

# Równanie Poissona i relaksacja

## Treść

- typowe liniowe równania różniczkowe cząstkowe
- typowe warunki brzegowe
- ilorazy różnicowe pierwszej i drugiej pochodnej
- dyskretyzacja równania
- procedura relaksacji i nadrelaksacji
- całka działania i zbieżność iteracji
- metody dokładne i iteracyjne rozwiązywania układów równań liniowych (metody Gaussa, dekompozycji LU, Jacobiego, Gaussa-Seidla)
- wygładzanie błędów
- iteracja wielosiatkowa

# typowe cząstkowe liniowe równanie różniczkowe

- $\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$  (hiperboliczne, klasyczne falowe)
- $i\hbar \frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2} + V(x, t)u$  (paraboliczne, Schrödingera, kwantowe falowe)
- $\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2} + S(x, t)$  (paraboliczne, przewodnictwa cieplnego, dyfuzji)
- $0 = \epsilon \frac{\partial^2 u}{\partial x^2} + S(x)$  (eliptyczne, Poissona)
- $\frac{\partial u}{\partial t} = c \frac{\partial u}{\partial x}$  (adwekcji, "równanie falowe pierwszego rzędu")
- obszar całkowania  $\Omega = [a, b]$
- $u(a) = C$  (warunek brzegowy typu Dirichleta)
- $u'_x(a) = D$  (warunek brzegowy typu Neumanna)
- $u'_x(a) + cu(a) = E$  (warunek brzegowy typu Robina lub mieszany)

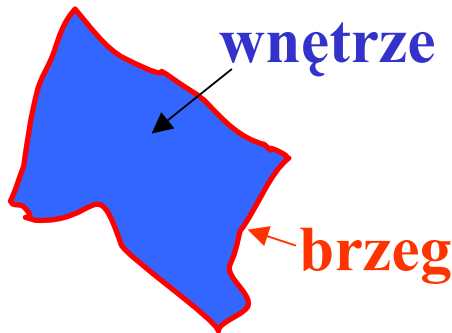
# równanie Poissona jako modelowe eliptyczne funkcjonał działania, zbieżność, relaksacje wielosiatkowe

Eliptyczne: opisuje stany stacjonarne

$$\nabla^2 \phi = -\rho$$

- 1) Rozkład potencjału elektrostatycznego  
[minimum *działania* w układzie ładunek/pole]
- 2) Rozkład temperatury przy stacjonarnym przepływie ciepła  
[ granica czasowa problemu parabolicznego ]

...



na brzegu musimy określić wartość  
rozwiązania lub jego pochodnej normalnej  
lub związek między nimi

# ilorazy różnicowe

różniczkowanie  
numerycz-  
ne

interpolacja  
wielomia-  
nowa

dynamika  
punktu  
material-  
nego

schematy  
Eulera

równania  
nieliniowe

metoda  
bisekcji

metoda  
Newtona

fraktal N.

krok w  
schemacie  
niejawnym

układy  
równań  
nielini-  
owych

schemat  
trapezów

- rachunek różniczkowy i pochodna funkcji  $f(x)$
- $f'(x) \equiv \lim_{\Delta x \rightarrow 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$
- najprostsze metody numeryczne rozwiązywania równań różniczkowych – przed przejściem granicznym
- pracujemy na ilorazach różnicowych oraz równaniach algebraicznych, które powstają z dyskretyzacji przestrzeni / czasu do punktów / chwil czasowych
- dyskretyzacja czasu / przestrzeni, ilorazy różnicowe zamiast pochodnych – metoda różnic skończonych (*finite difference method*)

# twierdzenie Taylora

różniczkowanie  
numerycz-  
ne

- szereg Taylora

- $y(t + \Delta t) = y(t) + \Delta t \frac{dy}{dt}|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2}|_t + \dots + \frac{\Delta t^n}{n!} \frac{d^n y}{dt^n}|_t + \dots$

- twierdzenie Taylora

- $y(t + \Delta t) = y(t) + \sum_{n=1}^N \left[ \frac{\Delta t^n}{n!} \frac{d^n y}{dt^n}|_t \right] + \frac{\Delta t^{N+1}}{(N+1)!} \frac{d^{N+1} y}{dt^{N+1}}|_\tau$ , gdzie  $\tau \in (t, t + \Delta t)$

- reszta znika w granicy małego kroku czasowego

- $\lim_{\Delta t \rightarrow 0} \left( \frac{\Delta t^{N+1}}{N!} \frac{d^{N+1} y}{dt^{N+1}}|_\tau \right) = 0$

- tempo zbieżności do zera jest  $\Delta t^{N+1}$ , co oznaczamy  
 $\left( \frac{\Delta t^{N+1}}{(N+1)!} \frac{d^{N+1} y}{dt^{N+1}}|_\tau \right) = O(\Delta t^{N+1})$

# różniczkowanie numeryczne: dwupunktowy iloraz różnicowy pierwszej pochodnej

różniczkowanie  
numerycz-  
ne

■ szereg Taylora:

$$\blacksquare y(t + \Delta t) = y(t) + \Delta t \frac{dy}{dt} \Big|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t + \cdots + \frac{\Delta t^n}{n!} \frac{d^n y}{dt^n} \Big|_t + \cdots$$

$$\blacksquare y(t + \Delta t) = y(t) + \Delta t \frac{dy}{dt} \Big|_t + O(\Delta t^2)$$

$$\blacksquare \Delta t \frac{dy}{dt} \Big|_t = y(t + \Delta t) - y(t) + O(\Delta t^2)$$

$$\blacksquare \frac{dy}{dt} \Big|_t = \frac{y(t + \Delta t) - y(t)}{\Delta t} + O(\Delta t)$$

# różniczkowanie numeryczne: schemat centralny trójpunktowy

różniczkowanie  
numerycz-  
ne

■ szereg Taylora:

$$\blacksquare y(t + \Delta t) = y(t) + \Delta t \frac{dy}{dt} \Big|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t + \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t - \Delta t) = y(t) - \Delta t \frac{dy}{dt} \Big|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t - \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t + \Delta t) - y(t - \Delta t) = 2\Delta t \frac{dy}{dt} \Big|_t + 2\frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + \dots$$

$$\blacksquare y(t + \Delta t) - y(t - \Delta t) = 2\Delta t \frac{dy}{dt} \Big|_t + O(\Delta t^3)$$

$$\blacksquare \frac{dy}{dt} \Big|_t = \frac{y(t+\Delta t) - y(t-\Delta t)}{2\Delta t} + O(\Delta t^2)$$

■ zamiast

$$\blacksquare \frac{dy}{dt} \Big|_t = \frac{y(t+\Delta t) - y(t)}{\Delta t} + O(\Delta t)$$

