UCRL- 101197
PREPRINT

# Detecting Stability Barriers in BDF Solvers

Alan C. Hindmarsh
Lawrence Livermore National Laboratory
Livermore, California

This paper was prepared for submittal to the
IMA Conference on Computational
Ordinary Differential Equations
July 3-7, 1989

June 1, 1989

# Detecting Stability Barriers in BDF Solvers*

Alan C. Hindmarsh
Computing & Mathematics Research Division, L-316
Lawrence Livermore National Laboratory
Livermore, CA 94550

June 1, 1989

**Abstract**

When a Backward Differentiation Formula method is used on an ODE problem with a damped but strongly oscillatory mode, the step size may be unduly limited if the order is three or more. The performance of existing BDF codes in this situation varies from poor to fair. Analysis of the computed solution and related quantities in model problems yields a set of relationships that allow for direct detection of the stability barrier. The analysis is fairly easy for a complex scalar model problem, but for a corresponding real system the diagonalizing transformation greatly complicates the relationships. But a solution of them leads to an algorithm for determining whether a dominant oscillatory mode is limiting the step size, and to estimate the magnitude of the associated characteristic root. It uses only the norms of the scaled derivatives of the solution, but bears no resemblance to the common practice of demanding monotonicity among these norms.

Running head: Detecting BDF Stability Barriers

*This work was performed under the auspices of the U.S Department of Energy by the Lawrence Livermore National Laboratory under contract W-7405-Eng-48.

# 1 Introduction

In the initial value problem for an ODE system, suppose that the solution contains a damped but strongly oscillatory mode. Then the numerical solution by a BDF (Backward Differentiation Formula) method may well be restricted in step size by the nature of the absolute stability regions. In fact, BDF solvers are infamous for getting stuck at an order $\geq 3$, with step size limited by absolute stability. Moreover, in tests with simple linear problems, existing BDF solvers are not particularly good at sensing this stability limitation. Of course, a direct test of the system eigenvalues against the boundary of the absolute stability region is impossible. Instead, a BDF solver will attempt to sense the stability limit by testing various available difference expressions. This generally involves calculating the norms of certain scaled derivatives of the solution. In some solvers, these are tested for a monotonicity condition of some kind. But in all cases, these algorithms perform erratically on model problems.

The question arises: Can one do better? Can one use available data to detect reliably when an oscillatory solution mode is causing the BDF solution to be stability-limited? In what follows, an affirmative answer is given. The case of a complex scalar model problem is analyzed first. Then a model $2 \times 2$ real linear system is reduced to the complex scalar case. The resulting relationships are solved, and the solution leads to a limit detection algorithm. Finally, some numerical test results are given.

# 2 The Scalar Model Problem

The simplest problem for which the stability limit issue makes sense is the standard complex scalar model,

$$\dot{u} = \lambda u \quad (Re(\lambda) < 0). \tag{2.1}$$

If $|Im(\lambda)/Re(\lambda)|$ is large enough, a BDF solution at order $q \geq 3$ will have a stability-limited step size.

Take the step size $h$ to be constant. Then the BDF solution values $u_n$ from (2.1) are linear combinations of powers of the $q$ characteristic roots $z_j$ of the BDF. When $h$ is near its stability limit, one of these, say $z_1$, approaches 1, while $|z_j| < 1$ for $j > 1$. After a sufficient number of steps, the term in $u_n$ involving $z_1$ dominates all the others. In fact we will neglect completely the nondominant terms and write

$$u_n = a z_1^n \quad (a = \text{complex constant}). \tag{2.2}$$

At time step $n$, let $\pi_n(t)$ be the interpolating polynomial of degree $\leq q$ which interpolates the solution data $\{u_n, \cdots, u_{n-q}\}$. Define scaled derivatives of the solution, denoted

$$\sigma_n(k) = h^k u_n^{(k)}.$$

These are defined by $u_n^{(k)} \equiv \pi_n^{(k)}(t_n)$ for $k \leq q$, whereas $\sigma_n(q+1) \equiv \nabla\sigma_n(q)$. In the case at hand, $\pi_n$ interpolates $a(z_1)^{n-j}$ $(j = 0, \cdots, q)$. Its leading coefficient is given by a $q^{th}$ order backward difference:

$$\sigma_n(q) = h^q \pi_n^{(q)} = \nabla^q \pi(t_n) = \nabla^q(az_1^n).$$

Clearly, the operator $\nabla$ has the same action upon $z_1^n$ as multiplication by $1 - z_1^{-1}$, and so

