International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: http://www.tandfonline.com/loi/tprs20

Scheduling continuous aluminium casting lines

Ekrem Duman , Mehmet Bayram Yildirim & Ali Fuat Alkaya

International Journal of Production Research,
Vol. 46, No. 20, 15 October 2008, 5701–5718

Scheduling continuous aluminium casting lines

EKREM DUMAN†, MEHMET BAYRAM YILDIRIM*‡ and
ALI FUAT ALKAYA§

†Industrial Engineering Department, Dogus University, Acibadem, Istanbul, Turkey
‡Department of Industrial and Manufacturing Engineering,
Wichita State University, 1845 Fairmount, Wichita, KS 67260-0035, USA
§Computer Science and Engineering Department, Marmara University,
Goztepe, Istanbul, Turkey

(*Revision received March 2007*)

This study considers the problem of scheduling casting lines of an aluminium casting and processing plant. In aluminium processing plants, continuous casting lines are the bottleneck resources, i.e. factory throughput is limited by the amount of aluminium that can be cast. The throughput of a casting line might be increased by minimizing total setup time between jobs. The objective is to minimize setup time on production lines for a given time period while balancing workload between production lines to accommodate potential new orders. A mathematical formulation for scheduling jobs to minimize the total setup time while achieving workload balance between the production lines is presented. Since the casting scheduling problem is an NP-hard problem, even with only one casting line, a four-step algorithm to find good solutions in a reasonable amount of time is proposed. In this process, a set of asymmetric travelling salesman problems is followed by a pairwise exchange heuristic. The proposed procedure is applied to a case study using real casting data.

Keywords: Aluminium cast scheduling; Asymmetric travelling salesman problem; Heuristics

1. Introduction

The research reported in this paper was inspired by a production scheduling problem in a large aluminium casting and processing factory in Istanbul, Turkey. The company casts aluminium coils of different alloys in various widths and thicknesses. According to customer specifications, these coils are then rolled to the desired dimensions and properties through cold milling, annealing, surface processing and cutting operations. Rolled aluminium coils are used widely in manufacturing, e.g. the production of aircrafts, buses, cans, packaging, wrapping materials, etc.

The first operation in the manufacturing process of rolled aluminium is the casting operation, whereby ingots or slabs of pure aluminium are melted in a furnace. Then, additive elements such as magnesium, vanadium, etc., which

*Corresponding author. Email: bayram.yildirim@wichita.edu

International Journal of Production Research
ISSN 0020–7543 print/ISSN 1366–588X online © 2008 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540701355212

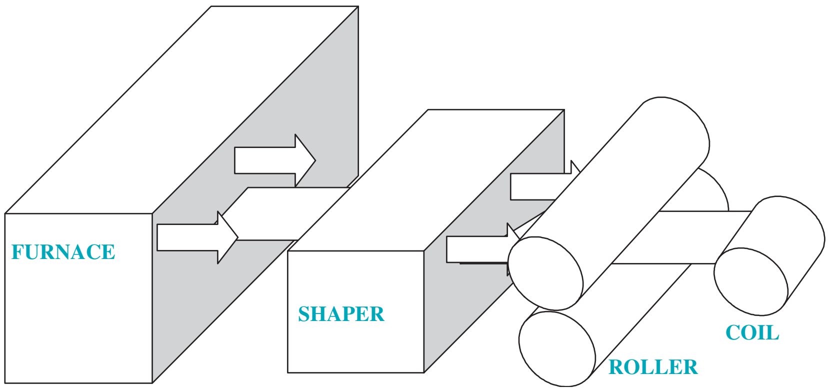

Figure 1. Schematic diagram of an aluminium casting line.

determine the alloy of the aluminium, are added to the furnace in specific amounts. The melted metal in the furnace flows through a shaper and a rolling mill (see figure 1). The entire process takes place on a casting line (CL). In the process shown in figure 1, continuous production is required to avoid reheating of a furnace after it cools down—a long and costly procedure. After the casting operation, to obtain the desired width and thickness, the metal is fed in coil form through a series of cold rolling mills, which successively reduce the metal thickness and recoil it after each rolling pass, thus preparing it for the next pass, until the required thickness is obtained. Annealing may be required between passes, depending on the final temper required. Surface processing and cutting operations then follow.

In the aluminium casting plant in this case study, casting is a determinant of factory throughput. Production is limited by the amount that can be cast. According to our observations, although the subsequent cold milling resources sometimes resulted in a bottleneck (in the case where the majority of orders consist of thin products), usually the CLs were the actual bottleneck resources. Therefore, maximally utilizing CLs is a goal that should be considered.

On a CL, several operations can be viewed as setups, most of which are sequence-dependent since the order of jobs on a production line affects the length of setup. These setups occur when different alloys are cast and/or when the width of the next job to be processed is different from the width of the current job. In our case, the scheduling objective is to maximize the utilization of CLs and, at the same time, balance their loads in anticipation of future orders.

In some ways, the steel manufacturing industry is similar to the aluminium industry. Tang *et al.* (2001) review various manufacturing technologies used in the steel industry and provide information on planning and scheduling methods. Tang *et al.* (2000) studied the scheduling of hot rolling (similar to cold milling in an aluminium factory) in a steelmaking factory. They formulate the problem as a multiple travelling salesman problem (TSP) and find a solution using genetic algorithms. Tang *et al.* (2002) construct an integer programming formulation and develop a solution methodology using dynamic programming and Lagrangian relaxation for steelmaking scheduling process. Harjunkoski and Grossmann (2001) discuss the scheduling of pre-casting and casting operations in a steel plant.

