

Technika modelowania

Fast Marching Method

Wojciech Dębski
IGF PAN

20.IV.2010

Syntetyczne seismogramy i ich charakterystyki

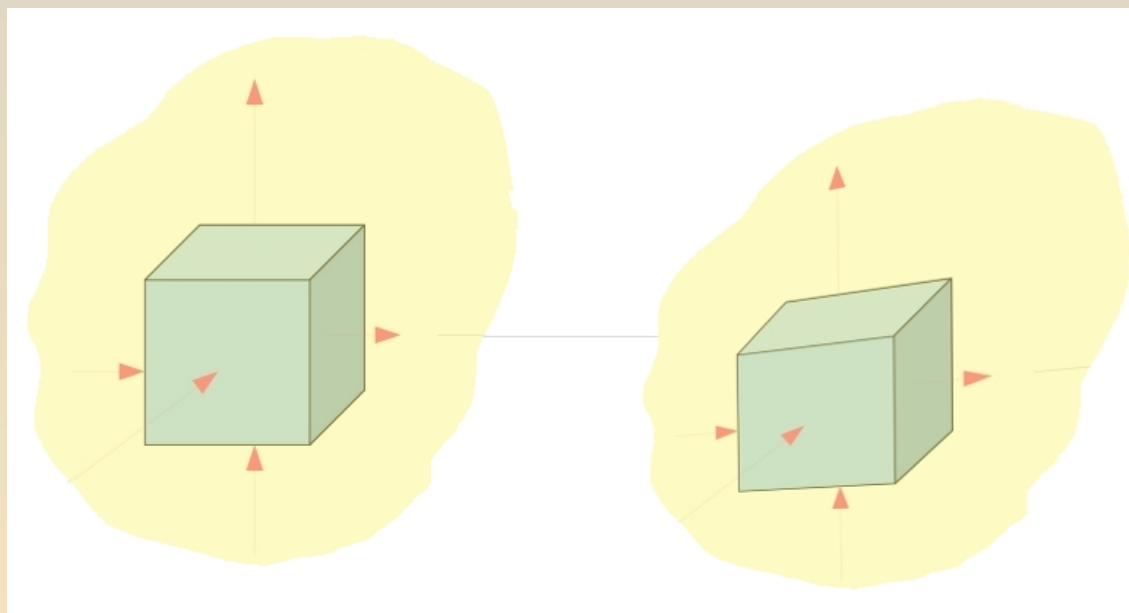
- ◆ inwersja pola falowego
- ◆ lokalizacja ognisk wstrząsów
- ◆ tomografia prędkościowa, tłumieniowa
- ◆ anizotropia
- ◆ odbicia, załamania, konwersje fal

Propagacja fal w ośrodku niejednorodnym

- ◆ równanie falowe
- ◆ trasowanie promieni (ang. ray tracing)
- ◆ propagacja frontu falowego

Równanie falowe

$$\vec{F} = m\vec{a}$$



Równanie falowe

$$dm = \rho dV$$

$$\epsilon^{kl} = \partial^k u^l + \partial^l u^k$$

$$\sigma^{ij} = c_{ijkl} \epsilon^{kl}$$

$$\rho \frac{\partial^2 u^i}{\partial t^2} - \partial_j \sigma^{ij} = S^i(\vec{r}, t)$$

Równanie falowe - ośrodek izotropowy jednorodny

$$c_{ijkl} = \lambda \delta_{ij} \delta_{kl} + 2\mu(\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk})$$

$$\rho \frac{\partial^2 \vec{u}}{\partial t^2} - (\lambda + 2\mu) \nabla (\nabla \cdot \vec{u}) + \mu \nabla \times (\nabla \times \vec{u}) = \vec{S}(\vec{r}, t)$$

Równanie falowe - ośrodek niejednorodny

$$\begin{aligned}\rho \frac{\partial^2 \vec{u}}{\partial t^2} = & (\lambda + \mu) \nabla (\nabla \cdot \vec{u}) + \mu \Delta \vec{u} \\ & + (\nabla \lambda) \nabla \cdot \vec{u} + \nabla \mu \vec{r} \cdot (\nabla \vec{r} \cdot \vec{u}) \\ & + 2(\nabla \mu \cdot \nabla) \vec{u}\end{aligned}$$

Metody rozwiązywania

- ◆ analityczne (asymptotyczne)
- ◆ metoda różnic skończonych
- ◆ metody spektralne (elementów spektralnych)
- ◆ metody elementów skończonych
- ◆ ...

