

# Metody Obliczeniowe w Nauce i Technice

## 12. Iteracyjne rozwiązywanie $Ax=B$

Marian Bubak, Katarzyna Rycerz

Department of Computer Science  
AGH University of Science and Technology  
Krakow, Poland  
[kzajac@agh.edu.pl](mailto:kzajac@agh.edu.pl)  
[dice.cyfronet.pl](http://dice.cyfronet.pl)

### Contributors

Anna Marciniec  
Radosław Kazior  
Łukasz Janeczko



# Plan wykładu

- 1 12.1 Wady metod bezpośrednich
- 2 12.2 Podział metod iteracyjnych
- 3 12.3 Istota metod iteracyjnych dla  $Ax=b$
- 4 12.4 Zbieżność procesu iteracyjnego rozwiązywania  $Ax=b$
- 5 12.6 Metoda Jacobiego
- 6 12.7 Metoda Gaussa-Seidla (G-S i S-R -successive relaxation)
- 7 12.8 Metoda kolejnych nadrelaksacji - SOR (successive over-relaxation)
- 8 12.9 Sposoby przeglądania węzłów siatki
- 9 12.10 Metoda Czebyszewa
- 10 12.11 Porównywanie jakości wybranych metod iteracyjnych

# Wady metod bezpośrednich

## Złożoność obliczeniowa $\sim N^3$

- $M = \frac{1}{3}n^3 + n^2 - \frac{1}{3}n$  ( $\cdot, /$ )
- $D = \frac{1}{3}n^3 + \frac{1}{2}n^2 - \frac{5}{6}n$  ( $+$ )

Np.  $10^4$  punktów siatki przestrzennej (mesh points)

- na przykład  $100 \times 100$  (2 – D)
- metoda eliminacji Gaussa to  $\sim (10^4)^3 = 10^{12}$  operacji
- jeśli 1 operacja trwa  $\sim 10^{-8}$  s.
- to całość trwa  $10^4$  czyli  $\sim 3$  godziny

## Zwykle

- czas symulacji  $\sim 1\text{h}$ ,
- liczba kroków czasowych  $\sim 1000$ ,
- cały krok - to rozwiązywanie układu równań,
- 1 operacja  $\sim 10^{-8}\text{ s}$ ,
- liczba punktów siatki  $\sim 10^4$  ( $\equiv$  liczba równań)

## Potrzebne są metody o znacznie mniejszej złożoności

Metody takie powinny być oparte na własnościach równań różniczkowych cząstkowych (zwykle - źródła równań liniowych)

## Modelowe zadanie

Równanie Poissona - opisuje niektóre procesy zachodzące w przyrodzie, np.

- potencjał pola grawitacyjnego
- potencjał pola elektrostatycznego
- rozkład temperatury wewnątrz ciała itp.

$$\nabla^2 \varphi(x, y) = -\rho(x, y)$$

gdzie:  $\varphi(x, y)$  - np. rozkład temperatury w 2D

$\rho(x, y)$  - funkcja rozkładu źródeł, zakładamy  $\infty$  szybkość propagacji informacji

$\nabla^2$  - operator Laplace'a, inaczej można zapisać to równanie:

$$\frac{\partial^2 \varphi(x, y)}{\partial x^2} + \frac{\partial^2 \varphi(x, y)}{\partial y^2} = -\rho(x, y)$$

## Warunki brzegowe

Dla przykładowego równania:

$$\frac{\partial^2 \varphi(x, y)}{\partial x^2} + \frac{\partial^2 \varphi(x, y)}{\partial y^2} = -\rho(x, y)$$

Zakładamy przykładowe warunki brzegowe (np. na brzegach temperatura zawsze 0) :

$$\varphi(0, y) = 0, \varphi(n+1, y) = 0, \varphi(x, 0) = 0, \varphi(x, n+1) = 0$$

## Częste źródło układów równań $Ax=B$

Metoda różnic skończonych

Cel: Przybliżenie pochodnych różnicami skończonymi

Wprowadzamy siatkę:

- $\varphi_{i,j} = \varphi(x_i, y_j)$ ,  $i = 1, \dots, N$ ,  $j = 1, \dots, M$
- $(x_i, y_i)$  - punkty siatki
- dla  $i = 0$ ,  $i = N + 1$ ,  $j = 0$ ,  $j = M + 1$  stosujemy warunki brzegowe
- zakładamy, że  $h = x_{i+1} - x_i = y_{j+1} - y_j$  - odstęp między punktami

$$\frac{\partial \varphi_{i,j}}{\partial x} = \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2h} + O(h^2)$$

$$\frac{\partial^2 \varphi_{i,j}}{\partial x^2} = \frac{\varphi_{i+1,j} - 2\varphi_{i,j} + \varphi_{i-1,j}}{h^2} + O(h^2)$$

Podobnie pochodne po  $y$

# Przejsie z równania różniczkowego na układ równań

Przybliżając pochodne różnicami skończonymi otrzymujemy:

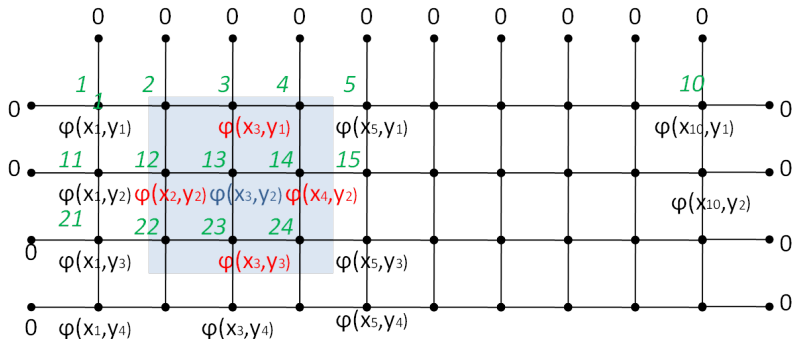
$$\frac{\varphi(x_{i+1}, y_j) - 2\varphi(x_i, y_j) + \varphi(x_{i-1}, y_j)}{h^2} + \frac{\varphi(x_i, y_{j+1}) - 2\varphi(x_i, y_j) + \varphi(x_i, y_{j-1}))}{h^2} = -\rho(x_i, y_j)$$

Czyli:

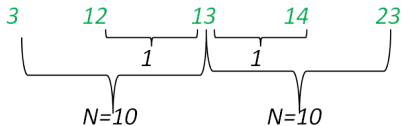
$$\frac{\varphi(x_i, y_{j-1}) + \varphi(x_{i-1}, y_j) - 4\varphi(x_i, y_j) + \varphi(x_{i+1}, y_j) + \varphi(x_i, y_{j+1}))}{h^2} = -\rho(x_i, y_j)$$



# Metoda różnic skończonych



$$\varphi(x_3, y_1) + \varphi(x_2, y_2) - 4\varphi(x_3, y_2) + \varphi(x_4, y_2) + \varphi(x_3, y_3) = -h^2 \rho(x_3, y_2)$$



# Metoda różnic skończonych - zapis macierzowy

Wtedy równanie do rozwiązania wygląda tak:

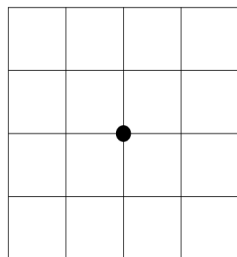
## Zapis macierzowy

$$\begin{bmatrix} -4 & 1 & & & 1 \\ 1 & -4 & 1 & & \\ & 1 & -4 & 1 & \\ 1 & & \dots & & \\ & 1 & & 1 & -4 & 1 \\ & & 1 & & 1 & -4 \end{bmatrix} \begin{bmatrix} \varphi(x_1, y_1) \\ \varphi(x_2, y_1) \\ \varphi(x_3, y_1) \\ \dots \\ \varphi(x_n, y_{n-1}) \\ \varphi(x_n, y_n) \end{bmatrix} = \begin{bmatrix} -h^2 \rho(x_1, y_1) \\ -h^2 \rho(x_2, y_1) \\ -h^2 \rho(x_3, y_1) \\ \dots \\ -h^2 \rho(x_n, y_{n-1}) \\ -h^2 \rho(x_n, y_n) \end{bmatrix}$$

## Metody bezpośrednie zaburzają strukturę macierzy rzadkich

Przykład: niewiadomą jest wielkość w każdym punkcie siatki (lewy rysunek), która zależy od swoich najbliższych sąsiadów.