Instead of trying to solve the aggregate problem, they dissect the problem and solve it in four phases. Assaf *et al*. (1997) propose a complete enumeration method to solve steel production scheduling problems in the presence of practical constraints. Lopez *et al*. (1998) formulate the hot strip mill production scheduling as a generalization of the prize-collecting TSP and solve it using the Tabu search metaheuristic procedure. Gravel *et al*. (2002) report on the scheduling of a single aluminium casting line, while considering several objectives (tardiness, unused capacity, draining and transport) at the same time using an ant colony optimization technique. They observe that in the simplest case when technological and practical scheduling constraints are not considered, their problem reduces to a TSP because of the setups between orders on the casting line. In Gravel *et al*. (2002), the final product is customers' ingots. In our case, the final product is aluminium coil rolls. Furthermore, we have several parallel CLs. As a result, our problem resembles a single-objective vehicle-routing problem with load-balancing constraints.

Our problem is also somewhat similar to parallel-machine scheduling problems with setups (in our case, only some of the machines are identical). In their survey paper, Allahverdi *et al*. (1999) present the state of the art for machine scheduling problems with setups. In a parallel-machine environment with identical machines, minimizing the maximum completion time is NP-complete (Kurz and Askin, 2001). Furthermore, Chen and Powell (2003) observe that solving non-identical parallel machines for any objective is more complex than that for identical parallel machines and a sequence-dependent setup time will further complicate the problem. Due to the nature of complexity involved in terms of formulation and computational time, several methods are proposed to solve parallel-machine problems with sequence-dependent setups, e.g. dynamic programming (Gascon and Leachman 1998) and branch and bound algorithms (Dietrich and Escudero 1989). Kurz and Askin (2001), Fowler *et al*. (2003) and Yildirim *et al*. (2007) use genetic algorithms to solve a parallel-machine problem with sequence-dependent setup times. In this paper, the load-balancing constraint is strict, i.e. it cannot be violated and there are sequence-dependent setups.

In this study, our goal is to formulate a mathematical model to solve the load-balancing problem in a parallel-machine environment with sequence-dependent setups and minimum sum of total completion time objective. In our setting, inspired from a real case study of a large aluminium casting and processing plant in Turkey, the machines (i.e. the CLs) are non-identical. Some of the CLs can process all of the incoming jobs while some others are less capable. The contributions of this paper can be summarized as follows. (1) We first present a mathematical model to formulate the unrelated parallel CL scheduling problem with load-balancing constraints and sequence-dependent setups given that the CLs are non-identical; the objective is to minimize the sum of total completion time on all CLs. (2) A four-step heuristic approach to solve this CL scheduling problem is proposed. (3) The results of the computational experimentation indicate that the heuristic approach described provides a significant financial benefit to the aluminium sheet manufacturer.

The following section describes the CL scheduling problem in more detail and then provides a mathematical formulation. In section 3, a four-phase solution methodology is developed to determine near-optimal solutions to the mathematical program. In section 4, the proposed solution methodology is illustrated with manufacturing data obtained from an aluminium processing plant and the results are analysed in detail. Finally, section 5 provides a short summary of the study and proposes possible future research directions.

2. Problem description and formulation

2.1 *Problem description*

Customers order products (i.e. jobs on CLs) with various properties. These orders are scheduled on parallel CLs based on their required size, width, thickness and alloy. Important factors that must be considered when designing a casting schedule can be categorized as line specifications and job specifications.

2.1.1 Line specifications.

(a) *Width*. The width of aluminium coils that each CL can cast is limited. It is assumed that some CLs can cast up to three different widths: narrow, medium or wide. In other words, CLs are not identical.
(b) *Speed*. It is assumed that the difference in the processing speeds of different CLs is negligible.

2.1.2 Job specifications.

(a) *Width*. If two consecutively scheduled jobs have different widths, then there will be a setup time after the job scheduled first. For two consecutive jobs, the setup time when the width of the second job is narrower than the first one is shorter than the setup time when the first job is narrower than the second.
(b) *Alloy*. If an alloy change occurs between two consecutive jobs, then a significant setup time may be required. Usually, no significant setup is required to process a more composite alloy after a purer alloy (i.e. some additive elements are added to the furnace and then production continues). However, in the reverse case, i.e. if a purer alloy is to be cast after a composite alloy, then the furnace must be cleaned thoroughly (hot cleaning) and this process usually takes significantly longer than the previous case.
(c) *Last job of previous schedule*. An important consideration in minimizing setup time involves the properties of the last jobs cast on lines in the previous planning period. The current planning period's schedule should take this into account as the initial condition.
(d) *Changes in the thickness of orders*. Other than the above specifications, jobs may have different casting thicknesses. However, any change in thickness can be handled in a very short amount of time. As a result, changes in thickness have a minimal impact on setup in a casting-scheduling problem.

