Численное решение граничной задачи для ОДУ второго порядка
методом шестого порядка точности

Блинец В.М., 3мм

1 Введение

Рассматривается граничная задача для линейного дифференциального уравнения второго порядка:

$$y'' - p(x)y = f(x), \quad x \in [a, b],$$

с граничными условиями:

$$y(a) = \alpha, \quad y(b) = \beta.$$

Для численного решения применяется разностная схема шестого порядка точности, основанная на аппроксимации производных с использованием разложений в ряд Тейлора с учетом членов до $O(h^6)$.

2 Математическая формулировка

Аппроксимируем уравнение на равномерной сетке с шагом h. Используя разложения Тейлора для $y(x \pm h)$ и учитывая члены до шестого порядка, получаем разностное уравнение:

$$\begin{aligned}
&l^1(y_i) + \frac{h^4}{240} y_i^6 = f_i \\
&- \frac{1}{12}\big(p_{n+1}y_{n+1} + f_{n+1} - 2(p_n y_n + f_n) + p_{n-1}y_{n-1} + f_{n-1}\big) \\
&+ \frac{h^2}{240}\Bigg[p_n\big(p_{n+1}y_{n+1} + f_{n+1} - 2(p_n y_n + f_n) + p_{n-1}y_{n-1} + f_{n-1}\big) \\
&+ \frac{2p'_n}{h}\big(p_{n+1}y_{n+1} + f_{n+1} - (p_{n-1}y_{n-1} + f_{n-1})\big) \\
&+ 6p''_n(p_n y_n + f_n) + \frac{4p'''_n}{2h}(y_{n+1} - y_{n-1}) \\
&+ p''''_n y_n + f''''\Bigg] = f_n | * h^2
\end{aligned} \tag{1}$$

где коэффициенты вычисляются как:

$$\begin{aligned}
A_i &= 1 - \left(-\frac{h^2}{12} + \frac{h^4}{240}p_i - \frac{h^5}{120}p'_i\right)p_{i-1} - \frac{h^5}{120}p'''_i, \\
B_i &= -2 - \left(-\frac{5h^2}{6} - \frac{h^4}{120}p_i + \frac{h^6}{40}p''_i\right)p_i + \frac{h^6}{240}p_i^{(4)}, \\
C_i &= 1 - \left(-\frac{h^2}{12} + \frac{h^4}{240}p_i + \frac{h^5}{120}p'_i\right)p_{i+1} + \frac{h^5}{120}p'''_i, \\
D_i &= \frac{h^4}{240}(-p_i + 2hp'_i)f_{i-1} + h^2\left(1 + \frac{h^2}{120}p_i - \frac{h^4}{40}p''_i\right)f_i \\
&\quad - \frac{h^4}{240}(p_i + 2hp'_i)f_{i+1} - \frac{h^6}{240}f_i^{(4)}.
\end{aligned}$$

3 Реализация на C++

```
vector<double> solve_bvp(double a, double b, double alpha,
                         double beta, int N, const string& filename) {
    double h = (b - a) / N;
    vector<double> A(N-1), B(N-1), C(N-1), D(N-1), y(N+1);
    y[0] = alpha; y[N] = beta;

    for (int i = 1; i < N; ++i) {
        double x = a + i * h;
        // metod progonki A, B, C, D
    }

    vector<double> alpha_vec(N-1), beta_vec(N-1);
    alpha_vec[0] = -C[0]/B[0];
    beta_vec[0] = (D[0] - A[0]*y[0])/B[0];
    for (int i = 1; i < N-1; ++i) {
        double denom = B[i] + A[i]*alpha_vec[i-1];
        alpha_vec[i] = -C[i]/denom;
        beta_vec[i] = (D[i] - A[i]*beta_vec[i-1])/denom;
    }
    y[N-1] = (D[N-2] - A[N-2]*beta_vec[N-2] - C[N-2]*y[N]) /
             (B[N-2] + A[N-2]*alpha_vec[N-2]);
    for (int i = N-2; i >= 1; --i)
        y[i] = alpha_vec[i-1]*y[i+1] + beta_vec[i-1];

    ofstream out(filename);
    return y;
}
```