# różniczkowanie numeryczne: schemat centralny pięciopunktowy

różniczkowanie  
numerycz-  
ne

## ■ szereg Taylora:

$$\blacksquare y(t + \Delta t) = y(t) + \Delta t \frac{dy}{dt} \Big|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t + \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t - \Delta t) = y(t) - \Delta t \frac{dy}{dt} \Big|_t + \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t - \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t + \Delta t) - y(t - \Delta t) = 2\Delta t \frac{dy}{dt} \Big|_t + 2 \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + O(\Delta t^5) \quad (1)$$

$$\blacksquare y(t + 2\Delta t) = y(t) + 2\Delta t \frac{dy}{dt} \Big|_t + 4 \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t + 8 \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + 16 \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t - 2\Delta t) = y(t) - 2\Delta t \frac{dy}{dt} \Big|_t + 4 \frac{\Delta t^2}{2} \frac{d^2 y}{dt^2} \Big|_t - 8 \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + 16 \frac{\Delta t^4}{4!} \frac{d^4 y}{dt^4} \Big|_t + \dots$$

$$\blacksquare y(t + 2\Delta t) - y(t - 2\Delta t) = 4\Delta t \frac{dy}{dt} \Big|_t + 2 \cdot 8 \frac{\Delta t^3}{3!} \frac{d^3 y}{dt^3} \Big|_t + O(\Delta t^5) \quad (2)$$

$$\blacksquare 8 \cdot (1) - (2) \rightarrow$$

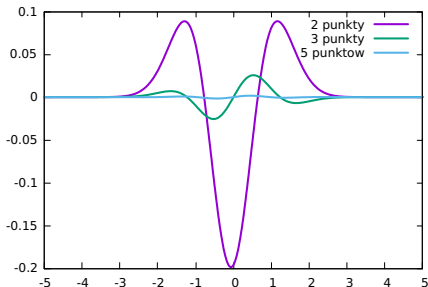
$$\blacksquare \frac{dy}{dt} \Big|_t = \frac{1}{12\Delta t} [y(t - 2\Delta t) - 8y(t - \Delta t) + 8y(t + \Delta t) - y(t + 2\Delta t)] + O(\Delta t^4)$$

■ etc.



# różniczkowanie numeryczne: schemat centralny pięciopunktowy

różniczkowanie  
numerycz-  
ne



- Rysunek pokazuje różnicę między dokładną pochodną funkcji gaussowskiej  $\exp(-x^2)$  oraz jej przybliżeniem ilorazem różnicowym przy  $\Delta x = \frac{1}{5}$ , z ilorazem 2, 3 oraz 5 punktowym

# iloraz różnicowy drugiej pochodnej

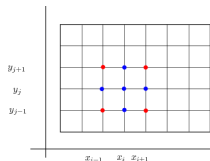
- szereg Taylora
- $f(x + dx) = f(x) + dx \frac{df}{dx} \Big|_x + \frac{dx^2}{2} \frac{d^2f}{dx^2} \Big|_x + \frac{dx^3}{3!} \frac{d^3f}{dx^3} \Big|_x + \frac{dx^4}{4!} \frac{d^4f}{dx^4} \Big|_x + \frac{dx^5}{5!} \frac{d^5f}{dx^5} \Big|_x + \frac{dx^6}{6!} \frac{d^6f}{dx^6} \Big|_\xi$
- $f(x - dx) = f(x) - dx \frac{df}{dx} \Big|_x + \frac{dx^2}{2} \frac{d^2f}{dx^2} \Big|_x - \frac{dx^3}{3!} \frac{d^3f}{dx^3} \Big|_x + \frac{dx^4}{4!} \frac{d^4f}{dx^4} \Big|_x - \frac{dx^5}{5!} \frac{d^5f}{dx^5} \Big|_x + \frac{dx^6}{6!} \frac{d^6f}{dx^6} \Big|_\eta$
- czerwone + niebieskie stronami, z tego wyliczyć wyrażenie z drugą pochodną
- $dx^2 \frac{d^2f}{dx^2} \Big|_x = f(x + dx) + f(x - dx) - 2f(x) + O(dx^4)$
- $\frac{d^2f}{dx^2} \Big|_x = \frac{f(x+dx)+f(x-dx)-2f(x)}{dx^2} + O(dx^2)$

# dyskretna forma równania Poissona

- $\frac{d^2 f}{dx^2} \Big|_x = \frac{f(x+dx) + f(x-dx) - 2f(x)}{dx^2} + O(dx^4)$
- $\nabla^2 \phi = -\rho$
- 1D:  $\frac{\phi(x+dx) + \phi(x-dx) - 2\phi(x)}{dx^2} = -\rho(x)$
- 2D:

$$\frac{\phi(x+dx, y) + \phi(x-dx, y) - 2\phi(x, y)}{dx^2} \quad (1)$$

$$\frac{\phi(x, y+dy) + \phi(x, y-dy) - 2\phi(x, y)}{dy^2} = -\rho(x, y) \quad (2)$$



Układ równań liniowych

# równanie Laplace'a

- 2D:

$$\frac{\phi(x+dx, y) + \phi(x-dx, y) - 2\phi(x, y)}{dx^2} \quad (3)$$

$$\frac{\phi(x, y+dy) + \phi(x, y-dy) - 2\phi(x, y)}{dy^2} = 0 \quad (4)$$

- dla  $dx = dy$ :
- 2D:  $\phi(x+dx, y) + \phi(x-dx, y) - 2\phi(x, y) + \phi(x, y+dx) + \phi(x, y-dx) - 2\phi(x, y) = 0$
- albo
- $\phi(x+dx, y) + \phi(x-dx, y) + \phi(x, y+dx) + \phi(x, y-dx) = 4\phi(x, y)$

Warto wiedzieć: (zasada maximum)  
rozwiązanie równanie Laplace'a osiąga wartości  
ekstremalne na brzegach (dowód np. u Weinbergera)

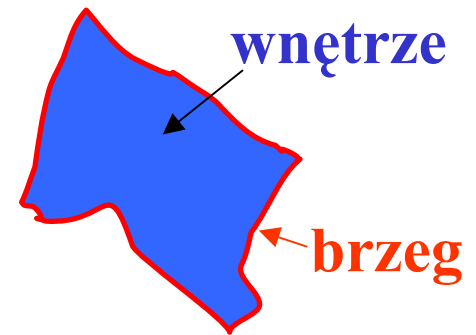
dla metody RS:

$$\nabla^2 \phi = 0$$



$$\Phi(i, j) = (\Phi(i - 1, j) + \Phi(i, j - 1) + \Phi(i + 1, j) + \Phi(i, j + 1))/4$$

skoro każdy punkt z wewnątrz  
obszaru całkowania jest średnią arytmetyczną z sąsiadów  
nigdy nie będzie większy od żadnego z nich



# iteracja relaksacyjna 2D

- $\nabla^2 \phi = -\rho(x)$
- z  $dx = dy$
- $\frac{\phi(x+dx, y) + \phi(x-dx, y) + \phi(x, y-dx) + \phi(x, y+dx) - 4\phi(x, y)}{dx^2} = -\rho(x, y)$
- dokładne rozwiązanie dyskretnej wersji równania Poissona spełnia warunek:
- $\phi(x, y) = \frac{dx^2 \rho(x, y) + \phi(x+dx, y) + \phi(x-dx, y) + \phi(x, y-dx) + \phi(x, y+dx)}{4}$
- pomysł na relaksację: przeiterować, może się zbliżać, jeśli tak mamy rozwiązanie
- $\phi(x, y) := \frac{dx^2 \rho(x, y) + \phi(x+dx, y) + \phi(x-dx, y) + \phi(x, y-dx) + \phi(x, y+dx)}{4}$