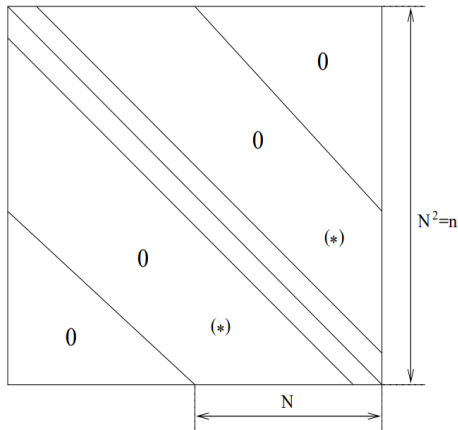
Prawy rysunek pokazuje odpowiadające równanie liniowe.



N

N

→ A:



## Metody bezpośrednie zaburzają strukturę macierzy rzadkich

- macierz wstęgowa ma  $\sim 5 \cdot N^2$  współczynników  $\neq 0$
- ale: po zastosowaniu metody eliminacji Gaussa znikają zera z wstęg (tam gdzie (\*) na rysunku)  
→ trzeba wtedy pamiętać  $2 \cdot N^3$  współczynników

# Podział metod iteracyjnych

## Metody stacjonarne (*stationary*)

- stałe współczynniki macierzy iteracyjnej,
- starsze,
- proste w zrozumieniu i implementacji,
- na przykład metody Jacobiego, G–S (SR), SOR, SSOR.

## Metody niestacjonarne (*nonstationary*)

- współczynniki macierzy iteracyjnej zmieniają się w kolejnych krokach iteracji,
- oparte na idei
  - ciągu wektorów ortogonalnych (CG, MINRES ...),
  - wielomianów ortogonalnych (metoda Czybyszewa),
- stosunkowo nowe,
- trudniejsze w zrozumieniu i implementacji,
- szybciej zbieżne.

# Istotne pojęcia

*iterate* – przybliżenie rozwiązania w kolejnej iteracji,

*residual* –  $r = Ay - b$ ,

*preconditioner*, *preconditioning matrix*: macierz transformująca układ równań do postaci o lepszych własnościach spektralnych

## Istota metod iteracyjnych dla $Ax = b$

Mamy do rozwiązania:

$$A \cdot x = b$$

- gdzie  $A$  jest macierzą  $n \times n$ ,
- $x$  – wektor  $n$  niewiadomych,
- $b$  – wektor danych (źródło).



## Rozkład:

$$A = B + R$$

- $B$  – macierz dla, której łatwo stworzyć  $B^{-1}$
- $R$  – pozostałość

$$A \cdot x = (B + R) \cdot x = b$$

$$B \cdot x = -R \cdot x + b$$

$$B \cdot x = -(A - B) \cdot x + b$$

## Metody iteracyjne dla $Ax = b$ (mesh relaxation methods) polegają na:

- odgadnięciu wektora początkowego  $x^{(0)}$
- generowaniu ciągu iteracyjnego  $x^{(t)}$  według postulowanego wzoru:

$$B \cdot x^{(t+1)} = -(A - B) \cdot x^{(t)} + b$$

$$x^{(t+1)} = -B^{-1} \cdot (A - B) \cdot x^{(t)} + B^{-1} \cdot b$$

$$x^{(t+1)} = \underbrace{I - B^{-1} \cdot A}_{M - \text{iteration matrix}} \cdot x^{(t)} + \underbrace{B^{-1} \cdot b}_W$$

$$x^{(t+1)} = M \cdot x^{(t)} + B^{-1} \cdot b$$

**Różne  $B \rightarrow$  rodzina metod iteracyjnych:**

$$\boxed{x^{(t+1)} = M \cdot x^{(t)} + B^{-1} \cdot b} \quad (\star)$$

Warunek *zgodności* formuły iteracyjnej z szukanym rozwiązaniem

$$\lim_{t \rightarrow \infty} x^{(t+1)} = \lim_{t \rightarrow \infty} (M \cdot x^{(t)} + B^{-1} \cdot b)$$

# Zbieżność procesu iteracyjnego rozwiązywania $Ax=b$

## Twierdzenie: Zbieżność procesu iteracyjnego

### Teza:

Ciąg  $(\star)$  z dowolnym wektorem startowym  $x^{(0)}$  jest zbieżny do jedyne go granicznego  $x^{(\text{inf})}$  wtedy i tylko wtedy, gdy *promień spektralny (spectral radius)* macierzy iteracji jest mniejszy od 1

$$\rho(M) < 1$$

Promień spektralny macierzy - wartość własna o maksymalnej wartości bezwzględnej.

