OR Spektrum (1989) 11:3–16

OR Spektrum
© Springer-Verlag 1989

A Survey of Priority Rule-Based Scheduling

R. Haupt

Lehrstuhl für Personalwirtschaft, Universität zu Köln, Albertus-Magnus-Platz, D-5000 Köln 41, Federal Republic of Germany

Received February 2, 1988 / Accepted June 30, 1988

Summary. In this paper, we survey the literature on heuristic priority rule-based job shop scheduling. Priority rules have been intensively investigated over the last 30 years by means of simulation experiments. They are also used in Shop Floor Control software systems. We present a classification, a characterization, and an evaluation of elementary priority rules. Some priority rule-related model extensions are discussed.

Zusammenfassung. In diesem Beitrag wird ein Überblick über heuristische, prioritätsregelgestützte Auftragsreihenfolgeplanung gegeben. Prioritätsregeln sind in den letzten 30 Jahren eingehend anhand von Simulations-Experimenten untersucht worden. Sie haben ebenso in Programmsysteme zur Produktionsplanung und -steuerung Eingang gefunden. Der Beitrag bemüht sich um eine Klassifizierung, Charakterisierung und Beurteilung von elementaren Prioritätsregeln. Abschließend werden einige prioritätsregelrelevante Modellerweiterungen angeschnitten.

1. Introduction

Production scheduling is part of the shop management decision system. Specifically, the day-to-day scheduling and dispatching of jobs through the shop is embedded in a Shop Floor Control (SFC) System.

The scheduling or sequencing problem is defined as the determination of the order in which a set of *jobs* (tasks) $\{i \mid i = 1, \ldots, n\}$ is to be processed through a set of *machines* (processors, work stations) $\{k \mid k = 1, \ldots, m\}$ [36, 10, 111, 59].

In real life situations beyond the manufacturing industries jobs may as well be interpreted as aircrafts queuing up to land, or as patients waiting for to be treated by a consultant surgeon, or as bank customers at a row of tellers' windows, respectively. Correspondingly, one would identify machines with an airport runway, or with surgical facilities, or with bank tellers, respectively [36, 59].

Job i is specified by a set of operations $\{j \mid j = 1, \ldots, m_i\}$ representing the processing requirements on various machines. Depending on whether the job processing order (routing) implies a predetermined sequence of operations, we distinguish between the case of arbitrary routings (*open* shop) and the case of given job routings which may be identical for all jobs (*flow* shop) or non-identical (*job* shop). Unless otherwise specified, we assume in what follows the case of individual but determined job processing orders (job shop).

A scheduling model usually involves the following assumptions [36, p. 5; 220; 38; 111, p. 11; 59, p. 8], the four last of which will be relaxed in Sect. 4:

(1) Each job is processed by one machine at a time (no splitting, no overlapping).

(2) Each machine processes one job at a time (no use of machining-centers).

(3) Each operation once started must be completed without interruption (no preemption).

(4) Each machine is continuously available for production (no breakdown).

(5) The only limiting resource is the machine (no lack of operator, tool, or material).

(6) Processing of an operation comprises all technologically determined times such as machining, setup, or move times. Processing times are independent of the schedule (no sequence-dependent setup-times).

(7) Jobs are strictly-ordered sequences of operations (no assembly).

(8) A given operation can be performed by only one type of machine (no alternate routing).

In Sect. 2 we introduce the concept of the "priority rule" by referring to the distinction in static vs. dynamic scheduling. In Sect. 3 a state-of-art survey of dynamic job shop scheduling research under the above mentioned assumptions is provided. In Sect. 4 we discuss some priority rule-based scheduling extensions by relaxing assumptions (5) to (8). In Sect. 5 we try to give an outlook on future dynamic scheduling research.

2. Dynamic Job Shop Scheduling

One basic distinction in scheduling research refers to the nature of the job arrivals in the shop [36, p. 7]. In a *static* model jobs arrive simultaneously and are available for to be scheduled at the same instant. Accordingly, their ready or release times r_i are 0, i.e. the total set of jobs is scheduled at time $t = 0$ ("all-at-once-scheduling"). New entering jobs are not admitted to the shop until the preceding scheduling cycle is finished. A *dynamic* model allows for a continuous stream of arriving orders in time that are intermittently released to the shop and are included in the current scheduling procedure. Reasonably, the distinction between simultaneous and intermittent job arrivals involves the one between known and fixed job data on one hand and stochastic data, in particular job interarrival times, on the other hand. Hence, we distinguish a static/deterministic scheduling problem from a dynamic/stochastic one. Here, we are exclusively concerned with the latter case.

Within the subset of dynamic/stochastic models we deal with experimental, simulation-based approaches, while ignoring the analytical procedures by means of queuing theory systems. The vast majority of simulation-based dynamic job shop scheduling literature assumes a Poisson distribution of job arrivals and, correspondingly, exponentially distributed interarrival times. As far as the processing time is considered as random variable, exponential as well as normal distributions occur.

The evident advantage of a dynamic scheduling approach is due to the fact that it allows for an up-to-date decision with respect to meanwhile entering (possibly rush) jobs, by loading a machine at the latest possible moment, namely as that machine gets idle [101]. Contrarily, a static model postpones the urgent job to the subsequent scheduling cycle. Accordingly, the dynamic property of a simulation model exhibits the obvious disadvantage that each sequencing decision is based on the constrained information horizon given by the set of currently scheduleable jobs, which prohibits the definition of an overall optimum: due to the real-time capability of a given sequencing decision, only the selection of the *first* job of the computed job sequence on that machine is actually performed, while the remainder of the scheduled jobs is rescheduled and possibly revised on the occasion of the subsequent loading moment. Thus, rather than to determine a global optimal sequencing policy, a dynamic job shop scheduling simulation at best is able to provide a heuristic optimum among alternative sequencing strategies by which a given job file is scheduled through the shop in successive simulation runs [25].

Such a policy that defines a specific sequencing decision each time a machine gets idle, is called a *priority rule*. A priority rule allows an idle machine to select its next operation from among those available. Primarily, "available" refers to currently waiting jobs at the corresponding machine; but, as we show in Sect. 3.2, the availability-property may also be extended to jobs being currently in the queue or on the machine of other work stations before proceeding to the queue in question.

3. Elementary Approaches in Priority Rule-Based Scheduling

3.1. Measures of Shop Performance

Basically, two groups of appropriate performance criteria, namely *flow-time*-based and *due-date*-based measures, are of importance in dynamic scheduling. The following definitions and notations are introduced [36, p. 11; 111, p. 18; 59, p. 10]:

p_{ij}: processing-time for operation j of job i

d_i: due-date, i.e. the promised delivery date of job i

a_i: allowance for job i, i.e. the allowed lead time that is assigned to job i at its arrival or release-time r_i;

$$a_i = d_i - r_i$$

W_{ij}: waiting-time preceding operation j of job i

C_i: completion-time of job i;

$$C_i = r_i + \sum_{j=1}^{m_i} (W_{ij} + p_{ij}),$$

since, according to assumption (6) in Sect. 1, a job in the shop is either on a machine or in a queue

F_i: flow-time (shop time, manufacturing interval) of job i;

$$F_i = C_i - r_i = \sum_{j=1}^{m_i} W_{ij} + \sum_{j=1}^{m_i} p_{ij}$$

L_i: lateness of job i;

$$L_i = C_i - d_i = C_i - (a_i + r_i) = F_i - a_i$$

T_i: tardiness of job i;

$$T_i = \max\,\{0, L_i\}$$

Flow-time-oriented performance measures are

$$\sum_{i=1}^{n} F_i \quad \text{or} \quad \bar{F} = \frac{1}{n} \cdot \sum_{i=1}^{n} F_i$$

respectively. By definition, $\bar{F}$, $\bar{C}$, $\bar{W}$, $\sum_{i=1}^{n} F_i$, $\sum_{i=1}^{n} C_i$, and $\sum_{i=1}^{n} \sum_{j=1}^{m_i} W_{ij}$ are equivalent performance criteria, i.e. a schedule that optimizes $\sum_{i=1}^{n} F_i$ or $\bar{F}$, also optimizes

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} W_{ij} \quad \text{or} \quad \bar{W} = \frac{1}{\sum_{i=1}^{n} m_i} \cdot \sum_{i=1}^{n} \sum_{j=1}^{m_i} W_{ij},$$

as well as

$$\sum_{i=1}^{n} C_i \quad \text{or} \quad \bar{C} = \frac{1}{n} \cdot \sum_{i=1}^{n} C_i,$$

since both processing-times and release-dates are unaffected by the sequencing decision (s. Assumption 6 in Sect. 1) [111, p. 21].