# Z elektrostatyki poprzez metodę różnic skończonych do równania Poissona i metod relaksacji i nadrelaksacji .

Działanie dla układu ładunków ( $\rho$ ) + pole :

/ potencjał pola elektrycznego

$$S = \int \left( \frac{1}{2} \mathbf{E}^2 - \rho \phi \right) dv = \int \left( \frac{1}{2} (\nabla \phi)^2 - \rho \phi \right) dv$$

(to **nie** jest energia układu  
energia będzie gdy znak  
przy  $\rho$  będzie +)



wektor pola elektrycznego

rozkład ładunku



funkcja podcałkowa:  
tzw. lagranżjan układu pole - ładunek

$$\mathbf{E} = -\nabla \phi$$

$$S = \min \longrightarrow \nabla^2 \phi = -\rho$$

Działanie jest najmniejsze dla potencjału, który spełnia równanie Poissona

Zobaczmy to w 1D:



## działanie a równanie Poissona w 1D

$$S = \left| \int_{-d/2}^{d/2} \left( \frac{1}{2} \left( \frac{d\phi}{dz} \right)^2 - \rho\phi \right) dz \right|$$

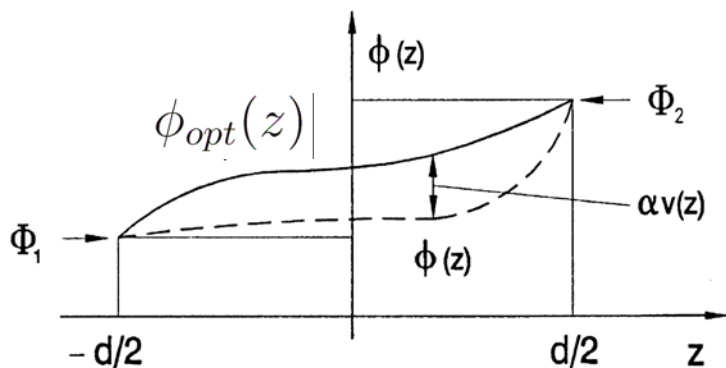
Z warunkami brzegowymi typu Dirichleta

$$\phi(z = -d/2) = \Phi_1, \quad \phi(z = d/2) = \Phi_2$$

Dla jakiego  $\phi$  wartość działania jest ekstremalna ? (w praktyce minimalna,  
bo maksymalna nie istnieje).

Ogólny problem minimum funkcjonału (całki funkcjonalnej)

$$F[\phi(z)] = \int_{-d/2}^{d/2} f[\phi(z), \phi'(z); z] dz$$



$\phi_{opt}(z)$  optymalny potencjał: minimalizuje działanie

$\phi(z)$  „bliski” optymalnemu  
i spełniający te same warunki brzegowe

$\phi(z) = \phi_{opt}(z) + \alpha v(z)$  ← dowolna funkcja  
ciągła z pochodną  
mały parametr

$$v(z = -d/2) = v(z = d/2) = 0$$



$$\phi(z) = \phi_{opt}(z) + \alpha v(z)$$

z definicji:  $F[\phi(z)] \geq F[\phi_{opt}(z)]$  |


$$F[\phi(z)] = \int_{-d/2}^{d/2} f[\phi_{opt}(z) + \alpha v(z), \phi'_{opt}(z) + \alpha v'(z); z] dz$$
 |

Wartość  $\alpha=0$  jest optymalna:


$$\frac{d}{d\alpha} \left( \int_{-d/2}^{d/2} f[\phi_{opt}(z) + \alpha v(z), \phi'_{opt}(z) + \alpha v'(z); z] dz \right)_{\alpha=0} = 0$$

Pochodna pod całkę:

$$\int_{-d/2}^{d/2} \left[ \frac{\partial f}{\partial \phi} \frac{\partial \phi}{\partial \alpha} + \frac{\partial f}{\partial \phi'} \frac{\partial \phi'}{\partial \alpha} \right]_{\alpha=0} dz = 0$$



$v(z)$



$v'(z)$

( wstawiamy  $\alpha=0$  )

$$\int_{-d/2}^{d/2} \left[ \frac{\partial f}{\partial \phi_{opt}} v(z) + \frac{\partial f}{\partial \phi'_{opt}} v'(z) \right] dz = 0$$

$$\int_{-d/2}^{d/2} \left[ \frac{\partial f}{\partial \phi_{opt}} v(z) + \frac{\partial f}{\partial \phi'_{opt}} v'(z) \right] dz = 0$$

pochoďna iloczynu [całkowanie przez części]

$$\frac{\partial f}{\partial \phi'_{opt}} v'(z) = \frac{d}{dz} \left[ \frac{\partial f}{\partial \phi'_{opt}} v(z) \right] - \left[ \frac{d}{dz} \frac{\partial f}{\partial \phi'_{opt}} \right] v(z)$$

$$\int_{-d/2}^{d/2} dz \left[ \frac{\partial f}{\partial \phi_{opt}} - \frac{d}{dz} \frac{\partial f}{\partial \phi'_{opt}} \right] v(z) + \left[ \frac{\partial f}{\partial \phi'_{opt}} v(z) \right]_{-d/2}^{d/2} = 0$$

dowolna

$v(-d/2)=v(d/2)=0$

czyli:

$$\frac{\partial f}{\partial \phi_{opt}} - \frac{d}{dz} \frac{\partial f}{\partial \phi'_{opt}} = 0$$

→ równanie Eulera-Lagrange'a  
na funkcję dla której całka funkcjonalna  
minimalna

## Równanie Eulera-Lagrange'a dla energii układu ładunek+pole

$$S = \int_{-d/2}^{d/2} \left( \frac{1}{2} \left( \frac{d\phi}{dz} \right)^2 - \rho\phi \right) dz \quad \Bigg| \longrightarrow \quad f(\phi, \phi'; z) = \frac{1}{2}(\phi')^2 - \rho\phi$$

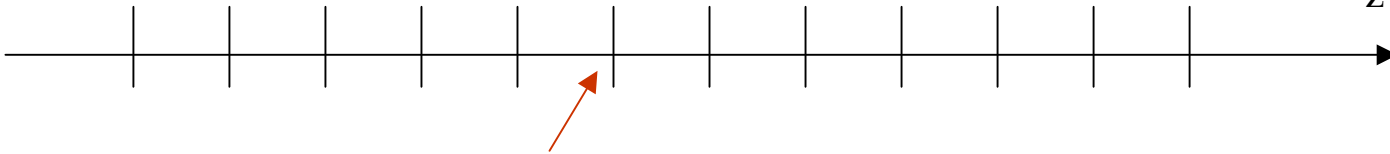
$$\frac{\partial f}{\partial \phi} - \frac{d}{dz} \frac{\partial f}{\partial \phi'} = 0 \quad \Bigg|$$

$$-\rho = \frac{d}{dz} \phi' = \phi'' \quad \Bigg|$$

---

minimalne działanie dostajemy  
dla potencjału spełniającego  
równanie Poissona

## Działanie na siatce różnicowej

$$S = \int_{-d/2}^{d/2} \left( \frac{1}{2} \left( \frac{d\phi}{dz} \right)^2 - \rho\phi \right) dz$$


