Sequencing and scheduling : an annotated bibliography

Memorandum COSOR 97-02, 1997, Eindhoven University of Technology

Sequencing and Scheduling: an annotated bibliography

J. A. Hoogeveen, J. K. Lenstra
Eindhoven University of Technology, Eindhoven

S. L. van de Velde
University of Twente, Enschede

Contents

E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys (1993). Sequencing and scheduling: algorithms and complexity. S.C. Graves, P.H. Zipkin, A.H.G. Rinnooy Kan (eds.). *Logistics of Production and Inventory; Handbooks in Operations Research and Management Science, Vol. 4*, North-Holland, Amsterdam, 445-522.

Sequencing and scheduling is concerned with the optimal allocation of scarce resources to activities over time. The area is motivated by questions that arise in production planning, project scheduling, and computer control, and to a lesser extent in routing, personnel scheduling, maintenance scheduling, and materials handling. The models in this area are highly standardized: they concern the scheduling of n *jobs* (the activities) on m *machines* (the resources), which can process no more than one activity at a time, so as to optimize some function of the job completion times.

The specification of a machine scheduling problem requires the description of a *machine environment*, *job characteristics*, and an *optimality criterion*. The simplest environment is the *single machine*, on which jobs j, each consisting of a single operation, have to spend given processing times p_j $(j = 1, \ldots, n)$. In a *parallel machine* environment, job j has to spend a given time on any of m machines. These can be *identical*, in which case the machines operate at the same speed; *uniform*, in which case each machine has its own speed; and *unrelated*, in which case the speed of a machine is job-dependent. Open shops, flow shops, and job shops are m-machine environments in which each job consists of several operations, each of which has to be executed on a designated machine; no job can undergo more than one operation at a time. In *job shops*, the order in which the operations of a job have to be executed is fixed; in *flow shops*, the order is fixed and the same for all jobs; and in *open shops*, the order is free and hence up to the scheduler. A generalization of these multi-stage models is the situation in which there are parallel machines available at one or more stages.

The job characteristics include the possibility of allowing preemption, and of specifying precedence constraints, release dates and deadlines. If *preemption* is allowed, then an operation may be interrupted and resumed later on; otherwise, an operation, once started, must be processed till completion without interruption. A *precedence constraint* stipulates that a certain job cannot start before another one has been completed. Job availability may be restricted by imposing *release dates* r_j, before which the jobs j cannot be started, and *deadlines* $\bar{d}_j$, by which they have to be completed.

A *feasible schedule* is an allocation of the operations to time intervals on the machines such that all restrictions are met. The optimality criterion is usually a function of the job completion times $C_1, \ldots, C_n$. Common criteria are *maximum completion time* or *makespan* $C_{\max} = \max_j C_j$ and *total completion time* $\sum C_j$. If job j has a *due date* d_j, we can compute its *lateness* $L_j = C_j - d_j$, its *tardiness* $T_j = \max\{0, C_j - d_j\}$, and its *earliness* $E_j = \max\{0, d_j - C_j\}$, for any given schedule. Important criteria involving due dates are *maximum lateness* $L_{\max} = \max_j L_j$, *total tardiness* $\sum T_j$, *total earliness* $\sum E_j$, and the *number of tardy jobs* (i.e., with $C_j > d_j$). If each job j has a *weight* w_j, then we can also have weighted versions of these criteria.

Machine scheduling has been an active field for many years. In this chapter, we review the main contributions to the area after the completion of the chapter by Lawler et al. (1993), which offers a comprehensive survey up to November 1990, providing 378 references.

Compiling this bibliography, we have focused on *deterministic* scheduling problems, where all parameters involved are given; such problems belong to the area of *combinatorial optimization*. We have put emphasis on classical models, new and relevant models such as on-line scheduling and scheduling with communication delays, and new and interesting techniques like polyhedral combinatorics. To save space, we have clustered papers and referred to surveys when possible and appropriate.

We do not cover purely experimental comparisons of heuristic rules. Models that, in our evaluation, are too specific or too artificial have been deleted, along with models that have not quite caught on or that are already dying off. We have also deleted models that are methodologically too different, such as stochastic scheduling models. Of the many papers surviving, we offer a survey of the most noteworthy results.

Acknowledgements. We are grateful for the comments of David Shmoys and Gerhard Woeginger. This research was partially supported by NSF grant CCR-9307391.

1 Books

J. Błazewicz, K. Ecker, G. Schmidt, J. Węglarz (1993). *Scheduling in Computer and Manufacturing Systems*, Springer, Berlin.
T.E. Morton, D.W. Pentico (1993). *Heuristic Scheduling Systems*, Wiley, New York.
V.S. Tanaev, V.S. Gordon, Y.M. Shafransky (1994). *Scheduling Theory: Single-Stage Systems*, Kluwer, Dordrecht.
V.S. Tanaev, Y.N. Sotskov, V.A. Strusevich (1994). *Scheduling Theory: Multi-Stage Systems*, Kluwer, Dordrecht.
P. Brucker (1995). *Scheduling Algorithms*, Springer, Berlin.
P. Chrétienne, E.G. Coffman, Jr., J.K. Lenstra, Z. Liu (eds.) (1995). *Scheduling Theory and Its Applications*, Wiley, Chichester.
M.L. Pinedo (1995). *Scheduling: Theory, Algorithms, and Systems*, Prentice Hall, Englewood Cliffs, NJ.

This is a selection of the scheduling books that have appeared in the 1990's. Morton & Pentico convey an engineering approach to scheduling manufacturing systems. Tanaev et al. integrate the Soviet and Western literature. Chrétienne et al. collected 17 tutorials and surveys on a broad range of topics. Pinedo's book is an attractive undergraduate text, also covering stochastic models and practical applications.

2 Single machine

2.1 Regular criteria

S.K. Gupta, J. Kyparisis (1987). Single machine scheduling research. *Omega* 15, 207-227.