Furthermore, $\bar{F}$ is proportional to the mean in-process inventory, since the steady-state behaviour of a waiting-system implies [36, p. 19]: $\bar{F} = 1/\lambda \cdot \bar{N}$, where λ denotes the mean arrival rate (and accordingly, $1/\lambda$ denotes the mean time between two arrivals), and $\bar{N}$ denotes the mean number of jobs in the system.

Another flow-time-based measure of effectiveness is the distribution of the individual flow-times, as expressed e.g. by the flow-time variance $\sigma^2(F) = \frac{1}{n} \cdot \sum_{i=1}^{n} (F_i - \bar{F})^2$.

In static problems, the measures $F_{\max} = \max\,\{F_i\}$ and $C_{\max} = \max\,\{C_i\}$ denote the scheduling time or make-span, since $r_i = 0$ for all $i = 1, \ldots, n$. In a dynamic model (i.e. different $r_i \geqslant 0$ for the jobs), the make-span is not a reasonable criterion. (Correspondingly, measures of machine or shop utilization are unusual in a dynamic scheduling environment.)

Due-date-based criteria are

$$\sum_{i=1}^{n} L_i \quad \text{or} \quad \bar{L} = \frac{1}{n} \cdot \sum_{i=1}^{n} L_i,$$

respectively (which is equivalent to $\sum_{i=1}^{n} F_i$ or $\bar{F}$, respectively, since L_i and F_i differ from one another by the sequence-independent allowances a_i), and

$$\sum_{i=1}^{n} T_i \quad \text{or} \quad \bar{T} = \frac{1}{n} \cdot \sum_{i=1}^{n} T_i,$$

respectively. In case, early deliveries involve penalties (which might be realistic under just-in-time management conditions), the measures

$$\sum_{i=1}^{n} E_i \quad \text{or} \quad \bar{E} = \frac{1}{n} \cdot \sum_{i=1}^{n} E_i,$$

where E_i, the earliness of job i, is defined as $E_i = \max\,\{-(C_i - d_i), 0\}$, could be of interest.

Rather than defining $L_{\max} = \max\,\{L_i\}$ or $T_{\max} = \max\,\{T_i\}$ (which are useful expressions in static approaches), one would prefer variance criteria for lateness and tardiness

$$\sigma^2(L) = \frac{1}{n} \cdot \sum_{i=1}^{n} (L_i - \bar{L})^2$$

$$\sigma^2(T) = \frac{1}{n} \cdot \sum_{i=1}^{n} (T_i - \bar{T})^2$$

or the fraction of jobs tardy $f_t (0 \leqslant f_t \leqslant 1)$ in dynamic models.

Some priority rule-based investigations introduce cost criteria rather than time-oriented measures. Flow-time-oriented performance criteria might be replaced by in-process inventory cost, while tardiness-based measures account for contractual penalties for late deliveries, for customer badwill, for lost sales, and rush shipping cost [122].

Due to the introduction of cost-based performance expressions one succeeds in coping with the multiple objective problem.

3.2. Classification of Priority Rules

As the definition of a priority rule given in Sect. 2 points out, the concept of a priority rule-based scheduling approach considers the sequencing decision as a set of independent decentralized one-machine problems [73, 101]. Hence, the idea of transferring algorithms which are optimal with respect to static one-machine problems to a dynamic job shop model environment seems to be intuitively appealing. For example, an order of jobs arranged according to non-decreasing processing-times minimizing $\bar{F}$, $\bar{C}$ and $\bar{W}$ in the one-machine case (Smith-rule) [36, p. 26] can be used as a reasonable heuristic flow-time-oriented sequencing principle ("shortest processing-time"-rule) in a dynamic job shop approach. Similarly, the application of the "due-date"-rule, i.e. a sequence according to non-decreasing job due-dates, which minimizes $T_{\max}$ and $L_{\max}$ in the one-machine problem (Jackson-rule) [36, p. 30] may serve as tardiness-oriented "rule of thumb" in a simulative approach. As these two examples show, evident job attributes, like processing-times, due-dates, and other data of the job file, are typical arguments of a priority function.

A priority function based on properties of an individual job assigns a priority value to that job, regardless of the urgency of other competing jobs. A basically different approach is one, in which the priority of a job depends on the urgency of other jobs too. This could be achieved for example by checking, if a selection of a job makes its competitor jobs critical (e.g. in terms of their slack). In fact, such a rule estimates the effect expressed by some performance measure that a hypothetical job selection has on the *total* of scheduleable jobs, i.e. on other jobs and on itself. Thus, this procedure tries first to "optimize" and then to select. The "alternate operation" rule of Gere [60] provides an example for this approach.

Priority rules on the basis of individual job information without considering priority interactions with other competing jobs first can be classified into time-independent and time-dependent functions. A processing-time-based rule can be regarded as time-independent, because the priority value, once computed at the entry of that operation in the corresponding queue, does not change over time, while time-dependent priorities, e.g. slack-based rules (s. Sect. 3.3), vary over time by definition and have to be (re)calculated, for reasons of real-time processing, at the last possible moment, i.e. as the machine gets idle and a loading decision is to be made.

Second, a distinction between local and global priority rules refers to the extension of the information status of rules. A local rule requires information only about jobs currently waiting at the machine in question, whereas global rules are based on information about jobs or about the shop beyond the corresponding queue. Since priority rules are typically applied in a decentralized sequencing procedure, the use of local queue information seems to be appropriate. But the

Table 1. Classification scheme for priority rules

Computation of a job's priority referred to

1. attributes only of the job in question:
   - 1.1. Local queue information:
     - 1.1.1. Time-independent priority computation:
       - Random attributes: RANDOM, FCFS, FASFS
       - Job processing attributes: SPT, LPT, LWKR, MWKR, FOPNR, GOPNR, TWORK
       - Job due-date attributes: DD, ODD
     - 1.1.2. Time-dependent priority computation: ALL, SL, CR, ALL/OPN, S/OPN, S/WKR, S/ALL, OSL, OCR
   - 1.2. Global shop information: NINQ, WINQ, XWINQ LA
2. both of the job in question and of other competing jobs: ALTOP

C_{ij} Completion time of operation j of job i ($C_{i0} = r_i$ = release time of job i, s. above) ▶

$N_{ij}(t)$ Number of waiting jobs at time t in the queue containing operation j of job i

P Set of jobs which are currently processed on machines preceding operation j

q Index of remaining operations ($q = j, \ldots, m_i$)

R Set of jobs in queue of operation j

X_{ij} Particular value of a random variable, uniformly distributed between 0 and 1, assigned to operation j of job i

$Y_{ij}(t)$ Total work, i.e. the sum of the imminent-operation processing-times, of waiting jobs at time t in the queue containing operation j of job i

$Y'_{ij}(t)$ Total work (including work that will "soon" arrive) of waiting jobs at time t in the queue containing operation j of job i (a job is expected to arrive "soon", if, at time t, its preceding operation is being performed

$Z_i(t)$ Priority value of job i at time t ($i \in R$ for rules (1)–(24) and (26), $i \in \{P \cup R\}$ for rule (25)) (Smallest values of $Z_i(t)$ have greatest priority)

