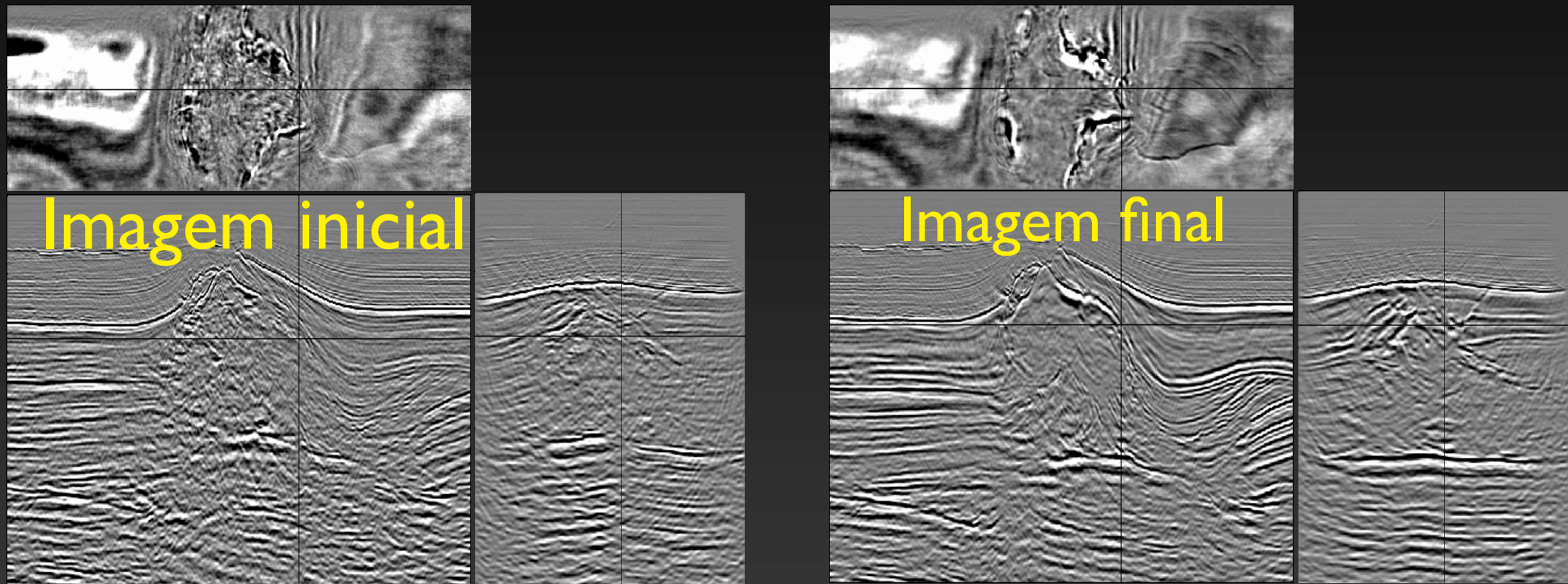


Análise de velocidade de migração com tomografia de campos de onda

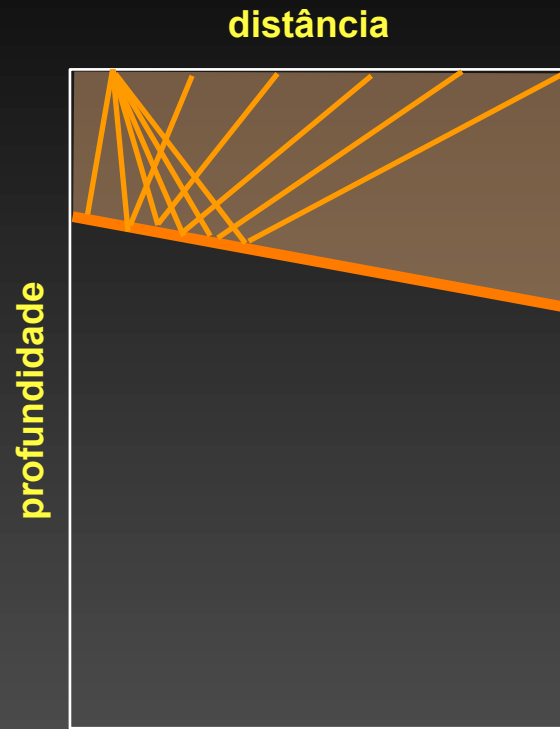
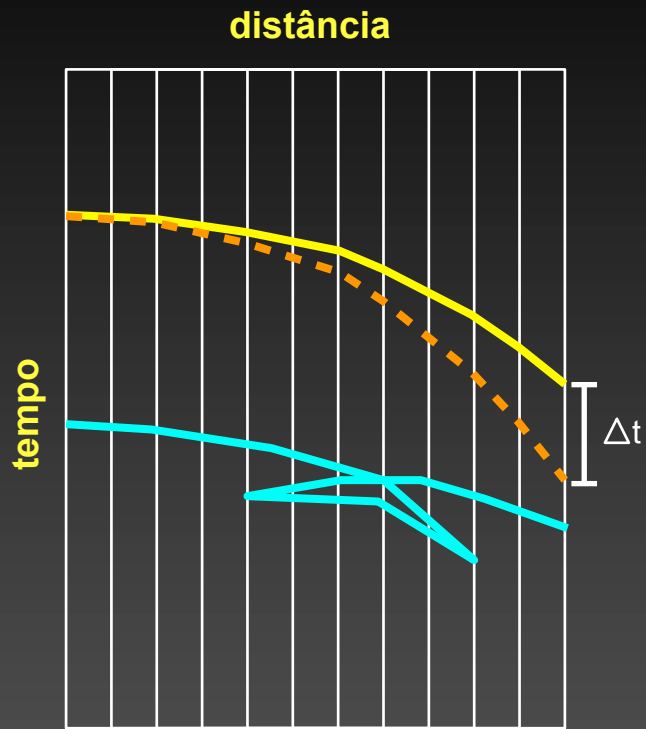
Claudio Guerra



Obtendo velocidade com tomografia

Domínio	dados	imagem
Física		
raios	Tomografia de tempo de trânsito	AVM por tomografia de raios
ondas	Inversão da forma de onda	AVM com tomografia de campos de ondas

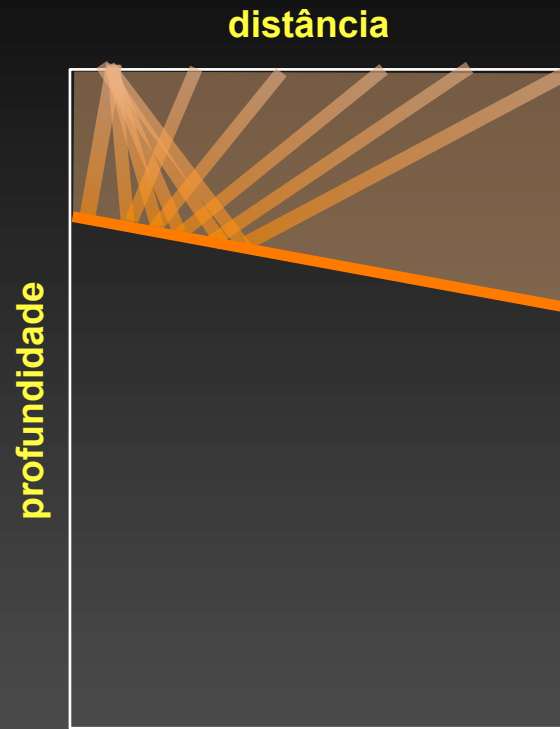
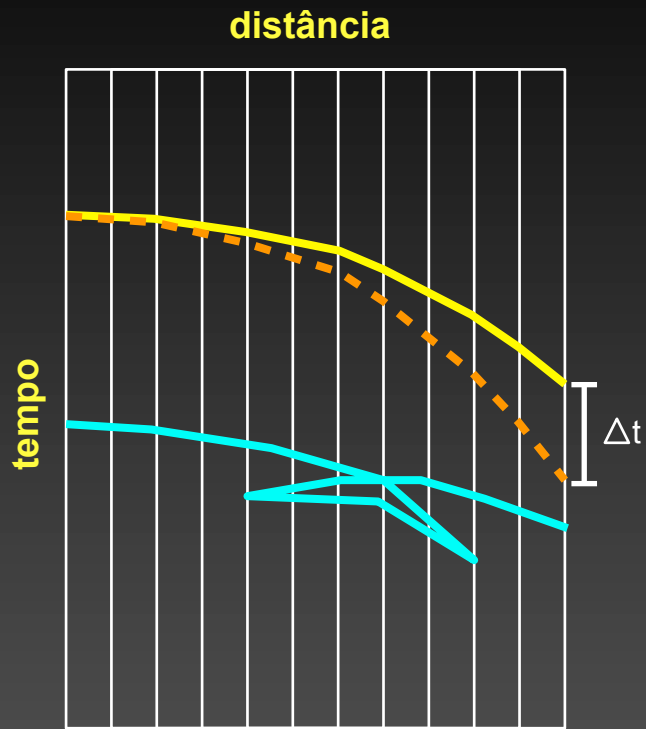
Tomografia de tempo de trânsito



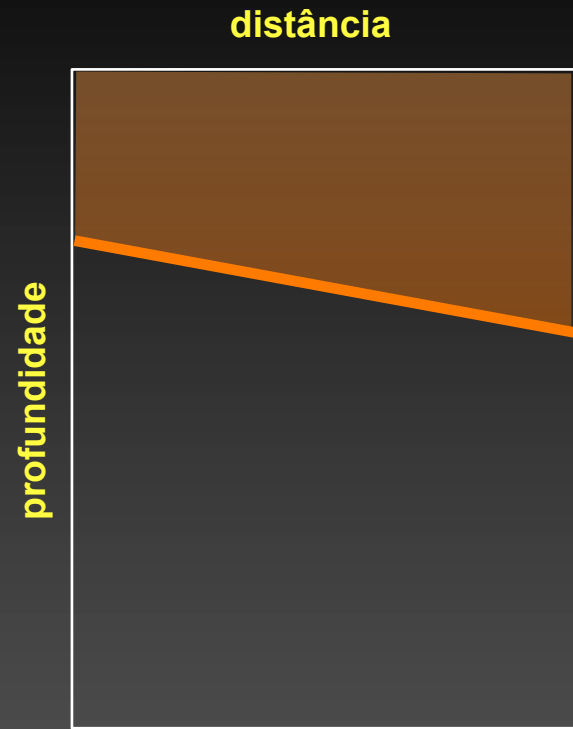
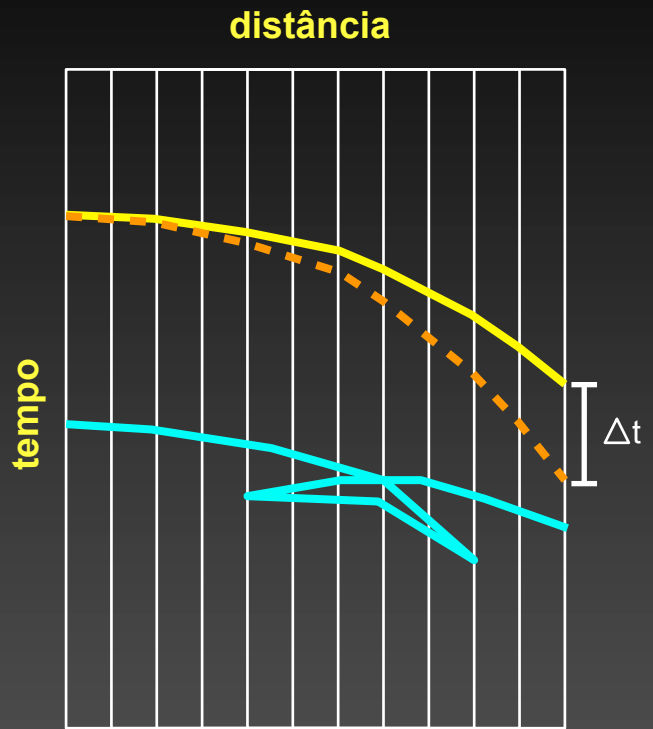
$$\Delta t = \mathbf{T} \Delta s$$

$$\Delta s = \mathbf{T}' \Delta t$$

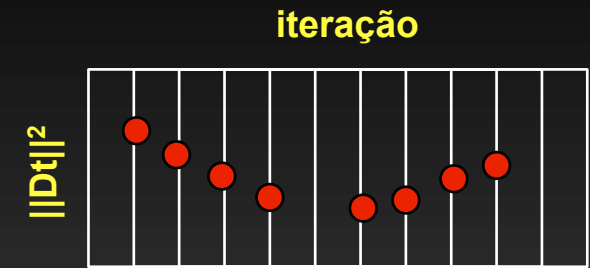
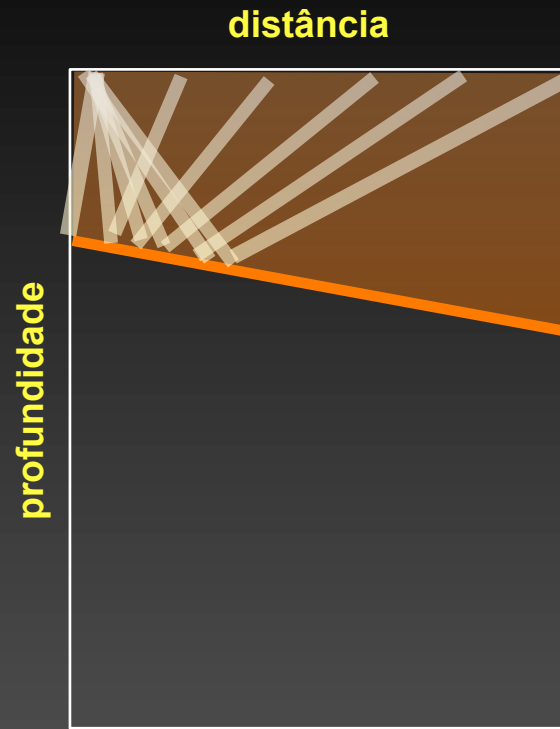
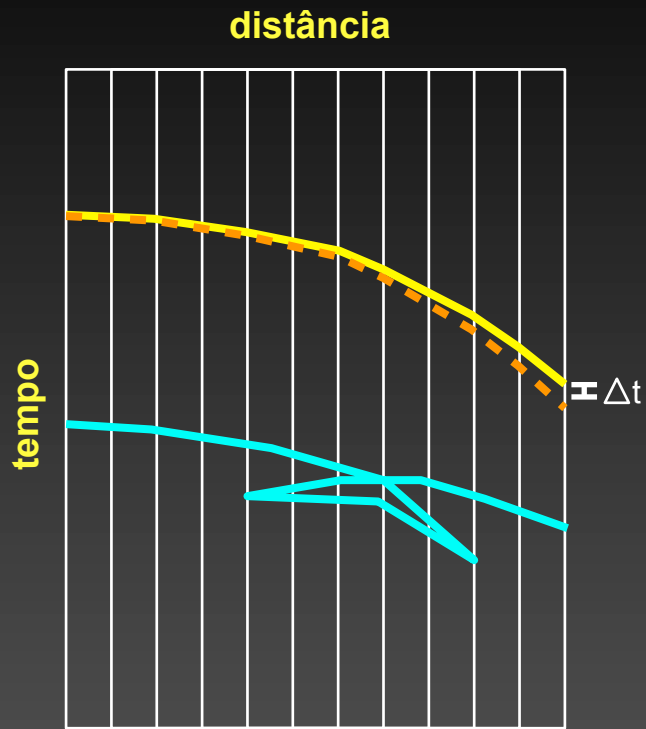
Tomografia de tempo de trânsito



Tomografia de tempo de trânsito

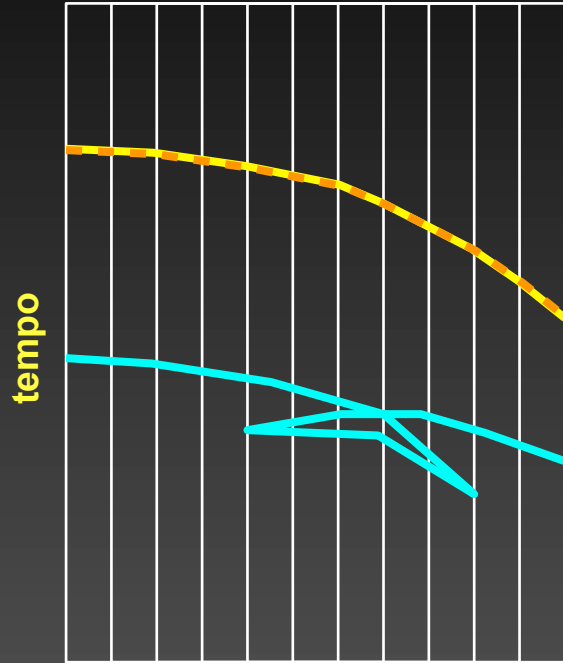


Tomografia de tempo de trânsito

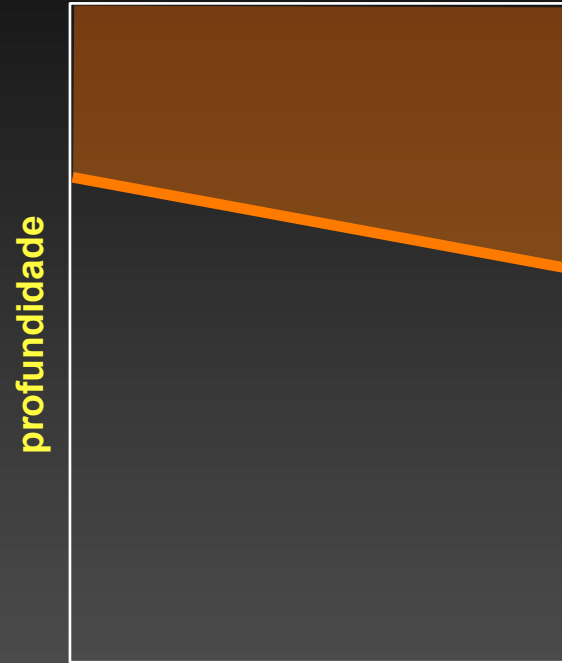


Tomografia de tempo de trânsito

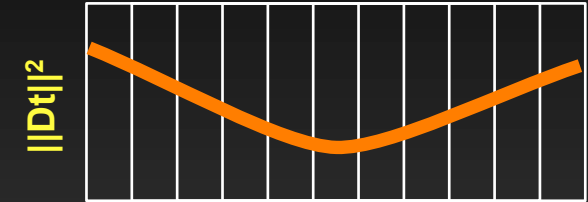
distância



distância

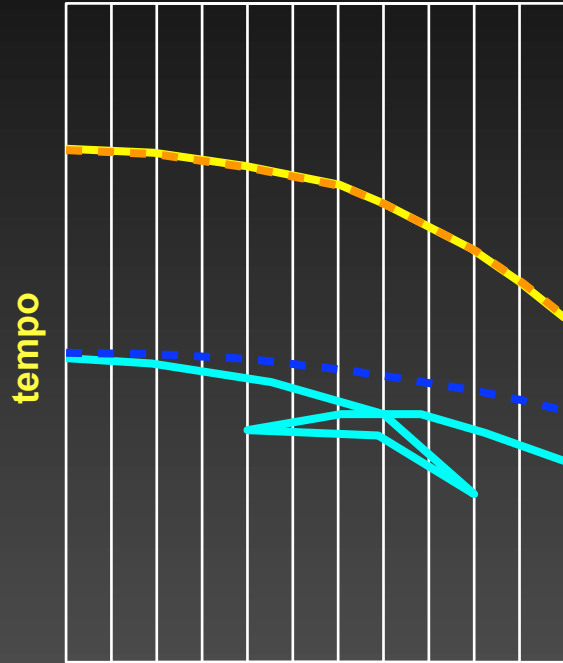


iteração

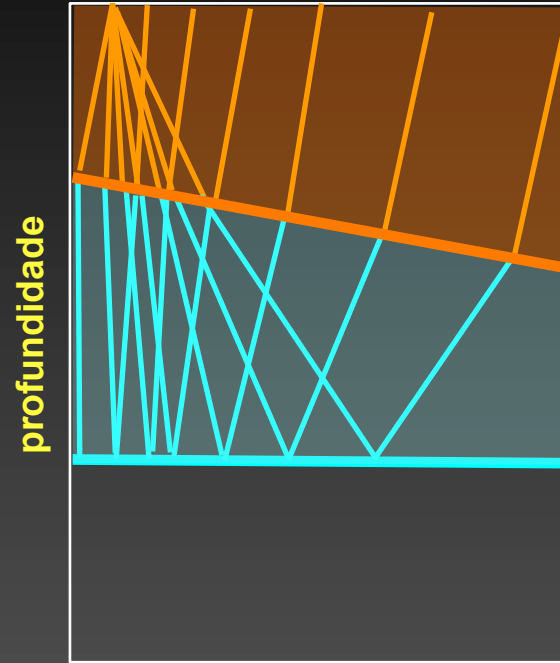


Tomografia de tempo de trânsito

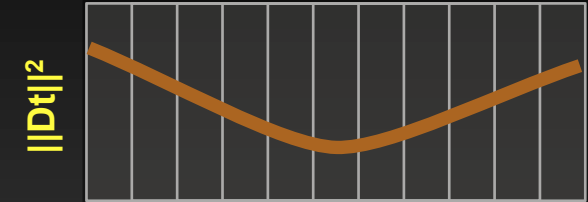
distância



distância

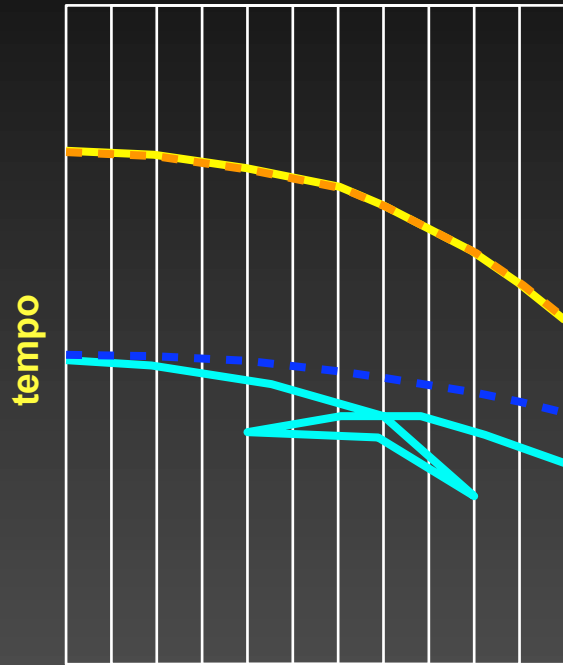


iteração

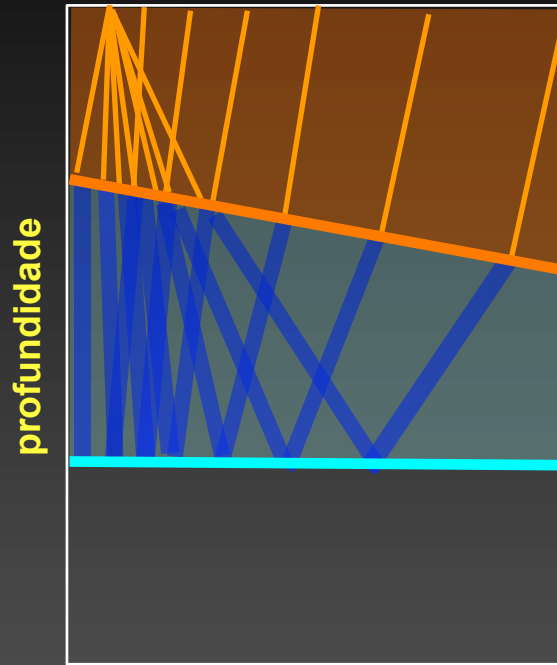


Tomografia de tempo de trânsito

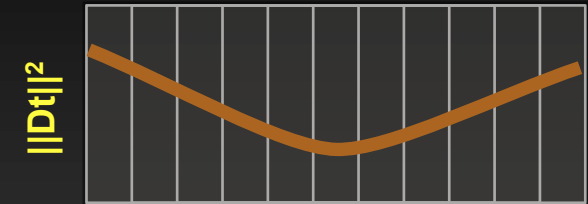
distância



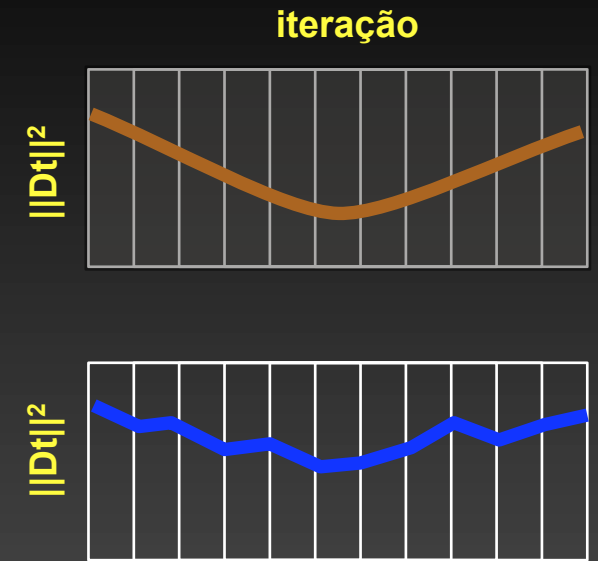
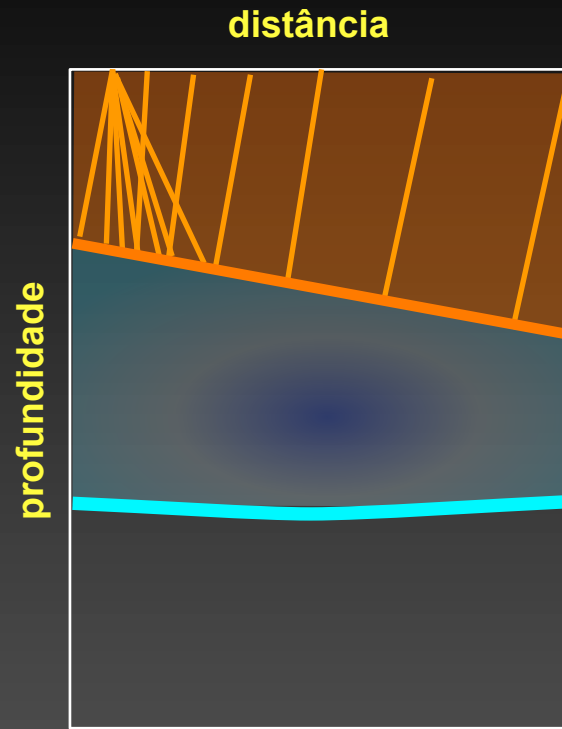
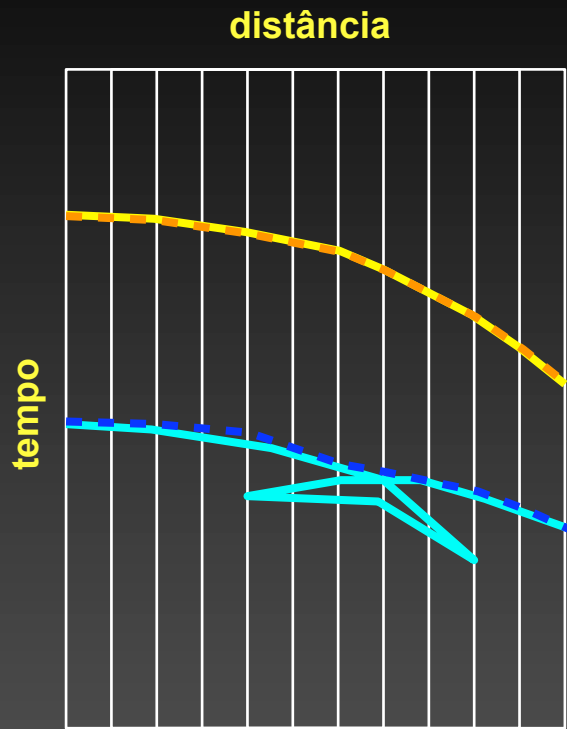
distância



iteração

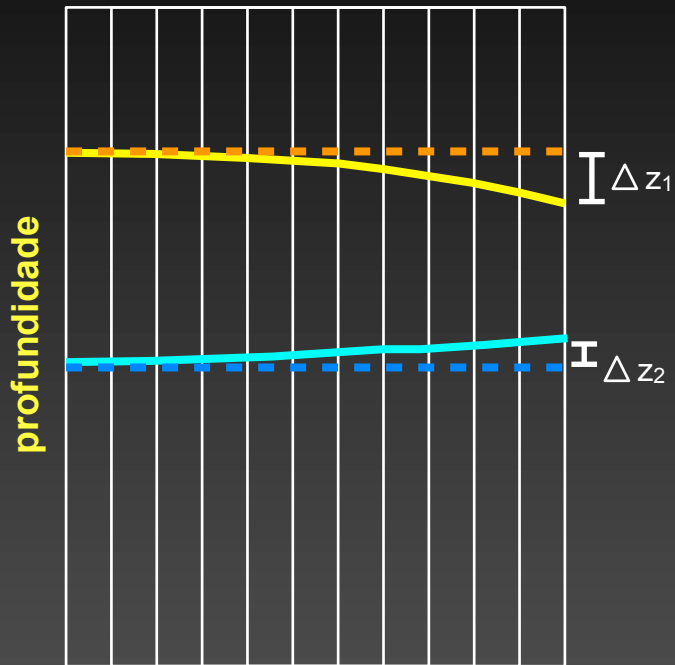


Tomografia de tempo de trânsito

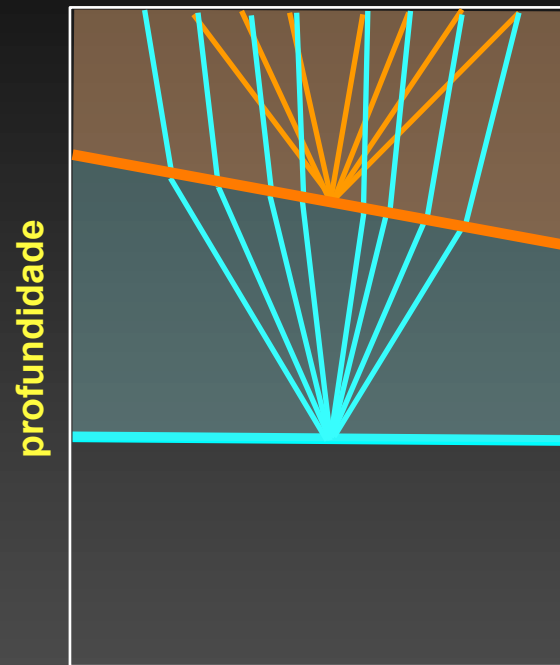


Domínio	dados	imagem
Física		
raios	Tomografia de tempo de trânsito	AVM por tomografia de raios
ondas	Inversão da forma de onda	AVM com tomografia de campos de ondas

offset



distância

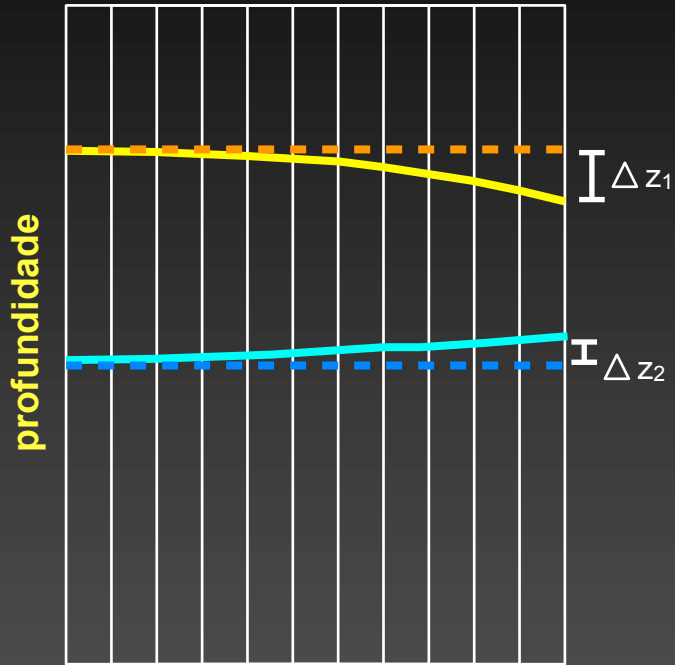


iteração

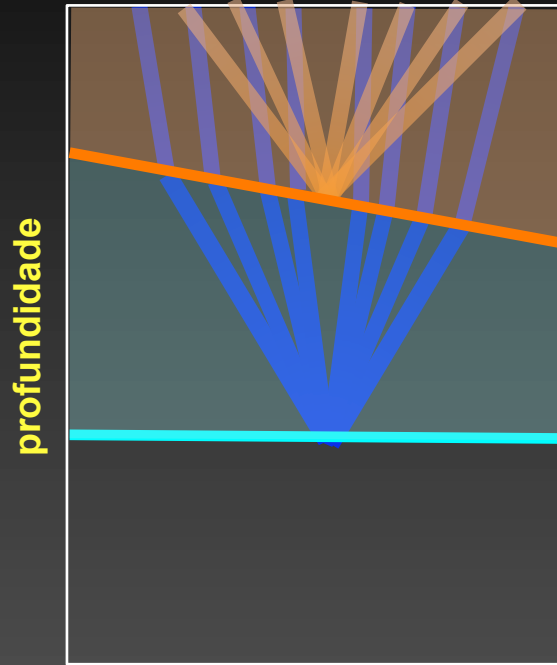


$$\Delta t(\Delta z) = \mathbf{T} \Delta s \quad \Delta s = \mathbf{T}' \Delta t(\Delta z)$$

