A Barge Sequencing Heuristic

JAMES S. DYER

Graduate School of Business Administration, University of California, Los Angeles, California

and

FRED GLOVER*

Department of Statistics and Computer Sciences, The University of Texas, Austin, Texas

This paper describes a heuristic method for sequencing barge unloadings at terminals that have regulated unloading rates or flows with the objective of minimizing the variation of a common constituent in the final combined output. The work was undertaken in response to the desire of a steel manufacturer to minimize the maximum sulfur content in coal entering the coking ovens after being mixed by the simultaneous unloading of barges onto a common conveyor. Although the problem may be given an integer programming formulation, the solution method we propose enables one to obtain optimal or near optimal solutions by hand. A hypothetical problem is solved to illustrate the procedure.

The following discussion presents a method for obtaining optimal or near optimal solutions to a class of scheduling problems. The prototype problem that motivated this work involves the sequencing of the unloading of coal barges at pre-assigned piers. However, a class of scheduling problems unrelated to the shipping of coal or the unloading of barges is also amenable to the same basic form of analysis. Although this class of problems can be

* The work of the second author was supported in part by a grant from McKinsey and Co., Inc.

given an integer programming formulation, the solution method we propose enables one to obtain optimal or near optimal solutions by hand (i.e., requiring only a pencil, paper, and perhaps, a desk calculator) in less than twenty minutes for problems whose integer programming formulation involves approximately 153 variables and 70 constraints. In addition, once the required 'tableau' for the method has been determined, problems involving different data sets may be solved even more rapidly.*

The problem for which this technique was initially developed may be described as follows. Coal that is to be subjected to a coking process is shipped by barge from several different suppliers (coal mines) to a common receiving area. Each supplier has his own pier at the receiving area at which barges from his mine(s) alone may be unloaded. The coal is moved by conveyors from each of the piers and dumped into a common receptacle. The composition of the coal in this receptacle is determined by two factors: (1) the composition of the coal in the barges that are being unloaded at a particular time, and (2) the rate at which the coal from each pier is conveyed to and deposited in the receptacle. For example, if the conveyor from one pier has twice the speed (or capacity) of the conveyor from another pier, then the final mix will contain twice as much coal from the first pier as from the second.

The proportions at which the coal from the different piers is mixed are important because the coal is not completely pure, but instead contains a certain amount of sulfur. This sulfur must be neutralized by the addition of chemicals (primarily lime) at the coking ovens. The objective of our scheduling technique is to minimize the cost of this neutralization process.

The parameter of the physical system that can be manipulated to achieve this cost minimization is the sequence in which barges are unloaded at the piers. We seek to determine an unloading schedule for the barges that minimizes the maximum percentage of sulfur occurring in the final mix over the unloading cycle. This 'min-max' objective arises from the fact that sufficient chemicals must be added at all times in the coking process to neutralize the maximum percentage of sulfur encountered over this unloading period. Thus, if the mixed coal flowing into the recepticle contains 1 percent of sulfur except for short time intervals during which it contains 3 percent of sulfur, sufficient chemicals to neutralize a 3 percent sulfur content must be added at all times in the coking ovens. Therefore, minimizing the maximum sulfur content constitutes a reasonable surrogate for minimizing the cost of the neutralization process.

* For a review of other approaches to scheduling problems, *see* references 2, 3, 6, 7, and 8.

THE SOLUTION METHOD

To DESCRIBE THE proposed solution method, we shall first introduce some basic terminology and notation. As remarked in the previous section, the proportions in which the materials from the barges are mixed are determined by the rates at which the flows from the piers or terminals enter into a common conveyor. Thus, with each of the k terminals we associate a 'mixing' proportion p_k that is independent of the barge being unloaded. The set of available barges or sources that supplies the kth terminal will be denoted by $S_k = (S_{1k}, S_{2k}, \cdots, S_{M_k k})$. Each barge S_{jk} supplies the same discrete unit of the basic material. In addition, the percentage q_{jk} of sulfur contained in each source S_{jk} is known. Therefore, the actual percentage Q_{jk} of sulfur that will go into the final mix from barge j at terminal k, S_{jk}, will be the percentage of sulfur in the barge q_{jk} weighted by the rate of mixing p_k; or $Q_{jk} = q_{jk}p_k$. Thus, the set $Q_k = (Q_{1k}, Q_{2k}, \cdots, Q_{M_k k})$ may be associated in one-to-one correspondence with the set S_k (*See* Fig. 1).