Table 2. Formalization of basic priority rules

Rule	Definition $Z_i(t)$	Description Job selected which has ...
(1) RANDOM	X_{ij}	the smallest value of a *random* priority
(2) FCFS	$C_{i,j-1}$	arrived at queue first; (*"first come, first served"*)
(3) FASFS	r_i	arrived at shop first; (*"first arrival at shop, first served"*)
(4) SPT	p_{ij}	the *shortest processing-time*
(5) LPT	$-p_{ij}$	the *longest processing-time*
(6) LWKR	$\sum_{q=j}^{m_i} p_{iq}$	the *least work remaining*
(7) MWKR	$-\sum_{q=j}^{m_i} p_{iq}$	the *most work remaining*
(8) FOPNR	$m_i - j + 1$	the *fewest* number of *operations remaining*
(9) GOPNR	$-(m_i - j + 1)$	the *greatest* number of *operations remaining*
(10) TWORK	$\sum_{j=1}^{m_i} p_{ij}$	the greatest *total work*
(11) DD	d_i	the earliest *due-date*
(12) ALL	$d_i - t$	the smallest *allowance*
(13) SL	$d_i - t - \sum_{q=j}^{m_i} p_{iq}$	the smallest *slack*
(14) CR	$(d_i - t)/\sum_{q=j}^{m_i} p_{iq}$	the smallest *critical ratio*
(15) ALL/OPN	$(d_i - t)/(m_i - j + 1)$	the smallest ratio of *allowance* per number of *operations remaining*
(16) S/OPN	$\left(d_i - t - \sum_{q=j}^{m_i} p_{iq}\right)/(m_i - j + 1)$	the smallest ratio of *slack* per number of *operations remaining*
(17) S/WKR	$\left(d_i - t - \sum_{q=j}^{m_i} p_{iq}\right)/\sum_{q=j}^{m_i} p_{iq}$	the smallest ratio of *slack* per *work remaining*
(18) S/ALL	$\left(d_i - t - \sum_{q=j}^{m_i} p_{iq}\right)/(d_i - t)$	the smallest ratio of *slack* per *allowance*
(19) ODD	d_{ij}	the earliest *operation due-date*
(20) OSL	$d_{ij} - t - p_{ij}$	the smallest *operation slack*
(21) OCR	$(d_{ij} - t)/p_{ij}$	the smallest *operation critical ratio*
(22) NINQ	$N_{i,j+1}(t)$	the least *number* of jobs in the *queue* of its *next operation*
(23) WINQ	$Y_{i,j+1}(t)$	the least total *work in* the *queue* of its *next operation*
(24) XWINQ	$Y'_{i,j+1}(t)$	the least total *work in* the *queue* of its *next operation* (both present and expected)
(25) LA	$Z_i\ (i \in \{P \cup R\})$	the highest priority among the jobs in the corresponding queue and those which are currently processed on machines preceding to the operation in question (*"look-ahead"*)
(26) ALTOP	$-\sum_{\substack{h \in R \\ h \neq i}} \min\{d_h - (t + p_{ij}) - \sum_{q=j}^{m_h} p_{hq}; 0\} - \min\{d_i - (t + p_{ij}) - \sum_{q=j+1}^{m_i} p_{iq}; 0\}$	the smallest sum of tardiness for all jobs in the queue due to the selected job (*"alternate operation"*)

more one is willing to consider a scheduling problem as a centralized approach, where interactions with parallel loading decisions at other machines are to be taken into account, the more the use of global shop information becomes adequate. Rather than to regard a machine as an island, one might expect scheduling improvements by extending the "myopic" information horizon to neighbor machines and subsequent sequencing decisions.

Similar to global queue information, the performance evaluation rationale, i.e. the priority interaction approach, equally tends to anticipate the future progress of a schedule, which, of course, in a dynamic model environment, is a rather limited "look ahead", since the expected performance of a job selection is no more than an approximate estimation of the objective function.

Summarizing, we present a classification scheme in Table 1. The rules to be defined in Sect. 3.3 are assigned to the appropriate cases. Though the heuristic scheduling literature is familiar with the two basic distinctions between time-independent and time-dependent rules and between local and global rules [77, 36, 91, 104], survey articles often ignore those classifications [38, 104, 24]. Moreover, the class 2 ("... attributes both of the job in question and of other jobs") does not occur at all.

3.3. Survey of Priority Rules

The rules presented in Table 2 and classified in Table 1 are *basic* ones, i.e. rules that are adequate to the elementary approaches under assumptions (1)–(8) in Sect. 1 (rules relating to more complex model extensions are introduced in Sect. 4). Moreover, the list in Tables 1 and 2 is limited to *simple* rules, since combinations of rules can be designed in a great variety, once a set of elementary simple rules having been defined (s. later in this section). Finally, the survey is not exhaustive; we discuss related modifications of some rules in this section and in Sect. 3.4.

Rules with random attributes, i.e. (1) RANDOM, (2) FCFS, and (3) FASFS, serve as benchmarks in comparison to reasonable heuristics which one would expect to perform better.

Among the rules with job processing information, (4) SPT is the most known, the most applied, and yet one of the most efficient rules. In line with (5) LPT, it requires the lowest information amount, since only operation data (not job data) from the local queue (not from other queues) are needed.

While (6) LWKR and (7) MWKR refer to the *work* remaining, (8) FOPNR and (9) GOPNR are based on the *number* of operations remaining. Similarly, as alternate version to (10) TWORK, a variant could be designed which favors the greatest number of total operations [24].

LWKR and FOPNR give preference to jobs the work completed of which is rather advanced. Thus, they can be regarded as value-oriented rules selecting jobs with a high fraction of their value added or cumulative value to their total value, whereas TWORK accounts for a static value version. Explicit value approaches are investigated by e.g. [5, 6, 7, 74, 106, 119, 120, 72].

The intent of MWKR and GOPNR is to speed up jobs with large processing work resulting in a well-balanced work progress of all jobs, at the expense of a high volume of in-process inventory, while LWKR and FOPNR tend to reduce the number of jobs in the shop.

Other modifications of job processing-oriented rules concern SPT or LPT multiplied with or divided by TWORK, respectively [125].

Among the due-date-related priority rules, regardless of whether they are time-independent (class 1.1.1) or time-dependent (class 1.1.2 in Table 1), (11) DD is the most important rule and the one that formalizes a natural, intuitive due-date behavior of everyday life. DD is equivalent to (12) ALL, i.e. job orders computed according to DD and ALL are identical, yet ALL would require actualized time-dependent rescheduling. Therefore, DD clearly dominates ALL, which has no practical importance.

Among the total set of due-date-related rules (11)–(21), only DD and (19) ODD are time-independent. Concerning the time-dependent rules, some authors discuss the possibility of varying rescheduling frequencies [25, 99, 56, 4, 96], which points out a trade-off between improved performance and the cost in terms of scheduling effort due to real-time-based rescheduling procedures.

While both (13) SL and (14) CR tend to reflect a job's urgency in a more appropriate manner than this is done by the simple allowance, they modify DD (or ALL) in their own ways, SL referring to the *difference* and CR referring to the *fraction* of a job's allowance and its remaining work. CR might possibly be the more appealing version, because a "critical" job is defined as one with CR < 1 [11].

CR is an allowance per remaining work expression. Hence, CR and (15) ALL/OPN differ from one another by their denominators, i.e. remaining work and remaining number of operations, respectively. They are explicitly compared with one another in [136, 88, 92]. CR and ALL/OPN belong to the class of time-dependent ratio type rules, which moreover comprises (16) S/OPN, (17) S/WKR, and (18) S/ALL. Adam et al. [3] have pointed out an anomaly in the behavior of those approaches in case of a negative numerator: For a negative slack, S/OPN, e.g. does not reflect the urgency of a job as intended, since a job with many operations left is not considered as urgent as another job with the same negative slack but with only few remaining operations. Kanet [79] has given the correct solution to this dilemma by replacing the ratio by a product of slack and number of operations remaining in the case of a negative slack (see similar suggestions [60; 84; 70, p. 115; 121]).

S/ALL has been proposed by Miyazaki [88]. Baker et al. [15] show the "critical ratio"-logic of this rule: S/ALL = 1 − 1/CR.

The idea of internally set operation due-dates has been suggested by Conway et al. [36]. Operation due-dates serve as "milestones" that pace the work progress through the shop. (The question, how to set those milestones, is treated in Sect. 3.4.) Baker [11] introduces an operation-based CR-version, i.e. (21) OCR, while ODD and (20) OSL were studied already earlier [36, p. 231; 85; 96].

