

A Class of Improved Algorithm for Solving Interval Nonlinear Equation

Liang Qiu, Haijun Wang*, Qi Wang

School of Mathematics, China University of Mining and Technology, Xuzhou Jiangsu
Email: qlcumt@163.com, wanghj@cumt.edu.cn

Received: Aug. 6th, 2017; accepted: Aug. 20th, 2017; published: Aug. 25th, 2017

Abstract

In this paper, we consider the problem of solving interval nonlinear equation with interval parameters. Dividing the initial interval by monotone segment technique, we extended the improved interval Newton algorithm proposed in [1], and established a class of improved algorithm for solving interval nonlinear equation. Besides, some relevant theoretical results and effectiveness tests are given. Numerical examples show the new algorithm can not only solve problems that can not be solved by improved interval Newton method, but also greatly improve the computational efficiency.

Keywords

Interval Nonlinear Equation with Interval Parameters, Improved Interval Newton Algorithm, Subinterval, Interval Zero

求解区间非线性方程的一类改进算法

邱亮, 王海军*, 王琪

中国矿业大学数学学院, 江苏 徐州
Email: qlcumt@163.com, wanghj@cumt.edu.cn

收稿日期: 2017年8月6日; 录用日期: 2017年8月20日; 发布日期: 2017年8月25日

摘要

本文研究了含区间参数的区间非线性方程的求解问题, 通过单调分割技术对初始区间进行并行划分, 对文献[1]中的拓展区间牛顿算法进行了改进, 建立了一类求解区间非线性方程的新算法, 给出相关的理论

*通讯作者。

结果并进行数值有效性测试。数值算例表明新算法不仅可以解决拓展区间牛顿法不能解决的问题，并且大大提高了计算效率。

关键词

含区间参数的区间非线性方程，拓展的区间牛顿算法，子区间，零解区间

Copyright © 2017 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

传统的非线性方程可以表示为：

$$f(x) = 0, \quad (1)$$

这里 $f: R \rightarrow R$ 是连续可微的函数， R 表示所有的实数。

在众多科学领域(如：化工[2]、计算机图形学[3]、机器人科学[4]、控制理论[5]等)中，都会涉及到(1)的求解。但是，在现实问题中考虑到实验过程中实验方法、实验仪器等存在的误差、计算过程中浮点数运算出现的误差，在实际计算中我们常常需要解决的是求解含不定参数的非线性方程：

$$f(x, p) = 0, \quad (2)$$

这里 $p \in [p]$ ， $[p] \in IR$ ， IR 表示所有实区间集合。下面我们给出含区间参数的区间非线性方程的定义：

定义 1.1 $f: R \times IR \rightarrow IR$ 是连续可微的函数， $[x]$ 是初始区间， $[p]$ 是包含 f 所有系数的区间向量，则含区间参数的区间非线性方程可表示为：

$$f(x, [p]) = 0. \quad (3)$$

定义 1.2 $f(x, p) = 0$ ， $p \in [p]$ 的每一个连续的零解集合构成区间方程 $f(x, [p]) = 0$ 的一个零解区间，如果该零解区间只包含一个值，称其为退化的零解区间。

区间方程(3)的所有零解就是方程(2)所有方程零解的集合。显然区间方程(3)的解有 3 种情况：空区间、退化区间(区间内只包含一个值)和非退化区间。对于非退化区间来说，包含在其中的零解是无穷多的，方程(2)的系数是不断变化的，其零解也是不断变化的。但是，区间方程(3)的解区间的个数是有限的，所以我们不直接求方程(2)的零解，而求其解区间。

1966 年，Moore [6] 在《Interval Analysis》一书中第一次用区间牛顿法逼近了一个简单的区间方程的零解；1992 年，Hansen [7] 利用牛顿法提出了区间搜索和启发式的终止条件求解区间方程的零解；2006 年，Nikas [8] 等，在理论上提出了一种边界近似的方法求解区间方程，并取得了很好的成果；2009 年，Nikas [1] 等，又基于牛顿法提出了通过分离零解区间端点的方法求解区间方程。

本文基于 Nikas 提出的分离零解区间端点的方法，首先考虑区间函数在指定初始区间内不单调的情况下，先分割出存在零解的各个单调子区间，再逼近其零解区间的各个端点。数值算例证明该方法可在更少的迭代步下确定零解区间。

2. 拓展的区间牛顿法(EIN)

对于 $f(x)=0$, $x \in [x] \subseteq \mathbb{R}$, $[x]^*$ 是方程的解, 且 $[x]^* \subseteq [x]$, 则传统的区间牛顿法公式为:

$$N([x]) = m - \frac{F(x)}{F'([x])}, \tag{4}$$

其中: m 是 $[x]$ 内的任意一点(通常取 m 为 $[x]$ 的中点), F 和 F' 是 f 和 f' 在区间 $[x]$ 上的区间扩张函数。则(4)的实际迭代公式为:

$$\begin{aligned} m^{(k)} &= \text{mid}([x]^{(k)}), \\ N([x]^{(k)}) &= m^{(k)} - \frac{F(m^{(k)})}{F'([x]^{(k)})}, \\ [x]^{(k+1)} &= [x]^{(k)} \cap N([x]^{(k)}). \end{aligned}$$

