Chapter 6
Time-Varying Lead Times and Iterative Multi-Model Approaches

The planning models in the previous chapter assume the planned lead times to be workload-independent, exogenous parameters that remain constant over the entire planning horizon. We now consider models with exogenous lead times that vary over time, seeking to accommodate time-varying levels of resource utilization. Since, as discussed in Chap. 2, cycle times depend on capacity utilization, which is determined by release decisions, obtaining time-varying estimates of lead time parameters requires observation or prediction of resource utilization across the time periods in the planning horizon. This tight linkage of utilization and cycle time suggests that releases and lead times should be jointly determined, i.e., the lead times should be endogenous to the model.

We begin this chapter with formulations based on exogenous, time-varying lead times, discuss the issues that arise in estimating these parameters, and then describe order release models that treat time-varying lead times as decision variables linked to the order releases. Noting that many of these formulations result in non-convex optimization models, we then discuss a class of iterative multi-model approaches that have been proposed in the literature.

6.1 Preliminaries

It is important to distinguish the problem addressed in this chapter, that of estimating planned lead times to represent cycle times that vary over time, from that of updating existing lead time estimates as new information becomes available from the market and the shop floor. The lead time parameters of MRP systems are reviewed relatively infrequently in practice (Jonsson and Matsson 2006), but must be updated periodically as the production system and its products evolve over time. In this chapter we consider time-varying lead time parameters within a single planning run, so this line of research is not directly relevant. Time-varying lead times are also of interest for due-date assignment (Ioannou and Dimitriou 2012), since the

H. Missbauer, R. Uzsoy, *Production Planning with Capacitated Resources and Congestion*, https://doi.org/10.1007/978-1-0716-0354-3_6

state of the shop at the time an order is placed will impact its planned finish date. This is again somewhat different from our problem since we use lead times as input parameters to an order release model that determines the release dates for all orders simultaneously, as opposed to predicting the cycle time of a particular order introduced into the shop at a particular time.

Flow factors or flow allowances, which estimate the lead time associated with an order at a workcenter as a multiple of its processing time, have been widely used for estimating lead times (Keskinocak and Tayur 2004). This approach appears to have originated in the literature on due date setting for make-to-order shops (Keskinocak and Tayur 2004) and has since been widely used in production planning and scheduling. Morton and Pentico (1993) extend this concept to suggest a load-dependent proportionality factor that can be estimated from historical data for lightly, moderately, and heavily loaded shops (p. 218), or by regression from historical data. However, at high levels of resource utilization, cycle times will consist mainly of waiting time in the queues, rendering a proportional relationship between processing and cycle time common to all orders in the shop unlikely except under specific conditions, such as lot sizes that depend strongly on resource utilization, or a sequencing rule that prioritizes jobs with short operation times. Ozturk et al. (2006) apply data mining based on regression tree techniques to this problem.

In Sects. 6.1–6.3 we discuss the representation and modeling of time-varying lead times. Sections 6.4–6.5 then present improved methods to adjust the lead times to the order release plan. In particular, Sect. 6.5 presents methods for iterative adjustment of order releases and time-varying lead times, an approach that has also been proposed for other production planning problems as discussed in Sect. 6.6.

6.2 Relaxing the Fixed Lead Time Constraint: Conceptual Issues

In discrete manufacturing systems, the cycle time of production orders at bottleneck resources consists mainly of waiting time in the queues and usually follows a probability distribution with substantial variance. The moments of this distribution, notably its mean, are highly nonlinear functions of the resource utilization as shown in Chap. 2. Planned lead times, which are parameters of the planning system, are derived from these cycle times or their distribution. In MRP, this is accomplished by treating the cycle times as a quantity to be forecast or predicted. In most order release models, planned lead times are obtained by specifying target lead times and controlling the WIP level to ensure that observed cycle times are consistent with these targets via Little's Law. The definition of "consistent" depends on how cycle time uncertainty is handled in the planning system—this uncertainty is (hopefully) reduced, but not eliminated by load-based order release. If the estimated average cycle time is used as the planned lead time, safety stock or a downstream time buffer can help to manage the uncertainty. The alternative is safety lead time achieved, for example, by setting planned lead times equal to the historical mean cycle time plus

a specified safety lead time (Hopp and Sturgis 2000). This approach amounts to setting the planned lead time to some percentile of the underlying cycle time distribution. A number of authors have addressed the problem of determining optimal lead times for different production and inventory systems, including Ben-Daya and Raouf (1994) for inventory systems and Gong et al. (1994) and Milne et al. (2015) for MRP systems.

If the relationship between the cycle time distribution and lead times can be specified, lead time parameters that remain constant over time can be consistent with the steady-state behavior of the production unit. When the aggregate demand faced by the production unit, and hence the average utilization of its bottleneck resources, exhibit little variation over time, this approach is likely to be quite satisfactory. However, if demand varies widely over time, even if the release model has some load-leveling capability, the releases, and thus the work input to the resources and their utilization, may also vary over time, and the constant lead times will not match the actual cycle times. This issue can arise due to both the total demand for all products varying over time and the time-varying demand for individual products with different production routings and resource requirements.

Inconsistency between constant lead times and load-dependent cycle time distributions causes two distinct difficulties. On the one hand, the lead times must allow high bottleneck utilization, which requires high WIP levels, high average cycle times, and thus a high value of the planned lead time. A temporary decrease in demand will lead to reduced releases, work input, and resource utilization, resulting in shorter cycle times. Material will be released earlier than is necessary to meet demand, causing unnecessarily high FGI levels. On the other hand, temporarily increasing releases, and hence WIP levels between workcenters, to improve load smoothing is not possible since this would raise realized lead times above the planned lead time. Directly addressing the latter issue within the release model requires estimates of lead times in the face of time-varying demand, either through a separate planning module that estimates the lead time parameters to be used in the release model or within the release procedure itself. The latter requires representation of time-varying lead times in the order release model, either explicitly as decision variables or implicitly as time-varying WIP, leading to additional complications discussed in the next two chapters.

To illustrate the issues that arise when considering time-varying lead times, consider the following example.

Example 6.1 The fixed lead times associated with the orders released in each period are given in Table 6.1 for 12 consecutive planning periods. We make no pretense that these lead times are realistic in any way; our purpose is to illustrate the issues that arise in selecting time-varying lead time estimates. The reader will note that the lead times increase and decrease by substantial jumps, with some being fractional and others integer.

Table 6.2 shows the loading factors that represent the fraction of material released in period τ that will emerge in period t based on these lead time estimates. The lower diagonal is, as expected, empty since a positive entry in this area would imply a

Table 6.1 Lead time parameters for Example 6.1

Period	0	1	2	3	4	5	6	7	8	9	10	11	12
Lead time	1	2	2.5	2.5	3	4.5	3	2.5	2.5	2	1.5	1	1

Table 6.2 Loading fractions for Example 6.1

Period	1	2	3	4	5	6	7	8	9	10	11	13	13
0	1	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0.5	0.5	0	0	0	0	0	0	0	0
3	0	0	0	0	0.5	0.5	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0.5	0.5	0	0	0
6	0	0	0	0	0	0	0	0	1	0	0	0	0
7	0	0	0	0	0	0	0	0	0.5	0.5	0	0	0
8	0	0	0	0	0	0	0	0	0	0.5	0.5	0	0
9	0	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0.5	0.5	0
11	0	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0	0	0	0	0	0	0	0	1

negative lead time, which, although it might be welcomed by many manufacturing managers, is difficult to achieve. There are, however, several areas of interest above the diagonal. No output emerges at all in period 8, due to the long lead times in preceding periods. Material released in period 4 emerges in period 7, but material released in period 5 emerges in periods 9 and 10. All materials released in period 6 emerge in period 9. However, half the material released in period 5 emerges in period 9 and the other half in period 10, indicating that the releases from period 5 are being overtaken by those from period 6.

This example illustrates that unless time-varying lead times are selected with some care, they can lead to quite unrealistic behavior in a planning model. It is thus useful to seek conditions to impose on lead time estimates that will ensure reasonable behavior of the planning models in which they are deployed. One such requirement would seem to be that of no-passing, or first-in-first-out (FIFO): material released in earlier periods should not emerge from the system before material that is released later. In the dynamic traffic assignment literature (Peeta and Ziliaskopoulos 2001), this implies no overtaking: vehicles entering a road segment at a point in time cannot exit before those entering earlier.

