THE SCHEDULE-SEQUENCING PROBLEM*

Edward H. Bowman

Massachusetts Institute of Technology, Cambridge, Massachusetts

(Received March 23, 1959)

A tentative solution to the general schedule-sequencing problem is presented in a linear-programming form Feasible solutions hinge on work recently presented on integer solutions to linear-programming problems As yet, the computation involved for a practical problem would be quite large

'THE' scheduling problem in its most simple form consists of a number of jobs to be done on a number of machines, each job having a number of operations to be performed by the various machines *in a specified sequence*, what feasible schedule covers the least total time?

The linear-programming approach presented here is a solution in the sense that the problem is stated in the form of a set of linear inequality (or equality) constraints and a linear-objective function employing the same variables as the constraints The variables are all to take the values zero or one, and therefore a solution hinges on the recent work presented on algorithms for integer solutions to linear programs (*see* especially ref 1, and also refs 2 and 3) Complete enumeration might be included in a list of the various approaches to this problem Work has been done, mostly of a partial enumeration form, on 'practical' approaches to this problem [4] No claim is made that the approach taken here is a practical one Statements exist in the literature to the effect that " here we have an unsolved problem "[5]

A solution to a version of this problem, restricted in several aspects, has been presented earlier [6] No formal restrictions, including the size of the problem to be solved, are inherent in the following method However for even a problem of modest size, the computation required will be quite large, and therefore the phrase is recalled—such solutions are very little solved It is hoped that mechanical and conceptual advances might help overcome this computational road block

* Thanks are due Dr S Vajda of the British Admiralty Research Laboratory, and Prof T P Frenckner and his associates of the Stockholm School of Economics for their suggestions concerning an earlier draft of this paper The referees, including Prof Alan S Manne of Yale University, are also thanked for their suggestions and additional bibliographic references (7–9)

THE PROBLEM

IN ORDER to make the approach taken here most clear, specific product, machine, and time notation will be used Let the products be x, y, and z, the machines be A, B, C, and D, and the time periods (small) run from 1, 2, 3, , T The times required (in time period units) are

	A	B	C	D
x	5	2	8	7
y	4	3	8	5
z	7	0	0	6

The required sequencing for product x is A, B, C, D, for product y is C, A, D, B, and for product z is D, A

STATEMENT OF PROBLEM CONSTRAINTS

THE BASIC variables in the formulation are of the nature $x_{A\,1}$ meaning product x having machine operation A during time period 1 All these variables are to take the values zero or one in the solution—that is, this process either is or is not taking place during this time period The form of the constraints are

$$1 \geqq x_{A\,1},\ x_{A\,2},\quad ,\ x_{A\,T},\ x_{B\,1},\ x_{B\,2},\quad ,\ y_{A\,1},\quad ,\ y_{D\,T},\quad ,\ z_{D\,T} \geqq 0$$

It is necessary to include constraints assuring that the individual operations will be performed For instance product x requires 5 time units of processing on machine A The form of the constraints are

$$\sum_{i=1}^{i=T} x_{A\,i}=5,\quad \sum_{i=1}^{i=T} x_{B\,i}=2,\qquad ,\quad \sum_{i=1}^{i=T} y_{A\,i}=4,\qquad ,\quad \sum_{i=1}^{i=T} z_{D\,i}=6$$

Two or more products may not be processed by the same machine at the same time, that is conflicting assignments are forbidden The form of the constraints are

$$x_{A\,1}+y_{A\,1}+z_{A\,1}\leqq 1,\quad x_{A\,2}+y_{A\,2}+z_{A\,2}\leqq 1,\qquad ,\quad x_{D\,T}+y_{D\,T}+z_{D\,T}\leqq 1$$

Proper sequencing is probably the key part of this problem No operation may be undertaken until the previous operation on the product in the specified sequence has been completed in a previous time period For example, product x requires 5 time units on machine A, before its operation on machine B can be started This operation on machine B (2 time units) in turn must precede the operation on machine C The form of the constraints are

$$5\,x_{B\,j}\leqq \sum_{i=1}^{i=j-1} x_{A\,i},\quad 2\,x_{C\,j}\leqq \sum_{i=1}^{i=j-1} x_{B\,i},\qquad ,\quad 6\,z_{A\,j}\leqq \sum_{i=1}^{i=j-1} z_{D\,i},$$

for all $j=1$ to $j=T$

There is no guarantee in the above formulation that operation runs will not be interrupted, only that sequencing will be correct If operation runs must not be broken (because of set-up costs for instance), then additional constraint sets such as the following can be added

$$5\,x_{A\,i}-5\,x_{A\,i+1}+\sum_{j=i+2}^{j=T} x_{A\,j} \leqq 5,\quad 2\,x_{B\,i}-2\,x_{B\,i+1}+\sum_{j=i+2}^{j=T} x_{B\,j} \leqq 2,$$

$$,\quad 6\,z_{D\,i}-6\,z_{D\,i+1}+\sum_{j=i+2}^{j=T} z_{D\,j} \leqq 6,$$

for all $i=1$ to $i=T$ This does not allow a 'one' variable to be followed by a 'zero' variable, and yet be followed by more 'one' variables Yet feasible scheduling is not excluded For instance the assignment of product x to machine A in the time sequence