$$\sigma_n(q) = az_1^n(1 - z_1^{-1})^q \quad \text{, and} \tag{2.3}$$

$$\sigma_n(q+1) = az_1^n(1 - z_1^{-1})^{q+1}. \tag{2.4}$$

To get $\sigma_n(q-1)$, use the relation $\nabla = 1 - e^{-hD}$ between $\nabla$ and differentiation $D$ to get

$$\nabla^{q-1} = (hD)^{q-1}\left[1 - \frac{1}{2}hD + \cdots\right]^{q-1} = (hD)^{q-1}\left[1 - \left(\frac{q-1}{2}\right)hD + \cdots\right],$$

$$\nabla^{q-1}\pi_n(t_n) = (hD)^{q-1}\pi_n(t_n) - \frac{q-1}{2}(hD)^q\pi(t_n) = \sigma_n(q-1) - \frac{q-1}{2}\sigma_n(q). \tag{2.5}$$

On the other hand, $\nabla^{q-1}\pi_n(t_n) = az_1^n(1 - z_1^{-1})^{q-1}$. Substituting this and (2.3) into (2.5) gives

$$\sigma_n(q-1) = az_1^n\left[(1 - z_1^{-1})^{q-1} + \frac{q-1}{2}(1 - z_1^{-1})^q\right] = az_1^n(1 - z_1^{-1})^{q-1}\left[\frac{q+1}{2} - \frac{q-1}{2}z_1^{-1}\right]. \tag{2.6}$$

We could evaluate the other $\sigma_n(k)$ in the same way, but we will not need them.

In practice, the complex scaled derivatives $\sigma_n(k)$ would not actually be available. But we would expect that their moduli would be. So define

$$\Sigma_n(k) \equiv |\sigma_n(k)|^2. \tag{2.7}$$

Each $\Sigma_n(k)$ is proportional to $|z_1|^{2n}$, with a constant depending on $k$. Our goal is to try to recover the value of $|z_1|$ from the $\Sigma_n(k)$. If we look at the two ratios

$$\rho_+ \equiv \Sigma_n(q+1)/\Sigma_n(q) \quad \text{and} \quad \rho_- \equiv \Sigma_n(q-1)/\Sigma_n(q), \tag{2.8}$$

we see that these are independent of $n$, but depend on both the modulus and argument of $z_1$. Moreover, actual values of these two ratios taken at various points on the BDF stability boundaries show that there is no correspondence between being in the absolute stability region and a monotonicity relation such as

$$\Sigma_n(q-1) > \Sigma_n(q) > \Sigma_n(q+1), \tag{2.9}$$

the failure of which is used as a signal to reduce the order in some BDF solvers. Relation (2.9) holds at some points inside the absolute stability region but not at others, and the same is true outside the region.

To eliminate the unknown $\arg(z_1)$, look first at the case $|z_1| = 1$, or $z_1 = e^{i\theta}$. We can evaluate

$$\begin{aligned} \rho_+ &= |1 - z_1^{-1}|^2 = |1 - e^{-i\theta}|^2 = 4\sin^2\theta/2, \qquad \text{and} \\ 4|1 - z_1^{-1}|^2 \rho_- &= |(q+1) - (q-1)z_1^{-1}|^2 \\ &= [(q+1) - (q-1)\cos\theta]^2 + [(q-1)\sin\theta]^2 = 2(q^2+1) - 2(q^2-1)\cos\theta \\ &= 4[1 + (q^2-1)\sin^2\theta/2]. \end{aligned}$$

On identifying the last expression with $4\rho_+\rho_-$, we have (for $|z_1| = 1$) a relation not involving $\theta$:

$$B \equiv (\rho_+\rho_-) - \left(\frac{q^2-1}{4}\right)(\rho_+) - 1 = 0. \tag{2.10}$$

Now for arbitrary $z_1 = re^{i\theta}$, consider the same expression $B$. In terms of $w = 1/z_1$ we find

$$4B = [(q+1)^2 - (q^2-1)(w+\bar{w}) + (q-1)^2 w\bar{w}] - (q^2-1)[1 - (w+\bar{w}) + w\bar{w}] - 4$$

$$= (2q-2)(1 - w\bar{w}) \quad \text{, or}$$

$$B = \left(\frac{q-1}{2}\right)(1 - |z_1|^{-2}). \tag{2.11}$$

This quantity $B$ is a "barrier function" in the following sense: First, from (2.8) and (2.10) it is computable in terms of the available data—the three values of $\Sigma_n(k)$. Secondly, by (2.11) it is precisely sensitive to the stability barrier, in that it changes sign when $|z_1|$ exceeds 1, and only then: $B > 0 \iff |z_1| > 1$. Moreover, given the $\Sigma_n(k)$, one can evaluate $B$ and deduce the value of $|z_1|$ exactly. (Since these identities rely only on the monomial form of the data, they hold for any linear multistep method of order $q$, when the multistep solution is dominated by a single characteristic root.)