A notable survey, containing 204 references.

M.C. Fields, G.N. Frederickson (1990). A faster algorithm for the maximum weighted tardiness problem. *Inform. Process. Lett.* 36, 39-44.

An $O(k + n \log n)$ algorithm for the problem with deadlines and k precedence constraints.

L.A. Hall, D.B. Shmoys (1992). Jackson's rule for single-machine scheduling: making a good heuristic better. *Math. Oper. Res.* 17, 22-35.
L.A. Hall, D.B. Shmoys (1990). Near-optimal sequencing with precedence constraints. R. Kannan, W.R. Pulleyblank (eds.). *Proc. 1st MPS Conf. IPCO*, University of Waterloo Press, Waterloo, 249-260.

A 4/3-approximation algorithm and two approximation schemes for minimizing maximum lateness in case of release dates and negative due dates that run in time $O(n^2 \log n)$, $O(16^{1/\epsilon}(n/\epsilon)^{3+4/\epsilon})$, and $O(n \log n + n(4/\epsilon)^{8/\epsilon^2+8/\epsilon+2})$ — and a polynomial approximation scheme for the precedence-constrained case.

E. Balas, J.K. Lenstra, A. Vazacopoulos (1995). The one-machine problem with delayed precedence constraints and its use in job shop scheduling. *Management Sci.* 41, 94-109.

Carlier's (1982) algorithm for minimizing $L_{\max}$ subject to release dates is adapted

to handle *precedence delays*, i.e., minimum delays between precedence-related jobs. The method is used in the *shifting bottleneck* heuristic for job shop scheduling.

R. Ahmadi, U. Bagchi (1990). Lower bounds for single-machine scheduling problems. *Naval Res. Log. Quart.* 37, 967-979.

Proof that the preemptive bound dominates all proposed polynomial-time lower bounds for the problem of minimizing total completion time subject to release dates.

J. Du, J.Y.-T. Leung (1993). Minimizing mean flow time with release time and deadline constraints. *J. Algorithms* 14, 45-68.

An NP-hardness proof for the preemptive problem, along with the identification of some well-solvable cases.

J.A. Hoogeveen, S.L. van de Velde (1995). Stronger Lagrangian bounds by use of slack variables: applications to machine scheduling problems. *Math. Program.* 70, 173-190.

A generic method to improve Lagrangian lower bounds; applications to the problems of minimizing total weighted completion time subject to precedence constraints, total weighted tardiness, and total completion time in a two-machine flow shop.

H. Kellerer, T. Tautenhahn, G.J. Woeginger (1996). Approximability and nonapproximability results for minimizing total flow time on a single machine. *Proc. 28th Annual ACM Symp. Theory of Comput.*, 418-426.

A polynomial $O(n^{1/2})$-approximation algorithm for minimizing $\sum(C_j - r_j)$, and a lower bound of $O(n^{1/2-\epsilon})$ on the performance of any polynomial-time algorithm.

T.S. Abdul-Razaq, C.N. Potts, L.N. Van Wassenhove (1990). A survey of algorithms for the single machine total weighted tardiness scheduling problem. *Discr. Appl. Math.* 26, 235-253.

An insightful survey of dynamic programming and branch-and-bound algorithms.

B.C. Tansel, I. Sabuncuoglu (1994). *Geometry based analysis of single machine tardiness problem and implications on solvability*, Report IEOR-9405, Bilkent University.
B.C. Tansel, B. Kara-Yetis, I. Sabuncuoglu (1995). *Advances in solvability of the single machine total tardiness scheduling problem*, Report IEOR-9517, Bilkent University.

A method to distinguish between presumably easy and hard instances of the total tardiness problem, and the best existing code for this problem.

D.S. Hochbaum, R. Shamir (1991). Strongly polynomial algorithms for the high multiplicity scheduling problem. *Oper. Res.* 39, 648-653.
D.S. Hochbaum, R. Shamir (1990). Minimizing the number of tardy job units under release time constraints. *Discr. Appl. Math.* 28, 45-57.

If there are g groups of identical unit-time jobs, one only needs to specify one copy of the job data per group and the size of each group. The weighted number of late jobs can be minimized in $O(g \log g)$ time, in $O(g^2)$ time in case of release dates, and in $O(g \log g)$ time in case of release dates and equal weights.

C.N. Potts, L.N. Van Wassenhove (1992). Single machine scheduling to minimize total late work. *Oper. Res.* 40, 586-595.
A.M.A. Hariri, C.N. Potts, L.N. Van Wassenhove (1995). Single machine scheduling to minimize total weighted late work. *ORSA J. Comput.* 7, 232-242.
M.Y. Kovalyov, C.N. Potts, L.N. Van Wassenhove (1995). A fully polynomial approximation scheme for scheduling a single machine to minimize total weighted late work. *Math. Oper. Res.* 19, 86-93.

The *late work* of job j is the amount of processing done after d_j. The preemptive weighted problem is solvable in $O(n \log n)$ time. The nonpreemptive problems, with or without weights, are both NP-hard but solvable in pseudopolynomial time. For the weighted version, a branch-and-bound algorithm is given, and a fully polynomial approximation scheme that runs in time $O(n^3 \log n + n^3/\epsilon)$.

2.2 Regular criteria, polyhedral techniques

M. Queyranne, A.S. Schulz (1997). Polyhedral approaches to machine scheduling. *Math. Program. (B)*, to appear.

A comprehensive survey of the area, which was independently initiated by Balas, Queyranne, and Wolsey in the mid-1980's. Originally meant to provide strong lower bounds, polyhedral methods are recently being used as a tool to develop polynomial approximation algorithms with excellent performance guarantees. Depending upon the variables used, several approaches can be distinguished.

M. Queyranne, Y. Wang (1991). Single-machine scheduling polyhedra with precedence constraints. *Math. Oper. Res.* 16, 1-20. Erratum. *Math. Oper. Res.* 20 (1995), 768.

