ЧИСЛЕННЫЙ АНАЛИЗ СХОДИМОСТИ НЕКОТОРЫХ ИТЕРАЦИОННЫХ МЕТОДОВ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Олифер В.И.

В данной работе проведен численный анализ сходимости некоторых итерационных методов высокого порядка для решения нелинейных уравнений вида $f(x) = 0$.

Далее рассматриваются следующие пять итерационных формул:

MN – метод Ньютона 2-го порядка [1]:

$$x_{n+1} = x_n - D_n(x_n); \tag{1}$$

$\mathbf{MN_c}$ – непрерывный аналог метода Ньютона 3-го порядка [2]:

$$x_{n+1} = s_n - D_n(s_n), \qquad s_{n+1} = x_{n+1} - D_n(x_{n+1}), \qquad s_0 = x_0 - D_n(x_0); \tag{2}$$

$\mathbf{CH_1}$ – метод Чебышева 3-го порядка [3]:

$$x_{n+1} = x_n - \left[1 + \frac{L_n(x_n)}{2}\right] \cdot D_n(x_n); \tag{3}$$

$\mathbf{CH_2}$ – метод Чебышева 4-го порядка [4]:

$$x_{n+1} = x_n - \left[1 + \frac{\left(L_n(x_n) + L_n^2(x_n)\right)}{2} - \frac{K_n(x_n)}{6}\right] \cdot D_n(x_n); \tag{4}$$

$\mathbf{CH_3}$ – метод Чебышева-Хэлли 5-го порядка [5]:

$$\begin{gathered} x_{n+1} = x_n - \left[1 + \frac{1}{2} \cdot \frac{L_n(x_n)}{1 - L_n(x_n)}\right] \cdot D_n(x_n), \\ \tilde{x}_{n+1} = x_{n+1} - \left[1 + \frac{M(x_n, x_{n+1})}{1 - M(x_n, x_{n+1})}\right] \cdot D_n(x_n, x_{n+1}), \qquad x_{n+1} = \tilde{x}_{n+1}, \end{gathered} \tag{5}$$

где:

$$D_n = \frac{f(x_n)}{f'(x_n)},\ \ L_n = \ D_n \cdot \frac{f''(x_n)}{f'(x_n)},\ \ K_n = D_i^2 \cdot \frac{f'''(x_n)}{f'(x_n)},$$
$$M(x_n, x_{n+1}) = L_n(x_n) \cdot \left(1 - \frac{f(x_{n+1})}{f(x_n)}\right),\ \ D_n(x_n, x_{n+1}) = \frac{f(x_{n+1})}{f'(x_n)},\ \ n = 0, 1, 2, \dots; \tag{6}$$

Для реализации формул (6) необходимо наличие простого и точного (с машинной точностью) способа определения значений производных (до третьего порядка включительно) исходной функции $f(x)$. Это может быть успешно решено, если воспользоваться методом автоматического дифференцирования (АД) [6], суть которого заключается в создании нового типа данных и переопределения операций и базовых функций над ними. В качестве такого нового типа данных могут быть исползованы гипер-дуальные числа 3-го класса [6].

Гипер-дуальное число 3-го класса имеет представление в виде $Z = z + z_1\boldsymbol{\varepsilon} + z_2\boldsymbol{\omega} + z_3\boldsymbol{\gamma}$, $z,\ z_1,\ z_2,\ z_3 \in \mathbb{R}$. Параметр z называется главной (*Re* – действительной) частью гипер-дуального числа, а z_1, z_2, z_3 – его мнимыми (*Im1*, *Im2*, *Im3* – инфинитезимальными) частями. Абстрактные элементы $\boldsymbol{\varepsilon},\ \boldsymbol{\omega},\ \boldsymbol{\gamma}$ образуют базис мнимых частей гипер-дуального числа, отвечающий следующими правилам: $\boldsymbol{\varepsilon}^2 = 2\boldsymbol{\omega},\ \boldsymbol{\varepsilon\omega} = 3\boldsymbol{\gamma},\ \boldsymbol{\varepsilon\gamma} = \boldsymbol{\omega\gamma} = \boldsymbol{\gamma}^2 = \boldsymbol{\omega}^2 = 0$. Алгебраические операции над дуальными числами 3-го класса определяются формулами:

$$\begin{aligned}
&X = x + x_1\boldsymbol{\varepsilon} + x_2\boldsymbol{\omega} + x_3\boldsymbol{\gamma}, \qquad Y = y + y_1\boldsymbol{\varepsilon} + y_2\boldsymbol{\omega} + y_3\boldsymbol{\gamma},\\
&X + Y = x + y + (x_1 + y_1)\boldsymbol{\varepsilon} + (x_2 + y_2)\boldsymbol{\omega} + (x_3 + y_3)\boldsymbol{\gamma},\\
&X{\cdot}Y = x{\cdot}y + (x{\cdot}y_1 + y{\cdot}x_1)\boldsymbol{\varepsilon} + (x{\cdot}y_2 + 2x_1{\cdot}y_1 + y{\cdot}x_2)\boldsymbol{\omega} + (x{\cdot}y_3 + y{\cdot}x_3 + 3(x_1{\cdot}y_2\\
&\qquad\qquad + y_1{\cdot}x_2))\boldsymbol{\gamma},\\
&Y^{-1} = y^{-1} - y^{-2}{\cdot}y_1\boldsymbol{\varepsilon} + y^{-2}(\,2{\cdot}y_1^2{\cdot}y^{-1} - y_2)\boldsymbol{\omega} + a^{-2}(6y_1{\cdot}y^{-1}(y_2 - y_1^2{\cdot}y^{-1}) - y_3)\boldsymbol{\gamma},\\
&X/Y = X{\cdot}Y^{-1},
\end{aligned} \tag{7}$$

а функция гипер-дуального аргумента 3-го класса, после ее разложения в ряд Тейлора, имеет вид

$$\begin{aligned}
F(X) &= f(x) + x_1 f'(x)\boldsymbol{\varepsilon} + [x_2 f'(x) + x_1^2 f''(x)]\boldsymbol{\omega} +\\
&+ [x_3 f'(x) + 3x_1 x_2 f''(x) + x_1^3 f'''(x)]\boldsymbol{\gamma}
\end{aligned} \tag{8}$$

При $x_1 = 1$ и $x_2 = x_3 = 0$ соотношение (8) принимает форму:

$$F(x + \boldsymbol{\varepsilon} + 0\boldsymbol{\omega} + 0\boldsymbol{\gamma}) = f(x) + f'\boldsymbol{\varepsilon} + f''(x)\boldsymbol{\omega} + f'''(t)\boldsymbol{\gamma},\quad f(x) = F(x + 0\boldsymbol{\varepsilon} + 0\boldsymbol{\omega} + 0\boldsymbol{\gamma}),$$

что позволяет представить (6) следующим образом

$$D_n = \frac{F(X_n).Re}{F(X_n).Im1}, \;\; L_n = \; D_n \cdot \frac{F(X_n).Im2}{F(X_n).Im1}, \;\; K_n = D_n^2 \cdot \frac{F(X_n).Im3}{F(X_n).Im1} \; ,$$

где: $X_n = x_n + \boldsymbol{\varepsilon} \; + \; 0\boldsymbol{\omega} + 0\boldsymbol{\gamma}$, $F(X)$ – целевая функция $f(x)$, записанная в терминах гипер-дуальных чисел 3-го класса.
Таким образом, если представить функцию $f(x)$ в терминах гипер-дуальных чисел 3-го класса $F(X)$, то реализация методов (1)-(5) не вызывает никаких проблем.

Классическое определение корня функции – всякое значение $\bar{x}$, обращающее функцию $f(x)$ в нуль, т.е. такое, что $f(\bar{x}) = 0$, называется корнем уравнения $f(x) = 0$ или нулем функции $f(x)$. При этом нулем функции могут быть: 1) точка пересечения кривой $y = f(x)$ с осью абцисс ($f(\bar{x}) = 0$); 2) точкой перегиба ($f(\bar{x}) = 0, f'(\bar{x}) = 0$ и $f''(\bar{x}) = 0$); 3) точкой локального экстремума ($f(\bar{x}) = 0$ и $f'(\bar{x}) = 0$). Это обстоятельство требует тщательного подхода в выборе итерационного метода и его начального приближения x_0 для решения уравнения $f(x) = 0$.