对于区间牛顿法, 当 $0 \in F([x])$ 时迭代失败, 对此 Hansen [9] 拓展了区间牛顿法使其可以处理 $0 \in F([x])$ 的情况。但对于 $m(x)=0$ 的情况目前还没有较好的处理方法, 只能用区间二分法避免此情况。为了求解区间方程(3), Nikas [1] 对区间牛顿法进行了拓展, 我们在这里对其进行简短的说明。

将区间牛顿法换一种表达方式, 令: $F(m) = [\underline{f}_m, \overline{f}_m]$, $m \in [x]$, $F'([x]) = [\underline{d}, \overline{d}]$, 则区间牛顿法可表示成:

$$N([x]) = m - \frac{[\underline{f}_m, \overline{f}_m]}{[\underline{d}, \overline{d}]}. \tag{5}$$

根据区间运算法则, (5)可表示为:

$$N([x]) = m - \left(\frac{[\underline{f}_m, \overline{f}_m]}{[\underline{d}, \overline{d}]} \cup \frac{[\overline{f}_m, \underline{f}_m]}{[\underline{d}, \overline{d}]} \right) = \left(m - \frac{[\underline{f}_m, \overline{f}_m]}{[\underline{d}, \overline{d}]} \right) \cup \left(m - \frac{[\overline{f}_m, \underline{f}_m]}{[\underline{d}, \overline{d}]} \right). \tag{6}$$

令: $N_L = m - \frac{[\underline{f}_m, \overline{f}_m]}{[\underline{d}, \overline{d}]}$, $N_U = m - \frac{[\overline{f}_m, \underline{f}_m]}{[\underline{d}, \overline{d}]}$, 则(6)的迭代公式是:

$$\begin{aligned} m^{(k)} &= \text{mid}([x]^{(k)}), \\ H([x]^{(k)}) &= N_L([x]^{(k)}) \cup N_U([x]^{(k)}), \\ [x]^{(k+1)} &= [x]^{(k)} \cap H([x]^{(k)}). \end{aligned}$$

引理 2.1 [1] 假设 f 是(3)定义的含区间参数的区间非线性方程, $[x]$ 是初始区间, $[x]^* \subseteq [x]$ 是方程 $f(x, [p])=0$ 的零解区间。如果 $0 \in f(m([x]))$, 且区间 $[r]$ 定义如下:

$$[r] = \left[\min\{\overline{N}_L, \overline{N}_U\}, \max\{\underline{N}_L, \underline{N}_U\} \right], \tag{7}$$

则 $[r]$ 总是存在且 $[r] \subseteq [x]^*$ (详见图 1)。

引理 2.2 [1] 假设引理 2.1 的条件成立, 如果 $0 \notin f(m([x]))$ 且 $0 \notin F'([x])$, $[r]$ 的定义同(7), 如果 $[r]$ 存在, 则 $[r] \subseteq [x]^*$ (详见图 2)。

