BIT 32 (1992), 334–344.

SOME EFFICIENT METHODS FOR ENCLOSING SIMPLE ZEROS OF NONLINEAR EQUATIONS

GÖTZ E. ALEFELD and FLORIAN A. POTRA[1]

Institut für Angewandte Mathematik
Universität Karlsruhe
Kaiserstrasse 12, Postfach 6380
D-W-7500 Karlsruhe, Germany

Department of Mathematics
University of Iowa
Iowa City, IA 52242
USA

Abstract.

In the present paper we propose three new methods for computing sequences of enclosing intervals for a zero of a real function without convexity assumptions.

The new methods have been tested on a series of published examples. The numerical experiments show that our methods are comparable in terms of efficiency with the well-known algorithms of Dekker and Brent.

AMS Subject Classifications: 65H05.

Keywords: Nonlinear equation, iteration method, high order of convergence.

1. Introduction.

Let $f:[a,b]\to\mathbb{R}$ be a continuous function which has a simple zero x^* in the interval $[a,b]$. We consider the problem of constructing a sequence of intervals $\{[a_n,b_n]\}_{n=0}^{\infty}$, such that:

$$x^*\in[a_{n+1},b_{n+1}]\subset[a_n,b_n]\subset\ldots\subset[a_0,b_0]=[a,b], \tag{1}$$

$$\lim_{n\to\infty}(b_n-a_n)=0. \tag{2}$$

In case f is monotone and convex on $[a,b]$ this may be accomplished by the classical Newton-Fourier procedure (see [7, p. 248]). The sequence of diameters $\{(b_n-a_n)\}_{n=0}^{\infty}$ of the enclosing intervals produced by this method converges Q-quadratically to zero. One step of this iterative procedure requires two function and one derivative evaluation so that its asymptotic efficiency index in the sense of Ostrowski [7] is $2^{1/3}=1.2599\ldots$. J. W. Schmidt [9] has proposed an iterative algorithm requiring the same amount of work and for which the sequence of diameters is Q-cubically convergent to zero. The efficiency index of the procedure is

[1] The present paper was written when this author was visiting the University of Karlsruhe. He would like to acknowledge the support provided by the University of Karlsruhe.

Received March 1991. Revised September 1991.

equal to $3^{1/3} = 1.4422\ldots$, He also proposed an algorithm with R-convergence order $1 + \sqrt{2} = 2.4142\ldots$ requiring only two function evaluations per step and therefore having the efficiency index equal to $(1 + \sqrt{2})^{1/2} = 1.5537\ldots$. Alefeld and Potra [2] suggested some modifications for the above-mentioned methods of J. W. Schmidt and showed that the Q-convergence order of the second method of J. W. Schmidt is also equal to $1 + \sqrt{2}$.

The convexity assumption required by the above-mentioned procedures is rather restrictive. Some interval versions of Newton's method and the secant method can successfully handle the nonconvex case, but the use of interval arithmetic and interval extensions of derivatives may be quite expensive (see Alefeld-Herzberger [1], and Alefeld-Potra [3]). Very efficient equation solvers, used in standard software packages (among the best known are the methods of Dekker [5] and Brent [4]), also produce enclosing intervals that satisfy (1). However with most such solvers (2) is not verified in an asymptotic sense. For example with Dekker's method [5], the diameters $b_n - a_n$ may remain greater than a relatively large positive quantity until the last iteration when a "δ-step" is taken and the diameter of the last interval is made less than a specified tolerance δ.

In the present paper we propose three new methods which produce sequences of enclosing intervals satisfying both (1) and (2) without any convexity assumptions. The first method requires asymptotically two function values per step. The diameters of the enclosing intervals are Q-quadratically convergent to zero so that the efficiency index of the method is $\sqrt{2} = 1.4142\ldots$.

The second method requires asymptotically three function values per step and the Q-convergence order of the sequence of diameters is four. Hence its efficiency index is $4^{1/3} = 1.5874\ldots$. Our third method uses three function values at each step and has R-convergence order $(3 + \sqrt{13})/2 = 3.3027\ldots$. The corresponding efficiency index is $1.4892\ldots$.

