

On Solving Systems of Equations Using Interval Arithmetic

By Eldon R. Hansen

Introduction. In this paper, we consider the problem of applying interval arithmetic to bound a solution of a system of nonlinear equations. Moore [1, Section 7.3] has discussed the same problem. His approach, as well as ours, is to extend the multidimensional Newton method and implement it in interval arithmetic.

In Section 2, it is shown that a particular detail of Moore's method can be modified to improve convergence and yield sharper bounds. In extreme cases, the modification can yield convergence where the original method fails.

To illustrate this procedure, we consider (in Section 3) the problem of bounding complex roots of polynomials. Previous literature on the use of interval arithmetic to bound polynomial roots was restricted to the case of real polynomials with real roots. We use the obvious expedient of separating a polynomial equation into real and imaginary parts. This yields two real equations in two real variables to be solved by the method of Section 2.

In Section 4, we consider the matrix eigenvalue-vector problem. Bounds for the solution of this problem are obtained by a method which is essentially that of Section 2. We show that our method is directly related to Wielandt inverse iteration.

1. First Formulation. Let $f_i(x)$ ($i = 1, 2, \dots, n$) be a real rational function of a real vector x . Let $f(x)$ denote the vector with components $f_i(x)$ ($i = 1, 2, \dots, n$). Assume

$$(1.1) \quad f(y) = 0;$$

that is, y is a desired solution vector. To obtain the method as described by Moore, we expand $f(x)$ in a Taylor series with remainder about the point y and obtain

$$(1.2) \quad f(x) = f(y) + \sum_{i=1}^n (x_i - y_i) \frac{\partial}{\partial x_i} f[y + \theta_i(x - y)]$$

where $\theta_i \in [0, 1]$. Since $f(y) = 0$, (1.2) becomes

$$(1.3) \quad J(x - y) = f(x)$$

where J is the Jacobian matrix with elements

$$(1.4) \quad J_{ji} = \frac{\partial}{\partial x_i} f_j[y + \theta_i(x - y)].$$

As in [3], we define an interval vector (matrix) to be a vector (matrix) whose elements are interval numbers.

Let $X^{(0)}$ be an interval vector containing both x and y . Then

$$(1.5) \quad y + \theta_i(x - y) \in X^{(0)}$$

since $\theta_i \in [0, 1]$. Let $J(X^{(0)})$ denote the interval matrix obtained from J by replacing $y + \theta_i(x - y)$ by $X^{(0)}$ for all i . We assume $J(X^{(0)})$ does not contain a singular matrix. Let $V(X^{(0)})$ be an interval matrix containing the inverse of $J(X^{(0)})$. (For a discussion of the inverse of an interval matrix, see [3], [9], or Chapter 5 of [1].) Then $J^{-1} \in V(X^{(0)})$ for all $\theta_i \in [0, 1]$ and hence the solution vector y is contained in

$$(1.6) \quad Y^{(0)} = x - V(X^{(0)})f(x).$$

Define $X^{(1)} = X^{(0)} \cap Y^{(0)}$. Note $X^{(1)}$ contains the solution Y since $X^{(0)}$ and $Y^{(0)}$ do. We next replace x by the midpoint of $X^{(1)}$, replace $X^{(0)}$ by $X^{(1)}$ and repeat the procedure. In this way, we obtain a sequence of interval vectors $X^{(i)}$ ($i = 0, 1, 2, \dots$) each containing the solution vector y . Under appropriate conditions, these interval vectors converge to the solution y .

This is the method proposed by Moore. Note that every element of J may be an interval.

2. Alternate Formulation. We now describe how certain elements of J can be replaced by real numbers while the remaining elements remain unchanged. Such a change improves accuracy, improves convergence, and succeeds in some extreme instances where Moore's procedure fails.

