

## Research Article

# A Generalized Regula Falsi Method for Finding Zeros and Extrema of Real Functions

Abel Gomes and José Morgado

Instituto de Telecomunicações, Universidade da Beira Interior, 6200-001 Covilhã, Portugal

Correspondence should be addressed to Abel Gomes; [agomes@di.ubi.pt](mailto:agomes@di.ubi.pt)

Received 18 November 2012; Revised 18 June 2013; Accepted 2 July 2013

Academic Editor: Hung Nguyen-Xuan

Copyright © 2013 A. Gomes and J. Morgado. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Many zero-finding numerical methods are based on the Intermediate Value Theorem, which states that a zero of a real function  $f : \mathbb{R} \rightarrow \mathbb{R}$  is bracketed in a given interval  $[A, B] \subset \mathbb{R}$  if  $f(A)$  and  $f(B)$  have opposite signs; that is,  $f(A) \cdot f(B) < 0$ . But, some zeros cannot be bracketed this way because they do not satisfy the precondition  $f(A) \cdot f(B) < 0$ . For example, local minima and maxima that annihilate  $f$  may not be bracketed by the Intermediate Value Theorem. In this case, we can always use a numerical method for bracketing extrema, checking then whether it is a zero of  $f$  or not. Instead, this paper introduces a single numerical method, called *generalized regula falsi* (GRF) method to determine both zeros and extrema of a function. Consequently, it differs from the standard *regula falsi method* in that it is capable of finding any function zero in a given interval  $[A, B] \subset \mathbb{R}$  even when the Intermediate Value Theorem is not satisfied.

## 1. Introduction

Numerical methods such as *bisection method* [1, 2], *secant method* [3–6], *regula falsi method* [7, 8], and *Newton-Raphson method* [9, 10], amongst others, are known as zero-finding algorithms. They are used for bracketing a zero. We say that a zero is *bracketed* in the interval  $[A, B]$  if  $f(A)$  and  $f(B)$  have opposite signs; that is,  $f(A) \cdot f(B) < 0$ . In this paper, a zero of this sort is called a *crossing zero*, as illustrated in Figure 1(a).

Although this class of methods satisfying the Intermediate Value Theorem succeeds in finding most zeros (i.e., crossing zeros), they fail to bracket zeros that are also extrema, here called *extremal zeros*. We say that an extremum is *bracketed* in the interval  $[A, B]$  if  $f(A)$  and  $f(B)$ , but not their derivatives, have identical signs; that is,  $f(A) \cdot f(B) > 0$  and  $f'(A) \cdot f'(B) < 0$ . If  $f(A) > 0$  and  $f(B) > 0$ , “go downhill, taking steps of increasing size, until your function starts back uphill” [4]. At this point, we have only to check that such a minimum is a zero. Analogously, if  $f(A) < 0$  and  $f(B) < 0$ , basically, “go uphill, taking steps of increasing size, until your function starts back downhill” [4]. Then, we have only to check that such a maximum is a zero.

Despite the existence of algorithms for finding extrema (maxima and minima) of functions in the numerical analysis

literature, there is no single iterative formula to find both crossing and extremum zeros of a real function. Such a method is described in the next section and is called *generalized regula falsi method*.

## 2. Generalized Regula Falsi Method

The generalized regula falsi (GRF) method is based on the ratio of similar triangles. Its main novelty is that it can be used to compute both zeros and extrema through a single interpolation formula.

Let us consider the interval  $[A, B]$ , as illustrated in Figures 1(a) and 1(b), being  $|f(A)|$  and  $|f(B)|$  the absolute values of the function at  $A$  and  $B$ , respectively. The endpoints of the interval  $[A, B]$  work as two distinct estimates for a zero of  $f(x) = 0$ . The next estimate  $X$  is the intersection point of the straight-line segments  $\overline{(A, 0)(B, 0)}$  and  $\overline{(A, |f(A)|)(B, |f(B)|)}$ , which is given by