The efficiency index is an asymptotic notion, and in practical applications methods with lower efficiency index may outperform methods with higher efficiency index. This is due to the fact that in practice we do not produce an infinite sequence of intervals, but stop the iteration as soon as, let us say, the diameter of the current enclosing interval becomes less than a prescribed tolerance δ.

For a given higher order method and given stopping criterion, examples can be constructed on which the respective method is outperformed by the classical bisection method which is only linearly convergent. Therefore it is important to know the worst case scenario behaviour of a method in case of a practical stopping criterium. Suppose we use $b_n - a_n \leq \delta$ as a stopping criterion. In the worst case our first method produces intervals that staisfy $b_{n+1} - a_{n+1} \leq 0.5(b_n - a_n)$, the same as at each step of the bisection method.

Because up to three function values may be needed at each step, our first method may need three times more function values than the bisection method in the worst case. In the same sense our second method may use four times more function values than bisection, while our third method may use three times more function values

than bisection in the worst case. Let us also remark that while the first two methods take no bisection steps asymptotically, our third method always does.

2. The algorithms.

In what follows we will always assume that

$$f(a)f(b) < 0. \tag{3}$$

If f is continuous on the interval $[a, b]$ this assumption guarantees the existence of a zero of f in the interval $[a, b]$. Suppose that by some procedure a point $c \in [a, b]$ is produced. Then a new interval $[\bar{a}, \bar{b}] \subset [a, b]$ containing at least one zero of f can be constructed by calling the following subroutine:

subroutine *bracket*$(a, b, c, \bar{a}, \bar{b})$

if $f(c) = 0$ then print c and stop

if $f(a)f(c) < 0$ then $\bar{a} = a, \bar{b} = c$

if $f(c)f(b) < 0$ then $\bar{a} = c, \bar{b} = b$.

The classical bisection method consists in repeated calls to this subroutine with c the midpoint of a and b. If the values of f at a and b are known then each call requires one function evaluation. In the algorithms to be described in what follows, after obtaining c and calling *bracket*$(a, b, c, \bar{a}, \bar{b})$ one attempts to obtain a better enclosing interval by means of a point $\bar{c}$ obtained via a "double length secant step":

$$\begin{cases} \text{if } |f(\bar{a})| < |f(\bar{b})| & \text{then } u = \bar{a} \text{ else } u = \bar{b} \\ \bar{c} = u - 2f[\bar{a}, \bar{b}]^{-1} f(u) \end{cases} \tag{4}$$

In the formula above $f[x_1, x_2]$ denotes the divided difference of f at x_1, x_2 i.e.

$$f[x_1, x_2](x_1 - x_2) = f(x_1) - f(x_2). \tag{5}$$

The higher order divided differences of f are defined recursively by

$$f[x_1, x_2, \ldots, x_i, x_{i+1}](x_i - x_{i+1}) = f[x_1, \ldots, x_i] - f[x_1, \ldots, x_{i-1}, x_{i+1}]. \tag{6}$$

We observe that (5) and (6) uniquely define the divided differences in the one-dimensional case. This is no longer true in several dimensions. Also in the one-dimensional case the divided differences are symmetric functions of their arguments, i.e. $f[x, y] = f[y, x]$, $f[x, y, z] = f[y, z, x] = \ldots$.