2.2 *Mathematical formulation*

Let K be the set of CLs where $|K|$ is its cardinality. Let $J = \{1, \ldots, n, n+1, \ldots, n+|K|\}$ be the set of jobs where the first n jobs are the jobs to be produced in the current planning period and the last $|K|$ jobs are the last jobs produced on the corresponding CLs in the previous period. Let K_j be the set of CLs on which job j can be processed and let J_k be the set of jobs that can be processed on CL k. Let p_{ik} be the processing time of job i on CL k, and s_{ijk} be the setup time incurred if job i immediately precedes job j on casting line k. The parameter α, which takes values between zero and one, constrains the level of maximum imbalance between CLs. Then, the casting-scheduling problem with sequence-dependent setups and load-balancing constraints (CSP-SDS-LBC) can be formulated as

$$\min C_{\text{total}}$$

$$C_{\text{total}} = \sum_{k \in K} C_k \tag{1}$$

$$C_k = \sum_{i=1}^{n} y_{ik} p_{ik} + \sum_{i \in J_k} \sum_{j \in jk} x_{ijk} s_{ijk} \quad k \in K \tag{2}$$

$$C_k \le \frac{1}{|K|} C_{\text{total}}(1+\alpha) \quad k \in K \tag{3}$$

$$C_k \ge \frac{1}{|K|} C_{\text{total}}(1-\alpha) \quad k \in K \tag{4}$$

$$\sum_{k \in K_i} y_{ik} = 1 \quad i \in J \tag{5}$$

$$y_{ik} = 1 \quad i = n + k \text{ and } k = 1..|k| \tag{6}$$

$$x_{ijk} \le y_{ik} \quad i \in J, k \in K_i, j \in J_k \tag{7}$$

$$\sum_{i \in J_k} x_{ijk} = 1 \quad k \in K, j \in J_k \tag{8}$$

$$\sum_{i \in J_k} x_{ijk} = 1 \quad k \in K, i \in J_k \tag{9}$$

$$\sum_{i \in S_k j \in S_k} \sum_{j \neq i} x_{ijk} \le |S_k| - 1 \quad S_k \subseteq J_k \tag{10}$$

where C_{total} is the total processing and setup time of all jobs, C_k is the total processing and setup time on line k,

$$y_{ik} = \begin{cases} 1 & \text{if job } i \text{ is assigned to casting line } k \\ 0 & \text{otherwise} \end{cases}$$

and

$$x_{ijk} = \begin{cases} 1 & \text{if job } i \text{ the immediate predecessor of job } j \text{ on casting line } k \\ 0 & \text{otherwise} \end{cases}$$

In the CSP-SDS-LBC formulation, the objective is to minimize the total processing and setup time needed to complete all jobs on all CLs.

Constraints (1) and (2) determine the total processing time of jobs on all lines and on a particular CL, respectively. Constraints (3) and (4) are workload-balancing constraints. Since a perfect balance may not (and most probably will not) be obtainable, a certain percentage (α) above or below the ideal average workload is acceptable. Typical values of α could be in the range of 0.05–0.30. Constraint (5) guarantees that each job is assigned to a CL. Similarly, using constraint (6), $|K|$ additional jobs corresponding to the last jobs of the previous month are assigned to the corresponding CLs. Constraint (7) ensures that a job cannot precede another job unless both jobs are assigned to the same line. Constraints (8) and (9) ensure that a job succeeds/precedes only one job. The subtour elimination constraints are given by constraint (10). In equation (10), S_k is a subset of jobs that can be processed on CL k.

3. Solution methodology

The CSP-SDS-LBC formulated in the previous section provides a means of minimizing the total setup time and attaining a balanced workload among the CLs. In fact, the formulation given resembles the well-known vehicle-routing problem (VRP), where the vehicle capacities are somewhat fuzzy. If the constraint sets (3) and (4) given for workload balancing are modified to, for example, 'the workload assigned to any line cannot exceed 30 days', then our problem can be modelled exactly as a VRP with non-identical vehicle capacities. Note that the VRP is one of the most difficult problems of combinatorial optimization and is NP-hard (Ghiani *et al*. 2003). Laporte *et al*. (2000) provides a survey of classical and modern heuristics for the VRP. When there is a single CL, the problem reduces to a TSP that can be solved using several heuristic methods including the ant colony optimization method (Dorigo *et al*. 1996).

For a problem involving 20 jobs in a plant with five CLs, running CPLEX 9.1 for three days (72 hours) on a 2.8 GHz Pentium IV computer with one gigabyte of memory did not provide proof of optimality. A typical problem in an aluminium processing plant is usually much larger. As a result, a four-step framework that would provide good solutions in a reasonable amount of time is proposed.

When the secondary load-balancing objective is disregarded (i.e. constraints (3) and (4) are ignored), maximization of the resource utilization objective is equivalent to minimization of the total setup time on each CL (which can be modelled as a variant of the VRP). Additionally, if the job-line assignment decision is known (i.e. the values of the y_{ik}-decision variables are fixed/known and the constraint sets (3), (4) and (5) are satisfied) then the objective will be minimization of the sum of setup times on $|K|$ CLs. Furthermore, the problem will decompose into $|K|$ subproblems, since the setup times on one CL do not depend on the setup times on other CLs. The resulting formulation of a subproblem with constraint sets (8), (9) and (10) is the asymmetric travelling salesman problem (ATSP) because the setup time matrix (which represents the total setup time needed between jobs as a result of width changes and alloy type changes) is asymmetric.

At this point, by naming the originally formulated problem (CSP-SDS-LBC) as the *master problem* and the resulting ATSPs after decomposition as the *subproblems*, it is possible and reasonable to define the following four-phase solution procedure to solve the master problem.