## Dowód

wektor błędu w iteracji  $t$

$$\varepsilon^{(t)} = x^{(t)} - x$$

$$x = M \cdot x + W \text{ oraz } x^{(t+1)} = M \cdot x^{(t)} + W$$

Odejmując mamy:

$$x^{(t+1)} - x = \varepsilon^{(t+1)} = M \cdot \varepsilon^{(t)}$$

Stosując wielokrotnie:

$$\varepsilon^{(t)} = M^t \cdot \varepsilon^{(0)}$$

## Dowód

$M$  – macierz iteracji ma  $n$  różnych wartości  $\rho_i$  i wektorów własnych  $s_i$

$$Ms_i = \rho_i s_i$$

Rozkładamy (przestawiamy) wektor błędu jako kombinację liniową wektorów własnych  $s_i$ ):

$$\varepsilon^{(0)} = \sum_{i=1}^n \alpha_i s_i, \alpha_i - \text{amplituda kierunku } s_i$$

$$\varepsilon^{(t)} = M^t \sum_{i=1}^n \alpha_i s_i = \sum_{i=1}^n \alpha_i M^{t-1} (\underbrace{Ms_i}_{\rho_i s_i}) = \dots = \sum_{i=1}^n \alpha_i \rho_i^t s_i$$

$$\boxed{\varepsilon^{(t)} = \sum_{i=1}^n \alpha_i \rho_i^t s_i}$$

# Dowód

$$\lim_{t \rightarrow \infty} \varepsilon^{(t)} = \lim_{t \rightarrow \infty} \alpha_m \rho_m^t s_m$$

gdzie:  $\rho_m = \max_i |\rho_i| = \rho(M)$  - promień spektralny macierzy iteracji  $M$

$\rho(M) < 1$  - **warunek zbieżności**

## Efektywność procedury iteracyjnej

*Dla określenia efektywności zbieżności - potrzebna jest miara długości wektora błędu  $\|\varepsilon^{(t)}\|$ .*

Pytanie: przy zadanej dokładności  $p \Rightarrow$  dla ilu iteracji  $t^*$

$$\frac{\|\varepsilon^{(t)}\|}{\|\varepsilon^{(0)}\|} = 10^{-p} ?$$

Można pokazać, że  $\lambda = \rho(M)$  jest asymptotyczną miarą zbieżności procedury iteracyjnej (mówi ile razy - w granicy - zmniejsza się średni błąd po każdej iteracji).

Po  $t^*$  iteracjach mamy:

$$\lambda^{t^*} = 10^{-p} \rightarrow t^* = \frac{-p}{\log_{10}(\lambda)} = \frac{-p}{\log_{10}(\rho(M))} = \frac{p}{R}$$

gdzie  $R$  to asymptotyczny wskaźnik zbieżności procedury iteracyjnej (rate of convergence):  $R = -\log_{10}(\rho(M))$



# Metoda Jacobiego

$$A = D + (L + U) \begin{cases} M = I - D^{-1}A \\ W = D^{-1}b \end{cases}$$

gdzie: L – poddiagonalna; U – nadddiagonalna;

D = B – diagonalna, z diagonalnych elementów macierzy A.

$$Ax = (D + (L + U))x = b \implies Dx = -(L + U)x + b$$

Korzystając z zależności

$$Dx^{(t+1)} = -(L + U)x^{(t)} + b$$

otrzymujemy wzór roboczy:

$$x_i^{(t+1)} = \frac{1}{a_{ii}} \left[ b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(t)} \right]; \quad a_{ii} \neq 0, \forall i \in 1, \dots, n$$

$$x_i^{(t+1)} = \frac{1}{a_{ii}} [b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(t)}] ; a_{ii} \neq 0, \forall i \in 1, \dots, n$$

### Procedura przestawiania wierszy

- 1) spośród kolumn z  $a_{ji} = 0$  wybieramy tę, która ma najwięcej zer,
- 2) w tej kolumnie wybieramy element o  $\max |a_{ji}|$  i przestawiamy wiersze tak, aby znalazł się on na diagonalu,
- 3) powtarzamy 1) i 2).

