Chapter 7

Economic Lot Scheduling

7.1 Introduction

In a job shop, each job has its own identity and its own set of processing requirements. In a flexible assembly system, there are a number of different types of jobs and jobs of the same type have identical processing requirements; in such a system, setup times and setup costs are often not important and a schedule may alternate many times between jobs of different types. In a flexible assembly system an alternating schedule is often more efficient than a schedule with long runs of identical jobs.

In the models considered in this chapter, a set of identical jobs may be large and setup times and setup costs between jobs of two different types may be significant. A setup typically depends on the characteristics of the job about to be started and the one just completed. If a job's processing on a machine requires a major setup then it may be advantageous to let this job be followed by a number of jobs of the same type.

In this chapter we refer to jobs as items and we call the uninterrupted processing of a series of identical items a run. If a facility or machine is geared to produce identical items in long runs, then the production tends to be Make-To-Stock, which inevitably involves inventory holding costs. This

M.L. Pinedo, *Planning and Scheduling in Manufacturing and Services*,
DOI: 10.1007/978-1-4419-0910-7_7,

form of production is, at times, also referred to as continuous manufacturing (in contrast to the forms of discrete manufacturing considered in the previous chapters). The time horizon in continuous manufacturing is often in the order of months or even years. The objective is to minimize the total cost, which includes inventory holding cost as well as setup cost. The optimal schedule is typically a trade-off between inventory holding costs and setup costs and is often repetitive or cyclic.

The associated scheduling problem has several aspects. First, the lengths of the runs have to be determined and, second, the order of the different runs has to be established. The run lengths are typically referred to as the lot sizes and they are the result of trade-offs between setup costs and inventory holding costs. The lots have to be sequenced in such a way that the setup times and setup costs are minimized. This scheduling problem is referred to as the *Economic Lot Scheduling Problem (ELSP).*

In the standard ELSP a single facility or machine has to produce n different items. The machine can produce items of type j at a rate of Q_j per unit time. If an item of type j is regarded as a job with processing time p_j, then $Q_j = 1/p_j$. We assume that the demand rate for type j is constant at D_j items per unit time. The inventory holding cost for one item of type j is h_j dollars per unit time. If an item of type j is followed by an item of type k a setup cost c_{jk} is incurred; moreover, a setup time s_{jk} may be required. In some models we assume that a setup involves a cost but no machine time and in other, more general, models we assume that a setup involves a cost as well as machine time. The setup cost and time may be either sequence dependent or independent. If the setup cost (time) is sequence independent, then $c_{jk} = c_k$ ($s_{jk} = s_k$). The problem can be viewed as one of deciding a cycle length x and a sequence of runs or cycle $j_1, j_2, \ldots, j_\nu$. This sequence may contain repetitions, so $\nu \geq n$. The associated run times are $\tau_{j_1}, \tau_{j_2}, \ldots, \tau_{j_\nu}$ and there may be idle time between two consecutive runs.

In practice, there are many applications of economic lot scheduling. In the process industries (e.g., the chemical, paper, pharmaceutical, aluminum and steel industries) setup costs and inventory holding costs are significant. When minimizing the total costs, a scheduling problem often reduces to an economic lot scheduling problem (see Example 1.1.4). There are applications of lot scheduling in the service industries as well. In the retail industry (e.g., Sears, Wal-Mart) the procurement of each item has to be controlled carefully. Placing an order for additional supplies entails an ordering cost and keeping a supply in inventory entails a holding cost. The retailer has to determine the trade-off between the inventory holding costs and the ordering costs.

7.2 One Type of Item and the Economic Lot Size

In this section we consider the simplest case, namely a single machine and one type of item. Since there is only one type of item the subscript j can be dropped, i.e., the production rate is Q and the demand rate is D items per unit time. We assume that the machine capacity is sufficient to meet the demand, i.e., $Q > D$. The problem is to determine the length of a production run. After a run has been terminated and sufficient inventory has been built up, the machine remains idle until the inventory has been depleted and a new run is about to start. Clearly, the length of a production run is determined by the trade-off between inventory holding costs and setup costs. In order to minimize the total cost per unit time we have to find an expression for the total cost over a cycle.

Let x denote the cycle time that has to be determined. If D denotes the demand rate, then the demand over a cycle is Dx and the length of a production run to meet the demand over a cycle is Dx/Q. If the inventory level at the beginning of the production run is zero, then the inventory level goes up during the run at a rate $Q - D$ until it reaches

$$(Q - D)\frac{Dx}{Q}.$$

During the idle period the inventory level goes down at a rate D until it reaches zero and the next production run starts. So the average inventory level is

$$\frac{1}{2}\Big(Dx - \frac{D^2x}{Q}\Big).$$

Each production run incurs a setup cost c. The average cost per unit time due to setups is therefore c/x. Let h denote the inventory holding cost per item per unit time. The total average cost per unit time due to inventory holding costs and setups is therefore

$$\frac{1}{2}h\Big(Dx - \frac{D^2x}{Q}\Big) + \frac{c}{x}.$$

To determine the optimal cycle length we take the derivative of this expression with respect to x and set it equal to zero, yielding

$$\frac{1}{2}hD\Big(1 - \frac{D}{Q}\Big) - \frac{c}{x^2} = 0.$$

Straightforward algebra gives the optimal cycle length

$$x = \sqrt{\frac{2\,Q\,c}{hD(Q - D)}}.$$

The total amount to be produced during a cycle, i.e., the lot size, is

$$Dx = \sqrt{\frac{2\,DQ\,c}{h(Q - D)}}.$$

The lot size Dx is not necessarily an integer number. (This is one of the differences between continuous models and discrete models; this difference is examined more closely in Example 7.2.2.)

The idle time of the machine during a cycle turns out to be

$$x\Big(1 - \frac{D}{Q}\Big).$$

The ratio D/Q is at times denoted by ρ, and may be regarded as the utilization of the machine, i.e., the proportion of time that the machine is busy.

Now consider the limiting case when the production rate Q is arbitrarily high, i.e., $Q \rightarrow \infty$. Then,

$$x = \lim_{Q\rightarrow\infty} \sqrt{\frac{2\,Q\,c}{hD(Q-D)}} = \sqrt{\frac{2c}{hD}}.$$

In this case the lot size is equal to

$$Dx = \sqrt{\frac{2Dc}{h}},$$

which is often called the *Economic Lot Size (ELS)* or *Economic Order Quantity (EOQ).*

All the expressions above are based on the assumption that there is a setup cost but not a setup time. If, in addition to the setup cost, there is also a setup time s and $s \leq x(1-\rho)$, then the solution presented above is still feasible and optimal. If

$$s > x(1-\rho),$$

then the lot size computed above is infeasible. The optimal solution then is the solution where the machine alternates between setups and production runs with a cycle length

$$x = \frac{s}{1-\rho}.$$

That is, the machine is either producing or being set up for the next run. The machine is never idle.

The first example illustrates the use of these formulae.

Example 7.2.1 (The ELSP with and without Setup Times). Consider a facility with a production rate $Q = 90$ items per week, a demand rate $D = 50$ items per week, a setup cost $c = \$2000$, a holding cost $h = \$20$ per item per week, and no setup times. From the analysis above it follows that the cycle time x is 3 weeks and the quantity produced in a cycle is 150. Figure 7.1.a depicts the inventory level over the cycle. The idle time during a cycle is $3(1 - 5/9) = 1.33$ weeks, which is approximately 9 days.

Now suppose that there are setup times. If the setup time is less than 9 days (the length of the idle period), then the 3 week cycle remains optimal.

