KSN — III FK — teoria 6

Iteracyjne rozwiązywanie układów równań liniowych

Praktyczne zastosowanie bezpośrednich metod rozwiązywania algebraicznych równań liniowych

$$\mathbf{A}\cdot\mathbf{x}=\mathbf{b} \tag{1}$$

jest ograniczone chociażby ze względu na skończoną precyzję obliczeń zmiennoprzecinkowych. Błędy zaokrągleń czynią je właściwie bezużytecznymi przy więcej niż 30-40 równaniach.

Metody iteracyjne rozwiązywania układu (1) sprowadzają się do próby rozłożenia macierzy $\mathbf{A}$ na składowe $\mathbf{B}+\mathbf{R}$ takie, by dla jednej z nich znalezienie macierzy odwrotnej (np. $\mathbf{B}^{-1}$) nie było zbyt kłopotliwe. Wówczas równanie (1) można przepisać jako:

$$\mathbf{Bx}=-(\mathbf{A}-\mathbf{B})\mathbf{x}+\mathbf{b}$$

i potraktować je jako iteracyjny przepis na znajdowanie kolejnych przybliżeń $\mathbf{x}$:

$$\mathbf{x}^{n+1}=-\mathbf{B}^{-1}(\mathbf{A}-\mathbf{B})\mathbf{x}^n+\mathbf{B}^{-1}\mathbf{b}.$$

Na przykład w metodzie *Jacobiego* $\mathbf{A}=\mathbf{D}+\mathbf{L}+\mathbf{U}$, gdzie składniki są odpowiednio macierzami diagonalną ($\mathbf{D}$), dolno- ($\mathbf{L}$) i górnotrójkątną ($\mathbf{U}$):

$$\mathbf{x}^{n+1}=-\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})\mathbf{x}^n+\mathbf{D}^{-1}\mathbf{b}=(\mathbf{1}-\mathbf{D}^{-1}\mathbf{A})\mathbf{x}^n+\mathbf{D}^{-1}\mathbf{b}. \tag{2}$$

Przy tym samym rozkładzie macierzy $\mathbf{A}$ metoda *Gaussa–Seidela* daje wzór iteracyjny:

$$\mathbf{Dx}^{n+1}=-\mathbf{Lx}^{n+1}-\mathbf{Ux}^n+\mathbf{b},$$

zaś metoda *kolejnych nadrelaksacji*

$$\mathbf{Dx}^{n+1}=(1-\omega)\mathbf{Dx}^n-\omega(\mathbf{b}-\mathbf{Lx}^{n+1}-\mathbf{Ux}^n), \tag{3}$$

gdzie $0<\omega<2$ jest parametrem nadrelaksacji.

Nietrudno też sobie wyobrazić schemat iteracyjnego poprawiania rozwiązania układu (1) jeśli znamy rozwiązanie różniące się od $\mathbf{x}$ o $\Delta\mathbf{x}$. Dla takiego rozwiązania iloczyn $\mathbf{A}\cdot(\mathbf{x}+\Delta\mathbf{x})$ różni się od prawej strony równania (1) o $\Delta\mathbf{b}$, które łatwo można obliczyć jako:

$$\Delta\mathbf{b}=\mathbf{A}\cdot\Delta\mathbf{x}=\mathbf{A}\cdot(\mathbf{x}+\Delta\mathbf{x})-\mathbf{b}. \tag{4}$$

Wyzanczając z równania (4) $\Delta\mathbf{x}=\mathbf{A}^{-1}\cdot\Delta\mathbf{b}$ i odejmując od „niedokładnego” rozwiązania $(\mathbf{x}+\Delta\mathbf{x})$ otrzymujemy poprawioną wartość $\mathbf{x}$. Procedurę tą można powtórzyć kilkakrotnie — na ogół jednak trzecia iteracja jest już wyrazem nadgorliwości.

W bibliotece *Numerical Recipies* schemat iteracyjnego poprawiania rozwiązania (4) realizuje procedura

```
SUBROUTINE mprove(a,alud,n,np,indx,b,x)
INTEGER n,np,indx(n),NMAX
REAL a(np,np),alud(np,np),b(n),x(n)
PARAMETER (NMAX=500)
...
```

Metoda nadrelaksacji (3) dla uogólniongo równania Poissona

$$a_{i,j}u_{i+1,j}+b_{i,j}u_{i-1,j}+c_{i,j}u_{i,j+1}+d_{i,j}u_{i,j-1}+e_{i,j}u_{i,j}=f_{i,j}.$$

jest zaimplementowana w procedurze:

```
SUBROUTINE sor(a,b,c,d,e,f,u,jmax,rjac)
INTEGER jmax
DOUBLE PRECISION rjac,a(jmax,jmax),b(jmax,jmax),c(jmax,jmax),
&  d(jmax,jmax),e(jmax,jmax),f(jmax,jmax),u(jmax,jmax)
...
```

Parametr `jmac` określa liczbę N na którą podzielono bok siatki L. Rozmiar oczka sieci jest więc równy $h \approx L/N$. Wartość parametru `rjac` służy do wyznaczenia optymalnego parametru nadrelaksacji ω i może być przyjęty za $r_{\mathrm{jac}} = \cos(\pi/N)$.

Biblioteka *GNU Scientific Library (GSL)* umożliwia iteracyjne poprawienie znalezionego wcześniej metodą *rozkładu LU* rozwiązania układu równań (1). Zadanie to wykonuje funkcja

```
int gsl_linalg_LU_refine (const gsl_matrix * A, const gsl_matrix * LU,
      const gsl_permutation * p, const gsl_vector * b, gsl_vector * x,
      gsl_vector * residual)
```

Oprócz macierzy `A` i `b` oraz wyznaczonego bezpośrednio rozwiązania `x`, przekazujemy do funkcji także `LU` i `p` będące wynikiem rozkładu macierzy `A` wykonanego wcześniej z użyciem `gsl_linalg_LU_decomp`.
W wektorze `residual` umieszczana jest początkowa wartość $\mathbf{r} = \mathbf{A} \cdot \mathbf{x} - \mathbf{b}$.
Opisana powyżej funkcja ma także swój odpowiednik zespolony – `gsl_linalg_complex_LU_refine`.

Krzysztof Malarz & Maciej Wołoszyn, Kraków, 30 listopada 2004