The time intervals required to consume a unit of material at terminal k may be computed from the relation $t_k = 1/p_k$. These times can be used to generate a schedule for the use of the terminals by the barges, since a new barge must be scheduled for unloading at terminal k every t_k units of time. It is to be emphasized that each terminal can be supplied by only one barge at a time, and that there is no mixing of successively received units of the material. We shall assume for convenience that all terminals begin to be supplied simultaneously, and that no terminal contains an initial supply of the material. However, this assumption may be eliminated through the provision of a lead factor. Finally, the time required to replace an empty barge at a terminal by the next scheduled barge is assumed to be negligible, and may be ignored. Thus, starting from time 0, the use of the ith barge scheduled at terminal k will begin at time $T_{ik} = (i - 1)t_k$.

To apply our solution method we identify a set of times $T_1, T_2, \cdots, T_{q(\max)}$ with the property that during any interval $T_q \leqq t < T_{q+1}$ exactly one unique set of barges is being unloaded at the terminals. Specifically, we accomplish this by setting $T_1 = 0$, $T_2 = \min(T_{ik})$, $T_{ik} > 0$, and generally, $T_q = \min(T_{ik})$, $T_{ik} > T_{q-1}$. The value of the largest q such that T_q is represented in the schedule will depend on the time period over which it is feasible or convenient to project the sequencing of barges.*

The decision to select barge j to be the ith one sequenced at terminal k is equivalent to adding the amount Q_{jk} of sulfur to each unit of the final

* If the number of available barges is larger than the number that can be incorporated in a particular schedule, it is reasonable to ask that the subset selected for the schedule be as nearly representative of the whole as possible. However, we do not concern ourselves with this problem in this paper.

mix over the time interval from T_{ik} to $T_{i+1,k}$. It is convenient to suppose the barges are indexed so that S_{ik} denotes that barge i at terminal k is put into use at time T_{ik}. The total amount X_q of sulfur in a unit of the final mix throughout any time interval $T_q \leqq t < T_{q+1}$ may now be identified. If, for

