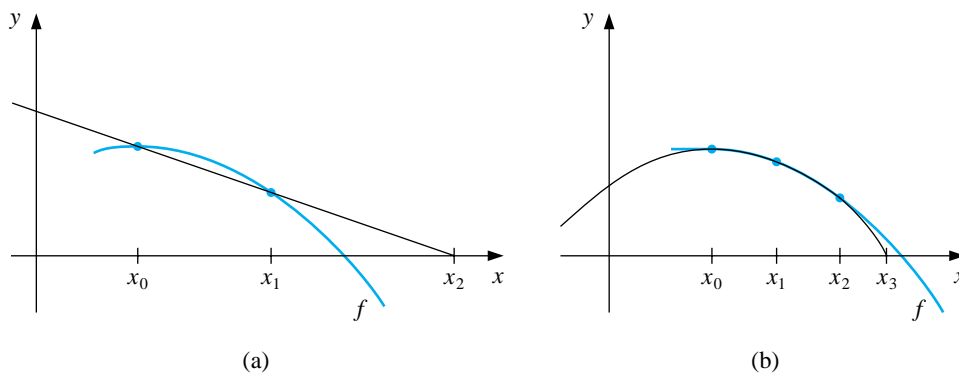


## 2.6 Müller's Method

There are a number of root-finding problems for which the Secant, False Position, and Newton's methods will not give satisfactory results. They will not give rapid convergence, for example, when the function and its derivative are simultaneously close to zero. In addition, these methods cannot be used to approximate complex roots unless the initial approximation is a **complex number whose imaginary part is nonzero**. This often makes them a poor choice for use in approximating the roots of polynomials, which, even with real coefficients, commonly have complex roots occurring in conjugate pairs.

In this section we consider Müller's method, which is a generalization of the Secant method. The Secant method finds the zero of the line passing through points on the graph of the function that corresponds to the two immediately previous approximations, as shown in Figure 2.8(a). Müller's method uses the zero of the parabola through the three immediately previous points on the graph as the new approximation, as shown in part (b) of Figure 2.8.

**Figure 2.8**



Suppose that three initial approximations,  $p_0, p_1$ , and  $p_2$ , are given for a solution of  $f(x) = 0$ . The derivation of Müller's method for determining the next approximation  $p_3$  begins by considering the quadratic polynomial

$$P(x) = a(x - p_2)^2 + b(x - p_2) + c$$

that passes through  $(p_0, f(p_0))$ ,  $(p_1, f(p_1))$ , and  $(p_2, f(p_2))$ . The constants  $a, b$ , and  $c$  can be determined from the conditions

$$\begin{aligned} f(p_0) &= a(p_0 - p_2)^2 + b(p_0 - p_2) + c, \\ f(p_1) &= a(p_1 - p_2)^2 + b(p_1 - p_2) + c, \end{aligned}$$

and

$$f(p_2) = a \cdot 0^2 + b \cdot 0 + c.$$

To determine  $p_3$ , the root of  $P(x) = 0$ , we apply the quadratic formula to  $P(x)$ . Because of round-off error problems caused by the subtraction of nearly equal numbers, however, we apply the formula in the manner prescribed in Example 1 of Section 1.4:

$$p_3 - p_2 = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}.$$

This gives two possibilities for  $p_3$ , depending on the sign preceding the radical term. In Müller's method, the sign is chosen to agree with the sign of  $b$ . Chosen in this manner, the denominator will be the largest in magnitude, which avoids the possibility of subtracting nearly equal numbers and results in  $p_3$  being selected as the closest root of  $P(x) = 0$  to  $p_2$ .

[Müller's Method] Given initial approximations  $p_0, p_1$ , and  $p_2$ , generate

$$p_3 = p_2 - \frac{2c}{b + \operatorname{sgn}(b)\sqrt{b^2 - 4ac}},$$

where

$$c = f(p_2),$$

$$b = \frac{(p_0 - p_2)^2[f(p_1) - f(p_2)] - (p_1 - p_2)^2[f(p_0) - f(p_2)]}{(p_0 - p_2)(p_1 - p_2)(p_0 - p_1)},$$

and

$$a = \frac{(p_1 - p_2)[f(p_0) - f(p_2)] - (p_0 - p_2)[f(p_1) - f(p_2)]}{(p_0 - p_2)(p_1 - p_2)(p_0 - p_1)}.$$

Then continue the iteration, with  $p_1, p_2$ , and  $p_3$  replacing  $p_0, p_1$ , and  $p_2$ .

