

ЧИСЛЕННОЕ РЕШЕНИЕ КОМПЛЕКСНЫХ УРАВНЕНИЙ МЕТОДОМ ЧЕБЫШЕВА 3-ГО ПОРЯДКА

Олифер В.И.

Рассмотрим комплексное уравнение:

$$f(z) = 0; \quad z = x + yi \in \mathbb{C}, \quad (x, y) \in \mathbb{R}, \quad i^2 = -1,$$

где $f(z)$ – функция комплексного аргумента.

Во многих случаях получение решения такого уравнения в замкнутом виде проблематично или вообще невозможно. Наиболее распространённый подход основан на разделении действительной и мнимой частей:

$$f(z) = f(x + yi) = u(x, y) + v(x, y)i = 0,$$

что приводит к системе двух вещественных уравнений:

$$\begin{cases} u(x, y) = 0 \\ v(x, y) = 0 \end{cases}$$

Однако в общем случае функции $u(x, y)$ и $v(x, y)$ являются нелинейными композициями элементарных функций, что делает решение системы сложным и вычислительно затратным. Поэтому для нахождения корней комплексных уравнений широко применяются итерационные численные методы различного порядка. Их использование требует вычисления производных первого и второго порядка, а также выбора подходящего начального приближения.

В настоящей работе предлагается метод решения комплексного уравнения $f(z) = 0$, основанный на методе Чебышёва третьего порядка, адаптированном для комплексных функций. Для точного вычисления производных используется автоматическое дифференцирование, реализованное на основе гипер-дуальных чисел над полем комплексных чисел. Для выбора начального приближения применяется метод разделения переменных с последующей визуализацией функций $u(x, y)$ и $v(x, y)$ с помощью графического калькулятора (например, Desmos).

Для реализации этой цели вводится новый тип гипер-дуального числа [1], использующий вместо вещественных чисел комплексные числа.

1. Алгебра гипер-дуальных чисел над полем комплексных чисел

Рассмотрим алгебру гипер-дуальных чисел второго порядка над полем комплексных чисел (CHDN). Элемент этой алгебры имеет вид:

$$X = x + x_1\epsilon + x_2\omega, \quad (x, x_1, x_2) \in \mathbb{C},$$

где: ε и ω – нильпотентные элементы, удовлетворяющие соотношениям, $\varepsilon^2 = 2\omega$, $\varepsilon\omega = \omega^2 = 0$.

Для двух элементов $X = x + x_1\varepsilon + x_2\omega$ и $Y = y + y_1\varepsilon + y_2\omega$ операции сложения и вычитания определяются покомпонентно:

$$X \pm Y = x \pm y + (x_1 \pm y_1)\varepsilon + (x_2 \pm y_2)\omega \quad (1.0)$$

Операция умножения задаётся правилом

$$X \cdot Y = x \cdot y + (x \cdot y_1 + y \cdot x_1)\varepsilon + (x \cdot y_2 + 2x_1 \cdot y_1 + y \cdot x_2)\omega \quad (1.1)$$

Пусть $F(X)$ – функция гипер-дуального комплексного аргумента.

Разложение $F(X)$ в ряд Тейлора с точностью до второго порядка даёт

$$F(X) = f(x) + x_1 \cdot f'(x)\varepsilon + (x_2 \cdot f'(x) + x_1^2 \cdot f''(x))\omega, \quad (1.2)$$

где $f(x) = F(x + 0\varepsilon + 0\omega)$, $f'(x) = \frac{df(x)}{dx}$, $f''(x) = \frac{d^2f(x)}{dx^2}$

При выборе $x_1 = 1$ и $x_2 = 0$ выражение (1.2) принимает вид:

$$F(X) = f(x) + f'(x)\varepsilon + f''(x)\omega, \quad (1.3)$$

что позволяет автоматически вычислять значения функции, её первой и второй производных в одной операции.

2. Алгебра комплексных чисел.

Алгебра комплексных чисел следует её классическому варианту [3, 4].

Каждое комплексное число представляется в виде

$$z_1 = x_1 + iy_1; \quad x, y \in \mathbb{R}, \quad i^2 = -1$$

Для двух комплексных чисел

$$z_1 = x_1 + iy_1, \quad z_2 = x_2 + iy_2,$$

операции сложения и умножения имеют вид

$$\begin{aligned} z_1 \pm z_2 &= (x_1 \pm x_2) \pm i(y_1 \pm y_2), \\ z_1 \cdot z_2 &= (x_1 \cdot x_2 - y_1 \cdot y_2) + i(x_1 \cdot y_2 + x_2 \cdot y_1), \end{aligned}$$

В дальнейшем эти соотношения используются при вычислении значений функции $f(z)$, а также её первых и вторых производных, необходимых для реализации формулы (1.3) метода Чебышёва третьего порядка.

