weight of machine priority
β – weight of lateness per unit time
μ - weight of earliness per unit time
o_{ij} – order number of arc (i,j)

Decision variables:
x_{ij} - flow of arc (i,j)

The network flow model is:

$$Min\ Z = \sum_{(i,j)\in A: i\in M_j} c_{ij} x_{ij}$$

subject to:

$$\sum_{j:(i,j)\in A \text{ and } i\in M_j} x_{ij} = p_i,\ \forall i \in N_1 \qquad (23)$$

$$\sum_{i:(i,j)\in A \text{ and } i\in M_j} x_{ij} \le b_j,\ \forall j \in N_2 \qquad (24)$$

$$x_{ij} \ge 0,\ \forall (i,j) \in A \qquad (25)$$

The value of c_{ij} is determined using equation *(26)*.

$$c_{ij} = \sum_{(i,j)\in A: d_i < t_j} \beta(t_j - d_i) + \sum_{(i,j)\in A: d_i > t_j} \mu(d_i - t_j) + \sum_{(i,j)\in A: d_i \ge t_j \text{ and } i\in M_j} \alpha o_{ij} \qquad (26)$$

This model is usually known as the transportation problem. In the objective function we minimize the sum of lateness, earliness and machine priority penalizations. Constraints *(23)* are the supply constraints, in which we force the sum of flows emanating from each item to be equal to its total production time. Constraints *(24)* are the demand constraints. In these constraints we force the flow entering in each interval/machine pair to not exceed the interval duration. In *(25)* we present the non-negativity constraints.

The interesting reader is referred to (Ahuda et al., 1993) for additional details about network flow models.

In Table 3 we present one of the optimal solutions to the network flow model of phase I of example 1.

Table 3 - Optimal solution of the network flow model of phase I of example 1

Component/size	Interval	Machine	Flow
CA1	I1	M1	1
CA1	I2	M1	1
CA1	I3	M1	6.67
CA1	I3	M3	6.33
CA2	I3	M4	18
CA2	I4	M4	1
CA2	I5	M4	1
MB2	I3	M1	4.67
MB2	I7	M1	6
MB2	I7	M3	6
MC3	I3	M1	6.66
MC4	I3	M3	2
MC4	I4	M1	1
MC4	I4	M3	1
MC4	I5	M1	1
MC4	I5	M3	1
MC4	I6	M1	2
MC4	I6	M3	2

Scheduling heuristic

After having solved the network flow model, we apply a scheduling heuristic in order to schedule in a Gantt chart the flow solution obtained. We apply the following three steps in the scheduling heuristic:

Step 1: schedule all the intervals that have associated only one item and that are fully occupied;

Step 2: for all the intervals that have associated only one item but that are not fully occupied:

Step 2.1: if possible schedule the item across an already scheduled part of it, if it is produced in the first position of the next interval or in the last position of the previous interval and go to step 3;

Step 2.2: schedule the item from the end time of the interval to its start time, if the finish time of the interval is smaller than or equal to the due date of the item. If the finish time of the interval is greater than the due date of the item, schedule it from the beginning of the interval;

Step 3: for all the intervals that have associated more than one item:

Step 3.1: select the item(s) that is(are) produced in the first position of the next interval or in the last position of the previous interval, and schedule it (them) across the already scheduled part(s) of it(them). If a given item is simultaneously produced in the first position of the next interval and in the last position of the previous interval, schedule it across the first position of the next interval. If all the items were meanwhile scheduled finish this procedure;

Step 3.2: for all the items not selected in step 3.1, schedule them according to the minimum slack first rule, starting in the beginning of the interval or in the first instant time free in the interval. If more than one item has the same slack, select arbitrarily one of them.

The minimum slack of a given item i in a given interval/machine j is given by: d_i-x_{ij}-t, where t is the actual instant time of interval j. Note that t can correspond to the beginning instant time of the interval j or to a certain instant time inside interval j. This second case can exist if in interval j there are one or more items already scheduled.

In Figure 2 we present the Gantt chart resulted from the application of the scheduling heuristic to the solution presented in Table 3.