The method continues until a satisfactory approximation is obtained. Since the method involves the radical  $\sqrt{b^2 - 4ac}$  at each step, the method approximates complex roots when  $b^2 - 4ac < 0$ , provided, of course, that complex arithmetic is used.

#### EXAMPLE 1

Consider the polynomial  $f(x) = 16x^4 - 40x^3 + 5x^2 + 20x + 6$ . Using the program MULLER25 with accuracy tolerance  $10^{-5}$  and various inputs for  $p_0, p_1$ , and  $p_2$  produces the results in Tables 2.9, 2.10, and 2.11.  $\square$

Table 2.9

$$p_0 = 0.5, \quad p_1 = -0.5, \quad p_2 = 0$$

$n$	$p_n$	$f(p_n)$
3	$-0.555556 + 0.598352i$	$-29.4007 - 3.89872i$
4	$-0.435450 + 0.102101i$	$1.33223 - 1.19309i$
5	$-0.390631 + 0.141852i$	$0.375057 - 0.670164i$
6	$-0.357699 + 0.169926i$	$-0.146746 - 0.00744629i$
7	$-0.356051 + 0.162856i$	$-0.183868 \times 10^{-2} + 0.539780 \times 10^{-3}i$
8	$-0.356062 + 0.162758i$	$0.286102 \times 10^{-5} + 0.953674 \times 10^{-6}i$

Table 2.10

$$p_0 = 0.5, p_1 = 1.0, p_2 = 1.5$$

$n$	$p_n$	$f(p_n)$
3	1.28785	-1.37624
4	1.23746	0.126941
5	1.24160	$0.219440 \times 10^{-2}$
6	1.24168	$0.257492 \times 10^{-4}$
7	1.24168	$0.257492 \times 10^{-4}$

Table 2.11

$$p_0 = 2.5, p_1 = 2.0, p_2 = 2.25$$

$n$	$p_n$	$f(p_n)$
3	1.96059	-0.611255
4	1.97056	$0.748825 \times 10^{-2}$
5	1.97044	$-0.295639 \times 10^{-4}$
6	1.97044	$-0.295639 \times 10^{-4}$

To use Maple to generate the first entry in Table 2.9 we define  $f(x)$  and the initial approximations with the **Maple statements**

```
>f:=x->16*x^4-40*x^3+5*x^2+20*x+6;
>p0:=0.5; p1:=-0.5; p2:=0.0;
```

We evaluate the polynomial at the initial values

```
>f0:=f(p0); f1:=f(p1); f2:=f(p2);
```

and we compute  $c = 6$ ,  $b = 10$ ,  $a = 9$ , and  $p_3 = -0.5555555558 + 0.5983516452i$  using the Müller's method formulas:

```

>c:=f2;
>b:=(p0-p2)^2*(f1-f2)-(p1-p2)^2*(f0-f2)/((p0-p2)*(p1-p2)*(p0-p1));
>a:=(p1-p2)*(f0-f2)-(p0-p2)*(f1-f2)/((p0-p2)*(p1-p2)*(p0-p1));
>p3:=p2-(2*c)/(b+(b/abs(b))*sqrt(b^2-4*a*c));

```

The value  $p_3$  was generated using complex arithmetic, as is the calculation

```
>f3:=f(p3);
```

which gives  $f_3 = -29.40070112 - 3.898724738i$ .

The actual values for the roots of the equation are  $-0.356062 \pm 0.162758i$ , 1.241677, and 1.970446, which demonstrate the accuracy of the approximations from Müller's method.

Example 1 illustrates that Müller's method can approximate the roots of polynomials with a variety of starting values. In fact, the technique generally converges to the root of a polynomial for any initial approximation choice. General-purpose software packages using Müller's method request only one initial approximation per root and, as an option, may even supply this approximation.

Although Müller's method is not quite as efficient as Newton's method, it is generally better than the Secant method. The relative efficiency, however, is not as important as the ease of implementation and the likelihood that a root will be found. Any of these methods will converge quite rapidly once a reasonable initial approximation is determined.

When a sufficiently accurate approximation  $p^*$  to a root has been found,  $f(x)$  is divided by  $x - p^*$  to produce what is called a *deflated* equation. If  $f(x)$  is a polynomial of degree  $n$ , the deflated polynomial will be of degree  $n - 1$ , so the computations are simplified. After an approximation to the root of the deflated equation has been determined, either Müller's method or Newton's method can be used in the original function with this root as the initial approximation. This will ensure that the root being approximated is a solution to the true equation, not to the less accurate deflated equation.