Не менее важным вопросом является критерий окончания итерационного процесса. К сожалению, нет универсального рецепта в этом вопросе. Обычно в качестве этого критерия используется одно или комбинация из следующих неравенств: $\|f(x_n)\| \le \delta$; $\|f(x_n) - f(x_{n-1})\| \le \delta$; $n \ge Imax$ (δ – заданная малая величина, $Imax$ – максимально допустимое количество итераций).

Скорость сходимости того или иного итерационного метода, как првило, определяется его порядком. Однако для решения конкретной прикладной задачи в первую очередь важно время нужное для её решения, а не количество итераций необходимое для его достижения. Поэтому в качестве критерия скорости сходимости того или иного итерационного процесса будем считать время затраченное на решение.

Для реализации описанных выше итерационных схем с использованием гипер-дуальных чисел 3-го класса на языке C# составлен соответствующий программный код (см. Приложение 1). Процедура `Chebyshev2_3_4(…)` реализирует методы **MN**, **CH$_1$**, **CH$_2$**, а процедуры `NewtonEx(…)` и `Ch_H5(…)` осуществляют расчет по методам **MN$_c$** и **CH$_3$** соответственно. Там же представлены процедуры, описывающие тестируемые функции. Статическая библиотека Sdn3Lib.dll выполняет поддержку гипер-дуальных чисел 3-го класса.

Численный эксперимент проводился на компьютере с операционной системой Microsoft Windows 10 Pro, CPU 3.40 GHz, при этом использовались тестируемые функции представленные в следующей таблице.

	Тестируемые функции	x_0		Тип корня
1	$f_1(x) = (sin(x) - x/2)^2$	1.75	2.0	3
2	$f_2(x) = x^6 - 6x^5 + 50x^3 - 45x^2 - 108x + 108$	2.7	4.0	2
3	$f_3(x) = (x \cdot exp(x^2) - sin^2(x) + 3\,cos(x) + 5)^4$	-1.4	1.0	3
4	$f_4(x) = ln^2(x) \cdot (exp(x-3) - 1) \cdot \sin(\pi x/3)$	2.5	4.0	3
5	$f_5(x) = x^3 - 6x^2 + 11x - 6$	2.8	4.0	1

Таблица 1. Список тестируемых функций

Графики этих функций даны на рис. 1.

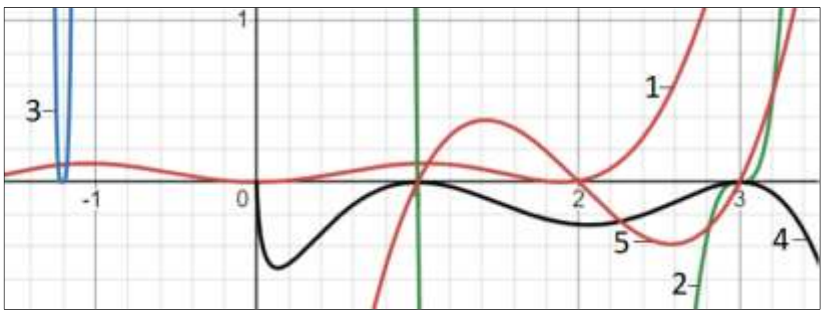

Рис.1. Графики тестируемых функций $f_1, \ldots, f_5$

Результататы численного эксперимента приведены в таблицах 2-6 (***t**ms* – время затраченное на выполнение соответствующей итерационной процедуры в миллисекундах).

Метод	$\overline{x}$	$f(\overline{x})$	i	t_{ms}
x_0=1.75				
MN_c	1.895494238333960	0.000000000000001	9	0.0054
MN	1.895494238333960	0.000000000000001	21	0.0046
CH_1	1.895494247673530	0.000000000000000	15	0.0033
CH_2	1.895494232307380	0.000000000000001	12	0.0028
CH_3	1.895494253754020	0.000000000000000	9	0.0039
x_0=2.0				
MN_c	1.895494294917640	0.000000000000001	9	0.0056
MN	1.895494294917640	0.000000000000001	21	0.0046
CH_1	1.895494284572520	0.000000000000000	15	0.0034
CH_2	1.895494297873340	0.000000000000001	12	0.0029
CH_3	1.895494275672970	0.000000000000000	9	0.0047