A formulation based on job completion times for the problem of minimizing total weighted completion time subject to precedence constraints, and a complete polyhedral characterization for the case of series-parallel constraints by two classes of inequalities, which are polynomially separable.

M.E. Dyer, L.A. Wolsey (1990). Formulating the single machine sequencing problem with release dates as a mixed integer program. *Discr. Appl. Math.* 26, 255-270.
J.P. Sousa, L.A. Wolsey (1992). A time indexed formulation of non-preemptive single machine scheduling problems. *Math. Program.* 54, 353-367.
J.M. van den Akker (1994). *LP-based solution methods for single-machine scheduling problems*, PhD Thesis, Eindhoven University of Technology. See also Reports COSOR 93-27, 95-24, 96-14, Dept. Math. & Comp. Sci., Eindhoven University of Technology.

Dyer & Wolsey analyze a time-indexed formulation, in which binary variables x_{jt} indicate if job j is started in time period t. It is stronger than other formulations but of pseudopolynomial size. Sousa & Wolsey derive a class of valid inequalities with right-hand side 1 and solve the separation problem in polynomial time. Van den Akker characterizes all facet-inducing inequalities with right-hand side 1 or 2 and solves the corresponding separation problem in polynomial time. She uses column generation to solve the LP-relaxation and presents a branch-and-cut algorithm.

J.B. Lasserre, M. Queyranne (1992). Generic scheduling polyhedra and a new mixed-integer formulation for single-machine scheduling. E. Balas, G. Cornuéjols, R. Kannan (eds.). *Proc. 2nd MPS Conf. IPCO*, University Printing and Publications, Carnegie Mellon University, Pittsburgh, 136-149.

A formulation in which the variables denote the start time of the jth job in the sequence, and its application to the problem of minimizing makespan in case of release dates and deadlines. The lower bound obtained is computationally competitive.

L.A. Hall, A.S. Schulz, D.B. Shmoys, J. Wein (1997). Scheduling to minimize average completion time: off-line and on-line approximation algorithms. *Math. Oper. Res.* 22, to appear.

A number of polynomial approximation algorithms with constant performance ratios based on the solution to an LP-relaxation. Completion time formulations are used for the problems of minimizing total weighted completion time subject to release dates and/or precedence constraints on a single machine and on identical parallel machines. The resulting bounds on the quality of the LP-relaxations are also valid for some other formulations, as these dominate the ones based on completion times in strength. A variant of the time-indexed formulation, using binary variables to indicate if a certain job completes in an interval of the form $(2^{k-1}, 2^k]$, is applied to the problem of minimizing $\sum w_j C_j$ on unrelated parallel machines subject to release dates.

2.3 Regular criteria, setup times

C.N. Potts, L.N. Van Wassenhove (1992). Integrating scheduling with batching and lot-sizing: a review of algorithms and complexity. *J. Oper. Res. Soc.* 43, 395-406.
S. Webster, K.R. Baker (1995). Scheduling groups of jobs on a single machine. *Oper. Res.* 43, 692-703.

In case of job setup times, the issue is to batch similar jobs to save setups, without delaying other jobs too much. Not surprisingly, most types of batching models are intractable. The above papers survey more of these than we consider here.

J.M.J. Schutten, S.L. van de Velde, W.H.M. Zijm (1996). Single-machine scheduling with release dates, due dates, and family setup times. *Management Science* 42, 1165-1174.
H.A.J. Crauwels, A.M.A. Hariri, C.N. Potts, L.N. Van Wassenhove (1997). Branch-and-bound algorithms for single machine scheduling with batch set-up times to minimize total weighted completion time. *Ann. Oper. Res.*, to appear.
A.M.A. Hariri, C.N. Potts (1997). Single machine scheduling with batch set-up times to minimize maximum lateness. *Ann. Oper. Res.*, to appear.
S. Zdrzałka (1992). *Analysis of an approximation algorithm for single-machine scheduling with delivery times and sequence independent batch setup times*, Manuscript, Inst. Eng. Cybernetics, Technical University of Wrocław.

Branch-and-bound algorithms for three single-machine problems complicated by setup times. The crux lies in the computation of good lower bounds. Schutten et al. handle setups as additional jobs, Crauwels et al. use Lagrangean bounds, while Hariri & Potts apply combinatorial bounds. For minimizing maximum lateness in case of negative due dates, the latter also present an $O(n \log n)$ time 5/3-approximation algorithm, while Zdrzałka gives an $O(n^2)$ time 3/2-approximation algorithm.

2.4 Irregular criteria

K.R. Baker, G.D. Scudder (1990). Sequencing with earliness and tardiness penalties. *Oper. Res.* 38, 22-37.

Irregular criteria are not monotone in the job completion times. Objective functions of the form $\sum(\alpha_j E_j + \beta_j T_j)$ have attracted much attention over the last ten years. Due to the intractability of problems with general due dates, most work assumes a due date d that is common to all jobs. The due date d is called *large* if $d \geq \sum p_j$: it does not restrict the decision to schedule a job early or tardy; otherwise, d is called *small*. Baker & Scudder give a comprehensive survey of the early work, including Kanet's (1981) $O(n \log n)$ algorithm for the problem with large d and $\alpha_j = \beta_j = 1$ for all j.

N.G. Hall, M.E. Posner (1991). Earliness-tardiness scheduling problems, I: weighted deviation of completion times about a common due date. *Oper. Res.* 39, 836-846.

N.G. Hall, W. Kubiak, S.P. Sethi (1991). Earliness-tardiness scheduling problems, II: deviation of completion times about a restrictive common due date. *Oper. Res.* 39, 847-856.

J.A. Hoogeveen, S.L. van de Velde (1991). Scheduling around a small common due date. *European J. Oper. Res.* 55, 237-242.

J.A. Hoogeveen, H. Oosterhout, S.L. van de Velde (1994). New lower and upper bounds for scheduling around a small common due date. *Oper. Res.* 42, 102-110.