Let us first note that the point $\bar{c}$ given by (4) always belongs to the interval $[\bar{a}, \bar{b}]$. In case $|f(\bar{a})| = |f(\bar{b})|$ it coincides with one of the endpoints of this interval. Also from (4) and (5) it follows that

$$f(\bar{c}) = f[\bar{c}, u](\bar{c} - u) + f(u) \tag{7}$$

$$= (1 - 2f[\bar{a}, \bar{b}]^{-1} f[\bar{c}, u]) f(u).$$

Suppose that f is continuously differentiable, and that the points $\bar{a}, \bar{b}, u, \bar{c}$ belong to a small neighbourhood of a simple zero x^* of f (i.e. $f(x^*) = 0$, $f'(x^*) \neq 0$.) Then $f[\bar{a}, \bar{b}] \approx f'(x^*) \approx f[\bar{c}, u]$ so that according to (7) $f(\bar{c}) \approx -f(u)$. This shows that if $[\bar{a}, \bar{b}]$ is a small enough interval containing a simple zero of f, then formula (4) can be used to obtain a better enclosure of this zero. In our first method we take for c the point obtained by the regula-falsi applied for the interval $[a, b]$. Then we take a "double length secant step" followed by a new bracket. If the length of the new enclosing interval is greater then $\mu_1(b - a)$ where μ_1 should be chosen equal to 0.25 then we call bracket again. More precisely we have the following.

Algorithm 1.

1.1. $c_n = a_n - f[a_n, b_n]^{-1} f(a_n)$

1.2. call *bracket*$(a_n, b_n, c_n, \bar{a}_n, \bar{b}_n)$

1.3. if $|f(\bar{a}_n)| < |f(\bar{b}_n)|$ then $u_n := \bar{a}_n$

else $u_n := \bar{b}_n$

1.4. $\bar{c}_n := u_n - 2f[\bar{a}_n, \bar{b}_n]^{-1} f(u_n)$

1.5. if $|\bar{c}_n - u_n| > 0.5(\bar{b}_n - \bar{a}_n)$ then $\hat{c}_n = 0.5(\bar{b}_n + \bar{a}_n)$

else $\hat{c}_n = \bar{c}_n$

1.6. call *bracket*$(\bar{a}_n, \bar{b}_n, \hat{c}_n, \hat{a}_n, \hat{b}_n)$

1.7. if $\hat{b}_n - \hat{a}_n < \mu_1(b_n - a_n)$ then $a_{n+1} = \hat{a}_n, b_{n+1} = \hat{b}_n$

else call *bracket*$(\hat{a}_n, \hat{b}_n, 0.5(\hat{a}_n + \hat{b}_n), a_{n+1}, b_{n+1})$

In the next algorithm the first two steps are the same as above. Then we consider the quadratic polynomial $p(a_n, b_n, c_n)(x)$ interpolationg f at a_n, b_n, c_n. Because $f(\bar{a}_n) f(\bar{b}_n) < 0$ and $\{\bar{a}_n, \bar{b}_n\} \subset \{a_n, b_n, c_n\}$ it follows that this polynomial has a change of sign in $[\bar{a}_n, \bar{b}_n]$ and therefore has a unique zero in $[\bar{a}_n, \bar{b}_n]$. This zero is used for a new bracketing and then we proceed similarly to 1.3–1.7. Summing up, we have

Algorithm 2.

2.1. $c_n = a_n - f[a_n, b_n]^{-1} f(a_n)$

2.2. call *bracket*$(a_n, b_n, c_n, \bar{a}_n, \bar{b}_n)$

2.3. $\bar{c}_n$ = the unique zero of $p(a_n, b_n, c_n)(x)$ belonging to the interval $[\bar{a}_n, \bar{b}_n]$

2.4. call *bracket*$(\bar{a}_n, \bar{b}_n, \bar{c}_n, \hat{a}_n, \hat{b}_n)$

2.5. if $|f(\hat{a}_n)| < |f(\bar{b}_n)|$ then $u_n := \hat{a}_n$

else $u_n := \hat{b}_n$

2.6. $\hat{c}_n := u_n - 2f[\hat{a}_n, \hat{b}_n]^{-1} f(u_n)$

2.7. if $|\hat{c}_n - u_n| > 0.5(\hat{b}_n - \hat{a}_n)$ then $\tilde{c}_n = 0.5(\hat{b}_n + \hat{a}_n)$

else $\tilde{c}_n = \hat{c}_n$

2.8. call *bracket*$(\hat{a}_n, \hat{b}_n, \tilde{c}_n, \tilde{a}_n, \tilde{b}_n)$