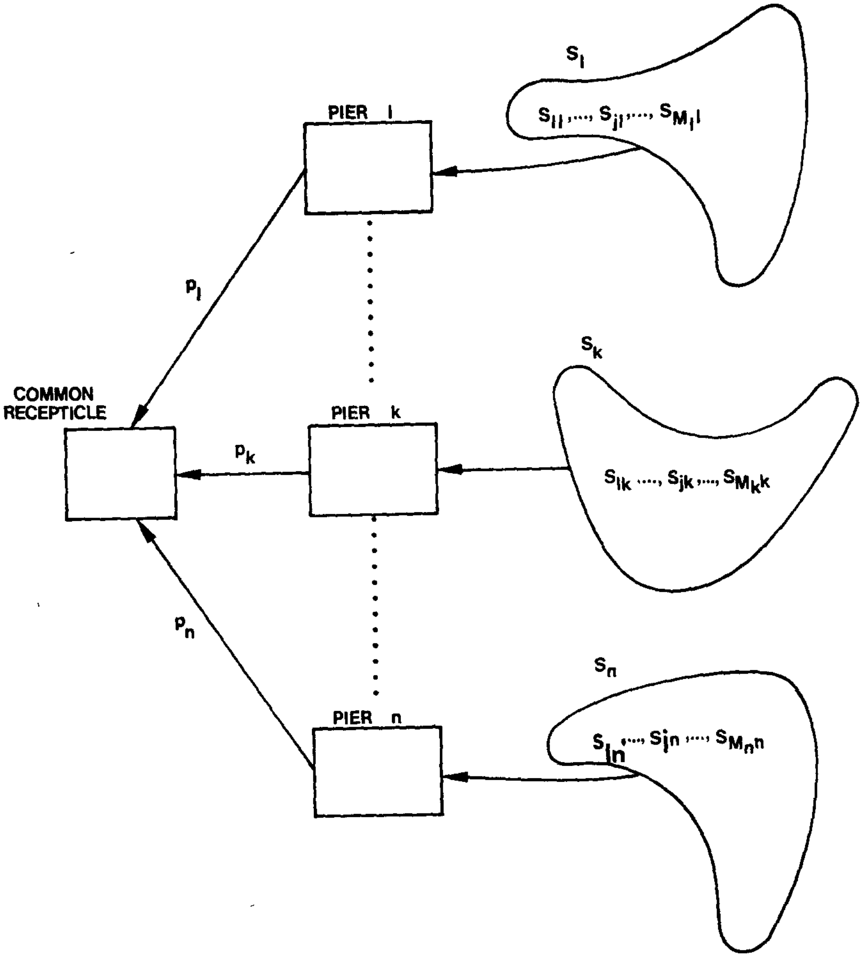

Fig. 1. Unloading system.

a given k, we let I_q denote the unique i such that $T_{ik} \leqq T_q < T_{i+1,k}$, then X_q is given by the formula $X_q = \sum_k Q_{I_q k}$.

The analytic structuring of the problem is completed via a scheduling matrix $A = (a_{qk})$, where the entries a_{qk} are the quantities $Q_{I_q k}$ from the formula for X_q. Thus for each time index q the entries of the row vector A_q

of A are the amounts of sulfur contributed per unit of mix by the barges in use during the time period $T_q \leqq t < T_{q+1}$. By affixing the column vector $X = (X_q)$ to the right of the A matrix we obtain a partitioned table in which the row elements of columns 1 through n sum to the corresponding row elements of column $n + 1$.

We will consider four related objectives that may be stated in terms of the foregoing definitions as follows: (1) minimize the maximum (X_q) for all q; (2) minimize the maximum $(X_q - M)$ for all q such that $X_q - M \geqq 0$ where $M = \text{mean } (X_q)$; (3) minimize the maximum $(|X_q - M|)$; and, (4) minimize the maximum $[(T_{q+1} - T_q)(|X_q - M|)]$. Other similar objectives are also readily amenable to the solution methods to be suggested. For example, the absolute deviations in objectives (3) and (4) above could be replaced by squared deviations.

A Method for Objective (1)

Given the indicated structuring of the problem, a method for obtaining optimal or near optimal solutions according to objective (1) may now be described. We first observe that the augmented matrix (A, X) reflects the consequences of sequencing the barges at each terminal k so that S_{ik} precedes $S_{i+1,k}$. To obtain a better sequencing in terms of objective (1), the maximum component of X, which we designate as $X_m = \max_q (X_q)$, must be reduced. Therefore, we seek a column c and a row r such that $a_{mc} > a_{rc}$, since switching the scheduling of the barges associated with these entries will reduce X_m by $a_{mc} - a_{rc}$.* However, the effect of this rescheduling will also include changes in X_q for q other than m, and for an improvement to occur we must ensure that we do not produce a new X_m as large as the old one. Thus, more extensive analysis of the augmented scheduling matrix (A, X) is required in order to project the complete consequences of a switch in the sequencing of two barges.

