CHAPTER V

THE MODIFIED DANILEWSKI METHOD

1. Introduction. In this chapter we describe an algorithm which uses residue arithmetic to reduce an nxn integral matrix A to Frobenius form, and obtains exact integral factors of the characteristic polynomial. The algorithm is based on the fact that if A has as its characteristic polynomial

$$f(\lambda) = \det(A-\lambda I)$$

$$= p_1(\lambda) \cdots p_\ell(\lambda), \tag{1.1}$$

where $p_i(\lambda)$ is an integral polynomial given by

$$p_i(\lambda) = (-1)^{n_i}(\lambda^{n_i} - b_1^{(i)}\lambda^{n_i-1} - \cdots - b_{n_i-1}^{(i)}\lambda - b_{n_i}^{(i)}), \tag{1.2}$$

then the characteristic polynomial modulo m of $|A|_m$, as defined in chapter III, section 5, is

$$|f(\lambda)|_m = |p_1(\lambda) \cdots p_\ell(\lambda)|_m. \tag{1.3}$$

Thus, if we compute a bound* β, where

$$\beta \geq \max_{i,j} |b_j^i|, \tag{1.4}$$

and if we choose m so that†

$$m \geq 2\cdot\beta, \tag{1.5}$$

and finally if we compute $|f(\lambda)|_m$ using modular arithmetic, then

$$|f(\lambda)|_m = f(\lambda). \tag{1.6}$$

---

* Methods for computing β will be discussed in chapter VI.

† The 2 is necessary because we are using the symmetric residue system.

In the next section we describe a method for computing the $|f_i|_m$ and, hence, also $|f(\lambda)|_m$, using modular arithmetic based on Danilewski's method.

2. Computing $|f(\lambda)|_m$. We recall from theorem (IV-2.4) that matrices which are similar modulo m have the same characteristic polynomial modulo m. Thus, if we can reduce $|A|_m$ to the form

$$C = \begin{bmatrix} 0 & 0 & 0 \ldots 0 & x_n \\ 1 & 0 & 0 \ldots 0 & x_{n-1} \\ 0 & 1 & 0 \ldots 0 & x_{n-2} \\ . & . & . \ldots . & . \\ 0 & 0 & 0 \ldots 0 & x_2 \\ 0 & 0 & 0 \ldots 1 & x_1 \end{bmatrix} , \tag{2.1}$$

where $C = |S^{-1}(m)\ |A|_m\ S|_m$, then the characteristic polynomial modulo m of C, and hence of $|A|_m$ is

$$|f(\lambda)|_m = |(-1)^n(\lambda^n - x_1\lambda^{n-1} - \ldots - x_{n-1}\lambda - x_n)|_m . \tag{2.2}$$

Therefore, since the elements of C are reduced modulo m, we have, from (1.2),

$$x_i = |b_i^{(1)}|_m . \tag{2.3}$$

This assumes that $|A|_m$ is nonderogatory modulo m. The case in which $|A|_m$ is derogatory modulo m is treated below.

We now consider the problem of reducing $|A|_m$ to the form (2.1). This is accomplished in a finite number of similarity transformations modulo m of the form

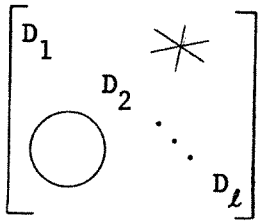
$$|A_{k+1}|_m = |S_k^{-1}(m)\ |A_k|_m\ S_k\ |_m , \tag{2.4}$$

where $|A_o|_m = |A|_m$. We shall let the S_k be elementary matrices modulo m as described in chapter II, section 3.

We could proceed in a manner analogous to the method for ordinary arithmetic described in chapter I, first reducing the matrix to Hessenberg form, then reducing the Hessenberg form to Frobenius form. However, the main reason for separating the computation for ordinary arithmetic into two stages is the difference in computational stability considerations involved in each step. When performing the analogous operations in modular arithmetic, all arithmetic is exact. Therefore, computational stability is of no concern. Furthermore, if the two stages are combined, then one transformation produces zeros in <u>all</u> the elements in one column except the first subdiagonal element (the pivotal element), and the inverse transformation modifies only the elements in a <u>single</u> column. Thus, we shall combine the two steps in the algorithm which uses modular arithmetic. We shall now describe the transformations which produce the columns of zeros.

In order to save arithmetic, we first reduce the matrix $|A_o|_m$ to the form

(2.5)



where each diagonal block D_i is in Frobenius form except for the subdiagonal elements which are nonzero but not yet reduced to unity. The reduction of $|A_o|_m$ to this form requires at most n-1 steps of the form

(2.6) $$|A_{k+1}|_m = \left| J_k^{-1}(m) \; |A_k|_m \; J_k \right|_m ,$$

where $|J_k|_m$ is a product of elementary matrices modulo m and $|A_o|_m = \left[a_k^{(o)}\right] = A$. The columns must be annihilated from left to right in order not to destroy zeros produced by previous transformations. Each transformation changes a matrix $|A_k|_m$ into a matrix $|A_{k+1}|_m$ in which there is an additional column with zeros everywhere except at the pivotal position.

Thus, after the first transformation we have

$$|A_1|_m = \left|J_o^{-1}(m)\ |A_o|_m\ J_o\ \right|_m$$

$$= \left[\begin{array}{c|ccc} 0 & x & \cdots & x \\ x & x & \cdots & x \\ \hline 0 & & & \\ \vdots & & * & \\ 0 & & & \end{array}\right], \tag{2.7}$$

where

$$J_o^{-1}(m) = \left[\begin{array}{cc|c} 1 & \mu_{11}^{(m)} & \\ 0 & 1 & \\ \hline 0 & \mu_{31}^{(m)} & \\ \vdots & \vdots & I_{n-2} \\ 0 & \mu_{n1}^{(m)} & \end{array}\right], \tag{2.8}$$

and

$$\mu_{i1}^{(m)} = \left|-a_{i1}^{(0)} . a_{21}^{(0)-1}(m)\right|_m, \quad (i = 1,3,4,\ldots,n). \tag{2.9}$$

The case in which a pivotal element is congruent to zero is discussed below.

The (j+1)st transformation produces

$$|A_{j+1}|_m = \left|J_j^{-1}(m)\ |A_j|_m\ J_j\ \right|_m$$

$$= \left|J_j^{-1}(m)\ \ldots\ J_o^{-1}(m)\ |A_o|_m\ J_o\ \ldots\ J_j\ \right|_m$$

$$= \left[F' \left| * \right.\right], \tag{2.10}$$

where F' is an nx(j+1) submatrix with zeros everywhere except on the first subdiagonal, and where

(2.11)
$$J_j^{-1}(m)= \left[\begin{array}{c|c|c} & \mu_{1,j+1}^{(m)} & \\ I_{j+1} & \vdots & \\ & \mu_{j+1,j+1}^{(m)} & \\ \hline 0 \ \ldots \ 0 & 1 & 0 \ \ldots \ 0 \\ \hline & \mu_{j+3,j+1}^{(m)} & \\ \text{○} & \vdots & I_{n-j-2} \\ & \mu_{n,j+1}^{(m)} & \end{array}\right]$$

and

(2.12)
$$\mu_{i,j+1}^{(m)} = \left| -a_{i,j+1}^{(j)} \cdot a_{j+2,j+1}^{(j)\ -1}(m) \right|_m , \quad (i=1,\ldots,j+1,j+3,..n)$$

Finally, at the (n-1)st step, if no pivots are congruent to zero (if $\ell = 1$ in (2.5)), we have a matrix in the form (2.5)

$$|A_{n-1}|_m = \left|J_{n-2}^{-1}(m)\ |A_{n-2}|_m\ J_{n-2}\right|_m$$

$$= \left|J_{n-2}^{-1}(m) \ \ldots \ J_o^{-1}(m)\ |A_o|_m\ J_o \ \ldots \ J_{n-2}\right|_m$$

(2.13)
$$= \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & x \\ x & 0 & 0 & \ldots & 0 & x \\ 0 & x & 0 & \ldots & 0 & x \\ . & . & . & \ldots & . & . \\ 0 & 0 & 0 & \ldots & 0 & x \\ 0 & 0 & 0 & \ldots & x & x \end{bmatrix} ,$$

where

(2.14)
$$J_{n-2}^{-1}(m) = \left[\begin{array}{c|c} & \mu_{1,n-1}^{(m)} \\ I_{n-1} & \vdots \\ & \mu_{n-1,n-1}^{(m)} \\ \hline 0 \ \ldots \ 0 & 1 \end{array}\right]$$

and

$$\mu_{i,n-1}^{(m)} = \left| -a_{i,n-1}^{(n-2)} \cdot a_{n,n-1}^{(n-2)\ -1}(m) \right|_m, \quad (i=1,\ldots,n-1). \tag{2.15}$$

If a pivotal element, $a_{j+1,j}^{(j-1)}$, is zero, then a search is made among the elements a_{ij} $(i=j+2,\ldots,n)$ for a nonzero element. If one is found, say a_{kj}, then <u>rows</u> k and j+1 are interchanged. This corresponds to pre-multiplying $|A_{j-1}|_m$ by the elementary matrix $|I_{k,j+1}|_m$. Then, to complete the similarity transformation modulo m, $|A_{j-1}|_m$ must be postmultiplied by $|I_{k,j+1}|_m$. This interchanges <u>columns</u> k and j+1. We should point out that the reason the search for a nonzero pivot is <u>not</u> made among the elements $a_{ij}^{(j-1)}$ $(i=1,\ldots,j)$ or $a_{j+1,k}^{(j-1)}$ $(k=j+1,\ldots,n)$ is that pre- and post-multiplying $|A_{j-1}|_m$ by $|I_{i,j}|_m$ $(i < j+1)$ or by $|I_{j+1,k}|_m$ $(k < j)$ destroys zeros produced by previous transformations.