Ниже приведены производные некоторых часто используемых комплексных функций.

Функция $f(z)$	Первая производная $f'(z)$	Вторая производная $f''(z)$
z^k	$k \cdot z^{k-1}$	$(k-1)k \cdot z^{k-2}$
a^z	$a^z \cdot \ln a$	$a^z \cdot \ln^2 a$
e^z	e^z	e^z
$\ln(z)$	$1/z$	$-1/z^2$

$\sin(z)$	$\cos(z)$	$-\sin(z)$
$\cos(z)$	$-\sin(z)$	$-\cos(z)$
$sh(z)$	$ch(z)$	$sh(z)$
$ch(z)$	$sh(z)$	$ch(z)$

Таблица 1. Первые и вторые производные некоторых базовых комплексных функций

3. Метод Чебышева

Метод Чебышёва является итерационным методом решения нелинейных уравнений [2], обладающим *кубической сходимостью* при условии, что искомый корень является простым и функция $f(z)$ достаточно гладкая. Для комплексных функций итерационная формула имеет вид

$$z_{n+1} = z_n - K(z_n) \cdot [1 + K(z_n) \cdot L(z_n)/2] \quad (3.1)$$

где: $K(z_n) = f(z_n) / f'(z_n)$, $L(z_n) = f'(z_n) / f''(z_n)$, $n = 0, 1, 2, \dots$

Формула (3.1) является комплексным аналогом классического метода Чебышёва и требует вычисления значений функции $f(z)$, её первой производной $f'(z)$ и второй производной $f''(z)$. В настоящей работе эти величины вычисляются автоматически с использованием гипер-дуальных чисел второго порядка (см. раздел 1), что обеспечивает точность и устойчивость вычислений.

Итерационный процесс продолжается до выполнения условия

$$|z_{n+1} - z_n| < \delta,$$

где: δ — заданная малая величина, определяющая требуемую точность решения.

Метод Ньютона является частным случаем метода Чебышёва. Если в формуле (3.1) положить $L(z_n) \equiv 0$, то получаем классическую итерацию Ньютона:

$$z_{n+1} = z_n - K(z_n), \quad (3.2)$$

которая обладает *квадратичной* сходимостью.

Таким образом, метод Чебышёва представляет собой естественное развитие метода Ньютона, обеспечивая более высокую скорость сходимости за счёт использования второй производной функции.

4. Определение начального приближения

Пусть дана функция комплексной переменной $f(z)$, $z = x + yi$.

Представим её в виде

$$f(x + yi) = u(x, y) + v(x, y)i; \quad x, y \in \mathbb{R}$$

где: $u(x, y) + v(x, y)i$ – действительная и мнимая части соответственно.

Подставляя $z = x + yi$ в исходное уравнение и далее используя формулы Эйлера (для экспонент и тригонометрических функций) или раскрытие скобок, явно выделяем функции $u(x, y)$ и $v(x, y)$.

Тогда уравнение $f(z) = 0$ эквивалентно системе двух неявных уравнений:

$$u(x, y) = 0, \quad v(x, y) = 0 \quad (4.1)$$

Таким образом, задача нахождения комплексного корня сводится к поиску точки пересечения двух кривых на плоскости (x, y) .

Для получения надёжного начального приближения удобно использовать графический анализ функций $u(x, y)$ и $v(x, y)$. Построив линии уровня, используя (4.1), можно визуально определить точки их пересечения. Каждая такая точка является приближением корня уравнения $f(z) = 0$. Этот подход особенно полезен при наличии нескольких корней, когда аналитическое выделение областей притяжения затруднено.

Графический метод позволяет:

- оценить количество корней;
- определить их примерное расположение;
- выбрать стартовые точки, обеспечивающие быструю сходимость итерационных методов.

Для корректного применения методов Чебышёва–Хэлли [2] необходимо, чтобы функция $f(z)$ была дифференцируемой в смысле комплексного анализа. Это требует выполнения условий Коши–Римана:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}$$

Эти условия гарантируют существование комплексной производной и обеспечивают корректность вычисления производных с использованием гипер-дуальных чисел. Нарушение условий Коши-Римана приводит к тому, что производные по x и y перестают согласовываться, что делает применение высокопорядковых итерационных методов некорректным.