## Modelowe zadanie

równanie Poissona

2-D

warunki brzegowe  $\Rightarrow \varphi \equiv 0$

siatka przestrzenna  $N \times N$

Dla modelowego zadania – met. Jacobiego:

$$\lambda_J = \rho(M_J) = \cos \frac{\pi}{N}$$

$$R_J = \log_{10}(\cos \frac{\pi}{N}) \approx 0.23(\frac{\pi^2}{N^2})$$

Im większe  $N$  tym metoda wolniej zbieżna !

## Charakterystyka metody Jacobiego

- prosta,
- ma znaczenie dydaktyczne,
- wolnozbieżna,
- nie wykorzystuje całej, dostępnej w danym kroku informacji,
- pamiętane  $x^{(t)}$  i  $x^{(t+1)}$ ,
- zbieżna dla  $A$  silnie diagonalnie dominujących,

$$\text{wierszowo : } |a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|,$$

$$\text{kolumnowo : } |a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ji}|,$$

# Metoda Gaussa-Seidla (G-S i S-R -successive relaxation)

$$A = \underbrace{(L + D)}_B + U$$

$$(D + L)x^{(t+1)} = -Ux^t + b$$

$$Dx^{(t+1)} = -Lx^{(t+1)} - Ux^{(t)} + b$$

Pamiętamy, że:

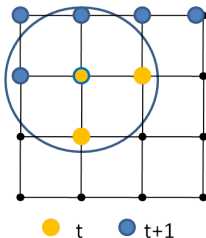
$$x^{(t+1)} = Mx^{(t)} + W, M = I - B^{-1}A, W = B^{-1}b$$

$$M = I - B^{-1}A = I - B^{-1}(B + U) = -B^{-1}U$$

$$x_i^{(t+1)} = \frac{1}{a_{ii}} \left[ b_i - \underbrace{\sum_{j=1}^{i-1} a_{ij} x_j^{(t+1)}}_{(*)} - \underbrace{\sum_{j=i+1}^n a_{ij} x_j^{(t)}}_{(**)} \right]$$

( $\star$ ) - otrzymujemy z rozwiązania w bieżącej ( $t + 1$ ) iteracji, ( $\star\star$ ) - z poprzedniej iteracji ( $t$ ).

$$\varphi^{(t+1)}(x_i, y_j) = \frac{\rho_{i,j} \cdot h^2 + \varphi^{(t+1)}(x_i, y_{j-1}) + \varphi^{(t+1)}(x_{i-1}, y_j) + \varphi^{(t)}(x_{i+1}, y_j) + \varphi^{(t)}(x_i, y_{j+1})}{4}$$



## Charakterystyka metody G-S

- elementy diagonalni powinny być  $\neq 0 \rightarrow$  przestawianie
- wystarczy pamiętać aktualne przybliżenie  $x^{(t+1)}$
- zbieżna dla  $A$ :
  - \* silnie diagonalnie dominujących wierszowo, kolumnowo,
  - \* symetrycznych,
  - \* dodatnio określonych ( $x^T Ax > 0 \wedge x \neq 0$ ).



Dla modelowego zadania – met. GS:

$$\rho(M_{GS}) = \rho^2(M_J)$$

$$\lambda_{GS} = \cos^2\left(\frac{\pi}{N}\right) \approx 1 - \left(\frac{\pi^2}{N^2}\right)$$

Metoda GS wymaga dwa razy mniej iteracji niż Jacobiego:

$$R_{GS} = -\log_{10}\left(\cos^2\left(\frac{\pi}{N}\right)\right) = 2R_J$$

$$R_{GS} \approx -\log_{10}\left(1 - \frac{\pi^2}{N^2}\right) \approx 0.4343 \frac{\pi^2}{N^2}$$

# Metoda kolejnych nadrelaksacji - SOR (successive over-relaxation)

Na podstawie metody successive relaxation (czyli Gaussa-Seidla) można zapisać :

$$x_i^{(t+1)} = x_i^{(t)} + \omega \underbrace{\frac{1}{a_{ii}} \left[ b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(t+1)} - \sum_{j=i}^n a_{ij} x_j^{(t)} \right]}_{r_i^{(t)} - \text{poprawka do starego rozwiązania } x_i^{(t)}}$$

