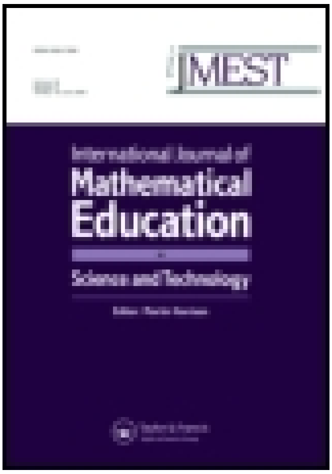

International Journal of Mathematical Education in Science and Technology
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tmes20

A partitioning algorithm for solving systems of linear equations

Theodore J. Sheskin [a]
[a] Industrial Engineering Department , Cleveland State University , Cleveland, Ohio 44115, USA
Published online: 09 Jul 2006.

To cite this article: Theodore J. Sheskin (1996) A partitioning algorithm for solving systems of linear equations, International Journal of Mathematical Education in Science and Technology, 27:5, 641-648, DOI: 10.1080/0020739960270502

To link to this article: http://dx.doi.org/10.1080/0020739960270502

INT. J. MATH. EDUC. SCI. TECHNOL., 1996, VOL. 27, NO. 5, 641–648

A partitioning algorithm for solving systems of linear equations

by THEODORE J. SHESKIN
Industrial Engineering Department, Cleveland State University, Cleveland, Ohio 44115, USA

(*Received 24 August 1994*)

An interesting new partitioning algorithm for solving simultaneous linear equations is presented. The partitioning algorithm consists of a procedure called matrix reduction which is followed by back substitution. During the matrix reduction procedure the algorithm generates a sequence of partitioned augmented matrices which are successively reduced in size by computing their Schur complements. A numerical example problem for a linear system of equations for an electrical network is solved.

1. Introduction

Systems of simultaneous linear equations arise in virtually every area of science, engineering, and social science. For example, in an electrical circuit we may apply Kirchhoff's voltage law to obtain a linear system of equations for the currents flowing in closed paths. In statistics we may use the method of least squares to find a line of best fit for experimental data by solving a system of linear equations for a slope and intercepts. We may analyse the economic behaviour of a society by formulating a Leontief input–output model in which linear equations represent the interrelationships among sectors.

In this paper we present an interesting new partitioning algorithm for solving simultaneous linear equations or inverting a matrix. The partitioning algorithm consists of two phases called matrix reduction and back substitution. Matrix reduction, which is algebraically equivalent to Gaussian elimination, is followed by back substitution just as Gaussian elimination is. The partitioning algorithm has the same operation count as Gaussian elimination with back substitution.

To begin the matrix reduction phase of the partitioning algorithm, we form the augmented matrix for a linear system of equations by adjoining the vector of right-hand constants to the coefficient matrix. We partition the augmented matrix into the following four submatrices: (1) a single element at the intersection of the first row and column; (2) a row vector consisting of the remaining elements in row one; (3) a column vector consisting of the remaining elements in column one; and (4) a submatrix containing all elements below row one and to the right of column one. We store the first row of the partitioned matrix. To execute step one of matrix reduction we compute the Schur complement [1] of the single element in the upper left-hand corner of the matrix. The Schur complement represents a new augmented matrix which has been reduced in size by one row and one column. We store the first row of the reduced matrix, partition the matrix as before, and once again compute the Schur complement of the element in its upper left hand corner. Matrix reduction is repeated in successive steps to produce a sequence of progressively smaller reduced matrices. We store the first row of each reduced matrix. Matrix reduction ends when a single row vector is produced. At this point we use back substitution to find the solution to the linear system.

