Round-Off Error Mitigation
In Solving Cubic and Quartic Equations

David J. Wolters

November 16, 2025

I. INTRODUCTION

Analytic techniques to mitigate computer round-off error are applied to algorithms for solving cubic and quartic equations. A computer, operating in double precision, usually produces a calculated-solution relative error on the order of 10^{-16} or less, but this small error value can increase by many orders of magnitude for certain conditions:

- multiplicity condition - Two or more equation solutions equal the same real value.
- magnitude condition - The absolute values (or magnitudes) of two equation solutions differ from each other by several orders of magnitude.
- symmetry condition – A quartic equation has a quartic polynomial P(Z) that is symmetric about some argument value Z_C: $P(Z_C + Z) = P(Z_C - Z)$.

Examples of these conditions in Table I are described below. Our design eliminates this error magnification by applying error mitigation to the Practical Algorithm for Solving the Cubic Equation and Modified Euler Algorithm for Solving the Quartic Equation. These two algorithms are provided below in Figures 1 and 2. They are described in detail at https://quarticequations.com/Cubic.pdf and https://quarticequations.com/Quartic.pdf.

Calculated-solution round-off error is caused by the computer's limited precision for real-number storage. The mitigation design presented here assumes standard double-precision computation for floating point numbers (binary64) [1], [2]. Of the number's 64 bits storage, one bit is the sign bit, 11 bits store the binary exponent, and 52 bits store the fraction. The mantissa storage of a non-zero real number consists of an implicit 1 followed by the binary point and 52 binary places. The maximum storage error is an incorrect least-significant bit. Thus, the maximum relative storage error for this format is

$$\varepsilon = 2^{-52} \approx 2.22\times10^{-16}.$$

The calculated solutions of most cubic and quartic equations have relative errors, if any, on the order of ε or less.

Solution error can be much greater for equations exhibiting the multiplicity, magnitude, or similarity conditions as demonstrated in the five example equations of Table I below. Examples 1 and 2 demonstrate the multiplicity condition. The cubic equation of Example 1 has two equal solutions (multiplicity 2) with relative solution error on the order of 10^{-8}. The quartic equation of Example 2 has three equal solutions (multiplicity 3) and relative error on the order of 10^{-5}. Example 3 demonstrates the symmetry condition. The quartic polynomial, and therefore, the four solutions 7, 4.2, –0.2 and –3, are symmetrical about the value $Z = Z_C = 2$. The solutions stay the same in Example 4 except that the third solution is changed very slightly from –0.2 to –0.2000001. Thus Example 4 is not symmetric, but it is a symmetry *near miss*. Example 3 and 4 relative error is on the order of 10^{-7}. Example 5 is an extreme example of the magnitude condition: the absolute values of the quartic equation's four solutions differ from each the other by many orders of magnitude. Round-off error swamps the two smallest-magnitude solutions so that their calculated values are worthless.

The mitigation design addresses all of these conditions of round-off error magnification, and calculates solutions for the tabulated equations with relative error less than 10^{-14}. With very rare exceptions for the quartic equation, any true real solution of a cubic or quartic equation is calculated as a real value, not a complex value with a small imaginary component.

Table I. Example Equations with Magnified Solution Error

1. The cubic equation $z_n^3 - 5\,z_n^2 + 8\,z_n - 4 = 0$ has true solutions 2, 2, and 1, but the calculated solutions are 1 and $2 \pm i\,1.676380642679 \times 10^{-8}$.
2. The quartic equation $Z_n^4 - 4.2\,Z_n^3 + 6.6\,Z_n^2 - 4.6\;Z_n + 1.2 = 0$ has true solutions 1.2, 1, 1, 1, but the calculated solutions are
   1.2, 0.999991545140, and $1.000004227430 \pm i\;0.000007322698$.
3. The quartic equation $Z_n^4 - 8\,Z_n^3 - 5.84\,Z_n^2 + 87.36\;Z_n + 17.64 = 0$ has true solutions 7, 4.2, –0.2 and –3, but the calculated solutions are
   7.000000042147, 4.199999957853 , –0.200000042147, and –2.999999957853.
4. The quartic equation $Z_n^4 - 7.9999999\,Z_n^3 - 5.84000082\,Z_n^2 + 87.35999958\;Z_n + 17.64000882 = 0$ has true solutions 7, 4.2, –0.2000001, and –3, but the calculated solutions are 7.000000017147, 4.199999982853, –0.200000117147, and –2.999999982853.
5. The quartic equation
   $Z_n^4 - 6.99970002\,Z_n^3 - 2.099860005965{\times}10^{-3}\,Z_n^2 + 4.20000104993{\times}10^{-11}\;Z_n - 2.1 \times 10^{-25} = 0$
   has true solutions
   7, $-3{\times}10^{-4}$, $2{\times}10^{-8}$, and $5{\times}10^{-15}$,
   but the calculated solutions are
   7, $-3.00019431496{\times}10^{-4}$ and
   $1.97157508097{\times}10^{-8} \pm i\,2.41435601527{\times}10^{-6}$.

Calculation results in this table are produced by coding the cubic- and quartic-equation algorithms (Figures 1, 2) in Excel 2016 Visual Basic for Applications (VBA) using double precision for floating-point numbers.

The mitigation design also addresses cubic and quartic equations with multiplicity near miss (two solutions are not equal, but are nearly equal).

Finally, the mitigation design addresses the special cases of cubic and quartic equation for which the constant coefficient is zero, which implies that at least one solution is zero. The algorithms in Figures 1 and 2 may return the zero solution as a small round-off error. Such a result for the resolvent cubic equation of a symmetric quartic equation produces magnified round-off error in the quartic-equation calculated solutions.

Summary

Our presentation of the mitigation design begins in Section II with a review of the Practical Algorithm for Solving the Cubic Equation and the Modified Euler Algorithm for Solving the Quartic Equation.

Section III describes cubic- and quartic-equation special cases that are incorporated into the round-off-error mitigation design. Solutions for these cases can be calculated more easily than by using the full cubic- and quartic-equation algorithms, and round-off error is also reduced. The set of special cases provides the preliminary logic structure in the form of algorithm flowcharts for the round-off error mitigation design. One of the special cases requires the solutions of a quadratic equation, so this section also introduces a quadratic-equation algorithm based on recommendations in *Numerical Recipes* [3, §5.6] by Press, et al. Unlike the quadratic formula, this quadratic-equation algorithm mitigates against round-off error when the absolute values of the two solutions differ by many orders of magnitude (magnitude condition).

Section IV addresses the multiplicity condition (two or more equation solutions equal the same real value). It shows how multiplicity magnifies solution round-off error, and then it eliminates the problem by introducing new calculations into the Section III algorithms. The algorithms for solving quadratic, cubic, and quartic equations take their final form. Demonstrations using Examples 1 and 2 from Table I above show how the mitigation calculations work. The mitigation technique for multiplicity also addresses the quartic-equation symmetry condition as demonstrated using Table I, Example 3.

The presentation also explains why the modified Euler algorithm was chosen from among the available quartic-equation algorithms for the round-off error mitigation design. The modified Euler algorithm reflects the conditions of quartic equation multiplicity and symmetry as a corresponding condition in the algorithm's resolvent cubic equation (Figure 11) and its three solutions. Euler's application of all three resolvent-cubic-equation solutions greatly simplifies the mitigation design for quartic equations.

Sections V and VI describe post-processing algorithms to address round-off-error magnification for the magnitude condition: cubic or quartic equations with at least two solutions whose absolute values differ by several orders of magnitude. The algorithms of Section IV provide good accuracy for the larger-magnitude solutions, but round-off error can swamp the smaller-magnitude solutions. To correct this situation, post processing applies the accurately-calculated, large-magnitude solution(s) to the cubic- or quartic-equation coefficients to accurately extract the small-magnitude solution(s).

The cubic-equation post processing, described in Section V, addresses not only the cubic-equation magnitude condition, but also quartic-equation symmetry near-miss. The reason is that such a quartic equation has a resolvent cubic equation with the magnitude condition. Section V works through the Table I, Example 4 quartic equation to demonstrate.

Section VI describes the quartic-equation post-processing algorithm, which it demonstrates using the Table I, Example 5 quartic equation. For its operation, the algorithm requires a generic value-ordering routine, which is also provided. Quartic-equation post processing is the final piece of the error mitigation design.

All calculations of the error-mitigation design for solving quadratic, cubic, and quartic equations are summarized in the figures listed below. The figures are found in Sections IV, V, and VI.

Table II. Calculations with Round-Off Error Mitigation for Solving Quadratic, Cubic, and Quartic Equations

Section #	Figure #	Page #	Title
IV	8	21	Final Calculation Algorithm for Solving the Quadratic Equation
IV	9	22	Final Calculation Algorithm for Solving the Cubic Equation
V	12	47	Cubic Equation Post Processing Algorithm
IV	10	28	Final Calculation Algorithm for Solving the Quartic Equation
VI	13	57	Quartic Equation Post Processing Algorithm
VI	14	59	Value-Ordering Routine

Sections VII through X provide an error analysis of the multiplicity and multiplicity near-miss conditions to show that the mitigation design provides excellent solution accuracy.

Unless noted otherwise, the radical sign √ denotes the principal square root. The principal square root of a positive real number is the positive square root. The principal square root of a negative real number is the positive imaginary square root. If z is complex with modulus r and argument ϕ such that $-\pi < \phi \leq \pi$, then $z = re^{i\phi}$ and the principal square root of z is $\sqrt{z} = \sqrt{r}\, e^{i\phi/2}$.

The following coding recommendations apply whenever calculation error is a concern.

- To calculate an integer power of a real number, use repeated multiplication rather than exponentiation. For example, code X^3 as X*X*X rather than X^3.
- To calculate an odd half power of a real number, use the square-root function rather than exponentiation. For example, code $X^{5/2}$ as X*X*SQRT(X) rather than X^(5/2).

I thank my correspondents Demetrius Papademetriou and Vadym Koliada, whose interest in the round-off error problem inspired my effort here.

See the website directory at https://quarticequations.com.

Contact the author at david@quarticequations.com.

II. REVIEW OF CUBIC- AND QUARTIC-EQUATION ALGORITHMS

This section reviews the starting algorithms prior to applying round-off error mitigation.

Review of the Cubic-Equation Algorithm

Figure 1 below shows the practical cubic-equation algorithm for solving the cubic equation

$$z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0, \qquad n = 1, 2, 3. \tag{1}$$

Solution z_1 is the greatest real solution. The other two solutions, $z_2 = x_2 + iy_2$ and $z_3 = x_3 - iy_2$, are either real ($y_2 = 0$) or a complex conjugate pair ($x_3 = x_2$). Given the equations three real coefficients a_2, a_1, and a_0, the algorithm calculates outputs z_1, x_2, x_3, and y_2 ($y_2 \geq 0$) so that z_1, $z_2 = x_2 + iy_2$, and $z_3 = x_3 - iy_2$ satisfy

$$z^3 + a_2 z^2 + a_1 z + a_0 \;=\; (z - z_1)\,(z - z_2)\,(z - z_3) \;\text{ for all z.}$$

Valid solutions z_1, $z_2 = x_2 + iy_2$, and $z_3 = x_3 - iy_2$ reproduce the input coefficients according to these check equations:

$$a_2 \;=\; -(z_1 + z_2 + z_3) \qquad a_1 \;=\; z_1z_2 + z_1z_3 + z_2z_3 \qquad a_0 \;=\; -z_1z_2z_3 \tag{2}$$

or

$$a_2 \;=\; -(z_1 + x_2 + x_3) \qquad a_1 \;=\; z_1(x_2+x_3) + x_2x_3 + y_2^2 \qquad a_0 \;=\; -z_1(x_2x_3+y_2^2) \tag{3}$$

Figure 1 Practical Algorithm for Solving the Cubic Equation

Given: Real coefficients a_2, a_1, and a_0, Find: z_1, $z_2=x_2+iy_2$, and $z_3=x_3-iy_2$ such that $z^3 + a_2 z^2 + a_1 z + a_0 \;=\; (z - z_1)\,(z - z_2)\,(z - z_3)$ for all z. Calculate q and r: $q = \frac{a_1}{3} - \frac{a_2^2}{9}$ $\quad r = \frac{a_1a_2 - 3a_0}{6} - \frac{a_2^3}{27}$		
Case: 1: $r^2 + q^3 > 0$ ⇔ Only One Real Solution (*Numerical Recipes*) $A = \left(\lvert r\rvert + \sqrt{r^2 + q^3}\right)^{1/3}$		**Case 2: $r^2 + q^3 \leq 0$ ⇔ Three Real Solutions (Viète)** $\theta = \begin{cases} 0 & \text{if } q = 0 \\ \mathrm{Cos}^{-1}\left(\mathrm{Max}\{\mathrm{Min}[r/(-q)^{3/2}, 1], -1\}\right) & \text{if } q < 0 \end{cases}$ Note: $0 \leq \theta \leq \pi$ $\phi_1 = \theta/3 \quad \phi_2 = \phi_1 - 2\pi/3 \quad \phi_3 = \phi_1 + 2\pi/3$ $t_1 = \; 2\sqrt{-q}\,\cos\phi_1$ $t_2 = t_{2x} = \; 2\sqrt{-q}\,\cos\phi_2 \qquad y_2 = 0$ $t_3 = t_{3x} = \; 2\sqrt{-q}\,\cos\phi_3$ Note: $t_1 \geq t_2 \geq t_3 \;\Rightarrow\; z_1 \geq z_2 = x_2 \geq z_3 = x_3$
for $r \geq 0$ $u = A, \quad v = -q/A$	for $r < 0$ $v = -A, \quad u = q/A$	
for $q \leq 0$, $\; t_1 = u + v$ (*Numerical Recipes*)	for $q > 0$, (La Porte) $t_1 = \dfrac{2r}{u^2 + v^2 + q}$	
$t_{2x} = t_{3x} = -t_1/2$ $t_2 = t_{2x} + iy_2,$	$y_2 = \dfrac{\sqrt{3}}{2}\left(A + \dfrac{q}{A}\right)$ $t_3 = t_{2x} - iy_2$	
$z_1 = t_1 - a_2/3,$	$x_2 = t_{2x} - a_2/3$	$x_3 = t_{3x} - a_2/3$ Note: $z_2 = x_2 + iy_2, \quad z_3 = x_3 - iy_2$

The algorithm converts the general cubic equation (1) to an equivalent *depressed cubic equation* with no quadratic term:

$$t_n^3 + 3q\,t_n - 2r = 0, \qquad n = 1, 2, 3. \tag{4}$$

The real values q and r are calculated from coefficients a_2, a_1, and a_0 as

$$q = \frac{a_1}{3} - \frac{a_2^2}{9} \qquad\qquad r = \frac{a_1 a_2 - 3a_0}{6} - \frac{a_2^3}{27} \tag{5}$$

The depressed solutions t_n in (4) are related to the general solutions z_n by

$$t_n = z_n + a_2/3 \quad \Leftrightarrow \quad z_n = t_n - a_2/3, \qquad n = 1, 2, 3. \tag{6}$$

The procedure to calculate the t_n depends on whether the cubic equation has only one real solution (Case 1, $r^2 + q^3 > 0$) or three real solutions (Case 2, $r^2 + q^3 \le 0$). The Case 1 calculation modifies Cardano's formula [4, Chapter XI] as suggested by Press, et al. in *Numerical Recipes* [3, §5.6]. If $q > 0$, however, the algorithm's final formula for t_1 is that of La Porte as described by Blinn [5, p14]. The algorithm applies Viète's trigonometric method [6] for cubic equations with three real solutions, Case 2. The last entry under Case 2 is

$$t_1 \ \ge\ t_2 \ \ge\ t_3 \quad\Rightarrow\quad z_1 \ \ge\ z_2 = x_2 \ \ge\ z_3 = x_3\,. \tag{7}$$

These inequalities are important to the mitigation design.

The algorithm above is expanded somewhat compared to the corresponding algorithm in this website's cubic-equation document https://quarticequations.com/Cubic.pdf. In the formula for θ in Case 2, that document gives the argument of Cos^{-1} as $r/(-q)^{3/2}$. That argument is theoretically bound to the range [–1, 1] by the definition of Case 2: $r^2 + q^3 \le 0$. In practice, however, round-off error may take the calculated value of $r/(-q)^{3/2}$ just outside this range and cause a run-time error in the Cos^{-1} calculation. The Figure 1 algorithm avoids this possibility by clamping the argument to the range [–1, 1] explicitly with the expanded expression $\text{Max}\{\text{Min}[r/(-q)^{3/2}, 1], -1\}$.

Also, the Figure 1 algorithm explicitly calculates the three solutions t_1, t_2, and t_3 of the depressed cubic equation (4). These depressed solutions are key to understanding solution error due to computer round off.

Review of the Modified Euler Quartic-Equation Algorithm

Figure 2 below shows the modified Euler quartic-equation algorithm. Inputs are four real coefficients A_3, A_2, A_1, and A_0, and the outputs are the four values Z_1, Z_2, Z_3 and Z_4 such that

$$Z^4 + A_3Z^3 + A_2Z^2 + A_1Z + A_0 \ = \ (Z-Z_1)\,(Z-Z_2)\,(Z-Z_3)\,(Z-Z_4) \text{ for all Z.}$$

The outputs are thus the four solutions of the general quartic equation

$$Z_n^4 + A_3Z_n^3 + A_2\,Z_n^2 + A_1\,Z_n + A_0 = 0, \qquad n = 1, 2, 3, 4. \tag{8}$$

Figure 2 Modified Euler Algorithm for Solving the Quartic Equation

Given: Real coefficients A_3, A_2, A_1, and A_0,

Find: Z_1, Z_2, Z_3 and Z_4 such that $Z^4 + A_3Z^3 + A_2Z^2 + A_1Z + A_0 = (Z–Z_1)(Z–Z_2)(Z–Z_3)(Z–Z_4)$ for all Z.

Calculation: $C = A_3 / 4$, $b_2 = A_2 – 6C^2$, $b_1 = A_1 – 2A_2C + 8C^3$, $b_0 = A_0 – A_1C + A_2C^2 – 3C^4$

Use the cubic-equation algorithm to find the three solutions z_1, z_2, and z_3 of the resolvent cubic equation:

$$z_k^3 + (b_2/2)\, z_k^2 + [(b_2^2 – 4b_0)/16]\, z_k – b_1^2/ 64 = 0.$$

Of the three cubic-equation solutions, z_1 is the greatest real solution and $z_1 \geq 0$. Solutions $z_2 = x_2 + iy_2$ and $z_3 = x_3 + iy_3$ are real ($z_2 = x_2$, $z_3 = x_3$, $y_2 = y_3 = 0$), or they form a complex conjugate pair ($z_2 = x_2 + iy_2$, $z_3 = x_2 – iy_2$, $x_2 = x_3$, $y_2 = –y_3 > 0$). In either case,

$$z_2 z_3 = x_2 x_3 + y_2^2 \geq 0 \quad \text{and} \quad x_2 x_3 \geq 0.$$

The calculation of z_1, z_2, and z_3 assures that if z_2 and z_3 are real, then $z_3 = x_3 \leq z_2 = x_2 \leq z_1$. To assure that round-off error does not cause a violation of $z_1 \geq 0$ and/or $x_2 x_3 \geq 0$, insert the following calculation logic:

If $z_1 < 0$, then $z_1 = 0$. If $x_2 x_3 < 0$, then (If $x_2 > –x_3$, then $x_3 = 0$; else $x_2 = 0$.)

$\Sigma = 1$ if $b_1 > 0$, $\Sigma = –1$ otherwise.

$D_1 = x_2 + x_3 - 2\Sigma\sqrt{x_2x_3 + y^2}$ $D_3 = x_2 + x_3 + 2\Sigma\sqrt{x_2x_3 + y^2}$ $R_1 = \sqrt{|D_1|}$ $R_3 = \sqrt{|D_3|}$

Case $D_1 \geq 0$:	Case $D_1 < 0$:	Case $D_3 \geq 0$:	Case $D_3 < 0$:
$T_{1X} = \sqrt{z_1} + R_1$ $T_{2X} = \sqrt{z_1} - R_1$ $Y_1 = 0$	$T_{1X} = T_{2X} = \sqrt{z_1}$ $Y_1 = R_1$	$T_{3X} = -\sqrt{z_1} + R_3$ $T_{4X} = -\sqrt{z_1} - R_3$ $Y_3 = 0$	$T_{3X} = T_{4X} = -\sqrt{z_1}$ $Y_3 = R_3$
$X_1 = T_{1X} – C$	$X_2 = T_{2X} – C$	$X_3 = T_{3X} – C$	$X_4 = T_{4X} – C$

Note: $T_1 = T_{1X} + iY_1$, $T_2 = T_{2X} – iY_1$ $T_3 = T_{3X} + iY_3$, $T_4 = T_{4X} – iY_3$

$Z_1 = T_1 – C = X_1 + iY_1$, $Z_2 = T_2 – C = X_2 – Y_1$,

$Z_3 = T_3 – C = X_3 + iY_3$, $Z_4 = T_4 – C = X_4 – Y_3$

The algorithm begins by calculating $C = A_3/4$, b_2, b_1, and b_0. The last three of these values are coefficients of the equivalent *depressed quartic equation* with no cubic term:

$$T_n^4 + b_2T_n^2 + b_1T_n + b_0 = 0 \qquad n = 1, 2, 3, 4. \qquad (9)$$

The solutions Z_n of (8) are related to the solutions T_n of (9) by

$$T_n = Z_n + C \quad \Leftrightarrow \quad Z_n = T_n – C, \qquad n = 1, 2, 3, 4. \qquad (10)$$

The coefficients a_2, a_1, and a_0 of the resolvent cubic equation are calculated from the depressed quartic equation coefficients b_n as:

$$a_2 = b_2/2 \tag{11}$$

$$a_1 = (b_2^2 - 4b_0)/16 \tag{12}$$

$$a_0 = -b_1^2/64. \tag{13}$$

The cubic-equation algorithm calculates the solutions z_1, z_2, and z_3 of the resolvent cubic equation, and from them the quartic-equation algorithm calculates the depressed quartic-equation solutions $T_n = T_{nX} + i\,Y_n$ and the general solutions $Z_n = X_n + i\,Y_n$ where $X_n = T_{nX} - C$.

The depressed solutions T_n are calculated as two pairs: T_1, T_2 and T_3, T_4. Solutions T_1 and T_2 are either both real:

$$T_1 = T_{1X}, \quad T_2 = T_{2x}, \quad Y_1 = Y_2 = 0 \quad \Rightarrow \quad Z_1 = X_1 = T_{1X} - C, \quad Z_2 = X_2 = T_{2X} - C,$$

or they form a complex conjugate pair:

$$T_{1X} = T_{2x}, \quad Y_2 = -Y_1 > 0, \quad T_1 = T_{1X} + i\,Y_1, \quad T_2 = T_{1X} - i\,Y_1 \quad \Rightarrow$$

$$X_1 = X_2 = T_{1X} - C, \quad Z_1 = X_1 + i\,Y_1, \quad Z_2 = X_1 - i\,Y_1.$$

In similar fashion, T_3, T_4 are either both real, or they form a complex conjugate pair. The pair Z_3, Z_4 are both real or a complex conjugate pair accordingly.

Valid solutions Z_1, Z_2, Z_3 and Z_4 of the quartic equation reproduce the input coefficients in compliance with the following check equations:

$$A_3 = -(Z_1+Z_2+Z_3+Z_4) \tag{14}$$

$$A_2 = Z_1Z_2+Z_1Z_3+Z_1Z_4+Z_2Z_3+Z_2Z_4+Z_3Z_4 \tag{15}$$

$$A_1 = -(Z_1Z_2Z_3+Z_1Z_2Z_4+Z_1Z_3Z_4+Z_2Z_3Z_4) \tag{16}$$

$$A_0 = Z_1Z_2Z_3Z_4\,. \tag{17}$$

OR

$$A_3 = -(X_1+X_2+X_3+X_4) \tag{18}$$

$$A_2 = X_1X_2+Y_1^2+(X_1+X_2)(X_3+X_4)+X_3X_4+Y_3^2 \tag{19}$$

$$A_1 = -[(X_1X_2+Y_1^2)(X_3+X_4) + (X_3X_4+Y_3^2)(X_1+X_2)] \tag{20}$$

$$A_0 = (X_1X_2+Y_1^2)(X_3X_4+Y_3^2). \tag{21}$$

Figure 2 lists some important inequality relationships among solutions z_1, z_2, and z_3 of the resolvent cubic equation. When all three solutions are real, the cubic-equation algorithm assures that the calculated solutions obey $z_3 = x_3 \le z_2 = x_2 \le z_1$. To assure that round-off error does not cause a violation of $z_1 \ge 0$ and/or $x_2\,x_3 \ge 0$, the algorithm inserts the following calculation logic:

```
If $z_1 < 0$, then $z_1 = 0$.
If $x_2 x_3 < 0$, then
    if $x_2 > -x_3$, then $x_3 = 0$
    else $x_2 = 0$.
```

This last logic is omitted in the more-compact version of the algorithm in *Practical Algorithms for Solving the Quartic Equation*, https://quarticequations.com/Quartic.pdf.

Another addition in the Figure 2 algorithm is the express calculation of the real and imaginary parts of solutions T_n of the depressed quartic equation (9).

III. DEFINITIONS OF THE SPECIAL CASES

This section describes cubic- and quartic-equation special cases, Figure 3, that are incorporated into the round-off-error mitigation design. Solutions for these cases can be calculated more easily than by using the full cubic- and quartic-equation algorithms, and round-off error is also reduced. The set of special cases provides the preliminary logic structure for the round-off error mitigation design.

Figure 3 Cubic- and Quartic-Equation Special Cases

Cubic-Equation Special Cases		
Case #	**Case Definition**	**Solutions z_n of the General Cubic Equation** $z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0$
1	$a_0 = 0$ At least one solution z_n is zero.	0 and the two solutions of $z_n^2 + a_2 z_n + a_1 = 0$
		Solutions t_n of the Depressed Cubic Equation $t_n^3 + 3q\,t_n - 2r = 0$
2	$q = r = 0$ All three solutions z_n equal the same real value (multiplicity 3 condition).	$t_1 = t_2 = t_3 = 0$
3	$R \equiv r^2 + q^3 = 0,\ r \neq 0$ Two of the solutions z_n equal the same real value (multiplicity 2 condition).	(a) $t_1 = 2\sqrt{-q}$, $t_2 = t_3 = -\sqrt{-q}$ if $r > 0$ (b) $t_1 = t_2 = \sqrt{-q}$, $t_3 = -2\sqrt{-q}$ if $r < 0$
4	$r = 0,\ q \neq 0$ (a) The three z_n have equal real parts if $q > 0$. (b) Three real z_n are evenly distributed if $q < 0$.	(a) $t_1 = 0$, $t_2 = -t_3 = i\sqrt{3q}$ if $q > 0$ (b) $t_1 = \sqrt{-3q}$, $t_2 = 0$, $t_3 = -\sqrt{-3q}$, if $q < 0$
Quartic-Equation Special Case		
5	$A_0 = 0$	Solution $Z_1 = 0$. Solutions Z_2, Z_3, and Z_4 are the solutions z_1, z_2, and z_3 respectively of the cubic equation $z_n^3 + A_3 z_n^2 + A_2 z_n + A_1 = 0$.
6	$b_0 = 0$	Depressed solution $T_1 = 0$. Solutions T_2, T_3, and T_4 are the solutions z_1, z_2, and z_3 respectively of the cubic equation $z_n^3 + b_2 z_n + b_1 = 0$.

Each special case corresponds to some parameter having a value of zero. The most obvious cases are cubic, quartic, and depressed equations whose constant coefficient is zero.

Cases 1, 5 and 6: $a_0 = 0$, $A_0 = 0$, and $b_0 = 0$

In a cubic equation with Case 1 ($a_0 = 0$) the left side of the cubic equation factors:

$$z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = z_n^3 + a_2 z_n^2 + a_1 z_n = (z_n^2 + a_2 z_n + a_1)\, z_n = 0. \qquad (a_0 = 0)$$

One solution is 0, and the other two are solutions of the quadratic equation $z_n^2 + a_2 z_n + a_1 = 0$. When the quadratic-equation solutions are real, we avoid the quadratic formula because it unnecessarily introduces round-off error into the solution of smaller magnitude. Instead, our design uses a quadratic-equation algorithm, described shortly, based on *Numerical Recipes* [3, §5.6] by Press, et al.

The Case 5 quartic equation ($A_0 = 0$) has solution Z_1 equal to 0, and solutions Z_2, Z_3, and Z_4 are the solutions z_1, z_2, and z_3 respectively of the cubic equation $z_n^3 + A_3 z_n^2 + A_2 z_n + A_1 = 0$.

The Case 6 quartic equation ($b_0 = 0$) has depressed solution T_1 equal to 0, and depressed solutions T_2, T_3, and T_4 are the solutions z_1, z_2, and z_3 respectively of the cubic equation $z_n^3 + b_2 z_n + b_1 = 0$.

Case 2: $q = r = 0 \Leftrightarrow$ all three solutions equal the same real value

If $q = r = 0$, then the depressed cubic equation (4) reduces to $t_n^3 = 0$. Then Equation (6) gives

$$t_1 = t_2 = t_3 = 0 \quad \text{and} \quad z_1 = z_2 = z_3 = -a_2 / 3 \qquad (q = r = 0) \tag{22}$$

Case 3: $R \equiv r^2 + q^3 = 0,\ r \neq 0 \Leftrightarrow$ two solutions equal the same real value

This case implies that $q < 0$, $r^2 = -q^3 = (-q)^3$, and $|r/(-q)^{3/2}| = 1$. The sign of $r/(-q)^{3/2}$ is the same as the sign of r. If $r > 0$, then $r/(-q)^{3/2} = 1$, and the Figure 1 cubic-equation algorithm, Case 2, shows that $\theta = \text{Cos}^{-1}(1) = 0$, $\phi_1 = 0$, $\phi_2 = -2\pi/3$, $\phi_3 = 2\pi/3$, and

$$t_1 = 2\sqrt{-q} \quad \text{and} \quad t_2 = t_3 = -\sqrt{-q} \qquad (R \equiv r^2 + q^3 = 0, r > 0). \tag{23}$$

If $r < 0$, then $r/(-q)^{3/2} = -1$, and the calculation becomes $\theta = \text{Cos}^{-1}(-1) = \pi$, $\phi_1 = \pi/3$, $\phi_2 = -\pi/3$, $\phi_3 = \pi$, and

$$t_1 = t_2 = \sqrt{-q} \quad \text{and} \quad t_3 = -2\sqrt{-q} \qquad (R \equiv r^2 + q^3 = 0, r < 0). \tag{24}$$

Whether r is positive or negative, all three solutions are real, and two of them equal the same real value.

Case 4: $r = 0,\ q \neq 0$

If $r = 0$, then the depressed cubic equation (4) reduces to $t_n^3 + 3q\, t_n = (t_n^2 + 3q)\, t_n = 0$. The solutions for t_n are 0 and $\pm\sqrt{-3q}$.

$$t_1 = 0, \qquad t_2 = i\sqrt{3q}, \qquad t_3 = -i\sqrt{3q} \qquad (r = 0,\ q > 0) \tag{25}$$

$$t_1 = \sqrt{-3q}, \qquad t_2 = 0, \qquad t_3 = -\sqrt{-3q}, \qquad (r = 0,\ q < 0) \tag{26}$$

Preliminary Logic Structure for the Round-Off Error Mitigation Design

The preliminary logic structure for the mitigation design combines the Numerical Recipes quadratic-equation method [3, §5.6], the Figure 1 cubic-equation algorithm, and the Figure 2 quartic-equation algorithm with the Figure 3 special cases.

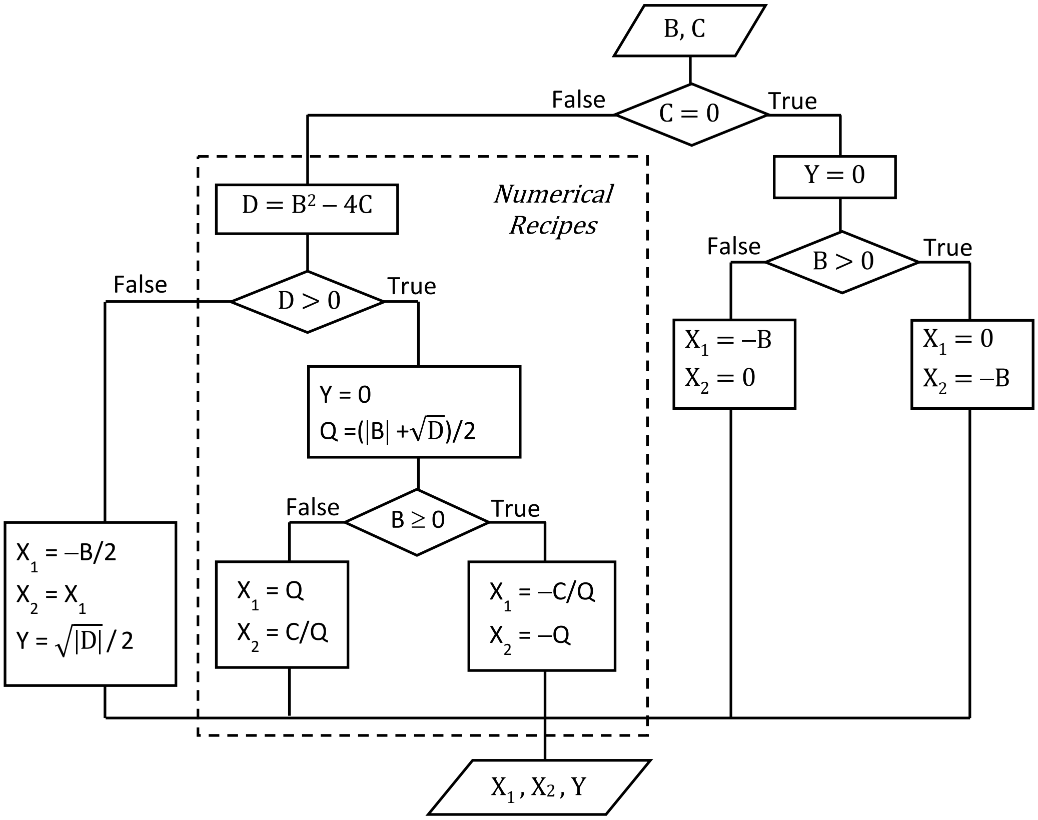
Quadratic-Equation Algorithm

We start with the quadratic-equation algorithm, Figure 4, which is needed by the cubic-equation algorithm for Special Case 1 ($a_0 = 0$) and by post processing for both cubic and quartic equations. The two solutions $Z_1 = X_1 + iY$ and $Z_2 = X_2 - iY$ of the quadratic equation $Z_n^2 + B\,Z_n + C = 0$ are calculated so that Z_1 is the greater of two real solutions or has the positive imaginary part when solutions are a complex conjugate pair. This convention simplifies the calculation logic in the cubic- and quartic-equation algorithms. The quadratic equation has its own special case when the constant coefficient C is 0. Then solution $Z_1 = X_1$ is the greater of the two solutions, 0 and –B.

Figure 4 Preliminary Calculation Logic for Solving the Quadratic Equation

Inputs: Real coefficients B and C of the quadratic equation $Z_n^2 + B\,Z_n + C = 0$

Outputs: X_1, X_2, Y so that $Z_1 = X_1 + iY$ and $Z_2 = X_2 - iY$ imply that $Z^2 + BZ + C = (Z - Z_1)(Z - Z_2)$ for all Z. Thus, Z_1 and Z_2 are the two solutions of the quadratic equation. By convention, solution Z_1 is the greater of two real solutions or has the positive imaginary part when solutions are a complex conjugate pair.



If $C \neq 0$, then the algorithm calculates the discriminate $D \equiv B^2 - 4C$. The quadratic formula is used only for $D \leq 0$.

QUADRATIC FORMULA

$$Z_1 = \frac{1}{2}\left(-B + \sqrt{D}\right) \qquad Z_2 = \frac{1}{2}\left(-B - \sqrt{D}\right), \qquad D \equiv B^2 - 4C \tag{27}$$

If $D > 0$, the formula is avoided because it unnecessarily introduces round-off error into the solution of smaller magnitude. Let X_A and X_B be the solutions of greater and smaller magnitude respectively. Then.

$$|X_A| = Q \equiv \frac{1}{2}\left(|B| + \sqrt{D}\right) \quad > \quad |X_B| = \frac{1}{2}\left||B| - \sqrt{D}\right| \tag{28}$$

If $B^2 >> |4C| > 0$, then $\sqrt{D} = \sqrt{B^2 - 4C} \approx |B|$. The calculated difference $|B| - \sqrt{D}$ and resulting X_B become less accurate as $|4C| / B^2$ decreases.

The algorithm's Numerical Recipes approach for $D > 0$ avoids this problem by using the relationships $B = -(Z_1 + Z_2) = -(X_1 + X_2) = -(X_A + X_B)$ and $C = Z_1 Z_2 = X_1 X_2 = X_A X_B$. If $B < 0$, then

$$X_1 = X_A = \frac{1}{2}\left(-B + \sqrt{D}\right) = \frac{1}{2}\left(|B| + \sqrt{D}\right) = Q \text{ and } X_2 = X_B = C/X_A = C/Q \quad (B < 0). \tag{29}$$

Otherwise,

$$X_2 = X_A = \frac{1}{2}\left(-B - \sqrt{D}\right) = -\frac{1}{2}\left(|B| + \sqrt{D}\right) = -Q \text{ and } X_1 = X_B = C/X_A = -C/Q \quad (B \geq 0). \tag{30}$$

Cubic-Equation and Quartic-Equation Algorithms

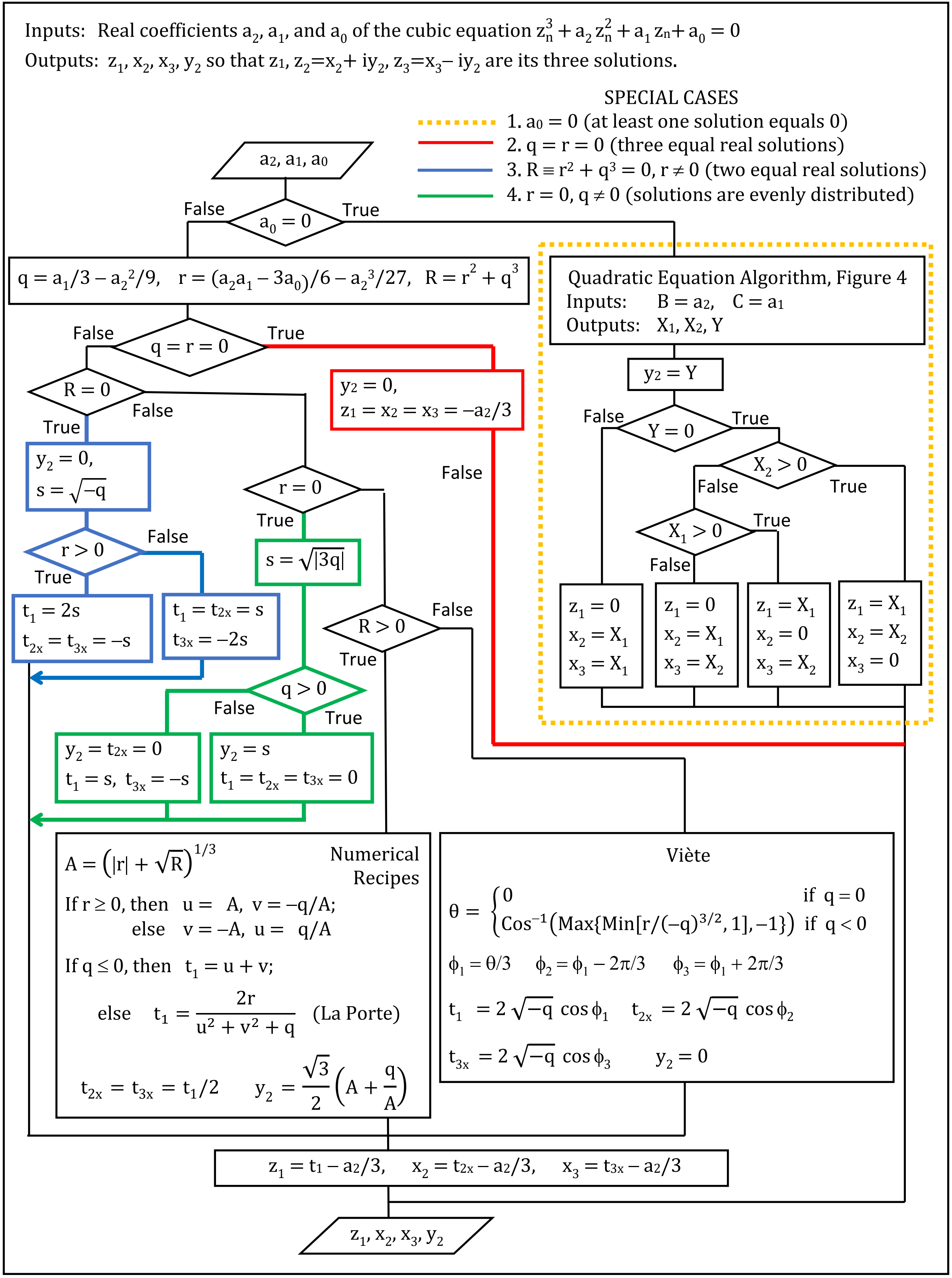
Figures 5 and 6 show the preliminary logic structure for the cubic-equation and quartic-equation algorithms. Figure 5 combines the Figure 1 cubic-equation algorithm with Figure 3 special cases 1 to 4. Figure 6 combines the Figure 2 quartic-equation algorithm with special cases 5 and 6.

Cubic-Equation Special Case 1 ($a_0 = 0$), shown in the Figure 5 dotted yellow box, requires some extra logic. One solution is zero and the other two are the solutions $Z_1 = X_1 + iY$ and $Z_2 = X_2 - iY$ of the quadratic equation $Z_n^2 + a_2 Z_n + a_1 = 0$. The quadradic-equation algorithm assures that $X_1 \geq X_2$. If the quadratic-equation solutions are complex conjugates, then the cubic-equation solutions are assigned $z_1 = 0$, $z_2 = Z_1$, and $z_3 = Z_2$. Otherwise, z_1, z_2, and z_3 are all real, and the extra logic assigns their values to comply with the convention:

$$z_1 \;\geq\; z_2 = x_2 \;\geq\; z_3 = x_3.$$

The following two sections add error-mitigation calculations to the quadratic-, cubic-, and quartic-equation algorithms of Figures 4, 5, and 6. However, the overall logic structure of the algorithms remains unchanged.

Figure 5 Preliminary Calculation Logic for Solving the Cubic Equation



Inputs: Real coefficients a_2, a_1, and a_0 of the cubic equation $z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0$

Outputs: z_1, x_2, x_3, y_2 so that z_1, $z_2 = x_2 + iy_2$, $z_3 = x_3 - iy_2$ are its three solutions.

SPECIAL CASES
1. $a_0 = 0$ (at least one solution equals 0)
2. $q = r = 0$ (three equal real solutions)
3. $R \equiv r^2 + q^3 = 0$, $r \neq 0$ (two equal real solutions)
4. $r = 0$, $q \neq 0$ (solutions are evenly distributed)
a_2, a_1, a_0
$a_0 = 0$
False
True
$q = a_1/3 - a_2^2/9, \quad r = (a_2 a_1 - 3a_0)/6 - a_2^3/27, \quad R = r^2 + q^3$
Quadratic Equation Algorithm, Figure 4
Inputs: $B = a_2$, $C = a_1$
Outputs: X_1, X_2, Y
$q = r = 0$
False
True
$y_2 = 0,$
$z_1 = x_2 = x_3 = -a_2/3$
$R = 0$
False
True
$y_2 = 0,$
$s = \sqrt{-q}$
$r > 0$
False
True
$t_1 = 2s$
$t_{2x} = t_{3x} = -s$
$t_1 = t_{2x} = s$
$t_{3x} = -2s$
$r = 0$
True
$s = \sqrt{|3q|}$
False
$R > 0$
False
True
$q > 0$
False
True
$y_2 = t_{2x} = 0$
$t_1 = s, \; t_{3x} = -s$
$y_2 = s$
$t_1 = t_{2x} = t_{3x} = 0$
$y_2 = Y$
$Y = 0$
False
True
$X_2 > 0$
False
True
$X_1 > 0$
True
False
$z_1 = 0$
$x_2 = X_1$
$x_3 = X_1$
$z_1 = 0$
$x_2 = X_1$
$x_3 = X_2$
$z_1 = X_1$
$x_2 = 0$
$x_3 = X_2$
$z_1 = X_1$
$x_2 = X_2$
$x_3 = 0$
Numerical Recipes
$A = \left(|r| + \sqrt{R}\right)^{1/3}$
If $r \geq 0$, then $u = A$, $v = -q/A$;
else $v = -A$, $u = q/A$
If $q \leq 0$, then $t_1 = u + v$;
else $t_1 = \dfrac{2r}{u^2 + v^2 + q}$ (La Porte)
$t_{2x} = t_{3x} = t_1/2 \qquad y_2 = \dfrac{\sqrt{3}}{2}\left(A + \dfrac{q}{A}\right)$
Viète
$\theta = \begin{cases} 0 & \text{if } q = 0 \\ \mathrm{Cos}^{-1}\left(\mathrm{Max}\{\mathrm{Min}[r/(-q)^{3/2}, 1], -1\}\right) & \text{if } q < 0 \end{cases}$
$\phi_1 = \theta/3 \quad \phi_2 = \phi_1 - 2\pi/3 \quad \phi_3 = \phi_1 + 2\pi/3$
$t_1 = 2\sqrt{-q}\cos\phi_1 \quad t_{2x} = 2\sqrt{-q}\cos\phi_2$
$t_{3x} = 2\sqrt{-q}\cos\phi_3 \quad y_2 = 0$
$z_1 = t_1 - a_2/3, \quad x_2 = t_{2x} - a_2/3, \quad x_3 = t_{3x} - a_2/3$
z_1, x_2, x_3, y_2

Figure 6 Preliminary Calculation Logic for Solving the Quartic Equation

Inputs: Real coefficients A_3, A_2, A_1, and A_0 of the quartic equation $Z_n^4 + A_3 Z_n^3 + A_2 Z_n^2 + A_1 Z_n + A_0 = 0$

Outputs: X_1, X_2, Y_1, X_3, X_4, Y_3 so that $Z_1=X_1+iY_1$, $Z_2=X_2-iY_1$, $Z_3=X_3+iY_3$, $Z_4=X_4-iY_3$ are its four solutions.

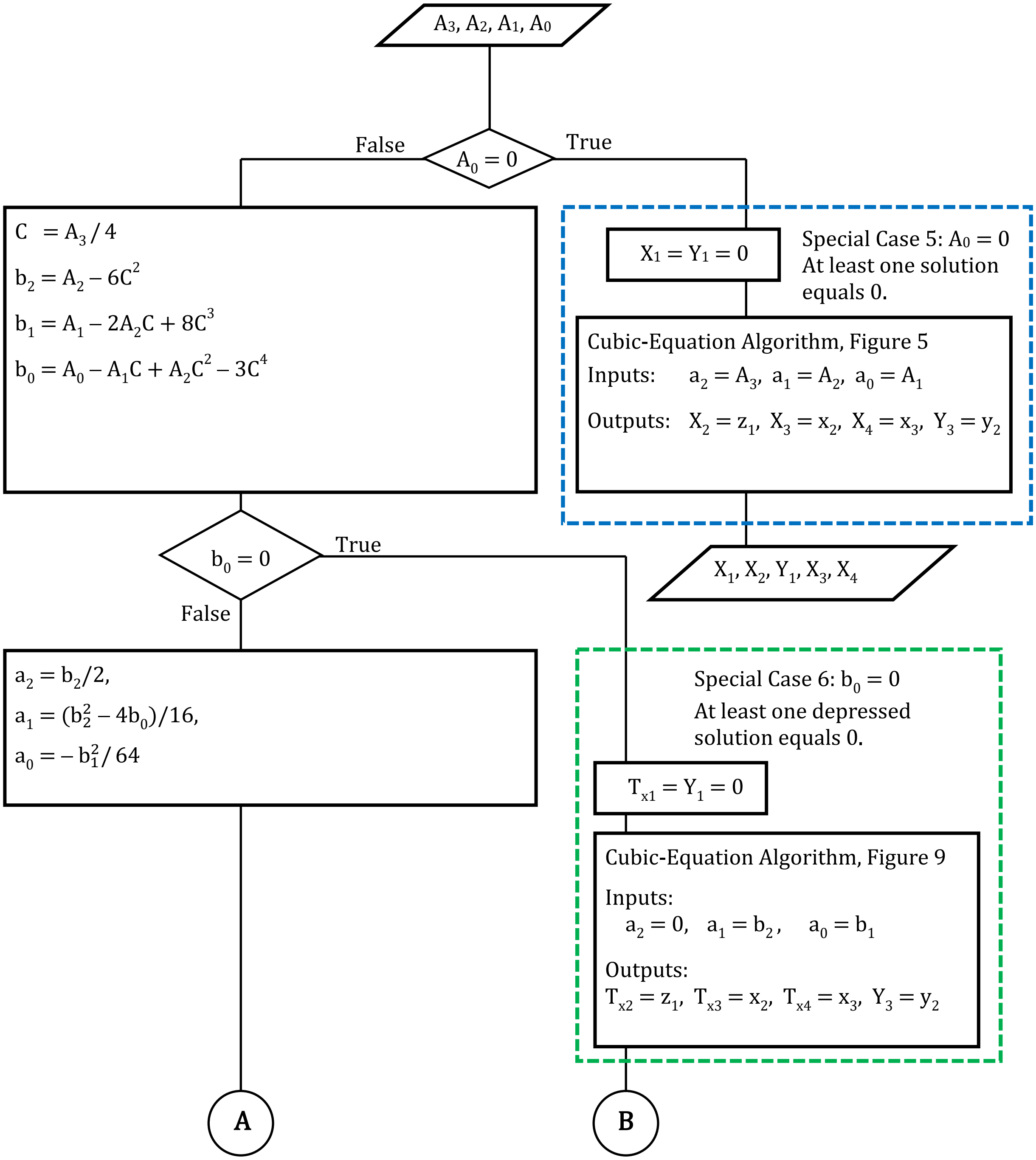
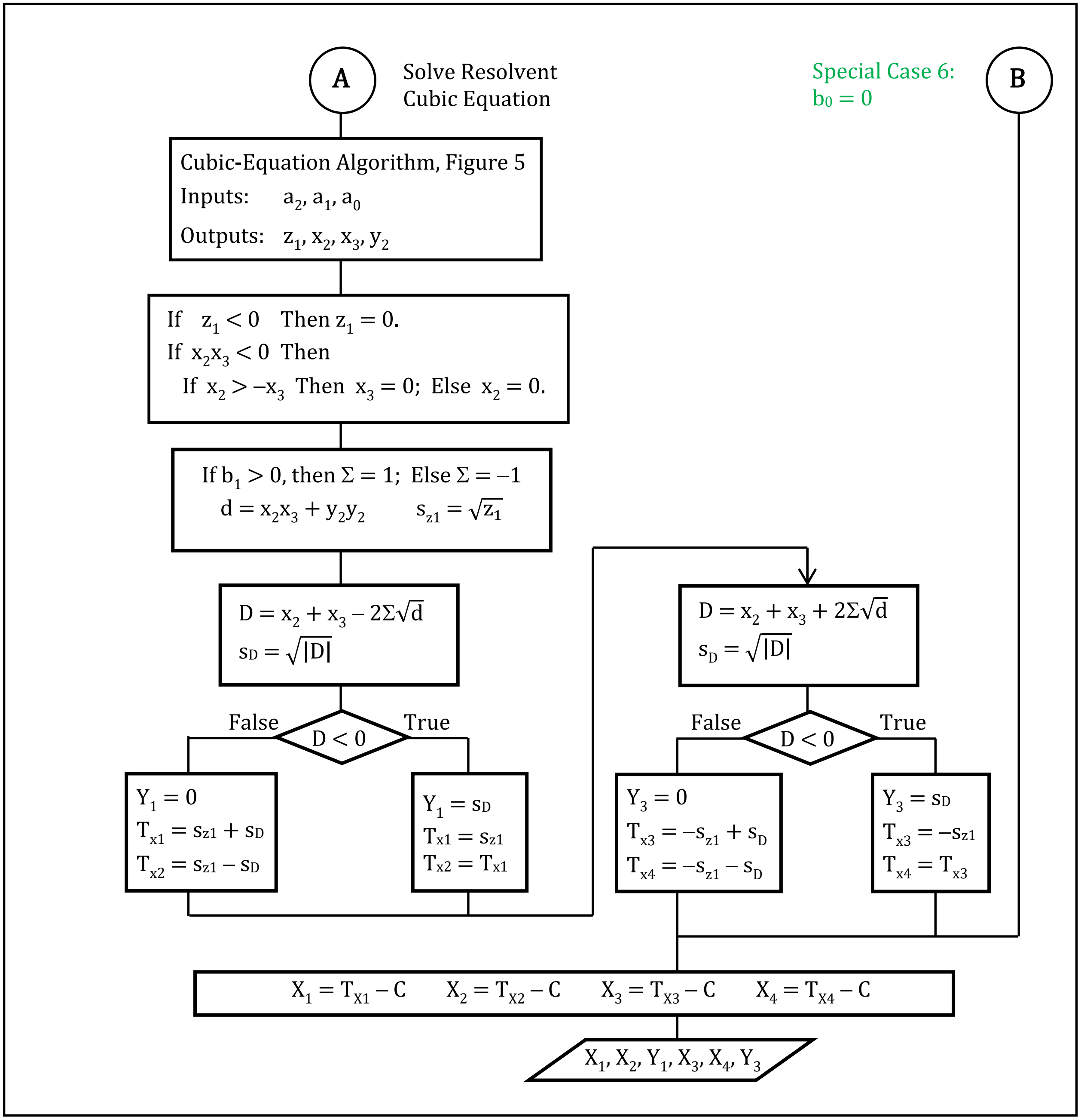

A_3, A_2, A_1, A_0
False
$A_0 = 0$
True
$C = A_3 / 4$
$b_2 = A_2 - 6C^2$
$b_1 = A_1 - 2A_2C + 8C^3$
$b_0 = A_0 - A_1C + A_2C^2 - 3C^4$
$X_1 = Y_1 = 0$
Special Case 5: $A_0 = 0$
At least one solution equals 0.
Cubic-Equation Algorithm, Figure 5
Inputs: $a_2 = A_3$, $a_1 = A_2$, $a_0 = A_1$
Outputs: $X_2 = z_1$, $X_3 = x_2$, $X_4 = x_3$, $Y_3 = y_2$
$b_0 = 0$
True
False
X_1, X_2, Y_1, X_3, X_4
$a_2 = b_2/2$,
$a_1 = (b_2^2 - 4b_0)/16$,
$a_0 = - b_1^2 / 64$
Special Case 6: $b_0 = 0$
At least one depressed solution equals 0.
$T_{x1} = Y_1 = 0$
Cubic-Equation Algorithm, Figure 9
Inputs:
$a_2 = 0$, $a_1 = b_2$, $a_0 = b_1$
Outputs:
$T_{x2} = z_1$, $T_{x3} = x_2$, $T_{x4} = x_3$, $Y_3 = y_2$
A
B

Figure 6 Preliminary Calculation Logic for Solving the Quartic Equation (Page 2)



IV. ALGORITHMS WITH MITIGATION FOR MULTIPLICITY CONDITION

This section addresses the multiplicity condition (two or more equation solutions equal the same real value). It shows how multiplicity magnifies solution round-off error, and then it eliminates the problem by introducing new calculations into the Section III algorithms. The algorithms for solving quadratic, cubic, and quartic equations take their final form. Demonstrations using Examples 1 and 2 from Table I above show how the mitigation calculations work. The mitigation technique for multiplicity also addresses the quartic-equation symmetry condition as demonstrated using Table I, Example 3. The presentation also explains why the Equation (round-off error mitigation design for quartic equations is based on the modified Euler algorithm rather than an alternative quartic-equation algorithm.

Figure 7 shows how multiplicity magnifies the solution round-off error. Figures 7-1 and 7-2 plot the cubic and quartic functions for the two example multiplicity equations in Table I, Section I. The function intersects the horizontal-axis at solution values. If the function has zero slope (zero first derivative) at the intersection point, then the solution is a multiple solution. At the intersection, if the function has both a zero slope and a point of inflection as in Figure 7-2, then the first two derivatives are zero, and the multiplicity is at least 3.

For any of the figure's multiple solution values, any small error in the function's vertical position produces a much greater error in the intersection location. That is, a small round-off error in the function value produces a magnified solution error.

The error analysis beginning in Section VII shows that magnification of residual solution error is an inherent feature of multiplicity. Calculated solutions for most simple solutions have maximum relative errors on the order of $\varepsilon = 2^{-52} \approx 2.22\times10^{-16}$, the computer's maximum relative storage error. However, maximum relative error is on the order of $\varepsilon^{1/2} \sim 10^{-8}$ for multiplicity 2 solutions and the order of $\varepsilon^{1/3} \sim 10^{-5}$ for multiplicity 3 solutions. See Examples 1 and 2 in Table I of Section I.

Our approach to mitigating this type of error magnification is to anticipate and accommodate the multiplicity condition.

Section III above has already described the first major feature of our error-mitigation design: incorporating logic for the Figure 3 special cases into our solution-calculation algorithms. Each special case is defined by a zero value for some calculated parameter. A cubic equation with $q = r = 0$ indicates a multiplicity of 3. Multiplicity 3 quartic equations, like Example 2, have a multiplicity 3 resolvent cubic equation. A cubic equation with $q < 0$ and $R \equiv r^2 + q^3 = 0$ has a multiplicity of 2.

This section adds the remaining major feature to mitigate multiplicity error magnification. For each of the special-case, zero-value parameters (e.g. r, q, R), we calculate a reasonable upper bound for its round-off error. The parameter R, for example, will have an upper-bound error $R_E\,\varepsilon$, where R_E is described below. The R_E value satisfies $R_E \geq |R|$. The computer's epsilon value $\varepsilon = 2^{-52}$ is stored as a universal constant of the mitigation design. If the absolute value of the calculated parameter is less than the upper-bound error, then the parameter is reset to zero. For example, if $|R| < R_E\,\varepsilon$, then R is reset to 0.

This approach assures that if multiple true solutions equal the same real value, then the corresponding calculated values also equal a common real value. Any residual round-off error in the calculated equal solutions is of the order of ε, not $\varepsilon^{1/2}$ or $\varepsilon^{1/3}$.

Figure 7 Cubic and Quartic Functions for Examples 1 and 2

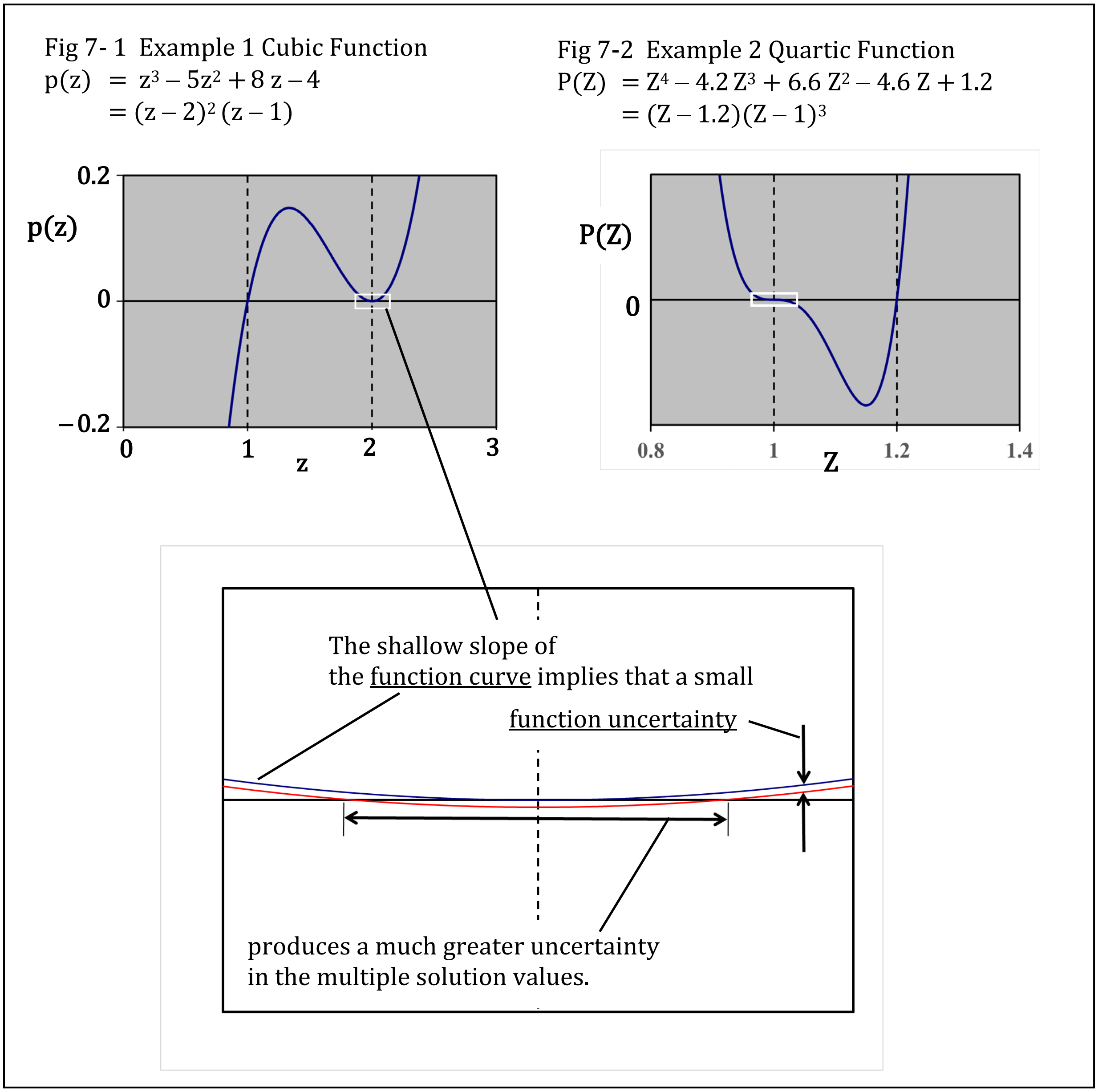


Fig 7- 1 Example 1 Cubic Function

$$p(z) = z^3 - 5z^2 + 8\,z - 4 = (z-2)^2\,(z-1)$$

Fig 7-2 Example 2 Quartic Function

$$P(Z) = Z^4 - 4.2\,Z^3 + 6.6\,Z^2 - 4.6\,Z + 1.2 = (Z-1.2)(Z-1)^3$$

The shallow slope of the function curve implies that a small function uncertainty produces a much greater uncertainty in the multiple solution values.