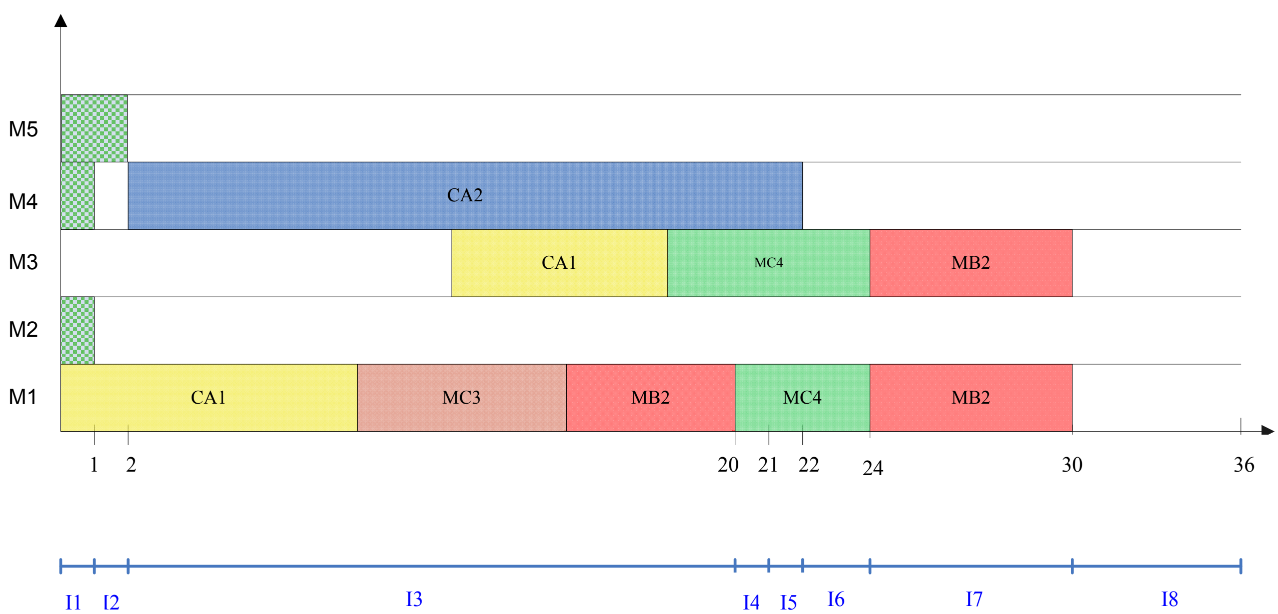

Figure 2 - Gantt chart for the solution of phase I of example 1

Phase II

In phase II we define a new network flow model that is based in the solution obtained in the first phase. In this model the supply nodes are associated with the items that were not considered during phase I and the supply associated with each node is given by the production time of the corresponding item. The demand nodes are also associated with time intervals and machines. The release times of the machines and the planning horizon are again used in the intervals definition. The due dates are used too, but they can be modified according to the solution obtained in phase I. The due date of a given item considered in phase II is given by the finish time of its brother in phase I. Consider for example final item A1 of example 1. The due date of this final item is 20 as can be seen in Table 1, but as CA1 finishes at 18, as can be seen in Figure 2, the due date of FA1 (that is the brother of CA1 considered in phase II) will be 18. Besides considering these information's to define the time intervals, in phase II the machines loads of phase I need to be considered too. In Figure 3 we present the time intervals of phase II for example 1. The demand associated with each time interval/machine is given by the duration of the time interval.

The arcs of the network of phase II are defined in the same manner as in phase I and the arc penalizations are calculated in the same way. The only difference is associated with the way we define the order in which the machines will be associated to the time intervals. In phase II the order has to be determined according to the machine release times and to the machines loads. This will force the order to be formed dynamically according to the machines available in each time interval. A possible order will be: I1-M3, I2-M3, I2-M2, I2-M4, I3-M3, I3-M2, I3-M5, I4-M2, I4-M5, I5-M2, I5-M5, I6-M2, I6-M5, I7-M2, I7-M5, I7-M4, I8-M2, I8-M5, I8-M4, I9-M2, I9-M5, I9-M4, I9-M1 and I9-M3.