The partitioning algorithm can be modified to invert a matrix by replacing the vector of right-hand side constants with an identity matrix. Other partitioning algorithms which have been published include the method of bordering [2] and matrix inversion by augmentation and reduction [3]. The matrix reduction procedure originated with Kemeny and Snell [4]. The name matrix reduction and its use in a partitioning algorithm for computing steady-state probabilities in a Markov chain are due to Sheskin [5]. Meyer [6] also used the matrix reduction procedure but called it stochastic complementation. Matrix reduction has, in addition, been used to compute a fundamental matrix [7] and absorption probabilities [8] for a Markov chain. We apply the partitioning algorithm to solve an example problem involving a linear system of equations obtained by applying Kirchhoff's voltage law to an electrical network.

2. Review of Gaussian elimination

Suppose that we want to solve a linear system of n equations in n unknowns. We let the n unknowns be $x_1, x_2, \ldots, x_n$. In equation form we can write the linear system as

$$\left.\begin{array}{l} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \quad\vdots \qquad\qquad\qquad\qquad\quad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{array}\right\} \tag{1}$$

To rewrite the linear system in matrix form we let

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

We rewrite the linear system (1) as

$$\mathbf{Ax} = \mathbf{b} \tag{2}$$

We form the augmented matrix, $\ddot{\mathbf{A}}$, of the linear system by adjoining the column vector $\mathbf{b}$ to the coefficient matrix $\mathbf{A}$ as the $(n + 1)$st column. In column $n + 1$ of the augmented matrix we rename the components b_i to be $a_{i,n+1}$, for $1 \leqslant i \leqslant n$. The augmented matrix is

$$\ddot{\mathbf{A}} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & a_{1,n+1} \\ a_{21} & a_{22} & \cdots & a_{2n} & a_{2,n+1} \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & a_{n,n+1} \end{array}\right] \tag{3}$$

Gaussian elimination is performed in $n-1$ steps to reduce the augmented matrix to upper triangular form. To begin step one we let $\ddot{\mathbf{A}}^{(1)} = \ddot{\mathbf{A}} = [a_{ij}]$ for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant n+1$. When we enter step k we have a system of $(n-k+1)$ equations in the $(n-k+1)$ unknowns $x_k, x_{k+1}, \ldots, x_n$ for which the augmented matrix is

$$\ddot{\mathbf{A}}^{(\mathbf{k})} = \begin{bmatrix} a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{k,n+1}^{(k)} \\ a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n+1}^{(k)} \\ \vdots & \vdots & & \vdots \\ a_{nk}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{n,n+1}^{(k)} \end{bmatrix} \tag{4}$$

We assume that $a_{kk}^{(k)} \neq 0$. We store the first row of $\ddot{\mathbf{A}}^{(\mathbf{k})}$. We perform elementary row operations to eliminate x_k from the last $(n-k)$ equations by subtracting from the ith equation, for $k+1 \leqslant i \leqslant n$, the multiple $a_{ik}^{(k)}/a_{kk}^{(k)}$ of the first equation. That is, in step k the elements $a_{ij}^{(k)}$ with $i, j > k$ are transformed according to the formula

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - a_{ik}^{(k)} a_{kj}^{(k)} / a_{kk}^{(k)}$$

for $k+1 \leqslant i \leqslant n$, and $k+1 \leqslant j \leqslant n+1$.

At the end of step k we obtain a system of $(n-k)$ equations in the $(n-k)$ unknowns $x_{k+1}, x_{k+2}, \ldots, x_n$ for which the augmented matrix is

$$\ddot{\mathbf{A}}^{(\mathbf{k+1})} = \begin{bmatrix} (a_{k+1,k+1}^{(k)} - a_{k+1,k}^{(k)} a_{k,k+1}^{(k)}/a_{kk}^{(k)}) & \cdots & (a_{k+1,n+1}^{(k)} - a_{k+1,k}^{(k)} a_{k,n+1}^{(k)}/a_{kk}^{(k)}) \\ \vdots & \vdots & \vdots \\ (a_{n,k+1}^{(k)} - a_{nk}^{(k)} a_{k,k+1}^{(k)}/a_{kk}^{(k)}) & \cdots & (a_{n,n+1}^{(k)} - a_{nk}^{(k)} a_{k,n+1}^{(k)}/a_{kk}^{(k)}) \end{bmatrix}$$