Hall & Posner prove that minimizing $\sum w_j |C_j - d|$ is NP-hard if d is large. Hall et al. and Hoogeveen & Van de Velde show that minimizing $\sum |C_j - d|$ becomes NP-hard for small d; the former authors also give an $O(n \sum p_j)$ time algorithm, while the latter give an $O(n^2 \sum p_j)$ algorithm for the weighted problem. For the unweighted problem, Hoogeveen et al. provide Lagrangean lower and upper bounds that always seem to concur in practice and present an $O(n \log n)$ 4/3-approximation algorithm.

M.R. Garey, R.E. Tarjan, G.T. Wilfong (1988). One-processor scheduling with symmetric earliness and tardiness penalties. *Math. Oper. Res.* 13, 330-348.

J.A. Hoogeveen, S.L. van de Velde (1997). Earliness-tardiness scheduling around almost equal due dates. *INFORMS J. Comput.*, to appear.

S. Verma, M. Dessouky (1998). Single-machine scheduling of unit-time jobs with earliness and tardiness penalties. *Math. Oper. Res.*, to appear.

Few memorable results are known for the case of unequal due dates. Garey et al. are the first to present a polynomial algorithm to find an optimal schedule for a *given* job sequence. Hoogeveen & Van de Velde present an $O(n^2)$ algorithm for the case that all intervals $[d_j - p_j, d_j]$ overlap, where the due dates are assumed to be large. Verma & Dessouky show that the problem is solvable as a linear program in case of unit processing times and identically ordered weights α_j and β_j.

H.G. Kahlbacher (1989). SWEAT – a program for a scheduling problem with earliness and tardiness penalties. *European J. Oper. Res.* 43, 111-112.

W. Kubiak (1993). Completion time variance minimization on a single machine is difficult. *Oper. Res. Lett.* 14, 49-59.

For small d, an $O(n \sum p_j)$ algorithm for minimizing $\sum(\alpha E_j^c + \beta T_j^c)$ with arbitrary positive c — and an NP-hardness proof for $\alpha = \beta$ and $c = 2$.

2.5 Multiple criteria

T.D. Fry, R.D. Armstrong, H. Lewis (1989). A framework for single machine multiple objective sequencing research. *Omega* 17, 595-607.

An early survey of this vast area, providing 43 references. We only highlight the few polynomial-time algorithms below.

J.A. Hoogeveen, S.L. van de Velde (1995). Minimizing total completion time and maximum cost simultaneously is solvable in polynomial time. *Oper. Res. Lett.* 17, 205-208.

Determining all $O(n^2)$ Pareto-optimal points takes $O(n^3 \min\{n, \log(\sum p_j)\})$ time.

J.A. Hoogeveen (1996). Minimizing maximum promptness and maximum lateness on a single machine. *Math. Oper. Res.* 21, 100-114.

An $O(n^2 \log n)$ algorithm to determine the trade-off curve for maximum lateness and maximum promptness, where the *promptness* of a job is the difference between its start time and its target start time. It can also be used to minimize $L_{\max}$ subject to release dates if $d_j - p_j - A \leq r_j \leq d_j - A$ $(j = 1, \ldots, n)$ for some constant A.

J.A. Hoogeveen (1996). Single-machine scheduling to minimize a function of two or three maximum cost criteria. *J. Algorithms* 21, 415-433.

Determining all Pareto-optimal points in $O(n^4)$ and $O(n^8)$ time, respectively.

3 Parallel machines

T.C.E. Cheng, C.C.S. Sin (1990). A state-of-the-art review of parallel-machine scheduling research. *European J. Oper. Res.* 47, 271-292.

A useful survey, providing 113 references.

3.1 Independent jobs, regular criteria

M. Dell'Amico, S. Martello (1995). Optimal scheduling of tasks on identical parallel processors. *ORSA J. Comput.* 7, 191-200.

A very effective branch-and-bound algorithm for minimizing makespan.

S.L. van de Velde (1993). Duality-based algorithms for scheduling unrelated parallel machines. *ORSA J. Comput.* 5, 192-205.
S. Martello, F. Soumis, P. Toth (1997). An exact algorithm for makespan minimization on unrelated parallel machines. *Discr. Appl. Math.*, to appear.

Optimization and approximation algorithms for the makespan problem.

A. Marchetti Spaccamela, W.S. Rhee, L. Stougie, S. van de Geer (1992). Probabilistic analysis of the minimum weighted flowtime scheduling problem. *Oper. Res. Lett.* 11, 67-71.

In a standard probabilistic model, the authors analyze the solution value obtained by the ratio rule for the single-machine problem and prove asymptotic optimality of this rule for the variant with identical parallel machines.

H. Belouadah, C.N. Potts (1994). Scheduling identical parallel machines to minimize total weighted completion time. *Discr. Appl. Math.* 48, 201-218.

A branch-and-bound algorithm with a polynomial Lagrangean lower bound obtained from a time-indexed formulation with a pseudopolynomial number of variables.

S. Webster (1995). Weighted flow time bounds for scheduling identical processors. *European J. Oper. Res.* 80, 103-111.

Latest paper in a series on obtaining lower bounds by job splitting.

L.M.A. Chan, P. Kaminsky, A. Muriel, D. Simchi-Levi (1995). *Machine scheduling, linear programming and list scheduling heuristics*, Manuscript, Dept. IE, Northwestern University, Evanston.
J.M. van den Akker, J.A. Hoogeveen, S.L. van de Velde (1995). *Parallel machine scheduling by column generation*, Report COSOR 95-35, Dept. Math. Comp. Sci., Eindhoven University of Technology.
Z.L. Chen, W.B. Powell (1995). *Solving parallel machine total weighted completion time problems by column generation*, Manuscript, Dept. Civil Eng. & Oper. Res., Princeton University.