Consider a scalar function $g(x)$ of a vector x . We can get an equation similar to (1.2) for $g(x)$ by expanding $g[y + \theta(x - y)]$ as a function of the single variable θ . Alternately, we may proceed sequentially as follows. Considering $g(x)$ as a function of the single variable x_1 , we obtain

$$(2.1) \quad g(x) = g(y_1, x_2, \dots, x_n) + (x_1 - y_1) \frac{\partial}{\partial x_1} g[y_1 + \theta_1(x_1 - y_1), x_2, \dots, x_n].$$

Next, consider $g(y_1, x_2, \dots, x_n)$ as a function of the single variable x_2 and obtain

$$(2.2) \quad g(y_1, x_2, \dots, x_n) = g(y_1, y_2, x_3, \dots, x_n) + (x_2 - y_2) \frac{\partial}{\partial x_2} g[y_1, y_2 + \theta_2(x_2 - y_2), x_3, \dots, x_n].$$

Next, expand $g(y_1, y_2, x_3, \dots, x_n)$ as a function of x_3 , and so forth. Finally, we expand $g(y_1, \dots, y_{n-1}, x_n)$ as a function of x_n . For each of these expansions, $\theta_i \in [0, 1]$ ($i = 1, 2, \dots, n$).

Combining these results,

$$(2.3) \quad g(x) = g(y) + \sum_{i=1}^n (x_i - y_i) \frac{\partial g}{\partial x_i},$$

where the argument of $\partial g / \partial x_i$ is $y_1, \dots, y_{i-1}, y_i + \theta_i(x_i - y_i), x_{i+1}, \dots, x_n$. Let g denote one of the components f_j in the above problem. We see that, while the arguments $y_1, \dots, y_{i-1}, y_i + \theta_i(x_i - y_i)$ are unknown, the remaining arguments x_{i+1}, \dots, x_n are known. Hence, only the first i arguments of $\partial f_j / \partial x_i$ need be replaced by the intervals X_1, \dots, X_i , respectively. The last $n - i$ elements remain scalars. Let J^i denote the matrix with elements

$$(\partial/\partial x_i)f_j(X_1, \dots, X_i, x_{i+1}, \dots, x_n).$$

Sharper results will be obtained, in general, using J^I in (1.3) rather than $J(X^{(0)})$. At any rate, the results cannot be worse. If J^I is used, then, as before, each derived vector $X^{(i)}$ contains the solution vector y .

A more useful expansion would be one in which even more of the arguments of the first partial derivatives are x_i rather than y_i or $y_i + \theta_i(x_i - y_i)$. The author knows of no such expansion.

This sequential derivation could be modified by changing the order in which the expansions were performed. In order to consider which is the most desirable order, consider the following:

Let J^* be the matrix with elements given by (1.4) with $\theta_i = 1$ ($i = 1, 2, \dots, n$). Using this matrix, Eq. (1.3) becomes $J^*(x - y^*) = f(x)$. If the solution y^* of this equation varies rapidly (in some sense) as x_i changes in row k of J^* and varies slowly as x_j changes in row k , then we prefer to use arguments x_i and X_j in row k of the interval matrix J^I rather than arguments X_i and x_j . The former choice yields a sharper interval solution Y . In the absence of such knowledge about y^* , we may use the natural order as given above. Thus, for example, if $n = 3$, the i th row of the Jacobian matrix J would become

$$(2.4) \quad \frac{\partial}{\partial x_1}f_i(X_1^{(0)}, x_2, x_3), \quad \frac{\partial}{\partial x_2}f_i(X_1^{(0)}, X_2^{(0)}, x_3), \quad \frac{\partial}{\partial x_3}f_i(X_1^{(0)}, X_2^{(0)}, X_3^{(0)}).$$

Note that the j th argument of $\partial f_i/\partial x_j$ is always an interval. Hence if, for all i , this derivative is a function of x_j only, no improvement in method occurs.