Ray tracing

Rozwinięcie w szereg *promieniowy*

$$\vec{u}(\vec{r}, t) = \sum_{k=0}^{\infty} \vec{u}_k(\vec{r}) F_k(t - \tau(\vec{r}))$$

$$F_k(\xi) = (-i\omega)^{-k-\gamma} \exp(-i\omega\xi)$$

$$\omega \gg 1 \implies F_k = 0; \quad k = 1, 2, \dots$$

Ray tracing - wniosek I

$$\rho \vec{u}_0 = (\lambda + \mu) \nabla \tau (\vec{u}_0 \cdot \nabla \tau) + \mu (\nabla \tau)^2 \vec{u}_0$$

Rozwiązanie niezerowe

$$(\nabla \tau)^2 = v_{\alpha, \beta}^{-2}$$

Ray tracing - wniosek II

Front falowy:

$$\tau(\vec{r}) = \text{const.} \quad \nabla\tau \perp \tau(\vec{r}) = \text{const.}$$

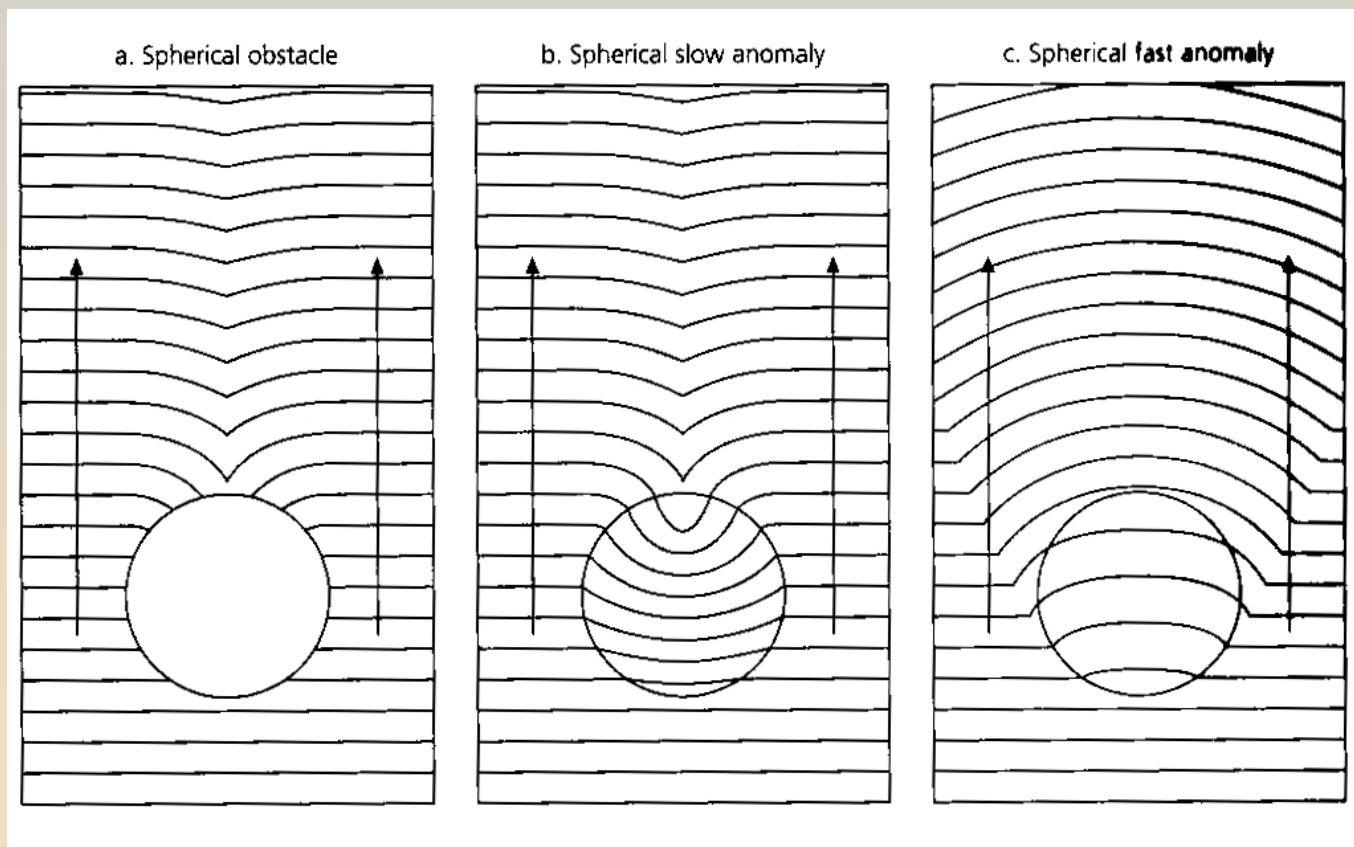
Koncepcja “promieni sejsmicznych”