2.9. if $\tilde{b}_n - \tilde{a}_n < \mu_2(b_n - a_n)$ then $a_{n+1} = \tilde{a}_n, b_{n+1} = \tilde{b}_n$

else call *bracket*$(\tilde{a}_n, \tilde{b}_n, 0.5(\tilde{a}_n + \tilde{b}_n), a_{n+1}, b_{n+1})$

In our third algorithm the first new bracketing point c_n is chosen to be simply the midpoint of the interval $[a_n, b_n]$. This results in a slight degradation of the asymptotic order of convergence, from 4 to $0.5(3 + \sqrt{13}) = 3.3027\ldots$. However, as mentioned in the introduction our third algorithm requires at each step three function values so that it can be only 3 times slower than bisection while Algorithm 2 can be four times slower than bisection in the worst case.

Algorithm 3.

3.1. $c_n := 0.5(a_n + b_n)$

3.2.–3.6. the same as 2.2.–2.6.

call *bracket*$(\hat{a}_n, \hat{b}_n, \hat{c}_n, a_{n+1}, b_{n+1})$

3. Convergence.

In what follows we study some convergence properties of the algorithms 1, 2 and 3 described in the previous section. The first theorem shows that under rather general conditions the sequence of intervals produced by these algorithms satisfy (1) and (2).

THEOREM 1. *Let f be a real function of a real variable which is continuous on an interval $[a, b]$ and satisfies* (3). *Consider any of the algorithms* 1, 2 *or* 3. *Then either a zero of f is found in a finite number of steps, or the sequence of intervals $\{[a_n, b_n]\}_{n=0}^{\infty}$ produced by this algorithm satisfies* (1) *and* (2), *where x^* is a zero of f.*

The proof of the theorem is straightforward and will be omitted. We note however that at *each* step the intervals produced by algorithms 1 and 2 will satisfy $b_{n+1} - a_{n+1} \leq 0.5(b_n - a_n)$, while in the case of algorithm 3 we have $b_{n+1} - a_{n+1} \leq 0.25(b_n - a_n)$. This shows at least linear convergence. In what follows we will show that under appropriate smoothness assumptions the algorithms 1 and 2 have Q-convergence orders 2 and 4 respectively, while algorithm 3 has the R-convergence order equal to $0.5(3 + \sqrt{13})$. We conjecture that the Q-convergence order of algorithm 3 is also equal to $0.5(3 + \sqrt{13})$. For a recent discussion of Q- and R-convergence orders see Potra [8].

THEOREM 2. *Under the hypothesis of Theorem* 1 *assume that algorithm* 1 *does not terminate after a finite number of steps. Suppose also that f is twice continuously differentiable on $[a, b]$ and that x^* is a simple zero off. Then the sequence of diameters $\{(b_n - a_n)\}_{n=0}^{\infty}$ converges Q-quadratically to zero, i.e. there is a constant γ such that*

$$b_{n+1} - a_{n+1} \leq \gamma(b_n - a_n)^2, \quad n = 0, 1, \ldots. \tag{8}$$

Moreover, there is a positive integer n_1 such that for all $n \geq n_1$ we have $a_{n+1} = \hat{a}_n$, $b_{n+1} = \hat{b}_n$ so that for $n \geq n_1$ algorithm 1 *requires only two function values per step.*

PROOF. From (5), (6) and 1.1. of algorithm 1 we deduce that

$$\begin{aligned} f(c_n) &= f(c_n) - f(a_n) - f[a_n, b_n](c_n - a_n) \\ &= (f[c_n, a_n] - f[a_n, b_n])(c_n - a_n) \\ &= (f[a_n, c_n] - f[a_n, b_n])(c_n - a_n) \\ &= f[a_n, c_n, b_n](c_n - b_n)(c_n - a_n). \end{aligned}$$