## EXERCISE SET 2.6

1. Find the approximations to within  $10^{-4}$  to all the real zeros of the following polynomials using Newton's method.

(a)  $P(x) = x^3 - 2x^2 - 5$

(b)  $P(x) = x^3 + 3x^2 - 1$

(c)  $P(x) = x^4 + 2x^2 - x - 3$

(d)  $P(x) = x^5 - x^4 + 2x^3 - 3x^2 + x - 4$

2. Find approximations to within  $10^{-5}$  to all the zeros of each of the following polynomials by first finding the real zeros using Newton's method and then reducing to polynomials of lower degree to determine any complex zeros.

(a)  $P(x) = x^4 + 5x^3 - 9x^2 - 85x - 136$

(b)  $P(x) = x^4 - 2x^3 - 12x^2 + 16x - 40$

(c)  $P(x) = x^4 + x^3 + 3x^2 + 2x + 2$

(d)  $P(x) = x^5 + 11x^4 - 21x^3 - 10x^2 - 21x - 5$

3. Repeat Exercise 1 using Müller's method.  
 4. Repeat Exercise 2 using Müller's method.  
 5. Find, to within  $10^{-3}$ , the zeros and critical points of the following functions. Use this information to sketch the graphs of  $P$ .

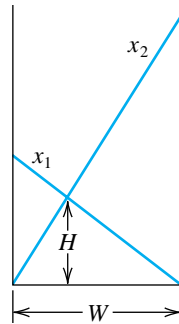
(a)  $P(x) = x^3 - 9x^2 + 12$

(b)  $P(x) = x^4 - 2x^3 - 5x^2 + 12x - 5$

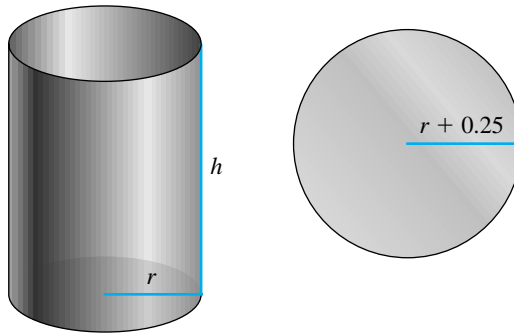
6.  $P(x) = 10x^3 - 8.3x^2 + 2.295x - 0.21141 = 0$  has a root at  $x = 0.29$ .  
 (a) Use Newton's method with  $p_0 = 0.28$  to attempt to find this root.  
 (b) Use Müller's method with  $p_0 = 0.275$ ,  $p_1 = 0.28$ , and  $p_2 = 0.285$  to attempt to find this root.  
 (c) Explain any discrepancies in parts (a) and (b).

7. Use Maple to find the exact roots of the polynomial  $P(x) = x^3 + 4x - 4$ .  
 8. Use Maple to find the exact roots of the polynomial  $P(x) = x^3 - 2x - 5$ .  
 9. Use each of the following methods to find a solution accurate to within  $10^{-4}$  for the problem

$$600x^4 - 550x^3 + 200x^2 - 20x - 1 = 0, \quad \text{for } 0.1 \leq x \leq 1.$$



- (a) Bisection method
- (b) Newton's method
- (c) Secant method
- (d) method of False Position
- (e) Müller's method
10. Two ladders crisscross an alley of width  $W$ . Each ladder reaches from the base of one wall to some point on the opposite wall. The ladders cross at a height  $H$  above the pavement. Find  $W$  given that the lengths of the ladders are  $x_1 = 20$  ft and  $x_2 = 30$  ft and that  $H = 8$  ft. (See the figure on page 58.)
11. A can in the shape of a right circular cylinder is to be constructed to contain  $1000 \text{ cm}^3$ . The circular top and bottom of the can must have a radius of  $0.25$  cm more than the radius of the can so that the excess can be used to form a seal with the side. The sheet of material being formed into the side of the can must also be  $0.25$  cm longer than the circumference of the can so that a seal can be formed. Find, to within  $10^{-4}$ , the minimal amount of material needed to construct the can.
12. In 1224 Leonardo of Pisa, better known as Fibonacci, answered a mathematical challenge of John of Palermo in the presence of Emperor Frederick II. His challenge was to find a root of the equation  $x^3 + 2x^2 + 10x = 20$ . He first showed that the equation had no rational roots and no Euclidean irrational root—that is, no root in one of the forms  $a \pm \sqrt{b}$ ,  $\sqrt{a} \pm \sqrt{b}$ ,  $\sqrt{a \pm \sqrt{b}}$ , or  $\sqrt{\sqrt{a} \pm \sqrt{b}}$ , where  $a$  and  $b$  are rational numbers. He then approximated the only real root, probably using an algebraic technique of Omar Khayyam