1. *Line assignment*. Assign jobs to casting lines in a balanced manner.
2. *Solution of subproblems*. Solve the resulting ATSPs.
3. *Improvement*. Try to improve the current solution of the master problem by considering changes in the line assignment made in phase 1.
4. *Re-solution of subproblems*. Solve the ATSPs again to detect any possible improvements, since the original solutions may have deteriorated in the third phase.

3.1 *Phase 1: Line assignment*

Order the jobs in non-decreasing order of their widths. The CL that can process narrower jobs has smaller index. The total processing time of all jobs that are planned to be scheduled during a planning period is calculated, and this amount is divided by the number of CLs ($|K|$) to find the desired workload per CL (DWPCL) without considering the effect of setup times. Starting from the top of the list and the smaller index CL, jobs are assigned to the CL if that assignment brings a decrease in the absolute value of the difference between the DWPCL and the total assigned workload. If not, the next job on the list is tried for the assignment. The same procedure is used for all lines until all jobs are assigned.

3.2 *Phase 2: Solution of subproblems*

Solve the resulting $|K|$ ATSPs optimally for each line. In order to solve each ATSP, the cost matrix (width-change setups plus alloy-change setups) should be calculated for the jobs assigned to each CL including the last job of the previous planning period as a city of the ATSP. Take the setup time between any regular job and the last job of the previous planning period as zero, and then arrange the directed ATSP route so that the last job of the previous planning period becomes the home city.

3.3 *Phase 3: Improvement*

Note that the quality of the solution procedure described thus far is primarily dependent on how well the line assignment of phase 1 is made. Changing the line assignment and then solving the ATSPs repeatedly is not easy and it is intractable in terms of the run time required. Therefore, a pairwise exchange on the solution obtained at the end of phase 2 is more reasonable and is described as follows.

Randomly determine any two jobs on all $|K|$ CLs (exclude the first jobs that correspond to the last jobs of the previous planning period) and check if they are exchangeable (i.e. width capacities of CLs are not violated and workloads of the corresponding lines are still within the acceptable range). If they are exchangeable, and if the total processing time is the same or would be improved, then perform the exchange of these two jobs. Continue this procedure until a predetermined number of exchange trials is made.

Note that, in this procedure, if two randomly determined jobs happen to be on the same CL, then selection is still considered. Such a selection will hardly bring an improvement in the earlier phases since the sequence of each CL is determined optimally, but it may be the base for improvement in later iterations. First, it may provide a means of moving away from the current local optimum and second, in later iterations, it may improve the sequence on a CL since the original optimum ATSP solution may have deteriorated.

3.4 *Phase 4: Re-solution of subproblems*

Note that the improvement phase may have caused deterioration in the ATSPs that were solved in the second phase. Thus, it is advisable to solve them again to detect any potential improvement.

4. Case study of a large aluminium processing plant

The four-phase solution procedure described above is illustrated using a case study of a large aluminium processing plant. We begin with a description of the scheduling practice in the plant and continue with the experimental runs and economic interpretation of the improvement obtained.

The aluminium processing plant analysed has five CLs. Each CL can process orders up to a certain width. These limits are 1400 mm for CL1, 1700 mm for CL2 and 2200 mm for CLs 3, 4 and 5. The processing speed of the CLs varies with respect to the alloy. Although there is a slight difference in processing time between the speed of CL1 and the others, the casting speed of any particular alloy is considered to be equal on different lines. In other words, p_{ik}, the processing time of order i on line k, is assumed to be p_i.

On any CL, when orders are scheduled, if the width of the next job is smaller, in order to ensure the quality of the product, the shaper (which determines the desired width) must be changed. This operation (which can be considered as a setup) takes 2.5 hours. On the other hand, if the width of the next job is larger, then both the shaper and the roller must be changed, a procedure that takes around 6 hours. To decrease setup times due to changing job widths, the plant is implementing some standard casting widths (mm): 1220, 1280, 1320, 1400, 1450, 1550, 1600, 1650, 1700, 1830, 2080, 2120, 2140 and 2200. The standard thicknesses are 3 mm and 5 mm for alloy 1050, and 4 mm and 6 mm for all other alloys.

When different types of alloys are scheduled consecutively, the CLs may require a setup time that includes a hot cleaning operation; this takes 10 hours on CL1 and 15 hours on CLs 2, 3, 4 and 5 (see table 1 for detailed setup times for the alloys produced at this plant).

4.1 *Structure of the demand data*

In the aluminium processing plant under study, the schedule of the CLs is prepared on a monthly basis. A typical order size is 3 to 10 tons. On average 600–1800 orders are received each month, a number so high that it is almost impossible to solve a problem of this size in the current practical setting. Thus, orders are combined in such a way that they have sufficient common properties to be cast at the same time (e.g. orders having similar alloy types and widths are grouped as much as possible).

Table 1. Alloy change setup time (in hours) matrix for CLs 2, 3, 4, and 5 (for CL1, replace 15 hours with 10 hours).