Denoting

$$\gamma_2 = 0.5 \max_{x \in [a,b]} |f''(x)|$$

it follows that

$$|f(c_n)| \leq \gamma_2 |c_n - b_n| |c_n - a_n| \leq 0.25\gamma_2(b_n - a_n)^2. \tag{9}$$

Because $\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = x^*$ and $f'(x^*) \neq 0$ there is a positive integer n_1 such that

$$\max_{x,y \in [a_n, b_n]} |f[x, y]^{-1}| \leq \gamma_1 \text{ for } n \geq n_1. \tag{10}$$

If n_1 is large enough then according to (7) we may assume that

$$f(\bar{c}_n) f(u_n) < 0 \text{ for } n \geq n_1.$$

Then from 1.4.–1.6. of algorithm 1 and the fact that $u_n, \bar{c}_n \in [\bar{a}_n, \bar{b}_n]$ we deduce that

$$\bar{b}_n - \bar{a}_n \leq |\bar{c}_n - u_n|. \tag{11}$$

On the other hand $c_n \in \{\bar{a}_n, \bar{b}_n\}$ implies $|f(c_n)| \geq |f(u_n)|$ so that from 1.4, (9), (10) and (11) it follows that

$$\hat{b}_n - \hat{a}_n \leq \gamma_3 (b_n - a_n)^2, \tag{12}$$

with $\gamma_3 = 0.5\gamma_1\gamma_2$. If n_1 is large enough then

$$\hat{b}_n - \hat{a}_n < \mu_1 (b_n - a_n).$$

This proves that for $n \geq n_1$ we have $a_{n+1} = \hat{a}_n$, $b_{n+1} = \hat{b}_n$. By taking $\gamma \geq \max\{\gamma_3, (b_{i+1} - a_{i+1})/(b_i - a_i)^2\}$ for $i = 0, 1, \ldots, n_1 - 1$ and using (12) we obtain (8). ■

THEOREM 3. *Under the hypothesis of Theorem* 1 *assume that algorithm* 2 *does not terminate in a finite number of steps. Suppose also that* f *is three times continuously differentiable on* $[a, b]$ *and that* x^* *is a simple zero of* f. *Then the sequence of diameters* $\{(b_n - a_n)\}_{n=0}^{\infty}$ *converges to zero with Q-order* 4, *i.e. there is a constant* $\bar{\gamma}$ *such that*

$$b_{n+1} - a_{n+1} \leq \bar{\gamma}(b_n - a_n)^4, \quad n = 0, 1, \ldots. \tag{13}$$

Moreover there is a positive integer n_2 *such that for all* $n \geq n_2$ *we have* $a_{n+1} = \tilde{a}_n, b_{n+1} = \tilde{b}_n$ *so that for* $n \geq n_2$ *algorithm* 2 *requires only three function values per step.*

PROOF. As in the proof of Theorem 2 we may assume that (10) is satisfied, and that, corresponding to (11), we have

$$\tilde{b}_n - \tilde{a}_n \leq |\hat{c}_n - u_n|, \quad n \geq n_1. \tag{14}$$

From 2.3.–2.6. it follows that

$$|\hat{c}_n - u_n| \leq 2\gamma_1 |f(\bar{c}_n)|. \tag{15}$$

Let

$$\gamma_4 = \max_{x \in [a,b]} \frac{1}{3!} |f'''(x)|. \tag{16}$$

Using the error formula for Lagrange interpolation (see e.g. Brent [4]) we obtain

$$\begin{aligned} |f(\bar{c}_n)| &\leq \gamma_4 |\bar{c}_n - a_n| |\bar{c}_n - b_n| |\bar{c}_n - c_n| \\ &\leq 0.25\gamma_4 (b_n - a_n)^2 |\bar{c}_n - c_n|, \end{aligned} \tag{17}$$

$$\begin{aligned} |c_n - \bar{c}_n| &= |f[c_n, \bar{c}_n]^{-1} (f(c_n) - f(\bar{c}_n))| \\ &\leq \gamma_1 (|f(c_n)| + |f(\bar{c}_n)|. \end{aligned} \tag{18}$$