$$\phi_n = \phi(z_n)$$

$$\rho_n = \rho(z_n)$$

$$S = \sum_n \Delta z \left( \frac{1}{2} (\phi'_n)^2 - \rho_n \phi_n \right)$$

Zdyskretyzowane działanie

$$\phi'_n = \frac{\phi_n - \phi_{n-1}}{\Delta z}$$

najprostszy iloraz różnicowy pierwszej pochodnej

Minimum zdyskretyzowanego działania

$$\frac{\partial S}{\partial \phi_i} = 0$$

dla wszystkich oczek siatki  $i$

$$\frac{\partial S}{\partial \phi_i} = \sum_n \left[ \frac{1}{2\Delta z^2} \frac{\partial}{\partial \phi_i} (\phi_n - \phi_{n-1})^2 - \rho_n \frac{\partial \phi_n}{\partial \phi_i} \right] = 0$$

$$\frac{\partial S}{\partial \phi_i} = \sum_n \left[ \frac{1}{\Delta z^2} \left( (\phi_n - \phi_{n-1}) \delta_{ni} - (\phi_n - \phi_{n-1}) \delta_{(n-1)i} \right) - \rho_n \delta_{ni} \right] = 0$$

wysumowane z deltami Kroneckera:

$$\frac{\partial S}{\partial \phi_i} = \frac{1}{\Delta z^2} ((\phi_i - \phi_{i-1}) - (\phi_{i+1} - \phi_i)) - \rho_i = 0$$

z zasady najmniejszego działania  
na siatce dostaliśmy dokładnie takie samo  
równanie, jak po bezpośredniej dyskretyzacji  
równania Poissona:

$$\frac{1}{\Delta z^2} (-\phi_{i-1} + 2\phi_i - \phi_{i+1}) = \rho_i$$

$$-\rho = \frac{d}{dz} \phi' = \phi''$$

z ilorazem różnicowym drugiej pochodnej

$$\phi''_i = \frac{\phi_{i+1} + \phi_{i-1} - 2\phi_i}{\Delta z^2}$$

wartość działania:

$$S = \int_{-d/2}^{d/2} \left( \frac{1}{2} \left( \frac{d\phi}{dz} \right)^2 - \rho\phi \right) dz \Bigg|$$

pozwała ocenić zbieżność procedur iteracyjnych

ponadto: nieoceniona do kontroli jakości

rozwiązania w metodzie elementów skończonych  
(wybór elementów, wybór funkcji kształtu)

## Zbieżność procedury relaksacyjnej

Przepis relaksacyjny

$$\boxed{\frac{1}{\Delta z^2} (-\phi_{i-1} + 2\phi_i - \phi_{i+1}) = \rho_i} \rightarrow \phi'_i = \frac{\phi_{i+1} + \phi_{i-1} + \Delta z^2 \rho_i}{2}$$

Działanie: 
$$a = \sum_n \left[ \frac{1}{2\Delta z^2} (\phi_n - \phi_{n-1})^2 - \rho_n \phi_n \right]$$

Przyczynek do  $a$  od punktu  $i$ :

$$\begin{aligned} a_i \times 2\Delta z^2 &= (\phi_i - \phi_{i-1})^2 + (\phi_{i+1} - \phi_i)^2 - 2\Delta z^2 \rho_i \phi_i \\ &= 2\phi_i^2 + \phi_{i-1}^2 + \phi_{i+1}^2 - 2\phi_i(\phi_{i-1} + \phi_{i+1}) - 2\Delta z^2 \rho_i \phi_i \end{aligned}$$

Wyliczmy zmianę funkcjonału działania jeśli iteracji poddany zostanie TYLKO  $i$ -ty punkt siatki.

$$\begin{aligned} (a'_i - a_i) \times (2\Delta z^2) &= 2\phi_i'^2 - 2\phi_i^2 - 2(\phi'_i - \phi_i)(\phi_{i-1} + \phi_{i+1}) - 2\Delta z^2 \rho_i (\phi'_i - \phi_i) \\ \dots &= -(\phi_{i+1} + \phi_{i-1} + \Delta z^2 \rho_i - 2\phi_i)^2 / 2 \end{aligned}$$

---

Relaksacja potencjału w każdym z punktów prowadzi do obniżenia wartości działania!  
[równanie P. zostanie rozwiązane gdy działanie obniży się do minimalnego]

## Nadrelaksacja i podrelaksacja

Uogólnijmy schemat relaksacyjny:  
do postaci:

$$\phi'_i = \frac{\phi_{i+1} + \phi_{i-1} + \Delta z^2 \rho_i}{2}$$

$$\phi'_i = (1 - \omega)\phi_i + \omega \frac{\phi_{i+1} + \phi_{i-1} + \Delta z^2 \rho_i}{2}$$

(wygląda jak  
część starego rozwiązania + część nowego)

Wracamy do równania na zmianę działania:

$$\begin{aligned} (a'_i - a_i) \times (2\Delta z^2) &= 2(\phi'_i - \phi_i)(\phi'_i + \phi_i - (\phi_{i-1} + \phi_{i+1}) - \Delta z^2 \rho_i) \\ \dots &= 2\omega(\omega - 2) \left[ -\phi_i + \frac{1}{2}(\phi_{i+1} + \phi_{i-1} + \Delta z^2 \rho_i) \right]^2 < 0 \text{ dla } \omega \in (0, 2) \end{aligned}$$

$\omega = 1$  relaksacja

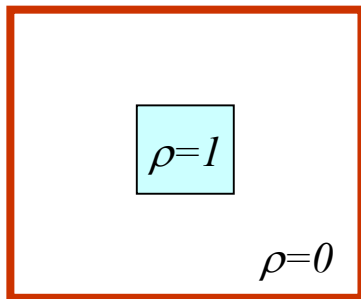
$\omega \in (0, 1)$  – podrelaksacja - mniej „nowego” rozwiązania akceptowane w iteracji

$\omega \in (1, 2)$  – nadrelaksacja – stare rozwiązanie jest usuwane z funkcji

uwaga: dotyczy relaksacji punktowej (nie globalnej) !

Na laboratorium zobaczymy, że globalna niebezpieczna dla  $\omega > 1$





Rozwiązanie:

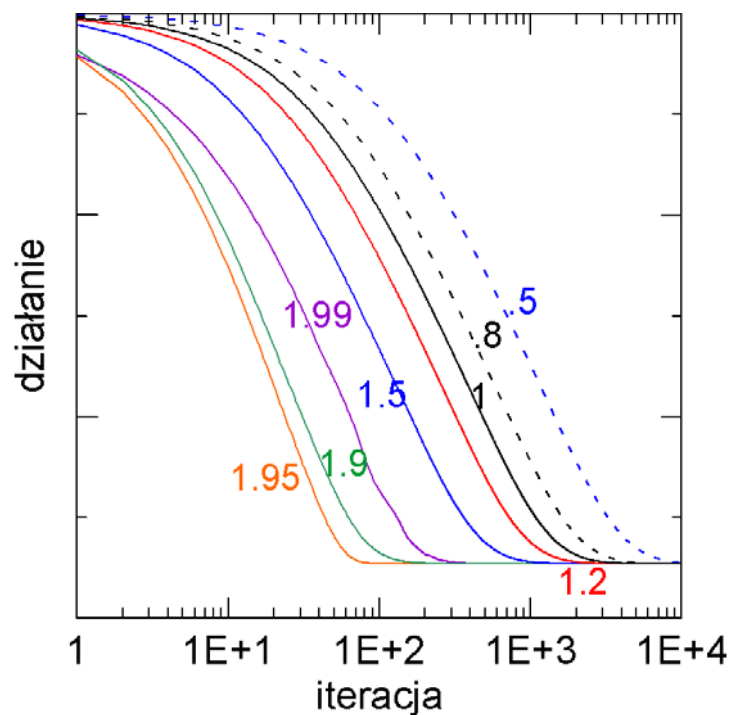
$$\phi = 0 \quad \nabla^2 \phi = -\rho$$

$$S = \int \left( \frac{1}{2} \mathbf{E}^2 - \rho \phi \right) dv = \int \left( \frac{1}{2} (\nabla \phi)^2 - \rho \phi \right) dv$$

pętla po i  
pętla po j

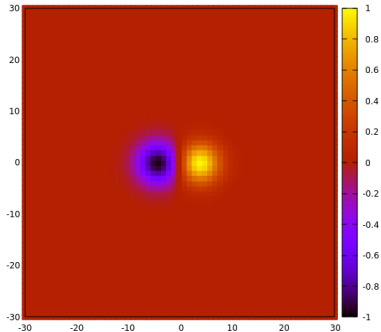
$$\phi'_{ij} = (1 - \omega) \phi_{ij} + \omega \frac{\phi_{(i+1)j} + \phi_{(i-1)j} + \phi_{i(j+1)} + \phi_{i(j-1)} + \Delta x^2 \rho_i}{4}$$

$$\phi_{ij} = \phi'_{ij}$$



# pierwsze zadanie projektowe

- równanie Poissona w 2D
- $\nabla^2 u = -\rho(x, y)$
- gdzie gęstość ładunku  $\rho(x, y) = \exp(-\frac{(x-x_0)^2+y^2}{d^2}) - \exp(-\frac{(x+x_0)^2+y^2}{d^2})$ ,  $d = 4$ ,  $x_0 = 4$ .



# pierwsze zadanie projektowe

- równanie Poissona w 2D
- $\nabla^2 u = -\rho(x, y)$
- gdzie gęstość ładunku  $\rho(x, y) = \exp(-\frac{(x-x_0)^2+y^2}{d^2}) - \exp(-\frac{(x+x_0)^2+y^2}{d^2})$ ,  $d = 4$ ,  $x_0 = 4$ .
- Siatka  $[-N, \dots, N] \times [-N, \dots, N]$  z krokiem  $dx = 1$  w obydwu kierunkach.
- Układ w metalowym uziemionym pudle  $u(x, y) = 0$  gdy  $|x| = N$  lub  $|y| = N$ .  
Przyjmujemy  $N = 31$ .
- Warunek początkowy do rozwiązania: przyjmujemy  $u = 0$ .
- Przepis relaksacyjny
- $u(i, j) := \frac{u(i+1, j) + u(i-1, j) + u(i, j+1) + u(i, j-1) + \rho(i, j)dx^2}{4}$ .
- Jedna iteracja procedury relaksacji polega na zastosowaniu wzoru (2) dla wszystkich punktów na siatce poza brzegiem.

# pierwsze zadanie projektowe

- Zbieżność iterowanej funkcji śledzić licząc całkę z lagranżjanu układu ładunek-pole

$$a = \int_S \left[ \frac{1}{2} \left( \frac{\partial u}{\partial x} \right)^2 + \frac{1}{2} \left( \frac{\partial u}{\partial y} \right)^2 - \rho(i, j)u(i, j) \right] dx dy, \quad (1)$$

- Operator pochodnej jest antyhermitowski  $(f(x), \nabla g(x)) = -(\nabla f(x), g(x))$ , więc

$$a = - \int_S \left[ \frac{1}{2} \left( u \frac{\partial^2 u}{\partial x^2} \right) + \frac{1}{2} \left( u \frac{\partial^2 u}{\partial y^2} \right) + \rho(i, j)u(i, j) \right] dx dy, \quad (2)$$

- w wersji dyskretnej

$$a = - \sum_{i,j=-N+1}^{N-1} \left[ \frac{1}{2} u(i, j) \frac{u(i+1, j) + u(i-1, j) - 2u(i, j)}{dx^2} \right] \quad (3)$$

$$+ \frac{1}{2} u(i, j) \frac{u(i, j+1) + u(i, j-1) - 2u(i, j)}{dy^2} \quad (4)$$

$$+ \rho(i, j)u(i, j)] dx dy. \quad (5)$$

- Im niższa wartość  $a$  tym bliżej dokładnego rozwiązania.
- Jakość rozwiązania przybliżonego można również ocenić odwracając równanie Poissona.

$$\rho'(x, y) = - \frac{u(i+1, j) + u(i-1, j) + u(i, j-1) + u(i, j+1) - 4u(i, j)}{dx^2}. \quad (6)$$

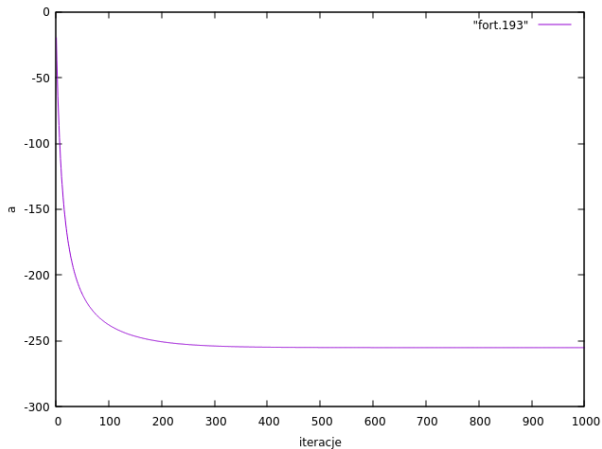
- W jakim stopniu  $\rho'$  reprodukuje  $\rho$ .

## Zadanie 1.1 (40 pkt)

- Liczymy do 500 iteracji pętli relaksacyjnej.
- Startujemy od  $u = 0$ .
- 1.1. Narysować  $a$  od numeru iteracji.
- 1.2. Po setnej i pięćsetnej iteracji narysować  $u$  oraz:
- 1.3.  $\rho'$  i  $\delta(x, y) = \rho'(x, y) - \rho(x, y)$ .