Multiplicity Error Mitigation in the Quadratic-Equation Algorithm

The mitigation design adds some simple calculations into the preliminary quadratic-equation algorithm, Figure 4, to address round-off error magnification for the multiplicity condition. The equation $Z_n^2 + B\,Z_n + C = 0$ has solutions $(-B \pm \sqrt{D})/2$ where D is the discriminate $D \equiv B^2 - 4C = 0$. When $D = 0$, then the equation has two equal solutions $Z_1 = Z_2 = X_1 = X_2 = -B/2$.

Suppose the true D value is zero, but D is calculated as a round-off error of $\delta D = \pm B^2 \times 10^{-16}$. Then the calculated solutions become

$$-(B/2)(1 \pm \sqrt{\pm 10^{-16}}) = \quad -(B/2)(1 \pm 10^{-8}) \quad \text{OR} \quad -(B/2)(1 \pm i\,10^{-8})$$

depending on the sign of δD. For the multiplicity condition, the discriminants relative error of 10^{-16} produces the magnified error of 10^{-8} in the solution.

The mitigation design addresses this situation by calculating a reasonable upper bound for the round-off error |δD| in D. Because D is the calculated value $D = B^2 - 4C$, we model the error δD as a function of the error δB in B and error δC in C by using the partial derivatives $\partial D/\partial B = 2B$ and $\partial D/\partial C = -4$:

$$\delta D \;\; = \frac{\partial D}{\partial B}\delta B + \frac{\partial D}{\partial C}\delta C = 2B\,\delta B - 4\,\delta C\,.$$

The true error contributions of B and C may either reinforce or cancel each other depending on their signs. For this purpose, we want an upper bound of |δD|, so we take the error contributions from B and C as reinforcing each other.

$$|\delta D|_{\max} \;\; = \left|\frac{\partial D}{\partial B}\right| |\delta B|_{\max} + \left|\frac{\partial D}{\partial C}\right| |\delta C|_{\max} = 2|B|\,|\delta B|_{\max} + 4\,|\delta C|_{\max}$$

Coefficients B and C may be supplied by the user to solve a quadratic equation, or they may be supplied by the cubic-equation or quartic-equation algorithm or a post-processing algorithm. For now, assume that B and C are user inputs. Then the worst-case errors $|\delta B|_{\max}$ and $|\delta C|_{\max}$ are just the computer's one-bit storage errors for B and C. That is,

$$|\delta B|_{\max} = |B|\varepsilon \;\text{ and }\; |\delta C|_{\max} = |C|\varepsilon \;\Rightarrow\; |\delta D|_{\max} = \left(\left|\frac{\partial D}{\partial B}\right| |B| + \left|\frac{\partial D}{\partial C}\right| |C|\right)\varepsilon$$

Each of the error upper bounds $|\delta B|_{\max}$, $|\delta C|_{\max}$, and $|\delta D|_{\max}$ is the product of a positive error size parameter times ε. These size parameters are given the corresponding label with the subscript E. Thus, B_E, C_E, and D_E are the error size parameters for B, C, and D, and

$$|\delta B|_{\max} = B_E\,\varepsilon \qquad |\delta C|_{\max} = C_E\,\varepsilon \qquad |\delta D|_{\max} = D_E\,\varepsilon \tag{31}$$

where
$$B_E = |B|, \quad C_E = |C|, \quad \text{and} \tag{32}$$

$$D_E = \left|\frac{\partial D}{\partial B}\right| B_E + \left|\frac{\partial D}{\partial C}\right| C_E \quad = \quad 2|B|B_E + 4C_E \tag{33}$$

The value $|\delta D|_{max} = D_E\, \varepsilon$ is the upper bound of round-off error $|\delta D|$ that we seek in order to provide error mitigation in the quadratic-equation multiplicity condition. The mitigation design includes the following three changes to the quadratic-equation algorithm of Figure 4.

- The computer's epsilon value $\varepsilon = 2^{-52}$ is stored as a universal constant.
- In addition to the coefficients B and C, the input values include the error size parameters B_E and C_E. If B and C are user inputs, then set $B_E = |B|$ and $C_E = |C|$. Otherwise, B_E and C_E are calculated and supplied by a higher-level algorithm.
- The following two calculation lines are included in the algorithm immediately following the calculation of determinate $D = B^2 - 4C$:
    - $D_E = 2|B|\, B_E + 4C_E$
    - If $|D| < D_E\, \varepsilon$ then $D = 0$

Figure 8 shows the final quadratic-equation algorithm. This mitigation design assures that two true equal real solutions are calculated accurately as two equal real solutions. Any residual round-off error is not magnified.

The mitigation design for the cubic- and quartic-equation algorithms requires the calculation of several additional error size parameters like B_E, C_E, and D_E. Equations (31) to (33) serve as models for the way these size parameters are used and calculated. If some value L is a user input, then the associated error size parameter is $L_E = |L|$ as in (32). If L is calculated, then L_E is calculated using the appropriate partial derivatives as in (33). If the cubic- or quartic-equation algorithm invokes the quadratic-equation algorithm, then the higher-level algorithm calculates B_E and C_E by employing appropriate partial derivatives as shown below.

Multiplicity Error Mitigation in the Cubic-Equation Algorithm

Figure 9 provides the final calculation algorithm for solving the cubic equation. It updates the preliminary calculation logic of Figure 5 with the calculations needed to prevent round-off error magnification for the multiplicity condition. The computer's epsilon value $\varepsilon = 2^{-52}$ is stored as a universal constant.

The input list for Figure 9 includes not only the cubic-equation coefficients a_2, a_1, and a_0, but also the corresponding error size parameters a_{2E}, a_{1E}, and a_{0E}. The coefficients may be supplied by the user to solve a cubic equation, or they may be supplied by the quartic-equation algorithm or its post-processing algorithm. If a_2, a_1, and a_0 are user inputs, then a_{2E}, a_{1E}, and a_{0E} are the corresponding absolute values:

$$a_{2E} = |a_2| \qquad a_{1E} = |a_1| \qquad a_{0E} = |a_0|. \qquad (34)$$

Otherwise, the higher-level algorithm calculates and supplies values for all of the cubic algorithm inputs: a_2, a_1, a_0, a_{2E}, a_{1E}, and a_{0E}.

Figure 8 Final Calculation Algorithm for Solving the Quadratic Equation

Universal Constant: $\varepsilon = 2^{-52} \approx 2.22\times10^{-16}$

Inputs:

1) Real coefficients B and C of the quadratic equation $Z_n^2 + B\,Z_n + C = 0$
2) Error size parameters B_E, and C_E. If these values are not calculated by a higher-level algorithm, then set them to $B_E = |B|$, $C_E = |C|$.

Outputs: X_1, X_2, Y so that $Z_1 = X_1 + iY$, $Z_2 = X_2 - iY$ are solutions of the quadratic equation.

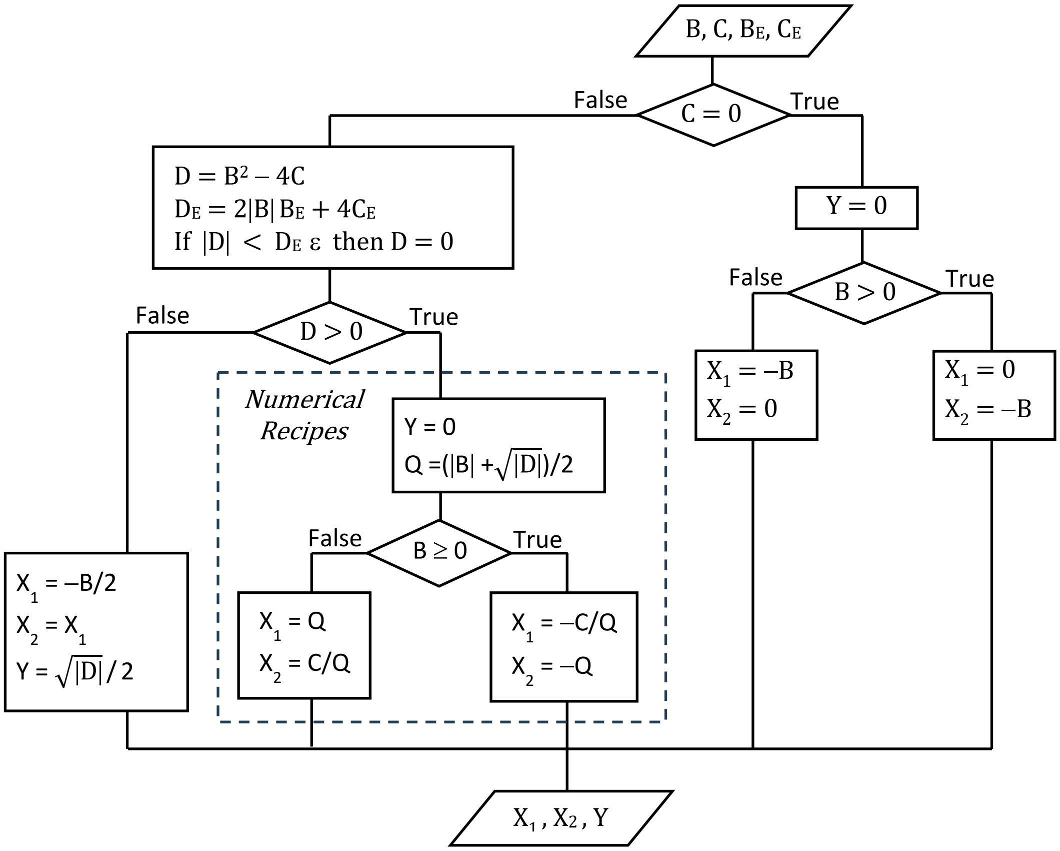


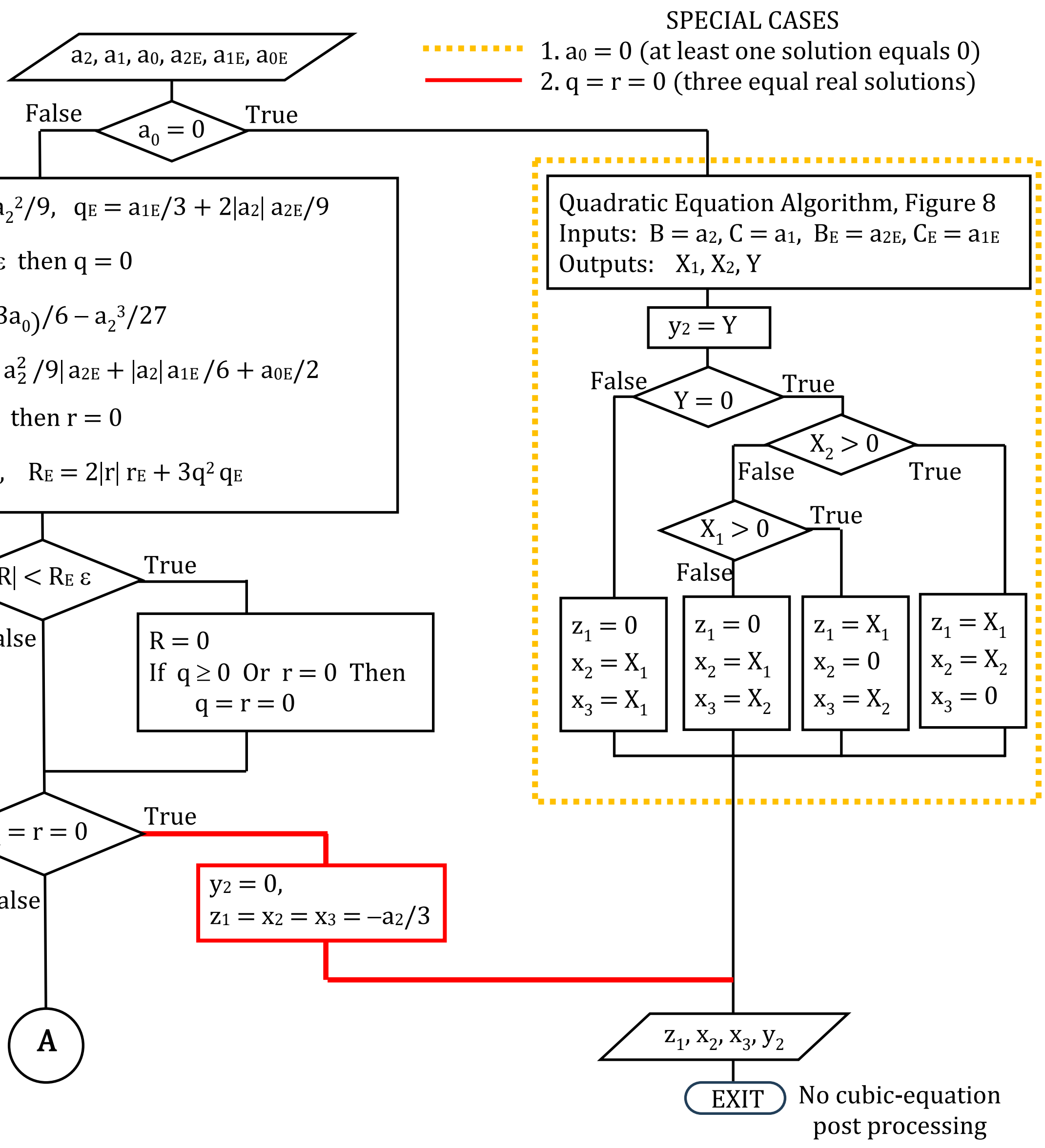
Figure 9 Final Calculation Algorithm for Solving the Cubic Equation

Universal Constant: $\varepsilon = 2^{-52} \approx 2.22\times10^{-16}$

Inputs:

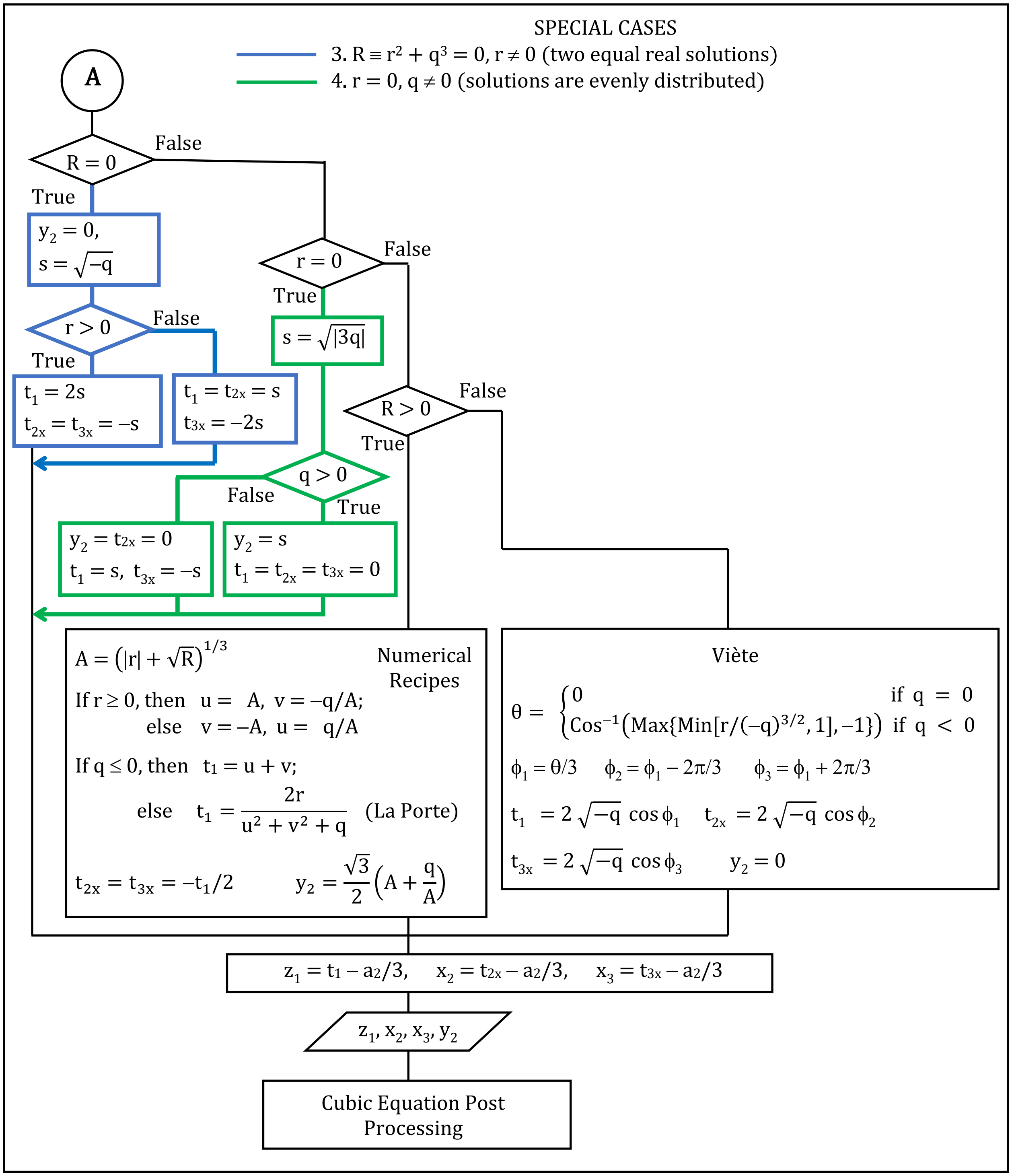
1) Real coefficients a_2, a_1, and a_0 of the cubic equation $z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0$
2) Error size parameters a_{2E}, a_{1E}, and a_{0E}. If these values are not calculated by a higher-level algorithm, set them to $a_{2E} = |a_2|$, $a_{1E} = |a_1|$, $a_{0E} = |a_{0E}|$.

Outputs: z_1, x_2, x_3, y_2 so that z_1, $z_2 = x_2 + iy_2$, $z_3 = x_3 - iy_2$ are the three solutions of the cubic equation.



SPECIAL CASES
1. $a_0 = 0$ (at least one solution equals 0)
2. $q = r = 0$ (three equal real solutions)
$a_2, a_1, a_0, a_{2E}, a_{1E}, a_{0E}$
False
$a_0 = 0$
True
$q = a_1/3 - a_2^2/9$, $q_E = a_{1E}/3 + 2|a_2|\, a_{2E}/9$
If $|q| < q_E\,\varepsilon$ then $q = 0$
$r = (a_2 a_1 - 3a_0)/6 - a_2^3/27$
$r_E = |a_1/6 - a_2^2/9|\, a_{2E} + |a_2|\, a_{1E}/6 + a_{0E}/2$
If $|r| < r_E\,\varepsilon$ then $r = 0$
$R = r^2 + q^3$, $R_E = 2|r|\, r_E + 3q^2\, q_E$
$|R| < R_E\,\varepsilon$
True
False
$R = 0$
If $q \geq 0$ Or $r = 0$ Then
$q = r = 0$
$q = r = 0$
True
False
$y_2 = 0$,
$z_1 = x_2 = x_3 = -a_2/3$
A
Quadratic Equation Algorithm, Figure 8
Inputs: $B = a_2$, $C = a_1$, $B_E = a_{2E}$, $C_E = a_{1E}$
Outputs: X_1, X_2, Y
$y_2 = Y$
False
$Y = 0$
True
$X_2 > 0$
False
True
$X_1 > 0$
True
False
$z_1 = 0$
$x_2 = X_1$
$x_3 = X_1$
$z_1 = 0$
$x_2 = X_1$
$x_3 = X_2$
$z_1 = X_1$
$x_2 = 0$
$x_3 = X_2$
$z_1 = X_1$
$x_2 = X_2$
$x_3 = 0$
z_1, x_2, x_3, y_2
EXIT
No cubic-equation post processing

Figure 9 Final Calculation Algorithm for Solving the Cubic Equation (Page 2)


SPECIAL CASES
3. $R \equiv r^2 + q^3 = 0, r \neq 0$ (two equal real solutions)
4. $r = 0, q \neq 0$ (solutions are evenly distributed)
A
$R = 0$
False
True
$y_2 = 0,$
$s = \sqrt{-q}$
$r > 0$
False
True
$t_1 = 2s$
$t_{2x} = t_{3x} = -s$
$t_1 = t_{2x} = s$
$t_{3x} = -2s$
$r = 0$
False
True
$s = \sqrt{|3q|}$
$R > 0$
False
True
$q > 0$
False
True
$y_2 = t_{2x} = 0$
$t_1 = s, \; t_{3x} = -s$
$y_2 = s$
$t_1 = t_{2x} = t_{3x} = 0$
$A = \left(|r| + \sqrt{R}\right)^{1/3}$
Numerical Recipes
If $r \geq 0$, then $u = A, \; v = -q/A;$
else $v = -A, \; u = q/A$
If $q \leq 0$, then $t_1 = u + v;$
else $t_1 = \dfrac{2r}{u^2 + v^2 + q}$ (La Porte)
$t_{2x} = t_{3x} = -t_1/2$
$y_2 = \dfrac{\sqrt{3}}{2}\left(A + \dfrac{q}{A}\right)$
Viète
$\theta = \begin{cases} 0 & \text{if } q = 0 \\ \mathrm{Cos}^{-1}\left(\mathrm{Max}\{\mathrm{Min}[r/(-q)^{3/2}, 1], -1\}\right) & \text{if } q < 0 \end{cases}$
$\phi_1 = \theta/3 \quad \phi_2 = \phi_1 - 2\pi/3 \quad \phi_3 = \phi_1 + 2\pi/3$
$t_1 = 2\sqrt{-q}\cos\phi_1 \quad t_{2x} = 2\sqrt{-q}\cos\phi_2$
$t_{3x} = 2\sqrt{-q}\cos\phi_3 \quad y_2 = 0$
$z_1 = t_1 - a_2/3, \quad x_2 = t_{2x} - a_2/3, \quad x_3 = t_{3x} - a_2/3$
z_1, x_2, x_3, y_2
Cubic Equation Post Processing

If $a_0 = 0$, then Special Case 1 applies. One solution is 0, and the other two are solutions of the quadratic equation $z_n^2 + a_2 z_n + a_1 = 0$. To find these two solutions, the cubic-equation algorithm invokes the Figure 8 quadratic-equation algorithm with the following input values:

$$B = a_2, \quad C = a_1, \quad B_E = a_{2E}, \quad C_E = a_{1E}.$$

If $a_0 \neq 0$, then the mitigation design calculates error size parameters q_E, r_E, and R_E corresponding to q, r, and R:

$$q = a_1/3 - a_2^2/9, \qquad r = (a_2 a_1 - 3a_0)/6 - a_2^3/27, \qquad R = r^2 + q^3.$$

It uses inputs a_{2E}, a_{1E}, and a_{0E} and the appropriate partial derivatives:

$$\partial q/\partial a_1 = 1/3, \qquad \partial q/\partial a_2 = -2a_2/9$$

$$\partial r/\partial a_2 = a_1/6 - a_2^2/9, \qquad \partial r/\partial a_1 = a_2/6, \qquad \partial r/\partial a_0 = -1/2$$

$$\partial R/\partial r = 2r, \qquad \partial R/\partial q = 3q^2 \,.$$

Values for q_E, r_E, and R_E are calculated using the following formulas.

$$q_E = \left|\frac{\partial q}{a_1}\right| a_{1E} + \left|\frac{\partial q}{a_2}\right| a_{2E} \;=\; \frac{a_{1E}}{3} + \frac{2|a_2|a_{2E}}{9} \tag{35}$$

$$r_E = \left|\frac{\partial r}{a_2}\right| a_{2E} + \left|\frac{\partial r}{a_1}\right| a_{1E} + \left|\frac{\partial r}{a_0}\right| a_{10E} \;=\; \left|\frac{a_1}{6} - \frac{a_2^2}{9}\right| a_{2E} + \frac{|a_2|a_{1E}}{6} + \frac{a_{0E}}{2} \tag{36}$$

$$R_E = \left|\frac{\partial R}{\partial r}\right| r_E + \left|\frac{\partial R}{\partial q}\right| q_E \;=\; 2|r|r_E + 3q^2 q_E \tag{37}$$

If the absolute value of q, r, and/or R are sufficiently small, then the value is reset to zero according to the following tests.

If $|q| < q_E\,\varepsilon$, then $q = 0$.

If $|r| < r_E\,\varepsilon$, then $r = 0$.

If $|R| < R_E\,\varepsilon$, then $\{R = 0$ and if $(q \geq 0$ or $r = 0)$ then $q = r = 0\}$.

This last line of logic is necessary to prevent round-off error from creating an illogical situation in which $R = r^2 + q^3 = 0$ and either $r = 0$ or $q = 0$, but not both. Also note that q cannot be positive if $R = 0$.

These logical tests involving $q_E\,\varepsilon$, $r_E\,\varepsilon$, and $R_E\,\varepsilon$ assure that the calculated value of q, r, and/or R is set to 0 whenever the corresponding true value is 0. If the cubic equation has multiple true solutions equal to the same real value, then the corresponding calculated solutions also equal a common real value.

If either Special Case 1 ($a_0 = 0$) or Special Case 2 ($q = r = 0$) apply, then cubic-equation post processing is not required. The Figure 9 algorithm exits immediately after output of the solution values z_1, x_2, x_3, y_2. Special Case 1 needs no post processing because it uses the Figure 8 quadratic-equation algorithm, whose Numerical Recipes design accurately calculates any nonzero solutions regardless of any differences in their magnitudes. Special Case 2 needs no post processing because it has three equal solutions; there are no differences in magnitude to create solution-error magnification.

Page 2 of Figure 9 shows the remainder of the algorithm for cubic-equation cases other than Special Cases 1 and 2. This portion of the algorithm is unchanged from the Figure 5 preliminary algorithm except for the indication of post processing at completion.

Table III below shows calculated parameters for the Example 1 cubic equation $z_n^3 - 5\,z_n^2 + 8\,z_n - 4 = 0$. The table compares calculations of the Figure 1 algorithm without round-off error mitigation to those of Figure 9 with mitigation. The equations true solutions are 2, 2, and 1. The true values of r and q are $r = -1/27$ and $q = -1/9$, so the true value of R is $R = r^2 + q^3 = 0$.

Both algorithms calculate R as a round-off error $R = \delta R = 1.04083\times10^{-17}$.

Because the calculated R is positive, the original Figure 1 algorithm uses Numerical Recipes to complete the calculation, starting with the calculation of A.

$$A = \left(|r| + \sqrt{R}\right)^{1/3} = \left(|r| + \sqrt{\delta R}\right)^{1/3}$$

With true values $r = -1/27$ and $R = 0$, the true value of A is 1/3. The erroneous calculated R value δR produces an error δA in A given by

$$\delta A = \left(|r| + \sqrt{\delta R}\right)^{1/3} - |r|^{1/3} \;=\; |r|^{1/3}\left[\left(1 + \sqrt{\delta R}/|r|\right)^{1/3} - 1\right] \;\approx\; |r|^{1/3}\left[\tfrac{1}{3}\sqrt{\delta R}/|r|\right] \;=3\sqrt{\delta R}$$

where $|r| = 1/27$. Taking the square root of δR in the A formula greatly magnifies the error: $\delta A \approx 3\sqrt{\delta R} = 9.678588\times10^{-9}$. To a first approximation, the error δA cancels itself out in the subtraction $t_1 = q/A - A$. The values $t_{x2} = t_{x3} = -t_1/2$, z_1, x_2, and x_3 are therefore unaffected by δA. The true y_2 value is 0, but the sum in the calculation $y_2 = \frac{\sqrt{3}}{2}(A + q/A)$ has the true value of A cancel itself out while the error δA is doubled:

$$\begin{aligned}
\text{calculated } A + q/A \quad &=\; A+\delta A + q/(A + \delta A) = A+\delta A + (q/A)[1/(1+\delta A/A)] \\
&\approx\; A+\delta A + (q/A)(1- \delta A/A) \;=\; A+q/A + \delta A(1-q/A^2) \\
&=\; 1/3 + (-1/9)/(1/3) + \delta A[1-(-1/9)/(1/3)^2] \\
\text{calculated } A + q/A \quad &=\; 2\delta A
\end{aligned}$$

$$\text{calculated } y_2 \;=\; \frac{\sqrt{3}}{2}\,(\text{calculated } A + q/A) \;=\; \sqrt{3}\,\delta A \;= 3\,\sqrt{3\delta R} \;=\; 3\sqrt{3\times1.04083\times10^{-17}}$$

$$\text{calculated } y_2 \;= 1.676381\times10^{-8}$$

The final algorithm, Figure 9, avoids this magnified error. After calculating R, it calculates $R_E = 1.481481481$ via equations (34) to (37). Because $|R| = 1.04083\times10^{-17} < R_E\,\varepsilon = 3.28955\times10^{-16}$, the algorithm resets R to 0 and invokes Special Case 3 to calculate the correct solutions.

Table III. Calculated Parameters for Example 1 Cubic Equation with Multiplicity 2

Example 1 Cubic Equation: $z_n^3 - 5\,z_n^2 + 8\,z_n - 4 = 0$ with solutions 2, 2, and 1		
Parameter Symbol	Figure 1 Cubic-Equation Algorithm (Value without error mitigation)	Figure 9 Final Algorithm (Value with error mitigation)
ε		$\varepsilon = 2^{-52} =$ 2.2204460E-16
a_2	-5	-5
a_1	8	8
a_0	-4	-4
a_{2E}		5
a_{1E}		8
a_{0E}		4
$a_0 = 0$	FALSE	FALSE
q	-0.111111111	-0.111111111
q_E		8.222222222
$\lvert q\rvert < q_E\,\varepsilon$		FALSE
r	-0.037037037	-0.037037037
r_E		15.88888889
$\lvert r\rvert < r_E\,\varepsilon$		FALSE
$R = r^2 + q^3$	1.04083E-17	1.04083E-17
R_E		1.481481481
$\lvert R\rvert < R_E\,\varepsilon$		TRUE
R reset		0
$q \geq 0$ Or $r=0$		FALSE
$R = 0$		TRUE
	Numerical Recipes	Special Case 3
	$r^2 + q^3 > 0$	$R = 0,\ r \neq 0$
A	0.333333343	
u	-0.3333333236547	
v	-0.3333333430119	
t_1	-0.666666667	0.333333333
t_{2x}	0.333333333	0.333333333
y_2	1.676381E-08	0
t_{3x}	0.333333333	-0.666666667
z_1	1	2
x_2	2	2
y_2	1.676381E-08	0
x_3	2	1

It is possible for the mitigation design to calculate two solutions as equal to each other when the corresponding true solutions of the cubic equation differ from each other by a very small relative value. Such a near-miss cubic equation has little practical significance because the coefficients would require extreme precision. Section X addresses this situation in detail, but for now consider the following example.

We modify the Example 1 cubic equation $z_n^3 - 5\,z_n^2 + 8\,z_n - 4 = 0$ (true solutions 2, 2, and 1) by decreasing the constant coefficient from -4 to -4.00000000000001, a change of 1×10^{-14}. The calculated R value decreases from 1.04083×10^{-17} to the new value $R = -3.8\times10^{-16}$ so that $|R| = 3.8\times10^{-16} > R_E\,\varepsilon = 3.3\times10^{-16}$. The calculated R keeps its negative value, and the algorithm correctly reports the two different solution values of about 2.0000001 and 1.9999999 ($2 \pm 1\times10^{-7}$). Only if the constant coefficient -4 changes by a nonzero magnitude less than 1×10^{-14} does the calculated $|R|$ become small enough that R is incorrectly reset to zero, and the algorithm produces the multiplicity result: 2, 2, 1.

Multiplicity Error Mitigation in the Quartic-Equation Algorithm

Figure 10 revises the Figure 6 quartic-equation calculation logic similar to the way that Figure 9 revises the Figure 5 cubic-equation calculation logic. Figure 10 updates the preliminary calculation logic with calculations to address round-off error magnification in the multiplicity condition. The computer's epsilon value $\varepsilon = 2^{-52}$ is stored as a universal constant.

Figure 10 uses the same inputs as Figure 6: the four quartic-equation coefficients A_3, A_2, A_1, A_0. The corresponding error size parameters are calculated immediately as the absolute values.

$$A_{3E} = |A_3|, \qquad A_{2E} = |A_2|, \qquad A_{1E} = |A_1| \qquad A_{0E} = |A_0| \tag{38}$$

If $A_0 = 0$, then Special Case 5 applies (dashed blue box). One solution, $Z_1 = X_1 + iY_1$, is zero, and the other three are solutions of the cubic equation $Z_n^3 + A_3\,Z_n^2 + A_2 Z_n + A_1 = 0$. To find them, the algorithm invokes the Figure 9 cubic-equation algorithm with the following input values:

$$a_2 = A_3, \quad a_1 = A_2, \quad a_0 = A_1, \quad a_{2E} = A_{3E}, \quad a_{1E}, = A_{2E}, \quad a_{0E} = A_{1E}.$$

If $A_0 \neq 0$, then Figure 10, like Figure 6, calculates

$$C\ = A_3 / 4, \qquad b_2 = A_2 - 6C^2, \qquad b_1 = A_1 - 2A_2C + 8C^3, \qquad b_0 = A_0 - A_1C + A_2C^2 - 3C^4.$$

The mitigation design also calculates the corresponding error size parameters C_E, b_{2E}, b_{1E}, and b_{0E}. These are derived in the usual way from A_{3E}, A_{2E}, A_{1E} and A_{0E}.

$$C_E = \left|\frac{dC}{dA_3}\right| A_{3E} \ = \ \frac{A_{3E}}{4} \tag{39}$$

$$b_{2E} = \left|\frac{\partial b_2}{\partial A_2}\right| A_{2E} + \left|\frac{\partial b_2}{\partial C}\right| C_E \ = \ A_{2E} + 12|C|C_E \tag{40}$$

$$b_{1E} = \left|\frac{\partial b_1}{\partial A_1}\right| A_{1E} + \left|\frac{\partial b_1}{\partial A_2}\right| A_{2E} + \left|\frac{\partial b_1}{\partial C}\right| C_E \ = \ A_{1E} + 2|C|A_{2E} + |-2A_2 + 24C^2|C_E \tag{41}$$

$$b_{0E} = \left|\frac{\partial b_0}{\partial A_0}\right| A_{0E} + \left|\frac{\partial b_0}{\partial A_1}\right| A_{1E} + \left|\frac{\partial b_0}{\partial A_2}\right| A_{2E} + \left|\frac{\partial b_0}{\partial C}\right| C_E$$

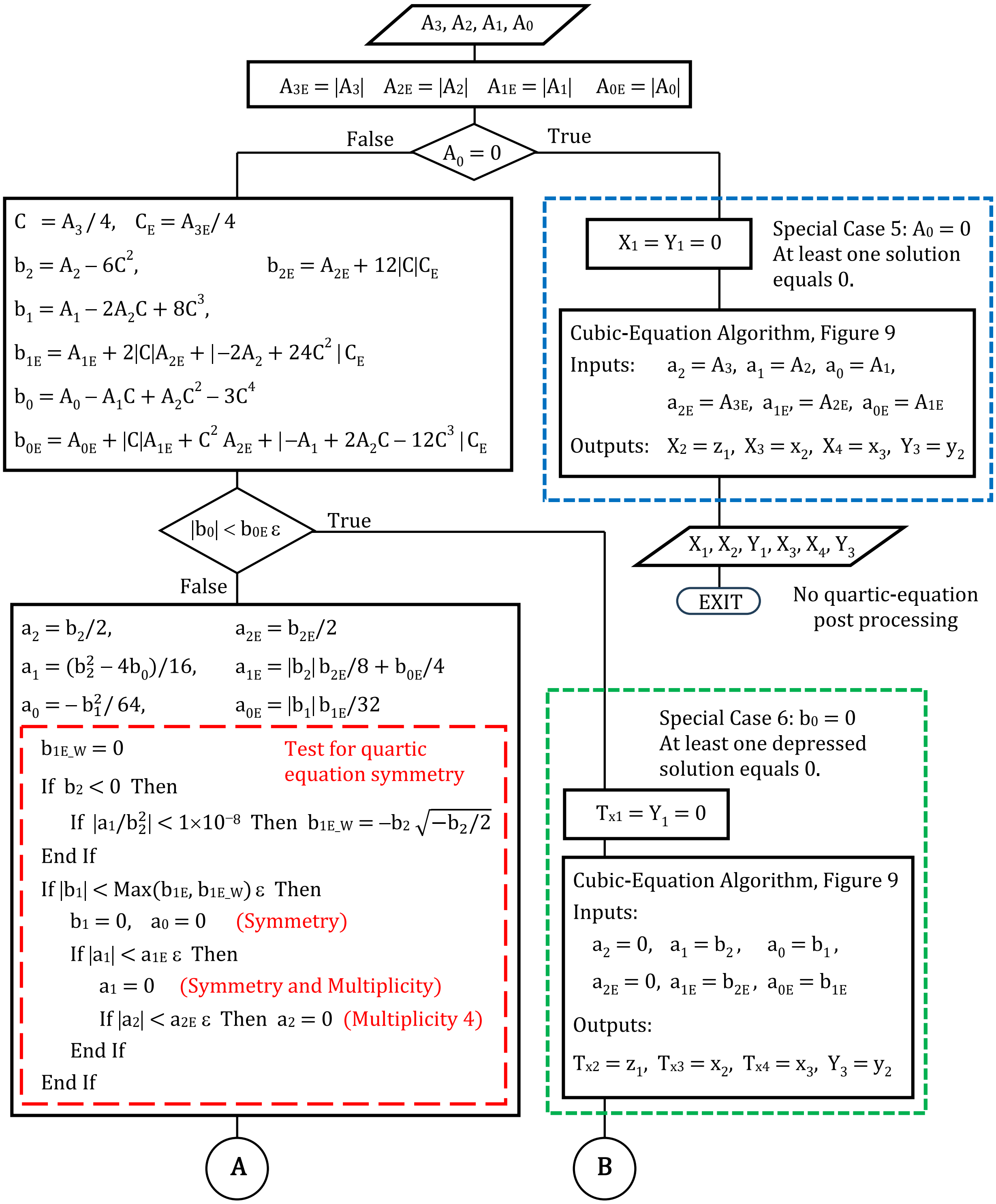
$$b_{0E} = \ A_{0E} + |C|A_{1E} + C^2A_{2E} + |-A_1 + 2A_2C - 12C^3|C_E \tag{42}$$

Figure 10 Final Calculation Algorithm for Solving the Quartic Equation

Universal Constant: $\varepsilon = 2^{-52} \approx 2.22\times10^{-16}$

Inputs: Real coefficients A_3, A_2, A_1, and A_0 of the quartic equation $Z_n^4 + A_3Z_n^3 + A_2Z_n^2 + A_1Z_n + A_0 = 0$

Outputs: X_1, X_2, Y_1, X_3, X_4, Y_3 so that $Z_1=X_1+iY_1$, $Z_2=X_2-iY_1$, $Z_3=X_3+iY_3$, $Z_4=X_4-iY_3$ are its four solutions.



A_3, A_2, A_1, A_0

$A_{3E} = |A_3| \quad A_{2E} = |A_2| \quad A_{1E} = |A_1| \quad A_{0E} = |A_0|$

$A_0 = 0$

False

True

$X_1 = Y_1 = 0$

Special Case 5: $A_0 = 0$
At least one solution equals 0.

Cubic-Equation Algorithm, Figure 9

Inputs: $a_2 = A_3,\ a_1 = A_2,\ a_0 = A_1,$

$a_{2E} = A_{3E},\ a_{1E,} = A_{2E},\ a_{0E} = A_{1E}$

Outputs: $X_2 = z_1,\ X_3 = x_2,\ X_4 = x_3,\ Y_3 = y_2$

$X_1, X_2, Y_1, X_3, X_4, Y_3$

EXIT

No quartic-equation post processing

$C = A_3/4, \quad C_E = A_{3E}/4$

$b_2 = A_2 - 6C^2, \qquad b_{2E} = A_{2E} + 12|C|C_E$

$b_1 = A_1 - 2A_2C + 8C^3,$

$b_{1E} = A_{1E} + 2|C|A_{2E} + |-2A_2 + 24C^2|C_E$

$b_0 = A_0 - A_1C + A_2C^2 - 3C^4$

$b_{0E} = A_{0E} + |C|A_{1E} + C^2A_{2E} + |-A_1 + 2A_2C - 12C^3|C_E$

$|b_0| < b_{0E}\,\varepsilon$

True

False

$a_2 = b_2/2, \qquad a_{2E} = b_{2E}/2$

$a_1 = (b_2^2 - 4b_0)/16, \qquad a_{1E} = |b_2|\,b_{2E}/8 + b_{0E}/4$

$a_0 = -\,b_1^2/64, \qquad a_{0E} = |b_1|\,b_{1E}/32$

Test for quartic equation symmetry

```
$b_{1E\_W} = 0$
If $b_2 < 0$ Then
    If $|a_1/b_2^2| < 1\times10^{-8}$ Then $b_{1E\_W} = -b_2\sqrt{-b_2/2}$
End If
If $|b_1| < \mathrm{Max}(b_{1E}, b_{1E\_W})\,\varepsilon$ Then
    $b_1 = 0$,  $a_0 = 0$    (Symmetry)
    If $|a_1| < a_{1E}\,\varepsilon$ Then
        $a_1 = 0$    (Symmetry and Multiplicity)
        If $|a_2| < a_{2E}\,\varepsilon$ Then $a_2 = 0$  (Multiplicity 4)
    End If
End If
```

Special Case 6: $b_0 = 0$
At least one depressed solution equals 0.

$T_{x1} = Y_1 = 0$

Cubic-Equation Algorithm, Figure 9

Inputs:

$a_2 = 0,\ a_1 = b_2,\ a_0 = b_1,$

$a_{2E} = 0,\ a_{1E} = b_{2E},\ a_{0E} = b_{1E}$

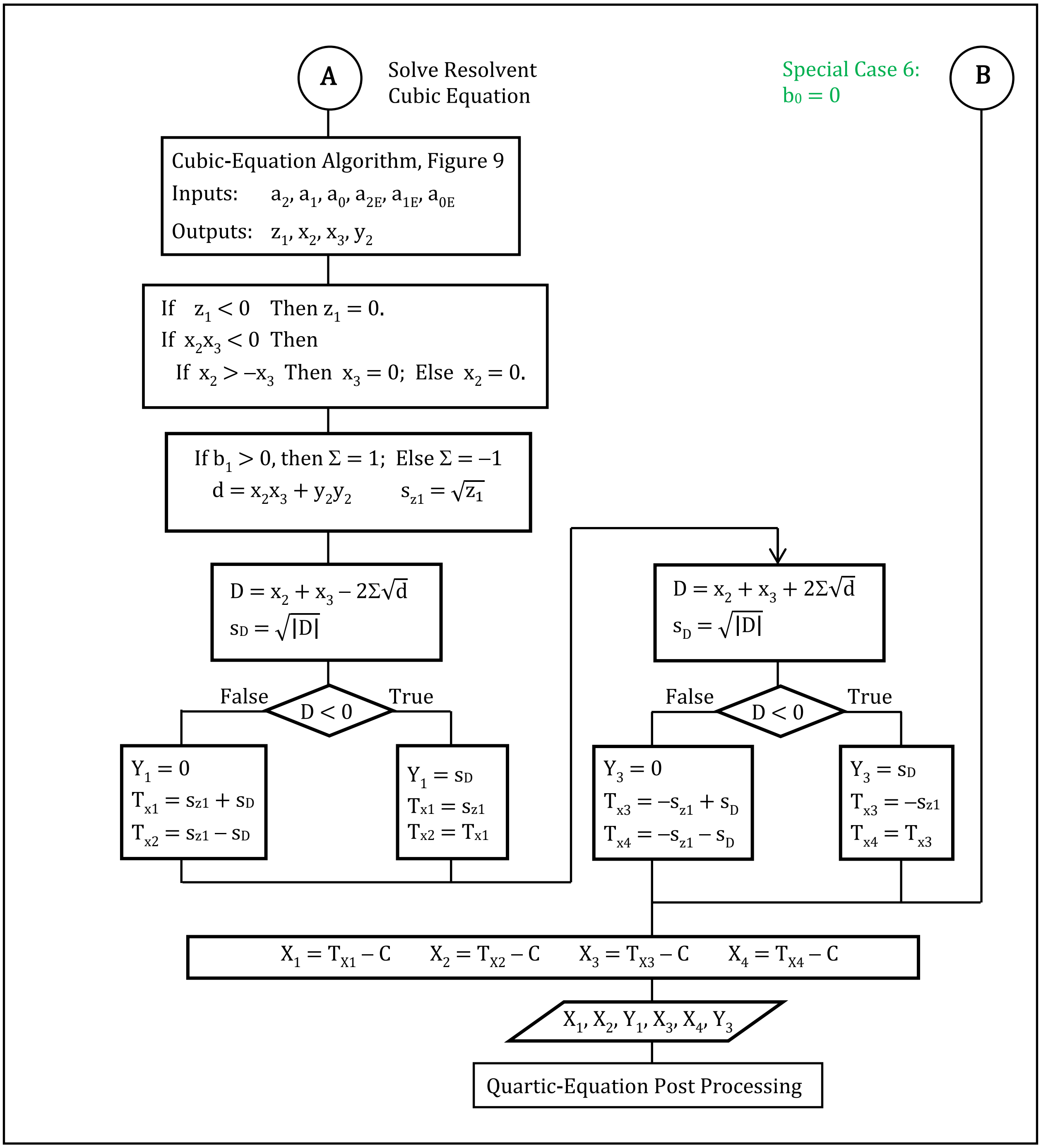
Outputs:

$T_{x2} = z_1,\ T_{x3} = x_2,\ T_{x4} = x_3,\ Y_3 = y_2$

A

B

Figure 10 Final Calculation Algorithm for Solving the Quartic Equation (Page 2)


A
Solve Resolvent Cubic Equation
Special Case 6: $b_0 = 0$
B
Cubic-Equation Algorithm, Figure 9
Inputs: $a_2, a_1, a_0, a_{2E}, a_{1E}, a_{0E}$
Outputs: z_1, x_2, x_3, y_2
If $z_1 < 0$ Then $z_1 = 0$.
If $x_2x_3 < 0$ Then
If $x_2 > -x_3$ Then $x_3 = 0$; Else $x_2 = 0$.
If $b_1 > 0$, then $\Sigma = 1$; Else $\Sigma = -1$
$d = x_2x_3 + y_2y_2$
$s_{z1} = \sqrt{z_1}$
$D = x_2 + x_3 - 2\Sigma\sqrt{d}$
$s_D = \sqrt{|D|}$
$D = x_2 + x_3 + 2\Sigma\sqrt{d}$
$s_D = \sqrt{|D|}$
False
$D < 0$
True
$Y_1 = 0$
$T_{x1} = s_{z1} + s_D$
$T_{x2} = s_{z1} - s_D$
$Y_1 = s_D$
$T_{x1} = s_{z1}$
$T_{x2} = T_{x1}$
False
$D < 0$
True
$Y_3 = 0$
$T_{x3} = -s_{z1} + s_D$
$T_{x4} = -s_{z1} - s_D$
$Y_3 = s_D$
$T_{x3} = -s_{z1}$
$T_{x4} = T_{x3}$
$X_1 = T_{X1} - C$
$X_2 = T_{X2} - C$
$X_3 = T_{X3} - C$
$X_4 = T_{X4} - C$
$X_1, X_2, Y_1, X_3, X_4, Y_3$
Quartic-Equation Post Processing

The algorithm tests for Special Case 6, $b_0 = 0$, by testing whether $|b_0| < b_{0E}\,\varepsilon$, that is, whether the calculated $|b_0|$ is less than the upper bound of its round-off error. If so, then b_0 is assumed to be 0, and Special Case 6 applies (dashed green box). One depressed solution, $T_1 = T_{x1}+iY_1$, is zero, and the other three are solutions of the cubic equation $T_n^3 + b_2T_n + b_1 = 0$.

To find these other three depressed solutions, the algorithm invokes the Figure 9 cubic-equation algorithm with the following input values:

$$a_2 = 0,\quad a_1 = b_2,\quad a_0 = b_1,\quad a_{2E} = 0,\quad a_{1E}, = b_{2E},\quad a_{0E} = b_{1E}.$$

If Special Case 6 does not apply, the algorithm calculates resolvent-cubic-equation coefficients a_2, a_1, and a_0 and their corresponding error size parameters a_{2E}, a_{1E}, and a_{0E}.

$a_2 = b_2/2, \qquad a_1 = (b_2^2 - 4b_0)/16, \qquad a_0 = -\,b_1^2/\,64$

$$a_{2E} = \left|\frac{da_2}{db_2}\right| b_{2E} \;=\; \frac{b_{2E}}{2} \tag{43}$$

$$a_{1E} = \left|\frac{\partial a_1}{\partial b_2}\right| b_{2E} + \left|\frac{\partial a_1}{\partial b_0}\right| b_{0E} \;=\; \frac{|b_2|b_{2E}}{8} + \frac{b_{0E}}{4} \tag{44}$$

$$a_{0E} = \left|\frac{da_0}{db_1}\right| b_{1E} \;=\; \frac{|b_1|b_{1E}}{32} \tag{45}$$

Additional calculations, shown in the dashed red box, are required for the quartic equations with symmetry or its near miss.

Quartic Equations with Symmetry or Its Near Miss

Our usual method of detecting Special Case 1, $a_0 = 0$, in the resolvent cubic equation can fail for quartic equations with a combination of symmetry (or its near miss) and multiplicity (or its near miss). The following discussion shows how the calculations in the red box address this situation.

The case $a_0 = 0$ in the resolvent cubic equation implies symmetry in the quartic equation. If $a_0 = 0$, then $b_1 = 0$ because $a_0 = -\,b_1^2/\,64$, Equation (13). The value b_1 is the linear coefficient in the depressed quartic equation, $T_n^4 + b_2T_n^2 + b_1T_n + b_0 = 0$, Equation (9). The case $a_0 = b_1 = 0$ reduces the depressed quartic equation to

$$T_n^4 + b_2T_n^2 + b_0 = 0.$$

This is a quadratic equation in T_n^2. The quartic equation's four depressed solutions T_n are the four values

$$\pm\sqrt{\left[-b_2 \pm \sqrt{b_2^2 - 4b_0}\,\right]/2}\,.$$

The negative of every depressed solution T_n is itself a depressed solution. The four T_n and the depressed quartic polynomial $P_T(T) = T^4 + b_2T^2 + b_0 = P_T(-T)$ are symmetric about $T = 0$. The four quartic-equation solutions $Z_n = T_n - C$ are symmetric about $Z = -C$. Thus, Special Case 1 ($a_0 = 0$) for the resolvent cubic equation corresponds to symmetry in the quartic equation: $b_1 = 0$ and $P_T(T) = P_T(-T)$.

The mitigation design's normal approach to detect this symmetry is to reset b_1 and a_0 to 0 if $|b_1| < b_{1E}\,\varepsilon$, but this approach fails for a type of quartic equation that has both symmetry (or its near miss) and multiplicity (or its near miss). The quartic equation has the four depressed solutions

$$T_1 = T_0, \qquad T_2 = T_0, \qquad T_3 = -T_0 + \Delta T, \qquad T_4 = -T_0 - \Delta T \qquad \text{where } |\Delta T| << T_0. \tag{46}$$

The quartic equation solutions are $Z_n = T_n - C$, and the quartic equation coefficients are given by check Equations (14) to (17):

$$A_3 = 4C \qquad\qquad A_2 = -2T_0^2 + 6C^2 - \Delta T^2 \tag{47}$$

$$A_1 = -4C(T_0^2 - C^2) + 2(T_0 - C)\,\Delta T^2 \qquad\qquad A_0 = (T_0^2 - C^2)^2 - (T_0 - C)^2\,\Delta T^2. \tag{48}$$

The check equations applied to the depressed solutions in (46) give the depressed quartic equation coefficients:

$$b_2 = -2T_0^2 - \Delta T^2 \qquad\qquad b_1 = 2T_0\,\Delta T^2 \qquad\qquad b_0 = T_0^2\,(T_0^2 - \Delta T^2). \tag{49}$$

Note that these expressions for the b_n are the same as those for the corresponding A_n with C set to zero. Apply Equations (38) and (39) for the error size parameters A_{1E}, A_{2E}, and C_E to Equation (41) for b_{1E}:

$$b_{1E} = |A_1| + 2|C||A_2| + |-2A_2 + 24C^2||C|\,. \tag{50}$$

The problem of using $|b_1| < b_{1E}\,\varepsilon$ to detect symmetry becomes evident by examining the special case $\Delta T = A_3 = C = 0$. The resulting quartic equation has symmetry and two double solutions: $T_1 = T_2 = T_0$ and $T_3 = T_4 = -T_0$. With $\Delta T = 0$, Equation (49) gives $b_1 = 0$. The problem is that Equations (47), (48), and (50) imply that $b_{1E} = 0$. An incremental change in C from 0 leaves $b_1 = 0$ and produces only an incremental increase in b_{1E}. That small increase cannot assure that $b_{1E}\,\varepsilon$ exceeds the round-off error in the calculated b_1 value.

We need an effective, new, upper bound of the round-off error in calculated b_1 for quartic equations that have both symmetry (or its near miss) and two double solutions (or near misses). The value $b_{1E}\,\varepsilon$ is inadequate for that situation.

To find that round-off error upper bound, examine the calculation of b_1 from check Equation (20) where b_1 replaces A_1, $Y_1 = Y_3 = 0$, and the X_n become the depressed quartic-equation solutions in (46): $T_1 = T_2 = T_0$, $T_3 = -T_0 + \Delta T$, $T_4 = -T_0 - \Delta T$.

$$b_1 = -[T_1T_2(T_3+T_4) + T_3T_4(T_1 + T_2)]$$

$$b_1 = -[T_0^2\,(-2T_0^2) + (T_0^2 - \Delta T^2)(2T_0)]$$

Suppose $\Delta T^2/T^2$ is very small. A computer may calculate the difference $(T_0^2 - \Delta T^2)$ in this last expression simply as T_0^2 and then calculate the resulting b_1 as 0 if

$$\Delta T^2 < T_0^2\,\varepsilon.$$

$T_0^2\varepsilon$ is the least significant bit value of the stored T_0^2 value. The true b_1 per (49) is $b_1 = 2T_0\,\Delta T^2$. Thus, if $|b_1| = 2|T_0|\,\Delta T^2 < 2|T_0^3|\varepsilon$, then b_1 might be calculated as zero. The value $2|T_0^3|\varepsilon$ is the one we seek for the upper bound of round-off error in calculated b_1.

The value $|T_0|$ in $2|T_0^3|\varepsilon$ is not available to us in practice, so we estimate $|T_0|$ from the calculated value of b_2 in (49): $b_2 = -2T_0^2 - \Delta T^2$. The assumption $|\Delta T| << T_0$ implies that $b_2 \approx -2T_0^2$ and allows us to estimate $|T_0|$ as $|T_0| \approx \sqrt{-b_2/2}$ and $2|T_0^3|$ as

$$2|T_0|^3 \approx -b_2\sqrt{-b_2/2} \quad \text{where} \quad b_2 < 0 \quad \text{and} \quad |\Delta T| << T_0.$$

We show later how we assure that $|\Delta T| << T_0$ for these calculations.

When the quartic equation has both symmetry (or its near miss) and two double solutions (or near misses), our new test for resetting b_1 and a_0 to zero becomes:

$$\text{For } b_2 < 0 \text{ and } |\Delta T| << T_0, \text{ reset } b_1 \text{ and } a_0 \text{ to 0 if } |b_1| < b_{1E\_W}\,\varepsilon \text{ where}$$

$$b_{1E\_W} = -b_2\sqrt{-b_2/2}. \tag{51}$$

If the resolvent-cubic-equation coefficients a_1 and a_0 both equal zero, then the quartic equation has both symmetry and multiplicity. Show this by applying Equation (49) to Equation (12) for a_1:

$$a_1 = (b_2^2 - 4b_0)/16 = [(-2T_0^2 - \Delta T^2)^2 - 4T_0^2\,(T_0^2 - \Delta T^2)]/16 = (8T_0^2\Delta T^2 + \Delta T^4)/16 \tag{52}$$

When ΔT is zero, then so is a_1.

The algorithm assures that $|\Delta T| << T_0$ when it applies the test $|b_1| < b_{1E\_W}\,\varepsilon$ for resetting b_1 and a_0 to zero. First it initializes b_{1E_W} to 0. If $b_2 < 0$, it recalculates $b_{1E\_W} = -b_2\sqrt{-b_2/2}$ only if $|a_1/b_2^2| < 1\times10^{-8}$. From Equations (49) and (52), this means that $b_{1E\_W} = 0$ unless

$$\left|\frac{a_1}{b_2^2}\right| = \left|\frac{(8T_0^2\Delta T^2 + \Delta T^4)/16}{4T_0^4 + 4T_0^2\Delta T^2 + \Delta T^4}\right| \approx \left|\frac{\Delta T^2}{8T_0^2}\right| < 1\times10^{-8}.$$

Thus, $b_{1E\_W} = 0$ unless $(\Delta T/T_0)^2$ is less than about 8×10^{-8}.

At greater values of $(\Delta T/T_0)^2$ when $|a_1/b_2^2| \geq 1\times10^{-8}$, b_{1E_W} is no longer needed. The value of b_{1E} in Equation (50) is sufficiently great that $b_{1E}\,\varepsilon$ can fulfill its proper role as round-off error upper bound for b_1.

Notice that the computation in the red box of Figure 10 keeps the "If $b_2 < 0$" and the "If $|a_1/b_2^2| < 1\times10^{-8}$ " as separate lines. Keeping them as separate lines of code is necessary to prevent a run-time error when $b_2 = 0$.

After the algorithm determines the value of b_{1E_W}, either 0 or $-b_2\sqrt{-b_2/2}$, it tests whether $|b_1| < \text{Max}(b_{1E}, b_{1E\_W})\,\varepsilon$. The maximum of $b_{1E}\,\varepsilon$ and $b_{1E\_W}\,\varepsilon$ serves as the upper bound of round-off-error magnitude in the calculated b_1. If $|b_1| < \text{Max}(b_{1E}, b_{1E\_W})\,\varepsilon$, the algorithm resets b_1 and a_0 to zero, indicating that the quartic equation is symmetric. Also, if $|b_1| < \text{Max}(b_{1E}, b_{1E\_W})\,\varepsilon$, the algorithm tests whether $|a_1| < a_{1E}\,\varepsilon$.

If $|a_1| < a_{1E}\,\varepsilon$, then the algorithm resets a_1 to zero indicating that the quartic equation has multiplicity as well as symmetry. Also, if $|a_1| < a_{1E}\,\varepsilon$, the algorithm tests whether $|a_2| < a_{2E}\,\varepsilon$.

If $|a_2| < a_{2E}\,\varepsilon$, then a_2 is reset to zero, which implies that all coefficients of the resolvent cubic equation are zero: $a_2 = a_1 = a_0 = 0$. This situation produces zero values for all three resolvent cubic equation solutions and all four depressed quartic equation solutions:

$$z_2 = z_1 = z_0 = T_1 = T_2 = T_3 = T_4 = 0.$$

The four quartic equation solutions $Z_n = T_n - C$, therefore, all equal the same real value (multiplicity 4): $Z_1 = Z_2 = Z_3 = Z_4 = -C = -A_3/4$.

This completes the Figure 10 calculations in the dashed red box for quartic equations with symmetry (or its near miss) and perhaps also multiplicity (or its near miss). Calculation of a_2, a_1, a_0, a_{2E}, a_{1E}, and a_{0E} is complete, and the algorithm is ready to invoke the Figure 9 cubic equation algorithm to solve the resolvent cubic equation.

Page 2 of Figure 10 is the same as that of the preliminary quartic-equation algorithm, Figure 6, with the following exceptions. The Figure 10 algorithm invokes the final cubic-equation algorithm of Figure 9 rather than the preliminary cubic-equation algorithm of Figure 5. At completion, the Figure 10 algorithm indicates the need for quartic-equation post processing.

Selection of the Modified Euler Quartic-Equation Algorithm for the Mitigation Design

The modified Euler quartic-equation algorithm is selected over alternative quartic-equation algorithms for the mitigation design because it requires relatively few changes.

Except for the if-statements in Figure 10's dashed red box, the Euler quartic-equation algorithm needs no branches for the multiplicity condition because it uses all three solutions z_1, z_2, and z_3 of its resolvent cubic equation. Most other quartic-equation algorithms use only one solution. Figure 11, described below, shows that multiplicity among the Euler resolvent-cubic-equation solutions z_n corresponds to multiplicity among solutions T_n of the depressed quartic equation. If the quartic equation has multiplicity, then the Figure 9 algorithm calculates resolvent-cubic-equation solutions with the appropriate multiplicity, which allows the normal Euler algorithm computation to produce quartic-equation solutions of the correct multiplicity.

Figure 11 below summarizes the relationships between the depressed quartic equation, the Euler resolvent cubic equation, and their solutions for multiplicity and quartic-equation symmetry. The figure's first section shows the Depressed Quartic Equation (Equation (9)) with solutions T_n. The T_n are related to solutions Z_n of the quartic equation

$$Z_n^4 + A_3 Z_n^3 + A_2 Z_n^2 + A_1 Z_n + A_0 = 0, \qquad n = 1, 2, 3, 4$$

by $\qquad T_n = Z_n + C \qquad \Leftrightarrow \qquad Z_n = T_n - C \quad$ where $\quad C = A_3/4.$

Thus, any multiplicity among the Z_n has a corresponding multiplicity among the T_n.