Таблица 2. Сравнение методов для функции $f_1(x) = (sin(x) - x/2)^2$

Метод	$\bar{x}$	$f(\bar{x})$	i	t_{ms}
		x_0=2.7		
MN_c	2.999988427595390	0.000000000000000	11	0.0062
MN	2.999988427595390	0.000000000000000	24	0.0053
CH_1	2.999994451694560	0.000000000000000	17	0.0041
CH_2	2.999996435794310	0.000000000000000	14	0.0035
CH_3	2.999993826695700	0.000000000000000	9	0.0043
		x_0=4.0		
MN_c	2.999995569570010	0.000000000000000	20	0.0091
MN	2.999995569570010	0.000000000000000	42	0.0088
CH_1	3.000007843742280	0.000000000000000	19	0.0043
CH_2	2.999998583778610	0.000000000000000	72	0.0155
CH_3	3.000008311692730	0.000000000000000	9	0.0043

Таблица 3. Сравнение методов для функции $f_2(x) = x^6 - 6x^5 + 50x^3 - 45x^2 - 108x + 108$

Метод	$\bar{x}$	$f(\bar{x})$	i	t_{ms}
		x_0= -1.4		
MN_c	-1.207653094875800	0.000000000000000	17	0.0204
MN	-1.207654850766060	0.000000000000000	36	0.0173
CH_1	-1.207655508092150	0.000000000000001	24	0.0155
CH_2	-1.207654409488220	0.000000000000000	20	0.0103
CH_3	-1.207656113901290	0.000000000000001	8	0.0107
		x_0=1.0		
MN_c	-1.207653670120870	0.000000000000000	238	0.2203
MN	-1.207655617753790	0.000000000000001	478	0.2216
CH_1	-1.207641606315270	0.000000000000000	27	0.0127
CH_2	-1.207654870937930	0.000000000000000	26	0.0142
CH_3	-1.207643567310770	0.000000000000000	12	0.0119

Таблица 4. Сравнение методов для функции $f_3(x) = (x \cdot exp(x^2) - sin^2(x) + 3\,cos(x) + 5)^4$

Метод	$\bar{x}$	$f(\bar{x})$	i	t_{ms}
		x_0=2.5		
MN_c	2.999898342608470	0.000000000000000	13	0.0129
MN	2.999864457576890	0.000000000000000	28	0.0095
CH_1	2.999876546204740	0.000000000000000	19	0.0069
CH_2	2.999835412768340	0.000000000000001	15	0.0056
CH_3	2.999921706602030	0.000000000000000	7	0.0057
		x_0=4.0		
MN_c	3.000157080983660	-0.000000000000001	13	0.0132
MN	3.000157080983660	-0.000000000000001	29	0.0096

CH_1	3.000127994706770	0.000000000000001	20	0.0072
CH_2	3.000157844586090	-0.000000000000001	16	0.0059
CH_3	3.000111245527990	0.000000000000001	8	0.0059

Таблица 5. Сравнение методов для функции $f_4(x) = ln^2(x) \cdot (exp(x-3)-1) \cdot sin(\pi x/3)$

Метод	$\overline{\boldsymbol{x}}$	$f(\overline{\boldsymbol{x}})$	$\boldsymbol{i}$	$\boldsymbol{t}_{ms}$
x_0=2.8				
MN_c	3.000000000000000	0.000000000000000	1	0.0024
MN	3.000000000000000	0.000000000000000	5	0.0016
CH_1	3.000000000000000	0.000000000000007	4	0.0013
CH_2	3.000000000000000	-0.000000000000007	3	0.0011
CH_3	3.000000000000000	0.000000000000004	2	0.0014
x_0=4.0				
MN_c	3.000000000000000	0.000000000000000	2	0.0028
MN	3.000000000000000	0.000000000000004	6	0.0014
CH_1	3.000000000000000	0.000000000000004	4	0.0011
CH_2	3.000000000000000	0.000000000000000	3	0.0009
CH_3	3.000000000000000	-0.000000000000004	2	0.0011

Таблица 6. Сравнение методов для функции $f_5(x) = x^3 - 6x^2 + 11x - 6$

Анализ полученных результатов тестируемых функций по критерию быстродействия $\boldsymbol{t}_{ms}$ показал:

1) методы **MN** и **$\mathbf{MN_c}$** не являются конкурентами для остальных рассматриваемых методов;
2) метод **$\mathbf{CH_1}$** вошел в тойку лидеров в одном случае из пяти;
3) метод **$\mathbf{CH_2}$** стал лидером во всех пяти случаях;
4) метод **$\mathbf{CH_3}$** разделил лидерство с методом **$\mathbf{CH_2}$** в трех случаях из пяти.