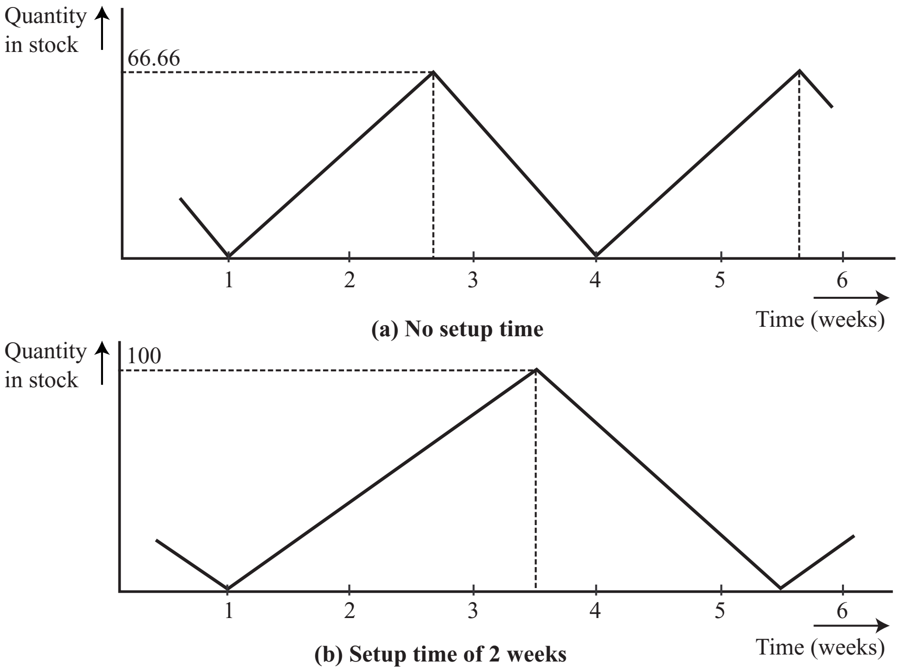

Fig. 7.1. Inventory levels in Example 7.2.1

If the setup time is longer than 9 days, then the cycle time has to be longer. For example, if a setup lasts 2 weeks (because of maintenance and cleaning), then the cycle time is 4.5 weeks. Figure 7.1.b depicts the inventory level over a cycle.

The next example highlights the differences between continuous and discrete settings.

Example 7.2.2 (Continuous Setting vs. Discrete Setting). Consider a production rate Q of 0.3333 items per day, a holding cost h of \$5.00 per item per day and a setup cost c of \$90.00. The demand rate D is 0.10 items per day. Applying the cycle length formula gives

$$x = \sqrt{\frac{60}{0.5(0.3333 - 0.1)}} = 22.678$$

and the number of items in a lot is $Dx = 2.2678$.

In a discrete setting such a number is not feasible. Consider the following discrete counterpart of this instance. The time to produce one item (or job) is $p = 1/Q = 3$ days. The demand rate is 1 item every 10 days. A lot of size k, k integer, has to be produced every $10k$ days. (The solution in the continuous setting suggests that the optimal solution in the discrete setting is either a lot of size 2 every 20 days or a lot of size 3 every 30 days.) The total cost per day with a lot of size 1 every 10 days is $90/10 = \$9.00$. The total cost per day with a lot of size 2 every 20 days is

$$(90 + 7 \times 5)/20 = \$6.25$$

and the total cost per day with a lot of size 3 every 30 days is

$$(90 + 7 \times 5 + 14 \times 5)/30 = \$6.50.$$

So in a discrete setting it is optimal to produce every 20 days a lot of size 2.

7.3 Different Types of Items - Rotation Schedules

Consider again a single machine, but now with n different items. The demand rate for item j is D_j and the machine is capable of producing item j at a rate Q_j. In order to start a production run for item j, a setup cost c_j is incurred. We assume, for the time being, that this setup cost is sequence independent. In this section we determine the best production cycle that contains a single run of each item. Thus, the cycle lengths of the n items have to be identical. Such a schedule is referred to as a *rotation* schedule. The length of the cycle determines the length of each of the production runs. Hence there is only a single decision variable, the cycle length x. In order to determine the optimal cycle length, it is again necessary to find an expression for the total cost per unit time as a function of the cycle length x.

If setups require machine time, then it may not be possible to make the cycle length arbitrarily small since frequent setups may take up too much machine time.

The length of the production run of item j in a cycle is $D_j x/Q_j$. Assume that the inventory level at the beginning of the production run of item j is zero. During the production run, the level increases at rate $Q_j - D_j$ until it reaches level $(Q_j - D_j)D_j x/Q_j$. During the idle period, the inventory decreases at a rate D_j until it reaches zero and the next production run starts. So the average inventory level of item j is

$$\frac{1}{2}\Big(D_j x - \frac{D_j^2 x}{Q_j}\Big).$$

The facility incurs a setup cost c_j for each production run of item j. The average cost per unit time due to setups for item j is therefore c_j/x. The total average cost per unit time due to inventory holding costs and setup costs is therefore

$$\sum_{j=1}^{n} \Big(\frac{1}{2} h_j \Big(D_j x - \frac{D_j^2 x}{Q_j}\Big) + \frac{c_j}{x}\Big).$$

To find the optimal cycle length we take the derivative with respect to x and set it equal to zero, obtaining

$$\sum_{j=1}^{n}\Big(\frac{1}{2}h_jD_j\Big(1-\frac{D_j}{Q_j}\Big)\Big)-\frac{\sum_{j=1}^{n}c_j}{x^2}=0,$$

Straightforward algebra yields the optimal cycle length

$$x=\sqrt{\Big(\sum_{j=1}^{n}\frac{h_jD_j(Q_j-D_j)}{2Q_j}\Big)^{-1}\sum_{j=1}^{n}c_j}.$$

The machine idle time during a cycle can be computed in a manner similar to the single item case. This idle time is equal to

$$x\Big(1-\sum_{j=1}^{n}\frac{D_j}{Q_j}\Big).$$

The ratio $\rho_j = D_j/Q_j$ can be regarded as the utilization factor of the machine due to item j.

Consider the limiting case where the production rates are arbitrarily fast, i.e., $Q_j = \infty$ for $j = 1, \ldots, n$. In this special case the optimal cycle length is

$$x=\sqrt{\Big(\sum_{j=1}^{n}\frac{h_jD_j}{2}\Big)^{-1}\sum_{j=1}^{n}c_j}.$$

Example 7.3.1 (Rotation Schedules without Setup Times). Consider four different items with the following production rates, demand rates, holding costs and setup costs.

items	1	2	3	4
D_j	50	50	60	60
Q_j	400	400	500	400
h_j	20	20	30	70
c_j	2000	2500	800	0

The optimal cycle length x is 1.24 months and the total idle time is $0.48x$ = 0.595 months. Figure 7.2 displays the optimal rotation schedule. The total average cost per unit time can be computed easily and is $2155+2559+1627+2213 = 8554$.

As the setup cost of item 4 is zero, it is clear that a rotation schedule does not make sense here. It makes more sense to spread the production of item 4 uniformly over the cycle to reduce inventory holding costs. In the next section we consider this example again and allow for more general schedules.

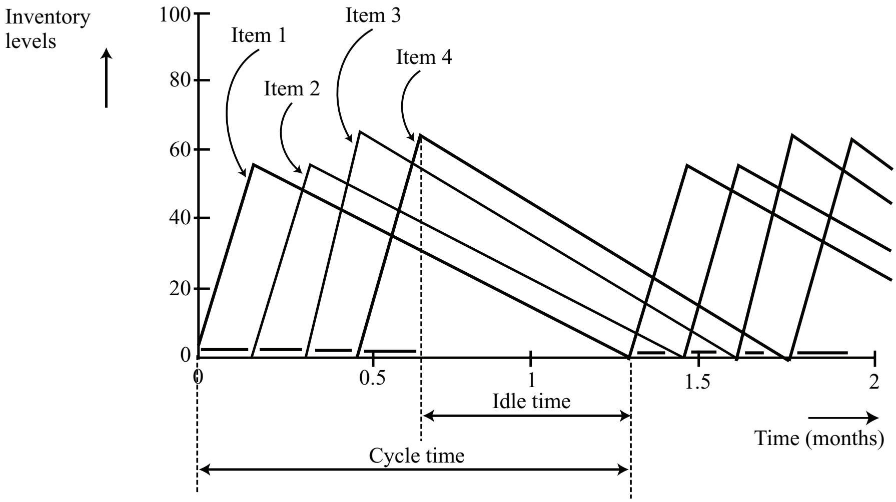

Fig. 7.2. Rotation schedule in Example 7.3.1

In the analysis above the order in which the different runs are sequenced does not matter. We assumed that there were no setup times and that setup costs were sequence independent. So, up to now, there was not any scheduling problem, only a lot sizing problem.