Figure 11 Relationships between the Depressed Quartic Equation, the Euler Resolvent Cubic Equation, and Their Solutions for Multiplicity and Quartic Symmetry

Depressed Quartic Equation

$$T_n^4 + b_2 T_n^2 + b_1 T_n + b_0 = 0, \quad n = 1, 2, 3, 4 \qquad T_1 + T_2 + T_3 + T_4 = 0$$

Euler Resolvent Cubic Equation

$$z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0, \qquad n = 1, 2, 3$$

$$\text{where} \qquad a_2 = b_2/2, \qquad a_1 = (b_2^2 - 4b_0)/16, \qquad a_0 = -b_1^2/64$$

Properties of the Resolvent-Cubic-Equation Solutions

Solution z_1 is real. Solutions z_2 and z_3 are real or they are a complex conjugate pair.
$z_1 \geq 0$. $z_2 z_3 \geq 0$. If z_2 and z_3 are real, then $z_1 \geq z_2 \geq z_3$
Thus, if z_2 and z_3 are real, then either $z_1 \geq z_2 \geq z_3 \geq 0$ or $z_1 \geq 0 \geq z_2 \geq z_3$.

Solutions of the Depressed Quartic Equation

$$T_1 = \sqrt{z_1} + \sqrt{z_2} - \Sigma s\sqrt{z_3}$$

$$T_2 = \sqrt{z_1} - \sqrt{z_2} + \Sigma s\sqrt{z_3}$$

$$T_3 = -\sqrt{z_1} + \sqrt{z_2} + \Sigma s\sqrt{z_3}$$

$$T_4 = -\sqrt{z_1} - \sqrt{z_2} - \Sigma s\sqrt{z_3}$$

$$\Sigma = \begin{cases} 1 & \text{if } b_1 > 0 \\ -1 & \text{otherwise} \end{cases} \qquad s = \begin{cases} 1 & \text{if } \sqrt{z_2}\sqrt{z_3} \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

Multiplicity 2 Relations

$\boxed{z_2 = z_3,\ \Sigma s = 1 \Leftrightarrow T_1 = T_2}$ $\quad T_1 = T_2 = \sqrt{z_1}$

If $z_2 \geq 0$, $T_3 = -\sqrt{z_1} + 2\sqrt{z_2}$, $\quad T_4 = -\sqrt{z_1} - 2\sqrt{z_2}$
If $z_2 < 0$, $T_3 = -\sqrt{z_1} + i2\sqrt{|z_2|}$, $\quad T_4 = -\sqrt{z_1} - i2\sqrt{|z_2|}$

$\boxed{z_2 = z_3,\ \Sigma s = -1 \Leftrightarrow T_3 = T_4}$ $\quad T_3 = T_4 = -\sqrt{z_1}$

If $z_2 \geq 0$, $T_1 = \sqrt{z_1} + 2\sqrt{z_2}$, $\quad T_2 = \sqrt{z_1} - 2\sqrt{z_2}$
If $z_2 < 0$, $T_1 = \sqrt{z_1} + i2\sqrt{|z_2|}$, $\quad T_2 = \sqrt{z_1} - i2\sqrt{|z_2|}$

$\boxed{z_1 = z_2 \geq z_3 \geq 0 \Leftrightarrow \text{real } T_2 = T_3}$ $\quad T_2 = T_3 = \Sigma\sqrt{z_3} \qquad T_1 = 2\sqrt{z_1} - \Sigma\sqrt{z_3}, \quad T_4 = -2\sqrt{z_1} - \Sigma\sqrt{z_3}, \quad s = 1$

Multiplicity 3 Relations

$$z_1 = z_2 = z_3,\ \Sigma s = 1 \Leftrightarrow T_1 = T_3 \Leftrightarrow T_1 = T_2 = T_3 = \sqrt{z_1}, \quad T_4 = -3\sqrt{z_1}$$

$$z_1 = z_2 = z_3,\ \Sigma s = -1 \Leftrightarrow T_2 = T_4 \Leftrightarrow T_2 = T_3 = T_4 = -\sqrt{z_1}, \quad T_1 = 3\sqrt{z_1}$$

Quartic Symmetry Relations

$z_1 = 0 \Leftrightarrow T_1 = -T_2 \Leftrightarrow T_3 = -T_4$ $\qquad$ $z_2 = 0 \Leftrightarrow T_1 = -T_3 \Leftrightarrow T_2 = -T_4$

$z_3 = 0 \Leftrightarrow T_1 = -T_4 \Leftrightarrow T_2 = -T_3$ $\qquad$ $z_2 = z_3 = 0 \Leftrightarrow T_1 = T_2 = -T_3 = -T_4 = \sqrt{z_1}$

$$z_1 = z_2 = 0,\ z_3 < 0 \Leftrightarrow s = -1, \quad T_1 = T_4 = -T_2 = -T_3 = i\Sigma\sqrt{-z_3} \Leftrightarrow s = -1, \quad T_1 = T_4$$

Quartic Multiplicity 4 Is Symmetric

$$z_1 = z_2 = z_3 = 0 \Leftrightarrow T_1 = T_2 = T_3 = T_4 = 0 \Leftrightarrow s = 1, \quad T_1 = T_4$$

The second section shows the Euler Resolvent Cubic Equation and the relationship between its coefficients a_n to the coefficients b_n of the depressed quartic equation. Solutions z_n of the resolvent cubic equation have the properties given in the third section, Properties of the Resolvent-Cubic-Equation Solutions. The property $z_2 z_3 \geq 0$ implies that if z_2 and z_3 are real, then z_2 and z_3 cannot have opposite signs. The property $z_1 \geq z_2 \geq z_3$ then implies that either $z_1 \geq z_2 \geq z_3 \geq 0$ or $z_1 \geq 0 \geq z_2 \geq z_3$.

The fourth section, Solutions of the Depressed Quartic Equation, provides formulas for T_1, T_2, T_3, and T_4 as functions of z_1, z_2, and z_3. These formulas are from this website's document https://quarticequations.com/Quartic.pdf, Equations (10) through (13). The T_n equations resemble the original Euler formulas, but they are configured so that the radical sign denotes the principal square root. Note that the value s in the figure equals 1 except when z_2 and z_3 are both real and are both less than zero.

The equivalent modified Euler formulation, the one applied in our quartic-equation algorithms, is:

$$T_1 = \sqrt{z_1} + \sqrt{x_2 + x_3 - 2\Sigma\sqrt{x_2 x_3 + y_2^2}} = \sqrt{z_1} + \sqrt{z_2 + z_3 - 2\Sigma\sqrt{z_2 z_3}}$$

$$T_2 = \sqrt{z_1} - \sqrt{x_2 + x_3 - 2\Sigma\sqrt{x_2 x_3 + y_2^2}} = \sqrt{z_1} - \sqrt{z_2 + z_3 - 2\Sigma\sqrt{z_2 z_3}}$$

$$T_3 = -\sqrt{z_1} + \sqrt{x_2 + x_3 + 2\Sigma\sqrt{x_2 x_3 + y_2^2}} = -\sqrt{z_1} + \sqrt{z_2 + z_3 + 2\Sigma\sqrt{z_2 z_3}}$$

$$T_4 = -\sqrt{z_1} - \sqrt{x_2 + x_3 + 2\Sigma\sqrt{x_2 x_3 + y_2^2}} = -\sqrt{z_1} - \sqrt{z_2 + z_3 + 2\Sigma\sqrt{z_2 z_3}}$$

These modified Euler expressions avoid the need for complex-number operations, but the formulas in Figure 11 are used there because of their simplicity in the multiplicity condition, for which the z_n are all real.

The remaining sections of Figure 11 (Multiplicity 2 Relations, Multiplicity 3 Relations, etc.) follow directly from the first four sections.

The Multiplicity 2 Relations show that a multiplicity 2 among the z_n implies multiplicity 2 among the T_n and vice versa. For the first case suppose that $z_2 = z_3$ and $\Sigma s = 1$. Then the T_n formulas in the figure give $T_1 = T_2 = \sqrt{z_1}$. T_1 and T_2 must be real because $z_1 \geq 0$.

We can also start with $T_1 = T_2$. Then the T_1 and T_2 formulas give

$$\sqrt{z_1} + \sqrt{z_2} - \Sigma s\sqrt{z_3} = \sqrt{z_1} - \sqrt{z_2} + \Sigma s\sqrt{z_3} \quad \Rightarrow \quad \sqrt{z_2} = \Sigma s\sqrt{z_3} \quad \text{AND} \quad T_1 = T_2 = \sqrt{z_1}\,.$$

T_1 and T_2 are again real because $z_1 \geq 0$. Σs can only be 1 or -1. If z_2 and z_3 are real, they cannot have opposite signs so $z_2 = z_3$ and $\Sigma s = 1$. If z_2 and z_3 are not real, then they are a complex conjugate pair. Let ϕ be the argument of z_2. Then

$$\sqrt{z_2} = \sqrt{|z_2|}e^{i\phi/2} = \Sigma s\sqrt{z_3} = \Sigma s\sqrt{|z_2|}e^{-i\phi/2} \Rightarrow e^{i\phi/2} = \pm e^{-i\phi/2}$$

If the plus sign applies, then $\phi = 0$, z_2 and z_3 are real, and $z_2 = z_3$. If the minus sign applies then

$$e^{i\phi/2} = -e^{-i\phi/2} \Rightarrow \phi/2 = \pi - \phi/2 \Rightarrow \phi = \pi,$$

which implies that z_2 and z_3 are real with equal magnitude and opposite sign, an impossibility for z_2 and z_3. Thus, only the plus sign can apply. We therefore have $[z_2 = z_3, \Sigma s = 1] \Leftrightarrow T_1 = T_2$.

A similar logic argument applies to the second multiplicity 2 case in Figure 11: $[z_2 = z_3, \Sigma s = -1] \Leftrightarrow T_3 = T_4$.

The third multiplicity 2 case is $z_1 = z_2 \geq z_3 \geq 0 \Leftrightarrow$ real $T_2 = T_3$. Demonstrate this by starting with $z_1 = z_2 \geq z_3 \geq 0$. Then the T_2 and T_3 formulas give $T_2 = T_3 = \Sigma s \sqrt{z_3}$, which must be real because $z_3 \geq 0$.

Alternatively, we can start with real $T_2 = T_3$. Then the T_2 and T_3 formulas give

$$\sqrt{z_1} - \sqrt{z_2} + \Sigma s \sqrt{z_3} = -\sqrt{z_1} + \sqrt{z_2} + \Sigma s \sqrt{z_3} \quad \Rightarrow \quad \sqrt{z_1} = \sqrt{z_2} \quad \text{AND} \quad T_2 = T_3 = \Sigma s \sqrt{z_3}$$

The value z_1 is always nonnegative real, so the equality $\sqrt{z_1} = \sqrt{z_2}$ implies $z_1 = z_2 \geq 0$. That $T_2 = T_3$ is given as real implies that z_3 is real and $z_3 \geq 0$. Thus, $z_1 = z_2 \geq z_3 \geq 0 \Leftrightarrow$ real $T_2 = T_3$.

Each of the remaining Figure 11 relations for multiplicity 3 and for symmetry can be demonstrated in similar fashion. A quartic equation is symmetric if for each depressed solution T_n, another solution T_n is its negative.

In summary, a quartic equation with multiplicity or symmetry has an Euler resolvent cubic equation with a corresponding multiplicity or zero value among its three solutions z_n.

Tables IV and V below demonstrate how the mitigation design works for the Example 2 multiplicity and Example 3 symmetry quartic equations.

Example 2 Quartic Equation with Multiplicity 3

The Example 2 quartic equation is

$$Z_n^4 - 4.2\, Z_n^3 + 6.6\, Z_n^2 - 4.6\ Z_n + 1.2 = 0 \quad \text{with true solutions} \quad 1.2,\ 1,\ 1,\ 1.$$

This is a multiplicity 3 equation: three of four solutions equal the same real value, 1. Table IV lists all of the pertinent parameters, calculated both without and with round-off error mitigation. The table is simplified by omitting calculated values from the Figure 10 dashed red box. These values apply only to quartic equations with symmetry, which is not a property of Example 2. Therefore, the inequality $|b_1| < \text{Max}(b_{1E}, b_{1E\_W})\ \varepsilon$ in the Figure 10 dashed red box is FALSE, and the red box makes no change to any relevant parameter.

Table IV. Calculated Parameters for Example 2 Quartic Equation with Multiplicity 3

Example 2 Quartic Equation: $Z_n^4 - 4.2\,Z_n^3 + 6.6\,Z_n^2 - 4.6\,Z_n + 1.2 = 0$ with solutions 1.2, 1, 1, 1				
Parameter Symbol	Figure 2 Quartic-Equation Algorithm (Value without error mitigation)	Figure 1 Cubic-Equation Algorithm (Value without error mitigation)	Figure 10 Final Quartic-Equation Algorithm (Value with error mitigation)	Figure 9 Final Cubic-Equation Algorithm (Value with error mitigation)
ε	--	--	$\varepsilon = 2^{-52} =$ 2.220446049E-16	$\varepsilon = 2^{-52} =$ 2.220446049E-16
A_3	[– 4.2]		[– 4.2]	
A_2	[6.6]		[6.6]	
A_1	[– 4.6]		[– 4.6]	
A_0	[1.2]		[1.2]	
A_{3E}	--		4.2	
A_{2E}	--		6.6	
A_{1E}	--		4.6	
A_{0E}	--		1.2	
$A_0 = 0$	FALSE		FALSE	
C	-1.05		-1.05	
C_E	--		1.05	
b_2	-0.015		-0.015	
b_{2E}	--		19.83	
b_1	-0.001		-0.001	
b_{1E}	--		32.383	
b_0	-1.875E-05		-1.875E-05	
b_{0E}	--		18.169575	
a_2	-0.0075	[-0.0075]	-0.0075	[-0.0075]
a_{2E}	--	--	9.915	[9.915]
a_1	1.875E-05	[1.875E-05]	1.875E-05	[1.875E-05]
a_{1E}	--	--	4.579575	[4.579575]
a_0	-1.5625E-08	[-1.5625E-08]	-1.5625E-08	[-1.5625E-08]
a_{0E}	--	--	0.001011969	[0.001011969]
Calculations from the Figure 10 dashed red box are irrelevant and omitted here.				
q		1.455880E-17		1.455880E-17
q_E		--		1.54305
$q_E\,\varepsilon$		--		3.42626E-16
q reset		--		0
r		-3.777899E-20		-3.777899E-20
r_E		--		0.006261438
$r_E\,\varepsilon$		--		1.39032E-18
r reset		--		0
$R = r^2 + q^3$		1.427252E-39		0
R_E				0
$\lvert R\rvert < R_E\,\varepsilon$				FALSE
$q = r = 0$				TRUE

Table IV Calculated Parameters for Example 2 Quartic Equation with Multiplicity 3 (Page 2)

Example 2 Quartic Equation: $Z_n^4 - 4.2\, Z_n^3 + 6.6\, Z_n^2 - 4.6\ Z_n + 1.2 = 0$ with solutions 1.2, 1, 1, 1				
Parameter Symbol	Figure 2 Quartic-Equation Algorithm (Value without error mitigation)	Figure 1 Cubic-Equation Algorithm (Value without error mitigation)	Figure 10 Final Quartic-Equation Algorithm (Value with error mitigation)	Figure 9 Final Cubic-Equation Algorithm (Value with error mitigation)
		Numerical Recipes		Special Case 2 q = r = 0
A		4.227596E-07		
u		3.443755E-11		
v		-4.227596E-07		
t_1		-4.227251E-07		
t_{2x}		2.113626E-07		
y_2		3.661504E-07		
t_{3x}		2.113626E-07		
z_1	[0.002499577]	0.002499577	[0.0025]	0.0025
x_2	[0.002500211]	0.002500211	[0.0025]	0.0025
y_2	[3.661504E-07]	3.661504E-07	[0]	0
x_3	[0.002500211]	0.002500211	[0.0025]	0.0025
Σ	–1		–1	
d	6.251057E-06		6.25E-06	
s_{z1}	0.049995773		0.05	
D	0.010000846		0.01	
s_D	0.100004227		0.1	
Y_1	0		0	
T_{x1}	0.15		0.15	
T_{x2}	-0.050008455		-0.05	
D	-5.36219E-11		0	
s_D	7.322698E-06		0	
Y_3	7.322698E-06		0	
T_{x3}	-0.049995773		-0.05	
T_{x4}	-0.049995773		-0.05	
X_1	1.2		1.2	
X_2	0.999991545		1	
Y_1	0		0	
X_3	1.000004227		1	
X_4	1.000004227		1	
Y_3	7.322698E-06		0	

The second and third table columns are the "without mitigation" columns. They list parameters calculated by the Figure 2 quartic-equation algorithm and Figure 1 cubic-equation algorithm. The fourth and fifth columns list parameters calculated by the final algorithms in Figures 10 and 9 with mitigation. Entries enclosed in square brackets are input values, either from the user or from another algorithm in the table.

The two quartic-equation algorithms produce identical values for the resolvent-cubic-equation coefficients: $a_2 = -0.0075$, $a_1 = 1.875\times10^{-5}$, $a_0 = -1.5625\times10^{-8}$. The respective cubic-equation algorithms use the calculated coefficient values to solve the resolvent cubic equation. There, the true values of parameters q and r are both zero: $q = r = 0$. This case (Special Case 2) implies that the depressed cubic equation is $t_n^3 = 0$, $n = 1, 2, 3$, and the resolvent-cubic-equation solutions all equal the same real value: $z_1 = z_2 = z_3 = -a_2/3 = 0.0025$.

Instead of zero, both cubic algorithms calculate the q and r values as round-off errors $q = 1.45588\times10^{-17}$ and $r = -3.7779\times10^{-20}$. The true value of $R = r^2 + q^3$ is zero, but the Figure 1 cubic-equation algorithm calculates $R = 1.42725\times10^{-39}$.

Because the calculated R is positive, the original Figure 1 algorithm uses Numerical Recipes to complete the calculation, starting with the calculation of A (true value is zero):

$$A = \left(|r| + \sqrt{R}\right)^{1/3}$$

$$= \left(|-3.7779\times10^{-20}| + \sqrt{1.42725\times10^{-39}}\right)^{1/3} = (7.5558\times10^{-20})^{1/3} = 4.227596\times10^{-7}$$

The square root and cube root operations in the formula for A greatly magnify the round-off errors of r and R. The Figure 1 and Figure 2 algorithms then go on to infect their calculated solutions of the resolvent cubic equation and quartic equation with this magnified round-off error. Instead of $z_1 = z_2 = z_3 = 0.0025$, the calculated resolvent-cubic-equation solutions are 0.002499577 and $0.002500211 \pm i\ 3.6615\times10^{-7}$. Instead of 1.2, 1, 1, 1, the calculated quartic-equation solutions are

$$1.2, \quad 0.999991545140, \quad \text{and} \quad 1.000004227430 \pm i\ 0.000007322698.$$

The algorithms with round-off error mitigation avoid these magnified errors. The Figure 9 cubic-equation algorithm calculates the error size parameters q_E and r_E, and finds that

$$|q| = 1.455880\times10^{-17} < q_E\,\varepsilon = 3.42626\times10^{-16} \quad \text{and}$$

$$|r| = 3.777899\times10^{-20} < r_E\,\varepsilon = 1.39032\times10^{-18}.$$

The algorithm therefore resets both q and r to zero. It proceeds with Special Case 2 to accurately calculate the resolvent-cubic-equation solutions. These are then used by the Figure 10 algorithm to calculate accurate quartic-equation solutions.

Example 3 Quartic Equation Symmetry Condition

The Example 3 quartic equation is

$$Z_n^4 - 8\,Z_n^3 - 5.84\,Z_n^2 + 87.36\ Z_n + 17.64 = 0 \text{ with true solutions 7, 4.2, –0.2 and –3.}$$

This equation is symmetric: the quartic polynomial and the four solutions are symmetrical about the value $Z = Z_C = 2$. Table V lists all of the pertinent parameters, calculated both without and with round-off error mitigation. The columns of the table correspond to those of the previous table (Table IV) except that a sixth column has been added for the Figure 8 Final Quadratic-Equation Algorithm.

Table V. Calculated Parameters for Example 3 Quartic Equation with Symmetry

Example 3 Quartic Equation: $Z_n^4 - 8\,Z_n^3 - 5.84\,Z_n^2 + 87.36\ Z_n + 17.64 = 0$ with solutions 7, 4.2, −0.2 and −3					
Parameter Symbol	Figure 2 Quartic-Equation Algorithm	Figure 1 Cubic-Equation Algorithm	Figure 10 Final Quartic-Equation Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 8 Final Quadratic-Equation Algorithm
	(Value without error mitigation)		(Value with error mitigation)		
ε			$\varepsilon = 2^{-52} =$ 2.2204460E-16	$\varepsilon = 2^{-52} =$ 2.2204460E-16	$\varepsilon = 2^{-52} =$ 2.2204460E-16
A_3	[-8]		[-8]		
A_2	[-5.84]		[-5.84]		
A_1	[87.36]		[87.36]		
A_0	[17.64]		[17.64]		
$A_0 = 0$	FALSE		FALSE		
A_{3E}			8		
A_{2E}			5.84		
A_{1E}			87.36		
A_{0E}			17.64		
C	-2		-2		
C_E			2		
b_2	-29.84		-29.84		
b_{2E}			53.84		
b_1	0		0		
b_{1E}			326.08		
b_0	121		121		
b_{0E}			279.72		
a_2	-14.92	[-14.92]	-14.92	[-14.92]	[-14.92] = B
a_{2E}			26.92	[26.92]	[26.92] = B_E
a_1	25.4016	[25.4016]	25.4016	[25.4016]	[25.4016] = C
a_{1E}			270.7532	[270.7532]	[270.7532] = C_E
a_0	0	[0]	0		
a_{0E}			0	[0]	
b_{1E_W}			0		
$b_2 < 0$			TRUE		
$\lvert a_1/b_2^2\rvert < 1\times10^{-8}$			FALSE		
$\lvert b_1\rvert < \mathrm{Max}(b_{1E}, b_{1E\_W})\ \varepsilon$			TRUE		
reset b_1			0		
reset a_0			0	[0]	
$\lvert a_1\rvert < a_{1E}\ \varepsilon$			FALSE		
q		-16.26684444			
r		59.8453357			
$r^2 + q^3$		-722.9092147			
		Viète $r^2 + q^3 \le 0$		Special Case 1 $a_0 = 0$	
θ		0.422250267			
ϕ_1		0.140750089			
ϕ_2		-1.953645014			
ϕ_3		2.235145191			

Table V Calculated Parameters for Example 3 Quartic Equation with Symmetry (Page 2)

Example 3 Quartic Equation: $Z_n^4 - 8\,Z_n^3 - 5.84\,Z_n^2 + 87.36\ Z_n + 17.64 = 0$ with solutions 7, 4.2, –0.2 and –3					
Parameter Symbol	Figure 2 Quartic-Equation Algorithm	Figure 1 Cubic-Equation Algorithm	Figure 10 Final Quartic-Equation Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 8 Final Quadratic-Equation Algorithm
	(Value without error mitigation)		(Value with error mitigation)		
t_1		7.986666667			
t_{2x}		-3.013333333			
y_2		0			
t_{3x}		-4.973333333			
D					121
D_E					1886.3056
Y				[0]	0
Q					12.96
X_1				[12.96]	12.96
X_2				[1.96]	1.96
z_1	[12.96]	12.96	[12.96]	12.96	
x_2	[1.96]	1.96	[1.96]	1.96	
y_2	[0]	0	[0]	0	
x_3	[1.776357E-15]	1.776357E-15	[0]	0	
Σ	-1		-1		
d	3.481659E-15		0		
s_{z1}	3.6		3.6		
D	1.960000118		1.96		
s_D	1.400000042		1.4		
Y_1	0		0		
T_{x1}	5.000000042		5		
T_{x2}	2.199999958		2.2		
D	1.959999882		1.96		
s_D	1.399999958		1.4		
Y_3	0		0		
T_{x3}	-2.200000042		-2.2		
T_{x4}	-4.999999958		-5		
X_1	7.000000042		7		
X_2	4.199999958		4.2		
X_3	-0.200000042		-0.2		
X_4	-2.999999958		3		

Both quartic-equation algorithms produce true values for the coefficients ($a_2 = -14.92$, $a_1 = 25.4016$, $a_0 = 0$) of the resolvent cubic equation, which is solved by the respective cubic-equation algorithms. The resolvent-cubic-equation true solutions are $z_1 = 12.96$, $z_2 = x_2 = 1.96$, and $z_3 = x_3 = 0$. The constant coefficient value $a_0 = 0$ indicates that one solution z_n must be zero.

The Figure 1 cubic-equation algorithm without mitigation does not treat the case $a_0 = 0$ as a special case, and so proceeds as usual. Instead of 0, the calculated z_3 is the small error value $z_3 = x_3 = 1.77636\times10^{-15}$. The parameter $d = x_2x_3 + y_2y_2$ in the quartic-equation algorithm has true value 0, but is now calculated as $d = 3.48\times10^{-15}$. This small error for d is magnified when the two formulas $D = x_2 + x_3 - 2\Sigma\sqrt{d}$ and $D = x_2 + x_3 + 2\Sigma\sqrt{d}$ take its square root. Both D values should be $D = x_2 = 1.96$, but they are calculated as 1.960000118 and 1.959999882 (relative error ≈ 6×10^{-8}). The resulting quartic-equation calculated solutions Z_n suffer a similar relative error.

$$Z_1 = 7.000000042,\quad Z_2 = 4.199999958,\quad Z_3 = -0.200000042,\quad Z_4 = -2.999999958$$

The algorithms with round-off error mitigation avoid these magnified errors. In solving the resolvent cubic equation, the case $a_0 = 0$ is Special Case 1 in the Figure 9 algorithm. The algorithm sets one solution to zero and invokes the Figure 8 quadratic-equation algorithm to calculate the other two (12.96 and 1.96) as solutions of $Z_n^2 + a_2Z_n + a_0 = 0$ (last column of the table on the second page). The Figure 9 cubic-equation algorithm conveys these three accurate solutions to the Figure 10 quartic-equation algorithm, which then accurately calculates the quartic-equation solutions.

We have shown that the final algorithms in Figures 8, 9, and 10 eliminate round-off error magnification in the first three example problems, but the algorithms address neither the symmetry near-miss condition in Example 4 nor the magnification condition in Example 5. Those two conditions are handled by post processing as described in the next two sections.

V. CUBIC-EQUATION POST PROCESSING FOR SMALL MAGNITUDE SOLUTIONS

This section describes cubic-equation post processing to eliminate round-off-error magnification for magnitude-condition equations: equations with at least two solutions that differ significantly in magnitude. The cubic-equation algorithm provides good accuracy for the larger-magnitude solution(s), but round-off error can swamp the smaller-magnitude solution(s) as we will demonstrate. To correct this situation, post processing applies the accurately-calculated, large-magnitude solution(s) to the cubic-equation coefficients to extract accurate values of the small-magnitude solution(s). The post-processing is summarized in a detailed calculation flow chart.

The cubic-equation post processing addresses not only the cubic-equation magnitude condition, but also quartic-equation symmetry near-miss. The reason is that such a quartic equation has a resolvent cubic equation with the magnitude condition. We work through the Table I, Example 4 quartic equation to demonstrate.

Simple Example Calculations

A simple example demonstrates why the cubic-equation algorithm has difficulty with extreme magnitude differences between equation solutions. Let the solutions z_1, z_2, and z_3 of a cubic equation $z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0$ be the three real values 2, 1, and 1×10^{-17}. From Equations (2), the equation coefficients are

$$\begin{aligned}
a_2 &= -(z_1 + z_2 + z_3) &&= -(2 + 1 + 1\times10^{-17}) &&= -(3 + 1\times10^{-17}) \\
a_1 &= z_1z_2 + z_1z_3 + z_2z_3 &&= 2\cdot1 + (2 + 1)\cdot(1\times10^{-17}) &&= 2 + 3\times10^{-17} \\
a_0 &= -z_1z_2z_3 &&= -2\cdot1\cdot1\times10^{-17} &&= -2\times10^{-17}
\end{aligned}$$

The computer's limited precision forces it to store a_2 as -3 and a_1 as 2. This limitation is not a problem because the value $a_0 = -2\times10^{-17}$ retains the needed information about the small-magnitude solution $z_3 = 1\times10^{-17}$.

The problem occurs in the cubic-equation algorithm with evaluation of parameters q and r:

$$q = \frac{a_1}{3} - \frac{a_2^2}{9} = \frac{2}{3} - \frac{(-3)^2}{9} = -\frac{1}{3}$$

$$r = \frac{a_1a_2 - 3a_0}{6} - \frac{a_2^3}{27} = \frac{2(-3) - 3(-2\times10^{-17})}{6} - \frac{(-3)^3}{27} = \frac{-6 + 6\times10^{-17}}{6} + 1$$

The last expression on the right shows that the true value of r is

$$r = \frac{-6 + 6\times10^{-17}}{6} + 1 = -1 + 1\times10^{-17} + 1 = 1\times10^{-17}$$

However, the computer with its limited precision first calculates the numerator $-6+6\times10^{-17}$ as -6. The calculated value of r becomes $-6/6 + 1 = 0$. At this point, all trace of $z_3 = 1\times10^{-17}$ has vanished from the computer calculation. With $q = -1/3$ and $r = 0$, the cubic-equation algorithm, Figure 9, proceeds with Special Case 4 to produce $s = \sqrt{|3q|} = 1$, $y_2 = t_{x2} = 0$, $t_1 = s = 1$, $t_{x3} = -s = -1$,

$$z_1 = t_1 - a_2/3 = 1 - (-3)/3 = 2,$$
$$z_2 = x_2 = t_{x2} - a_2/3 = 0 + 1 = 1,$$
$$z_3 = x_3 = t_{x3} - a_2/3 = -1 + 1 = 0.$$

The two larger solutions are accurate, but z_3 cannot possibly be 0 because a_0 is not 0.

Realizing that the Figure 9 algorithm produces two solutions, z_1 and $z_2 = x_2$, of the same order of magnitude and the third one, $z_3 = x_3$, much smaller, we can safely assume that z_1 and z_2 are accurate and, therefore, accurately recalculate z_3 by using the constant coefficient $a_0 = -z_1z_2z_3$:

$$z_3 = -\frac{a_0}{z_1z_2} = -\frac{-2\times10^{-17}}{2\cdot1} = 1\times10^{-17}.$$

To describe the general case, we relabel solutions from the Figure 9 algorithm as z_A, z_B, and z_C where the indices for the new labels indicate the order of absolute value: $|z_A| \geq |z_B| \geq |z_C|$. Equations (2) for the cubic-equation coefficients become

$$a_2 = -(z_A + z_B + z_C) \qquad a_1 = z_Az_B + z_Az_C + z_Bz_C \qquad a_0 = -z_Az_Bz_C. \qquad (53)$$

When $|z_A|$ and $|z_B|$ are of the same order of magnitude, but $|z_C|$ is much smaller, then the calculated z_A and z_B values are accurate, but z_C is suspect. Post processing applies the Equation (53) formula for a_0 to recalculate z_C from a_0, z_A, and z_B:

$$z_C = -\frac{a_0}{z_Az_B}. \qquad (54)$$

Unlike the z_C calculated by the Figure 9 algorithm, this z_C is recalculated directly from a_0 using the accurately calculated z_A and z_B values.

This approach is the same applied in the Numerical Recipes algorithm (Figure 4) to solve the quadratic equation $Z_n^2 + B\,Z_n + C = 0$. Coefficients B and C are related to the two solutions Z_1 and Z_2 by

$$B = -(Z_1 + Z_2) \text{ and } C = Z_1\,Z_2. \qquad (55)$$

For two solutions of unequal magnitude, the solutions must be real, the determinate $D = B^2 - 4C$ is positive, and parameter $Q = (|B| + \sqrt{D})/2$ is the absolute value of the larger-magnitude solution Z_A. The sign of Z_A is the opposite that of B, and the smaller magnitude solution is calculated as $Z_B = C/\,Z_A$.

A related approach applies to cubic equations when the magnitude of one solution from the Figure 9 algorithm is significantly greater than that of the other two, that is, $|z_A| >> |z_B| \geq |z_C|$. This time the accuracy of the Figure 9 algorithm can be trusted only for the largest magnitude value z_A, and post processing develops a quadratic equation $Z_n^2 + B\,Z_n + C = 0$ to accurately recalculate the values of $z_B = Z_1$ and $z_C = Z_2$. We apply the accurate solution z_A to extract the values of B and C from the cubic-equation coefficients a_0 and a_1. From

Equation (55), $C = Z_1\,Z_2 = z_B\,z_C$. Equation (53) gives $a_0 = -z_Az_Bz_C$. We therefore divide $-a_0$ by z_A to obtain C:

$$C = -a_0/z_A. \tag{56}$$

This value of $C = z_B\,z_C$ and the solution z_A allow us to extract the formula for $B = -(z_B + z_C)$ from a_1 by using the Equation (53) expression for a_1:

$$a_1 = z_Az_B + z_Az_C + z_Bz_C = z_A(z_B + z_C) + C = -\,z_AB + C \quad \Rightarrow$$

$$B = (C - a_1)/z_A \tag{57}$$

Post processing uses (56) and (57) for B and C, then solves the associated quadratic equation. The resulting quadratic-equation solutions are the recalculated values of the cubic equation's two smaller-magnitude solutions z_B and z_C.

Consider the following example of a cubic equation with solutions $z_A = -3$, $z_{B,} = 2\times10^{-17}$, and $z_C = 1\times10^{-17}$. The computer-stored, cubic-equation coefficients are

$a_2 = 3$ $\qquad$ $a_1 = -9\times10^{-17}$ $\qquad$ $a_0 = 6\times10^{-34}$

The Figure 9 cubic-equation algorithm calculates the solutions as 0, 0, –3.

The calculated values $z_B = z_C = 0$ cannot be correct because $a_0 \neq 0$. Our post-processing calculates C and B as

$C = -a_0/z_A = -(6\times10^{-34})/(-3) = 2\times10^{-34}$

$B = (C - a_1)/\,z_A = [2\times10^{-34} - (-9\times10^{-17})]/(-3) = -\,3\times10^{-17}$

The recalculated values of z_B and z_C are the two solutions of

$$Z_n^2 - 3\times10^{-17}\;Z_n + 2\times10^{-34} = 0.$$

The quadratic formula is adequate to accurately produce the proper solutions for this particular case:

$$z_{B,C} = \frac{1}{2}\left(-B \pm \sqrt{B^2 - 4C}\right) = \frac{1}{2}\left(3\times10^{-17} \pm \sqrt{9\times10^{-34} - 8\times10^{-34}}\right)$$

$$z_B = \frac{1}{2}(3\times10^{-17} + 1\times10^{-17}) = 2\times10^{-17} \qquad z_C = \frac{1}{2}(3\times10^{-17} - 1\times10^{-17}) = 1\times10^{-17}$$

Cubic-Equation Post-Processing Algorithm

In principle, post processing is quite simple. The Figure 9 cubic-equation provides its solutions z_A, z_B, and z_C of the cubic equation $z_n^3 + a_2\,z_n^2 + a_1\,z_n + a_0 = 0$ where $|z_A| \geq |z_B| \geq |z_C|$.

- If $|z_A| >> |z_B|$, then z_A is real and post processing recalculates z_B and z_C as the solutions of $Z_n^2 + BZ_n + C = 0$ where

$$C = -a_0/z_A \text{ and } B = (C - a_1)/\,z_A.$$

- Otherwise, if $|z_B| >> |z_C|$, then z_C is real and post processing recalculates real z_C as

$$z_C = -\frac{a_0}{z_A z_B}.$$

The actual cubic-equation post processing algorithm, Figure 12, is more complicated. It addresses the specific circumstances that trigger solution recalculation and the mechanism for relating z_A, z_B, and z_C to the Figure 9 algorithm outputs z_1, x_2, x_3, y_2.

The post-processing inputs are the cubic-equation coefficients a_2, a_1, and a_0, the associated error size parameters a_{2E}, a_{1E}, and a_{0E}, and the Figure 9 outputs z_1, x_2, x_3, and y_2. Post processing also utilizes a stored adjustable constant $\zeta = 0.345$ for determining whether solutions z_B and z_C are to be recalculated. Post processing recalculates both z_B and z_C if $|z_B| \leq \zeta\,|z_A|$. It recalculates only z_C if $|z_C| < \zeta\,|z_A| < |z_B|$. A ζ-value of 1 implies that z_B and z_C are always recalculated unless they have the same absolute value as z_A. A ζ-value of 0 implies that z_B and z_C are never recalculated. The theoretical range of ζ is $0 \leq \zeta \leq 1$, but the Section X error analysis shows that the selected value $\zeta = 0.345$ provides the best solution accuracy for our mitigation design.

The post-processing first step is calculating the absolute values of the Figure 9 solutions:

$$z_{1M} = |z_1|, \qquad z_{2M} \equiv |z_2| = \sqrt{x_2^2 + y_2^2}, \qquad z_{3M} \equiv |z_3| = \sqrt{x_3^2 + y_2^2}\,.$$

The algorithm then branches on the logical variable $\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\,\mathrm{MAX}(z_{1M}, z_{3M})$. The algorithm never explicitly evaluates z_A, z_B, and z_C (which are generally complex), but rather applies the corresponding solution components z_1, x_2, x_3, and y_2 as appropriate. Note that $|z_C| = \mathrm{MIN}(z_{1M}, z_{2M}, z_{3M})$, and $|z_A| = \mathrm{MAX}(z_{1M}, z_{3M})$. The greatest magnitude solution z_A equals either z_1 or z_3 because z_1 is the greatest real solution and $z_3 \leq z_2 \leq z_1$ when $y_2 = 0$.

If $\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\,\mathrm{MAX}(z_{1M}, z_{3M}) \Leftrightarrow |z_C| < \zeta\,|z_A|$, then at least one solution requires recalculation.

Then if $y_2 \neq 0$, solutions z_2 and z_3 are a complex conjugate pair, and there are only two possibilities:

1 $z_{1M} > z_{3M}$. In this case, the greatest magnitude solution is the real value $z_A = x_A = z_1$, and the complex conjugate pair z_2 and z_3 need to be recalculated using a quadratic equation.

2 Otherwise, z_A and z_B are the complex conjugate pair z_2 and z_3. The small-magnitude solution $z_C = z_1$ is recalculated via Equation (54).

$$z_1 = -\frac{a_0}{z_2 z_3} = -\frac{a_0}{x_2^2 + y_2^2} = -a_0/x_{2M}^2\,.$$

Figure 12 Cubic-Equation Post-Processing Algorithm

Post -Processing Constant: $\zeta = 0.345$

Inputs: For the cubic equation $z_n^3 + a_2 z_n^2 + a_1 z_n + a_0 = 0$,

1) the real coefficients a_2, a_1, and a_0 and error size parameters a_{2E}, a_{1E}, a_{0E}
2) real values z_1, x_2, x_3, y_2 as calculated by the Figure 9 algorithm so that z_1, $z_2 = x_2 + iy_2$, $z_3 = x_3 - iy_2$ are the three solutions.

Outputs: Recalculated real and imaginary component (z_1, x_2, x_3, and/or y_2) of any solution z_n such that $|z_n| < \zeta$ Max($|z_1|$, $|z_3|$).

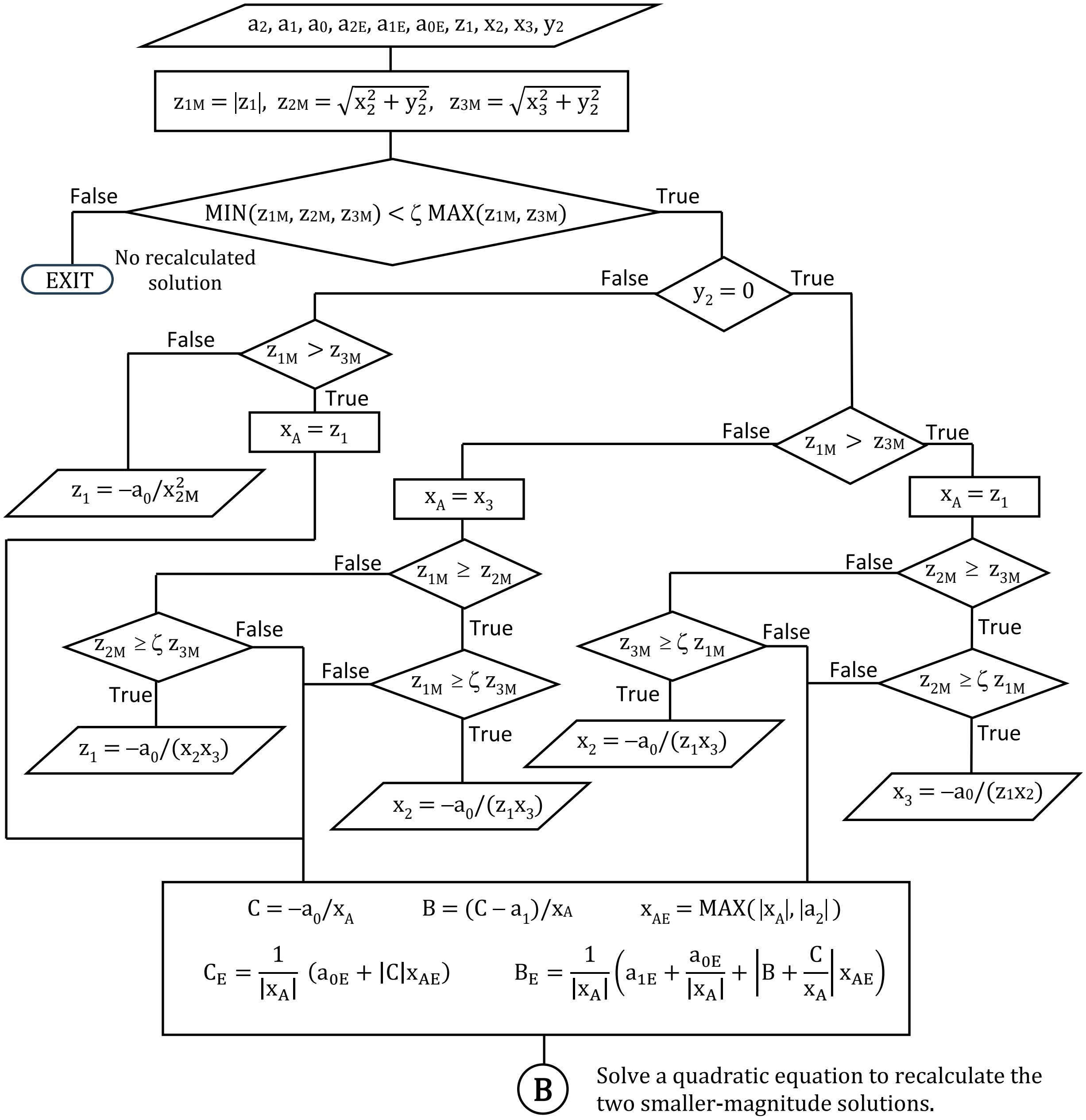
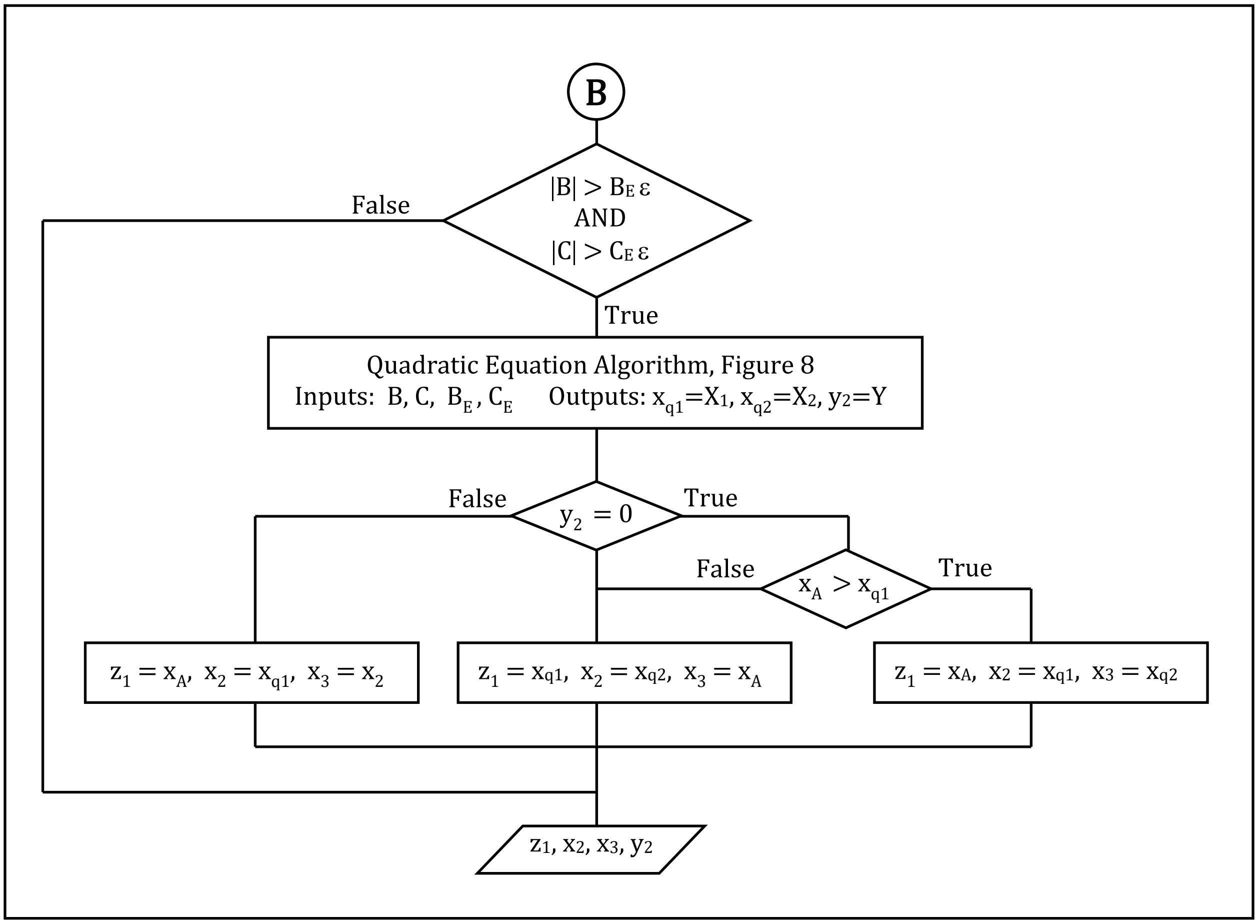

$a_2, a_1, a_0, a_{2E}, a_{1E}, a_{0E}, z_1, x_2, x_3, y_2$
$z_{1M} = |z_1|,\ z_{2M} = \sqrt{x_2^2 + y_2^2},\ z_{3M} = \sqrt{x_3^2 + y_2^2}$
False
$\text{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\ \text{MAX}(z_{1M}, z_{3M})$
True
EXIT
No recalculated solution
False
$y_2 = 0$
True
False
$z_{1M} > z_{3M}$
True
$x_A = z_1$
$z_1 = -a_0/x_{2M}^2$
False
$z_{1M} > z_{3M}$
True
$x_A = x_3$
$x_A = z_1$
False
$z_{1M} \geq z_{2M}$
True
False
$z_{2M} \geq z_{3M}$
True
$z_{2M} \geq \zeta\ z_{3M}$
False
True
False
$z_{1M} \geq \zeta\ z_{3M}$
True
$z_{3M} \geq \zeta\ z_{1M}$
False
True
False
$z_{2M} \geq \zeta\ z_{1M}$
True
$z_1 = -a_0/(x_2 x_3)$
$x_2 = -a_0/(z_1 x_3)$
$x_2 = -a_0/(z_1 x_3)$
$x_3 = -a_0/(z_1 x_2)$
$C = -a_0/x_A$
$B = (C - a_1)/x_A$
$x_{AE} = \text{MAX}(|x_A|, |a_2|)$
$C_E = \frac{1}{|x_A|}(a_{0E} + |C| x_{AE})$
$B_E = \frac{1}{|x_A|}\left(a_{1E} + \frac{a_{0E}}{|x_A|} + \left|B + \frac{C}{x_A}\right| x_{AE}\right)$
B
Solve a quadratic equation to recalculate the two smaller-magnitude solutions.

Figure 12 Cubic Equation Post Processing Algorithm (Page 2)


B
False
$|B| > B_E\,\varepsilon$
AND
$|C| > C_E\,\varepsilon$
True
Quadratic Equation Algorithm, Figure 8
Inputs: B, C, B_E, C_E Outputs: $x_{q1}=X_1$, $x_{q2}=X_2$, $y_2=Y$
False
$y_2 = 0$
True
False
$x_A > x_{q1}$
True
$z_1 = x_A,\ x_2 = x_{q1},\ x_3 = x_2$
$z_1 = x_{q1},\ x_2 = x_{q2},\ x_3 = x_A$
$z_1 = x_A,\ x_2 = x_{q1},\ x_3 = x_{q2}$
z_1, x_2, x_3, y_2

If $y_2 = 0$, then all three solutions are real, and the algorithm logic determines which of four possible orders applies among z_{1M}, z_{2M}, and z_{3M}.

		z_C	z_B	z_A
1	$z_{1M} < z_{2M} < z_{3M}$	z_1	x_2	x_3
2	$z_{2M} \le z_{1M} < z_{3M}$	x_2	z_1	x_3
3	$z_{2M} < z_{3M} \le z_{1M}$	x_2	x_3	z_1
4	$z_{3M} \le z_{2M} \le z_{1M}$	x_3	x_2	z_1

Variable x_A is assigned to z_A, which is either x_3 or z_1. The algorithm determines whether $|z_B| > \zeta\,|z_A|$ for each possible order. If so, then only solution z_C is recalculated via $z_C = -a_0/(z_A z_B)$, Equation (54).

Otherwise, both z_B and z_C are recalculated by solving a quadratic equation. The algorithm calculates the quadratic-equation coefficients C and B via Equations (56) and (57) with z_A replaced by its equivalent x_A:

$$C = -a_0/x_A \quad \text{and} \quad B = (C - a_1)/\,x_A. \tag{58}$$

The quadradic-equation algorithm, Figure 8, also requires the error size parameters C_E and B_E. To find C_E, take the partial derivatives of C with respect to a_0 and x_A.

$$C_E = \left|\frac{\partial C}{\partial a_0}\right| a_{0E} + \left|\frac{\partial C}{\partial x_A}\right| x_{AE} \quad = \quad \frac{a_{0E}}{|x_A|} + \frac{|a_0|}{x_A^2} x_{AE}$$

$$C_E = \frac{1}{|x_A|}\,(a_{0E} + |C| x_{AE}) \tag{59}$$

The values of a_0, a_{0E}, and x_A are all known, but the error-size parameter x_{AE} is not. We know that x_A is the real, greatest-magnitude, accurately-calculated solution of the cubic equation. So, one option is to just set $x_{AE} = |x_A|$. Another is to set $x_{AE} = |a_2|$ because

$$a_2 = -(z_1 + z_2 + z_3) = -(x_A + z_B + z_C) \approx -x_A$$

when $|x_A| >> |z_B|$ and $|x_A| >> |z_C|$. This condition is the primary reason for performing the post processing. Because we want $x_{AE}\,\varepsilon$ to be an easily-calculated, reasonable, upper bound of the round-off error in x_A, we opt to calculate x_{AE} as

$$x_{AE} = \mathrm{MAX}(\,|x_A|, |a_2|\,). \tag{60}$$

Derive the formula for B_E by again taking partial derivatives of B in Equation (58) with respect to a_1, a_0 and x_A.

$$B_E = \left|\frac{\partial B}{\partial a_1}\right| a_{1E} + \left|\frac{\partial B}{\partial C}\,\frac{\partial C}{\partial a_0}\right| a_{0E} + \left|\frac{\partial B}{\partial x_A} + \frac{\partial B}{\partial C}\frac{\partial C}{\partial x_A}\right| x_{AE}$$

$$B_E = \frac{1}{|x_A|}\left(a_{1E} + \frac{a_{0E}}{|x_A|} + \left|B + \frac{C}{x_A}\right| x_{AE}\right) \tag{61}$$

After the post-processing algorithm executes Equations (58) to (61) to obtain B, C, B_E, and C_E, it checks whether $|B| > B_E\,\varepsilon$ and $|C| > C_E\,\varepsilon$. That is, it checks whether B and C have absolute values that exceed their upper-bound, round-off error magnitudes. This step is necessary for cubic equations that are a resolvent cubic equation of a quartic equation, the quartic equation has multiplicity or multiplicity near miss, and the cubic coefficient A_3 has a very large absolute value. In such a situation, round-off error may dominate B and/or C, and solutions recalculated with the quadradic algorithm would be less accurate than the original calculated solutions. Therefore, if $[\,|B| > B_E\,\varepsilon$ AND $|C| > C_E\,\varepsilon\,]$ = FALSE, post processing performs no recalculation, and it simply returns the solutions calculated by the cubic-equation algorithm.

The usual situation is $[\,|B| > B_E\,\varepsilon$ AND $|C| > C_E\,\varepsilon\,]$ = TRUE, and post processing invokes the Figure 8 quadratic-equation algorithm. The Figure 8 outputs x_{q1}, x_{q2}, and y_2 are components of cubic-equation solutions $z_B = x_{q1} + iy_2$ and $z_C = x_{q2} - iy_2$. The imaginary component y_2 is nonnegative. If $y_2 = 0$, then $x_{q1} \geq x_{q2}$.

Finally, post processing assigns the real components x_A, x_{q1}, x_{q2} of the cubic-equation solutions to z_1, x_2, and x_3 where z_1 is the greatest real solution. Recall that the real solution

x_A is the greatest-magnitude solution, which must equal either z_1 or x_3. If $y_2 \neq 0$, then $z_1 = x_A$, and $x_2 = x_3 = x_{q1} = x_{q2}$. Otherwise, $y_2 = 0$ and z_1 is the greater of x_A or x_{q1}. If $x_A > x_{q1}$, then $z_1 = x_A$, $x_2 = x_{q1}$, and $x_3 = x_{q2}$. Otherwise, $z_1 = x_{q1}$, $x_2 = x_{q2}$, and $x_3 = x_A$.

Quartic-Equation Symmetry Near Miss (Table I, Example 4)

Solving the Example 4 quartic equation with the symmetry near-miss condition demonstrates the operation and effectiveness of cubic-equation post processing. Cubic-equation post processing is relevant because a symmetry-near-miss quartic equation has a resolvent cubic equation with the magnitude condition. If a quartic equation is perfectly symmetric, then the resolvent cubic equation has at least one solution equal to zero. The resolvent cubic equation therefore has a constant coefficient $a_0 = 0$, Special Case 1. If the quartic equation is a symmetry near miss, then the resolvent cubic equation has a solution of very small magnitude relative to the greatest-magnitude solution. That is, the resolvent cubic equation has the magnitude condition and needs cubic-equation post processing.

The Example 4 quartic equation is

$$\boxed{Z_n^4 - 7.9999999\,Z_n^3 - 5.84000082\,Z_n^2 + 87.35999958\;Z_n + 17.64000882 = 0}$$

with true solutions 7, 4.2, –0.2000001, and –3. This example 4 equation is a modification of the Example 3 quartic equation

$$Z_n^4 - 8\,Z_n^3 - 5.84\,Z_n^2 + 87.36\;Z_n + 17.64 = 0$$

with true solutions 7, 4.2, –0.2 and –3 and with symmetry about the value $Z = Z_C = 2$.

When we use cubic-equation post processing, the solutions of the Example 4 symmetry-near-miss equation are calculated accurately: solution relative error is less than 10^{-16}. Solution relative error without cubic-equation post processing is on the order of 10^{-7}.

EXAMPLE 4 CALCULATED SOLUTIONS
WITH AND WITHOUT CUBIC-EQUATION POST PROCESSING

with	7.000000000000	4.200000000000	–0.200000100000	–3.000000000000
without	7.000000017147	4.199999982853	–0.200000117147	–2.999999982853

Table VI, which is over two pages long, lists all of the pertinent Example 4 parameters, calculated both with and without cubic-equation post processing. The values listed in the second column are those from the Figure 10 quartic-equation algorithm with no post processing; values listed in last column are those from the same algorithm with cubic-equation post processing. Entries on each row of the table's first page are the same for these two columns. The third column corresponds to the Figure 9 cubic-equation algorithm, which calculates solutions to the resolvent cubic equation. The fourth column corresponds to the Figure 12 cubic-equation post-processing algorithm, which invokes the Figure 8 quadratic-equation algorithm in the fifth column.

Table VI. Calculated Parameters for Example 4 Quartic Equation with Symmetry Near Miss

Example 4 Quartic Equation:

$$Z_n^4 - 7.9999999\,Z_n^3 - 5.84000082\,Z_n^2 + 87.35999958\,Z_n + 17.64000882 = 0$$

with solutions 7, 4.2, –0.2000001, and –3

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm with no Post Processing	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm	Figure 10 Final Quartic-Equation Algorithm with Cubic-Equation Post Processing
	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]	[$\zeta = 0.345$]	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]
A_3	[-7.9999999]				[-7.9999999]
A_2	[-5.84000082]				[-5.84000082]
A_1	[87.35999958]				[87.35999958]
A_0	[17.64000882]				[17.64000882]
$A_0 = 0$	FALSE				FALSE
A_{3E}	7.9999999				7.9999999
A_{2E}	5.84000082				5.84000082
A_{1E}	87.35999958				87.35999958
A_{0E}	17.64000882				17.64000882
C	-1.999999975				-1.999999975
C_E	1.999999975				1.999999975
b_2	-29.84000022				-29.84000022
b_{2E}	53.83999962				53.83999962
b_1	-1.008000E-06				-1.0080000E-06
b_{1E}	326.0799984				326.0799984
b_0	121.0000055				121.0000055
b_{0E}	279.7200073				279.7200073
a_2	-14.92000011	[-14.92000011]	[-14.92000011]		-14.92000011
a_{2E}	26.91999981	[26.91999981]	[26.91999981]		26.91999981
a_1	25.40159945	[25.40159945]	[25.40159945]		25.40159945
a_{1E}	270.7532019	[270.7532019]	[270.7532019]		270.7532019
a_0	-1.5876E-14	[-1.5876E-14]	[-1.5876E-14]		-1.5876E-14
a_{0E}	1.02715E-05	[1.02715E-05]	[1.02715E-05]		1.02715E-05
Calculations from the Figure 10 dashed red box are irrelevant and omitted here.					
q		-16.26684499			
q_E		179.5058229			
$q_E\,\varepsilon$		3.985830E-14			
r		59.84533934			
r_E		1225.144945			
$r_E\,\varepsilon$		2.720368E-13			
$R = r^2+q^3$		-722.909216			
R_E		289135.6698			
$\lvert R\rvert < R_E\,\varepsilon$		FALSE			
$q = r = 0$		FALSE			
$R = 0$		FALSE			
$r = 0$		FALSE			
$R > 0$		FALSE			

Table VI Calculated Parameters for Example 4 Quartic Equation with Symmetry Near Miss (Page 2)

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm with no Post Processing	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm	Figure 10 Final Quartic-Equation Algorithm with Cubic-Equation Post Processing
		Viète $r^2 + q^3 \le 0$			
θ		0.422250244			
ϕ_1		0.140750081			
ϕ_2		-1.953645021			
ϕ_3		2.235145184			
t_1		7.98666681			
t_{2x}		-3.01333344			
t_{3x}		-4.97333337			
y_2	[0]	0	[0]		
z_1	[12.96000018]	12.96000018	[12.96000018]		
x_2	[1.95999993]	1.95999993	[1.95999993]		
x_3	[1.776357E-15]	1.776357E-15	[1.776357E-15]		
ζ			0.345		
z_{1M}			12.96000018		
z_{2M}			1.95999993		
z_{3M}			1.77636E-15		
$\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\, \mathrm{MAX}(z_{1M}, z_{3M})$			TRUE (Recalculate small-magnitude solution(s))		
$y_2 = 0$			TRUE		
$z_{1M} > z_{3M}$			TRUE		
x_A			12.96000018		
$z_{2M} \ge z_{3M}$			TRUE		
$z_{2M} > \zeta\, z_{1M}$			FALSE (Recalculate two smallest-magnitude solutions)		
C			1.225E-15	[1.225E-15]	
B			-1.95999993	[-1.95999993]	
x_{AE}			14.92000011		
C_E			7.92556E-07	[7.92556E-07]	
B_E			23.14787019	[23.14787019]	
$\lvert B \rvert > B_E\, \varepsilon$ AND $\lvert C \rvert > C_E\, \varepsilon$			TRUE (Solve quadratic equation)		
D				3.841599726	
D_E				90.73965108	
$\lvert D \rvert < D_E\, \varepsilon$				FALSE	
$D > 0$				TRUE	
$y_2 = Y$			[0]	0	
Q				1.95999993	
$B \ge 0$				FALSE	
$x_{q1}=X_1$			[1.95999993]	1.95999993	
$x_{q2}=X_2$			[6.250000E-16]	6.250000E-16	
$x_A > x_{q1}$			TRUE		
z_1	[12.96000018]		12.96000018		[12.96000018]
x_2	[1.95999993]		1.95999993		[1.95999993]
x_3	[1.776357E-15]		6.250000E-16		[6.250000E-16]
y_2	[0]		0		[0]

Table VI Calculated Parameters for Example 4 Quartic Equation with Symmetry Near Miss (Page 3)

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm with no Post Processing	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm	Figure 10 Final Quartic-Equation Algorithm with Cubic-Equation Post Processing
Σ	-1				-1
d	3.4816594E-15				1.2249999E-15
s_{z1}	3.600000000				3.600000000
D	1.960000118				1.960000000
s_D	1.400000042				1.400000000
$D < 0$	FALSE				FALSE
Y_1	0				0
T_{x1}	5.000000042				5.000000000
T_{x2}	2.199999958				2.200000000
D	1.959999882				1.960000000
s_D	1.399999958				1.400000000
$D < 0$	FALSE				FALSE
Y_3	0				0
T_{x3}	-2.200000042				-2.200000000
T_{x4}	-4.999999958				-5.000000000
X_1	7.000000017				7.000000000
X_2	4.199999983				4.200000000
X_3	-0.200000117				-0.200000100
X_4	-2.999999983				-3.000000000

Entries enclosed in square brackets are input values, either from the user or from another algorithm in the table.