$$\Delta T = \int_{\Gamma} \frac{ds}{v}$$

Problem:

$$\nabla\tau \angle \tau(\vec{r}) = \text{const.}$$

Ray tracing - problem



Propagacja frontu falowego

$T(\vec{r})$ - czas zaobserwowania fali (frontu falowego)

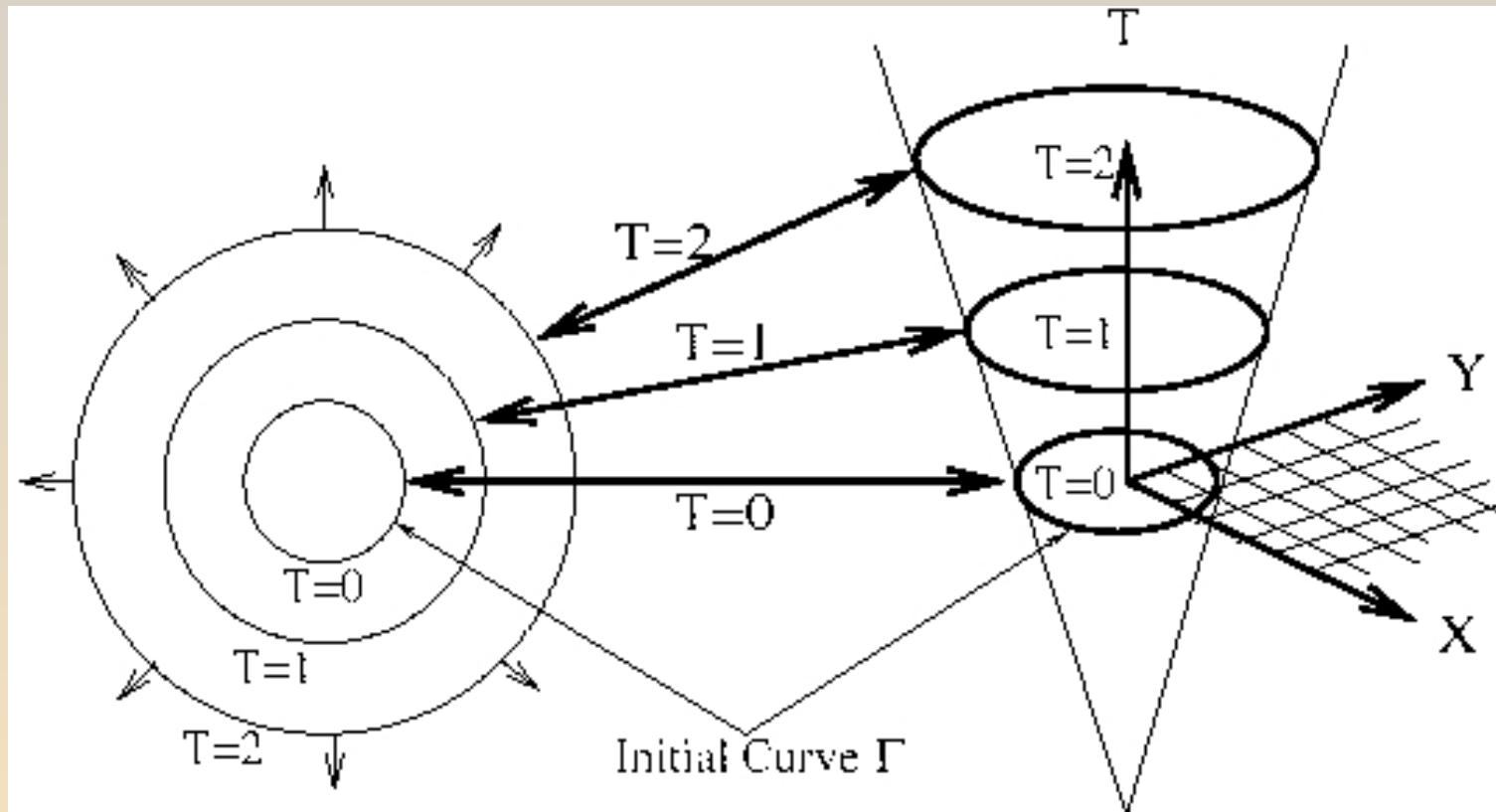
$$v(\vec{r}) |\nabla T| = 1$$

$$T|_{\Gamma_0} = 0$$

Rozwiązanie (front falowy):

$$\Gamma(t) = \{\vec{r} = (x, y, z) : T(\vec{r}) = t\}$$

Front falowy - zagadnienie brzegowe



Przybliżenie FD

$$D_{ijk}^{-x}T = 1/\Delta_x (T(x_{i-1}, y_j, z_k) - T(x_i, y_j, z_k))$$

$$D_{ijk}^{+x}T = 1/\Delta_x (T(x_{i+1}, y_j, z_k) - T(x_i, y_j, z_k))$$

$$\left[\begin{array}{l} \max \left(D_{ijk}^{-x}T, -D_{ijk}^{+x}T, 0 \right)^2 \\ \max \left(D_{ijk}^{-y}T, -D_{ijk}^{+y}T, 0 \right)^2 \\ \max \left(D_{ijk}^{-z}T, -D_{ijk}^{+z}T, 0 \right)^2 \end{array} + \right]^{1/2} = s_{ijk}$$

Przybliżenie FD

Klasyczne podejście iteracyjne jest numerycznie nieefektywne:
 $O(N^4)$

Fast Marching Method (FMM)

- ◆ Zasada Huyghensa
- ◆ technika Dijkstrasa

Fast Marching Method - podstawy

- ◆ węzły siatki do których dotarł front falowy stają się źródłem fal kulistych o promieniu $\sim v_{ijk}$
- ◆ Upwind FD schemat jest użyty do “przesunięcia” frontu falowego