Dla metody GS  $\omega = 1$

Przyspieszenie zbieżności:

$$\boxed{x_i^{(t+1)} = x_i^{(t)} + \omega r_i^{(t)}}, \omega - \text{pewna liczba}$$

Można też zapisać:

$$x_i^{(t+1)} = (1 - \omega)x_i^{(t)} + \omega \frac{1}{a_{ii}} \left[ b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(t+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(t)} \right]$$

W zapisie macierzowym

$$Dx^{(t+1)} = (1 - \omega)Dx^{(t)} + \omega[b - Lx^{(t+1)} - Ux^{(t)}]$$

po uporządkowaniu:

$$x^{(t+1)} = \underbrace{(D + \omega L)^{-1}[D - \omega(D + U)]}_M x^{(t)} + \underbrace{\omega(D + \omega L)^{-1}b}_{W(=B^{-1}b)}$$

## Twierdzenie

### Założenia

Dla dowolnej nieosobliwej macierzy  $A$  i dowolnej liczby  $\omega$  zachodzi:

### Teza

$$\rho(M) \geq |\omega - 1|$$

Stąd:

$$\omega \in (0, 2) \Rightarrow \begin{cases} \omega \in (0, 1] & \text{- podrelaksacja} \\ \omega \in (1, 2) & \text{- nadrelaksacja} \end{cases}$$

Dla ważnych praktycznie klas macierzy znana jest optymalna wartość  $\omega$ , która zależy od pochodzenia macierzy (zwykły różniczkowy)

## Twierdzenie

### Założenia

Dla  $A$  - symetrycznej, dodatnio określonej o postaci blokowo - trójkątnej:

$$A = \begin{bmatrix} D_1 & U_1 & & & \\ L_2 & D_2 & U_2 & & \\ & & \dots & \dots & \\ & & & L_{n-1} & D_{n-1} & U_{n-1} \\ & & & & L_n & D_n \end{bmatrix}$$

## Teza

$$\begin{aligned}\rho(M_{GS}) &= \rho^2(M_J) \\ \omega_{opt} &= \frac{2}{1 + \sqrt{1 - \rho(M_{GS})}} \\ \lambda_{SOR} &= \omega_{opt} - 1\end{aligned}$$

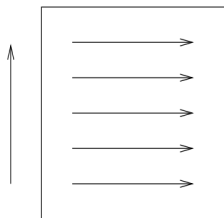
## Dla równania modelowego

$$\rho(M_{GS}) = \cos^2\left(\frac{\pi}{N}\right), \omega_{opt} \approx 2\left(1 - \frac{\pi}{N}\right), \lambda_{SOR} \approx 1 - \frac{2\pi}{N}$$

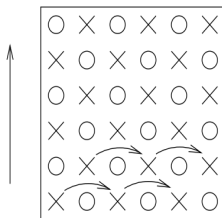
$$R_{SOR} \approx -\log_{10}\left(1 - \frac{2\pi}{N}\right) = 0.4343 \frac{2\pi}{N}$$

SOR wymaga  $\frac{R_{SOR}}{R_{GS}} = \frac{2N}{\pi}$  razy mniej iteracji niż GS

## Sposoby przeglądania węzłów siatki



typewriter ordering



odd/even

najpierw: X

potem: O

<- każdy węzeł komunikuje się  
z najbliższymi sąsiadami.

X - odd  
O - even

# Metoda Czebyszewa

Jak pamiętamy, ogólny wzór na macierz iteracji to:

$$M = I - B^{-1}A$$

Zadaniem jest znaleźć takie  $B$ , dla którego  $\rho(M)$  ma jak najmniejszą wartość bezwzględną (zbieżność)

Wprowadzamy wielomian macierzowy  $W(A)$  - daje macierz uzyskaną z wielomianu  $W(t)$  po podstawieniu za  $t$  macierzy  $A$



Zakładamy, że  $B^{-1} = W(A)$  dla jakiegoś  $W$ . Wtedy

$$M = 1 - W(A)A$$

i szukamy najlepszego takiego wielomianu  $W$ .