Carey (1992) examines several such conditions in the context of the dynamic traffic assignment problem, focusing on the need to preserve the FIFO property, in our context to ensure that material that is released earlier does not emerge after material released later. He first considers the case of a single product where x_{ts} represents the amount of the product arriving at the resource in period t and completing its processing in period s. Thus the amount of material x_{ts} will remain at the resource for $(s - t)$ periods, and the average time a unit of work arriving in period t will spend at the resource will be given by

$$\bar{m}_t = \frac{\sum_{\tau \geq t} x_{t\tau} \left(\tau - t\right)}{\sum_{\tau \geq t} x_{t\tau}} \tag{6.1}$$

Note that since it represents an average, the value of $\bar{m}_t$ need not be an integer. To maintain FIFO on the basis of the average flows, material that arrives at the resource in period t must exit by period $t + \bar{m}_t$. Thus, to ensure that on average material arriving later exits later, material entering in a later period $s > t$ must exit in period $s + \bar{m}_s$. Thus to preserve FIFO on average, we must have

$$\bar{m}_t \leq \bar{m}_s + \left(s - t\right), \quad \text{for} \quad s \geq t \tag{6.2}$$

yielding

$$t + \bar{m}_t \leq s + \bar{m}_s, \quad \text{for all} \quad s \geq t \tag{6.3}$$

Since $(s - t) \geq 1$, this implies that the constraints

$$\bar{m}_t \leq \bar{m}_{t+1} + 1 \tag{6.4}$$

are necessary and sufficient to ensure FIFO for the average flows, although, as he shows by a counterexample, necessary but not sufficient for the individual components x_{ts}. Note that in this representation, the planned lead times are not represented explicitly as a parameter, but through the definition of the decision variables x_{ts}, with $\bar{m}_t$ defined as in (6.1). The explicit inclusion of condition (6.4) in an optimization formulation with a planning horizon of T periods requires $O(T^2)$ non-convex constraints, resulting in a model that is significantly more difficult to solve. He goes on to show that analogous conditions are necessary and sufficient for the average flow in the presence of multiple vehicle classes, analogous to multiple products in our context, and necessary but not sufficient to maintain FIFO at the level of individual items. This necessary condition plays an important role in the formulation of the Allocated Clearing Function model in Chap. 7 and will be revisited in that context. However, Carey's findings are, in general, discouraging: they show that a variety of approaches to maintain the FIFO property all lead to planning models with non-convex feasible regions.

6.3 Modeling Time-Varying Lead Times

We can distinguish two different types of planned lead times for a single workcenter using a continuous representation of time and orders as seen in Fig. 6.1. The *forward* lead time $L^{\mathrm{f}}(t)$ represents the lead time of an order that arrives at time t, i.e., the estimated time spent in the workcenter by an order arriving at time t. Similarly, the *backward* lead time $L^{\mathrm{b}}(t)$ represents the planned amount of time spent in the

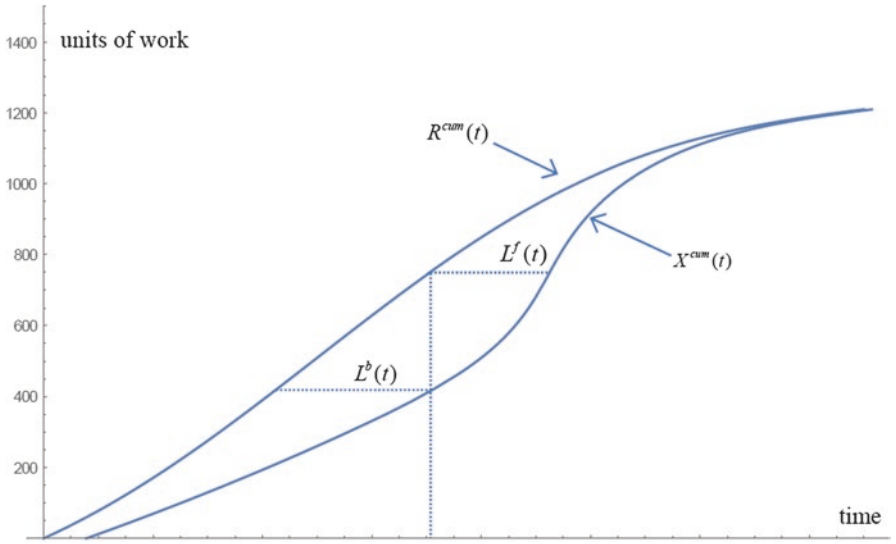

Fig. 6.1 Evolution of forward and backward lead time over time

workcenter by an order leaving the workcenter at time t. In other words, a unit of work arriving at the workcenter at time t departs at time $t + L^{\mathrm{f}}(t)$, while one departing at time t must have arrived at time $t - L^{\mathrm{b}}(t)$.

Following our previous notation, let $R(t)$ denote the rate of material release into the workcenter at time t, and $X(t)$ its output rate at time t. We shall denote the cumulative releases and output up to t by $R^{\mathrm{cum}}(t)$ and $X^{\mathrm{cum}}(t)$, respectively, and let $W(t)$ denote the planned WIP level at time t. If we require the FIFO or no-passing property, under which work released at time t cannot complete before work released at any time $s < t$ and production orders (or work particles in the continuous representation) depart the workcenter in the same sequence as they arrive, the cycle times are determined by the evolution of WIP over time. Based on Fig. 6.1, we have the material balance relations

$$W(0) + \int_0^t R(\tau)\,d\tau = \int_0^{t+L^{\mathrm{f}}(t)} X(\tau)\,d\tau \tag{6.5}$$

$$W(0) + \int_0^{t-L^{\mathrm{b}}(t)} R(\tau)\,d\tau = \int_0^t X(\tau)\,d\tau \tag{6.6}$$

which calculate the time-dependent output of the workcenter from its time-dependent input, constituting a *dynamic production function* (Hackman 2008). Equation (6.5) states that all materials entering the system by time t must, by the definition of $L^{\mathrm{f}}(t)$, have been converted into output by time $t + L^{\mathrm{f}}(t)$. Similarly, (6.6)

states that all materials leaving the system by time t must have entered by time $t - L^{b}(t)$. Hence $L^{f}(t)$ and $L^{b}(t)$ are related as

$$\begin{aligned} L^{b}(t) &= L^{f}\left(t - L^{b}(t)\right) \\ L^{f}(t) &= L^{b}\left(t + L^{f}(t)\right) \end{aligned} \tag{6.7}$$

Extending this logic to discrete-time models is not straightforward. The simplest analogy is period-based, integer lead times L_t^{f} and L_t^{b} representing the lead times of orders arriving or departing in period t, respectively. Thus the fixed lead time formulation in Chap. 5 represents a backward lead time implying

$$R_{t-L_t^{b}} = X_t \tag{6.8}$$

This is perfectly adequate when $L_t^{b} = L_{t+1}^{b}$ for all periods $t = 1,..., T - 1$ in the planning horizon; each unit of work emerging as output at any time within period t was released exactly L_t^{b} time units earlier. However, if the planned lead time at the workcenter increases by 1 period from period t to period t+1 such that $L_{t+1}^{b} = L_t^{b} + 1$, (6.8) implies that the output of two or more consecutive periods was released in the same period, as was the case for period 9 in Example 5.1. Hence, (6.8) must be formulated as an inequality constraint of the form

$$\sum_{k=1}^{t-L_t^{b}} R_k \geq \sum_{k=1}^{t} X_k \tag{6.9}$$

for all t, which only gives a lower bound on the releases or, expressed in terms of time, the latest possible release period for given output over time. Thus it represents time-varying lead time parameters only in the context of a release model that delays releases as much as possible, usually due to positive WIP holding costs in the objective function (as in the release models in Chap. 5). This is the first shortcoming of representing lead times directly as parameters L_t^{f} or L_t^{b}.

A second problem is that this representation cannot express lead time distributions. Empirical cycle time distributions often exhibit high coefficients of variation as seen in Fig. 2.3, and an effective planning model should be able to represent this. One approach to representing lead time distributions is the use of *loading factors* $w_{\tau t}$ defined as the fraction of the work released in period τ that emerges as output in period t.

A backward lead time L_t^{b} can be converted into a loading factor by noting that

$$w_{\tau t} = \begin{cases} 1, & \text{if } L_t^{b} = t - \tau \\ 0, & \text{otherwise} \end{cases} \tag{6.10}$$

yielding the relationship between releases and output as

$$R_t = \sum_{\tau=t}^{T} X_\tau w_{t\tau} \tag{6.11}$$

The loading factors can be interpreted as the expected fraction of work released in a certain period that leaves the workcenter after a certain time, representing a discrete probability distribution for lead times. In (6.11), and most of the iterative approaches discussed below, this expectation is treated as a deterministic fraction, resulting in a set of linear constraints.

6.4 Epoch-Based Lead Times

Until now we have assumed period-based lead times such that lead times are associated with specified planning periods, implying that that all releases (for forward lead times) or output (for backward lead times) associated with a period is subject to the same lead time. Hung and Leachman (1996) suggest the use of epoch-based lead times defined at the period boundaries, which permits a more general representation of fractional lead times. We now describe this approach since it forms the basis for many of the iterative approaches in Sect. 6.5.

The basic formulation is derived from the model discussed in Chap. 5, which requires lead time estimates L_{jk} representing the time required for a unit of product j to reach the k'th resource in its product routing after being released into the plant. However, instead of fixed lead times that remain constant over the entire planning horizon, Hung and Leachman (1996) associate lead time parameters with the start of each planning period. In the following we shall assume unit-length planning periods such that period t starts at time $t-1$, i.e., $t=0$ is the start of period 1, $t=1$ the start of period 2, etc. Equivalently, this can be viewed as period t ending at time t. The lead time parameters L_{jkt}, which may take on fractional values, represent the lead time after its release required for an order of product j to reach the k'th resource on its routing if the order reaches that resource at the end of period t. This definition of epoch-based lead time parameters is depicted in Fig. 6.2. The key assumption is that the releases associated with a planning period take place at a uniform rate over the planning period, as discussed in Chap. 5 and in Hackman and Leachman (1989).