We do not present in this subsection the network model analytically as it is similar to the one presented in phase I. The scheduling heuristic used in phase II is equal to the one developed to phase I. In Table 4 and in Figure 4 we present the flow solution and the Gantt chart of phase II, respectively.

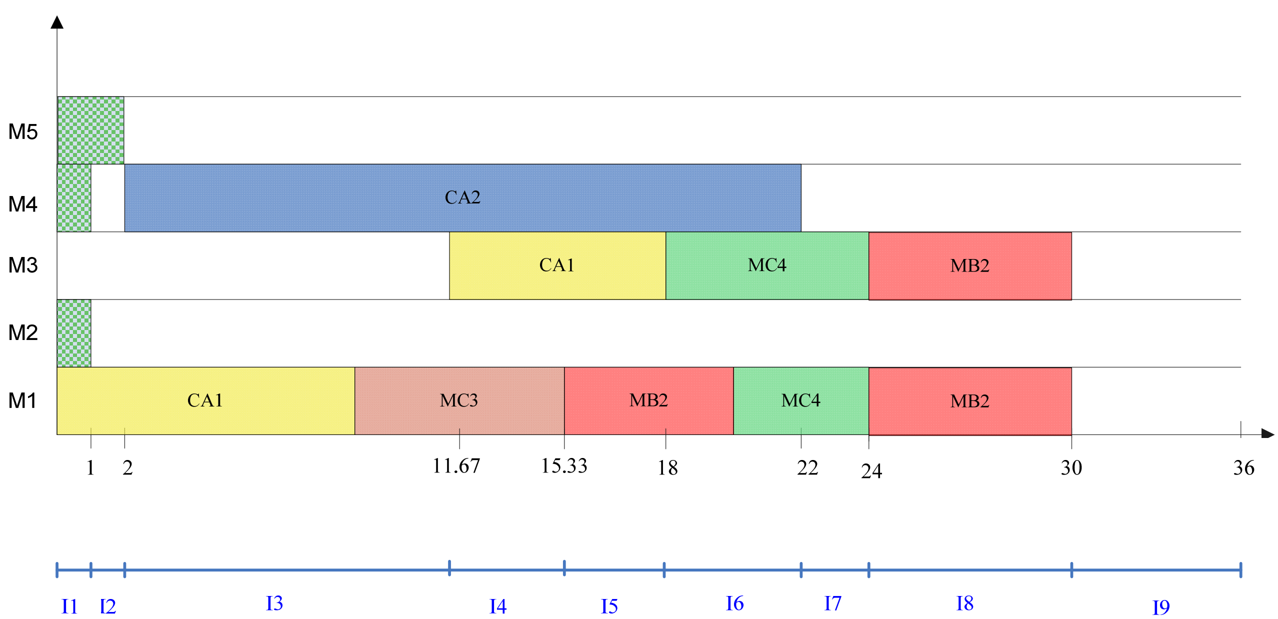

Figure 3 – Gantt chart of phase I and time intervals of phase II of example 1

Table 4 - Optimal solution of the network flow model of phase II of example 1

Component/size	Interval	Machine	Flow
FA1	I1	M3	1
FA1	I3	M3	4.01
FA1	I3	M2	3.66
FA1	I4	M2	3.66
FA1	I5	M2	2.67
FA2	I3	M5	9.67
FA2	I4	M5	3.66
FA2	I5	M5	2.67
FA2	I6	M5	4
CB2	I6	M2	0.33
CB2	I7	M4	2
CB2	I8	M4	6
FB2	I6	M2	2.33
FB2	I8	M2	6
CC3	I3	M3	3.33
FC3	I2	M3	1
FC3	I3	M3	2.33
CC4	I3	M2	4.66
CC4	I6	M2	0.34
FC4	I6	M2	1
FC4	I7	M2	2
FC4	I7	M5	2