With the coefficient inputs, A_3, A_2, A_1, A_0, the quartic-equation algorithm detects no special cases, and so proceeds to calculate in straight-forward manner the coefficients a_2, a_1, and a_0 of the resolvent cubic equation and also the corresponding error size parameters a_{2E}, a_{1E}, and a_{0E}.

The cubic-equation algorithm takes over the parameter calculation with parameter q about 2/3 of the way down the first page of the table. The algorithm detects no special case, so with $R = r^2 + q^3 = -722.9$ (not positive), the algorithm proceeds with Viète to set $y_2 = 0$ and calculate the resolvent cubic equation's three real solutions, whose values are listed about 1/4 of the way down the table's second page.

$$z_1 = 12.96000018, \quad z_2 = x_2 = 1.95999993, \quad \text{and} \quad z_3 = x_3 = 1.776357\times10^{-15}$$

The values $y_2 = 0$, z_1, x_2, and x_3 are boxed in red.

The second column of the table (quartic-equation algorithm with no post processing) uses these solutions of the resolvent cubic equation.

The ultra small magnitude of the third solution $x_3 = 1.77636\times10^{-15}$ makes its accuracy suspect. In fact, most of this x_3 value is round-off error.

Cubic-equation post processing remedies the problem. The cubic-equation algorithm passes on to the post-processing algorithm the values of a_2, a_1, a_0, a_{2E}, a_{1E}, and a_{0E} on the first page of the table, as well as its solution values y_2, z_1, x_2, and x_3 boxed in red.

Post processing's first step is calculation of the absolute values of the three solutions: $z_{nM} = |z_n|$. In this particular case, the absolute values are in value order (greatest to least) and equal to the solutions themselves.

Next, the algorithm finds that

$$\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) = z_{3M} = 1.77636\times10^{-15} < \zeta\ \mathrm{MAX}(z_{1M}, z_{3M}) = 0.345 \times 12.96000018,$$

so, some form of solution recalculation is required. The conditions $y_2 = 0$ and $z_{1M} > z_{3M}$ imply that solution z_1 has the greatest absolute value of three real solutions. Therefore, set $x_A = z_1 = 12.96000018$. Also, $z_{2M} \geq z_{3M}$, so $z_2 = x_2$ has the second greatest magnitude, and $z_3 = x_3$ has the least. Because z_{2M} does not satisfy $z_{2M} = 1.96 > \zeta\ z_{1M} = 0.345\times12.96$, both z_2 and z_3 will be recalculated as solutions of a quadratic equation.

Post processing proceeds to calculate the quadratic-equation coefficients C and B and error size parameters x_{AE}, C_E and B_E. Coefficients C and B have absolute values greater than their error upper bounds $B_E\,\varepsilon$ and $C_E\,\varepsilon$, so the algorithm invokes the quadratic-equation algorithm, whose parameters are listed in the table's fifth column.

The quadratic-equation algorithm, finding there are no special cases and that discriminate $D = B^2 - 4C$ is positive, proceeds to calculate the two real quadratic-equation solutions using the Numerical Recipes process. The solution components, $x_{q1} = X_1$, $x_{q2} = X_2$, and $y_2 = Y$, are returned to so, cubic-equation post-processing algorithm.

The post-processing algorithm finds that $y_2 = 0$ and $x_A = 12.96 > x_{q1} = 1.96$, and so reports back to the quartic-equation algorithm the components of the revised resolvent-cubic-equation solutions as follows:

$$z_1 = 12.96000018,\quad x_2 = 1.95999993,\quad x_3 = 6.250000\times10^{-16},\quad y_2 = 0$$

These values, at the bottom of page 2 of the table, are boxed in green. Recalculation does not change the x_2 value, but the accurate recalculated $x_3 = 6.25\times10^{-16}$ is less than half of the original value of 1.78×10^{-15}. The original, incorrect value of the resolvent-cubic-equation solution x_3 is the source of error in the quartic-equation solutions when no post processing is used.

The recalculated resolvent-cubic-equation solutions boxed in green are reported back to the quartic-equation algorithm, as reflected in the last column of the table and going forward to page 3 of the table. Calculated solutions of the quartic equation in the table's last column are accurate with relative error less than 10^{-16}.

Quartic-equation entries in the second column on page 3 of the table (no post processing) reflect the original calculated values of the resolvent-cubic-equation solutions, boxed in red. Relative error of these calculated quartic-equation solutions is on the order of 10^{-7}.

There is one final note on our post processing. Recalculation did not change the value of resolvent-cubic-equation solution x_2 because the original value was already accurate. The algorithm recalculates x_2 because the stored parameter $\zeta = 0.345$ is large, and $z_{2M} = |x_2| = 1.96$ fails to exceed $\zeta\, z_{1M} = 0.345 \times 12.96 = 4.47$. The Section X error analysis shows that a ζ value of 0.345 minimizes error. The side effect of this large ζ is that solution values calculated originally are sometimes accurate but are recalculated anyway.

VI. QUARTIC-EQUATION POST PROCESSING FOR SMALL MAGNITUDE SOLUTIONS

This section expands the post-processing techniques from the last section to work with quartic equations. Given the coefficients A_3, A_2, A_1, and A_0 of the quartic equation $Z_n^4 + A_3Z_n^3 + A_2Z_n^2 + A_1Z_n + A_0 = 0$, the quartic-equation algorithm, Figure 10, calculates the components X_1, X_2, Y_1, X_3, X_4, Y_3 of the four solutions $Z_1=X_1+iY_1$, $Z_2=X_2-iY_1$, $Z_3=X_3+iY_3$, and $Z_4=X_4-iY_3$. If the accuracy of one, two, or three of the calculated solutions is suspect because the solution absolute value is sufficiently small, then the post processing accurately recalculates the suspect solutions. The post-processing design addresses the magnitude-condition quartic equations like Table I, Example 5. We work through Example 5 to demonstrate the operation of the post-processing algorithm, Figure 13.

Quartic-Equation Post Processing Algorithm

The post-processing inputs are the quartic-equation coefficients A_3, A_2, A_1, and A_0, the associated error size parameters A_{3E}, A_{2E}, A_{1E}, and A_{0E}, and the calculated solution components $X(1)=X_1$, $X(2)=X_2$, $Y(1)=Y_1$, $X(3)=X_3$, $X(4)=X_4$, and $Y(3)=Y_3$ from Figure 10. The algorithm also stores the constant $\zeta = 0.345$, the same used in the Figure 12 cubic equation post processing algorithm. Even though the imaginary components of Z_2 and Z_4 are $-iY_1$ and $-iY_3$, the post processing algorithm sets $Y(2)=Y(1)$ and $Y(4)=Y(3)$. The algorithm requires the nonnegative Y values.

To determine which solutions need recalculation, we need to place the four solutions in order of their magnitudes (absolute values), or equivalently in order of the square of absolute values. The algorithm starts by calculating $Z_{SQ}(k) = |Z_k|^2$, the square of absolute value of each solution:

$$Z_{SQ}(k) = X^2(k) + Y^2(k), \quad k = 1 \text{ to } 4 \tag{62}$$

The next task is to relabel the $Z_{SQ}(k)$ as $Z_{MSQ}(k)$ so that the $Z_{MSQ}(k)$ are in value order:

$$Z_{MSQ}(1) \geq Z_{MSQ}(2) \geq Z_{MSQ}(3) \geq Z_{MSQ}(4).$$

Additionally, we need an index function $I_{IN}(k)$ that associates each ordered $Z_{MSQ}(k)$ with the appropriate Z_{SQ}. That is, $Z_{SQ}[I_{IN}(k)] = Z_{MSQ}(k)$.

The following table provides an example to help clarify.

Table VII. Example of Solution Ordering

Input Index I_{IN}	$X(I_{IN})$	$Y(I_{IN})$	$Z_{SQ}(I_{IN})$		Output Index I_{OUT}	$Z_{MSQ}(I_{OUT})$	Index Function $I_{IN}(I_{OUT})$
1	3×10^{-8}	4×10^{-8}	2.5×10^{-15}		1	49	4
2	3×10^{-8}	-4×10^{-8}	2.5×10^{-15}		2	2.5×10^{-15}	1
3	2×10^{-12}	0	4×10^{-24}		3	2.5×10^{-15}	2
4	7	0	49		4	4×10^{-24}	3

The first column contains the input index I_{IN} values in order 1 to 4. The next two columns list the solution components, $X(I_{IN})$ and $Y(I_{IN})$, of an example quartic equation. Equation (62) above gives the $Z_{SQ}(I_{IN})$ values in the fourth column. Think of $Z_{SQ}(I_{IN})$ as the input function.

Figure 13 Quartic-Equation Post-Processing Algorithm

Post -Processing Constant: $\zeta = 0.345$

Inputs: For the quartic equation $Z_n^4 + A_3 Z_n^3 + A_2 Z_n^2 + A_1 Z_n + A_0 = 0$,

1) real coefficients A_3, A_2, A_1, A_0 and error size parameters A_{3E}, A_{2E}, A_{1E}, A_{0E}
2) real values X(1), X(2), Y(1), X(3), X(4), and Y(3) as calculated by the Figure 10 algorithm so that $Z_1 = X(1) + iY(1)$, $Z_2 = X(2) - iY(1)$, $Z_3 = X(3) + iY(3)$, and $Z_4 = X(4) - iY(3)$ are the four solutions.

Outputs: Recalculated values X(1), X(2), Y(1), X(3), X(4), and Y(3) as required. Components of any solution Z_n are recalculted if $|Z_n| < \zeta \text{ Max}(|Z_1|, |Z_2|, |Z_3|, |Z_4|)$.

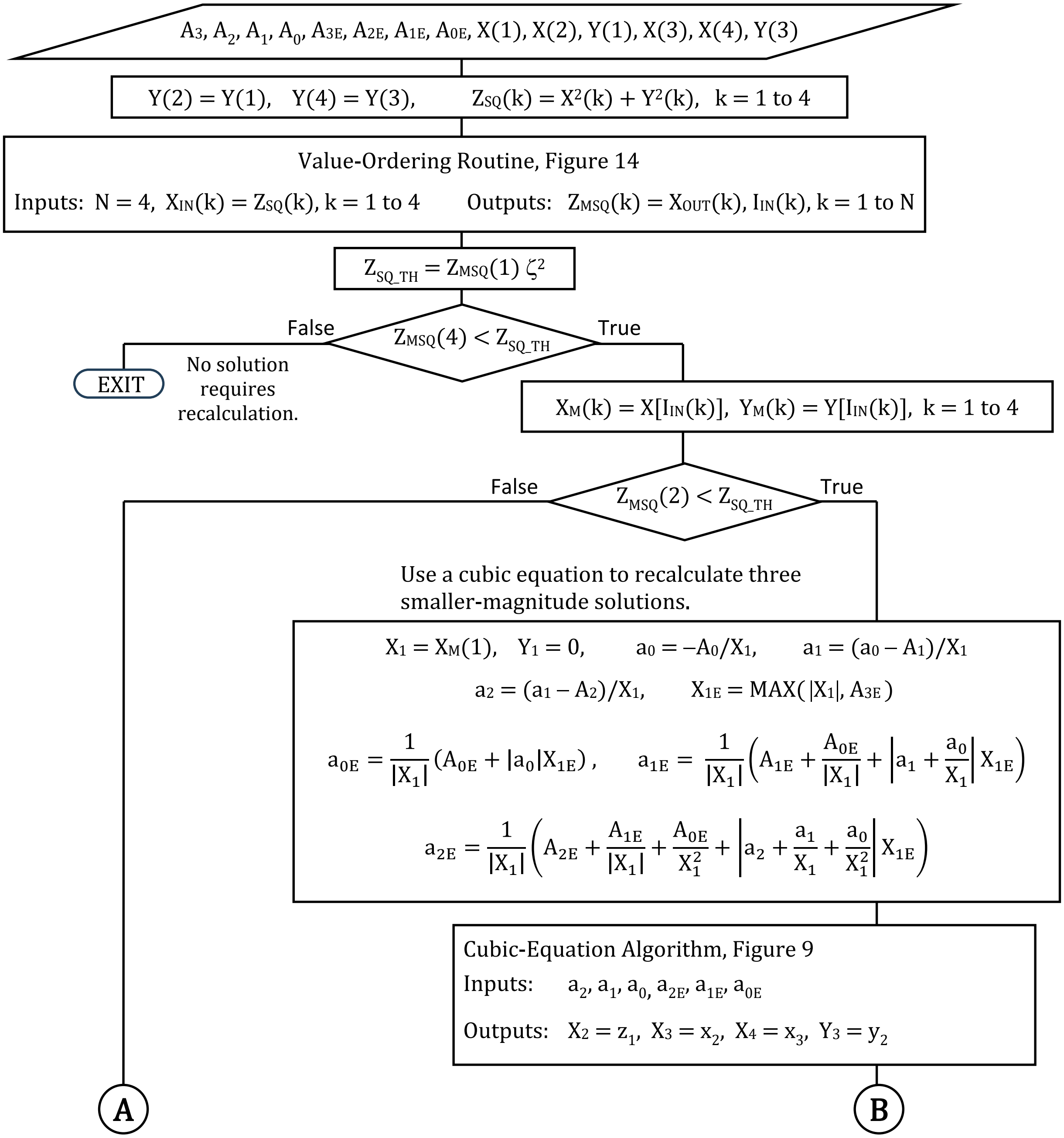
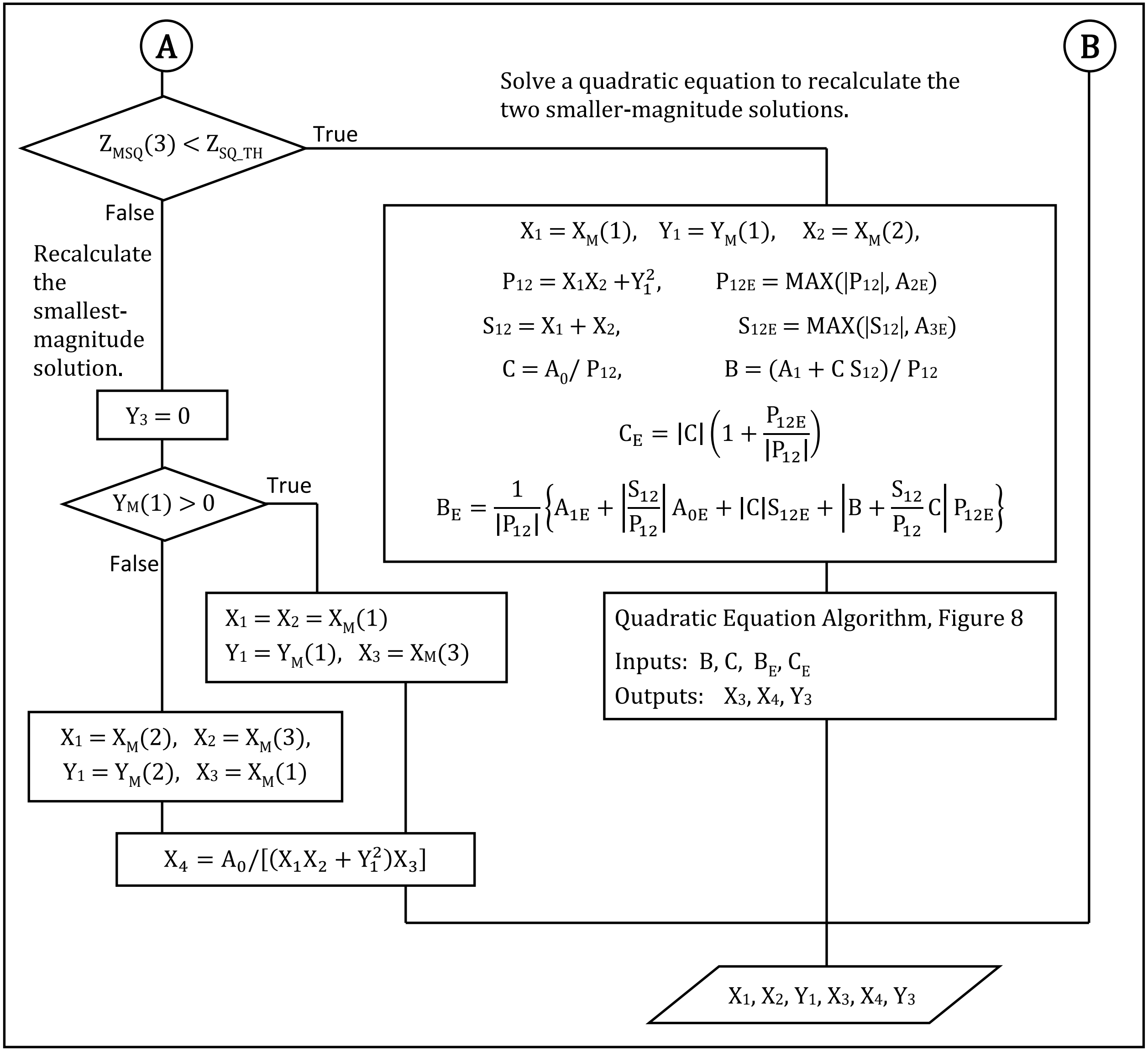

$A_3, A_2, A_1, A_0, A_{3E}, A_{2E}, A_{1E}, A_{0E}$, X(1), X(2), Y(1), X(3), X(4), Y(3)
$Y(2) = Y(1)$, $Y(4) = Y(3)$, $Z_{SQ}(k) = X^2(k) + Y^2(k)$, k = 1 to 4
Value-Ordering Routine, Figure 14
Inputs: N = 4, $X_{IN}(k) = Z_{SQ}(k)$, k = 1 to 4
Outputs: $Z_{MSQ}(k) = X_{OUT}(k)$, $I_{IN}(k)$, k = 1 to N
$Z_{SQ\_TH} = Z_{MSQ}(1)\,\zeta^2$
False
$Z_{MSQ}(4) < Z_{SQ\_TH}$
True
EXIT
No solution requires recalculation.
$X_M(k) = X[I_{IN}(k)]$, $Y_M(k) = Y[I_{IN}(k)]$, k = 1 to 4
False
$Z_{MSQ}(2) < Z_{SQ\_TH}$
True
Use a cubic equation to recalculate three smaller-magnitude solutions.
$X_1 = X_M(1)$, $Y_1 = 0$, $a_0 = -A_0/X_1$, $a_1 = (a_0 - A_1)/X_1$
$a_2 = (a_1 - A_2)/X_1$, $X_{1E} = \text{MAX}(|X_1|, A_{3E})$
$a_{0E} = \frac{1}{|X_1|}(A_{0E} + |a_0| X_{1E})$, $a_{1E} = \frac{1}{|X_1|}\left(A_{1E} + \frac{A_{0E}}{|X_1|} + \left|a_1 + \frac{a_0}{X_1}\right| X_{1E}\right)$
$a_{2E} = \frac{1}{|X_1|}\left(A_{2E} + \frac{A_{1E}}{|X_1|} + \frac{A_{0E}}{X_1^2} + \left|a_2 + \frac{a_1}{X_1} + \frac{a_0}{X_1^2}\right| X_{1E}\right)$
Cubic-Equation Algorithm, Figure 9
Inputs: $a_2, a_1, a_0, a_{2E}, a_{1E}, a_{0E}$
Outputs: $X_2 = z_1$, $X_3 = x_2$, $X_4 = x_3$, $Y_3 = y_2$
A
B

Figure 13 Quartic Equation Post Processing Algorithm (Page 2)


A
B
$Z_{MSQ}(3) < Z_{SQ\_TH}$
True
False
Solve a quadratic equation to recalculate the two smaller-magnitude solutions.
Recalculate the smallest-magnitude solution.
$Y_3 = 0$
$Y_M(1) > 0$
True
False
$X_1 = X_M(1), \quad Y_1 = Y_M(1), \quad X_2 = X_M(2),$
$P_{12} = X_1X_2 + Y_1^2, \quad P_{12E} = MAX(|P_{12}|, A_{2E})$
$S_{12} = X_1 + X_2, \quad S_{12E} = MAX(|S_{12}|, A_{3E})$
$C = A_0/ P_{12}, \quad B = (A_1 + C\,S_{12})/ P_{12}$
$C_E = |C|\left(1 + \frac{P_{12E}}{|P_{12}|}\right)$
$B_E = \frac{1}{|P_{12}|}\left\{A_{1E} + \left|\frac{S_{12}}{P_{12}}\right| A_{0E} + |C|S_{12E} + \left|B + \frac{S_{12}}{P_{12}}C\right| P_{12E}\right\}$
$X_1 = X_2 = X_M(1)$
$Y_1 = Y_M(1), \quad X_3 = X_M(3)$
Quadratic Equation Algorithm, Figure 8
Inputs: B, C, B_E, C_E
Outputs: X_3, X_4, Y_3
$X_1 = X_M(2), \quad X_2 = X_M(3),$
$Y_1 = Y_M(2), \quad X_3 = X_M(1)$
$X_4 = A_0/[(X_1X_2 + Y_1^2)X_3]$
$X_1, X_2, Y_1, X_3, X_4, Y_3$

The first output function is $Z_{MSQ}(I_{OUT})$ where the Z_{MSQ} are the same values as the Z_{SQ} but in proper order. The index function $I_{IN}(I_{OUT})$ in the last column is also an output function. It is defined so that $Z_{SQ}[I_{IN}(I_{OUT})] = Z_{MSQ}(I_{OUT})$ for all I_{OUT}.

We obtain the $Z_{MSQ}(k)$ and $I_{IN}(k)$ functions from the generic ordering routine in Figure 14 below. The inputs are $N = 4$ and $X_{IN}(k) = Z_{SQ}(k)$; the outputs are $Z_{MSQ}(k) = X_{OUT}(k)$ and $I_{IN}(k)$.

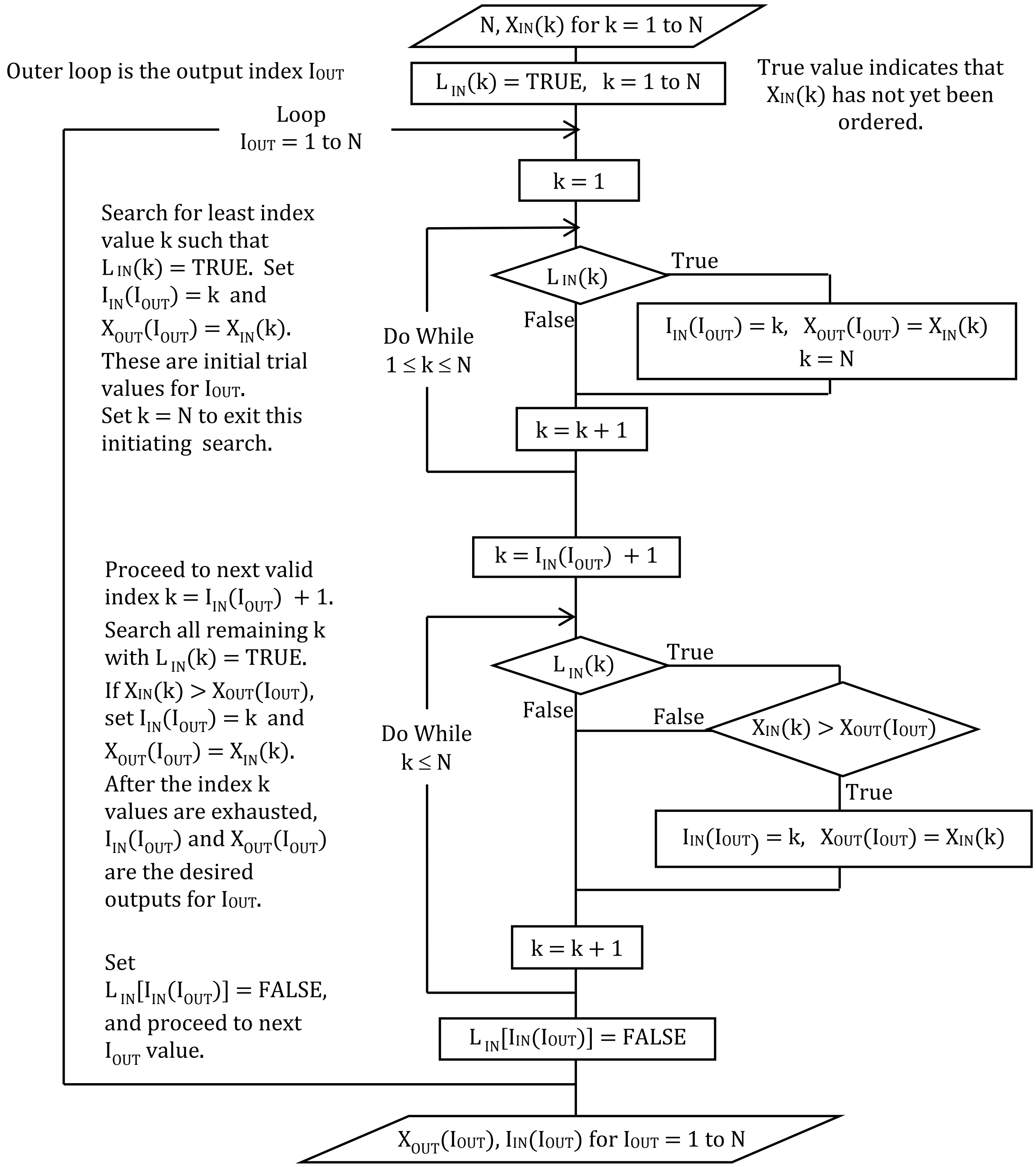
The greatest Z_{MSQ} is $Z_{MSQ}(1)$. This value is the standard against which the other $Z_{MSQ}(k)$ are compared to determine whether the corresponding solution $Z_M[I_{IN}(k)]$ requires recalculation. Figure 13 post processing calculates the threshold value

$$Z_{SQ\_TH} = Z_{MSQ}(1)\,\zeta^2. \tag{63}$$

Figure 14 Value-Ordering Routine

Inputs: N = number of elements in the input and output arrays
$X_{IN}(k)$ = array of values to be ordered for $k = 1$ to N

Outputs: $X_{OUT}(I_{OUT})$ = array of X_{IN} values ordered from greatest to least for $I_{OUT} = 1$ to N
$I_{IN}(I_{OUT})$ = array of order index values such that $X_{IN}[I_{IN}(I_{OUT})] = X_{OUT}(I_{OUT})$


N, $X_{IN}(k)$ for k = 1 to N
$L_{IN}(k)$ = TRUE, k = 1 to N
Loop
I_{OUT} = 1 to N
k = 1
$L_{IN}(k)$
True
False
$I_{IN}(I_{OUT}) = k$, $X_{OUT}(I_{OUT}) = X_{IN}(k)$
k = N
Do While
$1 \le k \le N$
k = k + 1
$k = I_{IN}(I_{OUT}) + 1$
$L_{IN}(k)$
True
False
False
$X_{IN}(k) > X_{OUT}(I_{OUT})$
True
Do While
$k \le N$
$I_{IN}(I_{OUT}) = k$, $X_{OUT}(I_{OUT}) = X_{IN}(k)$
k = k + 1
$L_{IN}[I_{IN}(I_{OUT})]$ = FALSE
$X_{OUT}(I_{OUT})$, $I_{IN}(I_{OUT})$ for I_{OUT} = 1 to N

Outer loop is the output index I_{OUT}

True value indicates that $X_{IN}(k)$ has not yet been ordered.

Search for least index value k such that $L_{IN}(k)$ = TRUE. Set $I_{IN}(I_{OUT}) = k$ and $X_{OUT}(I_{OUT}) = X_{IN}(k)$. These are initial trial values for I_{OUT}. Set k = N to exit this initiating search.

Proceed to next valid index $k = I_{IN}(I_{OUT}) + 1$. Search all remaining k with $L_{IN}(k)$ = TRUE. If $X_{IN}(k) > X_{OUT}(I_{OUT})$, set $I_{IN}(I_{OUT}) = k$ and $X_{OUT}(I_{OUT}) = X_{IN}(k)$. After the index k values are exhausted, $I_{IN}(I_{OUT})$ and $X_{OUT}(I_{OUT})$ are the desired outputs for I_{OUT}.

Set $L_{IN}[I_{IN}(I_{OUT})]$ = FALSE, and proceed to next I_{OUT} value.

The branch, $Z_{MSQ}(4) < Z_{SQ\_TH}$, checks whether the smallest Z_{MSQ}, $Z_{MSQ}(4)$, is less than the threshold. If not, then all Z_{MSQ} are sufficiently large that no recalculation is necessary, and the algorithm exits. Otherwise, some form of recalculation is required. In that case, the algorithm assigns the ordered solutions $Z_M(k)$ their real and imaginary components using the integer function $I_{IN}(k)$:

$$X_M(k) = X[I_{IN}(k)], \quad Y_M(k) = Y[I_{IN}(k)] \qquad k = 1 \text{ to } 4. \tag{64}$$

where $Z_M(k) = X_M(k) + iY_M(k)$ and $Z_{MSQ}(k) = |\,Z_M(k)\,|^2$.

The algorithm then works its way through a series of branches to determine which of the solutions $Z_M(k)$ require recalculation. The second greatest Z_{MSQ} is $Z_{MSQ}(2)$. If $Z_{MSQ}(2) < Z_{SQ\_TH}$, then $Z_M(2)$, $Z_M(3)$, and $Z_M(4)$ all require recalculation by solving a cubic equation. Otherwise, if $Z_{MSQ}(3) < Z_{SQ\_TH}$, then $Z_M(3)$ and $Z_M(4)$ require recalculation by solving a quadratic equation. Otherwise, only $Z_M(4)$ requires recalculation.

Use a Cubic Equation to Recalculate Three Small-Magnitude Solutions

Suppose $Z_{MSQ}(2) < Z_{SQ\_TH}$, so that $Z_M(2)$, $Z_M(3)$, and $Z_M(4)$ all require recalculation by solving a cubic equation. The greatest-magnitude solution $Z_M(1) = X_M(1) + iY_M(1)$ is accurate where components $X_M(1)$ and $Y_M(1)$ are known from Equation (64): $X_M(1) = X[I_{IN}(1)]$ and $Y_M(1) = Y[I_{IN}(1)]$. This solution must be real because $Z_{MSQ}(2) < Z_{SQ\_TH} < Z_{MSQ}(1)$. That is, $Z_M(1) = X_M(1)$. The solution $Z_M(1)$ components are relabeled

$$X_1 = X_M(1), \qquad Y_1 = 0, \tag{65}$$

and we recalculate the components of the three other quartic-equation solutions as the components of the three solutions of a cubic equation $z_n^3 + a_2\, z_n^2 + a_1\, z_n + a_0 = 0$.

$$X_2 = z_1 \qquad X_3 = x_2 \qquad X_4 = x_3 \qquad Y_3 = y_2. \tag{66}$$

The algorithm generates the cubic-equation coefficients a_2, a_1, and a_0 from the values of X_1, Y_1=0, and the quartic-equation coefficients A_2, A_1, and A_0. Derivation of the a_2, a_1, and a_0 formulas starts with Equations (3) and (66):

$$a_2 = -\,(z_1 + x_2 + x_3) \qquad = \; -(X_2 + X_3 + X_4) \tag{67}$$

$$a_1 = \quad z_1(x_2 + x_3) + x_2x_3 + y_2^2 \quad = \quad X_2(X_3 + X_4) + X_3X_4 + Y_3^2 \tag{68}$$

$$a_0 = -\,z_1(x_2x_3 + y_2^2) \qquad = \; -\,X_2(X_3X_4 + Y_3^2) \tag{69}$$

The quartic-equation coefficients are related to the solution components by the check equations, Equations (19) to (21) with Y_1=0.

$$A_2 \;=\; X_1X_2 + (X_1{+}X_2)(X_3{+}X_4) + X_3X_4 + Y_3^2 \;=\; X_1(X_2{+}X_3{+}X_4) + X_2(X_3 + X_4) + X_3X_4 + Y_3^2$$

$$A_1 \;=\; -[\,X_1X_2(X_3{+}X_4){+}(X_3X_4{+}Y_3^2)(X_1{+}X_2) \;=\; -\,X_1[X_2(X_3{+}X_4){+}X_3X_4{+}Y_3^2] \,-\, X_2(X_3X_4{+}Y_3^2)$$

$$A_0 \;=\; X_1X_2(X_3X_4{+}Y_3^2)$$

These expressions for A_2, A_1, and A_0 combine with Equations (67) to (69) to produce

$$A_0 = -X_1a_0, \qquad A_1 = -X_1a_1 + a_0, \qquad A_2 = -X_1a_2 + a_1.$$

Now these three equations combine to produce the post-processing expressions for the cubic-equation coefficients a_0, a_1, and a_2:

$$a_0 = -A_0/X_1 \qquad a_1 = (a_0 - A_1)/X_1 \qquad a_2 = (a_1 - A_2)/X_1 \tag{70}$$

In addition to the three coefficients a_0, a_1, and a_2, the Figure 9 cubic-equation algorithm also requires the associated error size parameters a_{0E}, a_{1E}, and a_{2E}. Obtain them by first expanding the equations in (70).

$$a_0 = -\frac{A_0}{X_1} \qquad a_1 = -\left(\frac{A_1}{X_1}+\frac{A_0}{X_1^2}\right) \qquad a_2 = -\left(\frac{A_2}{X_1}+\frac{A_1}{X_1^2}+\frac{A_0}{X_1^3}\right)$$

Calculate the error size parameters in the usual way by taking partial derivatives.

$$a_{0E} = \left|\frac{\partial a_0}{\partial A_0}\right| A_{0E} + \left|\frac{\partial a_0}{\partial X_1}\right| X_{1E} \ = \frac{A_{0E}}{|X_1|} + \frac{|A_0|}{X_1^2} X_{1E}$$

$$a_{0E} = \frac{1}{|X_1|}(A_{0E} + |a_0|X_{1E}) \tag{71}$$

$$a_{1E} = \left|\frac{\partial a_1}{\partial A_1}\right| A_{1E} + \left|\frac{\partial a_1}{\partial A_0}\right| A_{0E} + \left|\frac{\partial a_1}{\partial X_1}\right| X_{1E} \ = \ \frac{A_{1E}}{|X_1|} + \frac{A_{0E}}{X_1^2} + \left|\frac{A_1}{X_1^2} + \frac{2A_0}{X_1^3}\right| X_{1E}$$

$$a_{1E} = \frac{1}{|X_1|}\left(A_{1E} + \frac{A_{0E}}{|X_1|} + \left|a_1 + \frac{a_0}{X_1}\right| X_{1E}\right) \tag{72}$$

$$a_{2E} = \left|\frac{\partial a_2}{\partial A_2}\right| A_{2E} + \left|\frac{\partial a_2}{\partial A_1}\right| A_{1E} + \left|\frac{\partial a_2}{\partial A_0}\right| A_{0E} + \left|\frac{\partial a_2}{\partial X_1}\right| X_{1E}$$

$$= \frac{A_{2E}}{|X_1|} + \frac{A_{1E}}{X_1^2} + \frac{A_{0E}}{|X_1^3|} + \left|\frac{A_2}{X_1^2} + \frac{2A_1}{X_1^3} + \frac{3A_0}{X_1^4}\right| X_{1E}$$

$$a_{2E} = \frac{1}{|X_1|}\left(A_{2E} + \frac{A_{1E}}{|X_1|} + \frac{A_{0E}}{X_1^2} + \left|a_2 + \frac{a_1}{X_1} + \frac{a_0}{X_1^2}\right| X_{1E}\right) \tag{73}$$

The constituent values of X_1, A_{2E}, A_{1E}, A_{0E}, a_2, a_1, and a_0 are known, but X_{1E} is not. We know that X_1 is the real, greatest-magnitude, accurately-calculated solution of the quartic equation. So, one option is to just set $X_{1E} = |X_1|$. Equation (14) for A_3 also suggests the option to set $X_{1E} = |A_3|$ because

$$A_3 \ = \ -(Z_1+Z_2+Z_3+Z_4) = -[Z_1+Z_M(2)+Z_M(3)+ Z_M(4)] \ \approx \ -Z_1 = -X_1$$

when $|Z_1| = |X_1| >> |Z_M(2)| \geq \ |Z_M(3)| \geq \ |Z_M(4)|$. This condition is the primary reason for performing the post processing. Because we want $X_{1E}\,\varepsilon$ to be an easily-calculated, reasonable, upper bound of the round-off error in X_1, we opt to calculate X_{1E} as

$$X_{1E} = \mathrm{MAX}(\,|X_1|,\, A_{3E}) \ = \mathrm{MAX}(\,|X_1|,\, |A_3|\,). \tag{74}$$

To recalculate three small-magnitude solutions using a cubic equation, the algorithm executes Equation (65) for $X_1 = X_M(1)$ and $Y_1 = 0$, Equation (70) for a_0, a_1, and a_2, Equation (74) for X_{1E}, and Equations (71) to (73) for a_{0E}, a_{1E}, and a_{2E}. It then invokes the Figure 9 cubic-equation algorithm, whose outputs z_1, x_2, x_3, and y_2 provide the components of the recalculated, small-magnitude, quartic-equation solutions in (66):

$$X_2 = z_1, \quad X_3 = x_2, \quad X_4 = x_3, \quad Y_3 = y_2$$

Use a Quadratic Equation to Recalculate Two Small-Magnitude Solutions

Suppose $Z_{MSQ}(2) \geq Z_{SQ\_TH}$, but $Z_{MSQ}(3) < Z_{SQ\_TH}$ so that $Z_M(3)$ and $Z_M(4)$ require recalculation by solving a quadratic equation. The two greatest-magnitude solutions $Z_M(1) = X_M(1) + iY_M(1)$ and $Z_M(2) = X_M(2) - iY_M(1)$ are accurate where components $X_M(1)$, $X_M(2)$, and $Y_M(1)$ are known from Equation (64): $X_M(1) = X[I_{IN}(1)]$, $X_M(2) = X[I_{IN}(2)]$, $Y_M(1) = Y[I_{IN}(1)]$. The components of these two larger-magnitude solutions are relabeled:

$$X_1 = X_M(1), \qquad X_2 = X_M(2), \qquad Y_1 = Y_M(1). \tag{75}$$

Label the product and sum of Z_1 and Z_2 as $P_{12} \equiv Z_1Z_2$ and $S_{12} \equiv Z_1 + Z_2$ and calculate them as

$$P_{12} = X_1X_2 + Y_2^2 \quad \text{and} \quad S_{12} = X_1 + X_2. \tag{76}$$

We find the two other quartic-equation solutions $Z_3 = X_3 + iY_3$ and $Z_4 = X_4 - iY_3$ as the two solutions of a quadratic equation $Z_n^2 + B\,Z_n + C = 0$ where coefficients B and C satisfy

$$B = -(Z_3 + Z_4) = -(X_3 + X_4) \quad \text{and} \quad C = Z_3Z_4 = X_3X_4 + Y_3^3. \tag{77}$$

The algorithm generates B and C from the values of X_1, X_2, Y_1, and the quartic-equation coefficients A_1, and A_0. Derivation of the B and C formulas starts with Equation (16) and (17).

$$A_1 = -(Z_1Z_2Z_3+Z_1Z_2Z_4+Z_1Z_3Z_4+Z_2Z_3Z_4) = -Z_1Z_2(Z_3 + Z_4) - Z_3Z_4(Z_1 + Z_2)$$

$$A_0 = Z_1Z_2Z_3Z_4$$

These expressions for A_1 and A_0 combine with (77) to produce

$$A_0 = Z_1Z_2C \qquad A_1 = Z_1Z_2B - C(Z_1 + Z_2).$$

Now these two equations combine with the definitions $P_{12} \equiv Z_1Z_2$ and $S_{12} \equiv Z_1 + Z_2$ to produce the post-processing expressions for B and C:

$$C = A_0/P_{12} \qquad B = (A_1 + C\,S_{12})/\,P_{12} \tag{78}$$

The algorithm calculates X_1, X_2, and Y_1 from (75), P_{12} and S_{12} from (76), and C and B from (78).

In addition to the coefficients C and B, the Figure 8 quadratic-equation algorithm also requires the associated error size parameters C_E and B_E. Obtain them from (78) in the usual way by taking partial derivatives.

$$C_E = \left|\frac{\partial C}{\partial A_0}\right| A_{0E} + \left|\frac{\partial C}{\partial P_{12}}\right| P_{12E} \quad = \quad \frac{A_{0E}}{|P_{12}|} + \frac{|A_0|}{P_{12}^2} P_{12E}$$

$$C_E = \ |C| \left(1 + \frac{P_{12E}}{|P_{12}|}\right) \qquad (79)$$

The last equation results from the fact that $A_{0E} = |A_0|$ and $A_0/P_{12} = C$.

For the B_E formula, first substitute A_0/P_{12} for C in the Equation (78) expression for B.

$$B = \frac{A_1}{P_{12}} + \frac{A_0 S_{12}}{P_{12}^2}$$

$$B_E = \left|\frac{\partial B}{\partial A_1}\right| A_{1E} + \left|\frac{\partial B}{\partial A_0}\right| A_{0E} + \left|\frac{\partial B}{\partial S_{12}}\right| S_{12E} + \left|\frac{\partial B}{\partial P_{12}}\right| P_{12E}$$

$$= \frac{A_{1E}}{|P_{12}|} + \frac{|S_{12}|}{P_{12}^2} A_{0E} + \frac{|A_0|}{P_{12}^2} S_{12E} + \left|\frac{A_1}{P_{12}^2} + \frac{2A_0 S_{12}}{P_{12}^3}\right| P_{12E}$$

$$B_E = \frac{1}{|P_{12}|}\left\{A_{1E} + \left|\frac{S_{12}}{P_{12}}\right| A_{0E} + |C| S_{12E} + \left|B + \frac{S_{12}}{P_{12}} C\right| P_{12E}\right\} \qquad (80)$$

The constituent values of X_1, X_2, Y_1, A_1, A_0, A_{1E}, and A_{0E} are known, but P_{12E} and S_{12E} are not. We know that $Z_1 = X_1+iY_1$ and $Z_2 = X_2 - iY_1$ are the two greatest-magnitude, accurately-calculated solutions of the quartic equation. So, one option is to just set $P_{12E} = |P_{12}|$ and $S_{12E} = |S_{12}|$ where $P_{12} = Z_1Z_2$ and $S_{12} = Z_1 + Z_2$. Equations (15) for A_2 and (14) for A_3 also suggest the option to set $P_{12E} = |A_2|$ and $S_{12E} = |A_3|$ because

$A_2 \ = \ Z_1Z_2+Z_1Z_3+Z_1Z_4+Z_2Z_3+Z_2Z_4+Z_3Z_4 \ \approx \ Z_1Z_2 = P_{12}$ and

$A_3 \ = \ -(Z_1+Z_2+Z_3+Z_4) \ \approx \ -(Z_1+Z_2) = -S_{12}$

when $|Z_1| \geq |Z_2| \ >> \ |Z_3| \geq |Z_4|$. This condition is the primary reason for performing the post processing. Because we want $P_{12E}\,\varepsilon$ and $S_{12E}\,\varepsilon$ to be easily-calculated, reasonable, upper bounds of the round-off error in P_{12} and S_{12}, we opt to calculate P_{12E} and S_{12E} as

$$P_{12E} = \mathrm{MAX}(|P_{12}|, |A_2|) \qquad \text{and} \qquad S_{12E} = \mathrm{MAX}(|S_{12}|, |A_3|) \qquad \Rightarrow$$

$$P_{12E} = \mathrm{MAX}(\,|P_{12}|, A_{2E}) \qquad \text{and} \qquad S_{12E} = \mathrm{MAX}(|S_{12}|, A_{3E}) \qquad (81)$$

To recalculate two small-magnitude solutions using a quadratic equation, the algorithm executes Equation (75) for X_1, X_2, and Y_1; (76) for P_{12} and S_{12}; (78) for C and B; (81) for P_{12E} and S_{12E}; and (79) and (80) for C_E and B_E. It then invokes the Figure 8 quadratic-equation algorithm, whose outputs X_3, X_4, Y_3 are the components of the recalculated, small-magnitude, quartic-equation solutions $Z_3 = X_3 + iY_3$ and $Z_4 = X_4 - iY_3$.

Recalculate a Simple Small-Magnitude Solution

If $Z_{MSQ}(3)$ is not less than Z_{SQ_TH}, then $Z_M(3)$, like $Z_M(2)$ and $Z_M(1)$, needs no recalculation; only $Z_M(4)$ has so small an absolute value that it requires recalculation. $Z_M(4)$ is the only solution whose absolute value is so small, so it must be real. The recalculated value will be

$Z_4 = X_4 - iY_3 = X_4$, so the algorithm sets $Y_3 = 0$. The value of $Z_3 = X_3 + iY_3 = X_3$ is likewise real and is selected from $X_M(3)$ and $X_M(1)$ as follows.

If $Y_M(1) > 0$, then $Z_M(1)$ and $Z_M(2)$ are a complex conjugate pair, and $Z_M(3)$ is real. The algorithm sets $X_1 = X_2 = X_M(1)$, $Y_1 = Y_M(1)$, and $X_3 = X_M(3)$.

If $Y_M(1)$ is not greater than 0, then it equals 0, and $Z_M(1) = X_M(1)$ is real. In that case, solutions $Z_M(2)$ and $Z_M(3)$ are either both real, or they form a complex conjugate pair. The algorithm accommodates either case by pairing $Z_M(2)$ and $Z_M(3)$ as output solutions Z_1 and Z_2 and pairing the real solutions $Z_M(1)$ and $Z_M(4)$ as output solutions Z_3 and Z_4. The output real and imaginary components are calculated as $X_1 = X_M(2)$, $X_2 = X_M(3)$, $Y_1 = Y_M(2)$, and $X_3 = X_M(1)$.

The algorithm finally calculates X_4 as

$$X_4 = A_0/[(X_1X_2 + Y_1^2)X_3].$$

This expression is correct because Equation (17) gives A_0 as $A_0 = Z_1Z_2Z_3Z_4$; $Z_3 = X_3$ and $Z_4 = X_4$ are real; and the product Z_1Z_2 is $X_1X_2 + Y_1^2$.

This concludes the description of the Figure 13 quartic-equation post-processing algorithm. We demonstrate its operation with the Table I, Example 5 quartic-equation with the magnitude condition.

Example Magnitude-Condition Quartic Equation (Table I, Example 5)

The Example 5 quartic equation is

$$Z_n^4 - 6.99970002\, Z_n^3 - 2.099860005965\times10^{-3}\, Z_n^2 + 4.20000104993\times10^{-11}\, Z_n - 2.1\times10^{-25} = 0$$

with true solutions: 7, -3×10^{-4}, 2×10^{-8}, and 5×10^{-15}. This is an extreme example of the magnitude condition: the absolute values of the quartic equation's four solutions differ from each the other by many orders of magnitude. Solving this magnitude-condition quartic equation demonstrates the operation and effectiveness of quartic-equation post processing.

Solutions calculated with the Figure 2 quartic-equation algorithm (no round-off error mitigation) are

7, $-3.00019431496\times10^{-4}$, and $1.97157508097\times10^{-8} \pm i\, 2.41435601527\times10^{-6}$.

Solutions calculated with the Figure 10 final quartic-equation algorithm, but without post processing are

7, $-3.00000001152\times10^{-4}$, $1.00010177917\times10^{-8}$ and $1.00001391612\times10^{-8}$

As expected, both algorithms calculate the large-magnitude solution, 7, accurately. The calculated value of the second-greatest-magnitude solution, -3×10^{-4}, is considerably more accurate when using the Figure 10 final algorithm. Without post processing, however, neither algorithm had success with the two smallest magnitude solutions.

Table VIII, three and one-half pages long, lists all of the parameter values calculated by the mitigation design for the Example 5 quartic equation. Each column corresponds to one of the mitigation-design algorithms: Figure 10 quartic-equation algorithm, Figure 13 Quartic-Equation Post-Processing Algorithm, Figure 9 Final Cubic-Equation Algorithm, Figure 12 Cubic-Equation Post-Processing Algorithm, and Figure 8 Final Quadratic-Equation Algorithm. Entries enclosed in square brackets are input values, either from the user or from another algorithm in the table.

Initial Solutions from the Quartic-Equation Algorithm

The Figure 10 quartic-equation algorithm, using the coefficient inputs, A_3, A_2, A_1, A_0, detects no special cases, and so proceeds to calculate in straight-forward manner the coefficients a_2, a_1, and a_0 of the resolvent cubic equation. Also, the corresponding error size parameters a_{2E}, a_{1E}, and a_{0E}. The algorithm invokes the Figure 9 cubic-equation algorithm to solve the resolvent cubic equation.

The cubic-equation algorithm takes over the parameter calculation with parameter q about 2/3 of the way down the first page of the table. The calculated absolute value of $R = r^2+q^3$ is so small that R is reset to zero producing Special Case 3. All three solutions of the resolvent cubic equation are real ($y_2 = 0$), and two of the three real components z_1, x_2, and x_3 have the same value. In this case r is negative, so z_1 and x_2 equal each other. On the table's second page, the real components are calculated as $z_1 = x_2 = 3.0627625056$, and only slightly smaller $x_3 = 3.0622374881$.

The resolvent-cubic-equation solution components y_2, z_1, x_2, and x_3 are reported to the Figure 12 Cubic-Equation Post-Processing Algorithm, which finds that the three solution magnitudes are so close in value that no recalculation is necessary.

These same values of y_2, z_1, x_2, and x_3 are therefore used by the quartic-equation algorithm to finish calculating the quartic-equation solution components: Y_1, Y_3, X_1, X_2, X_3, and X_4. Their values in the table are boxed in red. If there were no post processing, these values would represent the final calculated solutions of the quartic equation.

The calculated values listed in the table for T_{x2}, T_{x3}, X_2, and X_3 show an inconsistency: values of depressed solutions T_{x2} and T_{x3} are displayed as equal to each other, but the corresponding solutions $X_2 = T_{x2} - C$ and $X_3 = T_{x3} - C$ are not equal to each other. Because the calculated solutions of the resolvent cubic equation reveal the multiplicity 2 condition $z_1 = x_2$ with $r \leq 0$, the quartic-equation calculated solutions should show a corresponding multiplicity 2 condition: $T_{x2} = T_{x3} = -\sqrt{x_3}$ and $X_2 = X_3 = -\sqrt{x_3} - C$.

Table VIII. Calculated Parameters for Example 5 Magnitude-Condition Quartic Equation

Example 5 Quartic Equation: $Z_n^4 - 6.99970002\, Z_n^3 - 2.099860005965\times 10^{-3} Z_n^2 + 4.20000104993\times 10^{-11} Z_n - 2.1\times 10^{-25} = 0$ with solutions 7, 2×10^{-8}, and 5×10^{-15}, and -3×10^{-4}					
Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm	Figure 13 Quartic-Equation Post-Processing Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm
	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]	[$\zeta = 0.345$]	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]	[$\zeta = 0.345$]	[$\varepsilon = 2^{-52} =$ 2.2204460E-16]
A_3	[-6.99970002]	[-6.99970002]			
A_2	[-2.099860006]	[-2.099860006]			
A_1	[4.2000011E-11]	[4.2000011E-11]			
A_0	[-2.100000E-25]	[-2.100000E-25]			
$A_0 = 0$	FALSE				
A_{3E}	6.99970002	[6.99970002]			
A_{2E}	2.099860006	[2.099860006]			
A_{1E}	4.2000011E-11	[4.2000011E-11]			
A_{0E}	2.100000E-25	[2.100000E-25]			
C	-1.749925005				
C_E	1.749925005				
b_2	-18.375524999				
b_{2E}	36.748950137				
b_1	-42.876837299				
b_{1E}	128.623162701				
b_0	-28.138326214				
b_{0E}	112.546874587				
a_2	-9.18776249938		[-9.1877624994]	[-9.1877624994]	
a_{2E}	18.374475069		[18.374475069]	[18.374475069]	
a_1	28.1383264898		[28.1383264898]	[28.1383264898]	
a_{1E}	112.5468751378		[112.546875138]	[112.546875138]	
a_0	-28.7253621366		[-28.725362137]	[-28.725362137]	
a_{0E}	172.3423256248		[172.342325625]	[172.342325625]	
Calculations from the Figure 10 dashed red box are irrelevant and omitted here.					
q			-3.06270422E-08		
q_E			75.0312501531		
$q_E\,\varepsilon$			FALSE		
r			-5.36459765E-12		
r_E			344.6846529378		
$r_E\,\varepsilon$			FALSE		
$R = r^2+q^3$			5.02610450E-26		
R_E			3.69840010E-09		
$\lvert R\rvert < R_E\,\varepsilon$			TRUE		
R reset			0		
$q\geq 0$ Or $r=0$			FALSE		
$q = r = 0$			FALSE		
$R = 0$			TRUE (Special Case 3)		
y_2	[0]		0	[0]	

Table VIII. Calculated Parameters for Example 5 Magnitude-Condition Quartic Equation (Page 2)

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm	Figure 13 Quartic-Equation Post-Processing Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm
$s = \sqrt{-q}$			1.75005835E-04		
$r > 0$			FALSE		
t_1			1.75005835E-04		
t_{2x}			1.75005835E-04		
t_{3x}			-3.50011670E-04		
z_1	[3.0627625056]		3.0627625056	[3.0627625056]	
x_2	[3.0627625056]		3.0627625056	[3.0627625056]	
x_3	[3.0622374881]		3.0622374881	[3.0622374881]	
z_{1M}				3.0627625056	
z_{2M}				3.0627625056	
z_{3M}				3.0622374881	
$\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\,\mathrm{MAX}(z_{1M}, z_{3M})$				FALSE (No Recalculation)	
Σ	-1				
d	9.37890616195				
s_{z1}	1.75007500000				
D	12.24999996500				
s_D	3.49999999500				
$D < 0$	FALSE				
$Y(1)=Y_1$	0	[0]			
T_{x1}	5.25007499500				
T_{x2}	-1.74992499500				
D	2.25015002E-08				
s_D	1.50005001E-04				
$D < 0$	FALSE				
$Y(3)=Y_3$	0	[0]			
T_{x3}	-1.74992499500				
T_{x4}	-1.75022500500				
$X(1)=X_1$	7	[7]			
$X(2)=X_2$	1.00010178E-08	[1.0001018E-08]			
$X(3)=X_3$	1.00001392E-08	[1.0000139E-08]			
$X(4)=X_4$	-3.00000001E-04	[-3.000000E-04]			
$Y(2)=Y(1)$		0			
$Y(4)=Y(3)$		0			
$Z_{SQ}(1)$		49			
$Z_{SQ}(2)$		1.00020357E-16			
$Z_{SQ}(3)$		1.00002783E-16			
$Z_{SQ}(4)$		9.00000007E-08			
$Z_{MSQ}(1)$		49			
$Z_{MSQ}(2)$		9.00000007E-08			
$Z_{MSQ}(3)$		1.00020357E-16			
$Z_{MSQ}(4)$		1.00002783E-16			
$I_{IN}(1)$		1			
$I_{IN}(2)$		4			
$I_{IN}(3)$		2			
$I_{IN}(4)$		3			

Table VIII. Calculated Parameters for Example 5 Magnitude-Condition Quartic Equation (Page 3)

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm	Figure 13 Quartic-Equation Post-Processing Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm
Z_{SQ_TH}		5.832225			
$Z_{MSQ}(4) < Z_{SQ\_TH}$		TRUE			
$X_M(1)$		7			
$X_M(2)$		-3.00000001E-04			
$X_M(3)$		1.00010178E-08			
$X_M(4)$		1.00001392E-08			
$Y_M(1)$		0			
$Y_M(2)$		0			
$Y_M(3)$		0			
$Y_M(4)$		0			
$Z_{MSQ}(2) < Z_{SQ\_TH}$		TRUE			
X_1		7			
Y_1		0			
a_0		3.00000000E-26	[3.0000000E-26]	[3.0000000E-26]	
a_1		-6.00000150E-12	[-6.000002E-12]	[-6.000002E-12]	
a_2		2.99980000E-04	[2.9998000E-04]	[2.9998000E-04]	
X_{1E}		7			
a_{0E}		6.00000000E-26	[6.0000000E-26]	[6.0000000E-26]	
a_{1E}		1.20000030E-11	[1.2000003E-11]	[1.2000003E-11]	
a_{2E}		5.99960001E-04	[5.9996000E-04]	[5.9996000E-04]	
q			-1.00006667E-08		
q_E			3.99986669E-08		
$\|q\| < q_E\,\varepsilon$			FALSE		
r			-1.00009999E-12		
r_E			6.00000001E-12		
$\|r\| < r_E\,\varepsilon$			FALSE		
$R = r^2+q^3$			-3.00039853E-32		
R_E			2.40024001E-23		
$\|R\| < R_E\,\varepsilon$			FALSE		
$q = r = 0$			FALSE		
$R = 0$			FALSE		
$r = 0$			FALSE		
$R > 0$			FALSE		
θ			3.14141945432		
ϕ_1			1.04713981811		
ϕ_2			-1.04725528429		
ϕ_3			3.14153492050		
t_1			1.00013333E-04		
t_{2x}			9.99933333E-05		
t_{3x}			-2.00006667E-04		
y_2			0	[0]	
z_1			2.00000001E-08	[2.0000000E-08]	
x_2			4.92168155E-15	[4.9216816E-15]	
x_3			-3.00000000E-04	[-3.000000E-04]	

Table VIII. Calculated Parameters for Example 5 Magnitude-Condition Quartic Equation (Page 4)

Parameter Symbol	Figure 10 Final Quartic-Equation Algorithm	Figure 13 Quartic-Equation Post-Processing Algorithm	Figure 9 Final Cubic-Equation Algorithm	Figure 12 Cubic-Equation Post-Processing Algorithm	Figure 8 Final Quadratic-Equation Algorithm
z_{1M}				2.00000001E-08	
z_{2M}				4.92168155E-15	
z_{3M}				3.00000000E-04	
$\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\, \mathrm{MAX}(z_{1M}, z_{3M})$				TRUE	
$y_2 = 0$				TRUE	
$z_{1M} > z_{3M}$				FALSE	
$x_A = x_3$				-3.00000000E-04	
$z_{1M} \geq z_{2M}$				TRUE	
$z_{1M} > \zeta\, z_{3M}$				FALSE	
C				1.00000000E-22	[1.0000000E-22]
B				-2.00000050E-08	[-2.000001E-08]
x_{AE}				3.00000000E-04	
C_E				3.00000000E-22	[3.0000000E-22]
B_E				6.00000150E-08	[6.0000015E-08]
$\|B\| > B_E\,\varepsilon$ AND $\|C\| > C_E\,\varepsilon$				TRUE	
$C = 0$					FALSE
D					3.99999800E-16
D_E					2.40000240E-15
$\|D\| < D_E\,\varepsilon$					FALSE
$D > 0$					TRUE
$y_2 = Y$				[0]	0
Q					2.00000000E-08
$B \geq 0$					FALSE
$x_{q1}=X_1$				[2.0000000E-08]	2.00000000E-08
$x_{q2}=X_2$				[5.0000000E-15]	5.00000000E-15
$y_2 = 0$				TRUE	
$x_A > x_{q1}$				FALSE	
z_1		[2.0000000E-08]		2.00000000E-08	
x_2		[5.0000000E-15]		5.00000000E-15	
x_3		[-3.000000E-04]		-3.00000000E-04	
y_2		[0]		0	
$X_2 = z_1$		2.00000000E-08			
$X_3 = x_2$		5.00000000E-15			
$X_4 = x_3$		-3.00000000E-04			
$Y_3 = y_2$		0			

The table to the right shows the pertinent calculated parameter values to 14 decimal places. The calculated values of T_{x2} and T_{x3} are not truly equal to each other; they differ starting in the 12th decimal place. They only appear to be equal in Table VIII because it shows

$T_{x2} = -\sqrt{x_3} =$	−1.74992499499898
$T_{x3} =$	−1.74992499499986
$C =$	−1.74992500500000
$X_2 =$	0.00000001000102
$X_3 =$	0.00000001000014

only the first eleven decimal places. The value of T_{x2} is calculated correctly as $-\sqrt{x_3}$, but the quartic-equation algorithm introduces some round-off error into the T_{x3} value.

The effect of the round-off error becomes obvious when C, which is close in value to T_{x2} and T_{x3}, is subtracted to produce X_2 and X_3.

This sort of round-off error discrepancy can be avoided if the algorithm reverts to the T_{xn} formulas in Figure 11 (Solutions of the Depressed Quartic Equation) whenever $y_2 = 0$ and $x_3 \geq 0$. That is, whenever the three resolvent-cubic-equation solutions are all nonnegative real. Such refinement is unnecessary, however, because the discrepancy problem occurs only under the magnitude condition, for which the relevant X_n values (X_2 and X_3) are recalculated in post processing.

Post-Processing Recalculation of Three Small-Magnitude Solutions

Using the initial solution component values boxed in red, Y(1), Y(3), X(1), X(2), X(3), and X(4), the quartic-equation post-processing algorithm sets Y(2) = Y(1), Y(4) = Y(3), and calculates the square $Z_{SQ}(k)$ of the solution absolute values, Equation (62).

The Value-Ordering Routine, Figure 14, returns these same square values in value order as $Z_{MSQ}(k)$ and the index function $I_{IN}(k)$ so that

$$Z_{MSQ}(1) \geq Z_{MSQ}(2) \geq Z_{MSQ}(3) \geq Z_{MSQ}(4) \quad \text{and} \quad Z_{SQ}[I_{IN}(k)] = Z_{MSQ}(k).$$

$Z_{MSQ}(1) = 49$ is the greatest $Z_{SQ}(k)$.