Inequalities (17) and (18) imply that

$$|f(\bar{c}_n)| \leq \gamma_5 (b_n - a_n)^2 (|f(c_n)| + |f(\bar{c}_n)|) \tag{19}$$

where $\gamma_5 = 0.25\gamma_1\gamma_4$. Because $\lim_{n\to\infty}(b_n - a_n) = 0$ we may assume that

$$\gamma_5(b_n - a_n)^2 \leq 0.5 \text{ for } n \geq n_2 \geq n_1. \tag{20}$$

Finally from (19) and (20) it follows that

$$|f(\bar{c}_n)| \leq 2\gamma_5(b_n - a_n)^2|f(c_n)|, \quad n \geq n_2. \tag{21}$$

If n_2 is large enough then from (14), (15) and (21) it follows that

$$\tilde{b}_n - \tilde{a}_n < \mu_2(b_n - a_n), \quad n \geq n_2,$$

so that $a_{n+1} = \tilde{a}_n$, $b_{n+1} = \tilde{b}_n$ and from (9), (14), (15), (21) we deduce that (13) holds with

$$\bar{\gamma} \geq \max\{\gamma_1\gamma_2\gamma_5, (b_{i+1} - a_{i+1})/(b_i - a_i)^4\} \text{ for } i = 0, \ldots, n_2 - 1. \quad \blacksquare$$

THEOREM 4. *Under the hypothesis of Theorem* 1 *assume that algorithm* 3 *does not terminate in a finite number of steps. Suppose also that* f *is three times continuously differentiable on* $[a, b]$ *and that* x^* *is a simple zero of* f. *Then the sequence of diameters* $\{(b_n - a_n)\}_{n=0}^{\infty}$ *converges to zero with R-convergence order equal to* $0.5(3 + \sqrt{13}) = 3.3027\ldots$.

PROOF. Inequalities (15) and (21) do not depend on a particular choice of $c_n \in [a_n, b_n]$. For any such point we have

$$|\hat{c}_n - u_n| \leq \gamma_6(b_n - a_n)^2|f(c_n)| \tag{22}$$

with $\gamma_6 = 4\gamma_1\gamma_5$. Using Theorem 1 and the fact that $\hat{c}_n, u_n \in [\hat{a}_n, \hat{b}_n] \subset [a_n, b_n]$ together with (7) we may assume that

$$f(\hat{c}_n)f(u_n) < 0 \text{ for } n \geq n_3 \geq n_2.$$

Then according to 3.5. and 3.6. of algorithm 3 we have

$$\{a_{n+1}, b_{n+1}\} = \{u_n, \hat{c}_n\} \text{ for } n \geq n_3. \tag{23}$$

Then $c_{n+1} = 0.5(a_{n+1} + b_{n+1}) = 0.5(u_n + \hat{c}_n)$ so that

$$c_{n+1} - u_n = 0.5(\hat{c}_n - u_n). \tag{24}$$

Using 3.6. and (5) it follows that

$$\begin{aligned} 2f(c_{n+1}) &= 2f(c_{n+1}) - 2f(u_n) - f[\hat{a}_n, \hat{b}_b](\hat{c}_n - u_n) \\ &= (f[c_{n+1}, u_n] - f[\hat{a}_n, \hat{b}_n])(\hat{c}_n - u_n). \end{aligned} \tag{25}$$

From standard mean value theorems it follows that

$$|f[c_{n+1}, u_n] - f[\hat{a}_n, \hat{b}_n]| \leq 2\gamma_2(b_n - a_n) \tag{26}$$

and from (22), (25) and (26) we get

$$|f(c_{n+1})| \leq \gamma_7 (b_n - a_n)^3 |f(c_n)| \tag{27}$$

with $\gamma_7 = \gamma_2 \gamma_6$.