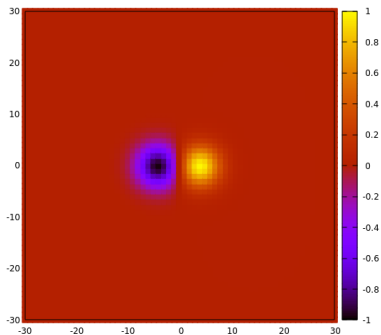
# pierwsze zadanie projektowe

- film labez.gif
- całka funkcjonalna



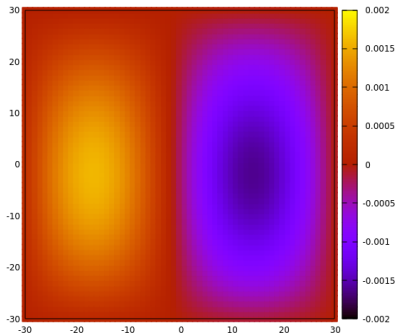
# pierwsze zadanie projektowe

- odtworzenie gęstości po 500 iteracjach



# pierwsze zadanie projektowe

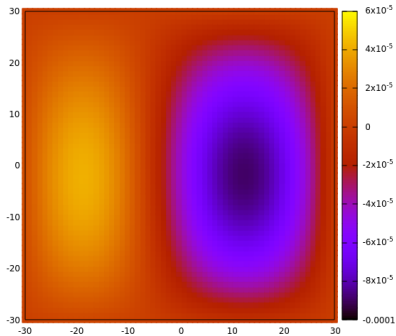
- różnica gęstość odtworzona i dokładna po 500 iteracjach





# pierwsze zadanie projektowe

- różnica gęstość odtworzona i dokładna po 1000 iteracjach



- uwaga: gładki błąd

## Zadanie 1.2 (40 pkt)

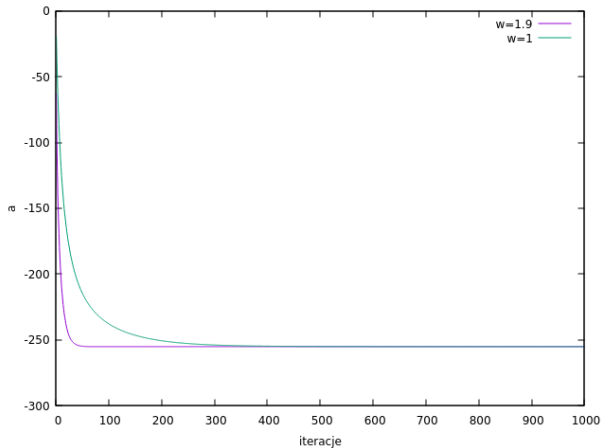
Modyfikacja wzoru relaksacyjnego

$$u(i, j) := (1 - w)u(i, j) + w \frac{u(i + 1, j) + u(i - 1, j) + u(i, j + 1) + u(i, j - 1) + \rho(i, j)dx^2}{4}, \quad (7)$$

z  $w > 1$  daje przepis na iterację nazywaną nadrelaksacją, która bywa szybciej zbieżna od relaksacji dla  $w > 1$ . Przyjąć  $w = 1.9$  i powtórzyć zadanie 1.1.

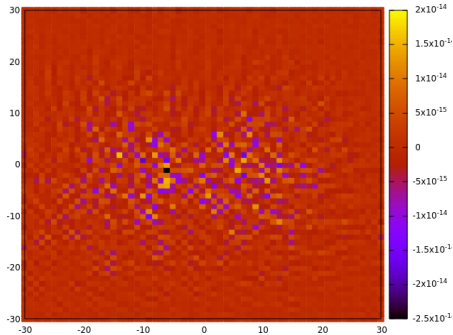
# pierwsze zadanie projektowe

- działanie: relaksacja i nadrelaksacja



# pierwsze zadanie projektowe

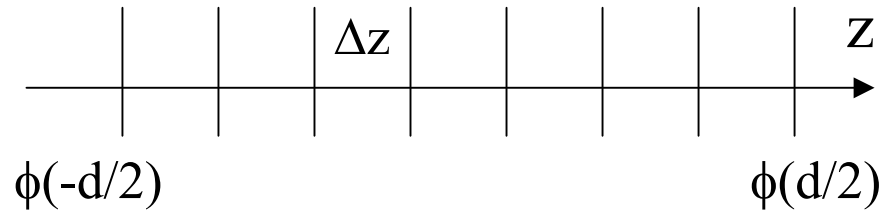
- różnica gęstość odtworzona i dokładna po 1000 iteracjach (nadrelaksacja)



- uwaga: błąd znacznie mniejszy, ale nie gładki.

# Zdyskretyzowane równanie Poissona jako układ równań liniowych

$$\nabla^2 \phi = -\rho \quad \Bigg| \quad \text{1D} \quad \phi(z = -d/2) = \Phi_1, \quad \phi(z = d/2) = \Phi_2$$



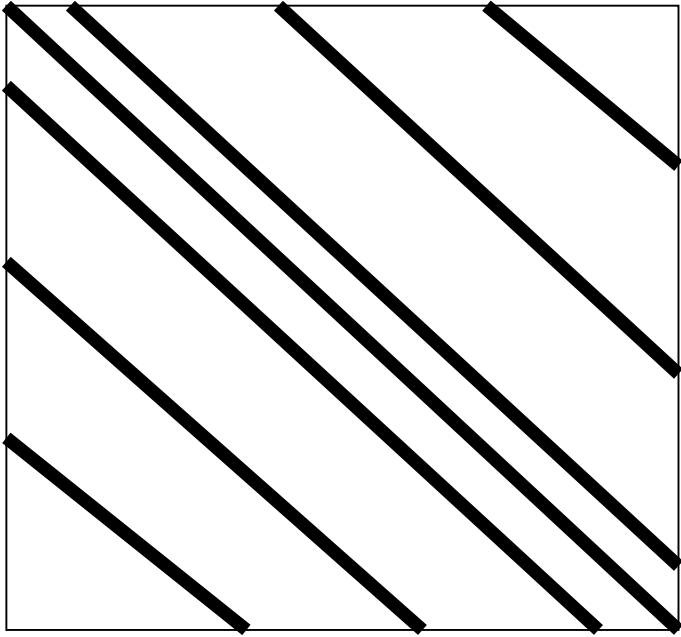
Metoda różnic skończonych

$$\phi_i'' = \frac{\phi_{i+1} + \phi_{i-1} - 2\phi_i}{\Delta z^2}$$

Układ równań  $Au=b$

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & \dots & 0 & 0 & 0 \\ \dots & & & & & & & & \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad u = \begin{pmatrix} \phi(-d/2) \\ \phi(-d/2 + \Delta z) \\ \phi(-d/2 + 2\Delta z) \\ \phi(-d/2 + 3\Delta z) \\ \phi(-d/2 + 4\Delta z) \\ \dots \\ \phi(d/2 - \Delta z) \\ \phi(d/2) \end{pmatrix} \quad b = \begin{pmatrix} \Phi_1 \\ -\Delta z^2 \rho(-d/2 + \Delta z) \\ -\Delta z^2 \rho(-d/2 + 2\Delta z) \\ -\Delta z^2 \rho(-d/2 + 3\Delta z) \\ -\Delta z^2 \rho(-d/2 + 4\Delta z) \\ \dots \\ -\Delta z^2 \rho(d/2 - \Delta z) \\ \Phi_2 \end{pmatrix}$$

niezerowe elementy w macierzy A (N na N), dla 3D równania Laplace'a



macierze są duże: rozmiar dla 3D:  $N = 10^6$  już  
w najprostszych zastosowaniach  
(100 na 100 na 100),  
ale do zapamiętania najwyżej  
 $7 \times 10^6 \times 8$  bajtów = 55 MB

faktycznie dla równania Laplace'a  
wystarczy pamiętać strukturę (kilka liczb)

Rozwiązać metodą dokładną czy iteracyjną?