Given these lead times, the loading of the production resource in period t is defined by releases occurring in the time interval $Q_t = [(t-1) - L_{j,k,t-1}, t - L_{jkt}]$, recalling that planning period t starts at time $(t-1)$ and ends at time t. There are two cases to consider here. In the first, simpler case, the time interval Q_t lies within a single planning period $\lceil (t-1) - L_{j,k,t-1} \rceil = \lceil t - L_{jkt} \rceil$ where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x. In this case the entire amount released in period $\lceil (t-1) - L_{j,k,t-1} \rceil$ arrives at workcenter k in period t. Hence the amount Y_{jkt} of product j loading workcenter k in period t is given by

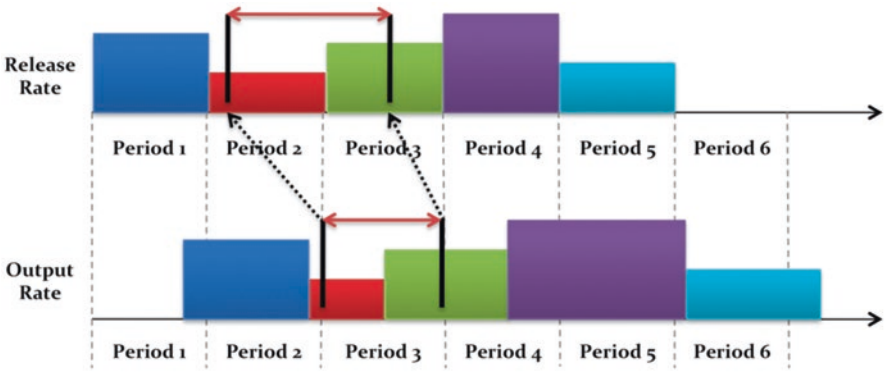

Fig. 6.2 Relationship between releases and loading with time-dependent lead times. Adapted from Hung and Leachman (1996)

$$Y_{jkt} = \left(\frac{\left(t - L_{jkt}\right) - \left(\left(t-1\right) - L_{j,k,t-1}\right)}{\Delta} \right) e_{jk} R_{j,\lceil (t-1) - L_{j,k,t-1} \rceil} \tag{6.12}$$

where Δ denotes the period length (set to 1 by definition) and e_{ij} the average fraction of the release quantity of product j that will arrive at resource k.

If, on the other hand, the time interval Q_t spans multiple planning periods, we allocate the load due to releases in that period in proportion to the fraction of that period's total duration included in the interval Q_t assuming uniform release rates over the planning periods, yielding

$$\begin{aligned} Y_{jkt} = & \left(\frac{\left(\left\lceil \left(t-1\right) - L_{j,k,t-1} \right\rceil - \left(t-1\right) - L_{j,k,t-1}\right)}{\Delta} \right) e_{jk} R_{j,\lceil (t-1) - L_{j,k,t-1} \rceil} \\ & + \sum_{\tau = \lceil (t-1) - L_{j,k,t-1} \rceil + 1}^{\lceil t - L_{jkt} - 1 \rceil} e_{jk} R_{j\tau} \\ & + \left(\frac{\left(t - L_{jkt}\right) - \left\lceil t - L_{jkt} - 1 \right\rceil}{\Delta} \right) e_{jkt} R_{j,\lceil t - L_{jkt} \rceil} \end{aligned} \tag{6.13}$$

The operation of this approach is illustrated in Fig. 6.2. The upper part of the figure shows the uniform release rates in each planning period, and the lower portion the resource loading resulting from these releases arriving at the resource after the specified fixed lead times. Releases in periods 2 and 3 contribute to the work input in period 3 at resource k corresponding to the first and the third term in (6.13); the second term is not relevant here because the release interval only spans the two periods 2 and 3. Due to the use of backward lead times, the lead times are associated with the boundary points between periods at the workcenter, not those between the release periods, and hence the lead time at the start of a period may not be the same

as that at the end. The coloring indicates the correspondence between the releases and the arrival of the material at the resource.

The loading factors $w_{j\tau t}$ that denote the fraction of releases of product j in period τ that contribute to output in period t follow immediately from (6.12) or (6.13), depending on the case. The amount Y_{jt} of product j arriving at the workcenter in period t is, analogously to (6.11), given by the linear expression

$$Y_{jt} = \sum_{\tau=1}^{t} R_{j\tau} w_{j\tau t} \tag{6.14}$$

If we could obtain the correct values of the loading factors $w_{j\tau t}$ efficiently, we would no longer need an explicit capacity constraint since the loading factors would reflect the ability of the resource to produce output over time. However, formulating and solving a model that encompasses both order release planning and estimation of the $w_{j\tau t}$ values turns out to be challenging, as we discuss below.

6.5 Lead Time Estimation Within the Order Release Procedure

The previous section described different ways to represent time-varying lead times in an order release model, assuming these were treated as exogenous parameters. We now turn to the crucial question of how to specify values for these lead times, that is, how to represent the functional relationship between capacity loading and lead times. This can be handled in two fundamentally different ways:

- Time-varying lead times can be treated as exogenous parameters whose values are determined based on information known prior to order release, such as historical flow times, capacity, and demand. Orders are then released based on these lead time parameters.
- The order release model, or the order release procedure of which it is a part, can treat the lead times as functionally related to the release schedule and hence must represent this functional relationship. This can, in turn, be accomplished in two ways:
  - The lead times can be defined as decision variables *endogenous* to the optimization model, which optimizes releases and lead times simultaneously.
  - The problem can be decomposed into two related subproblems: one that determines an optimal release schedule given the estimates of lead times, and another that estimates lead times based on a given release schedule. An iterative procedure then solves these subproblems in sequence until some convergence criterion is satisfied.

The first approach, that of setting lead times prior to order release, has been treated extensively in the MRP literature. “MRP treats lead times as attributes of the part and possibly the job, but *not* of the status of the shop floor” (Hopp and Spearman 2008: 124). Planned lead times “serve as a proxy for dealing with capacity constraints; a

longer planned lead time leads to a longer planned queue that permits more production smoothing" (Graves 2011: 93). As described in Chap. 2, using this approach, lead times are usually estimated from historical cycle times and are updated only infrequently. Due to the importance of planned lead times for manufacturing performance, there has been extensive research on improving lead time estimation (Milne et al. 2015). Since the role of the lead times in this framework is to coordinate the various planning levels of the PPC system, their values are determined by a parameter-setting function that seeks to ensure this coordination (see Chap. 1). Since the cycle times are closely related to capacity utilization, they must be coordinated with the production smoothing decisions made at the master production scheduling level, but we are not aware of any research on jointly determining the master production schedule and time-dependent lead times.

The principal difficulty in determining time-varying lead times derives from the nonlinear relation between cycle time and resource utilization described in Chap. 2. Since cycle times depend on resource utilization, and resource utilization on the release decisions, determining time-varying lead times for use in an order release model requires knowledge of capacity loading over time, and hence of the order releases, at least at an aggregate level. If the aggregate level of capacity loading, and hence resource utilization, remains largely constant over time, this may not be a major issue. However, even this may be moot at high utilization levels, where small changes in resource utilization may lead to large changes in cycle times. Directly addressing this interdependence between lead time estimates and release decisions requires models that simultaneously determine time-varying lead times and order releases, which we discuss in the following section.

6.5.1 Models Without WIP Evolution

The evolution of cycle times over time is closely related to the evolution of WIP over time as expressed in (6.5) and (6.6); Hackman (2008: 309ff.) gives a more detailed discussion. For complex manufacturing systems modeled as networks of queues that need not be in steady state, models that accurately anticipate the evolution of WIP and cycle time over time generally involve some type of simulation, either of the discrete-event type (Law and Kelton 2004) or continuous-time models based on ordinary or partial differential equations (Armbruster and Uzsoy 2012), which are difficult to incorporate into a tractable mathematical programming model. Therefore, a number of approaches that estimate load-dependent lead times within the order release model without explicitly considering the evolution of WIP over time have been proposed.

Since in steady state the average cycle time increases nonlinearly with utilization, it is intuitively appealing that this pattern also holds within each planning period of an order release model. A number of authors have developed models that select an appropriate lead time for each planning period based on the resource loading in that period. These are closely related to those developed for dynamic traffic

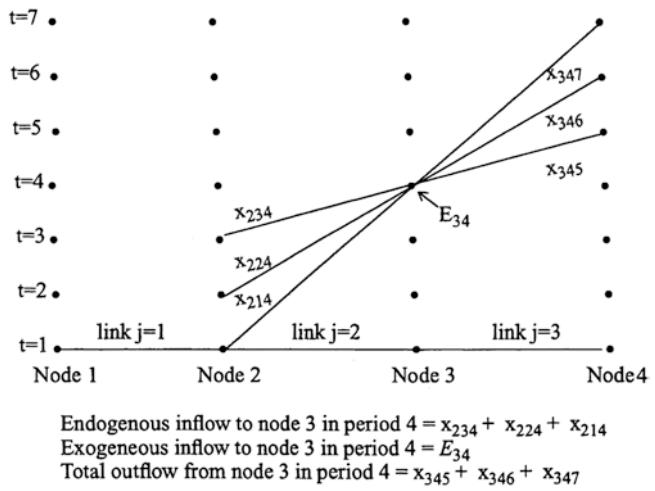

Fig. 6.3 Conservation of flows on a time-expanded network (Carey and Subrahmanian 2000)

assignment models (Peeta and Ziliaskopoulos 2001) that seek to determine the routing of vehicles through a road network to optimize some measure of performance. Since individual traffic links (road segments) are subject to congestion, considerable effort has been devoted to developing models that capture the relationship between the volume of flow on a traffic link and the velocity of that flow.

