stichting
mathematisch
centrum

MC

AFDELING MATHEMATISCHE BESLISKUNDE (DEPARTMENT OF OPERATIONS RESEARCH)

BC 22/81

JUNI

J.K. LENSTRA, A.H.G. RINNOOY KAN (editors)

DETERMINISTIC AND STOCHASTIC SCHEDULING: EXTENDED ABSTRACTS

kruislaan 413 1098 SJ amsterdam

BIBLIOTHEEK MATHEMATISCH CENTRUM
AMSTERDAM

Printed at the Mathematical Centre, 413 Kruislaan, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

1980 Mathematics Subject Classification: 60K25, 68C15, 90B35

DETERMINISTIC AND STOCHASTIC SCHEDULING:
EXTENDED ABSTRACTS

edited by

J.K. LENSTRA (*Mathematisch Centrum, Amsterdam*)
A.H.G. RINNOOY KAN (*Erasmus University, Rotterdam*)

This report collects the abstracts of the lectures to be given at an Advanced Study and Research Institute on Theoretical Approaches to Scheduling Problems, to be held in Durham, England, July 6-17, 1981. The Institute is sponsored by the NATO Advanced Study Institutes Programme and Systems Science Panel, The Institute of Mathematics and Its Applications, and the Mathematisch Centrum.

KEY WORDS & PHRASES: *combinatorial optimization, deterministic sequencing and scheduling, stochastic processes, queueing theory, stochastic sequencing and scheduling.*

ADVANCED STUDY INSTITUTE

Introduction to combinatorial optimization

E.L. LAWLER — Mon 06 15.00-16.00
Design and analysis of algorithms for combinatorial optimization

J.K. LENSTRA — Mon 06 16.30-17.30
Computational complexity of combinatorial problems

A.H.G. RINNOOY KAN — Tue 07 15.30-16.30
Enumerative methods

M.L. FISHER — Tue 07 17.00-18.00
Analysis of heuristics

Deterministic sequencing and scheduling

J.K. LENSTRA — Thu 09 09.30-10.30
Single machine scheduling to minimize maximum cost

M.L. FISHER — Thu 09 11.00-12.00
Single machine scheduling to minimize total cost

M.L. FISHER — Thu 09 15.30-16.30
Nonpreemptive scheduling of parallel machines

E.L. LAWLER — Fri 10 09.30-10.30
Scheduling precedence-constrained unit-time jobs on parallel machines

B. SIMONS — Fri 10 11.00-12.00
**On scheduling with release times and deadlines*

E.L. LAWLER — Mon 13 09.30-10.30
Preemptive scheduling of parallel machines

C.U. MARTEL — Mon 13 11.00-12.00
**Preemptive scheduling of uniform machines with release times and deadlines*

A.H.G. RINNOOY KAN — Tue 14 09.30-10.30
Open shop, flow shop and job shop scheduling

Introduction to analysis and optimization of stochastic processes

M.A.H. DEMPSTER — Tue 07 09.30-10.30
An elementary introduction to stochastic processes

G. WEISS — Tue 07 11.00-12.00
Markov decision processes

E. GELENBE — Wed 08 09.00-10.00
Single server queues

S.M. ROSS — Wed 08 10.00-11.00
Discrete state stochastic systems

K.C. SEVCIK — Wed 08 11.30-12.30
Queueing networks

Stochastic sequencing and scheduling

J.C. GITTINS — Thu 09 17.00-18.00
Discrete time stochastic scheduling

G. WEISS — Fri 10 15.30-16.30
Scheduling stochastic jobs on several machines

E. GELENBE — Fri 10 17.00-18.00
Scheduling in computer systems and networks

S.M. ROSS — Mon 13 15.30-16.30
Multi-server queues

K.C. SEVCIK — Mon 13 17.00-18.00
Applications of queueing networks to computer systems

Interfaces between deterministic and stochastic scheduling

L.E. SCHRAGE — Tue 14 11.00-12.00
The multiproduct lot scheduling problem

E.G. COFFMAN, JR. — Tue 14 15.30-18.00
Probability models of sequencing and packing algorithms I, II

M. PINEDO, L.E. SCHRAGE — Wed 15 09.00-10.00
Stochastic shop scheduling: a survey

M.A.H. DEMPSTER — Wed 15 10.00-11.00
A stochastic approach to hierarchical scheduling

W. GAUL — Wed 15 11.30-12.30
**On stochastic analysis of project-networks*

Invited lecture
**Contributed lecture*

DETERMINISTIC SEQUENCING AND SCHEDULING

J.K. LENSTRA
Mathematisch Centrum, Amsterdam

The purpose of this brief survey is to point out the relations between the twelve lectures in the deterministic part of the Advanced Study Institute.

Introduction to combinatorial optimization

1. E.L. LAWLER. *Design and analysis of algorithms for combinatorial optimization.*
2. J.K. LENSTRA. *Computational complexity of combinatorial problems.*
3. A.H.G. RINNOOY KAN. *Enumerative methods.*
4. M.L. FISHER. *Analysis of heuristics.*

Combinatorial optimization involves the study of problems in which an optimal ordering, selection or assignment of a finite set of objects has to be determined. Examples are routing problems (such as the celebrated traveling salesman problem of finding the shortest closed tour through a number of cities), location problems, and scheduling problems. While combinatorial optimization, as a subarea of operations research, is rooted in the theory of *mathematical programming*, the last years have witnessed an increasing application of tools from *computer science*.

Lawler's lecture is an excellent demonstration of this phenomenon. After a discussion of a number of *standard problems* and some *general algorithmic techniques* for their solution, the principles of the *implementation* of algorithms are reviewed and the concepts underlying the *theoretical analysis* of algorithms are outlined.

The second lecture provides an introduction to the theory of *computational complexity*. This theory allows us to make a formal distinction between *well-solved* problems, which can be solved by an algorithm whose running time is bounded by a polynomial function of problem size, and *NP-hard* problems, for which the existence of such an algorithm is very unlikely. In solving an NP-hard problem, one has to choose between using slow *optimization*

algorithms or fast *approximation* algorithms. These alternative approaches form the subject of the remaining two lectures.

Rinnooy Kan illustrates the use of *enumerative methods*, such as *dynamic programming* and *branch-and-bound*. These methods are guaranteed to produce an optimal solution, but only after an often time consuming search through the set of feasible solutions.

Fisher discusses the use of *heuristics* and three different approaches for analyzing their performance: *empirical*, *worst-case*, and *probabilistic*. Special attention is paid to the second approach: given a problem (*i.e.* the knapsack problem) and a heuristic for its solution, how does one determine an upper bound on the ratio between the approximate solution value and the optimal one?

Deterministic sequencing and scheduling: the state of the art

1. J.K. LENSTRA. *Single machine scheduling to minimize maximum cost.*
2. M.L. FISHER. *Single machine scheduling to minimize total cost.*
3. M.L. FISHER. *Nonpreemptive scheduling of parallel machines.*
4. E.L. LAWLER. *Scheduling precedence-constrained unit-time jobs on parallel machines.*
5. E.L. LAWLER. *Preemptive scheduling of parallel machines.*
6. A.H.G. RINNOOY KAN. *Open shop, flow shop and job shop scheduling.*

In the generic *single machine* scheduling problem, a number of jobs, each with a given processing time, has to be executed on a single machine that can handle at most one job at a time, subject to a variety of constraints that may include release dates, deadlines, and precedence constraints; it is also specified whether preemption (job splitting) is allowed or not. Each job incurs at its completion time a certain cost, where the cost function is nondecreasing over time. The problem is to find a schedule that minimizes a given optimality criterion, which is usually the maximum or the sum of the job completion costs.

The first two lectures deal with *min-max* and *min-sum* single machine problems, respectively. Although either of these classes contains many well-solved and NP-hard problems to which all the tools of combinatorial

optimization have been applied, the relative emphasis will be on polynomial-time algorithms in the first lecture and on enumerative methods in the second one.

This model can be generalized in two directions. In the first generalization, each job has to be processed on any one of a number of *parallel* machines.

Fisher considers the *nonpreemptive* case and concentrates in particular on the maximum completion time criterion. For this model, many results on the worst-case performance of approximation algorithms have been obtained.

Lawler investigates the addition of *precedence constraints* to this model, under the assumption that all jobs have unit processing times. The problem is NP-hard in general, but a number of special cases can be solved in polynomial time.

Lawler next surveys the *preemptive* case for several optimality criteria. The most notable recent advances in scheduling theory, concerning polynomial-time algorithms as well as NP-hardness proofs, have been made in this area.

In another generalization of the single machine model, each job consists of a set of operations, each of which has to be executed on a specific machine.

Rinnooy Kan discusses the resulting *open shop*, *flow shop* and *job shop* models. Except for a couple of well-solved special cases, these problems are very difficult, and the presentation is mainly concerned with the systematic development of enumerative methods for their solution.

Deterministic sequencing and scheduling: two recent developments

1. B. SIMONS. *On scheduling with release times and deadlines.*
2. C.U. MARTEL. *Preemptive scheduling of uniform machines with release times and deadlines.*

Suppose each of n jobs has to be executed during a given processing time between a given release date and a given deadline. Does there exist a *feasible* schedule?

The *nonpreemptive single machine* version of this model is well known to be NP-hard. Simons' contribution is a polynomial-time algorithm for this

problem under the additional assumption that all processing times are equal. The method can be extended to the case of *identical* parallel machines.

Martel considers the *preemptive* problem on *uniform* parallel machines (*i.e.*, parallel machines of different speeds). His polynomial-time algorithm involves *polymatroidal* network flow techniques, an extension of classical network flow theory.

References

Polynomial-time algorithms in combinatorial optimization:

E.L. LAWLER (1976) *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York.

A comprehensive treatment of computational complexity theory:

M.R. GAREY, D.S. JOHNSON (1979) *Computers and Intractability: a Guide to the Theory of NP-Completeness*, Freeman, San Francisco.

An introduction to the worst-case analysis of heuristics:

M.L. FISHER (1980) Worst-case analysis of heuristic algorithms. *Management Sci.* 26,1-14.

A survey of deterministic sequencing and scheduling theory:

R.L. GRAHAM, E.L. LAWLER, J.K. LENSTRA, A.H.G. RINNOOY KAN (1979) Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann. Discrete Math.* 5,287-326.

Revised and updated versions of the last two papers will appear in the ASRI Proceedings.

STOCHASTIC SEQUENCING AND SCHEDULING

E. GELENBE

Université de Paris Sud/INRIA, Rocquencourt

Not received

INTERFACES BETWEEN DETERMINISTIC AND STOCHASTIC SCHEDULING

M.A.H. DEMPSTER
Balliol College, Oxford/IIASA, Laxenburg

In the following lectures attempts will be made to relate the deterministic and stochastic approaches to some specific scheduling problems:

1. L.E. SCHRAGE. *The multiproduct lot scheduling problem.*
2/3. E.G. COFFMAN, JR. *Probability models of sequencing and packing algorithms.*
4. M. PINEDO, L.E. SCHRAGE. *Stochastic shop scheduling: a survey.*
5. M.A.H. DEMPSTER. *A stochastic approach to hierarchical scheduling.*

To place these contributions in perspective, it may be useful to review the relevance of some general considerations in stochastic optimization to the present context.

When deterministic combinatorial optimization models of practical scheduling problems are extended to more realistic stochastic models by assuming various data are *random variables* several options in *problem formulation* arise immediately. Five important *alternative* assumptions present themselves:

1. Is *expectation* of random costs or rewards an appropriate valuation criterion *or* must some *more complicated* stochastic optimality criterion be used?
2. Are optimizing decisions to be taken *before or after* the random variables are realized?
3. Are all data, such as processing time distributions, available at the *outset or* is a stochastic process generating *arrivals* to the system involved?
4. Is the model posed over a *finite or* an *infinite* horizon?
5. Are probability distributions of random variables or stochastic processes *known completely* in advance *or* are they known only up to certain parameters which must be *estimated* as the data is realized?

All these questions are familiar in stochastic system theory for systems involving continuous decision variables - *i.e. stochastic programming* and *stochastic optimal control* models (see *e.g.* [Dempster 1980; Fleming & Rishel 1975]). Their importance may however not as yet be fully appreciated by

researchers in related fields. Here we are principally interested in probabilistic algorithm analysis, and other questions in theoretical computer science and the mathematics of operations research, and in the control of queueing systems and networks.

Question 1 concerns the stochastic nature of the valuation criterion involved in the model. For most stochastic scheduling models this will be the *expected value* of the (now random) criterion, such as makespan or flow-time, used in the corresponding combinatorial optimization model. Such a criterion is entirely appropriate for these models in that they generally apply to *repetitive* situations in which relatively *small* costs or gains are involved per unit time. In the contrary situation, when a *once for all* decision must be taken in the face of uncertainties involving relatively *large* gains or losses, total preference ordering of reward distributions using Von Neumann-Morgenstern *expected utility theory* is a more general tool (see *e.g.* [Luce & Raiffa 1957]). For certain scheduling models it is possible to establish optimality in *distribution* for the random criterion - *i.e.* the probability of achieving a given criterion level is *everywhere* at least as great under the optimal policy as for any other. This is of course a very stringent optimality criterion which guarantees optimality for any expected utility criterion involving a monotonic utility function.

The second question, concerning the *timing* of the realization of the random variables in the problem, is of *crucial* importance for the nature of the analysis. If the random data is realized *before* optimization (or approximate optimization) is performed, in the present context of combinatorial optimization the solution of the resulting *distribution problem* - find the distribution of the optimal criterion value, or its expectation, or other moments - is termed *probabilistic analysis* of an *algorithm*, either optimal or heuristic. That is, a (multivariate) distribution is assumed for the problem data and the question of interest is (often) the *a priori* expected performance of the algorithm. For certain simple scheduling problems and parameter distributions such results may be obtained for an *optimal* algorithm. More often, in order to evaluate the performance of a *heuristic* algorithm, an upper (say) bound on the expected criterion value produced by the heuristic is compared with a lower (say) bound on the expected criterion value produced by an optimal algorithm. Once it is assumed that some of the

problem data is realized *sequentially* only *after* some decisions have been taken, the resulting decision problems generally become more difficult to analyze. In some simple cases - *e.g.* for list scheduling heuristics applied to m-machine scheduling problems - the analysis remains the same independent of the timing of the realization of the random problem data. However, this situation seldom applies to the case of an *optimal* policy for a problem with any complexity of structure.

Question 3 - whether or not a stochastic process of arrivals to the system is assumed - delimits the boundary between the *static* models which form the bulk of the problems analyzed in operations research and computer science and the dynamic *queueing* models analyzed in both disciplines. In computer science, the study of queueing systems has recently become of fundamental importance for *computer* system and network *performance modelling* (see *e.g.* [Kleinrock 1976]). Most of the models used so far differ from (static) stochastic scheduling models in that no *active* scheduling policy - other than arrival order, possibly by priority class - is normally assumed, since processing times are usually taken as independent identically distributed random variables. A current research topic involves the extension of recent results for stochastic scheduling problems to models involving arrival processes. Such models are natural stochastic extensions of deterministic scheduling models incorporating job *release dates*. The natural setting for their analysis is in *continuous time* - although certain optimality results for m-machine scheduling problems are so far only established in *discrete* time. Ultimately - for example, to improve our understanding of real job shops - it would be useful to analyze a *network* of m-machine problems. Each node of the appropriate network would be not simply a single server, but rather a scheduled m-machine system, so that node input and output processes would need to be more complicated than have so far been analyzed. Nevertheless, the recent general theory of the optimal control of stochastic systems driven by point processes should be relevant (*cf.* [Walrand & Varaiya 1978]).