An additional improvement of Moore's method is obtained if we solve an interval version of the system of linear equations (1.3), $J(X^{(0)})(x - y) = f(x)$, without inverting $J(X^{(0)})$. For details of such methods, see [4] or [9].

3. Polynomial Roots. Moore [1] also discusses the problem of using an interval version of Newton's method to find real roots of real polynomials. Given an initial interval containing a root, his method has the important feature that succeeding intervals, obtained by the method, also contain the root. Unfortunately, this property does not carry over to the complex case. This is because the mean value theorem (which assures this property) fails to hold.

However, by expressing the polynomial in terms of its real and imaginary parts we obtain two equations in two unknowns; therefore, the method of the previous section can be used. Starting with a rectangle in the complex plane known to contain a root, we obtain a nested sequence of rectangles converging (under appropriate conditions) to the root of the polynomial.

In practice, it seems best to obtain an approximate solution using ordinary (real or complex) arithmetic. An error bounding procedure can then be used to provide the initial region containing a root. This error bound can then be refined using the interval procedure.

The initial error bound can be rather crude. The bound expressed in the following theorem is derived (and refined) by Champagne [2].

THEOREM. *Given z_1 and a polynomial $p(z)$ of degree n , there exists a zero of $p(z)$*

in the circle in the complex plane with center z_1 and radius $|p(z_1)/a_n|^{1/n}$, where a_n is the coefficient of z^n in $p(z)$.

We illustrate the methods of this and the previous section by an example. Consider the polynomial

$$(3.1) \quad \begin{aligned} p(z) &= 9z^2 - 6iz - 19 \\ &= 9(z - i/3 - 2^{1/2})(z - i/3 + 2^{1/2}). \end{aligned}$$

Let $z = x_1 + ix_2$. Then

$$(3.2) \quad p(z) = 9(x_1^2 - x_2^2) + 6x_2 - 19 + 6ix_1(3x_2 - 1).$$

Denote the real and imaginary parts of $p(z)$ by

$$(3.3) \quad f_1(x_1, x_2) = 9(x_1^2 - x_2^2) + 6x_2 - 19$$

and

$$(3.4) \quad f_2(x_1, x_2) = 6x_1(3x_2 - 1),$$

respectively. We thus have the vector function $f(x)$. We seek a vector

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

such that $f(y) = 0$ as in Eq. (1.1).

For illustrative purposes, we begin with the poor approximation $z = 1.41 + 0.33i$. That is, $x_1 = 1.41$ and $x_2 = 0.33$. Thus,

$$|p(z)|^2 = [f_1(x_1, x_2)]^2 + [f_2(x_1, x_2)]^2 = 0.018649$$

and $|p(z)/a_n|^{1/2} < 0.12319$. Applying the above theorem, we see that $p(z)$ has a root in the circle with center $1.41 + 0.33i$ and radius 0.12319 . We shall use a cruder bound obtained in another way. We replace the value of the radius by 0.36955 and replace the circular bounding region by a square containing it. Thus, we know there is a root of $p(x)$ whose real part is in the interval $X_1^{(0)} = [1.41 - 0.36955, 1.41 + 0.36955] \subset [1.0404, 1.7796] = X_1^{(0)}$ and whose imaginary part is in

$$[0.33 - 0.36955, 0.33 + 0.36955] = [-0.03955, 0.69955] = X_2^{(0)}.$$

We now apply the method of Section 2. Note that $\partial f_1/\partial x_i$ ($i = 1, 2$) is a function of x_i alone. Hence, the order in which we perform the Taylor expansions of f_1 is irrelevant. This is not the case for f_2 . Denote the elements of the matrix J by

$$A = \frac{\partial f_1}{\partial x_1}, \quad B = \frac{\partial f_1}{\partial x_2}, \quad C = \frac{\partial f_2}{\partial x_1}, \quad \text{and} \quad D = \frac{\partial f_2}{\partial x_2}.$$