The time interval $T_{ik} \leqq t < T_{i+1,k}$ during which the ith barge is sequenced at terminal k is partitioned by a subset of the T_q; i.e., there are some q and some $h > q$ for which $T_{ik} = T_q$ and $T_{i+1,k} = T_h$. Thus for the ith sequenced barge at terminal k we can define the set $T_b^k = (q \mid T_{ik} \leqq T_q < T_{i+1,k}$ if and only if $T_{ik} \leqq T_b < T_{i+1,k})$, which associates any time T_b with range of q values such that T_q and T_b belong to the time interval during which a unique barge at terminal k is in use. The elements of T_b^k are the indexes of a set of consecutive rows in the A matrix. Thus, to the first criterion for a scheduling change, $a_{mc} > a_{rc}$, we add the second:

$$X_q + (a_{mc} - a_{rc}) < X_m$$

* The validity of this and related assertions may be established directly by reference to the definitions, and detailed proof is omitted.

for all q in T_r^c. This assures that the switch in the sequencing of the two barges will leave the altered values of X_q less than the specified X_m.

Using this same notation, we can specify the new matrix A' resulting from A and the new column vector X' resulting from X as a consequence of the switch in the sequencing of the two barges as follows: (i) $a'_{qc} = a_{rc}$ and $X_q' = X_q - (a_{mc} - a_{rc})$ for all $q \epsilon T_m^c$; (ii) $a'_{qc} = a_{mc}$ and $X_q' = X_q + (a_{mc} - a_{rc})$ for all $q \epsilon T_r^c$; (iii) $a'_{qk} = a_{qk}$ and $X_q' = X_q$ for all other q and k.

Thus, the method may be summarized as follows:

0. Set up the matrix A and the vector X.
1. Identify indexes m, r, and c such that $X_m = \max_q (X_q)$, $a_{mc} > a_{rc}$, and $X_q + (a_{mc} - a_{rc}) < X_m$ for all q in T_r^c. If such a set of indexes does not exist, the method stops. Otherwise,
2. Determine A' and X' as indicated in (i), (ii), and (iii). Redesignate A' and X' to be the new A and X and return to instruction 1.

The process is recursive and ends when there are no more reschedulings that satisfy the specified criteria. It must terminate since the X_q cannot be reduced indefinitely.

ILLUSTRATIVE EXAMPLE

THIS SECTION presents a numerical example illustrating the method for accommodating objective (1). Although the problem considered is of moderate size, it is representative of the class of problems to which the heuristic solution method may be applied. The example problem considers a receiving dock of a company that consists of five piers, each with an associated fleet of coal-laden barges waiting to be scheduled for unloading. We wish to discover a schedule that will minimize the maximum percentage of sulfur in the coal during unloading in accordance with the situation described in the first section.

Using the notation of the preceding section, the vector $P = (p_k; k = 1, \cdots, 5)$ of the proportions in which the materials from the piers are mixed is

$$P = (2/25, 3/25, 5/25, 7/25, 8/25).$$

There are 2 barges in the set S_1 of those available for unloading at pier 1, 3 in S_2, 5 in S_3, 7 in S_4, and 8 in S_5. These numbers have been selected so that the table will represent a scheduling cycle. From P we can determine the time required for a barge to be unloaded at pier k using the relation $t_k = 1/p_k$ (each barge carries an equal load). From the relation $T_{ik} = (i - 1)t_k$, the times at which the successive barges should begin to unload at each pier are given by

T_{i1}: 0, $12\frac{1}{2}$,
T_{i2}: 0, $8\frac{1}{3}$, $16\frac{2}{3}$,

T_{i3}:0, 5, 10, 15, 20,
T_{i4}:0, $3\frac{4}{7}$, $7\frac{1}{7}$, $10\frac{5}{7}$, $14\frac{2}{7}$, $17\frac{6}{7}$, $21\frac{3}{7}$,
T_{i5}:0, $3\frac{1}{8}$, $6\frac{1}{4}$, $9\frac{3}{8}$, $12\frac{1}{2}$, $15\frac{5}{8}$, $18\frac{3}{4}$, $21\frac{7}{8}$.
From the T_{ik} we see that the vector $T = (T_q)$ defined in the preceding section is given by $T = (0, 3\frac{1}{8}, 3\frac{4}{7}, 5, 6\frac{1}{4}, \cdots, 20, 21\frac{3}{7}, 21\frac{7}{8})$. The unloading of the barges will be completed in 25 units of time.

