10-11. Eliptyczne równania różniczkowe cząstkowe

Metody iteracyjne dla rzadkich układów liniowych

⋆ *przerabiane na zajęciach*: 17 grudnia 2018

† *do oddania*: 14 stycznia 2019

(L) Zadanie 1. Zaimplementuj funkcję, która rozwiązuje równanie Poissona $u_{xx} + u_{yy} = f$ w prostokącie $\Omega = [0, a] \times [0, b]$ z warunkiem brzegowym

(**1**) typu Dirichleta: $u_{|\partial\Omega} = g$, (**2**) typu Neumanna: $\frac{\partial u}{\partial n}_{|\partial\Omega} = g$.

W przypadku warunku typu Neumanna należy dodatkowo określić jeden punkt (o współrzędnych (x_0, y_0), niekoniecznie na brzegu), w którym zdefiniowana jest wartość funkcji $u_0 = u(x_0, y_0)$.

Równanie należy rozwiązać metodą SOR (5-punktowy laplasjan), dobierając krok siatki $h = \Delta x = \Delta y$ oraz parametr relaksacyjny ω. Punkt startowy algorytmu iteracyjnego zadany jest funkcją $u^{[0]}$ wewnątrz zbioru Ω. Obliczenia należy zakończyć, gdy zmiany rozwiązania w kolejnych iteracjach (mierzone za pomocą normy L^2 na siatce) zaczynają być mniejsze niż zadany parametr ε. Nagłówek funkcji powinien mieć postać

```
function [u, X, Y, M, N] = PoissonEquationSOR(f, g, Omega, options)
```

gdzie `f` i `g` to funkcje anonimowe lub macierze opisujące f i g, `Omega` to wektor współrzędnych górnego prawego wierzchołka prostokąta Ω, a `options` to struktura z polami:

(**1**) `boundary` – o wartości `'Dirichlet'` lub `'Neumann'`,

(**2**) `additionalPoint` – o wartości `[x0, y0, u0]`, jeśli pole `boundary` równe `'Neumann'`,

(**3**) `fType`, `gType` – o wartości `'function'` lub `'matrix'`,

(**4**) `h`, `omega`, `error` – o wartościach liczbowych,

(**5**) `iter0` – punkt startowy algorytmu (macierz), jeśli nie istnieje to macierz zer,

(L) Zadanie 2. Korzystając z metody SOR rozwiąż równanie Poissona

$$u_{xx} + u_{yy} = -2\cos x \sin y$$

w kwadracie jednostkowym. Warunki brzegowe oraz dokładne rozwiązanie dane są wzorem $u = \cos x \sin y$. Użyj standardowej siatki 5-punktowej z krokiem $h = \Delta x = \Delta y = 0{,}1$, $0{,}05$ oraz $0{,}025$. Jako punktu startowego algorytmu iteracyjnego użyj rozwiązania zerowego wewnątrz kwadratu. Jak zmienia się dokładność metody przy różnych wartościach kroku siatki?

Użyj $\omega = \frac{2}{1+\pi h}$ i zatrzymaj algorytm iteracyjny, gdy zmiany rozwiązania w kolejnych iteracjach (mierzone za pomocą normy L^∞ na siatce) są mniejsze niż 10^{-7}.