Solving nonlinear Vandermonde systems

G. E. Trapp, Jr. and W. Squire*

Department of Statistics and Computer Science, West Virginia University, Morgantown, West Virginia 26506, USA
**Department of Aerospace Engineering,*

The nonlinear system of equations $V(x)x = b$, where $V(x)$ is a Vandermonde matrix, may be used to determine roots of polynomials, eigenvalues of matrices and Chebychev quadrature formulae. A convergent iterative scheme is given to solve $V(x)x = b$, based on known methods for solving linear Vandermonde systems.
(Received September 1973)

This note describes an iterative procedure for solving nonlinear systems of equations of the form

$$\sum_{i=1}^{n} x_i^j = b_j,\ j = 1, \ldots, n\ . \qquad (1)$$

The numerical examples given in this paper are the determination of the roots of polynomials, but the method may also be used to find eigenvalues of matrices and nodes of generalized Chebychev quadrature formulae (Fröberg, 1965).

Let $x = (x_1, \ldots, x_n)^T$ and let $V(x)$ denote the Vandermonde matrix of order n with generating vector x, i.e.

$$V(x) = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_n \\ \vdots & & & \\ x_1^{n-1} & x_2^{n-1} & \ldots & x_n^{n-1} \end{bmatrix} .$$

Equation (1) may be rewritten as $V(x)x = b$, with $b = (b_1, \ldots, b_n)^T$.

Considerable work has been done on the solution of a linear Vandermonde system $-V(x)y = b$, with y the unknown. Traub (1966) gives two algorithms for generating the inverse of a Vandermonde matrix. Björck and Pereyra (1970), and Gustafson (1971) have developed algorithms for solving the linear Vandermonde system which only require n^2 operations, which is an order of magnitude less than the requirement for an arbitrary linear system.

We first considered the following iteration scheme for solving equation (1), guess x^0 and for $k = 0, 1, 2, \ldots$, determine $x^{(k+1)}$ from the equation $V(x^{(k)})x^{(k+1)} = b$. Unfortunately, this method proves to be highly unstable. An improved scheme is obtained by equating the solution to an approximate solution plus a correction term ε so that

$$V(x^{(k)} + \varepsilon)(x^{(k)} + \varepsilon) = b\ .$$

Expanding and retaining only first order terms in ε gives

$$V(x^{(k)})\varepsilon = D(b - V(x^{(k)})x^{(k)})\ ,$$

where D is a diagonal matrix with $D_{jj} = 1/j$. It is a measure of the efficiency of the algorithms for solving linear Vandermonde systems that the calculation of the right hand side requires as many multiplications as the solution of the system of linear equations.

Our iteration scheme becomes: given x^0, determine $\varepsilon^{(k)}$ and $x^{(k+1)}$ for $k = 0, 1, 2, \ldots$, as follows.

$$\begin{aligned} &(a)\ V(x^{(k)})\varepsilon^{(k)} = D(b - V(x^{(k)})x^{(k)}) \\ &(b)\ x^{(k+1)} = x^{(k)} + \varepsilon^{(k)} \end{aligned} \qquad (2)$$

Notice that in solving equation (2a), one has a linear Vandermonde system.

In solving problems where the solution has all real components, we noticed that using real initial guesses led to instability. We therefore modified the scheme to prevent large variations in the x's as follows.

If

$$|\varepsilon_j^{(k)}| \leqslant A\ |x_j^{(k)}|$$

then

$$x_j^{(k+1)} = x_j^{(k)} + \varepsilon_j^{(k)}$$

but if

$$|\varepsilon_j^{(k)}| > A\ |x_j^{(k)}|$$

then

$$x_j^{(k+1)} = x_j^{(k)} + A\varepsilon_j^{(k)}\ |\ x_j^{(k)}|/|\varepsilon_j^{(k)}|.$$

We found $A = 4/3$ satisfactory.

We found this correction was not necessary in the case of complex initial guesses. Our procedure for selecting initial guesses is discussed in the next section.

A simple way to analyse convergence is to recast the above algorithm in terms of Newton's method. We refer to Rheinboldt and Ortega (1970) for the appropriate background.

Let $F(x) = [f_1(x), \ldots, f_n(x)]^T$, with $f_j(x) = \sum_{i=1}^{n} x_i^j - b_j$.