Let us denote

$$\varepsilon_n = b_n - a_n, \quad \eta_n = f(c_n). \tag{28}$$

From (22), (23) and (27) we deduce that

$$\varepsilon_{n+1} \leq \gamma_6 \varepsilon_n^2 \eta_n, \quad \eta_{n+1} \leq \gamma_7 \varepsilon_n^3 \eta_n, \quad n \geq n_3.$$

According to a theorem of J. W. Schmidt [10] the R-order of the sequences $\{\varepsilon_n\}_{n=0}^{\infty}, \{\eta_n\}_{n=0}^{\infty}$ is the spectral radius of the matrix $\begin{bmatrix} 2 & 1 \\ 3 & 1 \end{bmatrix}$ which is $0.5(3 + \sqrt{13}) = 3.3027\ldots$. ■

4. Numerical results.

In what follows we present our numerical experiments comparing the three methods here in this paper with two recognized efficient equation solvers, Dekker's method [5] and Brent's method [4].

The machine used was Encore-multimax. Double precision was used. The test problems are listed in Table 1. For Algorithm 1 and Algorithm 2 of this paper we chose $\mu_1 = \mu_2 = 0.5$. For Dekker's method we picked up the ALGOL 60 routine *Zeroin* presented in [5] and translated it into Fortran, while for Brent's method we simply used the Fortran routine *Zero* presented in the Appendix of [4]. The termination criterion used for all five methods was the one used by Brent in the above-mentioned routine *Zero*. Namely, we terminate when

$$b - a \leq 2 \cdot tole(a, b)$$

where $[a, b]$ is the current enclosing interval, and

$$tole(a, b) = 2 \cdot macheps \cdot |u| + tol$$

where *macheps* is the relative machine precision which in our case is $2.2204460492504 \times 10^{-16}$, $u \in \{a, b\}$ such that $|f(u)| = \min\{|f(a)|, |f(b)|\}$, and *tol* is a user-given non-negative tolerance which should be chosen positive if the given initial interval $[a_0, b_0]$ contains Zero. (However, in our experiments we also tried $tol = 0$, and all methods worked properly). According to the above termination criterion, a natural modification was employed in our implementation of all three methods of this paper. That is, every time before we call the subroutine *bracket*$(a, b, c, \bar{a}, \bar{b})$, we do the following first:

```
if       b − a ≤ 4·λ·tole(a, b)   then c := ½(a + b)
else if  c < a + 2·λ·tole(a, b)   then c := a + 2·λ·tole(a, b)
else if  c > b − 2·λ·tole(a, b)   then c := b − 2·λ·tole(a, b)
endif.
```

Table 1. *Test problems: (n is a positive integer).*

	Function $f(x)$	Initial interval $[a, b]$
# 1	$x^3 - 1$	[0.5, 1.5]
# 2	$x^2(x^2/3 + \sqrt{2}\sin x) - \sqrt{3}/18$	[0.1, 1]
# 3	$11x^{11} - 1$	[0.1, 1]
# 4	$x^3 + 1$	[−1.8, 0]
# 5	$2xe^{-n} + 1 - 2e^{-nx}$	[0, 1]
# 6	$[1 + (1-n)^2]x^2 - (1-nx)^2$	[0, 1]
# 7	$x^2 - (1-x)^n$	[0, 1]
# 8	$[1 + (1-n)^4]x - (1-nx)^4$	[0, 1]
# 9	$e^{-nx}(x-1) + x^n$	[0, 1]
# 10	$x^2 + \sin(x/n) - 1/4$	[0, 1]
# 11	$(nx-1)/((n-1)x)$	[0.01, 1]

After calling the subroutine *bracket*$(a, b, c, \bar{a}, \bar{b})$, we do this:

$$\text{if } \bar{b} - \bar{a} \leq 2 \cdot tole(\bar{a}, \bar{b}) \text{ then terminate.}$$

In the modification above, λ is a user-given parameter such that $0 < \lambda < 1$. In our experiments, we take $\lambda = 0.7$.

We used all the test problems listed in Table 1 (for problem # 5 to # 11, we tested them with $n = 5, 10, 20$) with different user-given *tol* ($tol = 10^{-2}, 10^{-5}, 10^{-7}, 10^{-10}, 10^{-15}$, and 0).