Использование гипер-дуальных чисел позволяет автоматически вычислять первые и вторые производные функции $f(z)$, что критически важно для методов Чебышёва. Однако корректность этих производных напрямую зависит от выполнения условий Коши–Римана. Поэтому графический анализ и проверка комплексной дифференцируемости функции являются обязательными предварительными шагами перед запуском итерационного процесса.

5. Автоматическое дифференцирование

Традиционные способы вычисления производных комплексной функции $f(z)$ обладают рядом существенных недостатков. Вычисление производной в виде отдельной

программной процедуры требует предварительного анализа структуры функции, что делает процесс трудоёмким и подверженным ошибкам. Метод конечных разностей, основанный на приближённом вычислении производной, требует выбора оптимального шага приращения аргумента z , что связано с компромиссом между погрешностью усечения и погрешностью округления.

При вычислении производных более высоких порядков трудности возрастают многократно, особенно для сложных композиционных функций вида $f(f_1(f_2(f_3(f_4(f_5(\dots))))))$, где аналитическое получение производных становится практически невозможным.

Эти проблемы эффективно решаются методом *автоматического дифференцирования* (AD), позволяющим вычислять значения производных с машинной точностью, непосредственно из компьютерного кода функции [1], [5]. Автоматическое дифференцирование является фундаментальным инструментом в задачах оптимизации, решения нелинейных уравнений, численного интегрирования дифференциальных уравнений, анализа чувствительности и других областях вычислительной математики.

Для реализации AD необходимо определить новый тип данных и перегрузить базовые операции и элементарные функции таким образом, чтобы при одном обращении к функции вычислялись значения самой функции и её производных вплоть до заданного порядка. В настоящей работе для этой цели используется алгебра гипер-дуальных чисел второго порядка над полем комплексных чисел (см. раздел 1), позволяющая автоматически получать значения $f(z)$, $f'(z)$ и $f''(z)$ в одной операции.

6. Программная реализация

Реализация метода Чебышёва на базе гипер-дуальных чисел требует создания специализированного типа данных, обеспечивающего автоматическое вычисление производных первого и второго порядка. В рамках данной работы на языке C# была разработана структура CHDN, включающая перегруженные арифметические операции, функции элементарной математики и процедуру выполнения итерационного метода `Chebyshev()`.

Основная цель реализации заключалась не в создании универсальной библиотеки, а в формировании компактного и функционального инструмента, достаточного для проведения численных экспериментов, представленных в разделе 7 и Приложении 1. Поэтому структура CHDN включает только те элементы, которые необходимы для вычисления значений функции, её производных и выполнения итерационного шага метода Чебышёва.

Для операций над комплексными числами применялась стандартная библиотека `System.Numerics`, предоставляющая тип `Complex` и набор базовых функций. Это позволило сосредоточиться на реализации гипер-дуальной части, не дублируя функциональность, уже доступную в .NET.

Тип `Complex` использовался:

- для хранения значений функции $f(z)$,
- для вычисления промежуточных выражений,
- для реализации комплексных арифметических операций,
- для передачи комплексных аргументов в гипер-дуальные вычисления.

Следовательно, структура CHDN работает поверх стандартного комплексного типа, расширяя его возможностями автоматического дифференцирования.

Тип CHDN представляет собой расширение концепции гипер-дуальных чисел, включающее:

- основное значение $f(z)$,
- первую производную $f'(z)$,
- вторую производную $f''(z)$,
- смешанные производные при необходимости.

Для корректной работы метода Чебышёва требуется вычисление:

- $f(z)$,
- $f'(z)$,
- $f''(z)$.

Гипер-дуальные числа позволяют получить все эти значения автоматически, без ручного дифференцирования, что значительно снижает вероятность ошибок и упрощает реализацию.

В структуре CHDN были переопределены:

- операции сложения, вычитания, умножения и деления,
- операции возведения в степень,
- некоторые элементарные функции, если они требовались в тестах.

Каждая операция реализована таким образом, чтобы корректно обновлять значения производных согласно правилам дифференцирования гипер-дуальных чисел.

В процедуре `Chebyshev()`:

1. аргумент z_n преобразуется в гипер-дуальное число;
2. вычисляются $f(z_n)$, $f'(z_n)$, $f''(z_n)$;
3. выполняется итерационный шаг;
4. результат возвращается как новое комплексное число.