$$X = A + \frac{|f(A)|}{|f(A)| + |f(B)|} \cdot (B - A). \quad (1)$$

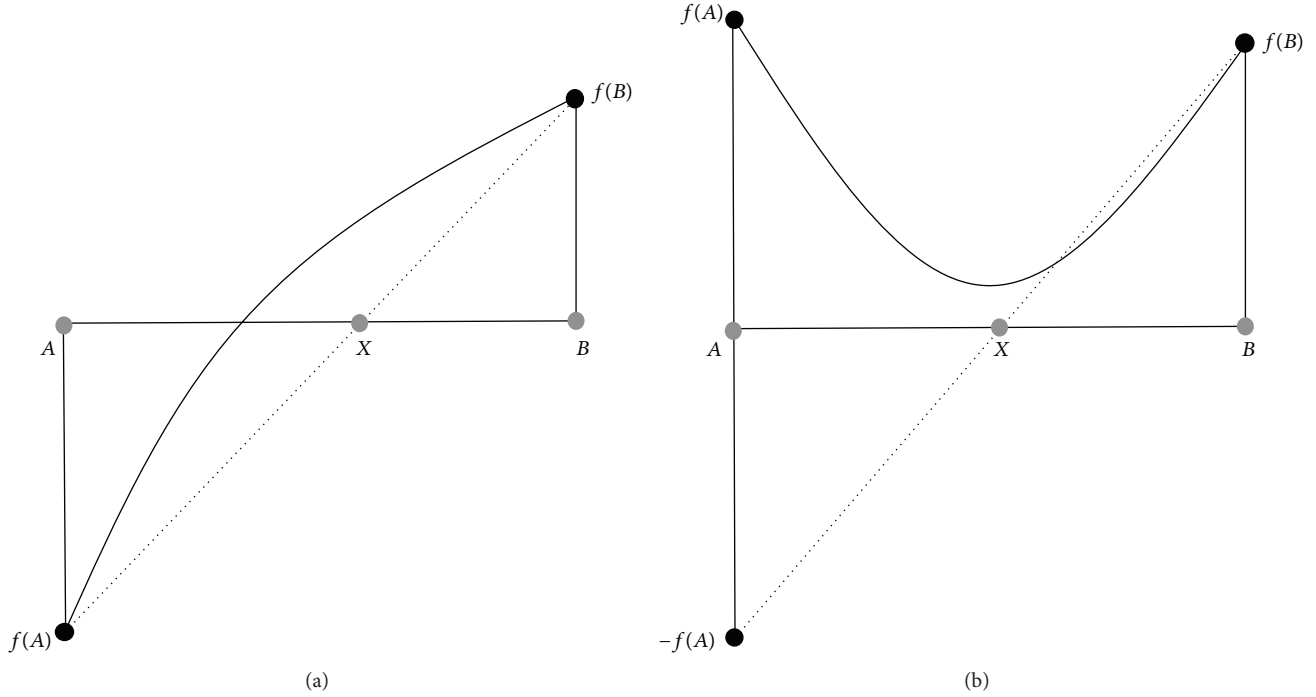


FIGURE 1: The generalized regula falsi method for (a) zeros and (b) extrema.

As known, the vectorial equation of the line that passes through the points  $A$  and  $B$  is given by

$$X(t) = A + t \cdot (B - A) \quad (2)$$

with  $A = X(0)$  and  $B = X(1)$ . Thus, the next estimate  $X$  occurs at  $t = |f(A)|/(|f(A)| + |f(B)|)$ , a result due to the following equality of triangles:

$$\frac{|f(A)|}{t} = \frac{|f(B)|}{1-t}. \quad (3)$$

Remarkably, as explained below, the iterative formula (1) applies to both zeros and extrema.

**2.1. Zeros.** The computation of a zero is illustrated in Figure 1(a). Let us assume that  $f(A) \cdot f(B) < 0$ , with  $f(A) < 0$ . By replacing  $|f(A)| = -f(A)$  and  $|f(B)| = f(B)$  into (1), we obtain

$$X = A - \frac{f(A)}{f(B) - f(A)} \cdot (B - A), \quad (4)$$

that is, the interpolation formula of the well-known regula falsi method [1]. It is used to determine crossing zeros.

**2.2. Extrema.** The computation of a local extremum is done exactly in the same way for minima and maxima. Let us first consider a local minimum as illustrated in Figure 1(b). In this case,  $f(A) \cdot f(B) > 0$ , with  $f(A) > 0$ . By replacing  $|f(A)| = f(A)$  and  $|f(B)| = f(B)$  into (1), we obtain

$$X = A + \frac{f(A)}{f(B) + f(A)} \cdot (B - A), \quad (5)$$

that is, the next estimate  $X$  of a local minimum.

Let us now consider that a reflection takes place across the  $x$ -axis in Figure 1(b). In this case, we have a function with a local maximum; that is,  $f(A) \cdot f(B) > 0$ , with  $f(A) < 0$ . Again, by replacing  $|f(A)| = -f(A)$  and  $|f(B)| = -f(B)$  into (1), we obtain the next estimate  $X$  through (5).

Thus, formula (5) works equally well for local minima and maxima, independently of whether  $f(A)$  and  $f(B)$  are either positive or negative. In fact, looking at (4) and (5), we see that it is enough to change the sign of either  $f(A)$  or  $f(B)$  to determine the next estimate  $X$  of an extremum through the regula falsi technique.

However, the condition  $f(A) \cdot f(B) > 0$  only indicates that there may be an extremum between  $A$  and  $B$ . We have to evaluate the finite difference  $d(x) = (f(x+h) - f(x))/h$ , with an infinitesimal  $h$ , at  $A$  and  $B$  and test whether they have opposite signs, that is,  $d(A) \cdot d(B) < 0$ , to guarantee that an extremum exists in  $[A, B]$ . We use the finite difference—the discrete counterpart of the derivative—of a function at a given point for two main reasons.

- (i) *Consistency*: the regula falsi method does not use derivatives. So, the generalized regula falsi method does not use them for consistency either.
- (ii) *Differentiability*: there are zeros and extrema at which derivatives do not exist. For example,  $f(x) = |x|^{1/3}$  in Figure 3(f) has a cusp at  $x = 0$  that is also a minimum. Derivatives cannot be used in this case because the program would break down.

Therefore, we have to check whether the finite difference  $d(X)$  of the estimate  $X$  is very close to zero. If so, we have an extremum at  $X$  approximately. Besides, if  $f(X) \approx 0$ , with an accuracy of  $10^{-6}$ , then  $X$  is a zero.

In short, in the literature, we find numerical methods for calculating zeros and extrema. The main novelty of the method introduced in this paper is its ability to compute zeros and extrema by means of a single interpolation formula (cf. formula (1)), that is, using the same formula.