When these elements become intervals, the second row can be either

$$(3.5) \quad C_1 = C(X_1, x_2), \quad D_1 = D(X_1, X_2),$$

or

$$(3.6) \quad C_2 = C(X_1, X_2), \quad D_2 = D(x_1, X_2).$$

That is, it can be either

$$18x_2 - 6, 18X_1 \quad \text{or} \quad 18X_2 - 6, 18x_1.$$

For this two-dimensional case, we can write down the solution to (1.3). We have

$$(3.7) \quad y_1 = x_1 - (1/\Delta)(Df_1 - Bf_2),$$

$$(3.8) \quad y_2 = x_2 - (1/\Delta)(Af_2 - Cf_1)$$

where

$$(3.9) \quad \Delta = AD - BC.$$

Using (3.5), the interval versions of (3.7) and (3.8) are

$$(3.10) \quad Y_1 = x_1 - \frac{D(X_1, X_2)f_1(x_1, x_2) - B(X_1, X_2)f_2(x_1, x_2)}{A(X_1, X_2)D(X_1, X_2) - B(X_1, X_2)C(X_1, x_2)},$$

$$(3.11) \quad Y_2 = x_2 - \frac{A(X_1, X_2)f_2(x_1, x_2) - C(X_1, x_2)f_1(x_1, x_2)}{A(X_1, X_2)D(X_1, X_2) - B(X_1, X_2)C(X_1, x_2)}.$$

For simplicity, we have written both arguments of A and B as intervals. Using (3.6), the interval versions of (3.7) and (3.8) are

$$(3.12) \quad Y_1 = x_1 - \frac{D(x_1, X_2)f_1(x_1, x_2) - B(X_1, X_2)f_2(x_1, x_2)}{A(X_1, X_2)D(x_1, X_2) - B(X_1, X_2)C(X_1, X_2)},$$

$$(3.13) \quad Y_2 = x_2 - \frac{A(X_1, X_2)f_2(x_1, x_2) - C(X_1, X_2)f_1(x_1, x_2)}{A(X_1, X_2)D(x_1, X_2) - B(X_1, X_2)C(X_1, X_2)}.$$

In (3.10) and (3.11) a noninterval argument of A , B , C , and/or D occurs in three places; and the same is true of the solution given by (3.12) and (3.13). However, if we use the solution expressed by (3.11) and (3.12), noninterval arguments occur in four places. Hence, this is the best choice (in the absence of further information as was indicated earlier). For higher-order systems, this device becomes impractical. We are using one form of J to solve for Y_1 and another to solve for Y_2 . Thus, we are essentially solving a set of n (here $n = 2$) linear equations n times. For the simple case $n = 2$, the sharper results usually are worth the extra work. This is especially true when A and/or B are also functions of an argument which can be noninterval.

We now continue with our example. Choose x_1 as the center of $X_1^{(0)} = [1.0404, 1.7796]$, i.e., $x_1 = 1.41$. Similarly, $x_2 = 0.33$, the center of $X_2^{(0)} = [-0.03955, 0.69955]$. Substituting these values for x and $X^{(0)}$ into (3.12) we obtain

$$Y_1^{(0)} = [1.4138, 1.4177]$$

and hence

$$X_1^{(1)} = X_1^{(0)} \cap Y_1^{(0)} = Y_1^{(0)}.$$

This result was obtained using five decimal digits of significance and rounding so that the interval resulting from each operation contains the exact result for that

operation. Using Moore's formulation in which all arguments of A , B , C , and D are intervals, we would obtain

$$Y_1^{(0)} = [1.4113, 1.4231].$$

This interval is more than three times as wide as that obtained above.

We now wish to find $Y_2^{(0)}$. If we used (3.13), we would not have to recalculate the denominator in the right member since this would have been evaluated in finding $Y_1^{(0)}$. However, we shall use (3.11) for reasons indicated above. It is important to note that we can use $X_1 = X_1^{(1)}$ rather than $X_1 = X_1^{(0)}$. This is the benefit gained at the expense of having to recalculate the denominator in the right member.