One way to model congestion in traffic networks is through the use of *time–space links* (Carey and Subrahmanian 2000). If two nodes i and j of a traffic network are connected by a spatial link (in other words, a road segment), this two-node network can be expanded over time to yield a network of time–space nodes as seen in Fig. 6.3. The flow on a time–space link represents the number of vehicles that pass the nodes at the times corresponding to the nodes at the end points of the link and hence requires the associated (integer) traversal time.

The impact of congestion is manifested as a link traversal time that increases with the volume of flow on the link and can be represented by having the upper bounds on the flow through the time–space links leaving node i at time t depend on the flow through node i at time t, i.e., the inflow to the time–space links leaving node (i,t). In the model of Carey and Subrahmanian (2000), the capacities of at most two neighboring time–space links leaving node (i,t) are positive and the other time–space links are closed for the given inflow. As the inflow increases, the time–space links with positive capacities move to higher traversal times, implying a flow-dependent traversal time distribution that is stationary over time for a given inflow. The relationship between the flow x through the time–space link and the time s it takes to traverse the link, referred to as the travel time function $s = f(x)$, is assumed to be convex, increasing, and piecewise linear, which allows the breakpoints of the function to be mapped onto the time–space links as in Fig. 6.4. If the inflow x is exactly at a breakpoint, only the corresponding time–space link is active. Otherwise

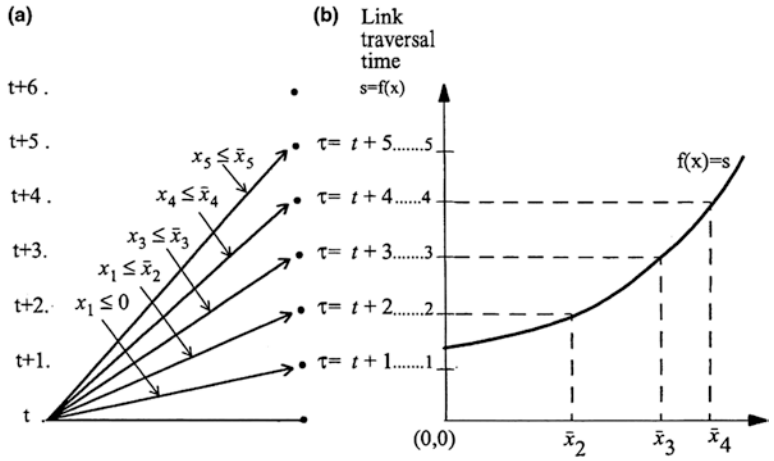

Fig. 6.4 Relationship between time–space link capacities (left) and travel time function (right) (Carey and Subrahmanian 2000)

the respective fractions of the inflow are assigned to the two adjacent time–space links (Carey and Subrahmanian 2000: 163). The authors consider a piecewise linear convex objective function and develop two alternative formulations based on this representation. Using concepts from separable programming (Bazaraa et al. 1979), they show that under these assumptions at most two adjacent time–space links will carry positive flow. They also show that as long as there is no holding back behavior, where traffic that has entered a link is not allowed to exit in order to alleviate congestion in later periods, the solutions will satisfy FIFO unless a sharp increase in inflow is followed by a sharp decrease. When holding back occurs, however, solutions may violate the FIFO property.

The traversal time of a spatial link in a traffic network is analogous to the cycle time at a workcenter, and models with similar structure have been developed for order release planning in manufacturing. Voss and Woodruff (2003) assume a steady-state relationship between workcenter utilization and the expected cycle time at that workcenter. They then discretize this curve using integer variables to ensure that only one segment of the discretized curve is active in a given time period. The relationship between utilization and expected lead time is evaluated at discrete utilization levels (breakpoints) BP_q, $q = 1,\ldots,U$ where L^q denotes the expected lead time value associated with the q'th utilization level BP_q. Thus the expected lead time of the resource is assumed to be L^q when its utilization level is between BP_q and BP_{q-1}. The authors suggest setting the breakpoints BP_q such that each lead time L^q corresponds to an integer number of periods. If a_j denotes the fraction of the available resource capacity required for one unit of product j, $j = 1,\ldots,P$, and R_{jt} the amount of product j released in period t, the utilization of the resource in period t is given by

$$u_t = \sum_{j=1}^{P} a_j R_{jt} \tag{6.15}$$

We now define binary variables y_{tq} that select a particular lead time value L^q to be active in a given period t as follows:

$$L_t = \sum_{q=1}^{U} y_{tq} L^q \text{ for all } t \tag{6.16}$$

$$\sum_{q=1}^{U} y_{tq} = 1, \text{for all } t \tag{6.17}$$

Additional constraints of the form

$$\sum_{q=1}^{U} BP_q y_{tq} \geq \sum_{j=1}^{P} a_j R_{jt}, \text{for all } t \tag{6.18}$$

ensure that the lead time selected is consistent with the workload. In addition, for any given period t, we require $L_t - L_{t+1} \leq 1$, giving

$$\sum_{q=1}^{U} y_{tq} L^q - \sum_{q=1}^{U} y_{t+1,q} L^q \leq 1, \quad \text{for all} \quad t \tag{6.19}$$

This latter constraint is interesting in that it restricts the changes in lead time from one period to the next to at most one period to avoid overtaking, i.e., material released into the system at a later time emerging before material released earlier. Note that (6.19) enforces the condition (6.4) shown by Carey (1992) to be necessary for the flow through a node to satisfy the first-in-first-out (FIFO) condition.

To complete the formulation, the authors present an objective function that includes an explicit holding cost for WIP, based on Little's Law (Hopp and Spearman 2008), leading to

$$\min \sum_{t=1}^{T} \sum_{j=1}^{P} h_{jt} \sum_{q=1}^{U} y_{tq} L^q R_{jt} \tag{6.20}$$

This objective function is nonlinear due to the product of the y_{tr} and R_{jt}, leading to a formulation that is computationally hard to solve.

Lautenschläger (1999) describes a similar approach. In order to consider load-dependent lead times for master production scheduling, this model determines the fraction of the planned production available in a period t that has to be started one period ahead in period $(t-1)$ assuming the rest is produced in period t. This fraction is a function of the planned utilization. Thus production on a resource can be performed in two modes, one with lead time of zero periods and the other with lead time of one period, essentially the same idea as the time-expanded network in Fig. 6.3. The maximum production volumes that can be realized in each mode are

limited, leading to a utilization-dependent lead time distribution. Short-term oscillations in capacity utilization over time, which are considered undesirable due to considerations not explicitly represented in the model, are reduced by a low-pass filter (Lautenschläger 1999: 114ff.). Many factory managers consider large variations in utilization to be detrimental to performance, perhaps due to their impact on staffing and other support services such as material procurement (Lautenschläger 1999: 114f). However, the high-frequency oscillations may also be due to the simplifications in the flow time modeling. Orcun and Uzsoy (2011) have shown that inconsistencies between the lead times used in a planning model and the cycle times in the production system can lead to significant oscillating behavior when the planning model is implemented in a rolling horizon environment, supporting the latter conclusion.

6.5.2 *Critique of Lead Time Estimation Without WIP Evolution*

While the models in the previous section address the load-dependent nature of lead times, they ignore the relationship between time-dependent lead times and WIP evolution over time expressed in (6.5) and (6.6) and formulated more generally in transient versions of Little's Law (Bertsimas and Mourtzinou 1997; Riaño 2003) in order to obtain a tractable mathematical programming formulation. As such, they must be viewed as approximations that exhibit several shortcomings:

- All the models described above assume a well-defined relationship between the workload or utilization of a resource in a planning period and its expected cycle time in that period. The form of this relationship is generally posited assuming steady-state is reached by all related queues during the planning period. However, since planning models assume discrete planning periods of a fixed length and work releases vary over time, planning models inherently operate in a transient regime, and the cycle time of work released in a given period may deviate quite substantially from the long-run steady-state average.
- If the amount of work released decreases sharply from period t to period t+1, the estimated lead time for the orders can decrease by more than one period from t to t+1, implying overtaking (Voss and Woodruff 2003: 165; Carey and Subrahmanian (2000)). This is unlikely to occur in practice—although it may be accomplished to a limited extent by expediting, which has its own disadvantages (Ehteshami et al. 1992; Narahari and Khan 1997)—and violates the assumption that the released work must be processed first-in-first-out. This suggests that these models can lead to unrealistic results. Voss and Woodruff (2003) add a constraint that keeps the lead time from decreasing by more than one period from t to t+1, which Carey (1992) has shown is a necessary condition for the preservation of the FIFO property.

Several researchers have sought to address these problems by using either a discrete-event simulation model or a transient queueing model to model the joint evolution of lead times and WIP levels. This leads to computationally intractable optimization models, requiring lead time estimation to be performed outside the optimization model. This approach will be discussed in the next section.

6.6 Lead Time Estimation Outside the Optimization Model: Iterative Multi-Model Approaches

6.6.1 Overview

Modeling the joint evolution of lead times and WIP levels in a transient setting usually leads to computationally intractable order release models even in simple cases. This can be seen from (6.5) and (6.9) where the lead times are elements of the integration or summation limits. However, this structure can be addressed by decomposing the order release problem into two separate subproblems: one that computes a release plan given a set of time-varying lead time estimates and another that computes the expected lead times or output associated with each period, or boundary between periods, for a given release plan. These are usually deployed within an iterative framework that seeks convergence to a pair of consistent subproblem solutions. A review of multi-model approaches combining optimization and simulation is given by Figueira and Almada-Lobo (2014).