	1050	1100	1200	1235	8011	8079	8006	8111	3003	3005	3105	5005	5049	5052	5182	5754
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1235	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8011	15	15	0	0	0	0	0	0	15	15	15	15	0	15	15	15
8079	15	15	15	15	0	0	0	0	15	0	0	15	15	15	15	15
8006	15	15	15	15	15	15	0	15	15	15	15	15	15	15	15	15
8111	15	15	15	15	0	15	15	0	15	15	15	15	15	15	15	0
3003	15	15	15	15	15	15	15	15	0	0	15	15	15	15	15	15
3005	15	15	15	15	15	15	15	15	15	0	15	15	0	15	15	15
3105	15	15	15	15	15	15	15	15	15	0	0	0	0	0	0	0
5005	15	15	15	15	15	15	15	15	15	0	15	0	0	0	0	0
5049	15	15	15	15	15	15	15	15	15	15	15	15	0	15	15	15
5052	15	15	15	15	15	15	15	15	15	15	15	15	15	0	0	0
5182	15	15	15	15	15	15	15	15	15	15	15	15	15	15	0	15
5754	15	15	15	15	15	15	15	15	15	15	15	15	15	0	0	0

As a result of this order consolidation, the number of jobs to be scheduled on a CL decreases significantly and casting jobs weigh 50–400 tons when the number of jobs on CLs is consolidated to 40–50 orders per month (i.e. 8–10 jobs per CL).

A week before the beginning of each month, production planners determine the schedule for that month. The desired schedule should make all CLs fully loaded and perfectly balanced (i.e. loading all CLs until midnight of the last day of the month). As expected, the casting jobs at hand do not allow for a schedule with such a perfect fit. In such cases, after preparing the schedule for each CL, planners typically increase the amount of some popular alloy types (such as 1050 and 3003) so that the lines are loaded exactly until the end of the month. In doing so, they assume that the extra amount of cast alloy can be allocated to customer orders received during that month or the following month at the latest. In this way, they obtain schedules that perfectly fit into the month. (We have to note that, during the time of our project, the plant was utilized at full capacity and some customer orders were occasionally refused.)

A sample of consolidated data (for February 2004) is given in table 2. These data provide information on the alloy, width, quantity and processing time required to finish a cast job. For this demand data, figure 2 shows a Gantt chart that represents the implemented schedule. In the figure, the setup times are lightly shaded. The dark shaded area on CL4 represents periodic maintenance. In table 2, the first 39 jobs are those cast in February, whereas the last five jobs are those cast as the last jobs in January. The production planning department at this plant designed a schedule that takes 3310.6 hours for overall processing and setup times.

Normally, we would assume zero setup time and zero processing time for the last five jobs completed in the previous month. However, to handle the time spent on maintenance, the processing time of the last jobs of the previous month is assumed to be equal to the total maintenance time during the month on the respective CL. The maintenance operations do not affect the setup requirements, i.e. a setup time determined by the jobs prior and after the maintenance is incurred.

4.2 *Experimentation on real data*

The implementation of phases 1, 3 and 4 of our solution procedure is quite straightforward. In phase 3 (the improvement phase) we need to decide on the number of pairwise exchange trials (i.e. the number of iterations). At first, n^2 pairwise exchanges were tried (n being the number of jobs) and then n^3 and even n^4 exchanges were tested (although n^4 trials take more time, it is still applicable since the schedules will be prepared once in a month).

The value of α, which indicates the level of tolerance in the workload imbalance of a CL (constraint sets (3) and (4) in the problem formulation), is an important design parameter. Three different values of α were tested (0.05, 0.10 and 0.30), where $\alpha = 0.05$ is an indication of an almost perfectly balanced schedule and $\alpha = 0.30$ is an indication of a loosely balanced but still reasonable schedule. However, regarding discussions on how the company manipulates job amounts, the choice of $\alpha = 0.30$ was considered suitable to represent the amount of benefit that the company might gain from the proposed solution. The total improvement in overall processing and setup times obtained for $\alpha = 0.30$ is given in table 3. Here, the first row represents the company's schedule (CS) for January–October 2004. When the proposed framework

Table 2. Casting data for February 2004.

Job No	Alloy	Width (mm)	Quantity (tons)	Proc. Time (hrs)
1	3105	1320	50	40.5
2	1050	1320	200	139.4
3	1100	1320	50	34.8
4	3105	1320	200	162.1
5	8006	1220	85	74.5
6	1050	1320	220	153.3
7	3003	1320	40	32.4
8	3105	1320	110	86.7
9	1050	1700	150	66.2
10	5005	1700	273	167
11	8011	1660	150	85.8
12	8111	1660	130	94
13	8011	1460	50	32.5
14	1050	1700	220	97.1
15	3003	2120	150	67.9
16	8006	2120	240	107.5
17	1050	2120	150	53.1
18	1050	2120	150	53.1
19	8111	2120	160	90.6
20	3003	2120	150	67.9
21	3003	2120	50	22.6
22	3003	2120	150	67.9
23	3003	2120	230	104.2
24	8111	1660	150	108.4
25	1050	2200	25	8.5
26	5005	2200	200	94.5
27	3005	2200	260	141.8
28	5049	2120	100	68.9
29	1050	2200	200	78.4
30	5005	2200	100	47.3
31	1050	2120	70	24.1
32	3003	2200	40	17.5
33	1050	2200	400	156.8
34	1200	2200	50	18
35	8111	2120	140	79.2
36	8111	1660	70	50.6
37	1050	2200	300	102.3
38	1100	2200	100	35.9
39	3105	2200	330	156
40	3105	1320	0	0
41	3105	1320	0	0
42	3003	2120	0	0
43	8111	1660	0	46
44	3003	2200	0	0

is applied to the CS, after phase 1 (P1), an increase in the overall time occurs (recall that phase one includes an assignment using the DWCPL rule). In phase 2 (P2), the setup time on each line is minimized using the GAMS/CPLEX solver to work out the resulting ATSPs. These results are further improved by performing random exchanges in phase 3 (P3). For each problem, P3 is performed ten times, each time