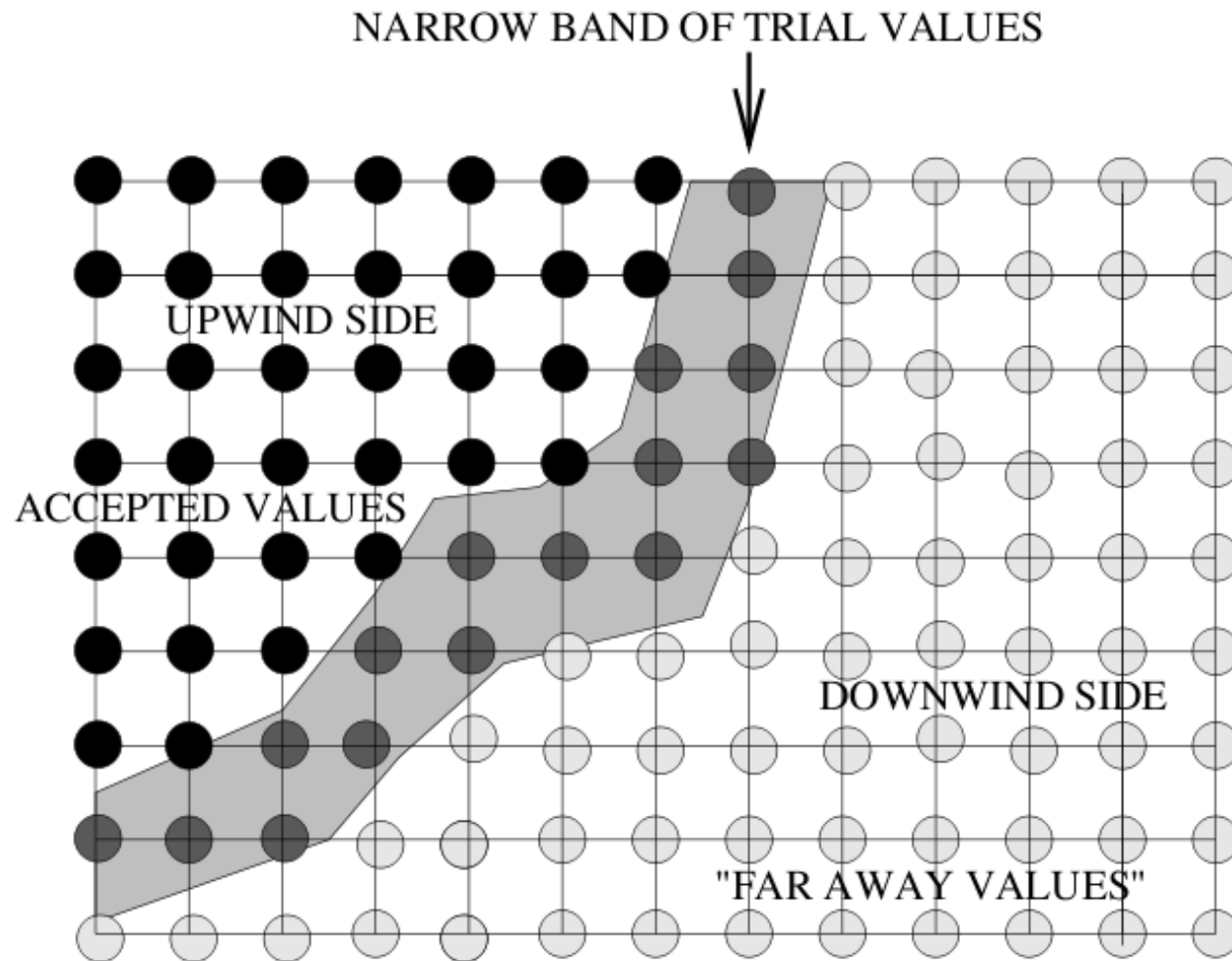
Ważna jest kolejność procesowania węzłów siatki przy “propagacji” frontu falowego

Fast Marching Method - algorytm

Podział siatki na części:

- ◆ wewnętrzną (*accepted*)
- ◆ brzegową - (*considered*) przynajmniej jeden z sąsiadów jest *accepted*
- ◆ zewnętrzną (*far*)