The group of rules based on global shop information (class 1.2 in Table 1) extends the sequencing decision horizon by anticipating machines the jobs in question will proceed to next, and by anticipating jobs that are expected to arrive soon at the machine in question. The first case comprises rules (22) NINQ, (23) WINQ, and (24) XWINQ, the latter case rule (25) LA.

The idea of NINQ, WINQ, and XWINQ is to give preference to jobs that would move on to queues with the least backlog, rather than to speed-up a job now that will be stopped in a congested queue after [36, p. 223]. WINQ and NINQ are related to one another like LWKR to FOPNR or like MWKR to GOPNR, i.e. WINQ allows for the weights given by the processing times of waiting operations. XWINQ, at the cost of a complex implementation effort, estimates the next queue backlog by the moment the job in question enters this next queue.

The rationale of LA on the other hand takes into account a rush job currently being processed on its predecessor machine; the urgency of such a job may require to book a machine-reservation in advance, even at the expense of introducing a deliberate idle period on that machine until then [60]. Of course, the "Insert"-option allows for other jobs that can be fitted into the idle time gap.

A last global shop information-based approach reported in the literature is one which takes into account the total number of waiting jobs in all queues of the shop, which is performed by the "processing-time-factor" [85] (see also [36, p. 235]). This information is used to control the weight of the SPT-rule within an additive combination of several components.

Finally, we are aware of only one example of rules in class 2 of Table 1 ("... attributes both of the job in question and of other jobs"), i.e. (26) ALTOP by Gere [60]. While Gere is not explicit enough in formalizing this rule, the definition of $Z_i(t)$ given in Table 2 assumes the objective function of minimizing the total tardiness of all jobs. Yet, one should point out that this evaluation approach is based at best on a rough *estimate* of the performance, since a job is considered as critical, only if its slack has become irreversibly negative at the moment of the loading decision and, hence, its delay is unavoidable.

The use of value-based rules rather than time-based ones is suggested in some investigations that introduce cost-oriented performance measures [5, 6, 74, 106, 119, 120, 22]. Besides expressions like "highest value added" (s. above), sales-based rules like "the most profitable job" or "the job with the highest selling price" are mentioned [74, 106, 119, 120]. Furthermore, the "expected delay cost" of a job is used as selection principle [106, 72], where the estimation of tardiness cost refers to a comparison between a regular, predetermined milestone-based work progress and the actual job status.

The idea to combine simple priority terms to more complex rules opens a field of great heuristic variety. In fact, it might be reasonable to link any of the introduced basic expressions, which would produce an immense number of possible priority functions. Rather than to survey specific combinations that are suggested in the literature, we characterize shortly the combination mechanisms.

Two basic procedures are standard: The additive and the alternative combination. (Ratios or products of rules are classified under simple rules, e.g. S/OPN or CR, s. above.) The *additive* or weighted combination determines the priority by an expression $Z_i(t) = \sum_{f=1}^{g} \alpha_f Q_{fi}$, where Q_{fi} denotes the priority value of the simple priority rule f $(f = 1, \ldots, g)$ for job i and α_f denotes the weighting or coefficient of rule f $(\alpha_f \geqslant 0)$ [73]. The special case of $0 \leqslant \alpha_f \leqslant 1$ and $\sum_{f=1}^{g} \alpha_f = 1$, i.e. a convex or linear combination, is the most usual one. One dominant example we will refer to in Sect. 3.4 too, is the linear combination $\alpha \cdot \mathrm{SPT} + (1 - \alpha) \cdot \mathrm{S/OPN}$ [see e.g. 36].

Another specific version of an additive combination is given by Oldziey [36, p. 235; 85]. He flexibilizes the weight α_f of a component within the combined expression according to current shop conditions.

The *alternative* or hierarchical approach is based on a conditional procedure. Typically, only two simple rules are combined in such an expression. Three ways of alternative combinations can occur that are shown for the two basic rules SPT and S/OPN:

– First, the decision for one out of the two rules may depend on the value of the job's slack:

$$Z_i(t) = \begin{cases} \text{SPT (slack} \leqslant 0) \\ \text{S/OPN (slack} > 0) \end{cases}$$

(A similar version is the "earliest modified due-date" rule, which is based on the original due-date or the earliest finish-time, whichever is greater [13, 14, 11]; see moreover [81].)

– Second, the selection of a rule may depend on a comparison of the numerical priority values of both rules, e.g. [100]: $Z_i(t) = \min \{\text{SPT} + r; \text{S/OPN}\}$ $(-\infty \leqslant r \leqslant \infty)$

– Third, one dominant rule is applied, but another rule is used to resolve ties; i.e. S/OPN might be used regularly but if more than one job have the same highest priority, SPT works as tie-breaker.

A recent approach from O'Grady et al. [63] formalizing the priority rule concept, may conclude the survey. The importance of that article results on one hand from the fact that a generalized additive combination expression is suggested covering a couple of simple rules with local queue information (class 1.1 in Table 1). The following notation is introduced:

- $\underline{c}, \underline{b}$: coefficient vector of the remaining processing times of a job
- $\underline{p}_i$: vector which contains the remaining processing times for job i
- s: coefficient of the due-date of a job
- y,z: coefficient of the number of remaining operations of a job

O'Grady et al. define the priority function:

$$Z_i(t) = (\underline{c} \times \underline{p}_i) + (sa_i)$$

(where $\underline{c} \times \underline{p}_i = \sum_{q=j}^{m_i} c_q p_{iq}$). Several rules are contained as special cases within this general framework, e.g.:

SPT: $\underline{c} = (1, 0, \ldots, 0)$; $s = 0$;

ALL (or DD): $\underline{c} = (0, \ldots, 0)$; $s = 1$;

SL: $\underline{c} = (-1, \ldots, -1)$; $s = 1$;

LWKR: $\underline{c} = (1, \ldots, 1)$; $s = 0$.

If one extends the approach of O'Grady et al. to the following priority function

$$Z_i(t) = \frac{(\underline{c} \times \underline{p}_i) + (sa_i) + y(m_i - j + 1)}{(\underline{b} \times \underline{p}_i) + z(m_i - j + 1)}$$

even a broader range of rules can be expressed, e.g.:

SPT: $\underline{c} = (1, 0, \ldots, 0)$; $\underline{b} = (0, \ldots, 0)$;
$s = y = 0; z = 1/(m_i - j + 1)$;

SL: $\underline{c} = (-1, \ldots, -1)$; $\underline{b} = (0, \ldots, 0)$;
$s = 1; y = 0; z = 1/(m_i - j + 1)$;

CR: $\underline{c} = (0, \ldots, 0)$; $\underline{b} = (1, \ldots, 1)$;
$s = 1; y = z = 0$;

FOPNR: $\underline{c} = (0, \ldots, 0)$; $\underline{b} = (0, \ldots, 0)$;
$s = 0; y = 1; z = 1/(m_i - j + 1)$;

LWKR: $\underline{c} = (1, \ldots, 1)$; $\underline{b} = (0, \ldots, 0)$;
$s = y = 0; z = 1/(m_i - j + 1)$;

S/OPN: $\underline{c} = (-1, \ldots, -1)$; $\underline{b} = (0, \ldots, 0)$;
$s = z = 1; y = 0$;

$\alpha\text{SPT} + (1 - \alpha)\text{S/OPN}$ $(0 \leqslant \alpha \leqslant 1)$:
$\underline{c} = ((m_i - j + 1)\alpha - (1 - \alpha), -(1 - \alpha), \ldots, -(1 - \alpha))$;
$\underline{b} = (0, \ldots, 0); s = 1 - \alpha; y = 0; z = 1$;

etc.

The idea of O'Grady et al. is also of major importance for quite another reason. Since the components of such a general combination do not represent *rules*, but *job properties*, arbitrary coefficients $\underline{c}$, $\underline{b}$, s, y, and z represent scheduling principles beyond heuristic priority rules. These principles belong to the area of "probabilistic dispatching" or Monte Carlo sampling procedures the application of which has been restricted so far to the static case [57; 36, pp. 121–129; 10, pp. 196–210]. The results of the experimental design of a sequence of simulation runs O'Grady et al. are concerned with, are very encouraging. At the same time, one should point out that this approach is equally accessible to traditional priority rule-based job shop scheduling.