If there are setup times that are sequence independent, i.e., $s_{jk} = s_k$ for all j and k, then the problem still does not have a sequencing component, since the sum of the setup times does not depend on the sequence. If the sum of the setup times is less than the idle time in the rotation schedule computed above, the length of the rotation schedule remains optimal. If the sum of the setup times exceeds the idle time computed above, then the actual optimal cycle length has to be larger than the optimal cycle length obtained before. Actually, the optimal cycle length again turns out to be the cycle length that corresponds to a schedule in which the machine is never idle, i.e.,

$$x = \Big(\sum_{j=1}^{n} s_j\Big)\Big/\Big(1 - \sum_{j=1}^{n} \rho_j\Big).$$

If the setup times are sequence dependent, then there is a sequencing problem and a sequence that minimizes the sum of the setup times has to be found. Minimizing the sum of the setup times in a rotation schedule is equivalent to the so-called *Travelling Salesman Problem (TSP)*, which can be described as follows. A salesman has to visit n cities and the distance from city j to city k is d_{jk}. His objective is to find a tour with the minimum total travel distance. That this TSP is equivalent to our sequencing problem can be shown easily. City j corresponds to item j and the distance from city j to city k, d_{jk}, is equivalent to the setup time needed when item k follows item j, i.e., s_{jk}. The TSP is known to be NP-hard.

If, in the case of sequence dependent setup times, a sequence can be found that minimizes the sum of the setup times and this sum is less than the idle time in the rotation schedule, then the lot sizes computed above, as well as the sequence, are optimal.

However, if the optimal sequence results in a total setup time that is larger than the machine idle time obtained before, then the optimal cycle length has to be larger than the cycle length given in the formula above. The optimal cycle length then again will be such that the machine is always either producing or being setup for the next production run. In any case, the lot sizing problem and the scheduling problem can still be analyzed separately.

The scheduling problem with arbitrary setup times is known to be extremely hard. However, when the setup times have a special structure, an easy solution may exist. Consider, for example, the following setup times:

$$s_{jk} = 0, \qquad j \leq k$$

and

$$s_{jk} = (j - k)s, \qquad j > k.$$

An optimal sequence can be obtained by starting out with the item with the lowest index, continuing with the item with the second lowest index, and so on. At the end of the run of the item with the highest index, a changeover is made to the item with the lowest index in order to start a new cycle. This sequence is obtained by applying the *Shortest Setup Time first (SST)* rule, which is often used as a heuristic in cases with arbitrary setup times (see Appendix C).

Example 7.3.2 (Rotation Schedules with Setup Times). Consider the same four items as in Example 7.3.1. However, there are now sequence dependent setup times. There are $3! = 6$ possible sequences. The setup times are given in the table below.

k	1	2	3	4
s_{1k}	-	0.064	0.405	0.075
s_{2k}	0.448	-	0.319	0.529
s_{3k}	0.043	0.234	-	0.107
s_{4k}	0.145	0.148	0.255	-

This setup matrix is asymmetric, i.e., s_{jk} is not necessarily equal to s_{kj}.

Recall that in the case without setup times the cycle time is 1.24 months and the total idle time is 0.595 months. Since there are only six sequences, all sequences can be enumerated and the best one can be selected. The sequence $1, 4, 2, 3$ requires a total setup of 0.585 months, which is feasible and therefore optimal. However, if SST is used starting with item 1, then the sequence $1, 2, 3, 4$ is selected. This sequence requires a total setup of 0.635 months which exceeds the idle time under the optimal cycle.

7.4 Different Types of Items - Arbitrary Schedules

We now generalize the model described in the previous section to allow for schedules that are more general than rotation schedules. Within a cycle there may be multiple runs of any given item. For example, if there are three different items 1, 2 and 3, then the cycle $1, 2, 1, 3$ is allowed. There may be setup costs as well as setup times.

If ρ_j denotes the utilization factor D_j/Q_j of item j, then a feasible solution exists if and only if

$$\rho = \sum_{j=1}^{n} \rho_j < 1.$$

This necessary and sufficient condition is the same as the condition for the model in Section 7.3. It is intuitive that the setup times do not have any impact on this feasibility condition. The setup times do take up machine time; but, if a machine is operating close to capacity, then the cycle time and individual production runs just have to be made long enough in order to minimize the impact of the setup times.

In contrast to the model in the previous section for which there exists a closed form solution (at least, when the setup times are sequence independent), the problem considered in this section is very hard. There does not exist an efficient algorithm for this problem. However, there are good heuristics that usually lead to satisfactory solutions. In what follows, we describe one such heuristic for the case with sequence independent setup times, i.e., $s_{jk} = s_k$. Let $\mathcal{S}$ denote the set of all possible sequences of arbitrary length and j_l the index of the item produced in position l of the sequence. So $j_1, \ldots, j_\nu$, where $\nu \geq n$, denotes the production sequence of a given cycle. The sequence may contain repetitions. Consider the item that is produced in the l-th position of the sequence. If $j_l = k$, then item k is produced in the l-th position of the sequence. In the remainder of this section, the superscript l is used to refer to data related to the item produced in the l-th position of the sequence, e.g., $Q^l = Q_{j_l}$ and if the item in the l-th position is item k, then $Q^l = Q_{j_l} = Q_k$.

The production of the item in position l involves a setup cost c^l, a setup time s^l, a production time τ^l, and a subsequent idle time u^l which may be zero. If item k is produced in the l-th position, item k may be produced again within the same cycle. Let x denote the cycle time and v the time from the start of the production of item k in the l-th position till the start of the next production of item k (this may be in the same cycle or in the next cycle). So

$$v = \frac{Q^l \tau^l}{D^l}$$

and if $j_l = k$, then

$$v = \frac{Q_k \tau^l}{D_k}.$$

The highest inventory level is $(Q^l - D^l)\tau^l$. The total inventory cost for the production run of item k in position l is

$$\frac{1}{2}h^l(Q^l - D^l)\Big(\frac{Q^l}{D^l}\Big)(\tau^l)^2.$$

Let I_k denote the set of all positions in the sequence in which item k is produced and L_l all the positions in the sequence starting with position l (when item k is produced) up to, but not including, the position in the sequence where item k is produced next. The definition of L_l assumes that the sequence $j_1, \ldots, j_\nu$ repeats itself. Let $\mathcal{S}$ denote the set of all possible cyclic schedules. The ELSP can now be written as

$$\min_{\mathcal{S}} \quad \min_{x,\tau^l,u^l} \quad \frac{1}{x}\Big(\sum_{l=1}^{\nu}\frac{1}{2}h^l(Q^l - D^l)\Big(\frac{Q^l}{D^l}\Big)(\tau^l)^2 \;+\; \sum_{l=1}^{\nu}c^l\Big)$$

subject to

$$\sum_{j\in I_k} Q_k\tau^j = D_k x \qquad\qquad \text{for } k = 1, \ldots, n,$$

$$\sum_{j\in L_l}(\tau^j + s^j + u^j) = \Big(\frac{Q^l}{D^l}\Big)\tau^l \qquad\qquad \text{for } l = 1, \ldots, \nu,$$

$$\sum_{j=1}^{\nu}(\tau^j + s^j + u^j) = x$$

The first set of constraints ensures that enough time is allocated to the production of item k to meet its demand over the cycle. The second set ensures that enough of the item in position l is produced to meet the demand till the next time that item is produced.

The problem described above may be viewed as being composed of a master problem and a subproblem. The master problem focuses on the search for the best sequence $j_1, \ldots, j_\nu$ (an element of $\mathcal{S}$), and the subproblem must determine the optimal production times, idle times, and cycle length (τ^l, u^l, x) given the sequence.