The percentages of coal contained in each barge are as follows:
q_{j1}:0.00500, 0.00875,
q_{j2}:0.00500, 0.00750, 0.01000,

TABLE I
Initial Table

q	T_q	$(T_{q+1} - T_q)$	S_1	S_2	S_3	S_4	S_5	X_q
1	0	$3\frac{1}{8}$	4	12	7	15	8	46
2	$3\frac{1}{8}$	$\frac{25}{56}$					12	50
3	$3\frac{4}{7}$	$1\frac{3}{7}$				14		49
4	5	$1\frac{1}{4}$			10			52
5	$6\frac{1}{4}$	$\frac{25}{28}$					14	54
6	$7\frac{1}{7}$	$1\frac{4}{21}$				10		50
7	$8\frac{1}{3}$	$1\frac{1}{24}$		9				47
8	$9\frac{3}{8}$	$\frac{5}{8}$					17	50
9	10	$\frac{5}{7}$			11			51
10	$10\frac{5}{7}$	$1\frac{11}{14}$				9		50
11	$12\frac{1}{2}$	$1\frac{11}{14}$	7				19	55
12	$14\frac{2}{7}$	$\frac{5}{7}$				8		54
13	15	$\frac{5}{8}$			13			56
14	$15\frac{5}{8}$	$1\frac{1}{24}$					24	61
15	$16\frac{2}{3}$	$1\frac{4}{21}$		6				58
16	$17\frac{6}{7}$	$\frac{25}{28}$				7		57
17	$18\frac{3}{4}$	$1\frac{1}{4}$					28	61
18	20	$1\frac{3}{7}$			16			64
19	$21\frac{3}{7}$	$\frac{25}{56}$				5		62
20	$21\frac{7}{8}$	$3\frac{1}{8}$					30	64

q_{j3}:0.00350, 0.00500, 0.00550, 0.00650, 0.00800,
q_{j4}:0.00179, 0.00250, 0.00286, 0.00321, 0.00357, 0..00500, 0.00536
q_{j5}:0.00250, 0.00375, 0.00437, 0.00531, 0.00594, 0,00750, 0.00875, 0.00937.
The quantities Q_{jk} are obtained from the relation $Q_{jk} = p_k q_{jk}$ for all j and k. For example,

$$Q_{11} = p_1 q_{11} = (\tfrac{2}{25})(0.00500) = 0.0004.$$

In order to facilitate subsequent computations, the Q_{jk} have been multiplied by 10,000 and are shown below.
Q_{j1}:4, 7,

Q_{j2}: 12, 9, 6,
Q_{j3}: 7, 10, 11, 13, 16,
Q_{j4}: 15, 14, 10, 9, 8, 7, 5,
Q_{j5}: 8, 12, 14, 17, 19, 24, 28, 30.

This scaling of the Q_{jk} implies that the resulting X_q values will be similarly scaled, and the 'true' X_q values can be obtained by dividing by 10,000. The operation of the heuristic is of course unaffected by such scaling. We have arranged the Q_{jk} in descending order for even k and in ascending order for odd k in order to reduce the variation of X_q in the initial table.

TABLE II

Results of First Iteration

q	S_1	S_2	S_3	S_4	S_5	X_q
1	4	12	16	15	8	55
2					12	59
3				14		58
4			10			52
5					14	54
6				10		50
7		9				47
8					17	50
9			11			51
10				9		50
11	7				19	55
12				8		54
13			13			56
14					24	61
15		6				58
16				7		57
17					28	61
18			7			55
19				5		53
20					30	55

This arrangement may be regarded as a first step toward achieving the desired objective. The Q_{jk} are inserted into the table with the appropriate column being determined by the subscript k. The eligible rows into which the values are placed are identified from the T_{ik} for each k. For example, the values for $k = 1$ are placed in column 1 in rows q determined by $T_{i1} = T_q$, i.e., in the rows q corresponding to times 0 and $12\frac{1}{2}$. The remaining cells under each value are left blank, although each is implicitly assigned the value of the first nonempty cell above it and in its column. The X_q may then be determined by summing the rows. The completed initial table appears as Table I. The column $(T_{q+1} - T_q)$ is not required for objective

(1), but is shown for illustrative purposes. [The column is useful for accommodating objective (4), as discussed later.]