3.4. Major Results of Priority Rule-Based Scheduling

From the output of the numerous simulation-based studies concerning the effectiveness of priority rules, some main results the representativeness of which seems to be confirmed through various investigations, are shortly discussed here. The basic research by Conway et al. [35, 33, 34, 37, 36], a pioneering and still today perhaps one of the most comprehensive experimental studies [38, p. 20], has dealt intensively with elementary processing and due-date attributes-oriented rules as well as with more sophisticated refinements. The majority of the following conclusions refers mainly to their work.

One major result is the remarkable efficiency of the SPT rule. Almost unanimously, the dominance of SPT with respect to the mean measures ($\bar{F}$, $\bar{L}$, $\bar{T}$) and with respect to f_t is pointed out. However, this advantage involves the evident disadvantage of prohibitively great flow-times and delays of individual jobs. Modifications of SPT in such a way as to overcome its weak points, have led to truncated versions of SPT, i.e. to alternative combinations with FCFS. On one hand, one has placed an upper bound on an operation's waiting-time. On the other hand, one has determined a lower bound for a queue's backlog. Thus, for jobs waiting for a long time and for queues with light load levels, SPT is replaced by FCFS. The corresponding simulation results exhibit obvious trade-offs between the variances and the means of flow-times and delays within the limits of the extreme cases (simple SPT rule, or simple FCFS rule, respectively) [36, p. 226].

While due-date-based rules, in particular SL and S/OPN, perform significantly worse than SPT with respect to the various means, they outperform SPT with respect to the tardiness and lateness variances and, under specific conditions, even with respect to the fraction of tardy jobs. An evident conclusion would be to combine both types of rules. While Conway et al. report good results for a linear combination of SPT and S/OPN [36, p. 233], Eilon et al. test an alternative combination of a SL-similar component (for tardy jobs) and SPT (for jobs ahead of schedule) [46]. Correspondingly, Oral et al. use an alternative combination version in the form $Z_i(t) = \min\{\text{SPT} + r; \text{S/OPN}\}$ $(-\infty \leqslant r \leqslant \infty)$, which involves, as expected, an increase of the tardiness variance and a decrease of $\bar{T}$ and f_t with increasing SPT-weights (decreasing r) [100].

Another well-confirmed result is that SPT is significantly less sensitive to shop load level variations than slack-based approaches [36, p. 233; 100; 112]. Baker [11] explains this behaviour with the "anti-SPT"-logic that is inherent in slack-expressions: Among two jobs with equal allowances, SL gives preference to the one with longer processing times left, which corresponds to an LPT idea (see also [129]). A strategy to cope with load fluctuations is to reinforce the SPT-weight during periods of high congestions and to weaken it under light load levels [36, p. 233; 85].

Carroll [30] has proposed a specific way to overcome the anti-SPT-behaviour of slack-based rules. His COVERT or "c over t"-rule favors jobs according to the risk to become tardy (slack-intent); but among two jobs, both with zero slack, the one with smaller processing times is preferred (SPT-intent). Recently, Vepsalainen et al. have modified the COVERT-rule with remarkable success [129].

The question of scheduling performance under various due-date conditions has been studied rather thoroughly. In particular, the due-date tightness, i.e. the mean allowance assigned to an arriving job, and the method of assigning due-dates are two crucial variables. Once more, SPT exhibits an evident robustness with respect to due-date tightness [36, 47]. Conway et al. were among the first to discuss the influence of various modes of due-date setting on sequencing effectiveness. They distinguish externally set dates (by means of random or constant allowances) from internally set ones (by means of job characteristics as the total work content or the total number of operations of a job). Due-dates on the basis of the total work content exhibit the best scheduling results, whereas all rules, with the exception of SPT, worsen, as the total work content method is replaced [36, p. 232]. Other authors deal with sequencing performances under current shop status-based due-dates, e.g. in terms of the workload [45, 130, 88, 12, 20, 15].

Additionally, Baker [11] introduces different ways to determine operation due-dates. Among the possible combinations of job and operation due-date setting modes, the total work content approach in both dimensions significantly outperforms all other methods. A consequence is that the operation due-date-based priority rules (OCR, ODD, OSL, etc.) can demonstrate their efficiency, if job due-dates and operation-milestones are set reasonably, i.e. related to the contained total work of a job or an operation.

Recent shop management systems consider the releasing of jobs to the shop more crucial than the dispatching of jobs [81, 135, 134]. Those systems might be characterized as workload-oriented approaches which limit the job release for purpose of flow-time and work-in-process inventory control. Under such a backlog reduction, different dispatching priority rules obviously do not contrast with one another in a significant way, while under an arrival-time-oriented, instantaneous job release the shop management efficiency depends mainly on the sequencing decision [82, 1, 2].

Summarizing, one has to point out once more again the performance and, in particular, the robustness of SPT. Finally, its advantages are obviously confirmed even under uncertainty of processing time prediction. Errors of processing time estimates do not present a serious problem for SPT. Of course, its performance worsens slightly with increasing uncertainty, but not as much as it is the case for other rules [36, p. 228; 96; 53; 54]. Hence, it seems that SPT remains a "mainstay" and a standard against which candidate procedures must demonstrate their virtue [38, 129].

4. Selected Extended Models of Dynamic Job Shop Scheduling

The relaxation of some of the assumptions (introduced in Sect. 1) concerning the job shop scheduling process leads to models for which more complex and more specific priority rules expressions might be adequate. The assumptions (6) ("no sequence-dependent setup-times") and (7) ("no assembly") describe conditions of the job and the manufacturing process, while the assumptions (5) ("no lack of operator, tool, or material") and (8) ("no alternate routing") restrict the shop management decision system to sequencing policies. Dealing briefly with the removal of these assumptions, we first discuss models of more complex shop conditions (i.e. sequence-dependent rather than sequence-independent setup-times, and tree-structured jobs rather than strictly-ordered sequences of operations). Second, we consider the interaction of the sequencing decision with other shop managerial policies (i.e. machine and labor (dual resource) constrained shops rather than machine limited systems, and alternate machine selection rather than fixed job routings).

The case of scheduling under *sequence-dependent setup-times* (including group technology-based characteristics of part family production [92]) were mainly studied by Baker [9] (see also [75, 56]). Typical priority rules under these conditions are the "shortest setup-time"-rule (MINSEQ) which can be considered as a dynamic version of the static heuristic "closest-unvisited-city"-algorithm, the "shortest sum of machine and setup-time"-rule, and the classical SPT-rule which has to be interpreted in this model environment as "shortest machine-time"-rule. All of these rules yield more or less the same remarkable results with respect to $\bar{F}$, light shop load levels given. Contrarily, Baker found significant relative improvements of $\bar{F}$ under heavy load levels by applying the job sequence of the static optimal solution (FIXSEQ). Equally, under all levels of work congestion, FIXSEQ was clearly best concerning the flow-time variance.

Tree-structured jobs are characteristic of the assembly-shop. The processing of the parts of an assembly-type job is restricted by precedence-constraints which require that a given item of the job cannot be operated on before the completion of its partner-item. To allow for serial-parallel-routed shops (rather than serial-routed ones) means to deal with staging delays which result from already finished parts waiting for the completion of their partners within the same assembly group.

A priority dispatching discipline focussing on minimizing these staging delays should try to reduce the differences of the completion-times for the various component parts which go into one assembly. This means that an optimal procedure with respect to $\bar{L}$ or $\bar{T}$ of the network-structured job, i.e. one allowing for both minimum queuing and minimum staging delays, has to look for completion of the branches of an assembly unit, not only *on time*, but also *at the same time*. The rationale of such a decision rule demands the continual comparison concerning the work progress over the assembly components. This procedure is referred to as "synchronization" meaning to speed up parts with lagging remaining work.