The central difficulty of multi-model approaches that decompose the release planning problem into separate release planning and lead time estimation problems is that of any decomposition procedure: that of efficiently achieving a solution simultaneously satisfying the constraints of both subproblems. In isolation, both subproblems can be addressed satisfactorily with well-known techniques. The release planning subproblem can be solved directly by the LP models described in Chap. 5, whose mathematical structure easily accommodates time-varying lead time estimates as long as reasonable estimates can be obtained as discussed in Sect. 6.1. The lead time estimation subproblem can be addressed by queueing or simulation models. What is required is a coordination mechanism that leads to mutually consistent solutions to the two subproblems that are at least feasible, and hopefully near-optimal, to the overall problem. In order to preserve the tractability of the release planning subproblem, its parameters (capacities and lead times) must be exogenous to whatever model is used to solve it, i.e., unaffected by the release schedule it produces. Similarly, the lead time/output estimation subproblem must treat the release schedule as an exogenous input. Hence these procedures combine mathematical programming and simulation or queueing models such that each model determines estimated values of parameters required by the other. Since the primary optimization mechanism is embedded in the mathematical programming model, the simulation or queueing model used for lead time estimation is subordinate

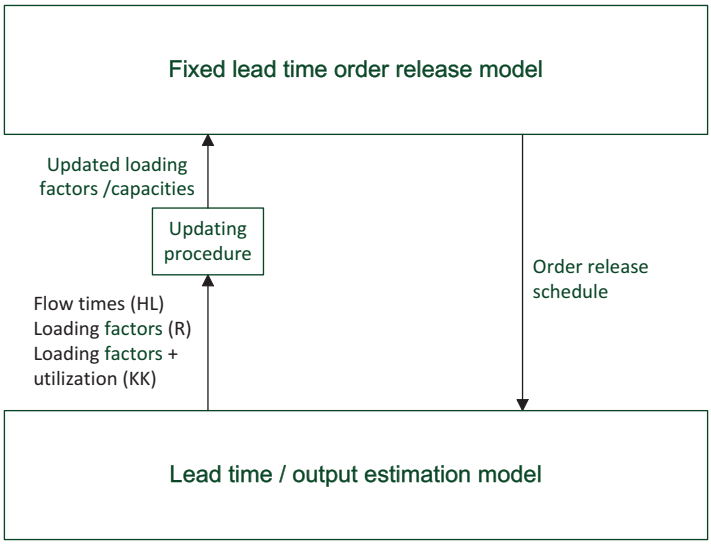

Fig. 6.5 Iterative simulation—LP approach for order release planning: the generic mechanism. HL: Hung/Leachman, KK: Kim/Kim, R: Riaño

to the optimization model per the taxonomy of hybrid simulation/analytic models by Shanthikumar and Sargent (1983). The procedure is outlined in Fig. 6.5.

How the parameters of each model are updated based on the results of the other is likely to have significant impact on both the convergence of the procedure and the quality of the solution to which it converges. The communication from the release planning model to the lead time estimation model is generally straightforward: the quantity of each product released in each period. The information passed from the lead time/output estimation model to the release planning model usually consists of the estimated mean cycle times associated with each period or epoch, while some approaches also consider average resource utilization levels in each period. The cycle times observed by the lead time estimation model may be represented in the release planning model as exogenous lead times or loading ratios as described in Sect. 6.2.

At each iteration, the current estimates of cycle times or loading ratios and utilization are used to update the lead times and capacities that constitute the parameters of the order release model, and the order release model is re-run. This iterative procedure is repeated until convergence, which can be defined as reaching a *fixed point* of the iterative mechanism, a solution where the parameters of the order release model lead to an order release schedule that results in the same cycle time and output estimates by which the order release schedule was produced. Thus, once the algorithm arrives at this solution, it remains there. Ideally, the optimal solution should be a fixed point, but there is as yet no rigorous proof that this is the case in general. There is considerable experimental evidence that the solution spaces of some formulations of this problem are non-convex, leading to the procedure

converging to different points from different initial solutions. Experimental evidence discussed later in the chapter suggests that even quite subtle differences in implementation may produce qualitative differences in computational behavior.

6.6.2 *Iterative Simulation-LP Algorithms*

Zaepfel (1984) was the first author to formulate such an iterative mechanism and its associated order release model. In his procedure only the estimated lead times are communicated from the lead time estimation (simulation) model to the order release model which assumes fixed lead times and unlimited capacities. The reasoning is that since overloading of capacities leads to higher flow times, information on capacity overload (excessive releases in certain periods) is captured in the revised lead time estimates in the feedback from the simulation model. No numerical results are provided.

Hung and Leachman (1996) were the first to provide numerical tests of this type of iterative scheme. Their order release model modifies the step-separated formulation of Leachman and Carmon (1992) to represent the lead times as loading factors per Sect. 6.2, with epoch-based backward lead times defined at the period boundaries as in Fig. 6.2. Updating these lead time parameters during the iterations requires observing the simulated flow times at the period boundaries, which are interpolated from the flow times of orders arriving at the workcenters immediately before and immediately after the boundary epoch (Hung and Leachman 1996: 262). The order release model includes capacity constraints and assumes that capacity is consumed at the end of the planned lead time. The release period determines the period in which the work is processed and capacity is required. Hung and Leachman (1996) examine the rate of convergence of the flow time estimates to the flow times observed in the simulation and find that convergence to the correct expected flow time values can be quite rapid but that the procedure can fail to converge in some cases which are not fully understood. Subsequent numerical tests by other authors (Irdem et al. 2010; Kacar et al. 2012) confirm that the convergence behavior of the general procedure is not well understood, as will be discussed further in Sect. 6.6.3.

Hung and Hou (2001) use the same basic procedure as Hung and Leachman (1996) but replace the simulation model with an analytical queueing model. The queueing model proceeds by dividing each planning period into a number of shorter subperiods and assumes steady-state behavior within the subperiods. The lead times applicable at the boundaries of the subperiods are obtained using the epoch-based lead time estimates obtained at a previous iteration. The *M/M/s* queueing model is used to predict average cycle times at individual workcenters, which are then composed into estimates of cycle times from the beginning of the process to each operation. They terminate the iterations when the percentage mean absolute deviation between the flow time estimates at successive iterations is sufficiently small. However, they find that especially at high utilization levels, the *M/M/s* queueing model predicts extremely high flow times, rendering the cycle time predictions

inaccurate. They also find that the method has difficulty in converging (specifically in Fig. 7 of Hung and Hou (2001)). Hence they develop an empirical approach that uses historical data to develop a model relating expected cycle times to workload at individual workcenters, similar to the function used by Voss and Woodruff (2003). They report short computation times and good convergence for longer sub-periods, but this issue is only described briefly.

Riaño (2003) proposes a rather different iterative technique in which loading factors w_{st} that describe the fraction of total releases in period s that will emerge by period $t \geq s$ are estimated using a transient model of a queueing network. To present the basic idea, we shall consider its application to a single-server workcenter; the extension to multiple stages and servers is discussed in Riaño et al. (2006). A job released to the workcenter at time s will see $Q(s)$ jobs ahead of it in the queue or in process. Hence the cycle time of that job will be given by

$$W(s) = \sum_{k=1}^{Q(s)} S_k + S \tag{6.21}$$

where S_k, $k = 2,\ldots,Q(s)$ denote the processing times of jobs ahead of this job in the queue, S_1 the residual (remaining) processing time of the job currently in process and S the processing time for the new arrival. The distribution function of the cycle time of the job introduced into the system at time s is then given by

$$G(s,t) = \sum_{n=0}^{\infty} F_1 * F^{n*}(t) P\{Q(s) = n\} \tag{6.22}$$

where F_1 denotes the distribution function of the residual processing time of the job currently in process, $*$ the convolution operation, and F^{n*} the n-fold convolution of the processing time distribution F at the server. $G(s,t)$ thus describes a state-dependent cycle time distribution that depends on the number of jobs $Q(s)$ in the system at the time s the job was released. We seek an approximation of this function that will allow us to calculate approximate values of the loading factors w_{st}. To develop this approximation, the author assumes that this time-dependent delay distribution of an arriving order will have the same form as the steady-state distribution of the waiting time for an *M/G/1* queue, which is given by Shortle et al. (2018: 273), as

$$(1-\rho)\sum_{n=0}^{\infty} \rho^n F_e^{n*}(t) \tag{6.23}$$

where F_e is the steady-state residual processing time distribution derived assuming that the time a new job enters the system is uniformly distributed over the duration of the current service time. This suggests an approximation of the form

$$G(s,t) = F_1 * (1-\beta(s)) \sum_{n=0}^{\infty} \beta(s)^n F_e^{n*}(t) \tag{6.24}$$

where $\beta(s)$ denotes a time-dependent traffic intensity. Noting that for a phase-type service time distribution (Neuts 1981), $G(s,t)$ will also be of phase type, the author proposes heuristic estimates of $\beta(s)$, obtaining an approximation for $G(s,t)$ that depends only on the expected WIP level at time s, denoted by $\phi(s)$, and its time derivative $\phi'(s)$. Hence, to obtain an approximation to $G(s,t)$, we now need a viable technique for estimating $\phi(s)$ and $\phi'(s)$. These quantities are clearly linked to the evolution of WIP over time, which, in turn, depends on the pattern of releases into the production system, suggesting a recursive technique. Given a release pattern, we can compute estimates of $\phi(t)$ for every planning period t in a recursive manner, starting from period $t = 1$ and moving forward in time. If the processing time distribution at the server is phase-type, these computations can be performed efficiently. The resulting approximation to $G(s,t)$ yields approximate values of the w_{st}, which can be interpreted as the probability that a job released in period s will complete in period t. The author suggests a successive approximation method to compute the w_{st}, where for a given release pattern estimates of the w_{st} are developed after which a planning problem is solved to estimate WIP levels over time. These new WIP levels are used to estimate new loading factors until the estimates of weights converge.