Parallel machine problems can be formulated as set partitioning problems with an exponential number of variables. For minimizing $\sum w_j C_j$, Chan et al. prove that the optimal solution value is at most $(1+\sqrt{2})/2$ times the value of the LP-relaxation and that the latter is, under mild conditions, asymptotically optimal. The other authors show that the LP-bound is effective. They give different column-generation algorithms to solve the LP-relaxation along with different branching strategies to close the gap.

J.Y.-T. Leung, V.K.M. Yu (1994). Heuristic for minimizing the number of late jobs on two processors. *Int. J. Foundations Comput. Sci. 5,* 261-279.

An $O(n \log n)$ 4/3-approximation algorithm for the case of two identical machines.

3.2 Independent jobs, multiple criteria

J.Y.-T. Leung, G.H. Young (1989). Minimizing schedule length subject to minimum flow time. *SIAM J. Comput.* 18, 314-326.

An $O(n \log n)$ algorithm for the preemptive minimization of makespan on identical parallel machines subject to minimum total flow time.

B.T. Eck, M. Pinedo (1993). On the minimization of the makespan subject to flowtime optimality. *Oper. Res.* 41, 797-801.

An $O(n \log n)$ 28/27-approximation algorithm for two identical parallel machines.

D.B. Shmoys, É. Tardos (1993). An approximation algorithm for the generalized assignment problem. *Math. Program.* 62, 461-474.

Among related results, a polynomial 2-approximation algorithm for minimizing a weighted combination of makespan and total cost on unrelated parallel machines.

3.3 Independent jobs, on-line models

J. Sgall (1997). On-line scheduling. A. Fiat, G.J. Woeginger (eds.). *On-Line Algorithms: the State of the Art*, Lecture Notes in Computer Science, Springer, Berlin, to appear.

A survey of the area. There are two types of on-line models. The first stems from on-line bin-packing: the jobs arrive in a list, and the next job in the list is only revealed after all previous jobs have been assigned irrevocably to a time slot on a machine. In the second type of model, the jobs arrive over time.

B. Chen, A. van Vliet, G.J. Woeginger (1994). New lower and upper bounds for on-line scheduling. *Oper. Res. Lett.* 16, 221-230.
Y. Bartal, A. Fiat, H. Karloff, R. Vohra (1995). New algorithms for an ancient scheduling problem. *J. Comput. Syst. Sci.* 51, 359-366.
D.R. Karger, S.J. Phillips, E. Torng (1996). A better algorithm for an ancient scheduling problem. *J. Algorithms* 20, 400-430.
J. Aspnes, Y. Azar, A. Fiat, S. Plotkin, O. Waarts (1993). On-line load balancing with applications to machine scheduling and virtual circuit routing. *Proc. 25th Annual ACM Symp. Theory of Comput.*, 623-631.

The papers in the first branch deal with the makespan problem on m identical parallel machines. Graham's (1966) list scheduling rule has worst-case ratio $2 - 1/m$. More than 25 years later, the problem got renewed attention, where the emphasis is on the questions 'Does there exist an on-line algorithm with a better performance ratio?' and 'How closely can any on-line algorithm approach the optimum off-line solution?' Chen et al. determine lower and upper bounds on the on-line performance for problems with $m \leq 10$. Bartal et al. present an algorithm with performance ratio smaller than $2 - 1/70$ for any m; Karger et al. achieve a ratio of 1.945 for $m \geq 6$. Extending this analysis to uniform and unrelated machines, Aspnes et al. give algorithms with worst-case ratios 8 and $\log m$, respectively.

For models in the second branch, the schedule needs to be built over time. At time t, the jobs released before or at t are fully known, but all other jobs and their characteristics are unknown. The questions that have to be answered are the same, though.

K.S. Hong, Y.-T. Leung (1992). On-line scheduling of real-time tasks. *IEEE Trans. Comput.* 41, 1326-1331.

In case of identical parallel machines, makespan can be minimized on-line when preemption is allowed, but maximum lateness cannot.

A.P.A. Vestjens (1997). Scheduling uniform machines on-line requires nondecreasing speed ratios. *Math. Program. (B)*, to appear.

Necessary and sufficient conditions on the machine speeds to allow the existence of an on-line algorithm that minimizes makespan on uniform parallel machines when preemption is allowed but processor-sharing is not.

G.J. Woeginger (1994). On-line scheduling of jobs with fixed start and end times. *Theor. Comput. Sci.* 130, 5-16.

Necessary and sufficient conditions for the existence of an on-line 1/4-approximation algorithm for the single-machine preemptive scheduling problem of maximizing the total weight of the fully processed jobs, where job j needs to be processed during the interval $[r_j, r_j + p_j]$. The algorithm is shown to be best possible.

B. Chen, A.P.A. Vestjens (1996). *Scheduling on identical machines: how good is LPT in an on-line setting?*, Report COSOR 96-11, Dept. Math. & Comp. Sci., Eindhoven University of Technology.

On-line 'longest processing time first' approximates the minimum makespan within a factor of 3/2 in case of identical parallel machines. No on-line algorithm can do better than 1.3820 for $m = 2$ and better than 1.3473 for $m \geq 3$.

J.A. Hoogeveen, A.P.A. Vestjens (1996). Optimal on-line algorithms for single-machine scheduling. M. Queyranne, W.H. Cunningham (eds.). *Proc. 5th MPS Conf. IPCO*, Lecture Notes in Computer Science 1084, Springer, Berlin, 404-414.

Best possible on-line algorithms for the respective minimization of total completion time and maximum lateness (in case of negative due dates) on a single machine.

C. Phillips, C. Stein, J. Wein (1997). Minimizing average completion time in the presence of release dates. *Math. Program. (B)*, to appear.

An algorithm that converts a preemptive schedule to a nonpreemptive one while increasing the completion times by a factor of at most 2 on a single machine or 3 on identical machines. On-line 2- and $(8 + \epsilon)$-approximation algorithms for the preemptive minimization of total completion time on identical machines and of total weighted completion time on unrelated machines, respectively.