offset



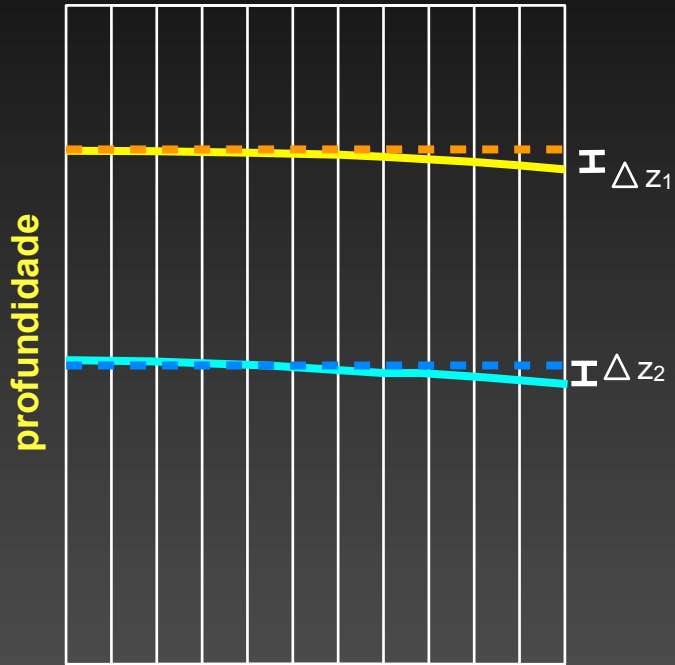
distância



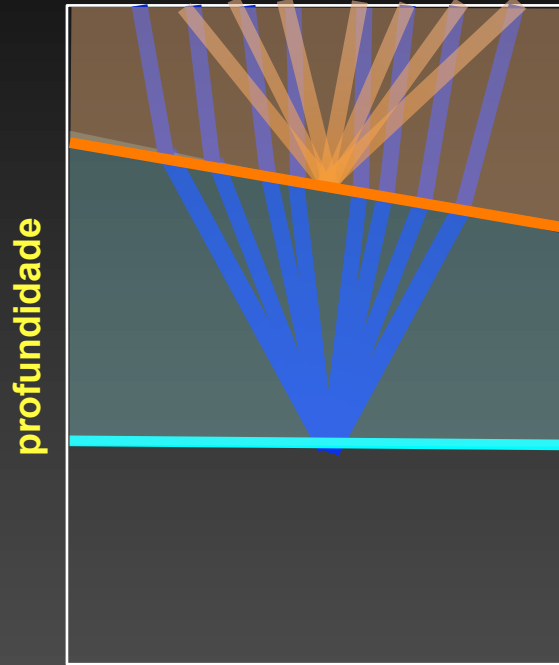
iteração



offset



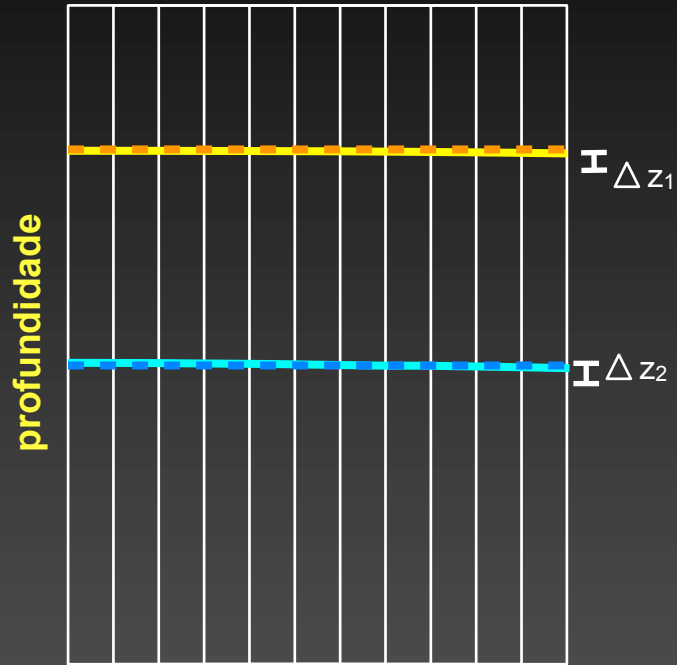
distância



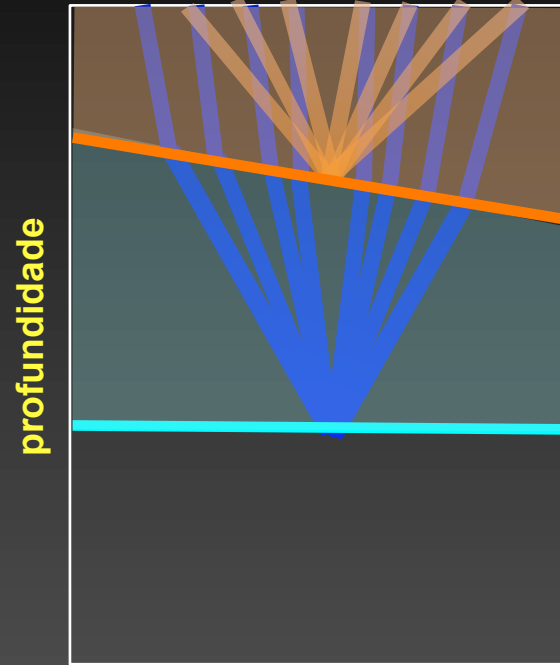
iteração



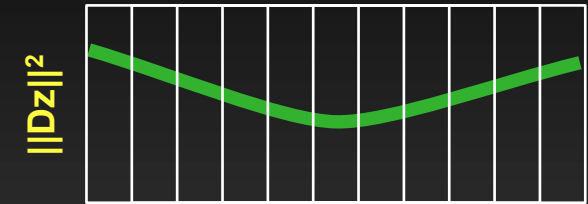
offset



distância

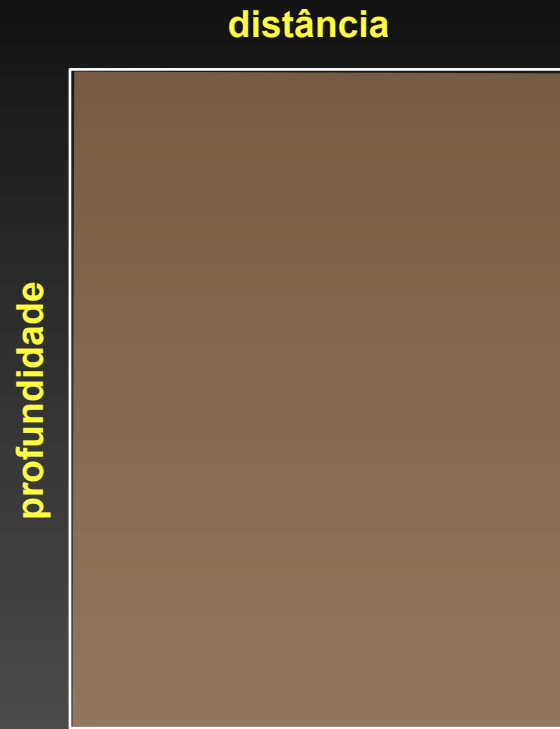
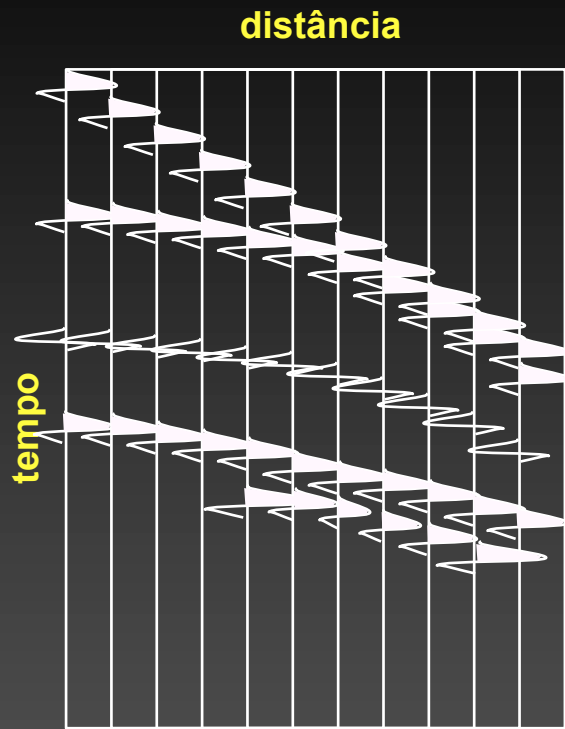


iteração

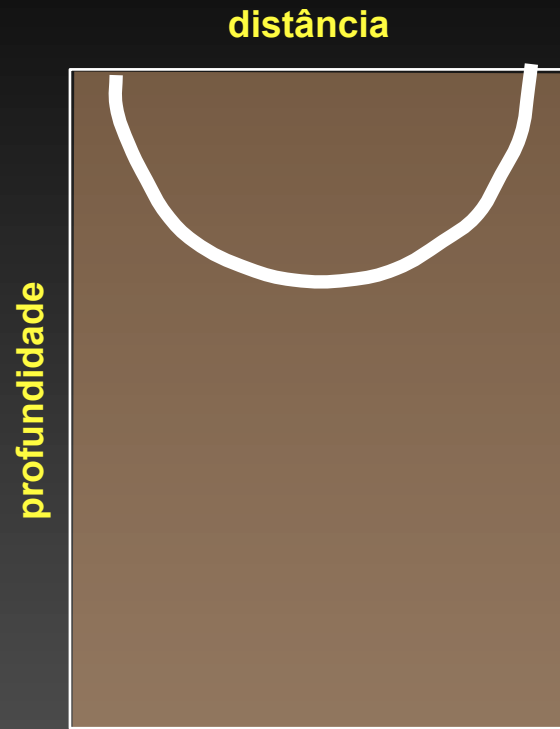
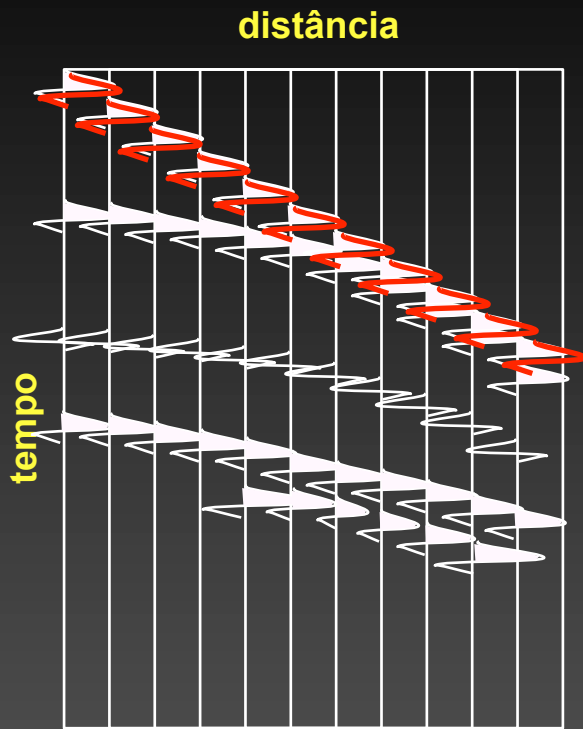


Domínio	dados	imagem
Física		
raios	Tomografia de tempo de trânsito	AVM por tomografia de raios
ondas	Inversão da forma de onda	AVM com tomografia de campos de ondas

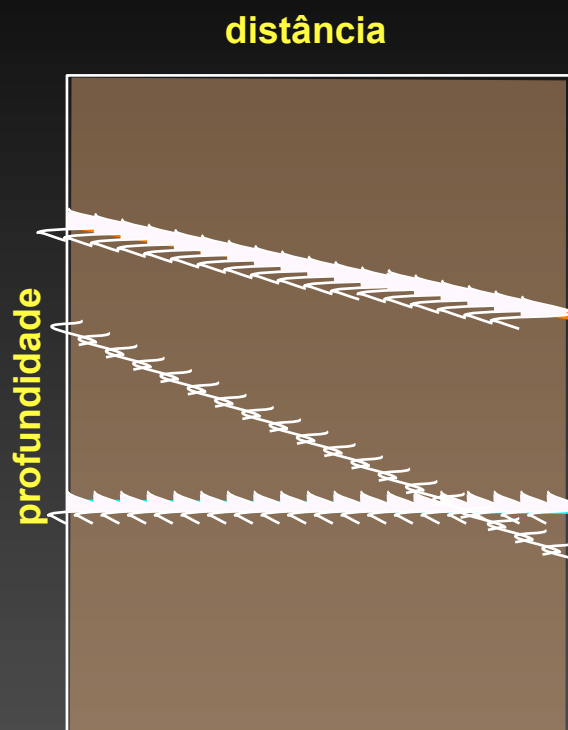
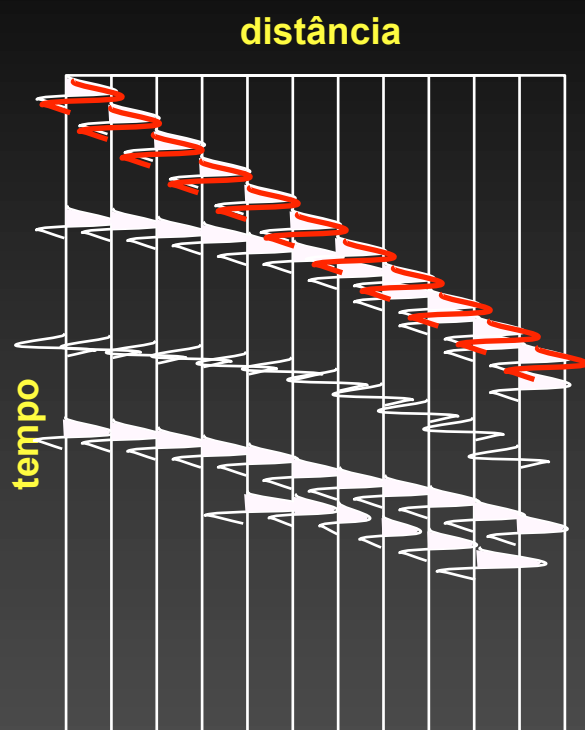
Inversão da forma de onda (FWI)



$$J(s) = \|F[s] - d_{obs}\|_n$$

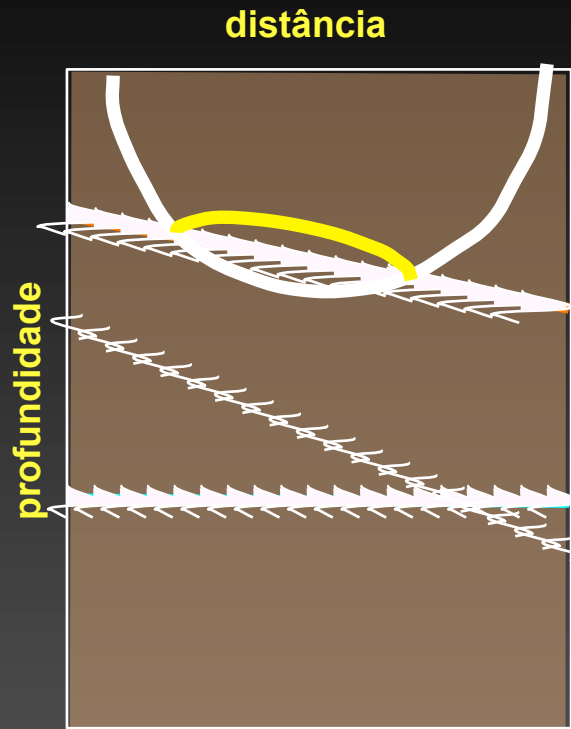
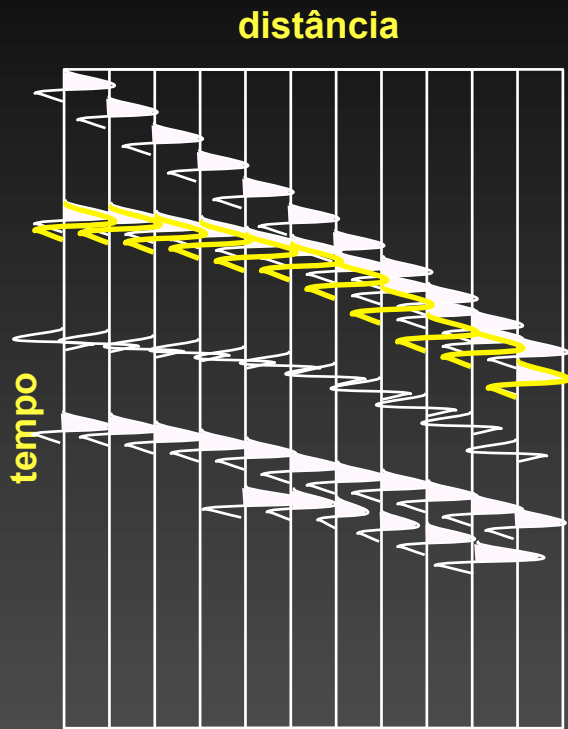


$$J(s) = \|F[s] - d_{obs}\|_n$$

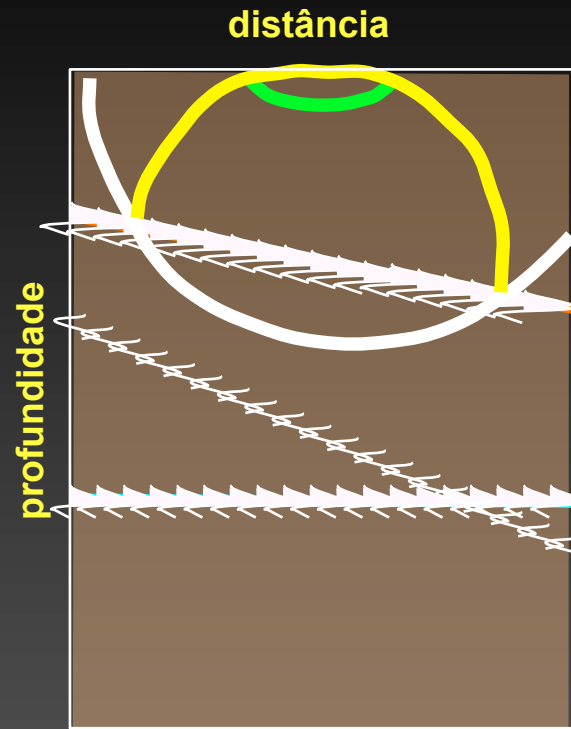
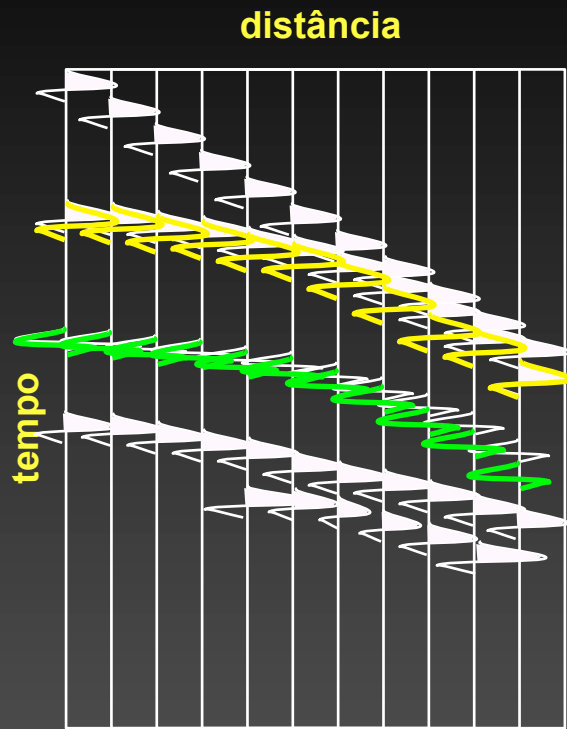


$$J(s) = \|F[s] - d_{obs}\|_n$$

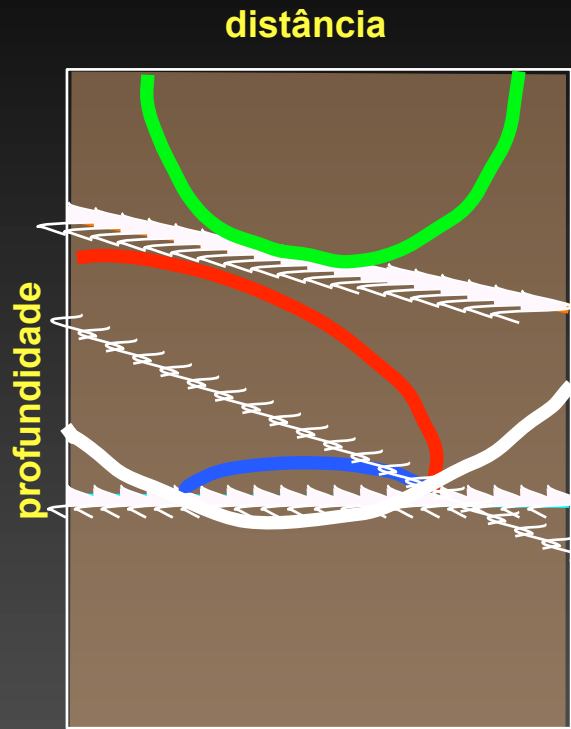
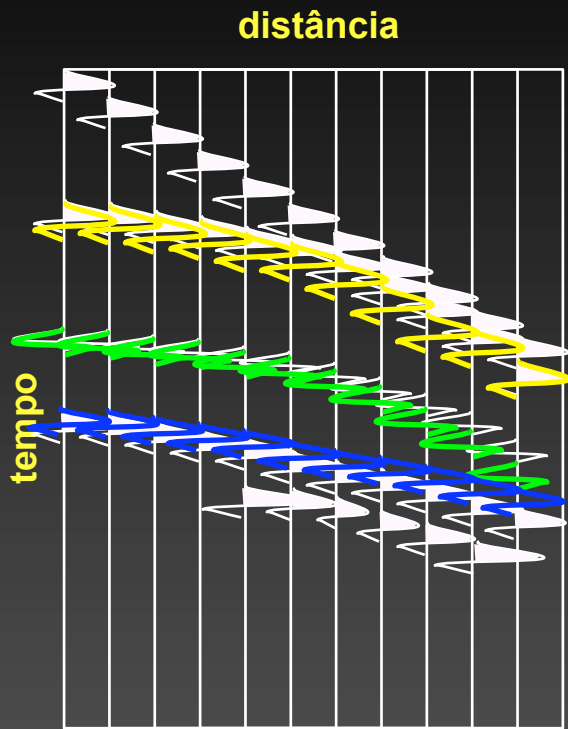
$$\nabla J(s) = F^* (d_{calc} - d_{obs})$$



$$J(s) = \|F[s] - d_{obs}\|_n$$

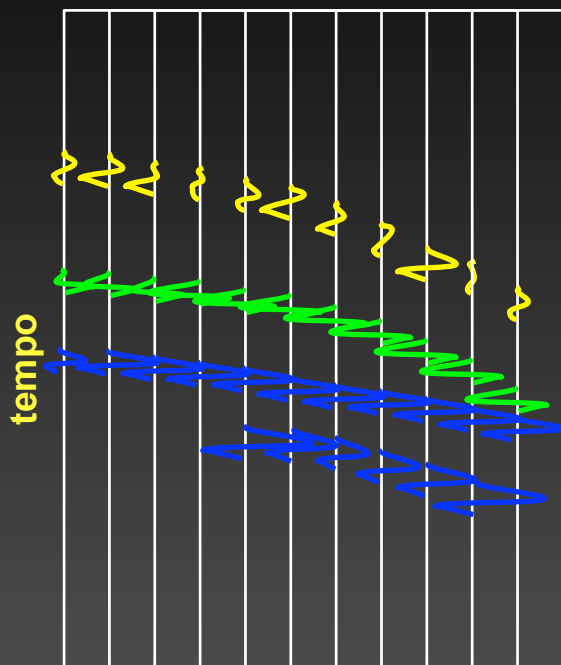


$$J(s) = \| \mathbf{F}[s] - d_{obs} \|_n$$

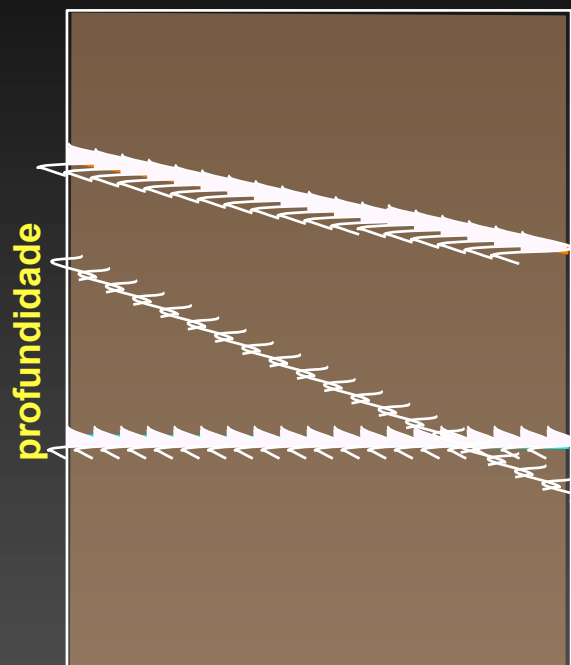


$$J(s) = \|F[s] - d_{obs}\|_n$$

distância



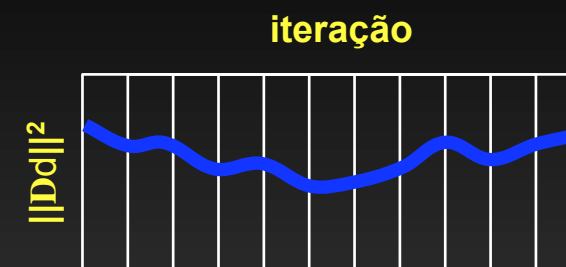
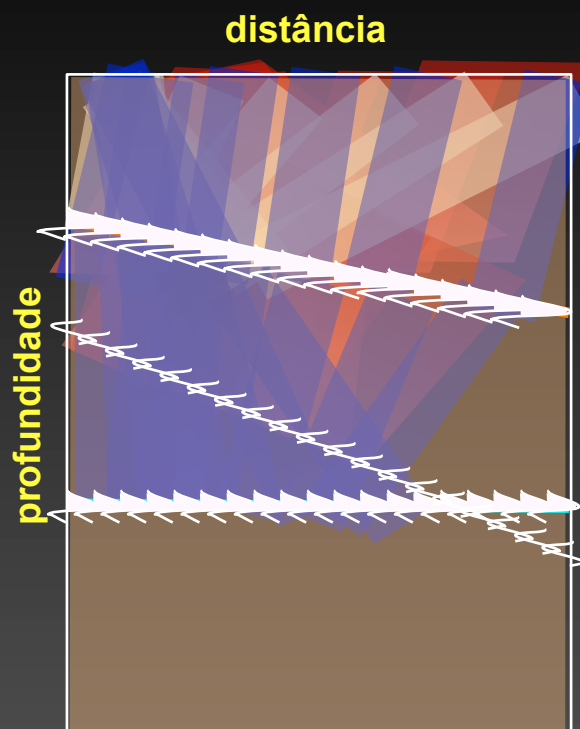
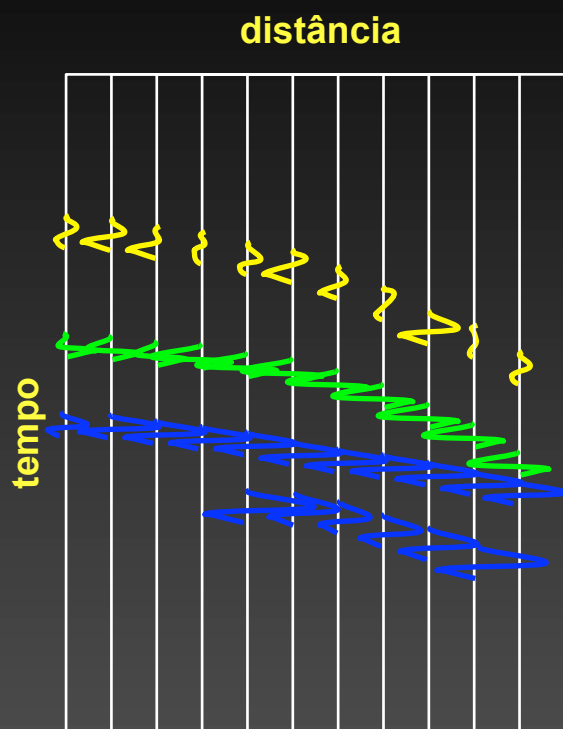
distância



iteração



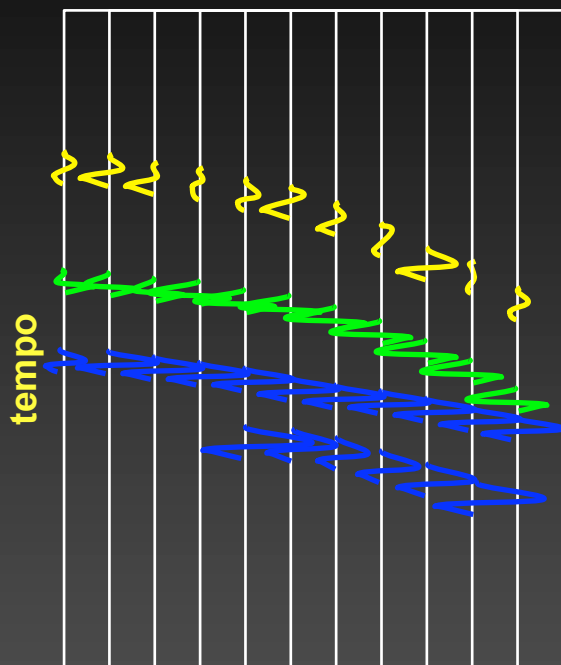
$$J(s) = \|\mathbf{F}[s] - d_{obs}\|_n$$



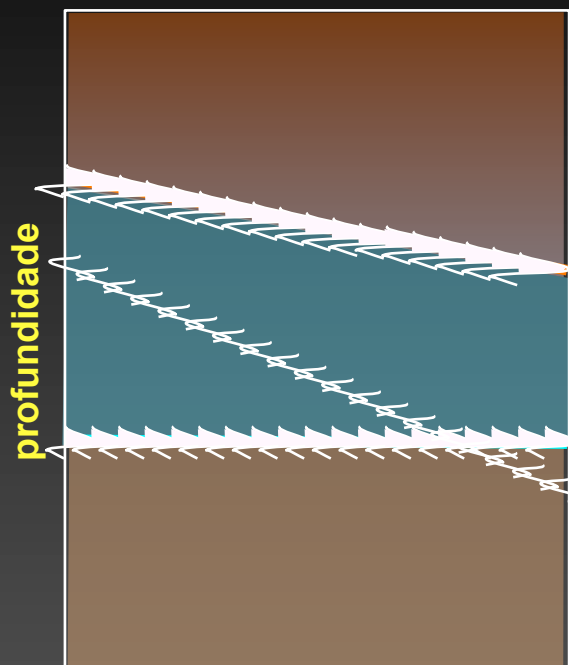
$$J(s) = \| \mathbf{F}[s] - d_{obs} \|_n$$

$$\nabla J(s) = \mathbf{F}^* (d_{calc} - d_{obs})$$

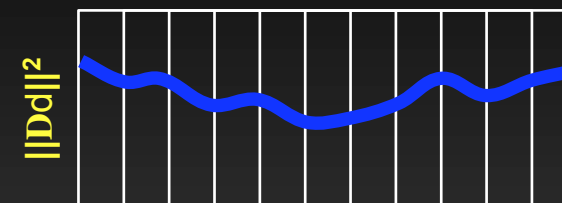
distância



distância



iteração



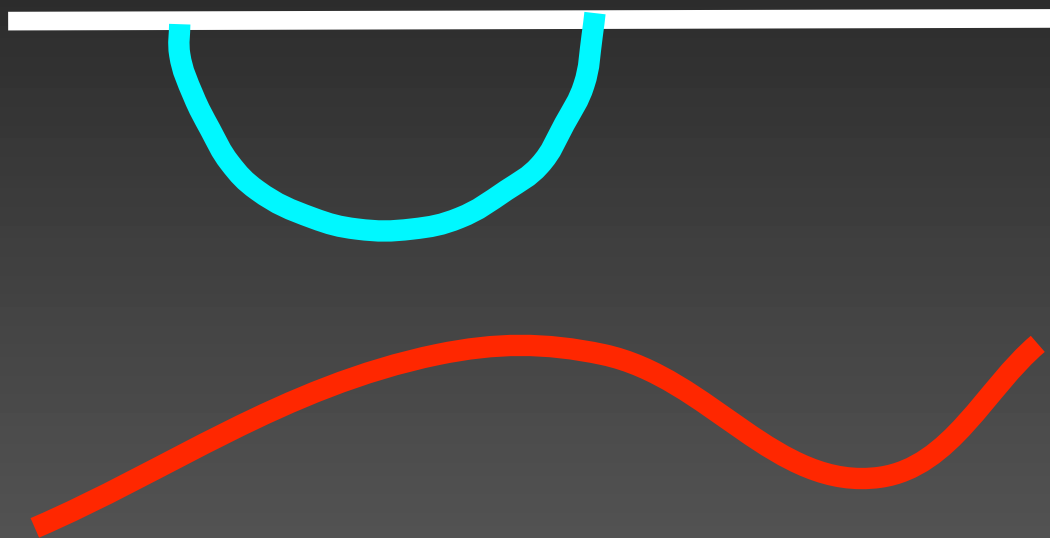
$$J(s) = \| \mathbf{F}[s] - d_{obs} \|_n$$

$$\nabla J(s) = \mathbf{F}^* (d_{calc} - d_{obs})$$

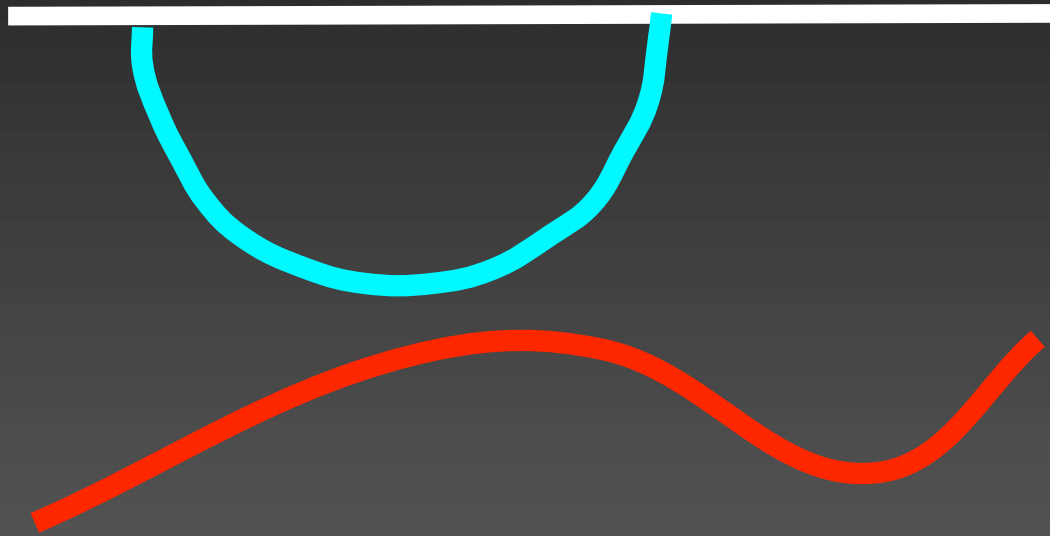
Domínio	dados	imagem
Física		
raios	Tomografia de tempo de trânsito	AVM por tomografia de raios
ondas	Inversão da forma de onda	AVM com tomografia de campos de ondas