The next question (4), concerning the length of the planning horizon, is closely connected to the assumption of an arrival process in that often the only tractable analysis - for example, for queueing models - refers to the *asymptotic* state distribution of the system as the underlying stochastic processes tend to their long run (stationary) *equilibrium* distributions at

infinity. In the probabilistic analysis of algorithms for finite horizon scheduling problems, *asymptotic analysis* as the number of jobs in the system tends to infinity is also useful and usually involves (implicitly or explicitly) a time horizon tending to infinity.

The final question concerning estimation of distributional parameters - usually be recursive *Bayesian* methods simultaneous with decision making - has so far received scant attention in the scheduling or queueing literature despite the fact that it is an area of current research effort in stochastic programming and stochastic control theory.

With this background the content and methods of the lectures in this session may be briefly described.

In the first lecture, Schrage discusses the deterministic NP-hard *multi-product scheduling* problem over a finite horizon. In this problem various products requiring different processing times on a single machine must be produced to meet given due dates. At each product change *set up costs* are incurred and products produced before their due dates may be stored upon payment of *inventory holding costs*. Schrage surveys linear programming approximations to the problem and he analyzes the asymptotic exactness of the solutions of these LP models relative to the exact integer optimum as the number of products in the system increase. Although product due dates (and more generally *demands*) are realistically random, so far little stochastic analysis of this practical problem has been undertaken.

Coffman is concerned in the next two lectures with the probabilistic analysis of the performance in expectation of certain heuristics for various scheduling and packing problems. He first analyzes *list scheduling* heuristics for minimizing makespan of a number n of jobs in an m *identical machine* system with random *independent identically distributed* processing times in terms of the Markov process of *variations* - increments to the latest completion time of the current jobs - defined on the job number. He analyzes the asymptotic state of this process as the number of jobs in the system tends to infinity and shows that its convergence is geometric so that its (asymptotic) equilibrium distribution allows an accurate approximation of expected makespan even for rather small n. Similar techniques are used to analyze the expected performance of the *next fit bin packing* algorithm for an independent identically distributed sequence of fractional piece sizes

and an infinite sequence of unit capacity bins. By specializing to *uniform* distributions on the unit interval more precise analyses of more complex heuristics for both problems can be performed. Coffman also treats the expected performance of some *2-dimensional packing* heuristics.

Pinedo and Schrage in the fourth lecture survey recent *optimality* results for a number of stochastic decision problems concerned with shop scheduling. They are principally concerned with 2-machine shops and a fixed number of jobs with *negative exponential* processing time distributions with different means. The optimality criteria utilized are expected makespan and expected flowtime, and optimal policies are presented for *open shops*, *flow shops* (both with and without *blocking* caused by no storage between machines) and *job shops*. Some results are also presented for the m-machine flow shop. The optimal policies are scheduling rules in which at job completions unprocessed jobs are assigned to machines in ascending (to minimize expected flowtime) or descending (to minimize expected makespan) order of *expected processing time*. These policies are for obvious reasons termed respectively *shortest expected processing time* (SEPT) and *longest expected processing time* (LEPT).

In the last lecture, Dempster surveys recent results in stochastic discrete programming models for hierarchical planning problems. Practical problems of this nature typically involve a sequence of decisions over time at an increasing level of detail and with increasingly accurate information. These may be modelled by multistage stochastic programmes whose lower levels (later stages) are stochastic versions of familiar NP-hard deterministic combinatorial optimization problems and hence require the use of approximations and heuristics for near-optimal solution. After a brief survey of distributional assumptions on processing times under which SEPT and LEPT policies remain optimal for m-machine scheduling problems, results are presented for various 2-level scheduling problems in which the first stage concerns the *acquisition* (or assignment) of machines. For example, heuristics which are asymptotically optimal in expectation as the number of jobs in the system increases are analyzed for problems whose second stages are either *identical* or *uniform* m-machine scheduling problems. A 3-level location, distribution and routing model in the plane is also discussed.

References

M.A.H. DEMPSTER (ed.) (1980) *Stochastic Programming*, Academic, London.

W.H. FLEMING, R.W. RISHEL (1975) *Deterministic and Stochastic Control*, Springer, New York.

L. KLEINROCK (1976) *Queueing Systems, Vol. II: Computer Applications*, Wiley, New York.

R.D. LUCE, H. RAIFFA (1957) *Games and Decisions*, Wiley, New York.

J. WALRAND, P. VARAIYA (1978) The outputs of Jacksonian networks are Poissonian. Memo ERL-M78/60, Electronics Research Laboratory, University of California, Berkeley.

PROBABILITY MODELS OF SEQUENCING AND PACKING ALGORITHMS

E.G. COFFMAN, JR.
Bell Laboratories, Murray Hill, NJ

Classical analyses of sequencing and packing rules have been based on combinatorial models. Among the important results of this research are: (1) complexity classifications for a variety of optimization problems (*e.g.* those concerning flow time performance measures and those concerning the number of resources needed to obtain a given flow time performance); (2) derivations of worst-case performance guarantees for simple approximation rules relative to optimization rules for NP-complete problems.

Following a brief review of the above results, our presentation will concentrate on recent *expected* performance results for approximation rules, based on probability models describing the lengths of the jobs being sequenced, and the dimensions of the pieces being packed. These models are intermediate between the combinatorial and the fully stochastic (*e.g.* queuing) models, in the sense that all jobs (pieces) are assumed available at the outset; *i.e.*, no arrival mechanism is assumed. In all models to date dimensions have been assumed to be independent, identically distributed random variables (i.i.d.r.v.'s); in some cases the distribution has been specialized (*e.g.* the uniform or exponential distribution), as a concession to tractability. Relative performance has been studied by comparing upper bounds on the expected performance of the approximation rule with lower bounds on the expected performance of an optimization rule.

Following the survey of the related combinatorial results, the presentation will be divided into the following five sections.

I. Expected makespans for list scheduling

We study the makespan performance of list schedules for sets of n independent jobs $\{J_1,\ldots,J_n\}$ on $m > 1$ machines. List schedules are produced by selecting the jobs in sequence from some given but arbitrary list, and assigning them to machines as they become available after finishing earlier jobs. We analyze the expected makespan (latest finishing time), $\pi_{m,n}$, under the assumption

that the job execution times are i.i.d.r.v.'s with a distribution function, $\Pr\{J \le x\} = G(x)$, having finite first and second moments, $E(J)$ and $E(J^2)$, and density, $g(x)$.

The study of makespans reduces essentially to the analysis of the *variations* for m = 2:

$$V_i = \begin{cases} J_i, & i = 1, \\ |V_{i-1} - J_i|, & 1 < i \le n. \end{cases}$$

The $\{V_i\}$ form a Markov process, the distribution for V_{i+1} being given by

$$F_{V_{i+1}}(y) = \int_0^\infty K(x,y)\,dF_{V_i}(x), \quad i \ge 1,$$

$$F_{V_1}(y) = G(y).$$

The density corresponding to the stochastic kernel $K(x,y)$ is obtained as

$$k(x,y) = \begin{cases} g(x+y) + g(x-y), & 0 < y \le x, \\ g(x+y), & y > x > 0. \end{cases}$$

For the limiting variation V one finds

$$f_V(y) = \frac{1}{E[J]}\,[1-G(y)]$$

with

$$E[V] = \frac{E[J^2]}{2E[J]}.$$

A key property of the chain $\{V_i\}$ that can be proved is that it converges geometrically. From these results we find for arbitrary m that $\pi_{m,n}$ converges geometrically to

$$\frac{n}{m}\,E[J] + E[\bar{V}] - \frac{m-1}{m}\,E[V]$$

where $E[\bar{V}]$ is the expectation of the maximum of m-1 i.i.d.r.v.'s with distribution F_V. The fast rate of convergence allows for an excellent approximation to be made even for rather small n. $\pi_{m,n}$ may be compared to the lower bound on the expected optimum, $\frac{n}{m}\,E[J]$, to assess the approximation rule.

II. Expected makespans under algorithms of greedy type

We use the assumptions above, except for m = 2 and specialization to the uniform distribution on [0,1], but we examine the more promising largest-processing-time-first (LPT) rule and a variant more amenable to analysis. Let $X_1 \le X_2 \le \ldots \le X_n$ be the order statistics of n samples from U[0,1]. We prove the bound on the expected final variation of processor finishing times

$$E[V_n] \le \sum_{i=1}^{n} E[Z_i]$$

where Z_i is the positive part of $X_i - \sum_{j<i} X_j$. Details are provided to show that $E[V_n] = O(\frac{1}{n})$ where the multiplicative constant is small.

In the variant studied, jobs are assigned two at a time, one to each processor in LPT order, the larger being placed on the processor with the earlier availability (if n is odd J_n is placed simply on the processor having earlier availability). This algorithm is subject to a more precise analysis showing that the expectation of the final variation is 1/n.

III. Expected performance of next-fit one-dimensional bin packing

Somewhat similar to §I above we approximate the solution for finite problems by results obtained for the asymptotic case $n \to \infty$. We assume an infinite sequence of bins $\langle B_i \rangle$ whose common capacity is taken, without loss of generality, to be 1. We assume an infinite sequence of pieces $\langle P_i \rangle$ whose sizes are i.i.d.r.v.'s drawn according to a distribution G(x) on [0,1].

We study the efficiency of the Next-Fit fule whereby the bins are packed in the sequence $B_1, B_2, \ldots$ as follows. First, pieces are drawn in sequence from the list and placed in B_1 until a piece, say P, is encountered which will not fit into the remaining unused capacity of B_1. At that point, starting with P, B_2 is packed in an identical manner; the first piece not fitting in B_2 commences B_3, *etc.* Next-Fit performance is analyzed by examining the Markov chain $\{L_i\}$, where L_i is the level of B_i once B_{i+1} has started. Once again,

$$F_{L_{i+1}}(y) = \int_0^{\infty} K(x,y)\, dF_{L_i}(x),$$

$$F_{L_1}(y) = K(1,y).$$

Solving for K(x,y)

$$K(x,y) = \begin{cases} \sum_{n=0}^{\infty} \int_{1-x}^{y} \int_{0}^{y-w} \frac{1-G(1-w-s)}{1-G(1-x)} \, dF_{S_n}(s)\,dG(w), & y > 1 - x, \\ 0, & y \le 1 - x, \end{cases}$$

where F_{S_n} is the distribution of the sum of n i.i.d.r.v.'s from G(x). A key result has been that F_{L_i} converges geometrically to the limiting distribution F_L. From this result it has been shown that there exists a constant γ such that for any m

$$|m\bar{L} - \sum_{i=1}^{m} E[L_i]| \le \gamma$$

where $\bar{L}$ is the expected value of F_L. Thus, the limiting-distribution results lead to an approximation for the finite case.

Specializing to $P \sim U[0,1]$ one finds

$$K(x,y) = \begin{cases} 1 - (1-y)\varepsilon^{-(1-y)}(\frac{\varepsilon^x}{x}), & 1-x < y \le 1, \\ 0, & 0 \le y \le 1-x, \end{cases}$$

$$F_L(y) = y^3, \quad \bar{L} = 3/4,$$

and a constant γ such that

$$m \le \frac{4}{3} \sum_{i=1}^{m} E[L_i] + 3.$$

IV. Expected performance of more effective rules in bin-packing

The distribution of piece sizes is specialized to U[0,1]. An algorithm is introduced whose performance, although not as good as the better approximation rules, is much more amenable to analysis. Basically, the algorithm scans a list of pieces in order by size, alternating between right-to-left and left-to-right, attempting to pair larger and smaller pieces in the same bin. If n is the length of the list, it is shown that the expected number of bins used by the algorithm is at most

$$\frac{n}{2} + (\frac{n}{2\pi})^{1/2} + o(n^{1/2}).$$

The term $\frac{n}{2} + o(n)$ is best possible in the sense that a lower bound for any

packing rule can be shown to be

$$\frac{n}{2} + \left(\frac{n}{24\pi}\right)^{1/2}(3^{1/2}-1) + O(1).$$

The analysis involves the use of a certain random walk.

V. Recent results in the expected performance of two-dimensional packing rules

In this research simple level-oriented approximation rules are studied for the problem of minimizing the length of a given strip necessary to pack orthogonally a collection of rectangles. Using methods similar to those already mentioned bounds on expected lengths are derived.

References

E.G. COFFMAN, JR., K. SO, M. HOFRI, A.C. YAO (1980) A stochastic model of bin-packing. *Inform. and Control* 44,105-115.

G.N. FREDERICKSON (1980) Probabilistic analysis for simple one- and two-dimensional bin-packing algorithms. *Inform. Process. Lett.* 11,272-314.

E.G. COFFMAN, JR., K. SO (1981) On the expected makespan of schedules of independent tasks. Presented at the TIMS/ORSA Symposium on Applied Probability and Computer Science/The Interface, Boca Raton, Jan. 1981; submitted for publication.

G.S. LUEKER. An average-case analysis of bin-packing. Preliminary draft.

E.G. COFFMAN, JR., G.S. LUEKER. On the expected makespan of schedules of independent tasks produced by rules of greedy type. In process.

M. HOFRI. On the expected performance of two dimensional bin-packing approximation rules. Submitted for publication.

AN ELEMENTARY INTRODUCTION TO STOCHASTIC PROCESSES

M.A.H. DEMPSTER
Balliol College, Oxford/IIASA, Laxenburg

This lecture is aimed at providing an intuitive understanding of the rigorous foundations of stochastic process theory as a foundation for the more advanced lectures on the topic at this Institute. It is based on excerpts from [Dempster 1970]. Definitions and results will be stated precisely within the framework of Kolmogorov's axiomatic approach to probability theory (see, for example, [Ross 1976; Tucker 1967]), but few proofs will be given.

The treatment will begin with a quick review of elementary axiomatic probability theory - including the concepts of probability space, random variables and vectors, their distributions and moments, independence, conditional probability and expectation, and Bayes' theorem. Participants will be expected to study the relevant sections of [Dempster 1970, §§1,2] *before the lecture*.

The lecture will concentrate on classification and analytic representation of stochastic processes, their moment properties and long run behaviour [Dempster 1970, §6] and on an elementary introduction to finite state Markov chains in discrete time [Dempster 1970, §7]. The latter topic is intended to lead to the treatment of countable state Markov chains in continuous time useful in queueing theory [Gelenbe & Mitrani 1980, Ch.1] and to provide background for the study of Markov and semi-Markov decision processes and other discrete state stochastic systems [Cinlar 1975, Ross 1970].

References

E. CINLAR (1975) *Introduction to Stochastic Processes*, Prentice-Hall, Englewood Cliffs, NJ.

M.A.H. DEMPSTER (1970) *Applied Probability in Operations Analysis*, Engineering Science Lecture Notes, University of Oxford.

E. GELENBE, I. MITRANI (1980) *Analysis and Synthesis of Computer Systems*, Academic, London.

S.M. ROSS (1970) *Applied Probability Models with Optimization Applications*, Holden-Day, San Francisco.