D.B. Shmoys, J. Wein, D.P. Williamson (1995). Scheduling parallel machines on-line. *SIAM J. Comput.* 24, 1313-1331.

General techniques for adjusting off-line algorithms to cope with jobs with unknown release dates and jobs with unknown processing times.

L.A. Hall, A.S. Schulz, D.B. Shmoys, J. Wein (1997) (see Section 2.2).

A framework to design on-line algorithms for minimizing total weighted completion time. The performance guarantees are $3 + \epsilon$, $4 + \epsilon$, and 8 in case of a single machine, identical, and unrelated machines, respectively.

S. Chakrabarti, C.A. Phillips, A.S. Schulz, D.B. Shmoys, C. Stein, J. Wein (1997). Improved scheduling algorithms for minsum criteria. Lecture Notes in Computer Science, Springer, Berlin, to appear.

Improvements of some of the bounds in the previous two papers. An on-line algorithm that approximates both $C_{\max}$ and $\sum w_j C_j$ within constant factors.

3.4 Independent multiprocessor jobs

Nonmalleable multiprocessor jobs need a given number of machines during a given processing time. For *malleable* jobs, the processing time is a nonincreasing function of the number of machines put on the job.

J. Błazewicz, M. Drozdowksi, J. Węglarz (1994). Scheduling multiprocessor tasks: a survey. *Int. J. Microcomputer Applications* 13, 89-97.

A survey of results for problems with nonmalleable jobs. See also Veltman et al. (1990) (Section 3.5).

W. Ludwig, P. Tiwari (1994). Scheduling malleable and nonmalleable parallel tasks. *Proc. 5th Annual ACM–SIAM Symp. Discr. Algorithms*, 167-176.

The most recent of a number of papers on approximating makespan. Performance bounds of 2 and 2.5 for malleable and nonmalleable jobs, respectively.

J. Turek, W. Ludwig, J.L. Wolf, L. Fleischer, P. Tiwari, J. Glasgow, U. Schwiegelshohn, P.S. Yu (1994). Scheduling parallelizable tasks to minimize average response time. *Proc. 6th Annual Symp. Parallel Algorithms and Architectures*, 200-209.

An $O(n^3 + mn)$ 2-approximation algorithm for minimizing total completion time in case of malleable jobs under mild conditions on the processing times.

U. Schwiegelshohn, W. Ludwig, J.L. Wolf, J. Turek, P.S. Yu (1997). Smart SMART bounds for weighted response time scheduling. *SIAM J. Comput.*, to appear.

An $O(n \log n)$ 8-(8.53-)approximation algorithm for minimizing total (weighted) completion time in case of nonmalleable jobs.

3.5 Precedence-constrained jobs

E.G. Coffman, Jr., M.R. Garey (1993). Proof of the 4/3 conjecture for preemptive vs. nonpreemptive two-processor scheduling. *J. ACM 40*, 991-1018.

In case of two identical machines and precedence constraints, the nonpreemptive minimum makespan is at most 4/3 times as large as the preemptive one.

B. Berger, L. Cowen (1995). Scheduling with concurrency constraints. *J. Algorithms* 18, 98-123.

The paper addresses the problem of scheduling unit-time jobs on identical parallel machines to minimize makespan with 'before', 'no later than' and 'at the same time' constraints. The problem is solvable in linear time for $m = 2$ but NP-hard for $m \geq 3$.

B. Veltman, B.J. Lageweg, J.K. Lenstra (1990). Multiprocessor scheduling with communication delays. *Parallel Comput.* 16, 173-182.
P. Chrétienne, C. Picouleau (1995). Scheduling with communication delays: a survey. Chrétienne, Coffman, Lenstra & Liu (eds.) (see Section 1), 65-90.

Two surveys of an active area. *Communication delays* occur between precedence-related jobs assigned to different machines. Some of the basic results: For the case that the number of identical machines is unrestricted and jobs may be duplicated, Papadimitriou & Yannakakis (1990) proved that minimizing makespan is NP-hard and gave a polynomial 2-approximation algorithm. For m part of the input, no job duplication, and unit-time delays and jobs, the problem is NP-hard, even for trees; for $m = 2$, the case of tree-type constraints is solvable in linear time, but the general case is open.

J.A. Hoogeveen, J.K. Lenstra, B. Veltman (1994). Three, four, five, six, or the complexity of scheduling with communication delays. *Oper. Res. Lett.* 16, 129-137.

The complexity of finding short schedules for unit-time jobs subject to unit-time communication delays on a restricted or unrestricted number of identical machines, and lower bounds on the polynomial-time approximability of the makespan.

A. Munier, J.-C. König (1997). A heuristic for a scheduling problem with communication delays. *Oper. Res.*, to appear.
C. Hanen, A. Munier (1995). An approximation algorithm for scheduling dependent tasks on m processors with small communication delays. *IEEE Symp. Emerging Technologies and Factory Automation*, Paris, 167-189.

A polynomial 4/3-approximation algorithm for the case of unrestricted m and unit-time delays and jobs, based on rounding the solution to an LP-relaxation — and an extension to the case of 'small' delays, both for unrestricted and restricted m.

4 Open, flow and job shops

R.A. Dudek, S.S. Panwalkar, M.L. Smith (1992). The lessons of flowshop scheduling research. *Oper. Res.* 40, 7-13.

A critical appraisal of the practical relevance of a substantial body of research.

4.1 Complexity

W. Kubiak, C. Sriskandarajah, K. Zaras (1991). A note on the complexity of openshop scheduling problems. *INFOR* 29, 284-294.

A detailed survey of complexity results for open shop scheduling.

P. Brucker, B. Jurisch, M. Jurisch (1993). Open shop problems with unit time operations. *Z. Oper. Res.* 37, 59-73.

This paper ties unit-time open shop problems to preemptive identical parallel machine problems and offers a survey as well as new results.