Процедура реализована так, чтобы её можно было применять к любой функции, определённой пользователем, при условии, что она поддерживает вычисления через CHDN. Разработанная структура не претендует на полноту или универсальность. Она создавалась как экспериментальный инструмент для:

- проверки корректности гипер-дуального подхода,
- тестирования метода Чебышёва,
- сравнения с классическим методом Ньютона,
- анализа поведения итераций на комплексной плоскости.

Тем не менее, структура легко расширяется и может служить основой для полноценной библиотеки автоматического дифференцирования комплексных функций.

7. Численное тестирование

Для оценки эффективности метода Чебышёва третьего порядка были проведены численные эксперименты на ряде комплексных уравнений различной структуры. В таблице ниже приведены:

- тестируемые функции,
- выбранные начальные приближения,
- найденные корни,
- количество итераций для методов Чебышёва и Ньютона.

Учитывая, что список входных параметров процедуры `Chebyshev()` (см. Приложение 1) содержит ссылку на тестируемую функцию, то последняя должна быть описана в терминах CHDN.

Некоторые результаты численного тестирования приводятся в следующей таблице. В которой указано тестируемые функции, начальные приближения корней, значения найденных корней, количество итераций для методов Чебышева и Ньютона.

Уравнение	z_0	Корень		Шагов
$Z^3 - 1 = 0$	$0.0 + 0.5i$	Чебышев	$-0.49999999999999994 + 0.8660254037844386i$	7
		Ньютон	$-0.49999999999999998 + 0.8660254037844388i$	10
	$1.5 + 0.5i$	Чебышев	$1.0 + 0i$	5
		Ньютон	$1.0 + 0i$	7
$Z^3 - Z = 0$	$0.5 + 0.5i$	Чебышев	$0 + 0i$	6
		Ньютон	$0 + 0i$	7
	$2.0 + 1.0i$	Чебышев	$1.0 + 0i$	6
		Ньютон	$1.0 + 0i$	8
$e^Z - Z = 0$	$0.5 + 0.5i$	Чебышев	$0.3181315052047642 + 1.3372357014306893i$	5
		Ньютон	$0.3181315052047641 + 1.3372357014306895i$	7
	$0.5 - 0.5i$	Чебышев	$0.3181315052047642 - 1.3372357014306893i$	4
		Ньютон	$0.3181315052047641 - 1.3372357014306895i$	6
$e^Z + Z^2 - 2 = 0$	$0.5 + 0.5i$	Чебышев	$0.5372744491738566 + 0i$	4
		Ньютон	$0.5372744491738566 + 0i$	6
	$4.0 + 4.0i$	Чебышев	$3.6891253851203896 + 5.073306753386286i$	6
		Ньютон	$3.6891253851203896 - 5.073306753386286i$	8
$(Z - 1)^2(Z + 2i) = 0$	$0.3 - 1.0i$	Чебышев	$1.000000000000 + 0i$	26

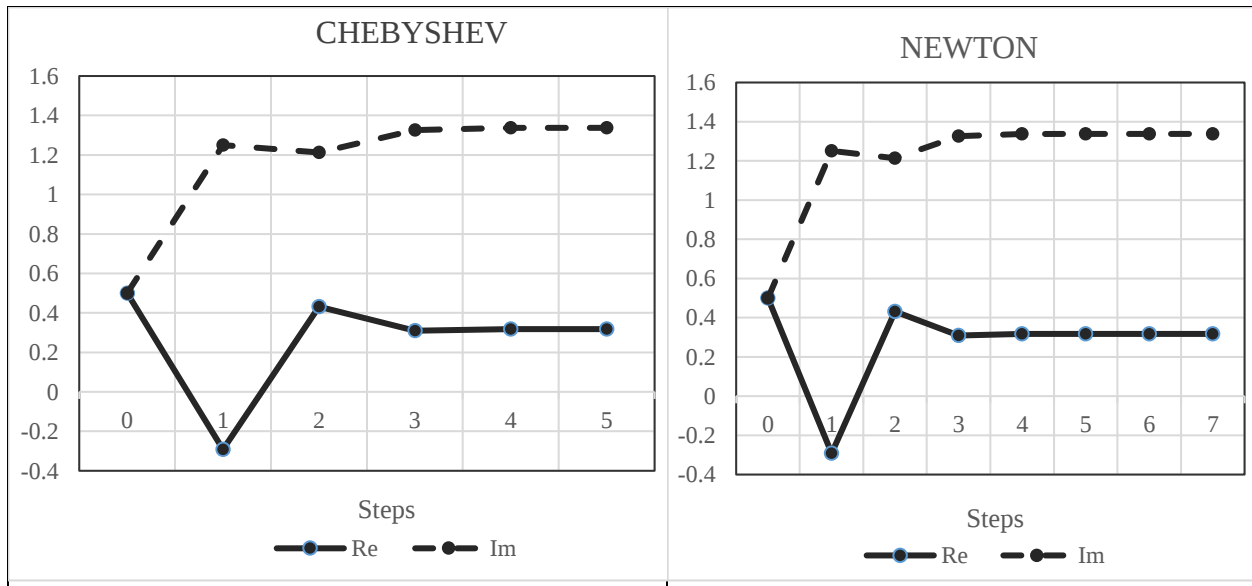
$Z^2i + (3 - 2i)Z - 6 = 0$	$0.3 - 3.0i$	Ньютон	$1.000000000000 + 0i$	34
		Чебышев	$0 - 2.0i$	5
		Ньютон	$0 - 2.0i$	7
	$1.0 + 2.0i$	Чебышев	$0 + 3.0i$	7
		Ньютон	$0 + 3.0i$	8
	$1.0 + 1.0i$	Чебышев	$2.0 + 0i$	7
		Ньютон	$1.999999999999 + 0i$	8