That the subproblem is relatively simple can be argued as follows. If the sequence $j_1, \ldots, j_\nu$ is fixed, then the first set of constraints in the nonlinear programming formulation is redundant, since substitution of the third set into the first set yields

$$\sum_{j\in I_k}\Big(\frac{Q^j}{D^j}\Big)\tau^j = \sum_{j=1}^{\nu}(\tau^j + s^j + u^j),$$

which is the sum of the second set over all positions in I_k. So, given a fixed sequence, the nonlinear programming problem that determines the optimal production times and idle times can be formulated as follows:

$$\min_{x,\tau^l,u^l} \quad \frac{1}{x}\Big(\sum_{l=1}^{\nu}\frac{1}{2}h^l(Q^l-D^l)\Big(\frac{Q^l}{D^l}\Big)(\tau^l)^2 \ + \ \sum_{l=1}^{\nu}c^l\Big)$$

subject to

$$\sum_{j\in L_l}(\tau^j+s^j+u^j)=\Big(\frac{Q^l}{D^l}\Big)\tau^l \qquad\qquad \text{for } l=1,\ldots,\nu,$$

$$\sum_{j=1}^{\nu}(\tau^j+s^j+u^j)=x$$

The master problem, i.e., finding the best sequence $j_1,\ldots,j_\nu$, is more complicated. One particular heuristic yields good sequences in practice. This heuristic is in what follows referred to as the *Frequency Fixing and Sequencing (FFS)* heuristic. This FFS heuristic consists of three phases:

(i) The computation of relative frequencies phase.
(ii) The adjustment of relative frequencies phase.
(iii) The sequencing phase.

The first phase determines the relative frequencies with which the various items have to be produced. The number of times item k is produced during a cycle is denoted by y_k. In the second phase, these production frequencies are adjusted so they can be spaced out evenly over the cycle; the adjusted frequency of item k is denoted by y_k'. In the third and last phase these adjusted frequencies are used to produce an actual sequence.

The first phase determines, besides the relative frequencies y_k, also the corresponding run times τ_k. If the runs of item k are of equal length and evenly spaced, then the frequency y_k and the cycle time x determine the run time τ_k, i.e.,

$$\tau_k=\frac{\rho_k x}{y_k}.$$

To compute the y_k we relax the original nonlinear programming formulation by dropping the second set of the three sets of constraints. Without these interlinking constraints the actual sequence is no longer important. Optimizing over sequences now becomes optimizing over the cycle time x and run times τ_k or, equivalently, over production frequencies y_k.

Substitutions lead to the following modifications in the objective function of the original nonlinear programming formulation of the ELSP problem:

$$\frac{1}{x}\Big(\sum_{l=1}^{\nu}\frac{1}{2}h^l(Q^l-D^l)\Big(\frac{Q^l}{D^l}\Big)(\tau^l)^2+\sum_{l=1}^{\nu}c^l\Big)$$
$$=\frac{1}{x}\Big(\sum_{k=1}^{n}\frac{1}{2}y_k h_k(Q_k-D_k)\Big(\frac{Q_k}{D_k}\Big)(\tau_k)^2+\sum_{k=1}^{n}y_k c_k\Big)$$

$$
\begin{aligned}
&= \sum_{k=1}^{n} \frac{1}{2} h_k (Q_k - D_k) \Big(\frac{Q_k}{D_k} \Big) \rho_k \tau_k + \sum_{k=1}^{n} \frac{c_k \rho_k}{\tau_k} \\
&= \sum_{k=1}^{n} \frac{1}{2} h_k (Q_k - D_k) \tau_k + \sum_{k=1}^{n} \frac{c_k \rho_k}{\tau_k} \\
&= \sum_{k=1}^{n} \frac{a_k \tau_k}{\rho_k} + \sum_{k=1}^{n} \frac{c_k \rho_k}{\tau_k} \\
&= \sum_{k=1}^{n} \frac{a_k x}{y_k} + \sum_{k=1}^{n} \frac{c_k y_k}{x},
\end{aligned}
$$

where

$$
a_k \ = \ \frac{1}{2} h_k (Q_k - D_k) \rho_k \ = \ \frac{1}{2} h_k (1 - \rho_k) D_k .
$$

Of course, in any feasible schedule the relative frequencies y_k have to be integers. This implies that the run times τ_k cannot assume just any values, since they are determined by the relative frequencies. However, in order to make the problem easier, we delete the integrality constraints on the y_k and thus relax the constraints on the τ_k as well (basically deleting the first set of constraints in the original nonlinear programming formulation). Disregarding the integrality constraints on the y_k results in a relatively easy nonlinear programming problem.

$$
\min_{y_k, x} \sum_{k=1}^{n} \frac{a_k x}{y_k} + \sum_{k=1}^{n} \frac{c_k y_k}{x},
$$

subject to

$$
\sum_{k=1}^{n} \frac{s_k y_k}{x} \leq 1 - \rho .
$$

The constraint in this problem is equivalent to the last constraint in the original formulation. If the left hand side of this constraint is strictly smaller than the right hand side, then the sum of the setup times is less than the time the machine is not producing, implying there is still some idle time remaining.

Before presenting a solution for this simplified nonlinear programming problem some observations have to be made. First, a solution that is feasible for the simplified nonlinear programming problem may not be feasible for the original nonlinear programming problem; the solution to the simplified problem may require some tweaking. Second, it is clear that the simplified nonlinear programming problem has an infinite number of optimal solutions. If the solution $x^*, y_1^*, \ldots, y_n^*$, is optimal and

$$
\sum_{k=1}^{n} \frac{s_k y_k}{x} < 1 - \rho
$$

(i.e., the inequality is strict), then any solution $kx^*, ky_1^*, \ldots, ky_n^*$, with k integer, is also optimal. Basically, multiplying the first cyclic schedule by an integer k results in an identical cyclic schedule. A third observation is the following: if the inequality above is strict, then the solution $x^*, y_1^*, \ldots, y_n^*$ would also be optimal if all setup times are zero. However, if in the optimal solution

$$\sum_{k=1}^{n} \frac{s_k y_k}{x} = 1 - \rho,$$

then the setup times play an important role. The fact that there is no idle time implies that the optimal solution requires relatively high production frequencies with relatively short run times (possibly due to high holding costs or low setup costs). These high frequencies require that the maximum proportion of machine time, i.e., $(1 - \rho)$, is dedicated to setup times.

The nonlinear programming problem can be dealt with as follows. Incorporating the constraint in the objective function using a Lagrangean multiplier λ ($\lambda \geq 0$), results in an unconstrained nonlinear optimization problem with objective function

$$\min_{y_k, x} \sum_{k=1}^{n} \frac{a_k x}{y_k} + \sum_{k=1}^{n} \frac{c_k y_k}{x} + \lambda\Big(\sum_{k=1}^{n} \frac{s_k y_k}{x} - (1 - \rho)\Big).$$

Taking the partial derivative of this function with respect to y_k and setting it equal to zero yields

$$y_k = x\sqrt{\frac{a_k}{c_k + \lambda s_k}}.$$

The cycle length x can be adjusted so that the production frequencies take appropriate values (for example, one may choose the cycle length x so that the smallest frequency value is approximately equal to 1). If there are idle times, then λ is set equal to zero. If there are no idle times, then the λ has to satisfy the equation

$$\sum_{k=1}^{n} \Big(s_k \sqrt{\frac{a_k}{c_k + \lambda s_k}}\Big) = 1 - \rho,$$

since

$$\sum_{k=1}^{n} s_k y_k = (1 - \rho) x.$$

The solution y_k is unlikely to be integer. To find an integer solution that is close to the values obtained for y_k may require the construction of a long sequence with high frequencies.

The second phase of the FFS heuristic makes adjustments in the frequencies y_k. It has been shown in the literature that it is possible to find a new set of frequencies y_k' that are integers and powers of 2 with the cost of this new solution being within 6% of the cost of the original solution. Of course, the run times of item k have to change then as well. The new run times, τ'_k, can be computed by assuming that the total idle time remains the same and the runs of item k are of equal length and equally spaced.