Then Newton's method may be written

$$x^{(k+1)} = x^{(k)} + [F'(x^{(k)})]^{-1}\ F(x^{(k)}) \qquad (2')$$

where $(F'(x))_{ji} = \partial_j f_i(x)$, the Jacobian of $F(x)$.

It is easily seen that for our case $F'(x) = D^{-1}V(x)$ and that equations (2′) and (2) are equivalent.

Letting $\bar{x}$ denote the true solution with components $\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n$, then if the x_i are distinct, $V(\bar{x})$ is nonsingular and hence $F'(\bar{x}) = D^{-1}V(x)$ is nonsingular. Therefore, there is a neighbourhood U about $\bar{x}$ such that Newton's Method converges for all starting values within U. Our results indicate that the convergence is quadratic. However since the first row of $V(x)x$ is linear, the second derivatives are zero, and the standard theorems concerning quadratic convergence do not apply.

Applications

Using the IBM 360/75 with (16S) complex arithmetic we tested our algorithm by determining roots of polynomials. Let $p(x)$ be a polynomial with roots $x_1, \ldots, x_n - p(x) = a_0x^n + a_1x^{n-1} + \ldots + a_n$—then

$\sum_{i=1}^{n} x_i^j = b_j$ where the b_j are computed from the triangular system

$$\begin{aligned} a_1 + a_0b_1 &= 0 \\ 2a_2 + a_1b_1 + a_0b_2 &= 0 \\ &\vdots \\ na_n + a_{n-1}b_1 + \ldots + a_0b_n &= 0\ , \end{aligned}$$

These equations, known as the Newton Identities may be found in Fröberg (1965).

In choosing initial guesses for our procedure, we followed the work of Aberth (1973), and selected initial values so that they all lie on a circle in the complex plane which includes all the roots. Obviously the method cannot give complex roots from a set of real initial guesses, however experience indicates that even when all the roots are real, convergence is better when complex initial guesses are used.

For other than roots of polynomial applications, note that the above triangular system may be solved for $a_1, \ldots, a_n$ given $b_1, \ldots, b_n$ and setting $a_0 = 1$. Therefore, one has a polynomial which has the solution to $V(x)x = b$ as its roots and the estimating procedure of Aberth may be used to determine initial guesses.

We let E be the maximum difference between the real and imaginary parts of the true roots and the approximate roots.

For $p(x) = x^5 - 10x^4 + 43x^3 - 104x^2 + 150x - 100$, with roots $1 \pm 2i$, 2, $3 \pm i$, see Aberth (1973), we found after 10 iterations that E was less than 10^{-14}.

Two close but unequal roots may also be solved. We let $p(x) = x^5 + 0{\cdot}001x^4 + x^2 + 0{\cdot}001x = x(x + 0{\cdot}001)(x^3 + 1)$ and after 20 iterations, E was smaller than 10^{-9}. The closeness of the two smaller roots accounts for the increase in iterations. This increase is more fully brought out by considering $p(x) = x^5 - 8x^3 + 16x$ where the roots are $-2, -2, 0, 2, 2$. In this case it took 30 iterations to reduce the error to below 10^{-8}. The linear system of equations becomes very ill-conditioned whenever the x's are closely spaced.

These results appear to indicate that this new method is competitive with other methods (Aberth, 1973; Grau, 1971) for simultaneously determining all the roots of polynomials.

References

ABERTH, OLIVA (1973). Iteration Methods for Finding All Zeroes of a Polynomial Simultaneously, *Math. Comp.* Vol. 27, pp. 339-344.

BJÖRCK, A., and PEREYRA, V. (1970). Solution of Vandermonde Systems of Equations, *Math. Comp.* Vol. 24, pp. 893-904.

FRÖBERG, C. E. (1965). *Introduction to Numerical Analysis*, Addison-Wesley, Reading, Massachusetts.

GRAU, A. A. (1971). The Simultaneous Newton Improvement of a Complete Set of Approximate Factors of a Polynomial, *Siam Num Anal.*, Vol. 8, pp. 435-438.

GUSTAFSON, S. A. (1971). Rapid Computation of General Interpolation Formulas and Mechanical Quadrature Rules, *CACM*, Vol. 14, pp. 797-801.

ORTEGA, J., and RHEINBOLDT, W. (1970). *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York.

TRAUB, J. F. (1966). Associated Polynomials and Uniform Methods for the Solution of Linear Problems, *Siam Review*, Vol. 8, pp. 277-301.