

Разностный метод решения краевой задачи для обыкновенного дифференциального уравнения второго порядка. Метод прогонки. Вариант 11.

Содержание

Комментарий к задаче.....	1
Точное решение.....	1
Алгоритм.....	1
Аппроксимация дифференциального уравнения разностными.....	1
Аппроксимация граничных условий.....	2
Итог.....	3
Решение с точностью $O(h^2)$ при $n = 10$	3
Определение сетки узлов.....	3
Поиск коэффициентов.....	3
Результат.....	4
Решение с точностью $O(h^2)$ при $n = 20$	4
Результат.....	4
Уточнение по Рундсону.....	5
Сравнение с точным значением.....	5
Таблица.....	5
Графики.....	5

Комментарий к задаче

Требуется получить численное решение следующей краевой задачи:

$$-\frac{7-x}{8+3x}u'' + \left(1 + \frac{x}{3}\right)u' + \left(1 - \frac{1}{2}e^{\frac{x}{2}}\right)u = \frac{1}{2} - \frac{x}{3}, \quad u(-1) = u(1) = 0$$

Заметим, что в граничных условиях $|\alpha_2| + |\beta_2| = 0 \implies$ суммарный порядок аппроксимации дифференциальной задачи разностной схемой второй уже на основной схеме. Поэтому все методы, описанные далее, реализованы на основной сетке и имеют порядок точности $O(h^2)$.

Точное решение

Получим "точное" решение краевой задачи встроенными методами языка MATLAB.

Примечание: см. acc_solution.m

```
Y_ex = acc_solution
```

```
Y_ex = 1x21
      0      0.0384      0.0733      0.1049      0.1330      0.1579      0.1796      0.1979 ...
```

Алгоритм

Аппроксимация дифференциального уравнения разностными

Дано дифференциальное уравнение второго порядка

$$-p(x)u'' + q(x)u' + r(x)u = f(x)$$

Пусть $p_i = (p(x_i))$, $q_i = (q(x_i))$, $r_i = (r(x_i))$, $f_i = (f(x_i))$. Заменяем производные конечно-разностными отношениями, получим

$$-p_i \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + q_i \frac{y_{i+1} - y_{i-1}}{2h} + r_i = f_i, \quad i = 1, \dots, n-1$$

Перегруппируем коэффициенты при y_i , получим

$$A_i y_{i-1} - B_i y_i + C_i y_{i+1} = G_i, \quad i = 1, \dots, n-1, \quad \text{где (1)}$$

- $A_i = -\left(\frac{p_i}{h^2} + \frac{q_i}{2h}\right)$
- $B_i = -\left(\frac{2p_i}{h^2} + r_i\right)$
- $C_i = \frac{q_i}{2h} - \frac{p_i}{h^2}$
- $G_i = f_i$

Аппроксимация граничных условий

Граничные условия имеют вид:

- $\alpha_1 y(a) + \alpha_2 y'(a) = \alpha$
- $\beta_1 y(b) + \beta_2 y'(b) = \beta$

Пусть $y_0 = y(a)$, $y_n = y(b)$. Заменяем производные конечно-разностными отношениями, получим

- $\alpha_1 y_0 + \alpha_2 \frac{y_1 - y_0}{h} = \alpha$
- $\beta_1 y_n + \beta_2 \frac{y_n - y_{n-1}}{h} = \beta$

Перегруппируем коэффициенты, получим

- $-B_0 y_0 + C_0 y_1 = G_0$
- $A_n y_{n-1} - B_n y_n = G_n$

Где (2)

- $B_0 = \alpha_1 + \frac{\alpha_2}{h}, C_0 = -\frac{\alpha_2}{h}$
- $A_n = \beta_1 + \frac{\beta_2}{h}, B_n = -\frac{\beta_2}{h}$

Итог

Получаем следующую систему (3):

$$\begin{cases} -B_0 y_0 + C_0 y_1 = G_0 \\ \dots \\ A_i y_{i-1} - B_i y_i + C_i y_{i+1} = G_i \\ \dots \\ A_n y_{n-1} - B_n y_n = G_n \end{cases}$$

Решим ее методом прогонки:

1) Найдем коэффициенты s_i, t_i по следующим формулам (5):

- $s_0 = \frac{C_0}{B_0}, t_0 = -\frac{G_0}{B_0}$
- $s_i = \frac{C_i}{B_i - A_i s_{i-1}}, t_i = \frac{A_i t_{i-1} - G_i}{B_i - A_i s_{i-1}}, i = 1, \dots, n$

2) Найдем y_i по следующим формулам (6):

- $y_n = t_n$
- $y_{i-1} = s_i y_{i+1} + t_i, i = 0, \dots, n-1$

Решение с точностью $O(h^2)$ при $n = 10$

Определение сетки узлов

Определим сетку узлов.

```
n1 = 10;  
a = -1;  
b = 1;  
h1 = (b-a)/n1;  
x1=a:h1:b;
```

Поиск коэффициентов

Найдем необходимые для решения задачи коэффициенты по формулам (1) и (2)

Комм: реализация описана в функции coefficients.m

```
[A1,B1,C1,G1] = coefficients(a,b,h1)
```

```
A1 = 1×11  
    0 -36.6548 -32.6452 -29.3725 -26.6577 -24.3750 -22.4341 -20.7681 ...  
B1 = 1×11  
 -5.0000 -70.3077 -61.9199 -55.0024 -49.1962 -44.2500 -39.9823 -36.2589 ...  
C1 = 1×11  
    0 -32.9881 -28.6452 -25.0392 -21.9910 -19.3750 -17.1008 -15.1014 ...
```

```
G1 = 1×11
      0      0.7667      0.7000      0.6333      0.5667      0.5000      0.4333      0.3667 ...
```

Результат

Решим систему (3) методом прогонки.