# dokładne metody rozwiązywania układów równań liniowych

Bartłomiej Szafran

14 kwietnia 2020

# eliminacja Gaussa

- $x_1 + x_2 + 0x_3 + 3x_4 = 4$  (1)
- $2x_1 + x_2 - x_3 + x_4 = 1$  (2)
- $3x_1 - x_2 - x_3 + 2x_4 = -3$  (3)
- $-x_1 + 2x_2 + 3x_3 - x_4 = 4$  (4)
- używamy (1) aby wyeliminować  $x_1$  z pozostałych równań
- (2) - 2(1) zamiast (2)
- (3) - 3(1) zamiast (3)
- (4) + (1) zamiast (4)
- $x_1 + x_2 + 0x_3 + 3x_4 = 4$  (1)
- $0x_1 - x_2 - x_3 - 5x_4 = -7$  (2)
- $0x_1 - 4x_2 - x_3 - 7x_4 = -15$  (3)
- $0x_1 + 3x_2 + 3x_3 + 2x_4 = 8$  (4)
- używamy (2) aby wyeliminować  $x_2$  z pozostałych równań
- (3) - 4(2) zamiast (3)
- (4) + 3(2) zamiast (4)

- $x_1 + x_2 + 0x_3 + 3x_4 = 4$  (1)
- $0x_1 - x_2 - x_3 - 5x_4 = -7$  (2)
- $0x_1 + 0x_2 + 3x_3 + 13x_4 = 13$  (3)
- $0x_1 + 0x_2 + 0x_3 - 13x_4 = -13$  (4)
- URL został zredukowany do postaci trójkątnej, dolnoprzekątniowej, która pozwala na rozwiązanie przed wyliczenie kolejnych  $x_i$  o malejącym indeksie  $i$  (backsubstitution)
- $x_4 = 1$ ,
- $3x_3 + 13 = -13 \rightarrow x_3 = 0$
- $-x_2 - 5 = -7 \rightarrow x_2 = 2$
- $x_1 + 2 + 3 = 4 \rightarrow x_1 = -1$



# eliminacja Gaussa

- $\mathbf{Ax} = \mathbf{b}$
- $E_1: a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$
- $E_2: a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$
- $\vdots$
- $E_n: a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n$
- macierz rozszerzona  $\tilde{\mathbf{A}} = [\mathbf{A}|\mathbf{b}]$  rozmiar  $n \times (n+1)$  z  $a_{k,n+1} = b_k$
- jeśli  $a_{11} \neq 0$  wykonujemy  $E_j - \frac{a_{j1}}{a_{11}}E_1 \rightarrow E_j$  dla  $j = 2, \dots, n$
- itd
- jeśli  $a_{ii} \neq 0$  wykonujemy  $E_j - \frac{a_{ji}}{a_{ii}}E_i \rightarrow E_j$  dla  $j = i+1, \dots, n$
- $\tilde{\mathbf{A}} \rightarrow \tilde{\mathbf{A}}'$

$$\bullet \tilde{\mathbf{A}}' = \begin{bmatrix} a'_{11} & a'_{12} & \dots & a'_{1n} & \vdots & a'_{1,n+1} \\ 0 & a'_{22} & \dots & a'_{2n} & \vdots & a'_{2,n+1} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & a'_{nn} & \vdots & a'_{n,n+1} \end{bmatrix}$$

# eliminacja Gaussa

- $$\tilde{\mathbf{A}}' = \begin{bmatrix} a'_{11} & a'_{12} & \dots & a'_{1n} & \vdots & a'_{1,n+1} \\ 0 & a'_{22} & \dots & a'_{2n} & \vdots & a'_{2,n+1} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & a'_{nn} & \vdots & a'_{n,n+1} \end{bmatrix}$$

- 

$$\begin{array}{cccccc} a'_{11}x_1 & a'_{12}x_2 & \dots & a'_{1n}x_n & = & a'_{1,n+1} \\ 0 & a'_{22}x_2 & \dots & a'_{2n}x_n & = & a'_{2,n+1} \end{array}$$

- 

$$\begin{array}{cccccc} \vdots & \ddots & \ddots & \vdots & & \vdots \end{array}$$

$$\begin{array}{cccccc} 0 & 0 & \ddots & a'_{n-1,n-1}x_{n-1} & a'_{n-1,n}x_n & = & a'_{n-1,n+1} \\ 0 & 0 & & \dots & a'_{nn}x_n & = & a'_{n,n+1} \end{array}$$

- 

$$x_n = \frac{a'_{n,n+1}}{a'_{n,n}}$$

- 

$$x_{n-1} = \frac{a'_{n-1,n+1} - a'_{n-1,n}x_n}{a'_{n-1,n-1}}$$

- 

$$x_i = \frac{a'_{i,n+1} - \sum_{j=i+1}^n a'_{i,j}x_j}{a'_{i,i}}$$

# eliminacja Gaussa

- (i)-ty krok: jeśli  $a_{ii}^{(i)} \neq 0$  wykonujemy  $E_j - \frac{a_{ji}}{a_{ii}} E_i \rightarrow E_j$  dla  $j = i + 1, \dots, n$
- procedura zawiedzie jeśli  $a_{ii}^{(i)}$  *pivot* będzie zerowy
- przykład  $\mathbf{A} = \begin{bmatrix} 1 & -1 & 2 & -1 \\ 2 & -2 & 3 & -3 \\ 1 & 1 & 1 & 0 \\ 1 & -1 & 4 & 3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -1 & 2 & -1 \\ 0 & 0 & -1 & -1 \\ 0 & 2 & -1 & 1 \\ 0 & 0 & 2 & 4 \end{bmatrix} \rightarrow (2) \leftrightarrow (3)$
- Jeśli wygeneruje się zerowy pivot, lub bardzo blisku zeru, należy zamienić miejscami równanie ( $i$ ) oraz ( $k$ )  $>$  ( $i$ ), takie że  $a_{kk}^{(i)}$  duże
- problem jest nie do usunięcia, jeśli URL jest osobiwy - wtedy nie ma jednoznacznego rozwiązania URL
- *Gaussian elimination with pivoting and backsubstitution*

# eliminacja Gaussa

- liczba operacji dla eliminacji Gaussa
- dla danego  $i$ , dla  $E_j - \frac{a_{ji}}{a_{ii}} E_i \rightarrow E_j$  dla  $j = i, \dots, n$
- liczba operacji dzielen:  $(n - i)$ , liczba mnożeń  $(n - i) \times (n - i)$
- liczba dodawań:  $(n - i) \times (n - i)$
- dla danego  $i$  operacji jest rzędu  $n^2$
- ogólnie liczba operacji dla eliminacji jest rzędu  $n^3$
- dla podstawienia wstecznego: liczba operacji jest  $n^2$