# 3 Real $2 \times 2$ Model Problem

We turn our attention now the case of a real ODE system, dominated by a damped strongly oscillatory mode. The system eigenvalues will occur in conjugate pairs. Heuristically at least, we can expect that the behavior of BDF methods on such a problem will be approximated by their behavior on a corresponding linear $2 \times 2$ problem having a nonreal spectrum. We therefore consider a linear problem of the form

$$\dot{y} = Ay \tag{3.1}$$

in which $A$ is a constant real $2 \times 2$ matrix with nonreal eigenvalues $\lambda$ and $\bar{\lambda}$. Thus $A$ is diagonalizable:

$$P^{-1}AP = D = \begin{pmatrix} \lambda & 0 \\ 0 & \bar{\lambda} \end{pmatrix}. \tag{3.2}$$

Again we consider a BDF solution at order $q \geq 3$ with a fixed step size $h$, such that $h$ is limited by the absolute stability region. We proceed by reducing this problem to that for the scalar model problem $\dot{u} = \lambda u$, and analyzing the resulting equations for the norms of the scaled derivatives of $y$. However, this is one situation in which the answer for the scalar problem does not extend trivially to a linear system.

Applying $P^{-1}$ to the BDF defining $y_n$, we see that the vectors $u_n = P^{-1}y_n$ constitute a BDF solution of

$$\dot{u} = Du \quad , \qquad \text{or} \qquad \dot{u}^1 = \lambda u^1, \quad \dot{u}^2 = \bar{\lambda} u^2. \tag{3.3}$$

For each $i$, $u_n^i$ is given in terms of the characteristic roots corresponding to $\lambda$ and $\bar{\lambda}$ respectively:

$$u_n^1 = \sum_{j=1}^{q} a_j^1 z_j^n \quad \text{and} \quad u_n^2 = \sum_{j=1}^{q} a_j^2 \overline{z_j}^n$$

for some coefficients $a_j^i$. From $y_n = Pu_n$, the components of $y_n$ are

$$y_n^i = p_{i1}u_n^1 + p_{i2}u_n^2 = \sum_{j=1}^{q}(p_{i1}a_j^1 z_j^n + p_{i2}a_j^2 \overline{z_j}^n). \tag{3.4}$$

For simplicity, we can assume (with no loss of generality) that the eigenvectors of $A$ are normalized so that $p_{11} > 0$ and the second column of $P$ is the conjugate of the first: $p_{i2} = \overline{p_{i1}}$. Excluding isolated special cases, we can also assume that the $z_j$ are are nonreal and independent, and so the $y_n^i$ being real implies $p_{i2}a_j^2 = \overline{p_{i1}a_j^1}$. (Actually, one can show directly that $a_j^2 = \overline{a_j^1}$.) Therefore in (3.4), $u_n^2 = \overline{u_n^1}$ and $y_n^i = 2Re(p_{i1}u_n^1)$. Assuming as before that $z_1$ is completely dominant, we have $u_n^1 = az_1^n$ $(a = a_1^1)$ and so

$$y_n^i = 2Re(p_{i1}az_1^n). \tag{3.5}$$

We next analyze the scaled derivatives of $y_n$, as obtained from the interpolatory polynomial at step $n$. For some $M = M(k)$ and real coefficients $\gamma_j = \gamma_j(k)$ depending only on $k$, these vectors are given by

$$s_n(k) \equiv h^k y_n^{(k)} = \sum_{j=0}^{M(k)} \gamma_j(k) y_{n-j}. \tag{3.6}$$

In terms of the corresponding scaled derivatives of $u_n^1$,

$$\sigma_n(k) \equiv h^k u_n^{1(k)} = \sum_{j=0}^{M} \gamma_j u_{n-j}^1 = \sum \gamma_j a z_1^{n-j}, \tag{3.7}$$

we can then express the components of the $s_n(k)$ as

$$s_n^i(k) = 2\sum_0^M \gamma_j Re(p_{i1} a z_1^{n-j}) = 2Re[p_{i1}\sigma_n(k)]. \tag{3.8}$$