J. Du, J.Y.-T. Leung (1993). Minimizing mean flow time in two-machine open shops and flow shops. *J. Algorithms* 14, 24-44.
C. Sriskandarajah, E. Wagneur (1994). On the complexity of preemptive openshop scheduling problems. *European J. Oper. Res.* 77, 404-414.

Minimizing total completion time in a preemptive two-machine shop is NP-hard in case of an open shop, strongly NP-hard for a flow shop — and also strongly NP-hard for an open shop with release dates.

M.M. Dessouky, M.I. Dessouky, S.K. Verma (1996). *Flowshop scheduling with identical jobs and uniform parallel machines*, Manuscript, Dept. Industrial & Systems Eng., University of Southern California, Los Angeles.

Minimizing makespan in a unit-time flow shop with uniform machines at each stage is easy for two stages and strongly NP-hard for three. For the latter case, a polynomial 7/4-approximation algorithm and a branch-and-bound method are given.

J.A. Hoogeveen, J.K. Lenstra, B. Veltman (1996). Minimizing makespan in a multiprocessor flow shop is strongly NP-hard. *European J. Oper. Res.* 89, 172-175.

Even the preemptive two-stage flow shop problem with two identical machines at one stage and a single machine at the other is strongly NP-hard.

W. Kubiak, S. Sethi, C. Sriskandarajah (1995). An efficient algorithm for a job shop problem. *Ann. Oper. Res.* 57, 203-216.

Various authors have observed that instances of the unit-time two-machine job shop can be encoded more compactly than in the 'standard' way. Here: an algorithm for minimizing maximum lateness that is polynomial with respect to such an encoding.

P. Brucker, R. Schlie (1990). Job-shop scheduling with multi-purpose machines. *Comput.* 45, 369-375.

A polynomial algorithm for the two-job case of a generalized job shop, in which an operation can be processed by any machine in its associated machine set.

P. Brucker (1994). A polynomial algorithm for the two machine job-shop scheduling problem with a fixed number of jobs. *OR Spektrum* 16, 5-7.
Y.N. Sotskov, N.V. Shakhlevich (1995). NP-hardness of shop-scheduling problems with three jobs. *Discr. Appl. Math.* 59, 237-266.

Job shop scheduling is easy for $m = 2$, n fixed, and NP-hard for $m = n = 3$.

P. Brucker, S.A. Kravchenko, Y.N. Sotskov (1997). Preemptive job-shop scheduling problems with a fixed number of jobs. *Math. Methods Oper. Res.*, to appear.

Preemptive job shop scheduling is NP-hard for $m = 2, n = 3$. Note that the non-preemptive problem is easy!

N.G. Hall, C. Sriskandarajah (1996). A survey of machine scheduling problems with blocking and no-wait in process. *Oper. Res.* 44, 510-525.

An overview of shop models with the additional feature of *no wait in process* or *no intermediate storage.*

4.2 Approximability

B. Chen, V.A. Strusevich (1993). Approximation algorithms for three-machine open shop scheduling. *ORSA J. Comput.* 5, 321-326.
B. Chen, V.A. Strusevich (1993). Worst-case analysis of heuristics for open shops with parallel machines. *European J. Oper. Res.* 70, 379-390.

Rácsmány proved that *dense* open shop schedules are shorter than twice the optimum. For three machines, the ratio becomes 5/3; a linear-time heuristic improves this to 3/2. The two-stage case with identical machines at each stage has a polynomial 2-approximation algorithm.

B. Chen, C.A. Glass, C.N. Potts, V.A. Strusevich (1996). A new heuristic for three-machine flow shop scheduling. *Oper. Res.* 44, 891-898.

An $O(n \log n)$ time 5/3-approximation algorithm for the three-machine flow shop.

C.-Y. Lee, G.L. Vairaktarakis (1994). Minimizing makespan in hybrid flowshops. *Oper. Res. Lett.* 16, 149-158.
B. Chen (1994). Scheduling multiprocessor flow shops. D.-Z. Du, J. Sun (eds.). *Advances in Optimization and Approximation*, Kluwer, Dordrecht, 1-8.
A polynomial algorithm with performance ratio $2 - 1/\max\{m_1, m_2\}$ for the two-stage flow shop with m_i identical machines at stage i — and the same algorithm along with a literature survey of the area.

C.N. Potts, D.B. Shmoys, D.P. Williamson (1991). Permutation vs. non-permutation flow shop schedules. *Oper. Res. Lett.* 10, 281-284.
The restriction to permutation schedules may cost a factor of more than $\sqrt{m}/2$.

E. Nowicki, C. Smutnicki (1994). A note on worst-case analysis of approximation algorithms for a scheduling problem. *European J. Oper. Res.* 74, 128-134.
The fourth paper in a series on performance bounds for permutation flow shops, which also summarizes the other three.

D.P. Williamson, L.A. Hall, J.A. Hoogeveen, C.A.J. Hurkens, J.K. Lenstra, S.V. Sevast'janov, D.B. Shmoys (1997). Short shop schedules. *Oper. Res.*, to appear.
Given an open, flow or job shop instance with integral processing times, does there exist a schedule of length at most c? This question is easy for $c = 3$ but NP-complete for $c = 4$. Hence, finding a schedule shorter than 5/4 times the optimum is NP-hard ...

L.A. Hall (1997). Approximability of flow shop scheduling. *Math. Program. (B)*, to appear.
S.V. Sevast'janov, G.J. Woeginger (1997). A polynomial approximation scheme for the open shop problem. *Math. Program. (B)*, to appear.
... but for minimizing makespan in flow shops and open shops with a *fixed* number of machines, there exist polynomial approximation schemes.

S.V. Sevast'janov (1994). On some geometric methods in scheduling theory: a survey. *Discr. Appl. Math.* 55, 59-82.
S.V. Sevast'janov (1995). Vector summation in Banach space and polynomial algorithms for flow shops and open shops. *Math. Oper. Res.* 20, 90-103.
The first paper surveys work by the author and others on using vector summation in obtaining absolute and relative performance bounds for shop scheduling and other problems. The second presents new results along the same lines.