3. Matrix reduction

To transform Gaussian elimination into matrix reduction we observe that the matrix $\ddot{\mathbf{A}}^{(\mathbf{k+1})}$ produced in step k of Gaussian elimination is equal to the Schur complement of the element $a_{kk}^{(k)}$ in the upper left hand corner of the augmented matrix $\ddot{\mathbf{A}}^{(\mathbf{k})}$. To verify this observation we shall compute the Schur complement of element $a_{kk}^{(k)}$ in $\ddot{\mathbf{A}}^{(\mathbf{k})}$. We partition the augmented matrix $\ddot{\mathbf{A}}^{(\mathbf{k})}$ in the following manner. Let

$$\ddot{\mathbf{A}}^{(\mathbf{k})} = \left[\begin{array}{c|ccc} a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{k,n+1}^{(k)} \\ \hline a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n+1}^{(k)} \\ \vdots & \vdots & & \vdots \\ a_{nk}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{n,n+1}^{(k)} \end{array}\right]$$

$$= \begin{bmatrix} a_{kk}^{(k)} & \mathbf{u}^{(\mathbf{k})} \\ \mathbf{r}^{(\mathbf{k})} & \mathbf{T}^{(\mathbf{k})} \end{bmatrix} \tag{5}$$

We compute

$$
\begin{aligned}
(\ddot{\mathbf{A}}^{(\mathbf{k})} | a_{kk}^{(k)}) &= \text{the Schur complement of } a_{kk}^{(k)} \text{ in } \ddot{\mathbf{A}}^{(\mathbf{k})} \\
&= \mathbf{T}^{(\mathbf{k})} - \mathbf{r}^{(\mathbf{k})}(a_{kk}^{(k)})^{-1}\mathbf{u}^{(\mathbf{k})} \\
&= \mathbf{T}^{(\mathbf{k})} - \begin{bmatrix} a_{k+1,k}^{(k)} \\ \vdots \\ a_{nk}^{(k)} \end{bmatrix} (a_{kk}^{(k)})^{-1} [a_{k,k+1}^{(k)} \cdots a_{k,n+1}^{(k)}] \\
&= \ddot{\mathbf{A}}^{(\mathbf{k+1})} = [a_{ij}^{(k+1)}] \quad \text{for } k+1 \leqslant i \leqslant n \quad \text{and} \quad k+1 \leqslant j \leqslant n+1
\end{aligned}
$$

At step k of both Gaussian elimination and matrix reduction we perform the same arithmetic operations to transform an augmented matrix $\ddot{\mathbf{A}}^{(\mathbf{k})}$ into a reduced matrix $\ddot{\mathbf{A}}^{(\mathbf{k+1})}$. The only difference is that in Gaussian elimination we execute elementary row operations on $\ddot{\mathbf{A}}^{(\mathbf{k})}$ while in matrix reduction we compute a Schur complement of $\ddot{\mathbf{A}}^{(\mathbf{k})}$. Since both procedures perform the same arithmetic operations at every step and both have $n-1$ steps, they are algebraically equivalent.

4. Partitioning algorithm

The partitioning algorithm consists of matrix reduction followed by back substitution. We present the detailed steps of the partitioning algorithm for solving the linear system (2).

4.1. *Matrix reduction*

1. Form the augmented matrix, $\ddot{\mathbf{A}} = [a_{ij}]$, for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant n+1$, as in relation (3).
2. Initialize $k = 1$.
3. Let $\ddot{\mathbf{A}}^{(\mathbf{k})} = \ddot{\mathbf{A}} = [a_{ij}]$ for $k \leqslant i \leqslant n$ and $k \leqslant j \leqslant n+1$, as in relation (4).
4. Perform partial pivoting [9]. Choose r as the smallest integer for which $|a_{rk}^{(k)}| = \max|a_{ik}^{(k)}|$, for $k \leqslant i \leqslant n$. Interchange rows k and r.
5. Partition $\ddot{\mathbf{A}}^{(\mathbf{k})}$ as in relation (5).