# faktoryzacja macierzy

- problem:
- $\mathbf{Ax} = \mathbf{b}$
- rozwiązywany wielokrotnie dla różnych prawych stron
- dobrą inwestycją jest faktoryzacja macierzy na iloczyn macierzy trójkątnych (koszt  $n^3$ , tak że kolejne prawe strony znajdziemy wykonując podstawienia podobne do backsubstitution).
- **dekompozycja LU** : iloczyn dolno-  $\mathbf{L}$  i górno-  $\mathbf{U}$  przekątniowych macierzy
- $\mathbf{A} = \mathbf{LU}$
- $\mathbf{LUx} = \mathbf{b}$
- $\mathbf{Ux} = \mathbf{y}$  (2)
- $\mathbf{Ly} = \mathbf{b}$  (1) - rozwiązujemy na  $\mathbf{y}$
- kiedy (1) rozwiązane, rozwiązujemy (2)

# dekompozycja LU

- problem:

- $\mathbf{A} = \mathbf{LU}$ ,  $a_{ij} = \sum_{k=1}^n l_{ik} u_{kj}$ ,  $a_{ij} = \sum_{k=1}^{i-1} l_{ik} u_{kj} + u_{ij}$

- $$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \ddots & u_{2n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & u_{nn} \end{bmatrix}$$

- $a_{1j} = u_{1j}$  (pierwszy wiersz  $\mathbf{U}$ )
- $a_{i1} = l_{i1} u_{11} \rightarrow l_{i1}$  (pierwsza kolumna  $\mathbf{L}$ )
- $a_{2j} = u_{2j} + l_{21} u_{1j} \rightarrow u_{2j}$  (drugi wiersz  $\mathbf{U}$ )
- $a_{j2} = l_{j1} u_{12} + l_{j2} u_{22} \rightarrow l_{j2}$  (druga kolumna  $\mathbf{L}$ )
- złożoność  $n^3$ , po faktoryzacji iloczyn diagonali  $\mathbf{U}$  to wyznacznik  $\mathbf{A}$ .

# dekompozycja LU

- $\mathbf{Ly} = \mathbf{b}$

- $$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

- $\mathbf{Ux} = \mathbf{y}$

- $$\begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \ddots & u_{2n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & u_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

# dekompozycja LU

- ogólne metody na rozwiązywanie układów równań liniowych, dla problemu  $n \times n$  wykorzystują pamięć rzędu  $n^2$
- z góry znany czas rozwiązania problemu, rzędu  $n^3$



# LU: wariant dla macierzy trójkątnej

dla  $n=N-1$

$$\mathbf{A} = \begin{pmatrix} a_1 & c_1 & & & \\ b_2 & a_2 & c_2 & & \\ & & \ddots & \ddots & \ddots \\ & & & b_n & a_n & c_n \end{pmatrix}$$

rozwiązać  $\mathbf{A}\mathbf{u}=\mathbf{f}$

dekompozycja

$$\mathbf{A}=\mathbf{LU}$$

$$\mathbf{LU}\mathbf{u}=\mathbf{f}$$

$$\mathbf{L} = \begin{bmatrix} 1 & & & & \\ \beta_2 & 1 & & & \\ & \beta_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & \beta_n & 1 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \alpha_1 & \gamma_1 & & & \\ & \alpha_2 & \gamma_2 & & \\ & & \ddots & \ddots & \\ & & & \alpha_{n-1} & \gamma_{n-1} \\ & & & & \alpha_n \end{bmatrix}$$

$$\gamma_i = c_i$$

$$\alpha_1 = a_1,$$

$$\beta_i = b_i / \alpha_{i-1}$$

$$\alpha_i = a_i - \beta_i \gamma_{i-1}$$

$$\mathbf{U}\mathbf{u}=\mathbf{z}$$

$$\mathbf{L}\mathbf{z}=\mathbf{f}$$

$$z_1 = f_1$$

$$z_2 = f_2 - \beta_2 z_1$$

$\vdots$

$$z_i = f_i - \beta_i z_{i-1}$$

$$u_n = z_n / \alpha_n$$

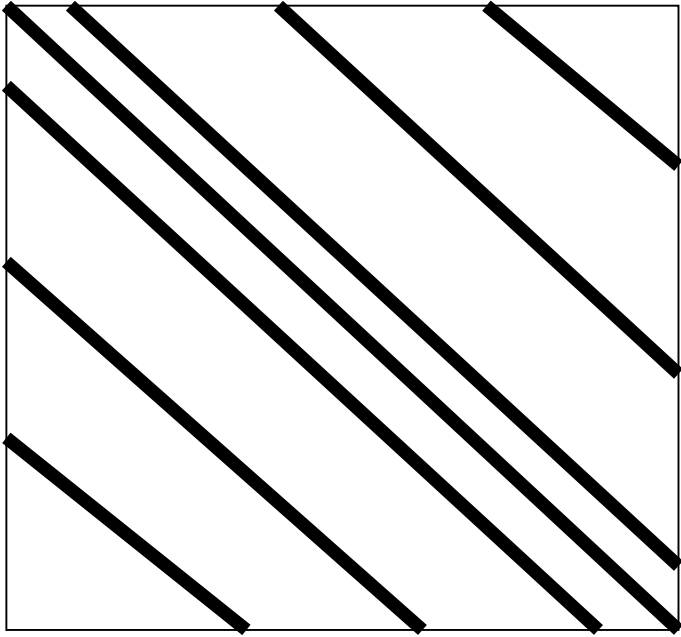
$$u_{n-1} = \frac{z_{n-1} - \gamma_{n-1} u_n}{\alpha_{n-1}}$$

itd..

5n mnożeń /dzielen  
3n dodawań / odejmowań

podczas gdy eliminacja Gaussa  
 $n^3/3$  operacji

niezerowe elementy w macierzy A (N na N), dla 3D równania Laplace'a



macierze są duże: rozmiar dla 3D:  $N = 10^6$  już  
w najprostszych zastosowaniach  
(100 na 100 na 100),  
ale do zapamiętania najwyżej  
 $7 \times 10^6 \times 8$  bajtów = 55 MB

faktycznie dla równania Laplace'a  
wystarczy pamiętać strukturę (kilka liczb)

Rozwiązać metodą dokładną czy iteracyjną?

Rozwiązujemy raczej metodą iteracyjną a nie dokładną:

metody „dokładne”:

- przepis dający rozwiązanie w ściśle określonej liczbie kroków
- złożoność rzędu  $N^3$  [(operacje na macierzy El. Gaussa, LU  $N^3$ )  
podstawienie –  $N^2$ ]
- operacje na macierzy – niszczą jej rzadką, pasmową strukturę (ogólna macierz double  $10^6$  na  $10^6$  - około 1 TB)

metody „iteracyjne”:  $x := Mx + c$  (dla układu  $Ax = b$ ,  
macierz iteracji  $M$ , różna od  $A$ )

- każda iteracja  $N^2$  operacji
- nie zmienia struktury macierzy
- problem zbieżności i strategii prowadzenia iteracji

układ równań:  $Ax=b$

metody iteracyjne, postać ogólna:  $x:=Mx+c$

konstrukcja  $M$ : dokładne rozwiązanie układu musi spełniać  
przepis iteracyjny  $x=Mx+c$   
dobrze gdy  $M$  rzadka (dla Jakobiego jest, ale dla GS – nie)

równanie własne  $Mv_l = \lambda_l v_l$

metody iteracyjne zbieżne wtedy i tylko wtedy, gdy promień  
spektralny macierzy iteracji  $M$  [największy moduł wartości własnej]

$$\rho(M) < 1$$