In case no nonzero pivot can be found, then we partition the matrix $|A_{j-1}|_m$ into blocks, as follows, and apply the algorithm to the $(n-j)x(n-j)$ submatrix H_1:

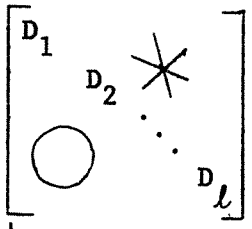
$$\begin{bmatrix} D_1 & * \\ \bigcirc & H_1 \end{bmatrix},$$

where D_1 is a jxj submatrix which is in Frobenius form except for the elements on the first subdiagonal which are not yet reduced to unity, and whose characteristic polynomial modulo m is $|p_1(\lambda)|_m$. If partitioning occurs when applying the algorithm to H_1, we obtain

$$\begin{bmatrix} D_1 & * & \\ & D_2 & \\ \bigcirc & & H_2 \end{bmatrix},$$

where the characteristic polynomial modulo m of D_2 is $|p_2(\lambda)|_m$. Proceeding in this manner, we obtain a block triangular matrix

(2.16)



which is similar modulo m to $|A|_m$, and where each D_i is in Frobenius form except for the nonunity subdiagonal elements and has characteristic polynomial modulo m $|p_i(\lambda)|_m$. Thus, the characteristic polynomial modulo m of $|A|_m$ is

$$|\det(|A|_m - \lambda I)|_m = |p_1(\lambda) \ \ldots \ p(\lambda)|_m \quad . \tag{2.17}$$

In a manner analogous to the one described in chapter I, we reduce the subdiagonal elements in the D_i to unity. If D_i has the form

$$D_i = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & y_{r_i}^{(i)} \\ d_{21}^{(i)} & 0 & 0 & \ldots & 0 & y_{r_i-1}^{(i)} \\ 0 & d_{32}^{(i)} & 0 & \ldots & 0 & y_{r_i-2}^{(i)} \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & 0 & \ldots & d_{r_i,r_i-1}^{(i)} & y_1^{(i)} \end{bmatrix} \tag{2.18}$$

then the coefficients, $x_j^{(i)}$, of the characteristic polynomial modulo m

$$|p_i(\lambda)|_m = |(-1)^{r_i}(\lambda^{r_i} - x_1^{(i)}\lambda^{r_i-1} - \ldots - x_{r_i-1}^{(i)}\lambda - x_{r_i}^{(i)})|_m \tag{2.19}$$

are given by

$$|x_1^{(i)}|_m = |y_1^{(i)}|_m \tag{2.20}$$

and

$$|x_j^{(i)}|_m = \Big|y_j^{(i)} \prod_{k=r_i-j+1}^{r_i-1} d_{k+1,k}\Big|_m , \quad j=2,\ldots,r_i. \tag{2.21}$$

This corresponds to performing a similarity transformation modulo m on D_i, producing

$$|F_i|_m = |P_i^{-1}(m) D_i P_i|_m \quad , \tag{2.22}$$

where

$$P_i^{-1}(m) = \begin{bmatrix} 1 & & & & \\ & (d_{21}^{(i)})^{-1}(m) & & & \\ & & (d_{21}^{(i)} d_{32}^{(i)})^{-1}(m) & & \\ & & & \ddots & \\ & & & & (\prod_{k=1}^{r_i - 1} d_{k+1,k}^{(i)})^{-1}(m) \end{bmatrix} . \tag{2.23}$$

Hence, after performing the transformations (2.22), $|A|_m$ is reduced to the following block triangular form in which the diagonal blocks are companion matrices for factors of the characteristic polynomial modulo m for $|A|_m$:

$$|S^{-1}(m)|A|_m S|_m = \left| \begin{bmatrix} F_1 & & & \\ & F_2 & * & \\ & & \ddots & \\ & & & F_\ell \end{bmatrix} \right|_m . \tag{2.24}$$

We note that if A is derogatory (or $|A|_m$ is derogatory modulo m) then this partitioning into a block triangular matrix must occur, by corollary (IV-4.6).

We illustrate the above method with an example.

EXAMPLE. Assume that by some scheme we have obtained the bound $\beta = 6$ for $\max_{i,j} |b_j^{(i)}|$, where

$$A = \begin{bmatrix} 3 & -1 & -4 & 2 \\ 2 & 3 & -2 & -4 \\ 2 & -1 & -3 & 2 \\ 1 & 2 & -1 & -3 \end{bmatrix} ,$$

$$\det (A - \lambda I) = p_1(\lambda) \ \ldots \ p(\lambda),$$

and

$$p_i(\lambda) = (-1)^{n_i}(\lambda^{n_i} - b_1^{(i)}\lambda^{n_i-1} - \ldots - b_{n_i}^{(i)}).$$

Then we choose m = 13. Letting

$$J_o^{-1}(13) = \begin{bmatrix} 1 & 5 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 6 & 0 & 1 \end{bmatrix},$$

the first transformation produces

$$A_1 = |J_o^{-1}(13)A_oJ_o|_{13}$$

$$= \begin{bmatrix} 0 & 4 & -1 & -5 \\ 2 & 2 & -2 & -4 \\ 0 & -2 & -1 & 6 \\ 0 & 0 & 0 & -1 \end{bmatrix}.$$

For the second transformation we choose

$$J_1^{-1}(13) = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and hence

$$A_2 = |J_1^{-1}(13)A_1J_1|_{13}$$

$$= \begin{bmatrix} 0 & 0 & -3 & -6 \\ 2 & 0 & 6 & 2 \\ 0 & -2 & 1 & 6 \\ 0 & 0 & 0 & -1 \end{bmatrix} .$$

Since the last pivotal element is 0, all that remains is to reduce the subdiagonal elements to unity. This is accomplished by the transformation

$$F = \left| P^{-1}(13)\, A_2\, P \right|_{13}$$

$$= \left| \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -6 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & -3 & -6 \\ 2 & 0 & 6 & 2 \\ 0 & -2 & 1 & 6 \\ 0 & 0 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & -4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \right|_{13}$$

$$= \left[\begin{array}{ccc|c} 0 & 0 & -1 & -6 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 5 \\ \hline 0 & 0 & 0 & -1 \end{array}\right] .$$

Hence, we have

$$\left| p_1(\lambda) \right|_{13} = \left| (-1)^3 (\lambda^3 - \lambda^2 - \lambda + 1) \right|_{13}$$

and

$$\left| p_2(\lambda) \right|_{13} = \left| -\lambda - 1 \right|_{13},$$

and by (1.6)

$$\det (A - \lambda I) = (\lambda^3 - \lambda^2 - \lambda + 1)(\lambda + 1).$$

We should point out that the Frobenius form (I-1.1) obtained using the above algorithm is not a canonical form, since similar matrices may yield different F_i. This is illustrated below.

EXAMPLE. Let m = 13 (we are assuming it is known that $13 \geq 2 \cdot \max_{i,j} |b^{(i)}|$) and

$$A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 4 & 5 \end{bmatrix} .$$

The modified Danilewski algorithm transforms A into the form

$$F = \left[\begin{array}{c|cc} 2 & 0 & 0 \\ \hline 0 & 0 & -5 \\ 0 & 1 & 6 \end{array}\right] .$$

Hence

$$\det (A-\lambda I) = (\lambda - 2)(\lambda^2 - 6\lambda + 5).$$

If we select

$$A' = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 5 \end{bmatrix} ,$$

(a matrix similar to A), we obtain

$$F' = \left[\begin{array}{c|c|c} 2 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 5 \end{array}\right] .$$

Hence

$$\det (A-\lambda I) = (\lambda-2)(\lambda-1)(\lambda-5).$$

Thus, A and A' yield different factorizations for the same characteristic polynomial.

CHAPTER VI

THE MULTIPLE MODULUS-ALGORITHM

1. Introduction. The algorithm described in chapter V uses single-modulus residue arithmetic to reduce a matrix A to Frobenius form (I-1.1). We recall that the size of the modulus m depends on the bound β (V-1.4). If 2β is too large to be representable in a computer as a single-precision integer, then m will have to be stored as a multiple-precision integer, making computations modulo m too difficult to be practical.

In order to avoid this problem, we select a set of moduli, m_1, m_2, ..., m_s, with

$$m = m_1 m_2 \cdots m_s \,, \tag{1.1}$$

because, as we shall see in section 4, this enables us to obtain results modulo m by doing most of the arithmetic modulo m_i, for $i = 1,2,\ldots,s$. More specifically, the moduli are chosen so that* $(m_i,m_j) = 1$, for $i \neq j$, and so that

$$\begin{aligned} m &\geq 2\beta \\ &\geq 2 \cdot \max_{i,j} \left| b_j^{(i)} \right|_m \,, \end{aligned} \tag{1.2}$$

where the $b_j^{(i)}$ are defined in (V-1.2).

We perform similarity transformations modulo m_i on $|A|_{m_i}$, for $i = 1,2,\ldots,s$, by using the single-modulus procedure described in chapter V in order to obtain the residue representations (see Szabó and Tanaka [1967, p. 12]) for the factors of the characteristic polynomial modulo m of A,

---

* The restriction given here is required by the Chinese Remainder Theorem, mentioned below.

$$p_i(\lambda) \sim \{|p_i(\lambda)|_{m_1}, |p_i(\lambda)|_{m_2}, \ldots, |p_i(\lambda)|_{m_s}\}. \tag{1.3}$$

From these s-tuples we can determine $|p_i(\lambda)|_m$ using the Chinese Remainder Theorem or some variation of it such as the mixed-radix conversion procedure. (See Szabó and Tanaka [1967, pp. 27, 43], Lipson [1971], and Howell and Gregory [1970].) This means that since m is chosen according to (1.2), we can determine $p_i(\lambda)$. Examples illustrating the algorithm are given in section 6.

Since different moduli may give us different factorizations, we must monitor the reductions, keeping a record of rows which are interchanged and pivots which vanish in order to use only the factorizations modulo m_i which give us the correct factorization over the integers for det $(A-\lambda I)$. This monitoring scheme and the multiple-modulus reduction are described in the next sections.

2. Block Structures in the Multiple-Modulus Algorithm. In the ideal situation all moduli used would yield the same block structure (blocks of the same order and arranged in the same pattern along the diagonal). This is not always the case, as an example illustrates.

EXAMPLE. Let

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ -5 & 2 & -1 \end{bmatrix}.$$

For m = 23, we obtain

$$A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 7 & -1 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 10 \\ 0 & 7 & 0 \end{bmatrix},$$

and

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Hence, we have obtained a 3x3 block, and

$$|\det (A-\lambda I)|_{23} = |(-1)^3(\lambda^3 - \lambda)|_{23}.$$

For m = 7, we have

$$A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$= F,$$

which gives us the factorization

$$|\det (A-\lambda I)|_7 = |(-1)^3(\lambda^2 - \lambda)(\lambda + 1)|_7.$$

Both of these are correct factorizations for the characteristic polynomial, but the factorizations are not the same for m = 7 and m = 23.

The following example illustrates that we can also obtain two different factorizations for the characteristic polynomial modulo m, one of which is <u>not</u> a correct factorization over the integers for det (A-λI).