**2.3. Convergence Analysis.** The convergence of the GRF method is guaranteed by the following theorem.

**Theorem 1.** Let  $f$  be a continuous real function on  $[A, B] \subset \mathbb{R}$  and  $f(A) \cdot f(B) < 0$  or  $d(A) \cdot d(B) < 0$ . If  $P$  is the only zero (i.e.,  $f(P) = 0$ ) or extremum (i.e.,  $d(P) = 0$ ) in  $[A, B]$ , then the GRF method generates a sequence  $\{P_n\}_{n=1}^{+\infty}$  converging to  $P$  for two initial approximations  $P_1 = A$  and  $P_2 = B$ .

*Proof.* Without loss of generality, that is equivalent to prove the existence of a sequence  $\{I_n\}_{n=1}^{+\infty}$  of intervals  $I_n = [P_n, P_{n+1}]$  converging to  $P$ , with  $P \in I_n$  for all  $n$ .

Let  $l(I_n)$  be the length of the interval  $I_n$  so that  $l(I_1) = |P_2 - P_1|$ ,  $l(I_2) = |P_3 - P_2|$ , and

$$\frac{l(I_2)}{l(I_1)} = \frac{|P_3 - P_2|}{|P_2 - P_1|}. \quad (6)$$

Now, from (1) we obtain

$$\begin{aligned} \frac{l(I_2)}{l(I_1)} &= \frac{|P_2 + (|f(P_2)| / (|f(P_1)| + |f(P_2)|)) \cdot (P_1 - P_2) - P_2|}{|P_2 - P_1|} \\ &= \frac{|f(P_2)|}{|f(P_1)| + |f(P_2)|} \end{aligned} \quad (7)$$

or

$$l(I_2) = l(I_1) \cdot \frac{|f(P_2)|}{|f(P_1)| + |f(P_2)|}. \quad (8)$$

In general, we can write

$$l(I_n) = l(I_{n-1}) \cdot \frac{|f(P_n)|}{|f(P_{n-1})| + |f(P_n)|}. \quad (9)$$

Taking into account that

$$0 < \frac{|f(P_n)|}{|f(P_{n-1})| + |f(P_n)|} < 1 \quad (10)$$

we have  $l(I_n) < l(I_{n-1})$  for all  $n$ . Thus, the sequence  $\{l(I_n)\}_{n=1}^{+\infty}$  is monotone decreasing. In addition, it is bounded because  $0 \leq l(I_n) \leq |B - A|$ . So, by the Monotone Convergence Theorem we can conclude that  $\{l(I_n)\}_{n=1}^{+\infty}$  is convergent. Taking into account that  $P \in I_n$  for all  $n$ , we conclude that the GRF method converges to  $P$ , being  $P$  a zero or an extremum of  $f$ .  $\square$

It is known that the zero-finding regula falsi method converges linearly [9]. Intuitively, we expect that the GRF method also converges linearly because it uses the same interpolation formula and bracketing strategy for both zeros and extrema.

**Theorem 2.** The GRF method has linear convergence.

*Proof.* Let  $P_0, P_1, P_2, \dots, P_n, \dots$  be a sequence of approximations to either a zero or an extremum  $P$  produced by the GRF method, where  $\lim_{n \rightarrow \infty} P_n = P$ . Let  $\epsilon_n = P_n - P$  be the error in the  $n$ th iterate.

By (1), the successive iterations are

$$P_{n+1} = P_n + \frac{|f(P_n)|}{|f(P_n)| + |f(P_{n-1})|} \cdot (P_{n-1} - P_n). \quad (11)$$

To determine the speed of convergence subtract  $P$  and divide by  $\epsilon_n = P_n - P$  to get

$$\begin{aligned} \frac{\epsilon_{n+1}}{\epsilon_n} &= 1 + \frac{|f(P_n)|}{|f(P_n)| + |f(P_{n-1})|} \cdot \frac{(P_{n-1} - P) - (P_n - P)}{P_n - P} \\ &= 1 + \left( \frac{P_{n-1} - P}{P_n - P} - 1 \right) \cdot \frac{|f(P_n)|}{|f(P_n)| + |f(P_{n-1})|} \\ &= 1 + \left( \frac{\epsilon_{n-1}}{\epsilon_n} - 1 \right) \cdot \frac{|f(P_n)|}{|f(P_n)| + |f(P_{n-1})|}. \end{aligned} \quad (12)$$

Since  $\lim_{n \rightarrow \infty} (\epsilon_{n+1}/\epsilon_n) = \lim_{n \rightarrow \infty} (\epsilon_n/\epsilon_{n-1})$  and  $\lim_{n \rightarrow \infty} P_{n-1} = \lim_{n \rightarrow \infty} P_n = P$ , we obtain

$$\lim_{n \rightarrow \infty} \frac{\epsilon_{n+1}}{\epsilon_n} = 1 + \left( \frac{1}{\lim_{n \rightarrow \infty} (\epsilon_{n+1}/\epsilon_n)} - 1 \right) \cdot \frac{1}{2} \quad (13)$$

from where we conclude that  $\lim_{n \rightarrow \infty} (\epsilon_{n+1}/\epsilon_n) = 1$ . This concludes the proof.  $\square$

**2.4. GRF Implementation.** The C function that implements the generalized regula falsi method returns either a zero or an extremum bracketed in a given interval  $[A, B]$  (see Algorithm 1).