Since the structures of the algorithms are quite different, the number of iterations used does not mean much in the comparison. Hence they are not listed here. Instead in Table 2 we list the total number of function evaluations used by each individual method in solving all the testing problems.

From Table 2 we see that in the sense of the number of function evaluations used, Algorithm 2 works the best from our three methods, especially when *tol* is small ($tol = 10^{-15}, 0$). This reconfirms the fact that Algorithm 2 has the highest asymptotic efficiency index of the three. It is encouraging that the practical behaviour of our Algorithm 2 and Algorithm 3 are close to those of Dekker and Brent. With some problems in our experiments such as problem # 1, # 6, and # 10, our Algorithm 2 and Algorithm 3 always work quicker than Dekker's and Brent's

Table 2. *Total number of function evaluations*

tol	Alg. 1	Alg. 2	Alg. 3	Dekker	Brent
10^{-2}	185	174	175	163	166
10^{-5}	278	250	246	222	221
10^{-7}	309	273	265	244	237
10^{-10}	337	285	279	259	252
10^{-15}	355	298	311	279	265
0	360	302	318	287	266

Table 3. *Total number of function evaluations for problems # 1, # 6, # 10.*

tol	Alg. 1	Alg. 2	Alg. 3	Dekker	Brent
10^{-2}	46	35	32	46	46
10^{-5}	68	40	34	59	56
10^{-7}	70	43	37	66	63
10^{-10}	82	44	39	70	67
10^{-15}	83	45	39	76	70
0	86	45	39	77	70

Table 4. *Total number of function evaluations for problems # 3, # 7, # 9, # 11.*

tol	Alg. 1	Alg. 2	Alg. 3	Dekker	Brent
10^{-2}	98	90	92	68	71
10^{-5}	140	141	140	97	98
10^{-7}	155	155	147	105	104
10^{-10}	159	160	153	111	110
10^{-15}	175	168	174	120	116
0	175	169	175	123	116

method, while for some other problems like # 3, # 7, # 9 and # 11 the reuslts are just opposite. The total number of function evaluations for these two groups are given in Table 3 and Table 4, respectively. The second group simply supports the fact indicated in Section 1 of this paper, that is, the efficiency index is an asymptotic notion and in practical applications methods with high efficiency index like our Algorithm 2 may sometimes very likely be outperformed by other efficient solvers.

Acknowledgement.

We would like to thank our students Armin Gienger and Yixun Shi for helping us with a number of numerical experiments.

REFERENCES

1. Alefeld G., Herzberger J.: *Introduction to Interval Computations.* Academic Press (1983).
2. Alefeld G., Potra F.: *On two higher order enclosing methods of J. W. Schmidt.* Z. angew. Math. Mech. 68 (1988) 8, 331–337.
3. Alefeld G., Potra F.: *A new class of interval methods with higher order of convergence.* Computing 42, (1989) 69–80.
4. Brent R. P.: *Algorithms for Minimization without Derivatives.* Prentice-Hall. Englewood Cliffs, New Jersey, 1972.
5. Dekker, T. J.: *Finding a zero by means of successive linear interpolation.* In *Constructive Aspects of the Fundamental Theorem of Algebra.* Edited by B. Dejon and P. Henrici. Wiley Interscience (1969).
6. King, R. F.: *Methods without secant steps for finding a bracketed root.* Computing 17, (1976), 49–57.
7. Ostrowski A. M.: *Solution of Equations in Banach Spaces.* Academic Press, New York 1973.
8. Potra F.: *On Q-order and R-order of convergence.* JOTA, 63 (1989), 415–431.
9. Schmidt J. W.: *Eingrenzung der Lösungen nichtlinearer Gleichungen mit höherer Konvergenzgeschwindigkeit.* Computing 8 (1971), 208–215.
10. Schmidt J. W.: *On the R-order of coupled sequences.* Computing 26 (1981), 333–342.