- **Modelagem**
- **Migração**
- **AVM com tomografia de campos de onda (AVMTO)**
- **Custo e flexibilidade**
- **Exemplos**

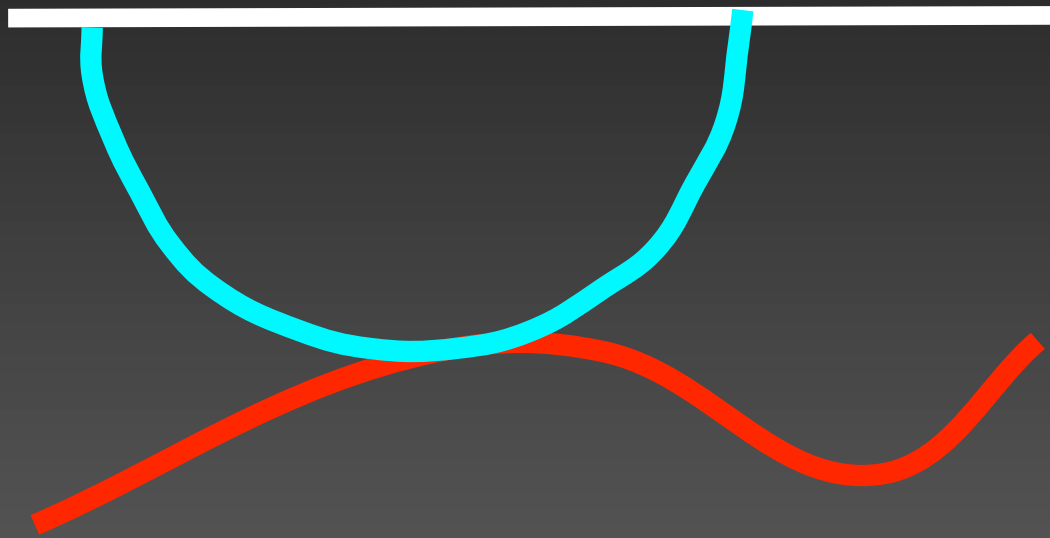
$$L_m = d$$



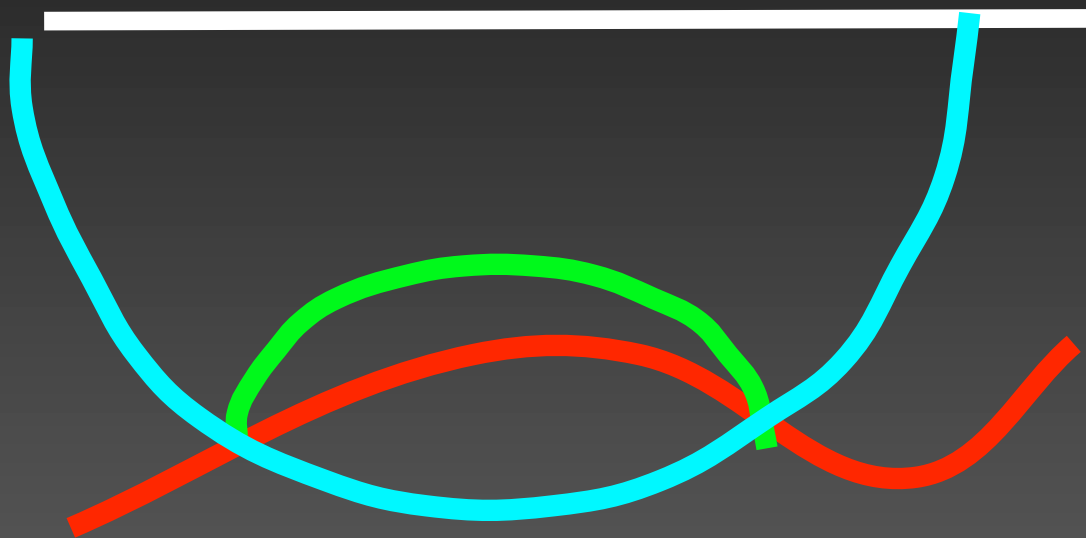
$$L_m = d$$



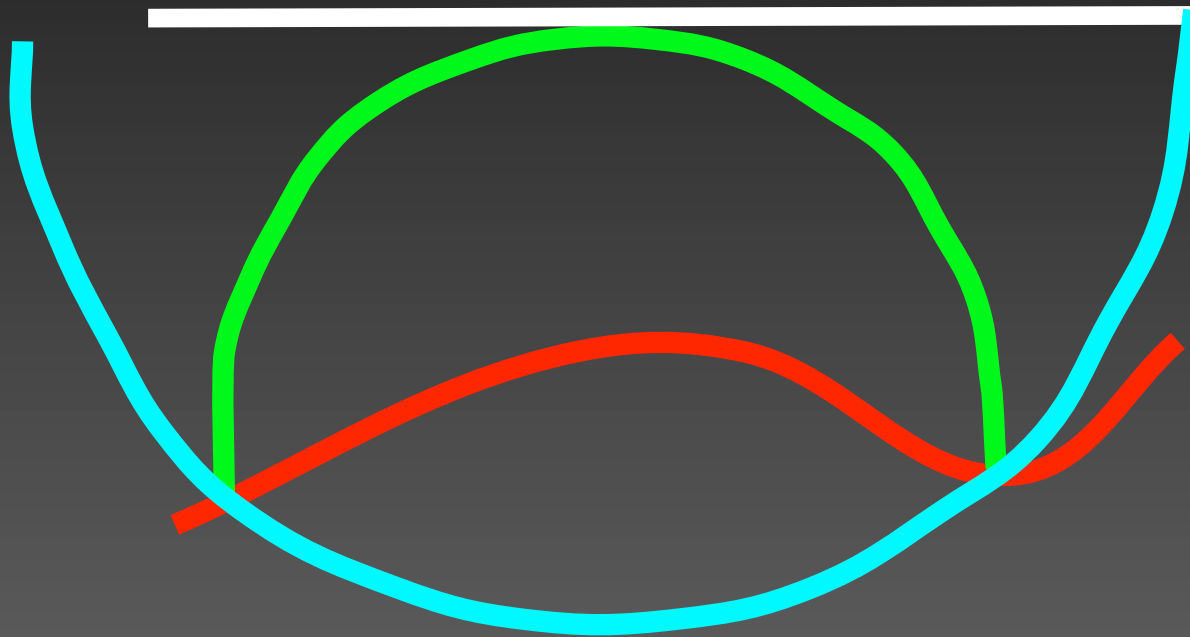
$$L_m = d$$



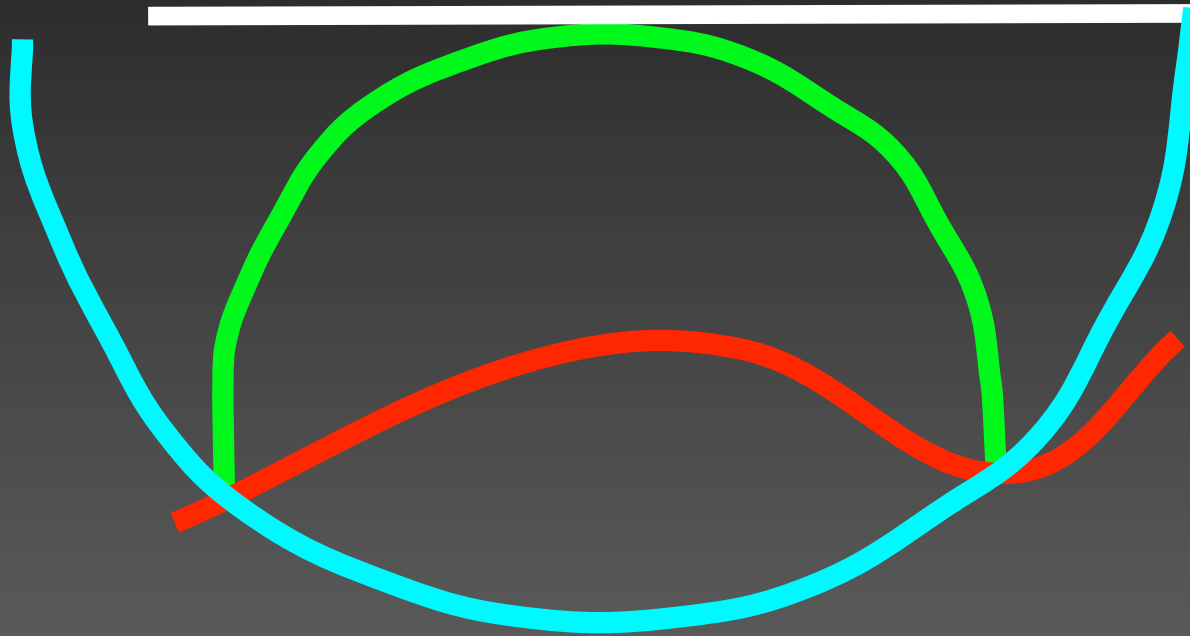
$$L_m = d$$



$$L_m = d$$



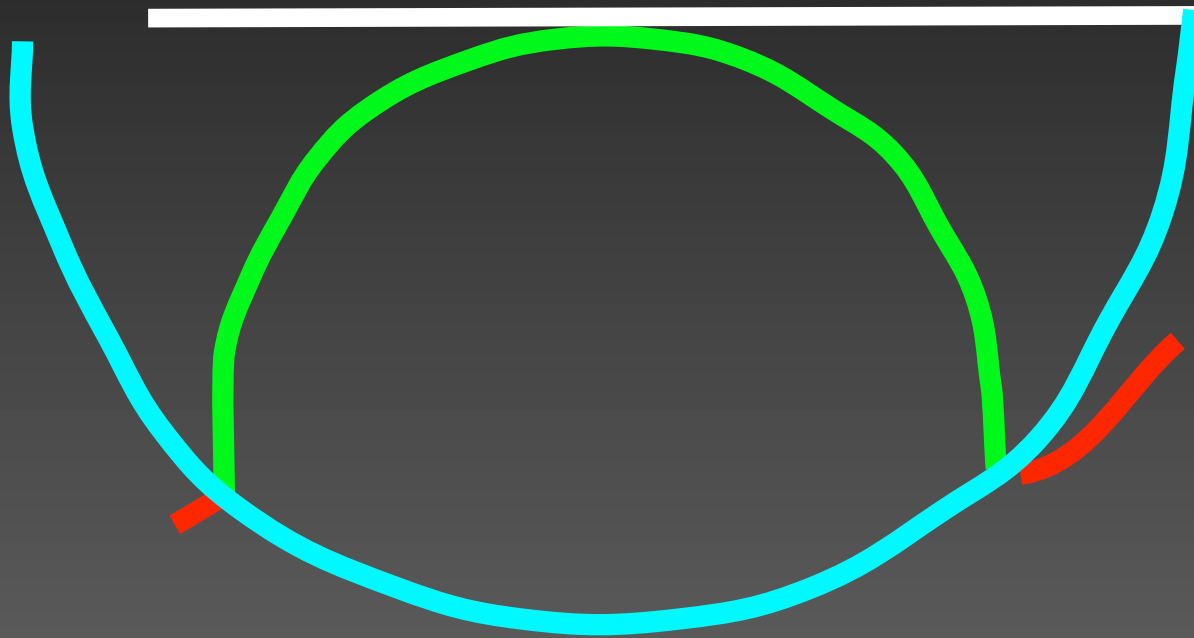
$$\mathbf{L}_m = d$$



$$\mathbf{L} \Leftrightarrow G(x_s, x) \otimes G(x, x_r)$$

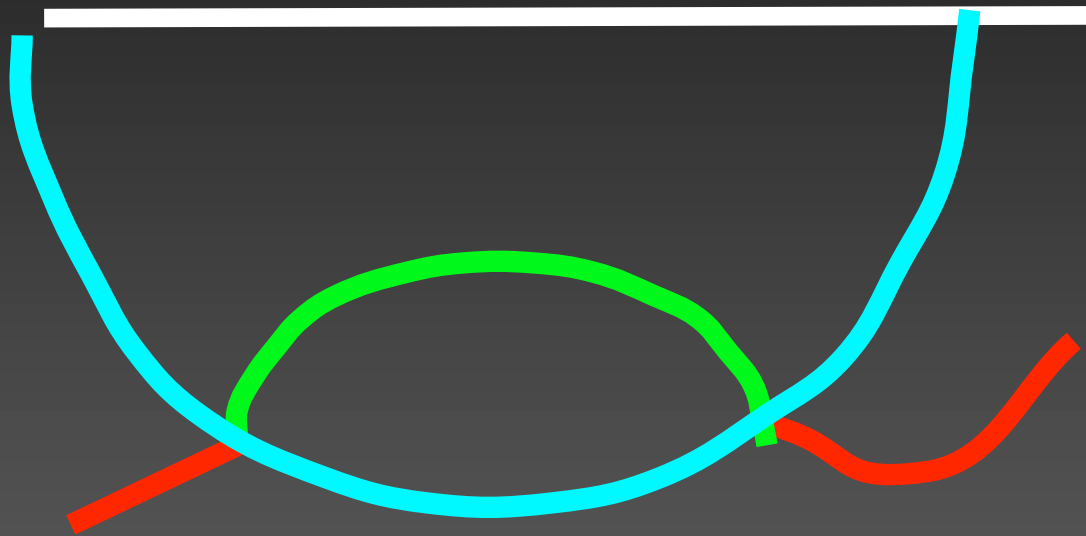
- Modelagem
- **Migração**
- AVM com tomografia de campos de onda (AVMTO)
- Custo e flexibilidade
- Exemplos

$$I = \mathbf{L}^* d$$

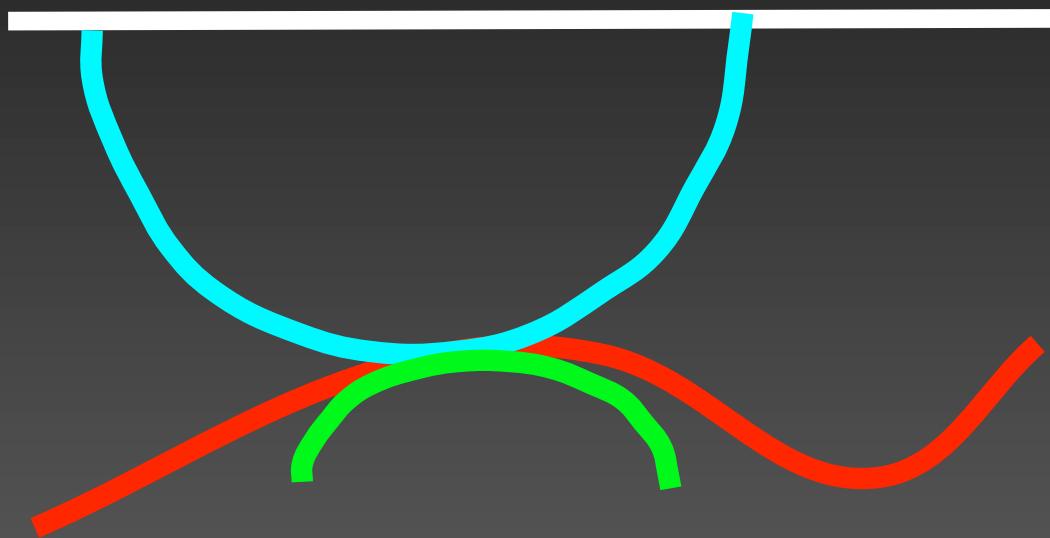


$$\mathbf{L}^* \Leftrightarrow G^*(x, x_r) \otimes G^*(x_s, x)$$

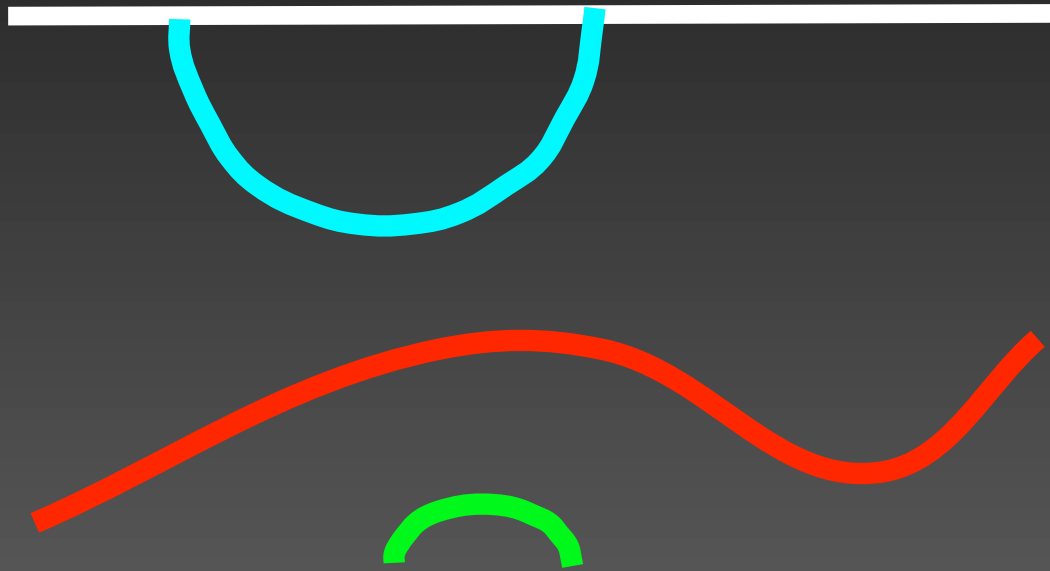
$$I = \mathbf{L}^* d$$



$$I = \mathbf{L}^* d$$



$$I = \mathbf{L}^* d$$



$$\mathbf{L}^* \Leftrightarrow G^*(x, x_r) \otimes G^*(x_s, x)$$

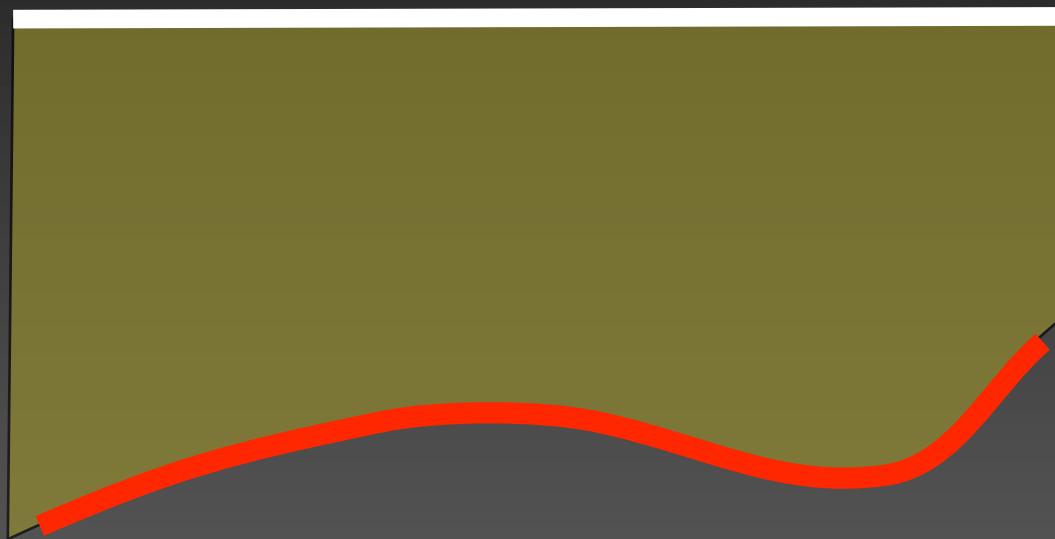
$$I = D^*U$$


$$D = G(x_s, x)$$

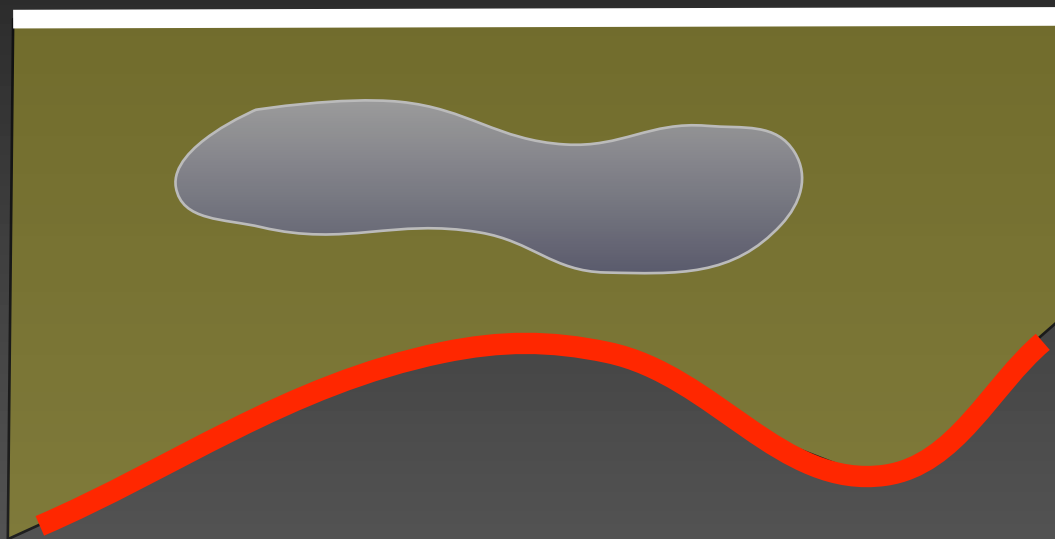
$$U = G^*(x, x_r)d$$

- Modelagem
- Migração
- **AVM com tomografia de campos de onda (AVMTO)**
- Custo e flexibilidade
- Exemplos

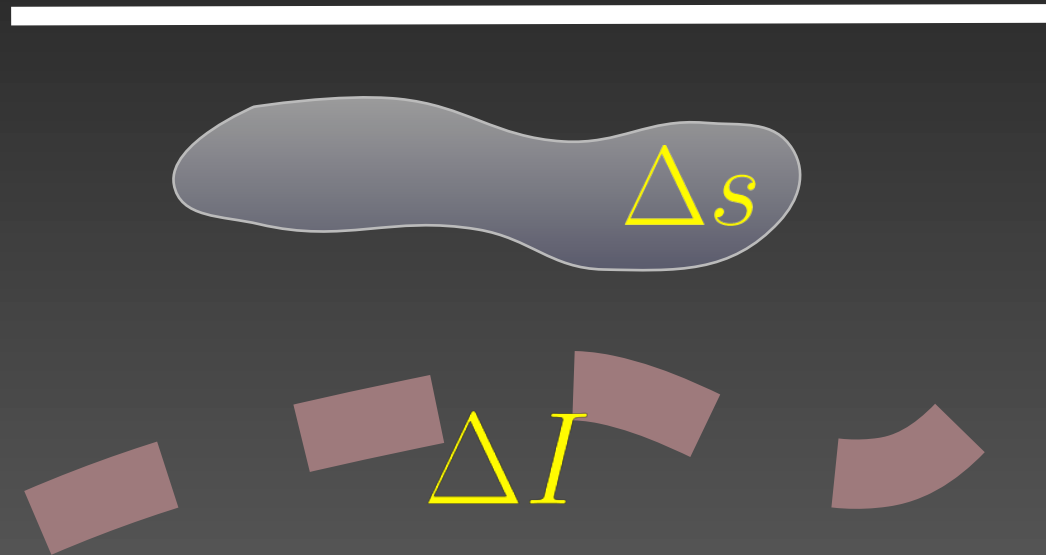
$$I_0 = D_0^* U_0$$



$$I = D^*U$$

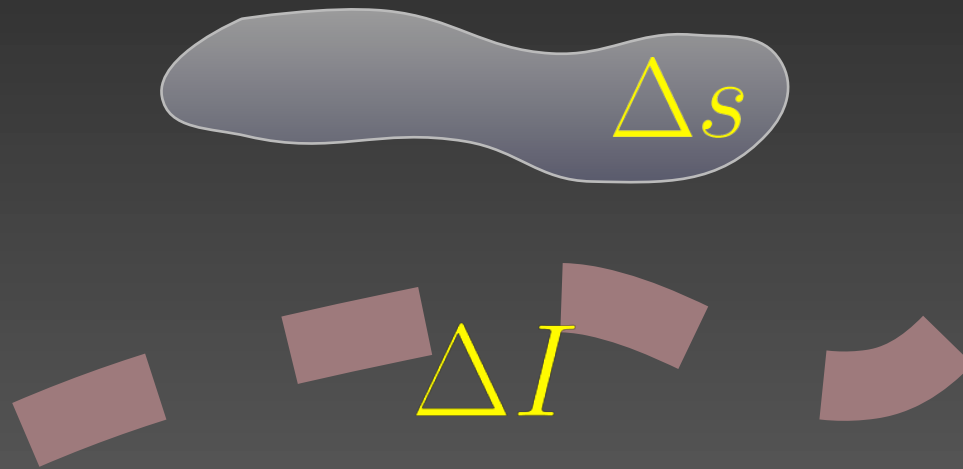


$$I = I_0 + \Delta I$$

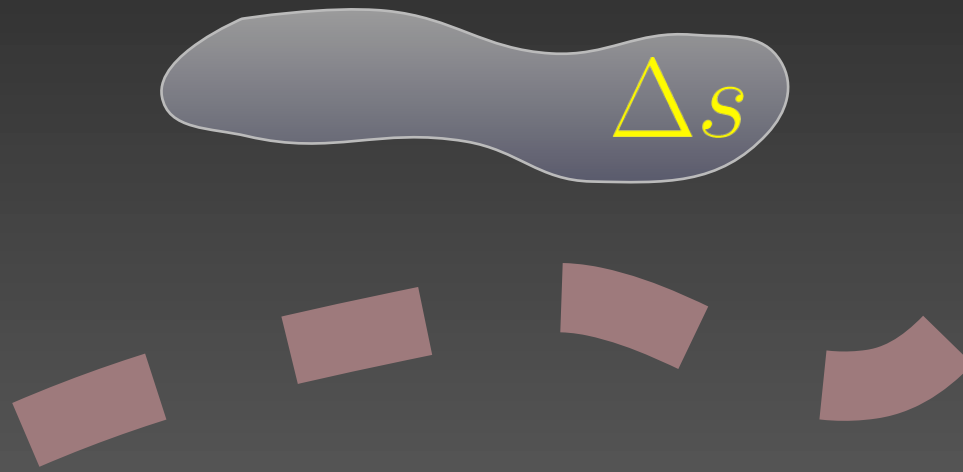


Relação entre perturbações ?

$$\Delta I \longleftrightarrow \Delta S$$

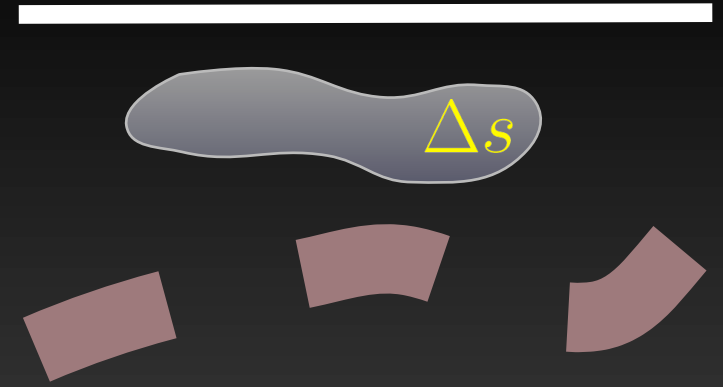


$$I(s) \approx I_0 + \left. \frac{\partial I}{\partial s} \right|_{s=s_0} \Delta s$$



$$\Delta I = \mathbf{T} \Delta s$$

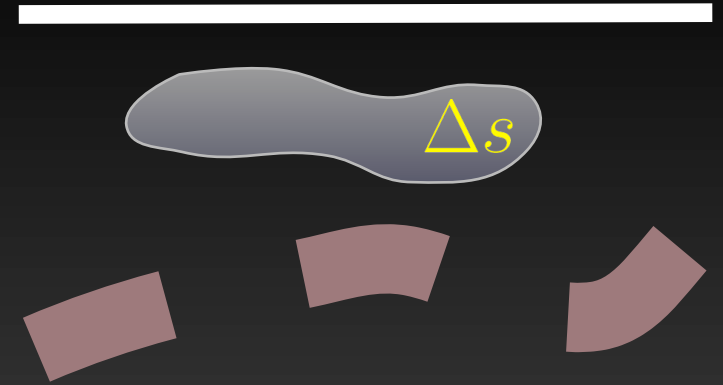
$$I = D^*U$$
$$\Delta I = \Delta D^*U + D^*\Delta U$$



$$\Delta I = \Delta D^* U + D^* \Delta U$$

$$P_{z+\Delta z} = P_z e^{ik_z \Delta z}$$

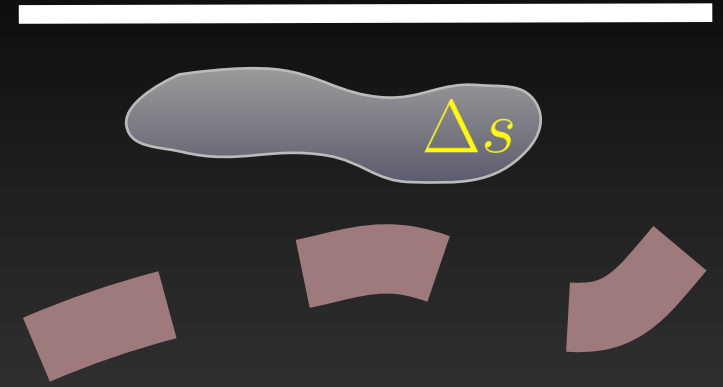
$$k_z = \sqrt{(\omega s_0)^2 - |\mathbf{k}|^2}$$



$$\Delta I = \Delta D^* U + D^* \Delta U$$

$$P_{z+\Delta z} = P_z e^{ik_z \Delta z}$$

$$\Delta P_{z+\Delta z} = i \frac{dk_z}{ds} P_z e^{ik_z \Delta z} \Delta z \Delta s$$

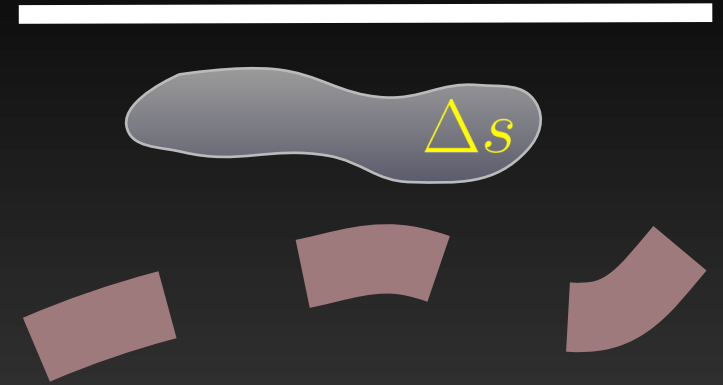


$$\Delta I = \Delta D^* U + D^* \Delta U$$

$$P_{z+\Delta z} = P_z e^{ik_z \Delta z}$$

$$\Delta P_{z+\Delta z} = i \frac{dk_z}{ds} P_z e^{ik_z \Delta z} \Delta z \Delta s$$

$$\Delta P = i \frac{dk_z}{ds} P \Delta z \Delta s \quad \frac{dk_z}{ds} = \frac{\omega^2 s}{k_z}$$