S.M. ROSS (1976) *A First Course in Probability*, Macmillan, New York.

H.G. TUCKER (1967) *A Graduate Course in Probability*, Academic, New York.

A STOCHASTIC APPROACH TO HIERARCHICAL SCHEDULING

M.A.H. DEMPSTER
Balliol College, Oxford/IIASA, Laxenburg

Practical hierarchical planning problems typically involve a sequence of decisions over time at an increasing level of detail and with increasingly accurate information. For example, for manufacturing operations a 3-level hierarchy of planning decisions in terms of increasingly finer time units is often utilized. The first level concerns medium term planning which works with projected monthly production averages and is primarily concerned with the acquisition of certain resources. The next level treats weekly production scheduling, while the third level is concerned with the real-time sequencing of jobs through various machine centres on the shop floor. The first two levels can currently be handled adequately by deterministic linear programming and combinatorial permutation procedures, but the third realistically involves a network of stochastic m-machine scheduling problems whose natural setting is in continuous time.

More generally, many hierarchical planning problems can be modelled by multistage stochastic programmes whose later stages (lower levels) are stochastic versions of familiar NP-hard deterministic combinatorial optimization problems. Hence they usually require the use of approximations and heuristics for near-optimal solutions.

Recently, computer-based planning systems have become popular for practical multilevel decision problems [Dempster *et al.* 1981A]. In principle, the performance of such systems can be evaluated relative to optimality for the appropriate multi-stage stochastic programming model.

This paper primarily reports on a programme of research conducted jointly with M.L. Fisher, B.J. Lageweg, J.K. Lenstra and A.H.G. Rinnooy Kan.

After a brief survey of distributional assumptions on processing times under which SEPT and LEPT policies remain optimal for m-machine scheduling problems [Coffman 1981; Weber 1979; Weiss & Pinedo 1980; Pinedo 1981; Frederickson 1981], results are presented for various 2-level scheduling problems in which the first stage concerns the *acquisition* (or assignment) of machines [Dempster *et al.* 1981B]. For example, heuristics which are

asymptotically optimal in expectation as the number of jobs in the system increases are analyzed for problems whose second stages are either *identical* or *uniform* m-machine scheduling problems. A 3-level location, distribution and routing model in the plane is also discussed [Beardwood *et al.* 1959; Fisher & Hochbaum 1980; Hochbaum & Steele 1981; Marchetti Spaccamela *et al.* 1981].

References

J. BEARDWOOD, H.J. HALTON, J.M. HAMMERSLEY (1959) The shortest path through many points. *Proc. Cambridge Phil. Soc.* 55,299-327.

E.G. COFFMAN, JR. (1981) Probability models of sequencing and packing algorithms. To be presented at this Institute.

M.A.H. DEMPSTER, M.L. FISHER, L. JANSEN, B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1981A) Analytical evaluation of hierarchical planning systems. *Oper. Res.*, to appear.

M.A.H. DEMPSTER, M.L. FISHER, L. JANSEN, B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1981B) Analysis of heuristics for stochastic programming: results for hierarchical scheduling problems. Report BW 142, Mathematisch Centrum, Amsterdam.

M.L. FISHER, D.S. HOCHBAUM (1980) Probabilistic analysis of the planar K-median problem. *Math. Oper. Res.* 5,27-34.

G.N. FREDERICKSON (1981) Probabilistic analysis of the LPT processor scheduling heuristic. To be presented at this Institute.

D. HOCHBAUM, J.M. STEELE (1981) Steinhaus' geometric location problem for random samples in the plane. *J. Appl. Probab.*, to appear.

A. MARCHETTI SPACCAMELA, A.H.G. RINNOOY KAN, L. STOUGIE (1981) Hierarchical vehicle routing. To appear.

M. PINEDO (1981) On the complexity of stochastic scheduling problems. To be presented at this Institute.

R.R. WEBER (1979) *Optimal Organization of Multi-Server Systems*, Ph.D. Thesis, University of Cambridge.

G. WEISS, M. PINEDO (1980) Scheduling tasks with exponential service times on non-identical processors to minimize various cost function. *J. Appl. Probab.* 17,187-202.

ANALYSIS OF HEURISTICS

M.L. FISHER
University of Pennsylvania, Philadelphia

This lecture will closely follow sections 1 and 2 of [Fisher 1980]. Results on the analysis of scheduling heuristics will be deferred to the advanced lectures on scheduling. A discussion of the analysis of scheduling heuristics is also available in [Garey *et al.* 1978].

1. Introduction

1.1. Why study heuristics?
1.2. Short history of the study of heuristics
1.3. Three different approaches for measuring the performance of a heuristic
  (a) Empirical
  (b) Worst-case analysis
  (c) Probabilistic analysis

2. Fundamentals of worst-case analysis

2.1. Definition of the worst-case performance ratio of a heuristic
2.2. Illustration of determining the worst-case performance ratio for various knapsack heuristics
  (a) Greedy heuristics
  (b) Partial enumeration [Johnson 1974; Sahni 1975]
  (c) Dynamic programming with rounded data [Ibarra & Kim 1975; Lawler 1979]
2.3. General observations on the knapsack results

References

M.L. FISHER (1980) Worst-case analysis of heuristic algorithms. *Management Sci.* 26,1-14.

M.R. GAREY, R.L. GRAHAM, D.S. JOHNSON (1978) Performance guarantees for scheduling algorithms. *Oper. Res.* 26,3-21.

O.H. IBARRA, C.E. KIM (1975) Fast approximation algorithms for the knapsack and sum of subset problems. *J. Assoc. Comput. Mach.* 22,463-468.

D.S. JOHNSON (1974) Approximation algorithms for combinatorial problems. *J. Comput. System Sci.* 9,256-278.

E.L. LAWLER (1979) Fast approximation algorithms for knapsack problems. *Math. Oper. Res.* 4,339-356.

S. SAHNI (1975) Approximate algorithms for the 0-1 knapsack problem. *J. Assoc. Comput. Mach.* 22,115-124.

SINGLE MACHINE SCHEDULING TO MINIMIZE TOTAL COST

M.L. FISHER
University of Pennsylvania, Philadelphia

1. Introduction and definition of the generic single machine mini-sum problem

2. Polynomially solvable cases

2.1. Min weighted completion [Smith 1956]
2.2. Min number of late jobs [Moore 1968]

3. Min weighted tardiness

3.1. NP-complete [Lenstra *et al.* 1977]
3.2. Conditions on an optimal sequence [Emmons 1969]
3.3. Exact algorithms
  (a) Dynamic programming [Held & Karp 1962; Srinivasan 1971; Baker & Schrage 1978]
  (b) Pseudopolynomial [Lawler 1977]
  (c) Branch-and-bound
    - Assignment relaxation [Rinnooy Kan *et al.* 1975]
    - Lagrangian relaxation [Fisher 1973, 1976, 1981; Held & Karp 1971]
3.4. Analysis of heuristics
  (a) Greedy [Fisher & Krieger 1981]
  (b) Fully polynomial approximation scheme [Lawler 1977]

References

K.R. BAKER, L.E. SCHRAGE (1978) Finding an optimal sequence by dynamic programming: an extension to precedence-related tasks. *Oper. Res.* 26, 111-120.

H. EMMONS (1969) One machine sequencing to minimize certain functions of job tardiness. *Oper. Res.* 17,701-715.

M.L. FISHER (1973) Optimal solution of scheduling problems using Lagrange multipliers, part I. *Oper. Res.* 21,1114-1127.

M.L. FISHER (1976) A dual algorithm for the one-machine scheduling problem. *Math. Programming* 11,229-251.

M.L. FISHER (1981) The Lagrangian relaxation method for solving integer programming problems. *Management Sci.* 27,1-18.

M.L. FISHER, A.M. KRIEGER (1981) Analysis of a linearization heuristic for single-machine scheduling to maximize profit. Working Paper, Department of Decision Sciences, The Wharton School, University of Pennsylvania, Philadelphia.

M. HELD, R.M. KARP (1962) A dynamic programming approach to sequencing problems. *J. SIAM* 10,196-210.

M. HELD, R.M. KARP (1971) The traveling-salesman problem and minimum spanning trees: part II. *Math. Programming* 1,6-25.

E.L. LAWLER (1977) A "pseudopolynomial" algorithm for sequencing jobs to minimize total tardiness. *Ann. Discrete Math.* 1,331-342.

J.K. LENSTRA, A.H.G. RINNOOY KAN, P. BRUCKER (1977) Complexity of machine scheduling problems. *Ann. Discrete Math.* 1,343-362.

J.M. MOORE (1968) An *n* job, one machine sequencing algorithm for minimizing the number of late jobs. *Management Sci.* 15,102-109.

A.H.G. RINNOOY KAN, B.J. LAGEWEG, J.K. LENSTRA (1975) Minimizing total costs in one-machine scheduling. *Oper. Res.* 23,908-927.

W.E. SMITH (1956) Various optimizers for single-stage production. *Naval Res. Logist. Quart.* 3,59-66.

V. SRINIVASAN (1971) A hybrid algorithm for the one-machine sequencing problem to minimize total tardiness. *Naval Res. Logist. Quart.* 18,317-327.

NONPREEMPTIVE SCHEDULING OF PARALLEL MACHINES

M.L. FISHER
University of Pennsylvania, Philadelphia

For simplicity, I will restrict attention to the case of identical parallel machines and general process times. Results for many variations on this base case are reviewed in [Graham *et al.* 1979].

1. Minimize sum of job completion times [Conway *et al.* 1967]

2. Minimize time to complete all jobs

2.1. NP-hard [Lenstra *et al.* 1977]
2.2. Exact algorithms
  (a) Branch-and-bound [Bratley *et al.* 1975; Stern 1976]
  (b) Dynamic programming [Lawler & Moore 1969; Rothkopf 1966]
2.3. Approximation methods
  (a) Longest process time [Graham 1966, 1969]
  (b) Partial enumeration [Graham 1969]
  (c) Multi fit [Coffman *et al.* 1978]
  (d) Dynamic programming and rounding [Sahni 1976]

3. Variable number of machines and stochastic process times [Dempster *et al.* 1981A,1981B]

4. A large-scale real example

4.1. Description of the problem and its economic significance
4.2. Relationship of this problem to research on theoretical algorithms
4.3. Importance of stochastic elements

References

P. BRATLEY, M. FLORIAN, P. ROBILLARD (1975) Scheduling with earliest start and due date constraints on multiple machines. *Naval Res. Logist. Quart.* 22,165-173.

E.G. COFFMAN, JR., M.R. GAREY, D.S. JOHNSON (1978) An application of bin-packing to multiprocessor scheduling. *SIAM J. Comput.* 7,1-17.

R.W. CONWAY, W.L. MAXWELL, L.W. MILLER (1967) *Theory of Scheduling*, Addison-Wesley, Reading, MA.

M.A.H. DEMPSTER, M.L. FISHER, L. JANSEN, B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1981A) Analytical evaluation of hierarchical planning systems. *Oper. Res.*, to appear.

M.A.H. DEMPSTER, M.L. FISHER, L. JANSEN, B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1981B) Analysis of heuristics for stochastic programming: results for hierarchical scheduling problems. *Math. Oper. Res.*, in process.

R.L. GRAHAM (1966) Bounds for certain multiprocessing anomalies. *Bell System Tech. J.* 45,1563-1581.

R.L. GRAHAM (1969) Bounds on multiprocessing timing anomalies. *SIAM J. Appl. Math.* 17,263-269.

R.L. GRAHAM, E.L. LAWLER, J.K. LENSTRA, A.H.G. RINNOOY KAN (1979) Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann. Discrete Math.* 5,287-326.

E.L. LAWLER, J.M. MOORE (1969) A functional equation and its application to resource allocation and sequencing problems. *Management Sci.* 16,77-84.

J.K. LENSTRA, A.H.G. RINNOOY KAN, P. BRUCKER (1977) Complexity of machine scheduling problems. *Ann. Discrete Math.* 1,343-362.

M.H. ROTHKOPF (1966) Scheduling independent tasks on parallel processors. *Management Sci.* 12,437-447.

S. SAHNI (1976) Algorithms for scheduling independent tasks. *J. Assoc. Comput. Mach.* 23,116-127.

H.I. STERN (1976) Minimizing makespan for independent jobs on nonidentical parallel machines - an optimal procedure. Working Paper 2/75, Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Beer-Sheva.

ON STOCHASTIC ANALYSIS OF PROJECT-NETWORKS

W. GAUL
Universität Karlsruhe

If the activity-completion-times of a project-network are random variables, the project-completion-time is a random variable the distribution function of which is difficult to obtain.

Thus, after a survey on results to determine bounds for the mean and the variance and bounding distribution functions for the distribution function of the project-completion-time, a new approach using stochastic programming for a cost-oriented project scheduling model is presented. Completion-time estimates for the random activity-completion-times have to be computed where planned time-reductions increase costs and nonconformity with the actual realizations of the random activity-completion-times yields additional compensation costs (gains). Taking into consideration a prescribed project-completion-time constraint the expected costs for performing the activities according to the computed activity-completion-time estimates are minimized. The solution procedure constructs a finite sequence of non-stochastic network circulation problems.

Examples of application-relevant size can be presented.

SINGLE SERVER QUEUES

E. GELENBE

Université de Paris Sud/INRIA, Rocquencourt

Not received

SCHEDULING IN COMPUTER SYSTEMS AND NETWORKS

E. GELENBE

Université de Paris Sud/INRIA, Rocquencourt

Not received

DISCRETE TIME STOCHASTIC SCHEDULING

J.C. GITTINS
Keble College, Oxford

The optimality criterion under consideration in this lecture is that of maximum expected total discounted reward, where rewards are associated with the completion of each job. This choice is natural when the jobs under consideration are substantial projects, more of which may become available as time goes by. In the limit as the discount factor tends to one it is equivalent to minimising a weighted version of the total expected flow-time.

It is plausible that a policy which maximises the expected discounted reward per unit time up to an arbitrary time depending on the durations of the various jobs, and then continues, stage by stage, in the same way, should be a good policy. In fact [Gittins 1979,1982] such a policy, called a forwards induction policy, is often optimal. When this is so optimal policies often reduce to giving priority at each stage to that job for which the value of a certain index, which typically varies as work progresses on the job, is largest.

The circumstances under which forwards induction policies are optimal will be discussed, and the forms of the appropriate indices. These circumstances include certain types of precedence constraints between jobs. To a considerable extent the lecture will be based on [Gittins 1979,1982]. The other references listed are also relevant.

References

O.I. BRONSTEIN, I.M. DUHOVNYI (1976) *Priority Queueing Models in Computer Information Systems* (in Russian), Isdat. Nauka, Moscow.

M.A.H. DEMPSTER (1979) Some programs in dynamic stochastic scheduling. Presented at 2nd Internat. Conf. Stochastic Programming, Oberwolfach, Germany, February 2, 1979.

J.C. GITTINS (1979) Bandit processes and dynamic allocation indices. *J. Roy. Statist. Soc. Ser. B* 41,148-177.

J.C. GITTINS (1982) *Sequential Resource Allocation*, Wiley, Chichester, to appear.