Podział siatki



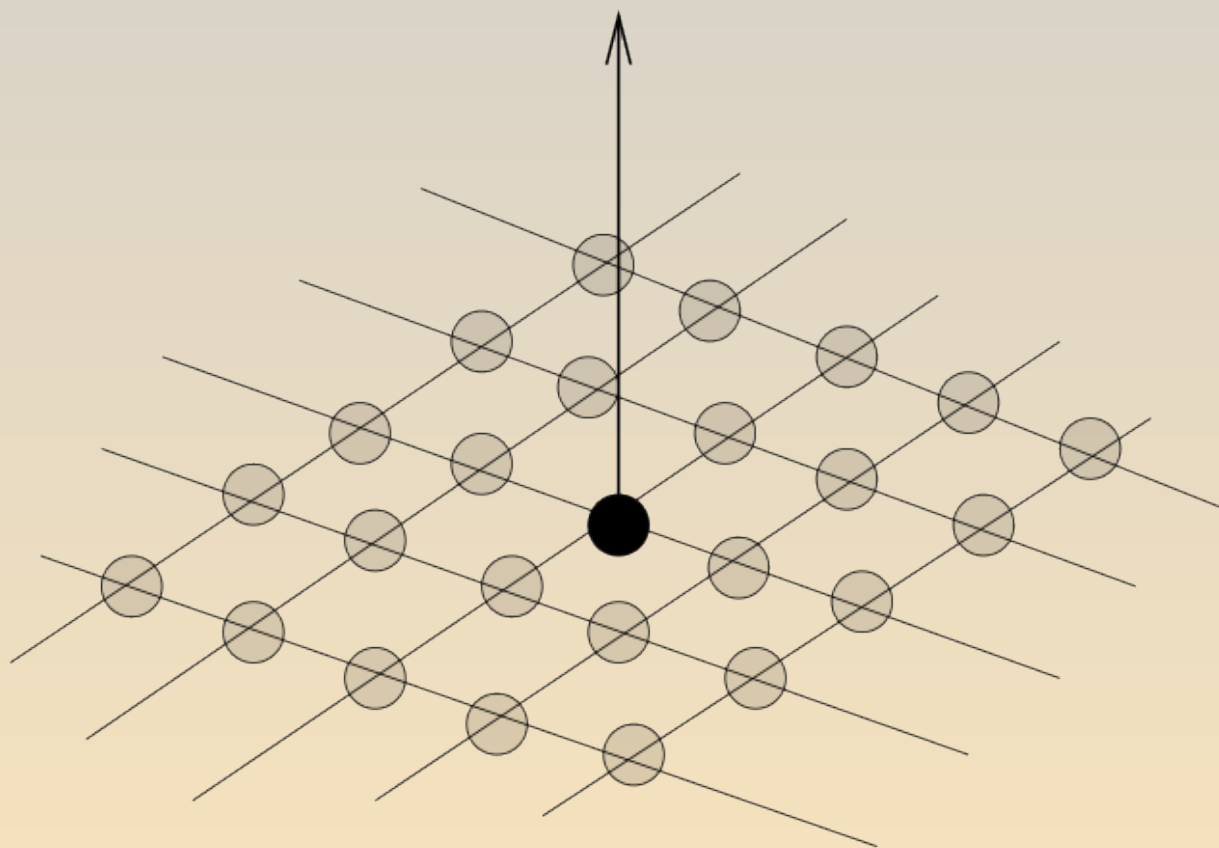
Fast Marching Method - algorytm

- ◆ dla wężła K *considered* o najmniejszym T wszystkich sąsiadów *far* oznaczamy jako *considered*
- ◆ obliczamy T dla wszystkich sąsiadów K używając formuły FD zakładając $T = \infty$ dla węzłów typu *far-away* i już znane wartości dla *accepted*
- ◆ węzeł K oznaczamy jako *accepted*
- ◆ powtarzamy dopóki wszystkie węzły nie staną się *accepted*