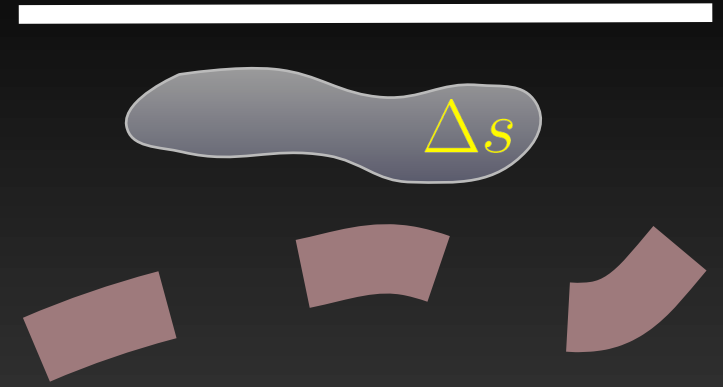
$$\Delta I = \Delta D^* U + D^* \Delta U$$

$$P_{z+\Delta z} = P_z e^{ik_z \Delta z}$$

$$\Delta P_{z+\Delta z} = i \frac{dk_z}{ds} P_z e^{ik_z \Delta z} \Delta z \Delta s$$

$$\Delta P = i \frac{dk_z}{ds} P \Delta z \Delta s$$

$$\Delta P = P_{sc} \Delta s$$



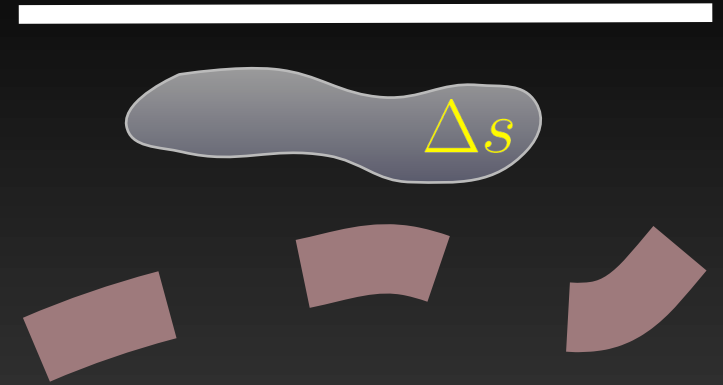
$$\Delta I = \Delta D^* U + D^* \Delta U$$

$$P_{z+\Delta z} = P_z e^{ik_z \Delta z}$$

$$\Delta P_{z+\Delta z} = i \frac{dk_z}{ds} P_z e^{ik_z \Delta z} \Delta z \Delta s$$

$$\Delta P = i \frac{dk_z}{ds} P \Delta z \Delta s \quad \Delta P = P_{sc} \Delta s$$

$$\Delta I = (D_{sc}^* U + D^* U_{sc}) \Delta s$$



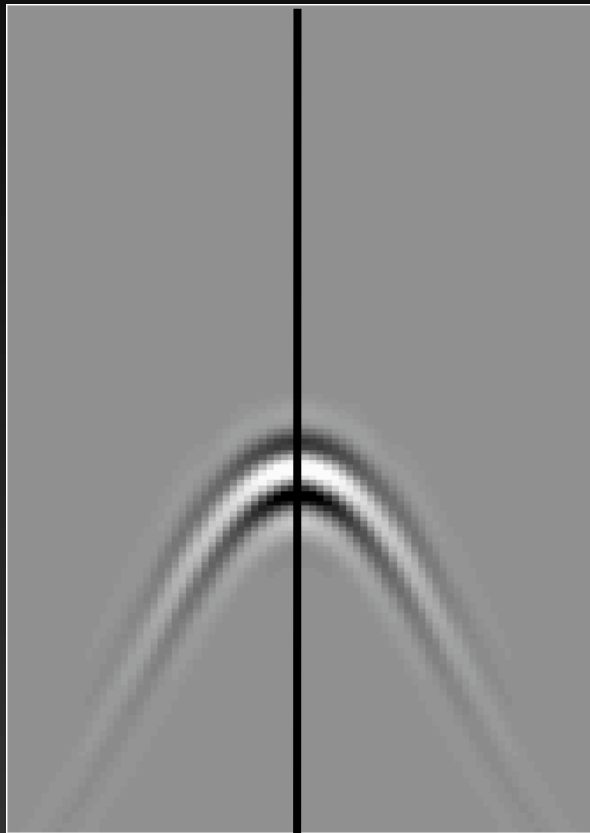
- **Como calcular o ΔI ?**
 - differential-semblance velocity analysis (DSVA)
(Shen and Symes, 2008)

**Velocidade
baixa**

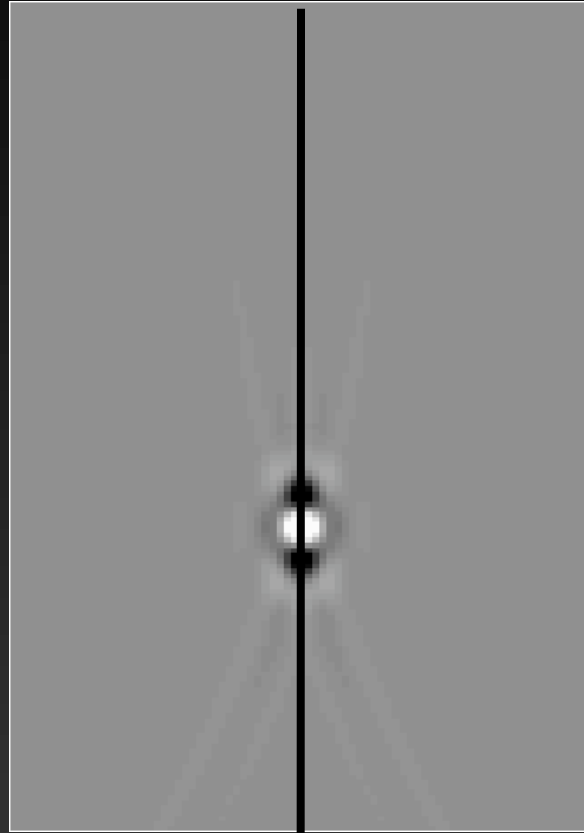
**Velocidade
correta**

**Velocidade
alta**

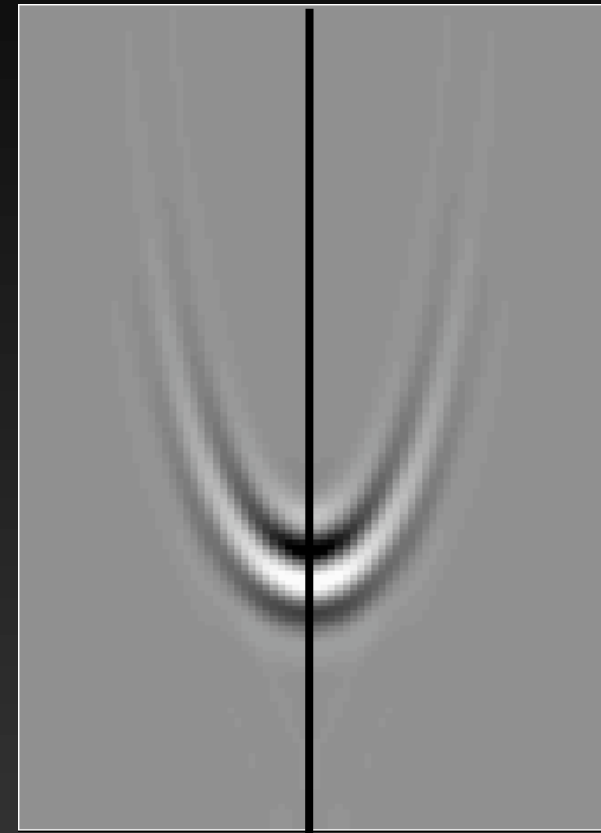
profundidade



offset



offset



offset

BR

PETROBRAS

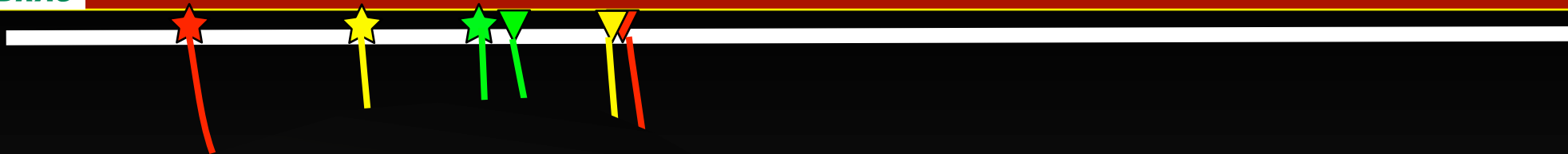
CIG (offset subsuperfície)



BR

PETROBRAS

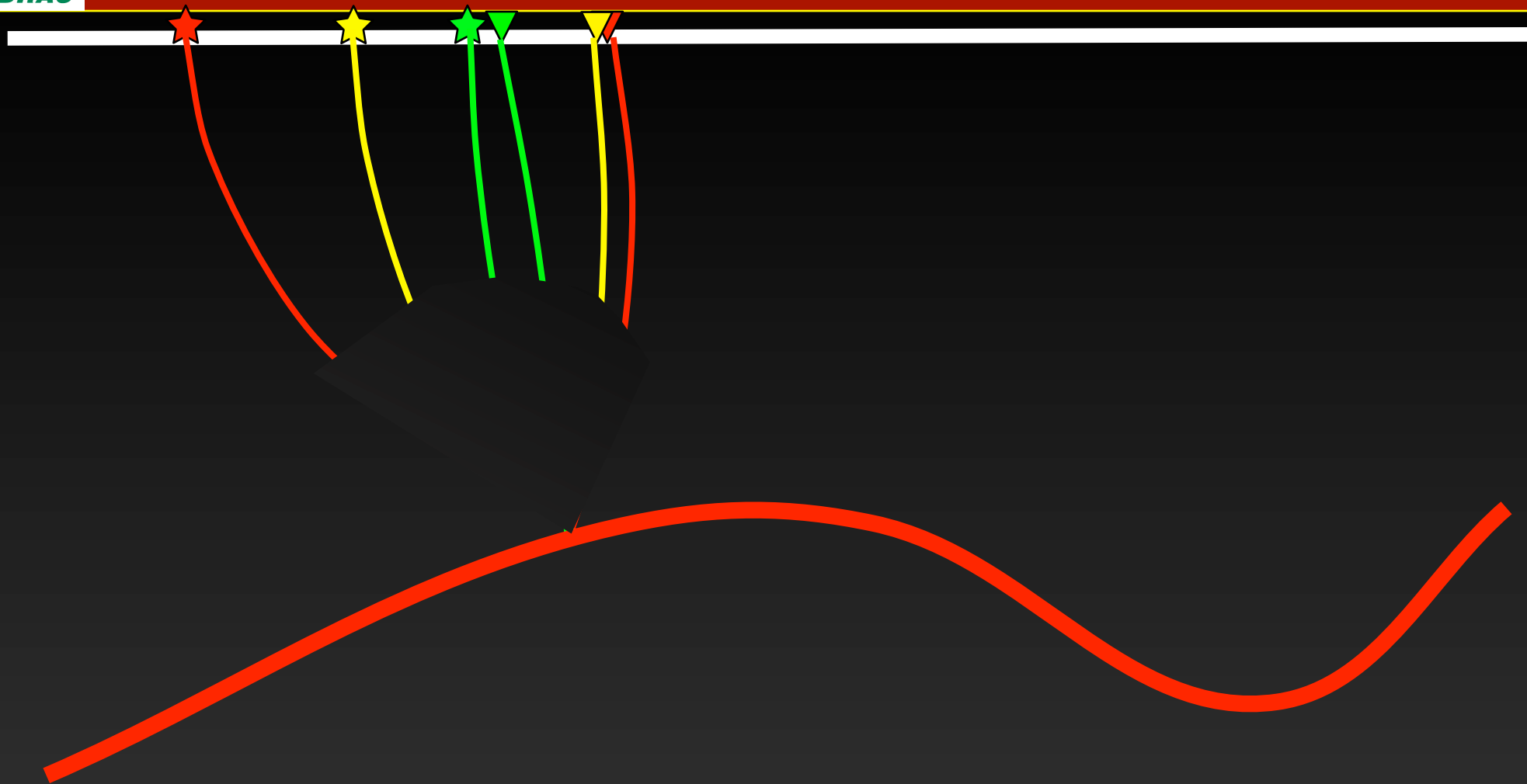
CIG (offset subsuperfície)

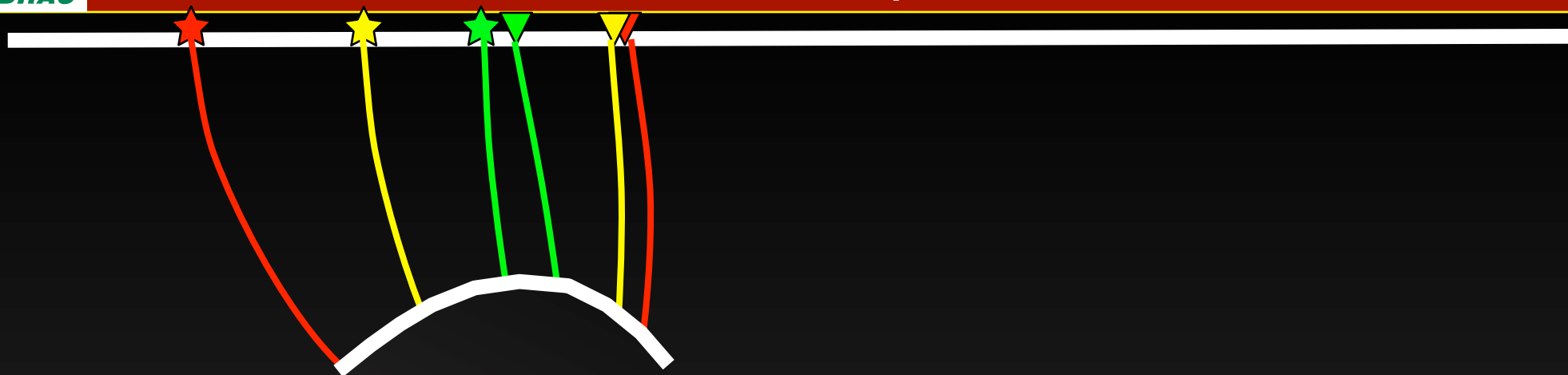


BR

PETROBRAS

CIG (offset subsuperfície)

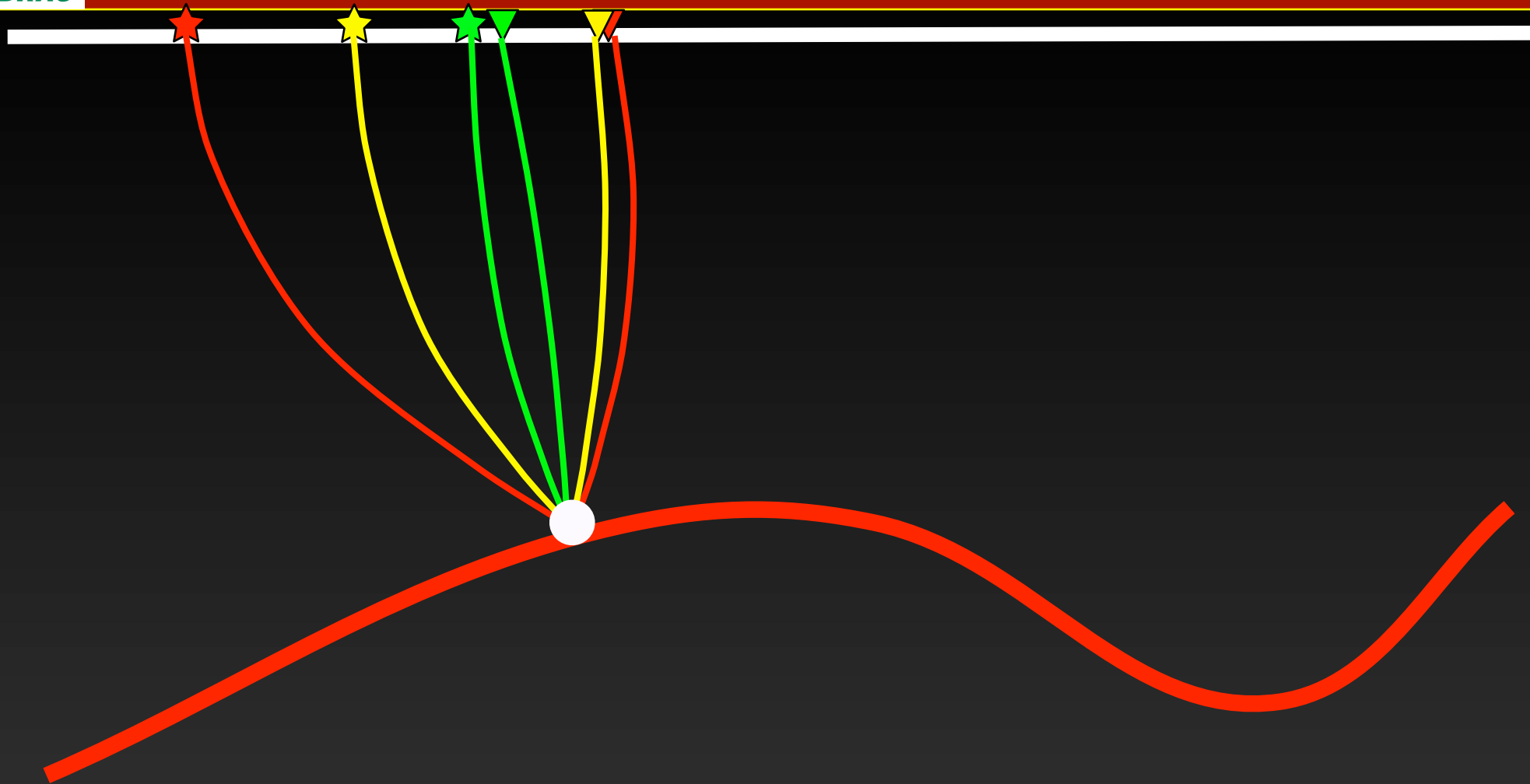




**Velocidade
baixa**

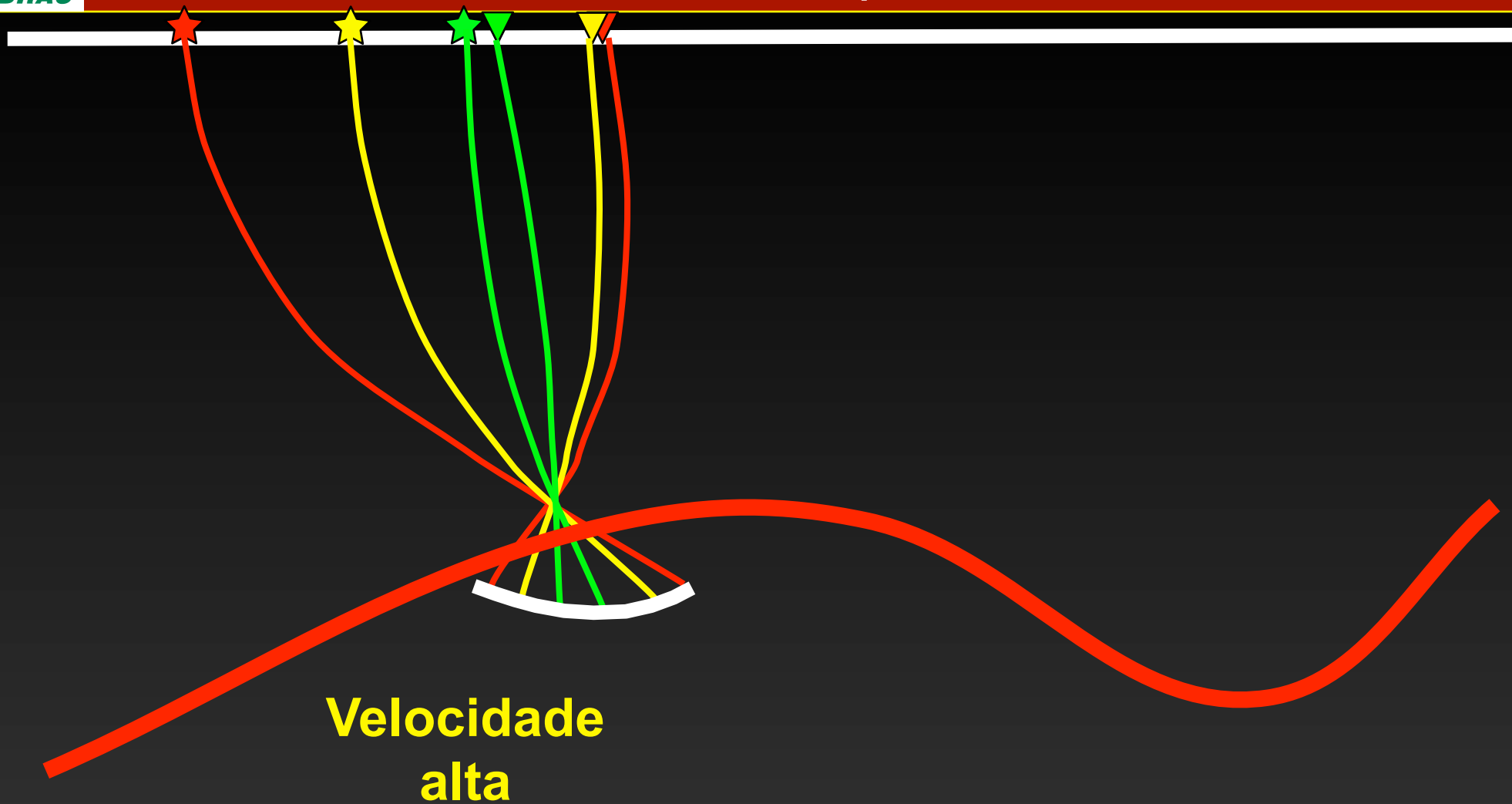


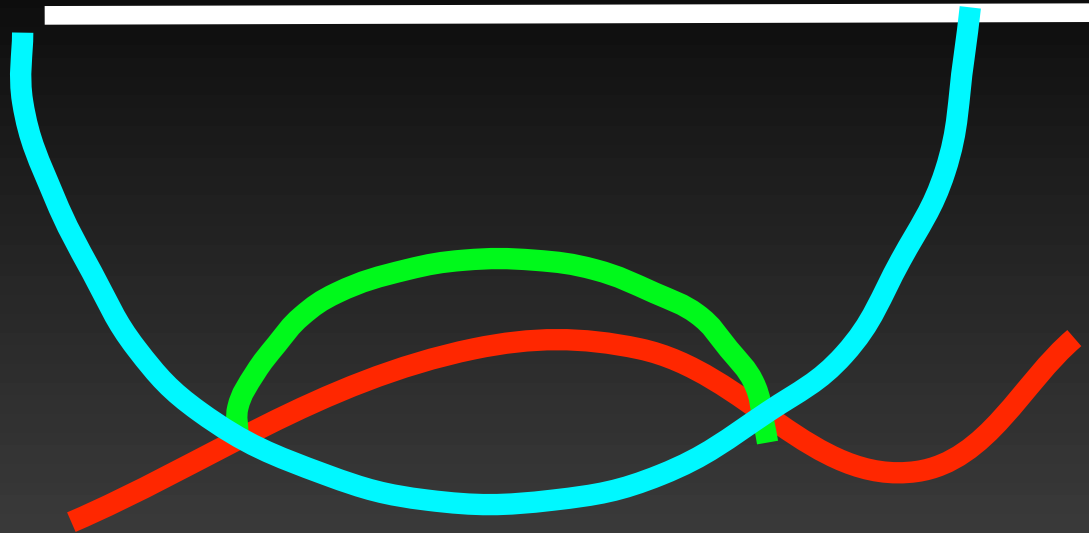
CIG (offset subsuperfície)

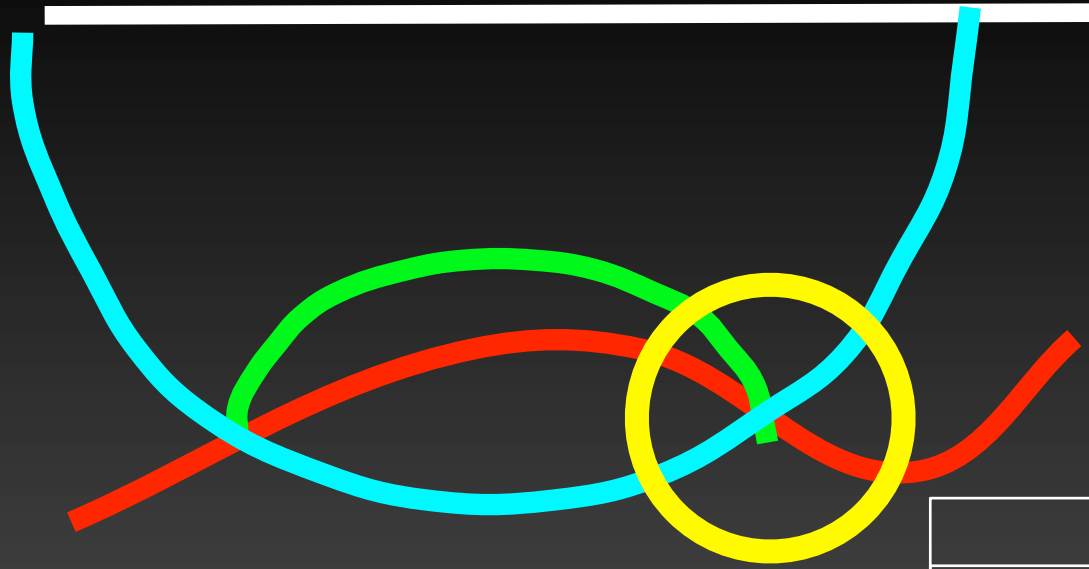


**Velocidade
correta**

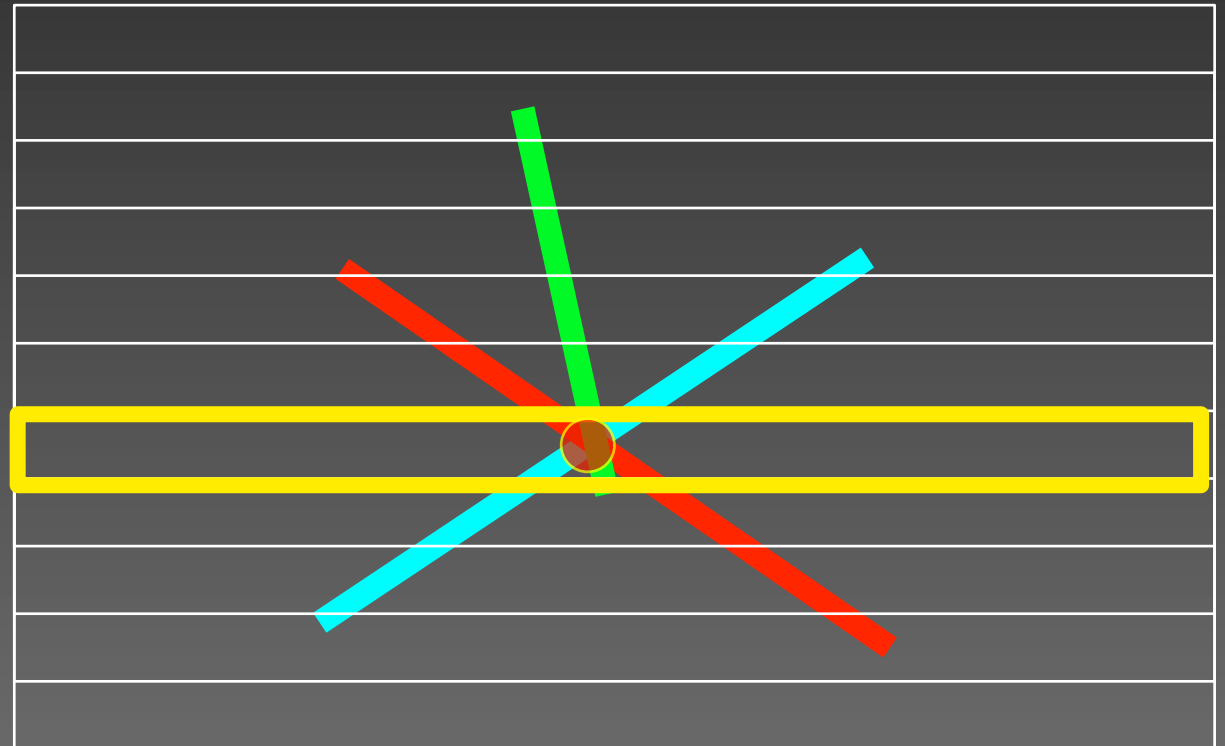
CIG (offset subsuperfície)