$$
\ddot{\mathbf{A}}^{(\mathbf{k})} = \left[\begin{array}{c|ccc}
a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{k,n+1}^{(k)} \\
\hline
a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n+1}^{(k)} \\
\vdots & \vdots & & \vdots \\
a_{nk}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{n,n+1}^{(k)}
\end{array}\right]
$$

$$
= \begin{bmatrix} a_{kk}^{(k)} & \mathbf{u}^{(\mathbf{k})} \\ \mathbf{r}^{(\mathbf{k})} & \mathbf{T}^{(\mathbf{k})} \end{bmatrix} \tag{5}
$$

6. Store the first row of $\ddot{\mathbf{A}}^{(\mathbf{k})}$.
7. Compute

$$
\ddot{\mathbf{A}}^{(\mathbf{k+1})} = \mathbf{T}^{(\mathbf{k})} - \mathbf{r}^{(\mathbf{k})}(a_{kk}^{(k)})^{-1}\mathbf{u}^{(\mathbf{k})}
$$

8. Increment k by 1. If $k < n$, go to step 3. Otherwise, go to back substitution.

4.2. *Back substitution*

1. $x_n = a_{n,n+1}^{(n)}/a_{nn}^{(n)}$.
2. $k = n - 1$.
3. $x_k = (a_{k,n+1}^{(k)} - \sum_{h=k+1}^{n} a_{kh}^{(k)} x_h)/a_{kk}^{(k)}$.
4. Decrement k by 1.
5. If $k > 0$, go to step 3. Otherwise, stop.

5. **Operation count**

Since matrix reduction performs the same arithmetic operations as Gaussian elimination, and both procedures are followed by back substitution, the partitioning algorithm has the same operation count as Gaussian elimination with back substitution [10]. At step k of matrix reduction or Gaussian elimination we have $(n - k)$ divisions, $(n - k)(n - k + 1)$ multiplications, and $(n - k)(n - k + 1)$ subtractions. After $n - 1$ steps we have a total of $n(n - 1)/2$ divisions, $n/\mathrm{n}^2 - 1)/3$ multiplications, and $n(n^2 - 1)/3$ subtractions, When we use back substitution to solve for a single variable x_k, we have one division, $(n - k)$ multiplications, and $(n - k)$ subtractions. When we use back substitution to evaluate n variables, we have n divisions, $n(n - 1)/2$ multiplications, and $n(n - 1)/2$ subtractions. Both the partitioning algorithm and Gaussian elimination with back substitution have a total of $n(n + 1)/2$ divisions, $n(n - 1)(2n + 5)/6$ multiplications, and $n(n - 1)(2n + 5)/6$ subtractions. In comparison with the multiplications, the subtractions are relatively easy and the divisions are relatively few. Therefore, for large n the number of operations is of the order of magnitude $n^3/3$.

6. **Example problem**

Kirchhoff's voltage law states that in an electrical network the algebraic sum of the voltage drops around any closed path in a given direction is zero [11]. In the network in Figure 1 we identify four closed paths to which we assign the circulating currents I_1, I_2, I_3, and I_4, as indicated by the arrows. When we apply

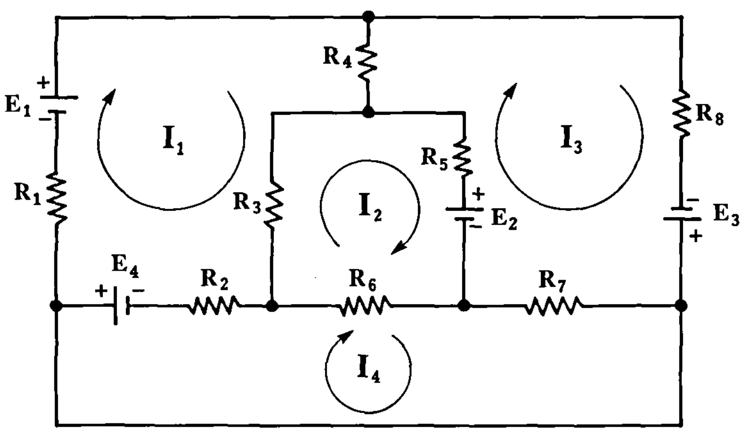

Figure 1. Electrical network.