Комм. алгоритм реализован в функции TMA_algorithm.m

```
[Y_n1,s1,t1] = TMA_algorithm(A1,B1,C1,G1);
```

Представим результаты в следующей таблице

```
table1 = table(transpose(x1),transpose(A1),transpose(B1),transpose(C1),transpose(G1),transpose(s1),transpose(t1),transpose(Y1));
table1.Properties.VariableNames = {'x_i','A_i','B_i','C_i','G_i','s_i','t_i','Y_i'}
```

```
table1 = 11×8 table
```

	x_i	A_i	B_i	C_i	G_i	s_i	t_i	Y_i
1	-1.0000	0	-5.0000	0	0	0	0	0
2	-0.8000	-36.6548	-70.3077	-32.9881	0.7667	0.4692	0.0109	0.0733
3	-0.6000	-32.6452	-61.9199	-28.6452	0.7000	0.6147	0.0227	0.1329
4	-0.4000	-29.3725	-55.0024	-25.0392	0.6333	0.6777	0.0352	0.1793
5	-0.2000	-26.6577	-49.1962	-21.9910	0.5667	0.7064	0.0483	0.2128
6	0	-24.3750	-44.2500	-19.3750	0.5000	0.7168	0.0621	0.2328
7	0.2000	-22.4341	-39.9823	-17.1008	0.4333	0.7154	0.0764	0.2382
8	0.4000	-20.7681	-36.2589	-15.1014	0.3667	0.7057	0.0913	0.2262
9	0.6000	-19.3265	-32.9781	-13.3265	0.3000	0.6891	0.1067	0.1913
10	0.8000	-18.0705	-30.0618	-11.7372	0.2333	0.6665	0.1227	0.1227
11	1.0000	0	-5.0000	0	0	0	0	0

Решение с точностью $O(h^2)$ при $n = 20$

Результат

Повторим описанные выше шаги для нового значения n

```
n2 = 20;
h2 = (b-a)/n2;
x2=a:h2:b;
[A2,B2,C2,G2] = coefficients(a,b,h2);
```

Получим

```
Y_n2 = TMA_algorithm(A2,B2,C2,G2)
```

```
Y_n2 = 1×21
      0      0.0384      0.0733      0.1048      0.1330      0.1579      0.1795      0.1979 ...
```

Уточнение по Ричардсону

Используя рассчитанные ранее значения Y_{n1} и Y_{n2} , найдем главный член погрешности по формуле:

$$R_m^{(h)} = \frac{y_m^{(\frac{h}{2})} - y_m^{(h)}}{2^s - 1}$$

```
R_n2 = (Y_n2(1:2:21) - Y_n1)/3;
```

Уточним значение в узлах сетки по формуле:

$$\hat{y}_m = y_m^{(\frac{h}{2})} + R_m^{(h)}$$

```
Y_ut = Y_n2(1:2:21) + R_n2;
```

Сравнение с точным значением

Таблица

Найдем разницу между точным и уточненным значениями

```
difference = Y_ut - Y_ex(1:2:21);
```

Результат оформим в виде таблицы

```
table2 = table(transpose(x1),transpose(Y_ex(1:2:21)),transpose(Y_ut),transpose(difference));  
table2.Properties.VariableNames = {'x_i', 'Y_ex', 'Y_ut', 'Y_ut - Y_ex'}
```