D.B. Shmoys, C. Stein, J. Wein (1994). Improved approximation algorithms for shop scheduling problems. *SIAM J. Comput.* 23, 617-632.
Polylogarithmic performance bounds for job shops and extensions with parallel machines at each stage or with jobs corresponding to a partial order of their operations.

4.3 Branch-and-bound

F. Della Croce, V. Narayan, R. Tadei (1996). The two-machine total completion time flow shop problem. *European J. Oper. Res.* 90, 227-237.

A survey and empirical comparison of old and new lower bounds.

A.M.G. Vandevelde, J.A. Hoogeveen, C.A.J. Hurkens, J.K. Lenstra (1997). *Lower bounds for the multiprocessor flow shop*, Manuscript, Dept. Math. & Comp. Sci., Eindhoven University of Technology.

A theoretical and computational investigation of lower bounds for the m-stage flow shop with identical parallel machines at each stage.

J. Carlier, E. Pinson (1990). A practical use of Jackson's preemptive schedule for solving the job shop problem. *Ann. Oper. Res.* 26, 269-287.
J. Carlier, E. Pinson (1994). Adjustment of heads and tails for the job-shop problem. *European J. Oper. Res.* 78, 146-161.
D. Applegate, W. Cook (1991). A computational study of the job-shop scheduling problem. *ORSA J. Comput.* 3, 149-156.
P. Brucker, B. Jurisch (1993). A new lower bound for the job-shop scheduling problem. *European J. Oper. Res.* 64, 156-167.
P. Brucker, B. Jurisch, B. Sievers (1994). A branch and bound algorithm for the job-shop scheduling problem. *Discr. Appl. Math.* 49, 107-127.
P. Brucker, B. Jurisch, A. Krämer (1994). The job-shop problem and immediate selection. *Ann. Oper. Res.* 50, 73-114.
P. Martin, D.B. Shmoys (1996). A new approach to computing optimal schedules for the job shop scheduling problem. M. Queyranne, W.H. Cunningham (eds.). *Proc. 5th MPS Conf. IPCO*, Lecture Notes in Computer Science 1084, Springer, Berlin, 389-403.

These papers represent the continuing efforts to develop optimization algorithms for job shop scheduling. The most effective lower bound is the classical single-machine bound, enhanced by techniques to increase heads and tails and to fix the order between operations on the same machine. The two-job bound of Brucker & Jurisch may work well in case m/n is large. LP-based bounds are not yet competitive, as shown by Applegate & Cook, who tested cutting-plane methods for standard disjunctive and mixed-integer formulations, and by Martin & Shmoys, who computed the linear relaxation of a giant packing formulation. The branching strategies used settle an essential conflict between two operations or choose an operation to precede (or follow) a larger set of operations. The infamous 10×10 problem is now within reach, but 15×15 instances may still pose a computational challenge.

4.4 Local search

E. Taillard (1990). Some efficient heuristic methods for the flow shop sequencing problem. *European J. Oper. Res.* 47, 65-74.
E. Nowicki, C. Smutnicki (1996). A fast tabu search algorithm for the permutation flow-shop problem. *European J. Oper. Res.* 91, 160-175.
H. Ishibuchi, S. Misaki, H. Tanaka (1995). Modified simulated annealing algorithms for the flow shop sequencing problem. *European J. Oper. Res.* 81, 388-398.
C.R. Reeves (1995). A genetic algorithm for flowshop sequencing. *Computers Oper. Res.* 22, 5-13.

Tabu search (twice), simulated annealing, and a genetic algorithm for flow shop scheduling.

E. Nowicki, C. Smutnicki (1996). A fast taboo search algorithm for the job shop problem. *Management Sci.* 42, 797-813.
E. Balas, A. Vazacopoulos (1994). *Guided local search with shifting bottleneck for job shop scheduling,* Report MSRR-609, GSIA, Carnegie Mellon University, Pittsburgh.
R.J.M. Vaessens, E.H.L. Aarts, J.K. Lenstra (1996). Job shop scheduling by local search. *INFORMS J. Comput.* 8, 302-317.

The champions in job shop scheduling by local search, based on variants of tabu search and variable-depth search, and a survey covering 71 references.

5 Miscellaneous

P. Serafini, W. Ukovich (1989). A mathematical model for periodic scheduling problems. *SIAM J. Discr. Math.* 2, 550-581.
C. Hanen, A. Munier (1995). Cyclic scheduling on parallel processors: an overview. Chrétienne, Coffman, Lenstra & Liu (eds.) (see Section 1), 193-226.

Two surveys. In periodic scheduling, operations have to be periodically executed at a constant rate over time. In cyclic scheduling, a set of generic tasks must be performed infinitely often.

E. Nowicki, S. Zdrzałka (1990). A survey of results for sequencing problems with controllable processing times. *Discr. Appl. Math.* 26, 271-287.

A survey of results on another variation of the standard model, in which the processing times may vary at some cost.

N.G. Hall, S.P. Sethi, C. Sriskandarajah (1991). On the complexity of generalized due date scheduling problems. *European J. Oper. Res.* 51, 100-109.

Complexity categorization of problems with positional due dates.

L. Özdamar, G. Ulusoy (1995). A survey on the resource-constrained project scheduling problem. *IIE Trans.* 27, 574-586.
E. Demeulemeester (1995). Minimizing resource availability costs in time-limited project networks. *Management Sci.* 41, 1590-1598.

Resource-constrained project scheduling is a practical and classical area. The focus is on branch-and-bound and local search. The first paper surveys the area, the second gives a state-of-the-art branch-and-bound algorithm.

Y. Crama (1997). Combinatorial optimization models for production scheduling in automated manufacturing systems. *European J. Oper. Res.*, to appear.

A survey of results on models involving tool management, part selection, robotic cells, and automated guided vehicles.