<u>EXAMPLE</u>. Let

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -3 & 1 \\ 0 & -7 & 2 \end{bmatrix}.$$

For m = 23, we obtain

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix},$$

which gives a correct factorization over the integers

$$|\det (A-\lambda I)|_{23} = |(-1)^3(\lambda - 1)(\lambda^2 + \lambda + 1)|_{23}.$$

For m = 7, $|A|_7$ becomes immediately

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -3 & 1 \\ 0 & 0 & 2 \end{bmatrix},$$

which gives the factorization $|(-1)^3(\lambda - 1)(\lambda + 3)(\lambda - 2)|_7$ for $|\det (A-\lambda I)|_7$, and this is <u>not</u> a correct factorization over the integers for det $(A-\lambda I)$.

In order to guarantee that we can reconstruct the factors by means of the Chinese Remainder Theorem, the factors obtained using the modulus m_i must be of the same degrees as those factors obtained using m_k (for all i and k). Moreover, the factors must appear in the same order along the diagonal. This implies that even if 2 or more factors of the same degree, n_j, are obtained using the modulus m_i, they must appear in the same order as their corresponding factors of degree n_j obtained using the modulus m_k.

We now show that if we have obtained blocks of corresponding orders for two or more moduli, then these blocks can be combined using the Chinese Remainder Theorem to obtain the blocks we would have obtained had we done

our calculations modulo $m_1 m_2 \ldots m_s = m$. That is to say, if the blocks obtained using the multiple-modulus algorithm are of corresponding sizes for different moduli, then they are in the proper order for obtaining the Frobenius form modulo m using the Chinese Remainder Theorem.

We prove this by considering the transforming matrices $J_j^{(m_i)}$, where

$$|J_j^{(m_i)-1}{}_{(m_i)} A_j^{(m_i)} J_j^{(m_i)}|_{m_i} = |J_j^{(m_i)-1}{}_{(m_i)} \ldots J_o^{(m_i)-1}{}_{(m_i)} A_o^{(m_i)} J_o^{(m_i)} \ldots J_j^{(m_i)}|_{m_i}$$

$$= |J^{(m_i)-1}{}_{(m_i)} A_o^{(m_i)} J^{(m_i)}|_{m_i}$$

$$= \left[\begin{array}{c|c} F_1^{(m_i)} & * \\ \hline O & P_1^{(m_i)} \end{array}\right], \tag{2.1}$$

and $F_1^{(m_i)}$ is a (j+2)x(j+2) submatrix which is in Frobenius form except for the subdiagonal elements which are not yet reduced to unity. We must show that if

$$|J_j^{(m)-1}{}_{(m)} A_j^{(m)} J_j^{(m)}|_m = |J_j^{(m)-1}{}_{(m)} \ldots J_o^{(m)-1}{}_{(m)} A_o^{(m)} J_o^{(m)} \ldots J_j^{(m)}|_m$$

$$= |J^{(m)-1}{}_{(m)} A_o^{(m)} J^{(m)}|_m$$

$$= \left[\begin{array}{c|c} F_1^{(m)} & * \\ \hline O & P_1^{(m)} \end{array}\right], \tag{2.2}$$

where $F_1^{(m)}$ is a (j+2)x(j+2) submatrix in Frobenius form except for the nonunity subdiagonal elements, then

$$|J^{(m)}|_{m_i} = |J^{(m_i)}|_{m_i} \tag{2.3}$$

for all i, and hence

$$|F_1^{(m)}|_{m_i} = F_1^{(m_i)} \tag{2.4}$$

and

$$|P_1^{(m)}|_{m_i} = P_1^{(m_i)} . \tag{2.5}$$

The same arguments can then be applied to the submatrices $P_1^{(m_i)}$ and $P_1^{(m)}$ to show that

$$|F_2^{(m)}|_{m_i} = F_2^{(m_i)} \tag{2.6}$$

and

$$|P_2^{(m)}|_{m_i} = P_2^{(m_i)} . \tag{2.7}$$

Continuing in this manner we can show that

$$|F_k^{(m)}|_{m_i} = F_k^{(m_i)} \tag{2.8}$$

for all k. We now prove (2.4) and (2.5).

First we examine $J_k^{(m)}$ (k = 1,...,j). We shall assume for the moment that no row interchanges have taken place. We have

$$J_k^{(m)} = \left[\begin{array}{c|c|c} & -\mu_{1,k+1}^{(m)} & \\ I_{k+1} & \vdots & \bigcirc \\ & -\mu_{k+1,k+1}^{(m)} & \\ \hline 0 \ \ldots \ 0 & 1 & 0 \ \ldots \ 0 \\ \hline & -\mu_{k+3,k+1}^{(m)} & \\ \bigcirc & \vdots & I_{n-k-2} \\ & -\mu_{n,k+1}^{(m)} & \end{array}\right] , \quad (k = 1,\ldots,j) \tag{2.9}$$

where

$$-\mu_{\ell,k+1}^{(m)} = |a_{\ell,k+1}^{(k)} \cdot a_{k+2,k+1}^{(k)\ -1}(m)|_m . \tag{2.10}$$

Since, from theorem (II-2.7c) we have

$$-\mu^{(m)}_{\ell,k+1} = \left| \left| a^{(k)}_{\ell,k+1} \cdot a^{(k)\ -1}_{k+2,k+1}(m) \right|_m \right|_{m_i}$$

$$= \left| a^{(k)}_{\ell,k+1} \cdot a^{(k)\ -1}_{k+2,k+1}(m_i) \right|_{m_i}$$

$$= \left| -\mu^{(m_i)}_{\ell,k+1} \right|_{m_i} , \tag{2.11}$$

then

$$\left| J^{(m)}_k \right|_{m_i} = \left| J^{(m_i)}_k \right|_{m_i}$$

$$= J^{(m_i)}_k \ . \tag{2.12}$$

Thus,

$$\left| J^{(m)} \right|_{m_i} = \left| J^{(m)}_o J^{(m)}_1 \ \ldots \ J^{(m)}_j \right|_{m_i}$$

$$= \left| J^{(m_i)}_o \ J^{(m_i)}_1 \ \ldots \ J^{(m_i)}_j \right|_{m_i}$$

$$= \left| J^{(m_i)} \right|_{m_i} , \tag{2.13}$$

and hence

$$\left| J^{(m)\ -1}(m) \right|_{m_i} = \left| J^{(m_i)\ -1}(m_i) \right|_{m_i} . \tag{2.14}$$

Therefore,

$$\left| J^{(m)\ -1}(m) A^{(m)}_o J^{(m)} \right|_{m_i} = \left| J^{(m)\ -1}(m) |A|_m J^{(m)} \right|_{m_i}$$

$$= \left| J^{(m_i)-1}(m_i) |A|_{m_i} J^{(m_i)} \right|_{m_i}$$

$$= \left| J^{(m_i)-1}(m_i) A^{(m_i)}_o J^{(m_i)} \right|_{m_i} , \tag{2.15}$$

and thus

$$\left| F^{(m)}_1 \right|_{m_k} = F^{(m_i)} \tag{2.16}$$

and

$$|P_1^{(m)}|_{m_i} = P_1^{(m_i)} \quad . \tag{2.17}$$

Applying the same arguments to $P_k^{(m)}$ and $P_k^{(m_i)}$ $(k = 2,\ldots,j-1)$ we can easily prove (2.8).

We see from the above that applying the Chinese Remainder Theorem to the $F_k^{(m_i)}$ gives us $F_k^{(m)}$, the blocks we would have obtained had we done our arithmetic modulo m instead of modulo m_i, $i = 1,\ldots,s$. It is important to note that this analysis is based on the assumption that partitioning occurred at the same point for all moduli.

If a zero pivot occurs somewhere between columns one and j+1 which can be removed by pivoting rows, then the same rows must be pivoted for all moduli. This necessitates a monitor on the rows being pivoted during the course of reduction. If it is impossible to pivot the same rows for all moduli, then the odd modulus (or moduli) must be discarded and another tried. This assures us that even when pivoting occurs, (2.8) and (2.13) still hold. An example illustrates the necessity of monitoring the pivots.

EXAMPLE. Let

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 7 & 1 & 0 & 1 \\ 5 & 0 & 0 & 0 \end{bmatrix} .$$

If we choose m = 5, we must interchange rows 2 and 3 (and then columns 2 and 3). We obtain

$$F = \left[\begin{array}{cc|c|c} 0 & 2 & 0 & 0 \\ 1 & 0 & 3 & 3 \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \end{array}\right] .$$

Thus, we have as factors

$$|\det (A-\lambda I)|_5 = |(\lambda^2 - 2)\cdot\lambda\cdot\lambda|_5 .$$

For m = 7, we must interchange rows 2 and 4 (and columns 2 and 4). We obtain

$$F = \left[\begin{array}{ccc|c} 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 \\ \hline 0 & 0 & 0 & 0 \end{array}\right]$$

and the factorization

$$|\det (A-\lambda I)|_7 = |(\lambda^3 - 3)\cdot\lambda|_7 .$$

For all other prime moduli, we interchange rows 2 and 3 in the first step. (A search is made downward in column one for a nonzero element, and the first one encountered is selected for pivoting.) This matrix is further discussed in section 5.

3. <u>The Vanishing Pivot</u>. From previous examples we see that different moduli may yield different reductions, and hence different factorizations of the characteristic polynomial. The problem which we discuss in this section is how to select the correct factorization from a set which contains several different ones. Since the factorizations are dependent on the vanishing of pivotal elements, we approach the problem from this standpoint.

We would like to determine whether the vanishing of a pivot has

occurred because it "should" (the same pivot would vanish in the real arithmetic algorithm) or whether it has vanished as a result of our choice of modulus. The following theorem exhibits the relationship between the vanishing of pivots in the real arithmetic algorithm and in the modular arithmetic algorithm.

(3.1) THEOREM. Let $A'_k = [a'^{(k)}_{pq}]$ and $A_k = [a^{(k)}_{pq}]$ each be the kth matrix in a sequence of matrices obtained in the reduction of a matrix to Frobenius form (but not producing unity subdiagonal elements) using real arithmetic and residue arithmetic modulo m, respectively. Then, the statement $a^{(i)}_{i+2,i+1} = 0$ if and only if $a'^{(i)}_{i+2,i+1} = 0$ $(1 \leq i \leq j-1)$ implies that

$$a^{(j)}_{j+2,j+1} = 0$$

if and only if either

(a) $a'^{(j)}_{j+2,j+1} = 0$

of

(b) $\left| a'^{(j)}_{j+2,j+1} \left(a'^{(j-1)}_{j+1,j}\right)^{b_1} \left(a'^{(j-2)}_{j,j-1}\right)^{b_2} \cdots \left(a'^{(0)}_{21}\right)^{b_r} \right|_m = 0,$

where

(i) $\begin{cases} b_1 = 2 \\ b_i = 1 + 2b_{i-1} + \sum_{k=1}^{i-2} b_k \end{cases}$

(ii) the pivots $a'^{(j-1)}_{j+1,j}, \ldots, a'^{(0)}_{21}$ are the nonzero pivots obtained between step 1 and step j-1, and r is the number of such pivots.