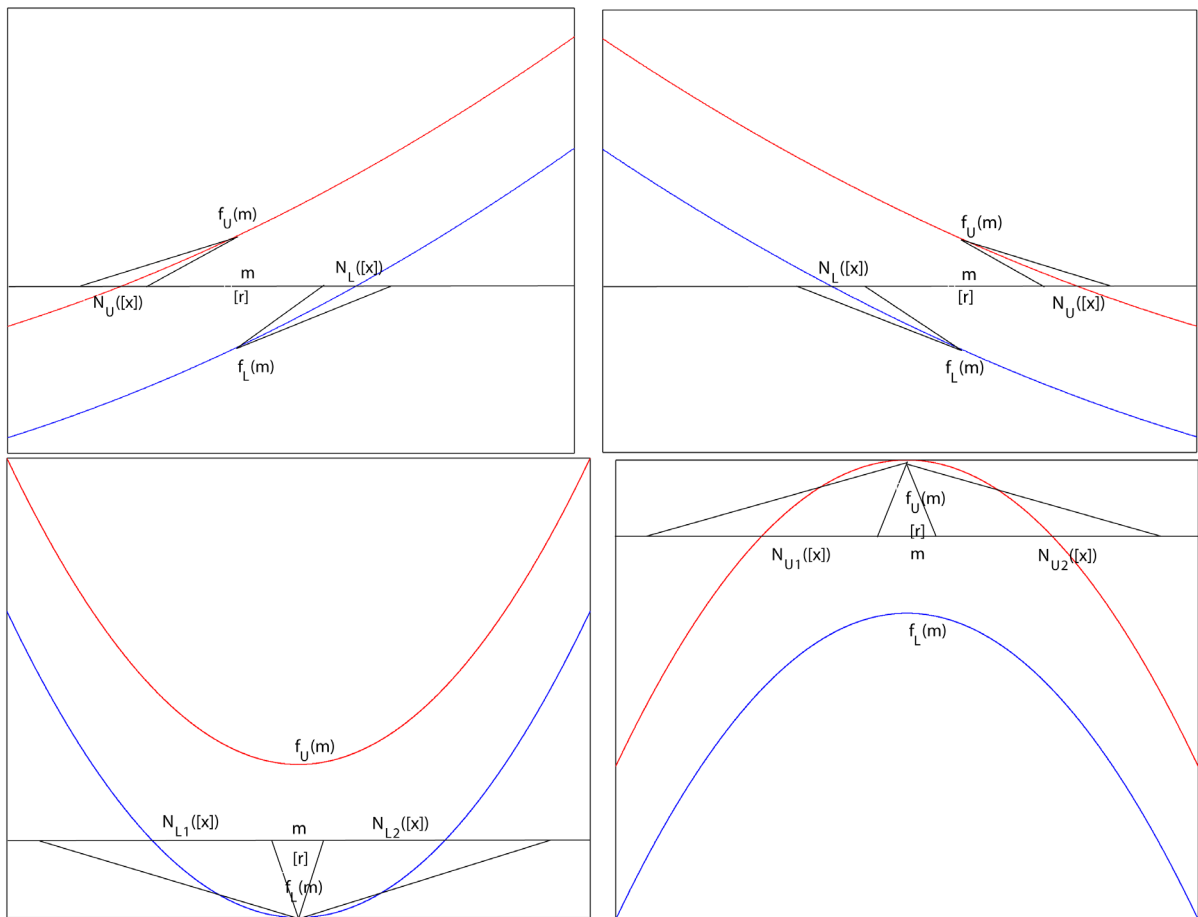


Figure 1. Four cases in lemma 2.1
图 1. 引理 2.1 中的 4 种情况

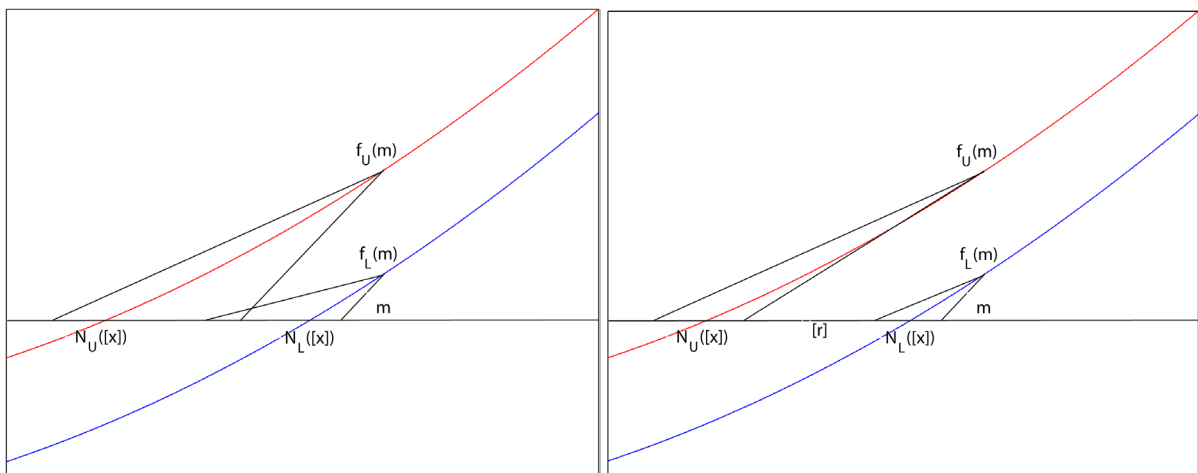


Figure 2. Two cases: $[r]$ not existing or existing in lemma 2.2
图 2. 引理 2.2 中 $[r]$ 不存在与 $[r]$ 存在的 2 种情况

对于一些在指定初始区间内非单调的含区间参数的区间非线性方程，由(6)构造的迭代法计算其零解区间时可能会失效。如下例：

例 2.1 求区间方程 $f(x) = x^2 - [-1, 1]$ 在初始区间 $[x] = [-3, 3]$ 内的零解(精度取 10^{-3})。考虑到 $0 \in F'([x])$ ，根据拓展牛顿法可以得到(详见图 3)：

$$N_{L_1} = \left(m([x]) - \frac{F([x])}{F'([x])} \right) \cap [x] = \left(0 - \frac{-1}{[-\infty, -0.1666666666666666]} \right) \cap [x] = [-3, -0.1666666666666666],$$

$$N_{L_2} = \left(m([x]) - \frac{F([x])}{F'([x])} \right) \cap [x] = \left(0 - \frac{-1}{[0.1666666666666666, +\infty]} \right) \cap [x] = [0.1666666666666666, 3],$$

$$[x]^{(2)} = \left(N_{L_1}([x]^{(1)}) \right) \cup \left(N_{L_2}([x]^{(1)}) \right) \cap [x] = [-3, 3] = [x]^{(1)}.$$