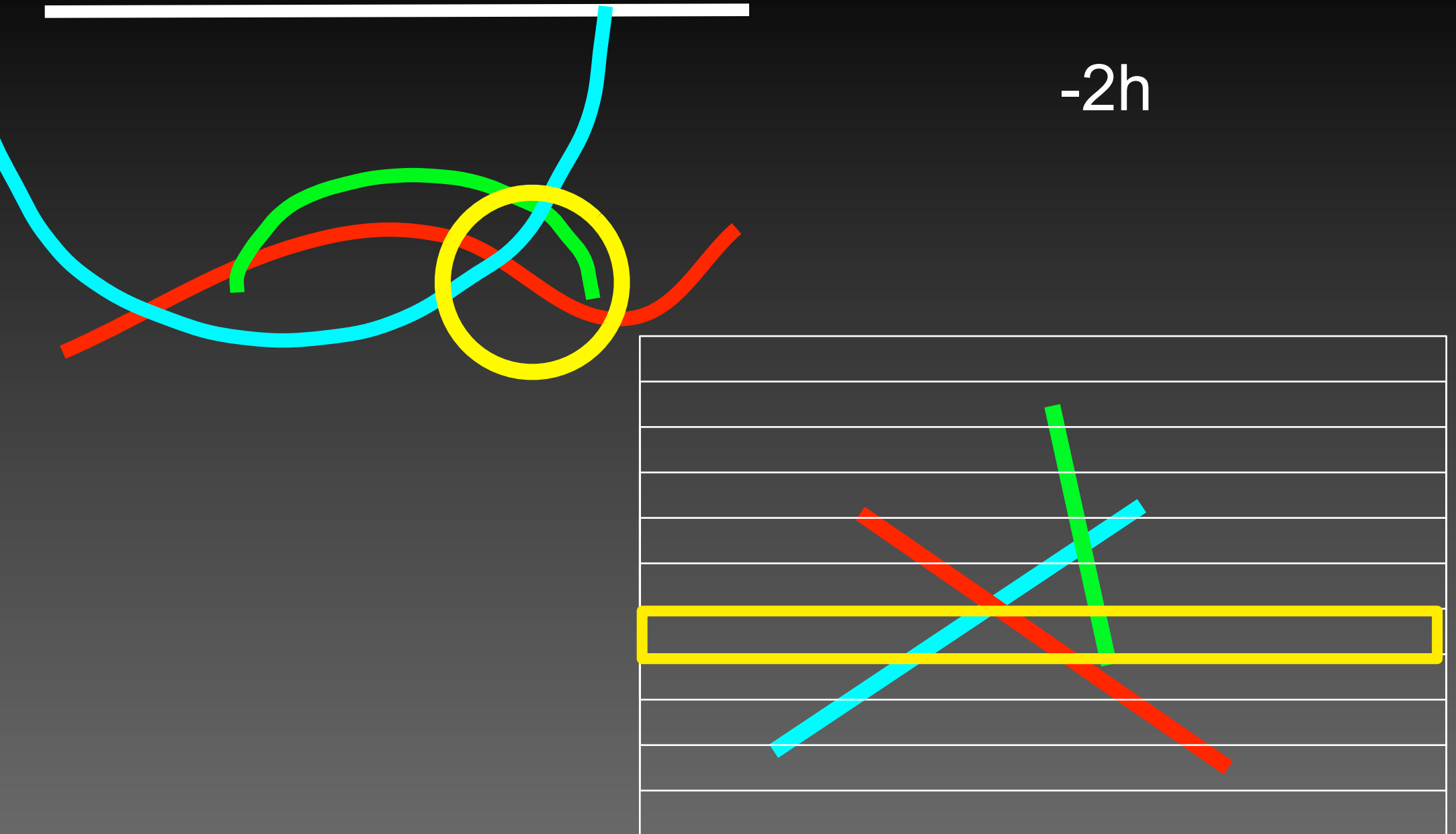


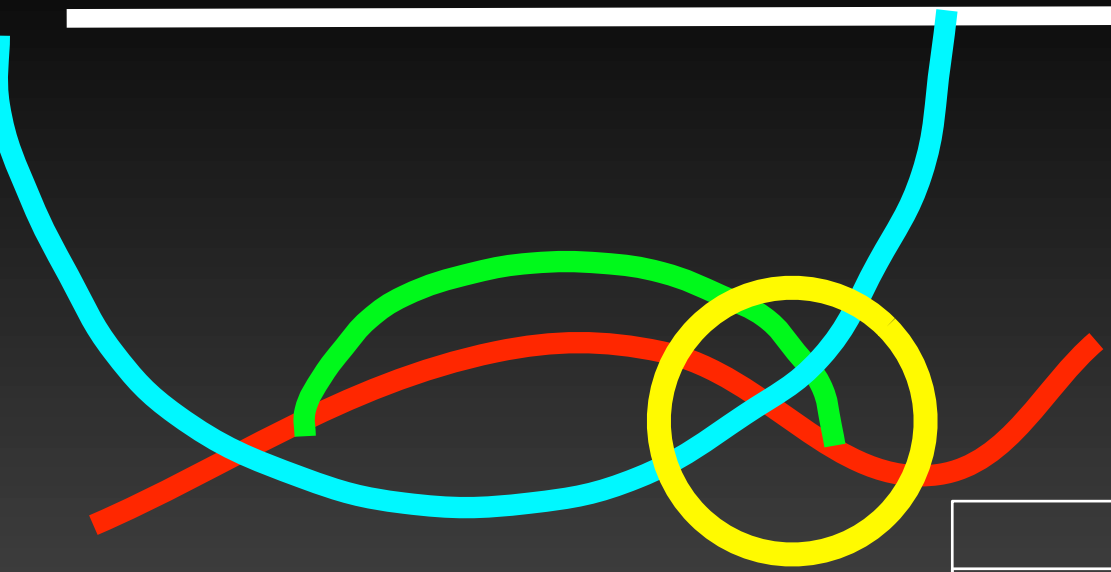


0h

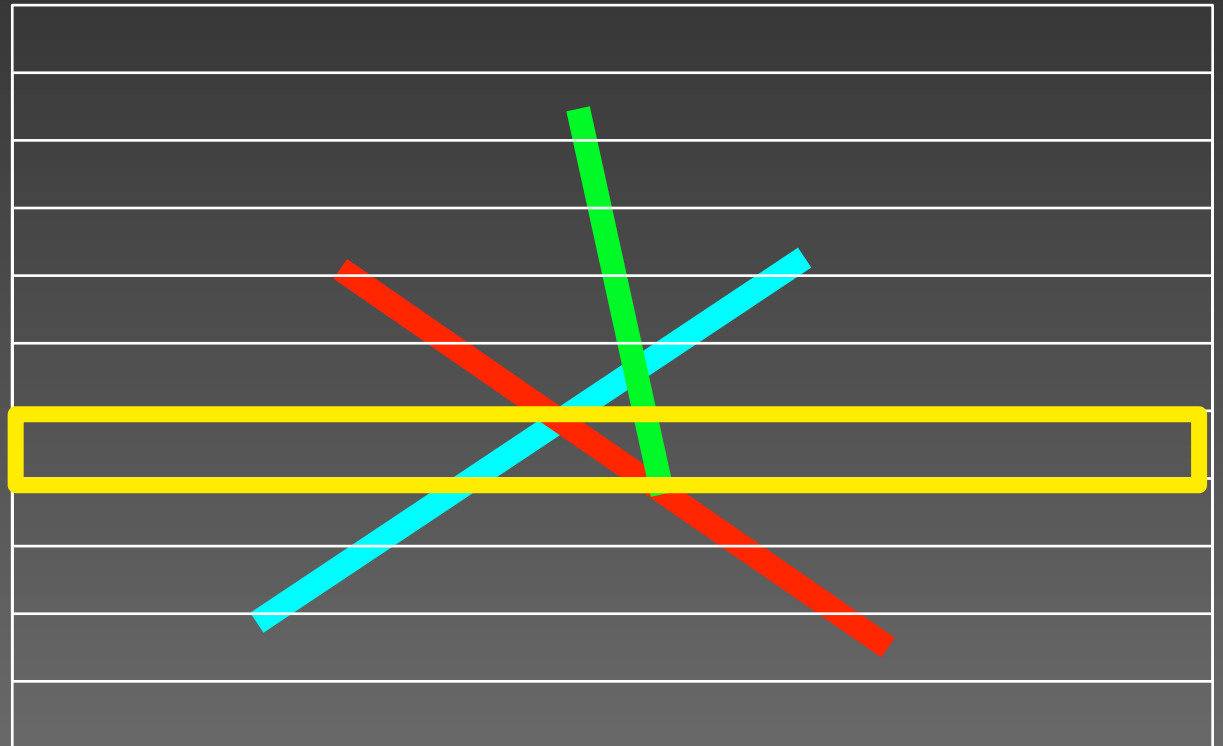


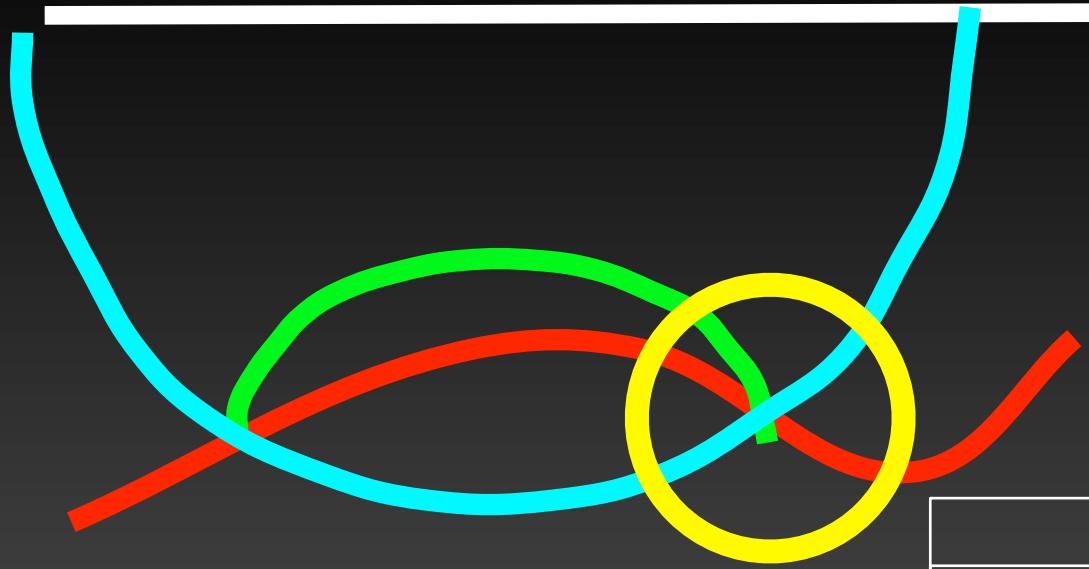
CIG (offset subsuperfície)



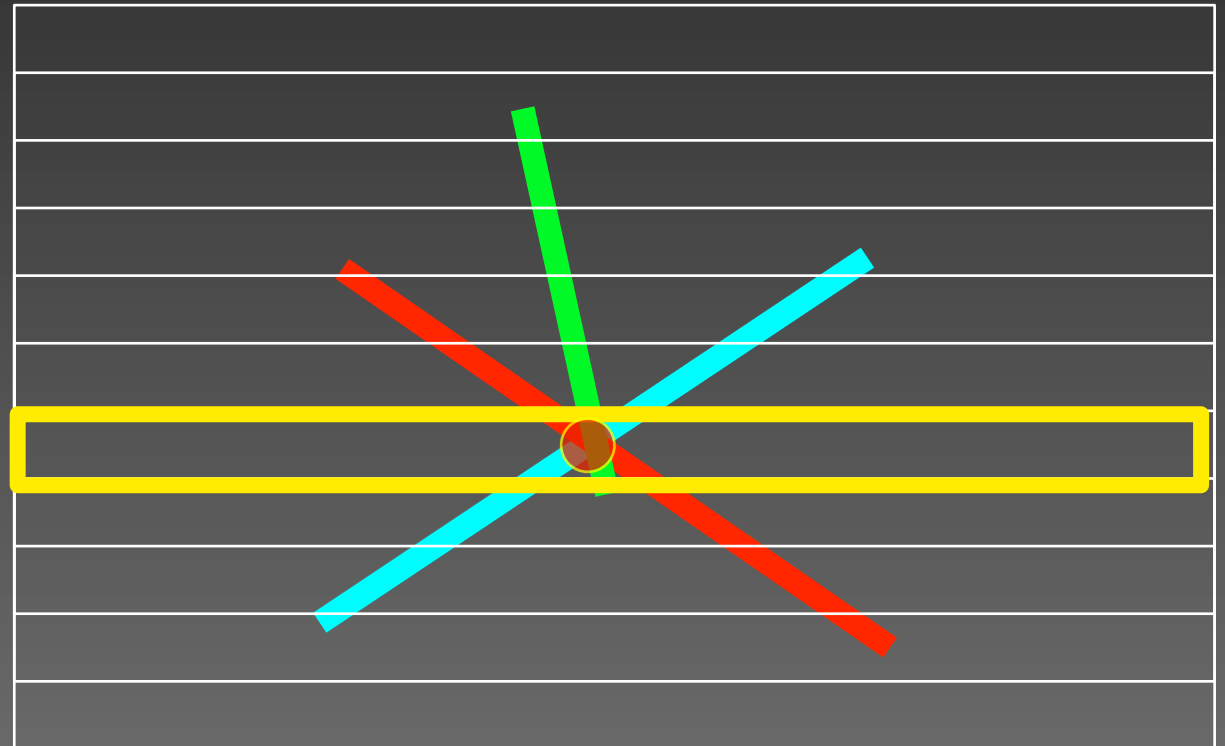


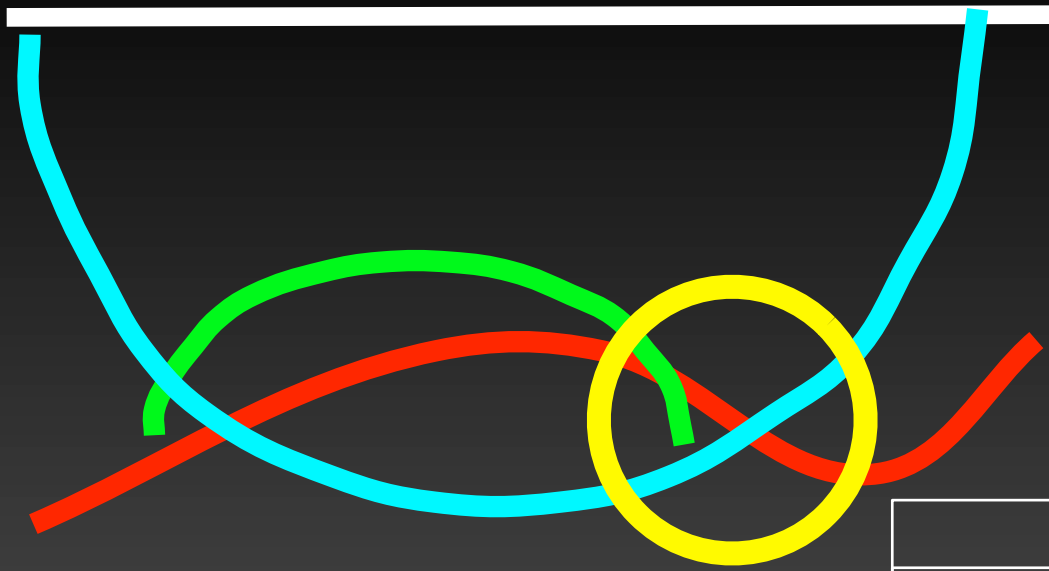
-1h



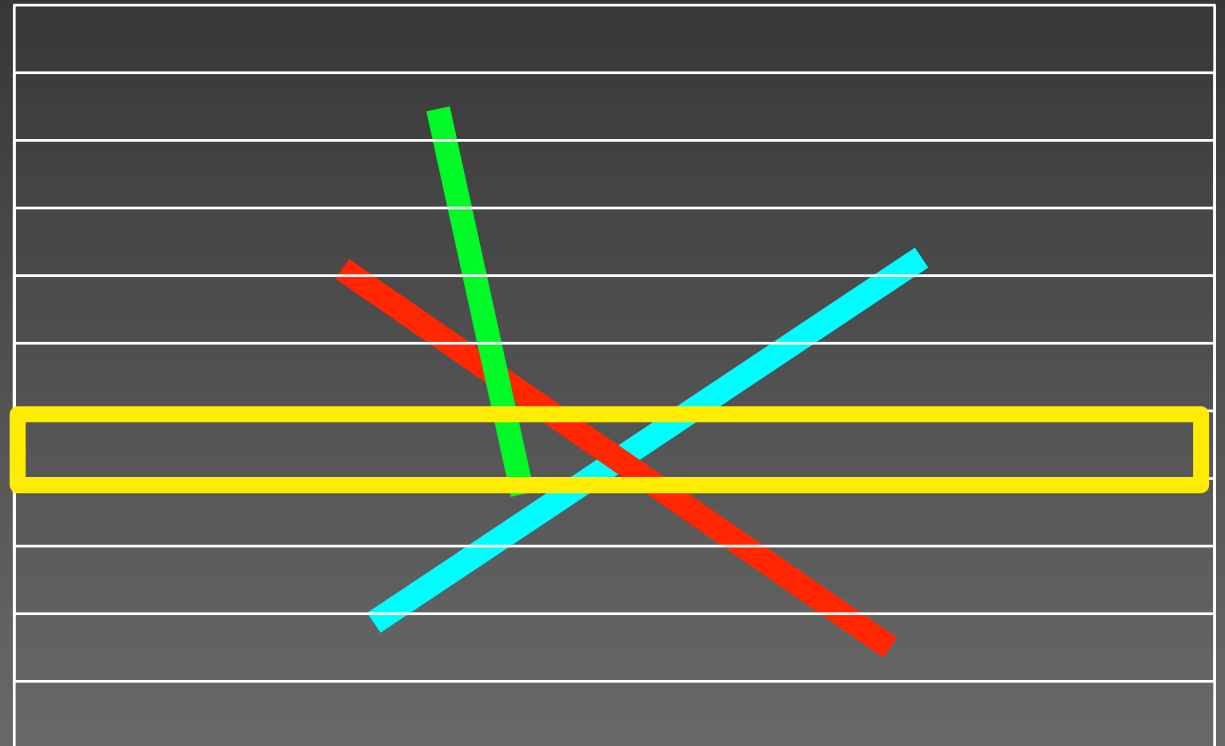


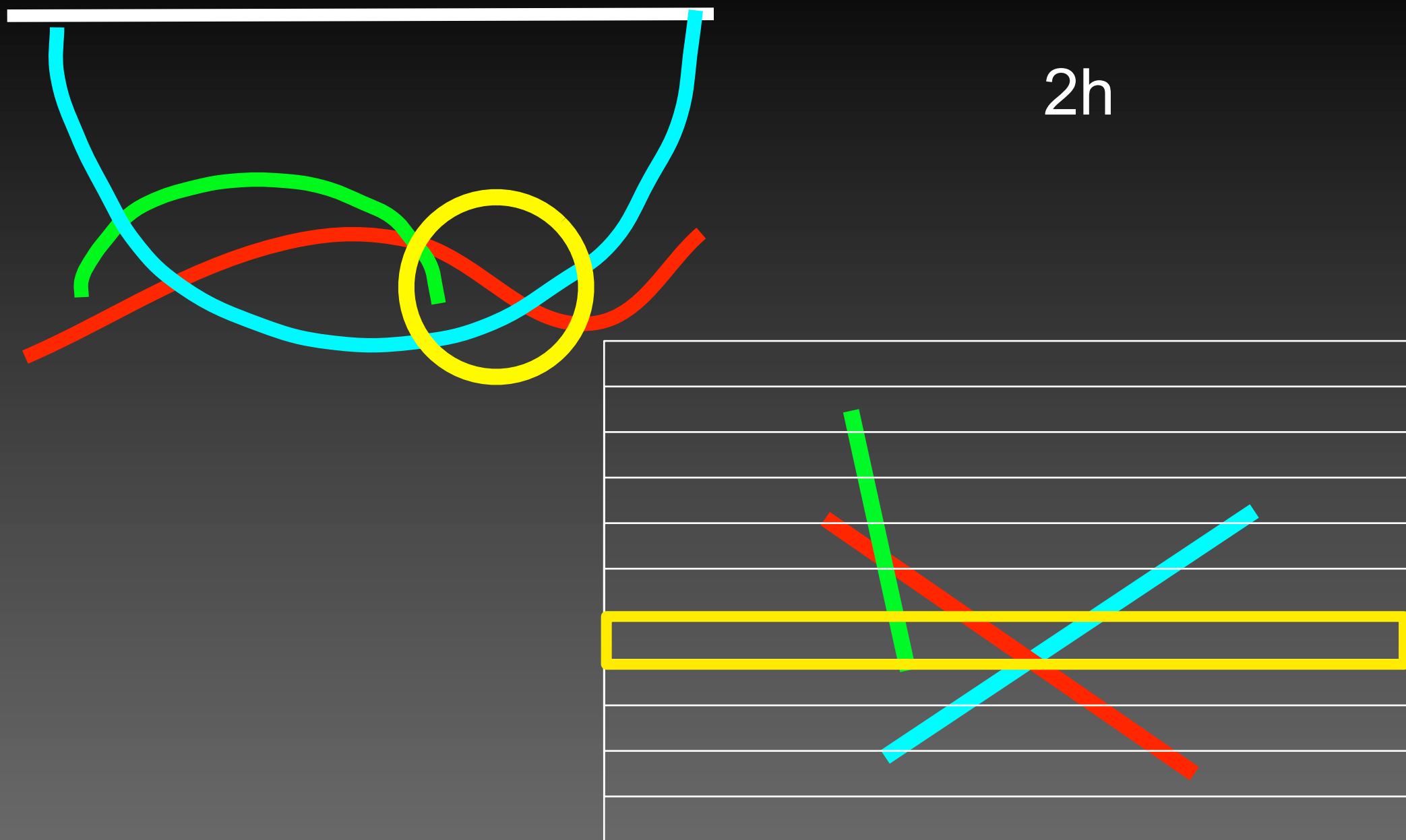
0h

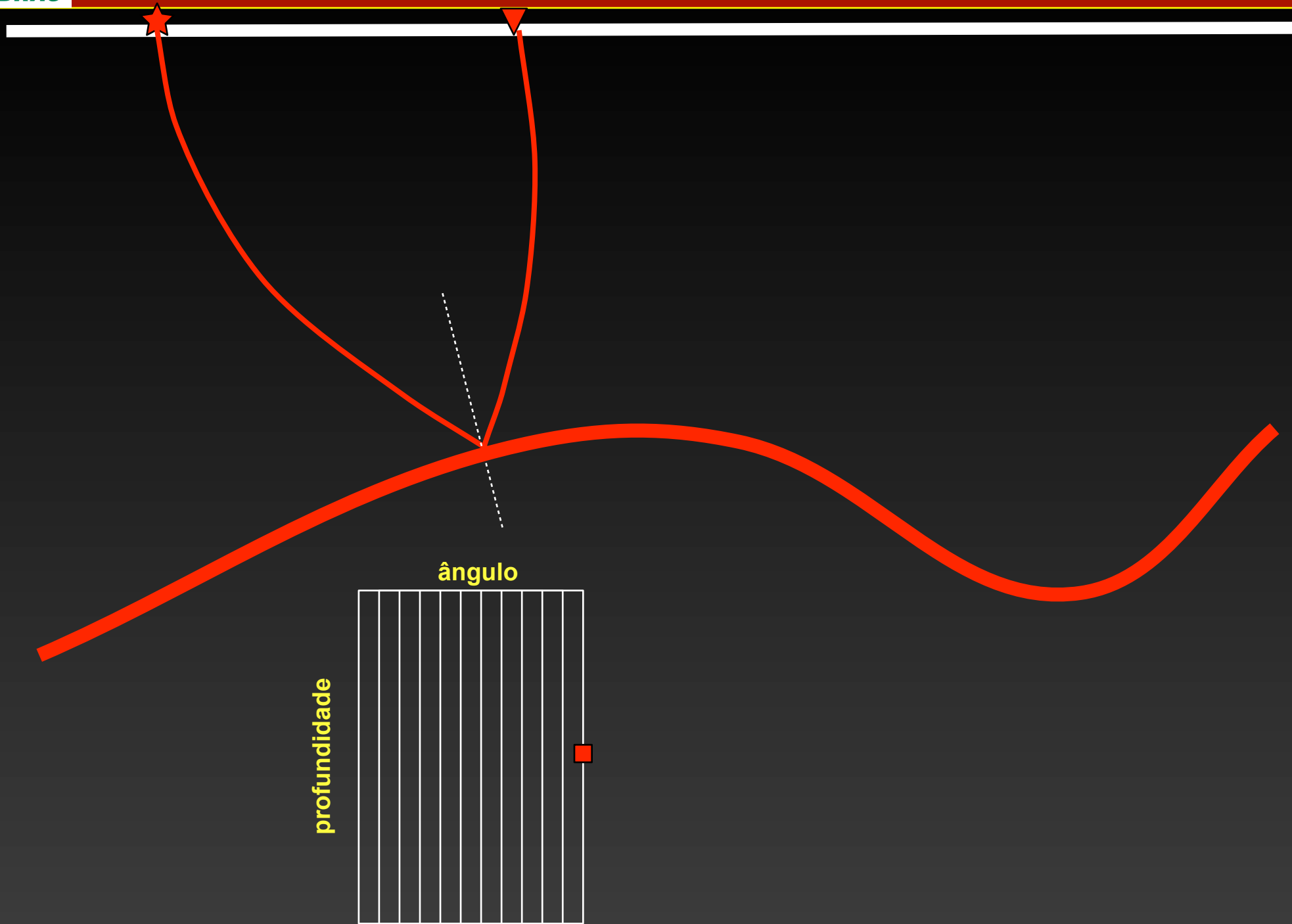


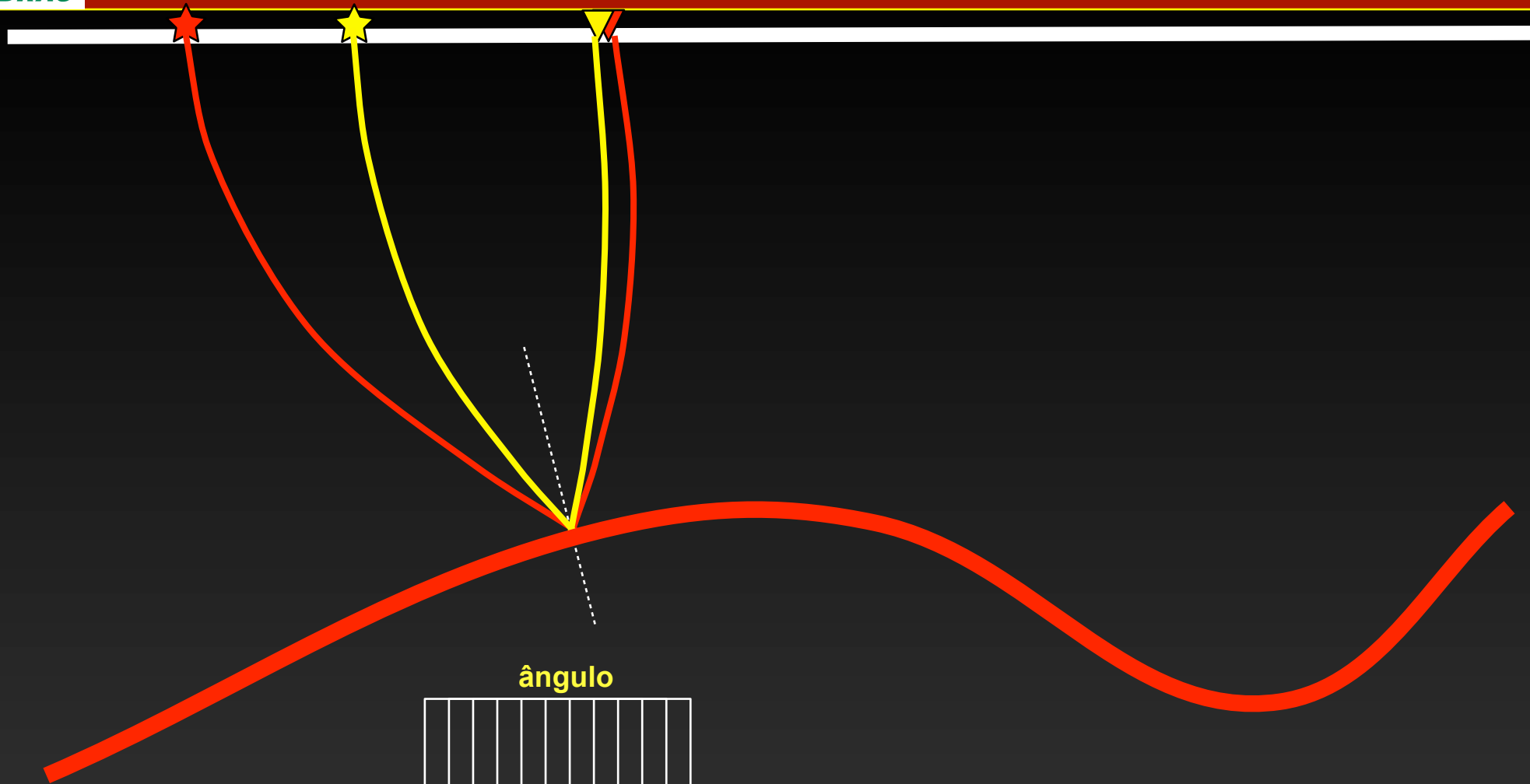


1h



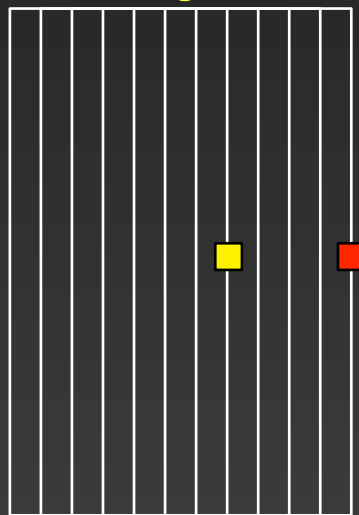


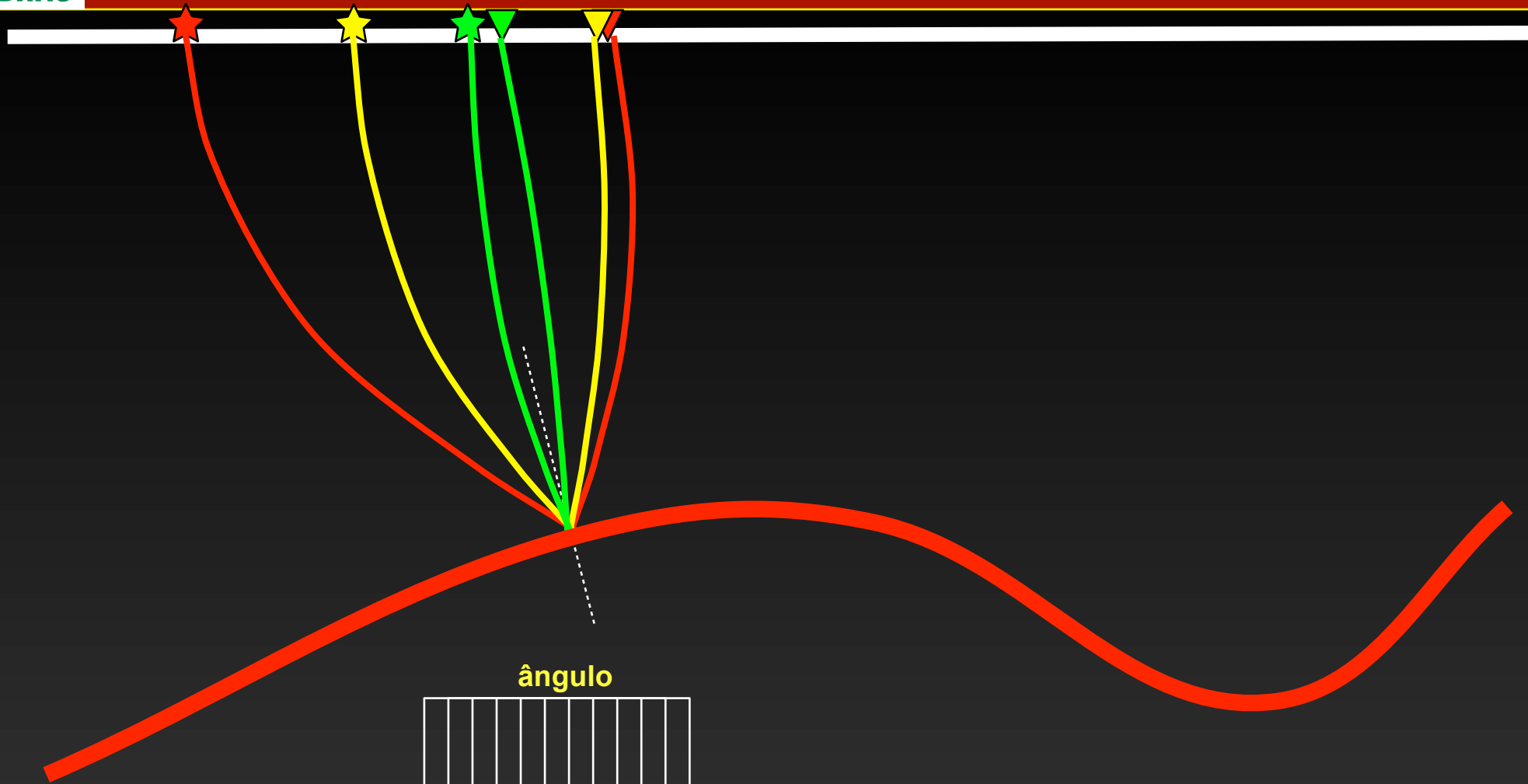




ângulo

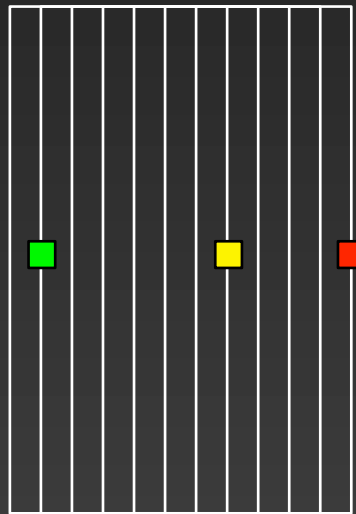
profundidade



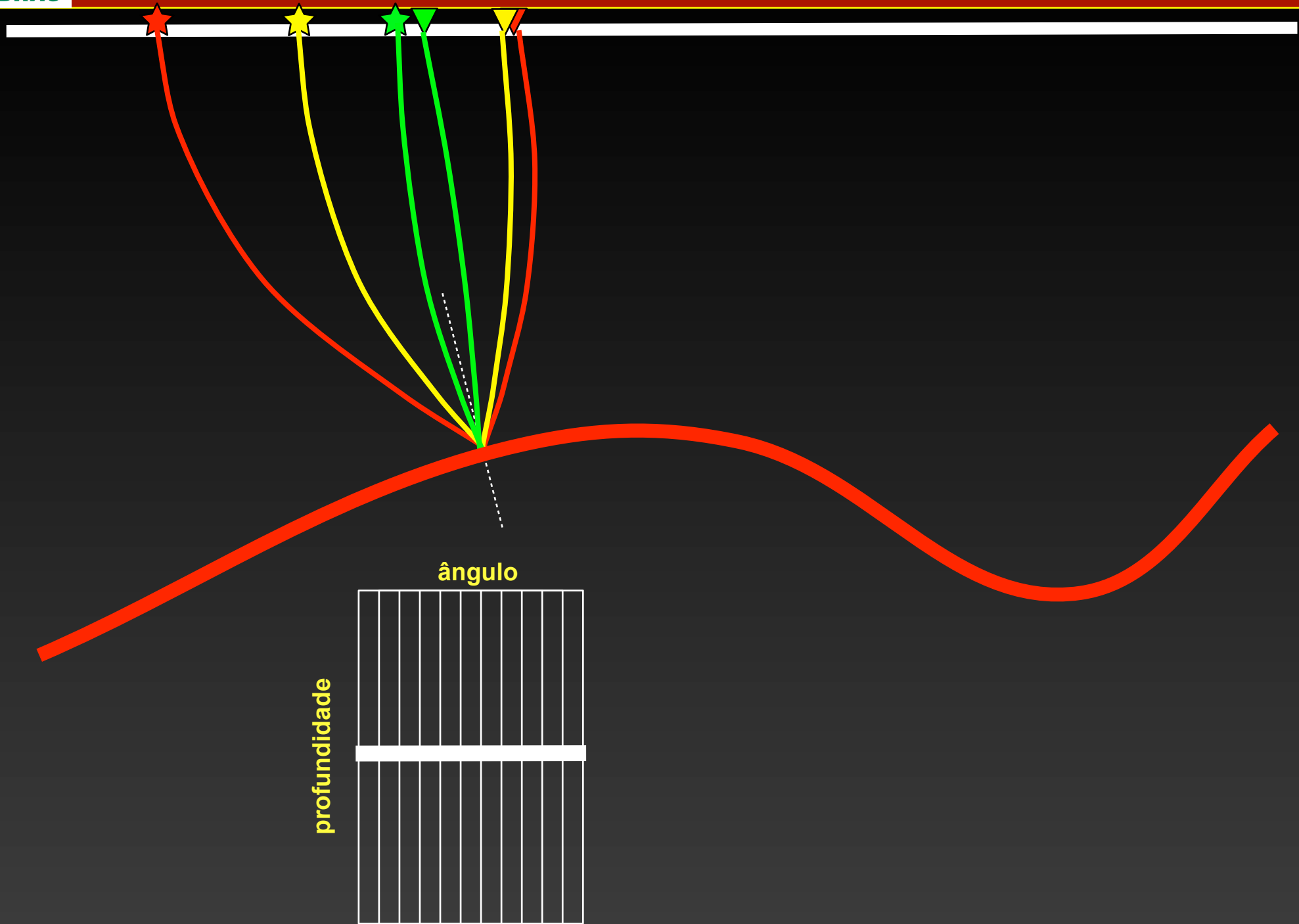


ângulo

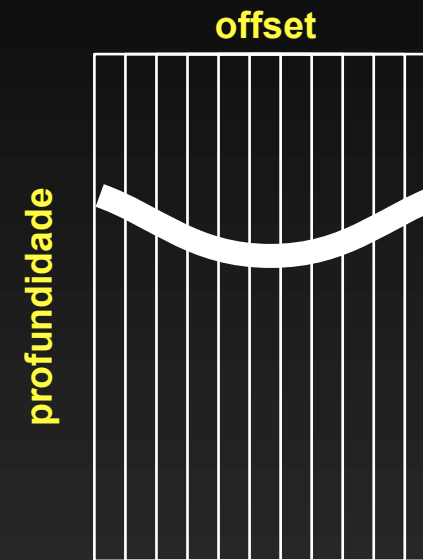
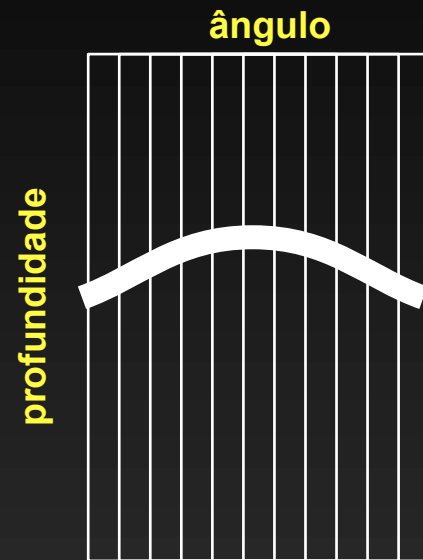
profundidade