1	2	3	4	5	6	7	8	9
0	0	1	1	1	0	0	1	1

is excluded, because

$$5\,x_{A\,5}-5\,x_{A\,6}+\sum (x_{A\,7}+x_{A\,8}+x_{A\,9})=5-0+2=7,$$

which is not $\leqq 5$

OBJECTIVE FUNCTION

To GET a solution to the linear-programming problem, the variables, of the form $x_{A\,1}$, must have values associated with them (many such values may be zero) In a sense, the objective is to have the *final* operations on all products performed as early as possible Prior operations, such as all those on machine C, will of course have preceded the final operations The following objective function is suggested, to be minimized

$$\begin{aligned} \text{O F} =1\,(x_{D\,23}+y_{B\,23}+z_{A\,23})+4\,(x_{D\,24}+y_{B\,24}+z_{A\,24}) \\ +16\,(x_{D\,25}+y_{B\,25}+y_{A\,25})+64\,(x_{D\,26}+y_{B\,26}+z_{A\,26}) \\ + \quad +K_T\,(x_{D\,T}+y_{B\,T}+z_{A\,T}), \end{aligned}$$

where $K_T=4\,K_{T-1}$, etc The rationale of the objective function is that it makes operations (the last ones on each product) toward the end of the time periods costly The number of time periods, chosen in advance of solution, may certainly be equal to or less than the simple sum of all operation times (55), and can be no less than the sum of operation times required on the longest product (22) The cost associated with any operation in a time period is a synthetic one equal to the sum of *all* prior costs plus one This exploding cost function thus forces operations toward the beginning for economic reasons No later time period will be ultimately used than the minimum (optimal), as this one 'cost' is larger than the sum of all prior costs That is, given some feasible solution, the latest (and last) operation would be moved earlier by one time period, and all other operations could be moved later by any number of time periods (excluding movement into or beyond the original last time period) and the exchange would be favorable Actually, it would only be a pathological case that requires such an exploding cost function A problem where different specific costs can be assigned to uncompleted products beyond certain dates would of course not require the generation of *synthetic* costs

THE COMPUTATIONAL PROBLEM

EVEN THE simple problem presented here for illustration has about 300 to 600 real variables depending on the number of time periods chosen [the number equals (products)×(machines)×(time periods)] The number of constraints is substantially larger than this While a dual formulation should ordinarily enable the smaller number (variables<constraints) to be used for computation, GOMORY'S[1] procedure for obtaining integer solutions is of the very nature of adding more constraints as progressive noninteger solutions reveal this to be necessary Other approaches to reducing the computational complexity of a linear-programming solution might also include (1) bounding procedures for choosing the number of time periods, (2) a choice of grosser units for time period length than the sensitivity of measurement available, (3) the elimination of the obvious redundancy in some of the constraints (many of the variables must be zero), and finally, (4) the 'on-off' (one or zero) nature of all the variables raises hopes that a computational procedure exploiting this property may become available

REFERENCES

1 R E GOMORY, "Outline of an Algorithm for Integer Solutions to Linear Programs," *Bull Am Math Soc* **64,** (Sept 1958)
2 G B DANTZIG, "Discrete-Variable Extremum Problems," *Opns Res* **5,** 266–276 (1957)
3 H M MARKOWITZ AND A S MANNE, "On the Solution of Discrete Programming Problems," *Econometrica* **25,** No 1 (1957)
4 W E ANDRUS AND A L BECKER, "A Production Schedule Computed on the IBM 701," *IBM Technical Report,* No 105,007,273, Jan 20, 1954, contained in *Analyses of Industrial Operations,* E H BOWMAN AND R B FETTER (eds), R D Irwin, 1959
5 A VAZSONYI, *Scientific Programming in Business and Industry,* p 416, Wiley, 1958
6 S M JOHNSON, "Optimal Two and Three Stage Production Schedules with Set-up Times Included," *Naval Res Log Quar* **1,** No 1 (March 1954)
7 HARVEY WAGNER, "An Integer Linear Programming Model for Machine Scheduling," Technical Report No 66, Department of Economics, Stanford University, December 24, 1958
8 L G MITTEN, "Sequencing *n* Jobs of Two Machines with Arbitrary Time Lags," *Management Sci* **5,** 293–298 (1959)
9 S M JOHNSON, "Discussion Sequencing *n* Jobs of Two Machines with Arbitrary Time Lags," *Management Sci* **5,** 299–303 (1959)