根据例 2.1 的计算结果我们可以发现： $[x]^{(2)} = [x]^{(1)}$ ，则对任意迭代步都有： $[x]^{(k+1)} = [x]^{(k)}$ ，则每一次迭代得到的 $[r]$ 都是一样的，永远达不到要求的精度。这样由(6)构造的迭代法并不能解决这个问题。

3. 新的改进算法

为了提高(5)和(6)在求解非单调的区间方程时的计算效率，我们首先把非单调的区间分割成若干个单调的子区间，在每个单调的子区间内再求解(3)的零解区间。

定理 3.1 假设 f 是(3)定义的含区间参数的区间非线性方程， $[x]$ 是初始区间， $[x]^* = [\underline{x}^*, \bar{x}^*] \subseteq [x]$ 是 f 的一个零解区间， N_L, N_U 是由(5)和(6)计算得到的，则 $\underline{x}^*, \bar{x}^* \in N_L$ 或 $\underline{x}^*, \bar{x}^* \in N_U$ 。

证明：

- 1) 若 $\underline{d} > 0$ ，则 $\underline{x}^* \in N_U, \bar{x}^* \in N_L$ 。
- 2) 若 $\bar{d} < 0$ ，则 $\underline{x}^* \in N_L, \bar{x}^* \in N_U$ 。
- 3) 若 $\underline{d} < 0, \bar{d} > 0$ ，根据 Hansen [9]提出的处理 $0 \in F'([x])$ 的情况有：

$$N_{L_1} = m - \left[\frac{\underline{f}_m}{-\infty, \frac{1}{\underline{d}}} \right], N_{L_2} = m - \left[\frac{\underline{f}_m}{\frac{1}{\bar{d}}, +\infty} \right], \tag{8}$$

则 $\underline{x}^* \in N_{L_1}, \bar{x}^* \in N_{L_2}$ 。

- 4) 若 $\underline{d} > 0, \bar{d} < 0$ ，同理有：

$$N_{U_1} = m - \left[\frac{\bar{f}_m}{-\infty, \frac{1}{\underline{d}}} \right], N_{U_2} = m - \left[\frac{\bar{f}_m}{\frac{1}{\bar{d}}, +\infty} \right], \tag{9}$$

则 $\underline{x}^* \in N_{U_1}, \bar{x}^* \in N_{U_2}$ 。

所以： $\underline{x}^*, \bar{x}^* \in N_L$ 或 $\underline{x}^*, \bar{x}^* \in N_U$ 。

定理 3.1 保证了在分割非单调的初始区间时不会丢失区间方程的零解区间的上下界。但是，分割得到的子区间有可能不包含方程的零解区间，对此我们给出定理 3.2。

定理 3.2 假设定理 3.1 的条件成立， $[x]$ 是初始区间，若(5)和(6)计算得到的 N_L, N_U 为空，则 $[x]$ 内不包含 f 的零解。

证明：假设 $[x]$ 内包含 f 的零解，由定理 3.1 可知： $\underline{x}^*, \bar{x}^* \in N_L$ 或 $\underline{x}^*, \bar{x}^* \in N_U$ ，则 $N_L \cap [x] \neq \emptyset, N_U \cap [x] \neq \emptyset$ ，与原假设矛盾，定理得证。