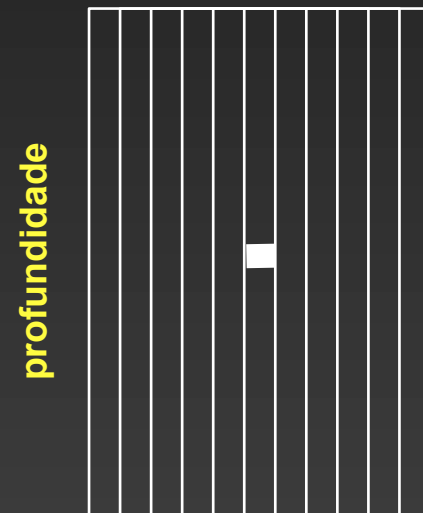
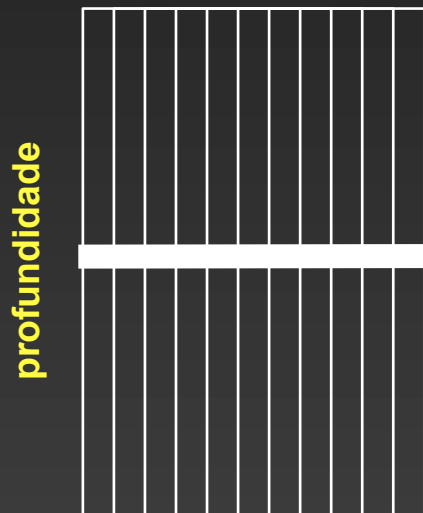
CIG (ângulo de reflexão)



**Velocidade
baixa**



**Velocidade
correta**

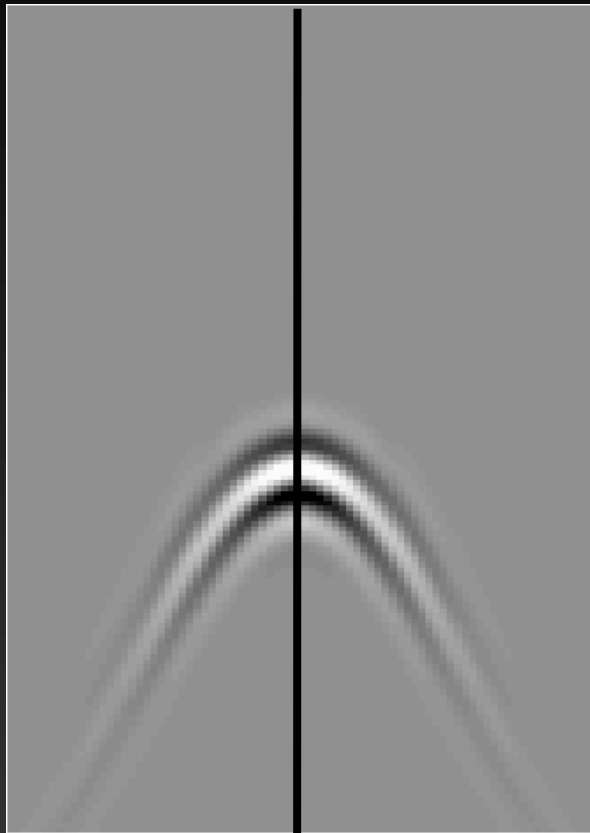


**Velocidade
baixa**

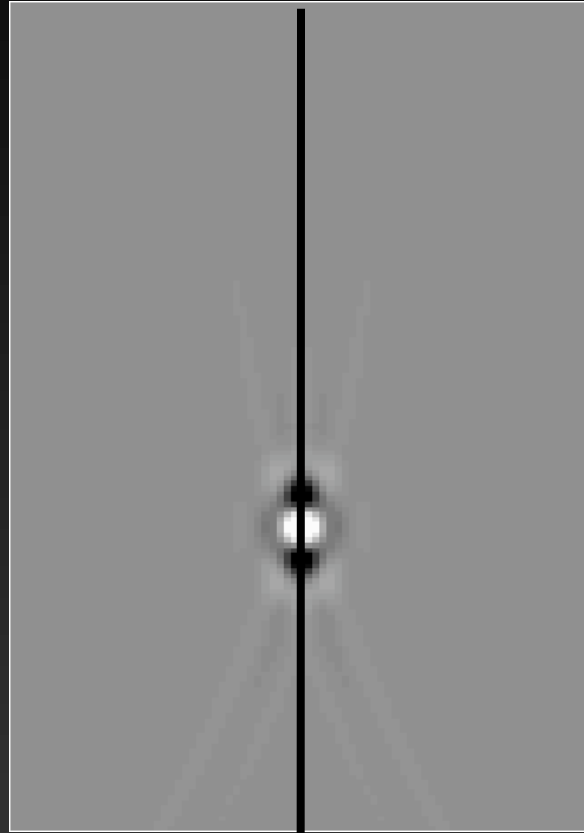
**Velocidade
correta**

**Velocidade
alta**

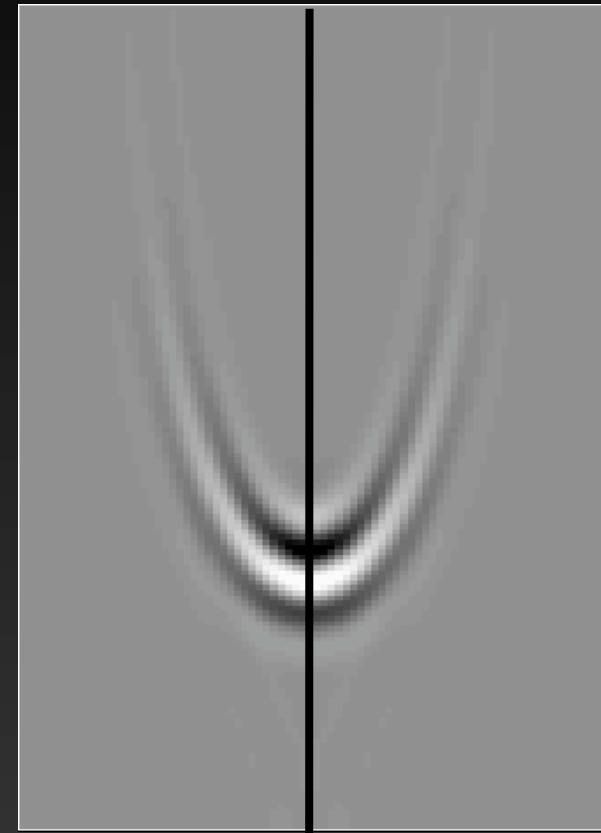
profundidade



offset



offset



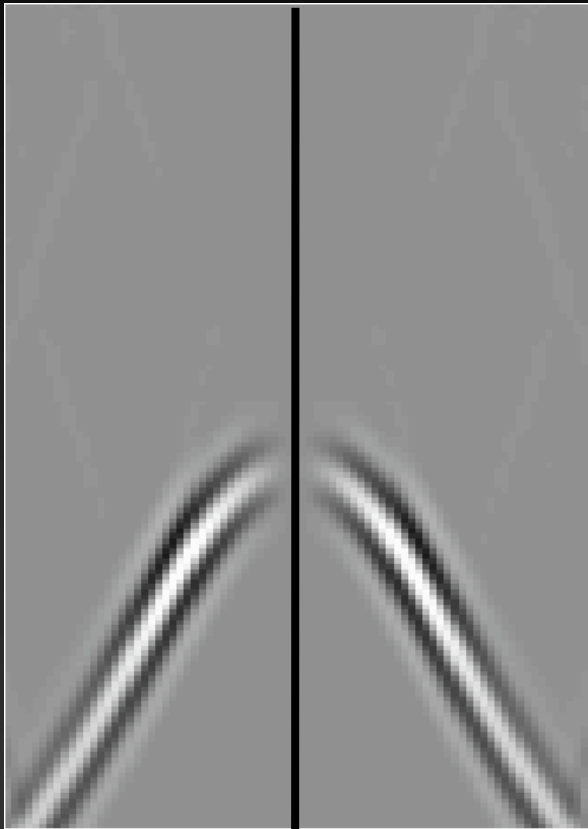
offset

**Velocidade
baixa**

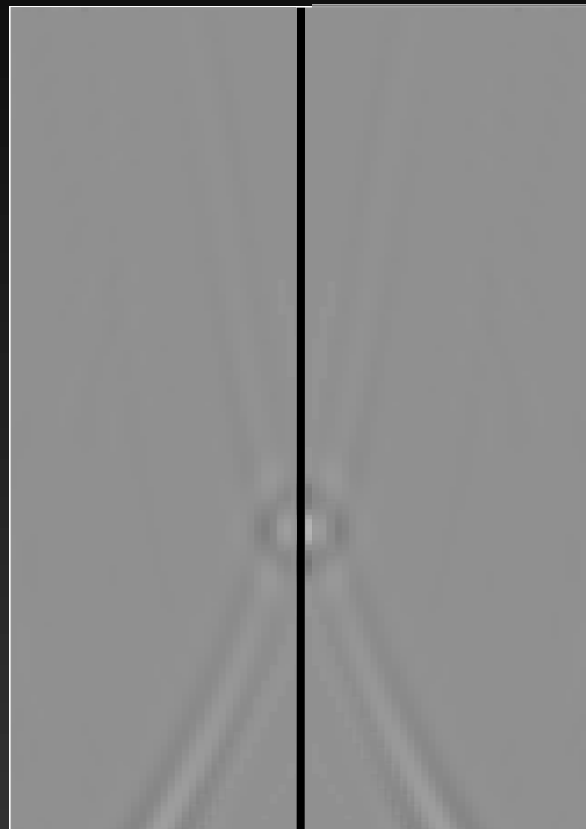
**Velocidade
correta**

**Velocidade
alta**

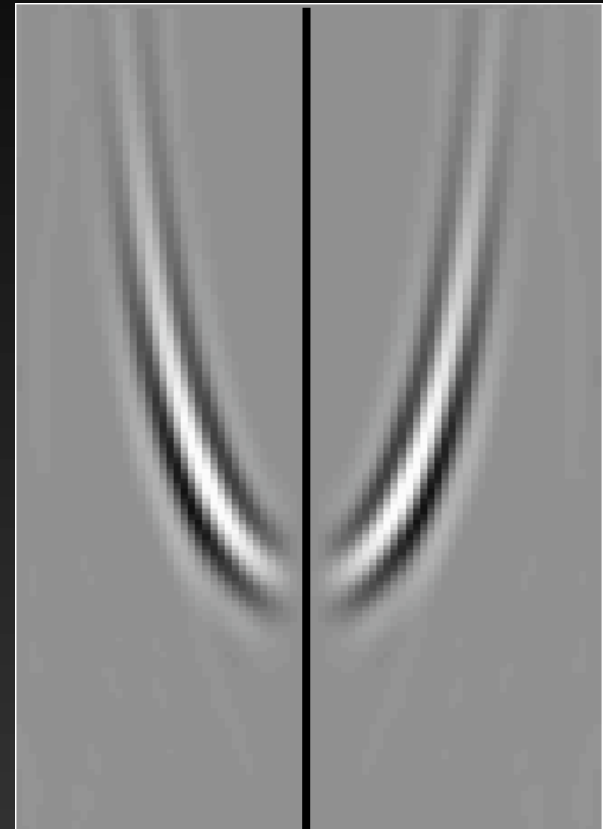
profundidade



offset



offset

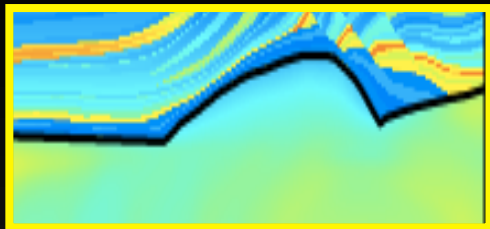


offset

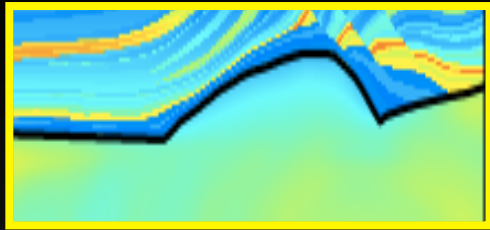
BR

PETROBRAS

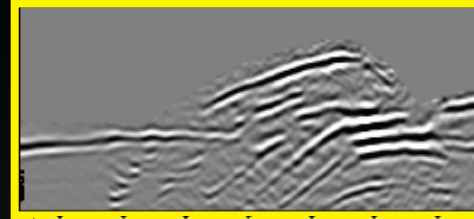
Calculando Δs



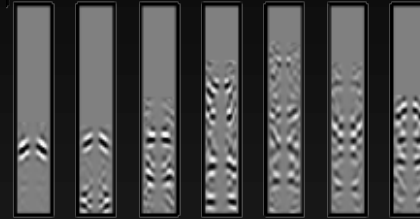
s_0

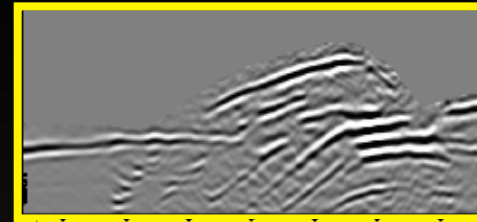
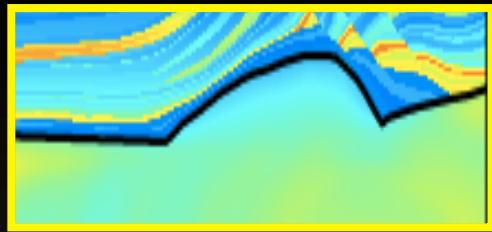


S_0

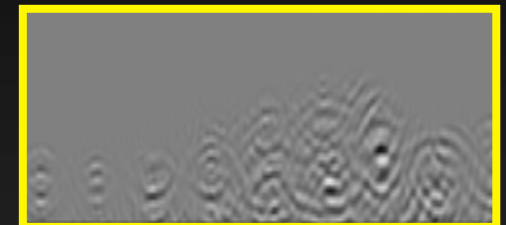
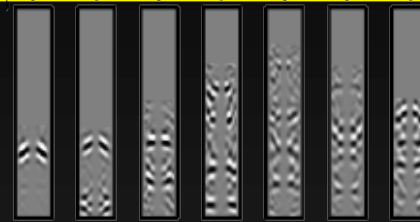
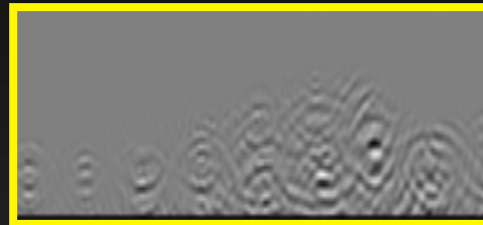


$$\Delta I = |h|I_0$$

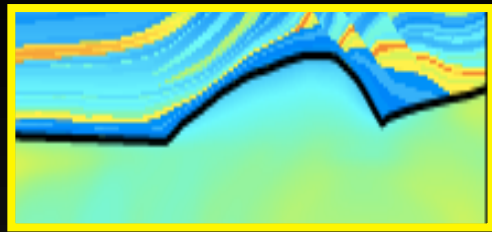




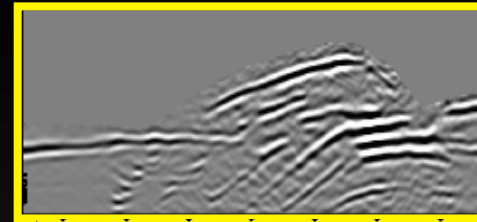
$$\Delta I = |h|I_0$$



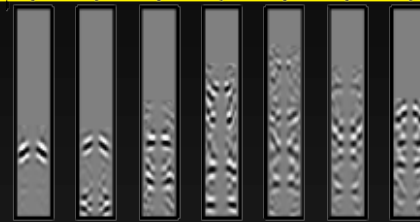
$$\Delta P^\pm = \mathbf{T}^\mp \mathbf{P}_0^\pm \Delta I$$



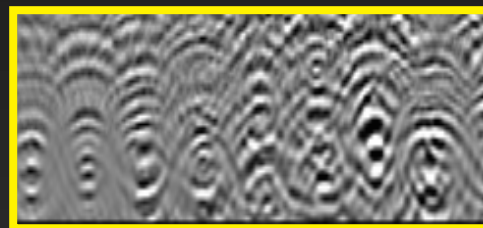
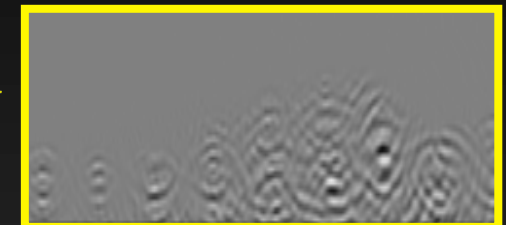
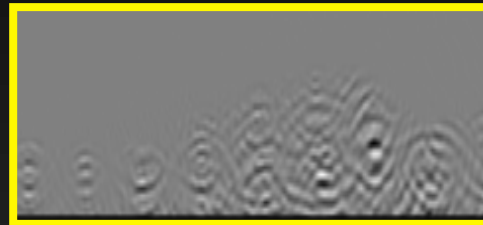
S_0



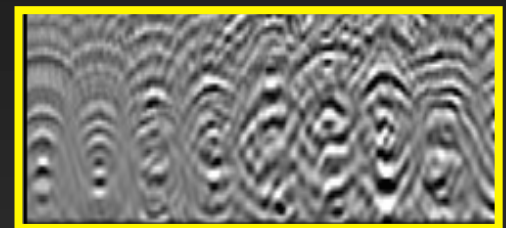
$$\Delta I = |h|I_0$$

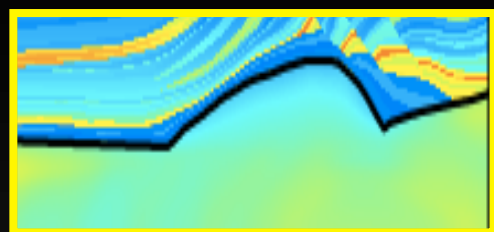


$$\Delta P^\pm = \mathbf{T}^\mp \mathbf{P}_0^\pm \Delta I$$

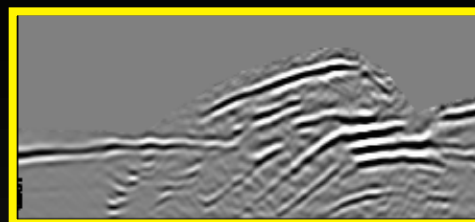


$$P_{sc}^\pm = \mathbf{B}^\pm P_0^\pm$$

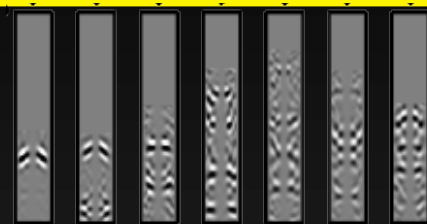




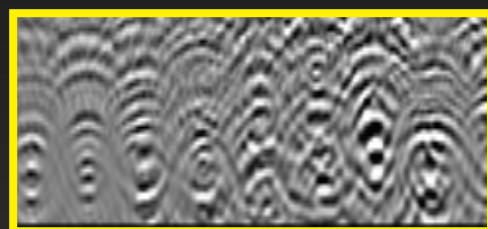
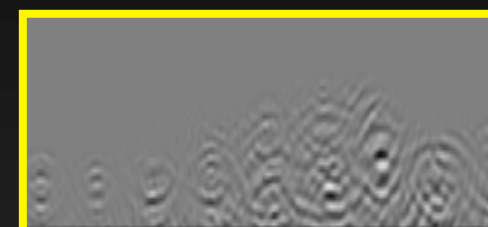
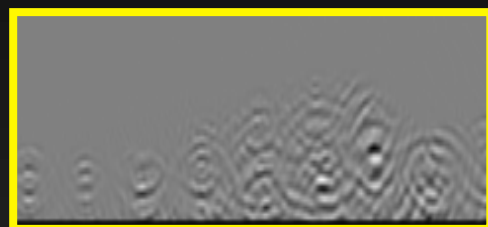
s_0



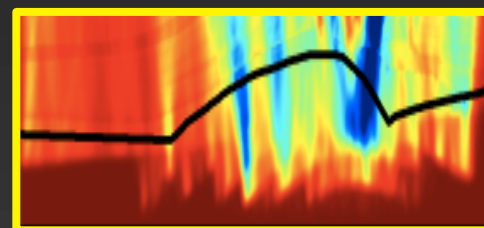
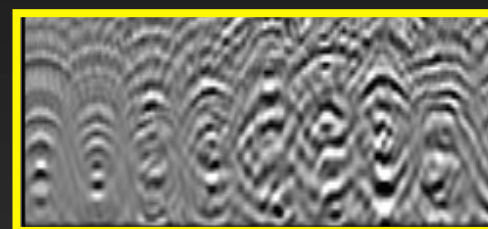
$\Delta I = |h|I_0$



$$\Delta P^\pm = \mathbf{T}^\mp \mathbf{P}_0^\pm \Delta I$$

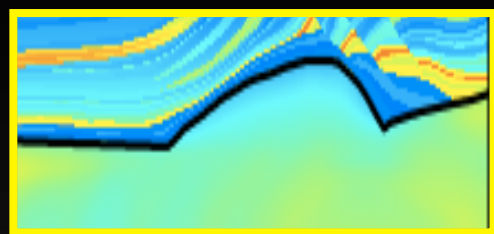


$$P_{sc}^\pm = \mathbf{B}^\pm P_0^\pm$$

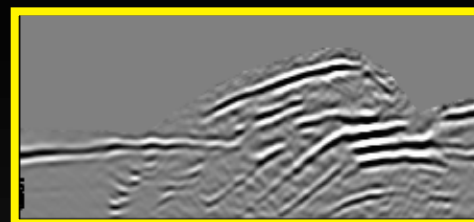


$$\Delta s = P_{sc}^{+*} \Delta P^+ + P_{sc}^{-*} \Delta P^-$$

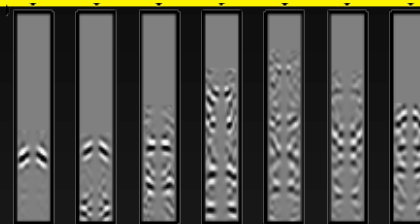




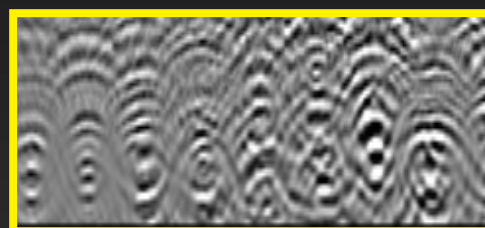
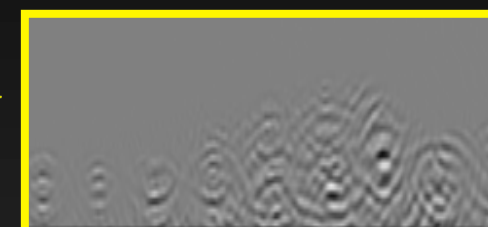
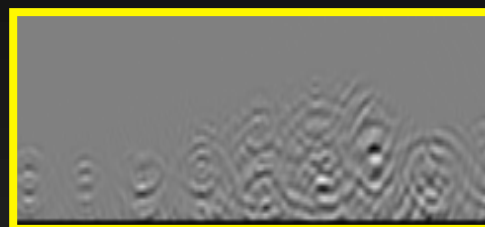
S_0



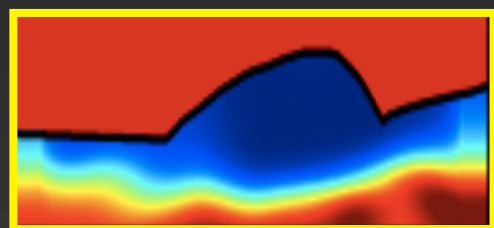
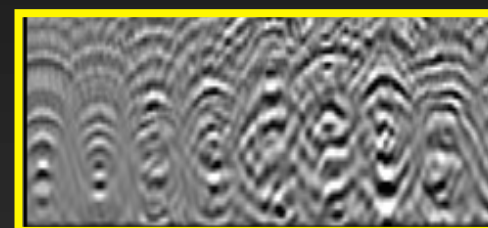
$\Delta I = |h|I_0$



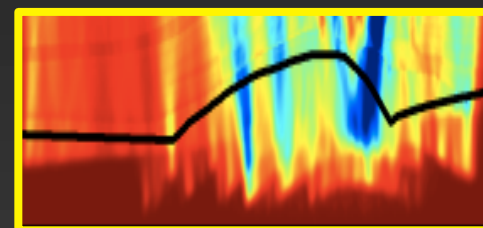
$\Delta P^\pm = \mathbf{T}^\mp \mathbf{P}_0^\pm \Delta I$



$P_{sc}^\pm = \mathbf{B}^\pm P_0^\pm$

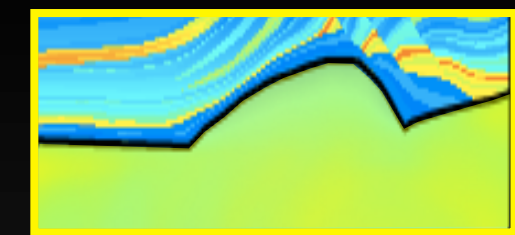


$\langle \Delta s \rangle = \mathbf{S} \Delta s$

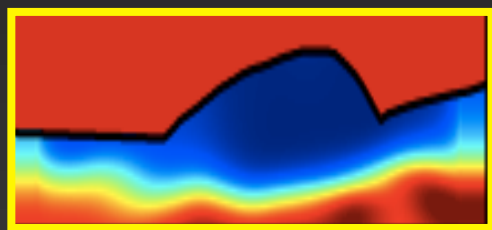


$\Delta s = P_{sc}^{+*} \Delta P^+ + P_{sc}^{-*} \Delta P^-$

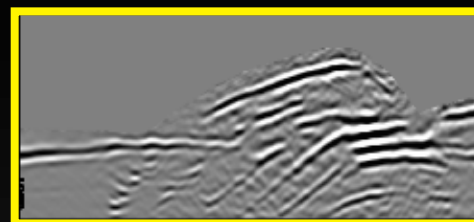




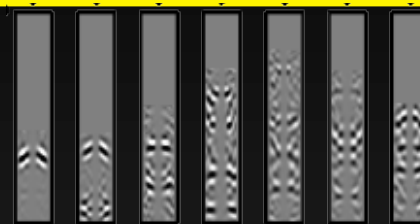
$$s_0 = s_0 - \lambda \langle \Delta s \rangle$$



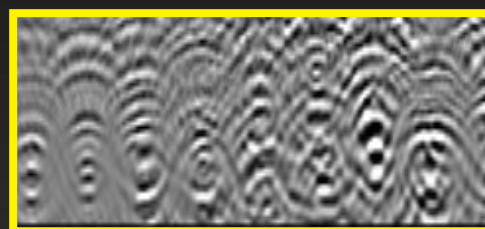
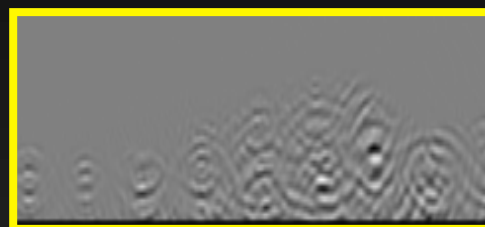
$$\langle \Delta s \rangle = \mathbf{S} \Delta s$$



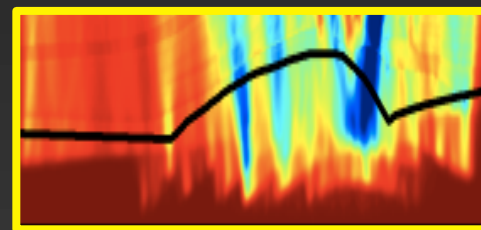
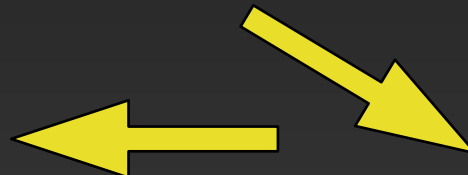
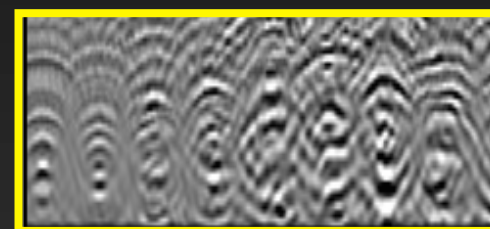
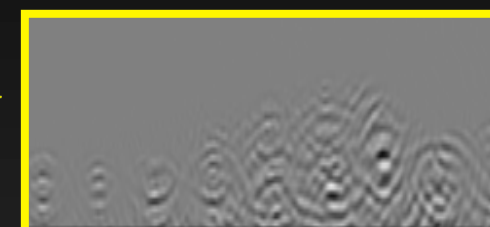
$$\Delta I = |h| I_0$$



$$\Delta P^\pm = \mathbf{T}^\mp \mathbf{P}_0^\pm \Delta I$$



$$P_{sc}^\pm = \mathbf{B}^\pm P_0^\pm$$



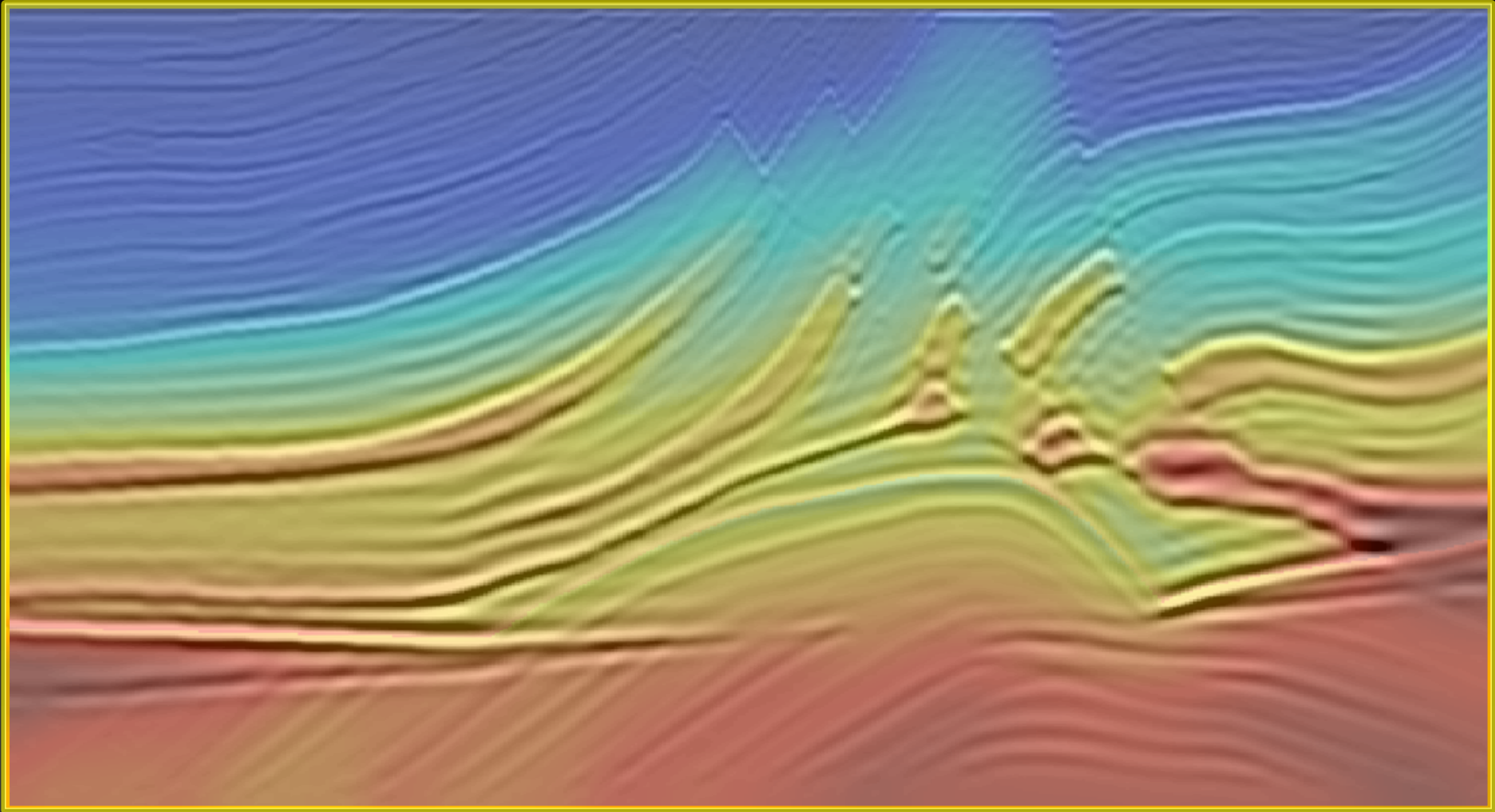
$$\Delta s = P_{sc}^{+*} \Delta P^+ + P_{sc}^{-*} \Delta P^-$$



- **Modelagem**
- **Migração**
- **AVM com tomografia de campos de onda (AVMTO)**
- **Custo e flexibilidade**
- **Exemplos**