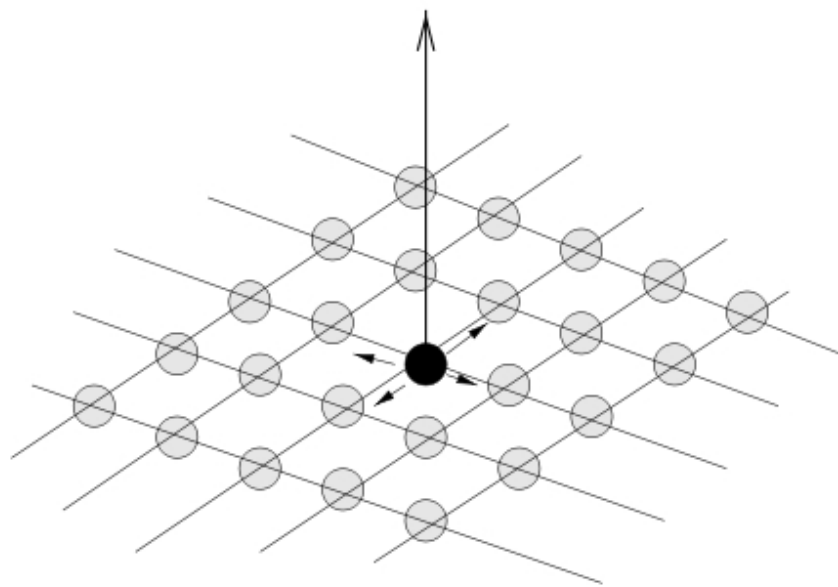
Fast Marching Method - algorytm

- ◆ efektywność $O(k_N N^3)$
 - ★ N^3 - wszystkie węzły sieci
 - ★ $k_N = \log N$ - dokładne sortowanie warstwy brzegowej (heap sorting)