The larger pattern of the iteration procedure is now clear: we begin with an initial release pattern, and calculate initial estimates of the w_{st}. We then calculate a new release pattern using these weights, and repeat until, hopefully, convergence is achieved. As in Zaepfel (1984), the model does not include separate capacity constraints because the load factors w_{st} reflect how the input is transformed into output. "If correctly computed, they will ensure the output is actually bounded. If too much input is placed into the system the weights will reflect these longer lead times" (Riaño 2003: 72).

As with the approach of Hung and Leachman (1996), the convergence behavior of this procedure is not well understood; when it converges, it converges quite rapidly to a solution that does not depend on the initial solution used, but in other cases, it can cycle through a limited number of solutions (Riaño 2003: 83). Further experimental and theoretical work is necessary to understand this convergence issue (see Sect. 6.5.3), but the overall approach stands as a very interesting and novel approach to modeling workload-dependent lead times in production planning, with a strong theoretical underpinning. Interesting discussions in this direction are given by Hackman (2008).

The iterative mechanisms discussed so far iterate solely on the lead times or on the loading factors. Byrne and Bakir (1999) iterate between a conventional multi-period LP production planning model that determines the optimal production levels for given capacity constraints and a simulation model that is used to update the available capacities if the production levels obtained from the initial optimization run turn out to be infeasible in the simulation. Lead times are not considered. Byrne and Hossain (2005) provide some extensions to this mechanism, again without considering lead times in the production planning model.

Kim and Kim (2001) also use loading factors to express lead times and include capacity constraints in their release model. Simulation is used to obtain estimates of the effective loading factors and resource utilization that are used to update the lead

times and the capacities in the release model within an iterative mechanism. The authors do not report convergence problems in their numerical tests. Irdem et al. (2010) report good convergence of this approach under both high and low levels of resource utilization. They conclude that "the convergence behavior of the KK (Kim and Kim 2001) procedure is qualitatively different from that of the HL (Hung and Leachman 1996) procedure" (452f.). Albey and Bilge (2014) conduct extensive experiments with the KK procedure and find that the procedure converges to different solutions from different initial release plans. They also observe that when the release planning model proposes a release plan that results in low capacity utilization, agreement with the lead time estimation model is often achieved fairly quickly, which may result in the procedure converging to a suboptimal solution. Once capacity estimates have been revised downwards and passed to the release planning model, they are implemented in a hard constraint that does not permit them to be revised upwards again at a subsequent iteration. They also find that combining the values of estimates from successive iterations using a smoothing constant improves performance and that convergence in aggregate convergence criteria such as total throughput over all periods and products is much easier to obtain that agreement for each product in each period. These authors also examine the performance of the KK procedure in the presence of routing flexibility and find that increasing flexibility improves its performance.

Bang and Kim (2010) formulate an iterative procedure using an aggregate production planning model designed for semiconductor wafer fabrication that uses a separate disaggregation stage to obtain the release quantities over time. Based on an extended (compared to Hung and Leachman 1996) simulation model, not only is cycle time information updated but also product types are regrouped for the next run of the aggregate production planning model. The authors report improvements compared to Hung and Leachman (1996) and good convergence for both methods in all problem instances tested, although convergence cannot be guaranteed. Kim and Lee (2016) propose an iterative scheme where the production planning level determines production and WIP levels (or the deviations from target values, respectively). These target values are updated based on the simulated cycle times, number of setups, and available WIP. The convergence of the procedure seems to depend on the variable used to specify the convergence criterion.

6.6.3 Critique of Iterative Simulation-LP Algorithms

The iterative simulation-LP approach to order release planning combines two familiar, off-the-shelf modeling techniques, linear programming, and simulation, in an iterative scheme that addresses the complex interdependency of releases and lead times. However, the simulation model requires large amounts of engineering effort and data to construct, validate, and maintain and increases run time significantly. The computational burden can be reduced by limiting the level of detail of the model to what is necessary for the specific purpose (Law and Kelton 2000: 267ff.),

e.g., by focusing on highly utilized workcenters and replacing operations at low-utilization workstations with fixed time lags (Hung and Leachman 1999). The ongoing increase in computational power alleviates this problem somewhat, but does not eliminate it. The overall procedure—starting with reasonable cycle time estimates that are refined based on the simulated dynamics of the material flow—is intuitive and easy to explain. Modeling the flow time dynamics outside the optimization model allows complex system dynamics to be embedded in the simulation or queueing models used to estimate lead times, permitting realistic modeling of the system within the limits of the available computational resources.

However, the behavior of this type of order release mechanism is not well understood. There is no guarantee of optimality and hardly any insight into its deviation from the optimum. Although convergence is ergodic in some numerical experiments (Riaño 2003), there is no proof of this property. The approaches often converge within a reasonable number of iterations (five or six in Hung and Leachman (1996)), but can frequently fail to converge, in which case it does not reach a feasible solution. This is not acceptable in real-life situations and largely precludes practical application. However, Kim and Kim (2001) do not report convergence problems in their numerical tests, which might indicate that including the capacities in the iterative mechanism makes a substantial difference. Note that updating the capacities changes the right-hand side of the order release model, while updating the lead times changes the coefficient matrix. However, it is not clear how this difference is related to the mechanism that coordinates the order release and lead time estimation models.

Irdem et al. (2008, 2010) and Kacar et al. (2012) perform numerical studies that explore both the convergence of the HL (Hung and Leachman 1996) and the KK (Kim and Kim 2001) method and, in the latter paper, their performance relative to a clearing function model of the type described in the next chapter. All three papers use the same simulation testbed, a scaled-down wafer fabrication facility first studied by Kayton et al. (1997). Irdem et al. (2008) find substantial convergence problems for the HL method, especially under high bottleneck utilization, which are confirmed in Irdem et al. (2010). This behavior is qualitatively different from the KK procedure for which they report good convergence (Irdem et al. 2010; Kacar et al. 2012). Kacar et al. (2012) compare a clearing function model with two parameter settings to the KK and the HL procedure using the same testbed. They find that for the KK method convergence is achieved after four iterations in most test cases, while the HL method is "consistently outperformed by the clearing function model" (p. 116). They also conclude that the dynamic behavior of the HL method is problematic due to large swings in releases from one period to the next. The KK procedure is mostly outperformed by the clearing function model, at least for the better of the two parameter settings.

The convergence issue highlights the fact that the theory behind the iterative simulation-LP approach is largely unclear, making it difficult to explain their behavior and the nature of the solution to which they converge. Missbauer (2020) analyzes a simplified version of the HL procedure assuming a production unit with a single workcenter. He shows that in the order release model the lead times, which

are time-varying parameters, act as prices for producing an item in a certain period. This is because the WIP holding costs are assigned to the production period due to the use of backward lead times and are proportional to the lead time assigned to this period. Similar insights arise in the analysis of fixed lead time models in Chap. 5, notably Equation (5.19). Hence an iterative order release procedure that iterates on the lead times behaves like a price coordination mechanism. Missbauer (2020) shows that the price coordination mechanism implied by the iterative order release mechanism does not meet the theoretical requirements for an effective price coordination mechanism, so a reasonable solution can only be expected under very specific conditions. This argument clearly does not extend, e.g., to the KK procedure that iterates on the capacities as well, suggesting different theoretical underpinning for different variants of the iterative mechanisms. These issues are largely unexplored, and more research is needed. Future research should link the design of iterative LP-simulation algorithms to the theory of mathematical decomposition and coordination that is available in the mathematical programming literature.

A comparison to the widely used techniques of simulation optimization (Fu 2002; Zapata et al. 2011) provides additional perspective on the performance of these iterative approaches. Simulation optimization is used when the objective function and constraints of the system of interest do not admit of a tractable mathematical representation but instead can be represented in a discrete-event simulation model. Thus the performance measure of interest cannot be computed directly, but must be estimated based on samples obtained from replications of the simulation. If we denote the vector of decision variables by θ and the estimate of the performance measure to be optimized obtained from the simulation replication w by $L(\theta, w)$, the general statement of a simulation optimization problem is then

$$\min_{\theta \in \Theta} J(\theta) \tag{6.25}$$

where $J(\theta) = E_w[L(\theta, w]$ where Θ denotes the set of all acceptable decision variable vector θ. The decision variables θ can be discrete or continuous. Fu (2002), Henderson and Nelson (2006), and Zapata et al. (2011) provide extensive reviews of this area. A wide variety of such algorithms exist, including genetic algorithms that use the simulation model to compute a fitness measure for different solutions and stochastic approximation methods for continuous state spaces. The latter methods start with an initial solution θ_0 that is updated iteratively using an estimate of the gradient $\nabla J(\theta)$ of $J(\theta)$. The general form of the stochastic approximation algorithm is as follows:

Step 1: Choose an initial solution θ_0. Set $n = 0$.