The algorithm calculates $Z_{SQ\_TH} = Z_{MSQ}(1)\,\zeta^2 = 12.25$, which value is listed at the top of Table VIII, Page 3. This Z_{SQ_TH} value becomes the threshold for $Z_{MSQ}(k)$ less than $Z_{MSQ}(1)$: if $Z_{MSQ}(k) < Z_{SQ\_TH}$, then solution $Z_M(k)$ will be recalculated. $Z_{MSQ}(4) = 1.00002783\times10^{-16}$ is the smallest $Z_{MSQ}(k)$, and $Z_{MSQ}(4) < Z_{SQ\_TH}$ is TRUE, so at least $Z_M(4)$ requires recalculation.

Post processing uses Equation (64) to calculate the real and imaginary components $X_M(k)$ and $Y_M(k)$ of the ordered solution $Z_M(k)$.

The second greatest $Z_{MSQ}(k)$ value is $Z_{MSQ}(2) = 9\times10^{-8}$, and $Z_{MSQ}(2) < Z_{SQ\_TH}$ is TRUE, so solutions $Z_M(2)$, $Z_M(3)$, and $Z_M(4)$ will all be recalculated as solutions of a cubic equation. Of the solutions calculated by the quartic-equation algorithm, only the greatest-magnitude real solution $Z_M(1)$ is reliable. Its components are relabeled $X_1 = X_M(1) = 7$ and $Y_1 = 0$. The values in the table are boxed in green to denote components of a final solution.

Post processing next calculates the cubic-equation coefficients a_2, a_1, and a_0 and their error size parameters a_{2E}, a_{1E}, and a_{0E}. These are the input parameters required for the cubic-equation algorithm to recalculate quartic-equation solutions $Z_M(2)$, $Z_M(3)$, and $Z_M(4)$.

The cubic-equation algorithm, Figure 9, takes over calculation about half way down the Page 3 of the table. No special cases apply, and $R > 0$ is FALSE, so Viète computation produces the three real cubic-equation solutions. The components y_2, z_1, x_2, and x_3 (bottom of page 3 of the table) are turned over to cubic-equation post processing, which takes over computation at the top of Page 4.

The cubic-equation post processing calculates the absolute values z_{1M}, z_{2M}, and z_{3M}. The least of these is $\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) = z_{2M} = 4.92\times10^{-15}$. The greatest is $\mathrm{MAX}(z_{1M}, z_{3M}) = z_{3M} = 3\times10^{-4}$. Because $\mathrm{MIN}(z_{1M}, z_{2M}, z_{3M}) < \zeta\,\mathrm{MAX}(z_{1M}, z_{3M})$ is TRUE, at least one of the three cubic-equation solutions requires recalculation. With imaginary component $y_2 = 0$, either z_1 or x_3 must be the real value of greatest absolute value. Computation finds $z_{1M} > z_{3M}$ is FALSE, so real solution x_3 has greatest magnitude, and x_A is set equal to $x_3 = -3\times10^{-4}$. This x_A value will become one of the quartic equation's computed solutions.

Evaluation shows $z_{1M} \geq z_{2M}$ is TRUE, so solution z_1 has the second greatest absolute value. Because $z_{1M} > \zeta\, z_{3M}$ is FALSE, the two small-magnitude solutions z_1 and x_2 will both be recalculated as solutions of a quadratic equation. The processing calculates the quadratic equation coefficients C and B and the corresponding error size parameters C_E and B_E.

These values are passed on to the quadratic-equation algorithm, which takes over computation half way down Page 4 of the table in the last column. No special case applies, and determinate $D > 0$, so calculation proceeds with Numerical Recipes to find the two real solutions $X_1 = 2\times10^{-8}$ and $X_2 = 5\times10^{-15}$ $(Y = 0)$.

These values are passed back to cubic-equation post processing with labels $x_{q1} = X_1$, $x_{q2} = X_2$, $y_2 = Y$, to complete its calculation. The cubic-equation's large-magnitude solution $x_A = -3\times10^{-4}$ is less than $x_{q1} = 2\times10^{-8}$, so the cubic-equation's solution components are assigned as follows:

$$z_1 = x_{q1} = 2\times10^{-8}, \qquad x_2 = x_{q2} = 5\times10^{-15}, \qquad x_3 = x_A = -3\times10^{-4} \quad \text{and} \quad y_2 = 0.$$

Finally, these components of the three cubic-equation solutions are passed back to the quartic-equation post-processing algorithm, where they are relabeled as components of the quartic equation's three small-magnitude solutions.

$$X_2 = z_1 = 2\times10^{-8}, \qquad X_3 = x_2 = 5\times10^{-15}, \qquad X_4 = x_3 = -3\times10^{-4} \quad \text{and} \quad Y_3 = y_2 = 0$$

These values at the end of the table are boxed in green to show that they are components of final solutions. They join the components $X_1 = 7$ and $Y_1 = 0$ calculated earlier.

All of these calculated values are accurate with solution relative error less than 1×10^{-16}.

This completes our description of quartic-equation post processing and the entire round-off-error mitigation design. The following section is an error analysis of that design for multiplicity and multiplicity near-miss conditions.

VII. ERROR ANALYSIS SUMMARY FOR MULTIPLICITY AND MULTIPLICITY NEAR-MISS

This section and the following three show that the mitigation design provides excellent solution accuracy for the multiplicity and multiplicity near-miss conditions. Previous sections have shown how the design addresses round-off error magnification for the multiplicity and magnitude conditions. We also showed how the design eliminates error magnification for quartic-equation symmetry and symmetry near miss. See the Section IV subsection on the Example 3 Quartic Equation Symmetry Condition and the Section V subsection on the Quartic-Equation Symmetry Near Miss (Table I, Example 4). Here for the first time, we address algorithm performance for the multiplicity near-miss condition.

This analysis, which examines quadratic and cubic equations, does not specifically address quartic equations. This is because the mitigation design automatically addresses solution-error magnification in quartic equations by providing accurate solutions to cubic and lower-order equations. Quartic-equation multiplicity, symmetry, and their near misses have a corresponding special-case condition in the Euler resolvent cubic equation as detailed in Figure 11. Accurate solutions to the resolvent cubic equation produce accurate quartic equation solutions. Post processing accurately calculates quartic-equation small-magnitude solutions as accurate solutions of cubic, quadratic, or linear equations.

The analysis in these last four sections is based on the fundamental concepts of quantum uncertainty, zero-guard range, and relative coefficient error.

Quantum Uncertainty

To establish the computer's solution accuracy limit, we define the concept of quantum uncertainty (QU). Let z_n be a true root of the polynomial function

$$p(z) = \sum_{k=0}^{N} a_k z^k$$

where coefficient $a_N = 1$. Then $p(z_n) = 0$. Now add a small positive value δz to z_n, and ask the question: "How great must δz be so that the computed value of $p(z_n+\delta z)$ changes from zero?". To be sure that $p(z_n+\delta z)$ is not computed as zero, δz must be at least as great as $|z_n|\varepsilon$, the magnitude represented by z_n's least significant bit. Otherwise, the value $z_n+\delta z$ could be stored in the computer as z_n. Usually, however, δz must be greater than $|z_n|\varepsilon$.

Consider the root $z_3 = 1$ from the Table I, Example 1 polynomial $p(z) = z^3 - 5z^2 + 8z - 4 = (z-1)(z-2)^2$. The magnitude of the root's least significant bit is $|z_3|\varepsilon = \varepsilon \approx 2.22\times10^{-16}$. The computer evaluates the polynomial $p(z)$ at $z = z_3 = 1$ as the sum of its four terms: $p(1) = 1 - 5 + 8 - 4 = 0$. The least significant bit of $p(1)$ is the least significant bit of the greatest-magnitude term 8 of the sum. The magnitude represented by that bit is therefore $\delta p = |8|\varepsilon = 8\varepsilon$. The true value of $p(z_3 + \delta z) = p(1 + \delta z)$ is

$$p(1+\delta z) = (1+\delta z - 1)(1+\delta z - 2)^2 = \delta z(\delta z - 1)^2 = \delta z^3 - 2\delta z^2 + \delta z \approx \delta z \text{ for } |\delta z| << 1.$$

To assure that the value of $p(1+\delta z)$ is not calculated as $p(1) = 0$, the value δz must satisfy

$$p(1+\delta z) \approx \delta z \;\; \geq \;\; \delta p = 8\varepsilon \approx 1.78\times10^{-15}.$$

The simple root $z_3 = 1$ of $p(z)$ could be calculated on the computer anywhere in the range $1 \leq z_n < 1+8\varepsilon$. We call the magnitude of this range the *quantum uncertainty (QU)* of root $z_3 = 1$. The quantum uncertainty value is designated $|\delta z|_{QU}$:

$$|\delta z|_{QU} = 8\varepsilon \approx 1.78\times10^{-15} \quad \text{for root } z_3 = 1.$$

We define the relative quantum uncertainty $|\delta z/z_3|_{QU}$ as the ratio of the quantum uncertainty $|\delta z|_{QU}$ to the absolute value of the true root $z_3 = 1$. For this case then

$$|\delta z/z_3|_{QU} \equiv |\delta z|_{QU}/|z_3| = 8\varepsilon/1 = 8\varepsilon \approx 1.78\times10^{-15}.$$

The double root $z_1 = z_2 = 2$ of this same polynomial has a much greater relative quantum uncertainty as we now show. The computer evaluates $p(2)$ as the sum of its terms: $p(2) = 2^3 - 5(2^2) + 8(2) - 4 = 8 - 20 + 16 - 4 = 0$. The least significant bit of $p(2)$ is the least significant bit of the greatest-magnitude term, 20. The magnitude represented by that bit is therefore $\delta p = |20|\varepsilon = 20\varepsilon$. The true value of $p(z_1 + \delta z) = p(2 + \delta z)$ is

$$p(2+\delta z) = (2+\delta z - 1)(2+\delta z - 2)^2 = (1+\delta z)\delta z^2 \approx \delta z^2 \text{ for } |\delta z| << 1.$$

To assure that the value of $p(2+\delta z)$ is not calculated as $p(2) = 0$, the value δz must satisfy

$$p(2+\delta z) \approx \delta z^2 \;\geq\; \delta p = 20\varepsilon \quad \Rightarrow \quad |\delta z| \geq \sqrt{20\varepsilon} \quad \Rightarrow |\delta z|_{QU} = \sqrt{20\varepsilon}.$$

The relative quantum uncertainty for the double root $z_1 = z_2 = 2$ is therefore

$$|\delta z/z_1|_{QU} \equiv |\delta z|_{QU}/|z_1| = \sqrt{20\varepsilon}/2 = \sqrt{5\varepsilon} \approx 3.33\times10^{-8} \text{ for double root } z_1 = z_2 = 2.$$

This relative quantum uncertainty for the double root $z_1 = z_2 = 2$ is over seven orders of magnitude greater than that for root $z_3 = 1$.

The relative quantum uncertainty $|\delta z/z_n|_{QU}$ is on the order of ε for a simple root, $\varepsilon^{1/2}$ for a double root, and $\varepsilon^{1/3}$ for a triple root.

The general procedure for calculating quantum uncertainty of root z_n of polynomial

$$p(z) = z^N + a_{N-1} z^{N-1} + \ldots + a_0$$

is as follows. Calculate the polynomial's bit size at $z = z_n$ as

$$\delta p = \text{MAX}(|z_n^N|, |a_{N-1}z_n^{N-1}|, \cdots, |a_0|)\varepsilon.$$

Solve the polynomial equation

$$p(z_n + \delta z) = \pm\delta p$$

for the uncertainty δz using either sign for δp on the right. The choice makes little difference. The quantum uncertainty for root z_n is the absolute value of δz, and is designated $|\delta z|_{QU}$. The relative quantum uncertainty is $|\delta z/z_n|_{QU}$.

We deviate slightly from this definition of relative quantum uncertainty when the root is a multiplicity near-miss root. Suppose roots z_n and z_{n+1} are a pair of such roots given by

$$z_n = x_0 + \Delta z \qquad \text{and} \qquad z_{n+1} = x_0 - \Delta z$$

where x_0 is nonzero real and $|\Delta z| < |x_0|$. The displacement Δz from x_0 is either real with $\Delta z = \Delta x > 0$, or Δz is pure imaginary with $\Delta z = i\Delta y$ and $\Delta y > 0$. The phrase "near-miss" implies that $|\Delta z| << |x_0|$. We simplify the calculation of the relative quantum uncertainty by using x_0 rather than $z_n = x_0 + \Delta z$ to normalize quantum uncertainty $|\delta z|_{QU}$. That is, we take $|\delta z/x_0|_{QU}$ as the relative quantum uncertainty rather than $|\delta z/z_n|_{QU}$.

Zero-Guard Processing and Zero-Guard Range

Our mitigation design avoids large error magnification for multiple solutions by anticipating and accommodating the multiplicity condition. For quadratic equations $Z_n^2 + BZ_n + C = 0$, the Figure 8 algorithm calculates the discriminate's error size parameter D_E. If the calculated discriminate magnitude $|D|$ is so small that $|D| < D_E\,\varepsilon$, then the calculated D value is reset to zero, and the algorithm calculates the two quadratic-equation solutions as the appropriate double solution $Z_1 = Z_2 = X_1 = X_2 = -B/2$ with imaginary component $Y = 0$.

The Figure 9 cubic-equation algorithm avoids multiplicity error magnification in similar fashion. If the magnitude $|R|$ is so small that $|R| < R_E\,\varepsilon$, then the calculated R value is reset to zero, and the algorithm properly calculates the three real solutions, two of which are the same real value (Special Case 3). If the cubic equation is the resolvent cubic equation of a quartic equation, then the quartic-equation solutions are properly calculated, with two of them set equal to the same real value.

This process whereby an algorithm performs the test $|D| < D_E\,\varepsilon$ or $|R| < R_E\,\varepsilon$ to determine whether D or R is reset to zero, we refer to as *zero-guard processing*. The range of D values $(-D_E\,\varepsilon,\ D_E\,\varepsilon)$ about $D = 0$ corresponds to a range of quadratic-equation solution values $(-B/2 - \Delta Z,\ -B/2 + \Delta Z)$ about the double solution value $Z_1 = Z_2 = X_0 = -B/2$. We call the positive, real value ΔZ the *zero-guard range* $|\Delta Z|_{ZG}$ about $-B/2$. Similarly, the range of R values $(-R_E\,\varepsilon,\ R_E\,\varepsilon)$ about $R = 0$ corresponds to a range of solution values $(z_0-\Delta z,\ z_0+\Delta z)$ about a double solution z_0 of a cubic equation. We call Δz the zero-guard range $|\Delta z|_{ZG}$ about z_0.

The relative zero-guard ranges for the quadratic equation and cubic equation are denoted

$$|\Delta Z/X_0|_{ZG} \equiv |\Delta Z|_{ZG}/|X_0| \qquad \text{and} \qquad |\Delta z/x_0|_{ZG} \equiv |\Delta z|_{ZG}/|x_0|.$$

The zero-guard range $|\Delta z|_{ZG}$ is a potential error caused by the zero-guard processing. For if two true near-miss solutions are $z_2 = z_0 + \Delta z$ and $z_3 = z_0 - \Delta z$ where $0 < |\Delta z| < |\Delta z|_{ZG}$, then the zero-guard processing will incorrectly calculate the two solutions as a double solution $z_2 = z_3$. We use the ratio of zero-guard range to the double-root quantum uncertainty as a measure of our ability to keep the size of $|\Delta z|_{ZG}$ under control.

$$ZG/QU = \frac{|\Delta z/x_0|_{ZG}}{|\delta z/x_0|_{QU}}.$$

Zero-guard processing also includes the Figure 9 cubic-equation processing for Special Case 2 ($q = r = 0$, multiplicity 3) and Special Case 4 ($r = 0$, three evenly spaced solutions).

Relative Coefficient Error

Relative coefficient error measures the accuracy with which the set of calculated solutions reproduces the input coefficients via the check equations. For a cubic equation with input coefficients a_2, a_1, and a_0, the three relative coefficient errors are

$$\delta a_{2u} \equiv \left|\frac{a_{2C} - a_2}{a_2}\right|, \quad \delta a_{1u} \equiv \left|\frac{a_{1C} - a_1}{a_1}\right|, \quad \delta a_{0u} \equiv \left|\frac{a_{0C} - a_0}{a_0}\right|. \tag{82}$$

where the calculated solution values z_{1C}, $z_{2C} = x_{2C} + iy_{2C}$, and $z_{3C} = x_{3C} - iy_{2C}$ are applied to the check equations (either Equations (2) or (3)) to calculate the *check coefficients* a_{2C}, a_{1C}, and a_{0C}. The relative coefficient errors provide a measure of an algorithm's accuracy without a priori knowledge of the true solutions.

Corresponding definitions apply to the quadratic equation $Z_n^2 + BZ_n + C = 0$ with coefficients B and C and calculated solutions Z_{1C} and Z_{2C}. The check coefficients are

$$B_C = -(Z_{1C} + Z_{2C}) \qquad C_C = Z_{1C}\,Z_{2C}$$

and the relative coefficient errors are

$$\delta B_u \equiv \left|\frac{B_C - B}{B}\right| \quad \text{and} \quad \delta C_u \equiv \left|\frac{C_C - C}{C}\right|.$$

Error Analysis Summary

Sections VIII through X examine the quantum uncertainty, zero-guard range, and relative coefficient error of multiple and multiple near-miss solutions of quadratic and cubic equations. We show that the relative coefficient error induced by zero-guard processing is a maximum of 3.3×10^{-15} for cubic equations and the ratio ZG/QU is less than 2.3. These values are even smaller for quadratic equations.

Section VIII addresses multiplicity 2 solutions (X_0) and multiplicity 2 near-miss solutions ($X_0 \pm \Delta Z$ where $|\Delta Z| << |X_0|$) in quadratic equations. We will also call these *double* solutions and *double near-miss* solutions. When $\Delta Z = 0$ so that two real solutions are exactly equal to each other, then the relative quantum uncertainty $|\delta Z/X_0|_{QU}$ is $\sqrt{2\varepsilon} \approx 2.11\times10^{-8}$. The relative zero-guard range is $|\Delta Z/X_0|_{ZG} = \sqrt{3\varepsilon} \approx 2.58\times10^{-8}$, 22% higher than the quantum uncertainty: ZG/QU = 1.22. When the zero-guard range is in effect ($\Delta Z < |\Delta Z|_{ZG}$), then the maximum relative coefficient errors are $\delta B_u = 0$ and $\delta C_u = 6.66\times10^{-16}$.

Sections IX and X demonstrate that our mitigation design provides good calculation accuracy for cubic equations.

Section IX addresses multiplicity 3, its near miss, and multiplicity 2. Two solutions have the same real value x_0, and a third real solution is x_A. The difference $x_A - x_0$ drops to zero to create the multiplicity 3 condition. For multiplicity 3 ($x_A = x_0$), we show that the relative quantum uncertainty $|\delta z/z_0|_{QU}$ is $(3\varepsilon)^{1/3} \approx 8.73\times10^{-6}$. The maximum relative solution error imposed by zero-guard processing is 15% greater at 1×10^{-5}, but the corresponding relative coefficient errors (Equation (82)) are small at $\delta a_{2u} = \delta a_{1u} = 0$, and $\delta a_{0u} = 2\times10^{-15}$. As the relative separation $|(x_A - x_0)/x_0|$ becomes large, then relative quantum uncertainty $|\delta z/z_0|_{QU}$ for the double solution x_0 approaches $(2\varepsilon)^{1/2} \approx 2.11\times10^{-8}$; relative quantum uncertainty $|\delta z/z_A|_{QU}$ for the simple solution x_A approaches $\varepsilon \approx 2.22\times10^{-16}$.

Section X addresses multiplicity 2 near miss. The three cubic-equation solutions are now $z_1 = x_A$, $z_2 = x_0 + \Delta z$ and $z_3 = x_0 - \Delta z$ where Δz is either a positive real value Δx or a positive pure imaginary number $i\Delta y$. Relative quantum uncertainties for the near-miss solutions are calculated as a function of $\eta \equiv x_A/x_0$ and $\Delta z/x_0$. The relative zero-guard range $|\Delta z/z_0|_{ZG}$ is a function η, but can change dramatically if post processing recalculates the two near-miss solutions when $|x_0| < |x_A|$. The recalculation occurs approximately when $|x_0/x_A| = |1/\eta| < |\zeta|$ where ζ is the Figure 12 post-processing constant. With the proper choice of ζ value, post processing not only eliminates magnification of magnitude-type round-off error, but it also controls the size of zero-guard range.

Section X shows that the selected value $\zeta = 0.345$ minimizes relative coefficient error induced by zero-guard processing. With this ζ value, the relative coefficient error induced by zero-guard processing is a maximum of 3.3×10^{-15} for all three coefficients and the maximum ratio ZG/QU is 2.3.

VIII. QUADRATIC EQUATION ERROR ANALYSIS

For quadratic equations $Z_n^2 + BZ_n + C = 0$, we show that zero-guard processing in the Figure 8 algorithm produces excellent solution accuracy: ZG/QU = 1.22 and relative coefficient errors $\delta B_u = 0$ and $\delta C_u = 6.66\times10^{-16}$ when the zero-guard range is in effect ($\Delta Z < |\Delta Z|_{ZG}$). This accuracy, however, is no better than that provided by the preliminary Numerical Recipes algorithm of Figure 4 for stand-alone quadratic equations in which the user enters coefficients B and C. As we demonstrate, it is solution accuracy for cubic and quartic equations with multiplicity (or multiplicity near miss) that requires the quadratic-equation algorithm to have zero-guard processing.

The quadratic polynomial P(Z) with roots $Z_1 = X_1 + iY$ and $Z_2 = X_2 - iY$ is written

$$P(Z) = Z^2 + BZ + C \qquad \text{where} \qquad B = -(Z_1 + Z_2) \quad \text{and} \quad C = Z_1Z_2.$$

The solutions Z_1 and Z_2 are both real with $X_1 \geq X_2$, $Y = 0$, or else they form a complex conjugate pair with $X_1 = X_2$, $Y > 0$. The quadratic equation is

$$P(Z_1) = P(Z_2) = Z_n^2 + BZ_n + C = 0.$$

To analyze quantum uncertainty and zero-guard range for multiplicity and multiplicity near-miss conditions, define the solutions Z_1 and Z_2 as

$$Z_1 = X_0 + \Delta Z \qquad \text{and} \qquad Z_2 = X_0 - \Delta Z$$

where X_0 is nonzero real and $|\Delta Z| < |X_0|$. The displacement ΔZ from X_0 is either real with $\Delta Z = \Delta X \geq 0$, or ΔZ is pure imaginary with $\Delta Z = i\Delta Y$ and $\Delta Y > 0$. The coefficients become

$$B = -2X_0 \qquad C = X_0^2 - \Delta Z^2 \qquad |\Delta Z| < |X_0|. \tag{83}$$

The polynomial at $Z = Z_1$ becomes

$$P(Z_1) = (X_0 + \Delta Z)^2 - 2X_0(X_0 + \Delta Z) + X_0^2 - \Delta Z^2 = 0. \tag{84}$$

Equation (84) is symmetric with respect to both $\Delta Z = 0$ and $X_0 = 0$. That is,

$$P(Z_1) = P(X_0 + \Delta Z) = P(X_0 - \Delta Z) = P(Z_2) = 0 \quad \text{and}$$

$$P(-X_0 + \Delta Z) = (-X_0 + \Delta Z)^2 - 2(-X_0)(-X_0 + \Delta Z) + (-X_0)^2 - \Delta Z^2 = 0 = P(X_0 + \Delta Z)$$

We therefore impose the convention

$$X_0 > 0$$

in order to simplify the analysis without loss of generality.

Quadratic Equation Quantum Uncertainty

The quantum uncertainty of solution Z_1 is the minimum absolute value $|\delta Z|$ which assures that the computed polynomial $P(Z_1 + \delta Z)$ in Equation (84) is nonzero. Of the three terms in $P(Z_1)$, the second term, $-2X_0(X_0 + \Delta Z)$, has the greatest magnitude under our restriction $|\Delta Z| < |X_0|$. The magnitude of the polynomial's least significant bit is therefore

$$\delta P = P_E\,\varepsilon = 2|X_0(X_0 + \Delta Z)|\varepsilon.$$

$P_E = 2|X_0(X_0+\Delta Z)|$ is the polynomial's error size parameter.

We find the quantum uncertainty of solution Z_1 by adding error δZ to Z_1, and solving the following equation for $|\delta Z|$:

$$P(Z_1 + \delta Z) = \pm\delta P \quad \text{where} \quad \delta P = P_E\,\varepsilon = 2|X_0(X_0+\Delta Z)|\,\varepsilon. \qquad (85)$$

The right side of this equation may be either $+\delta P$ or $-\delta P$. We start with $+\delta P$ and then show that the resulting quantum efficiency differs little from that using $-\delta P$.

The easiest way to proceed is to first normalize $Z_1 + \delta Z$ by X_0. The normalization allows us to calculate the relative quantum uncertainty $|\delta Z/X_0| = |\delta Z/X_0|_{QU}$ as a function of the multiplicity relative miss $\Delta Z/X_0$.

Use $+\delta P$ on the right side of Equation (85), and divide Equations (84) and (85) by X_0^2:

$$\frac{P(Z_1)}{X_0^2} = \left(1 + \frac{\Delta Z}{X_0}\right)^2 - 2\left(1 + \frac{\Delta Z}{X_0}\right) + 1 - \left(\frac{\Delta Z}{X_0}\right)^2 = 0$$

$$\frac{P(Z_1 + \delta Z)}{X_0^2} = \left(1 + \frac{\Delta Z}{X_0} + \frac{\delta Z}{X_0}\right)^2 - 2\left(1 + \frac{\Delta Z}{X_0} + \frac{\delta Z}{X_0}\right) + 1 - \left(\frac{\Delta Z}{X_0}\right)^2 = 2\left(1 + \frac{\Delta Z}{X_0}\right)\varepsilon\,.$$

Subtract the first equation from the second and simplify to obtain the general quadratic equation for relative quantum uncertainty $\delta Z/X_0$.

$$\left(\frac{\delta Z}{X_0}\right)^2 + 2\,\frac{\Delta Z}{X_0}\,\frac{\delta Z}{X_0} - 2\left(1 + \frac{\Delta Z}{X_0}\right)\varepsilon = 0 \quad \text{for} \quad P(Z_1 + \delta Z) = \delta P \qquad (86)$$

If the displacement ΔZ is real ($\Delta Z = \Delta X \geq 0$), then the pertinent solution is also real and given by

$$\left|\frac{\delta Z}{X_0}\right|_{QU} = \frac{\delta X}{X_0} = \frac{2(1 + \Delta X/X_0)\varepsilon}{|\Delta X/X_0| + \sqrt{(\Delta X/X_0)^2 + 2(1 + \Delta X/X_0)\varepsilon}} \qquad \begin{array}{c}\text{for } \Delta Z = \Delta X \text{ and}\\ P(Z_1 + \delta Z) = \delta P.\end{array} \qquad (87)$$

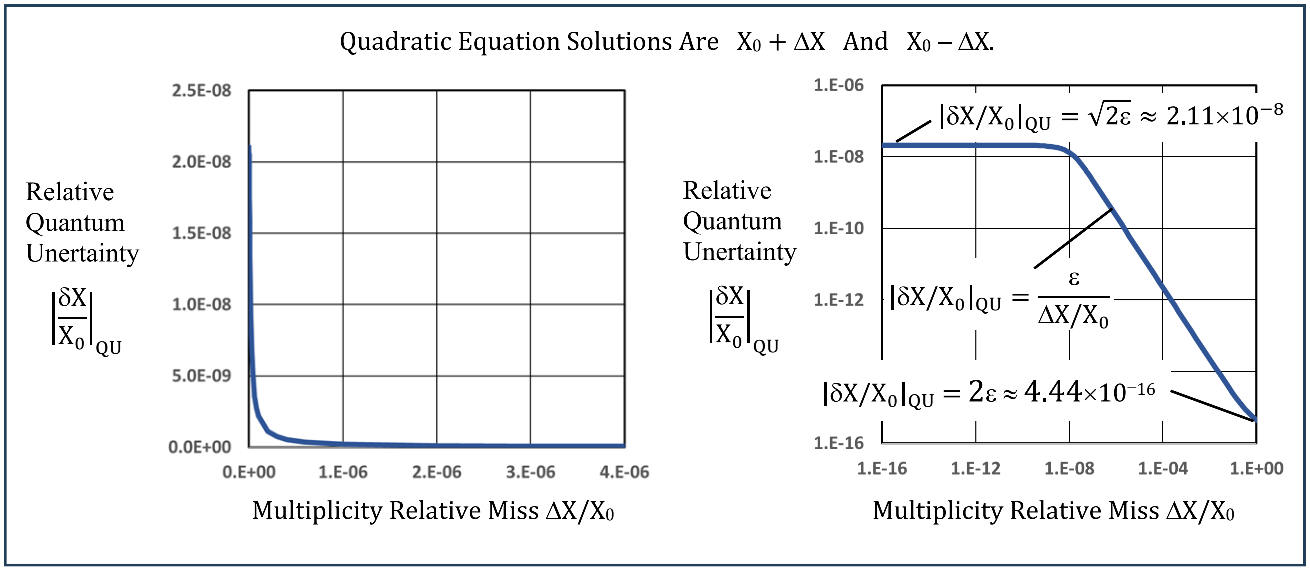
This solution provides the relative quantum uncertainty as a function of the real relative near-miss value $\Delta X/X_0$. Figure 15 below provides linear and log-log plots of $|\delta X/X_0|_{QU}$ vs $\Delta X/X_0$ that show how quantum uncertainty decreases as $\Delta X/X_0$ increases from zero. At the multiplicity condition $\Delta X = 0$, the right side of Equation (87) collapses to $\sqrt{2\varepsilon}$.

$$\left|\frac{\delta Z}{X_0}\right|_{QU} = \frac{\delta X}{X_0} = \sqrt{2\varepsilon} \approx 2.11\times10^{-8} \quad \text{for the multiplicity condition } \Delta Z = \Delta X = 0. \quad (88)$$

When relative miss $|\Delta Z/X_0|$ satisfies $\sqrt{2\varepsilon} << \Delta X/X_0 << 1$, then the numerator in (87) collapses to 2ε, the denominator collapses to $2|\Delta X/X_0|$, and $|\delta X/X_0|_{QU}$ becomes

$$|\delta Z/X_0|_{QU} = \frac{\varepsilon}{\Delta X/X_0} \qquad \text{for } \sqrt{2\varepsilon} \ll \Delta X/X_0 \ll 1. \qquad (89)$$

Figure 15 Quantum Uncertainty for Quadratic-Equation Multiplicity Real Near Miss



Finally, when $\Delta X/X_0 = 1$, then the radicand in (87) is $1 + 4\varepsilon \approx 1$, the denominator is 2, the numerator is 4ε, and $|\delta Z/X_0|_{QU}$ is

$$|\delta Z/X_0|_{QU} = \delta Z/X_0 = 2\varepsilon \qquad \text{for} \quad \Delta X/X_0 = 1. \tag{90}$$

The displacement ΔZ may also be imaginary: $\Delta Z = i\Delta Y$ where $\Delta Y \geq 0$. In that case, the error value δZ is complex: $\delta Z = \delta X + i\delta Y$. The left side of Equation (86) then consists of both a real part and an imaginary part.

$$\left(\frac{\delta X}{X_0} + i\frac{\delta Y}{X_0}\right)^2 + 2i\frac{\Delta Y}{X_0}\left(\frac{\delta X}{X_0} + i\frac{\delta Y}{X_0}\right) - 2\left(1 + i\frac{\Delta Y}{X_0}\right)\varepsilon = 0$$

$$\left(\frac{\delta X}{X_0}\right)^2 - \left(\frac{\delta Y}{X_0}\right)^2 - 2\frac{\Delta Y}{X_0}\frac{\delta Y}{X_0} - 2\varepsilon + i2\left[\frac{\delta X}{X_0}\frac{\delta Y}{X_0} + \frac{\Delta Y}{X_0}\frac{\delta X}{X_0} - \frac{\Delta Y}{X_0}\varepsilon\right] = 0$$

The real and imaginary components each equal zero.

$$\left(\frac{\delta X}{X_0}\right)^2 - \left(\frac{\delta Y}{X_0}\right)^2 - 2\frac{\Delta Y}{X_0}\frac{\delta Y}{X_0} - 2\varepsilon = 0 \qquad \text{(Real Part)} \tag{91}$$

$$\frac{\delta X}{X_0}\left(\frac{\delta Y}{X_0} + \frac{\Delta Y}{X_0}\right) - \frac{\Delta Y}{X_0}\varepsilon = 0 \qquad \text{(Imaginary Part)} \tag{92}$$

Add $(\Delta Y/X_0)^2$ to both sides of (91) and rearrange as

$$\left(\frac{\delta X}{X_0}\right)^2 + \left(\frac{\Delta Y}{X_0}\right)^2 - 2\varepsilon \; = \left(\frac{\delta Y}{X_0} + \frac{\Delta Y}{X_0}\right)^2. \tag{93}$$

Rearrange (92) as

$$\frac{\delta Y}{X_0}+\frac{\Delta Y}{X_0}=\left(\frac{\Delta Y}{X_0}\varepsilon\right)\Big/\left(\frac{\delta X}{X_0}\right). \tag{94}$$

The right side of (93) is the square of the left side of (94). Square Equation (94); then substitute the squared right side for the right side of (93). Simplify to obtain a quadratic equation in $(\delta X/X_0)^2$.

$$\left[\left(\frac{\delta X}{X_0}\right)^2\right]^2+B_{\delta X}\left(\frac{\delta X}{X_0}\right)^2+C_{\delta X}=0 \quad \text{where} \quad B_{\delta X}=\left(\frac{\Delta Y}{X_0}\right)^2-2\varepsilon, \quad C_{\delta X}=-\left(\frac{\Delta Y}{X_0}\right)^2\varepsilon^2 \tag{95}$$

Apply the Equation (95) solution value $(\delta X/X_0)^2$ to Equation (91) to produce a quadratic equation in $\delta Y/X_0$.

$$\left(\frac{\delta Y}{X_0}\right)^2+B_{\delta Y}\left(\frac{\delta Y}{X_0}\right)+C_{\delta Y}=0 \quad \text{where} \quad B_{\delta Y}=2\left(\frac{\Delta Y}{X_0}\right), \quad C_{\delta Y}=2\varepsilon-\left(\frac{\delta X}{X_0}\right)^2 \tag{96}$$

The Figure 8 quadratic-equation algorithm provides the two solutions for each of Equations (95) and (96). The two solutions of (95) have opposite signs as indicated by the equation's negative constant coefficient, $C_{\delta X} = -(\Delta Y/X_0)^2\,\varepsilon^2$. We use the positive solution for $(\delta X/X_0)^2$, and then take its positive square root for $\delta X/X_0$. The positive square-root value is required in order that $\delta X/X_0$ matches Equation (88), $\delta X/X_0 = \sqrt{2\varepsilon}$, at the multiplicity condition $\Delta Z = \Delta X = \Delta Y = 0$. Notice that at multiplicity, when $\Delta Y = 0$ and $\delta X/X_0 = \sqrt{2\varepsilon}$, Equation (96) collapses to $(\delta Y/X_0)^2 = 0$.

The two solutions of Equation (96) for $\delta Y/X_0$ are each negative or zero because both coefficients, $B_{\delta Y} = 2(\Delta Y/X_0)$ and $C_{\delta Y} = 2\varepsilon - (\delta X/X_0)^2$, are nonnegative. We select the greater solution for $|\delta Y/X_0|_{QU}$ (solution of lesser absolute value) because it satisfies (94) as required.

Figure 16 below plots the calculated relative quantum uncertainty for imaginary near miss $\Delta Z = i\Delta Y$. The dashed blue curve is the real component $\delta X/X_0$ of $\delta Z/X_0$, the dashed yellow curve is the negative imaginary component $-\delta Y/X_0$ of $\delta Z/X_0$, and the solid black curve is the total (the relative quantum uncertainty): $|\delta Z/X_0|_{QU} = [(\delta X/X_0)^2 + (\delta Y/X_0)^2]^{1/2}$.

The most obvious feature of the plot is the dramatic change in behavior of both the real and imaginary components at the *critical displacement* $\Delta Y/X_0 = \sqrt{2\varepsilon} \approx 2.11\times10^{-8}$. As $\Delta Y/X_0$ increases through this value, the linear coefficient $B_{\delta X} = (\Delta Y/X_0)^2 - 2\varepsilon$ of Equation (95) changes sign from negative to positive. The sign change in $B_{\delta X}$ then changes the nature of the equation's positive solution $(\delta X/X_0)^2$. The equation's constant coefficient is $C_{\delta X} = -(\Delta Y/X_0)^2\,\varepsilon^2$, so the determinate is

$$D_{\delta X} = B^2{}_{\delta X} - 4C_{\delta X} = (\Delta Y/X_0)^4 - 4(\Delta Y/X_0)^2(\varepsilon - \varepsilon^2) + 4\varepsilon^2. \tag{97}$$

When $\Delta Y/X_0 = \sqrt{2\varepsilon}$, and $B_{\delta X} = 0$, then $D_{\delta X} = -\,4C_{\delta X} = 4(\Delta Y/X_0)^2\,\varepsilon^2$. Unless $\Delta Y/X_0$ is close to the critical displacement $\sqrt{2\varepsilon}$, however, the difference $(\varepsilon - \varepsilon^2) = \varepsilon(1 - \varepsilon)$ in Equation (97) may be written as ε, and $D_{\delta X}$ becomes

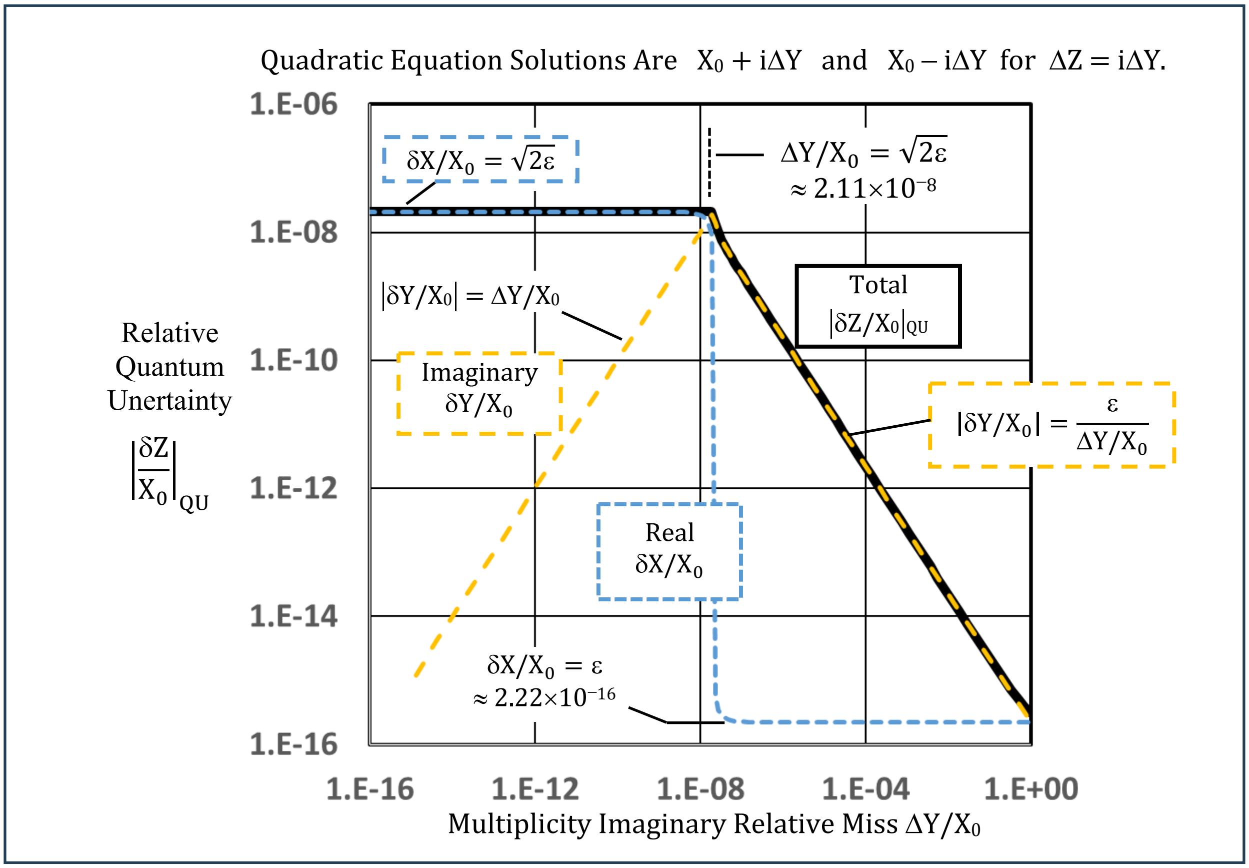
$$D_{\delta X} \approx (\Delta Y/X_0)^4 - 4(\Delta Y/X_0)^2\varepsilon + 4\varepsilon^2 = [(\Delta Y/X_0)^2 - 2\varepsilon]^2 = B_{\delta X}^2.$$

The formula for Q in the Figure 8 quadratic-equation algorithm is $Q = (|B| + \sqrt{|D|})/2$. So, Q for Equation (95) is

$$Q_{\delta X} = \frac{1}{2}\left(|B_{\delta X}| + \sqrt{|D_{\delta X}|}\right) \approx \frac{1}{2}(|B_{\delta X}| + |B_{\delta X}|) = |B_{\delta X}| = |(\Delta Y/X_0)^2 - 2\varepsilon|.$$

This approximation for $Q_{\delta X}$ is excellent unless $|(\Delta Y/X_0)^2 - 2\varepsilon|/(2\varepsilon) < 1\times10^{-6}$. That is, $Q_{\delta X} \cong |B_{\delta X}|$ unless $\Delta Y/X_0$ is extremely close to $\sqrt{2\varepsilon}$.

Figure 16 Quantum Uncertainty for Quadratic-Equation Multiplicity Imaginary Near Miss



When $(\Delta Y/X_0)^2 < 2\varepsilon$, then $B_{\delta X} = (\Delta Y/X_0)^2 - 2\varepsilon$ is negative. The Figure 8 algorithm gives the solution of Equation (95) for $(\delta X/X_0)^2$ as $Q_{\delta X}$, which implies that $\delta X/X_0$ is:

$$\delta X/X_0 = \sqrt{Q_{\delta X}} = \sqrt{|B_{\delta X}|} = \sqrt{|2\varepsilon - (\Delta Y/X_0)^2|} \quad \text{for } \Delta Y/X_0 < \sqrt{2\varepsilon} \approx 2.11\times10^{-8} \qquad (98)$$

When $(\Delta Y/X_0)^2 > 2\varepsilon$, then $B_{\delta X} = (\Delta Y/X_0)^2 - 2\varepsilon$ changes sign to positive, and the Figure 8 quadratic-equation algorithm gives the solution of Equation (95) as $-C_{\delta X} / Q_{\delta X}$. The formula for $\delta X/X_0$ becomes

$$\delta X/X_0 = \sqrt{\frac{-C_{\delta X}}{Q_{\delta X}}} = \sqrt{\frac{(\Delta Y/X_0)^2\varepsilon^2}{|(\Delta Y/X_0)^2 - 2\varepsilon|}} = \frac{(\Delta Y/X_0)\varepsilon}{\sqrt{|(\Delta Y/X_0)^2 - 2\varepsilon|}} \quad \text{for } \Delta Y/X_0 > \sqrt{2\varepsilon} \qquad (99)$$

Equation (99) simplifies even further when there is an order of magnitude separation between $(\Delta Y/X_0)^2$ and 2ε. If $(\Delta Y/X_0)^2 >> 2\varepsilon$, then $[(\Delta Y/X_0)^2 - 2\varepsilon] \rightarrow (\Delta Y/X_0)^2$.

$$\delta X/X_0 \;\approx\; \varepsilon \;\approx\; 2.22{\times}10^{-16} \;\text{ for } \Delta Y/X_0 > 3.2{\times}\sqrt{2\varepsilon} \qquad (100)$$

The values of $\delta X/X_0$ in Equations (98) and (100) are clearly evident as the two horizontal dashed blue lines for $[\delta X/X_0]_{QU}$ in Figure 16 to the left and to the right of $\Delta Y/X_0 = \sqrt{2\varepsilon} \approx 2.11{\times}10^{-8}$.

These results for $\delta X/X_0$ produce corresponding formulas for $\delta Y/X_0$ via Equation (96). For the condition $\Delta Y/X_0 < \sqrt{2\varepsilon}$, substitute Equation (98) into (96).

$$\left(\frac{\delta Y}{X_0}\right)^2 + 2\frac{\Delta Y}{X_0}\frac{\delta Y}{X_0} + \left(\frac{\Delta Y}{X_0}\right)^2 = \left(\frac{\delta Y}{X_0} + \frac{\Delta Y}{X_0}\right)^2 = 0 \;\;\Rightarrow$$

$$\left|\frac{\delta Y}{X_0}\right| = -\frac{\delta Y}{X_0} = \frac{\Delta Y}{X_0} \qquad \text{for } \;\Delta Y/X_0 < \sqrt{2\varepsilon}.$$

This result is shown as the diagonal, increasing yellow dashed line for $|\delta Y/X_0|_{QU}$ in Figure 16.

For the condition $\Delta Y/X_0 > 3.2{\times}\sqrt{2\varepsilon}$, substitute Equation (100) into (96).

$$\left(\frac{\delta Y}{X_0}\right)^2 + 2\frac{\Delta Y}{X_0}\frac{\delta Y}{X_0} + 2\varepsilon - \varepsilon^2 = 0 \qquad \text{for } \;\frac{\Delta Y}{X_0} > 3.2{\times}\sqrt{2\varepsilon}\,. \qquad (101)$$

The small value of ε allows the constant coefficient $C_{\delta Y} = 2\varepsilon - \varepsilon^2 = (2{-}\varepsilon)\varepsilon$ to simplify to 2ε, so the equation's determinate becomes $D_{\delta Y} = 4[(\Delta Y/X_0)^2 - 2\varepsilon]$. The determinate simplifies to $D_{\delta Y} \approx B_{\delta Y}^2 = 4(\Delta Y/X_0)^2$ because of the condition $\Delta Y/X_0 > 3.2{\times}\sqrt{2\varepsilon}$. The corresponding Q value becomes

$$Q_{\delta Y} = \tfrac{1}{2}\left(|B_{\delta Y}| + \sqrt{|D_{\delta Y}|}\right) \approx \tfrac{1}{2}(|B_{\delta Y}| + |B_{\delta Y}|) = |B_{\delta Y}| = 2(\Delta Y/X_0).$$

The value $B_{\delta Y} = 2(\Delta Y/X_0)$ is positive, so the Figure 8 quadratic-equation algorithm gives the desired greater solution of (101) as $\delta Y/X_0 = -C_{\delta Y}/Q_{\delta Y} = -2\varepsilon/[2(\Delta Y/X_0)]$:

$$\left|\frac{\delta Y}{X_0}\right| = -\frac{\delta Y}{X_0} = \frac{\varepsilon}{\Delta Y/X_0} \qquad \text{for } \;\Delta Y/X_0 > 3.2{\times}\sqrt{2\varepsilon}. \qquad (102)$$

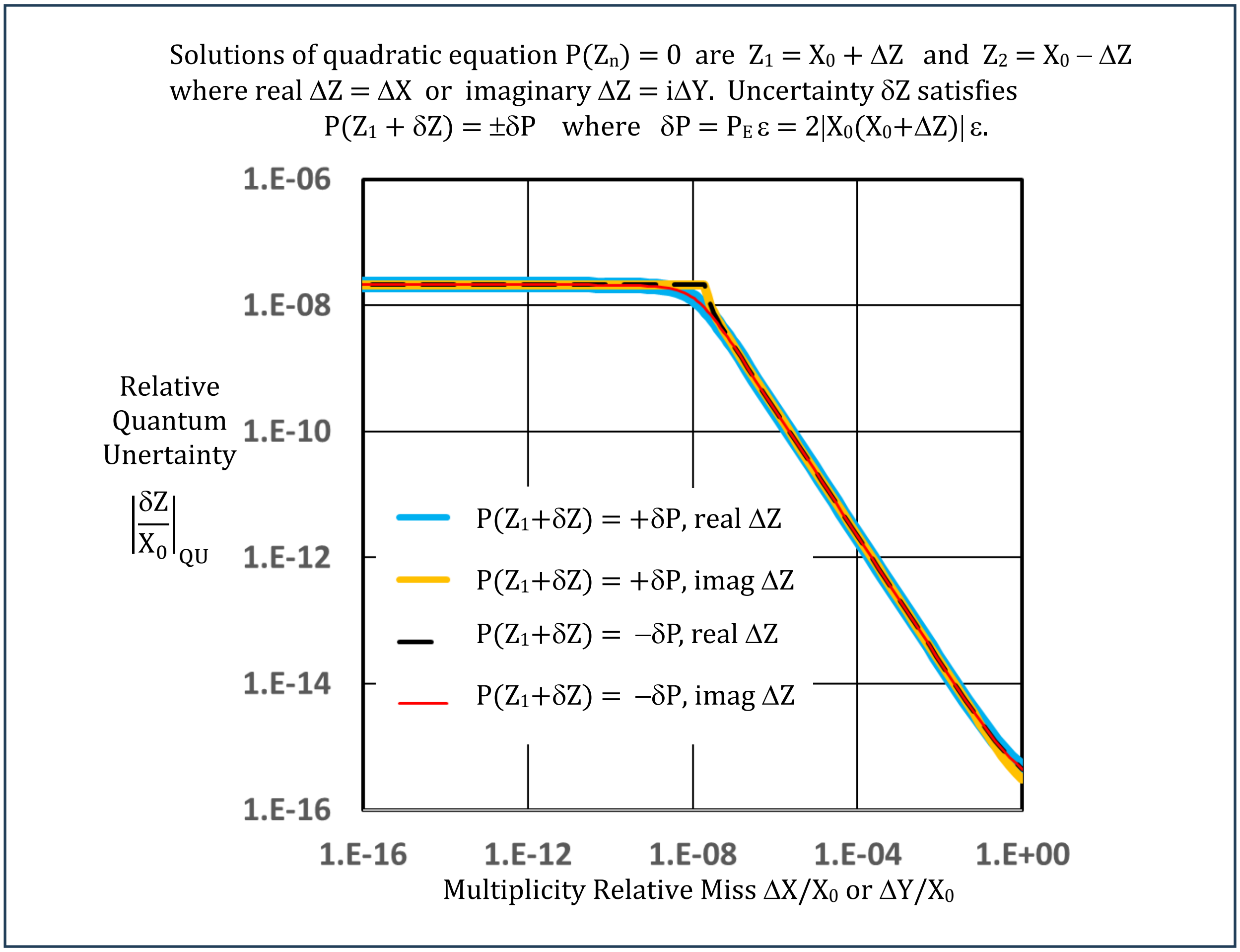
This result is shown as the diagonal decreasing yellow dashed line for $|\delta Y/X_0|$ in Figure 16, which is similar to that for $|\delta Z/X_0|_{QU}$ in Equation (89) when $\Delta Z = \Delta X$ is real and $\sqrt{2\varepsilon} << \Delta X/X_0 << 1$.

Equations (100) for $\delta X/X_0$ and (102) for $|\delta Y/X_0|$ show that

$$|\delta X/X_0| = |\delta Y/X_0| = \varepsilon \qquad \text{for} \qquad \Delta Y/X_0 = 1$$

Figure 17 below compares the relative quantum uncertainties from Figures 15 (real $\Delta Z = \Delta X$) and 16 (imaginary $\Delta Z = i\Delta Y$). These are the light blue and yellow curves respectively in Figure 17. Recall that these values apply to the quadratic equation $P(Z_1+\delta Z) = +\delta P$, Equation (85) using $+\delta P$ on the right side. The Figure 17 dashed black curve and red curve show the relative quantum uncertainties using $-\delta P$ on the right side of Equation (85): $P(Z_1+\delta Z) = -\delta P$. The formulas for these latter two curves are derived below.

Figure 17 Quantum Uncertainty for Quadratic-Equation Multiplicity Near Miss -- Comparison of Four Calculations



Solutions of quadratic equation $P(Z_n) = 0$ are $Z_1 = X_0 + \Delta Z$ and $Z_2 = X_0 - \Delta Z$ where real $\Delta Z = \Delta X$ or imaginary $\Delta Z = i\Delta Y$. Uncertainty δZ satisfies $P(Z_1 + \delta Z) = \pm\delta P$ where $\delta P = P_E\,\varepsilon = 2|X_0(X_0+\Delta Z)|\,\varepsilon$.

Using $P(Z_1+\delta Z) = -\delta P$ instead of $P(Z_1+\delta Z) = +\delta P$, the constant coefficient in Equation (86) changes sign.

$$\left(\frac{\delta Z}{X_0}\right)^2 + 2\,\frac{\Delta Z}{X_0}\,\frac{\delta Z}{X_0} + 2\left(1 + \frac{\Delta Z}{X_0}\right)\varepsilon = 0 \quad \text{for} \quad P(Z_1 + \delta Z) = -\delta P \qquad (103)$$

The case of real $\Delta Z = \Delta X$ produces the determinate of this quadratic equation as

$$D = 4[(\Delta X/X_0)^2 - 2(1 + \Delta X/X_0)\varepsilon].$$

The determinate is negative for $\Delta X/X_0$ less than about $\sqrt{2\varepsilon}$. The pertinate solution of Equation (103) is then

$$\delta Z/X_0 = -\,\Delta X/X_0 + i\sqrt{-D}/2, \qquad \Delta X/X_0 < \sqrt{2\varepsilon}$$

whose absolute value is

$$|\delta Z/X_0|_{QU} = \sqrt{(\Delta X/X_0)^2 + 2(1+\Delta X/X_0)\varepsilon - (\Delta X/X_0)^2} = \sqrt{2(1+\Delta X/X_0)\varepsilon} \;\approx\; \sqrt{2\varepsilon}$$

$$|\delta Z/X_0|_{QU} \;\approx\; \sqrt{2\varepsilon} \qquad \text{for} \qquad \Delta X/X_0 < \sqrt{2\varepsilon} \;\text{ and }\; P(Z_1+\delta Z) = -\delta P.$$

This is the same value plotted in Figure 15 for $\Delta X/X_0 < \sqrt{2\varepsilon}$.

For $\Delta X/X_0 > \sqrt{2\varepsilon}$, the determinate D is positive. The solution of Equation (103) is that given by Equation (87) with a sign change for terms that contain ε.

$$\left|\frac{\delta Z}{X_0}\right|_{QU} = \left|\frac{-2(1+\Delta X/X_0)\varepsilon}{|\Delta X/X_0| + \sqrt{(\Delta X/X_0)^2 - 2(1+\Delta X/X_0)\varepsilon}}\right| \qquad \begin{array}{c}\text{for } \Delta Z = \Delta X > X_0\sqrt{2\varepsilon} \text{ and} \\ P(Z_1+\delta Z) = -\delta P.\end{array}$$

This value of $|\delta Z/X_0|_{QU}$ versus $\Delta X/X_0$ is plotted as the dashed black curve in Figure 17.

Finally, we solve Equation (103) for the case of imaginary ΔZ: $\Delta Z = i\Delta Y$. Just as Equation (86) leads to Equations (95) and (96) for $(\delta X/X_0)^2$ and $\delta Y/X_0$, so Equation (103) produces the following.

$$\left[\left(\frac{\delta X}{X_0}\right)^2\right]^2 + B_{\delta X}\left(\frac{\delta X}{X_0}\right)^2 + C_{\delta X} = 0 \;\text{ where }\; B_{\delta X} = \left(\frac{\Delta Y}{X_0}\right)^2 + 2\varepsilon, \quad C_{\delta X} = -\left(\frac{\Delta Y}{X_0}\right)^2\varepsilon^2 \qquad (104)$$

$$\left(\frac{\delta Y}{X_0}\right)^2 + B_{\delta Y}\left(\frac{\delta Y}{X_0}\right) + C_{\delta Y} = 0 \;\text{ where }\; B_{\delta Y} = 2\left(\frac{\Delta Y}{X_0}\right), \quad C_{\delta Y} = -2\varepsilon - \left(\frac{\delta X}{X_0}\right)^2 \qquad (105)$$

The only difference between these and Equations (95) and (96) is that the sign of 2ε is reversed in the formulas for $B_{\delta X}$ and $C_{\delta Y}$.

The values of both $C_{\delta X}$ and $C_{\delta Y}$ are negative, so Equations (104) and (105) each have two solutions of opposite sign. The positive solutions are the ones of interest. Quantity $(\delta X/X_0)^2$ in Equation (104) must be positive, so it is calculated as the positive solution. The maximum value of $(\delta X/X_0)^2$ is ε^2, so $C_{\delta Y} \approx -2\varepsilon$. The value of $\delta Y/X_0$ in Equation (105) must then have an absolute value several orders of magnitude less than 1. The negative solution of Equation (105) grows to –2 at $\Delta Y/X_0 = 1$ and, therefore, cannot be $\delta Y/X_0$. The positive solution is $\delta Y/X_0$.

From $(\delta X/X_0)^2$ and $\delta Y/X_0$ as the positive solutions of Equations (104) and (105), relative quantum uncertainty is calculated as

$$|\delta Z/X_0|_{QU} = \sqrt{(\delta X/X_0)^2 + (\delta Y/X_0)^2}$$

and plotted as the red curve in Figure 17.

Quadratic Equation Zero-Guard Range

In the multiplicity near-miss condition where $Z_1 = X_0 + \Delta Z$, $Z_2 = X_0 - \Delta Z$, and $|\Delta Z| < X_0$, there is a range of small ΔZ values for which the determinate $D = B^2 - 4C$ fails to exceed its quantum error value $D_E\,\varepsilon$ in the Figure 8 final quadratic-equation algorithm. Zero-guard processing resets D to zero in these cases, and calculates $Z_1 = Z_2 = -B/2$. The maximum absolute value $|\Delta Z|$ of such ΔZ values is called the zero-guard range $|\Delta Z|_{ZG}$.

Calculation of $|\Delta Z|_{ZG}$ for the near-miss condition is straight-forward. Equation (83) gives the coefficients as $B = -2X_0$ and $C = X_0^2 - \Delta Z^2 > 0$, so the determinate is

$$D = B^2 - 4C = 4\Delta Z^2.$$

Equation (33) gives D_E as

$$D_E = 2|B|\,B_E + 4C_E = 2|B|^2 + 4|C| = 2B^2 + 4C = 12X_0^2 - 4\Delta Z^2$$

The zero-guard range is found by equating $D = 4\Delta Z^2$ to $D_E\varepsilon$ and solving for ΔZ.

$$D = 4\Delta Z^2 = (12X_0^2 - 4\Delta Z^2)\varepsilon \qquad \Rightarrow \qquad \Delta Z^2 = 3X_0^2\,\varepsilon/(1-\varepsilon) \approx 3X_0^2\,\varepsilon$$

The relative zero-guard range $|\Delta Z/X_0|_{ZG}$ becomes

$$|\Delta Z/X_0|_{ZG} \approx \sqrt{3\varepsilon} \approx 2.58{\times}10^{-8},$$

a value only 22% greater than the relative quantum uncertainty of $\sqrt{2\varepsilon} \approx 2.11{\times}10^{-8}$.

Figure 18 below shows the relative zero-guard range in relation to the Figure 15 quantum uncertainty for real near miss ($\Delta Z = \Delta X$) and the Figure 16 total quantum uncertainty for imaginary near miss ($\Delta Z = i\Delta Y$). The zero-guard range is shown as the vertical dashed red line at the relative miss value of $\sqrt{3\varepsilon}$ on the horizontal axis. For all relative miss values $|\Delta Z/X_0|$ to the left, i.e. for $|\Delta Z/X_0| < \sqrt{3\varepsilon}$, the zero-guard processing causes the calculated quadratic-equation solutions to be $Z_{1C} = Z_{2C} = -B/2 = X_0$. Zero-guard processing does not affect the calculated solutions for $|\Delta Z/X_0| \geq \sqrt{3\varepsilon}$.

The diagonal red line plots the theoretical zero-guard relative error $|\delta Z/X_0|_{ZG}$, which is the absolute value of the calculated solution Z_{1C} minus the true solution Z_1, all normalized by X_0: $|\delta Z/X_0|_{ZG} = |[X_0 - (X_0 + \Delta Z)]/X_0| = |\Delta Z/X_0|$.

$$|\delta Z/X_0|_{ZG} = |\Delta Z/X_0| \qquad \text{for} \qquad |\Delta Z/X_0| < \sqrt{3\varepsilon}$$

In actual practice, the effective solution error produced by zero-guard processing is less dramatic than appears in Figure 18.