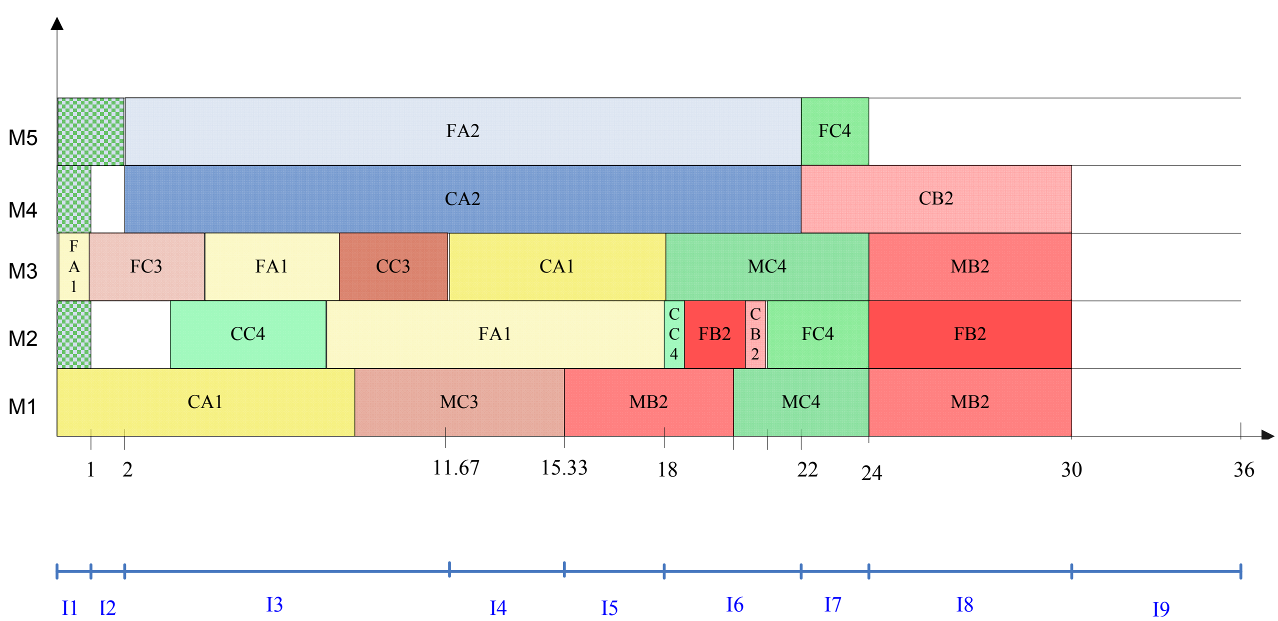

Figure 4 – Gantt chart for solution of phase II of example 1

NETWORK MODELS AND SCHEDULING HEURISTIC EVALUATION

In order to test our method we developed an instance generator in which we try to produce instances as much faithful as possible with the real data. In this text we present results for one of those instances with the aim of showing the correctness of our method as long as the main measures we used to evaluate the performance of the schedules.

The selected instance has 65 final items that are associated with the four types of orders and a planning horizon of one week. The 65 final items are decomposed in 501 items, considering the components and sizes, in which 91 of them belong to 21 gauge, 205 to 24 gauge and the others 153 to 27 gauge. There are five machines available in 21 gauge, 13 in 24 gauge and 11 in 27 gauge. We coded the transportation problems and the scheduling heuristic in visual C++ and solved the transportation problems with Cplex 11.0 (ILOG, 2007). The code produces three solution outputs: the first one has the weekly production plans for the three gauges represented in Gantt charts; the second one has the main performance measures as long as some of the main characteristics of the instance solved, and the third one has information related with the machines, *i. e.*, the number of machines used for each final item/size, the number of machines used for each item and the number of items produced in each machine.

For the selected instance, the transportation problem of phase I was created and solved in 0.136 seconds and the one of phase II in 0.761 seconds. The time needed by the scheduling heuristic was 0.001 and 0.003 seconds in phase I and II, respectively. In Table 5 we present the lateness, the total deviation between the several items that belong to the same final item, the gauge utilization and the number of setups associated with the production plans obtained for each gauge. In average there are 1.16, 2.55 and 1.58 setups associated with each item for gauge 21, 24 and 27 respectively. The mean number of machines used per final item/size is 3.66 and per item 1.56.