Note that $x_1 = 1.41$ is not contained in $X_1^{(1)}$. Hence, we must redefine x_1 (and recompute f_1 and f_2) in order to assure that $Y_2^{(0)}$ contains the solution element y_2 . We choose $x_1 = 1.4158$, the approximate center of $X_1^{(1)}$. Equation (3.11) yields

$$Y_2^{(0)} = [0.33290, 0.33377], \text{ and } X_2^{(1)} = X_2^{(0)} \cap Y_2^{(0)} = Y_2^{(0)}.$$

We now begin again with $x_2 = 0.33334$, the center of $X_2^{(1)}$. Eq. (3.12) yields

$$Y_1^{(1)} = [1.4141, 1.4143] = X_1^{(2)}.$$

We next use $x_1 = 1.4142$, apply (3.11) and obtain

$$Y_2^{(1)} = [0.33333, 0.33335] = X_2^{(2)}.$$

The ultimate accuracy (i.e., the widths of the intervals containing the solution values) depends on precision used in the computation. Using five decimal digits, greater or less accuracy than obtained here can be gotten by appropriate use, or lack of use, of semi-double-precision operations such as double-precision accumulation of scalar products.

Using infinite precision to continue the process begun above, the iterates converge to the solution vector

$$\begin{bmatrix} 2^{1/2} \\ 1/3 \end{bmatrix}.$$

4. The Matrix Eigenvalue Problem. The matrix eigenvalue problem can be solved by the method of Section 2 to yield bounds on the eigensolutions. However, we shall derive a procedure that permits use of complex arithmetic so that it is not necessary to separate the problem into real and imaginary parts.

Given a matrix A of order n , let λ be an eigenvalue and x the corresponding eigenvector. Then

$$(4.1) \quad (A - \lambda I)x = 0.$$

Let j be such that $x_j \neq 0$ and normalize x so that $x_j = 1$. In practice we might choose j so that $|x_j| \geq |x_i|$ for all $i = 1, 2, \dots, n$. For simplicity assume $j = n$. Eq. (4.1) now consists of n equations in the unknowns x_1, x_2, \dots, x_{n-1} , and λ . We could solve these equations by the method of Section 2.

However, we now consider an alternative procedure. Let $\mu = \lambda + \Delta\lambda$ be an approximate eigenvalue and $v = x + \Delta x$ an approximation for the corresponding eigenvector. Assume $v_n \neq 0$ (note $x_n \neq 0$ by assumption) and normalize v so that $v_n = 1$. Then $\Delta x_n = 0$.

Expanding in Taylor series (without remainder), we obtain

$$(4.2) \quad (A - \mu I)v = (A - \lambda I)x - x\Delta\lambda + (A - \lambda I)\Delta x - \Delta\lambda\Delta x.$$

Note the series terminates since $(A - \mu I)v$ is bilinear in μ and v . Since $(A - \lambda I)x = 0$, we can rewrite (4.2) either as

$$(4.3) \quad (A - \mu I)v = -x\Delta\lambda + (A - \mu I)\Delta x$$

or as

$$(4.4) \quad (A - \mu I)v = -v\Delta\lambda + (A - \lambda I)\Delta x.$$

Other forms could be obtained by combining separate components of $\Delta\lambda\Delta x$ with other terms in the equation. However, the two given forms seem to be of most interest.