Figure 18 Comparison of Quadratic-Equation Zero-Guard Range to Quantum Uncertainty

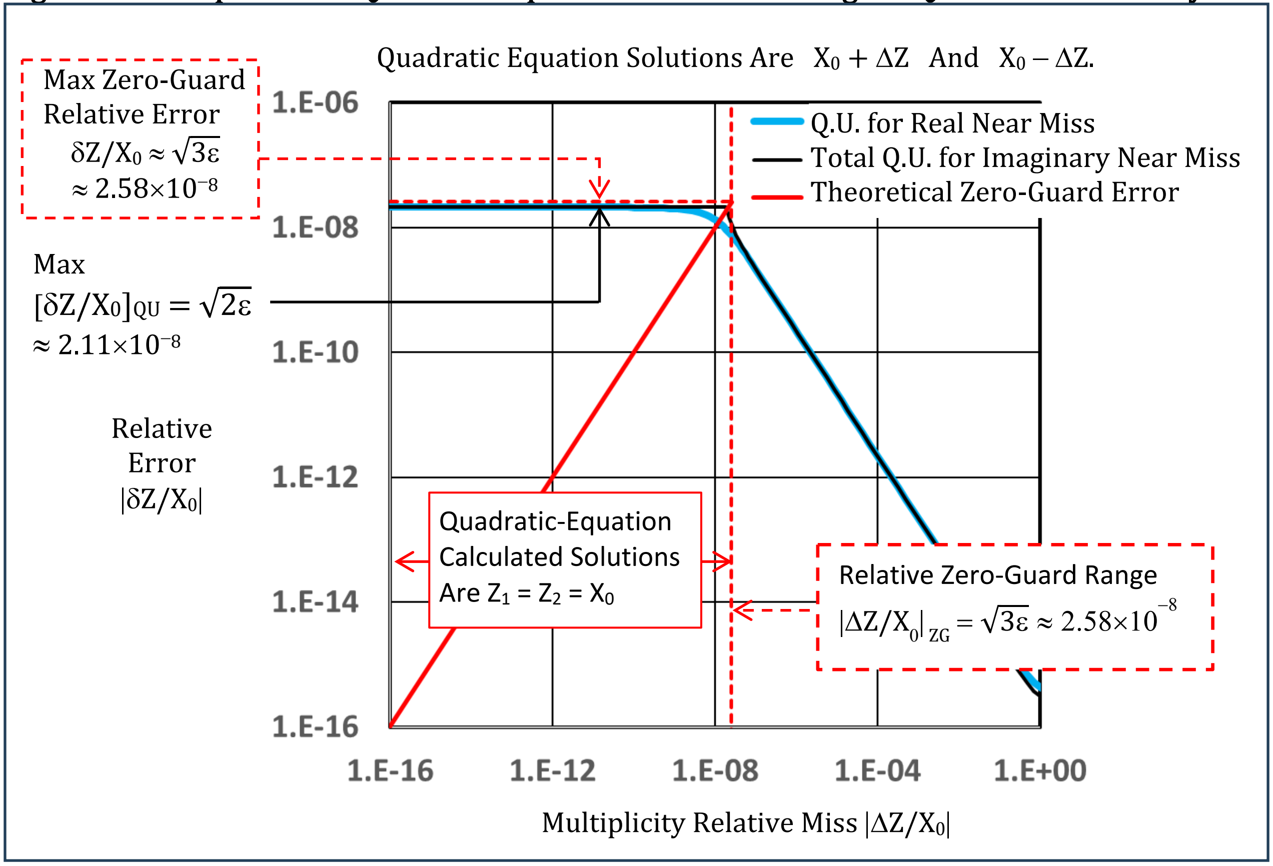
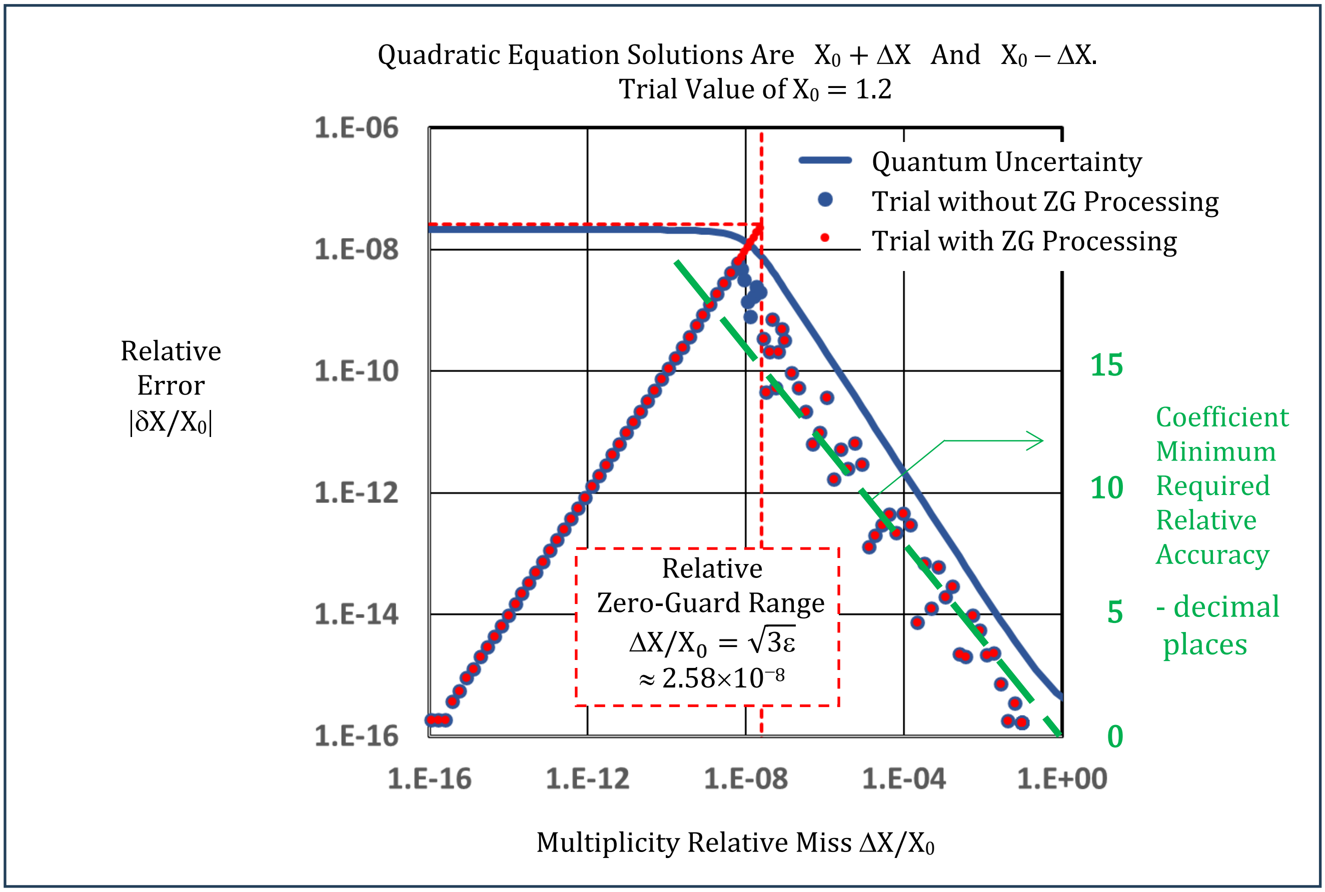


Figure 19 on page 87 compares the relative zero-guard range to computer trial results for the real relative near miss $\Delta X/X_0$ at $X_0 = 1.2$. The larger blue dots plot error results when zero-guard processing is disabled; the smaller red dots show error with zero-guard processing enabled. These results are typical for calculated solutions of stand-alone quadratic equations. The zero-guard processing affects the error results only for a small range of relative miss values just less than the zero-guard cut-off. In this example, zero-guard processing affects the error results only over the range
$7\times10^{-9} < \Delta X/X_0 < \sqrt{3\varepsilon} \approx 2.58\times10^{-8}$.

The normal Numerical Recipes algorithm of Figure 4 accurately calculates quadratic-equation solutions for the multiplicity condition ($\Delta X/X_0 = 0$). I am unable to find any quadratic-equation example where this is not so. As $\Delta X/X_0$ increases from zero, the Figure 4 algorithm continues to calculate the multiplicity result $Z_{1C} = Z_{2C} = X_0$ until $\Delta X/X_0$ is great enough that the computer can store the constant coefficient $C = X_0^2 - \Delta Z^2 = X_0^2\ (1 - \Delta Z^2/X_0^2)$ as something other than X_0^2. Until that point, the theoretical zero-guard error is meaningless because the calculated solutions are $Z_{1C} = Z_{2C} = X_0$, whether or not zero-guard processing is enabled.

Figure 19 Effect of ZG Processing on Quadratic-Equation Trial Solution Error for Multiplicity Real Near Miss



Only when computed results differ between zero-guard processing enabled and disabled can we say that zero-guard processing produces an effective error, as shown in Figure 20, an expanded view of Figure 19. Even this effective zero-guard error is not a practical concern. At the zero-guard range, we have $\Delta X/X_0 = (\Delta X/X_0)_{ZG} = \sqrt{3\varepsilon}$. The coefficients B and C are

$$B = -(Z_1 + Z_2) = -(X_0{+}\Delta X + X_0{-}\Delta X) = -2X_0$$

$$C = Z_1 Z_2 = (X_0{+}\Delta X)(X_0{-}\Delta X) = X_0^2(1 - \Delta X^2/X_0^2) = X_0^2(1 - 3\varepsilon)$$

The calculated solutions are $Z_{1C} = Z_{2C} = X_0$, so the check coefficients are

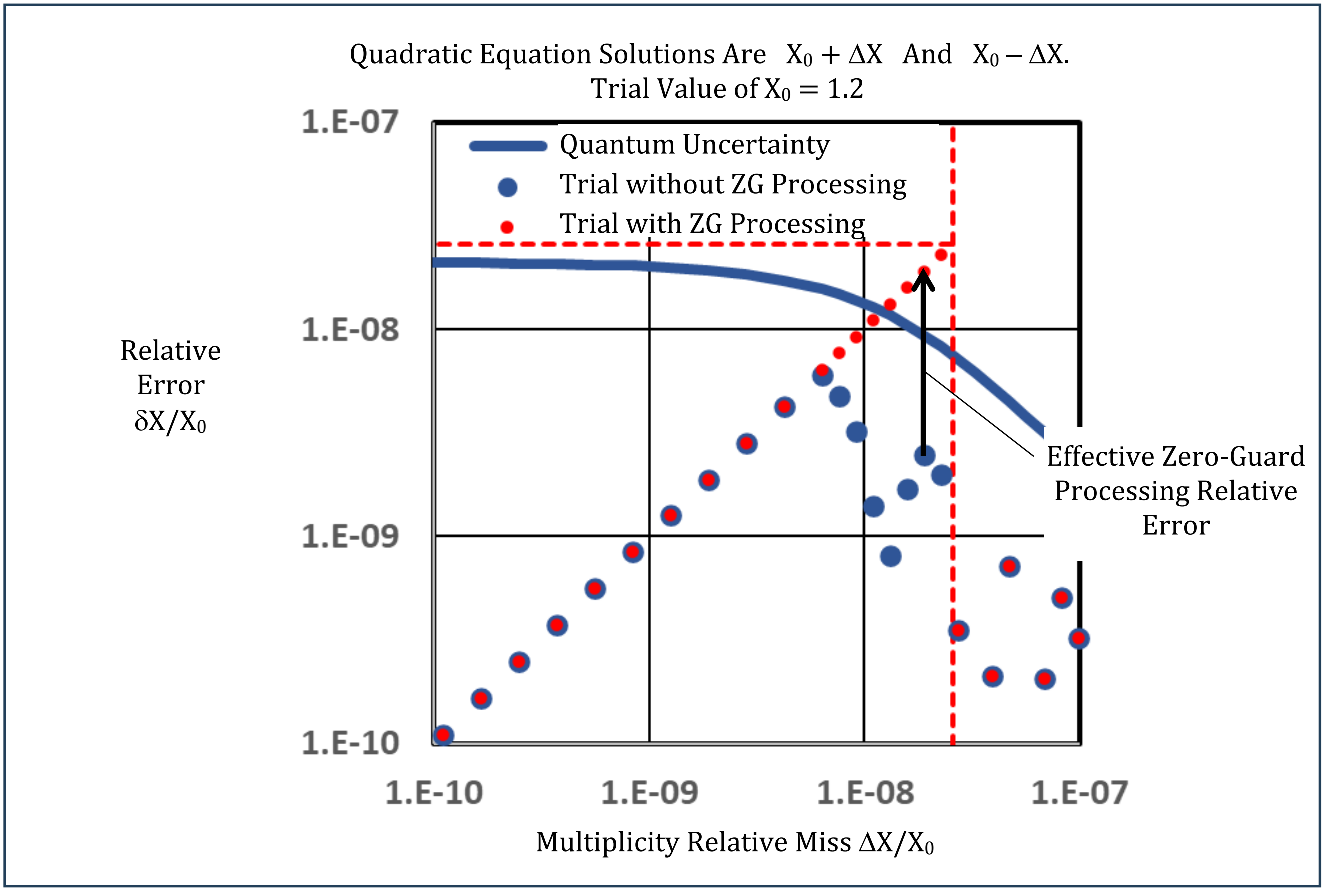
$$B_C = -(Z_{1C} + Z_{2C}) = -(X_0{+}X_0) = -2X_0 \;=\; B$$

$$C_C = Z_{1C}\, Z_{2C} \;=\; X_0^2$$

The relative coefficient errors are

$$\delta B_u \equiv \left|\frac{B_C - B}{B}\right| = 0 \quad \text{and} \quad \delta C_u \equiv \left|\frac{C_C - C}{C}\right| = \left|\frac{3\varepsilon}{1 - 3\varepsilon}\right| \approx 3\varepsilon \;\approx\; 6.66\times10^{-16}.$$

Figure 20 Effect of ZG Processing on Quadratic-Equation Trial Solution Error for Multiplicity Real Near Miss -- Expanded View



Thus, the user would have to supply coefficient relative accuracy to better than 15 decimal places in order to write a quadratic equation with nonzero, near-miss relative displacement $\Delta X/X_0$ equal to $(\Delta X/X_0)_{ZG} = \sqrt{3\varepsilon}$ or less.

For the example case of Figures 19 and 20 with $X_0 = 1.2$, the value of C at $\Delta X/X_0 = 0$ is 1.44. At $\Delta X/X_0 = (\Delta X/X_0)_{ZG} = \sqrt{3\varepsilon}$, the C value is 1.43999999999999904. The final 9 appears in the fifteenth decimal place. This C value is so close to 1.44 that it is displayed as 1.44 in the Excel spreadsheet, even though the internal spreadsheet value is less than 1.44.

We conclude that zero-guard processing in the quadratic-equation algorithm is superfluous for solving stand-alone quadratic equations. The normal Numerical Recipes algorithm of Figure 4 accurately calculates quadratic-equation solutions for the multiplicity condition without engaging zero-guard processing. On the other hand, any theoretical error induced by the zero-guard processing is not a practical concern.

Necessity of Zero-Guard Processing in the Quadratic-Equation Algorithm

Zero-guard processing in the quadratic-equation algorithm of Figure 8 is necessary to produce accurate solutions for some cubic- and quartic-equations that require post-processing. Two such equations are:

$z_n^3 - 5.20000001\, z_n^2 + 5.2000000025{\times}10^{-8}\, z_n - 1.3{\times}10^{-16} = 0$ with true solutions 5.2, $5{\times}10^{-8}$, and $5{\times}10^{-8}$

$Z_n^4 - 12.20000001\, Z_n^3 - 36.400000122\, Z_n^2 - 3.64000000305{\times}10^{-7}\, Z_n + 9.1{\times}10^{-16} = 0$ with true solutions 7, 5.2, $5{\times}10^{-8}$, and $5{\times}10^{-8}$

Post processing for these two equations engages the Figure 8 quadratic-equation algorithm, which resets its initial non-zero D value to zero and thereby calculates the double solution $X_1 = X_2 = 5{\times}10^{-8}$ with $Y = 0$. Relative solution error is less than $\varepsilon \approx 2.22{\times}10^{-16}$.

If we turn off the quadratic algorithm's zero-guard processing (if we use the Figure 4 algorithm instead of Figure 8), then the two calculated solutions differ from each other, and relative solution error is on the order of 10^{-8} instead of 10^{-16}.

Coefficient Minimum Required Relative Accuracy

Figure 19 above demonstrates one final point: the dashed green line plots the coefficient minimum required relative accuracy on the right vertical axis as a function of $\Delta X/X_0$. For any displacement ΔZ from the multiplicity solution X_0, the constant coefficient is

$$C = X_0^2 - \Delta Z^2 = X_0^2[1 - (\Delta Z/X_0)^2]$$

$$C = X_0^2[1 - (\Delta X/X_0)^2] \ \text{ if } \Delta Z = \Delta X \qquad \text{OR} \qquad C = X_0^2[1 + (\Delta Y/X_0)^2] \ \text{ if } \Delta Z = i\Delta Y$$

For the multiplicity near-miss condition where $|\Delta Z/X_0|$ is small, the user must supply the C value with sufficient accuracy to distinguish it from $C = X_0^2$, or equivalently, to distinguish the factor $[1 - (\Delta Z/X_0)^2]$ from 1. Suppose for example, $\Delta Z/X_0 = \Delta X/X_0 = 1{\times}10^{-4}$. Then $[1 - (\Delta X/X_0)^2] = [1 - 1{\times}10^{-8}] = 0.99999999$. The minimum required relative accuracy is eight decimal places. The number $\mathcal{N}$ of decimal places, plotted as the dashed green in line in Figure 19, is calculated as

$$\mathcal{N} = -2\log_{10}(|\Delta Z/X_0|).$$

If the constant coefficient has a relative accuracy limited to eight decimal places, then the smallest, nonzero near-miss relative displacement that the quadratic-equation can represent is $\Delta Z/X_0 = 1{\times}10^{-4}$.

This concludes error analysis for quadratic equations, and we proceed with cubic equations.

IX. CUBIC EQUATION ANALYSIS – MULTIPLICITY 3, ITS NEAR MISS, AND MULTIPLICITY 2

This section and the next demonstrate how zero-guard processing and post processing work together to produce good solution accuracy for the multiplicity conditions. Section X addresses multiplicity 2 near miss.

This section addresses multiplicity 3, its near miss, and multiplicity 2. The analysis is based on a cubic polynomial p(z) with two equal real roots x_0 and the third real root x_A. Neither x_0 nor x_A is zero; otherwise, the Special Case 1, $a_0 = 0$ applies. For multiplicity 3 where $x_A = x_0$, we show that the relative quantum uncertainty $|\delta z/z_0|_{QU}$ is $(3\varepsilon)^{1/3} \approx 8.73\times10^{-6}$. The maximum relative solution error imposed by zero-guard processing is 15% greater at 1×10^{-5}, but the corresponding relative coefficient errors (Equation (82)) are small at $\delta a_{2u} = \delta a_{1u} = 0$, and $\delta a_{0u} = 2\times10^{-15}$.

The cubic polynomial is

$$p(z) = z^3 + a_2\, z^2 + a_1\, z + a_0 \;=\; (z - x_0)^2\, (z - x_A) \tag{106}$$

where

$$a_2 = -(x_A + 2x_0), \qquad a_1 = 2x_Ax_0 + x_0^2, \qquad a_0 = -x_A\, x_0^2, \qquad x_0 \neq 0, \quad x_A \neq 0. \tag{107}$$

Therefore:

$$p(z) = 0 \qquad \text{for} \qquad z = x_0 \quad \text{or} \quad z = x_A. \tag{108}$$

The cubic polynomial's inherent symmetry allows us to take $x_A \geq x_0$ without sacrificing generality. To demonstrate, let $p_2(z)$ have two equal roots x_{02} and the third real root x_{A2} where $x_{A2} < x_{02}$. Then $p_2(z) = (z - x_{02})^2\, (z - x_{A2})$. Define $x_0 = -x_{02}$ and $x_A = -x_{A2}$. Then $x_{A2} < x_{02} \Rightarrow x_A > x_0$ and

$$\begin{aligned} p_2(z) &= (z + x_0)^2\, (z + x_A) &&= z^3 + (2x_0 + x_A)\, z^2 + (2x_0x_A + x_0^2)\, z + x_A\, x_0^2 \\ &= z^3 - a_2\, z^2 + a_1\, z - a_0 &&= -p(-z). \end{aligned}$$

The properties of $p_2(z)$ are the properties of p(z) with all of the signs reversed. The roots and root errors of $p_2(z)$ are the negatives of the roots and root errors of p(z). The error δp_2 in function $p_2(z)$ is the negative of error δp in p(z). That is: $|\delta p_2| = |\delta p|$. We therefore adopt

$$x_A \;\geq\; x_0 \quad \text{by convention.} \tag{109}$$

With $x_A > x_0$, the double root x_0 occurs at a local maximum of p(z), and the derivative p′(z) at $z = x_A$ is positive.

In addition to taking $x_A \geq x_0$, normalization by x_0 simplifies the analysis even further.

Define the following. $\qquad u \equiv z/x_0 \qquad \eta \equiv x_A/x_0 \qquad p_u(u) \equiv p(z)/x_0^3 \qquad$ (110)

Then

$$p_u(u) = u^3 + a_{2u}\, u^2 + a_{1u}\, u + a_{0u} \tag{111}$$

where

$$a_{2u} \equiv a_2/x_0 = -(\eta + 2) \qquad a_{1u} \equiv a_1/x_0^2 \;= 2\eta + 1 \qquad a_{0u} \equiv a_0/x_0^3 = -\eta. \tag{112}$$

The cubic $p_u(u)$ has roots 1, 1, and η.

$$p_u(u) = (u-1)^2\,(u-\eta) \qquad (113)$$

$$p_u(u) = 0 \quad \text{for} \quad u = 1 \quad \text{and} \quad u = \eta. \qquad (114)$$

When x_0 is positive, then u and z have the same sign, and $\eta \geq 1$. If $x_A \neq x_0$, then the double root $u = 1$ occurs at a local maximum of $p_u(u)$, and the simple root $u = \eta$ is greater than 1.

When x_0 is negative, then u and z have opposite signs, and $\eta \leq 1$. If $x_A \neq x_0$, then the double root $u = 1$ occurs at a local minimum of $p_u(u)$, and the simple root $u = \eta$ is less than 1.

Whatever the sign of x_0, the derivative $p_u'(u)$ at the simple root $u = \eta$ is positive.

Setting $x_A = x_0 \Leftrightarrow \eta = 1$ produces the multiplicity 3 condition in p(z) and $p_u(u)$. The coefficients in (111) and (112) become

$$a_{2u} = -3 \qquad a_{1u} = 3 \qquad a_0 = -1 \qquad \text{for} \qquad \eta = 1.$$

Quantum Uncertainty

We can now calculate the quantum uncertainty $|\delta u|_{QU} = |\delta z/x_0|_{QU}$ of the multiplicity 3 root $u = 1$. The cubic $p_u(1)$ is evaluated as the sum

$$p_u(1) = 1^3 - 3(1^2) + 3(1) - 1 = 1 - 3 + 3 - 1 = 0.$$

The magnitude of the sum's least significant bit is that of its greatest-magnitude term. The second and third terms both have the greatest magnitude of 3; therefore, the magnitude of $p_u(1)$'s least significant bit is $\delta p_u = 3\varepsilon$. This δp_u is the magnitude of the range of p_u values that would be stored in the computer as $p_u = 0$. To find the resulting uncertainty $|\delta u|_{QU}$ in the root $u = 1$, solve $p_u(1 + \delta u) = \pm\delta p_u$ for δu. Use Equation (113) with $\eta = 1$.

$$p_u(1 + \delta u) = (1 + \delta u - 1)^3 = \delta u^3 = \pm\delta p_u = \pm 3\varepsilon \qquad \Rightarrow \qquad |\delta u| = (3\varepsilon)^{1/3}$$

The relative quantum uncertainty of solution x_0 of a multiplicity-three cubic equation is

$$\boxed{|\delta z/x_0|_{QU} \;=\; |\delta u|_{QU} \;=\; (3\varepsilon)^{1/3} \;\approx\; 8.73\times 10^{-6} \;\; \text{(multiplicity 3 quantum uncertainty).} \qquad (115)}$$

We study multiplicity 3 near miss and multiplicity 2 by incrementally increasing the x_A value above x_0. The corresponding $\eta = x_A/x_0$ increases above 1 if $x_0 > 1$ or decreases below 1 if $x_0 < 0$. For either sign of x_0, the magnitude change in η from 1 is $|\eta - 1|$. The cubic $p_u(u)$ now has two different quantum uncertainty values of interest: one at the double root $u = 1$ and one at the simple root $u = \eta$.

Start with the quantum uncertainty at the double root $u = 1$. From Equation (111), the cubic $p_u(1)$ is evaluated as the sum

$$p_u(1) = 1^3 + a_{2u}\,(1^2) + a_{1u}\,(1) + a_{0u} \;=\; 1 + a_{2u} + a_{1u} + a_{0u} \;=\; 0$$

The magnitude of the polynomial's least significant bit is that of its greatest-magnitude term. The magnitude of the greatest-magnitude term is MAX(1, $|a_{2u}|$, $|a_{1u}|$, $|a_{0u}|$). Applying the coefficient values from Equation (112), the magnitude of the polynomial's least significant bit is therefore

$$\delta p_u = \mathrm{MAX}(1, |a_{2u}|, |a_{1u}|, |a_{0u}|)\,\varepsilon \;=\; \mathrm{MAX}(1, |\eta+2|, |2\eta+1|, |\eta|)\,\varepsilon\,. \qquad (116)$$

Use Equation (113) to find the quantum uncertainty $|\delta z/x_0|_{QU}$. Set $p_u(1 + \delta u)$ equal to $\pm\delta p_u$, and solve for δu.

$$p_u(1 + \delta u) = \delta u^2\,(1 + \delta u - \eta) = \pm\delta p_u$$

The desired δu is a solution of the cubic equation

$$\delta u^3 + (1-\eta)\delta u^2 - (\pm\delta p_u) = 0 \quad \text{where} \quad \delta p_u = \mathrm{MAX}(1, |\eta+2|, |2\eta+1|, |\eta|)\,\varepsilon. \qquad (117)$$

CUBIC EQUATION FOR RELATIVE QUANTUM UNCERTAINTY $|\delta u|_{QU} = |\delta z/x_0|_{QU}$ OF DOUBLE ROOT x_0 $(u=1)$

Selection of this equation's appropriate solution is described shortly.

For now, consider the quantum uncertainty at the simple root $u = \eta$. From (111), the cubic $p_u(\eta)$ is evaluated as the sum

$$p_u(\eta) = \eta^3 + a_{2u}\,\eta^2 + a_{1u}\,\eta + a_{0u} \;=\; 0$$

The magnitude of the polynomial's least significant bit is that of its greatest-magnitude term. The magnitude of the greatest-magnitude term is $\mathrm{MAX}(|\eta^3|, |a_{2u}\eta^2|, |a_{1u}\eta|, |a_{0u}|)$, and the magnitude of the polynomial's least significant bit is therefore

$$\delta p_u = \mathrm{MAX}(|\eta^3|,\; |\eta+2|\eta^2,\; |(2\eta+1)\eta|,\; |\eta|)\varepsilon.$$

Use Equation (113) to find $|\delta u| = |\delta z/x_0|$. Set $p_u(\eta + \delta u)$ equal to $\pm\delta p_u$, and solve for δu.

$$p_u(\eta + \delta u) = (\eta - 1 + \delta u)^2\,\delta u = \pm\delta p_u$$

The desired δu is a solution of the cubic equation

$$\delta u^3 + 2(\eta-1)\delta u^2 + (\eta-1)^2\,\delta u - (\pm\delta p_u) = 0 \qquad (118)$$

$$\text{where } \delta p_u = \mathrm{MAX}(|\eta^3|, |\eta+2|\eta^2, |(2\eta+1)\eta|, |\eta|)\varepsilon.$$

CUBIC EQUATION FOR RELATIVE QUANTUM UNCERTAINTY $|\delta z/x_A|_{QU} = |\delta u/\eta|$ OF SIMPLE ROOT x_A $(u=\eta)$

The relative quantum uncertainty $|\delta z/x_A|_{QU}$ for x_A is normalized by x_A, but u is defined as $u \equiv z/x_0$. We therefore divide δu by $\eta \equiv x_A/x_0$ to obtain $|\delta z/x_A|_{QU}$:
$|\delta z/x_A|_{QU} = |\delta z/x_0|/|x_A/x_0| = |\delta u/\eta|$.

Each of the Equations (117) and (118) has three solutions from which to choose for δu. The proper choice of solution for each equation depends on the range of x_0 and on the sign of the function error $\pm\delta p_u$. Table IX summarizes the proper solutions as discussed below. Following that discussion, Figure 21 plots the resulting quantum uncertainties $|\delta z/x_0|_{QU}$ and $|\delta z/x_A|_{QU}$ as functions of $|\eta - 1| = |(x_A - x_0)/\,x_0|$.

Table IX. Selecting from Among Equation (117) and (118) Solutions for δu

<table>
<tr><td colspan="4">Cubic equations (117) and (118) each have three solutions: z_1, $z_2 = x_2 + iy_2$, and $z_3 = x_3 - iy_2$. The third and fourth columns below show which of these three solutions is the proper value of δu for the condition defined in the first two columns. The relative quantum uncertainties $|\delta z/x_0|_{QU}$ and $|\delta z/x_A|_{QU}$ follow directly from δu.</td></tr>
<tr><th>Range of Double Root x_0</th><th>Sign of Function Error $\pm\delta p_u$</th><th>Double Root
$z = x_0 \quad u = 1$
Equation (117)</th><th>Simple Root
$z = x_A \quad u = \eta$
Equation (118)</th></tr>
<tr><td>$x_0 > 0$</td><td>$+\delta p_u$</td><td>$\delta u = z_2 = x_2 + iy_2$</td><td>$\delta u = z_1 > 0$</td></tr>
<tr><td>$x_0 > 0$</td><td>$-\delta p_u$</td><td>$\delta u = z_1 \quad < 0 \quad \text{if } y_2 \neq 0$
$\delta u = z_3 = x_3 < 0 \quad \text{if } y_2 = 0$</td><td>$\delta u = z_2 = x_2 + iy_2 \quad \text{if } y_2 \neq 0$
$\delta u = z_1 < 0 \quad \text{if } y_2 = 0$</td></tr>
<tr><td>$x_0 < 0$</td><td>$+\delta p_u$</td><td>$\delta u = z_1 > 0$</td><td>$\delta u = z_2 = x_2 + iy_2 \quad \text{if } y_2 \neq 0$
$\delta u = z_1 > 0 \quad \text{if } y_2 = 0$</td></tr>
<tr><td>$x_0 < 0$</td><td>$-\delta p_u$</td><td>$\delta u = z_2 = x_2 + iy_2$</td><td>$\delta u = z_1 \quad < 0 \quad \text{if } y_2 \neq 0$
$\delta u = z_3 = x_3 < 0 \quad \text{if } y_2 = 0$</td></tr>
<tr><td colspan="2">Quantum Uncertainty =</td><td>$|\delta z/x_0|_{QU} = |\delta u|$</td><td>$|\delta z/x_A|_{QU} = |\delta u/\eta|$</td></tr>
</table>

Whether the double root $u = 1$ of $p_u(u)$ occurs at a local maximum or local minimum of $p_u(u)$ depends on the sign of x_0. The double root x_0 of $p(z)$ always occurs at a local maximum of $p(z)$ because the simple root x_A is greater than or equal to x_0. If $x_0 > 0$, then $u = z/x_0$ has the same sign as z, and like double root $z = x_0$ of $p(z)$, the double root $u = 1$ of $p_u(u)$ occurs at a local maximum. A positive function error $+\delta p_u$ implies that δu at the $u = 1$ local maximum cannot be real. The proper solution of (117) for δu at the double root is the complex solution $\delta u = z_2 = x_2 + iy_2$. Thus, the double-root quantum uncertainty is $|\delta z/x_0|_{QU} = |\delta u| = \sqrt{x_2^2 + y_2^2}$. A negative function error $-\delta p_u$ implies, however, that δu at $u = 1$ is the only negative real solution of (117). This is either solution z_1 at small $\eta - 1$ when (117) has only one real solution or solution $z_3 = x_3$ when all three solutions are real.

If $x_0 < 0$, then u has the opposite sign of z, and the double root $u = 1$ of $p_u(u)$ occurs at a local minimum. Also, $\eta \leq 1$. A positive function error $+\delta p_u$ implies that δu is (117)'s only positive real solution z_1. A negative function error $-\delta p_u$ implies that δu at $u = 1$ is the complex solution $\delta u = z_2 = x_2 + iy_2$.

Regardless of the x_0 value, the double-root quantum uncertainty is always given by $|\delta z/x_0|_{QU} = |\delta u|$.

At the simple root $u = \eta$, the cubic $p_u(u)$ has a positive derivative $p_u'(u)$ provided that $x_A \neq x_0$ ($\eta \neq 1$). Typically, the proper solution of Equation (118) for δu is the real solution of least magnitude that has the same sign as $\pm\delta p_u$. An exception to this rule occurs as follows.

The cubic $p_u(u)$ has two local extrema where $p_u'(u) = 0$. They occur at $u = 1$ and at $u = (2\eta + 1)/3$. The extremum value of $p_u(u)$ at $u = (2\eta + 1)/3$ is $p_{ext} = -(4/27)(\eta - 1)^3$, a value opposite in sign to both $\eta - 1$ and x_0. If the sign of $\pm\delta p_u$ is the same as that of p_{ext}, and if $|\eta - 1|$ is so small that $|p_{ext}|$ is less than $|\pm\delta p_u|$, then δu cannot be real and must be complex. Consequently, the proper solution of Equation (118) is $\delta u = x_2 + iy_2$ when the sign of $\pm\delta p_u$ is opposite that of x_0 and

$$|p_{ext}| = |(4/27)(\eta - 1)^3| < |\pm\delta p_u| \quad \Rightarrow \quad |\eta - 1| < (27|\pm\delta p_u|/4)^{1/3}.$$

Greater values of $|\eta - 1|$ produce an Equation (118) with three real solutions.

Figure 21 below plots the relative quantum uncertainties $|\delta z/x_0|_{QU}$ and $|\delta z/x_A|_{QU}$, produced from the solutions δu of Equations (117) and (118), as functions of $|\eta - 1| = |(x_A - x_0)/ x_0|$. Figure 21a plots the uncertainties for the range $x_0 > 0$ and for both $+\delta p_u$ and $-\delta p_u$. Figure 21b plots the same uncertainties except that the range $x_0 < 0$ applies. Notice that the overall characteristics of the $|\delta z/x_0|_{QU}$ and $|\delta z/x_A|_{QU}$ curves in Figure 21b are similar to those in Figure 21a. In both 21a and 21b, the double-root uncertainty $|\delta z/x_0|_{QU}$ for $-\delta p_u$ (dashed yellow curve) is almost identical to that for $+\delta p_u$ (solid black curve). This same comment applies to the simple-root uncertainty $[\delta z/\delta x_A]_{QU}$ with an exception in the region of $|\eta - 1| = 10^{-5}$. In 21a, $[\delta z/\delta x_A]_{QU}$ values for $-\delta p_u$ (dashed green curve) are slightly greater than for $+\delta p_u$ (solid blue curve). This order is reversed in 21b.

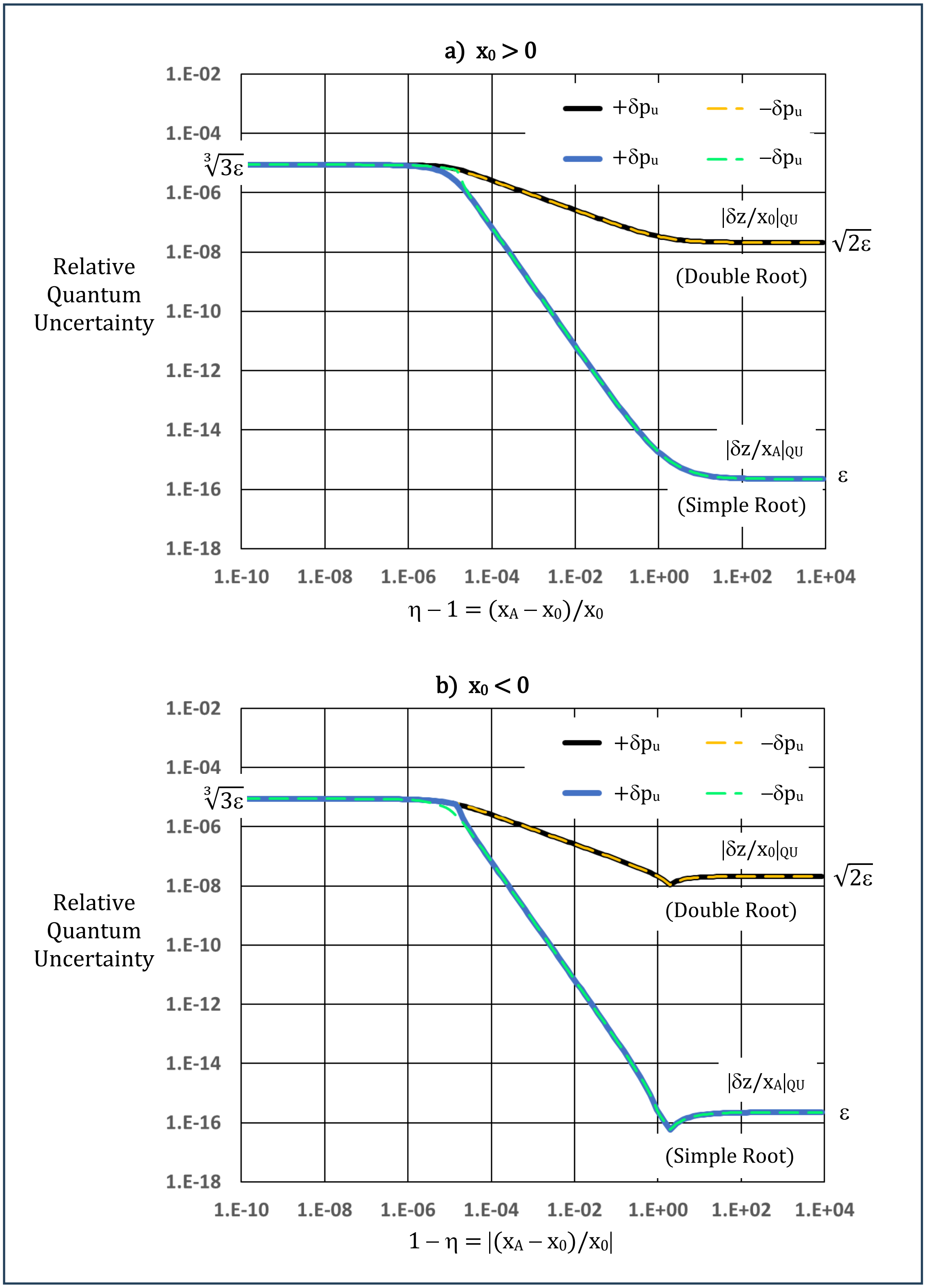
The limiting values of $|\delta z/x_0|_{QU}$ and $|\delta z/x_A|_{QU}$ at very small and very large values of $|\eta - 1|$ can be found easily from Equations (117) and (118). In the limit as $|\eta - 1|$ approaches zero, η is 1, δp_u is 3ε in both (117) and (118), and the cubic equations in (117) and (118) both become $\delta u^3 = \pm 3\varepsilon$. This implies that $|\delta u| = (3\varepsilon)^{1/3}$, and the uncertainties become

$$|\delta z/x_0|_{QU} = |\delta u| = |\delta u/\eta| = |\delta z/x_A|_{QU} = (3\varepsilon)^{1/3} \approx 8.73\times10^{-6} \quad \text{for} \quad |\eta - 1| \to 0.$$

This value of $(3\varepsilon)^{1/3}$ is noted on the vertical axis on the left side of the plots.

As $|\eta - 1|$ in (117) increases without limit, δp_u becomes $2\eta\varepsilon$, and the cubic equation in δu becomes $-\eta\delta u^2 - (\pm 2\eta\varepsilon) = 0$ or $\delta u^2 = \pm 2\varepsilon$. This implies that the relative quantum uncertainty for the double root becomes $|\delta z/x_0|_{QU} = |\delta u| = \sqrt{2\varepsilon} \approx 2.11\times10^{-8}$. This is the same multiplicity quantum-uncertainty level as that for quadratic equations ($[\delta Z/\delta X_0]_{QU}$ in Equation (88)). As $|\eta - 1|$ in (118) increases without limit, δp_u becomes $|\eta^3|\varepsilon$, and the cubic equation in δu becomes $\eta^2\,\delta u - (\pm|\eta^3|\varepsilon) = 0$ or $\delta u = \pm|\eta|\varepsilon$. Thus $|\delta u| = |\eta|\varepsilon$, and the relative quantum uncertainty for the simple root becomes $|\delta z/x_A|_{QU} = |\delta u/\eta| = \varepsilon \approx 2.22\times10^{-16}$. These relative uncertainty values of $|\delta z/x_0|_{QU} = \sqrt{2\varepsilon}$ for the double root and $|\delta z/x_A|_{QU} = \varepsilon$ for the simple root are appropriately noted on the right border of the charts in Figure 21a and 21b.

Figure 21 Cubic-Equation Quantum Uncertainty for Multiplicity 2



Calculated Solution Error

Figure 22 shows example relative solution errors plotted as a function of $\eta - 1$ for the example double root $z_2 = x_2 = z_3 = x_3 = x_0 = 1.2$. At each η trial value, the third root is $z_1 = x_A = x_0\eta = 1.2\eta$. Equation (107) then produces coefficients a_2, a_1, and a_0 for the cubic equation $p(z) = z^3 + a_2 z^2 + a_1 z + a_0 = 0$ to be solved.

The figure plots solution errors for two different cubic-equation computation methods. Figure 22a shows results for solutions calculated using the round-off-error mitigation design: the Figure 9 cubic-equation algorithm, the Figure 12 cubic-equation post-processing algorithm, and the Figure 8 quadratic-equation algorithm. Figure 22b shows results for calculation without mitigation using the Figure 1 algorithm. For reference, Figure 22 also plots the quantum uncertainties $|\delta z/x_0|_{QU}$ and $\delta z/x_A|_{QU}$ from Figure 21a.

The figure's relative errors for solutions z_1, z_2, and z_3 are calculated as follows:

$$|\delta z_{1u}| \equiv \left|\frac{\delta z_1}{x_A}\right| = \left|\frac{z_1 - x_A}{x_A}\right|, \quad |\delta z_{2u}| \equiv \left|\frac{\delta z_2}{x_0}\right| = \left|\frac{z_2 - x_0}{x_0}\right|, \quad |\delta z_{3u}| \equiv \left|\frac{\delta z_3}{x_0}\right| = \left|\frac{z_3 - x_0}{x_0}\right|. \quad (119)$$

The pattern of relative solution errors displayed in Figure 22a is typical for any real x_0 value. Compared to the errors in 22b, which are calculated without round-off error mitigation, the errors in 22a calculated with mitigation are, for the most part, substantially less and produce a more regular plot pattern. This plot pattern for $\eta - 1 < 1$ in Figure 22a is due entirely to zero-guard processing in the Figure 9 cubic-equation algorithm as now explained.

The far-left portion of Figure 22a is labeled Special Case 2. In this region where, $\eta - 1 < 1.3{\times}10^{-7}$, the $\eta - 1$ value is so small (η is so close to 1, x_0 is so close to x_A) that zero-guard processing resets q, r, and R to zero. With $q = r = 0$, the algorithm branches to Special Case 2, and calculates the three cubic-equation solutions as the same real value:

$z_1 = x_2 = x_3 = -a_2/3 = (x_A + 2x_0)/3$ instead of the true values $z_1 = x_A$, $x_2 = x_3 = x_0$.

In normalized form the calculated solutions are

$$\frac{z_1}{x_A} = \frac{1}{3\eta}(\eta + 2) \text{ instead of } 1, \qquad \frac{x_2}{x_0} = \frac{x_3}{x_0} = \frac{1}{3}(\eta + 2) \text{ instead of } 1.$$

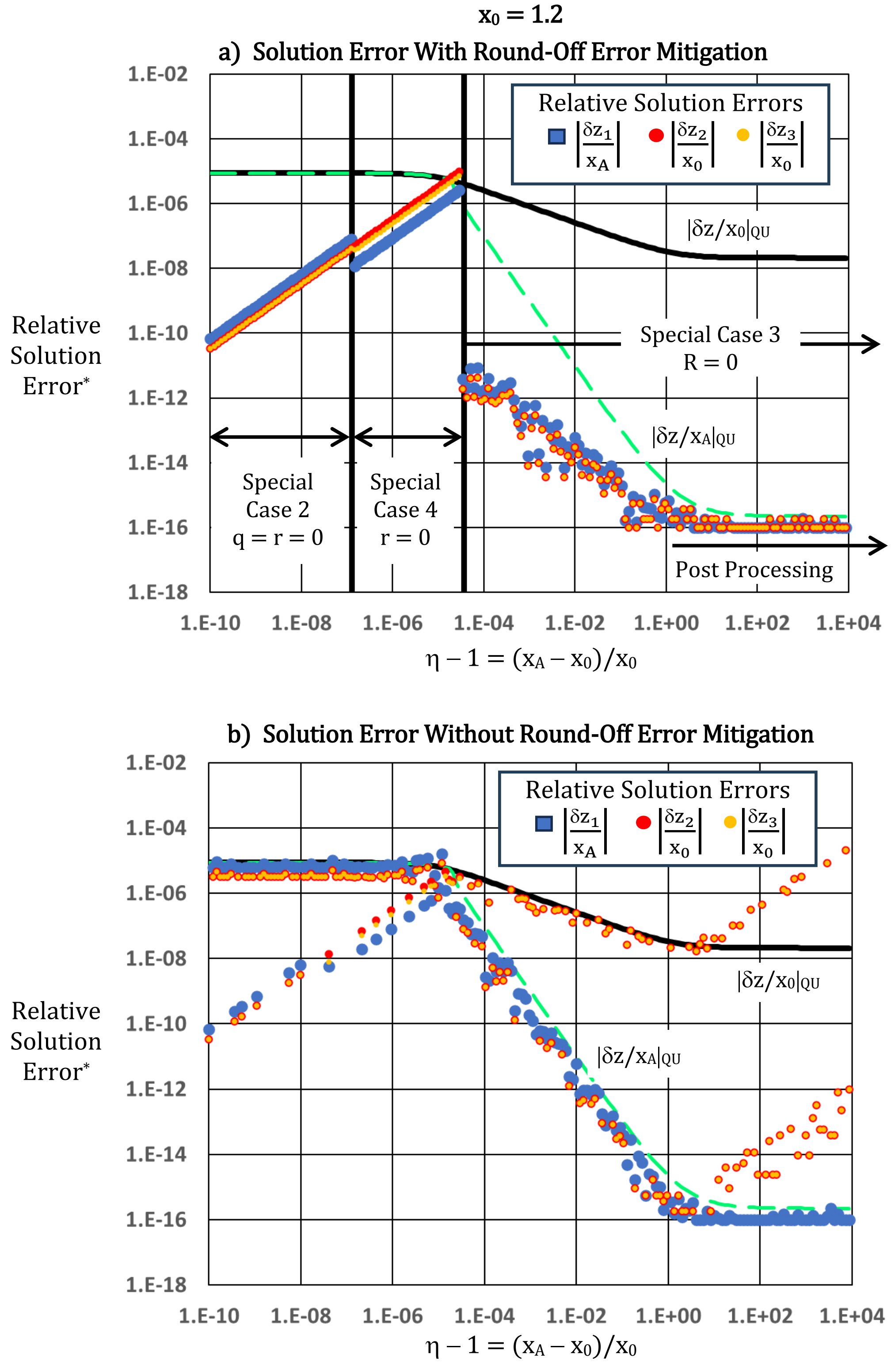
The relative errors become

$|\delta z_{1u}| \quad = |(\eta + 2)/(3\eta) - 1| = |-2(\eta-1)/(3\eta)| \approx 2(\eta-1)/3$ for small $\eta - 1$, and

$|\delta z_{2u}| = |\delta z_{3u}| = |(\eta + 2)/3 - 1| \quad = (\eta-1)/3$.

All three are proportional to $\eta-1$. The z_2 and z_3 relative errors equal each other; the z_1 relative error is twice their value.

Figure 22 Example Cubic-Equation Relative Solution Error – Multiplicity 2



*** Note: Any relative solution error value less than 10^{-16} is plotted as 10^{-16}.**

The next portion of Figure 22a is labeled Special Case 4 and includes $\eta - 1$ values from 1.3×10^{-7} to 3×10^{-5}. Here, the calculated value of q is negative (as it should be), but r is still reset to zero, and Special Case 4 applies. The Figure 9 algorithm calculates the cubic-equation solutions as three evenly-distributed real values:

$z_1 = -a_2/3 + s, \quad x_2 = -a_2/3, \quad x_3 = -a_2/3 - s \quad \text{where} \quad s = \sqrt{|3q|} \quad \text{and} \quad a_2 = -(x_A + 2x_0).$

Equations (5) and (107) show that q is

$$q = -(x_A - x_0)^2/9, \qquad \text{which implies that} \qquad s = \sqrt{3}\,(x_A - x_0)/3.$$

The calculated solutions are

$$z_1 = \frac{1}{3}\left[x_A + 2x_0 + \sqrt{3}(x_A - x_0)\right] \quad \text{instead of} \quad x_A$$

$$x_2 = \frac{1}{3}\left[x_A + 2x_0\right] \quad \text{instead of} \quad x_0$$

$$x_3 = \frac{1}{3}\left[x_A + 2x_0 - \sqrt{3}(x_A - x_0)\right] \quad \text{instead of} \quad x_0$$

with $y_2 = 0$. The relative errors become

$$|\delta z_{1u}| = \left|\frac{1}{3\eta}\left[\eta + 2 + \sqrt{3}(\eta - 1)\right] - 1\right| = \frac{2 - \sqrt{3}}{3\eta}(\eta - 1) \approx 0.089\,(\eta - 1)$$

$$|\delta z_{2u}| = \left|\frac{1}{3}[\eta + 2] - 1\right| = \frac{1}{3}(\eta - 1) \approx 0.333\,(\eta - 1)$$

$$|\delta z_{3u}| = \left|\frac{1}{3}\left[\eta + 2 - \sqrt{3}(\eta - 1)\right] - 1\right| = \frac{\sqrt{3} - 1}{3}(\eta - 1) \approx 0.244\,(\eta - 1).$$

Again, all three relative solution errors are proportional to $\eta - 1$. In the expression above for $|\delta z_{1u}|$, the fraction $(2 - \sqrt{3})/(3\eta)$ is approximately equal to $(2 - \sqrt{3})/3$ because $\eta - 1$ is less than 10^{-4}, so η is very close to 1.

The maximum relative solution error occurs at $\eta - 1 = 3\times10^{-5}$, the right edge of the Special Case 4 region in Figure 22a. This maximum error value is $|\delta z_{2u}| \approx 1\times10^{-5}$, only 15% greater than the multiplicity 3 quantum uncertainty of $(3\varepsilon)^{1/3} \approx 8.73\times10^{-6}$.

More importantly, the solutions calculated with the mitigation design are very accurate when judged against the cubic-equation coefficients. The coefficients a_2 and a_1 generated from the calculated solutions using Equations (2) or (3) are identical to the algorithm input coefficients a_2 and a_1. The coefficient a_0 generated from the calculated solutions using Equations (2) or (3) reproduces the algorithm input a_0 accurate to 15 significant figures.

To demonstrate these facts, normalize by x_0 the formulas above for the calculated solutions z_1, x_2, and x_3:

$$z_{1u} \equiv z_1/x_0 = [\eta + 2 + \sqrt{3}(\eta - 1)]/3$$
$$x_{2u} \equiv x_2/x_0 = (\eta + 2)/3$$
$$x_{3u} \equiv x_3/x_0 = [\eta + 2 - \sqrt{3}(\eta - 1)]/3$$

where $y_{2u} = 0$. Use Equation (3) to calculate the corresponding cubic-equation coefficients:

$$a_{2u} = -(\eta + 2) \qquad a_{1u} = 2\eta + 1 \qquad a_{0u} = -\eta + \frac{2}{27}(\eta - 1)^3.$$

The results for a_{2u} and a_{1u} are the same as the normalized input coefficient values in Equation (112). The a_{0u} value differs from the one in Equation (112) by $\frac{2}{27}(\eta - 1)^3$. This difference, evaluated at $\eta - 1 = 3\times10^{-5}$ (the $\eta - 1$ corresponding to maximum solution error) is 2×10^{-15}. Thus, the calculated cubic-equation solutions reproduce the input coefficients a_2 and a_1 exactly and a_0 accurate to nearly 15 significant figures.

Relative solution error in Figure 22a instantly drops six orders of magnitude at $\eta - 1 = 3\times10^{-5}$. This sudden decrease occurs because q and r now have sufficient magnitude that zero-guard processing maintains their original calculated values. The processing correctly resets any non-zero calculated R to zero (Special Case 3), so calculated solutions reflect the multiplicity 2 condition. Relative solution error continues to drop to 10^{-16} or less as $\eta - 1$ increases to about 10. As noted at the bottom of the figure, any relative solution error less than 10^{-16} is plotted as 10^{-16}.

The mitigation design invokes post processing to recalculate the two near-miss solutions z_2 and z_3 for $\eta - 1$ greater than about 1.9. The recalculation maintains the very small relative solution errors near 10^{-16} as $\eta - 1$ grows large. The onset of recalculation near $\eta - 1 = 1.9$ corresponds to the value of $\zeta = 0.345$ in Figure 12. To show this correspondence, recall that $\eta \equiv x_A/x_0$ where $x_A > x_0$. The Figure 12 algorithm recalculates the two smaller-magnitude solutions near x_0 approximately when $|x_0| < \zeta|x_A|$, that is, when $|x_A/x_0| = x_A/x_0 = \eta > 1/\zeta$. Since $\zeta = 0.345$, the algorithm recalculates the two smaller solutions when $\eta > 1/\zeta = 1/0.345 \approx 2.9$, which implies $\eta - 1 > 1.9$.

By contrast, solution errors without post processing in Figure 22b grow larger due to magnitude-type error magnification as $\eta - 1$ increases above about 10.

X. CUBIC EQUATION ANALYSIS – MULTIPLICITY 2 NEAR MISS

This section extends the previous section's analysis to the multiplicity 2 near miss condition, where again the mitigation design provides good solution accuracy. The three cubic-equation solutions are now

$$z_1 = x_A, \quad z_2 = x_0 + \Delta z \quad \text{and} \quad z_3 = x_0 - \Delta z \quad \text{where} \quad x_A \geq x_0.$$

Quantity Δz is either a nonnegative real value Δx or a positive pure imaginary number $i\Delta y$. Relative quantum uncertainties for the near-miss solutions are calculated as a function of $\eta \equiv x_A/x_0$ and $\Delta z/x_0$. The relative zero-guard range $|\Delta z/z_0|_{ZG}$ is a function η, but can change dramatically if post processing recalculates the two near-miss solutions when $|x_0| < |x_A|$. The recalculation occurs approximately when $|x_0/x_A| = |1/\eta| < \zeta$ where ζ is the Figure 12 post-processing constant. With the proper choice of ζ value, post processing not only eliminates magnitude-type round-off error magnification, but it also controls the size of the zero-guard range and its induced error.

The value $\zeta = 0.345$ is selected to minimize relative coefficient error produced by zero-guard processing. With this ζ value, the relative coefficient error induced by zero-guard processing is a maximum of 3.3×10^{-15} for all three coefficients and the maximum ratio ZG/QU is 2.3.

Although the title of this section specifies multiplicity 2 near miss, we allow the combination $|(x_A - x_0)/x_0| << 1$ and $|\Delta z/x_0| << 1$, which is actually an alternate form of multiplicity 3 near miss. We also allow $\Delta z = 0$ for multiplicity 2.

The cubic polynomial for multiplicity 2 near miss is

$$p(z) = z^3 + a_2 z^2 + a_1 z + a_0 \tag{120}$$

$$p(z) = (z - z_1)(z - z_2)(z - z_3) = (z - x_A)(z - x_0 - \Delta z)(z - x_0 + \Delta z) \tag{121}$$

where

$$a_2 = -(z_1 + z_2 + z_3) \quad = \quad -(x_A + 2x_0) \tag{122}$$

$$a_1 = z_1z_2 + z_1z_3 + z_2z_3 \quad = \quad 2x_Ax_0 + x_0^2 - \Delta z^2 \tag{123}$$

$$a_0 = -z_1z_2z_3 \quad = \quad -x_A\,(x_0^2 - \Delta z^2). \tag{124}$$

Quantum Uncertainty for Real Δz

This subsection derives quantum uncertainty for real Δz, but it also plots quantum uncertainty and sample calculated solution error for both real and imaginary Δz.

The case of real $\Delta z = \Delta x \geq 0$ implies that p(z) has three real roots:

$$z_1 = x_A, \qquad z_2 = x_2 = x_0 + \Delta x, \qquad z_3 = x_3 = x_0 - \Delta x \quad \text{for real } \Delta z = \Delta x. \tag{125}$$

The real roots x_2 and x_3 are separated by the difference $2\Delta x$.

The equation $p(z) = 0$ holds for z equal to any of the three roots: x_A, x_2, or x_3. We first examine $z = x_3$, for which $p(x_3)$ is evaluated as the sum

$$p(x_3) \;=\; x_3^3 + a_2x_3^2 + a_1x_3 + a_0 \;=\; 0.$$

The magnitude of the sum's least significant bit is that of its greatest-magnitude term. The least-significant-bit value δp of the sum is therefore

$$\delta p_3 = \mathrm{MAX}(|x_3^3|, |a_2x_3^2|, |a_1x_3|, |a_0|)\ \varepsilon \;>\; 0. \tag{126}$$

This δp_3 is the magnitude of the range of p values that would be stored in the computer as $p = 0$.

To find the corresponding uncertainty $|\delta x_3|_{QU}$ in the root $z = x_3$, solve the equation $p(x_3 + \delta x_3) = -\delta p_3$ for δx_3. (The rationale for the negative sign in this expression will be explained shortly.) Use Equation (121) for p(z) with $z = x_3 + \delta x_3 = x_0 - \Delta x + \delta x_3$.

$$(x_3 + \delta x_3 - z_1)(x_3 + \delta x_3 - x_2)(\delta z_3) = (x_0 - \Delta x + \delta x_3 - x_A)(-2\Delta x + \delta x_3)\ (\delta x_3) = -\delta p_3$$

Simplify to arrive at the cubic equation in δx_3.

$$\delta x_3^3 - (x_A - x_0 + 3\Delta x)\delta x_3^2 + 2\Delta x(x_A - x_0 + \Delta x)\delta x_3 + \delta p_3 = 0 \tag{127}$$

We choose the negative sign in $p(x_3 + \delta x_3) = -\delta p_3$ to assure that δx_3 always has a negative real value. This is so because $x_A \geq x_0$, which implies that $p(x_0)$ is a local maximum and that the derivative $p'(x)$ is positive for $x < x_0$. The root $x_3 = x_0 - \Delta x < x_0$, so $p'(x_3) > 0$. Thus, $p(x_3 + \delta x_3) = -\delta p_3$ assures that $\delta x_3 < 0$ regardless of the magnitudes of Δx and δp_3.

The upper bound of Δx is $(x_A - x_0)/3$. This value corresponds to the root $x_2 = x_0 + \Delta x$ having a value midway between roots $x_3 = x_0 - \Delta x$ and $z_1 = x_A$. That is, $x_2 - x_3 = x_A - x_2$. We cannot use Δx any greater than $(x_A - x_0)/3$, for then x_2 would be closer to x_A than it is to x_3.

Normalize Equation (127) by x_3^3 to obtain a cubic equation in $\delta x_3/x_3$.

$$\left[\frac{\delta x_3}{x_3}\right]^3 - \frac{x_A - x_0 + 3\Delta x}{x_0 - \Delta x}\left[\frac{\delta x_3}{x_3}\right]^2 + \frac{2\Delta x(x_A - x_0 + \Delta x)}{(x_0 - \Delta x)^2}\left[\frac{\delta x_3}{x_3}\right] + \frac{\delta p_3}{(x_0 - \Delta x)^3} = 0 \tag{128}$$

It appears that the coefficients depend on three variables: x_A, x_0, and Δx. Note that δp_3 depends on these same three variables via Equations (126) and (122) to (124) where $x_3 = x_0 - \Delta x$ and $\Delta z = \Delta x$. Our normalization, however, allows us to reduce the three variables x_A, x_0, and Δx to only two. In the special case $x_0 = 0$, the coefficients depend only on x_A and Δx.

$$\left[\frac{\delta x_3}{x_3}\right]^3 + \frac{x_A + 3\Delta x}{\Delta x}\left[\frac{\delta x_3}{x_3}\right]^2 + \frac{2\Delta x(x_A + \Delta x)}{\Delta x^2}\left[\frac{\delta x_3}{x_3}\right] - \frac{\delta p_3}{\Delta x^3} = 0, \qquad x_0 = 0 \tag{129}$$

Otherwise, the coefficients of Equation (128) become functions of a sign function and the two normalized variables $\eta \equiv x_A/x_0$ and $\Delta u \equiv \Delta x/x_0$:

$$\left[\frac{\delta x_3}{x_3}\right]^3 - \frac{\eta - 1 + 3\Delta u}{1 - \Delta u}\left[\frac{\delta x_3}{x_3}\right]^2 + \frac{2\Delta u(\eta - 1 + \Delta u)}{(1 - \Delta u)^2}\left[\frac{\delta x_3}{x_3}\right] + \frac{\delta p_{3u}}{(1 - \Delta u)^3} = 0, \quad x_0 \neq 0 \quad (130)$$

where

$$\eta \equiv \frac{x_A}{x_0}, \qquad \Delta u \equiv \frac{\Delta x}{x_0}, \qquad \delta p_{3u} \equiv \frac{\delta p_3}{x_0^3}, \quad (131)$$

$$\delta p_{3u} = \text{sgn}(x_0)\,\text{MAX}[|(1-\Delta u)^3|, |(\eta+2)(1-\Delta u)^2|, |(2\eta+1-\Delta u^2)(1-\Delta u)|, |\eta(1-\Delta u^2)|]\,\varepsilon, \quad (132)$$

and

$$\text{sgn}(x_0) = \begin{cases} 1 & \text{if } x_0 > 0 \\ 0 & \text{if } x_0 = 0 \text{ (not used).} \\ -1 & \text{if } x_0 < 0 \end{cases} \quad (133)$$

For $x_0 \neq 0$, the relative quantum uncertainty $|\delta x_3/x_3|_{QU}$ for solution x_3 is the minimum absolute value of the three Equation (130) solutions. When $x_0 > 0$, $|\delta x_3/x_3|_{QU}$ is the absolute value of the only negative solution.