Step 2: Compute a new solution $\theta_{n+1} = \Pi_\Theta(\theta_n + a_n \nabla J(\theta_n))$ where θ_n is the variable set at the n'th iteration, a_n a step size, and Π_Θ denotes a projection onto Θ such that if θ_{n+1} lies outside the feasible region, Π_Θ returns it to the feasible region; one such projection is setting $\theta_{n+1} = \theta_n$. If a specified stopping criterion is satisfied, stop and return θ_{n+1} as the estimated optimal solution. Otherwise set $n = n+1$ and return to Step 2.

The quality of the solution obtained and the speed of convergence to that solution depend on the choices of the step sizes a_n and the manner in which the gradient $\nabla J(\theta_n)$ is computed. There are four general gradient estimation techniques: finite differences, likelihood ratio, perturbation analysis, and frequency domain experimentation. The finite difference technique estimates the gradient by running multiple simulations to obtain an approximation of the gradient. One version of finite differences is $\hat{\nabla}\left(J\left(\theta_n\right)\right)=\left[\hat{\nabla}_1 J\left(\theta_n\right)\cdots\hat{\nabla}_p J\left(\theta_n\right)\right]^T$ where p denotes the number of decision variables and

$$\hat{\nabla}_i J\left(\theta_n\right)=\frac{\hat{J}\left(\theta_n+c_i e_i\right)-\hat{J}\left(\theta_n-c_i e_i\right)}{2c_i} \tag{6.26}$$

Convergence requires that $c_i \to 0$. Here e_i denotes the i'th unit vector and the c_i difference parameters whose values represent a trade-off between too much noise (small values) and too much bias (large values). This gradient estimation technique is broadly applicable, but requires $2p$ simulation runs at each iteration.

The direct application of simulation optimization to release planning would treat the release quantities R_{it} of each product i in each period t as the decision variables and seek to optimize some objective function. Although simulation optimization is generally employed in the presence of random variables such as processing times, machine failures and yields, the basic approach can be implemented in completely deterministic simulations. Although models based on this approach have been developed and shown to yield good solutions (Liu et al. 2011; Kacar and Uzsoy 2015), their computational requirements are usually very high due to the time required to run multiple independent replications of a large simulation model. Some recent work attempts to reduce the computational burden of these procedures by replacing the simulation model with a metamodel based on extensive offline simulation experiments, with promising results (Li et al. 2016).

The iterative multi-model approaches can be viewed as simulations of a particular decision process: initial estimates of planning parameters such as lead times and resource utilizations are obtained, the release planning model is run, and the resulting release pattern is simulated. This perspective provides some insight into their performance. First, most multi-model iterative approaches do not consider the objective function value in their convergence criteria; instead they focus on achieving consistency in the flow time estimates obtained from successive iterations. Hence there is no *a priori* evidence that these procedures will converge to even a locally optimal solution with respect to the objective function of concern, as implemented in the release planning model; the best that can be hoped for is a feasible solution. Although the primary concern is the reduction of the differences in lead time estimates obtained at successive iterations, this is never explicitly formulated as an objective function to be reduced from one iteration to the next, nor is any information on the gradient of this quantity used. Simulation optimization methods that explicitly consider the gradient of the objective function generally yield good solutions, although their computational burden is very high.

Viewing these techniques as applications of fixed point iteration also raises concerns. The basic fixed point iteration procedure, common in numerical analysis, generates a sequence of solutions $x_{n+1} = f(x_n)$, $n = 0, 1, ...$ In the context of the iterative multi-model methods, the solution x_n at iteration n represents a vector of lead time estimates, while the function $f(x_n)$ represents the simulation of the decision process by which a release schedule is obtained by the LP model from the previous iteration's lead time estimates x_{n-1}. This release schedule is then simulated to obtain revised lead time estimates. Per the Banach Fixed Point Theorem (O'Regan et al. 2001), the existence of a fixed point in general requires the existence of a contraction mapping such that for any two points x_i and x_j there exists a constant $0 \leq q < 1$ such that $||f(x_i) - f(x_j)|| \leq q||x_i - x_j||$. In the current iterative methods, no conditions of this type are considered, let alone satisfied.

Our discussion of simulation optimization and fixed point iteration in relation to the iterative multi-model procedures is clearly heuristic in nature and provides no mathematically rigorous evidence. However, these considerations do suggest that most existing iterative methods are, in mathematical terms, ill-posed and require the imposition of additional conditions to ensure reliable performance in terms of solution quality and convergence.

6.7 Iterative Methods for Production Planning and Scheduling

The iterative methods described in Sect. 6.5.2 represent a small and rather specialized research direction in order release planning. However, a closer look at the literature reveals that this is a special case of a more general problem: Order release planning—as a subproblem of production planning—requires information on lead times and maximum possible production which, in turn, depend on the detailed schedule within the production unit. While it is true that, as stated in the optimized production technology (OPT) approach, "lead times are the result of a schedule and can't be predetermined" (Vollmann et al. 1997: 797), the monolithic approach to production planning and control, at least for the bottleneck workcenters, that results from this view is not always applicable, motivating the hierarchical approaches described in Chap. 1. Planned lead times allow decomposition of the complex planning problem into planning and scheduling levels (Graves 2011: 93) and thus are necessary within this planning concept, but both lead times and capacities should anticipate the outcomes of the scheduling level reliably (Kanet and Sridharan 1998).

It is thus not surprising that iteration between the planning and scheduling levels has also been proposed for other production planning tasks. Integrating the planning and scheduling levels is particularly important in lot sizing. This can be achieved, e.g., by anticipating the queuing effects of lot sizes using stochastic models and determining lot sizes accordingly, as discussed in Chap. 9, or by lot streaming, that is, splitting up production lots into smaller transfer batches whose processing on

different workcenters can be overlapped in time (Cheng et al. 2013). Dauzere-Peres and Lasserre (2002) present an integrated model for lot sizing and scheduling and an algorithm that iterates between a lot-sizing module that assumes a fixed production sequence and a scheduling module that sequences the given lots. In this approach the lead time acts as a capacity constraint (p. 789). Negenman (2000) presents an algorithm that iterates between an LP model that calculates the production plan for a production network and a flexible flow shop scheduling model that is solved by a heuristic. The feedback information provided by the scheduling level is the completion times of the orders. If the planned lead times are exceeded, the planning level reduces the available capacities of the workcenters in the next iteration. A detailed analysis of the convergence behavior is not provided. Albey and Bilge (2011) present a hierarchical production planning and control system framework for a Flexible Manufacturing System that consists of three levels: aggregate planning, loading, and detailed planning. The behavior of the shop floor for a given production plan is anticipated using simulation. The simulation result is used to update capacity coefficients in the upper-level modules. Again, convergence is not analyzed in detail, but the authors indicate that capacity updating is complex due to the special problem and decision structure.

Planned lead times can also be required for dispatching decisions when the dispatching rule compares a job's current slack time to its remaining lead time. In this case the lead time estimate must be consistent with the schedule that is based on this estimate (Vepsalainen and Morton 1988). The lead time iteration method (Vepsalainen and Morton 1988; Morton and Pentico 1993) updates initial lead time estimates used for scheduling using the actual flow times obtained from the scheduling algorithm using exponential smoothing (Morton and Pentico 1993: 218f). Convergence is not guaranteed, and "it is then an empirical question whether such a procedure obtains good results or not" (Morton and Pentico 1993: 219). Lu et al. (1994) provide an interesting illustration of the lead time iteration procedure in a semiconductor wafer fabrication facility. Note that the role of planned lead times in scheduling algorithms is different from that in order release models, and thus the relationship of these results to the convergence issue of the LP-simulation approaches discussed in the previous section is not straightforward. A unifying view of algorithms that iterate between a production planning model, independent of its formulation, and a scheduling model, independent of the scheduling algorithm, is a challenge for future research.

6.8 Conclusions

The various models discussed in this chapter highlight the difficulty of the central problem addressed in this volume: how to anticipate the behavior of the scheduling level in planning models in a manner that is both sufficiently accurate and computationally tractable. The linear programming models presented in Chap. 5 can be extended easily to handle time-varying exogenous lead times, but this begs the

question of how to obtain such estimates since lead times are determined by utilization and utilization by the release decisions the model seeks to address. Work in traffic modeling suggests that optimization models with lead times as an endogenous decision variable are often non-convex and hence hard to solve. Attempts to preserve computational tractability have led to the use of multi-model approaches that separate the problems of release planning and lead time estimation, but the convergence behavior of these is not well understood, and the use of a simulation model to construct the planning solution (as opposed to estimating its parameters offline, outside the planning run) result in high computational burden for large production systems. What is needed is a way of representing the behavior of the scheduling level within the release planning model that is consistent with the queueing view of production resources in Chap. 2, but which yields tractable optimization models. The clearing functions discussed in the next two chapters seek to provide such a model.