定理 3.2 保证了不包含零解的区间在迭代中会被自动删除，进而不会影响迭代结果。

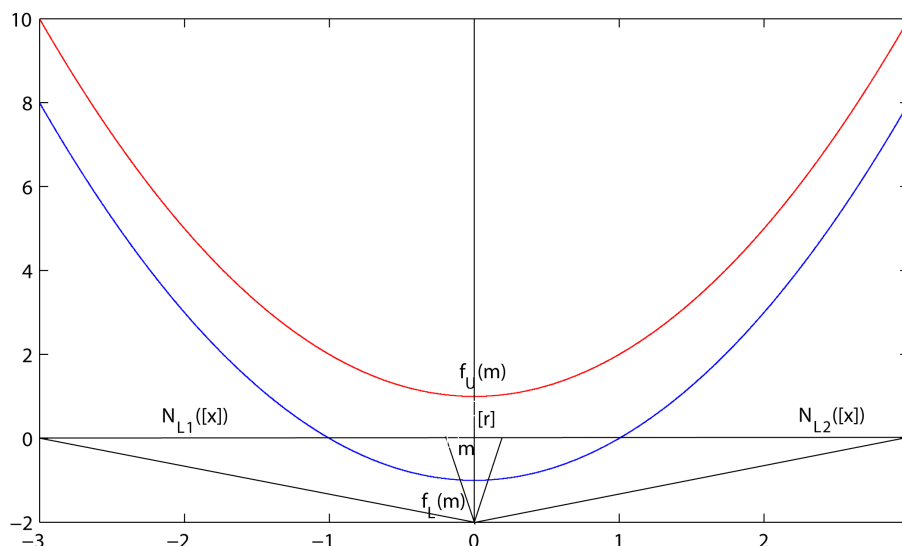


Figure 3. Interval equation $f(x) = x^2 - [-1, 1] = 0$, $[x]^{(0)} = [-3, 3]$

图 3. 区间函数 $f(x) = x^2 - [-1, 1] = 0$, $[x]^{(0)} = [-3, 3]$

定理 3.3 假设 $[\tilde{x}]$ 是通过分割初始区间 $[x]$ 得到的一个单调子区间, $[x]^* = [\underline{x}^*, \bar{x}^*]$ 是 $f(x, [p])$ 的一个区间零解, f_L, f_U 是 f 的边界函数, N_L, N_U 分别是建立在 f_L, f_U 上的区间牛顿算子, 若 $[x]^* \cap [\tilde{x}] \neq \emptyset$, 则 N_L, N_U 可以以任意精度二阶收敛到 \underline{x}^* 或 \bar{x}^* .

证明: 因为 $[\tilde{x}]$ 是单调区间且 $[x]^* \cap [\tilde{x}] \neq \emptyset$, 所以在 $[\tilde{x}]$ 上解的情况只可能有 3 种: $[x]^* \in [\tilde{x}]$, $\underline{x}^* \in [\tilde{x}]$ 和 $\bar{x}^* \in [\tilde{x}]$. 对于 $[x]^* \in [\tilde{x}]$, N_L, N_U 是分别建立在 f_L, f_U 上的区间牛顿算子且 f_L, f_U 是非区间函数, Hansen 在文献[7]中证明了区间牛顿算子可以以任意精度收敛到非区间函数的零解。同理可证: 对于 $\underline{x}^* \in [\tilde{x}]$, N_U 的迭代结果为空, N_L 可以以任意精度收敛到 \underline{x}^* . 对于 $\bar{x}^* \in [\tilde{x}]$, N_L 的迭代结果为空, N_U 可以以任意精度收敛到 \bar{x}^* .

又 Moore 在文献[6]中证明了区间牛顿算子在逼近单调的非区间方程的零解时是二阶收敛的, 故定理得证。

下面给出通过分割非单调区间求解区间方程的零解区间的算法。在如下的算法 1 中, N 用来存储非单调的区间, N 的初始值为 $[x]^{(0)}$, M 用来存储单调的子区间, M 的初始值为空, h 用来取出 N 内不单调的子区间, l 用来将迭代过后的仍不单调的子区间重新存入 N 内, j 用来将单调的子区间存入 M 内和取出 M 内单调的子区间, R 用来存储零解区间, $ready$ 用来决定是否还需要迭代。

算法 1: 分割法求解含区间参数的区间非线性方程的数值算法

- 1: $l=1; j=1; h=0; ready=0;$
- 2: while $ready == 0$
- 3: if $h \leq \text{length}(N)$
- 4: 如果 $F_m = 0, \bar{F}_m = 0$ 或者 $[x]_{new} = [x]$, 采用区间二分法分割区间, 将其存入 N ;
- 5: 根据(8)或(9)计算 N_{l_1} 和 N_{l_2} 或 N_{v_1} 和 N_{v_2} ;
- 6: 在 N_{l_1} 和 N_{l_2} 或 N_{v_1} 和 N_{v_2} 中, 把单调的存入 M 中, 不单调的存入 N 中;

```

7: 如果  $N_{L_1}$  和  $N_{L_2}$  或  $N_{U_1}$  和  $N_{U_2}$  中出现空集, 从  $N$  中取出非单调区间将其赋值;
8: end if
9: 对  $M$  中的区间进行从小到大排序;
10: 令  $N_L = N_U = M(1)$ ,  $j = 1$ ;
11: while  $j \leq \text{length}(M)$ 
12: 根据(6)计算  $N_L$  和  $N_U$ ;
13: 如果  $N_L$  或  $N_U$  为空, 从  $M$  中取出单调区间将其赋值;
14:  $[x]_{\text{new}} = [N_L; N_U]$ ;
15: if  $\max(\text{rad}([x]_{\text{new}})) < 10^{-14}$ 
16: 计算  $r$ ;
17:  $R = R + r$ ;
18: if  $j \leq \text{length}(M)$ 
19:  $j = j + 1$ ;
20:  $N_L = M(j)$ ;  $N_U = M(j)$ ;
21:  $[x]_{\text{new}} = [N_L; N_U]$ ;
22: else
23:  $\text{ready} = 1$ 
24: break;
25: end if
26:  $[x] = [x]_{\text{new}}$ 
27: end if
28: end while
29: end while
30: return  $R$ 