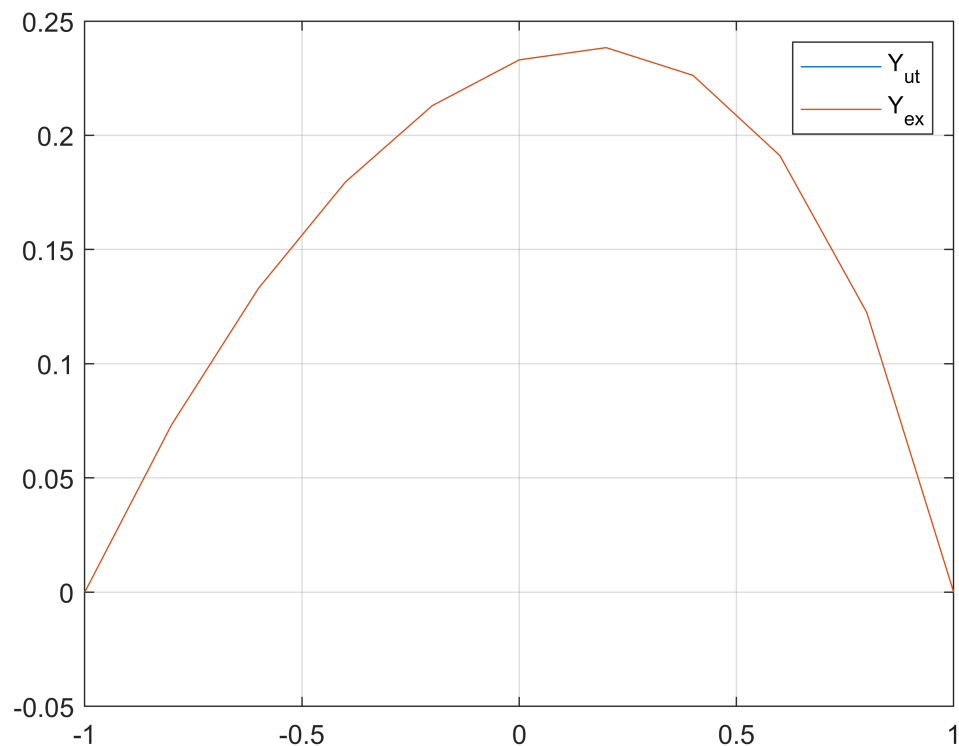
table2 = 11x4 table

	x_i	Y_ex	Y_ut	Y_ut - Y_ex
1	-1.0000	0	0	0
2	-0.8000	0.0733	0.0733	5.2204e-07
3	-0.6000	0.1330	0.1331	1.0470e-06
4	-0.4000	0.1796	0.1796	1.6191e-06
5	-0.2000	0.2130	0.2130	2.2387e-06
6	0	0.2330	0.2330	2.9204e-06
7	0.2000	0.2384	0.2384	3.6426e-06
8	0.4000	0.2263	0.2263	4.2802e-06
9	0.6000	0.1911	0.1911	4.4447e-06
10	0.8000	0.1224	0.1224	3.3378e-06
11	1.0000	-0.0000	0	5.3784e-20

Графики

Построим графики "точного" и уточненного решения.

```
plot(x1,Y_ut)
hold on
plot(x1,Y_ex(1:2:21));
hold off
grid on
legend ('Y_{ut}','Y_{ex}')
```

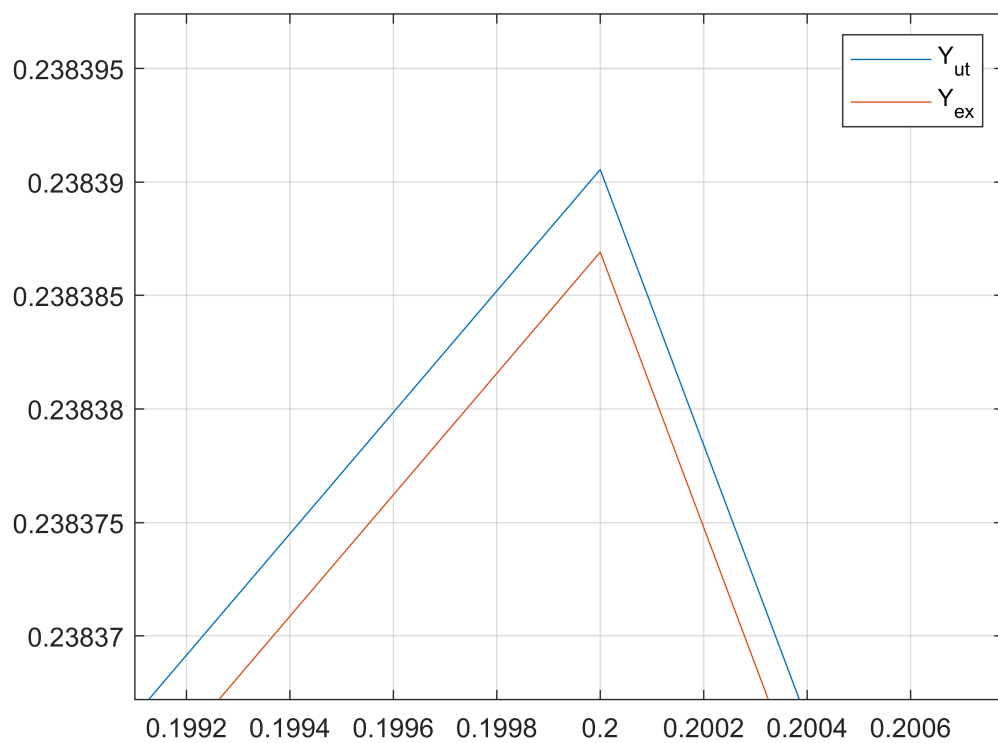


При таком масштабе разница между значениями функций не заметна.

Рассмотрим окрестность точки $(0.2, 0.238385)$.

```
plot(x1,Y_ut)
hold on
plot(x1,Y_ex(1:2:21));
hold off
grid on
legend ('Y_{ut}','Y_{ex}')

xlim([0.19910 0.20078])
ylim([0.2383672 0.2383974])
```



На данном графике видно, что уточненное значение получилось немного больше "точного" значения.