J.C. GITTINS, K.D. GLAZEBROOK (1977) On Bayesian models in stochastic scheduling. *J. Appl. Probab.* 14,556-565.

J.C. GITTINS, D.M. JONES (1974A) A dynamic allocation index for the sequential design of experiments. In: J. GANI (ed.) (1974) *Progress in Statistics*, North-Holland, Amsterdam, 241-260.

J.C. GITTINS, D.M. JONES (1974B) A dynamic allocation index for new-product chemical research. CUED/A-MgtStud/TR13, Engineering Department, University of Cambridge.

J.C. GITTINS, P. NASH (1977) Scheduling, queues, and dynamic allocation indices. *Proc. EMS, Prague 1974*, Czechoslovak Academy of Sciences, Prague, 191-202.

K.D. GLAZEBROOK (1976A) Stochastic scheduling with order constraints. *Internat. J. Systems Sci.* 7,657-666.

K.D. GLAZEBROOK (1976B) A profitability index for alternative research projects. *Omega* 4,79-83.

K.D. GLAZEBROOK (1979A) Stoppable families of alternative bandit processes. *J. Appl. Probab.* 16,843-854.

K.D. GLAZEBROOK (1979B) Decomposable jump processes. *Stochastic Process. Appl.* 9,19-34.

K.D. GLAZEBROOK (1980A) On stochastic scheduling with precedence relations and switching costs. *J. Appl. Probab.* 17,1016-1024.

K.D. GLAZEBROOK (1980B) On single-machine sequencing with order constraints. *Naval Res. Logist. Quart.* 27,123-130.

K.D. GLAZEBROOK, J.C. GITTINS (1981) On single-machine scheduling with precedence relations and linear or discounted costs. *Oper. Res.* 29,to appear.

G.J. HALL (1978) Nonstationary stochastic gold-mining: a time-sequential tactical allocation problem. *Naval Res. Logist. Quart.* 25,81-93.

J.B. KADANE, H.A. SIMON (1977) Optimal strategies for a class of constrained sequential problems. *Ann. Statist.* 5,237-255.

G. KIMELDORF, F.H. SMITH (1979) Binomial searching for a random number of multinomially hidden objects. *Management Sci.* 25,1115-1126.

M. LEHNERDT (1980) *On the Structure of Discrete Sequential Search Problems and of their Solutions*, Doctoral Dissertation, University of Hamburg.

I. MEILIJSON, G. WEISS (1977) Multiple feedback at a single server station. *Stochastic Process. Appl.* 5,195-205.

I. MEILIJSON, U. YECHIALI (1977) On optimal right-of-way policies at a single-server station when insertion of idle times is permitted. *Stochastic Process. Appl.* 6,25-32.

P. NASH (1973) *Optimal Allocation of Resources between Research Projects*, Ph.D. Thesis, University of Cambridge.

P. NASH (1980) A generalised bandit problem. *J. Roy. Statist. Soc. Ser. B* 42,165-169.

P. NASH, J.C. GITTINS (1977) A Hamiltonian approach to optimal stochastic resource allocation. *Adv. in Appl. Probab.* 9,55-68.

M. PINEDO (1980) Spheres with exponential life-times in a system with two parallel components. *J. Appl. Probab.* 17,1025-1032.

K.W.S. ROBERTS, M.L. WEITZMAN (1980) On a general approach to search and information gathering. Working Paper 263, Department of Economics, M.I.T., Cambridge, MA.

J.B. SIDNEY (1975) Decomposition algorithms for single machine sequencing with precedence relations and deferral costs. *Oper. Res.* 23,283-298.

A. SIMONOVITS (1973) Direct comparison of different priority queueing disciplines. *Studia Sci. Math. Hungar.* 8,225-243.

D.-W. TCHA, S.R. PLISKA (1977) Optimal control of single-server queueing networks and multi-class M/G/1 queues with feedback. *Oper. Res.* 25, 248-258.

R.R. WEBER (1980) *Optimal Organisation of Multi-Server Systems*, Doctoral Dissertation, University of Cambridge.

G. WEISS, M. PINEDO (1980) Scheduling tasks with exponential service times on non-identical processors to minimize various cost functions. *J. Appl. Probab.* 17,187-202.

M.L. WEITZMAN (1979) Optimal search for the best alternative. *Econometrica* 47,641-654.

P. WHITTLE (1980) Multi-armed bandits and the Gittins index. *J. Roy. Statist. Soc. Ser. B* 42,143-149.

DESIGN AND ANALYSIS OF ALGORITHMS FOR COMBINATORIAL OPTIMIZATION

E.L. LAWLER
University of California, Berkeley

0. Introduction

Much of the material I propose to present is "bread-and-butter" for computer scientists. Other material is well known to the operations research community. Because of the diverse backgrounds of participants in this conference, I have chosen to emphasize very basic topics, but to try to indicate one or two results that may be new even to those who are familiar with most of the subject area.

Topics to be discussed are:

1. Standard problem formulations of combinatorial optimization.
2. Generally applicable techniques for solving combinatorial problems, *e.g.* divide-and-conquer, dynamic programming, branch-and-bound.
3. Fundamental data structures needed for the implementation of algorithms.
4. "Time" and "space" as measures of the effectiveness of combinatorial algorithms.
5. Methods for analyzing time and space requirements of specific algorithms.
6. Possible trade-offs between time and space.
7. Techniques for establishing lower bounds on time.

1. Standard problem formulations

Certain standard combinatorial optimization problems occur with considerable frequency in the subject area of deterministic scheduling. Among these are:

Traveling Salesman Problem,
Assignment Problem,
Knapsack Problem,
Quadratic Assignment Problem,
Bin Packing Problem,
Set Covering Problem,
Chromatic Number Problem.

It will be indicated, by example, how each of these problem types arises in scheduling theory.

2. Techniques for solving combinatorial optimization problems

Sometimes it is possible to formulate and solve a combinatorial problem by a standard and well understood technique, *e.g.* linear programming, network flows, the "greedy" algorithm. Often it is necessary to devise a special algorithm for the problem at hand.

One general approach is that of "divide-and-conquer". As an example, consider sorting by merging. To sort n numbers, divide them into two sets of nearly equal size ($\lfloor \frac{n}{2} \rfloor$ and $\lceil \frac{n}{2} \rceil$), sort these smaller sets (by recursive application of the same procedure) and merge the two sorted sets, with at most n-1 additional comparisons. Let c(n) denote the number of comparisons required, in the worst case, to sort n numbers by this method. For simplicity, let $n = 2^k$. Then we have:

$$c(n) = 2c(\tfrac{n}{2}) + (n-1), \tag{1}$$

$$c(1) = 0,$$

and so

$$c(n) = n \log n - n + 1.$$

Dynamic programming is another useful technique. Consider the case of a single-machine sequencing problem in which there are n jobs, $j = 1,2,\ldots,n$, for each of which there is a specified processing requirement p_j and a cost function f_j. If job j is completed at time t, the cost incurred for job j is $f_j(t)$. The object is to sequence the jobs so that the sum of the costs is minimized. Let $S \subseteq \{1,2,\ldots,n\}$, and let F(S) denote the minimum cost of a schedule for the subset S. Then:

$$F(S) = \min_{k \in S} \{F(S-\{k\}) + f_k(p(S))\}, \tag{2}$$

where

$$p(S) = \sum_{j \in S} p_j,$$

and with the initial condition

$$F(\emptyset) = 0.$$

There are 2^n subsets S for which equation (2) is to be solved, and each equation requires at most n additions and n-1 comparisons. Hence the time and space requirements are $O(n2^n)$ and $O(2^n)$, respectively.

Branch-and-bound is a widely applicable technique for solving combinatorial optimization problems. For example, one could devise an algorithm in which each node of the branch-and-bound search tree is identified with a sequence $\Pi(S)$ for some subset of jobs S. "Branching" can then be performed on the choice of the next job $k \in S$. A possible lower bound on the cost of completing the sequence is given by:

$$\sum_{j \notin S} f_j(p(S \cup \{j\})).$$

As another example, consider the problem of determining whether a given digraph G = (N,A) contains a Hamilton path between two specified nodes. Let N = {0,1,...,n,n+1} and suppose a Hamilton path is sought from node 0 to node n+1. A well-known method of solution by dynamic programming is as follows. For arbitrary $S \subseteq \{1,2,...,n\}$, let F be a Boolean-valued function such that:

$$F(S,j) = \begin{cases} 1 & \text{if there is a path from node 0 to node j passing through each of the nodes in S exactly once (and through no other nodes),} \\ 0 & \text{otherwise.} \end{cases}$$

Then we have the recurrence

$$F(S,j) = \bigvee_{k \in S} (F(S-\{k\},k) \wedge a_{kj}),$$

where

$$a_{kj} = \begin{cases} 1 & \text{if there is an arc from k to j in G,} \\ 0 & \text{otherwise,} \end{cases}$$

$$F(\emptyset,j) = a_{0j}.$$

The value of $F(\{1,2,\ldots,n\},n+1)$ can be computed in $O(n^2 2^n)$ time and $O(n2^n)$ space.

A very dramatic reduction in space requirements can be effected by the use of the principle of inclusion and exclusion. (This idea is due to R.M. Karp.) Let $w(S)$ denote the number of walks of length $n+1$ from node 0 to node $n+1$ which do not pass through any nodes in S. (A walk is like a directed path, but with repetitions of arcs permitted.) By inclusion and exclusion, the number of Hamilton paths is equal to

$$\sum_{|S| \text{even}} w(S) - \sum_{|S| \text{odd}} w(S).$$

For given S, $w(S)$ can easily be computed in $O(n^3)$ time and $O(n)$ space. It follows that the existence of a Hamilton path can be determined in $O(n^3 2^n)$ time and only $O(n)$ space (in addition to the space required to specify the digraph).

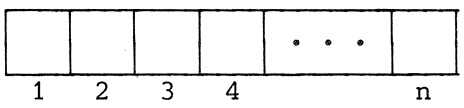
Although the same trick can be applied to the problem solved by recurrence (2), it appears that the time-space trade-off which can be achieved is not nearly so impressive.

Estimates of the time requirements for branch-and-bound algorithms are nearly always horrifying. The effectiveness of branch-and-bound algorithms must generally be demonstrated by empirical tests. (One remarkable exception is a branch-and-bound algorithm devised by D. Matula for the purpose of finding a subgraph of maximum connectivity.)

3. Data structures

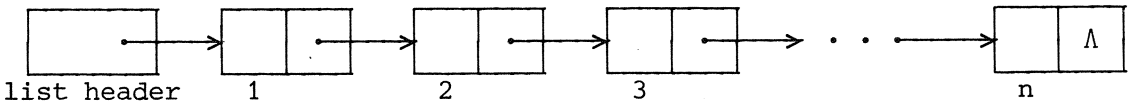
It is usually not possible to make a meaningful theoretical analysis of the efficiency of an algorithm without anticipating some details of implementation. This means thinking about data structures, which we review briefly.

A sequence of n numbers may be stored in an *array* or in a *list*. An array can be thought of as a sequence of consecutive locations in memory. Pictorially:



A list consists of a set of records, presumably dispersed throughout memory,

joined together by *links* or *pointers*. Pictorially:



An array is easy to program and provides access to any given element in constant time. However, it is not well suited for adding and deleting elements, and there may be severe complications if the size of the array is not known in advance, or if its size changes dynamically. A list is more complicated to program, and does not provide easy access to any given element. (O(n) time is required.) However, it is well suited for insertions and deletions, and it is particularly well suited for applications in which the number of elements is not known in advance, or the number changes dynamically.

A list is a particularly good way to implement a *stack*, which operates as a LIFO storage device. By providing a pointer to the last entry in a list, one can implement a *queue*, which operates in FIFO mode.

A *priority queue* is a data structure which is intended to support the operations

FINDMAX,
DELETEMAX,
INSERT X.

A *dictionary* is a data structure which is intended to support the operations

FIND X,
INSERT X,
DELETE X.

Dictionaries are commonly implemented by means of hash tables and search trees. Properly constructed hash tables allow dictionary operations to be performed in effectively (but not theoretically) constant time.

A very useful type of data structure is one which performs the operations

FIND X,
UNION i,j,

where FIND X means "find the name of the equivalence class of which X is a member", and UNION i,j means to join the existing equivalence classes i and j.

4. "Time" and "space"

When analyzing algorithms the computer scientist ordinarily does not try to estimate "time" in the sense of milliseconds of running time or "space" in the sense of words of storage. Instead, he adopts some measure which abstracts and (hopefully) approximates these notions. Example: In the case of algorithms for sorting, it is common to count only the number of comparisons performed.

Two measures of time and space are commonly employed: *worst case* and *average case*. Worst case analysis is usually easier, but pessimistic. Average case analysis is often complicated by the difficulty of determining a realistic probability distribution for problem instances.

We mention here the concept of polynomial-time boundedness, and the reasons for its importance.

5. Analysis of time and space requirements

Worst-case estimates of time requirements of combinatorial algorithms are obtained in various ways: by loop-counting, by solution of recurrence relations like (1), or by direct counting arguments, as in the case of the dynamic programming equations (2).

6. Time-space trade-offs

One well-known example of time space trade-off is "depth-first" *vs*. "breadth-first" search in branch-and-bound.

7. Methods of lower bounding

The methodology of lower bounding has been most highly developed in the case of algorithms based on comparisons of key values, *e.g.* for problems in sorting and searching, merging lists, and so forth. Lower bounding arguments may involve:

(1) combinatorial analysis of decision trees,

(2) arguments involving "adversaries" or "oracles",

(3) analysis of state descriptions.

Possibly the most common lower bound is the "information theoretic" lower bound: If there are N possible answers, and these are to be obtained by making comparisons only, then any algorithm for solving the problem must make at least $\lceil \log_2 N \rceil$ comparisons, in the worst case. Thus, for example, at least $\lceil \log_2 n! \rceil = O(n \log n)$ comparisons must be performed, in the worst case, by any algorithm which sorts n numbers.

It should be noted that the information-theoretic lower bound for sorting can be applied to show that certain scheduling algorithms are "optimal", *e.g.* Smith's rule for minimizing total weighted completion time.

References

A.V. AHO, J.E. HOPCROFT, J.D. ULLMAN (1976) *The Design and Analysis of Computer Algorithms*, Addison-Wesley.

E. HOROWITZ, S. SAHNI (1978) *Fundamentals of Computer Algorithms*, Computer Science Press.

D.E. KNUTH (1980) *The Art of Computer Programming*, Vols. 1, 3 (2nd ed.), Addison-Wesley.

E.L. LAWLER (1976) *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston.

SCHEDULING PRECEDENCE-CONSTRAINED UNIT-TIME JOBS ON PARALLEL MACHINES

E.L. LAWLER
University of California, Berkeley

Scheduling of unit-time jobs on parallel machines is generally an easy matter if the jobs are independent. Such problems can often be formulated and solved as assignment or transportation problems. When precedence constraints exist, the problems become much more interesting and challenging.

One of the older and more important results of deterministic scheduling theory is a simple and elegant algorithm of T.C. Hu [Hu 1961] for the scheduling of unit-time jobs on any number of identical machines, when the precedence constraints are in the form of a rooted tree. Brucker, Garey and Johnson [Brucker *et al.* 1977] succeeded in generalizing Hu's procedure (which simply minimizes the length of the schedule) to minimize maximum lateness, provided the precedence constraints are in the form of an *intree*. They also showed that the corresponding problem is NP-hard for *outtrees*.