```

注: 算法 1 中, 第 3~7 步是将非单调的初始区间分割成若干个单调的子区间(初始区间单调时不执行), 第 8~9 步是对各单调子区间排序, 以确保迭代出的 r 是零解区间, 第 10~28 步是在各单调子区间内计算 r 。

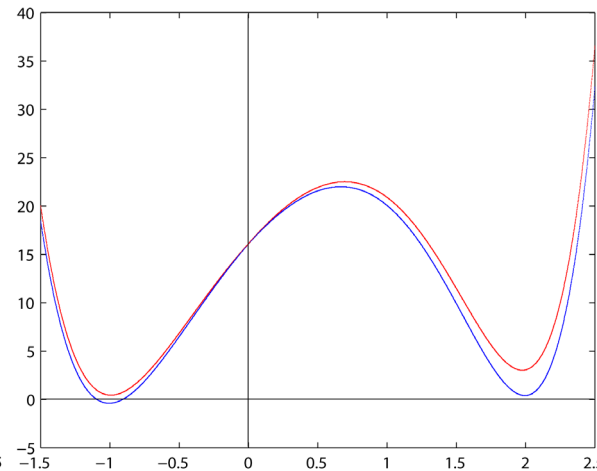
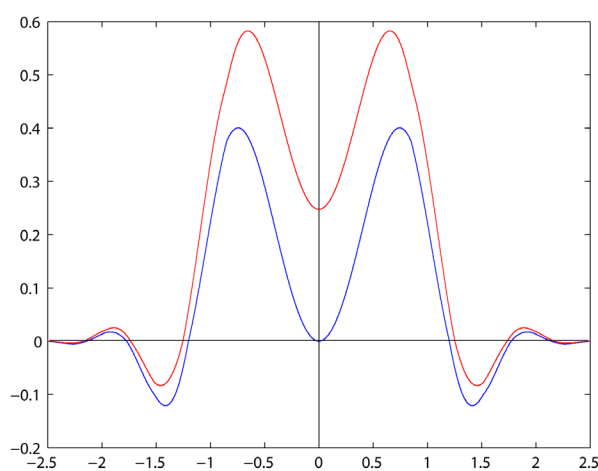
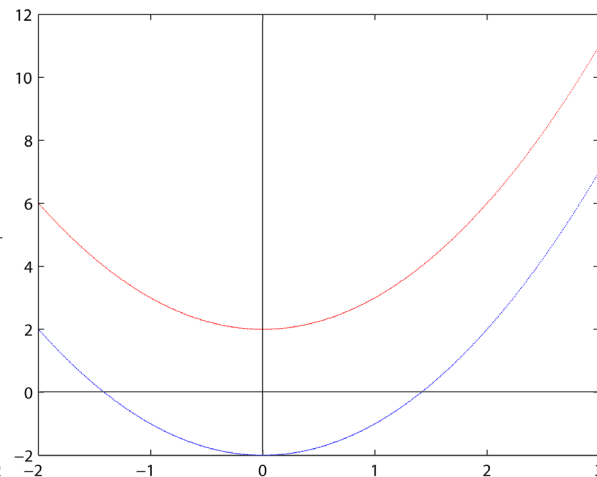
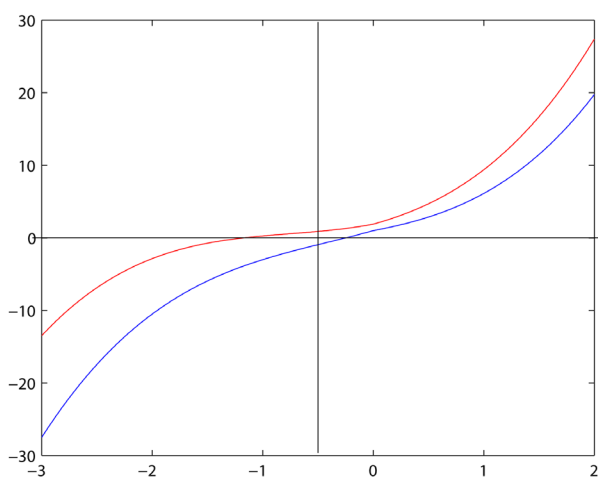
4. 数值算例

本文所使用的数值算例全部来自[1], 计算精度 $\varepsilon = 10^{-14}$ 。通过比较初始区间(SI)、迭代步数(IT)、 F 的计算次数(F)、 F' 的计算次数(F')、二分区间数(B)、获得解区间数(Z)来比较算法的优劣。表 1 给出了 6 个测试函数的具体表现形式, 图 4 给出了每个测试函数的图像, 表 2 是用 EIN 方法计算得到的结果, 表 3 是用算法 1 计算得到的结果。