Kirchhoff's voltage law to each of the four closed paths we get the following linear system of equations:

$$\begin{bmatrix} (R_1+R_2+R_3+R_4) & -R_3 & -R_4 & -R_2 \\ -R_3 & (R_3+R_5+R_6) & -R_5 & -R_6 \\ -R_4 & -R_5 & (R_4+R_5+R_7+R_8) & -R_7 \\ -R_2 & -R_6 & -R_7 & (R_2+R_6+R_7) \end{bmatrix}$$

$$\times \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} E_1+E_4 \\ -E_2 \\ E_2+E_3 \\ -E_4 \end{bmatrix}$$

We substitute the following numerical values for the resistances, in ohms, and the direct-current voltage sources, in volts.

R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	E_1	E_2	E_3	E_4
12	47	20	33	75	15	62	10	9	12	6	36

The system of equations becomes

$$\begin{bmatrix} 112 & -20 & -33 & -47 \\ -20 & 110 & -75 & -15 \\ -33 & -75 & 180 & -62 \\ -47 & -15 & -62 & 124 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 45 \\ -12 \\ 18 \\ -36 \end{bmatrix}$$

We will use the partitioning algorithm to solve for the four currents I_1, I_2, I_3, and I_4, in amperes. (We will ignore the fact that the coefficient matrix is symmetric in order to demonstrate the generality of the partitioning algorithm.)

6.1. *Matrix reduction*

$$\ddot{\mathbf{A}} = \left[\begin{array}{cccc|c} 112 & -20 & -33 & -47 & 45 \\ -20 & 110 & -75 & -15 & -12 \\ -33 & -75 & 180 & -62 & 18 \\ -47 & -15 & -62 & 124 & -36 \end{array}\right]$$

$$\ddot{\mathbf{A}}^{(1)} = \left[\begin{array}{c|cccc} 112 & -20 & -33 & -47 & 45 \\ \hline -20 & 110 & -75 & -15 & -12 \\ -33 & -75 & 180 & -62 & 18 \\ -47 & -15 & -62 & 124 & -36 \end{array}\right] = \begin{bmatrix} a_{11}^{(1)} & \mathbf{u}^{(1)} \\ \mathbf{r}^{(1)} & \mathbf{T}^{(1)} \end{bmatrix}$$

$$\ddot{\mathbf{A}}^{(2)} = \mathbf{T}^{(1)} - \mathbf{r}^{(1)}(a_{11}^{(1)})^{-1}\mathbf{u}^{(1)}$$

$$\ddot{\mathbf{A}}^{(2)} = \left[\begin{array}{c|ccc} 106{\cdot}429 & -80{\cdot}893 & -23{\cdot}393 & -3{\cdot}964 \\ \hline -80{\cdot}893 & 170{\cdot}277 & -75{\cdot}848 & 31{\cdot}259 \\ -23{\cdot}393 & -75{\cdot}848 & 104{\cdot}277 & -17{\cdot}116 \end{array}\right] = \begin{bmatrix} a_{22}^{(2)} & \mathbf{u}^{(2)} \\ \mathbf{r}^{(2)} & \mathbf{T}^{(2)} \end{bmatrix}$$

$$\ddot{\mathbf{A}}^{(3)} = \mathbf{T}^{(2)} - \mathbf{r}^{(2)}(a_{22}^{(2)})^{-1}\mathbf{u}^{(2)}$$