Define the column vectors

$$(4.5) \quad p = (x_1, \dots, x_{n-1}, \lambda)^T$$

and

$$(4.6) \quad \Delta p = (\Delta x_1, \dots, \Delta x_{n-1}, \Delta\lambda)^T$$

where the superscript T denotes the transpose. Let e_i denote the i th column of the identity matrix. Then

$$(4.7) \quad \Delta x = (I - e_n e_n^T) \Delta p$$

and

$$(4.8) \quad \Delta\lambda = e_n^T \Delta p.$$

Hence, we can rewrite (4.3) and (4.4) as

$$(4.9) \quad (A - \mu I)v = [(A - \mu I)(I - e_n e_n^T) - x e_n^T] \Delta p$$

and

$$(4.10) \quad (A - \mu I)v = [(A - \lambda I)(I - e_n e_n^T) - v e_n^T] \Delta p,$$

respectively.

Define the matrices

$$(4.11) \quad B(v, \lambda) = (A - \lambda I)(I - e_n e_n^T) - v e_n^T$$

and

$$(4.12) \quad C(x, \mu) = (A - \mu I)(I - e_n e_n^T) - x e_n^T$$

and let

$$(4.13) \quad b = (A - \mu I)v.$$

Then (4.9) and (4.10) can be written as

$$(4.14) \quad C(x, \mu) \Delta p = b$$

and

$$(4.15) \quad B(v, \lambda) \Delta p = b ,$$

respectively. If we could solve one of these equations for Δp , we could obtain the desired solution vector p . However, the coefficient matrices involve unknowns. The matrix B involves λ and C involves x . If we knew a bound for x such as $x \in x^I$ where x^I is an interval vector, we could replace (4.14) by

$$(4.16) \quad C(x^I, \mu) \Delta p = b$$

and solve for Δp (now an interval) by a method described in [4] and [9]. The solution Δp^I thus obtained will contain the exact correction Δp . More commonly, we obtain a bound on λ , say $\lambda \in \lambda^I$, where λ^I is an interval. We can replace λ in B by λ^I and solve (4.15) in the interval form

$$(4.17) \quad B(v, \lambda^I) \Delta p = b$$

for the (interval) vector Δp . Again, the solution Δp^I thus obtained will contain the exact correction Δp . Using the (hopefully) improved bound for λ , the procedure can be repeated.

We shall use (4.17). We have derived (4.16) also to illustrate the choices of formulation. These choices correspond to the previous choices of order in which Taylor series expansions were made in Section 2.

Note that the intervals being discussed could be complex, that is, rectangles in the complex plane. (For analytical purposes, circles could be used; but no practical procedure for an "interval" arithmetic using anything but rectangles has yet been devised.) For a discussion of complex interval arithmetic, see [5].

Since the solution Δp^I to (4.17) contains the exact Δp , we obtain a bound on the eigenvector x knowing an initial bound on the eigenvalue alone.

Note that the matrix A could itself be an interval matrix. We shall discuss this case and other aspects of the method in another paper.

4.1. *A shortcoming of the method.* Note also that if the matrix $B(v, \lambda^I)$ contains a singular matrix, Eq. (4.17) cannot be solved. This must occur if $v = x$ and λ^I contains more than one eigenvalue (including the case of a multiple eigenvalue). To see this, let λ' denote an eigenvalue of A , where $\lambda' \in \lambda^I$. If $\lambda' \neq \lambda$, there is a row eigenvector y^T such that

$$(4.18) \quad y^T(A - \lambda'I) = 0$$

and

$$(4.19) \quad y^T x = 0 .$$

Now $B(x, \lambda) \in B(x, \lambda^I)$ and, from (4.11)

$$y^T B(x, \lambda^I) = y^T(A - \lambda'I)(I - e_n e_n^T) - y^T x e_n^T$$

so that from (4.18) and (4.19),

$$y^T B(x, \lambda^I) = 0 .$$

Hence, $B(x, \lambda^I)$ must be singular.

If $\lambda' = \lambda$, then $A - \lambda I$ is of rank $r \leq n - 2$ and hence the rank of $(A - \lambda I)(I - e_n e_n^T)$ cannot exceed $n - 2$. Therefore, since $x e_n^T$ is of rank 1,

$B(x, \lambda)$ is of rank $n - 1$ at most. That is, $B(x, \lambda)$ is singular if λ is a multiple eigenvalue.