Note that, for the sake of convenience, we have used the setting  $h = 1.0E - 7$  for evaluating finite differences, but this is clearly prone to cancellation errors. A better choice to  $h$  would be  $\sqrt{\epsilon}X$ , where  $\epsilon$  denotes the machine precision, which is typically of the order  $2.2 \times 10^{-16}$  for doubles on a 32-bit computer.

### 3. Moving GRF Method

As illustrated in Figure 2(a), the regula falsi method may converge slowly to a zero, spending many iterations in pulling distant bounds closer. For example, the crossing zero of the function  $f_4(x) = \tan(x)^{\tan(x)} - 10^3$  on  $[1.3, 1.4]$  (Figure 3(d)) was found after 222 iterations. Likewise, the crossing zero of the function  $f_5(x) = xe^x - 10$  on  $[-10.0, 10.0]$  (Figure 3(e)) was only found after 999 iterations. The reason behind the slow convergence to some zeros is due to the fact that GRF method always retains a fixed endpoint of interval  $[A, B]$ .

The leading idea of speeding up the convergence of the GRF method is then moving the fixed point as well. To achieve that we replace the fixed endpoint of the interval

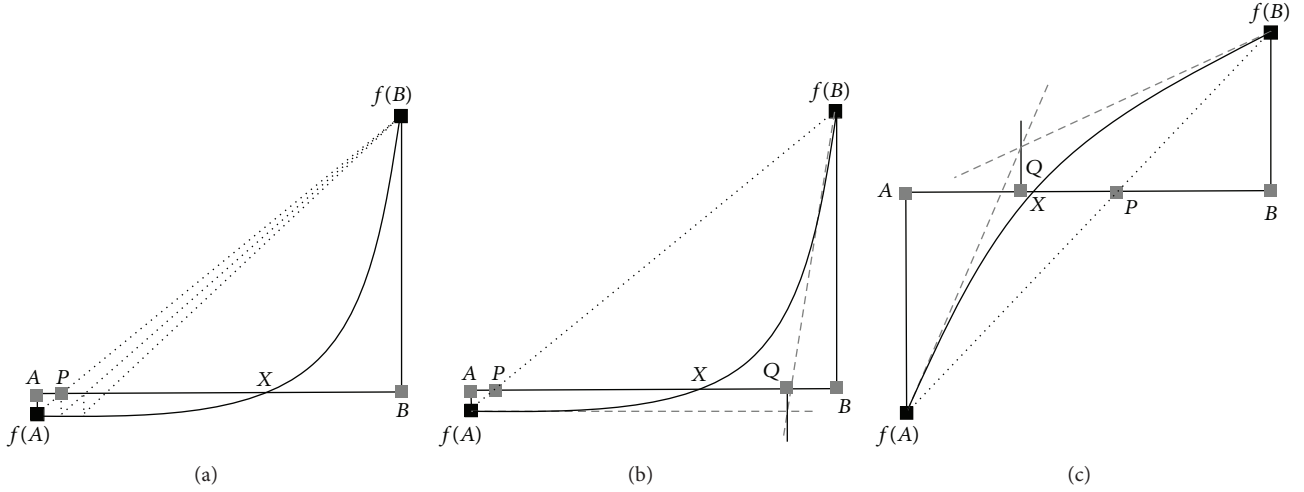


FIGURE 2: The mGRF method.

```

#include <math.h>
#define ACY 1.0E-6    accuracy for estimates, function, and finite difference values
#define h 1.0E-7      infinitesimal for finite differences
double GRF(float (*func)(double), double A, double B, int n)
{
    double X, fX, dX, A, fA, dA, B, fB, dB, a, fa, da, b, fb, db;
    while ( (fabs(B-A)>ACY) && (--n>0) )
    {
        double fA=(*func)(A);
        double dA=(((*func)(A+h)-(*func)(A))/h);
        if ((fabs(fA)<ACY) || (fabs(dA)<ACY)) // zero or extremum
            return A;
        fB=(*func)(B);
        double dB=(((*func)(B+h)-(*func)(B))/h);
        if ((fabs(fB)<ACY) || (fabs(dB)<ACY)) // zero or extremum
            return B;
        X=A+fabs(fA)/(fabs(fA)+fabs(fB))*(B-A);
        fX=(*func)(X);
        dX=(((*func)(X+h)-(*func)(X))/h);
        if ((fabs(fX)<ACY) || (fabs(dX)<ACY)) // zero or extremum
            return X;
        if ((fA*fX<0) || (dA*dX<0))
            B = X;
        else
            if ((fX*fB<0) || (dX*dB<0))
                A = X;
    }
}

```

ALGORITHM 1

$[A, B]$  by the abscissa of a point  $Q$  that results from intersecting the tangents at the bound points  $(A, f(A))$  and  $(B, f(B))$  as shown in Figures 2(b) and 2(c). That is, the abscissa  $Q$  is the solution  $x$  of the following system of equations:

$$\begin{aligned} y - f(A) &= d(A) \cdot (x - A) \\ y - f(B) &= d(B) \cdot (x - B) \end{aligned} \quad (14)$$

which yields

$$Q = \frac{f(B) - f(A) + A \cdot d(A) - B \cdot d(B)}{d(A) - d(B)}, \quad (15)$$