Another substantial advance occurred when Fujii, Kasami and Ninomiya [Fujii *et al.* 1969, 1971] provided a polynomial algorithm for the scheduling of unit-time jobs on two machines, with completely arbitrary precedence constraints. Soon after, Coffman and Graham [Coffman & Graham 1972] suggested a faster algorithm; a variation of this algorithm by Gabow [Gabow 1980] enables one to find a minimum-length schedule in *nearly* linear time. Garey and Johnson [Garey & Johnson 1976, 1977] have provided algorithms for minimizing maximum lateness, even in the presence of arbitrary release dates for jobs.

We shall review the algorithms mentioned above, and also indicate some recent results from [Simons 1980; Warmuth 1980; Dolev 1981; Garey *et al.* 1981].

We conclude by noting that the "three-processor" problem is one of the most vexing open questions in deterministic scheduling.

References

P. BRUCKER, M.R. GAREY, D.S. JOHNSON (1977) Scheduling equal-length tasks under treelike precedence constraints to minimize maximum lateness. *Math. Oper. Res.* 2,275-284.

E.G. COFFMAN, JR., R.L. GRAHAM (1972) Optimal scheduling for two-processor systems. *Acta Inform.* 1,200-213.

D. DOLEV (1981) Scheduling wide graphs. Unpublished manuscript.

M. FUJII, T. KASAMI, K. NINOMIYA (1969) Optimal sequencing of two equivalent processors. *SIAM J. Appl. Math.* 17,784-789.

M. FUJII, T. KASAMI, K. NINOMIYA (1971) Erratum. *SIAM J. Appl. Math.* 20,141.

H.N. GABOW (1980) An almost-linear algorithm for two processor scheduling. Technical Report CU-CS-169-80, Department of Computer Science, University of Colorado, Boulder.

M.R. GAREY, D.S. JOHNSON (1976) Scheduling tasks with nonuniform deadlines on two processors. *J. Assoc. Comput. Mach.* 23,461-467.

M.R. GAREY, D.S. JOHNSON (1977) Two-processor scheduling with start-times and deadlines. *SIAM J. Comput.* 6,416-426.

M.R. GAREY, D.S. JOHNSON, R.E. TARJAN, M. YANNAKAKIS (1981) Scheduling opposing forests. Unpublished manuscript.

T.C. HU (1961) Parallel sequencing and assembly line problems. *Oper. Res.* 9, 841-848.

B. SIMONS (1980) A fast algorithm for multiprocessor scheduling. *Proc. 21st Annual IEEE Symp. Foundations of Computer Science*, 50-53.

M.K. WARMUTH (1980) M Processor unit-execution-time scheduling reduces to M-1 weakly connected components. M.S. Thesis, Department of Computer Science, University of Colorado, Boulder.

PREEMPTIVE SCHEDULING OF PARALLEL MACHINES

E.L. LAWLER
University of California, Berkeley

In this lecture we review a number of algorithmic results concerning the preemptive scheduling of parallel machines. We shall generally assume the machines are *uniform*, *i.e.* differing only in the speed with which they can process jobs.

Topics to be covered include:

(1) Minimizing the sum of completion times [Gonzalez 1977].

(2) Minimizing schedule length [Gonzalez & Sahni 1978].

(3) Minimizing schedule length, with release times (or alternatively, minimizing maximum lateness) [Sahni & Cho 1979; Labetoulle *et al.* 1979].

(4) Minimizing the weighted number of late jobs [Lawler 1979].

(5) Coping with precedence constraints.

In the case of the last topic, we shall point out similarities with procedures for dealing with unit-time tasks.

References

T. GONZALEZ (1977) Optimal mean finish time preemptive schedules. Technical Report 220, Computer Science Department, Pennsylvania State University, University Park, PA.

T. GONZALEZ, S. SAHNI (1978) Preemptive scheduling of uniform processor systems. *J. Assoc. Comput. Mach.* 25,92-101.

J. LABETOULLE, E.L. LAWLER, J.K. LENSTRA, A.H.G. RINNOOY KAN (1979) Preemptive scheduling of uniform machines subject to release dates. Report BW 99, Mathematisch Centrum, Amsterdam.

E.L. LAWLER (1979) Preemptive scheduling of uniform parallel machines to minimize the weighted number of late jobs. Report BW 105, Mathematisch Centrum, Amsterdam.

S. SAHNI, Y. CHO (1979) Nearly on line scheduling of a uniform processor system with release times. *SIAM J. Comput.* 8,275-285.

COMPUTATIONAL COMPLEXITY OF COMBINATORIAL PROBLEMS

J.K. LENSTRA
Mathematisch Centrum, Amsterdam

The inherent computational complexity of a combinatorial problem obviously has to be related to the computational behavior of algorithms designed for its solution. This behavior is usually measured by the *running time* of the algorithm (*i.e.*, the number of elementary operations such as additions and comparisons) as related to the *size* of the problem (*i.e.*, the number of bits occupied by the data).

If a problem of size n can be solved by an algorithm with running time $O(p(n))$ where p is a *polynomial* function, then the algorithm may be called *good* and the problem *well solved*. These notions were introduced by Edmonds [Edmonds 1965] in the context of the matching problem; his algorithm can be implemented to run in $O(n^3)$ time on graphs with n vertices. Polynomial algorithms have been developed for a wide variety of combinatorial optimization problems [Lawler 1976]. On the other hand, many such problems can only be solved by enumerative methods which may require *exponential* time.

When encountering a combinatorial problem, one would naturally like to know if a polynomial algorithm exists or if, on the contrary, any solution method must require exponential time in the worst case. Results of the latter type are still rare, but it is often possible to show that the existence of a polynomial algorithm is at the very least extremely unlikely. One may arrive at such a result by proving that the problem in question is *NP-complete* [Cook 1971; Karp 1972]. The NP-complete problems are equivalent in the sense that none of them has been well solved and that, if one of them would be well solved, then the same would be true for all of them. Since all the classical problems that are notorious for their computational intractability, such as traveling salesman, job shop scheduling and integer programming problems, are known to be NP-complete, the polynomial-time solution of such a problem would be very surprising indeed. For practical purposes, this implies that in solving those problems one may just as well accept the inevitability of a bad (superpolynomial) *optimization* algorithm or resort to using a good (polynomial) *approximation* algorithm.

References

S.A. COOK (1971) The complexity of theorem-proving procedures. *Proc. 3rd Annual ACM Symp. Theory of Computing*, 151-158.

J. EDMONDS (1965) Paths, trees, and flowers. *Canad. J. Math.* 17,449-467.

M.R. GAREY, D.S. JOHNSON (1979) *Computers and Intractability: a Guide to the Theory of NP-Completeness*, Freeman, San Francisco.

R.M. KARP (1972) Reducibility among combinatorial problems. In: R.E. MILLER, J.W. THATCHER (eds.) (1972) *Complexity of Computer Computations*, Plenum Press, New York, 85-103.

R.M. KARP (1975) On the computational complexity of combinatorial problems. *Networks* 5,45-68.

E.L. LAWLER (1976) *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York.

J.K. LENSTRA, A.H.G. RINNOOY KAN (1979) Computational complexity of discrete optimization problems. *Ann. Discrete Math.* 4,121-140.

SINGLE MACHINE SCHEDULING TO MINIMIZE MAXIMUM COST

J.K. LENSTRA
Mathematisch Centrum, Amsterdam

Suppose n jobs are to be processed on a single machine, subject to release dates and precedence constraints. The problem is to find a schedule that minimizes the maximum job completion cost.

If no preemption (job splitting) is allowed, the case of equal release dates if solvable in $O(n^2)$ time [Lawler 1973] and the case of arbitrary release dates is NP-hard in the strong sense [Garey & Johnson 1977; Lenstra *et al.* 1977].

The latter result is still true if no precedence constraints are specified and the maximum lateness is to be minimized. This problem has received ample attention in the literature. It possesses an interesting symmetric structure [Lageweg *et al.* 1976] and has important applications in job shop scheduling theory [McMahon & Florian 1975; Lageweg *et al.* 1977]. Branch-and-bound algorithms have been designed in [Baker & Su 1974; McMahon & Florian 1975; Lageweg *et al.* 1976] and approximation algorithms have been theoretically analyzed in [Kise *et al.* 1979; Potts 1980]. A polynomial time optimization algorithm for the special case of equal processing times was proposed in [Simons 1978].

If preemption is permitted, the most general case of arbitrary release dates, arbitrary precedence constraints and arbitrary nondecreasing cost functions is solvable in $O(n^2)$ time [Baker *et al.* 1980], which generalizes the result of [Lawler 1973].

References

K.R. BAKER, E.L. LAWLER, J.K. LENSTRA, A.H.G. RINNOOY KAN (1980) Preemptive scheduling of a single machine to minimize maximum cost subject to release dates and precedence constraints. Report BW 128, Mathematisch Centrum, Amsterdam.

K.R. BAKER, Z.-S. SU (1974) Sequencing with due-dates and early start times to minimize maximum tardiness. *Naval Res. Logist. Quart.* 21,171-176.

M.R. GAREY, D.S. JOHNSON (1977) Two-processor scheduling with starttimes and deadlines. *SIAM J. Comput.* 6,416-426.

H. KISE, T. IBARAKI, H. MINE (1979) Performance analysis of six approximation algorithms for the one-machine maximum lateness scheduling problem with ready times. *J. Oper. Res. Soc. Japan* 22,205-224.

B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1976) Minimizing maximum lateness on one machine: computational experience and some applications. *Statist. Neerlandica* 30,25-41.

B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1977) Job-shop scheduling by implicit enumeration. *Management Sci.* 24,441-450.

E.L. LAWLER (1973) Optimal sequencing of a single machine subject to precedence constraints. *Management Sci.* 19,544-546.

J.K. LENSTRA, A.H.G. RINNOOY KAN, P. BRUCKER (1977) Complexity of machine scheduling problems. *Ann. Discrete Math.* 1,343-362.

G.B. McMAHON, M. FLORIAN (1975) On scheduling with ready times and due dates to minimize maximum lateness. *Oper. Res.* 23,475-482.

C.N. POTTS (1980) Analysis of a heuristic for one machine sequencing with release dates and delivery times. *Oper. Res.* 28,1436-1441.

B. SIMONS (1978) A fast algorithm for single processor scheduling. *Proc. 19th Annual IEEE Symp. Foundations of Computer Science*, 246-252.

PREEMPTIVE SCHEDULING OF UNIFORM MACHINES WITH RELEASE TIMES AND DEADLINES

C.U. MARTEL
University of California, Davis

Given n jobs each of which has a release time, a deadline, and a processing requirement, we examine the problem of determining whether there exists a preemptive schedule on m uniform machines which completes each job in the time interval between its release time and its deadline. An $O(mn^5)$ algorithm is presented which uses a generalization of network flow techniques to construct such a schedule whenever one exists. This algorithm is then used with search techniques to find a schedule which minimizes maximum lateness.

STOCHASTIC SHOP SCHEDULING: A SURVEY

M. PINEDO
Georgia Institute of Technology, Atlanta

L.E. SCHRAGE
University of Chicago

In this paper a survey is presented of some of the recent results in stochastic shop scheduling. The models dealt with include open shops, flow shops and job shops. For the majority of the models we shall call a policy optimal if it minimizes the expected completion time of the last job, *i.e.* the expected makespan. We discuss the difficulties encountered when other objectives are desired. The two machine open shop is discussed in detail. For this model optimal policies are presented in case the jobs have exponentially distributed processing times. For flow shops two different versions are treated: (1) infinite storage space between the machines (no blocking), and (2) no storage space between the machines (blocking possible). Optimal policies can be found easily for both versions when there are two machines and the jobs have exponentially distributed processing times. Some additional results are presented for the m machine case with and without intermediate storage. Optimal policies are presented also for the two machine job shop when the jobs have exponentially distributed processing times.

References

B. AVI-ITZHAK (1965) A sequence of service stations with arbitrary input and regular service times. *Management Sci.* 11,565-573.

P.C. BAGGA (1970) N-job, 2 machine sequencing problem with stochastic service times. *Opsearch* 7,184-199.

J. BRUNO, P. DOWNEY, G.N. FREDERICKSON (1981) Sequencing tasks with exponential service times to minimize the expected flow time or makespan. *J. Assoc. Comput. Mach.* 28,100-113.

H.D. FRIEDMAN (1965) Reduction methods for tandem queuing systems. *Oper. Res.* 13,121-131.

F.S. HILLIER, R.W. BOLING (1967) Finite queues in series with exponential or Erlang service times - a numerical approach. *Oper. Res.* 15,286-303.

E. MUTH (1979) The reversibility property of production lines. *Management Sci.* 25,152-159.

G. OLIVIER (1972) Kostenminimale Prioritäten in Wortesystemen vom Type M/G/1. *Elektron. Rechenanlagen* 14,262-271.

M.L. PINEDO (1981) Minimizing the makespan in a stochastic flow shop. *Oper. Res.*, to appear.

M.L. PINEDO (1981) On the optimal order of stations in tandem queues. *Proc. Symp. Applied Probability and Computer Science: the Interface*, to appear.

M.L. PINEDO (1981) A note on the two machine job shop with exponential processing times. *Naval Res. Logist. Quart.*, to appear.

M.L. PINEDO (1981) Minimizing the expected flow time in a stochastic open shop. In preparation.

M.L. PINEDO, S.M. ROSS (1980) Scheduling jobs subject to nonhomogeneous Poisson shocks. *Management Sci.* 26,1250-1257.

M.L. PINEDO, S.M. ROSS (1981) Minimizing the expected makespan in a stochastic open shop. *J. Appl. Probab.*, in process.

M. PINEDO, G. WEISS (1979) Scheduling stochastic tasks on two parallel processors. *Naval Res. Logist. Quart.* 26,527-535.

M.H. ROTHKOPF (1966) Scheduling with random service times. *Management Sci.* 12,707-713.

K. SEVCIK (1974) Scheduling for minimum expected loss using service time distributions. *J. Assoc. Comput. Mach.* 21,66-75.

S.V. TEMBE, R.W. WOLFF (1974) The optimal order of service in tandem queues. *Oper. Res.* 22,824-833.

L. VAN DER HEYDEN (1979) A note on scheduling jobs with exponential processing times on identical processors so as to minimize makespan. *Math. Oper. Res.*

G. WEISS, M. PINEDO (1980) Scheduling tasks with exponential service times on nonidentical processors to minimize various cost functions. *J. Appl. Probab.* 17,187-202.

R.R. WEBER (1979) The interchangeability of tandem ·/M/1 queues in series. *J. Appl. Probab.* 16,690-695.

ENUMERATIVE METHODS

A.H.G. RINNOOY KAN
Erasmus University, Rotterdam

The only way to solve a scheduling problem to optimality is often to submit it to an *enumerative approach* in which all feasible solutions are implicitly or explicitly considered. Although the time required by such approaches generally grows as an exponential function of problem size in the worst case, the average case behavior of some of the more sophisticated methods can be quite satisfactory.