To illustrate the rescheduling process, Table II will be derived showing the results of the first application of instructions 1 and 2 of the method. We use the notation $(a, c) - (b, c)$ to denote that the rescheduling consists of exchanging the barge associated with row a, column c, for that associated with row b, column c of the A matrix. We select $X_{20} = 64$ as X_m in the initial table. Several potential changes exist; however, the difference between $a_{20,3} = 16$ and $a_{1,3} = 7$ is the greatest in the relevant columns and the addition of the difference of 9 to X_1, X_2, and X_3 does not produce a value as large as 64. Therefore, we carry out the change (18, 3) − (1, 3), with the results shown in Table II.

TABLE III

Final Solution

q	S_1	S_2	S_3	S_4	S_5	X_q
1	4	12	16	10	14	56
2					8	50
3				15		55
4			13			52
5					12	56
6				14		55
7		9				52
8					17	57
9			11			55
10				9		50
11	7				19	55
12				8		54
13			10			53
14					24	58
15		6				55
16				5		52
17					28	56
18			7			53
19				7		55
20					30	57

We now select X_{14} as X_m, and make the exchange (13, 3) − (4, 3), which decreases X_{13} through X_{17} by 3, and increases X_4 through X_8 by the same amount. The subsequent reschedulings, not accompanied by tables, are (1, 4) − (6, 4), (2, 5) − (1, 5), (5, 5) − (1, 5), (6, 4) − (3, 4), and (16, 4) − (19, 4). The final table shown in Table III has a maximum X_q of 58. In reality, this value for the maximum was reached after three steps, but there were five occurrences of it. The last four steps eliminated four of the five, leaving only X_{14}.

It should be observed that this method also substantially reduced overall variation in the X_q and so may be used to produce a starting schedule for the methods tailored specifically to attain objectives (2), (3), and (4).

Modifications and Extension of the Method for Objective (1)

It may be noted that, because of the redundancy of data, only very small amounts of information are handled in accomplishing each single rescheduling. For practical purposes the data to be considered might be further reduced by ignoring rows associated with extremely small values of $(T_{q+1} - T_q)$. This would be particularly appropriate for objective (4).

Note too that a complete new table is not required for each exchange or iteration. The changes can be made by merely erasing or striking-out the old values and writing in the new ones.

The heuristic rescheduling process may be strengthened or quickened in several ways. For objective (1), the exchange that leaves X_m the smallest may be sought instead of selecting any exchange reducing X_m or the number of X_m's. This rule was applied to the illustrative example and the number of additional side computations required was not excessive.

The solution method as it is presently formulated does not consider the exchange of scheduled barges for more than one terminal at a time, or the exchange of more than two barges at a time. There is no procedural difficulty involved in removing these restrictions; however, the amount of additional computation required rises rapidly with increases in the numbered of alternatives considered.*

It is of course theoretically possible that a better schedule could be determined by other than a strict monotonic reduction of the greatest X_q. With larger and more intricate problems it might be profitable to 'perturb' a locally optimal schedule by allowing some number of exchanges that merely constrain X_m to lie inside a certain interval, and then begin the original process again. Because the solution process is very fast, it would also be entirely reasonable to restart the method several times from various randomly generated initial schedules. The results of successive convergences could then be compared, and the best schedule generated put to use.

This solution method can readily be extended to the more general case where the barges are not preassigned to specific terminals. For example, assume that a total of M_S barges are available for scheduling at *any* of k terminals, each having a unique mixing rate P_k. Once again, the t_k and

* The method used to determine an initial scheduling table can make a significant difference in the number of iterations required to bring the X_q close to their optimum. For such considerations, *see* reference 4.