Case: $x_0 \neq 0$

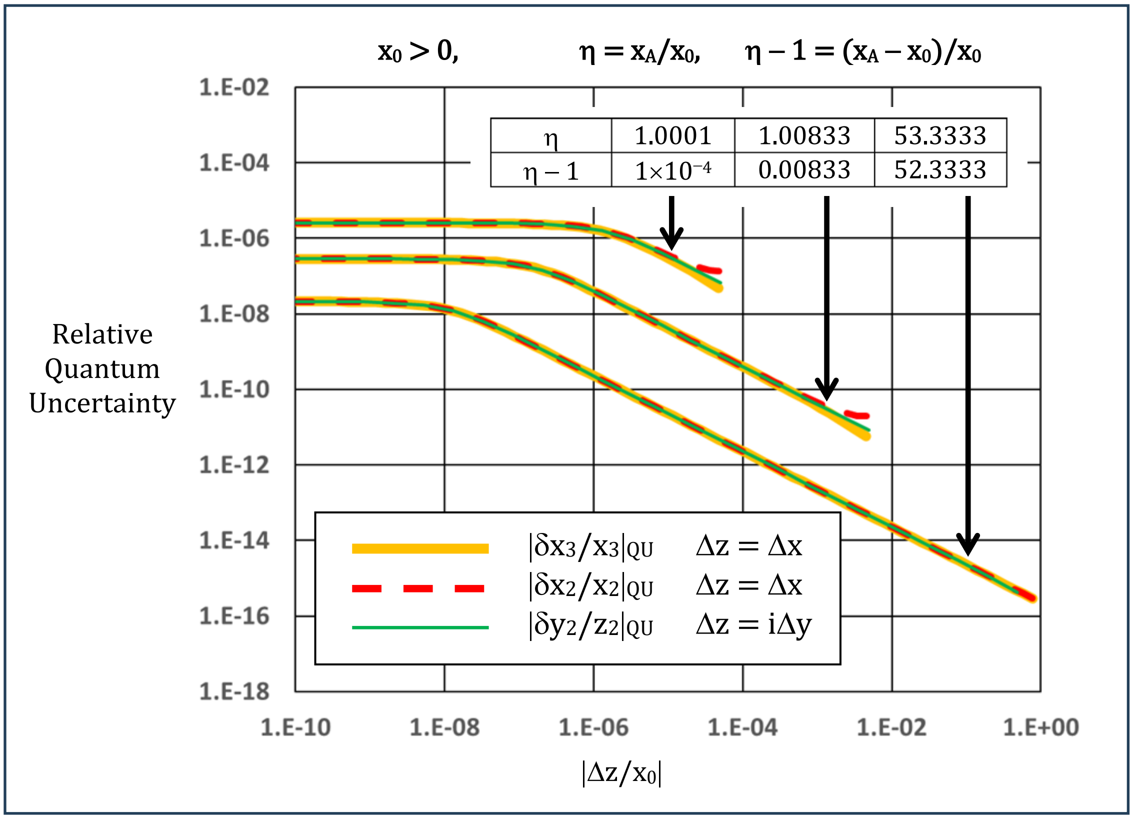
Figure 23 below applies Equations (130) through (133) to plot the relative quantum uncertainty $|\delta x_3/x_3|_{QU}$ versus $|\Delta u| = |\Delta z/x_0| = |\Delta x/x_0|$ for three representative values of $\eta - 1$ using the heavy, yellow curves. The uppermost curve corresponds to $\eta - 1 = 1\times10^{-4}$. At the smallest $|\Delta z/x_0|$ values, x_2 and x_3 are nearly equal, so that the $|\delta x_3/x_3|_{QU}$ value of about 2×10^{-6} in Figure 23 corresponds to this same value for the multiplicity 2 quantum uncertainty $|\delta z/x_0|_{QU}$ in Figure 21 (black and dashed yellow curves) at $\eta - 1 = 1\times10^{-4}$.

This same correspondence between Figures 21 and 23 applies to the other two $|\delta x_3/x_3|_{QU}$ curves in Figure 23. For the middle curve, $\eta - 1 = 0.00833$, and the maximum $|\delta x_3/x_3|_{QU}$ is about 2.8×10^{-7}. For the lowest curve, $\eta - 1 = 52.3333$, and the maximum $|\delta x_3/x_3|_{QU}$ is about 2.1×10^{-8}.

The relative quantum uncertainty $|\delta x_3/x_3|_{QU}$ in all three curves tends to decrease as the relative separation $|\Delta z/x_0|$ increases until the curve terminates. The upper two curves terminate at the point where the root $x_2 = x_0 + \Delta x$ is midway between roots $x_3 = x_0 - \Delta x$ and $z_1 = x_A$. That is, where $x_2 - x_3 = x_A - x_2$, which implies $\Delta x/x_0 = (\eta - 1)/3$. The lowest curve terminates at $\Delta x/x_0 = 1$ where $x_3 = 0$.

In addition to $|\delta x_3/x_3|_{QU}$, Figure 23 plots the relative quantum uncertainties $|\delta x_2/x_2|_{QU}$ (dashed, red curve) and $|\delta y_2/z_2|_{QU}$ (thin green curve). Both of these values are nearly equal to $|\delta x_3/x_3|_{QU}$. The value $|\delta x_2/x_2|_{QU}$ is the relative quantum uncertainty for root x_2 plotted versus real $\Delta z = \Delta x$. Notice how the two upper dashed red curves tend to level out at the curve termination where root x_2 is midway between roots x_3 and $z_1 = x_A$. The derivation of $|\delta x_2/x_2|_{QU}$ is given presently and is similar to that of $|\delta x_3/x_3|_{QU}$. The value $|\delta y_2/z_2|_{QU}$, whose derivation is given later, applies when $\Delta z = i\Delta y$ is imaginary. See Equations (146) to (148) below.

Figure 23 Cubic-Equation Quantum Uncertainty for Multiplicity 2 Near Miss



The derivation of $|\delta x_2/x_2|_{QU}$ starts by using Equation (121) for p(z) to solve the equation $p(x_2 + \delta x_2) = -\delta p_2$ for δx_2 where

$$\delta p_2 = \mathrm{MAX}(|x_2^3|, |a_2 x_2^2|, |a_1 x_2|, |a_0|)\,\varepsilon \ > \ 0. \qquad (134)$$

The result is the following cubic equation for δx_2.

$$\delta x_2^3 - (x_A - x_0 - 3\Delta x)\delta x_2^2 - 2\Delta x(x_A - x_0 - \Delta x)\delta x_2 + \delta p_2 = 0. \qquad (135)$$

Comparing Equations (134) and (135) to (126) and (127), we see that δp_2, x_2, and δx_2 replace δp_3, x_3, and δx_3 and that $-\Delta x$ replaces Δx.

We choose the negative sign in $p(x_2 + \delta x_2) = -\delta p_2$ to assure that Equation (135) produces a proper positive real solution for δx_2. The value δp_2 in Equation (134) is positive. Given any Δx in the range $(0, (x_A - x_0)/3)$, both the quadratic term and linear term of Equation (135) are negative for positive δx_2. One or both of these terms dominate the cubic term depending on the value of Δx. Thus Equation (135) has the desired positive real solution for δx_2.

The cubic equations in $\delta x_2/x_2$ corresponding to Equations (128), (129), and (130) become

$$\left[\frac{\delta x_2}{x_2}\right]^3 - \frac{x_A - x_0 - 3\Delta x}{x_0 + \Delta x}\left[\frac{\delta x_2}{x_2}\right]^2 - \frac{2\Delta x(x_A - x_0 - \Delta x)}{(x_0 + \Delta x)^2}\left[\frac{\delta x_2}{x_2}\right] + \frac{\delta p_2}{(x_0 + \Delta x)^3} = 0 \qquad (136)$$

$$\left[\frac{\delta x_2}{x_2}\right]^3 - \frac{x_A - 3\Delta x}{\Delta x}\left[\frac{\delta x_2}{x_2}\right]^2 - \frac{2\Delta x(x_A - \Delta x)}{\Delta x^2}\left[\frac{\delta x_2}{x_2}\right] + \frac{\delta p_2}{\Delta x^3} = 0, \qquad x_0 = 0,\ \ \Delta z = \Delta x \qquad (137)$$

$$\left[\frac{\delta x_2}{x_2}\right]^3 - \frac{\eta - 1 - 3\Delta u}{1 + \Delta u}\left[\frac{\delta x_2}{x_2}\right]^2 - \frac{2\Delta u(\eta - 1 - \Delta u)}{(1 + \Delta u)^2}\left[\frac{\delta x_2}{x_2}\right] + \frac{\delta p_{2u}}{(1 + \Delta u)^3} = 0, \qquad x_0 \neq 0. \quad (138)$$

The value δp_{2u} corresponds to δp_{3u} in Equation (132):

$\delta p_{2u} = \mathrm{sgn}(x_0)\ \mathrm{MAX}[|(1+\Delta u)^3|, |(\eta+2)(1+\Delta u)^2|, |(2\eta+1-\Delta u^2)(1+\Delta u)|, |\eta(1-\Delta u^2)|\,]\ \varepsilon.$ (139)

The relative quantum uncertainty $|\delta x_2/x_2|_{QU}$ for solution x_2 at $x_0 \neq 0$ is the minimum absolute value of the three Equation (138) solutions. It is plotted as the dashed red curve versus $\Delta u = |\Delta z/x_0|$ in Figure (23) above. When $x_0 > 0$ and $\Delta x/x_0 \leq (\eta - 1)/3$, the solution of Equation (138) with the minimum absolute value is a positive solution. This corresponds to the positive solution δx_2 of Equation (135).

Figures 24, 25, and 26 below plot trial values of solution relative error versus $|\Delta z/x_0|$ for the three $\eta - 1$ values in Figure 23 using $x_0 = 1.2$. The x_A value for each of the three figures is $x_A = x_0\eta = 1.2\eta$.

Figure 24	$\eta - 1 = 52.3333$	$x_A = 64$
Figure 25	$\eta - 1 = 0.00833$	$x_A = 1.21$
Figure 26	$\eta - 1 = 1\times10^{-4}$	$x_A = 1.20012$

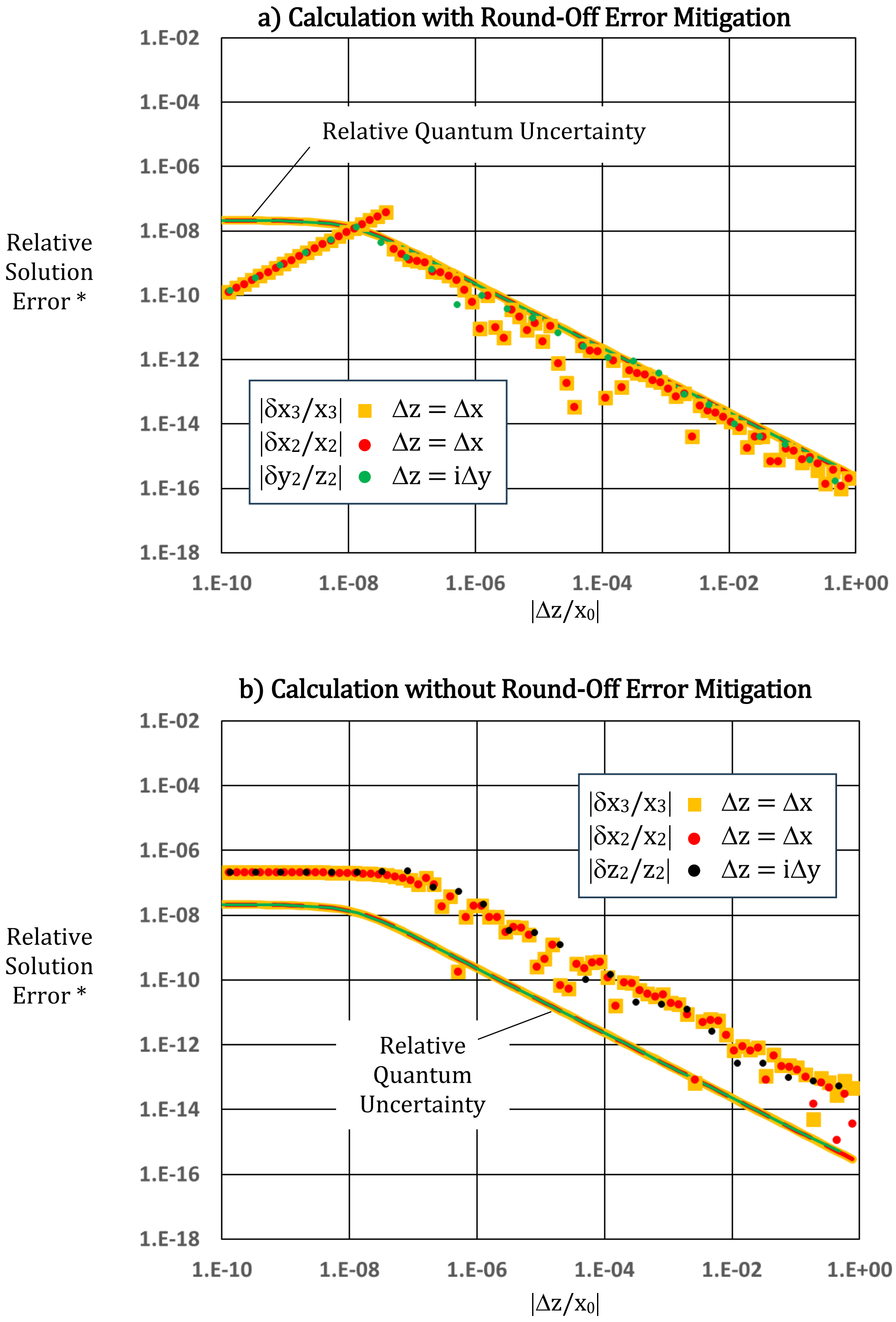
Figures 24a, 25a, and 26a show error for solutions calculated with the mitigation design: the Figure 9 cubic-equation algorithm, the Figure 12 cubic-equation post-processing algorithm, and the Figure 8 quadratic-equation algorithm. Figures 24b, 25b, and 26b show error for solutions calculated without the mitigation design using the Figure 1 algorithm.

For the relative errors $|\delta x_3/x_3|$ (yellow squares) and $|\delta x_2/x_2|$ (red circles), the separation $\Delta z = \Delta x$ is real, and the true x_2 and x_3 values are $x_2 = x_0 + \Delta x$ and $x_3 = x_0 - \Delta x$. The horizontal-axis variable is $|\Delta z/x_0| = |\Delta x/x_0| = |\Delta u|$.

Figures 24a, 25a, and 26a with the mitigation design also plot relative error $|\delta y_2/z_2|$ (green circles); Figures 24b, 25b, and 26b without the mitigation design also plot relative error $|\delta z_2/z_2|$ (black circles). The corresponding separation $\Delta z = i\Delta y$ is imaginary, and the true x_2 and x_3 values are $x_2 = x_0 + i\Delta y$ and $x_3 = x_0 - i\Delta y$. The horizontal-axis variable is $|\Delta z/x_0| = |\Delta y/x_0|$. The mitigation design produces solution error with a negligible real component, so only the imaginary error component is contained in $|\delta y_2/z_2|$ (green circles). Without the mitigation design, either the real or imaginary component may dominate the solution error, so $|\delta z_2/z_2|$ (black circles) includes both components.

Figure 24 Example Solution Error, Cubic Multiplicity 2 Near Miss for $\eta - 1 = 52.3333$

$x_0 = 1.2$, $x_A = 64$ $\eta = x_A/x_0$, $\eta - 1 = (x_A - x_0)/x_0$



a) Calculation with Round-Off Error Mitigation

b) Calculation without Round-Off Error Mitigation

* Note: Any relative solution error value less than 10^{-16} is plotted as 10^{-16}.

Figure 25 Example Solution Error, Cubic Multiplicity 2 Near Miss for η – 1 = 0.00833

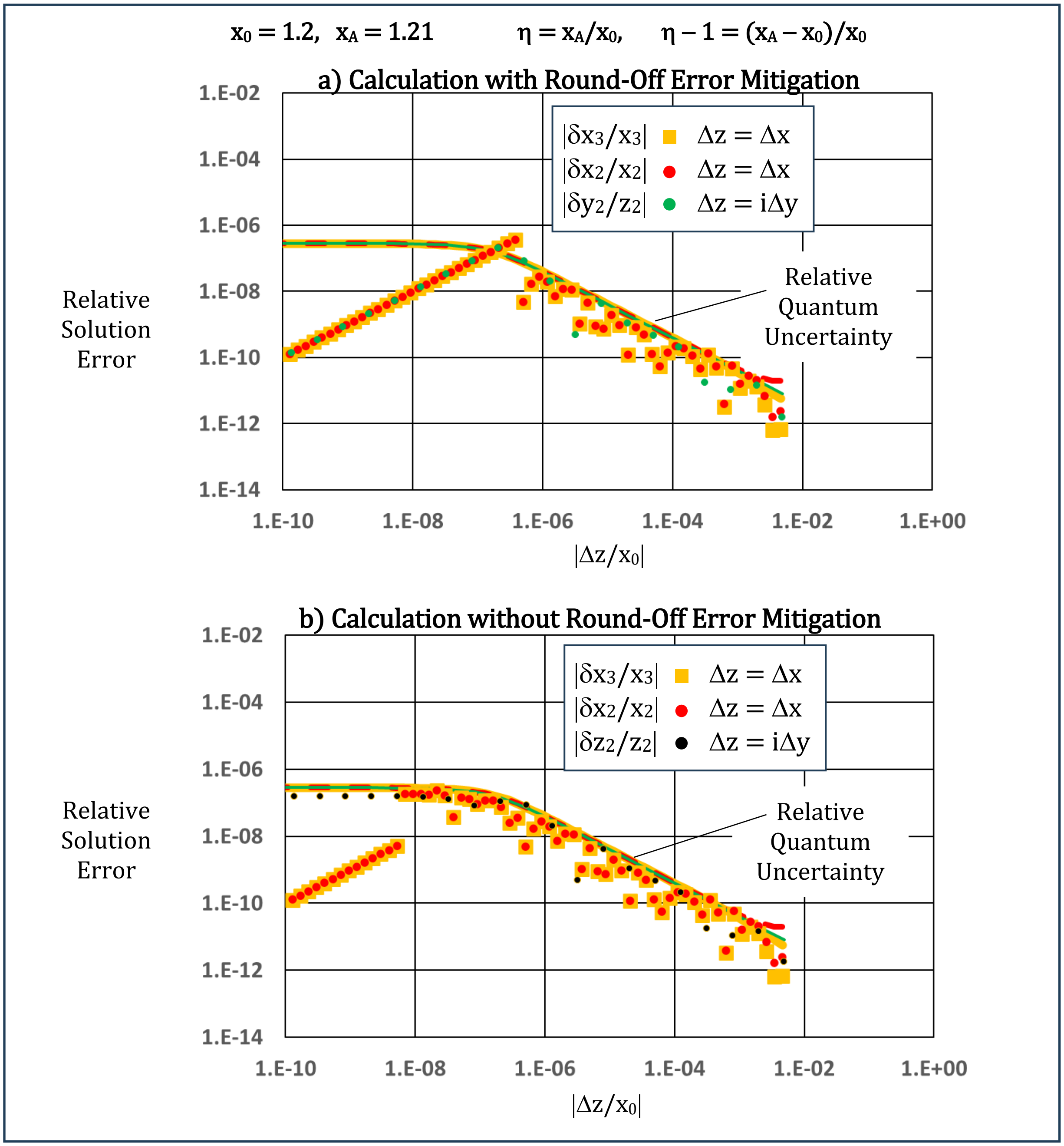
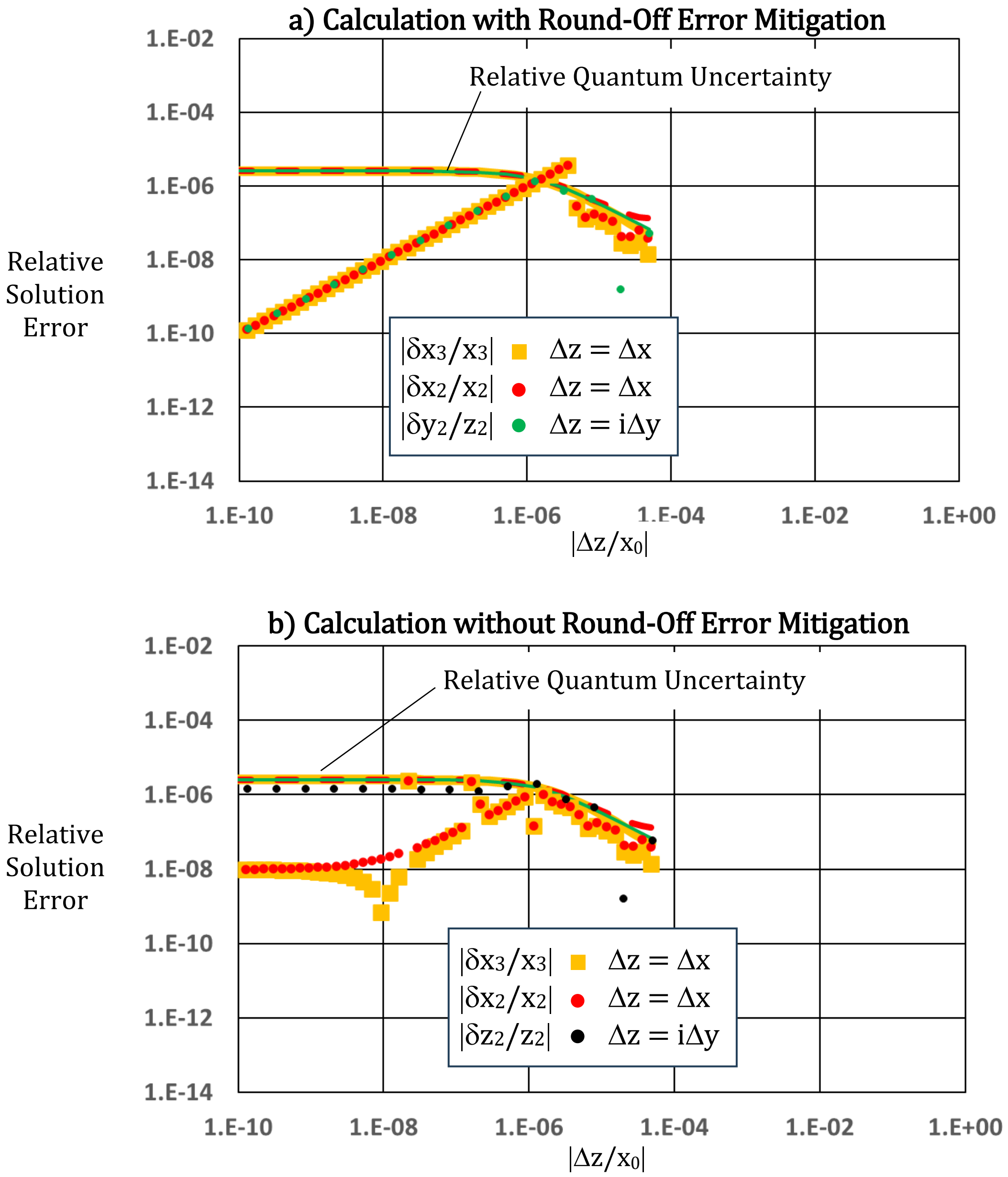


Figure 26 Example Solution Error, Cubic Multiplicity 2 Near Miss for $\eta - 1 = 1\times10^{-4}$

$x_0 = 1.2,\ \ x_A = 1.20012 \qquad \eta = x_A/x_0, \quad \eta - 1 = (x_A - x_0)/x_0$



a) Calculation with Round-Off Error Mitigation

b) Calculation without Round-Off Error Mitigation

As a reference, the figures also plot the appropriate relative quantum uncertainties from Figure 23.

The pattern of quantum uncertainty and error values in Figure 24a ($\eta - 1 = 52.3333$) with the mitigation design are typical for any multiplicity 2 near-miss cubic equation with a $|\eta - 1|$ value of 1 or greater, just as the double-root values of $|\delta z/x_0|_{QU}$ in Figure 21 change little for $|\eta - 1|$ of 1 or greater. It makes little difference whether x_0 is positive or negative or whether Δz is real or imaginary, the plots of quantum uncertainty and computed solution error with the mitigation design are all similar.

Zero-guard processing assures accurate calculated solutions for the multiplicity 2 condition $\Delta z = 0$ and $z_2 = z_3 = x_0$. As $\Delta z/x_0$ increases up to multiplicity the relative zero-guard range $|\Delta z/z_0|_{ZG}$ (in this case 3.82×10^{-8}), the calculated z_2 and z_3 values remain equal to each other. The plotted relative error values increase in proportion with $\Delta z/x_0$. The peak relative error becomes the relative zero-guard range $|\Delta z/z_0|_{ZG}$. That value in this case is 3.82×10^{-8}, about 1.8 times the maximum quantum uncertainty at $\Delta z = 0$. However, this measure of relative error assumes that we have a priori knowledge of the true solutions. The only true values available in actual practice are the cubic-equation coefficients.

To evaluate solution accuracy against the input cubic-equation coefficients a_2, a_1, and a_0, we use the calculated solution values z_{1C}, $z_{2C} = x_{2C} + iy_{2C}$, and $z_{3C} = x_{3C} - iy_{2C}$ and Equations (3) to calculate the check coefficients a_{2C}, a_{1C}, and a_{0C}. Solution accuracy can then be judged from the relative coefficient errors defined in Equation (82) and repeated here.

$$\delta a_{2u} \equiv \left|\frac{a_{2C} - a_2}{a_2}\right|, \quad \delta a_{1u} \equiv \left|\frac{a_{1C} - a_1}{a_1}\right|, \quad \delta a_{0u} \equiv \left|\frac{a_{0C} - a_0}{a_0}\right|. \qquad (82)$$

Against this measure, solutions calculated with the mitigation design in the examples above are very accurate. The relative coefficient errors are consistently on the order of 10^{-15} or less.

We shall return to this topic of solution error induced by the zero-guard range at the end of this section after we derive formulas for the zero-guard range. There we show that relative coefficient error induced by the zero-guard range is a maximum of 3.3×10^{-15} for all η with the post-processing constant $\zeta = 0.345$.

Figure 24a shows that the relative solution error abruptly drops an order of magnitude as $|\Delta z/x_0|$ surpasses the relative zero-guard range $|\Delta z/z_0|_{ZG} = 3.82\times10^{-8}$, and the calculated z_2 and z_3 values are no longer reset to x_0. Because z_2 and z_3 approximate x_0 and $|x_0/x_A| = |1/\eta| < \zeta = 0.345$, the Figure 12 cubic-equation post-processing algorithm uses the accurately-calculated $z_1 = x_A$ solution to recalculate the smaller-magnitude solutions z_2 and z_3. The post processing holds the relative solution errors to the quantum-uncertainty level or less.

If instead the cubic-equation solutions are calculated with the Figure 1 algorithm (no round-off error mitigation), then the z_2 and z_3 relative errors, shown in Figure 24b, consistently exceed the quantum uncertainty by an order of magnitude. These larger errors occur because the ratios $|z_1/z_2|$ and $|z_1/z_3|$ are close to $\eta = |x_A/x_0| = 53.3333$, a ratio large enough that magnitude-type error magnification exacerbates the multiplicity error magnification. The mitigation-design post processing eliminates the magnitude-type error magnification.

The particular values $x_0 = 1.2$ and $x_A = 64$ ($\eta = |x_A/x_0| = 53.3333$) were selected for this trial to show how large the calculated solution error can grow without the mitigation design. Any combination of x_0 and x_A values such that $|\eta - 1| \geq 1$ produces an error plot similar to Figure 24a when mitigation is used. Solution error is much more variable without mitigation. Sometimes (for example $x_0 = 1.2$, $x_A = 60 \Rightarrow \eta = 50$) calculation without mitigation produces accurate solutions for the multiplicity condition ($\Delta x = 0$). Then the error plot at small $\Delta z/x_0$ is the same as that with mitigation. The values $x_0 = 1.2$ and $x_A = 64$ avoid this situation, and the ratio x_A/x_0 is great enough to produce significant magnitude-type error magnification.

Figures 25 and 26 above plot calculated-solution relative error versus $\Delta z/x_0$ for the two small $\eta - 1$ values of Figure 23. Again $x_0 = 1.2$. Figure 25 has $x_A = 1.21$ and $\eta - 1 = 0.00833$; Figure 26 has $x_A = 1.20012$ and $\eta - 1 = 1\times10^{-4}$. The errors in Figures 25a and 26a, where solutions are calculated with the mitigation design, show the same regular plot pattern as in Figure 24a with its large $\eta - 1$. The errors in Figures 25b and 26b, where solutions are calculated without the mitigation design, have plot patterns that are less regular, but the errors do not exceed the quantum uncertainty. The small $\eta - 1$ values imply that the 0. three cubic-equation solutions are close in value so that there is no magnitude-type error magnification. The $\eta - 1$ is so small and the three cubic-equation solutions are so close in value that we could properly label Figures 25 and 26 as multiplicity 3 near miss. The major benefit of the mitigation design at these small $\eta - 1$ values is its consistent accurate solutions at the true multiplicity condition $\Delta z = 0$.

Case: $x_0 = 0$

We now return to Equations (129) and (137) to show that the relative quantum uncertainties for x_3 and x_2 are very small, on the order ε, for $x_0 = 0$ and real $\Delta z = \Delta x$. The mitigation design produces solution relative error that is likewise small, but error grows very large without mitigation. The case of $x_0 = 0$ and imaginary $\Delta z = i\Delta y$ is examined later.

The condition $x_0 = 0$, $\Delta z = \Delta x$ implies that $x_2 = -x_3 = \Delta x$. Equations (122) to (124) for the cubic-equation coefficients simplify to

$$a_2 = -x_A, \quad a_1 = -\Delta x^2, \quad a_0 = x_A\,\Delta x^2 \quad \text{for} \quad x_0 = 0, \Delta z = \Delta x.$$

The upper bound of Δx is limited by the restriction that $x_2 = \Delta x$ cannot exceed the midpoint between $x_3 = -\Delta x$ and $z_1 = x_A$:

$$x_2 = \Delta x \;\leq\; (x_A + x_3)/2 = (x_A - \Delta x)/2 \quad \Rightarrow \quad 0 < \Delta x \;\leq\; x_A/3.$$

From the foregoing, Equations (126) for δp_3 and (134) for δp_2 simplify to

$$\delta p_3 = \delta p_2 = x_A \Delta x^2 \varepsilon, \text{ and}$$

Equations (129) for $\delta x_3/x_3$ and (137) for $\delta x_2/x_2$ become

$$\left[\frac{\delta x_3}{x_3}\right]^3 + \left(3 + \frac{x_A}{\Delta x}\right)\left[\frac{\delta x_3}{x_3}\right]^2 + 2\left(1 + \frac{x_A}{\Delta x}\right)\left[\frac{\delta x_3}{x_3}\right] - \frac{x_A}{\Delta x}\varepsilon = 0, \qquad x_0 = 0,\ \ \Delta z = \Delta x$$

$$\left[\frac{\delta x_2}{x_2}\right]^3 + \left(3 - \frac{x_A}{\Delta x}\right)\left[\frac{\delta x_2}{x_2}\right]^2 + 2\left(1 - \frac{x_A}{\Delta x}\right)\left[\frac{\delta x_2}{x_2}\right] - \frac{x_A}{\Delta x}\varepsilon = 0, \qquad x_0 = 0,\ \ \Delta z = \Delta x.$$

We could solve these two cubic equations for $\delta x_3/x_3$ and $\delta x_2/x_2$, but the cubic and quadratic terms are negligible and can be dropped. Solving the resultant linear equations produces the same absolute values $|\delta x_3/x_3|_{QU}$ and $|\delta x_2/x_2|_{QU}$ as do the cubic equations:

$$\left|\frac{\delta x_3}{x_3}\right|_{QU} = \frac{\varepsilon}{2(1 + \Delta x/x_A)} \qquad \left|\frac{\delta x_2}{x_2}\right|_{QU} = \frac{\varepsilon}{2(1 - \Delta x/x_A)} \qquad x_0 = 0,\ \ 0 < \Delta x/x_A \le 1/3.$$

When $\Delta x/x_A = 0$, then $x_3 = x_2 = 0$, and $|\delta x_3/x_3|_{QU} = |\delta x_2/x_2|_{QU} = \varepsilon/2$. When $\Delta x/x_A = 1/3$, its maximum value, then $|\delta x_3/x_3|_{QU} = 3\varepsilon/8$, and $|\delta x_2/x_2|_{QU} = 3\varepsilon/4$. Thus, the relative quantum uncertainties $|\delta x_3/x_3|_{QU}$ and $|\delta x_2/x_2|_{QU}$ are always less than ε for $x_0 = 0$.

Figure 27 below plots trial values of solution relative error versus $|\Delta z/x_A|$ for multiplicity 2 near miss with $x_0 = 0$. The figure shows error for calculation with the mitigation design and without mitigation. The mitigation design holds relative solution error to the order of the quantum uncertainty around 10^{-16}. Without mitigation, relative solution error increases as the reciprocal of $|\Delta z/x_A|$. The small-magnitude cubic-equation solutions x_2 and x_3 equal $\pm\Delta z$, whereas the large solution is $z_1 = x_A$. Thus $|x_3/x_A| = |x_2/x_A| = |\Delta z/x_A|$. Without the mitigation design's post processing, magnitude-type error magnification overwhelms the small $|\Delta z/x_A|$ values.

Quantum Uncertainty for Imaginary $\Delta z = i\Delta y$

This section derives the formula for the relative quantum uncertainty $|\delta y_2/z_2|_{QU}$ when Δz has the imaginary value $\Delta z = i\Delta y$. This is the same $|\delta y_2/z_2|_{QU}$ that appears as the green curve in Figures 23 through 26.

The cubic polynomial $p(z)$ has the three roots $z_1 = x_A \neq 0$, $z_2 = x_0 + i\Delta y$ and $z_3 = x_0 - i\Delta y$.

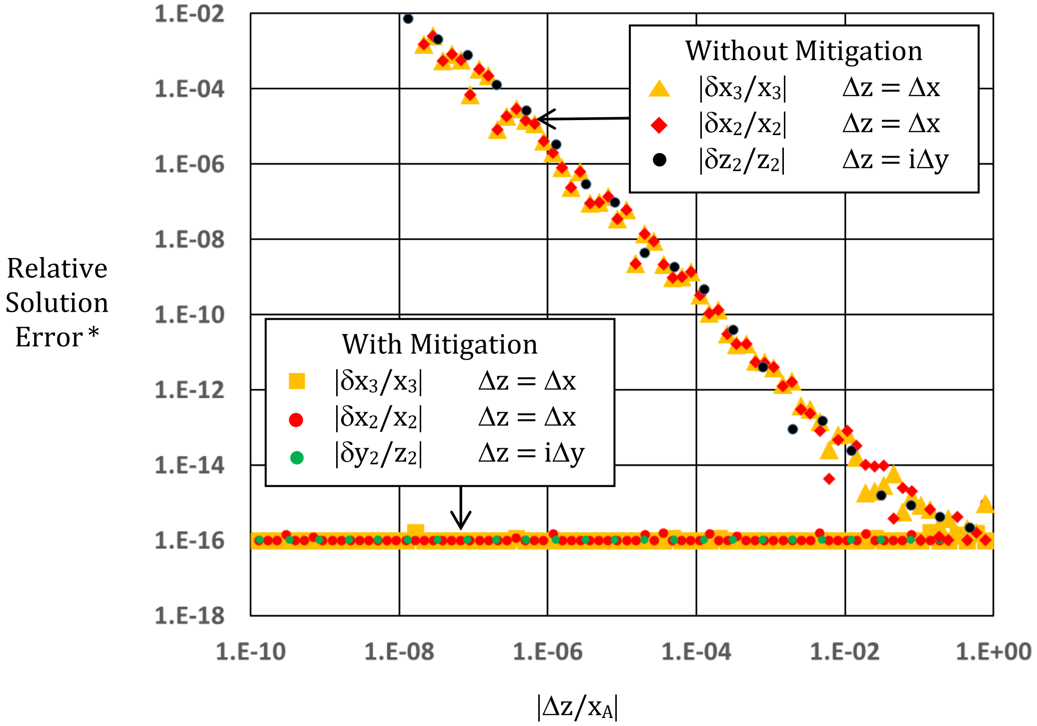
$$p(z) = z^3 + a_2 z^2 + a_1 z + a_0 = (z - x_A)(z - z_2)(z - z_3) = (z - x_A)(z - x_0 - i\Delta y)(z - x_0 + i\Delta y) \qquad (140)$$

Equation (3) gives the coefficients as

$$a_2 = -(x_A + 2x_0),\ \ a_1 = 2x_0 x_A + x_0^2 + \Delta y^2,\ \ a_0 = -x_A(x_0^2 + \Delta y^2) \ \text{ for } \ \Delta z = i\Delta y. \qquad (141)$$

Because the multiplicity 2 near-miss roots z_2 and z_3 are a complex conjugate pair, we need examine only one of them: $z_2 = x_0 + i\Delta y$. The total quantum uncertainty $|\delta z_2|_{QU} = |\delta x_2 + i\delta y_2|_{QU}$ is dominated by the imaginary component as we shall see. Thus, the relative value $|\delta z_2/z_2|_{QU}$ becomes $|\delta y_2/z_2|_{QU}$.

Figure 27 Example Solution Error, Cubic Multiplicity 2 Near Miss for $x_0 = 0$



*** Note: Any relative solution error value less than 10^{-16} is plotted as 10^{-16}.**

The derivation of $|\delta z_2/z_2|_{QU}$ is similar to that of $|\delta x_3/x_3|_{QU}$ above for real $\Delta z = \Delta x$. The cubic $p(z_2)$ for $z_2 = x_0 + i\Delta y$ is

$$p(z_2) = x_0^3 + i3x_0^2\Delta y - 3x_0\Delta y^2 - i\Delta y^3 + a_2[(x_0^2 - \Delta y^2) + i2x_0\Delta y] + a_1(x_0 + i\Delta y) + a_0\,.$$

The cubic is real and equal to zero, so $p(z_2)$ is evaluated as the sum of its real terms only.

$$p(z_2) = x_0^3 - 3x_0\Delta y^2 + a_2x_0^2 - a_2\Delta y^2 + a_1x_0 + a_0 = 0.$$

The least significant bit of this calculated $p(z_2)$ is the least significant bit of the term having the greatest absolute value. The value of this least significant bit is therefore

$$\delta p_y = \mathrm{MAX}(|x_0^3|, |3x_0\Delta y^2|, |a_2x_0^2|, |a_2\Delta y^2|, |a_1x_0|, |a_0|)\varepsilon\,. \qquad (142)$$

For z very close to z_2, δp_y is the range of $p(z)$ values that could be stored in the computer as $p(z) = 0$.

We find the quantum uncertainty $|\delta z_2|_{QU}$ in root $z_2 = x_0 + i\Delta y$ corresponding to δp_y by solving the equation $p(z_2 + \delta z_2) = \delta p_y$ for $\delta z_2 = \delta x_2 + i\delta y_2$. When the imaginary displacement $i\Delta y$ is 0, then root z_2 is the double root x_0, which occurs at the local maximum of cubic $p(z)$. Any real value of δz_2 produces a negative value of $p(z_2 + \delta z_2) = p(x_0 + \delta z_2)$. The relevant solution of $p(z_2 + \delta z_2) = p(x_0 + \delta z_2) = \delta p_y > 0$, in that case, cannot be real. To the contrary, error value $\delta z_2 = \delta x_2 + i\delta y_2$ is nearly pure imaginary for $\Delta y = 0$. The real component δx_2 is possibly significant only when Δy increases above 0. Even then, $|\delta x_2| << |\delta y_2|$, as we will show.

Solve the equation $p(z_2 + \delta z_2) = \delta p_y$ by using Equation (140) with $z = z_2 + \delta z_2 = x_0 + \delta x_2 + i(\Delta y + \delta y_2)$.

$$p(z_2 + \delta z_2) = (z_2 + \delta z_2 - x_A)(z_2 + \delta z_2 - z_2)(z_2 + \delta z_2 - z_3) = \delta p_y$$

$$[x_0 - x_A + \delta x_2 + i(\Delta y + \delta y_2)](\delta x_2 + i\delta y_2)[\delta x_2 + i(2\Delta y + \delta y_2)] - \delta p_y = 0$$

Expand and simplify the left side of this equation to produce real and imaginary components, each with several terms. Both the sum of real components and the sum of imaginary components must equal zero. The result is two equations for the two unknowns δx_2 and δy_2.

REAL

$$\begin{gathered}\delta x_2^3 - 3\delta x_2 \delta y_2^2 - (x_A - x_0)\delta x_2^2 - 6\Delta y \delta x_2 \delta y_2 + (x_A - x_0)\delta y_2^2 - 2\Delta y^2 \delta x_2 \\ +2(x_A - x_0)\Delta y \delta y_2 - \delta p_y = 0\end{gathered} \tag{143}$$

IMAGINARY

$$\begin{gathered}3\delta x_2^2 \delta y_2 - \delta y_2^3 + 3\Delta y \delta x_2^2 \\ -2(x_A - x_0)\delta x_2 \delta y_2 - 3\Delta y \delta y_2^2 - 2(x_A - x_0)\Delta y \delta x_2 - 2\Delta y^2 \delta y_2 = 0\end{gathered} \tag{144}$$

The following derivation and simplifying assumptions produce reasonably accurate solutions δx_2 and δy_2 to the above simultaneous equations.

Start with the IMAGINARY equation. Drop the first and third terms, which are quadratic in δx_2. Solve for δx_2.

$$\delta x_2 = -\left[\frac{2\Delta y^2 + 3\Delta y \delta y_2 + \delta y_2^2}{2(x_A - x_0)(\delta y_2 + \Delta y)}\right] \delta y_2$$

This equation shows that $|\delta x_2| << |\delta y_2|$ because $(x_A - x_0)$ and Δy are both positive, δy_2 is assumed positive, and $(x_A - x_0)$ is assumed much greater than Δy and δy_2.

Because $|\delta x_2| << |\delta y_2|$, the uncertainty value $\delta z_2 = \delta x_2 + i\delta y_2$ is dominated by its imaginary component, and all of the terms containing δx_2 in the REAL equation, Equation (143), are dropped. Write Equation (143) as a quadratic equation in δy_2.

$$\delta y_2^2 + 2\Delta y \delta y_2 - \frac{\delta p_y}{x_A - x_0} = 0\,.$$

Divide this equation by $|z_2|^2$ to normalize δy_2 by $|z_2| = \sqrt{x_0^2 + \Delta y^2}$.

$$\left[\frac{\delta y_2}{|z_2|}\right]^2 + 2\frac{\Delta y}{|z_2|}\left[\frac{\delta y_2}{|z_2|}\right] - \frac{\delta p_y}{|z_2|^2(x_A - x_0)} = 0 \tag{145}$$

We refine this quadratic equation separately for the two cases:
$x_0 \neq 0$, which produces $|\delta y_2/z_2|_{QU}$ in Figure (23) above and
$x_0 = 0$ for which $|\delta y_2/z_2|_{QU}$ is on the order of ε.

Case: $x_0 \neq 0$

Define the following:

$$\Delta v \equiv \Delta y/|x_0| = |\Delta z/x_0| \qquad \delta p_{uy} \equiv \delta p_y/x_0^3. \tag{146}$$

Equation (141) for the coefficients a_2, a_1, and a_0 and Equation (142) for δp_y give us δp_{uy} as

$$\delta p_{uy} = \mathrm{sgn}(x_0)\ \mathrm{MAX}(1{,}3\Delta v^2, |\eta + 2|, |(\eta + 2)\Delta v^2|, |2\eta + 1 + \Delta v^2|, |\eta(1 + \Delta v^2)|)\varepsilon \tag{147}$$

where $\eta = x_A/x_0$ from Equation (110). The function $\mathrm{sgn}(x_0)$ is the sign of x_0 given by (133). Equation (145) finally becomes

$$\left[\frac{\delta y_2}{|z_2|}\right]^2 + 2\frac{\Delta v}{\sqrt{1 + \Delta v^2}}\left[\frac{\delta y_2}{|z_2|}\right] - \frac{\delta p_{uy}}{(1 + \Delta v^2)(\eta - 1)} = 0 \quad \text{for} \quad \Delta z = i\Delta y \ \text{ and } \ x_0 \neq 0. \tag{148}$$

The relative quantum uncertainty $|\delta y_2/z_2|_{QU}$ for y_2 is the minimum positive value of the two Equation (148) solutions. It is plotted as the green curve versus $\Delta v = |\Delta z/x_0|$ in Figures 23 to 26.

Case: $x_0 = 0$

The cubic polynomial $p(z)$ for the case $x_0 = 0$ has the three roots $z_1 = x_A \neq 0$, $z_2 = i\Delta y$ and $z_3 = -\,i\Delta y$, $(\Delta y \neq 0)$. Thus $|z_2| = \Delta y$. Equation (140) becomes

$$p(z) = z^3+a_2z^2+a_1z+a_0 = (z-x_A)(z-z_2)(z-z_3) = (z - x_A)(z - i\Delta y)(z + i\Delta y).$$

The coefficients of $p(z)$ from Equation (141) are

$$a_2 = -\,x_A, \quad a_1 = \Delta y^2, \quad a_0 = -x_A\,\Delta y^2 \quad \text{for} \ \ \Delta z = i\Delta y \ \text{ and } \ x_0 = 0,$$

and δp_y in Equation (142) is

$$\delta p_y = \mathrm{MAX}(0{,}0{,}0, x_A\Delta y^2, 0, x_A\Delta y^2)\varepsilon = x_A\Delta y^2\varepsilon.$$

Quadratic Equation (145) becomes

$$\left[\frac{\delta y_2}{|z_2|}\right]^2 + 2\left[\frac{\delta y_2}{|z_2|}\right] - \varepsilon = 0.$$

The value of $\varepsilon \approx 2.22\times10^{-16}$ is so small that the two solutions are $\delta y_2/z_2 \approx -2$ and the relative quantum uncertainty

$$|\delta y_2/z_2|_{QU} \approx \varepsilon/2.$$

Zero-Guard Range and Lower Bound of Post-Processing Constant ζ

This subsection derives relative zero-guard range $|\Delta z/z_0|_{ZG}$ for the multiplicity 2 near-miss condition and shows that the Figure 12 post-processing constant ζ should have a minimum value of about 0.25. At this minimum ζ value, post processing holds the zero-guard range to 3.1 times the quantum uncertainty ($ZG/QU < 3.1$). The next (final) subsection shows that the selected $\zeta = 0.345$ value minimizes relative coefficient error produced by zero-guard processing and holds the ratio ZG/QU to 2.3.

The cubic-equation post-processing algorithm (Figure 12) uses the constant ζ as follows. Given the calculated solutions z_A, z_B, and z_C of a cubic equation such that $|z_C| \leq |z_B| \leq |z_A|$, post processing recalculates both z_B and z_C if $|z_B| < \zeta\ |z_A|$. It recalculates only z_C if $|z_C| < \zeta|z_A| \leq |z_B|$. The recalculation prevents contamination of the smaller-magnitude solution(s) from magnitude-type error magnification. The two smaller-magnitude solutions are most sensitive to magnitude-type error magnification when they are nearly equal to each other: multiplicity 2 near miss. We therefore adjust the value of ζ to accommodate the multiplicity 2 near-miss condition.

Unless explicitly stated otherwise, the term “post processing” in the following discussion refers specifically to post-processing recalculation of the two near-miss solutions $z_2 = x_2 = x_0 + \Delta x$ and $z_3 = x_3 = x_0 - \Delta x$ when they are also the two smallest-magnitude solutions. If $x_0 < 0$ and $x_0 < x_A < -x_0$ so that solution $z_1 = x_A$ has the smallest magnitude, then we will explicitly indicate post-processing recalculation of the simple small-magnitude solution z_1 as appropriate.

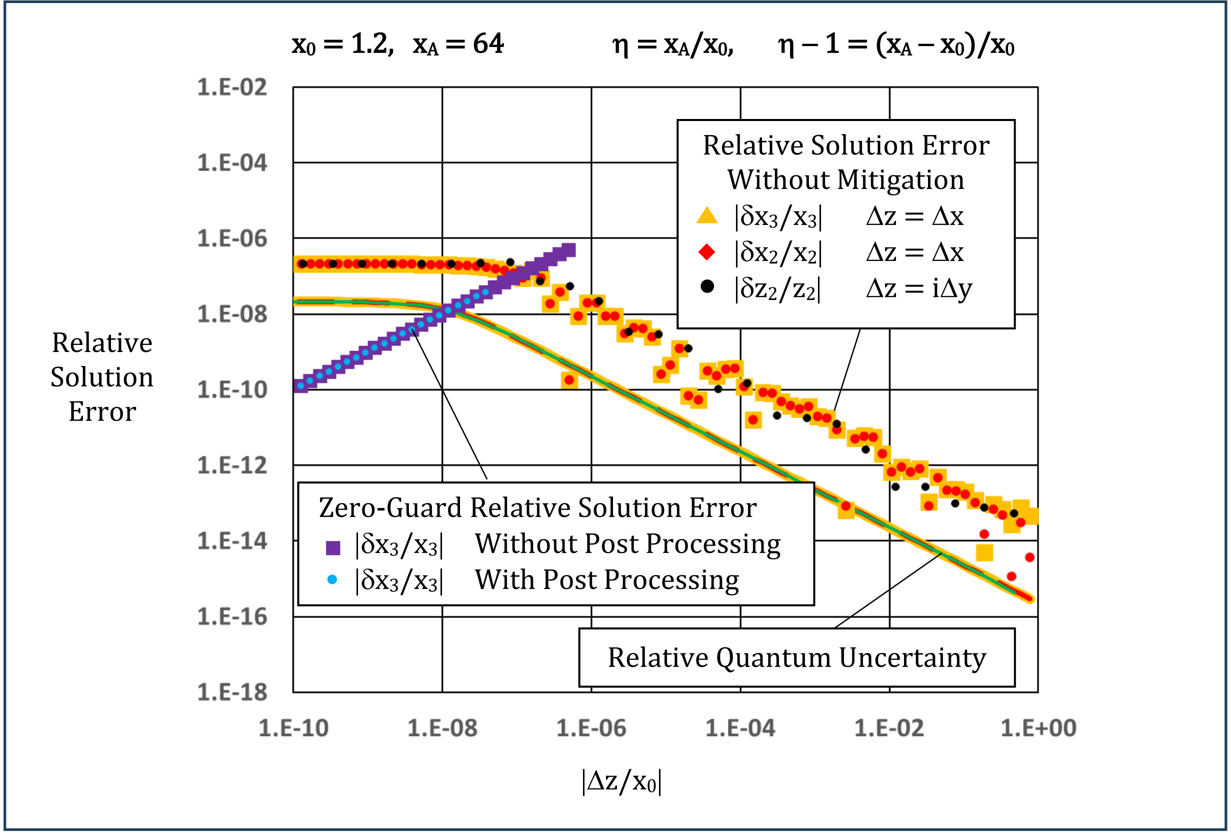
We have already demonstrated a primary benefit of using the relatively high value of $\zeta = 0.345$ in Figure 22 for the multiplicity 2 condition and in Figure 24 for multiplicity 2 near miss. Without error mitigation’s post processing, Figure 22b shows that relative solution error for the small-magnitude double solution $z_2 = z_3 = x_0$ (red dots and yellow dots) begins a steady increase as the ratio $\eta - 1$ on the horizontal axis climbs above 10. Here $\eta = x_A/x_0$ where x_A is the value of the large-magnitude solution z_1. The z_2 and z_3 worst-case error values at $\eta - 1 > 10$ are considerably greater than the quantum uncertainty of 2×10^{-8} shown as the solid black line. By contrast, post processing in Figure 22a holds the relative solution error to around 10^{-16} at $\eta - 1 > 10$.

Given our three cubic-equation solutions $z_1 = x_A$, $z_2 = x_0 + \Delta z$, and $z_3 = x_0 - \Delta z$, the Figure 12 post-processing algorithm recalculates the small-magnitude solutions z_2 and z_3 when $|z_2| < \zeta|z_1| = \zeta|x_A|$. The displacement magnitude $|\Delta z|$ is small relative to $|x_0|$ in our case, so that recalculation occurs approximately when $|x_0| < \zeta|x_A| \Leftrightarrow |x_0/x_A| = |1/\eta| < \zeta$. The value $\zeta = 0.345$ implies that post-processing recalculates z_2 and z_3 for $|\eta| > 1/0.345 \approx 2.9$. In Figure 22a, post processing recalculates z_2 and z_3 for $\eta - 1 > 1.9$.

Figure 24 compares relative solution error with and without error mitigation for the multiplicity 2 near-miss condition $\eta - 1 = 52.3333$. The solution error in Figure 24a with mitigation is markedly less than in Figure 24b without mitigation. At this high $\eta - 1$ value, mitigation’s post processing is responsible for holding down the solution error.

Post processing not only eliminates multiplicity error magnification, but it also controls solution error caused by a high zero-guard range as exemplified in Figure 28 below. As a reference, the figure duplicates the relative quantum uncertainty and solution error from Figure 24b in which solutions are calculated without round-off error mitigation. The value of $\eta = x_A/x_0$ is large: $\eta - 1 = 52.3333$. The purple and light-blue markers indicate solution error caused by zero-guard processing when it is turned on. Post processing is also turned on for the light-blue circles, just as in Figure 24a. Here the zero-guard range and corresponding relative solution error is limited to $3.82{\times}10^{-8}$. This value is 1.8 times the relative quantum uncertainty. The purple squares indicate zero-guard solution error when zero-guard processing is turned on but post processing is turned off. Now the zero-guard range and corresponding relative solution error climb to $5.6{\times}10^{-7}$, a value over 26 times greater than the relative quantum uncertainty.

Figure 28 Effect of Post Processing on Zero-Guard Solution Error, Cubic Multiplicity 2 Near Miss for $\eta - 1 = 52.3333$



At large values of $\eta = x_A/x_0$, the potential solution error caused by zero-guard processing becomes the driving need for post processing; we therefore use zero-guard range as a guide for selecting the minimum ζ value.

The three cubic-equation solutions for the multiplicity 2 near-miss condition are

$$z_1 = x_A \neq 0, \qquad z_2 = x_0 + \Delta z, \qquad z_3 = x_0 - \Delta z \tag{149}$$

where $\Delta z = \Delta x$ (real) or $\Delta z = i\Delta y$ (imaginary). Also, $x_A > x_0$, $\Delta x > 0$, and $\Delta y > 0$.

Without post processing, the zero-guard range $|\Delta z|_{ZG}$ is the maximum Δx or Δy value such that the zero-guard processing in the Figure 9 cubic-equation algorithm resets the calculated parameter $R = r^2 + q^3$ to zero and then calculates solutions z_2 and z_3 as the same real value. The corresponding relative zero-guard range is $|\Delta z/x_0|_{ZG}$. Because $|\Delta z/x_0|_{ZG} << 1$, we approximate its value from x_A and x_0 without regard for the displacement Δz in Equation (149). That is, $|\Delta z/x_0|_{ZG}$ is calculated as a property of the multiplicity 2 condition $z_1 = x_A$, $z_2 = z_3 = x_0$.

Post processing, which mitigates against excessive zero-guard range, has its own zero-guard range. Post processing recalculates z_2 and z_3 as solutions of a quadratic equation. The post-processing zero-guard range $|\Delta z|_{ZG}$ is the maximum Δx or Δy value such that the Figure 8 quadratic-equation algorithm resets the determinate D to zero and then calculates solutions z_2 and z_3 as the same real value.

Later, this section derives formulas to calculate the relative zero-guard range both without and with post processing. First, however, we present plots of the resulting zero-guard range values, which indicate that ζ should have a value of at least 0.25.

Figure 29 provides a global view of relative zero-guard range $|\Delta z/x_0|_{ZG}$ versus $|\eta - 1|$ across many orders of magnitude. As a reference, the figure also includes the double-root, relative quantum uncertainty $|\delta z/x_0|_{QU}$ from Figure 21 as the black curve.

Figure 29a for the case $x_0 > 0$ is straight forward because $x_A > x_0 > 0$, so $\eta = x_A/x_0 > 1$ and $\eta - 1 > 0$. Because the double root $x_0 = |x_0|$ is always less than the simple root $x_A = |x_A|$, post-processing recalculation of the smaller-magnitude roots x_2 and x_3 is possible for any value of $\eta - 1$. The green curve shows relative zero-guard range with post-processing recalculation, the blue curve without. The “With Post Processing” green curve is dashed where it predicts zero-guard range values less than the quantum uncertainty; such a prediction is unreliable in real-world computation.

The Post Processing green curve in Figure 29b for $x_0 < 0$ has a limited extent because post processing is possible only if $|x_0| < |x_A|$. With $x_0 < 0$, $|x_0|$ may be greater than, equal to, or less than $|x_A|$. The inequalities $x_0 < 0$ and $x_0 < x_A$ are given, so dividing the inequality $x_0 < x_A$ by $-x_0$ produces $-1 < -\eta = x_A/(-x_0)$ and $0 < 1 - \eta$. Thus, the horizontal axis of our log-log plot is $1 - \eta$. The post-processing recalculation requirement $|x_0| < |x_A|$ implies that $x_A > -x_0$, $-\eta > 1$, and $1 - \eta > 2$. Thus, the green curve for relative zero-guard range with post processing applies only to $1 - \eta > 2$.

In both Figures 29a and 29b, the blue curves show that zero-guard range without post processing is only a small multiple of quantum uncertainty for $|\eta - 1|$ less than about 1. As $|\eta - 1|$ becomes a bit greater than 1, zero-guard range grows many times greater than quantum uncertainty. This large ratio of zero-guard range to quantum uncertainty at $|\eta - 1|$ greater than 1 is a potential source of calculation error. We prevent that problem by employing post processing with the proper choice of constant ζ.

Figure 29 Relative Zero-Guard Range With And Without Post Processing versus |η – 1| -- Cubic Equation Multiplicity 2

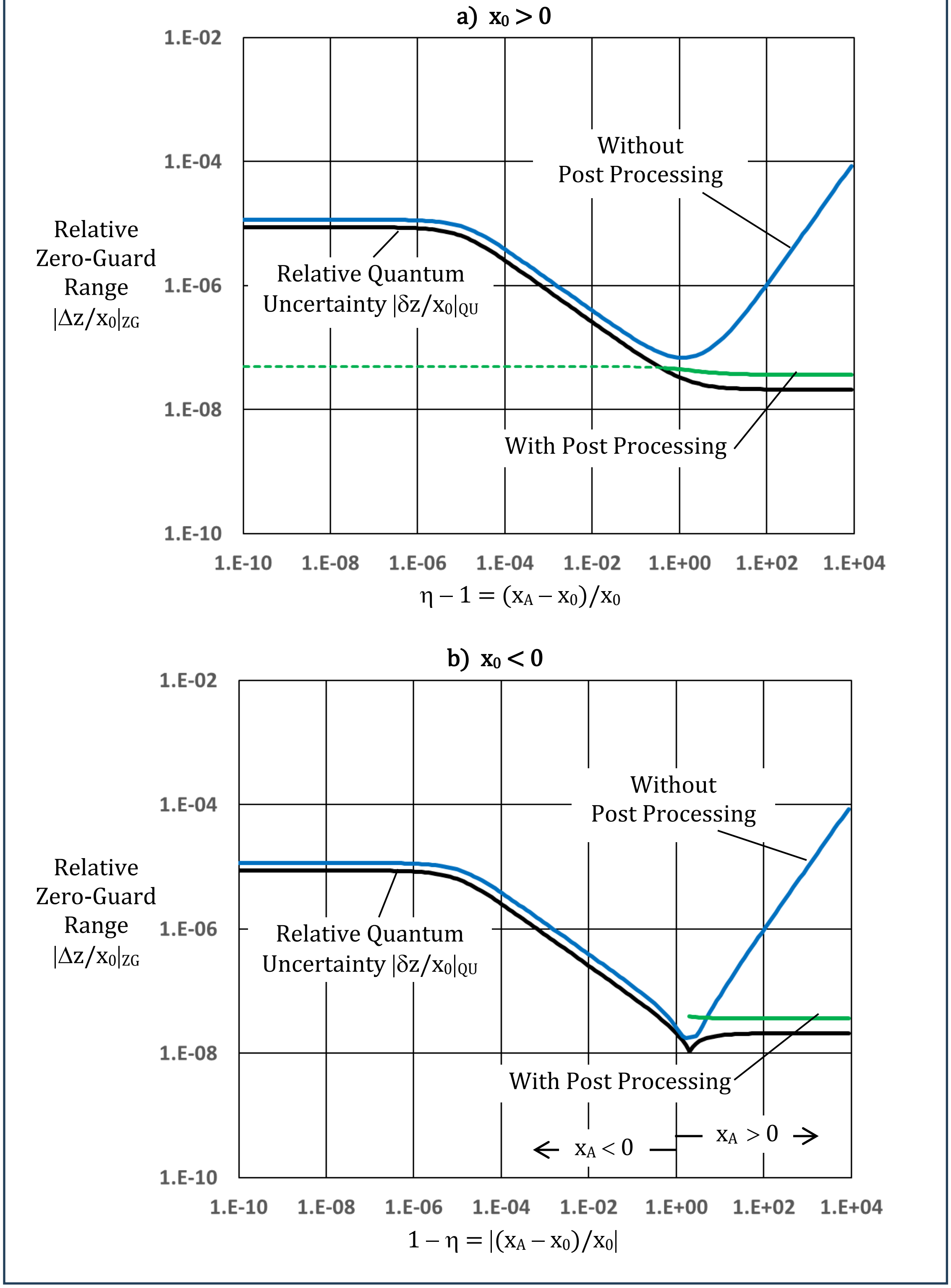


Figure 30 replots the relative zero-guard range from Figure 29 as a function of $|1/\eta|$ on a linear horizontal scale so that we can determine an appropriate minimum value for ζ. Post-processing recalculates the small-magnitude solutions z_2 and z_3 approximately when $|1/\eta| < \zeta$. Figure 30 shows that, without post processing (blue curve), the relative zero-guard range increases rapidly as $|1/\eta|$ falls below 0.25. Therefore ζ should have a value of at least 0.25 to avoid excessive zero-guard range and the resultant solution error.