根据表 2 和表 3 的计算结果我们能得到: 算法 1 在迭代步数上有明显的减少, F' 的计算次数有所增加是因为我们需要检测子区间是否单调, 如果不单调就需要继续迭代, 因此, 这一类检测子区间是否单调的 F' 的计算量与 F 的计算量相同, 而算法 1 的 F 的计算次数有明显的改善, 总的来说算法 1 的计算效率高于 EIN。

Table 1. Test function (from reference [1])
表 1. 测试函数(来自参考文献[1])

NO.	$f(x, [p])$
1.	$x^3 + [p]_2 x^2 + [p]_1 x + [p]_0, [p]_0 = [1, 1.8907], [p]_1 = [2.8749, 4.2501], [p]_2 = [1.2499, 2.2501]$
2.	$x^2 - [p], [p] = [-2, 2]$
3.	$\sin([p]^2 + 2x^2)e^{-[p]^2 - x^2}, [p] = [-0.5, 0.5]$
4.	$x^6 - p_3 x^5 + p_4 x^4 + [p]_3 x^3 - p_2 x^2 + [p]_1 x + p_0, p_0 = 16.1024, [p]_1 = [15.8448, 16.52], p_2 = 7.872, [p]_3 = [-4.0388, -3.875], p_4 = 1.0256, p_5 = 2$
5.	$x^4 + [p]_1^3 [p]_2 x^3 + [p]_1^2 [p]_2^2 [p]_3 x^2 + [p]_1 [p]_3^3 [p]_2^2 x + [p]_3, [p]_1 = [1.15, 1.65], [p]_2 = [1.3, 1.7], [p]_3 = [0.6, 1.0]$
6.	$\left(\frac{5}{\pi}x - \frac{5.1}{4\pi^2}x^2 + [p] - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x, [p] = [-2, 0]$



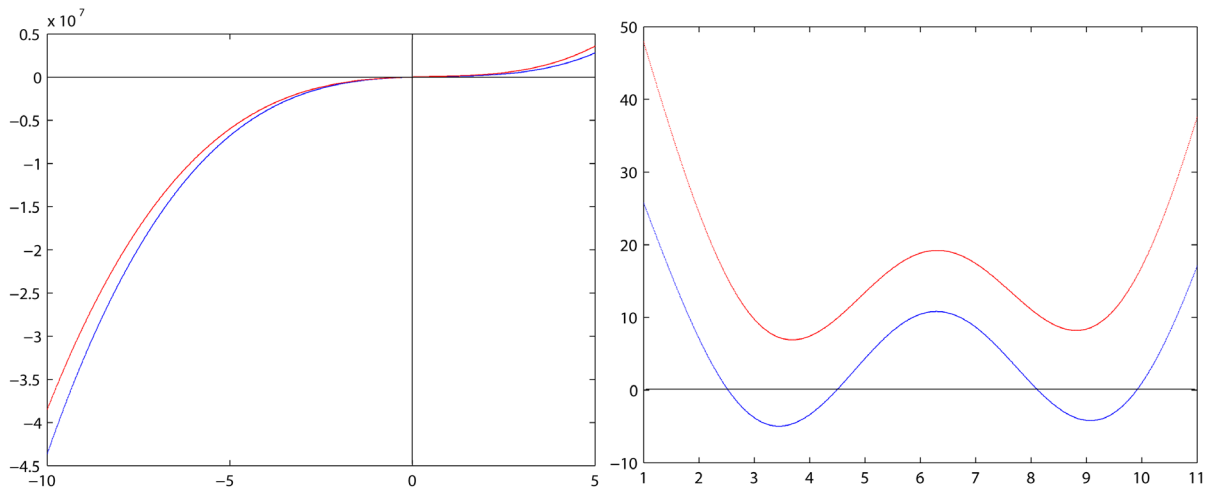


Figure 4. The image of test function
图 4. 测试函数的图像

Table 2. Numerical results are calculated by EIN
表 2. EIN 计算得到的数值结果

NO.	SI	IT	F	F'	B	Z	零解区间
1.	[-3,2]	45	88	43	0	1	$R = [-1.173264124091346, -0.24991803609978]$
2.	[-2,3]	15	28	13	0	1	$R = [-1.414213562373095, 1.414213562373095]$ $R_1 = [-2.500000000000000, -2.481569121983017]$ $R_2 = [-2.170803763674813, -2.141819082085289]$ $R_3 = [-1.772453850905517, -1.736834089252561]$ $R_4 = [-1.253314137315509, -1.202412710695869]$
3.	[-2.5,2.5]	317	606	289	1	9	$R_5 = [-0.000000000000008, 0.000000000000008]$ $R_6 = [1.202412710695869, 1.253314137315509]$ $R_7 = [1.736834089252561, 1.772453850905517]$ $R_8 = [2.141819082085289, 2.170803763674813]$ $R_9 = [2.481569121983017, 2.500000000000000]$
4.	[-1.5,2.5]	55	103	48	0	1	$R = [-1.094076044826976, -0.908642763062733]$
5.	[-10,5]	69	134	65	0	1	$R = [-7.599154609171094, -0.074548001830275]$
6.	[1,11]	62	118	56	0	2	$R_1 = [2.528141250340096, 4.488909517022945]$ $R_2 = [8.122294734375085, 9.909824008961696]$