where  $d(A)$  and  $d(B)$  are the finite differences at  $A$  and  $B$ , respectively. Note that the estimate  $Q$  may be closer to the solution than the estimate produced by Newton's method that results from the intersection between a single tangent and the  $x$ -axis (Algorithm 2).

```

double mGRF(float (*func)(double), double A, double B, int n)
{
    double X, fX, dX, A, fA, dA, B, fB, dB, a, fa, da, b, fb, db;
    while ( (fabs(B-A)>ACY) && (--n>0) )
    {
        fA=(*func)(A);
        dA=((*func)(A+h)-(*func)(A))/h;
        if ((fabs(fA)< ACY) || (fabs(dA)< ACY)) // zero or extremum
            return A;
        fB=(*func)(B);
        dB=((*func)(B+h)-(*func)(B))/h;
        if ((fabs(fB)< ACY) || (fabs(dB)< ACY)) // zero or extremum
            return B;
        X=A+fabs(fA)/(fabs(fA)+fabs(fB))*(B-A);
        fX=(*func)(X);
        dX=((*func)(X+h)-(*func)(X))/h;
        if ((fabs(fX)< ACY) || (fabs(dX)< ACY)) // zero or extremum
            return X;
        if ((fA*fX<0) || (dA*dX<0))
        {
            B = X;
            fB=(*func)(B); dB=((*func)(B+h)-(*func)(B))/h;
            a =(fB-fA+A*dA-B*dB)/(dA-dB); // moving fixed point
            fa=(*func)(a); da=((*func)(a+h)-(*func)(a))/h;
            if ((a>A) && (a<B)) // bracketing moving point
                if ((fa*fA<0) || (da*dA<0))
                    B=a;
            else
                A=a;
        }
        else if ((fX*fB<0) || (dX*dB<0))
        {
            A = X;
            fA=(*func)(A); dA=((*func)(A+h)-(*func)(A))/h;
            b =(fA-fB+B*dB-A*dA)/(dB-dA); // moving fixed point
            fb=(*func)(b); db=((*func)(b+h)-(*func)(b))/h;
            if ((b>A) && (b<B)) // bracketing moving point
                if ((fb*fB<0) || (db*dB<0))
                    A=b;
            else
                B=b;
        }
    }
}

```

ALGORITHM 2

So, the C function that implements the moving generalized regula falsi (mGRF) method differs from GRF in that we have to include statements concerning the calculation of the moving fixed point  $Q$  that results from intersecting tangents at  $A$  and  $B$ , as well as some statements to make sure that  $Q$  remains bracketed in the interval.

**3.1. Convergence of mGRF Method.** The convergence of the mGRF method is stated by the following theorem.

**Theorem 3.** *Let  $f$  be a continuous real function on  $[A, B] \subset \mathbb{R}$  and  $f(A) \cdot f(B) < 0$  or  $d(A) \cdot d(B) < 0$ . If  $X \in [A, B]$  is such that  $f(X) = 0$  or  $d(X) = 0$ , then the mGRF method generates two*

*sequences  $\{A_i\}_{i=1}^{+\infty}$  and  $\{B_j\}_{j=1}^{\infty}$  converging to  $X$  for two initial approximations  $A_1 = A$  and  $B_1 = B$ .*

*Proof.* The mGRF method uses two preliminary estimates  $A$  and  $B$  (i.e., the endpoints of the interval  $[A, B]$ ) to generate two sequences of estimates,  $\{A_i\}$  and  $\{B_j\}$ , that supposedly march towards the solution  $X$ .

Without loss of generality, we can say that the GRF method generates the sequence  $\{A_i\}_{i=1}^{\infty}$  converging to  $X$ , as proved by Theorem 1.

It remains to prove that the sequence  $\{B_j\}_{j=1}^{\infty}$  generated by the computation of  $Q$  (cf. formula (15)) also converges to