Jeśli  $A$  ma wartość własną  $\lambda$  to

$$P(A) = 1 - W(A)A$$

ma wartość własną

$$P(\lambda) = 1 - W(\lambda)\lambda$$

Zadaniem jest znaleźć taki wielomian  $P$ , dla którego maksymalna wartość w przedziale  $[a, b]$ , gdzie znajdują się wszystkie wartości własne macierzy  $A$  jest najmniejsza:

$$\min! \max_{u \in [a, b]} |1 - W(u)u|$$

Jest to klasyczny problem aproksymacji  $\Rightarrow$  wielomiany Czebyszewa

# Metoda Czebyszewa

- stosujemy wzór roboczy SOR

$$x_i^{(t+1)} = x_i^{(t)} + \omega \underbrace{\frac{1}{a_{ii}} \left[ b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(t+1)} - \sum_{j=i}^n a_{ij} x_j^{(t)} \right]}_{r_i^{(t)} - \text{poprawka do starego rozwiązania } x_i^{(t)}}$$

- ale  $\omega$  zmienia się w każdym kroku
- $\omega$  wyliczamy zgodnie z zależnościami rekurencyjnymi (użycie przeskalowanych wielomianów Czebyszewa)

# Metoda Czebyszewa

Przeglądanie siatki odd-even:

$$\rho = \rho(M_J)$$

$$\omega^{(0)} = 1$$

$$\omega^{\frac{1}{2}} = \frac{1}{1 - \frac{1}{2}\rho^2}$$

$$\omega^{(t+\frac{1}{2})} = \frac{1}{1 - \frac{1}{4}\rho^2\omega^{(t)}}, \text{ dla } t=\frac{1}{2}, 1, 1\frac{1}{2}, \dots$$

$$\omega^{(\infty)} = \omega_{opt}$$

```

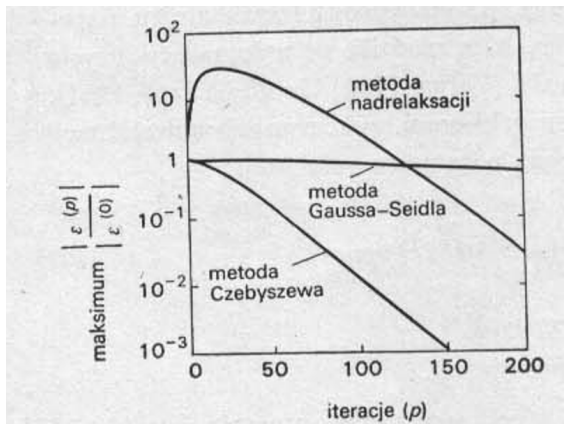
1   $\omega = 1.0$ 
DO 2 t = 1, MAXIT                                ( $\approx 100, 1000$ )
3      Norm = 0.0
DO 1 p = 1, n
5      DO 1 q = 1, n
          IF ( MOD(p+q,2) .EQ. MOD(t,2) ) THEN
7              Residual =  $a_{p,q}\phi_{p,q-1} + b_{p,1}\phi_{p,q+1} + c_{p,q}\phi_{p-1,q} +$ 
                         $d_{p,q}\phi_{p+1,q} + e_{p,q}\phi_{p,q} - f_{p,q}$ 
9              Norm = Norm + (Residual)2
                         $\phi_{p,q} = \phi_{pq} - \omega * \text{Residual} / e_{p,q}$ 
11             ENDIF
1      CONTINUE
13  (*)  $\omega = 1.0 / (1.0 - 0.25\rho^2\omega)$ 
    (*) IF (t .EQ. 1)  $\omega = 1.0 / (1.0 - 0.5\rho^2)$ 
15  IF (Norm .LT. EPS ||  $\vec{f}$  || solution obtained
2  CONTINUE

```

### Uwagi do algorytmu

- $\text{EPS} \approx 10^{-6}$ ,
- $\|f^{\rightarrow}\|$  – obliczona wcześniej norma prawej strony,
- gdy usuniemy instrukcje (\*) otrzymamy metodę G-S,
- jeżeli dodatkowo zastąpimy  $\omega = 1$  przez  $\omega = \omega_{opt}$  - metodę SOR,
- (Dla r. Poissona:  $a = b = c = d = 1, e = -4$ )

# Porównywanie jakości wybranych metod iteracyjnych



źródło: David Potter Metody obliczeniowe fizyki

$$A \leftarrow \begin{cases} \text{r. Poissona, 2-D} \\ \text{siatka } 128 \times 128 \end{cases}$$