To avoid the inspection of every single feasible schedule, one usually tries to find sharp *lower bounds* on the quality of a subset of schedules; if the lower bound exceeds the value of a schedule found already, the subset can be discarded. *Dynamic programming* techniques [Held & Karp 1962] can be viewed as a special case of these *branch-and-bound* approaches [Lawler & Wood 1966].

The design of a successful enumerative method is strongly problem dependent. We will illustrate the principle by describing several enumerative approaches [Fisher 1976; Held & Karp 1962] to the problem of minimizing the total costs of scheduling a number of jobs on a single machine.

References

M.L. FISHER (1976) A dual algorithm for the one-machine scheduling problem. *Math. Programming* 11,229-251.

M. HELD, R.M. KARP (1962) A dynamic programming approach to sequencing problems. *J. SIAM* 10,196-210.

E.L. LAWLER, D.E. WOOD (1966) Branch-and-bound methods: a survey. *Oper. Res.* 14,699-719.

OPEN SHOP, FLOW SHOP AND JOB SHOP PROBLEMS

A.H.G. RINNOOY KAN
Erasmus University, Rotterdam

In *multi-operation models* a job consists of a number of operations, each of which has to be executed on a particular machine. If the operations can be executed in any order (though not simultaneously), we have an *open shop* model. If the operations have to be executed in a prespecified order, the model is called a *flow shop* if this order is the same one for each job and a *job shop* if this is not necessarily the case.

For all these models, the criterion that has been studied most frequently is the minimization of the time required to process all the jobs. Barring a few exceptions that occur when the number of machines is equal to two [Johnson 1954] or when preemption is allowed [Gonzalez & Sahni 1976] all these problems are very difficult and can only be solved to optimality by enumerative methods [Lageweg *et al.* 1977,1978]. Both special cases that can be solved efficiently and branch-and-bound techniques for general cases will be discussed. In the case of the general job shop problem, an ingenious problem representation known as the *disjunctive graph* model [Roy & Sussmann 1964] will pay a crucial role.

References

T. GONZALEZ, S. SAHNI (1976) Open shop scheduling to minimize finish time. *J. Assoc. Comput. Mach.* 23,665-679.

S.M. JOHNSON (1954) Optimal two- and three-stage production schedules with setup times included. *Naval Res. Logist. Quart.* 1,61-68.

B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1977) Job-shop scheduling by implicit enumeration. *Management Sci.* 24,441-450.

B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1978) A general bounding scheme for the permutation flow-shop problem. *Oper. Res.* 26,53-67.

B. ROY, B. SUSSMANN (1964) Les problèmes d'ordonnancement avec contraintes disjonctives. Note DS No.9 bis, SEMA, Montrouge.

DISCRETE STATE STOCHASTIC SYSTEMS

S.M. ROSS
University of California, Berkeley

We will present some of the major results in Markov chain theory and then consider applications to (i) modelling of algorithmic efficiency, (ii) optimal computer list scheduling, and (iii) theory of runs. The first application is an attempt to obtain a simple model so as to give an intuitive feel as to why the simplex algorithm of linear programming performs much more efficiently in practice than one might suppose by a consideration of worst case principles. The second application deals with the determination of the optimal way to reorder a list of elements when every unit of time an element is selected with some fixed (but unknown) probability and no memory of previous selections is allowed.

Some results on time reversible chains will also be considered.

MULTI-SERVER QUEUES

S.M. ROSS
University of California, Berkeley

We will survey a variety of multiserver models in which the arrival stream is a Poisson process. In particular, we will consider the Erlang loss model in which arrivals finding all servers busy are lost. In this system, we assume a general service distribution. We will also consider finite and infinite capacity versions of this model. Another model of this type is the shared processor system in which service is shared by all customers.

Another model to be considered is the G/M/k in which arrivals are in accordance with a renewal process and the service distribution is exponential. We will analyze this model by means of the embedded Markov chain approach.

References

F. KELLY (1979) *Reversibility and Stochastic Networks*, Wiley, Chichester.

S. NOZAKI, S. ROSS (1978) Approximations in finite capacity multi-server queues with Poisson arrivals. *J. Appl. Probab.* 15,826-834.

S. ROSS (1980) *Introduction to Probability Models*, Second Edition, Academic Press, New York.

THE MULTIPRODUCT LOT SCHEDULING PROBLEM

L.E. SCHRAGE
University of Chicago

An NP-hard problem of considerable practical interest is the multi-product lot scheduling problem. In its simplest form there are P products to be scheduled on a single machine over a finite interval (0,T). Associated with each product i is a due date d_i, a per unit time holding cost h_i, a processing time p_i and a changeover cost vector C_{ji} which is the cost of starting production on i if the machine previously produced product j. In practical problems one might wish to treat the d_i and p_i as random variables, although this feature is typically disregarded by solution procedures. Example situations might be a television manufacturer who produces several different styles and sizes of televisions on a single line or a chemical processor who produces several different chemicals in batches on a single expensive machine. We briefly summarize previous approaches to this problem starting with the work of Manne, Dzielinski, Gomory, Lasdon and Terjung and then analyze LP-like approximations to this model and provide bounds on the closeness of the LP solution to the exact IP solution as the problem size gets large.

References

P. AFENTAKIS, B. GAVISH, U. KARMARKAR (1980) Exact solutions to the lot-sizing problem in multistage assembly systems. Technical Report, University of Rochester, NY.

K.R. BAKER, P.S. DIXON, M.J. MAGAZINE, E.A. SILVER (1978) An algorithm for the dynamic lot-size problem with time-varying production capacity constraints. *Management Sci.* 24,1710-1720.

G.R. BITRAN, H.H. YANASSE (1981) Computational complexity of the capacitated lot size problem. Technical Report, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA.

J. BLACKBURN, R. MILLEN (1979) Selecting a lot-sizing technique for a single-level assembly process: part I - analytical results. *Production and Inventory Management*, 3rd Quarter.

J. BLACKBURN, R. MILLEN (1979) Selecting a lot-sizing technique for a single-level assembly process: part II - empirical results. *Production and Inventory Management*, 4th Quarter.

W. CROWSTON, M. WAGNER, J. WILLIAMS (1973) Economic lotsize determination in multi-stage assembly systems. *Management Sci.*

M.A.H. DEMPSTER, M.L. FISHER, L. JANSEN, B.J. LAGEWEG, J.K. LENSTRA, A.H.G. RINNOOY KAN (1981) Analytical evaluation of hierarchical planning systems. *Oper. Res.*, to appear.

C. DELPORTE, J. THOMAS (1977) Lotsizing and sequencing for N products on one facility. *Management Sci.* 23,1070-1079.

B. DZIELINSKI, R. GOMORY (1965) Optimal programming of lot sizes, inventories and labor allocations. *Management Sci.* 11,874-890.

S. ELMAGHRABY (1978) The economic lot scheduling problem (ELSP): review and extensions. *Management Sci.* 24,587-598.

M. FLORIAN, M. KLEIN (1971) Deterministic production planning with concave costs and capacity constraints. *Management Sci.* 18,12-20.

M. FLORIAN, J.K. LENSTRA, A.H.G. RINNOOY KAN (1980) Deterministic production planning: algorithms and complexity. *Management Sci.* 26,669-679.

A.M. GEOFFRION, G.W. GRAVES (1976) Scheduling parallel production lines with changeover costs: practical application of a quadratic assignment/LP approach. *Oper. Res.* 24,595-610.

S. GORENSTEIN (1970) Planning tire production. *Management Sci.* 17,72-82.

M.R. LAMBRECHT, J. VANDER EECKEN (1978) A facilities in series capacity constrained dynamic lot-size model. *European J. Oper. Res.* 2,42-49.

M.R. LAMBRECHT, J. VANDER EECKEN (1978) Capacity constrained single-facility dynamic lot-size model. *European J. Oper. Res.* 2,1-8.

M.R. LAMBRECHT, H. VANDERVEKEN (1979) Heuristic procedures for the single operation, multi-item loading problem. *AIIE Trans.* 11,319-326.

M.R. LAMBRECHT, H. VANDERVEKEN (1979) Production scheduling and sequencing for multi-stage production systems. *OR Spektrum* 1,103-114.

L.S. LASDON, R.C. TERJUNG (1971) An efficient algorithm for multi-item scheduling. *Oper. Res.* 19.

S.F. LOVE (1972) A facilities in series inventory model with nested schedules. *Management Sci.* 18,327-338.

A.S. MANNE (1958) Programming of economic lot sizes. *Management Sci.* 4,115-135.

T. MORTON, J. MENSCHING (1978) A finite production rate dynamic lot size model. Working Paper 59-77-78, GSIA, Carnegie-Mellon University, Pittsburgh, PA.

L.G. SCHWARZ, L. SCHRAGE (1975) Optimal and system myopic policies for multi-echelon production/inventory assembly systems. *Management Sci.* 21.

S. SETHI, S. CHAND (1980) Multiple finite production rate dynamic lot size inventory models. Technical Report, Purdue University.

E.A. SILVER (1976) A simple method of determining order quantities in joint replenishments under deterministic demand. *Management Sci.* 22,1351-1361.

E.A. SILVER, H. MEAL (1973) A heuristic for selecting lot-size quantities for the case of a deterministic time varying demand rate and discrete opportunities for replenishment. *Production and Inventory Control Management*, 2nd Quarter, 64-74.

J.A.E.E. VAN NUNEN, J. WESSELS (1978) Multi-item lot size determination and scheduling under capacity constraints. *European J. Oper. Res.* 2,36-41.

H.M. WAGNER, T.M. WHITIN (1959) Dynamic version of the economic lot size model. *Management Sci.* 5,86-96.

W. ZANGWILL (1968) Minimum concave cost flows in certain networks. *Management Sci.* 14,429-450.

QUEUEING NETWORKS

K.C. SEVCIK
University of Toronto

Not received

APPLICATIONS OF QUEUEING NETWORKS TO COMPUTER SYSTEMS

K.C. SEVCIK
University of Toronto

Not received

ON SCHEDULING WITH RELEASE TIMES AND DEADLINES

B. SIMONS
IBM, San José

We shall discuss problems in which there are n jobs and 1 or more identical parallel machines. Each job becomes available for running at its release time, must be completed by its deadline, and cannot be interrupted once it has started to run. Can we determine in polynomial time whether or not a set of jobs has a feasible schedule if:

1. each job has unit processing time and the release times and deadlines are integers;
2. each job has arbitrary processing time and the release times and deadlines are integers;
3. each job has unit processing time and the release times and deadlines are real numbers;
4. each job has unit processing time, the release times and deadlines are real numbers, and there may be two or more release time/deadline intervals for each job?

The answers to the first two questions had been known for some time. We shall survey these results, discuss a simple algorithm which resolves the third question, and show why the last question probably has no fast algorithm.

We shall also discuss how with an additional cost of only O(log n) time we can minimize the maximum tardiness for instances of problem 3 for which there is no feasible schedule.

MARKOV DECISION PROCESSES

G. WEISS
Tel-Aviv University

This introductory lecture is intended to outline some of the basic ideas of discrete time Markov decision processes, with particular stress on the finite decision space case, which avoids measure theoretic problems - the coverage follows in part the presentation in [Ross 1970]. The list of references includes some of the more important papers in the development of the subject; two other texts are [Derman 1970] and [Bertsekas 1976].

The topics covered are: The optimality principle of dynamic programming - state space and decision space. The elements of Markov decision processes - state space, decision space, transition law and immediate return function. Policies - general, Markov and stationary. Classification of problems - discounted, positive, negative and average return per period. Optimal return, the optimality equation and optimal policies. Contraction mappings and the discounted case. Solution via successive approximation, policy iteration, or mathematical programming. The negative and positive problems. The average return per period problem. Comments on general state and action spaces. Extension to semi Markov decision processes.

References

R. BELLMAN (1957) *Dynamic Programming*, Princeton University Press, Princeton, NJ.

D.P. BERTSEKAS (1976) *Dynamic Programming and Stochastic Control*, Academic Press, New York.

D. BLACKWELL (1962) Discrete dynamic programming. *Ann. Math. Statist.* 33,719-726.

D. BLACKWELL (1965) Discounted dynamic programming. *Ann. Math. Statist.* 36, 226-235.

D. BLACKWELL (1965) Positive dynamic programming. *Proc. 5th Berkeley Symp. Mathematical Statistics and Probability*, 415-418.

E.V. DENARDO (1967) Contraction mappings in the theory underlying dynamic programming. *SIAM Rev.* 9,165-177.

C. DERMAN (1970) *Finite State Markovian Decision Processes*, Academic Press, New York.

R. HOWARD (1960) *Dynamic Programming and Markov Processes*, MIT Press, Cambridge, MA.

A. HORDIJK (1974) *Dynamic Programming and Markov Potential Theory*, Mathematical Centre Tracts 51, Mathematisch Centrum, Amsterdam.

S.M. ROSS (1970) *Applied Probability Models with Optimization Applications*, Holden Day, San Francisco.

R.E. STRAUCH (1966) Negative dynamic programming. *Ann. Math. Statist.* 37, 871-890.

SCHEDULING STOCHASTIC JOBS ON SEVERAL MACHINES

G. WEISS
Tel-Aviv University

Unlike the wide range of solved problems in deterministic scheduling and the results on scheduling general stochastic jobs on a single machine (see previous lectures), only little is known about scheduling stochastic jobs on several machines.

Most of the results are for jobs whose durations are exponentially distributed - for such jobs there is usually no distinction between preemptive and nonpreemptive schedules, and between continuous and discrete time schedules. In particular, if only two machines are involved, some problems can be solved by fairly direct methods. Bruno and Downey [2] use an exchange argument to show that SEPT (shortest expected processing time) and LEPT (longest expected processing time) rules respectively minimize expected values of $\sum C_i$ and of C_{max} for two parallel machines. Pinedo and Weiss [14] prove the latter by examining the last remaining job, and Pinedo [12] uses similar arguments to partially characterize optimal schedules for maximizing expected value of C_{max}. Minimization and maximization of C_{max} has a reliability interpretation. For the two machine flowshop with exponential job durations on both machines, Cunningham and Dutta [6] prove by an exchange argument that a rule equivalent to Johnson's [11] rule for deterministic jobs minimizes expected value of C_{max}. Some very simple cases of two machine open shop can also be treated [8]. The calculation of the expected values of $\sum C_i$ and C_{max} for a given schedule is in itself a nontrivial problem, and various efficient algorithms exist.

For m machines and n exponential jobs, scheduling on parallel machines was investigated with increasing generality by several authors [1;2;7;9;14; 16;20;21]. Weiss and Pinedo [21] consider m uniform machines in parallel, with preemptions, and show that SEPT and LEPT minimize expected $\sum C_i$ and expected C_{max} respectively, as well as optimizing various other expected cost criteria. The proof is by using Markov decision processes in continuous time. When machines are identical the schedules are nonpreemptive.

Optimality of LEPT and SEPT preemptive scheduling rules for m parallel

identical machines is investigated by Weber [17;18;19], for jobs whose durations belong to MHR (monotone hazard rate) or MHR* families, defined as follows: there is a basic job duration distribution with a monotone hazard rate (a PF2 type probability density for MHR*), and the various jobs start at different ages along that distribution. Using an elegant induction argument in discrete time, both on the time and on the jobs, he shows the optimality of SEPT for minimizing $\sum C_i$, and of LEPT for minimizing C_{max} and for maximizing the time to the first machine idleness (the Nylon Stocking problem of Cox [4], also life time of a series system with m components and n spares in reliability). For some of these problems, the optimality is in the strong sense of stochastic majorization, and job arrivals (equivalently release dates) as well as varying numbers of machines are allowed.