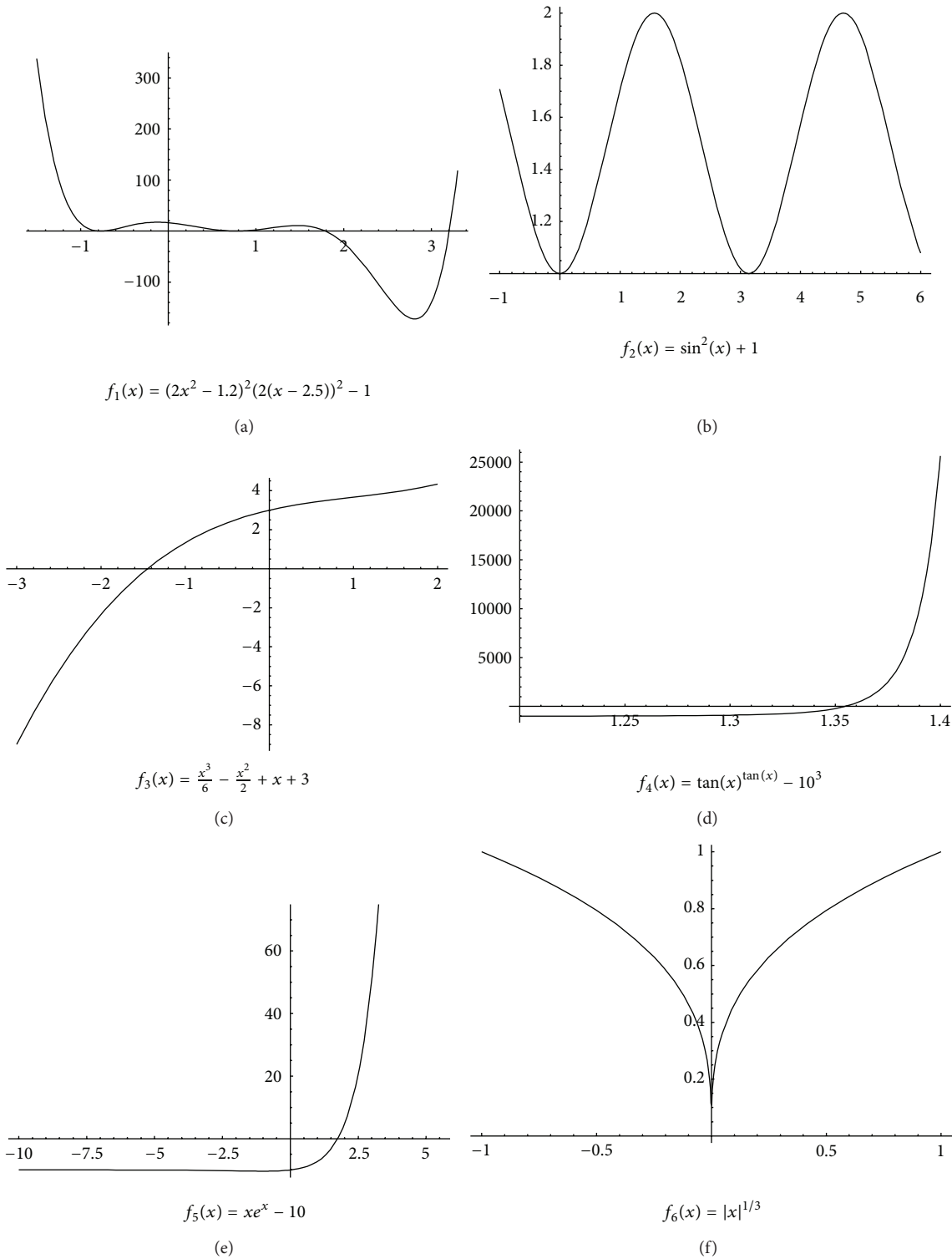


FIGURE 3: Graphs of six real functions: (a)  $f_1$  has 2 local maxima, 1 local minimum, 2 zeroed minima, and 2 crossing zeros; (b)  $f_2$  has an alternate sequence of local maxima and zeroed minima; (c)  $f_3$  has a single crossing zero; (d)  $f_4$  has a single crossing zero; (e)  $f_5$  has also a single crossing zero; (f)  $f_6$  has a zeroed minimum that is a cusp.



$X$ . Without loss of generality, let us assume that  $d(B_{j-2}) > 0$  and  $d(B_{j-1}) < 0$  so that  $|d(B_{j-2})| = d(B_{j-2})$  and  $|d(B_{j-1})| = -d(B_{j-1})$ . Now, formula (15) can be rewritten as follows:

$$B_j = \frac{f(B_{j-1}) - f(B_{j-2}) + B_{j-2} \cdot |d(B_{j-2})| + B_{j-1} \cdot |d(B_{j-1})|}{|d(B_{j-2})| + |d(B_{j-1})|} \quad (16)$$

or, equivalently

$$B_j = B_{j-1} + t \cdot (B_{j-2} - B_{j-1}) \quad (17)$$

which is the vectorial equation of the line defined by the points  $B_{j-1}$  and  $B_{j-2}$ , being

$$t = \frac{f(B_{j-1}) - f(B_{j-2})}{(|d(B_{j-1})| + |d(B_{j-2})|) \cdot (B_{j-2} - B_{j-1})} + \frac{|d(B_{j-2})|}{|d(B_{j-1})| + |d(B_{j-2})|}. \quad (18)$$

Let  $l(I_j)$  be the length of the interval  $I_j$  so that  $l(I_j) = |B_j - B_{j-1}|$ ,  $l(I_{j-1}) = |B_{j-1} - B_{j-2}|$ . Now, from (17) we obtain

$$\begin{aligned} \frac{l(I_j)}{l(I_{j-1})} &= \frac{|B_j - B_{j-1}|}{|B_{j-1} - B_{j-2}|} \\ &= \frac{|B_{j-1} + t \cdot (B_{j-2} - B_{j-1}) - B_{j-1}|}{|B_{j-1} - B_{j-2}|} = t. \end{aligned} \quad (19)$$

But, as said before, the new estimate  $B_j$  is only taken into account in the convergence process if it lies in the interval  $]B_{j-2}, B_{j-1}[$ ; otherwise,  $B_j$  equals to the estimate  $A_j \in ]B_{j-2}, B_{j-1}[$  determined by the GRF method; that is,  $B_j = A_j$ . From (17) and (19), this implies that  $0 < t < 1$  and  $l(I_j) < l(I_{j-1})$  for all  $j$ . Thus, the sequence  $\{l(I_j)\}_{j=1}^{+\infty}$  is monotone decreasing. In addition, it is bounded because  $0 < l(I_j) \leq |B - A|$ . So, by the Monotone Convergence Theorem we can conclude that  $\{l(I_j)\}_{j=1}^{+\infty}$  is convergent. Taking into account that  $X \in I_j$  for all  $j$ , we can also conclude that the mGRF method converges because the sequence  $\{B_j\}_{j=1}^{\infty}$  converges to  $X$ , being  $X$  a zero or an extremum of  $f$ .  $\square$

## 4. Experimental Results

We have carried out a number of the convergence tests on computer in order to assess the convergence of the superlinear version of generalized regula falsi (GRF) method. For that purpose, we have used a MacBook Pro laptop powered by a 2.33 GHz Intel Core 2 Duo processor, with 3 GB 667 MHz DDR2 SDRAM, running Mac OS X operating system (version 10.6.8).