Table 3. Numerical results are calculated by algorithm1
表 3. 算法 1 计算得到的数值结果

NO.	SI	IT	F	F'	B	Z	零解区间
1.	[-3,2]	30	31	31	0	1	$R = [-1.17326412409134, -0.24999180360997]$
2.	[-2,3]	6	12	15	0	1	$R = [-1.41421356237310, 1.41421356237310]$

Continued

							$R_1 = [-2.500000000000000, -2.481569121983017]$
							$R_2 = [-2.17080376367481, -2.14181908208529]$
							$R_3 = [-1.77245385090552, -1.73683408925256]$
							$R_4 = [-1.25331413731551, -1.20241271067587]$
3.	$[-2.5, 2.5]$	196	324	410	1	9	$R_5 = [-0.0000001814388, 0.0000001814388]$
							$R_6 = [1.20241271067587, 1.25331413731551]$
							$R_7 = [1.73683408925256, 1.77245385090552]$
							$R_8 = [2.14181908208529, 2.17080376367481]$
							$R_9 = [2.48156912198302, 2.500000000000000]$
4.	$[-1.5, 2.5]$	39	84	50	0	1	$R = [-1.09407604482696, -0.90864276306273]$
5.	$[-10, 5]$	64	118	68	0	1	$R = [-7.59915460917109, -0.07454800183027]$
6.	$[1, 11]$	42	55	72	0	2	$R_1 = [2.52814125034008, 4.48890951702295]$
							$R_2 = [8.12229473437509, 9.90982400896169]$

5. 总结

本文通过分割非单调的区间得到单调的子区间的方法,对求解含区间参数的区间非线性方程在指定区间内的所有零解区间的拓展区间牛顿算法进行了改进,并且可以解决拓展区间牛顿算法不能解决的问题,在计算效率上也有所提高。

基金项目

本文工作受到江苏省自然科学基金(NO.BK20151139)支持。

参考文献 (References)

- [1] Nikas, I. and Grapsa, T. (2009) Bounding the Zeros of an Interval Equation. *Applied Mathematics and Computation*, **213**, 466-478. <https://doi.org/10.1016/j.amc.2009.03.041>
- [2] Balaji, G. and Seader, J. (1995) Application of Interval Newton's Method to Chemical Engineering Problems. *Reliable Computing*, **1**, 215-223. <https://doi.org/10.1007/BF02385253>
- [3] Carpani, O. Hvidegaard, L., Mortensen, M. and Schneider, T. (2000) Robust and Efficient Ray Intersection of Implicit Surfaces. *Reliable Computing*, **6**, 9-21. <https://doi.org/10.1023/A:1009921806032>
- [4] The COPRIN Project (2003) Minimal and Maximal Real Roots of Parametric Polynomials Using Interval Analysis. In: Bliiek, C., Jermann, C. and Neumaier, A., Eds., *Global Optimization and Constraint Satisfaction, First International Workshop Global Constraint Optimization and Constraint Satisfaction, COCOS 2002, Valbonne-Sophia Antipolis, France, October 2-4, 2002, Revised Selected Papers, Lecture Notes in Computer Science, Vol. 2861*, Springer.
- [5] Šiljak, D. and Stipanović, D. (1999) Robust D-Stability via Positivity. *Automatica*, **35**, 1477-1484. [https://doi.org/10.1016/S0005-1098\(99\)00042-4](https://doi.org/10.1016/S0005-1098(99)00042-4)
- [6] Moore, R. (1966) *Interval Analysis*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- [7] Hansen, E. (1992) *Global Optimization Using Interval Analysis*. Monographs and Textbooks in Pure and Applied Mathematics, Vol. 165, Marcel Dekker, New York.
- [8] Nikas, I. Sotiropoulos, D. and Grapsa, T. (2006) Extending Interval Newton Method for Nonlinear Parameterized Equations. In: Simos, T., Psihoyios, G. and Tsitouras, C., Eds., *ICNAAM-International Conference on Numerical Analysis and Applied Mathematics, Wiley-VCH, Hersonisos, Crete, 512-515*, ISBN 3-527-40743-X.
- [9] Hansen, E. (1978) Interval Forms of Newton's Method. *Computing*, **20**, 153-163. <https://doi.org/10.1007/BF02252344>