If $B(v, \lambda')$ contains a singular matrix so that the above method fails, alternative procedures can be used. In another paper, we shall describe the interval arithmetic implementation of methods discussed by Wilkinson in [7] and pp. 170–188 and 637–647 of [8]. For the non-Hermitian case, this requires computation of approximations for all the eigenvectors of A in order to bound one eigenvector. The above method requires only an approximation for the eigenvector we seek to bound. Another (noninterval) method, somewhat similar to the one discussed above, is derived by Rall [6]. It, too, does not require approximations for all the eigenvectors.

4.2. *Relation to inverse iteration.* An ordinary arithmetic version of the above method is equivalent to inverse (Wielandt) iteration (see p. 619 of [8]). The ordinary Newton method would use (4.9) with x replaced by v or, what is the same, (4.10) with λ replaced by μ . We thus have

$$(4.20) \quad [(A - \mu I)(I - e_n e_n^T) - v e_n^T] \Delta p = (A - \mu I)v.$$

Assume $A - \mu I$ is nonsingular. Then (4.20) can be rewritten as

$$(4.21) \quad [I - e_n e_n^T - (A - \mu I)^{-1} v e_n^T] \Delta p = v.$$

It can be easily verified that the solution to (4.21) is

$$(4.22) \quad \Delta p = \{I - (1/N)[e_n e_n^T + (A - \mu I)^{-1} v e_n^T]\}v$$

where

$$N = e_n^T (A - \mu I)^{-1} v.$$

Thus, the next approximation for x is

$$(4.23) \quad v' = v - (I - e_n e_n^T) \Delta p = (1/N)(A - \mu I)^{-1}v$$

since $v_n = 1$. The scalar N is merely a normalization such that $v_n' = 1$. Equation (4.23) shows that the change in v using (4.20) is the same as one step of inverse iteration.

Solving (4.20) also yields a new approximation for the eigenvalue. From (4.21),

$$\mu' = \mu - e_n^T \Delta p = \mu + 1/N,$$

since $v_n = 1$. This is a rather obvious choice for μ' ; note that for v equal to the exact eigenvector, x , we have $N = 1/(\lambda - \mu)$ and hence $\mu' = \lambda$. More sophisticated choices using Aitken's acceleration technique (see Section 2.1.5 of [10]) or the Rayleigh quotient (see p. 636 of [8]) could be used.

We thus see that one step of the Newton method is equivalent to one step of inverse iteration with a somewhat crude improvement of the approximate eigenvalue.

4.3. *An example.* We conclude by presenting a simple example. Consider the matrix

$$(4.24) \quad A = \begin{bmatrix} 1 & 3 \\ -4 & 2 \end{bmatrix}$$

which has eigenvalues $\frac{1}{2} [3 \pm i(47)^{1/2}]$ and eigenvectors

$$\begin{bmatrix} \frac{1}{8} [1 \mp i(47)^{1/2}] \\ 1 \end{bmatrix}$$

Assume that, by some means, we obtain approximations $1.4991 \pm 3.4271 i$ and

$$\begin{bmatrix} 0.12490 \mp 0.85656 i \\ 1 \end{bmatrix}$$

for the eigenvalues and eigenvectors, respectively. Using these approximations, and implementing procedures in [8] in interval arithmetic, we find

$$\lambda_1 \in \lambda_1^I = [1.4978, 1.5021] + i[3.4253, 3.4301].$$

The details of how this bound was obtained need not concern us here. A full description will appear elsewhere. Suffice it to say that the bound was obtained from the initial approximations by use of five-significant-digit interval arithmetic with double-precision (ten-digit) accumulation of scalar products. Computations described below were performed in the same manner.