Таким образом, несмотря на то что метод **$\mathbf{CH_3}$** является методом 5-го порядка в ряде случаев он уступает методу 4-го порядка **$\mathbf{CH_2}$**.

Очевидно, помимо всего прочего, быстродействие итерационного метода зависит от сложности функции цели и от количества обращений к ней её производным на каждом шаге итерации. Более того, можно несколько улучшить производительность конкретного итерационного метода, если использовать более подходящий класс гипер-дуальное чисел.

Так например, для методов **MN** и **$\mathbf{MN_c}$** применить гипер-дуальные числа 1-го класс, учитывающего только первые производные, а для методов **$\mathbf{CH_1}$** и **$\mathbf{CH_3}$** гипер-дуальные числа 2-го класс, учитывающего только первые и вторые производные.

Приложение 1.

Исходный код на C#, использующий статическую библиотеку Sdn3Lib.dll.

```
using System;
using System.Diagnostics;
using Sdn3Lib;

const double δ = 1e-15;
const int maxIter = 1000;

enum Order {
        order2,
        order3,
        order4
}

struct tuple {
        public double x;    public sdn3 y;  public int i;
        public tuple(double value1, sdn3 value2, int value3){
            x = value1;
            y = value2;
            i = value3;
        }
}

//------------ непрерывный аналог метода Ньютона
   tuple NewtonEx( Func<double, sdn3> f,double x0){
        int i = -1;  double Sn = 0;  double Xn = x0;  sdn3 Fn = f(Xn);
        Sn = Xn - Fn.re / Fn.im1;
        while (true){
            Fn = f(Sn);
            Xn = Sn - Fn.re / Fn.im1;
            Fn = f(Xn);
            if (Math.Abs(Fn.re) <= δ || i > maxIter || double.IsNaN(Xn))
            {return new tuple(Xn, Fn, i);}
            Sn = Xn - Fn.re / Fn.im1;
            i += 1;
        }

//------------ методы Ньютона, Чебышева 3-го и 4-го порядков
tuple Chebyshev2_3_4(Func<double, sdn3> f, double x0, Order order){
        int i = -1; double D = 0.0, L = 0.0, K = 0.0, x = x0; sdn3 F = f(x);
        do
        {   D = F.re / F.im1;  L = D * F.im2 / F.im1;
            switch (order)
```

```
            {
                case Order.order2: x -= D;  break;
                case Order.order3: x -= (1.0 + 0.5 * L) * D;  break;
                case Order.order4:
                    K = D * D * F.im3 / F.im1;
                    x -= (1.0 + 0.5 * (L + L * L) - K / 6.0) * D;
                    break;
            }
            i += 1;
            if (i > maxIter || double.IsNaN(x)) break;
            F = f(x);
        } while (Math.Abs(F.re) > δ);
        return new tuple(x, F, i);
    }

//------------ метод Чебышева-Хэлли 5-го порядка
tuple Ch_H5(Func<double, sdn3> f, double x0){
        int i = -1;double xi = x0; sdn3 F = new sdn3(); sdn3 Φ = new sdn3();
        double L, D, M;
        while (true){
            i += 1;
            F = f(xi);
            D = F.re / F.im1;
            L = D * F.im2 / F.im1;
            xi -= (1.0 + 0.5 * L / (1.0 - L)) * D;
            Φ = f(xi);
            if (Math.Abs(Φ.re) <= δ || i > maxIter || double.IsNaN(xi))
               { break; }
            M = L * (1.0 - Φ.re / F.re);
            xi -= (1.0 + M / (1.0 - M)) * Φ.re / F.im1;
        }
        return new tuple(xi, Φ, i);
    }

//------------ тестируемые функции
sdn3 sin(double x){
        var X = new sdn3(x, 1.0, 0.0, 0.0);
        var sX = (sdn3.sin(X) - 0.5 * X);
        return sX*sX;
    }

    sdn3 x6(double x){
        var A = new sdn3(108.0, 0.0, 0.0, 0.0);
        var X = new sdn3(x, 1.0, 0.0, 0.0);
        var X2 = X * X;
        var X3 = X * X2;
        var X5 = X2 * X3;
        var X6 = X5 * X;
        return X6 - 6.0*X5 + 50.0*X3 - 45.0*X2 - A*X + A;
```

```
    }

    sdn3 xexp(double x){
        var A = new sdn3(5.0, 0.0, 0.0, 0.0);
        var X = new sdn3(x, 1.0, 0.0, 0.0);
        var Z = X * sdn3.exp(X * X) - sdn3.sin(X) * sdn3.sin(X) +
                3.0 * sdn3.cos(X) + A;
        return Z * Z * Z * Z;
    }

    sdn3 ln(double x){
        var X = new sdn3(x, 1.0, 0.0, 0.0);
        var A1 = new sdn3(1.0, 0.0, 0.0, 0.0);
        var A2 = new sdn3(2.0, 0.0, 0.0, 0.0);
        var A3 = new sdn3(3.0, 0.0, 0.0, 0.0);
        var sin = sdn3.sin((Math.PI / 3.0) * X);
        var ln1 = sdn3.ln(X - A2);
        return ln1 * ln1 * (sdn3.exp(X - A3) - A1) * sin;
    }

    sdn3 x3(double x){
        var A = new sdn3(6.0, 0.0, 0.0, 0.0);
        var X = new sdn3(x, 1.0, 0.0, 0.0);
        var X2 = X * X;
        var X3 = X * X2;
        return X3 - 6.0 * X2 + 11.0 * X - A;
    }

//------------ пример вызова метода Chebyshev2_3_4(...) с функцией xexp(...)
        Stopwatch sw = Stopwatch.StartNew();
          tuple R4 = Chebyshev2_3_4(xexp, 2.5, Order.order4);
              sw.Stop();
          string t = (string)sw.Elapsed.TotalMilliseconds.ToString();
         //  R4.x - root , xexp(R4.x)= R4.y.re,  R4.i- iterations, t - ms
```