The third phase of the FFS heuristic generates the actual sequence. The heuristic used here has its roots in the heuristic used for scheduling n different jobs on a number of parallel machines to minimize the makespan. (Recall from the second section of Chapter 5 that the most popular heuristic for this problem is the LPT rule). Let

$$y'_{\max} = \max(y'_1, \ldots, y'_n).$$

For each item k, there are y'_k jobs with the same estimated processing time τ'_k (assuming that the lots will be equally spaced). Now consider a scheduling problem with $y'_{\max}$ machines in parallel and y'_k jobs of length τ'_k, $k = 1, \ldots, n$, (implying a total of $\sum_{k=1}^{n} y'_k$ jobs). There is an additional restriction in that item k with frequency y'_k must have the y'_k lots (jobs) placed on machines that are equally spaced. For example, if $y'_{\max} = 6$ and $y'_k = 3$, then there are two choices: the three jobs are assigned either to machines 1, 3 and 5 or to machines 2, 4 and 6. Now the following variation of the LPT heuristic can be used: the pairs (y'_k, τ'_k) are listed in decreasing order of y'_k. Pairs with identical frequencies y'_k are listed in decreasing order of the estimated processing time τ'_k. The pairs are taken from the list one by one starting at the top. When the pair (y'_k, τ'_k) is taken from the list, the corresponding y'_k jobs of length τ'_k are put on the machines (satisfying the spacing restriction) so that the maximum of the total processing assigned so far to the selected y'_k machines is minimized. After all pairs in the list have been assigned, the resulting sequences on the $y'_{\max}$ machines are concatenated, i.e., machine 1, followed by machine 2, and so on, to obtain a single sequence. This idea is based on the fact that after all jobs have been scheduled and the total processing is more or less equally partitioned over all the machines, the concatenated sequence will maintain the equal spacing of the various runs of any given item.

The following example illustrates the application of the FFS heuristic to an instance without setup times.

Example 7.4.1 (Application of the FFS Heuristic without Setup Times). Consider again the situation described in Example 7.3.1. However, now the schedule does not necessarily have to be a rotation schedule. There are setup costs but no setup times. Since item 4 has no setup cost and a fairly high holding cost, its production should be spread out as uniformly as possible in between the production of the other three items.

In order to find the frequencies y_k, we have to first solve the unconstrained optimization problem. Note that

$$(1 - \rho)x = 0.48x,$$

$$y_k = \frac{\rho_k x}{\tau_k},$$

and

$$a_k = \frac{1}{2} h_k (Q_k - D_k)\rho_k.$$

The following values can be computed easily.

items	1	2	3	4
D_j	50	50	60	60
Q_j	400	400	500	400
h_j	20	20	30	70
c_j	2000	2500	800	0
ρ_j	0.125	0.125	0.12	0.15
a_j	437.5	437.5	792	1785

It immediately follows that

$$\begin{aligned} y_1 &= 0.46x \\ y_2 &= 0.42x \\ y_3 &= 0.99x \\ y_4 &= \infty \end{aligned}$$

Suppose that the cycle time x is set equal to 2 months. This cycle time corresponds to the following approximate values for $y_1, \ldots, y_4$: $y_1 = y_2 = 1$, $y_3 = 2$ and $y_4 = 16$. The choice of y_4 is somewhat arbitrary but it has to be made high. The higher y_4, the more uniform the production of item 4 can be made in the final solution. Given a cycle time of 2 months and the production frequencies above, the runtimes τ_k of the four items are $\tau_1 = \tau_2 = 0.25$, $\tau_3 = 0.12$, $\tau_4 = 0.3/16$.

Now we apply the LPT-like heuristic. The number of machines in parallel is $y_{\max} = 16$. Item 4, the first to be assigned, is assigned to all 16 machines with all 16 processing times equal to 0.3/16. Item 3 is assigned next and is assigned to machines 1 and 9 with the two processing times equal to 0.12. Item 1 is then put on machine 5 and item 2 on machine 13. Concatenating the sequences of the 16 parallel machines results in the cyclic schedule

$$| \, 4\,,3 \,|\, 4 \,|\, 4 \,|\, 4 \,|\, 4\,,1 \,|\, 4 \,|\, 4 \,|\, 4 \,|\, 4\,,3 \,|\, 4 \,|\, 4 \,|\, 4 \,|\, 4\,,2 \,|\, 4 \,|\, 4 \,|\, 4 \,|.$$

Item 4 goes first for 0.3/16 months, followed by item 3 for 0.12 months. Item 4 goes next four times in a row, each time for 0.3/16 months (these four runs are separated in the final solution by idle times). Item 1 follows for 0.25 months. Item 4 goes again four times, and so on.

A feasibility check has to be done. It is clear that, in an ideal solution, the runs of each item are spaced evenly over the cycle. If the runs of an item are evenly spaced, we are assured that there are no stockouts. Attempting to

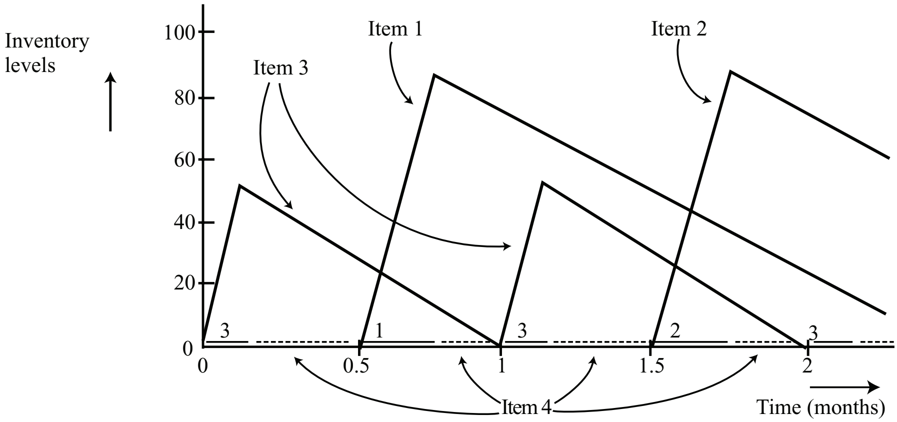

Fig. 7.3. Schedule in Example 7.4.1

uniformize the production of item 4 over the cycle results in many short runs that are either separated by idle times or by production runs of other items. We need to check whether, whenever items 1, 2, or 3 are produced, the stock of item 4 is sufficient to cover the demand in the periods that the other items are in production.

This solution could have been obtained in another way as well. As mentioned above, the y_4 was selected somewhat arbitrarily. The reason for choosing a high value is that it enables us to uniformize the item's production over the cycle. It is clear that the y_4 has to be chosen at least as large as the sum of all the other y's, i.e., at least 4. If $y_4 = 4$, then the algorithm yields the sequence

$$4, 3, 4, 1, 4, 3, 4, 2.$$

A schedule can now be constructed as follows. First the production runs of items 1, 2, and 3 are spaced evenly over the cycle and fixed. Then the production of item 4 is scheduled separately in between the remaining idle times. This production of item 4 is scheduled evenly over time (in *many* short runs). However, before any of the items 1, 2, or 3 has to go into production, a given amount of inventory of item 4 has to be built up. The inventory level of item 4 depends on the length of the runs of items 1, 2, and 3. It is only during these buildups of inventory that holding costs are incurred for item 4.

The entire schedule is depicted in Figure 7.3. The average total cost per unit time can be computed and is equal to

$$1875 + 2125 + 1592 + 190 = 5782.$$

Recall that the cost of the rotation schedule in Example 7.3.1 is 8554.

The next example illustrates the application of the FFS heuristic on an instance with setup times. In contrast to the previous example there is now, under the optimal sequence, no idle time on the machine.