Using this bound with the approximations $\mu = 1.4991 + 3.4271 i$ and

$$v = \begin{bmatrix} 0.12490 - 0.85656 i \\ 1 \end{bmatrix}$$

and substituting into (4.17), we find

$$\begin{aligned} & \begin{bmatrix} [-0.50210, -0.49780] + i[-3.4301, -3.4253] - 0.12490 \mp 0.85656 i \\ -4 \qquad \qquad \qquad -1 \end{bmatrix} \Delta p \\ &= \begin{bmatrix} [0.0021456, 0.0021457] - i[0.00053567, 0.00053570] \\ 0.0013 - 0.00086 i \end{bmatrix}. \end{aligned}$$

In general, this equation would be solved by a method described in [4] or [9]. However, for the simple case here, we can write down analytic expressions for the elements of Δp . Evaluating these expressions using interval arithmetic, we obtain

$$10^5 \Delta p = \begin{bmatrix} [-10.025, -9.9755] + i[39.632, 39.734] \\ [-90.195, -89.798] + i[-72.823, -72.547] \end{bmatrix}.$$

So far we have used five-digit arithmetic. As a last step we subtract Δp from p using double precision and obtain (after separating out λ)

$$\lambda \in [1.4999\ 9798, 1.5000\ 0195] + i[3.4278\ 2547, 3.4278\ 2823]$$

and

$$x \in \begin{bmatrix} [0.12499\ 9755, 0.12500\ 025] + i[-0.85695\ 734, -0.85695\ 632] \\ 1 \end{bmatrix}.$$

Note that the left and right endpoints of the intervals bounding λ and x agree to about six digits. The initial approximations were correct to only about three or four digits. The endpoints of the intervals in the initial bound on λ agreed to about three digits.

5. Conclusion. In this paper, we have presented procedures and simple examples for bounding solutions of nonlinear equations. In later papers, we shall present more complicated examples and discuss computational details.

Note that in the examples used here, we obtained a posteriori bounds using approximations obtained by ordinary arithmetic. In the course of computing the bounds, we improved the approximations as well. In practice, we recommend the use of ordinary arithmetic as much as possible. Thus, the initial approximations should be accurately obtained to reduce the number of (slower) computations in interval arithmetic.

Lockheed Palo Alto Research Laboratory
Palo Alto, California 94304

1. R. E. MOORE, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, N. J., 1966.
2. W. P. CHAMPAGNE, JR., *On Finding Roots of Polynomials by Hook or by Crook*, Master's Thesis, Univ. of Texas, 1964.
3. ELDON HANSEN, "Interval arithmetic in matrix computations. I," *J. Soc. Indust. Appl. Math. Ser. B, Numer. Anal.*, v. 2, 1965, pp. 308-320. MR 32 #4833.
4. ELDON HANSEN & ROBERTA SMITH, "Interval arithmetic in matrix computations. II," *SIAM J. Numer. Anal.*, v. 4, 1967, pp. 1-9.
5. RAY BOCHE, "Complex interval arithmetic with some applications," Lockheed Missiles and Space Company Report #4-22-66-1, 1966.
6. L. B. RALL, "Newton's method for the characteristic value problem $Ax = \lambda Bx$," *J. Soc. Indust. Appl. Math.*, v. 9, 1961, pp. 288-293. MR 23 #B1110.
7. J. H. WILKINSON, "Rigorous error bounds for computed eigensystems," *Comput. J.*, v. 4, 1961/62, pp. 230-241. MR 23 #B2161.
8. J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965. MR 32 #1894.
9. ELDON HANSEN & ROBERTA SMITH, "A computer program for solving a system of linear equations and matrix inversion with automatic error bounding using interval arithmetic," Lockheed Missiles and Space Company Report #4-22-66-3, June 1966. (This program has been submitted to SHARE under the name, Linear Systems Dynam (LSD).)
10. E. BOEDEWIG, *Matrix Calculus*, Interscience, New York, 1959. MR 23 #B563.