While many articles deal with priority rule-based scheduling of assembly-type jobs [30, 128, 103, 105, 68, 69, 127, 17, 112, 61, 76, 23], only few authors investigate assembly-specific synchronizing rules, in particular Maxwell [84] and Maxwell et al. [85] (see also [123, 70, 71, 121]). A synchronizing rule selects the job with the largest lag or the smallest advance, respectively, of the corresponding part with respect to its other partner-components, where "lag" and "advance" may refer to remaining work differences, or remaining number of operation differences [84, 85], or finally to slack differences [70, 121].

Synchronizing procedures, when used as single-factor rules, do not result in satisfactory performance, due to the fact that they focus on the relative work progress of a part rather than on the overall delay structure of the job. But when used with modest weight within an additive combination with basic due-date-related terms, they provide remarkable improvements.

The dispatching decision in an assembly-shop is most similar to the resource-constrained project scheduling (RCPS) problem. A slack-difference-based synchronization rule can be shown to correspond to a "least total slack"-heuristic in the RCPS case, if the RCPS-critical path-analysis is revised and updated at each activity completion [10, 121].

Dual resource constrained shops reflect a semi-automated real-world environment, where a given number of workers (l) operate a given number of machines (m) ($l < m$), and where hand-times are performed manually, while machine-times are performed automatically. Since one operator is responsible for more than one machine, the problem of assigning labor to machines occurs in addition to the job dispatching problem. Thus, the sequencing decision interacts with the labor allocation decision. Pioneering studies are the articles from Nelson [98] and Weeks et al. [131] (see also [41, 130, 114, 118]).

Labor assignment criteria that are adequate to the dual-constrainedness might be e.g. the hourly cost rate of the machine waiting for an operator, or its backlog [41, 114]. A job dispatching priority rule specific for a dual constrained shop might be the one that gives preference to the job with the smallest ratio of hand-time to machine-time [41].

The *alternate machine selection* allows for a routing flexibility, where the assignment of a given operation to a machine is not fixed. Routing flexibility involves the use of multi-purpose machines that may serve as alternate capacities. The policy by which the sequencing policy is affected, is the loading or machine-assignment decision [36, p. 239; 125; 26; 72]. The alternate machine selection might focus on machines with small backlog, eventually at the expense of a higher processing cost rate [36, p. 239; 72].

In particular, in new computer-based manufacturing systems, e.g. FMS's, the interaction of machine loading and job dispatching is evident. Pioneering studies, such as [125], deal with the assignment of work stations to operations (loading) in combination with priority rules (sequencing). Stecke et al. found the maximum production rate under a discipline of pooling the machining centers into one single group and under a sequencing rule, where each operation is inversely weighted by the total processing time of the workpiece.

Other approaches, e.g. [126], point out the complexity of the scheduling problem in a FMS environment: job dispatching and machine selection policies are interrelated with some other decisions, such as the assignment of tools, pallets, and fixtures to machines, or the selection and the routing of transportation facilities, e.g. automated guided vehicles. Yet, such a multi-dimensional shop management problem is beyond the scope of this priority rule-based scheduling survey [137].

5. Future Research

The classification of priority rules (Table 1) has evidenced that future research on heuristic dynamic job shop scheduling should be directed towards the more sophisticated cases of global shop information and the evaluation of performance criteria. This might also include an interaction of both approaches.

Another conclusion concerns priority rule combinations. Following O'Grady et al. [63], probabilistic dispatching mechanisms should be connected with heuristic scheduling procedures. This implies that the experimental design of successive simulation runs becomes an integral part of the process of priority rule formulation and application.

Last but not least, the detailed investigation of extended models of shop managerial decision making, where sequencing interacts with other shop policies such as lot-sizing, lot-splitting, alternate machine selection, overtime usage, work subcontracting etc., will get increasing importance, in particular in view of the progress of computer-integrated manufacturing systems.

Acknowledgement. The author would like to thank two anonymous referees for helpful comments and valuable suggestions concerning an earlier version of this paper.

References

1. Adam D (1987) Ansätze zu einem integrierten Konzept der Fertigungssteuerung bei Werkstattfertigung. In: Adam D (Hrsg) Neuere Entwicklungen in der Produktions- und Investitionspolitik. Festschrift zum 60. Geburtstag von Herbert Jacob. Gabler, Wiesbaden, pp 17–52
2. Adam D (1988) Die Eignung der belastungsorientierten Auftragsfreigabe für die Steuerung von Fertigungsprozessen mit diskontinuierlichem Materialfluß. Z Betriebsw 58:98–115
3. Adam N, Bertrand JWM, Surkis J (1984) Effective priority rules related to due date performance in multi-level assembly jobs. Paper presented at the TIMS XXVI meeting, Copenhagen, TB 11.1
4. Adam N, Surkis J (1980) Priority update intervals and anomalies in dynamic ratio type job shop scheduling rules. Manag Sci 26:1227–1237
5. Aggarwal SC, McCarl BA (1974) The development and evaluation of a cost-based composite scheduling rule. Nav Res Log Q 21:155–169
6. Aggarwal SC, Wyman FP, McCarl BA (1973) An investigation of a cost-based rule for job-shop scheduling. Int J Prod Res 11:247–261
7. Arumugam V, Ramani S (1980) Evaluation of value time sequencing rules in a real world job shop. J Op Res Soc 31: 895–904
8. Baker CT, Dzielinsky BP (1960) Simulation of a simplified job shop. Manag Sci 6:311–323
9. Baker KR (1968) Priority dispatching in the single channel queue with sequence-dependent setups. J Ind Eng:203–206
10. Baker KR (1974) Introduction to sequencing and scheduling. Wiley, New York
11. Baker KR (1984) Sequencing rules and due-date assignments in a job shop. Manag Sci 30:1093–1104
12. Baker KR, Bertrand JWM (1981) An investigation of due-date assignment rules with constrained tightness. J Op Manag 1:109–120
13. Baker KR, Bertrand JWM (1982) A dynamic priority rule for sequencing against due dates. J Op Manag 3:37–42
14. Baker KR, Kanet JJ (1983) Job shop scheduling with modified due dates. J Op Manag 4:11–22
15. Baker KR, Kanet JJ (1984) Improved decision rules in a combined system for minimizing job tardiness. Int J Prod Res 22:917–921
16. Berg CC (1979) Prioritätsregeln in der Reihenfolgeplanung. In: Kern W (Hrsg) HWProd, Poeschel, Stuttgart, pp 1425–1433