Proof. See Appendix.

From the above theorem we see that if $a'^{(j)}_{j+2,j+1}$ is nonzero for some

modulus m_i, then the same pivot must be nonzero in the real arithmetic algorithm, provided previous pivots for the two algorithms vanished at the same point. Thus, if $a'^{(j)}_{j+2,j+1}$ vanishes for m_j $(j \neq i)$, it must be vanishing because

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \cdots (a'^{(0)}_{21})^{b_r}|_{m_j} = 0 \tag{3.2}$$

and <u>not</u> because $a'^{(j)}_{j+2,j+1} = 0$. Thus, m_j should be discarded.

This implies that if we compare the size of the initial (leading) blocks $(F_1^{(m_i)})$ obtained by reductions modulo m_1, m_2, ..., m_s, then only the moduli which have produced those blocks of maximum size should be retained, and all others should be discarded. Thus, if the remaining moduli each produced an initial block of order j+1, then either $a'^{(j)}_{j+2,j+1} = 0$ or

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \cdots (a'^{(0)}_{21})^{b_r}|_{M} = 0, \tag{3.3}$$

where M is the product of the remaining moduli. By demanding that we have at least k moduli which produce like factorizations ($k > 1$ being some input parameter dependent upon the size of the computer word and the size of the moduli used) we can make M as large as we like. The larger M is, the lesser the chance of having

$$a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \cdots (a'^{(0)}_{21})^{b_r} = C \cdot M\ , \tag{3.4}$$

where C is an integer not equal to zero. Thus, a large M will increase the probability that $a^{(j)}_{j+2,j+1} = 0$ if and only if $a'^{(j)}_{j+2,j+1} = 0$.

By comparing the sizes of blocks $2,\ldots,\ell$, in a similar manner we can eliminate "bad" choices of m_i until we are left with a set of moduli which have all produced the same block structures with blocks of maximal size. If their pivoting patterns are all the same, if the number of "successful" moduli is greater than k, and if the product of the moduli is

greater than 2β, then we can apply the results of the last section and use the Chinese Remainder Theorem to get the factors modulo m, where m is the product of the moduli.

We must emphasize that even though we have <u>at least</u> k matching reductions, and this <u>increases</u> the probability that $a^{(j)}_{j+2,j+1} = 0$ if and only $a'^{(j)}_{j+2,j+1} = 0$, we <u>can not guarantee</u> that this is the case. For example, if $k = 3$ and the m_i are approximately 10^7, then $M \doteq 10^{21}$. Then, for the method to fail to produce the correct factors, we must have $a^{(j)}_{j+2,j+1} = 0$ and

$$a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \cdots (a'^{(0)}_{21})^{b_r} = C \cdot m_1 m_2 m_3 \doteq C \cdot 10^{21}, \tag{3.5}$$

where C is a positive integer. Although it is extremely unlikely for the left side of (3.5) to be an exact integer multiple of $m_1 m_2 m_3$, a product of primes, the possibility nevertheless still exists.

4. <u>Selection of the Moduli</u>. In practice, the moduli are chosen as large prime numbers. The choice of the moduli as primes is necessary in order to guarantee the existence of inverses for integers and matrices and in order to be able to utilize the results of chapters III and IV. We recall that when m is a prime, the integers modulo m form a field. Furthermore, by choosing the m_i as primes, we guarantee that

$$(m_i, m_j) = 1, \tag{4.1}$$

for $i \neq j$, as required by the Chinese Remainder Theorem.

Ideally, the primes should be chosen as large as possible and so that $m_i m_j$ does not overflow a fixed-point computer word, for all i and j. This guarantees that an intermediate result will not overflow before it

can be reduced modulo m_i for all i. In addition to this, time can be saved by using a small number of large primes rather than a large number of small primes. Furthermore, by choosing the moduli as large prime numbers we greatly increase the probability that the disappearance of a pivot during the reduction modulo m_i has occurred because the same pivot would disappear during the real arithmetic algorithm. (See theorem (3.1).) We must further have $m_i > n$ for all i in order to be able to reconstruct the characteristic polynomial from its residue representation.

5. Calculation of a Bound for m. Let

$$\det (A-\lambda I) = (-1)^n (\lambda^n - x_1\lambda^{n-1} - \ldots - x_{n-1}\lambda - x_n)$$

$$= p_1(\lambda) \ldots p_\ell(\lambda), \tag{5.1}$$

where

$$p_i(\lambda) = (-1)^{n_i}(\lambda^{n_i} - b_1^{(i)}\lambda^{n_i-1} - \ldots - b_{n_i-1}^{(i)}\lambda - b_{n_i}^{(i)}). \tag{5.2}$$

We wish to compute a lower bound for m so that if we have a prestored set of primes, we select and use as many moduli as necessary to guarantee a solution (i.e. to guarantee that (V-1.6) holds).

If it is known that the matrix A has a characteristic polynomial which is irreducible over the integers, then $\ell = 1$. In this case we obtain a bound for $\max_j |x_j|$ by utilizing the fact that x_j is plus or minus the sum of the principal minors of order j. From Hadamard's inequality we have

$$|x_n| \leq \left(\sum_{j=1}^{n} |a_{1j}|^2 \ldots \sum_{j=1}^{n} |a_{nj}|^2\right)^{1/2}$$

$$= k. \tag{5.3}$$

Thus, any principal minor of order less than n is also bounded by k.

Since the number of principal minors of order j is equal to $\binom{n}{j}$, we have

$$|x_j| \leq \binom{n}{j} \cdot k \ . \tag{5.4}$$

Hence

$$\max_j |x_j| \leq \max_j \binom{n}{j} \cdot k$$

$$= \binom{n}{[n/2]} \cdot k \tag{5.5}$$

and we should choose m so that

$$m \geq 2 \cdot \binom{n}{[n/2]} \cdot k. \tag{5.6}$$

Therefore, in (V-1.4) we have

$$\beta = \binom{n}{[n/2]} \cdot k. \tag{5.7}$$

If it is not known that $\ell = 1$, then the bound (5.7) <u>may</u> not be sufficient. It is possible for some of the coefficients of the $p_i(\lambda)$ to be greater in absolute value than $\max_j |x_j|$. An example of such a situation is given by the characteristic polynomial

$$\det (A-\lambda I) = \lambda^9 - 3\lambda^6 + 3\lambda - 1$$

$$= (\lambda^3 - 1)^3$$

$$= (\lambda - 1)^3(\lambda^2 + \lambda + 1)^3$$

$$= (\lambda^3 - 3\lambda^2 + 3\lambda - 1)(\lambda^6 + 3\lambda^5 + 6\lambda^4 + 7\lambda^3 + 6\lambda^2 + 3\lambda + 1). \tag{5.8}$$

Thus, for this example, a bound, β, for the coefficients of the characteristic polynomial might not be large enough to guarantee that $\beta \geq 7$. (This polynomial is a special case of the polynomial $(\lambda^n - 1)^i = (\lambda - 1)(\lambda^{n-1} + \ldots + \lambda + 1)^i$ suggested by Musser [1971].)

A method for bounding the coefficients $|b_j^{(i)}|$ proposed by Collins (see Knuth [1969, p. 392]) is based on the fact that we can write

$$p_i(\lambda) = (-1)^{n_i}(\lambda^{n_i} - b_1^{(i)}\lambda - \ldots - b_{n_i-1}^{(i)}\lambda - b_{n_i}^{(i)})$$

$$= (-1)^{n_i}(\lambda - \gamma_1^{(i)}) \ldots (\lambda - \gamma_{n_i}^{(i)}), \tag{5.9}$$

where the $\gamma_j^{(i)}$ are eigenvalues of the matrix A. Thus, if we have a bound, α, for the eigenvalues of A, then

$$|b_j^{(i)}| \leq \binom{n_i}{j}\alpha^j . \tag{5.10}$$

Therefore

$$\max_{i,j} |b_j^{(i)}| \leq \max_{i,j} \binom{n_i}{j}\alpha^j$$

$$= \binom{n}{[n/2]}\alpha^n , \tag{5.11}$$

and we should choose m so that

$$m \geq 2\binom{n}{[n/2]}\alpha^n$$

$$= 2\beta. \tag{5.12}$$

Bounds for α such as $\|A\|_\infty$, $\|A\|_1$, or the bound given by Ostrowski [1952] are suitable. In practice, the bounds computed using either (5.6) or (5.12) are larger than necessary to guarantee that $|b_j^{(i)}|_m = b_j^{(i)}$.

6. Examples of the Multiple-Modulus Algorithm. Let A be the matrix used in chapter V,

$$A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 4 & 5 \end{bmatrix} .$$

We let the stored set of moduli be {7, 11, 13, 17, 19}. We shall assume that regardless of the computed bound β we require <u>at least</u> 2 moduli to give the same block structures. If we compute a bound, β, b (5.7) we have

$$\beta = \binom{3}{2} (4 \cdot 1 \cdot 41)^{1/2}$$

$$\doteq 3\ (2.8)$$

$$\doteq 38.4 \quad .$$

The bound from (3.12) with $\alpha = \|A\|_1$ is

$$\beta = \binom{3}{2} 5^3$$

$$= 375,$$

and with $\alpha = \|A\|_\infty$ it is

$$\beta = \binom{3}{2} 9^3$$

$$= 2187.$$

If α is computed from Ostrowski [1952], we have

$$\alpha = R - (1-\sigma)K,$$

where

$$R = \|A\|_\infty$$

$$= 9,$$

$$r = \min_i \sum_{j=1}^{n} |a_{ij}|$$

$$= 1,$$

$$K = \min_{i,j} |a_{ij}|$$

$$= 1,$$

and

$$\sigma = \left(\frac{r - K}{R - K}\right)^{1/2}$$

$$= 0.$$

Then

$$\alpha = 9 - 1$$

$$= 8$$

and

$$\beta = \binom{3}{2} 8^3$$

$$= 1,536.$$

Clearly, all of these bounds are larger than necessary to guarantee that $|p_i(\lambda)|_m = p_i(\lambda)$. In practice, the first bound is usually adequate, even if it is not known that $\ell = 1$.