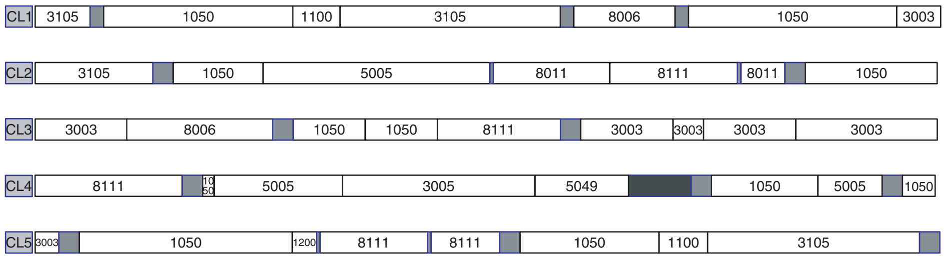

Figure 2. Gantt chart for the company schedule monthly data (table 2).

with a different random seed; the average of these runs is shown in the tables. Finally, in phase 4 (P4), the best available solution P3 is further improved by resolving the ATSPs optimally using GAMS. The overall procedure is relatively fast. On average, for n^4 pairwise exchange trials in phase 3, it takes 15 minutes per run on a Pentium IV 2.8 GHz Windows XP machine with 512 MB of RAM and 80 GB hard disk. As can be expected, the biggest improvement occurs in P2. Similar observations can be made on improvements in total setup time, which are presented in table 4: the average time spent on setups decreases from 232.45 hours to 122.78 hours. In other words, almost half of the time spent on setups is gained as potential production time. Table 4 also shows the number of pairwise exchange trials (# of iterations), the number of exchanges made (# of exchanges) and their ratio (# of exc./# of iter.).

The effectiveness of the proposed approach was tested using a genetic algorithm (GA) developed for a load-balancing problem in an unrelated parallel-machine scheduling environment in which the last job of the previous month and maintenance constraints were not considered (Yildirim *et al*. 2007). A GA with a population size of 20, number of iterations of 2000, and mutation and crossover rates of 0.05 and 0.95 respectively was run ten times for 2000 iterations per run to obtain the average total completion time (see table 3). In four out of ten cases (months), the two-exchange procedure provided better results. On average, the GA provided a solution that was 1.45% better than the two-exchange procedure solution. Thus, the GA solution does not dominate the proposed procedure. Note that the unrelated parallel-machine scheduling problem is a relaxation of the aluminium casting scheduling problem. In other words, if the GA was able to provide an optimal solution, then this solution would have been a lower bound to the aluminium casting problem solution.

A summary of the results for overall processing and setup times for different values of α and numbers of iterations in phase 3 is given in table 5 (table 6 shows the results of total setup time). As can be seen in table 5, the improvement, compared to the company schedule, is 2.59% for $\alpha = 0.05$, 3.04% for $\alpha = 0.10$ and 3.40% for $\alpha = 0.30$. As expected, the improvement increases further as more imbalance is tolerated. These improvements are obtained in the total processing time and they are quite meaningful. For example, if we look at improvements in the total setup time, we see that these figures range from 33.09–46.24%. Regarding unavoidable setup times due to width changes (at the least), we can conclude that our solutions are very close to the optima.

Another observation that could be made from table 5 and/or table 6 is that although increasing the number of iterations from n^2 to n^3 seems to be quite beneficial, there is very little gain in further increasing the number of iterations to n^4. However, since run time is not a critical factor in our case, n^4 iterations can be implemented. Note also that, the gain from increasing the number of iterations is less for smaller alpha values. This is natural since when the tolerance to imbalance is smaller, some pairwise exchanges could not be made that otherwise could have improved the objective value.

If we look at the improvements gained at each phase, we observe that phase 1 makes the company solution worse, which is natural. Phase 2 yields the best improvement, followed by phase 3. On the other hand, the improvement created by phase 4 is very small. This indicates that pairwise exchanges perform fairly well, with little room for optimization after their completion.

Finally, out of curiosity, we speculated what would happen if we omitted phase 2, in which case pairwise exchanges would first apply (phase 3) and then ATSPs would be solved optimally using GAMS (phase 4). The results of this experiment are shown in table 7: the improvement gained in step 3 is reduced (from 3.33% to 2.97%). This is normal because when we started with a worse solution (phase 1 result), the pairwise exchange procedure failed to obtain its previous best result, even though more exchanges were implemented successfully (2.07% versus 2.03%). However, fortunately, the end result obtained after phase 4 was better than the previous case (3.46% versus 3.40%). This phenomenon can be explained as follows.

In the standard design, optimal solutions of ATSPs are based on initial line assignments. But in the latter case, line assignment is improved initially by pairwise exchanges. Then, when the ATSPs are solved optimally, it is possible to arrive at better results in a shorter time. Thus, in this case study, on average it is observed that the solution procedure using phases 1, 3 and 4 and skipping phase 2 could be as good as the original proposed procedure.

The economic benefit to the aluminium factory on using the proposed solution procedure can be calculated as follows. In the base scenario ($\alpha = 0.30$), the improvement in total processing time is 3.46%. This is equivalent to an additional 125 hours of casting capacity in a 30-day month. The average amount that can be cast in 125 hours is 232 tons. The current price of processed aluminium for different alloys and products varies between \$2500 and \$4000 USD per ton. Assuming an average price of \$3000 USD per ton of processed aluminium coils, the company could make an additional \$696,000 USD revenue per month, or \$8,352,000 USD more revenue per year. Using the proposed methodology, a 5% profit margin implies \$417,600 USD additional profit per year. Given that the company believes that their own schedule is currently close to optimal, the additional potential profit using the proposed approach is quite significant.