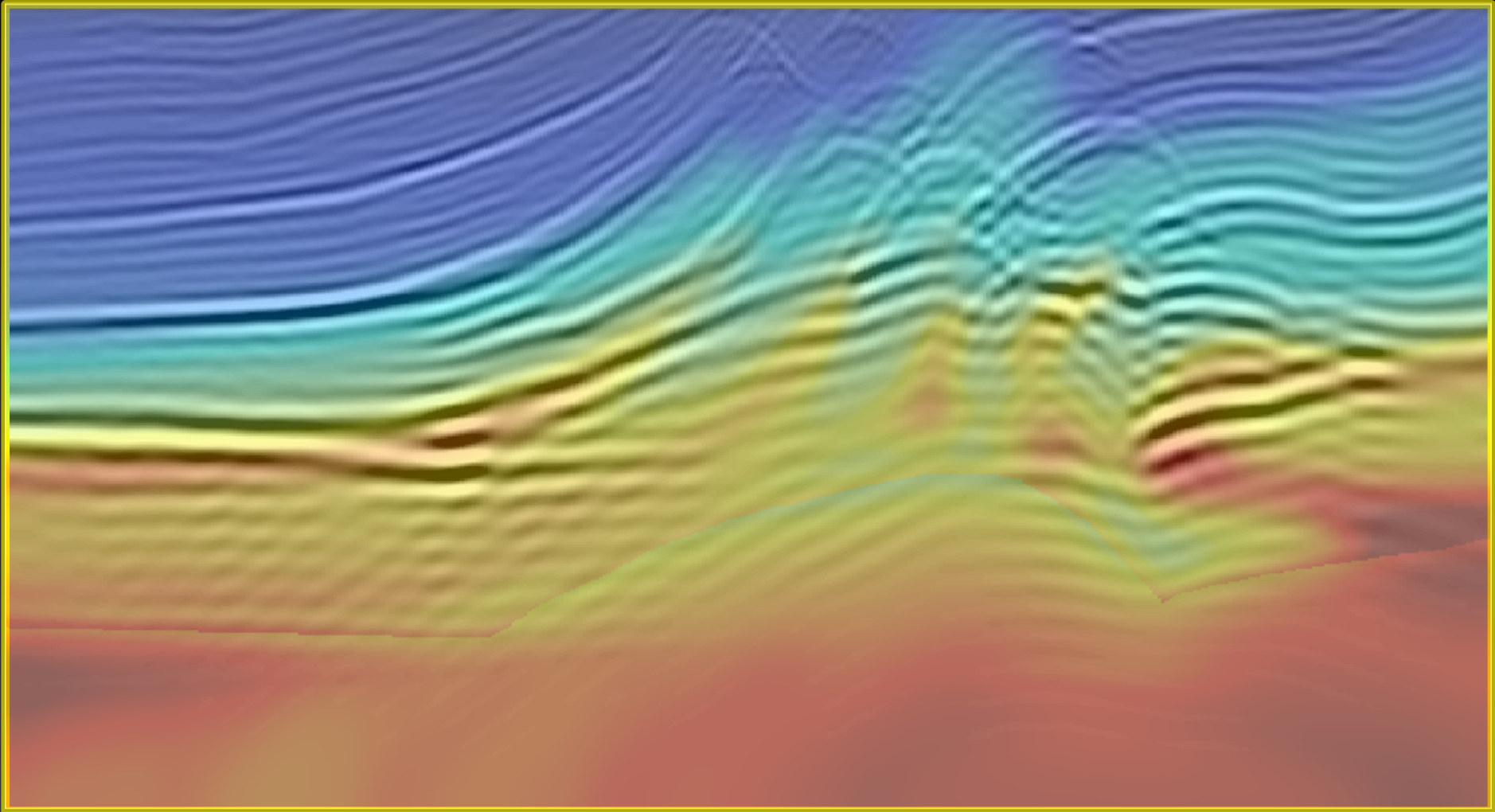
distância

profundidade



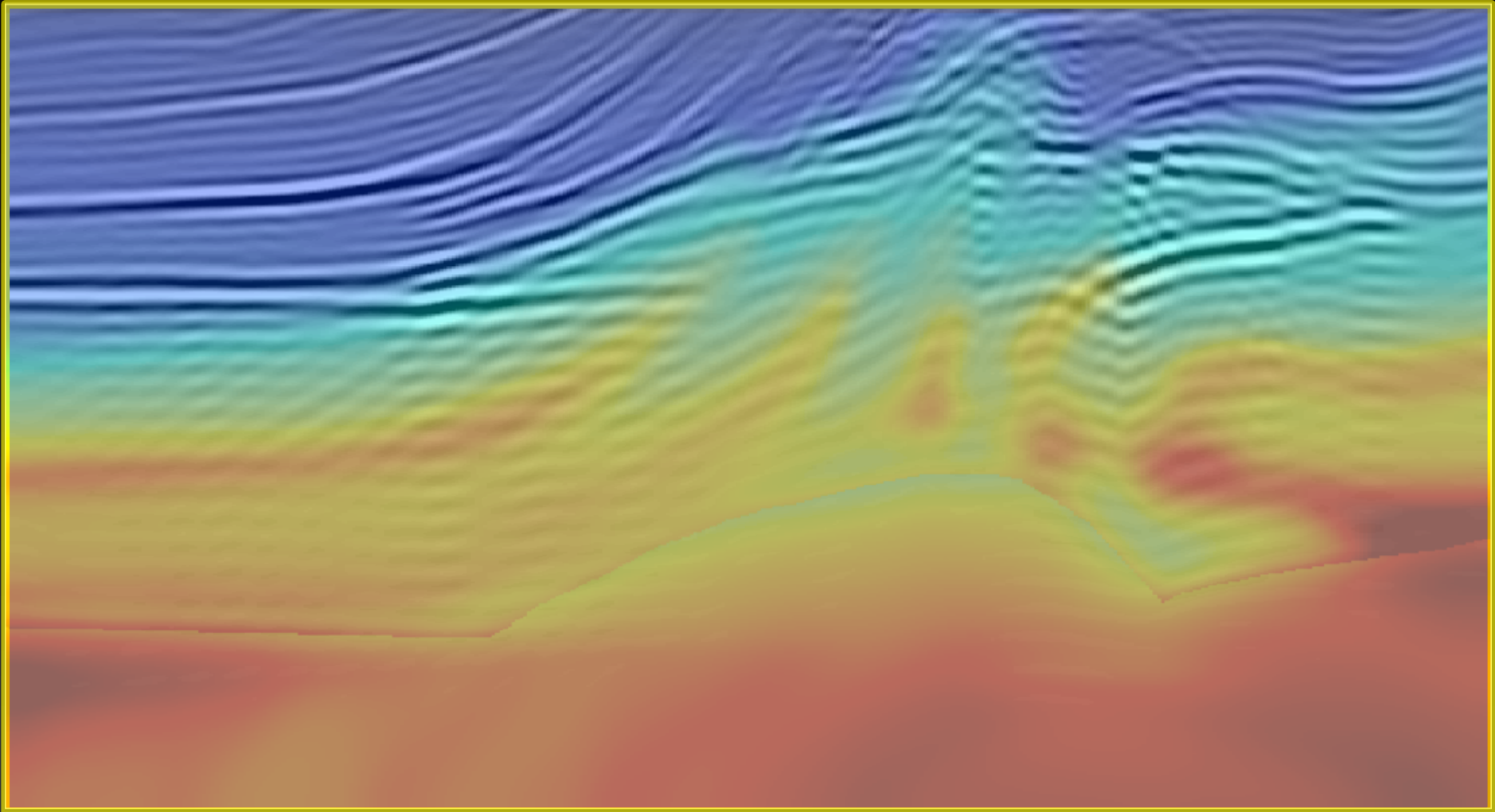
distância

profundidade



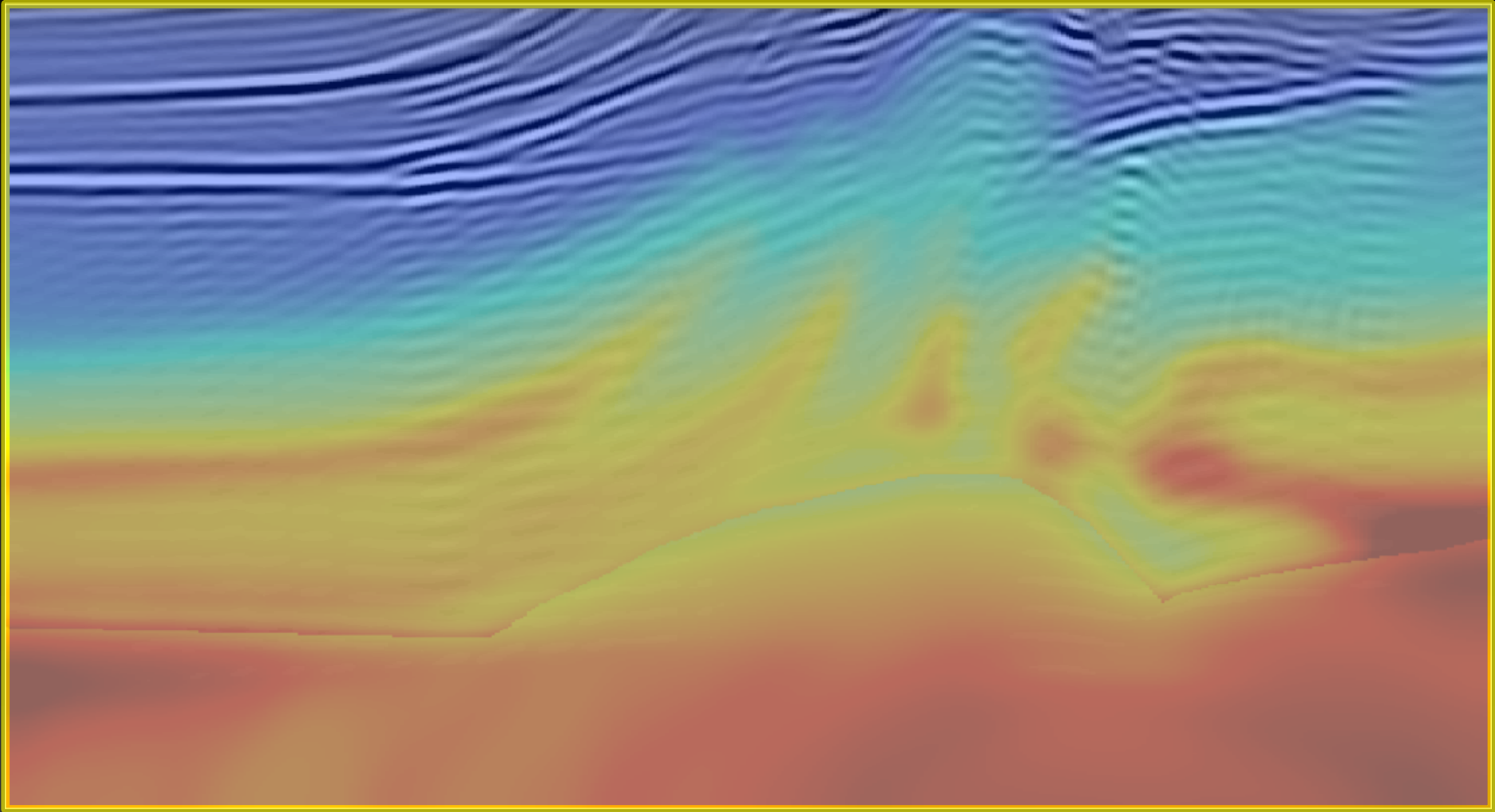
distância

profundidade



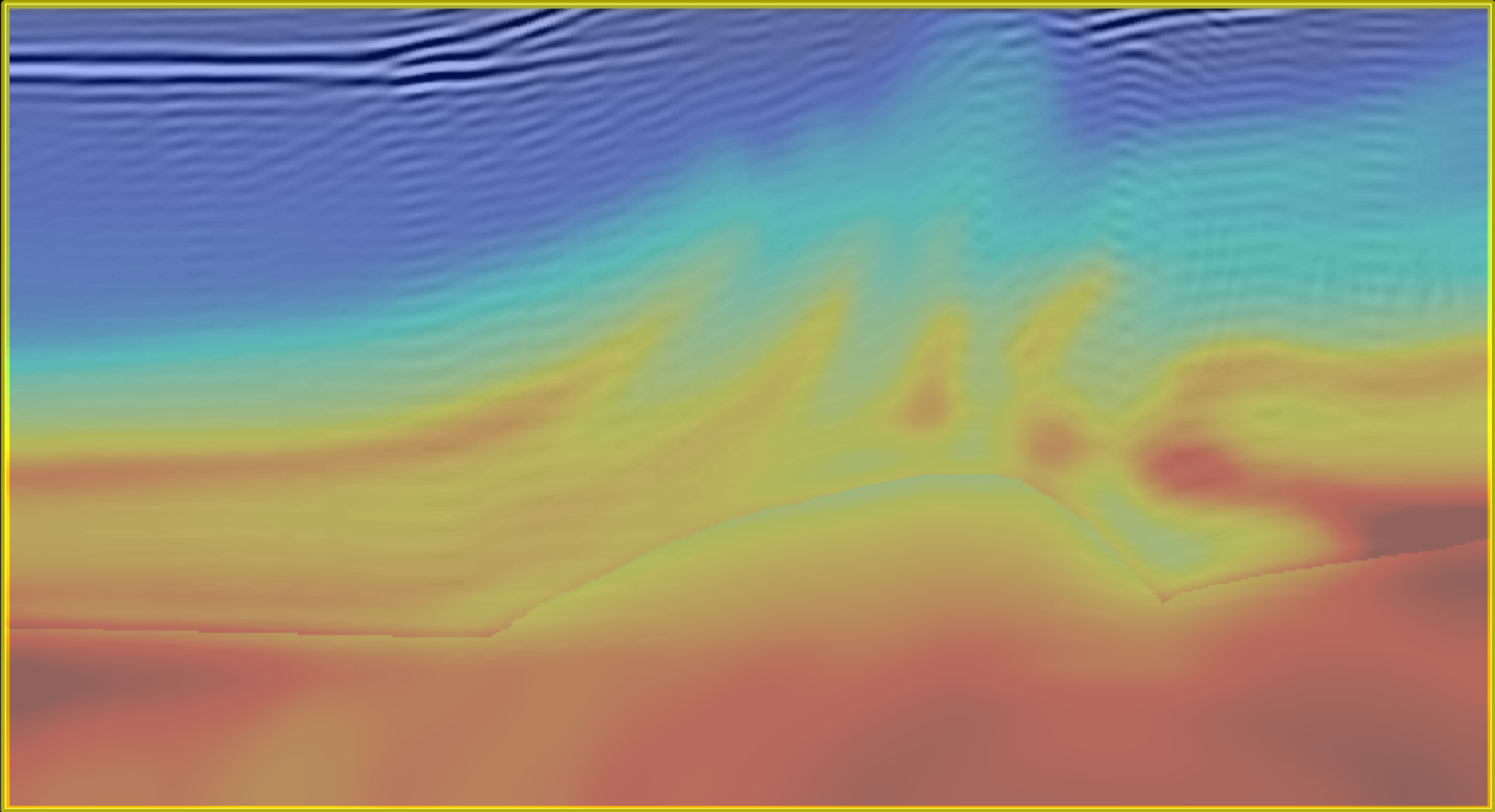
distância

profundidade



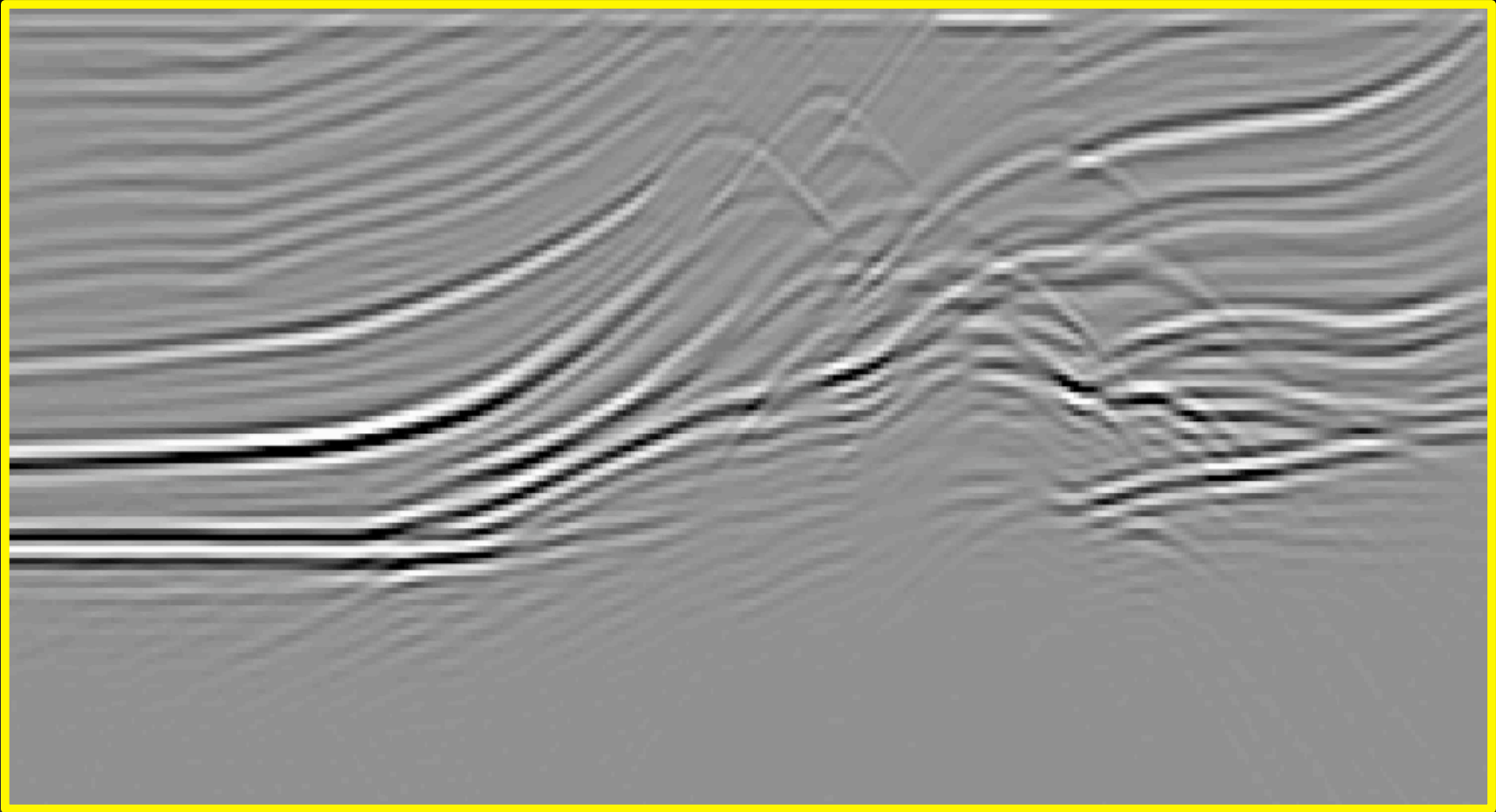
distância

profundidade



distância

tempo

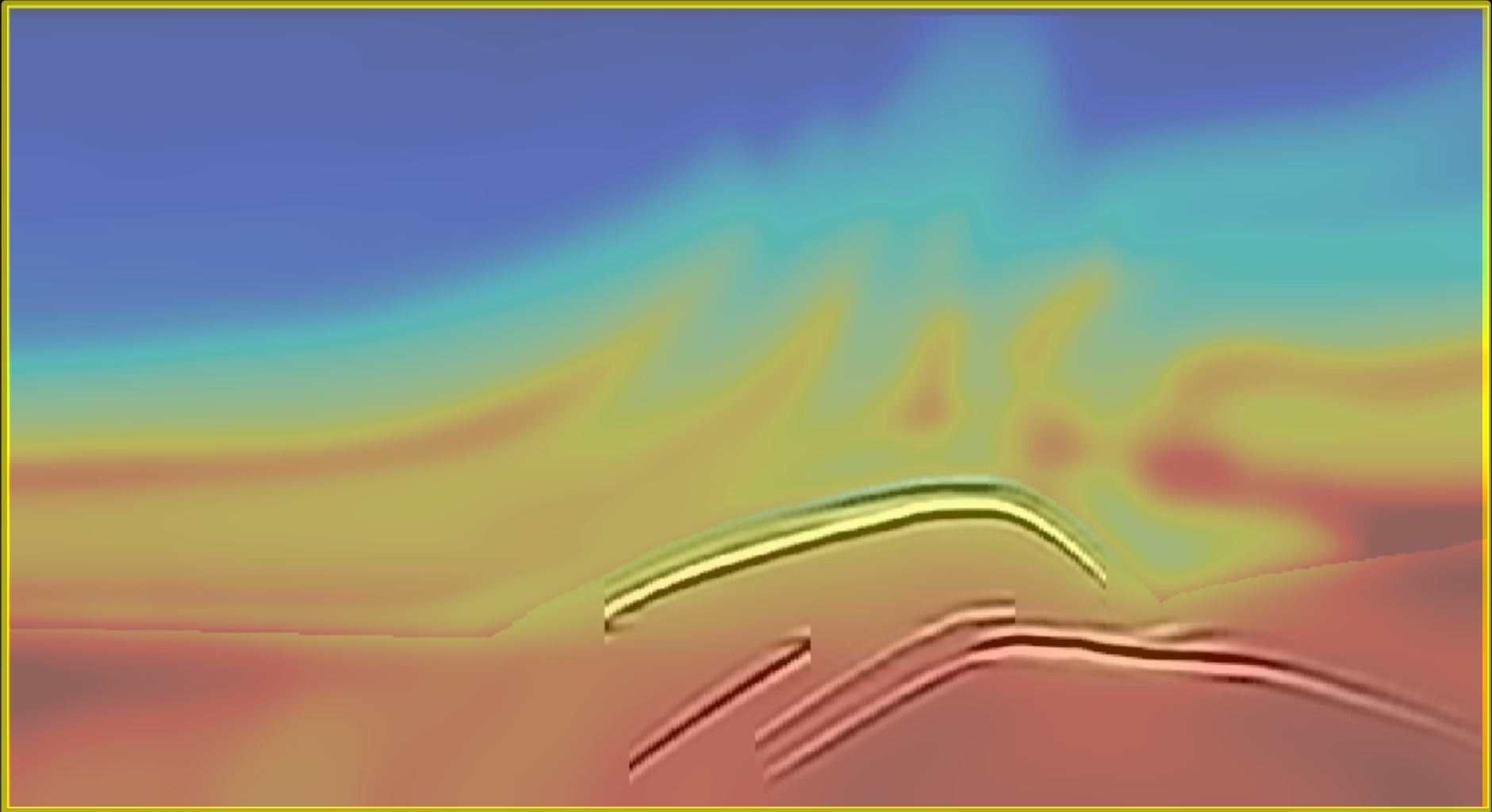


- Usa CIG's para modelar dados que carreguem informação sobre a velocidade

- Usa CIG's para modelar dados que carreguem informação sobre a velocidade
- Usa refletores chave selecionados na imagem original pré-stack, naturalmente incorporando uma estratégia baseada em horizontes