Table 5 – Some performance measures for the selected instance

Gauge	Lateness (hours)	Total deviation between items (hours)	Gauge utilization (%)	Number of setups
21	0	1149.16	27.9	106
24	0	8040.5	79.8	524
27	0	1407.44	48.7	242

CONCLUSION

In this paper we present two approaches developed with the aim of supporting the knitting production planning of a textile company. In both of them a weekly production schedule is developed for the main knitting subsection. The first approach is a MIP model that takes into account all the characteristics of the problem, but that takes huge amounts of time to solve exactly. We only tested it in small instances and even in that case an optimal solution is not found within one hour. As we are dealing with a huge real problem we decided to develop an alternative method that gives approximate solutions in few seconds. The major disadvantage of this method is that it doesn't consider the sequence dependent setup times. Another disadvantage is related with the possibility of having imbalances in the machines utilization due to the machines prioritization, especially if the load of the subsection is low. In order to somehow prevent this we test several orders of machines and select the best solution according to some specific measures above presented. Nevertheless we consider this method an effective tool, with possibilities of practical application mainly due to its solution times. Our major concern at present is to take into account the sequence dependent setups.

In the near future we intend too to use local search to try to improve the scheduling heuristic. The aim is to improve the quality of the solution either inside a machine either between a set of machines by providing items exchanges. Beside this, we plan to consider the production planning problem of the other two subsections of the knitting section and to test our model using real data.

ACKNOWLEDGMENTS

The authors acknowledge financial support provided by Fundação para a Ciência e a Tecnologia (FCT) through PhD research grant (SFRH/BD/38582/2007).

REFERENCES

Ahuda, R., Magnanti, T. and Orlin, J. (1993), *Network Flows: Theory, Algorithms and Applications,* Prentice Hall, New Jersey.

Allahverdi, A., Gupta, J. and Aldowaisan, T. (1999), "A review of scheduling research involving setup considerations," *Omega The International Journal of Management Science,* Vol.27**,** pp. 219-239.

Chen, K. and Ji, P. (2007), "A mixed integer programming model for advanced planning and scheduling (APS)," *European Journal of Operational Research,* Vol.181**,** pp. 515-522.

Dauzère-Pérès, S. and Lasserre, J. (1997), "Lot streaming in job-shop scheduling," *Operations Research,* Vol.45**,** No.4, pp. 584-595.

Doganis, P. and Sarimveis, H. (2007), "Optimal scheduling in a yogurt production line based on mixed integer linear programming," *Journal of Food Engineering,* Vol.80**,** No.2, pp. 445-453.

Drexl, A. and Kimms, A. (1997), "Lot sizing and scheduling - survey and extensions," *European Journal of Operational Research,* Vol.99**,** No.2, pp. 221-235.

ILOG (2007), CPLEX 11.0 User's Manual.

ILOG (2008), www.ilog.com (last access on March, 31, 2008).

Serafini, P. (1996), "Scheduling jobs on several machines with the job splitting property," *Operations Research,* Vol.44**,** No.4, pp. 617-628.

Sheen, G. and Liao, L. (2007), "Scheduling machine-dependent jobs to minimize lateness on machines with identical speed under availability constraints," *Computers and Operations Research,* Vol.34**,** pp. 2266-2278.

Tahar, D., Yalaoui, F., Chu, C. and Amodeo, L. (2006), "A linear programming approach for identical parallel machine scheduling with job spliting and sequence-dependent setup times," *International Journal of Production Economics,* Vol.99**,** pp. 63-73.

Yalaoui, F. and Chu, C. (2003), "An efficient heuristic approach for parallel machine schedulinng with job splitting and sequence-dependent setup times," *IIE Transactions,* Vol.35**,** pp. 183-190.

Zhu, X. and Wilhelm, W. (2006), "Scheduling and lot sizing with sequence-dependent setup: a literature review," *IIE Transactions,* Vol.38**,** No.11, pp. 987-1007.