17. Berr U, Tangermann HP (1976) Einfluß von Prioritätsregeln auf die Kapazitätsterminierung der Werkstattfertigung. Wt-Z Ind Fertig 66:7–12
18. Berry WL, Finley RA (1976) Critical ratio scheduling with queue waiting information: an experimental analysis. AIIE-Transact 8:161–168
19. Berry WL, Rao V (1975) Critical ratio scheduling: an experimental analysis. Manag Sci 22:192–201
20. Bertrand JWM (1983) The effect of workload dependent due-dates on job shop performance. Manag Sci 29:799–816
21. Biendl P (1984) Ablaufsteuerung von Montagefertigungen. Haupt, Bern Stuttgart
22. Biggs JR (1979) Heuristic lot-sizing and sequencing rules in a multistage production-inventory system. Dec Sci 10: 96–115
23. Biggs JR (1985) Priority rules for shop floor control in a material requirements planning system under various levels of capacity. Int J Prod Res 23:33–46
24. Blackstone JH Jr, Phillips DT, Hogg GL (1982) A state-of-the-art survey of dispatching rules for manufacturing job shop operations. Int J Prod Res 20:27–45
25. Brown RG (1968) Simulations to explore alternative sequencing rules. Nav Res Log Q 15:281–286
26. Browne J, Davies BJ (1984) The design and validation of a digital simulation model for job shop control decision making. Int J Prod Res 22:335–357
27. Bulkin MH, Colley JL, Steinhoff HW (1967) Load forecasting, priority sequencing, and simulation in a job shop control system. Manag Sci 13:B29–B51
28. Bunnag P, Smith SB (1985) A multifactor priority rule for jobshop scheduling using computer search. IIE-Transact 17:141–146
29. Buzi K (1971) Die Simulation optimaler Prioritätsregeln für die Ablaufplanung in der Werkstattfertigung. Westdeutscher Verlag, Köln Opladen
30. Caroll DC (1965) Heuristic sequencing of single and multiple component jobs. PhD thesis, MIT, Cambridge/Mass
31. Cheng TCE (1986) Optimal due-date assignment in a job shop. Int J Prod Res 24:503–515
32. Colley JL Jr (1968) Principles for scheduling job shops. Syst Proced J 19, No 4
33. Conway RW (1965) Priority dispatching and work-in-process inventory in a job shop. J Ind Eng 16:123–130
34. Conway RW (1965) Priority dispatching and job lateness in a job shop. J Ind Eng 16:228–237
35. Conway RW, Maxwell WL (1962) Network dispatching by the shortest operation discipline. Oper Res 10:51–73
36. Conway RW, Maxwell WL, Miller LW (1967) Theory of scheduling. Addison-Wesley, Reading/Mass
37. Conway RW, Maxwell WL, Oldziey JW (1966) Sequencing against due-dates. In: Hertz DB, Melèse J (eds) Proceedings of the Fourth International Conference on Operational Research. Wiley, New York, pp 599–617
38. Day JE, Hottenstein MP (1970) Review of sequencing research. Nav Res Log Q 17:11–39
39. Deane RH, Moodie CL (1972) A dispatching methodology for balancing workload assignments in a job shop production facility. AIIE-Transact 4:277–283
40. Dörken W (1973) Simulationsmodelle und ihre Anwendung bei der Analyse von Prioritätsregeln zur Maschinenbelegungsplanung. Arbeitsvorb 10:89–95, 115–122
41. Dornia M (1976) Aufbau- und ablauforientierte Untersuchungen an einem Simulationsmodell der Mehrstellenbedienung. Dissertation, Köln
42. Eilon S (1973) Research note: The job shop simulation programme. Int J Prod Res 11:299–300
43. Eilon S (1979) Production scheduling. In: Haley KB (ed) OR '78. North-Holland, Amsterdam, pp 237–266
44. Eilon S, Chowdhury IG (1976) Due dates in job shop scheduling. Int J Prod Res 14:223–237
45. Eilon S, Chowdhury IG, Serghiou SS (1975) Experiments with the SI rules in job-shop scheduling. Simulation 24: 45–48
46. Eilon S, Cotterill DJ (1968) A modified SI rule in job shop scheduling. Int J Prod Res 7:135–145
47. Eilon S, Hodgson RM (1967) Job shops scheduling with due dates. Int J Prod Res 6:1–13
48. Ellinger T (1976) Reihenfolgeplanung. In: Grochla E, Wittmann W (eds) HWB 4. Aufl, Poeschel, Stuttgart, pp 3412–3420
49. Ellinger T, Schmitz P, Goldbecker H, Haupt R, Oellers B (1978) Simulationssystem für Reihenfolgeprobleme der industriellen Fertigung. Rationalisierung 29:17–21, 43–45
50. Elmaghraby SE (ed) (1973) Symposium on the theory of scheduling and its applications. Springer, Berlin Heidelberg New York
51. Elvers DA (1973) Job shop dispatching using various due date setting criteria. Prod Inv Manag 14:62–69
52. Elvers DA (1974) The sensitivity of the relative effectiveness ob job shop dispatching rules with respect to various arrival distributions. AIIE-Transact 6:41–49
53. Elvers DA, Taube LR (1983) Deterministic/stochastic assumptions in job shops. Eu J Oper Res 14:89–94
54. Elvers DA, Taube LR (1983) Time completion for various dispatching rules in job shops. Omega 11:81–89
55. Fabrycky WJ, Shamblin JE (1966) A probability based sequencing algorithm. J Ind Eng 17:308–312
56. Farn CK, Muhlemann AP (1979) The dynamic aspects of a production scheduling problem. Int J Prod Res 17:15–21
57. Fischer H, Thompson GL (1963) Probabilistic learning combinations of local job shop scheduling rules. In: Muth JF, Thompson GL (eds) Industrial scheduling, Prentice-Hall, Englewood Cliffs, chap 15
58. Fleischmann B (1987) Operations-Research-Modelle und -Verfahren in der Produktionsplanung (Übersichtsvortrag). In: Isermann H et al. (eds) Operations Research Proceedings 1986. Springer, Berlin Heidelberg New York, pp 278–285
59. French S (1982) Sequencing and scheduling. An introduction to the mathematics of the job-shop. Horwood, Chichester
60. Gere WS (1966) Heuristics in job shop scheduling. Manag Sci 13:167–190
61. Goodwin JS, Goodwin JC Jr (1982) Operating policies for scheduling assembled products. Dec Sci 13:585–603
62. Goodwin JS, Weeks JK (1986) Evaluating scheduling policies in a multi-level assembly system. Int J Prod Res 24:247–257
63. O'Grady PJ, Harrison C (1985) A general search sequencing rule for job shop sequencing. Int J Prod Res 23:961–973
64. Gräßler D (1968) Der Einfluß von Auftragsdaten und Entscheidungsregeln auf die Ablaufplanung von Fertigungsstraßen. Dissertation, TH Aachen
65. Graves SC (1977) A note on "Critical ratio scheduling: an experimental analysis" (Note). Manag Sci 23:1358–1359
66. Graves SC (1981) A review of production scheduling. Oper Res 29:646–675
67. Green GI, Appel LB (1981) An empirical analysis of job shop dispatch rule selection. J Oper Manag 1:197–203
68. Hauk W (1972) Beitrag zur Lösung des Reihenfolgeproblems bei der Auftragsplanung. Z Wirtsch Fert 67:45–49