5. Concluding remarks

Lean manufacturing strategies aim for manufacturing without waste. Waste might take several forms, e.g. material, time, inventory. The goal of this work was to minimize the total sequence-dependent setup time involved in the scheduling of CLs in a major aluminium casting and processing factory while achieving balance

Table 3. Improvements obtained in total (processing + setup) time and setup time (n^4 iterations, $\alpha = 0.30$), in hours.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Average
CS	3688.2	3310.6	3712.7	3590.6	3712.7	3565.6	3696.0	3707.7	3590.0	3701.6	3627.57
After P1	3771.4	3397.3	3894.2	3714.7	3726.1	3617.1	3736.3	3704.3	3626.6	3752.6	3694.06
After P2	3646.9	3270.8	3654.2	3520.2	3559.6	3473.6	3585.8	3531.3	3494.1	3644.6	3538.11
After P3	3618.4	3252.6	3615.4	3506.0	3548.6	3445.9	3543.4	3471.1	3443.1	3617.5	3506.16
After P4	3616.40	3247.40	3610.10	3503.10	3542.30	3445.60	3537.30	3470.00	3438.70	3616.50	3502.34
GA	3415.80	3218.80	3756.00	3671.20	3693.40	3616.40	3146.80	3288.40	3297.20	3518.00	3462.20
Improvement											
(CS–P1)/CS	−2.26%	−2.62%	−4.89%	−3.46%	−0.36%	−1.44%	−1.09%	0.09%	−1.02%	−1.38%	−1.84%
(CS–P2)/CS	1.12%	1.20%	1.58%	1.96%	4.12%	2.58%	2.98%	4.76%	2.67%	1.54%	2.45%
(CS–P3)/CS	1.89%	1.75%	2.62%	2.36%	4.42%	3.36%	4.13%	6.38%	4.09%	2.27%	3.33%
(CS–P4)/CS	1.95%	1.91%	2.76%	2.44%	4.59%	3.37%	4.29%	6.41%	4.21%	2.30%	3.42%
(GA–P4)/GA	−5.87%	−0.89%	3.88%	4.58%	4.09%	4.72%	−12.41%	−5.52%	−4.29%	−2.80%	−1.45%

Table 4. Improvements obtained in total setup time (in hours) (n^4 iterations, $\alpha = 0.30$).

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Average
CS	193.5	175	221	239	316	262.5	255	227.5	247.5	187.5	232.45
After P1	276.5	262	403	360	329	314	310.5	338.5	284	253.5	313.1
After P2	152	135.5	163	165.5	162.5	170.5	160	165.5	151.5	145.5	157.15
After P3	123.5	117.25	124.15	151.25	151.45	142.75	117.55	105.3	100.45	118.35	125.2
After P4	121.5	112.05	122.85	148.6	151.2	142.5	111.45	104.15	96.1	117.4	122.78
# of exchanges	89743.3	81212.3	77963.6	238112.7	150635.7	217709.4	94256.7	119585.6	36843.2	137666.9	124372.9
# of iterations	5308416	3748096	6250000	9834496	9834496	11316496	4100625	5308416	2560000	4100625	6236167
Improvement											
(CS–P1)/CS	−42.89%	−49.71%	−82.35%	−50.63%	−4.11%	−19.62%	−21.76%	−48.79%	−14.75%	−35.20%	−36.98%
(CS–P2)/CS	21.45%	22.57%	26.24%	30.75%	48.58%	35.05%	37.25%	27.25%	38.79%	22.40%	31.03%
(CS–P3)/CS	36.18%	33.00%	43.82%	36.72%	52.07%	45.62%	53.90%	53.71%	59.41%	36.88%	45.13%
(CS–P4)/CS	37.21%	35.97%	44.41%	37.82%	52.15%	45.71%	56.29%	54.22%	61.17%	37.39%	46.24%
# of exc./# of iter.	1.69%	2.17%	1.25%	2.42%	1.53%	1.92%	2.30%	2.25%	1.44%	3.36%	2.03%

Table 5. Improvement figures (in%) obtained for different settings (total time).

Number of iterations	Phase	Total time		
		$\alpha = 0.05$	$\alpha = 0.10$	$\alpha = 0.30$
n^2	(CS–P1)/CS	−1.84	−1.84	−1.84
	(CS–P2)/CS	2.45	2.45	2.45
	(CS–P3)/CS	2.52	2.69	2.96
	(CS–P4)/CS	2.55	2.79	3.12
n^3	(CS–P1)/CS	−1.84	−1.84	−1.84
	(CS–P2)/CS	2.45	2.45	2.45
	(CS–P3)/CS	2.57	2.96	3.28
	(CS–P4)/CS	2.59	3.03	3.34
n^4	(CS–P1)/CS	−1.84	−1.84	−1.84
	(CS–P2)/CS	2.45	2.45	2.45
	(CS–P3)/CS	2.58	2.97	3.33
	(CS–P4)/CS	2.59	3.04	3.40

Table 6. Improvement figures (in%) obtained for different settings (setup time).