We seek precise relations among the norms of the vectors $s_n(k)$. So we now need to specify the norm used, and in fact we must assume that it is a weighted $\ell_2$-norm,

$$||v|| = [(w_1 v^1)^2 + (w_2 v^2)^2]^{1/2}$$

with constant positive weights $w_i$. Now we can evaluate the norms of the scaled derivatives, defining

$$S_n(k) \equiv ||s_n(k)||^2 = 4\{Re[w_1 p_{11}\sigma_n(k)]^2 + Re[w_2 p_{21}\sigma_n(k)]^2\}. \tag{3.9}$$

Recall from (2.3) - (2.6) that for each $k$, $\sigma_n(k)$ is a complex constant (depending on $k$) times $z_1^n$, and as in (2.7) denote $\Sigma_n(k) \equiv |\sigma_n(k)|^2$ . To simplify the results, define constants

$$\rho = (w_2 p_{21})/(w_1 p_{11}) \quad , \quad \gamma \equiv \frac{|1+\rho^2|}{1+|\rho|^2} \quad (0 \le \gamma \le 1) \quad \text{, and} \tag{3.10}$$

$$c_0 \equiv 2(w_1 p_{11})^2(1+|\rho|^2) = 2[(w_1 p_{11})^2 + (w_2 |p_{21}|)^2] \quad , \tag{3.11}$$

and angles

$$\nu_n(k) = \arg\left[\sigma_n(k)^2(1+\rho^2)\right] = 2n\theta + \nu_0(k) \quad , \tag{3.12}$$

where $\theta = \arg(z_1)$ and $\nu_0(k)$ is a constant. With these definitions, we obtain from (3.9) (with $\sigma = \sigma_n(k)$)

$$\begin{aligned} S_n(k) &= (w_1 p_{11}\sigma + w_1 p_{11}\bar{\sigma})^2 + (w_2 p_{21}\sigma + w_2\overline{p_{21}}\bar{\sigma})^2 = (w_1 p_{11})^2[(\sigma+\bar{\sigma})^2 + (\rho\sigma + \bar{\rho}\bar{\sigma})^2] \\ &= 2(w_1 p_{11})^2\{|\sigma|^2(1+|\rho|^2) + Re[\sigma^2(1+\rho^2)]\} = 2(w_1 p_{11})^2|\sigma|^2[(1+|\rho|^2) + |1+\rho^2|\cos\nu_n(k)] \\ &= 2(w_1 p_{11})^2|\sigma|^2(1+|\rho|^2)[1+\gamma\cos\nu_n(k)] \quad \text{, or} \end{aligned}$$

$$S_n(k) = c_0\Sigma_n(k)[1+\gamma\cos\nu_n(k)]. \tag{3.13}$$

# 4 Limit Detection

We now suppose that the $S_n(k)$ are given, and are known to satisfy (3.13). The quantities $\Sigma_n(k)$ etc. are unknown, and the next task is to recover these, and $|z_1|$ in particular. Begin by defining

$$R = |z_1|^2 \quad \text{and} \quad G_0(k) = c_0|\sigma_n(k)/z_1^n|^2 = c_0\Sigma_n(k)/R^n \tag{4.1}$$

so that

$$S_n(k) = G_0(k)R^n[1+\gamma\cos\nu_n(k)]. \tag{4.2}$$

An important special case arises when the matrix $A$ is <u>normal</u>. Then it is easy to show that $p_{21}/p_{11} = \pm i$. If in addition the two weights $w_i$ are equal, then we have $\rho = \pm i$ and $\gamma = 0$ by (3.10). In this case, $S_n(k)$ is proportional to $\Sigma_n(k)$ and hence to $R^n$, and so the ratios $S_{n+1}(k)/S_n(k)$ yield $|z_1|$ directly.

For the general case, we must eliminate the unknown oscillatory term in (4.2). Fix $n$ and $k$, and (dropping the $k$ for brevity) consider the consecutive values $\{S_{n-2}, S_{n-1}, S_n, S_{n+1}, S_{n+2}\}$. On expanding $S_{n\pm1}$ and $S_{n\pm2}$, Eqns. (4.2) for these constitute five equations in the five unkown parameters $G_0$, $G_0\gamma\cos\nu_n$, $G_0\gamma\sin\nu_n$, $R$, and $\cos 2\theta$. They can be solved by [redacted] exploiting their linearity in the first three unknowns.