Example 7.4.2 (Application of the FFS Heuristic with Setup Times). Consider the instance in the previous example but now with setup times. The setup times are sequence independent.

items	1	2	3	4
D_j	50	50	60	60
Q_j	400	400	500	400
h_j	20	20	30	70
c_j	2000	2500	800	0
s_j	0.5	0.2	0.1	0.2
ρ_j	0.125	0.125	0.12	0.15
a_j	437.5	437.5	792	1785

Note that, because of the nonzero setup time of item 4 the frequency of item 4 cannot be made arbitrarily high. In order to find the frequencies y_k, we first have to find a λ that satisfies the equation

$$\sum_{k=1}^{n}\left(s_k\sqrt{\frac{a_k}{c_k+\lambda s_k}}\right)=1-\rho.$$

It can be verified easily that $\lambda \approx 8000$ satisfies this equation. With this value of λ the y_k frequencies can be computed as a function of the cycle time x, i.e.,

$$y_k = x\sqrt{\frac{a_k}{c_k+\lambda s_k}}.$$

$$\begin{aligned} y_1 &= 0.27x \\ y_2 &= 0.33x \\ y_3 &= 0.70x \\ y_4 &= 1.05x \end{aligned}$$

If the cycle time x is fixed at 3 months, then the approximate values $y_1', \ldots, y_4'$ can be either $(1,1,2,2)$ or $(1,1,2,4)$. Both solutions are power of two solutions. Compare these solutions with the frequency values in the previous example, i.e., $(1,1,2,16)$. It is clear that, because of the setup times, the frequency of item 4 cannot be as high as in the previous example. There is simply not enough idle time for that many setups.

Consider the solution $x = 3$ with frequencies $(1,1,2,2)$. The item sequence is $1,3,4,2,3,4$. It has to be verified whether this solution is feasible. The idle time before taking setups into account is $0.48 \times 3 = 1.44$ months and the total amount of setup time required is 1.3, which implies that the schedule is feasible. Actually, this means that the cycle time x can be made slightly smaller than 3, and a slightly smaller cycle time will give a better solution. (In the previous example, without setup times, the cycle length was 2 months; here the cycle time was made 3 months assuming that there would not be any idle time.) The average total cost per unit time can be computed as in the previous examples.

Consider the solution $x = 3$ with frequencies $(1, 1, 2, 4)$. The order of the items is

$$1, 4, 3, 4, 2, 4, 3, 4.$$

The total setup time required during a cycle is 1.7 months. This implies that a cycle length of 3 months is not feasible with these frequencies. In order to have these frequencies the cycle length has to be made larger (see Exercise 7.10).

7.5 More General ELSP Models

All models considered in the previous sections are single machine models. Some of these models can be extended fairly easily. For example, consider the model with multiple products on m identical machines in parallel. There are setup costs but no setup times. Any particular item has to be processed on one and only one of the m machines. The utilization factor of item j is again defined as $\rho_j = D_j/Q_j$ and in order for a feasible solution to exist we must have

$$\sum_{j=1}^{n} \rho_j \leq m.$$

Suppose that the schedule for each of the machines has to be a rotation schedule. If the cycle times of the m rotation schedules have to be equal, the problem is relatively easy and not much different from the one described in Section 7.3. The only additional issue that needs to be resolved is the assignment of the items to the different machines. Assuming that each item has to be produced on one and only one machine, the loads have to be balanced and the sum of the ρ_j's of the items assigned to any one machine has to be less than one. To find a good balance or, equivalently, a good partition of the n different items over the m machines, we can use the LPT heuristic with the ρ_j values playing the role of processing times. The LPT heuristic (used for minimizing the makespan in a parallel machine environment) will result in a reasonably good assignment of the items to the m machines.

Example 7.5.1 (Rotation Schedules with Machines in Parallel). Consider the situation in Example 7.3.1. Instead of a single machine, we now have 2 machines in parallel. The production rate of each of the two machines is half the production rate of the machine in Example 7.3.1. The data are presented below.

items	1	2	3	4
D_j	50	50	60	60
Q_j	200	200	250	200
h_j	20	20	30	70
c_j	2000	2500	800	0

Because the two machines have the same cycle length, the formula in Example 7.3.1 can be used here as well. The optimal cycle x in this case turns out to be 1.35 months (the optimal cycle length with a single machine at twice the speed is 1.24 months).

However, if the cycle times of the m rotation schedules are allowed to be different, then the problem becomes more difficult. We can reduce total cost by taking advantage of different cycle times. Again, an assignment of the items to the machines has to be found while maintaining a proper machine load balance. There is now an additional difficulty. If an item with a cost structure that favors a long cycle time is assigned to the same machine as an item with a cost structure that favors a short cycle time, then the solution is not likely to be a good one. One can deal with this difficulty as follows. Consider each item as a single product model (as in Section 7.2) and compute its cycle time. Rank the items in decreasing order of their cycle times. Start taking the items from the top of the list and put them on one machine. Keep assigning items to this machine until its capacity is exhausted, i.e., the allocation of an item makes the sum of the ρ_j's larger than one. This last item is then reallocated to the second machine, and so on. This procedure may not lead to a good load balance, and may even lead to an infeasible solution (i.e., the sum of the ρ_j's on the last machine may be larger than one). If that is the case, then items on adjacent machines have to be swapped in order to obtain a better balance.

The parallel machine model with rotation schedules and sequence dependent setups is of course harder. The assignment of items to the m machines now has to consider machine balance, preferred cycle times, as well as setup times on all machines. The setup time structure becomes especially important when there are large differences between setup times. Very little research has been done on this problem.

When the schedules on the different machines do not have to be rotation schedules, i.e., the parallel machines generalization of the model considered in Section 7.4, the problem is even harder. However, now it is no longer necessary to assign items with similar cycle times to the same machine. It is clear that in the parallel machine environment there are still many unresolved issues that require more research.

Another important extension of the single machine setting is the environment with machines in series, i.e., the flow shop. Consider a single machine feeding another single machine with the production rates and setup costs of the two machines being identical. Hence the cost structures of an item on the upstream machine and on the downstream machine are the same. The machines can be scheduled and synchronized so that the items, when they leave the upstream machine, can immediately start their processing on the downstream machine without having to wait. Because of this synchronization, the results in Sections 7.3 and 7.4 can be extended to the case of similar machines with identical costs in series.

Example 7.5.2 (Rotation Schedules with Machines in Series). Consider the same product mix as in Example 7.3.1. However, now instead of a single machine, we have two machines in series. After an item has completed its processing on the upstream machine, it has to go to the downstream machine and complete its processing there. There are setup costs but no setup times. The setup costs of an item on the two machines are the same. A rotation schedule is needed for the two machines (i.e., the same rotation schedule must be used for both machines).

If the rotation schedule is such that an item with a long processing time (i.e., with a low production rate) is followed by an item with a short processing time (i.e., with a high production rate), then the item with the short processing time may have to wait between the two machines. Assume that at the beginning of the rotation schedule machine 1 starts with the production of the item with the shortest processing time (i.e., with the highest production rate), it then continues, without stopping, with the item with the second shortest processing time, and so on. After it completes the run of the item with the longest processing time machine 1 remains idle until it is necessary to start the new cycle. In this way any item that comes out of machine 1 can start immediately on machine 2 without having to wait, i.e., there is no Work-In-Process in between the two machines.

The inventory costs of the finished goods are exactly the same as in the single machine case, so this system can be analyzed as a single machine. However, there are now two setup costs, instead of only one. This implies that the optimal cycle length is $\sqrt{2} = 1.4142$ times longer than the optimal cycle length for a single machine. This result can be extended easily to m identical machines in series.

The results in the previous example can be further generalized. Consider m machines in series with identical production rates for each product type, but different setup costs. This problem can still be reduced to a single machine problem with production rates identical to those of one of the machines in the original problem. However, now the setup costs have to be set equal to the sum of the setup costs of the original m machines, i.e., the setup cost for item j in the new problem is

$$c_j = \sum_{i=1}^{m} c_{ij}.$$

When the machines do not have identical production rates for each product type the problem is not that easy. Consider first the case with two machines that have identical setup costs but different speeds. But the speed structure is uniform over the items, i.e., the production rate of item j on machine i is $Q_{ij} = v_i Q_j$, where v_i is a speed factor of machine i. One approach is to first analyze the slow machine as a single machine in isolation and then adapt the fast machine accordingly (since the high speed machine provides more flexibility). The schedule of the fast machine can be adapted in such a way that there is no WIP in between the two machines. This model can then be analyzed as a single machine with the production rates of the slow machine and setup costs that are the sum of the setup costs of the two machines.

When the production rates are not uniform, it may become necessary to schedule Work-In-Process (WIP) between the machines. The carrying cost of the WIP in between the machines may be different from the holding cost of the finished goods. This makes the model more complicated. Very little research has been done on this problem.