To illustrate the multiple-modulus algorithm for the matrix A, we choose $m_1 = 7$ and $m_2 = 11$. (Note that $77 \geq 2\beta$, where β is computed by the first method.) Transforming the matrix modulo 7, we obtain

$$\left[\begin{array}{c|cc} 2 & 0 & 0 \\ \hline 0 & 0 & 2 \\ 0 & 1 & -1 \end{array}\right] .$$

Thus, the residue modulo 7 of the factors of the characteristic polynomial are

$$|p_1(\lambda)|_7 = \lambda - 2$$

and

$$|p_2(\lambda)|_7 = \lambda^2 + \lambda - 2.$$

No rows were interchanged to produce a nonzero pivot. Transforming the matrix modulo 11, we have

$$\left[\begin{array}{c|cc} 2 & 0 & 0 \\ \hline 0 & 0 & -5 \\ 0 & 1 & -5 \end{array}\right].$$

Thus,

$$|p_1(\lambda)|_{11} = \lambda - 2$$

and

$$|p_2(\lambda)|_{11} = \lambda^2 + 5\lambda + 5.$$

Again, no rows were interchanged to produce a nonzero pivot.

The residue representations for $p_1(\lambda)$ and $p_2(\lambda)$ for moduli $m_1 = 7$ and $m_2 = 11$ are, thus,

$$p_1(\lambda) \sim \{\lambda - 2, \lambda - 2\}$$

and

$$p_2(\lambda) \sim \{\lambda^2 + \lambda - 2,\ \lambda^2 + 5\lambda + 5\}.$$

Since the two moduli used yield the same block structures, and since the same pivoting strategies were used in both cases, we can apply the Chinese Remainder Theorem to the coefficients of the polynomials $|p_i(\lambda)|_{m_j}$, and obtain results modulo $m_1 m_2 = 77$,

$$|p_1(\lambda)|_{77} = \lambda - 2$$

and

$$|p_2(\lambda)|_{77} = \lambda^2 - 6\lambda + 5.$$

Hence,

$$p_1(\lambda) = \lambda - 2$$

and

$$p_2(\lambda) = \lambda^2 - 6\lambda + 5.$$

In the next example we let

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 7 & 1 & 0 & 1 \\ 5 & 0 & 0 & 0 \end{bmatrix},$$

and the stored set of moduli be {5, 7, 13, 17, 19}. Again we require that <u>at least</u> 2 moduli give the same block structures. Computing β by (5.7) we obtain

$$\beta = \binom{4}{2}(1 \cdot 51 \cdot 25)^{1/2}$$

$$\doteq 6(35.7)$$

$$= 214.2 \quad .$$

Thus, we should have

$$m \geq 428.4 \quad .$$

We saw in section 3 that transforming A modulo 5 leads to an interchange of rows 2 and 3, and we obtain

$$F = \left[\begin{array}{cc|c|c} 0 & 2 & 0 & 0 \\ 1 & 0 & 3 & 3 \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \end{array}\right] .$$

Thus, we have 3 factors

$$|\det (A-\lambda I)|_5 = |(\lambda^2 - 2)\cdot\lambda\cdot\lambda|_5$$

For $m_2 = 7$, we interchange rows 2 and 4 and obtain

$$F = \left[\begin{array}{ccc|c} 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 \\ \hline 0 & 0 & 0 & 0 \end{array}\right]$$

and the factorization

$$|\det (A-\lambda I)|_7 = |(\lambda^3 - 3)\cdot\lambda|_7.$$

For $m_3 = 13$, we interchange rows 2 and 3 in step one, and later rows 3 and 4. We obtain

$$F = \left[\begin{array}{ccc|c} 0 & 0 & 5 & -4 \\ 1 & 0 & -6 & 0 \\ 0 & 1 & 0 & -5 \\ \hline 0 & 0 & 0 & 0 \end{array}\right]$$

and hence the factorization

$$|\det (A-\lambda I)|_{13} = |(\lambda^3 + 6\lambda - 5)\cdot\lambda|_{13}.$$

At this point we note that the product of moduli used so far exceeds the bound 2β ($m_1 m_2 m_3 = 455$). If all block structures are the same at this point, and if all pivoting strategies are the same, we can apply the Chinese Remainder Theorem to the factors obtained. However, this is not the case in this example. We can immediately discard m_1 since it produced an initial block which is smaller than the one obtained using m_2 and m_3. The remaining two moduli yield identical block structures, but different pivoting strategies. It is not apparent at this point which is the correct one. Hence we must try other moduli. Since the bound 2β indicates that at least 3 moduli will have to yield identical reductions (blocks of corresponding orders and the same pivoting strategies) we will have to try at least 2 more primes.

For $m_4 = 17$, we obtain

$$F = \left[\begin{array}{ccc|c} 0 & 0 & 5 & 2 \\ 1 & 0 & 7 & 0 \\ 0 & 1 & 0 & 7 \\ \hline 0 & 0 & 0 & 0 \end{array}\right]$$

and the factorization

$$|\det (A-\lambda I)|_{17} = |(\lambda^3 - 7\lambda - 5)\cdot\lambda|_{17}.$$

The rows interchanged are 2 and 4 and rows 3 and 4.

For $m_5 = 19$, we obtain

$$F = \left[\begin{array}{ccc|c} 0 & 0 & 5 & -9 \\ 1 & 0 & 7 & 0 \\ 0 & 1 & 0 & 4 \\ \hline 0 & 0 & 0 & 0 \end{array}\right]$$

and the factorization

$$|\det (A-\lambda I)|_{19} = |(\lambda^3 - 7\lambda - 5)\cdot\lambda|_{19}.$$

The rows interchanged are rows 2 and 4 and rows 3 and 4.

We compare the results obtained using m_2, m_3, m_4. and m_5. The block structures are all the same. For m_3, m_4, and m_5 the pivoting strategies are are the same. Since $m_3m_4m_5 \geq 2\beta$ we can use the Chinese Remainder Theorem to obtain the coefficients of the factors modulo $m_3m_4m_5 = 4199$. We thus obtain

$$|\det (A-\lambda I)|_{4199} = |(\lambda^3 - 7\lambda - 5)\cdot\lambda|_{4199},$$

and hence

$$\det (A-\lambda I) = (\lambda^3 - 7\lambda - 5)\cdot\lambda .$$

In a computer program it is more efficient to use the largest stored primes first, since this may decrease the number of primes which must be used to guarantee that $m \geq 2\beta$.

CHAPTER VII

NUMERICAL RESULTS

A program for reducing a matrix to Frobenius form and obtaining a factorization of its characteristic polynomial by the method described in this paper was written in FORTRAN for the CDC 6600 at the University of Texas at Austin. The set of stored primes used are as follows [Lehmer, 1914]:

10,000,019
10,000,079
10,000,103
10,000,121
10,000,139
10,000,141
10,000,169
10,000,189
10,000,223
10,000,229.

The bound β was computed using (VI-3.7). In each of the following examples we exhibit a matrix and the factorization of its characteristic polynomial obtained using the program. We required at least 3 like reductions regardless of the size of β.

EXAMPLE 1. [Slotnick, 1963, p. 4-43]

$$A = \begin{bmatrix} 3 & -1 & -4 & 2 \\ 2 & 3 & -2 & -4 \\ 2 & -1 & -3 & 2 \\ 1 & 2 & -1 & -3 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = 1$ $\lambda_3 = 1$

$\lambda_2 = -1$ $\lambda_4 = -1$

β computed by program: 3.10×10^3

Number of moduli used: 3

Factorization of det $(A-\lambda I)$ from program:

$$\det (A-\lambda I) = (\lambda^3 - \lambda^2 - \lambda + 1)(\lambda + 1)$$

EXAMPLE 2. [Varah, 1967, pp. 103, 207] [Gregory and Karney, 1969, p. 108]

$$A = \begin{bmatrix} -9 & 21 & -15 & 4 & 2 & 0 \\ -10 & 21 & -14 & 4 & 2 & 0 \\ -8 & 16 & -11 & 4 & 2 & 0 \\ -6 & 12 & -9 & 3 & 3 & 0 \\ -4 & 8 & -6 & 0 & 5 & 0 \\ -2 & 4 & -3 & 0 & 1 & 3 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = 2 + i$ $\quad \lambda_4 = 1$

$\lambda_2 = 2 - i$ $\quad \lambda_5 = 3$

$\lambda_3 = 1$ $\quad \lambda_6 = 3$

β computed by program : 4.05×10^8

Number of moduli used: 3

Factorization of det $(A-\lambda I)$ from program:

$$\det (A-\lambda I) = (\lambda^2 - 4\lambda + 5)(\lambda^2 - 2\lambda + 1)(\lambda - 3)(\lambda - 3)$$

EXAMPLE 3. [Eberlein, 1962] [Gregory and Karney, 1969, p. 90]

$$A = \begin{bmatrix} 15 & 11 & 6 & -9 & -15 \\ 1 & 3 & 9 & -3 & -8 \\ 7 & 6 & 6 & -3 & -11 \\ 7 & 7 & 5 & -3 & -11 \\ 17 & 12 & 5 & -10 & -16 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = 1.5 + \sqrt{12.75}\ i$

$\lambda_2 = 1.5 + \sqrt{12.75}\ i$

$\lambda_3 = 1.5 - \sqrt{12.75}\ i$

$\lambda_4 = 1.5 - \sqrt{12.75}\ i$

$\lambda_5 = -1$

β computed by program: 2.41×10^7

Number of moduli used: 3

Factorization* of det $(A-\lambda I)$ from program:

$$\det (A-\lambda I) = (-1)^5(\lambda^5 - 5\lambda^4 + 33\lambda^3 - 51\lambda^2 + 135\lambda + 225)$$

EXAMPLE 4. [Varah, 1967, pp. 211-212] [Gregory and Karney, 1969, p. 110]

$$A = \begin{bmatrix} 1 & 1 & 1 & -2 & 1 & -1 & 2 & -2 & 4 & -3 \\ -1 & 2 & 3 & -4 & 2 & -2 & 4 & -4 & 8 & -6 \\ -1 & 0 & 5 & -5 & 3 & -3 & 6 & -6 & 12 & -9 \\ -1 & 0 & 3 & -4 & 4 & -4 & 8 & -8 & 16 & -12 \\ -1 & 0 & 3 & -6 & 5 & -4 & 10 & -10 & 20 & -15 \\ -1 & 0 & 3 & -6 & 2 & -2 & 12 & -12 & 24 & -18 \\ -1 & 0 & 3 & -6 & 2 & -5 & 15 & -13 & 28 & -21 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -11 & 32 & -24 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -14 & 37 & -26 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -14 & 36 & -25 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = 2$ $\lambda_4 = 2$ $\lambda_7 = 3$ $\lambda_{10} = 1$

$\lambda_2 = 2$ $\lambda_5 = 3$ $\lambda_8 = 3$

$\lambda_3 = 2$ $\lambda_6 = 3$ $\lambda_9 = 2$

---

* We note that this is a case in which the characteristic polynomial is factorable over the integers, $(-1)^5(\lambda + 1)(\lambda^4 - 6\lambda^3 + 39\lambda^2 - 90\lambda + 225)$, but the program finds only one factor, that factor being the characteristic polynomial itself.