$$\ddot{\mathbf{A}}^{(3)} = \left[\begin{array}{c|cc} 108{\cdot}793 & -93{\cdot}628 & 28{\cdot}246 \\ \hline -93{\cdot}628 & 99{\cdot}135 & -17{\cdot}987 \end{array}\right] = \begin{bmatrix} a_{33}^{(3)} & \mathbf{u}^{(3)} \\ \mathbf{r}^{(3)} & \mathbf{T}^{(3)} \end{bmatrix}$$

$$\ddot{\mathbf{A}}^{(4)} = \mathbf{T}^{(3)} - \mathbf{r}^{(3)}(a_{33}^{(3)})^{-1}\mathbf{u}^{(3)}$$

$$\ddot{\mathbf{A}}^{(4)} = [18{\cdot}558 | 6{\cdot}322] = [a_{44}^{(4)} \quad a_{45}^{(4)}]$$

6.2. *Back substitution*

1. $I_4 = a_{45}^{(4)}/a_{44}^{(4)}$
   $= 6{\cdot}322/18{\cdot}558 = 0{\cdot}341$ amperes
2. $I_3 = (a_{35}^{(3)} - a_{34}^{(3)}I_4)/a_{33}^{(3)}$
   $= (28{\cdot}246 - (-93{\cdot}628)(0{\cdot}341))/108{\cdot}793 = 0{\cdot}553$ amperes
3. $I_2 = (a_{25}^{(2)} - a_{23}^{(2)}I_3 - a_{24}^{(2)}I_4)/a_{22}^{(2)}$
   $= (-3{\cdot}964 - (-80{\cdot}893)(0{\cdot}553) - (-23{\cdot}393)(0{\cdot}341))/106{\cdot}429$
   $= 0{\cdot}458$ amperes
4. $I_1 = (a_{15}^{(1)} - a_{12}^{(1)}I_2 - a_{13}^{(1)}I_3 - a_{14}^{(1)}I_4)/a_{11}^{(1)}$
   $= (45 - (-20)(0{\cdot}458) - (-33)(0{\cdot}553) - (-47)(0{\cdot}341))/112$
   $= 0{\cdot}790$ amperes

7. Conclusion

The matrix reduction phase of the partitioning algorithm is an interesting alternative procedure to Gaussian elimination for solving a linear system of equations. By transforming Gaussian elimination into matrix reduction we have shown that both procedures perform the same arithmetic operations. We have used the partitioning algorithm to solve a linear system of equations for an electrical network.

Several variations of the partitioning algorithm are possible. For example, we can partition an augmented matrix with respect to a single element in the lower right-hand corner and follow matrix reduction with forward substitution. We can also do block reduction by partitioning an augmented matrix so that we take the Schur complement of an invertible submatrix consisting of more than one element.

References

[1] Cottle, R. W., 1974, *Lin. Alg. and Applic.*, **8**, 189–211.
[2] Faddeeva, V. N., 1959, *Computational Methods of Linear Algebra* (New York: Dover).
[3] Sheskin, T. J., 1991, *Int. J. Math. Educ. Sci. Technol.*, **22**, 103–110.
[4] Kemeny, J. G., and Snell, J. C., 1960, *Finite Markov Chains* (Princeton, N.J.: Van Nostrand).
[5] Sheskin, T. J., 1985, *Op. Res.*, **33**, 228–235.
[6] Meyer, C. D., 1989, *SIAM Rev.*, **31**, 240–272.

[7] Sheskin, T. J., 1991, in *Numerical Solutions of Markov Chains*, edited by W. J. Stewart (New York: Marcel Dekker) pp. 671–672.
[8] Sheskin, T. J., 1991, *Int. J. Math. Educ. Sci. Technol.*, **22**, 799–806.
[9] Dahlquist, G., and Bjorck, A., 1974, *Numerical Methods* (Englewood Cliffs, NJ: Prentice-Hall).
[10] Gastinel, N., 1970, *Linear Numerical Analysis* (New York: Academic Press).
[11] Cheng, D. K., 1959, *Analysis of Linear Systems* (Reading, MA: Addison-Wesley).