**4.1. Functions, Zeros, and Extrema.** We have used a number of real analytic functions in testing of the method convergence, including algebraic (or polynomial) and transcendental functions, some of which are depicted in Figure 3; for example,  $f_1$  and  $f_3$  are algebraic functions, but not  $f_2$  and  $f_4$  which are transcendental functions. These functions have been chosen because they cover most types of zeros and (local) extrema, namely, as shown in what follows.

- (i) *Crossing Zeros (cZ)*: a crossing zero is a point at which the function graph crosses the  $x$ -axis. We find 5 crossing zeros in Figure 3. More specifically,  $f_1$  has 2 crossing zeros in the interval  $[1.5, 3.5]$ ,  $f_3$  has 1 crossing zero in the interval  $[-2.0, -1.0]$ ,  $f_4$  has 1 crossing zero in the interval  $[1.3, 1.4]$ , and  $f_5$  has 1 crossing zero in the interval  $[1.5, 2.5]$ .
- (ii) *Minima (m)*: we find 1 minimum in Figure 3;  $f_1$  has 1 local minimum in the interval  $[2.0, 3.0]$ .
- (iii) *Maxima (M)*: we have 4 maxima in Figure 3;  $f_1$  has 2 local maxima in the interval  $[-1.0, 2.0]$ , while  $f_2$  possesses 2 local maxima, the first in the interval  $[1.0, 2.0]$  and the second in the interval  $[4.0, 5.0]$ .
- (iv) *Zeroed Minima (Zm)*: a local zeroed minimum is a point at which the function takes on the value 0 and has a local minimum; that is, the function graph touches but does not cross the  $x$ -axis. We have 5 zeroed minima in Figure 3. In particular,  $f_1$  has 2 local zeroed minima in the interval  $[-1.0, 1.0]$ ,  $f_2$  also possesses 2 local zeroed minima, the first at  $x = 0$  and the second in the interval  $[3.0, 4.0]$ , whereas  $f_6$  has 1 local minimum at  $x = 0$  that is a cusp.
- (v) *Zeroed Maxima (ZM)*: a local zeroed maximum is a point at which the function takes on the value 0 and has a local maximum.

For the purpose of testing, we carried out the convergence experiments for crossing zeros and extrema separately, as described below.

**4.2. Convergence to Crossing Zeros.** The experimental results shown in Table 1 put in evidence how different GRF and mGRF methods are in terms of convergence to crossing zeros. According to Theorem 2, the GRF has linear convergence. But, following the results shown in Figure 4(top), mGRF seems to have superlinear convergence.

As known, the GRF converges very slowly to a crossing zero when the slope of the function graph changes very fast near to such a zero, as it is the case of those shown in Figures 3(d)-3(e). For example, in Figure 3(d), the crossing zero of  $f_4$  on the narrow interval  $]1.3, 1.4[$  was found after 222 iterations, while that one of  $f_5$  in Figure 3(e) was found after 999 iterations, though in the bigger interval  $[-10, 10]$  (cf. Table 1).

As explained above, the mGRF is faster than GRF because both endpoints of the bracketing interval converge to the solution, while GRF keeps one of the endpoints fixed. The mGRF convergence rate seems to be at least quadratic because the estimates produced by (15) are usually closer to

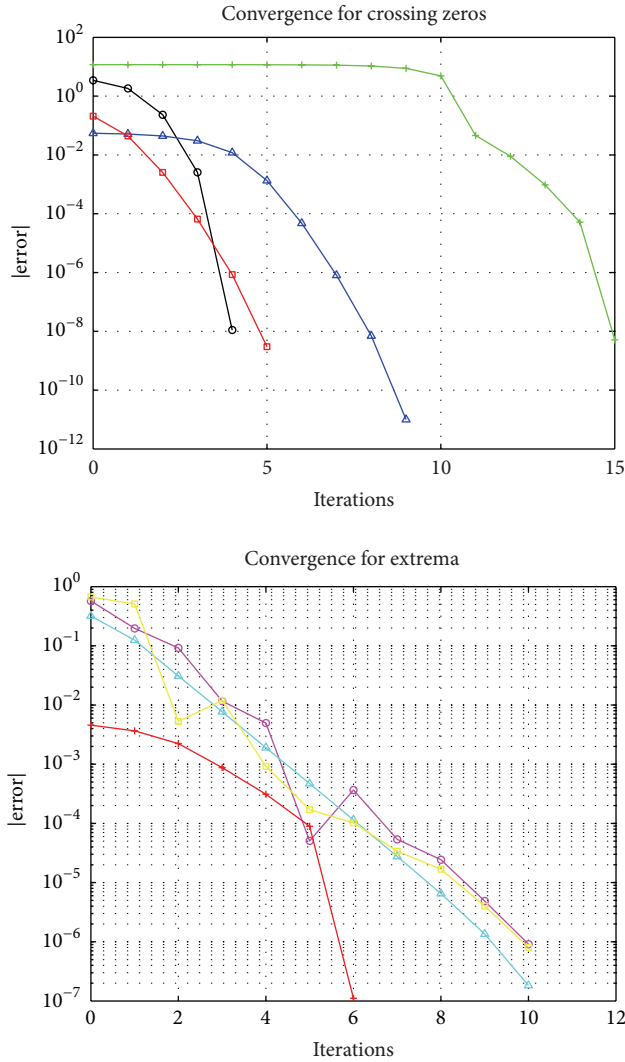


FIGURE 4: Semilogarithmic graphs relative to convergence to zeros and extrema.