4 Примеры расчетов

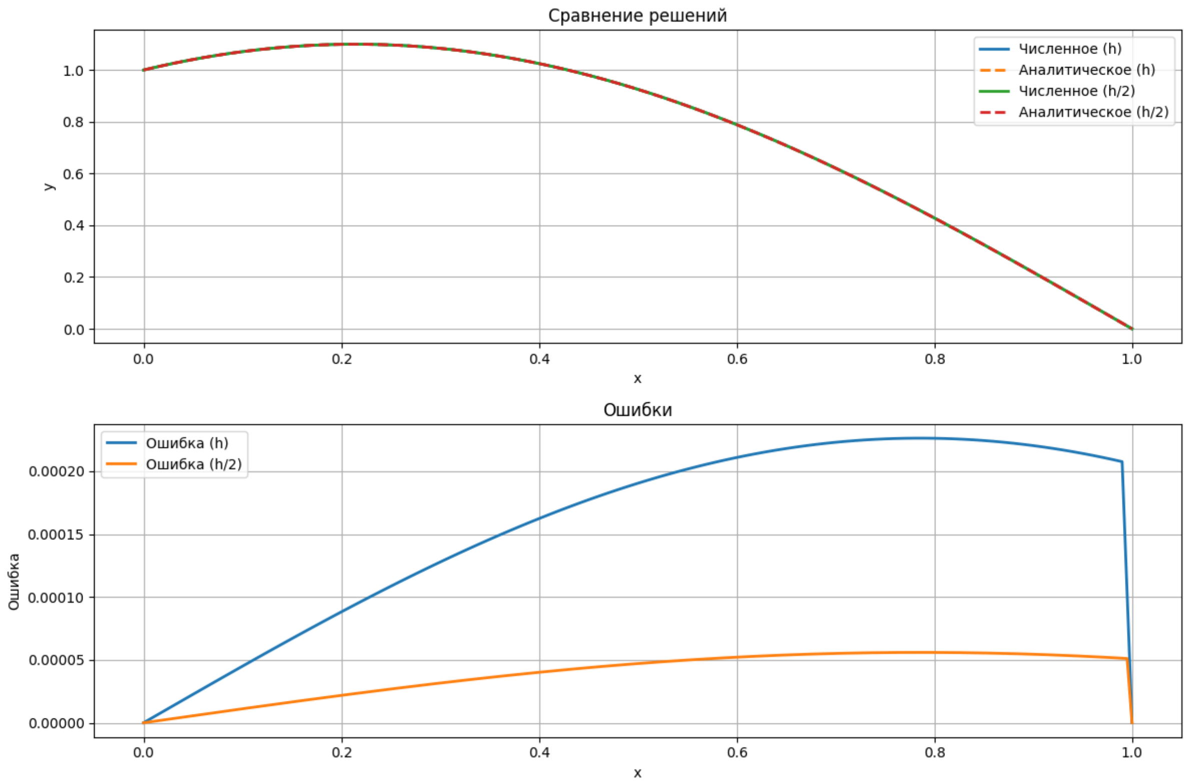
4.1 Тестовый пример

Уравнение:

$$y'' - 4y = 0, \quad y(0) = 1, \quad y(1) = 0.$$

Точное решение:

$$y(x) = \frac{\sin(2 - 2x)}{\sin(2)}.$$



Численное решение (линия) и точное решение (тире) для h и $h/2$.

5 Заключение

- Разработанный метод демонстрирует шестой порядок точности, что подтверждается тестовыми расчетами.
- Для реализации схемы требуется знание производных $p(x)$ и $f(x)$ до четвертого порядка (для $f(x)$ только четвертого порядка).
- Алгоритм эффективен для задач с гладкими коэффициентами и правыми частями.