

2. semestrální práce z předmětu Numerika 2 (3NU)

Pokyny

Semestrální práci zpracujte sami nebo ve dvojicích a odevzdejte ji vytištěnou do 17.5.2013.

Práce bude obsahovat:

1. Vaše jméno(a), zadaný okrajový problém.
2. Krátký popis vámi implementovaných metod včetně vámi samostatně zpracovaného úplného, řádně komentovaného kódu.
3. Grafické výstupy s popisky.

Zadání

Zadáním semestrální práce je v Matlabu naprogramovat diferenční metodu pro difúzně-konvekční rovnici na čtvercové síti.

Nechť je dán počáteční problém

$$\begin{aligned}\nabla \cdot (v u(x, y) - \alpha \nabla u(x, y)) &= f(x, y), & (x, y) \in (0, 1) \times (0, 1), \\ u(x, y) &= 0 & \text{na } \Gamma_0, \\ u(x, y) &= 1 & \text{na } \Gamma_1, \\ \frac{\partial u(x, y)}{\partial n} &= 0 & \text{na } \Gamma_2,\end{aligned}\tag{1}$$

kde $f(x, y) = 0$. Problém vyřešte pro čtyři různé hodnoty parametru α , $\alpha = 1, \frac{1}{10}, \frac{1}{100}, \frac{1}{1000}$.

Váš výpočet by měl probíhat následovně:

- Vytvořte síť s 1024 uzlovými body.
- Vytvořte matici K o velikosti (počet uzlových bodů)² a vektor pravých stran f o velikosti (počet uzlových bodů) $\times 1$.
- Projděte všechny řádky matice K a vektoru f a naplňte je. Zohledněte okrajovou podmínku na hranici Γ_2 . Okrajové podmínky na hranicích Γ_0 a Γ_1 se zpracují později. Konvekční člen aproximujte pomocí up-wind metody.
- Okrajové podmínky na Γ_0 a Γ_1 zapracujte pomocí následujícího kódu.

Předpokládejme, že sloupcový vektor `Gamma_0` obsahuje uzly na hranici Γ_0 a vektor `Gamma_0_hodnoty` předepsané hodnoty v těchto uzlech. Podobně vektor `Gamma_1` obsahuje uzly na hranici Γ_1 , vektor `Gamma_1_hodnoty` jejich hodnoty.

```
f=f-K(:,Gamma_0)*Gamma_0_hodnoty;
f=f-K(:,Gamma_1)*Gamma_1_hodnoty;
free=setdiff(1:pocet_rovnic,[Gamma_0;Gamma_1]);
f=f(free);K=K(free,free); % Redukce systemu
```

- Systém vyřešte příkazem

```
u=K\f;
```

- Řešení vykreslete pomocí

```

v=zeros(pocet_rovnic,1);
v(Gamma_0) = Gamma_0_hodnoty;
v(Gamma_1) = Gamma_1_hodnoty;
v(free)=u;

Vysledek = zeros(pocet_rovnic,pocet_rovnic);
for i = 1:nx+1
    for j = 1:ny+1
        Vysledek(i,j) = v((i-1)*(pocet_rovnic)+j );
    end
end
surface(Vysledek);

```

- Nakonec změňte up-wind aproximaci za aproximaci pomocí centrální difference a porovnejte řešení.