References

1. J. BRUNO (1976) Sequencing tasks with exponential service times on parallel machines. Technical Report, Department of Computer Science, Pennsylvania State University.
2. J. BRUNO, P. DOWNEY (1977) Sequencing tasks with exponential service times on two machines. Technical Report, Department of Electrical Engineering and Computer Science, University of California, Santa Barbara.
3. V.Ya. BURDYUK (1969) The stochastic problem of two machines. *Cybernetics* 5,651-661 (translated from Russian).
4. D.R. COX (1959) A renewal problem with bulk ordering of components. *J. Roy. Statist. Soc. Ser. B* 21,180-189.
5. T.B. CRABILL, D. GROSS, M.J. MAGAZINE (1977) A classified bibliography of research on optimal design and control of queues. *Oper. Res.* 25,219-232.
6. A.A. CUNNINGHAM, S.K. DUTTA (1973) Scheduling jobs with exponentially distributed processing times on two machines of a flow shop. *Naval Res. Logist. Quart.* 16,69-81.
7. G.N. FREDERICKSON (1978) Sequencing tasks with exponential service times to minimize the expected flow time or makespan. Technical Report, Department of Computer Science, Pennsylvania State University.
8. E. FROSTIG (1980) Personal communication.
9. K.D. GLAZEBROOK (1979) Scheduling tasks with exponential service times on parallel processors. *J. Appl. Probab.* 16,685-689.

10. K.D. GLAZEBROOK, P. NASH (1976) On multi server stochastic scheduling. *J. Roy. Statist. Soc. Ser. B* 38,67-72.

11. S.M. JOHNSON (1954) Optimal two- and three-stage production schedules with setup times included. *Naval Res. Logist. Quart.* 1,61-68.

12. M. PINEDO (1980) Scheduling spares with exponential lifetimes in a two component parallel system. *J. Appl. Probab.* 17,1025-1032.

13. M. PINEDO (1980) On stochastic flowshops. Technical Report, IVIC, Caracas, Venezuela.

14. M. PINEDO, G. WEISS (1979) Scheduling of stochastic tasks on two parallel processors. *Naval Res. Logist. Quart.* 26,527-535.

15. M.H. ROTHKOPF (1966) Scheduling with random service times. *Management Sci.* 12,707-713.

16. L. VAN DER HEYDEN (1979) A note on scheduling jobs with exponential processing times on identical processors so as to minimize makespan. *Math. Oper. Res.*, to appear.

17. R.R. WEBER (1978) On the optimal assignment of customers to parallel servers. *J. Appl. Probab.* 15,406-413.

18. R.R. WEBER (1979) *Optimal Organisation of Multiserver Systems*, Ph.D. Thesis, University of Cambridge.

19. R.R. WEBER, P. NASH (1979) An optimal strategy in multiserver stochastic scheduling. *J. Roy. Statist. Soc. Ser. B* 40,322-327.

20. G. WEISS (1977) A 2 machine n job scheduling problem. Technical Report, Department of Statistics, Tel-Aviv University.

21. G. WEISS, M. PINEDO (1980) Scheduling tasks with exponential service times on non-identical processors to minimize various cost functions. *J. Appl. Probab.* 17,187-202.

ADVANCED RESEARCH INSTITUTE

P. WHITTLE — Thu 16 09.00-10.00
Sequential project selection (multi-armed bandits) and the Gittins index

P. NASH, R.R. WEBER — Thu 16 10.00-11.00
Sequential open-loop scheduling strategies

M. PINEDO — Thu 16 11.30-12.30
On the complexity of stochastic scheduling problems

P. NASH, R.R. WEBER — Thu 16 15.30-16.30
Stochastic dominance in allocation and scheduling problems

K.D. GLAZEBROOK — Thu 16 17.00-18.00
**On the evaluation of non-preemptive strategies in stochastic scheduling*

G.N. FREDERICKSON — Fri 17 09.00-10.00
Probabilistic analysis of the LPT processor scheduling heuristic

J. BRUNO — Fri 17 10.00-11.00
Deterministic and stochastic scheduling problems with treelike precedence constraints

I. MITRANI — Fri 17 11.30-12.30
On the delay functions achievable by non-preemptive scheduling strategies in M/G/1 queues

U. HERZOG — Fri 17 15.30-16.30
Modelling for multiprocessor projects

A. FEDERGRUEN, P. ZIPKIN — Fri 17 17.00-18.00
**A combined vehicle routing and inventory allocation problem*

Invited lecture
**Contributed lecture*

DETERMINISTIC AND STOCHASTIC SCHEDULING PROBLEMS WITH TREELIKE PRECEDENCE CONSTRAINTS

J. BRUNO
University of California, Santa Barbara

In this talk we survey the known results for sequencing unit execution time tasks on parallel machines subject to treelike precedence constraints. We shall discuss the intree and outtree versions of this problem and for each of these contrast the known results for deterministic versus stochastic processing times.

References

T.C. HU (1961) Parallel sequencing and assembly line problems. *Oper. Res.* 9, 841-848.

A.B. BARAK, P.J. DOWNEY (1978) Scheduling probabilistic two-processor systems using reincarnation policies. Technical Report CS-78-01, Computer Science Department, Pennsylvania State University.

K.M. CHANDY, P.F. REYNOLDS (1975) Scheduling partially ordered tasks with probabilistic execution times. *Proc. 5th Symp. Operating Systems Principles* 161-177.

P. BRUCKER, M.R. GAREY, D.S. JOHNSON (1977) Scheduling equal-length tasks under treelike precedence constraints to minimize maximum lateness. *Math. Oper. Res.* 2,275-284.

G.I. DAVIDA, D.J. LINTON (1976) A new algorithm for the scheduling of tree structured task. *Proc. 1976 Conf. Information Sciences and Systems*, Johns Hopkins University, 543-548.

J. BRUNO, J.W. JONES, K. SO (1980) Deterministic scheduling with pipelined processors. *IEEE Trans. Comput.* C-29,308-316.

A COMBINED VEHICLE ROUTING AND INVENTORY ALLOCATION PROBLEM

A. FEDERGRUEN, P. ZIPKIN
Columbia University, New York

We consider the combined problem of allocating inventories of a scarce resource available at some central depot among a given set of delivery points while grouping these locations in minimal cost vehicle routes. A stochastic demand pattern is assumed at each of the delivery points: In each point the inventory carrying and shortage costs depend upon the end of period inventory levels. Two solution approaches are discussed, an estension of the r-opt method used for deterministic vehicle routing problems, and a generalized Benders decomposition algorithm which achieves the exact solution when pursued till convergence.

The final part of the talk briefly discusses extensions of the basic model incorporating additional complications such as: (1) multiple commodities, (2) multiple age classes for perishable products, (3) dynamic allocation procedures.

PROBABILISTIC ANALYSIS OF THE LPT PROCESSOR SCHEDULING HEURISTIC

G.N. FREDERICKSON
Pennsylvania State University

We consider the following processor scheduling problem: Assign n tasks with known execution times to m identical processors, with no preemptions, so as to minimize the finish time. The problem has been shown to be NP-hard [Karp 1972], indicating that there is probably no polynomial time algorithm to solve it. As a consequence, polynomial time approximation algorithms have been developed that guarantee a constant worst-case bound on the ratio of the cost of a heuristic solution to the cost of an optimal solution. For example, the LPT heuristic (largest processing time first) has been shown [Graham 1969] to have a bound of $4/3 - 1/(3m)$. The best algorithm to date is MULTIFIT, with a bound of 6/5 [Coffman *et al.* 1978; Friesen 1981]. It has been observed by several authors that the worst-case bounds for these heuristics are not indicative of average performance, which simulation results suggest is considerably better. Unfortunately, relatively little probabilistic analysis has been applied to heuristics for any of the NP-hard problems [Karp 1977; Lueker 1978].

In this paper, we analyze the average performance of the LPT heuristic, under the assumption that task times are drawn from a uniform distribution on (0,1]. (This distribution is chosen for the sake of tractability, but we note that simulations of various heuristics have been made using this distribution [Coffman *et al.* 1978].) We bound the ratio of the expected finish time for the heuristic to the expected finish time of an optimal preemptive schedule. (This type of ratio was employed in [Frederickson 1980] to analyze the average performance of two simple bin packing heuristics.) We show that this ratio is $1 + O((m-1)^2/n^2)$ for the LPT heuristic, confirming analytically that the heuristic does do well on average. For the case $m = 2$, we also demonstrate that this ratio is $1 + \Omega(1/n^2)$.

References

E.G. COFFMAN, JR., M.R. GAREY, D.S. JOHNSON (1978) An application of bin packing to multiprocessor scheduling. *SIAM J. Comput.* 7,1-17.

G.N. FREDERICKSON (1980) Probabilistic analysis for simple one- and two-dimensional bin packing algorithms. *Inform. Process. Lett.* 11,156-161.

D.K. FRIESEN (1980) Tighter bounds for the MULTIFIT processor scheduling algorithm. *SIAM J. Comput.*, to appear.

R.L. GRAHAM (1969) Bounds on multiprocessing timing anomalies. *SIAM J. Appl. Math.* 17,416-429.

R.M. KARP (1972) Reducibility among combinatorial problems. In: R.E. MILLER, J.W. THATCHER (eds.) (1972) *Complexity of Computer Computations*, Plenum Press, New York, 85-103.

R.M. KARP (1977) Probabilistic analysis of partitioning algorithms for the traveling salesman problem in the plane. *Math. Oper. Res.* 2,209-224.

G.S. LUEKER (1978) Maximization problems on graphs with edge weights chosen from a normal distribution. *Proc. 10th Annual ACM Symp. Theory of Computing*, 13-18.

ON THE EVALUATION OF NON-PREEMPTIVE STRATEGIES IN STOCHASTIC SCHEDULING

K.D. GLAZEBROOK
University of Newcastle upon Tyne

A collection of stochastic jobs is to be processed by a single machine. The jobs must be processed in a manner which is consistent with a precedence relation but the machine is free to switch from one job to another at any time; such switches are costly, however.

A general model is proposed for the above problem. Sufficient conditions are given which ensure that there is an optimal strategy given by a fixed permutation of the job set. These conditions are then used as a starting point for the important task of evaluating permutations as strategies in more general circumstances where no permutation is optimal.

MODELLING FOR MULTIPROCESSOR PROJECTS

U. HERZOG
Universität Erlangen-Nürnberg

Important performance problems for multiprocessor computer systems have been discussed, modelled and investigated since many years. These fundamental results are - although derived without experience with real systems - still of great importance.

Nowadays, however, there are several multiprocessor-projects operational. Experiences with such experimental systems give us a deeper insight and many impulses for performance modelling.

This contribution discusses two multiprocessor projects at the University of Erlangen-Nuremberg and related performance problems.

EGPA, the Erlangen General Purpose Array

Goal of the EGPA-project is to design powerful general purpose computers by means of hierarchically structured, modulary extendable multiprocessor systems. A pilot-implementation with five AEG-Telefunken control-computers is in operation. Software and hardware monitors allow to investigate the internal flow of information and to detect bottlenecks in hardware, system software and application programs, as well.

Rather than running independent tasks on different processors one also tries to take advantage of the parallelism inherent in many problems, *i.e.* application programs are decomposed into sets of cooperating subtasks and processed in parallel, when possible. So we many increase not only the throughput of a system: run-times (and therefore response-times) for individual application programs may be reduced significantly, too. Then, however, difficult coordination problems (synchronization between tasks, data- and load-sharing, *etc.*) may occur and have to be considered in modelling such systems. Measurements also show that system overhead due to interprocessor communication can be significant and has to be taken into account. Realistic modelling is possible by introducing a new class of queueing systems.

DIRMU, a DIstributed Reconfigurable MUltiprocessor kit

Goal of the DIRMU-project is to implement and test a module-computer kit for dedicated and user-configurable multimicrocomputer systems. Prototypes for basic elements (general purpose processors, memories, *etc.*) have been built and can be used to construct efficient multiprocessor networks for given sets of user problems.

When implementing such specialized computer networks a major problem is to allocate subtasks and data in order to guarantee performance requirements, fault tolerance and reasonable cost, as well. These questions and related modelling activities will be discussed in general and by example.

References

U. HERZOG, W. HOFFMANN, W. KLEINÖDER (1979) Performance modelling and evaluation for hierarchically organized multiprocessor computer systems. *Proc. 1979 Int. Conf. Parallel Processing*, IEEE, ACM & Wayne State University, Detroit, 103-114.

W. HÄNDLER, H. ROHRER (1980) Thoughts on a computer construction kit. *Elektron. Rechenanlagen* 1,3-13.

W.W. CHU *et al.* (1980) Task allocation in distributed data processing. *Comput.* 57-68.

ON THE DELAY FUNCTIONS ACHIEVABLE BY NON-PREEMPTIVE SCHEDULING STRATEGIES IN M/G/1 QUEUES

I. MITRANI
University of Newcastle upon Tyne

For a queueing system in equilibrium, the delay function, $W(x)$, is defined as the expected time spent in the queue by a job whose required service is x ($x \geq 0$). In an M/G/1 queue with a given arrival rate, λ, and distribution of required service times, $F(x)$, the delay function depends on the job scheduling strategy employed. A function $W(x)$ is said to be achievable in that queue if there exists a scheduling strategy such that the corresponding delay function is $W(x)$.

This note addresses the problem of characterising the set of delay functions that are achievable by non-preemptive scheduling strategies. That is, scheduling decisions are made at service completion instants only; the selection of the next job to be served may be influenced by the required service times of the jobs in the queue and by the past queue behaviour (during the current busy period), but not by future arrivals. The idea is to generalise some existing results which are valid when $F(x)$ is a step function with a finite number of jumps (*i.e.* when the set of required service times is finite). That generalisation leads to an integral equality constraint and a set of integral inequalities which the achievable functions $W(x)$ must satisfy. In addition, it is shown that the set of those functions is convex, and its extreme elements are given. This allows the construction of scheduling strategies whose delay functions approximate a pre-defined, achievable delay function $W(x)$ to any accuracy.

The presentation is informal, relying on intuitive arguments rather than rigorous proofs. The results should therefore be treated as conjectures at this stage.

Reference

E. GELENBE, I. MITRANI (1980) *Analysis and Synthesis of Computer Systems*, Academic Press, London.

SEQUENTIAL OPEN-LOOP SCHEDULING STRATEGIES

P. NASH, R.R. WEBER
University of Cambridge

A large number of stochastic scheduling problems can be reduced to problems in deterministic optimal control by a dynamic programming formulation over suitable spaces of functions. The basis of the method is to seek an allocation of processor effort for every future time, to be followed only until some event (usually an arrival or completion) occurs. This allocation is optimized on the assumption that an optimal schedule will be followed after the first event. Control theorists call the resulting strategy a closed-loop controller. This approach has produced a number of theoretical results, and can in principle be used as the basis of a computational method. In practice, the derived deterministic control problems are complicated, and the computational requirements for a realistic application prohibitively great.