 - ★ $k_N = \text{const}$ - sortowanie przybliżone
- ◆ algorytm bezwzględnie stabilny

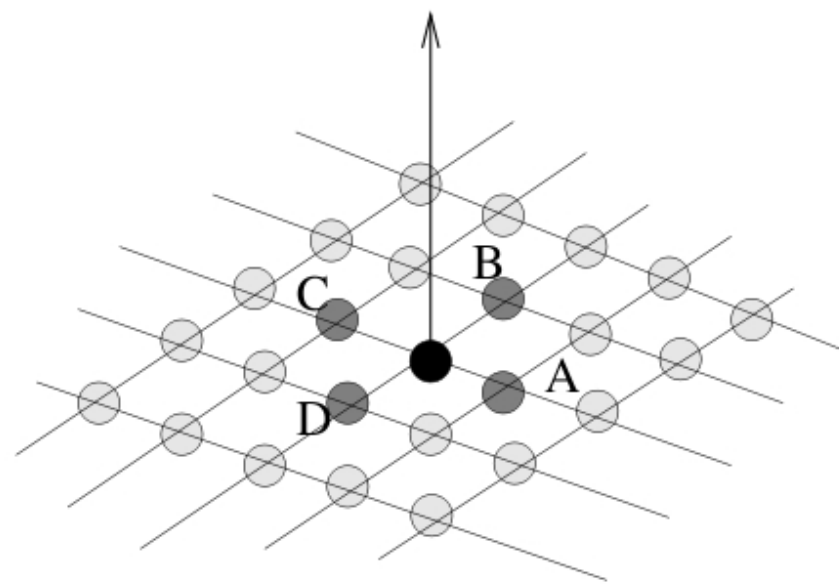
Algorytm



Algorytm

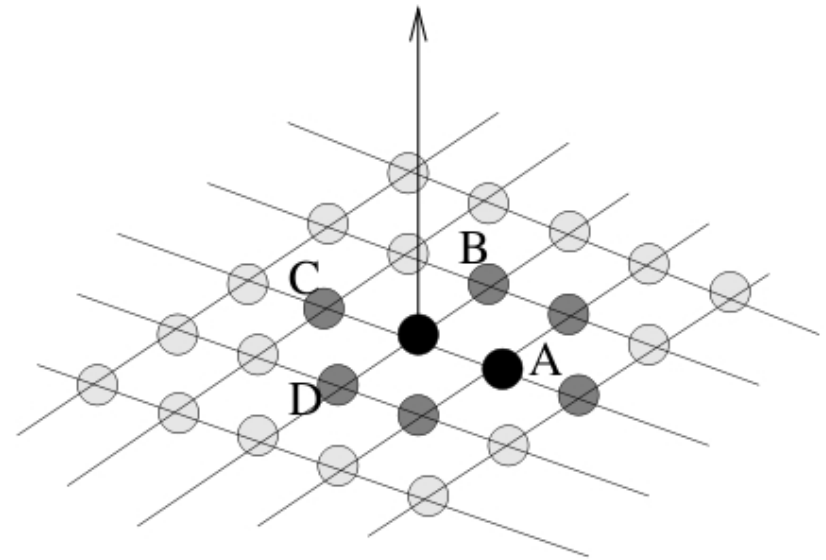
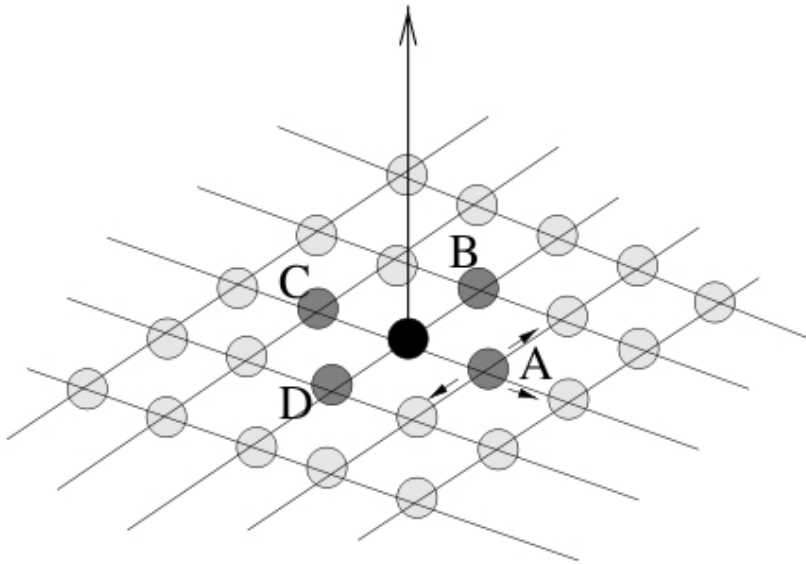