β computed by program: 4.56×10^{16}

Number of moduli used: 3

Factorization of det $(A-\lambda I)$ from program:

$$\det (A-\lambda I) = (\lambda^2 - 4\lambda + 4)(\lambda^2 - 4\lambda + 4)(\lambda^2 - 6\lambda + 9)(\lambda^2 - 6\lambda + 9)(\lambda - 2)(\lambda - 1)$$

EXAMPLE 5. [Gregory and Karney, 1969, p. 7]

$$A = \begin{bmatrix} B & C \\ C & B \end{bmatrix}$$

where

$$B = \begin{bmatrix} -364,270 & 0 & 0 & \bigcirc \\ 1 & -364,270 & 0 & \\ 0 & 1 & -364,270 & \\ \hline \bigcirc & & -918,326 & 0 \\ & & 1 & -918,326 \end{bmatrix}$$

and

$$C = \begin{bmatrix} -694,488 & 0 & 0 & \bigcirc \\ 0 & 694,488 & 0 & \\ 0 & 0 & 694,488 & \\ \hline \bigcirc & & 965,197 & 0 \\ & & 0 & 965,197 \end{bmatrix}$$

Eigenvalues:

$\lambda_1 = 330,218$	$\lambda_6 = -1,058,758$
$\lambda_2 = 330,218$	$\lambda_7 = -1,058,758$
$\lambda_3 = 330,218$	$\lambda_8 = -1,058,758$
$\lambda_4 = 46,871$	$\lambda_9 = -1,883,523$
$\lambda_5 = 46,871$	$\lambda_{10} = -1,883,523$

β computed by program: 1.79×10^{62}

Number of moduli used: 9

Factorization of det (A-λI) from program:

$$(\lambda^2 + 1{,}836{,}652\lambda - 88{,}282{,}606{,}533)\cdot(\lambda^2 + 1{,}836{,}652\lambda - 88{,}282{,}606{,}533)\cdot$$
$$\cdot(\lambda^6 + 2{,}185{,}620\lambda^5 + 543{,}448{,}747{,}068\lambda^4 - 1{,}141{,}589{,}515{,}081{,}478{,}560\lambda^3$$
$$- 1{,}901{,}066{,}815{,}376{,}621{,}816{,}592\lambda^2$$
$$+ 267{,}158{,}841{,}389{,}405{,}409{,}701{,}792{,}512{,}320\lambda$$
$$- 42{,}735{,}849{,}656{,}157{,}591{,}523{,}087{,}007{,}405{,}518{,}784)$$

EXAMPLE 6. [Gregory and Karney, 1969, p.7]

$$A = \left[\begin{array}{c|c} B & C \\ \hline C & B \end{array}\right]$$

where

$$B = \begin{bmatrix} -1{,}604{,}858 & 0 & 0 \\ 1 & -1{,}604{,}858 & 0 \\ 0 & 1 & -1{,}604{,}858 \end{bmatrix}$$

and

$$C = \begin{bmatrix} -8{,}314{,}154 & 0 & 0 \\ 0 & -8{,}314{,}154 & 0 \\ 0 & 0 & -8{,}314{,}154 \end{bmatrix}$$

Eigenvalues: $\lambda_1 = -9{,}919{,}012$ $\lambda_4 = 6{,}709{,}296$

$\lambda_2 = -9{,}919{,}012$ $\lambda_5 = 6{,}709{,}296$

$\lambda_3 = -9{,}919{,}012$ $\lambda_6 = 6{,}709{,}296$

β computed by program: 0.72×10^{43}

Number of moduli used: 7

Factorization of det (A-λI) from program:

$$\lambda^6 + 9{,}629{,}148\lambda^5 - 168{,}741{,}932{,}204{,}688\lambda^4 - 1{,}248{,}564{,}272{,}754{,}076{,}565{,}696\lambda^3$$

$$+ 11{,}229{,}705{,}988{,}174{,}065{,}139{,}941{,}067{,}776\lambda^2$$

$$- 42{,}646{,}029{,}020{,}938{,}523{,}316{,}320{,}811{,}418{,}632{,}192\lambda$$

$$-294{,}737{,}981{,}114{,}491{,}044{,}619{,}180{,}056{,}066{,}964{,}562{,}116{,}608$$

CHAPTER VIII

SUMMARY AND CONCLUDING REMARKS

In chapter I of this thesis is a discussion of the reduction of a matrix A to Frobenius form using rational arithmetic. This form displays the coefficients of the characteristic polynomial of A or the coefficients of a factorization of it.

Chapters II, III, and IV summarize the main theorems and definitions for residue arithmetic. Since the integers modulo a prime form a finite field, F, all the theorems relative to matrices and polynomials over a field can be applied. The most important theorems for matrices over F are those regarding the similarity transformation modulo m.

Since similar matrices have the same characteristic polynomial, we can use similarity transformations over F to reduce A to a Frobenius form over F. The Frobenius form over F is the residue of the Frobenius form obtained using rational arithmetic. Thus, if m is sufficiently large, we can use residue arithmetic modulo m to obtain the Frobenius form which we would have obtained had we used rational arithmetic.

An algorithm for reducing A to Frobenius form over F is described in chapter V. This algorithm is analogous to the one described in chapter I. The size of the modulus used depends upon a bound β (V-1.5), and since β may be quite large, it is desirable to use multiple-modulus residue arithmetic. The multiple-modulus algorithm is described in chapter VI. When using this algorithm we obtain results modulo m_1, m_2, ... , and m_s, where the m_i are large primes. Then the Chinese Remainder Theorem is used to obtain the results modulo $m = m_1 m_2 \ldots m_s$. This, in turn, yields the characteristic polynomial for A (or a factorization of it over the integers). Thus, the

size of the moduli can be limited to a size which can be stored in a single-precision computer word. Multiple-precision arithmetic usually must be used for the Chinese Remainder Theorem, however.

Care must be taken in using the multiple-modulus algorithm, since different moduli may yield different Frobenius forms, and hence different factorizations for the characteristic polynomial modulo m. An algorithm is described in chapter VI for deciding which reductions are the incorrect ones. On all reductions remaining, a check must be made to insure that their pivoting patterns are identical. The moduli yielding reductions with identical block structures and pivoting patterns are then checked to see if their product is greater than the predetermined bound, 2β, and if their number is greater than some preset constant,k. If so, then the Chinese Remainder Theorem is applied to the residue representations for the coefficients of the factors.

The number k should be some number greater than 1 such that a product of k moduli yields some "large" number. As k becomes larger, the probability becomes greater that the vanishing of a pivot for all moduli means that the same pivot would have vanished had we used rational arithmetic. (See theorem (VI-3.1).) It cannot be overemphasized that the method <u>can fail</u> if either k is too small or if not enough moduli are stored to give $m > 2\beta$. (See the comments at the end of section 3 of chapter VI.)

The bound β can be computed by several methods, most of which yield bounds which are larger than necessary. By choosing the moduli as large as possible (see chapter VI,section 4), we can reduce the amount of work to be performed even though the bound is too large.

Numerical results indicate that in some cases the characteristic polynomial is obtained in a factored form which readily yields some of the

eigenvalues. In other cases we obtain a factorization which does not yield eigenvalues (without finding the zeros of a high-degree polynomial), but it may give us some information about the structure of the matrix. For instance, it may tell us that we have multiple eigenvalues. Another type of result obtained from numerical testing is the case in which no immediate information is obtained other than the coefficients of the characteristic polynomial in unfactored form (the nonderogatory case). This is not to say that results in this case are unusable, for the characteristic polynomial itself gives us some information merely from its coefficients. The fact that all results obtained are exact is of some use.

We emphasize that the block structure obtained for a given matrix is not unique. The form obtained depends upon the order in which the elements below the first subdiagonal are annihilated. Changing the order in which the elements are annihilated may change either the order of the blocks on the diagonal or the size of the blocks. Clearly, the form obtained is not a canonical form. (See the example at the end of chapter V.)

This algorithm is not the sort of algorithm that one would normally use by itself. It certainly should not be used with the idea that it can obtain eigenvalues for matrices (although in some cases it can). It can be a powerful tool for supplementing information obtained from other programs in which approximate results are obtained. It is also a powerful tool in that it yields (exactly) the characteristic polynomial or a factorization over the integers of the characteristic polynomial. With conventional computer arithmetic only an approximation can be obtained even when we resort to multiple-precision arithmetic.

APPENDIX

Proof of Theorem (VI-3.1). We first assume, for $i = 1,2,\ldots,j-1$, that $a'^{(i)}_{i+2,i+1} \neq 0$ and $a^{(i)}_{i+2,i+1} \neq 0$. Furthermore we assume that the real arithmetic algorithm is performed with no roundoff error. We shall prove the main theorem by proving first (by induction) that

$$a'^{(j)}_{j+2,j+1}\{(a'^{(j-1)}_{j+1,j})^{b_1}(a'^{(j-2)}_{j,j-1})^{b_2}\cdots (a'^{(0)}_{21})^{b_j}\}$$

is an integer and that

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1}\cdots (a'^{(0)}_{21})^{b_j}|_m = |a^{(j)}_{j+2,j+1}(a^{(j-1)}_{j+1,j})^{b_1}\cdots (a^{(0)}_{21})^{b_j}|_m$$

for all j, $1 \leq j \leq n-2$. Then for $r = j$, the main theorem follows.

In the following we let

$$u_{i,\ell+1} = -a'_{i,\ell+1}\cdot a'^{-1}_{\ell+2,\ell+1}$$

for all i and ℓ.