A modification of this approach is possible, which leads to much simpler control problems. This is to seek sequential open-loop scheduling strategies, rather than fully closed-loop ones. That is, we still look for an allocation of processor effort for every future time, but now to be followed only until some fixed or possibly random review time, and optimize this allocation without reference to what happens after the review. This calculation is carried out sequentially, at each review time. The resulting schedules are sub-optimal, but in many cases approximate optimal schedules quite well. In a number of cases, the sequential open-loop strategy is actually optimal, if the review period is small enough.

In this paper, we discuss some models for which the closed-loop formulation leads to as-yet unsolved control problems, but where the sequential open-loop formulation has analytic solutions. Among these are a number of different single and parallel processor models, including models where jobs leave before completion and models where the cost of waiting is itself a discrete-state random process. As well as analytic results, some computational results are presented which compare the performance of open- and closed-loop strategies and illustrate the behaviour of the open-loop sequential strategy as the review period changes.

References

The use of dynamic programming to reduce the stochastic scheduling problem to a deterministic optimal control problem is described in

P. NASH, J.C. GITTINS (1977) A Hamiltonian approach to stochastic resource allocation. *Adv. in Appl. Probab.* 9,55-68.

P. NASH (1979) Controlled jump process models for stochastic scheduling. *Internat. J. Control* 29,1011-1025.

P. NASH (1974) *Optimal Allocation of Resources between Research Projects*, Ph.D. Thesis, University of Cambridge.

R.R. WEBER (1982) Scheduling jobs with unknown processing requirements on parallel machines to minimize makespan and flowtime. *J. Appl. Probab.*, to appear.

The basic optimal control theory underlying these papers can be found in

E.B. LEE, L. MARKUS (1968) *Foundations of Optimal Control Theory*, Wiley, New York.

Approaches related to the open-loop formulation for the stochastic scheduling problem appear in

M. ALAM (1979) An application of modern control theory to a time-dependent queueing system for optimal operation. *Internat. J. Systems Sci.* 10, 693-700.

C.H. SCOTT, T.R. JEFFERSON (1976) Optimal regulation of service rate for a queue with finite waiting room. *J. Optim. Theory Appl.* 20,245-250.

The theory of maximum principles for discrete state, continuous-time Markov processes is discussed in, for example,

R. RISHEL (1974) Lecture notes in economic and mathematical systems 107, Springer, Berlin.

D.D. SWORDER (1969) *IEEE Trans. Automat. Control* AC14.9.

The following paper considers the reduction to discrete semi-Markov decision problems of controlled jump process models in general.

A.A. YUSHKEVITCH (1981) On reducing a jump-controllable Markov model to a model with discrete time. *SIAM J. Probab. Theory Appl.* 25,1.

STOCHASTIC DOMINANCE IN ALLOCATION AND SCHEDULING PROBLEMS

P. NASH, R.R. WEBER
University of Cambridge

The aim in many stochastic allocation and scheduling problems is to minimize in expectation or distribution the time required to meet a certain objective. An example is the problem of minimizing makespan for a number of jobs on parallel processors. Although for many problems there exists a strategy minimizing in expectation the time to meet the objective, it is only for rather special problems that there exists a stochastically dominant strategy minimizing the time in distribution. When a stochastically dominant strategy does exist, it is usually easy to construct, since it can be found by just maximizing the one-step probability of meeting the objective.

This paper discusses the problem of determining when a problem has a stochastically dominant strategy. We consider this question first in a quite general class of Markov decision problems, and illustrate the results by applying them to a number of stochastic scheduling and allocation problems. The main result is a way of deriving sufficient conditions for the one-step look-ahead strategy to be dominant, in terms of simple conditions on the transition matrix P^* associated with this strategy, of the form

$$QP^*Q^{-1} \geq 0,$$

for certain matrices Q. This result essentially tells us that stochastic dominance of the one-step look-ahead strategy is guaranteed by certain sorts of easily checkable dominance among the rows of the associated transition matrix. We show how a number of already-known and some new results for scheduling, customer-assignment and search problems can be easily derived by this route.

References

A paper which argues for the use of stochastic dominance as a criterion for judging scheduling strategies is

J.K. WEEKS, T.R. WINGLER (1979) A stochastic dominance ordering of scheduling rules. *Decision Sci.* 10,245-257.

The following papers deal with problems for which stochastically dominant optimal strategies can be derived.

C. DERMAN, G.U. LIEBERMAN, S.M. ROSS (1980) On the optimal assignment of servers and a repairman. *J. Appl. Probab.* 17,577-581.

D.R. SMITH (1968) Optimal repair of the series system. *Oper. Res.* 4,653-662.

D.R. SMITH (1978) A new proof of the optimality of the shortest remaining processing time discipline. *Oper. Res.* 26,197-199.

R.R. WEBER (1978) On the optimal assignment of customers to parallel servers. *J. Appl. Probab.* 15,406-413.

W. WINSTON (1977) Optimality of the shortest line discipline. *J. Appl. Probab.* 14,181-189.

Some ideas related to those in this paper are discussed in

R. SERFOZO (1981) Optimal control of random walks, birth and death processes and queues. *Adv. in Appl. Probab.* 13,1.

ON THE COMPLEXITY OF STOCHASTIC SCHEDULING PROBLEMS

M. PINEDO
Georgia Institute of Technology, Atlanta

For any deterministic scheduling problem, one can formulate a stochastic counterpart. Stochastic models with exponentially distributed processing times, due dates and release dates usually have a rather "nice" structure. It will be mainly with these models with which we will deal. For determining the optimal policy, *i.e.*, the policy that minimizes either the expected makespan or the expected flow time, in any class of policies, one might have to develop algorithms which will have a certain complexity. In some cases these optimal policies can be determined in polynomial time, in other cases it is not clear whether they can be determined in polynomial time or not. And in some very special cases we will be able to prove that we have no polynomial time algorithm. Often, when a deterministic scheduling problem can be solved in polynomial time, it turns out that the same model with processing times and other relevant data exponentially distributed also can be analyzed in polynomial time. Examples of models where this is the case are $F2||E(C_{max})$, $J2||E(C_{max})$. This, however, is not always the case. An exception for example is $O2||E(C_{max})$, where the exponential version is much harder that the deterministic version.

However, on the other hand, when a deterministic problem is NP-complete it does *not* imply that determining the optimal policy (in any particular class of policies) for the same model with exponentially distributed data is a hard problem, too. We discuss a number of models where the deterministic version is NP-complete and where for its counterpart with all data exponentially distributed, we have "nice" optimal policies in several classes of policies. Examples of these models are:

(1) $P||C_{max}$ [Weber -; Weiss & Pinedo 1980];
(2) $1|d_j=d|\sum w_j U_j$ [Derman *et al.* 1978; Pinedo -];
(3) $1|d_j=d|\sum w_j T_j$ [Pinedo -].

Of the last problem (3) it has not been determined yet whether the deterministic version is NP-complete or not. However, the structure of the deterministic version of this problem is certainly not as "nice" as the

structure of the exponential version. These three examples indicate that problems with exponential distributions often have nicer structures than problems with deterministic distributions. So, in these cases, having less information with regard to the processing times makes it easier to determine the optimal policy.

In some cases it turns out that the optimal policy for the deterministic model has the same structure as the optimal policy for the same model with exponential distributions, which makes us conclude that optimal policies for these stochastic problems cannot be determined in polynomial time, when the deterministic version is NP-complete. An example is the deterministic problem $1|prec,p_j=1|\sum w_jC_j$, and its stochastic counterpart $1|prec,p_j\sim exp(1)|E(\sum w_jC_j)$ [Lawler 1978].

Determining the complexity of an algorithm for finding the optimal policy turns out to be less hard for one particular class of policies. Consider the class of policies where the decision-maker is required to determine all his actions for the whole duration of the process, in advance, at time t = 0 and may not deviate from this predetermined course when more information becomes available during the process. We will call this class of policies the class of static policies, in contrast to the class of dynamic policies, where the decision-maker is allowed to make his decisions sequentially. The problem of determining the optimal policy in the class of static policies, when distributions are exponential, usually has a very special structure. Such a problem can be compared easily with other deterministic problems. It is now possible to reduce well-known deterministic NP-complete problems to the problem of determining the optimal policy in the class of static policies for models with exponential distributions. For example, it is shown that LINEAR ARRANGEMENT is reducible to $1|prec,p_j\sim exp(1),d_j\sim exp(\mu_j)|E(\sum U_j)$. Several other examples are presented, too.

Only for very special cases we are able to reduce the problem of determining the optimal policy in the class of static policies to the problem of determining the optimal policy in classes of preemptive and non-preemptive dynamic policies.

References

C. DERMAN, G. LIEBERMAN, S. ROSS (1978) A renewal decision problem. *Management Sci.* 24,554-561.

E.L. LAWLER (1978) Sequencing jobs to minimize total weighted completion time subject to precedence constraints. *Ann. Discrete Math.* 2,75-90.

M. PINEDO (-) Scheduling exponential tasks with random due dates and their relationship with flow time. *European J. Oper. Res.*, in editorial process.

R.R. WEBER (-) Scheduling jobs with stochastic processing requirements on parallel machines to minimize makespan or flow time. *J. Appl. Probab.*, to appear.

G. WEISS, M. PINEDO (1980) Scheduling tasks with exponential service times on non-identical processors to minimize various cost functions. *J. Appl. Probab.* 17,187-202.

SEQUENTIAL PROJECT SELECTION (MULTI-ARMED BANDITS) AND THE GITTINS INDEX

P. WHITTLE
University of Cambridge

A direct proof is given of the optimality of the Gittins index policy, and a related identity demonstrated for the loss function. Especial attention is paid to the case when new projects also arrive in a statistically homogeneous stream. A number of general results are obtained of which those derived by J.M. Harrison *etc.* are shown to be a special case.

References

J.C. GITTINS (1975) The two-armed bandit problem: variations on a conjecture by H. Chernoff. *Sankhyā Ser. A* 37,287-291.

J.C. GITTINS (1979) Bandit processes and dynamic allocation indices. *J. Roy. Statist. Soc. Ser. B* 41,148-164.

J.C. GITTINS, K.D. GLAZEBROOK (1977) On Bayesian models in stochastic scheduling. *J. Appl. Probab.* 14,556-565.

J.C. GITTINS, D.M. JONES (1974) A dynamic allocation index for the sequential design of experiments. In: J. GANI (ed.) (1974) *Progress in Statistics*, North-Holland, Amsterdam, 241-266.

J.C. GITTINS, P. NASH (1977) Scheduling, queues, and dynamic allocation indices. *Proc. EMS, Prague 1974*, Czechoslovak Academy of Sciences, Prague, 191-202.

K.D. GLAZEBROOK (1976A) A profitability index for alternative research projects. *Omega* 4,79-83.

K.D. GLAZEBROOK (1976B) Stochastic scheduling with order constraints. *Internat. J. Systems Sci.* 7,657-666.

K.D. GLAZEBROOK (1978A) On a class of non-Markov decision processes. *J. Appl. Probab.* 15,689-698.

K.D. GLAZEBROOK (1978B) On the optimal allocation of two or more treatments in a controlled clinical trial. *Biometrika* 65,335-340.

J.M. HARRISON (1975A) A priority queue with discounted linear costs. *Oper. Res.* 23,260-269.

J.M. HARRISON (1975B) Dynamic scheduling of a multi-class queue: discount optimality. *Oper. Res.* 23,270-282.

J.M. HARRISON (1976) Dynamic scheduling of a two-class queue: small interest rates. *SIAM J. Appl. Math.* 31,51-61.

P. NASH (1973) *Optimal Allocation of Resources between Research Projects*, Ph.D. Thesis, University of Cambridge.

P. NASH, J.C. GITTINS (1977) A Hamiltonian approach to optimal stochastic resource allocation. *Adv. in Appl. Probab.* 9,55-68.

P. WHITTLE (1980) Multi-armed bandits and the Gittins index. *J. Roy. Statist. Soc. Ser. B* 42,143-149.

P. WHITTLE (1981) Arm-acquiring bandits. *Ann. Probab.* 9.

ADDRESSES

J. BRUNO
Department of Computer Science
University of California
Santa Barbara, CA 93106
U.S.A.

E.G. COFFMAN, JR.
Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974
U.S.A.

M.A.H. DEMPSTER
IIASA
2361 Laxenburg
Austria

A. FEDERGRUEN
Graduate School of Business
Uris Hall 419
Columbia University
New York, NY 10027
U.S.A.

M.L. FISHER
Decision Sciences Department
The Wharton School
University of Pennsylvania
Philadelphia, PA 19104
U.S.A.

G.N. FREDERICKSON
Computer Science Department
Pennsylvania State University
University Park, PA 16802
U.S.A.

W. GAUL
Institut für Entscheidungstheorie
und Unternehmensforschung
Universität Karlsruhe (TH)
Kollegium am Schloss, Bau III
Postfach 6380
7500 Karlsruhe 1
F.R.G.

E. GELENBE
Laboratoire de Recherche en Informatique
Université de Paris-Sud
Bâtiment 490
Centre d'Orsay
91405 Orsay Cédex
France

J.C. GITTINS
Mathematical Institute
University of Oxford
24-29 St. Giles
Oxford OX1 3LB
U.K.

K.D. GLAZEBROOK
Department of Statistics
University of Newcastle upon Tyne
Newcastle upon Tyne NE1 7RU
U.K.

U. HERZOG
Institut für Mathematische Maschinen
und Datenverarbeitung VII
Universität Erlangen-Nürnberg
Martensstrasse 3
8520 Erlangen
F.R.G.

E.L. LAWLER
Computer Science Division
591 Evans Hall
University of California
Berkeley, CA 94720
U.S.A.

J.K. LENSTRA
Mathematisch Centrum
P.O. Box 4079
1009 AB Amsterdam
The Netherlands

C.U. MARTEL
Department of Electrical and
Computer Engineering
University of California
Davis, CA 95616
U.S.A.

I. MITRANI
Computing Laboratory
University of Newcastle upon Tyne
Claremont Tower
Claremont Road
Newcastle upon Tyne NE1 7RU
U.K.

P. NASH
Engineering Department
University of Cambridge
Mill Lane
Cambridge CB2 1RX
U.K.

M.L. PINEDO
School of Industrial and
Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332
U.S.A.

A.H.G. RINNOOY KAN
Econometric Institute H2-11
Erasmus University
P.O. Box 1738
3000 DR Rotterdam
The Netherlands

S.M. ROSS
Operations Research Center
University of California
Berkeley, CA 94720
U.S.A.

L.E. SCHRAGE
Graduate School of Business
University of Chicago
1101 East 58th Street
Chicago, IL 60637
U.S.A.

K.C. SEVCIK
Computer Systems Research Group
University of Toronto
121 St. Joseph Street
Toronto, Ontario M5S 1A1
Canada

B. SIMONS
IBM
6500 Cottle Road
San José, CA 95123
U.S.A.

R.R. WEBER
Queen's College
Cambridge CB3 9ET
U.K.

G. WEISS
Department of Statistics
Tel-Aviv University
Ramat-Aviv
Tel-Aviv
Israel 69978

P. WHITTLE
Statistical Laboratory
University of Cambridge
16 Mill Lane
Cambridge CB2 1SB
U.K.

ONTVANGEN 15 JULI 1981