Табл. Результаты расчёта тестируемых комплексных функций

Для всех рассмотренных примеров наблюдается характерная особенность: на начальных шагах методы Чебышёва и Ньютона демонстрируют схожее поведение, однако на заключительных этапах метод Чебышёва существенно опережает метод Ньютона.

Это связано с тем, что метод Чебышёва обладает кубической сходимостью, тогда как метод Ньютона — квадратичной.

Для наглядности ниже в приведены графики сходимости для уравнения $e^Z - Z = 0$.

Рис. Характер сходимости функции $e^Z - Z = 0$

До третьего шага оба метода ведут себя одинаково, после чего метод Чебышёва начинает сходиться примерно в два раза быстрее.

8. Заключение

- В работе представлен подход к численному решению нелинейных уравнений комплексной переменной с использованием гипер-дуальных чисел и метода Чебышёва.

Применение гипер-дуальных чисел обеспечивает автоматическое вычисление производных первого и второго порядка, что позволяет реализовать высокопорядковые итерационные методы без ручного дифференцирования.

- Разработанная структура CHDN на языке C# включает необходимые перегруженные операции и процедуру `Chebyshev()`, достаточные для проведения численных экспериментов. Использование стандартного типа `Complex` из библиотеки `System.Numerics` позволило сосредоточиться на автоматическом дифференцировании и избежать дублирования функциональности комплексной арифметики.
- Показано, что выбор начального приближения можно эффективно осуществлять через графический анализ линий уровня действительной и мнимой частей функции. Проверка условий Коши–Римана гарантирует корректность комплексной дифференцируемости, что является необходимым условием для применения гипер-дуальных чисел и высокопорядковых методов.
- Численные эксперименты подтверждают высокую точность и быструю сходимость метода Чебышёва при использовании гипер-дуальных чисел.
- Перспективы развития подхода включают расширение на системы нелинейных уравнений комплексной переменной, исследование областей притяжения итерационных методов, создание универсальной библиотеки автоматического дифференцирования комплексных функций и применение гипер-дуальных чисел в задачах оптимизации комплексных функций.

ПРИЛОЖЕНИЕ 1.