involving the intersection of a circle and a parabola. His answer was given in the base-60 number system as

$$1 + 22 \left(\frac{1}{60}\right) + 7 \left(\frac{1}{60}\right)^2 + 42 \left(\frac{1}{60}\right)^3 + 33 \left(\frac{1}{60}\right)^4 + 4 \left(\frac{1}{60}\right)^5 + 40 \left(\frac{1}{60}\right)^6.$$

How accurate was his approximation?

## 2.7 Survey of Methods and Software

In this chapter we have considered the problem of solving the equation  $f(x) = 0$ , where  $f$  is a given continuous function. All the methods begin with an initial approximation and generate a sequence that converges to a root of the equation, if the method is successful. If  $[a, b]$  is an interval on which  $f(a)$  and  $f(b)$  are of opposite sign, then the Bisection method and the method of False Position will converge. However, the convergence of these methods may be slow. Faster convergence is generally obtained using the Secant method or Newton's method. Good initial approximations are required for these methods, two for the Secant method and one for Newton's method, so the Bisection or the False Position method can be used as starter methods for the Secant or Newton's method.

Müller's method will give rapid convergence without a particularly good initial approximation. It is not quite as efficient as Newton's method, but it is better than the Secant method, and it has the added advantage of being able to approximate complex roots.

Deflation is generally used with Müller's method once an approximate root of a polynomial has been determined. After an approximation to the root of the deflated equation has been determined, use either Müller's method or Newton's method in the original polynomial with this root as the initial approximation. This procedure will ensure that the root being approximated is a solution to the true equation, not to the deflated equation. We recommended Müller's method for finding all the zeros of polynomials, real or complex. Müller's method can also be used for an arbitrary continuous function.

Other high-order methods are available for determining the roots of polynomials. If this topic is of particular interest, we recommend that consideration be given to Laguerre's method, which gives cubic convergence and also approximates complex roots (see [Ho, pp. 176–179] for a complete discussion), the Jenkins-Traub method (see [JT]), and Brent's method. (see [Bre]), Both IMSL and NAG supply subroutines based on Brent's method. This technique uses a combination of linear interpolation, an inverse quadratic interpolation similar to Müller's method, and the bisection method.

The netlib FORTRAN subroutine `fzero.f` uses a combination of the Bisection and Secant method developed by T. J. Dekker to approximate a real zero of  $f(x) = 0$  in the interval  $[a, b]$ . It requires specifying an interval  $[a, b]$  that contains a root and returns an interval with a width that is within a specified tolerance. The FORTRAN subroutine `sdzro.f` uses a combination of the bisection method, interpolation, and extrapolation to find a real zero of  $f(x) = 0$  in a given interval  $[a, b]$ . The routines `rpzero` and `cpzero` can be used to approximate all zeros of a real polynomial or complex polynomial, respectively. Both methods use Newton's method for systems, which will be considered in Chapter 10. All routines are given in single and double precision. These methods are available on the Internet from netlib at <http://www.netlib.org/slatec/src>.

Within MATLAB, the function `ROOTS` is used to compute all the roots, both real and complex, of a polynomial. For an arbitrary function, `FZERO` computes a root near a specified initial approximation to within a specified tolerance.



Maple has the procedure `fsolve` to find roots of equations. For example,  
`>fsolve(x^2 - x - 1, x);`  
returns the numbers  $-0.6180339887$  and  $1.618033989$ . You can also specify a particular variable and interval to search. For example,  
`>fsolve(x^2 - x - 1, x, 1..2);`  
returns only the number  $1.618033989$ . The command `fsolve` uses a variety of specialized techniques that depend on the particular form of the equation or system of equations.

Notice that in spite of the diversity of methods, the professionally written packages are based primarily on the methods and principles discussed in this chapter. You should be able to use these packages by reading the manuals accompanying the packages to better understand the parameters and the specifications of the results that are obtained.

There are three books that we consider to be classics on the solution of nonlinear equations, those by Traub [Tr], by Ostrowski [Os], and by Householder [Ho]. In addition, the book by Brent [Bre] served as the basis for many of the currently used root-finding methods.