Step 1. For the first step of the reduction (that is, $j = 1$) in real arithmetic we have

$$a'^{(1)}_{i2} = a'^{(0)}_{i2} + u_{i1}a'^{(0)}_{22} + \sum_{k=3}^{n}(-u_{k1})(a'^{(0)}_{ik} + u_{i1}a'^{(0)}_{2k})$$

$$= a'^{(0)}_{i2} - a'^{(0)}_{i1}a'^{(0)}_{21}{}^{-1}a'^{(0)}_{22} + \sum_{k=3}^{n} a'^{(0)}_{k1}a'^{(0)}_{21}{}^{-1}(a'^{(0)}_{ik} - a'^{(0)}_{i1}a'^{(0)}_{21}{}^{-1}a'^{(0)}_{2k}),$$

for $i = 3,\ldots,n$. Thus,

$$a'^{(1)}_{i2}(a'^{(0)}_{21})^2 = a'^{(0)}_{i2}(a'^{(0)}_{21})^2 - a'^{(0)}_{i1}a'^{(0)}_{21}a'^{(0)}_{22} + \sum_{k=3}^{n} a'^{(0)}_{k1}(a'^{(0)}_{ik}a'^{(0)}_{21} - a'^{(0)}_{i1}a'^{(0)}_{2k}).$$

Since $a'^{(0)}_{ij} = a^{(0)}_{ij}$ for all i and j, each quantity on the right hand side is an integer. Therefore $a'^{(1)}_{i2}(a'^{(0)}_{21})^2$ is an integer. Hence, we can consider its residue modulo m which is

$$|a'^{(1)}_{i2}(a'^{(0)}_{21})^2|_m = |a'^{(0)}_{i2}(a'^{(0)}_{21})^2 - a'^{(0)}_{i1}a'^{(0)}_{21}a'^{(0)}_{22}$$
$$+\sum_{k=3}^{n} a'^{(0)}_{k1}(a'^{(0)}_{ik}a'^{(0)}_{21} - a'^{(0)}_{i1}a'^{(0)}_{2k})|_m$$

$$= |a^{(0)}_{i2}(a^{(0)}_{21})^2 - a^{(0)}_{i1}a^{(0)}_{21}a^{(0)}_{22}$$
$$+ \sum_{k=3}^{n} a^{(0)}_{k1}(a^{(0)}_{ik}a^{(0)}_{21} - a^{(0)}_{i1}a^{(0)}_{2k}|_m$$

$$= |a^{(1)}_{i2}(a^{(0)}_{21})^2|_m ,$$

for $i = 3,\ldots,n$. Furthermore, for $i,t = 3,\ldots,n$,

$$a'^{(1)}_{ti} = a'^{(0)}_{ti} + u_{t1}a'^{(0)}_{2i},$$

$$= a'^{(0)}_{ti} - a'^{(0)}_{t1}a'^{(0)}_{21}{}^{-1}a'^{(0)}_{2i},$$

and so

$$a'^{(0)}_{ti}(a'^{(0)}_{21}) = a'^{(0)}_{ti}a'^{(0)}_{21} - a'^{(0)}_{t1}a'^{(0)}_{2i}.$$

Thus $a'^{(1)}_{ti}(a'^{(0)}_{21})$ is an integer. (This fact is needed for the inductive step.)

Step j-1. At step j-1 ($j < n-1$) in the real arithmetic algorithm we have, for $i = j+1,\ldots,n$,

$$a'^{(j-1)}_{i,j} = a'^{(j-2)}_{i,j} + u_{i,j-1}a'^{(j-2)}_{j,j} + \sum_{k=j+1}^{n}(-u_{k,j-1})(a'^{(j-2)}_{ik} + u_{i,j-1}a'^{(j-2)}_{jk})$$

$$= a'^{(j-2)}_{i,j} - a'^{(j-2)}_{i,j-1}a'^{(j-2)}_{j,j-1}{}^{-1}a'^{(j-2)}_{jj}$$

$$+ \sum_{k=j+1}^{n}(a'^{(j-2)}_{k,j-1}a'^{(j-2)}_{j,j-1}{}^{-1})\cdot(a'^{(j-2)}_{i,k} - a'^{(j-2)}_{i,j-1}a'^{(j-2)}_{j,j-1}{}^{-1}a'^{(j-2)}_{j,k}).$$

We assume that

$$a'^{(j-1)}_{ij}(a'^{(j-2)}_{j,j-1})^2 a'^{(j-3)}_{j-1,j-2})^5 \cdots (a'^{(0)}_{21})^{b_{j-1}}$$

is an integer, and that

$$|a'^{(j-1)}_{ij}(a'^{(j-2)}_{j,j-1})^2(a'^{(j-3)}_{j-1,j-2})^5 \cdots (a'^{(0)}_{21})^{b_{j-1}}|_m$$

$$= |a^{(j-1)}_{ij}(a^{(j-2)}_{j,j-1})^2(a^{(j-3)}_{j-1,j-2})^5 \cdots (a^{(0)}_{21})^{b_{j-1}}|_m.$$

Furthermore, we assume that, for $t,i = j+1,\ldots,n$,

$$a'^{(j-1)}_{ti}(a'^{(j-2)}_{j,j-1})(a'^{(j-3)}_{j-1,j-2})^3 \cdots (a'^{(0)}_{21})^{c_{j-1}}$$

is an integer, where

$$\begin{cases} c_1 = 1 \\ c_i = b_{i-1} + c_{i-1}, \end{cases}$$

and that

$$|a'^{(j-1)}_{ti}(a'^{(j-2)}_{j,j-1})(a'^{(j-3)}_{j-1,j-2})^3 \cdots (a'^{(0)}_{21})^{c_{j-1}}|_m$$

$$= |a^{(j-1)}_{ti}(a^{(j-2)}_{j,j-1}(a^{(j-3)}_{j-1,j-2})^3 \cdots (a^{(0)}_{21})^{c_{j-1}}|_m .$$

This completes the inductive bypothesis.

Step j. In the real arithmetic algorithm we have

$$a'^{(j)}_{j+2,j+1} = a'^{(j-1)}_{j+2,j+1} + u_{j+2,j}a'^{(j-1)}_{j+1,j+1} + \sum_{k=j+1}^{n}(-u_{kj})(a'^{(j-1)}_{j+2,k} + u_{j+2,j}a'^{(j-1)}_{j+1,k})$$

$$= a'^{(j-1)}_{j+2,j+1} - a'^{(j-1)}_{j+2,j}\, a'^{(j-1)-1}_{j+1,j}\, a'^{(j-1)}_{j+1,j+1}$$

$$+ \sum_{k=j+1}^{n} (a'^{(j-1)}_{kj}\, a'^{(j-1)-1}_{j+1,j}) \cdot (a'^{(j-1)}_{j+2,k} - a'^{(j-1)}_{j+2,j}\, a'^{(j-1)-1}_{j+1,j}\, a'^{(j-1)}_{j+1,k}).$$

Then

$$a'^{(j)}_{j+2,j+1} (a'^{(j-1)}_{j+1,j})^2 (a'^{(j-2)}_{j,j-1})^5 \cdots (a'^{(0)}_{21})^{b_j}$$

$$= a'^{(j-1)}_{j+2,j+1} (a'^{(j-1)}_{j+1,j})^2 (a'^{(j-2)}_{j,j-1})^5 \cdots (a'^{(0)}_{21})^{b_j}$$

$$- a'^{(j-1)}_{j+2,j}\, a'^{(j-1)}_{j+1,j}\, a'^{(j-1)}_{j+1,j+1} (a'^{(j-2)}_{j,j-1})^5 \cdots (a'^{(1)}_{32})^{b_{j-1}} (a'^{(0)}_{21})^{b_j}$$

$$+ \sum_{k=j+1}^{n} a'^{(j-1)}_{kj} (a'^{(j-1)}_{j+2,k}\, a'^{(j-1)}_{j+1,j} - a'^{(j-1)}_{j+2,j}\, a'^{(j-1)}_{j+1,k}) (a'^{(j-2)}_{j,j-1}) \cdots (a'^{(0)}_{21})^{b_j}$$

$$= \{a'^{(j-1)}_{j+2,j+1} (a'^{(j-2)}_{j,j-1}) (a'^{(j-3)}_{j-1,j-2})^3 \cdots (a'^{(0)}_{21})^{c_{j-1}}\} \{(a'^{(j-1)}_{j+1,j})^2 (a'^{(j-2)}_{j,j-1})^4 \cdots$$

$$\cdots (a'^{(0)}_{21})^{b_j - c_j}\}$$

$$- \{a'^{(j-1)}_{j+2,j} (a'^{(j-2)}_{j,j-1})^2 (a'^{(j-3)}_{j-1,j-2})^5 \cdots (a'^{(0)}_{21})^{b_{j-1}}\} \{a'^{(j-1)}_{j+1,j} (a'^{(j-2)}_{j,j-1})^2 \cdot$$

$$\cdot (a'^{(j-3)}_{j-1,j-2})^5 \cdots (a'^{(0)}_{21})^{b_{j-1}}\}$$

$$\cdot \{a'^{(j-1)}_{j+1,j+1} (a'^{(j-2)}_{j,j-1}) (a'^{(j-3)}_{j-1,j-2})^3 \cdots (a'^{(0)}_{21})^{c_{j-1}}\}$$

$$+ \sum_{k=j+1}^{n} \{a'^{(j-1)}_{kj} (a'^{(j-2)}_{j,j-1})^2 (a'^{(j-3)}_{j-1,j-2})^5 \cdots (a'^{(0)}_{21})^{b_{j-1}}\}$$

$$\cdot (\{a'^{(j-1)}_{j+2,k} (a'^{(j-2)}_{j,j-1}) (a'^{(j-3)}_{j-1,j-2})^3 \cdots (a'^{(0)}_{21})^{c_{j-1}}\} \{a'^{(j-1)}_{j+1,j} (a'^{(j-2)}_{j,j-1})^2 (a'^{(j-3)}_{j-1,j-2})^5$$

$$\cdots (a'^{(0)}_{21})^{b_{j-1}}\}$$

$$- \{a'^{(j-1)}_{j+2,j}(a'^{(j-2)}_{j,j-1})^2(a'^{(j-3)}_{j-1,j-2})^5 \ldots (a'^{(0)}_{21})^{b_{j-1}}\}\{a'^{(j-1)}_{j+1,k}(a'^{(j-2)}_{j,j-1})(a'^{(j-3)}_{j-1,j-2})^3 \ldots (a'^{(0)}_{21})^{c_{j-1}}\}).$$

Since each product contained in braces is an integer (by the inductive hyposhesis), the entire quantity must be an integer. By the previous step, the, we have