An even more general machine environment is the flexible flow shop, i.e., a number of stages in series with at each stage a number of machines in parallel. Under very special conditions optimal rotation schedules can be determined for such a machine environment. For example, consider two stages in series with at each stage two machines in parallel. For any given product type the production rates of the four machines are the same and so are the setup costs. Under these circumstances there is no need for any WIP in between two stages. This makes it possible to determine the optimal rotation schedules relatively easily when the cycle times of all four machines have to be the same.

7.6 Multiproduct Planning and Scheduling at Owens-Corning Fiberglas

Owens-Corning Fiberglas is a leading manufacturer of fiberglass products and has a large manufacturing facility in Anderson, South Carolina. In its manufacturing process, molten fiberglass is formed and the glass is spun onto spools of various sizes. This material is used to weave fabric and to produce chopped strand mat. Fiberglass mat is sold in rolls of various widths and weights, treated with one of three process binders, and trimmed at one or both edges or not at all. The demand for the products comes mainly from the marine industry for the manufacture of boat hulls, and from the residential construction industry for bath tubs and shower booths.

At the time when a production planning and scheduling system was developed for this facility, the product line consisted of over 200 distinct mat items. Twenty-eight of these represented over 80% of the total annual demand and were treated as high volume standard (stock) products. The remaining items were made to order. The manufacturing facility had two main production lines referred to as Mat Lines 1 and 2. Line 1 had approximately three times the capacity of Line 2 and could produce mat 76 inches wide, whereas Line 2 was limited to 60 inches. The product came off these lines in the form of 175- to 230-pound cylinders. The average cost of down time on Line 1 was approximately \$300 per hour; the average cost of down time on Line 2 was less. Maintenance costs were related to the frequency of job changeovers. In addition, each time a product change was made on a line, there was a sequence dependent setup cost partly due to material waste. The monthly costs due to setups ranged from \$15,000 to \$50,000 (with approximately 75 job changes and 50 hours of down time).

The production planning and scheduling system developed for the mat lines focused on three issues, namely aggregate planning (focusing on inventory costs and workforce scheduling), production run quantities and lot sizing (taking line assignments and inventory levels into account), and detailed line sequencing of Make-to-Stock and Make-to-Order products (taking setup costs into account). The system developed for the mat lines consisted therefore of three main modules, namely,

(i) the aggregate planning module,
(ii) the lot sizing module, and
(iii) the sequencing module.

The aggregate planning module used as input the aggregate demand forecast for the next twelve months. Its objective was to minimize the sum of direct payroll costs, overtime costs and hiring and firing costs. The time horizon ranged from three to twelve months. The optimization method in this module was based on a production switching heuristic; this rule considered inventory levels and forecasts of future demand and, based on these data, determined the appropriate production rates. The output of this model included targets with respect to aggregate inventory levels, production rates and levels of employment.

The output of the aggregate planning module served as input to the lot sizing module. The lot sizing module also required a detailed short term demand forecast for each stock item. The time horizon considered in this module was up to three months. The output from this module were the line assignments and the lot sizes. The optimization in this module was based on a linear program formulation. The objective was to minimize all the relevant setup costs and production costs subject to several sets of constraints. The inventory level constraints ensured that aggregate inventory levels and safety stock levels were met and the production balance constraints guaranteed demand satisfaction as well as inventory conservation. The linear program was solved using the MPSX package and the program was run on a monthly basis providing the plant with specific inventory levels, lot sizes and line assignments for the coming months.

The output of the lot sizing module served as input for the sequencing module. The time horizon of this module was one month, and its main objective was the minimization of the sequence dependent setup costs. The dominant components of setup costs were direct downtime and mat waste. Changeovers were classified as fiber changes, width changes, weight changes, and slitter changes. A distinction could be made within each family of changeovers based upon the direction of the change. For example, it was easier to decrease weight and width than to increase them. The sequencing heuristic was based on sim-

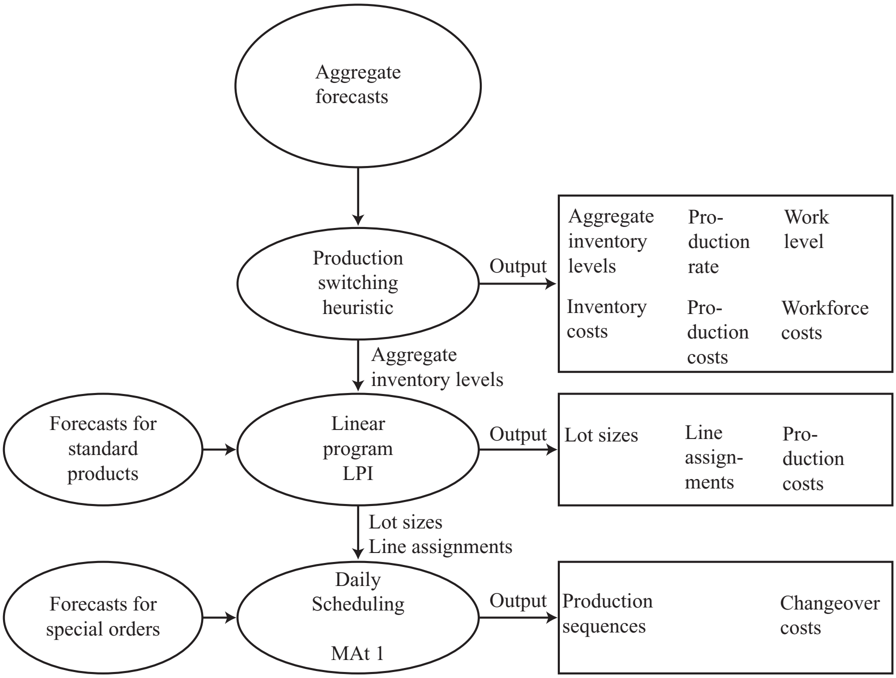

Fig. 7.4. Overview of system at Owens-Corning Fiberglass

ple dispatching rules and the sequencing module was run on a weekly basis. The entire system is depicted in Figure 7.4.

Implementation of the system led to major improvements in the operation of the plant. The average number of changeovers went down from an average of 70 before the system's implementation to an average of 40 after the system's implementation.

The Owens-Corning Fiberglas environment is somewhat similar to the setting described in Section 7.4. However, Owens-Corning had two machines in parallel instead of the single machine in Section 7.4 (a parallel machine environment was considered in Section 7.5). Note that the FFS heuristic described in Section 7.4 is based on a nonlinear programming formulation, whereas the lot sizing module in the Owens-Corning Fiberglas system was based on a linear programming formulation.

7.7 Discussion

The models discussed in this chapter have similarities as well as differences with the models for the the flexible assembly systems described in Chapter 6. In both chapters the planning horizons are basically unbounded. (This is in contrast to the models described in Chapters 4 and 5, which all have a finite number of jobs.) However, the objectives considered in Chapter 6 are fundamentally different from the objectives considered in this chapter. In Chapter 6, the typical objective is to maximize the throughput or, equivalently, to minimize the cycle time. In this chapter, the throughput is basically given, since the demand levels are known. The objective is to minimize the sum of the inventory carrying costs and the setup costs. Nonetheless, the objectives in this chapter display some similarities with the objectives in paced assembly systems.

The models considered in this chapter are very important for industries that produce Make-To-Stock and for environments with setup times and costs. Examples of these industries include the paper industry, the aluminum industry and the steel industry. If one compares the models described in this chapter with the problems that have to be solved in those industries, then a number of issues arise. The problems in practice are, of course, more complicated than the models considered in this chapter. Often, if there are multiple machines in parallel, the production rates of any given item on the various machines may vary. Run lengths have in practice, besides an impact on the inventory costs and the total change-over costs, also an effect on various other factors, including

(i) the quality of the finished product,
(ii) the production yield or the amount of waste incurred,
(iii) the productivity of the facility and its total production capacity.

