A Constructive Algorithm for Computing the Reachability Matrix

R. S. H. MAH

Department of Chemical Engineering
Northwestern University
Evanston, Illinois 60201

The use of the reachability matrix as a means of partitioning process flow sheets and equations in chemical engineering computation has been proposed at various times in published literature (Steward, 1962; Himmelblau, 1966; Ledet and Himmelblau, 1970). The attractive feature of the reachability matrix is its conceptual simplicity. Although significant improvements have been made in computational efficiency, the schemes proposed so far are basically enumerative in character. The purpose of this note is to point out an alternative approach which is based on successive augmentation of the graph and which appears to be simpler and faster.

The reachability matrix R of a graph G may be defined in terms of its adjacency matrix A

$$R = (A + A^2 + A^3 + \ldots + A^n)^* = B^* \qquad (1)$$

where B^* denotes a matrix of zeros and ones (a Boolean matrix) formed from B as follows:

$$B^*_{ij} = \begin{cases} 0, & \text{if } B_{ij} = 0 \\ 1, & \text{if } B_{ij} \neq 0 \end{cases} \qquad (2)$$

Alternative schemes for computing the reachability matrix were discussed by Ledet and Himmelblau (1970). They are

1. Direct computation of powers of A using Boolean algebra [that is, $a + b := \max(a, b)$ and $a \times b := \min(a, b)$]. To form the product of two matrices in this way requires $n^2[n + (n-1)]$ comparisons, and since there are $(n-1)$ products to be formed, approximately $2n^4$ comparisons are required to compute R.

2. Since A is a matrix of zeros and ones, row i of the product $(A\,B)^*$ is the union of the rows of B corresponding to the nonzero elements in row i of A. If union of rows is used in place of Boolean multiplications and additions to form the powers of A, the computing time may be reduced by as much as a factor of four, according to Ledet and Himmelblau (1970).

3. The reachability matrix can also be expressed as

$$R = [A(I + A^2 + \ldots + A^{n-1})]^* = [A(I + A)^{n-1}]^* \qquad (3)$$

and since no additional nonzero elements will be generated by higher powers of A beyond A^n, instead of computing $(I + A)^{n-1}$, we could just as well compute $(I + A)^p$ so long as $p = 2^m \geq (n-1)$. If successive squaring is used, $m \geq \log_2(n-1)$. Thus the total number of operations required to compute R is approximately $\frac{1}{2}n^3 \log_2 n$.

Now recalling that $(A^k)_{ij}$ gives the number of paths in G of length k from node i to node j (Norman, 1965), it becomes evident that Equation (1) represents the direct enumeration of all possible paths in G. If we are only interested in the existence or absence of paths, irrespective of lengths or number of possible paths, this is clearly a rather wasteful procedure.

The alternative approach which is advanced in this note is to augment G successively by the addition of an arc whenever a path is discovered between two nodes in G. After all such possible arcs have been added, the augmented graph will correspond to the reachability matrix R.

We shall now outline a scheme for systematic augmentation of arcs.

Let $C^{(i)}$ denote the adjacency matrix of the augmented graph after i steps. Then we define

$$C^{(0)} = A \qquad (4)$$

$$C^{(i)}_{jk} = \begin{cases} 1 & , \text{ if } C^{(i-1)}_{ji} = 1 \;\&\; C^{(i-1)}_{ik} = 1 \\ C^{(i-1)}_{jk} & \text{otherwise} \end{cases} \qquad (5)$$

If Equation (5) is applied successively to $i = 1, 2, \ldots, n$, we claim that

$$C^{(n)} = R \qquad (6)$$

To prove the validity of this claim, let us note that after the first step all paths of the neighboring nodes of node 1 through node 1 will be represented by augmented arcs. Similarly, after the second step all paths of the nodes, which are neighbors of node 1, node 2 or both, through node 1, node 2 or both will be represented by augmented arcs, and so on. Since any node in a connected graph is the neighbor of at least one other node, all paths in G will be represented by augmented arcs in $C^{(n)}$. Note that the outcome is independent of the order of the nodes and that the status of an isolated node will be correctly reflected.

At each step of this algorithm $2(n-1)$ comparisons are made to search for nonzero elements in row i and column i, and s^2 assignments or additions are made for arc augmentation, where $s \leq n - 1$. Hence to compute R using this procedure will require $2n(n-1) + ns^2$ operations.

The application of this algorithm to a simple example is illustrated in Figure 1 which shows a graph G, its adja-

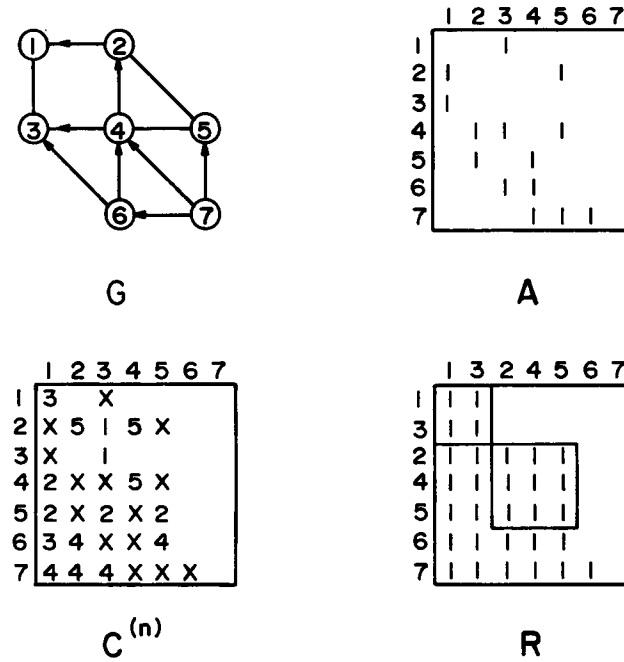

Fig. 1. An illustration of arc-augmentation algorithm.

cency matrix A, the augmented matrix $C^{(n)}$, and finally the reachability matrix after permutation. In order to make it easier to follow the intermediate steps, the original entries are denoted by "x", and "1", "2", "3", etc. are used to denote the successive arc augmentation in $C^{(i)}$. In actual implementation the storage of only one matrix will be required.