69. Hauk W (1973) Einplanung von Produktionsaufträgen nach Prioritätsregeln. Eine Untersuchung von Prioritätsregeln mit Hilfe der Simulation. Beuth, Berlin Köln Frankfurt
70. Haupt R (1977) Reihenfolgeplanung im Sondermaschinenbau. Gabler, Wiesbaden
71. Haupt R (1978) Simulationsuntersuchungen zur Reihenfolgeplanung unter dem Einfluß von Vereinheitlichungs-Elementen in der Produktion. In: Brockhoff K et al. (eds) Proceedings in Operations Research 7. Physica, Würzburg Wien, pp 247–256
72. Haupt R (1987) Produktionstheorie und Ablaufmanagement. Zeitvariable Faktoreinsätze und ablaufbezogene Dispositionen in Produktionstheorie- und -planungs-Modellen. Poeschel, Stuttgart
73. Hershauer JC, Ebert RJ (1975) Search and simulation selection of a job shop sequencing rule. Manag Sci 21:833–843
74. Hoffmann TR, Scudder GD (1983) Priority scheduling with cost considerations. Int J Prod Res 21:881–889
75. Hollier RH (1986) A simulation study of sequencing in batch production. Oper Res Q 19:389–407
76. Huang PY (1984) A comparative study of priority dispatching rules in a hybrid assembly job shop. Int J Prod Res 22:375–387
77. Jackson JR (1957) Simulation research on job shop production. Nav Res Log Q 4:287–295
78. Jones CH (1973) An economic evaluation of job shop dispatching rules. Manag Sci 20:293–307
79. Kanet JJ (1982) On anomalies in dynamic ratio type scheduling rules: a clarifying analysis. Manag Sci 28: 1337–1341
80. Kanet JJ, Hayya JC (1982) Priority dispatching with operation due-dates in a job shop. J Oper Manag 2:155–163
81. Kettner H, Bechte W (1981) Neue Wege der Fertigungssteuerung durch belastungsorientierte Auftragsfreigabe. VDI-Z 123:459–465
82. Knolmayer G, Holdhof J (1986) BORA versus DORA: Ein Vergleich zweier Terminplanungsverfahren. Arbeitsvorbereitung 23:163–165
83. Le Grande E (1963) The development of a factory simulation system using actual operating data. Manag Tech 3: 1–19
84. Maxwell WL (1969) Priority dispatching and assembly operations in a job shop. The RAND Corp Memorandum RM-5370-PR, Santa Monica/Cal
85. Maxwell WL, Mehra M (1968) Multiple-factor rules for sequencing with assembly constraints. Nav Res Log Q 15:241–254
86. Mellor P (1966) A review of job shop scheduling. Oper Res Q 17:161–171
87. Miller LW, Ginsberg AS, Maxwell WL (1975) An experimental investigation of priority dispatching in a simple assembly shop. In: Geisler MA (ed) Logistics, TIMS Studies in the Management Sciences, vol 1. North-Holland, Amsterdam, pp 125–142
88. Miyazaki S (1981) Combined scheduling system for reducing job tardiness in a job shop. Int J Prod Res 19:201–211
89. Moodie CL, Novotny DJ (1968) Computer scheduling and control systems for discrete part production. J Ind Eng 19:336–341
90. Moodie CL, Roberts SD (1968) Experiments with priority dispatching rules in a parallel processor shop. Int J Prod Res 6:303–312
91. Moore JM, Wilson RC (1967) A review of simulation research in job shop scheduling. Prod Inv Manag 8:1–10
92. Mosier CT, Elvers DA, Kelly D (1984) Analysis of group technology scheduling heuristics. Int J Prod Red 22:857–875
93. Müller A (1987) Produktionsplanung und Pufferbildung bei Werkstattfertigung. Gabler, Wiesbaden
94. Müller G (1978) Job-Shop-Simulation auf dem Minicomputer: Ein Instrument zur Reihenfolge- und Terminplanung in der Textillohnveredelungsindustrie. Dissertation, Universität Zürich
95. Müller-Merbach H (1979) Ablaufplanung, Optimierungsmodelle zur. In: Kern W (Hrsg) HWProd, Poeschel, Stuttgart, pp 38–52
96. Muhlemann AP, Lockett AG, Farn CI (1982) Job shop scheduling heuristics and frequency of scheduling. Int J Prod Res 20:227–241
97. Nanot YR (1963) An experimental investigation and comparative evaluation of priority disciplines in job shop-like queuing networks. Manag Sci Res Proj, UCLA, Report No 87–1963
98. Nelson RT (1967) Labor and machine limited production systems. Manag Sci 13:648–671
99. New CC (1975) Job shop scheduling: is manual application of dispatching rules feasible? Oper Res Q 21:35–43
100. Oral M, Malouin JL (1973) Evaluation of the shortest processing time scheduling rule with truncation process. AIIE-Transact 5:357–365
101. Ow PS (1985) Focused scheduling in proportionate flowshops. Manag Sci 31:852–869
102. Pabst HJ (1985) Analyse der betriebswirtschaftlichen Effizienz einer computergestützten Fertigungssteuerung mit CAPOSS-E in einem Maschinenbauunternehmen mit Einzel- und Kleinserienfertigung. Lang, Frankfurt Bern New York
103. Pai AR, McRoberts KL (1971) Simulation research in interchangeable part manufacturing. Manag Sci 17:B732–B743
104. Panwalkar SS, Iskander W (1977) A survey of scheduling rules. Oper Res 25:45–61
105. Papendieck AJ (1971) Reihenfolgen und Losgrößen in der Serienfertigung, untersucht an einem praxisbezogenen Simulationsmodell. Dissertation TH Braunschweig
106. Paulik R (1984) Kostenorientierte Reihenfolgeplanung in der Werkstattfertigung. Haupt, Bern Stuttgart
107. Pinto PA, Khumawala BM, Susko JA (1983) Application of a heuristic scheduling rule for a tyre testing department. J Opl Res Soc 34:243–248, 910
108. Putnam AO, Everdell R, Dormann GH, Cronan RR, Lindgren LH (1971) Updating critical ratio and slack time priority scheduling rules. Prod Inv Manag 12:51–73
109. Ragatz GL, Mabert VA (1984) A framework for the study of due date management in job shops. Int J Prod Res 22: 685–695
110. Reiter S (1966) A system for managing job shop production. J Business 39:371–393
111. Rinnooy Kan AHG (1976) Machine scheduling problems. Classification, complexity and computations. Nijhoff, Den Haag
112. Rochette R, Sadowski RP (1976) A statistical comparison of the performance of simple dispatching rules for a particular set of job shop. Int J Prod Res 14:63–75
113. Rowe AJ (1960) Toward a theory of scheduling. J Ind Eng 11:125–136
114. Russell RS, Taylor BW (1985) An evaluation of scheduling policies in a dual resource constrained assembly shop. IIE-Transact 17:219–232

115. Sägesser R (1976) Analytische und heuristische Methoden zur Lösung des Reihenfolgeproblems mit besonderer Berücksichtigung der Werkstattfertigung. Dissertation, Sankt Gallen
116. Sammler A (1970) Diskussion von Vorrangregeln (Prioritäten) zur Lösung des Reihenfolgeproblems in der Planung und Lenkung der Einzelteilefertigung. Fertigungstechnik Betrieb 20:408–411
117. Scudder GD (1984) Priority scheduling and spares stocking policies for a repair shop: the multiple failure case. Manag Sci 30:739–749
118. Scudder GD (1986) Scheduling and labor-assignment policies for a dual-constrained repair shop. Int J Prod Res 24:623–634
119. Scudder GD, Hoffmann TR (1985) An evaluation of value based dispatching rules in a flow shop. E J Oper Res 22: 310–318
120. Scudder GD, Hoffmann TR (1985) Composite cost-based rules for priority scheduling in a randomly routed job shop. Int J Prod Res 23:1185–1195
121. Sculli D (1980) Priority dispatching rules in job shops with assembly operations and random delays. Omega 8: 227–234
122. Sen T, Gupta SK (1984) A state-of-art survey of static scheduling research involving due dates. Omega 12:63–76
123. Siegel GB (1971) An investigation of job shop scheduling for jobs with assembly constraints. PhD Thesis, Cornell University, Ithaca/NY
124. Sprotte HG (1970) Simultane Termin- und Reihenfolgeplanung bei Mehrproduktunternehmen mit Auftragsfertigung und stochastischem Auftragseingang. Dissertation, Universität Köln
125. Stecke KE, Solberg JJ (1981) Loading and control policies for a flexible manufacturing system. Int J Prod Res 19: 481–490
126. Subramanyam S, Askin RG (1986) An expert systems approach to scheduling in flexible manufacturing systems. In: Kusiak A (ed) Flexible manufacturing systems: Methods and Studies. North-Holland, Amsterdam, pp 243–256
127. Tangermann HP (1973) Auftragsreihenfolgen und Losgrößen als Instrument der Fertigungsterminplanung, untersucht an einem praxisbezogenen Simulationsmodell. Dissertation, TH Braunschweig
128. Trilling DR (1966) Job-shop simulation of orders that are networks. J Ind Eng 17:59–71
129. Vepsalainen APJ, Morton TE (1987) Priority rules for job shops with weighted tardiness costs. Manag Sci 33:1035–1047
130. Weeks JK (1979) A simulation study of predictable due-dates. Manag Sci 25:363–373
131. Weeks JK, Fryer JS (1976) A simulation study of operating policies in a hypothetical dual-constrained job shop. Manag Sci 22:1362–1371
132. Weeks JK, Fryer JS (1977) A methodology for assigning minimum cost due-dates. Manag Sci 23:872–881
133. Wegner N (1978) Simulation von Einplanungs- und Abfertigungsstrategien bei Werkstattfertigung. Dissertation, TU Hannover
134. Wiendahl HP (1987) Die belastungsorientierte Fertigungssteuerung. Oldenbourg, München Wien
135. Wiendahl HP, Buchmann W (1984) Realisierung alternativer Fertigungssteuerungsstrategien. VDI-Z 126:327–334
136. Worrall BM, Mert B (1980) Application of dynamic scheduling rules in maintenance planning and scheduling. Int J Prod Res 18:57–74
137. Zijm WHM (1988) Flexible Manufacturing Systems: background, examples, and models. In: Schellhaas H et al. (ed) Operations Research Proceedings 1987. Springer, Berlin Heidelberg New York, pp 142–161