We can temporarily eliminate the variable $R$ by working instead with quantities

$$\hat{S}_m \equiv S_m R^{n-m} = c_0\Sigma_n(1+\gamma\cos\nu_m) \quad (m = n-2, \cdots, n+2). \tag{4.3}$$

The expansion of these quantities is best expressed in terms of difference operations. Temporarily define

$$C \equiv \cos 2\theta, \quad S \equiv \sin 2\theta, \qquad c_m \equiv \cos\nu_m, \quad s_m \equiv \sin\nu_m.$$

We use (3.12), or $\nu_m = 2m\theta + \nu_0$, to express the differences of the $c_m$, and among the resulting equations is $\Delta^2 c_m = -2(1-C)c_{m+1}$. Applying differences to $\hat{S}_m = c_0\Sigma_n(1+\gamma c_m)$, we obtain

$$\Delta\hat{S}_n = c_0\Sigma_n\gamma\Delta c_n \tag{4.4}$$

$$\Delta^2\hat{S}_{n-1} = -2(1-C)c_0\Sigma_n\gamma c_n \tag{4.5}$$

$$\Delta^3\hat{S}_{n-1} = -2(1-C)c_0\Sigma_n\gamma\Delta c_n. \tag{4.6}$$

Together, these equations easily yield the value of $1-C$ and then $c_0\Sigma_n\gamma c_n$:

$$-2(1-C) = \Delta^3\hat{S}_{n-1}/\Delta\hat{S}_n \tag{4.7}$$

$$c_0\Sigma_n\gamma c_n = \Delta^2\hat{S}_{n-1}/[-2(1-C)] = \Delta^2\hat{S}_{n-1}\Delta\hat{S}_n/\Delta^3\hat{S}_{n-1}. \tag{4.8}$$

Finally, we have

$$c_0\Sigma_n = \hat{S}_n - c_0\Sigma_n\gamma c_n = \hat{S}_n - \Delta^2\hat{S}_{n-1}\Delta\hat{S}_n/\Delta^3\hat{S}_{n-1}. \tag{4.9}$$

Eq. (4.9) gives $c_0\Sigma_n$ in terms of $R$ and $S_{n-1}, S_n, S_{n+1}, S_{n+2}$. To get another equation in these unknown parameters, we repeat the above process with the 4-tuple $S_{n-2}, \cdots, S_{n+1}$. In analogy with (4.7), we obtain

$$-2(1-C) = \Delta^3\hat{S}_{n-2}/\Delta\hat{S}_{n-1}. \tag{4.10}$$

Of course, (4.7) and (4.10) must agree, giving one more equation, which is (in terms of the $S_m$)

$$\left(\frac{S_{n+2}}{R^2} - 3\frac{S_{n+1}}{R} + 3S_n - RS_{n-1}\right)(S_n - RS_{n-1}) = \left(\frac{S_{n+1}}{R} - 3S_n + 3RS_{n-1} - R^2S_{n-2}\right)\left(\frac{S_{n+1}}{R} - S_n\right).$$

Multiplication by $R^2$ gives a quartic equation in $R$,

$$Q(R) = b_4R^4 + b_3R^3 + b_1R + b_0 = 0 \quad \text{with} \tag{4.11}$$

$$\begin{aligned} b_4 &= (S_{n-1})^2 - S_nS_{n-2}, \quad & b_3 = S_{n+1}S_{n-2} - S_nS_{n-1}, \\ b_1 &= S_{n+1}S_n - S_{n+2}S_{n-1}, \quad & b_0 = S_{n+2}S_n - (S_{n+1})^2. \end{aligned}$$

For each $k = q-1, q, q+1$, the above procedure gives rise to a quartic $Q_k(R)$. The normal matrix case must be isolated first, because the coefficients in (4.11) all vanish when $\gamma \to 0$. Otherwise, if in fact the data do correspond to a dominant oscillatory mode, these three quartics will have a common root $R$. Thus we need not solve them individually, but instead can use elimination on them. Because they have no $R^2$ term, this yields a linear equation for $R$, assuming that the quartics are linearly independent. Once that common root is obtained, one can return to (4.7) - (4.9) to evaluate the other unknowns, particularly the three $c_0\Sigma_n(k)$. Using the ratios of these, we can evaluate the right-hand side of the relation

$$\left(\frac{q-1}{2}\right)(1 - R^{-1}) = B = \frac{\Sigma_n(q+1)}{\Sigma_n(q)}\left[\frac{\Sigma_n(q-1)}{\Sigma_n(q)} - \frac{q^2-1}{4}\right] - 1 \tag{4.12}$$

[from (2.8), (2.10), and (2.11)], and solve this for $R$, giving $R = R_B \equiv [1 - 2B/(q-1)]^{-1}$. This value can be checked for agreement with the value of $R$ obtained directly from the quartics.