Methods based on reachability matrix computation have not been previously favored in partitioning process flow sheets and equations because they are generally slower than the alternative algorithms using path tracing and list processing (Sargent and Westerberg, 1964; Steward, 1965; Billingsley, 1967). For instance, comparison of computing times reported by Ledet and Himmelblau (1970) indicates that the "Boolean powers" method (Method 3) is slower than either Steward's algorithm or Billingsley's algorithm by a factor of up to 3. The disparity is greatest for the largest problem tested which contains 126 equations. It is interesting that the maximum computing time ratio is almost the same as $\frac{1}{2}\log_2(126)$ which is the approximate ratio of operation counts for the "Boolean powers" method and the arc-augmentation algorithm.*

* The author is indebted to the referee for pointing out a partitioning method due to Kevorkian and Snoek (1973), who arrived at a similar result based on a formula for computing the successors (and predecessors) of each node. In this instance, path tracing and arc augmentation appear to lead to the same result through different intermediate steps. Instead of tracing the successors of each node, the constructive algorithm augments all the paths associated with the neighbors of each node in turn.

NOTATION

: = replaced by

LITERATURE CITED

Billingsley, D. S., *Chem. Eng. Sci.*, **22**, 719 (1967).

Himmelblau, D. M., "Decomposition of Large Scale Systems—I. Systems Composed of Lumped Parameter Elements," *ibid.*, **21**, 425 (1966).

Ledet, W. P., and D. M. Himmelblau, "Decomposition Procedures for the Solving of Large Scale Systems," *Adv. Chem. Eng.*, 8, p. 185, Academic Press, N. Y. (1970).

Norman, R. L., "A Matrix Method for Location of Cycles of a Directed Graph," *AIChE J.*, **11**, 450 (1965).

Sargent, R. W. H., and A. W. Westerberg," 'SPEED-UP' in Chemical Engineering Design," *Trans. Instn. Chem. Engrs.*, **42**, T190 (1964).

Steward, D. V., "On an Approach to Techniques for the Analysis of the Structure of Large Systems of Equations," *SIAM Review*, **4**, 321 (1962).

———., "Partitioning and Tearing Systems of Equations," *J. SIAM Numer. Anal., Ser. B*, **2**, 345 (1965).

Kevorkian, A. K., and J. Snoek, "Decomposition in Large Scale Systems. Theory and Applications of Structural Analysis in Partitioning, Disjointing and Constructing Hierarchical Systems," in *Decomposition of Large Scale Problems*, D. M. Himmelblau (ed.), North Holland, Amsterdam (1973).

Manuscript received May 20, 1974; revision received and accepted July 25, 1974.

Experimental Almost Isothermal Reactor

JENN-MING STEPHEN HUANG and ALLEN J. BARDUHN

Department of Chemical Engineering and Materials Science
Syracuse University, Syracuse, New York 13210

In those cases where laboratory or pilot plant determination of chemical reaction rate data are taken in a tubular flow reactor, the equipment may consist of a long tube immersed in a bath of water, molten lead, or molten salt at some fixed temperature. Due to the difficulty of measuring the actual reaction temperatures except at the inlet and outlet of such a reactor, the conditions are usually so chosen that the reaction and bath temperatures may be assumed equal and the reaction thus isothermal. Isothermal rate data are much simpler to interpret and correlate and they minimize the question of how much the accuracy of the results depend on the method of treating the data. Towell and Martin (1961) present a method for analysis of data from a nonisothermal plug flow reactor, but it requires measuring the complete axial temperature profile and thus both the measurements and the analysis are complicated.

No tubular flow reactor is truly isothermal since the heat of reaction changes the temperature of the reaction mixture and some temperature difference is necessary to carry away or supply the heat to bring the mixture back to the bath temperature. The isothermal condition can be approached to any degree necessary, however, by making the reactants adequately dilute in the feed stream (adding inert gas or solvent), by increasing the heat transfer area per unit volume of reactor (using a small diameter tube), by conducting the reaction at temperatures where the rate is low, or by improving the rate of heat transfer between the tube wall and reacting mixture (for example, using high velocities). Some of these remedies may not be desired, convenient, or possible, however, and thus the reactor may not operate isothermally to the degree necessary for assuming the observed reaction rate to be truly for the assumed temperature.

It is the purpose of this note (1) to present a simple method for predicting whether the reaction is adequately isothermal and (2) to provide a simple relation for correcting the rate constants to the assumed isothermal temperature when it is not. The method does not require measurement of the temperatures except at the reactor entrance, but it does require that the actual reaction temperature be typically no more than a few tens of degrees different from the initial temperature, that is, that the reaction be almost isothermal.

ANALYSIS

The system envisioned consists of a plug flow tubular reactor immersed in a bath which maintains the tube wall temperature at T_w. The reacting mixture enters the reactor at T_w and the composition is known at the entrance and exit of the tube. For small tubes the effect of axial disper-

Correspondence concerning this note should be addressed to A. J. Barduhn.