Теоретический материал к семинару №4

Дифференциально-алгебраическая система содержит как дифференциальные, так и алгебраические уравнения. В общем виде она записывается как

$$\begin{cases} \mathbf{G}\dfrac{d\mathbf{u}}{dt} = \mathbf{F}\left(\mathbf{u}, t\right) \\ \mathbf{u}(t_0) = \mathbf{u}_0 \end{cases} \tag{1}$$

где $\mathbf{G}$ - матрица коэффициентов при производных.

Для ее решения можно использовать семейство схем Розенброка, заменив в них матрицу $\mathbf{E}$ матрицей $\mathbf{G}$

$$\begin{cases} \left(\mathbf{G} - \alpha\tau\mathbf{F}_\mathbf{u}\right)\boldsymbol{\omega} = \mathbf{F}\left(\mathbf{u}, t + \dfrac{\tau}{2}\right) \\ \hat{\mathbf{u}} = \mathbf{u} + \tau\,\mathrm{Re}\,\boldsymbol{\omega} \end{cases}. \tag{2}$$

Здесь α - параметр схемы, τ - шаг схемы по времени, $\mathbf{F}_\mathbf{u}$ - производная правой части по переменной $\mathbf{u}$, $\mathbf{u}$ и $\hat{\mathbf{u}}$ - численное решение в текущий и следующий моменты времени соответственно.

Для получения второго порядка в одностадийной схеме Розенброка для задачи Коши необходимо брать $\alpha = \frac{1}{2}$ или $\alpha = \frac{1+i}{2}$. Для неавтономных дифференциально-алгебраических систем даже при этих α получить второй порядок точности не удается из-за противоречивых требований к выбору смещения по времени при вычислении правой части. Поэтому для получения второго порядка необходимо провести автономизацию, то есть убрать явную зависимость от времени в правой части.

Задачи к семинару №4

1. Требуется рассчитать работу транзисторного усилителя. Для этого необходимо решить дифференциально-алгебраическую систему.

   Параметры электрической схемы:

```
r0, r1, r2, r3, r4, r5 = 1000, 9000, 9000, 9000, 9000, 9000
c1, c2, c3 = 1e-6, 2e-6, 3e-6
ub = 6
```

   Начальные условия:

   Python:

```
u0 = np.array([0, ub*r1/(r1+r2), ub*r1/(r1+r2), ub, 0])
```

   MATLAB:

```
u0 = [0; ub*r1/(r1+r2); ub*r1/(r1+r2); ub; 0];
```

   Матрица коэффициентов при производных:

   Python:

```
G = np.array([[-c1,  c1,  0,   0,   0],
              [ c1, -c1,  0,   0,   0],
              [ 0,    0, -c2,  0,   0],
              [ 0,    0,  0,  -c3,  c3],
              [ 0,    0,  0,   c3, -c3]])
```

   MATLAB:

```
G = [-c1  c1  0   0   0;
      c1 -c1  0   0   0;
      0   0  -c2  0   0;
      0   0   0  -c3  c3;
      0   0   0   c3 -c3];
```

Правая часть:

Python:

```python
def ue(t):
    return 0.1 * np.sin(200 * np.pi * t)

def ff(u):
    return 1e-6 * (np.exp(u / 0.026) - 1)

def F(t, u):
    global r0, r1, r2, r3, r4, r5, ub
    y = np.array([u[0]/r0 - ue(t)/r0,
                  0.01 * ff(u[1]-u[2]) - ub/r2 + u[1]*(1/r1 + 1/r2),
                  u[2]/r3 - ff(u[1]-u[2]),
                  0.99 * ff(u[1]-u[2]) - ub/r4 + u[3]/r4,
                  u[4]/r5])
    return y
```

MATLAB:

```matlab
function y = ue(t)
    y = 0.1*sin(200*pi*t);
end

function y = ff(u)
    y = 1e-6*(exp(u/0.026)-1);
end

function y = F(t, u)
    global r0 r1 r2 r3 r4 r5 ub;
    y = [u(1)/r0-ue(t)/r0;
         0.01*ff(u(2)-u(3))-ub/r2+u(2)*(1/r1+1/r2);
         u(3)/r3-ff(u(2)-u(3));
         0.99*ff(u(2)-u(3))-ub/r4+u(4)/r4;
         u(5)/r5                                  ];
end
```

Расчет провести с шагом $h = 1/5000$ на временном отрезке от 0 до 0.3 при $\alpha = \frac{1}{2}$ и $\alpha = \frac{1+i}{2}$. Вывести результат расчета на график (на одном графике будет 2 семейства кривых для двух разных α, в каждом семействе по 5 кривых, соответствующих 5 компонентам $\mathbf{u}$). Объяснить разницу между численными решениями при разных α.

2. Определить эффективный порядок метода с помощью сгущения сеток. Для экономии времени расчет вести до $t = 0.01$. Провести автономизацию и вновь определить эффективный порядок.