These three factors, which are not independent, are seldom included in medium term or long term planning models. The three factors are somewhat related to one another. The quality of the products in process industries (which typically is a continuous measure rather than a discrete measure) depends strongly on the length of the run. The longer the length of the run, the higher the average quality of production. In the process industries there is usually also a yield or waste problem (cutting stock or trim related issues). If the run length is very small, then the average waste tends to increase. So the more changeovers there are, the lower the average quality of the product, and the lower the productivity and the capacity. The cost of machine capacity is basically determined by opportunity costs (i.e., the shadow prices or dual prices of the resources involved).

Practical problems are often a combination of Make-To-Stock and Make-To-Order. The Make-To-Stock aspects involve problems such as those described in this chapter whereas the Make-To-Order aspects are related to job shop scheduling problems. Researchers have been analyzing inventory problems in which the facilities are assumed to be set up in series. This area of research, often referred to as multi-echelon inventory theory or supply chain management, is considered in the next chapter.

Exercises

7.1. Consider four different products with the following demand rates, production rates, holding costs and setup costs.

items	1	2	3	4
D_j	50	50	60	60
Q_j	400	500	500	400
h_j	20	20	30	70
c_j	2000	1000	1000	100

(a) Find the optimal rotation schedule. Determine its cycle length, and the total idle time.

(b) Suppose now that item 4 can be produced many times during a cycle. Items 1, 2 and 3 still can be produced only once during a cycle. Find the optimal production schedule. How does the optimal cycle length compare with the original optimal cycle length?

7.2. Consider two identical machines in parallel. Four items have to be produced.

items	1	2	3	4
D_j	50	50	60	60
Q_j	200	200	300	300
h_j	20	20	30	70
c_j	2000	2500	800	0

(a) Find the optimal rotation schedule assuming that the cycle lengths of the two machines have to be the same. Compute the total average cost per unit time.

(b) Find the optimal rotation schedules of the two machines assuming the cycle lengths of the two machines do not have to be the same (determine which items have to be combined with one another on the same machine to obtain the best result). Compute the average cost per unit time and compare the result with the result found under (a).

7.3. Consider the following two stage production process in a paper mill with a downstream converting operation. At the first stage there is a single paper machine. The output of this operation consists of large rolls of paper. The second stage is a single machine cutting operation that produces cutsize paper. To simplify the problem assume that only two items have to be produced. Also, each item that comes out of the second stage corresponds to one of the items that comes out of the first stage. The production rates, setup costs and holding costs are different at the two stages. In the table below Q_{ij} denotes the production rate of item j at stage i, c_{ij} the setup cost of item j at stage i, and h_{ij} the holding cost of item j after processing at stage i (so h_{1j} denotes the holding cost of keeping item j in inventory in between the two stages, while h_{2j} denotes the holding cost of item j as a finished good).

items	1	2
D_j	100	50
Q_{1j}	400	400
Q_{2j}	600	1000
h_{1j}	20	20
h_{2j}	60	80
c_{1j}	3000	2500
c_{2j}	1000	1250

The schedules at both stages have to be rotation schedules (i.e., it is, for example, not allowed to produce item 1 at the first stage for a while, leave the machine idle for some time, produce item 1 again, and then item 2).

(a) Assuming that the cycle length x of the two stages have to be the same, what is the cycle length with the minimum total cost?

(b) Assume that the cycle lengths at the two stages are allowed to be different. Determine the optimal cycle lengths of the two stages.

7.4. Connsider the environment with two machines in parallel in Example 7.5.1. Suppose now that the production rate of one machine is .7 times the production rate of the machine in Example 7.3.1 and the production rate of the second machine .3 times.

a) Determine the optimal rotation schedules when both machines must have the same cycle time.

b) Determine the optimal rotation schedules when the two machines do not have to have the same cycle times.

7.5. Consider the data in Example 7.3.1. Consider now m identical machines in parallel. The production rate of item j on any one of the machines is Q_j/m. (This implies that the total production capacity does not depend on the number of machines in parallel.) Assume that all the machines have to be scheduled according to rotation schedules with the same cycle time x. Compute the optimal x and the total cost for $m = 2, 3, 4$. Plot the total cost against m.

7.6. Consider m identical facilities with each facility having a production rate Q_j/m. There are no setup times. Assume that all the facilities have to follow rotation schedules with the same cycle length x. Derive an expression for the optimal cycle length x. How does x depend on m? Discuss the monotonicity and the convexity of the function.

7.7. Consider a paper mill with two paper machines. There are 5 different types of paper that have to be produced. Items 1 and 2 have to be produced on machine 1 and item 3 has to be produced on machine 2. Items 4 and 5 can be produced on either one of the two machines.

items	1	2	3	4	5
D_j	60	60	80	80	100
Q_j	200	200	300	300	400
h_j	20	30	40	20	20
c_j	3000	2000	800	4000	1500

(a) Determine the optimal rotation schedule assuming that the cycle lengths have to be the same.

(b) Determine the optimal rotation schedules assuming the cycle lengths do not have to be the same.

(c) Determine the optimal schedule if the schedule on machine 1 has to be a rotation schedule and the schedule on machine 2 may be an arbitrary schedule.

7.8. Consider the following generalization of the paper making facility of Exercise 7.3. Again, there are two stages. The entire facility produces three different items. However, the paper machine at the first stage produces only two different intermediate products. One of the intermediate products that comes out of stage 1 is used at stage 2 to produce items 1 and 2. (This implies that the data correponding to items 1 and 2 regarding stage 1 are the same.) The other intermediate product that comes out of stage 1 is converted at stage 2 into item 3.

items	1	2	3
D_j	50	70	100
Q_{1j}	250	250	300
Q_{2j}	200	400	300
h_{1j}	30	30	40
h_{2j}	40	20	30
c_{1j}	2000	2000	900
c_{2j}	1000	3000	2000

Determine the optimal rotation schedule and its cycle length.

7.9. Consider the setting in Exercise 7.2. Instead of a single stage with two machines in parallel, we have now two stages in series with two machines in parallel at each stage. All four machines are identical with regard to production rates and setup costs (the data being the same as in Exercise 7.2). Determine the optimal rotation schedules of the four machines assuming that the cycle times of the four machines are the same.

7.10. Consider the setting in Example 7.4.2.

(a) What is the minimum cycle time when the frequency values are $y'_1 = y'_2 = 1$ and $y'_3 = y'_4 = 2$? Compute the total average cost of this solution.

(b) What is the minimum cycle time when the frequency values are $y'_1 = y'_2 = 1$, $y'_3 = 2$ and $y'_4 = 4$? Compute the total average cost of this solution.

(c) Compare the results obtained under (a) and (b) with the total average cost obtained in Example 7.4.1. Explain your results.

Comments and References

Various books and monographs focus on lot sizing and scheduling; see, for example, Haase (1994), Brüggemann (1995), Kimms (1997), and Zipkin (2000). Several excellent survey papers cover this topic also in depth, see Graves (1981) and Drexl and Kimms (1997).

The material in Section 2 is very basic. The EOQ formula was first mentioned by Harris (1915) and studied in detail by Wilson (1934). This material is covered in every elementary textbook on production planning and operations management.

Maxwell (1964) did an exhaustive study of rotation schedules. Gallego (1988) and Gallego and Roundy (1992) generalized these results allowing for backorder costs. Jones and Inman (1989, 1996) made an in depth study of the worst case behavior of rotation schedules and compared rotation schedules with other types of schedules. Gallego and Queyranne (1995) extended some of these results.

The FFS heuristic, described in Section 7.4, for generating arbitrary schedules is due to Dobson (1987, 1992). Gallego and Shaw (1997) established the NP-hardness of the ELSP with arbitrary cyclic schedules.

A fair amount of work has been done on lot scheduling in more complicated machine environments; see Crowston, Wagner and Williams (1973), Carreño (1990), Brüggemann (1995), Jones and Inman (1996), Chao and Pinedo (1996), and Pinedo and Chao (1999).

The production planning and scheduling system developed for Owens-Corning Fiberglas is discussed in Oliff and Burch (1985).

http://www.springer.com/978-1-4419-0909-1

Planning and Scheduling in Manufacturing and Services
Pinedo, M.L.
2009, XVIII, 536 p., Hardcover
ISBN: 978-1-4419-0909-1