This solution procedure, along with a test for the normal (or nearly normal) case, leads to the following crude algorithm. It includes various consistency checks to verify the validity of the dominant mode model.

<u>Limit Detection Algorithm:</u>

1. If $h$ has been constant $\geq 5$ steps and $q \geq 3$, collect the 15 values of $S_m(k)$.
2. For each $k$, look at the variance of the $S_{m+1}(k)/S_m(k)$. If all are small, get $R$ from these ratios ($A$ nearly normal). If the $R$ values are not consistent, exit. If a consistent $R$ is found, go to 4.
3. Form the three quartics $Q_k$ in (4.11) and eliminate to get a tentative $R$. If the $Q_k$ are dependent, exit. Evaluate the $Q_k(R)$ and do Newton corrections to $R$ if necessary. If the new $Q_k(R)$ are not all small, exit.
4. From $R$, get the $c_0\Sigma_n(k)$ by (4.9), then $B$ and the solution $R_B$ from (4.12). If this disagrees with $R$, exit.
5. If $R \approx 1$, or $R > 1$, signal a reduction in order.

## 5 Numerical Tests

The above algorithm, filled in with heuristic factors, was tested on data from $2 \times 2$ systems (3.1), as follows.

For a choice of ODE system, consider first the Toronto test Problem B5, which is dominated by mode with $\lambda = -10 + 100i$. But in that system, the $2 \times 2$ block is a normal matrix

$$A_0 = \begin{pmatrix} -10 & 100 \\ -100 & -10 \end{pmatrix}. \tag{5.1}$$

We can generate non-normal $A$ by a similarity transformation

$$T = \begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}, \quad A = TA_0T^{-1}. \tag{5.2}$$

The initial values and analytic solution for $\dot{y} = Ay$ are those of Problem B5 multiplied by $T$.

A BDF solver (VODE) was run on this problem with scalar absolute error control, and all of the squared scaled derivatives $S_n(k)$ were written to a file. The limit detection algorithm was applied separately to selected subsequences. Generally, the performance of the algorithm is quite good. On data too early in the solution, or too soon after a change in step size or order, it fails, as expected. But it succeeds in the situations for which it was meant, although it is subject to further tuning. The following example is typical.

Example: $\alpha = 1$, $q = 5$, $h_{limit} = .008979$, $h/h_{limit} = .989$

$S_n(k)$ data:

| n | q-1 | q | q+1 |
|---|---|---|---|
| 149 | 6.370379026002e+02 | 4.855836384281e+01 | 5.953518464302e+01 |
| 150 | 1.876023089446e+02 | 5.597922189165e+01 | 1.416568269282e+02 |
| 151 | 9.246250495632e+02 | 1.471330276210e+02 | 4.919635341570e+01 |
| 152 | 6.910744968119e+02 | 5.673251338505e+01 | 4.764253375368e+01 |
| 153 | 1.547105532689e+02 | 4.374817040913e+01 | 1.355841534733e+02 |

Elimination among the three quartics..

| | | | | | |
|---|---|---|---|---|---|
| $Q_4$.. | -5.538266e+05 | 2.667789e+05 | 0. | 6.099607e+05 | -3.345347e+05 |
| reduced $Q_5$.. | | -7.413593e+03 | 0. | 1.480843e+03 | 5.640956e+03 |
| reduced $Q_6$.. | | 4.122677e+03 | 0. | 2.012189e+03 | -5.951487e+03 |
| reduced $Q_6$.. | | | 0. | 2.835682e+03 | -2.814569e+03 |

Root from elimination is.. $R = 0.9925546$

From Newton iteration 1.. $R = 0.9856203$

Estimated $c_0\Sigma_n(k)$: 5.907911e+02 8.424955e+01 7.964971e+01

Estimated $B$ = -0.04288 $R_B = 0.9790100$

From a separate calculation of the true root $z_1$.. $R_{true} = 0.9827322$

The errors in the two computed values of $R$ are both under 0.4%.

*Technical Information Department* • Lawrence Livermore National Laboratory
University of California • Livermore, California 94550