Number of iterations	Phase	Setup time		
		$\alpha = 0.05$	$\alpha = 0.10$	$\alpha = 0.30$
n^2	(CS–P1)/CS	−36.98	−36.98	−36.98
	(CS–P2)/CS	31.03	31.03	31.03
	(CS–P3)/CS	32.06	34.72	39.25
	(CS–P4)/CS	32.41	36.31	41.90
	# of exc./# of iter.	1.02	2.32	3.38
n^3	(CS–P1)/CS	−36.98	−36.98	−36.98
	(CS–P2)/CS	31.03	31.03	31.03
	(CS–P3)/CS	32.80	39.12	44.37
	(CS–P4)/CS	33.07	40.28	45.31
	# of exc./# of iter.	0.91	1.77	2.18
n^4	(CS–P1)/CS	−36.98	−36.98	−36.98
	(CS–P2)/CS	31.03	31.03	31.03
	(CS–P3)/CS	32.82	39.23	45.13
	(CS–P4)/CS	33.09	40.28	46.24
	# of exc./# of iter.	0.86	1.62	2.03

among CLs. Using the proposed four-phase solution procedure, the total setup time is almost halved and the factory could profit from an additional \$8,352,000 USD in revenue in one year.

Applying the proposed method to an order combination procedure (the case of more casting jobs of smaller amounts) is an interesting idea that merits further study. Another problem of interest would consider surface quality specifications. Depending on customers' specifications and/or production characteristics, a production scheduling department might decide on the minimum surface quality

Table 7. Improvements (in %) when phase 2 is omitted (n^4 iterations, $\alpha = 0.30$).

	Total time	Setup time
(CS–P1)/CS	−1.84	−36.98
(CS–P3)/CS	2.97	38.81
(CS–P4)/CS	3.46	47.05

requirement for each casting job. When a new roller is installed, products with the best surface quality are obtained for up to 400 hours. For the next 400 hours of operation, the surface quality obtained is classified as medium. The same roller can be continually used for an additional 400 hours and after 1200 hours the roller is changed. Another extension of this problem could be to use other scheduling objectives, e.g. due date related objectives such as total weighted tardiness. Furthermore, the development of order acceptance policies in the case when there are more orders than production capacity allows might also be an interesting study area.

References

Allahverdi, A., Gupta, J.N.D. and Aldowaisan, T., A review of scheduling research involving setup consideration. *Int. J. Manage. Sci.*, 1999, **27**, 219–239.

Assaf, I., Chen, M. and Katzberg, J., Steel production schedule generation. *Int. J. Product. Res.*, 1997, **35**, 467–477.

Chen, Z.L. and Powell, W.B., Exact algorithms for scheduling multiple families of jobs in parallel machines. *Naval Res. Logist.*, 2003, **50**, 823–840.

Dietrich, B.L. and Escudero, L.F., On solving a 0–1 model for workload allocation on parallel unrelated machines with setups, in *Proceedings 3rd ORSA/TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, 1989, pp. 181–186.

Dorigo, M., Maniezzo, V. and Colorni, A., Ant system: optimization by a colony of cooperating agents. *IEEE Trans. Syst., Man Cyber.*, 1996, **26**, 29–41.

Fowler, J.W., Horng, S.M. and Cochran, J.K., A hybridized genetic algorithm to solve parallel machine scheduling problems with sequence dependent setups. *Int. J. Indust. Eng.*, 2003, **10**, 232–243.

Gascon, A. and Leachman, R.C., A dynamic programming solution to the dynamic, multi-item, single-machine scheduling problem. *Operat. Res.*, 1998, **36**, 50–56.

Ghiani, G., Guerriero, F., Laporte, G. and Musmanno, R., Real-time vehicle routing: solution concepts, algorithms and parallel computing strategies. *Eur. J. Operat. Res*, 2003, **151**, 1–11.

Gravel, M., Price, W.L. and Gagne, C., Scheduling continuous casting of aluminum using a multiple objective ant colony optimization metaheuristic. *Eur. J. Operat. Res.*, 2002, **143**, 218–229.

Harjunkoski, I. and Grossmann, I.E., A decomposition approach for the scheduling of a steel plant production. *Comput. Chem. Eng.*, 2001, **25**, 1647–1660.

Kurz, M.E. and Askin, R.G., Heuristic scheduling of parallel machines with sequence-dependent setup times. *Int. J. Product. Res.*, 2001, **39**, 3747–3769.

Laporte, G., Gendreau, M., Potvin, J. and Semet, F., Classical and modern heuristics for the vehicle routing problem. *Int. Trans. Operat. Res.*, 2000, **7**, 285–300.

Lopez, L., Carter, M.W. and Gendreau, M., The hot strip mill production scheduling problem: a Tabu search approach. *Eur. J. Operat. Res.*, 1998, **106**, 317–335.
Tang, L., Liu, J., Rong, A. and Yang, Z., A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Iron & Steel Complex. *Eur. J. Operat. Res.*, 2000, **124**, 267–282.
Tang, L., Liu, J., Rong, A. and Yang, Z., A review of planning and scheduling systems and methods for integrated steel production. *Eur. J. Operat. Res.*, 2001, **133**, 1–20.
Tang, L., Luh, P.B., Liu, J. and Fang, L., Steel-making process scheduling using lagrangian relaxation. *Int. J. Product. Res.*, 2002, **40**, 55–70.
Yildirim, M. B., Duman, E., Krishnan, K. and Senniappan, K., Parallel machine scheduling with load balancing and sequence dependent setups. *Int. J. Operat. Res.*, 2007, **4**, 42–49.