TABLE 1: Experimental convergence results for zeros using GRF and mGRF methods.

Function	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$
Estimate type	cZ	—	cZ	cZ	cZ	—
Interval	[3.0, 3.3]	—	[-3.0, 2.0]	[1.3, 1.4]	[-10, 10]	—
GRF iterations	11	—	19	222	999	—
mGRF iterations	4	—	5	9	15	—

TABLE 2: Experimental convergence results for extrema using GRF and mGRF methods.

Function	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$
Estimate type	m	M	—	—	—	Zm
Interval	[2.0, 3.0]	[1, 1.8]	—	—	—	[-0.5, 1.5]
GRF iterations	23	19	—	—	—	18
mGRF iterations	10	10	—	—	—	10

the solution than those produced by Newton's method. This is confirmed by the error semilogarithmic graphs versus number of iterations shown in Figure 4(top). The convergence

curve in black concerns the crossing zero of  $f_1$  in the interval [3.0, 3.3], while the red, blue, and green curves represent the convergence to zero crossings of  $f_3$ ,  $f_4$ , and  $f_5$  (cf. Table 1).

**4.3. Convergence to Extrema.** Unlike the numerical methods found in the literature, the GRF and mGRF methods are capable of computing both zeros and extrema of a real function through a single interpolation formula.

However, as shown in Figure 4(bottom), the convergence rate for extrema is lower than for crossing zeros and tends to be linear. This is explained by the fact that some estimates of the lateral sequence  $\{A_i\}_{i=1}^{+\infty}$  jump to the sequence  $\{B_j\}_{j=1}^{\infty}$ , and vice versa.

The data shown in Table 2 concerns some extrema of  $f_1$ ,  $f_2$ , and  $f_6$ , whose convergence curves are plotted in Figure 4(bottom); the red curve concerns the minimum of  $f_1$  in the interval [2.0, 3.0], the magenta curve has to do with the convergence to the maximum  $f_2$  in the interval [1, 1.8], and the cyan curve refers to zeroed minimum (cusp) of  $f_6$  in the interval [-0.5, 1.5]. This latter case shows that the method proposed in this paper is also capable of calculating a zero at which  $f$  is not differentiable. These semilogarithmic graphs suggest that mGRF has linear convergence for extrema. Nevertheless, the red convergence curve associated to the local maximum of  $f_1$  in the interval [-0.9, -0.4] may suggest that we can possibly obtain superlinear convergence for extrema too, but that would require redesigning the algorithm somehow.

## 5. Conclusions

The experimental results presented in the previous section show us the following.

- (i) It is feasible to have a single numerical method to determine both zeros and extrema of a real function.
- (ii) The algorithm (m)GRF works well even under conditions of lack of differentiability. So, singular zeros such as cusps and corners can be also computed.
- (iii) It is also feasible to compute inflection points. For that, the algorithm must include the computation of the second finite differences.

This algorithm has been designed and implemented to solve a difficult problem in computer graphics and geometric modeling, which has to do with rendering sign-invariant components of implicit curves. For example, the circle described by function  $f(x, y) = (x^2 + y^2 - 1)^2 = 0$  in  $\mathbb{R}^2$  cannot be sampled and, thus, rendered by algorithms that make usage of the Intermediate Value Theorem because the function is positive everywhere, with the exception of the curve points on which the function takes on the value 0; that is, all the circle points are extrema.

## Acknowledgment

The authors would like to acknowledge the helpful suggestions of anonymous reviewers.



## References

- [1] S. D. Conte and C. de Boor, *Elementary Numerical Analysis: An Algorithmic Approach*, McGraw-Hill, New York, NY, USA, 1965.
- [2] G. Corliss, "Which root does the bisection algorithm find?" *SIAM Review*, vol. 19, no. 2, pp. 325–327, 1977.
- [3] R. P. Brent, "An algorithm with guaranteed convergence for finding a zero of a function," *The Computer Journal*, vol. 14, pp. 422–425, 1971.
- [4] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, Cambridge, UK, 1992.
- [5] S. A. Stage, "Comments on an improvement to the Brent's method," *International Journal of Experimental Algorithms*, vol. 4, no. 1, pp. 1–16, 2013.
- [6] Z. Zhang, "An improvement to the Brent's method," *International Journal of Experimental Algorithms*, vol. 2, no. 1, pp. 21–26, 2011.
- [7] A. Ralston and P. Rabinowitz, *A First Course in Numerical Analysis*, International Series in Pure and Applied Mathematics, McGraw-Hill, New York, NY, USA, 1978.
- [8] C. F. Ridders, "A new algorithm for computing a single root of a real continuous function," *Transactions on Circuits and Systems*, vol. 26, no. 11, pp. 979–980, 1979.
- [9] J. Kou, Y. Li, and X. Wang, "On modified Newton methods with cubic convergence," *Applied Mathematics and Computation*, vol. 176, no. 1, pp. 123–127, 2006.
- [10] J. R. Sharma, "A family of Newton-like methods based on an exponential model," *International Journal of Computer Mathematics*, vol. 84, no. 3, pp. 297–304, 2007.