Код программы на C#

```
using System;
using System.Numerics;
using System.Diagnostics;
using c = System.Numerics.Complex;

public struct Cdn    // struct of CHDN type
{
    public enum Method
    {
        Newton,      // 0
        Chebyshev     // 1
    }

    public Complex Re, Im1, Im2;
    public Cdn(Complex x, Complex x1, Complex x2) { Re = x; Im1 = x1; Im2 = x2;}
    public Cdn(Complex x ) { Re = x; Im1 = Complex.Zero; Im2 = Complex.Zero;}
    public Cdn(Complex x, Complex x1) { Re = x; Im1 = x1; Im2 = Complex.Zero;}
}
```

```

#region OPERATORS (overloading):
public static Cdn operator -(Cdn X) { return new Cdn(-X.Re, -X.Im1, -X.Im2); }
public static Cdn operator +(Cdn X, Cdn Y)
{ return new Cdn(X.Re + Y.Re, X.Im1 + Y.Im1, X.Im2 + Y.Im2); }
public static Cdn operator -(Cdn X, Cdn Y)
{ return new Cdn(X.Re - Y.Re, X.Im1 - Y.Im1, X.Im2 - Y.Im2); }

public static Cdn operator *(Cdn X, Cdn Y)
{ return new Cdn(X.Re * Y.Re,
    X.Re * Y.Im1 + Y.Re * X.Im1,
    X.Re * Y.Im2 + 2.0 * X.Im1 * Y.Im1 + Y.Re * X.Im2);
}
public static implicit operator Cdn(double value) => new(new Complex(value, 0),
    Complex.Zero);
public static implicit operator Cdn(Complex value) => new(value, Complex.Zero);
#endregion

#region PUBLIC FUNCTIONS:
public static Cdn Inverse(Cdn X)
{
    Complex y0 = X.Re, y1 = X.Im1, y2 = X.Im1, y3 = y0 * y0 * y0;
    return new Cdn((y0 * y0) / y3, (y0 * y1) / y3, (2.0 * y1 * y1 - y0 * y2) / y3);
}

public static Cdn exp(Cdn X) {
    var ex = Complex.Exp(X.Re);
    return new Cdn(ex, ex, ex);
}
public static Complex Chebyshev(Func<Cdn, Cdn> F, Complex z,
    Method m = Method.Chebyshev, int maxIter=50, double δ = 1e-10 )
{
    int step = 0;
    for (step = 0; step < maxIter; step++)
    {
        Cdn zDual = new Cdn(z, Complex.One), f = F(zDual);
        Complex val = f.Re, der1 = f.Im1, der2 = f.Im2;
        if (der1.Magnitude < 1e-12){Debug.WriteLine("Error: der1≈0."); break;}
        Complex zNext = z - (val / der1) * (1.0 +
            ((m == Method.Newton) ? 0.0: (val / der1) * der2 / (2.0 * der1)));
        Debug.WriteLine($"Step {step}: z = {zNext}, |f(z)| = {val.Magnitude:E4}");
        if ((zNext - z).Magnitude < δ)
        {
            Debug.WriteLine($"Convergence achieved at step {step + 1}!");
            Debug.WriteLine($"The root found: {zNext}"); break;
        }
        z = zNext;
    }
    return z;
}

public static int mRoot(Cdn Z) // get multiplicity of a root
{
    int m = 1;
    if (Z.Im1 == Complex.Zero)
    {
        if (Z.Im2 != Complex.Zero) m = 2;
        else m = 3;
    }
    return m;
}

public override string ToString() => $"{Re} + {Im1}ε + {Im2}ω";
}

```

//ПРИМЕР РАСЧЁТА

```
Cdn MyFunc (Cdn Z) { return Cdn.exp(Z) - Z; } //F(Z) = exp(Z)-Z
с zo = new с(0.5, 0.5); Func<Cdn, Cdn> fun = MyFunc;
Debug.WriteLine("CHEBYSHEV");
var Rch = Cdn.Chebyshev(fun, zo, Cdn.Method.Chebyshev);
Debug.WriteLine("\nNEWTON");
var Rn = Cdn.Chebyshev(fun, zo, Cdn.Method.Newton);
```

ЛИТЕРАТУРА

1. Олифер В.И. Усеченные гипер-дуальные числа в автоматическом дифференцировании.
URL: http://viosolutions.amerihomesrealty.com/pdf/усеченные_гипер-дуальные_числа_в_автоматическом_дифференцировании.pdf, – 2020.
2. Олифер В.И. 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, – 2023.
3. Краснов М. Л., Киселев А. И., Макаренко Г. И. Функции комплексного переменного. Операционное исчисление. Теория устойчивости. – М.: Наука, 1981. – 304 с.
4. Минькова Р.М. Функции комплексного переменного и операционное исчисление. – Екатеринбург: УГТУ-УПИ, 2013. – 72 с.
5. Naumann U. The Art of Differentiating Computer Programs. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, – 2012.

Абстракт

В статье рассматривается метод решения комплексных уравнений с использованием гипер-дуальной алгебры над полем комплексных чисел, метода Чебышева и автоматического дифференцирования. Приведен код программного обеспечения и рассмотрены численные примеры, полученные на его основе.

The article discusses a method for solving complex equations using hyper-dual algebra over the field of complex numbers, the Chebyshev method, and automatic differentiation. Software code is provided, and numerical examples obtained using it are examined.

Ключевые слова *комплексные уравнения, гипер-дуальные числа, метод Чебышева, автоматическое дифференцирование, complex equations, hyper-dual numbers, Chebyshev method, automatic differentiation*