The following two subsections derive formulas for relative zero-guard range $|\Delta z/x_0|_{ZG}$ without and with post processing. These formulas produce the plots in Figures 29 and 30 above.

Derivation of Zero-Guard Range without Post-Processing Recalculation

The zero-guard range for a cubic equation without post-processing recalculation is defined in terms of parameters R and $R_E \geq 0$ calculated in the Figure 9 cubic equation algorithm. If $R = 0$, then two solutions of the cubic equation equal the same real value $x_0 \neq 0$. If the calculated R value satisfies $|R| < R_E\,\varepsilon$, the algorithm resets R to zero and calculates two solutions as the same real value. The range of R values $(-R_E\,\varepsilon, R_E\,\varepsilon)$ about $R = 0$ corresponds to a range of solution values $(z_0 - \Delta z,\ z_0 + \Delta z)$ about z_0. We call this Δz value the zero-guard range $|\Delta z|_{ZG}$ about z_0. The relative zero-guard range is $|\Delta z/x_0|_{ZG} = |\Delta z|_{ZG}/|z_0|$.

We start with the multiplicity 2 cubic equation with solutions $z_1 = x_A$, $z_2 = z_3 = x_0$. The cubic polynomial and its coefficients are given in Equations (106) to (109) and repeated here.

$$p(z) = z^3 + a_2 z^2 + a_1 z + a_0 \;=\; (z - x_0)^2\,(z - x_A) \tag{150}$$

where

$$a_2 = -(x_A + 2x_0), \qquad a_1 = 2x_A x_0 + x_0^2, \qquad a_0 = -x_A\,x_0^2, \qquad x_0 \neq 0, \quad x_A \neq 0. \tag{151}$$

$$p(z) = 0 \qquad \text{for} \qquad z = x_0 \;\text{ or }\; z = x_A$$

$$x_A \;>\; x_0 \;\text{ by convention.}$$

The Inequality (109) has $x_A \geq x_0$, but the equality condition $x_A = x_0$ for multiplicity 3 does not apply here where the topic is multiplicity 2 and its near miss. We therefore apply only the strict inequality $x_A > x_0$.

The Figure 9 cubic equation algorithm calculates parameters a_{2E}, a_{1E}, a_{0E}, q, q_E, r, r_E, R, and R_E as follows.

$$a_{2E} = |a_2| \qquad\qquad a_{1E} = |a_1| \qquad\qquad a_{0E} = |a_1| \tag{152}$$

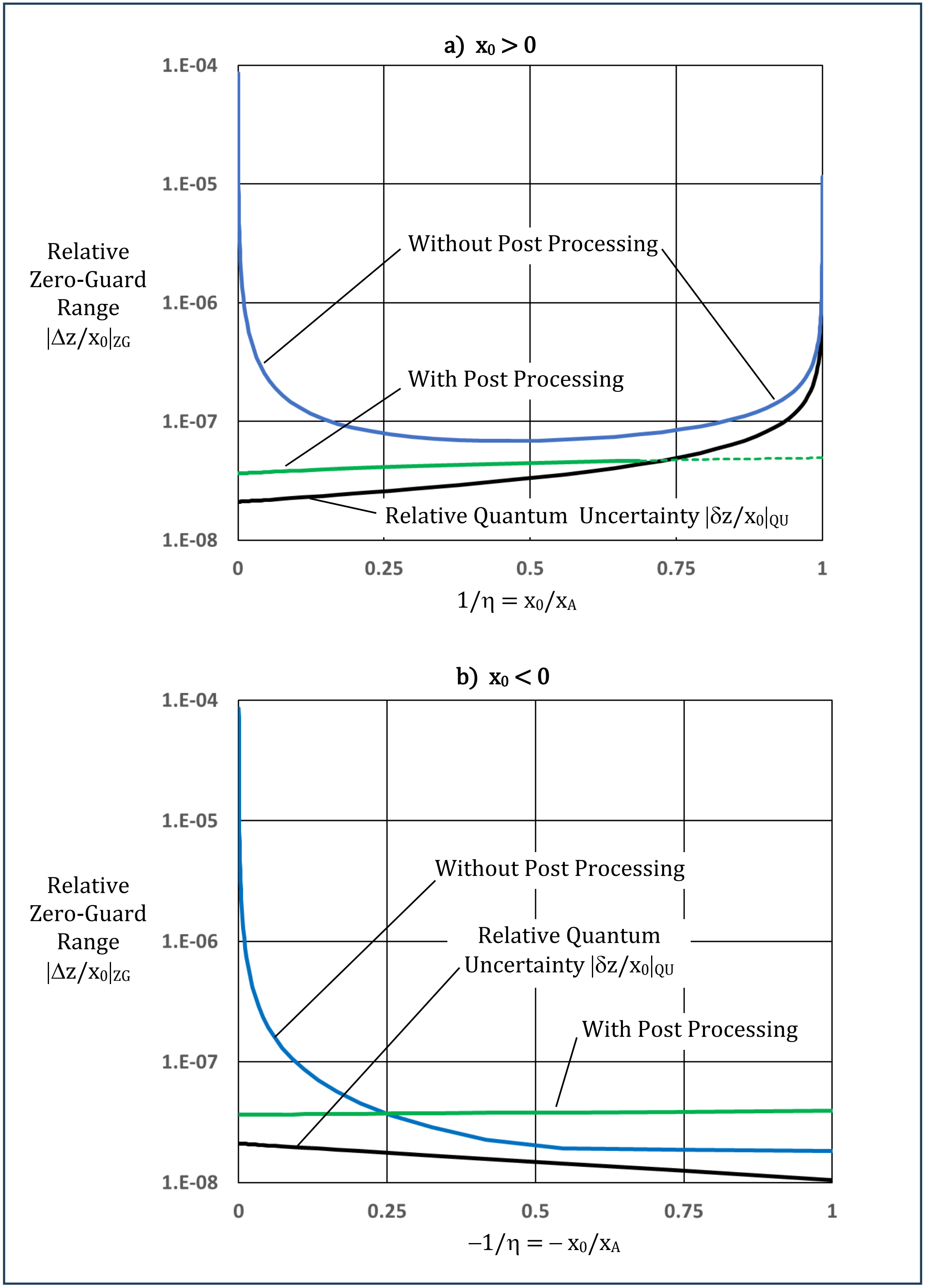
$$q = a_1/3 - a_2^2/9 \qquad\qquad r = (a_2 a_1 - 3a_0)/6 - a_2^3/27 \tag{153}$$

$$q_E = a_{1E}/3 + 2|a_2|\,a_{2E}/9 \qquad\qquad r_E = |a_1/6 - a_2^2\,/9|\,a_{2E} + |a_2|\,a_{1E}\,/6 + a_{0E}/2 \tag{154}$$

$$R = r^2 + \;q^3 \tag{155}$$

$$R_E = 2|r|\,r_E + 3q^2\,q_E \tag{156}$$

Figure 30 Relative Zero-Guard Range With And Without Post Processing versus $|1/\eta|$ -- Cubic Equation Multiplicity 2



a) $x_0 > 0$

b) $x_0 < 0$

Apply Equations (151) for coefficients a_2, a_1, and a_0 and Equation (152) for a_{2E}, a_{1E}, and a_{0E} to the formulas above to express q, r, q_E, r_E, and R_E as functions of x_A and x_0.

$$q = -(x_A - x_0)^2/9 < 0 \qquad\qquad r = (x_A - x_0)^3/27 > 0 \tag{157}$$

$$q_E = \frac{|2x_Ax_0 + x_0^2|}{3} + \frac{2(x_A + 2x_0)^2}{9} \tag{158}$$

$$r_E = \left|\frac{2x_Ax_0 + x_0^2}{6} - \frac{(x_A + 2x_0)^2}{9}\right| |x_A + 2x_0| + \frac{|x_A + 2x_0||2x_Ax_0 + x_0^2|}{6} + \frac{|x_Ax_0^2|}{2} \tag{159}$$

$$R_E = \frac{2}{27}|x_A - x_0|^3 r_E + \frac{1}{27}(x_A - x_0)^4 q_E \tag{160}$$

Note that Equation (155) for R and (157) for q and r produce R = 0:

$$R = r^2 + q^3 = \frac{(x_A - x_0)^6}{3^6} + \frac{(-1)^3(x_A - x_0)^6}{3^6} = 0$$

The convention $x_A > x_0$ implies that the double root $x_2 = x_3 = x_0$ of p(z) occurs at a local maximum of p(z) for real values x of z. We simplify this derivation by taking z to be a real value z = x so that p(z) = p(x). The cubic p(x), its derivative p′(x), and its second derivative p′′(x) satisfy $p(x_0) = p'(x_0) = 0$ and $p''(x_0) < 0$. Thus p(x) has a negative incremental value ($\Delta p < 0$) when x deviates from x_0 by a positive real increment Δx. That is, $\Delta p \equiv p(x_0 - \Delta x) < 0$ and $p(x_0 - \Delta x) - \Delta p = 0$. This means that the new cubic polynomial $p(x) - \Delta p$ has a root $x_0 - \Delta x$, which corresponds to root $x_3 = x_0$ of p(x). If the deviation $|x_A - x_0|$ is not too small, then $p(x) - \Delta p$ also has a root nearly equal to $x_0 + \Delta x$, which corresponds to root $x_2 = x_0$ of p(x).

We define Δx with respect to the least real root $x_0 - \Delta x$ rather $x_0 + \Delta x$ to assure that Δp is negative regardless of how small $|x_A - x_0|$ is. Suppose we had instead chosen $\Delta p \equiv p(x_0 + \Delta x) < 0$ and it happens that $x_A = x_0$. Then $p(x) = (x - x_0)^3$, and $\Delta p \equiv p(x_0 + \Delta x) = \Delta x^3 < 0$, which is impossible for our positive real increment Δx.

The increment Δx is the relative zero-guard range $|\Delta z|_{ZG}$ we seek if the parameter R for the cubic $p(x) - \Delta p$ is ΔR such that $|\Delta R| = R_E\,\varepsilon$. The derivation of the zero-guard range $\Delta x = |\Delta z|_{ZG}$ proceeds as follows.

Evaluate the cubic polynomial increment $\Delta p \equiv p(x_0 - \Delta x)$ using Equations (150) and (151).

$$\Delta p = p(x_0 - \Delta x) = (x_0 - \Delta x)^3 - (2x_0 + x_A)(x_0 - \Delta x)^2 + (2x_0x_A + x_0^2)(x_0 - \Delta x) - x_A\,x_0^2$$

$$\Delta p = -\Delta x^3 - (x_A - x_0)\Delta x^2$$

We have $x_A > x_0$ and $\Delta x > 0$ by definition, so $\Delta p < 0$, $\Delta p = -|\Delta p|$, and the last expression may be written

$$\Delta x^3 + (x_A - x_0)\Delta x^2 - |\Delta p| = 0. \tag{161}$$

The cubic polynomial p(x) – Δp has the same quadratic and linear coefficients a_2 and a_1 as does p(x), but the constant coefficient for p(x) – Δp is $a_0 + \Delta a_0$ where

$$\Delta a_0 = -\Delta p = |\Delta p| > 0.$$

Equation (153) for q and r shows that this incremental change to a_0 does not affect q but does produce a corresponding incremental change to r:

$$\Delta r = -\Delta a_0/2 = \Delta p/2 < 0.$$

The incremental change in R, Equation (155), becomes

$$\Delta R = 2r\Delta r = r\,\Delta p, \quad \text{which implies} \quad \Delta p = \Delta R/r < 0.$$

The value of r in Equation (157) is positive, so ΔR is negative. Set $\Delta R = -R_E\,\varepsilon$ so that Δp becomes

$$\Delta p = -R_E\,\varepsilon/r,$$

and Δx in Equation (161) becomes the relative zero-guard range $|\Delta z|_{ZG}$.

Substitute this result for Δp, Equation (160) for R_E, and Equation (157) for r into Equation (161) to obtain the following cubic equation in Δx.

$$\Delta x^3 + (x_A - x_0)\Delta x^2 - |2r_E + (x_A - x_0)q_E|\varepsilon = 0 \qquad (162)$$

Divide this equation through by x_0^3 to obtain a simplified cubic equation in the normalized increment $\Delta u \equiv \Delta x/x_0$. The equation is simplified because it contains a single parameter, $\eta \equiv x_A/x_0$, as will be demonstrated. The relative zero-guard range will then be $|\Delta z/x_0|_{ZG} = |\Delta u|$.

$$\Delta u^3 + (\eta - 1)\Delta u^2 - \frac{1}{x_0^3}[2r_E + (x_A - x_0)q_E]\varepsilon = 0 \qquad (163)$$

$$u \equiv z/x_0 \qquad \eta \equiv x_A/x_0$$

The normalized coefficients a_{2u}, a_{1u}, and a_{0u} are given in Equation (112) as

$$a_{2u} \equiv a_2/x_0 = -(\eta + 2) \qquad a_{1u} \equiv a_1/x_0^2 = 2\eta + 1 \qquad a_{0u} \equiv a_0/x_0^3 = -\eta.$$

The corresponding error size parameters are

$$a_{2uE} = |a_{2u}| = |\eta + 2| \qquad a_{1uE} = |a_{1u}| = |2\eta + 1| \qquad a_{0uE} = |a_{0u}| = |\eta|.$$

The normalized versions of q, r, q_E, and r_E in Equations (157) to (159) are

$$q_u \equiv q/x_0^2 = -(\eta - 1)^2/9 \qquad r_u \equiv r/x_0^3 = (\eta - 1)^3/27$$

$$q_{uE} \equiv \frac{q_E}{x_0^2} = \frac{|2\eta + 1|}{3} + \frac{2(\eta + 2)^2}{9} \qquad (164)$$

$$r_{uE} \equiv \frac{r_E}{|x_0^3|} = \left|\frac{2\eta + 1}{6} - \frac{(\eta + 2)^2}{9}\right| |\eta + 2| + \frac{|\eta + 2||2\eta + 1|}{6} + \frac{|\eta|}{2} \qquad (165)$$

Note that to maintain r_{uE} as a positive value, r_E is normalized by $|x_0^3|$ rather than x_0^3.

For the case $x_0 > 0$, we have $x_0^3 = |x_0^3|$, and Equation (163) becomes

$$\boxed{\Delta u^3 + (\eta - 1)\Delta u^2 - [2r_{uE} + (\eta - 1)q_{uE}]\varepsilon = 0 \qquad x_0 > 0, \quad \Delta u > 0. \qquad (166)}$$

CUBIC EQUATION FOR RELATIVE ZERO-GUARD RANGE $|\Delta z/x_0|_{ZG} = |\Delta u|$
WITHOUT POST PROCESSING FOR $x_0 > 0$

For the case $x_0 < 0$, we have $x_0^3 = -|x_0^3| = -x_0 x_0^2$, so the constant coefficient in Equation (163) must be positive. The coefficient becomes

$$+\frac{2r_E + (x_A - x_0)q_E}{|x_0^3|}\varepsilon = \left(2r_{uE} + \frac{x_A - x_0}{-x_0}q_{uE}\right)\varepsilon = [2r_{uE} + (1 - \eta)q_{uE}]\varepsilon.$$

Equation (163) becomes

$$\boxed{\Delta u^3 - (1 - \eta)\Delta u^2 + [2r_{uE} + (1 - \eta)q_{uE}]\varepsilon = 0 \qquad x_0 < 0, \quad \Delta u < 0. \qquad (167)}$$

CUBIC EQUATION FOR RELATIVE ZERO-GUARD RANGE $|\Delta z/x_0|_{ZG} = |\Delta u|$
WITHOUT POST PROCESSING FOR $x_0 < 0$

The calculation of relative zero-guard range $|\Delta z/x_0|_{ZG} = |\Delta u|$ (blue curves in Figures 29 and 30) is summarized as follows. By convention $x_A > x_0$ where x_A is the simple root and x_0 is the double root of the relevant multiplicity 2 cubic polynomial p(z). Calculate q_{uE} and r_{uE} from $\eta \equiv x_A/x_0$ using Equations (164) and (165). For the case $x_0 > 0$, solve Equation (166) for Δu. The relative zero-guard range $|\Delta z/x_0|_{ZG}$ is the positive real solution. For the case $x_0 < 0$, solve Equation (167) for Δu. The relative zero-guard range $|\Delta z/x_0|_{ZG}$ is the absolute value of the negative real solution.

Equations (166) and (167) show how relative zero-guard range $|\Delta z/x_0|_{ZG}$ without post processing must increase in proportion to $|\eta-1|$ for large $|\eta-1|$ in the blue curve of Figures 29 and 30. The quadratic coefficient in those two equations is $\eta-1$, so at large $|\eta-1|$, the cubic term becomes irrelevant. The equations become quadratic equations in Δu. Equations (164) and (165) show that both r_{uE} and $|\eta-1|q_{uE}$ increase as $|\eta^3|$ for large $|\eta|$. Equations (166) and (167) without the cubic term have the quadratic coefficient increase as $|\eta|$ and the constant coefficient increase as $|\eta^3|$, so Δu^2 must increase as η^2 and $|\Delta z/x_0|_{ZG} = |\Delta u|$ must increase as $|\eta|$.

Post processing avoids this troublesome growth of relative zero-guard range as now demonstrated.

Derivation of Zero-Guard Range with Post-Processing Recalculation

This derivation considers the case in which the real, multiplicity near-miss roots $x_2 = x_0 + \Delta x$ and $x_3 = x_0 - \Delta x$ of the cubic p(x) are calculated by the Figure 9 cubic-equation algorithm and are then both recalculated by the Figure 12 post-processing algorithm. The post-processing algorithm invokes the Figure 8 quadratic equation algorithm to recalculate $z_2 = x_2$ and $z_3 = x_3$ as solutions of a quadratic equation. The quadratic equation algorithm calculates the determinate D and its error magnitude D_E. The case D = 0 corresponds to the multiplicity condition $x_2 = x_3 = x_0 \Leftrightarrow \Delta x = 0$. If $|D| < D_E\,\varepsilon$, then the algorithm resets D to zero and calculates x_2 and x_3 as the same real value. The zero-guard range $|\Delta z|_{ZG}$ is the Δx value that produces the determinate value such that $|D| = D_E\,\varepsilon$. The relative zero-guard range is $|\Delta z/x_0|_{ZG} = |\Delta z|_{ZG} / |x_0|$.

The post-processing algorithm receives the following inputs from the cubic-equation algorithm: the cubic-equation coefficients a_2, a_1, a_0, the corresponding error size parameters a_{2E}, a_{1E}, a_{0E}, and the calculated real values z_1, x_2, x_3, and y_2 such that the three cubic-equation solutions are z_1, $z_2{=}x_2{+}\ iy_2$, $z_3{=}x_3{-}\ iy_2$. We calculate zero-guard range at the multiplicity condition $z_1 = x_A$, $z_2 = z_3 = x_2 = x_3 = x_0$, $y_2 = 0$, therefore a_2, a_1, a_0, a_{2E}, a_{1E}, and a_{0E} are given by Equations (151) and (152).

The post-processing algorithm recalculates x_2 and x_3 as solutions $x_2 = Z_1$ and $x_3 = Z_2$ of the quadratic equation $Z_n^2 + B\,Z_n + C = 0$. Post processing uses the accurately calculated large-magnitude solution $z_1 = x_A$ and coefficients a_0 and a_1 to calculate C and B as

$$C = -a_0/x_A \quad \text{and} \quad B = (C - a_1)/\,x_A.$$

The values of a_0 and a_1 in Equation (151) for our multiplicity condition are $a_0 = -x_A\,x_0^2$ and $a_1 = 2x_Ax_0 + x_0^2$, so C and B are calculated as $C = x_0^2$ and $B = -2x_0$.

The algorithm also calculates the error size parameters x_{AE}, C_E, and B_E corresponding to x_A, B, and C. The formulas are given in Equations (60), (59), and (61) respectively.

$$x_{AE} = \mathrm{MAX}(\,|x_A|, |a_2|\,) \tag{168}$$

$$C_E = \frac{1}{|x_A|}\,(a_{0E} + |C|x_{AE}) \tag{169}$$

$$B_E = \frac{1}{|x_A|}\left(a_{1E} + \frac{a_{0E}}{|x_A|} + \left|B + \frac{C}{x_A}\right|x_{AE}\right) \tag{170}$$

The post-processing algorithm then provides its values of B, C, B_E, and C_E to the Figure 8 quadratic equation algorithm to recalculate $x_2 = Z_1$ and $x_3 = Z_2$ as solutions of the quadratic equation $Z_n^2 + B\,Z_n + C = 0$. The quadratic equation algorithm calculates determinate D and its error size parameter D_E as

$$D = B^2 - 4C \quad \text{and} \quad D_E = 2|B|\,B_E + 4C_E \tag{171}$$

For the multiplicity condition $z_2 = z_3 = x_2 = x_3 = x_0$, we have

$$C = x_0^2 \ \text{ and } \ B = -2x_0, \tag{172}$$

so, $D = B^2 - 4C = 0$.

However, the determinate D is not zero for the near-miss condition $x_2 = x_0 + \Delta x$, $x_3 = x_0 - \Delta x$. Coefficient B given by $B = -(x_2 + x_3) = -2x_0$ is independent of Δx. Coefficient C and determinate D are $C = x_2x_3 = x_0^2 - \Delta x^2$ and $D = B^2 - 4C = 4\Delta x^2$.

The increment Δx is the zero-guard range $|\Delta z|_{ZG}$ when $4\Delta x^2 = D = D_E\,\varepsilon$, so

$$|\Delta z|_{ZG} = \sqrt{\frac{D_E\varepsilon}{4}} \quad \text{and} \quad |\Delta z/x_0|_{ZG} = \sqrt{\left(\frac{D_E}{4x_0^2}\right)\varepsilon} \tag{173}$$

The value $|\Delta z/x_0|_{ZG}$ is the relative zero-guard range, which we now derive and which is plotted as the green curves in Figures 29 and 30. Apply Equations (151) for a_0, (152) for a_{0E}, and (172) for C to Equation (169) for C_E to obtain

$$C_E = x_0^2\left(1 + \frac{x_{AE}}{|x_A|}\right).$$

Substitute this equation and $B = -2x_0$ into Equation (171) for D_E and divide through by $4x_0^2$.

$$\frac{D_E}{4x_0^2} = \frac{B_E}{|x_0|} + 1 + \frac{x_{AE}}{|x_A|}$$

Use Equation(170) for B_E, and apply Equations (151) and (152) for a_{1E} and a_{0E} and (172) for B and C. Finally apply the definition $\eta \equiv x_A/x_0$.

$$\frac{D_E}{4x_0^2} = \left|2 + \frac{1}{\eta}\right| + \left|\frac{1}{\eta}\right| + \left|\frac{1}{\eta} - 2\right|\frac{x_{AE}}{|x_A|} + 1 + \frac{x_{AE}}{|x_A|} \tag{174}$$

To determine $x_{AE}/|x_A|$, apply $a_2 = -(2x_0 + x_A)$ from Equation (151) to $x_{AE} = \mathrm{MAX}(|x_A|, |a_2|)$ in Equation (168).

$$x_{AE} = \mathrm{MAX}(|x_A|, |x_A + 2x_0|) \tag{175}$$

The case $x_0 > 0$ implies $x_A > x_0 > 0$ and $x_{AE} = |a_2| = x_A + 2x_0$.

$$\text{Then} \quad x_0 > 0 \;\Rightarrow\; \frac{x_{AE}}{|x_A|} = \frac{x_A + 2x_0}{x_A} = 1 + \frac{2}{\eta}.$$

For the case $x_0 < 0$, recall that $x_A > x_0$ and post-processing recalculation of z_2 and z_3 can occur only if $|x_A| > |x_0|$, that is, if $x_A > -x_0 > 0$. Thus, $|x_A + 2x_0| = |x_A - 2|x_0|| < |x_A| = x_A$, and by Equation (175), $x_{AE} = |x_A| = x_A$.

$$\text{Then} \quad x_A > -x_0 > 0 \;\Rightarrow\; \frac{x_{AE}}{|x_A|} = 1.$$

The final expressions for $D_E/(4x_0^2)$ in (174)become

$$\text{For}\ \ x_0 > 0\ \ \text{and}\ \ \eta > 1,\qquad \frac{D_E}{4x_0^2} = \left|2+\frac{1}{\eta}\right| + \left|\frac{1}{\eta}\right| + \left|\frac{1}{\eta}-2\right|\left|1+\frac{2}{\eta}\right| + 1 + \left|1+\frac{2}{\eta}\right| \qquad (176)$$

$$\text{For}\ \ x_0 < 0\ \ \text{and}\ \ \eta < -1,\qquad \frac{D_E}{4x_0^2} = \left|2+\frac{1}{\eta}\right| + \left|\frac{1}{\eta}\right| + \left|\frac{1}{\eta}-2\right| + 2. \qquad (177)$$

Use these equations and Equation (173), repeated below, to find relative zero-guard range $|\Delta z/x_0|_{ZG}$ with post-processing recalculation. These values are the green curves in Figures 29 and 30.

$$|\Delta z/x_0|_{ZG} = \sqrt{\left(\frac{D_E}{4x_0^2}\right)\varepsilon} \qquad (173)$$

Minimize Zero-Guard-Induced Relative Coefficient Error with ζ = 0.345

This section calculates the relative coefficient error induced by zero-guard processing and shows that such error is minimized by selecting the value 0.345 for the post-processing constant ζ. For a cubic equation with double solution $z_2 = z_3 = x_0$, zero-guard processing assures that the corresponding calculated values z_{2C} and z_{3C} are equal to each other regardless of computer round-off error. The zero-guard range $|\Delta z|_{ZG}$ is, however, a potential source of error. If the true solutions are $z_2 = x_0 + \Delta z$ and $z_3 = x_0 - \Delta z$ (multiplicity 2 near-miss) and $|\Delta z| < |\Delta z|_{ZG}$, then zero-guard processing will incorrectly produce calculated solutions z_{2C} and z_{3C} that are equal to each other. This section shows that any such zero-guard processing error is very small: the maximum relative coefficient error is 3.3×10^{-15}.

The relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} correspond to the cubic-equation coefficients a_2, a_1, and a_0 and are defined by Equation (82):

$$\delta a_{2u} \equiv \left|\frac{a_{2C}-a_2}{a_2}\right|,\quad \delta a_{1u} \equiv \left|\frac{a_{1C}-a_1}{a_1}\right|,\quad \delta a_{0u} \equiv \left|\frac{a_{0C}-a_0}{a_0}\right|. \qquad (82)$$

The check coefficients a_{2C}, a_{1C}, and a_{0C} in these formulas are produced from the calculated solutions z_{1C}, $z_{2C} = x_{2C} + iy_{2C}$, and $z_{3C} = x_{3C} - iy_{2C}$ using the check equations, Equation (3). Given cubic-equation true solutions

$$z_1 = x_A,\qquad z_2 = x_0 + \Delta z\qquad \text{and}\qquad z_3 = x_0 - \Delta z,$$

the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} are functions of $\eta \equiv x_A/x_0$, but they depend also on whether x_0 is positive or negative and on whether or not post processing recalculates any solutions.

By convention $x_A > x_0$, Equation (109).

The following paragraphs show that the greatest relative coefficient error is δa_{0u} when $x_0 > 0$ and δa_{1u} when $x_0 < 0$. Each of these errors has a maximum value of 3.3×10^{-15} with $\zeta = 0.345$. Lesser values of ζ increase the maximum δa_{0u}; greater values of ζ increase the maximum δa_{1u}. Thus assigning ζ the value 0.345 minimizes the greatest relative coefficient error to 3.3×10^{-15}.

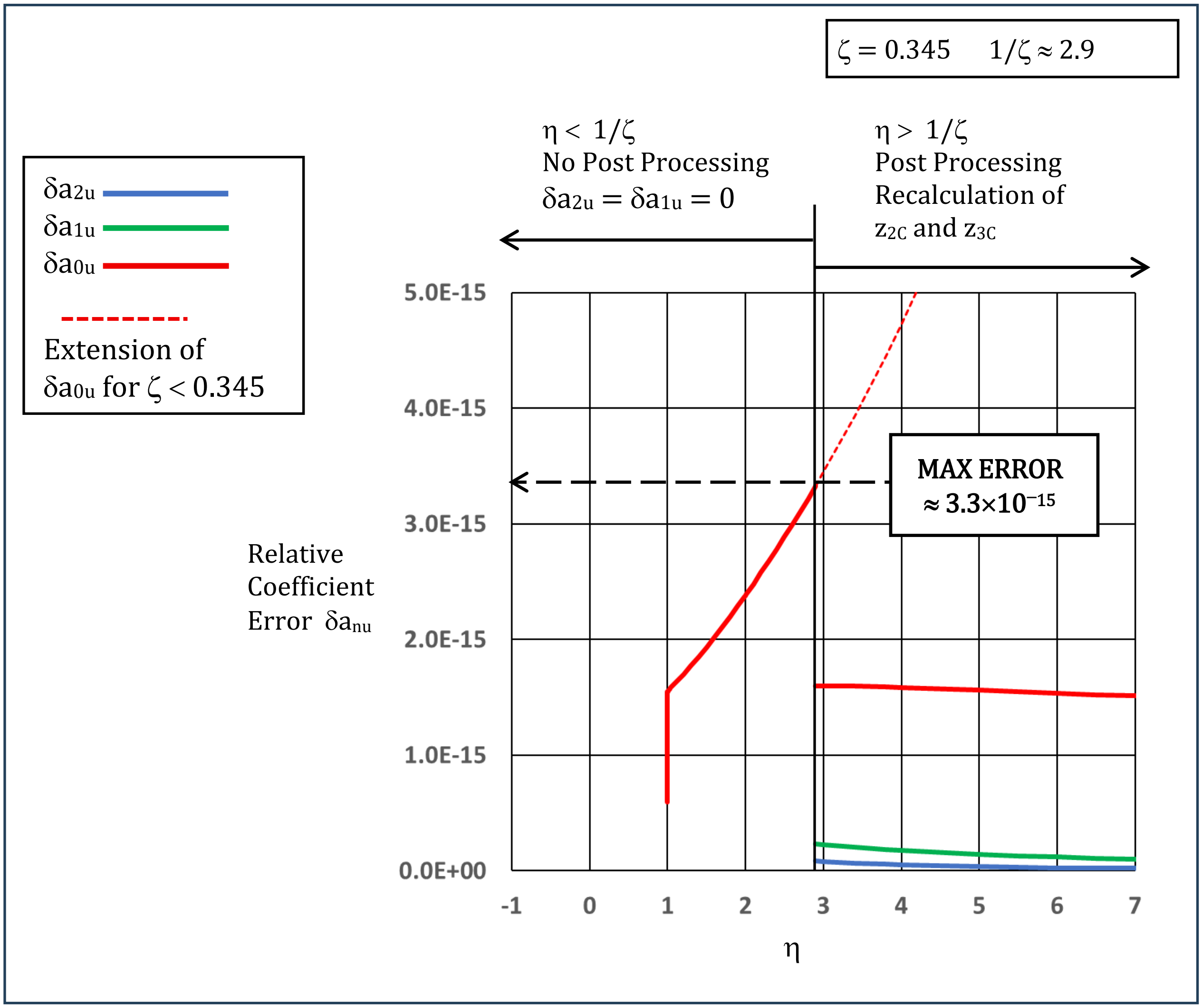
We first present plots of the δa_{2u}, δa_{1u}, and δa_{0u} and then derive their formulas.

Relative Coefficient Error Results

Figures 31 and 32 below plot δa_{2u}, δa_{1u}, and δa_{0u} versus η with $\zeta = 0.345$. Figure 31 presents the case $x_0 > 0$, Figure 32, the case $x_0 < 0$.

Figure 31 shows the simpler case: $x_0 > 0$, which implies that $x_A > x_0 > 0$ and $\eta = x_A/x_0 > 1$. Post processing recalculates the multiplicity 2 near-miss solutions when $|z_2| \approx x_0 < \zeta x_A$, that is when $\eta = x_A/x_0 > 1/\zeta \approx 2.9$. When $\eta \leq 1/\zeta$, there is no post-processing recalculation, which produces $\delta a_{2u} = \delta a_{1u} = 0$ and a δa_{0u} (red curve) that increases monotonically with η.

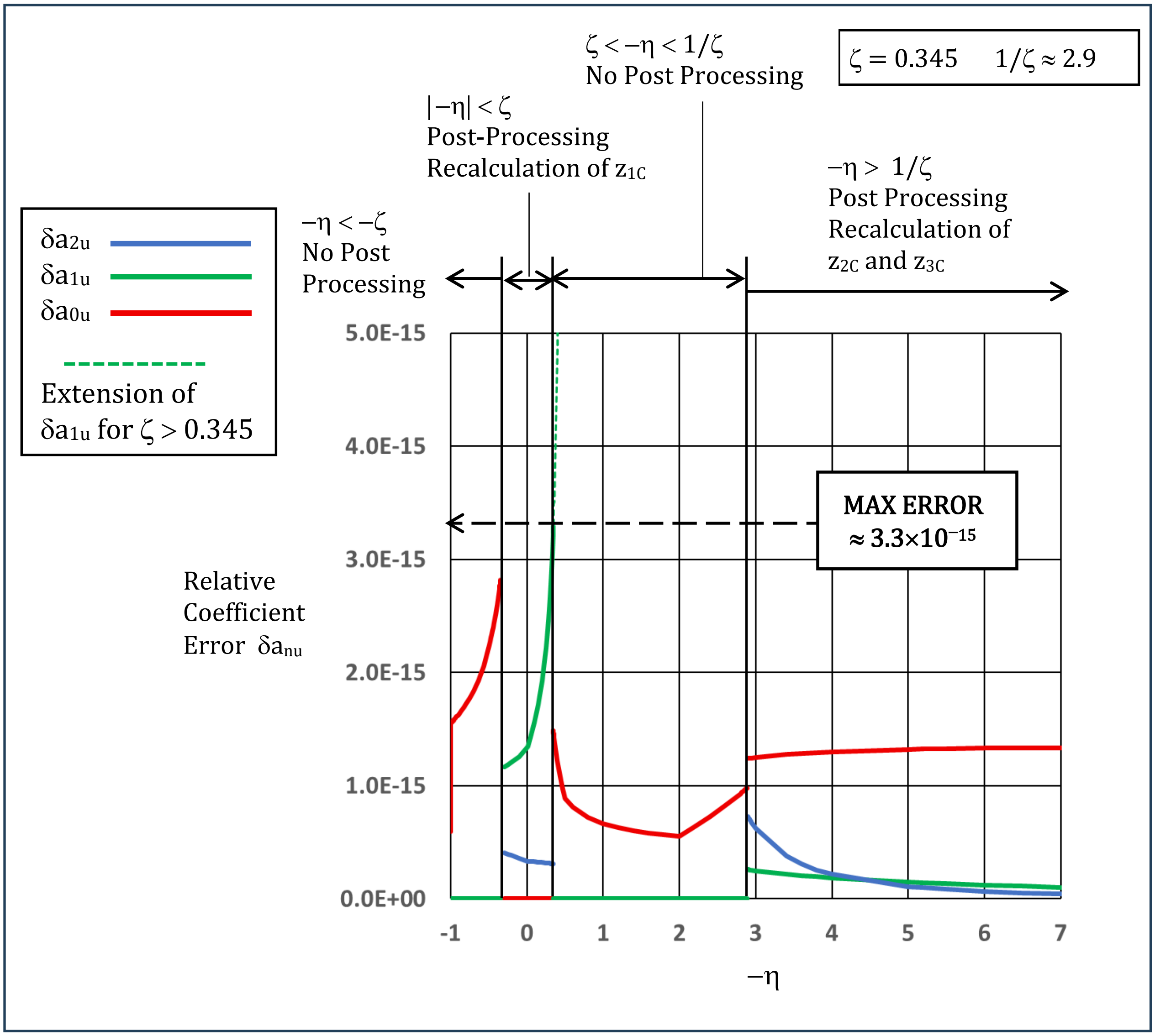
Figure 31 Zero-Guard-Induced Relative Coefficient Error with $\zeta = 0.345$ and $x_0 > 0$



The maximum δa_{0u} is $3.3{\times}10^{-15}$. This is the no-post-processing value of δa_{0u} at the transition $\eta = 1/\zeta \approx 2.9$. If ζ were any less than 0.345, then the transition value $\eta = 1/\zeta$ would be greater than 2.9, the transition point would move to the right, and the maximum relative coefficient error would increase above $3.3{\times}10^{-15}$ as shown by the red dashed curve.

Figure 32 below plots the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} for the case $x_0 < 0$.

Figure 32 Zero-Guard-Induced Relative Coefficient Error with $\zeta = 0.345$ and $x_0 < 0$



The inequalities $x_0 < 0$ and $x_A > x_0$ imply that $\eta \equiv x_A/x_0$ has a maximum value of 1, which occurs when $x_A = x_0 < 0$. As x_A increases above $x_0 = -|x_0|$, η decreases without limit.

Figure 32 above plots δa_{2u}, δa_{1u}, and δa_{0u} versus $-\eta = x_A/(-x_0)$, which increases as x_A increases. Post processing recalculates z_{1C} (the calculated value of the simple root $z_1 = x_A$)

when $|x_A| < \zeta|x_0|$, that is when $|-\eta| < \zeta$ or $-\zeta < -\eta < \zeta$. Post processing recalculates the two near-miss roots z_{2C} and z_{3C} when $|x_0| < \zeta|x_A|$, that is when $-\eta > 1/\zeta$.

The maximum relative coefficient error is $\delta a_{1u} = 3.3\times10^{-15}$, which occurs at the upper bound $-\eta = \zeta = 0.345$ of post processing recalculation of z_{1C}. If ζ were any greater than 0.345, then the transition point $-\eta = \zeta$ would move to the right, and the maximum relative coefficient error would increase above 3.3×10^{-15} as shown by the green dashed curve. The dramatic increase of error $\delta a_{1u} = |(a_{1C} - a_1)/a_1|$ with $-\eta$ occurs because coefficient a_1 goes to 0 at $-\eta = 0.5$. Equation (112) shows this:

$$a_1 = (2\eta + 1)x_0^2 \;\Rightarrow\; a_1 = 0 \text{ when } \eta = -0.5.$$

This problem affects δa_{1u} only when there is post-processing recalculation as shown below. Without post processing, zero-guard processing calculates roots z_{1C}, z_{2C}, and z_{3C} such that check coefficients a_{2C} and a_{1C} are exactly equal to the true coefficient values. Then $a_{2C} = a_2$ and $a_{1C} = a_1$, which implies that $\delta a_{2u} = \delta a_{1u} = 0$.

The remainder of this paper derives the formulas for the zero-guard-induced relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} as plotted in Figures 31 and 32 above.

Derivation of Relative Coefficient Errors without Post Processing

We examine first the case of no post processing. The Figure 9 cubic-equation algorithm calculates solutions z_1, z_2, and z_3, but there is no post-processing recalculation of any of the solutions. The true solution values of the multiplicity 2 cubic equation are

$$z_1 = x_A, \quad z_2 = x_0 + \Delta z \quad \text{and} \quad z_3 = x_0 - \Delta z$$

where $x_A > x_0$ and Δz is positive real.

The Figure 9 algorithm calculates parameters a_{2E}, a_{1E}, a_{0E}, q, q_E, r, r_E, R, and R_E according to Equations (152) to (156) above. The zero-guard condition $|\Delta z| < |\Delta z|_{ZG}$ is equivalent to $|R| < R_E\,\varepsilon$. Under this condition, the algorithm calculates the three cubic-equation solutions as

$$z_{1C} = 2s - a_2/3 \quad \text{and} \quad z_{2C} = z_{3C} = -s - a_2/3 \quad \text{where} \quad s = \sqrt{-q} \text{ and } q \le 0. \tag{178}$$

The check coefficient a_{2C} from Equation (2) is

$$a_{2C} = -(z_{1C} + z_{2C} + z_{3C}) = -(2s - a_2/3 - s - a_2/3 - s - a_2/3) = a_2$$

$$\boxed{a_{2C} = a_2\,.}$$

The check coefficient a_{1C} from Equation (2) is

$$a_{1C} = z_{1C}(z_{2C} + z_{3C}) + z_{2C}z_{3C} = (2s - a_2/3)(-2s - 2a_2/3) + (-s - a_2/3)^2 = -3s^2 + a_2^2/3$$

$$a_{1C} = 3q + a_2^2/3.$$

Equation (153) gives q as $q = a_1/3 - a_2{}^2/9$ so that

$$\boxed{a_{1C} = a_1.}$$

Thus, with no post processing and $|\Delta z| < |\Delta z|_{ZG}$, the zero-guard processing in the Figure 9 algorithm calculates solutions z_{1C}, z_{2C}, and z_{3C} such that the check coefficients a_{2C} and a_{1C} are identical to their input coefficient counterparts a_2 and a_1. The corresponding relative coefficient errors δa_{2u} and δa_{1u} of Equation (82) are therefore both 0.

$$\boxed{\delta a_{2u} = \delta a_{1u} = 0 \qquad \text{without post processing}} \tag{179}$$

We can now calculate the relative coefficient error δa_{0u} based on the results $a_{2C} = a_2$ and $a_{1C} = a_1$ for the zero-guard condition $|\Delta z| < |\Delta z|_{ZG} \Leftrightarrow |R| < R_E\, \varepsilon$. Equations (122) to (124) give the input coefficients as

$$a_2 \;=\; -(x_A + 2x_0) \qquad\qquad a_1 \;=\; 2x_Ax_0 + x_0^2 \;-\; \Delta z^2 \qquad\qquad a_0 \;=\; -x_A\,(x_0^2 - \Delta z^2).$$

Equations (2) give the check coefficients as

$$a_{2C} \;=\; -(z_{1C} + z_{2C} + z_{3C}) \qquad a_{1C} \;=\; z_{1C}(z_{2C} + z_{3C}) + z_{2C}z_{3C} \qquad a_{0C} \;=\; -z_{1C}z_{2C}z_{3C}\,. \tag{180}$$

Define the quantities δ and δ_A as

$$\delta \equiv z_{2C} - x_0, \qquad \delta_A \equiv z_{1C} - x_A\,.$$

Calculated solutions z_{2C} and z_{3C} are equal ($z_{2C} = z_{3C} = -s - a_2/3$), so they must be real, and we may write the three calculated solutions as

$$z_{1C} = x_A + \delta_A$$

$$z_{2C} = z_{3C} = x_0 + \delta. \tag{181}$$

The check coefficient a_{2C} is

$$a_{2C} = -(z_{1C} + z_{2C} + z_{3C}) \;= -(x_A + \delta_A \;+ 2x_0 + 2\delta)\text{, and the input coefficient } a_2 \text{ is}$$

$$a_2 \;= -(x_A \qquad + 2x_0).$$

The equality $a_{2C} = a_2$ therefore implies that $\delta_A = -2\delta$, and

$$z_{1C} = x_A - 2\delta. \tag{182}$$

We can now find δ from the equality $a_{1C} = a_1$. The check coefficient a_{1C} is

$$a_{1C} \;=\; z_{1C}(z_{2C} + z_{3C}) + z_{2C}z_{3C} \;=\; (x_A - 2\delta)(2x_0 + 2\delta) + (x_0 + \delta)^2.$$

It is equal to the input coefficient a_1 given by

$$a_{1C} \;=\; a_1 \;=\; 2x_Ax_0 + x_0^2 \;-\; \Delta z^2.$$

These last two equations combine to produce the following quadratic equation in δ.

$$\delta^2 - \frac{2}{3}(x_A - x_0)\delta - \frac{1}{3}\Delta z^2 = 0$$

Normalize this equation by dividing it through by x_0^2.

$$\delta_u^2 - \frac{2}{3}(\eta - 1)\delta_u - \frac{1}{3}\Delta u^2 = 0 \qquad \text{where} \quad \delta_u \equiv \delta/x_0 \quad \text{and} \quad \Delta u \equiv \Delta z/x_0 \qquad (183)$$

Of the two δ_u solutions, we use the one of smaller absolute value. Use the Numerical Recipes solution of Figure 8.

$$\delta_u = -\mathrm{sgn}(\eta - 1)\frac{\Delta u^2}{|\eta - 1| + \sqrt{(\eta - 1)^2 + 3\Delta u^2}} \qquad (184)$$

The function sgn(x) is the sign of x, (Equation (133)). Note that $\mathrm{sgn}(\eta - 1) = \mathrm{sgn}(x_0)$ because $\eta \equiv x_A/x_0$ and $x_A > x_0$.

Apply the calculated solutions in Equations (181) and (182) to the formula for check coefficient a_{0C} in Equation(180). Then normalize by x_0^3.

$$a_{0C} = -z_{1C}z_{2C}z_{3C} = -(x_A - 2\delta)(x_0 + \delta)^2 \Rightarrow a_{0C}/x_0^3 = -(\eta - 2\delta_u)(1 + \delta_u)^2$$

The corresponding formulas for a_0 and a_0/x_0^3 are

$$a_0 = -x_A\,(x_0^2 - \Delta z^2) \qquad \Rightarrow a_0/x_0^3 = -\eta(1 - \Delta u^2).$$

Use these formulas for a_{0C}/x_0^3 and a_0/x_0^3 to find the zero-guard relative coefficient error δa_{0u}.

$$\delta a_{0u} \equiv \left|\frac{a_{0C} - a_0}{a_0}\right| = \left|\frac{a_{0C}/x_0^3 - a_0/x_0^3}{a_0/x_0^3}\right|$$

The result is

$$\delta a_{0u} = \left|\frac{2(\eta - 1)\delta_u + (\eta - 4)\delta_u^2 - 2\delta_u^3 + \eta\Delta u^2}{\eta(1 - \Delta u^2)}\right|. \qquad (185)$$

The maximum δa_{0u} corresponds to a $\Delta u \equiv \Delta z/x_0$ equal to the relative zero-guard range $|\Delta z/x_0|_{ZG}$.

Calculate the maximum δa_{0u} versus η for $x_0 > 0$ and $x_0 < 0$ as follows. If $x_0 > 0$, then $\eta > 1$. If $x_0 < 0$, then $\eta < 1$. Calculate q_{uE} and r_{uE} with Equations (164) and (165). For $x_0 > 0$, calculate Δu as the positive real solution of Equation (166). For $x_0 < 0$, calculate Δu as the absolute value of the negative real solution of Equation (167). Finally, calculate δ_u and δa_{0u} using Equations (184) and (185).

This δa_{0u} is plotted as the solid red curve in Figures 31 and 32 for those ranges of η where there is no post processing. The corresponding δa_{2u} and δa_{1u} are zero for those ranges of η per Equation (179).

Derivation of Relative Coefficient Errors with Post-Processing Recalculation of z_{2C} and z_{3C}

We now derive the zero-guard-induced relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} when the Figure 12 cubic-equation post-processing algorithm recalculates the multiplicity 2 near-miss solutions z_{2C} and z_{3C} from the Figure 9 cubic-equation algorithm. For the most part, these relative coefficient errors are smaller than the δa_{0u} just derived for the no-post-processing case. The major reason is that zero-guard range is usually smaller when post processing recalculates z_{2C} and z_{3C}. See the comparison of zero-guard ranges in Figure 29 above. As in the no-post-processing case, relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} are to be expressed as functions of η for $x_0 > 0$ and $x_0 < 0$.

To find δa_{2u}, δa_{1u}, and δa_{0u}, we first need expressions for the calculated solutions z_{1C}, z_{2C}, and z_{3C} produced by the combination of Figure 9 and Figure 12 algorithms. We can then calculate the corresponding check coefficients a_{2C}, a_{1C}, and a_{0C} and relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u}.

The post-processing algorithm does not change the Figure 9 calculated value z_{1C}, so the relevant z_{1C} is that of the zero-guard processing in the Figure 9 algorithm and is given above in Equation (178).

$$z_{1C} = 2s - a_2/3 \;=\; 2\sqrt{-q}\; - a_2/3 \quad \text{where} \quad q = a_1/3 - a_2^2/9$$

From Equations (122) and (123),

$$a_2 \;=\; -(x_A + 2x_0) \qquad\qquad a_1 \;=\; 2x_Ax_0 + x_0^2 \;-\; \Delta z^2.$$

Combine the equations above to obtain the normalized calculated solution, $u_{1C} \equiv z_{1C}/x_0$.

$$u_{1C} \equiv\; z_{1C}/x_0 = \frac{1}{3}\left[\eta + 2 + 2\,\mathrm{sgn}(x_0)\,\sqrt{(\eta-1)^2 + 3\Delta u^2}\right] \tag{186}$$

The relevant values of $(\eta - 1)^2$ and $3\Delta u^2$ in the radicand differ by almost 15 orders of magnitude. Figures 31 and 32 show that post-processing recalculation of z_{2C} and z_{3C} occurs for $|\eta|$ greater than 2.9, so $(\eta - 1)^2$ is greater than 3.6. The value $\Delta u \equiv \Delta z/x_0$ is evaluated as the post-processing relative zero-guard range $|\Delta z/x_0|_{ZG}$, whose value is about 4×10^{-8} as shown by the green curves in Figure 29. The value of $3\Delta u^2$ is thus about 5×10^{-15}.

This great magnitude difference between $(\eta - 1)^2$ and $3\Delta u^2$ means that round-off error will swamp the contribution of $3\Delta u^2$ to u_{1C} when Equation (186) is evaluated. The equation needs to be modified so that the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} can be accurately determined.

Extract $(\eta - 1)^2$ from the radical in Equation (186) to give

$$u_{1C} \;= \frac{1}{3}\left[\eta + 2 + 2\,\mathrm{sgn}(x_0)\;|\eta - 1|\;\sqrt{1+2\theta}\,\right]$$

where

$$\theta \equiv \frac{3\Delta u^2}{2(\eta-1)^2} \qquad \text{and} \quad 0 \;<\; \theta \;<<\; 1.$$

Approximate $\sqrt{1+2\theta}$ as $1 + \theta$.

$$u_{1C} \;= \frac{1}{3}\left[\eta + 2 + 2\,\mathrm{sgn}(x_0)\;|\eta - 1|\;(1+\theta)\right]$$

Whether $x_0 > 0$ or $x_0 < 0$, the quantity $\mathrm{sgn}(x_0)|\eta - 1|$ is $\eta - 1$, and the last two equations combine to give

$$u_{1C} = \eta + \delta\eta \qquad \text{where} \qquad \delta\eta \equiv \Delta u^2/(\eta - 1). \tag{187}$$

This is the expression we seek. The contribution of Δu^2 to u_{1C} is obvious, and the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u} will be easy to calculate accurately.

The Figure 12 post processing uses the simple solution z_{1C} from the Figure 9 algorithm and the cubic-equation coefficients a_0 and a_1 to calculate the remaining solutions z_{2C} and z_{3C}. The post processing invokes the Figure 8 quadratic-equation algorithm to calculate z_{2C} and z_{3C} as the two solutions of the quadratic equation

$$z^2 + Bz + C = 0 \qquad \text{where} \quad C = \frac{-a_0}{z_{1C}} \qquad \text{and} \quad B = \frac{C - a_1}{z_{1C}}.$$

In the situation of interest, the difference $z_{2C} - z_{3C}$ is smaller than the quadratic-equation zero-guard range, which implies that $|D| < D_E\,\varepsilon$. The quadratic-equation algorithm sets determinate D to zero and calculates z_{2C} and z_{3C} as the equal values

$$z_{2C} = z_{3C} = -\frac{B}{2} = \frac{a_0 + a_1 z_{1C}}{2z_{1C}^2}.$$

To this equation, apply the expressions $a_1 = 2x_A x_0 + x_0^2 - \Delta z^2$ and $a_0 = -x_A\,(x_0^2 - \Delta z^2)$ from Equations (123) and (124). Then divide through by x_0 to obtain the expression for the normalized solutions $u_{2C} \equiv z_{2C}/x_0$ and $u_{3C} \equiv z_{3C}/x_0$.

$$u_{2C} = u_{3C} = \frac{-\eta(1 - \Delta u^2) + (2\eta + 1 - \Delta u^2)u_{1C}}{2u_{1C}^2}.$$

Apply $u_{1C} = \eta + \delta\eta$ from Equation (187) and simplify.

$$u_{2C} = u_{3C} = \frac{2\eta^2 + (2\eta + 1)\delta\eta - \delta\eta\Delta u^2}{2\left(\eta^2 + 2\eta\delta\eta + \delta\eta^2\right)}$$

Drop second-order error terms: $-\delta\eta\Delta u^2$ in the numerator and $\delta\eta^2$ inside the denominator parentheses.

$$u_{2C} = u_{3C} = \frac{2\eta^2 + (2\eta + 1)\delta\eta}{2\eta^2(1 + 2\delta\eta/\eta)}$$

Multiply numerator and denominator by $(1 - 2\delta\eta/\eta)$ and simplify, dropping terms that contain $\delta\eta^2$ from both the numerator and denominator. The result for $u_{2C} = u_{3C}$ corresponds to u_{1C} in Equation (187).

$$u_{2C} = u_{3C} = 1 + \delta u_{2C} \qquad \text{where} \qquad \delta u_{2C} \equiv -\frac{2\eta - 1}{2\eta^2}\delta\eta \tag{188}$$

Equations (187) and (188) are the desired expressions for the normalized calculated solutions $u_{1C} \equiv z_{1C}/x_0$, $u_{2C} \equiv z_{2C}/x_0$, and $u_{3C} \equiv z_{3C}/x_0$. We can now find the corresponding formulas for the normalized check coefficients and the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u}.

Normalize the cubic-equation coefficients a_2, a_1, and a_0 in Equations (122) to (124) by the appropriate power of x_0.

$$a_2/x_0 = -(\eta + 2) \qquad a_1/x_0^2 = 2\eta + 1 - \Delta u^2 \qquad a_0/x_0^3 = -\eta\,(1 - \Delta u^2). \qquad (189)$$

Do the same for the check coefficients in Equation (180).

$$a_{2C}/x_0 = -(u_{1C} + u_{2C} + u_{3C}) \quad a_{1C}/x_0^2 = u_{1C}(u_{2C} + u_{3C}) + u_{2C}u_{3C} \quad a_{0C}/x_0^3 = -u_{1C}u_{2C}u_{3C} \quad (190)$$

We apply Equations (187) to (190) to Equation (82) to find the formulas for the relative coefficient errors δa_{2u}, δa_{1u}, and δa_{0u}.

$$\delta a_{2u} = \left|\frac{a_{2C}/x_0 - a_2/x_0}{a_2/x_0}\right|, \qquad \delta a_{1u} = \left|\frac{a_{1C}/x_0^2 - a_1/x_0^2}{a_1/x_0^2}\right|, \qquad \delta a_{0u} = \left|\frac{a_{0C}/x_0^3 - a_0/x_0^3}{a_0/x_0^3}\right|$$

The results are

$$\delta a_{2u} = \left|\frac{\eta - 1}{\eta^2(\eta + 2)}\right|\Delta u^2, \qquad \delta a_{1u} = \left|\frac{\eta^2 - 1}{\eta^2(2\eta + 1)}\right|\Delta u^2, \qquad \delta a_{0u} = \left|\frac{\eta^2 - 1}{\eta^2}\right|\Delta u^2$$

In deriving these formulas, we dropped all second-order error terms like $\delta\eta^2$, $\delta\eta\delta u_{2C}$, and δu_{2C}^2.

Evaluate these formulas for δa_{2u}, δa_{1u}, and δa_{0u} by setting Δu equal to the relative zero-guard range $|\Delta z/x_0|_{ZG}$ given by Equations (176), (177), and (173) for post-processing recalculation of z_2 and z_3. The resulting values of δa_{2u}, δa_{1u}, and δa_{0u} are plotted as the blue, green, and red curves on the right-hand portions of Figures 31 and 32 where $|\eta| > 1/\zeta \approx 2.9$.

Derivation of Relative Coefficient Errors with Post-Processing Recalculation of z_{1C}

This final derivation calculates δa_{2u}, δa_{1u}, and δa_{0u} when $|x_A| < |x_0|$ and post processing recalculates the simple solution z_{1C}. Our analysis convention that $x_A > x_0$ then requires that

$$x_0 < 0 \qquad \text{and} \qquad |\eta| = |x_A/x_0| < 1.$$

The Figure 12 cubic-equation post-processing algorithm uses solutions z_{2C} and z_{3C} from the Figure 9 cubic-equation algorithm to recalculate z_{1C} as

$$z_{1C} = \frac{-a_0}{z_{2C}z_{3C}}\,. \qquad (191)$$

This z_{1C} formula implies that the input coefficient a_0 is identical to the corresponding check coefficient a_{0C}: $a_0 = -z_{1C}\,z_{2C}\,z_{3C} = a_{0C}$. The conclusion that $a_0 = a_{0C}$ holds regardless of how z_{2C} and z_{3C} are calculated or whether zero-guard processing is involved. Because $a_0 = a_{0C}$, the relative coefficient error δa_{0u} is zero.

$$\delta a_{0u} \equiv \left|\frac{a_{0C} - a_0}{a_0}\right| = 0 \qquad (192)$$

The recalculation of z_{1C} does not affect solutions z_{2C} and z_{3C} from the Figure 9 algorithm. We therefore use Equations (181), (183), and (184) from the no-post-processing case to calculate z_{2C} and z_{3C}.

$$z_{2C} = z_{3C} = x_0 + \delta = x_0(1 + \delta_u) \tag{193}$$

This equation for z_{2C} and z_{3C} and Equation (191) for z_{1C} enable us to now calculate the check coefficients a_{2C} and a_{1C} and the relative coefficient errors δa_{2u} and δa_{1u}.

Equation (2) gives the check coefficient a_{2C} as $a_{2C} = -(z_{1C} + z_{2C} + z_{3C})$, so with the equations above and a_0/x_0^3 from Equation (189) we have

$$a_{2C} = \frac{a_0}{z_{2C}z_{3C}} - (z_{2C} + z_{3C}) = \frac{a_0 - 2z_{2C}^3}{z_{2C}^2}$$

$$a_{2C}/x_0 = \frac{-\eta(1 - \Delta u^2) - 2(1 + \delta_u)^3}{(1 + \delta_u)^2}.$$

From this expression, Equation (82) for δa_{2u}, and Equation (189) for a_2/x_0, the desired expression for relative coefficient error δa_{2u} becomes

$$\delta a_{2u} = \left|\frac{2(\eta - 1)\delta_u + (\eta - 4)\delta_u^2 - 2\delta_u^3 + \eta\Delta u^2}{(\eta + 2)(1 + \delta_u)^2}\right|. \tag{194}$$

We derive the expression for δa_{1u} in similar manner. Equation (2) gives the check coefficient a_{1C} as $a_{1C} = z_{1C}z_{2C} + z_{1C}z_{3C} + z_{2C}z_{3C} = 2z_{1C}z_{2C} + z_{2C}^2$. Apply Equations (191) for z_{1C}, (193) for $z_{2C} = z_{3C}$, (189) for a_0/x_0^3 and a_1/x_0^2, and (82) for δa_{1u}.

$$a_{1C} = 2\frac{-a_0}{z_{2C}z_{3C}}z_{2C} + z_{2C}^2 = \frac{-2a_0 + z_{2C}^3}{z_{2C}}.$$

$$a_{1C}/x_0^2 = \frac{2\eta(1 - \Delta u^2) + (1 + \delta_u)^3}{1 + \delta_u}.$$

$$\delta a_{1u} = \left|\frac{-2(\eta - 1)\delta_u + 3\delta_u^2 + \delta_u^3 + (1 - 2\eta + \delta_u)\Delta u^2}{(2\eta + 1 - \Delta u^2)(1 + \delta_u)}\right|. \tag{195}$$

Recall that, in this case, post processing does not affect solutions z_{2C} and z_{3C}, the calculated multiplicity 2 near-miss solutions. Therefore, Δu is set equal to the relative zero-guard range $|\Delta z/x_0|_{ZG}$ for the no-post-processing case. Also, $x_0 < 0$ and $|\eta| < 1$.

Evaluate Equations (194) and (195) for δa_{2u} and δa_{1u} versus η as follows. Calculate q_{uE} and r_{uE} with Equations (164) and (165). Calculate Δu as the absolute value of the negative real solution of Equation (167). Finally, calculate δ_u, δa_{2u}, and δa_{1u} using Equations (184), (194), and (195) respectively. The resulting values of δa_{2u} and δa_{1u} are plotted as the blue and green curves in Figure 32 for $-\eta$ values satisfying $|-\eta| < \zeta = 0.345$. The value of δa_{0u} (red curve) for that range of η is zero, Equation (192). Equation (195) also provides the δa_{1u} values for the dashed green curve at $-\eta > \zeta = 0.345$.

XI. REFERENCES

1. “Machine epsilon”, *Wikipedia*: https://en.wikipedia.org/wiki/Machine_epsilon

2. “IEEE 754-1985”, *Wikipedia*: https://en.wikipedia.org/wiki/IEEE_754-1985

3. Press, W.H., et al., *Numerical Recipes. The Art of Scientific Computing*, 3rd Edition, 2007, Cambridge University Press, ISBN 978-0-521-88068-8, https://iate.oac.uncor.edu/~mario/materia/nr/numrec/f5-6.pdf.

4. Cardano, Girolamo, *The Rules of Algebra (Ars Magna)* [1545], translated and edited by T. Richard Witmer. 2007 reissue, Dover Publications, Inc., Mineola, NY (1993) ISBN 0-486-45873-3.

5. Blinn, James F., “How to Solve a Cubic Equation Part 5 -- Back to Numerics”, https://courses.cs.washington.edu/courses/cse590b/13au/lecture_notes/solvecubic_p5.pdf.

6. Nickalls, R. W. D., "Viète, Descartes and the cubic equation", *The Mathematical Gazette* 2006; **90** (July; No. 518), 203–208, https://pdfs.semanticscholar.org/a4ca/119b8a221fdbe2bd3f253472f932b76e673e.pdf, https://web.archive.org/web/20190503030505/https://www.cambridge.org/core/services/aop-cambridge-core/content/view/AD8AEC94661F645DC7B46781E087DA30/S0025557200179598a.pdf/div-class-title-viete-descartes-and-the-cubic-equation-div.pdf.