(i) Update “downwind”



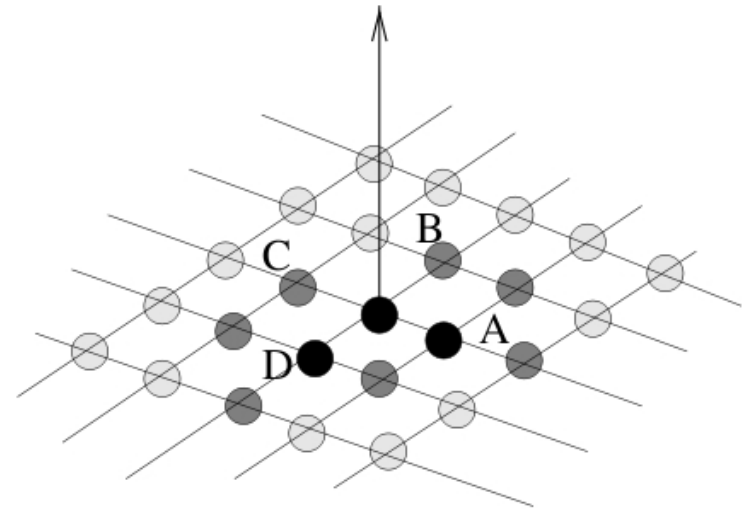
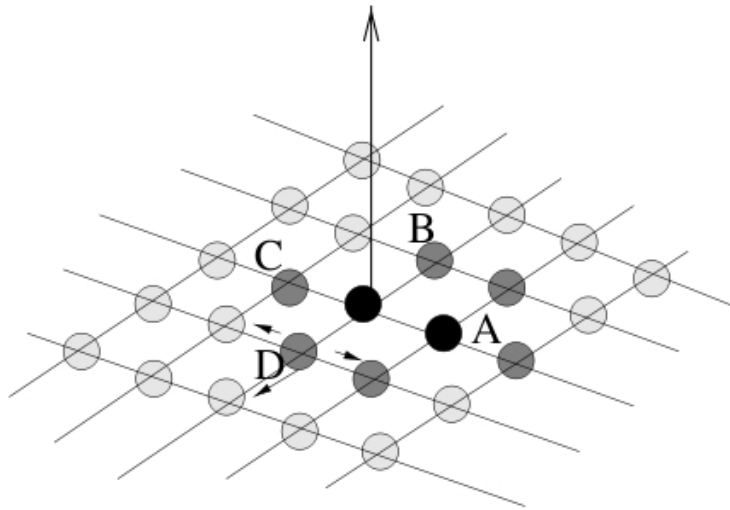
(ii) Compute new possible values

Algorytm



) Choose smallest dark gray sphere (for example, “A”) (iv) Freeze value at A, update neighboring downwind p

Algorytm



Choose smallest dark gray sphere (for example, “D”) (vi) Freeze value at D, update neighboring downwind points

Przykład- model Marmousi