Литература

1. *Тынкевич М. А., Пимонов А. Г.* Введение в численный анализ. - КузГТУ. – Кемерово, 2017. - 176 с.

2. *Жанлав Т., Чулуунбаатар О.* О некоторых итерационных методах высокого порядка сходимости для решения нелинейных уравнений. - Вестник РУДН. Серия Математика. Информатика. Физика. No 4. 2009. С. 47–55.

3. *Olifer V.I.* Implementation of chebyshev-halley type methods based on hyper-dual numbers. –

URL:http://viosolutions.amerihomesrealty.com/pdf/Implementation_of_chebyshev-halley_type_methods_based_on_hyper-dual_numbers.pdf

4. *Антонюк П.* О первой работе П. Л. Чебышева. – Научно-Исслед. Семинар по истории и методологии мат. и мех.: МГУ, 2014. –14 с.

5. *Jisheng Kou, Yitian Li.* The improvements of Chebyshev–Halley methods with fifth-order convergence. Applied Mathematics and Computation 188 (2007) 143–147.

6. *Олифер В. И.* Автоматическое дифференцирование на основе супер-дуальных чисел. – URL:http://viosolutions.amerihomesrealty.com/pdf/Автоматическое_дифференцирование_на_основе_супер-дуальных_чисел.pdf, 2020.

Абстракт

В данной публикации производится анализ быстродействия итерационных методов 2-го – 5-го порядков для решения нелинейных уравнений. Используется автоматическое дифференцирование со специальными дуальными чиселами (супер-дуальными чиселами 3-го класса). Представлена компьютерная реализация этого подхода для языка C# операционной системы Windows 10 Pro. Проведены численные эксперименты.

This publication analyzes the performance of iterative methods of the 2nd to 5th orders for solving nonlinear equations. Automatic differentiation with special dual numbers (super-dual numbers of the 3rd class) is used. A computer implementation of this approach for the C# language of the Windows 10 Pro operating system is presented. Numerical experiments are conducted.

Ключевые слова: *итерационные методы, автоматическое дифференцирование, супер-дуальные числа, iterative methods, automatic differentiation, super-dual numbers.*

---

20 февраля 2025 г