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^2(a'^{(j-2)}_{j,j-1})^5 \ldots (a'^{(0)}_{21})^{b_j}|_m$$

$$= |a^{(j)}_{j+2,j+1}\,(a^{(j-1)}_{j+1,j})^2(a^{(j-2)}_{j,j-1})^5 \ \ldots\ (a^{(0)}_{21})^{b_j}|_m\ .$$

Also, for $t,i = j+2,\ldots,n$,

$$a'^{(j)}_{ti}(a'^{(j-1)}_{j+1,j})(a'^{(j-2)}_{j,j-1})^3 \ \ldots\ (a'^{(0)}_{21})^{c_j}$$

$$= (a'^{(j-1)}_{ti} + a'^{(j-1)}_{tj}a'^{(j-1)}_{j+1,j}a'^{(j-1)}_{j+1,i})(a'^{(j-1)}_{j+1,j})(a'^{(j-2)}_{j,j-1})^3 \ \ldots\ (a'^{(0)}_{21})^{c_j}$$

$$= (a'^{(j-1)}_{ti}a'^{(j-1)}_{j+1,j} + a'^{(j-1)}_{tj}a'^{(j-1)}_{j+1,i})(a'^{(j-2)}_{j,j-1})^3 \ \ldots\ (a'^{(0)}_{21})^{c_j}$$

$$= \{a'^{(j-1)}_{ti}(a'^{(j-2)}_{j,j-1})(a'^{(j-3)}_{j-1,j-2})^3 \ldots (a'^{(0)}_{21})^{c_{j-1}}\}\{a'^{(j-1)}_{j+1,j}(a'^{(j-2)}_{j,j-1})^2(a'^{(j-3)}_{j-1,j-2})^5 \ldots (a'^{(0)}_{21})^{c_j - c_{j-1}}\}$$

$$+ \{a'^{(j-1)}_{tj}(a'^{(j-2)}_{j,j-1})^2(a'^{(j-3)}_{j-1,j-2})^5 \ldots (a'^{(0)}_{21})^{b_{j-1}}\}\{a'^{(j-1)}_{j+1,i}(a'^{(j-2)}_{j,j-1}) \cdot (a'^{(j-3)}_{j-1,j-2})^3 \ldots (a'^{(0)}_{21})^{c_{j-1}}\}.$$

Since each product in braces is an integer, the entire quantity must be an integer. Thus, from step j-1, we have

$$|a'^{(j)}_{ti}(a'^{(j-1)}_{j+1,j})(a'^{(j-2)}_{j,j-1})^3 \ \ldots\ (a'^{(0)}_{21})^{c_j}|_m$$

$$= |a_{ti}^{(j)}(a_{j+1,j}^{(j-1)})(a_{j,j-1}^{(j-2)})^3 \ \ldots \ (a_{21}^{(0)})^{c_j}|_m \ .$$

This completes the induction. Thus, for all j, $1 \leq j \leq n-2$, we have

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a'^{(0)}_{21})^{b_j}|_m = |a^{(j)}_{j+2,j+1}(a^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a^{(0)}_{21})^{b_j}|_m .$$

We recall that we assumed that no previous pivots for either algorithm had vanished. In order to include the case in which K pivots have vanished at the same place in both algorithms (in other words $a^{(i)}_{i+2,i+1} = 0$ if and only if $a'^{(i)}_{i+2,i+1} = 0$), we note that whenever a nonzero pivot cannot be found, then no divisions take place on or below the first subdiagonal for that step of the algorithm. Thus, this eliminates K factors form the product of pivots used in scaling. Then, for r = j-K, the scaling factor is

$$(a'^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a'^{(0)}_{21})^{b_r} .$$

From the above proof by induction we see that if either $a'^{(j)}_{j+2,j+1}=0$ or $|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a'^{(0)}_{21})^{b_r}|_m=0$, then

$$|a^{(j)}_{j+2,j+1}(a^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a^{(0)}_{21})^{b_r}|_m = 0.$$

Since none of $(a^{(j-1)}_{j+1,j})^{b_1}, \ldots, (a^{(0)}_{21})^{b_r}$ is zero, we must have

$$a^{(j)}_{j+2,j+1} = 0.$$

Conversely, if $a^{(j)}_{j+2,j+1} = 0$, then either

$$a'^{(j)}_{j+2,j+1} = 0$$

or

$$|a'^{(j)}_{j+2,j+1}(a'^{(j-1)}_{j+1,j})^{b_1} \ \ldots \ (a'^{(0)}_{21})^{b_r}|_m = 0.$$

This completes the proof of the theorem.

BIBLIOGRAPHY

Chartres, B. A. [1964], "Controlled Precision Calculations and the Danilewski Method", Brown University, Division of Applied Mathematics Report.

Danilewski, A. [1937], "O Čislennom Resĕnii Vekovogo Uravnenija", Mat. Sbornik 2, pp. 169-171.

Eames, W. P. [1968], The Elementary Theory of Numbers, Polynomials, and Rational Functions, American Elsevier, New York.

Eberlein, P. J. [1962], "A Jacobi-like Method for the Automatic Computation of Eigenvalues and Eigenvectors", Jour. SIAM 10, pp. 74-78.

Eves, H. [1966], Elementary Matrix Theory, Allyn and Bacon, Boston.

Frank, W. L. [1958], "Computing Eigenvalues of Complex Matrices by Determinant Evaluation and by Methods of Danilewski and Wielandt", Jour. SIAM 6 pp. 378-392.

Gantmacher, F. R. [1960], The Theory of Matrices, Volume I, Chelsea, New York.

Gregory, R. T. and Karney, D. [1969], A Collection of Matrices for Testing Computational Algorithms, Wiley, New York.

Griffin, H. [1954], Elementary Theory of Numbers, McGraw-Hill, New York.

Hansen, E. R. [1963], "On the Danilewski Method", Journal of the A.C.M. 10, pp. 102-109.

Hohn, F. E. [1964], Elementary Matrix Algebra, Macmillan, New York.

Householder, A. S. [1964], The Theory of Matrices in Numerical Analysis, Blaisdell, New York.

Householder, A. S., and Bauer, F. L. [1959], "On Certain Methods for Expanding the Characteristic Polynomial", Num. Math. 1, pp. 29-37.

Howell, J. A. [1969], "Solving Systems of Linear Algebraic Equations Using Residue Arithmetic", Unpublished Masters Thesis, The University of Texas at Austin.

Howell, J. A., and Gregory, R. T. [1969a], "An Algorithm for Solving Linear Algebraic Equations Using Residue Arithmetic I", BIT 9, pp. 200-224.

Howell, J. A., and Gregory, R. T. [1969b], "An Algorithm for Solving Linear Algebraic Equations Using Residue Arithmetic II, BIT 9, pp. 324-327.

Howell, J. A., and Gregory, R. T. [1969c], "Solving Systems of Linear Algebraic Equations Using Residue Arithmetic", The University of Texas at Austin Computation Center Report TNN-82 (Revised).

Howell, J. A., and Gregory, R. T. [1970], "Solving Linear Equations Using Residue Arithmetic-Algorithm II", BIT 10, pp. 23-37.

Knuth, D. E. [1969], The Art of Computer Programming, Volume II, Addison-Wesley, Reading.

Lehmer, D. N. [1914], List of Prime Numbers from 1 to 10,006,721, Carnegie Institute of Washington (165), Washington, D.C.

Lipson, J. D. [1971], "Chinese Remainder and Interpolation Algorithms", Proceedings of the Second Symposium on Symbolic and Algebraic Manipulation, SIGSAM-ACM.

Musser, D. [1971], Private communication, The University of Texas at Austin.

Nagell, T. [1964], Introduction to Number Theory, Chelsea, New York.

Niven, I., and Zuckerman, H. S. [1966], An Introduction to the Theory of Numbers, Wiley, New York.

Ostrowski, A. [1952], "Bounds for the Greatest Latent Root of a Positive Matrix", Jour. of the London Math. Soc. 27, pp. 253-256.

Perlis, S. [1952], Theory of Matrices, Addison-Wesley, Cambridge.

Slotnick, D. L. [1963], "Modular Arithmetic Computing Techniques", Westinghouse Electric Corporation, Technical Report ASD-TDR-63-280, Baltimore.

Szabó, N. S., and Tanaka, R. I. [1967], Residue Arithmetic and Its Applications to Computer Technology, McGraw-Hill, New York.

Varah, J. [1967], "The Computation of Bounds for the Invariant Subspaces of a General Matrix Operator", Stanford University, Computer Science Department Report-CS66.

Wayland, H. [1945], "Expansion of Determinantal Equations into Polynomial Form", Quart. of Appl. Math. 2, pp. 277-306.

Wilkinson, J. H. [1965], The Algebraic Eigenvalue Problem, Clarendon Press, Oxford.

VITA

Jo Ann Shaw Howell was born in Bonham, Texas on September 28, 1945, daughter of Lela Mae Turner Shaw and Joseph Frederick Shaw. After graduating from Bonham High School in June, 1964, as valedictorian, she entered the University of Texas at Austin. In June, 1967, she received the degree of Bachelor of Arts with honors from the University of Texas at Austin, with a major in mathematics and minor in chemistry. On June 10, 1967, she married Robert Benton Howell, a graduate student in plasma physics from Midland, Texas. During the summer of 1967, they resided in Dayton, Ohio, where she worked as a computer programmer for Data Corporation of Dayton.

Returning to Austin in September, 1967, she entered the Graduate School at the University of Texas at Austin, majoring in computer sciences. Until she received the degree of Master of Arts in January, 1969, she was employed as a research scientist assistant at Applied Research Laboratories of Austin. In January, 1969, she became a teaching assistant in the Department of Computer Sciences of the University of Texas at Austin.

The work done on her masters' thesis resulted in the publication of four papers. The first three were published in BIT ("An Algorithm for Solving Linear Algebraic Equations Using Residue Arithmetic", I and II, volume 9 (1969), pp. 200-224 and pp. 324-327, and "Solving Linear Equations Using Residue Arithmetic - Algorithm II", volume 10, pp. 23-37), and the fourth was published in Communications of the ACM ("Exact Solution of Linear Equations Using Residue Arithmetic", volume 14 (1971), pp. 180-184.

From June, 1969, until August, 1970, she was employed as a computer programmer for the Computation Center at the University of Texas at Austin. When the Center for Numerical Analysis was formed from Computation Center personnel, she became a research assistant with the Center.

Permanent address: c/o 420 Meadow Lane Road

Bonham, Texas 75418

This dissertation was typed by Jo Ann Shaw Howell.