T_{ik} may be calculated for each terminal. The T_{ik} will determine the number of barges required at each terminal during the scheduling cycle, which may be denoted by M_k. Randomly assign M_k unique barges from the available M_s barges to each of the k terminals and proceed as previously described. The only stipulation is that $\sum_k M_k \leqq M_S$.

ADDITIONAL OBJECTIVES

OBJECTIVES (2), (3), AND (4) defined in the second section may be handled in a manner similar to our proposed treatment of objective (1) by reference to an evaluation vector V whose components are given by $V_q = X_q - M$ where M = mean (X_q). An alternative definition of M, which would be a constant for a particular sequence of tables, would be $M = \sum_q (T_{q+1} - T_q)X_q/(T_c)$, where T_c represents the total time involved in the scheduling cycle.

Objective (2) may in fact be accommodated exactly as objective (1), using the criterion $V_q > 0$ as a guide to possible reschedulings. Objective (3) requires a slightly more complicated modification of the basic procedure as follows: Select V_m from the components V_q such that $|V_m| \geqq |V_q|$ for all q. If $V_m > 0$, find a row r and a column c so that

(i) $a_{mc} > a_{rc}$,
(ii) $V_q + (a_{mc} - a_{rc}) < V_m$ for all $q \in T_r^c$, and
(iii) $V_q - (a_{mc} - a_{rc}) > -V_m$ for all $q \in T_m^c$. If $V_m < 0$, find row r and column c so that
(i) $a_{rc} > a_{mc}$,
(ii) $V_q + (a_{rc} - a_{mc}) < -V_m$ for all $q \in T_m^c$, and
(iii) $V_q - (a_{rc} - a_{mc}) > V_m$ for all $q \in T_r^c$. The creation of the remainder of the new table, (AX), is exactly the same as before.

The procedure for objective (4) may be made identical to that of objective (3) by substituting for each variable V_q the product of that variable and $(T_{q+1} - T_q)$ in conditions (ii) and (iii) above.

OTHER CONSIDERATIONS AND APPLICATIONS

A VARIETY OF problems other than those involving the sequencing of barges can be put into the same general framework provided by the notation of the second section, including certain types of machine scheduling and manpower smoothing problems. The common abstract representation of these problems can in turn be formulated as an integer linear program.* To conserve space, we have not detailed these problems and their integer programming formulation here, but refer the interested reader to reference 4. However, it is worth remarking that the specific problem that generated

* For a review of solution techniques, *see* references 1 and 5.

this study, which involves about 100 barges and four terminals, would require on the order of 2,800 variables and 270 constraints. Thus, for practical application it appears that an approach (such as the one developed in the previous sections) for obtaining answers with considerably less effort than required with the integer programming formulation is particularly desirable.

REFERENCES

1. Jean Abadie (ed.), *Nonlinear and Integer Programming*, North Holland, The Netherlands, 1970.
2. Leif H. Applegren, "A Column Generation Algorithm for a Ship Scheduling Problem," *Trans. Sci.* **3**, 53–68 (1969).
3. Richard W. Conway, William L. Maxwell, AND Louis W. Miller, *Theory of Scheduling*, Addison-Wesley, Reading, Mass., 1967.
4. Fred Glover AND James S. Dyer, "A Heuristic for Sequencing Barge Unloadings at Regulated Flow Terminals to Minimize the Variation of a Common Constituent in the Final Output," Working Paper 70–8, Graduate School of Business, The University of Texas at Austin, Austin, Texas, September 1969.
5. T. C. Hu, *Integer Programming and Network Flows*, Addison-Wesley, Reading, Mass., 1969.
6. E. L. Lawler, "The Quadratic Assignment Problem," *Management Sci.* **9**, 586–599 (1963).
7. J. F. Muth AND G. L. Thompson, *Industrial Scheduling*, Prentice-Hall, 1963.
8. Harvey M. Wagner, "An Integer Linear-Programming Model for Machine Scheduling," *Naval Res. Log. Quart.* **6**, 131–140 (1959).

(Received September 1969)