References

Albey E, Bilge Ü (2011) A hierarchical approach to FMS planning and control with simulation-based capacity anticipation. Int J Prod Res 49(11):3319–3342

Albey E, Bilge U (2014) An improved iterative linear programming-simulation approach for production planning. Department of Industrial Engineering, Ozyegin University, Istanbul

Armbruster D, Uzsoy R (2012) Continuous dynamic models, clearing functions, and discrete-event simulation in aggregate production planning. INFORMS Tutorials in Operations Research

Bang JY, Kim YD (2010) Hierarchical production planning for semiconductor wafer fabrication based on linear programming and discrete-event simulation. IEEE Trans Autom Sci Eng 7(2):326–336

Bazaraa MS, Sherali HD, Jarvis J (1979) Nonlinear programming: theory and algorithms. Wiley, New York

Ben-Daya M, Raouf A (1994) Inventory models involving lead time as a decision variable. J Oper Res Soc 45(5):579–582

Bertsimas D, Mourtzinou G (1997) Transient laws of non-stationary queueing systems and their applications. Queue Syst 25:115–155

Byrne MD, Bakir MA (1999) Production planning using a hybrid simulation-analytical approach. Int J Prod Econ 59:305–311

Byrne MD, Hossain MM (2005) Production planning: an improved hybrid approach. Int J Prod Econ 93-94:225–229

Carey M (1992) Nonconvexity of the dynamic traffic assignment problem. Transport Res B 26B(2):127–133

Carey M, Subrahmanian E (2000) An approach to modelling time-varying flows on congested networks. Transport Res B 34:157–183

Cheng M, Mukherjee NJ, Sarin SC (2013) A review of lot streaming. Int J Prod Res 51(23/24):7023–7046

Dauzere-Peres S, Lasserre JB (2002) On the importance of sequencing decisions in production planning and scheduling. Int Trans Oper Res 9:779–793

Ehteshami B, Petrakian R, Shabe P (1992) Trade-offs in cycle time management: hot lots. IEEE Trans Semicond Manuf 5(2):101–106

Figueira G, Almada-Lobo B (2014) Hybrid simulation–optimization methods: a taxonomy and discussion. Simul Model Pract Theor 46:118–134

Fu MC (2002) Optimization for simulation: theory vs practice. INFORMS J Comput 14(3):192–215
Gong L, de Kok T, Ding J (1994) Optimal leadtimes planning in a serial production system. Manag Sci 40(5):629–632
Graves SC (2011) In: Kempf KG, Keskinocak P, Uzsoy R (eds) Uncertainty and Production Planning. Planning Production and Inventories in the Extended Enterprise: A State of the Art Handbook, Volume 1, International Series in Operations Research and Management Science, vol 151. Springer, New York and Heidelberg, pp 83–101
Hackman S (2008) Production economics: integrating the microeconomic and engineering perspectives. Springer, Berlin
Hackman ST, Leachman RC (1989) A general framework for modeling production. Manag Sci 35(4):478–495
Henderson SG, Nelson BL eds (2006) Simulation. In: Handbooks in operations research and management science. North-Holland, Amsterdam
Hopp WJ, Spearman ML (2008) Factory physics: foundations of manufacturing management. Irwin/McGraw-Hill, Boston
Hopp WJ, Sturgis MLR (2000) Quoting manufacturing due dates subject to a service level constraint. IIE Trans 32(9):771–784
Hung YF, Hou MC (2001) A production planning approach based on iterations of linear programming optimization and flow time prediction. J Chin Inst Indus Eng 18(3):55–67
Hung YF, Leachman RC (1996) A production planning methodology for semiconductor manufacturing based on iterative simulation and linear programming calculations. IEEE Trans Semicond Manuf 9(2):257–269
Hung YF, Leachman RC (1999) Reduced simulation models of wafer fabrication facilities. Int J Prod Res 37(12):2685–2701
Ioannou G, Dimitriou S (2012) Lead time estimation in MRP/ERP for make-to-order manufacturing systems. Int J Prod Econ 139(2):551–563
Irdem DF, Kacar NB, Uzsoy R (2008) An experimental study of an iterative simulation-optimization algorithm for production planning. In: Mason SJ, Hill R, Moench L, Rose O (eds) 2008 Winter Simulation Conference, Miami, FL
Irdem DF, Kacar NB, Uzsoy R (2010) An exploratory analysis of two iterative linear programming-simulation approaches for production planning. IEEE Trans Semicond Manuf 23:442–455
Jonsson P, Matsson SA (2006) A longitudinal study of material planning applications in manufacturing companies. Int J Oper Prod Manag 26(9):971–995
Kacar NB, Uzsoy R (2015) Estimating clearing functions for production resources using simulation optimization. IEEE Trans Autom Sci Eng 12(2):539–552
Kacar NB, Irdem DF, Uzsoy R (2012) An experimental comparison of production planning using clearing functions and iterative linear programming-simulation algorithms. IEEE Trans Semicond Manuf 25(1):104–117
Kanet JJ, Sridharan V (1998) The value of using scheduling information in planning material requirements. Decis Sci 29(2):479–496
Kayton D, Teyner T, Schwartz C, Uzsoy R (1997) Focusing maintenance improvement efforts in a wafer fabrication facility operating under theory of constraints. Prod Invent Manag 38(Fourth Quarter):51–57
Keskinocak P, Tayur S (2004) Due-date management policies. In: Simchi-Levi D, Wu SD, Shen ZM (eds) Supply chain analysis in the e-business era: handbook of quantitative supply chain analysis. Kluwer Academic, Dordrecht
Kim B, Kim S (2001) Extended model for a hybrid production planning approach. Int J Prod Econ 73:165–173
Kim SH, Lee YH (2016) Synchronized production planning and scheduling in semiconductor fabrication. Comput Indus Eng 96:72–85
Lautenschläger M (1999) Mittelfristige Produktionsprogrammplanung mit auslastungsabhängigen Vorlaufzeiten. Peter Lang, Frankfurt am Main
Law AM, Kelton WD (2000) Simulation modeling and analysis, 3rd edn. McGraw Hill, New York

Law AM, Kelton WD (2004) Simulation modeling and analysis. McGraw-Hill, New York
Leachman RC, Carmon TF (1992) On capacity modeling for production planning with alternative machine types. IIE Trans 24(4):62–72
Li M, Yang F, Uzsoy R, Xu J (2016) A metamodel-based monte carlo simulation approach for responsive production planning of manufacturing systems. J Manuf Syst 38:114–133
Liu J, Li C, Yang F, Wan H, Uzsoy R (2011) Production planning for semiconductor manufacturing via simulation optimization. In: Jain S, Creasey RR, Himmelspach J, White KP, Fu R (eds) Winter simulation conferemce. IEEE, Piscataway, NJ
Lu S, Ramaswamy D, Kumar PR (1994) Efficient scheduling policies to reduce mean and variance of cycle time in semiconductor plants. IEEE Trans Semicond Manuf 7:374–388
Milne RJ, Mahapatra S, Wang C-T (2015) Optimizing planned lead times for enhancing performance of MRP systems. Int J Prod Econ 167:220–231
Missbauer H (2020) Order release planning by iterative simulation and linear programming: theoretical foundation and analysis of its shortcomings. Eur J Oper Res 280:495–507
Morton TE, Pentico D (1993) Heuristic scheduling systems: with applications to production systems and project management. Wiley, New York
Narahari Y, Khan LM (1997) Modeling the effect of hot lots in semiconductor manufacturing systems. IEEE Trans Semicond Manuf 10(1):185–188
Negenman EG (2000) Material coordination under capacity constraints. Industrial engineering. Eindhoven University of Technology, Eindhoven
Neuts MF (1981) Matrix-geometric solutions in stochastic models. Johns Hopkins University Press, Baltimore, MD
O'Regan D, Meehan M, Agarwal RP (2001) Contractions. In: Fixed point theory and applications. Cambridge University Press, Cambridge, pp 1–11
Orcun S, Uzsoy R (2011) The effects of production planning on the dynamic behavior of a simple supply chain: an experimental study. In: Kempf KG, Keskinocak P, Uzsoy R (eds) Planning in the extended enterprise: a state of the art handbook. Springer, Berlin, pp 43–80
Ozturk A, Kayaligil S, Ozdemirel NE (2006) Manufacturing lead time estimation using data mining. Eur J Oper Res 173:683–700
Peeta S, Ziliaskopoulos AK (2001) Foundations of dynamic traffic assignment: the past, the present and the future. Netw Spat Econ 1(3-4):233–265
Riaño G (2003) Transient behavior of stochastic networks: application to production planning with load-dependent lead times. School of Industrial and Systems Engineering. Georgia Institute of Technology, Atlanta, GA
Riaño G, Hackman S, Serfozo R (2006) Transient behavior of queueing networks. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA
Shanthikumar JG, Sargent RG (1983) A unifying view of hybrid simulation/analytic models and modeling. Oper Res 31(6):1030–1052
Shortle JF, Thompson JM, Gross D, Harris CM (2018) Fundamentals of queueing theory. Wiley, Hoboken, NJ
Vepsalainen AP, Morton TE (1988) Improving local priority rules with global lead-time estimates: a simulation study. J Manuf Oper Manag 1:102–118
Vollmann T, Berry W, Whybark D (1997) Manufacturing planning and control systems. Irwin, Boston
Voss S, Woodruff DL (2003) Introduction to computational optimization models for production planning in a supply chain. Springer, New York
Zaepfel G (1984) Systemanalytische Konzeption der Produktionsplanung und –steuerung für Betriebe der Fertigungsindustrie. In: Zink C (ed) Sozio-Technologische Systemgestaltung als Zukunftsaufgabe, (in German). Carl Hanser Verlag, Munich
Zapata JC, Pekny J, Reklaitis GV (2011) Simulation-optimization in support of tactical and strategic enterprise decisions. In: Kempf KG, Keskinocak P, Uzsoy R (eds) Planning production and inventories in the extemnded enterprise: a state of the art handbook, vol 1. Springer, New York, pp 593–628