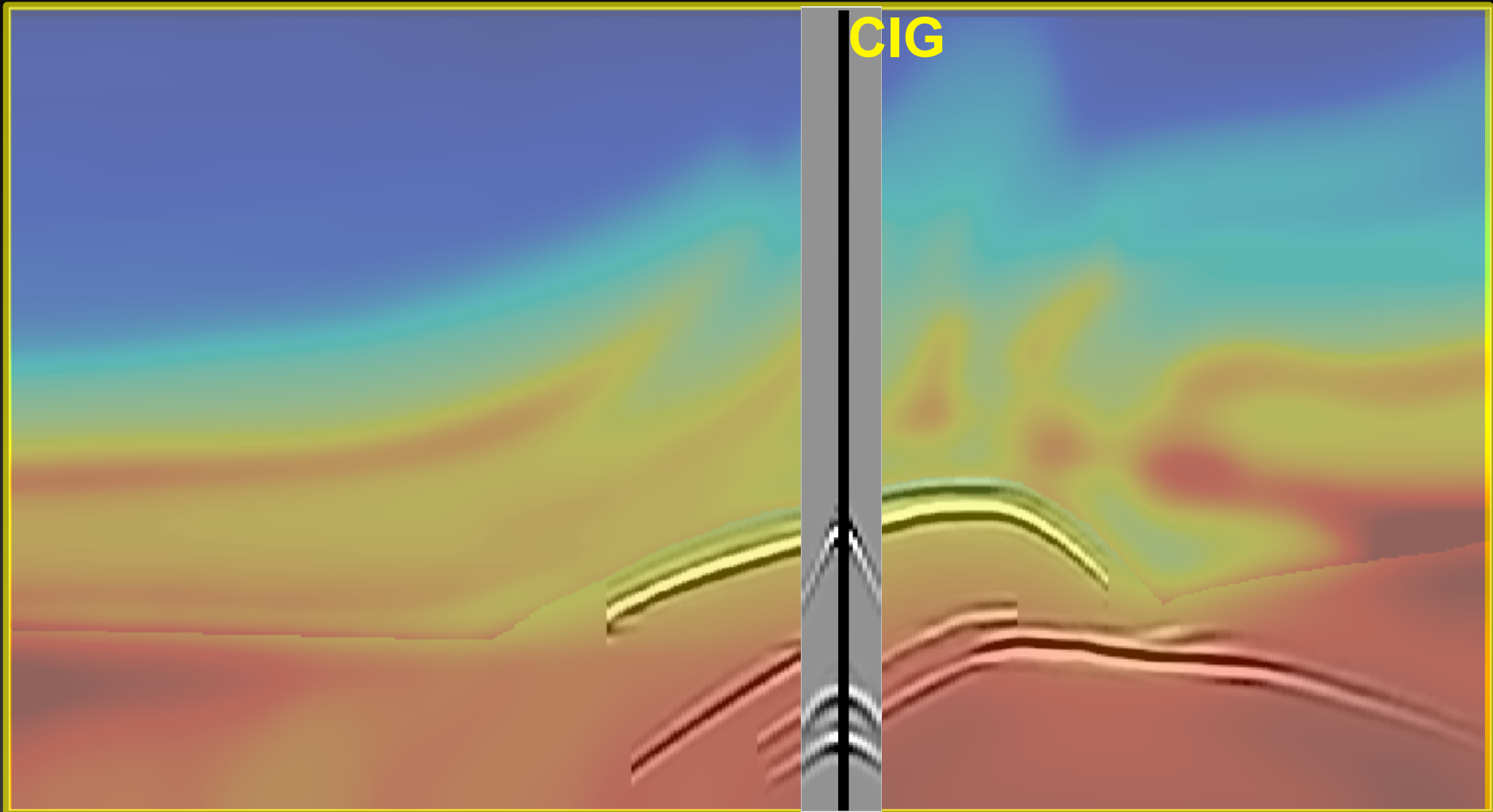
distância

profundidade



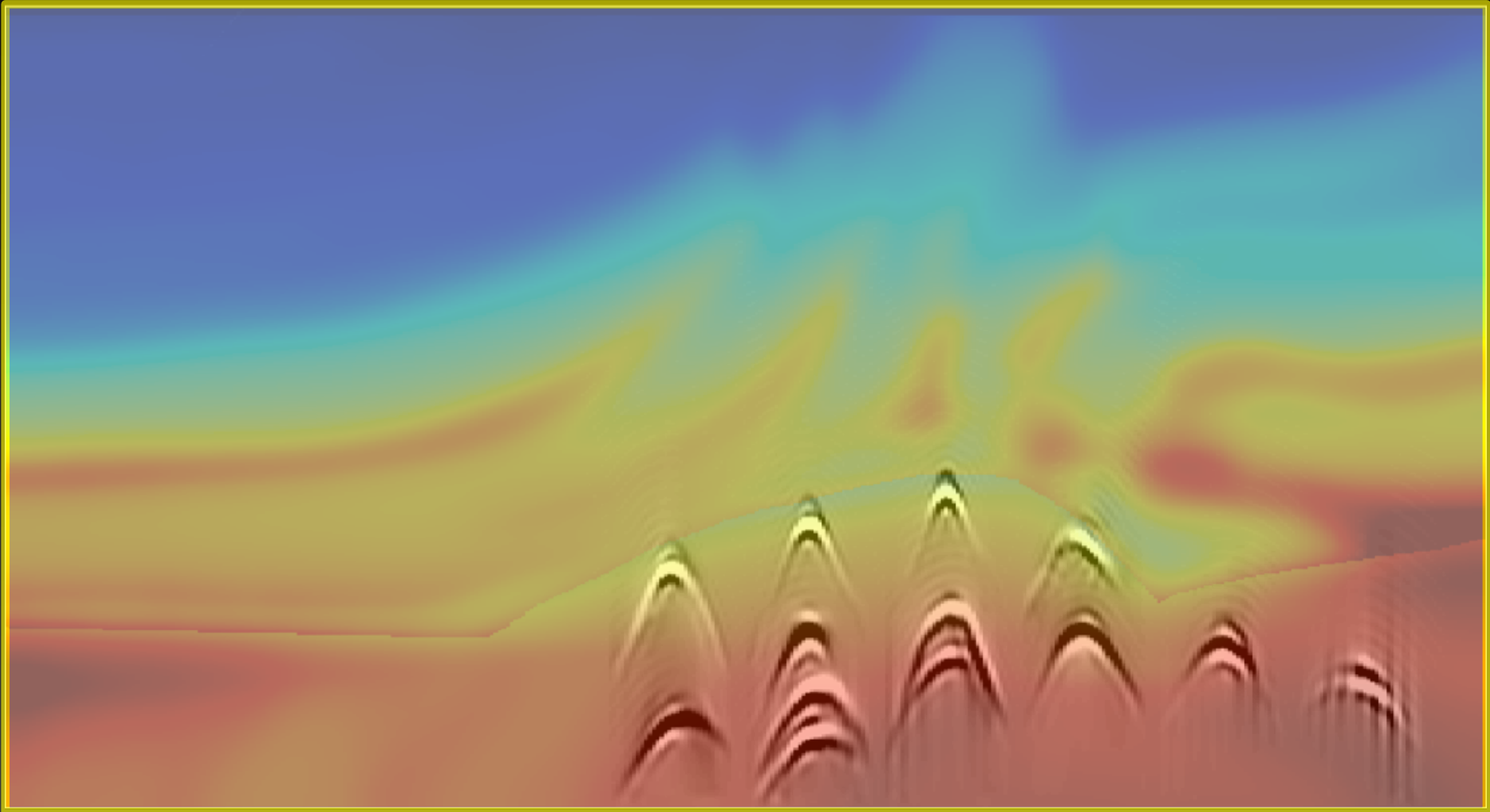
distância

profundidade



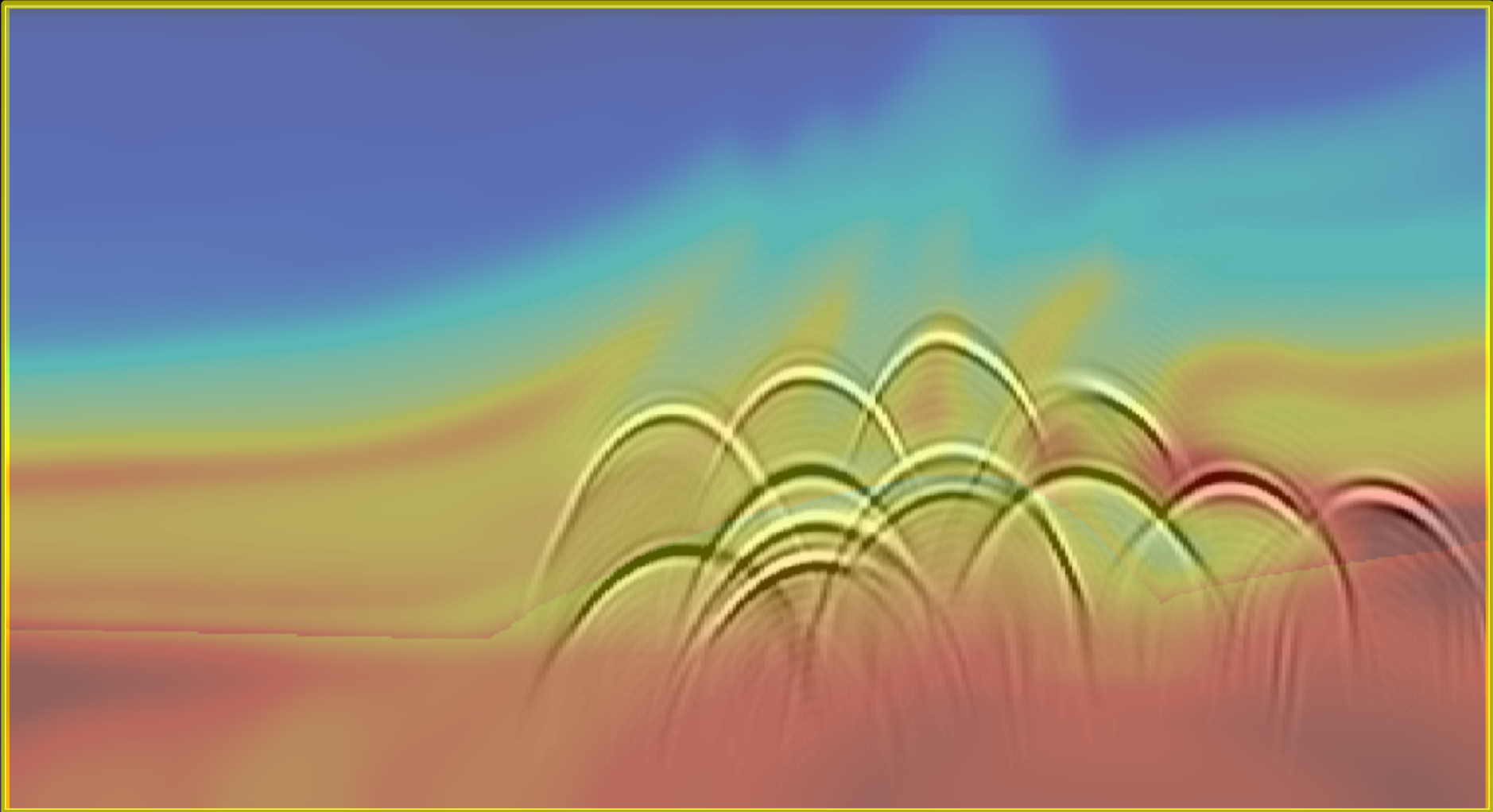
distância

profundidade



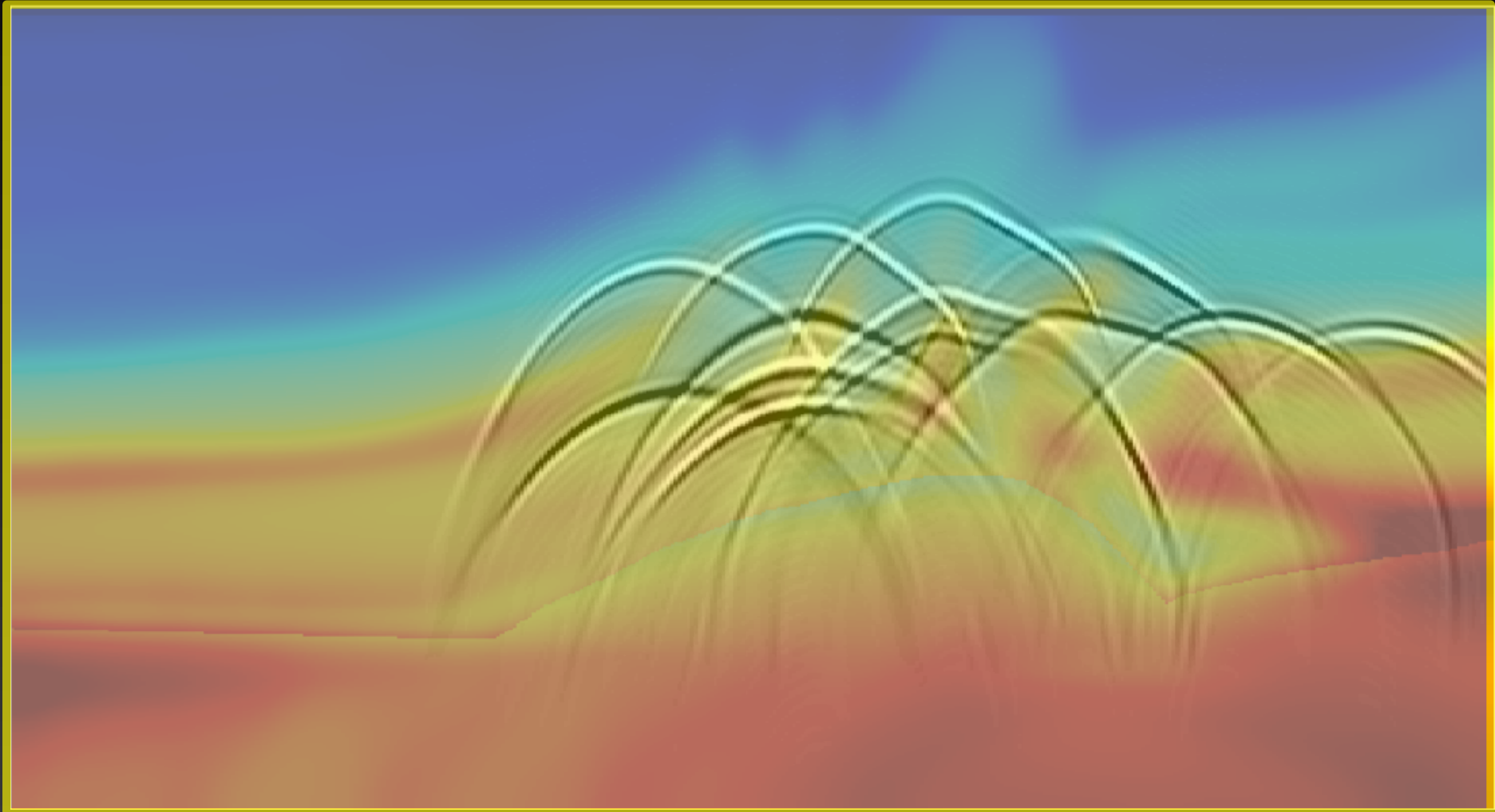
distância

profundidade



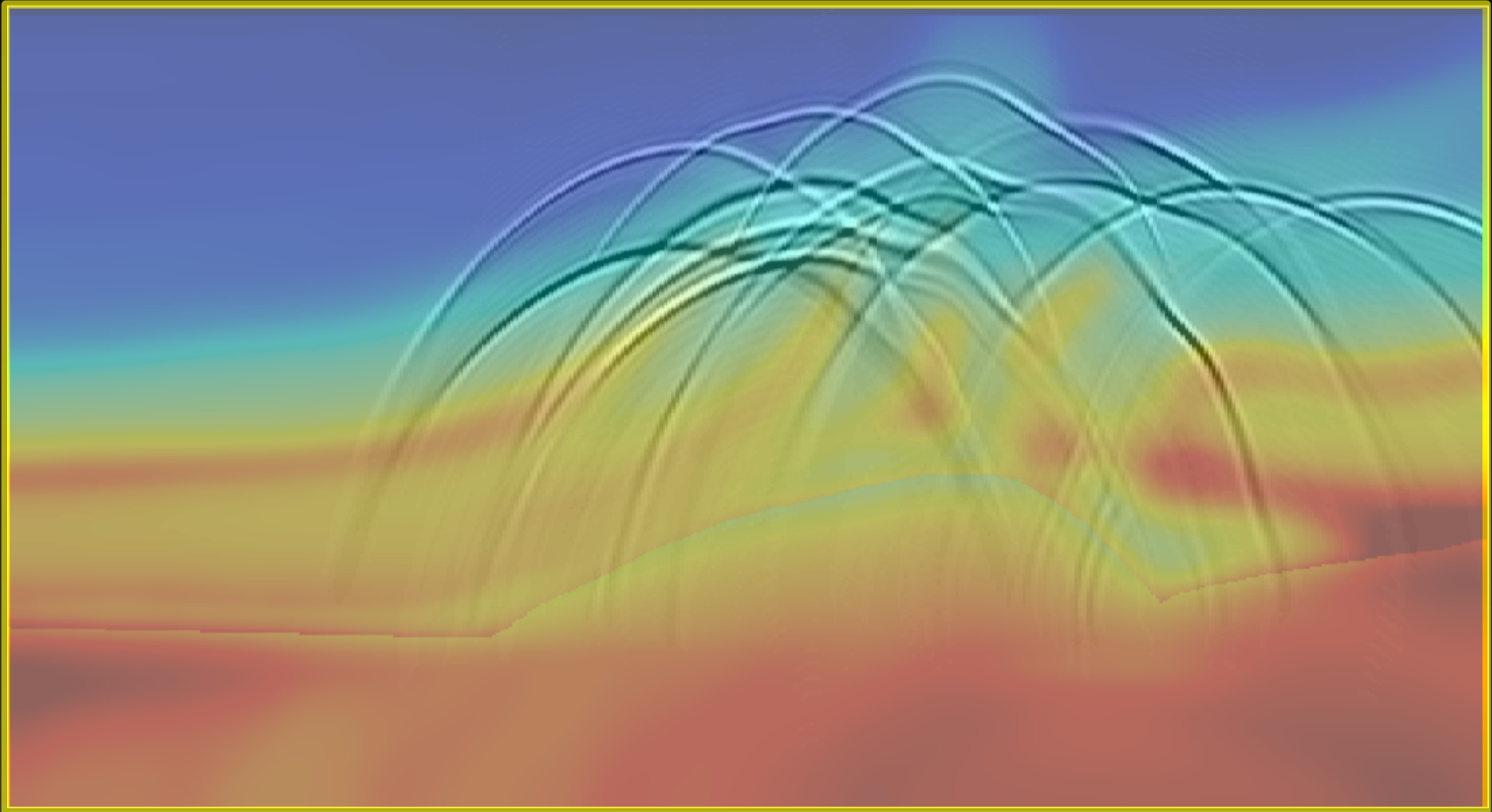
distância

profundidade



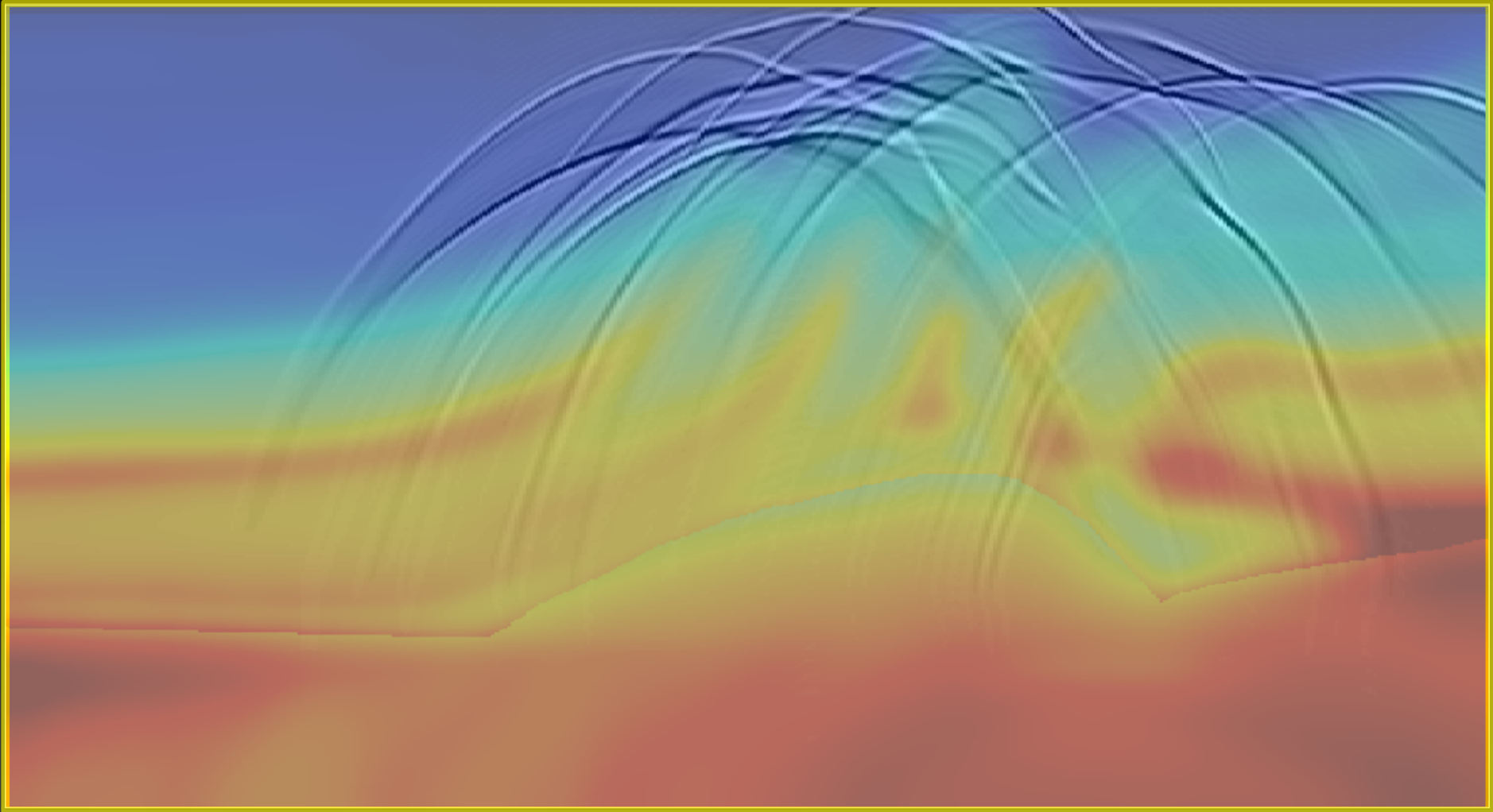
distância

profundidade



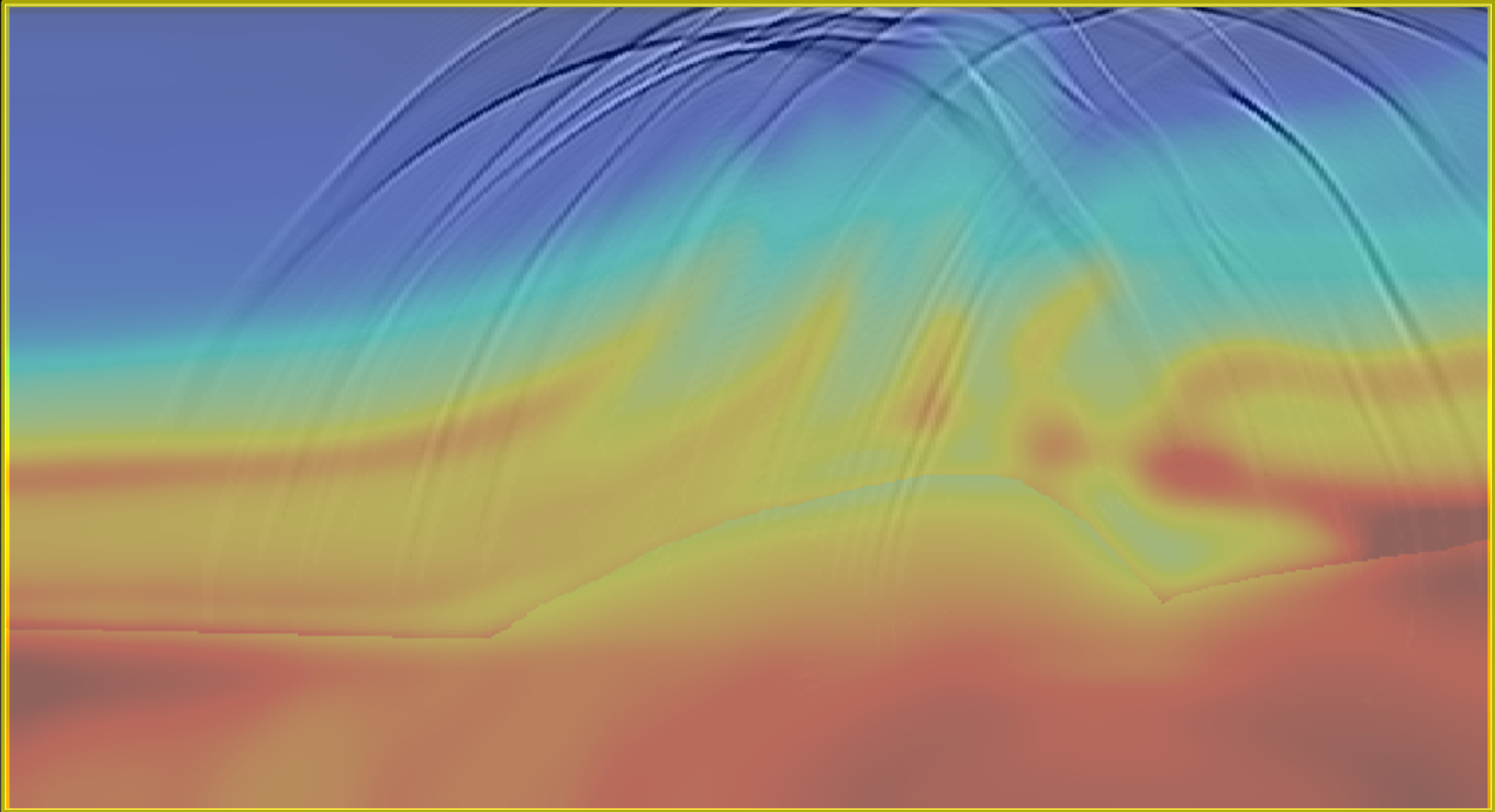
distância

profundidade

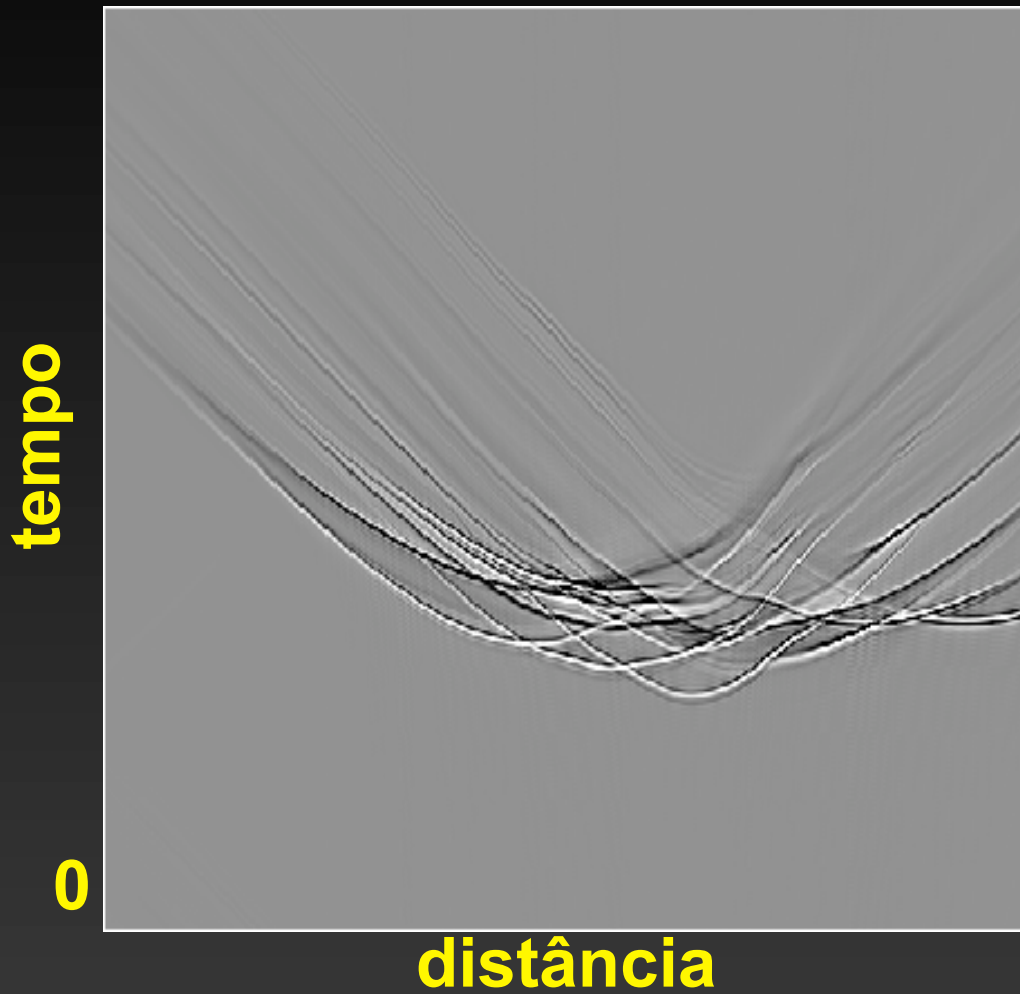


distância

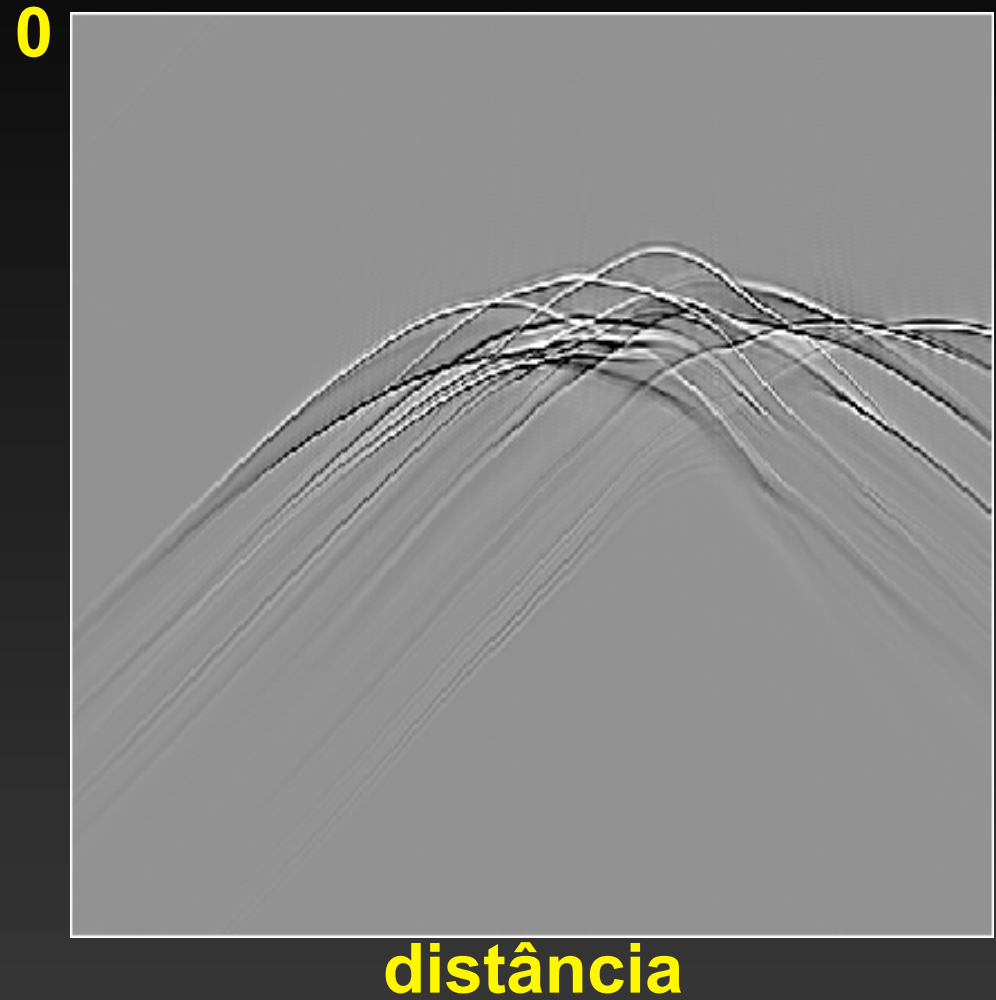
profundidade



Upgoing

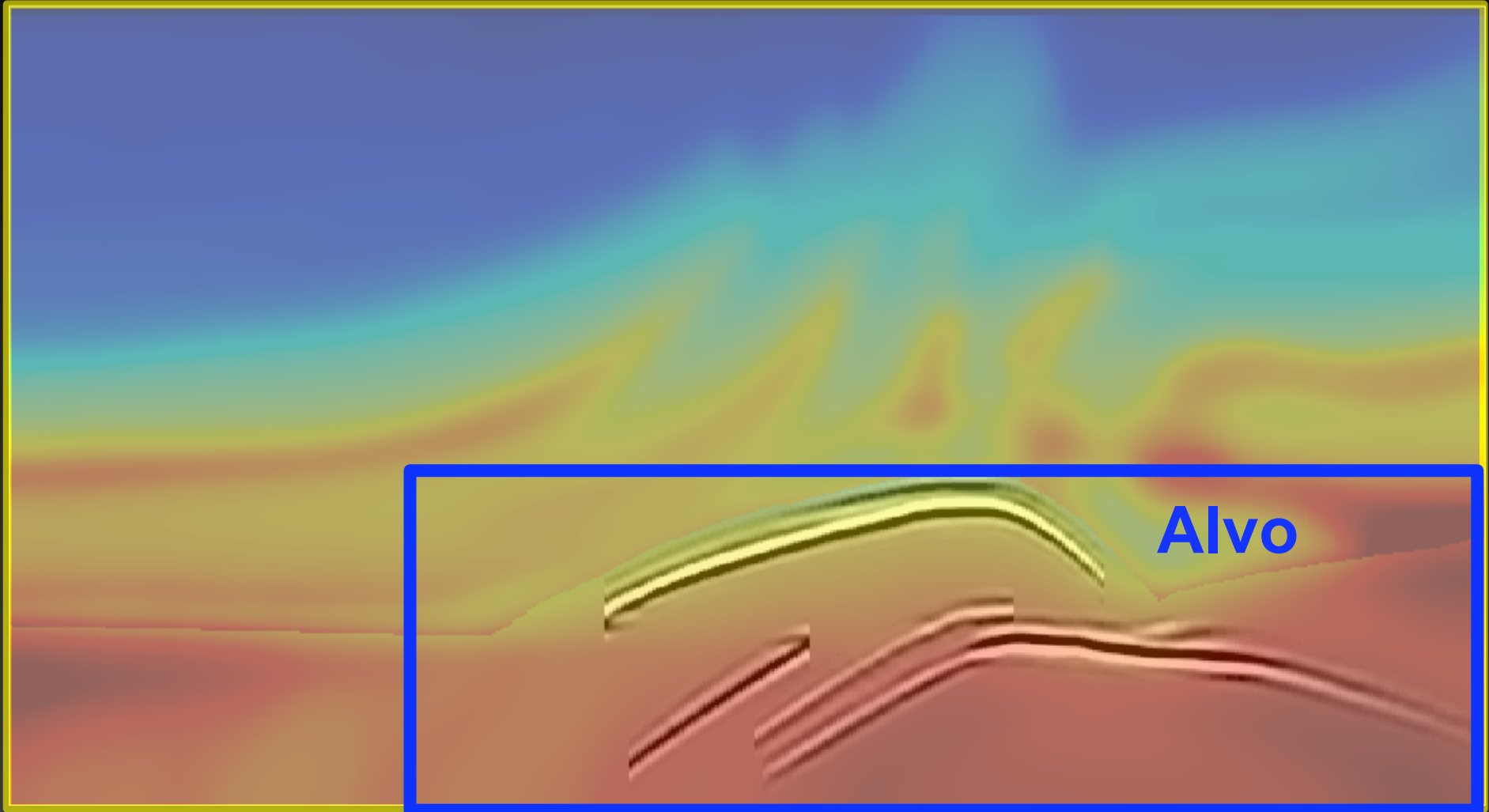


Downgoing



distância

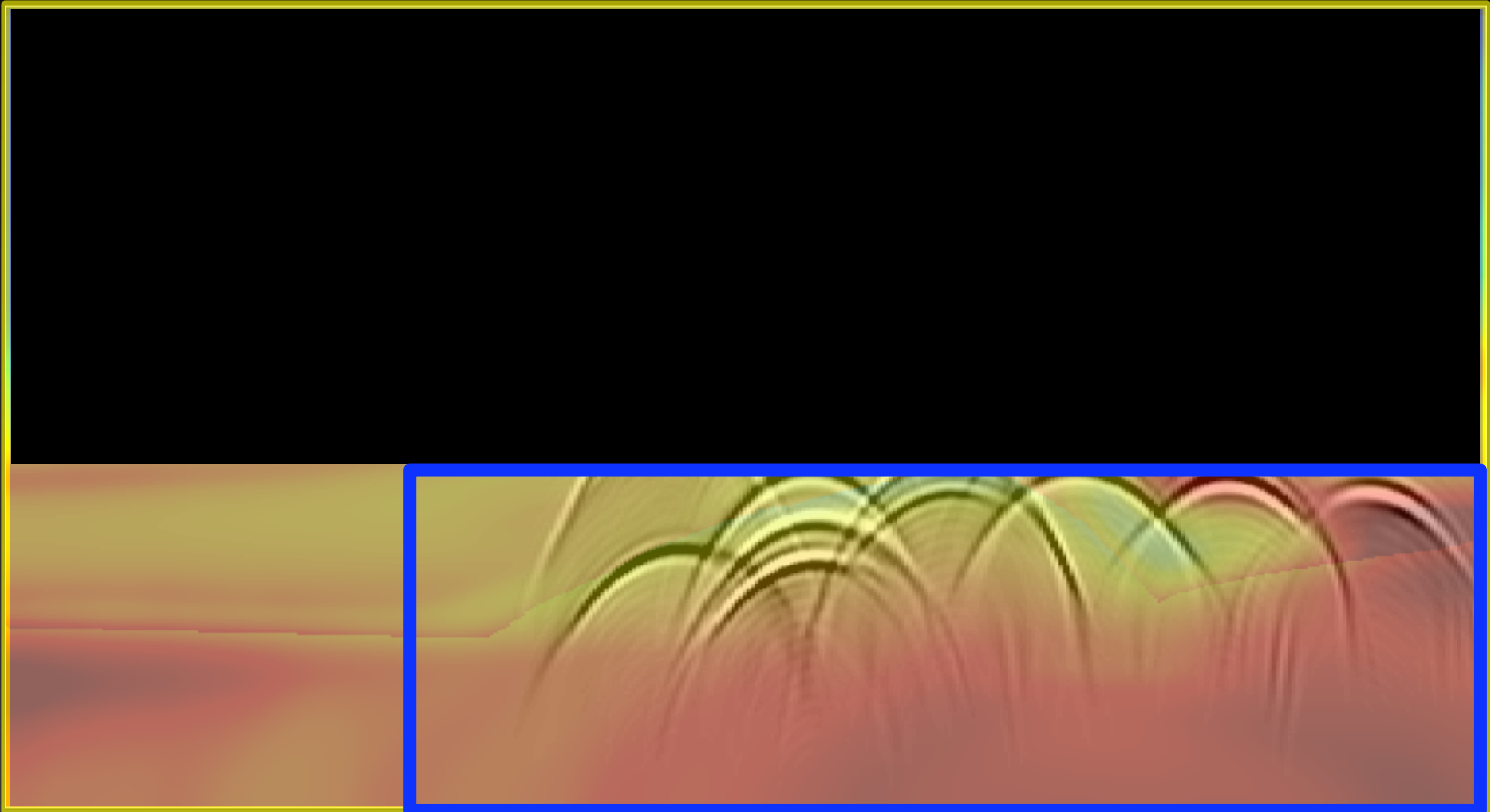
profundidade



Campos de onda até o topo do alvo

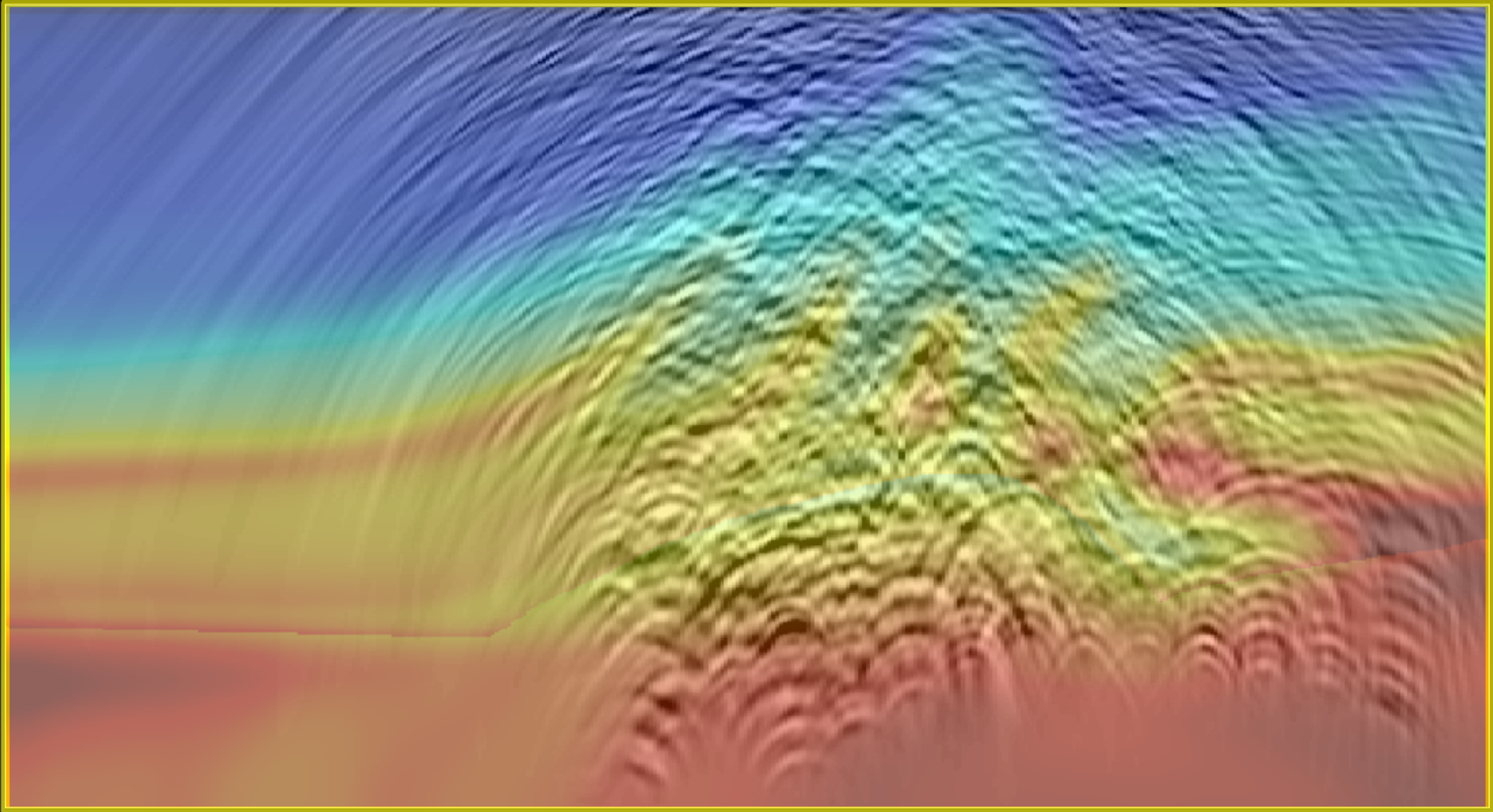
distância

profundidade



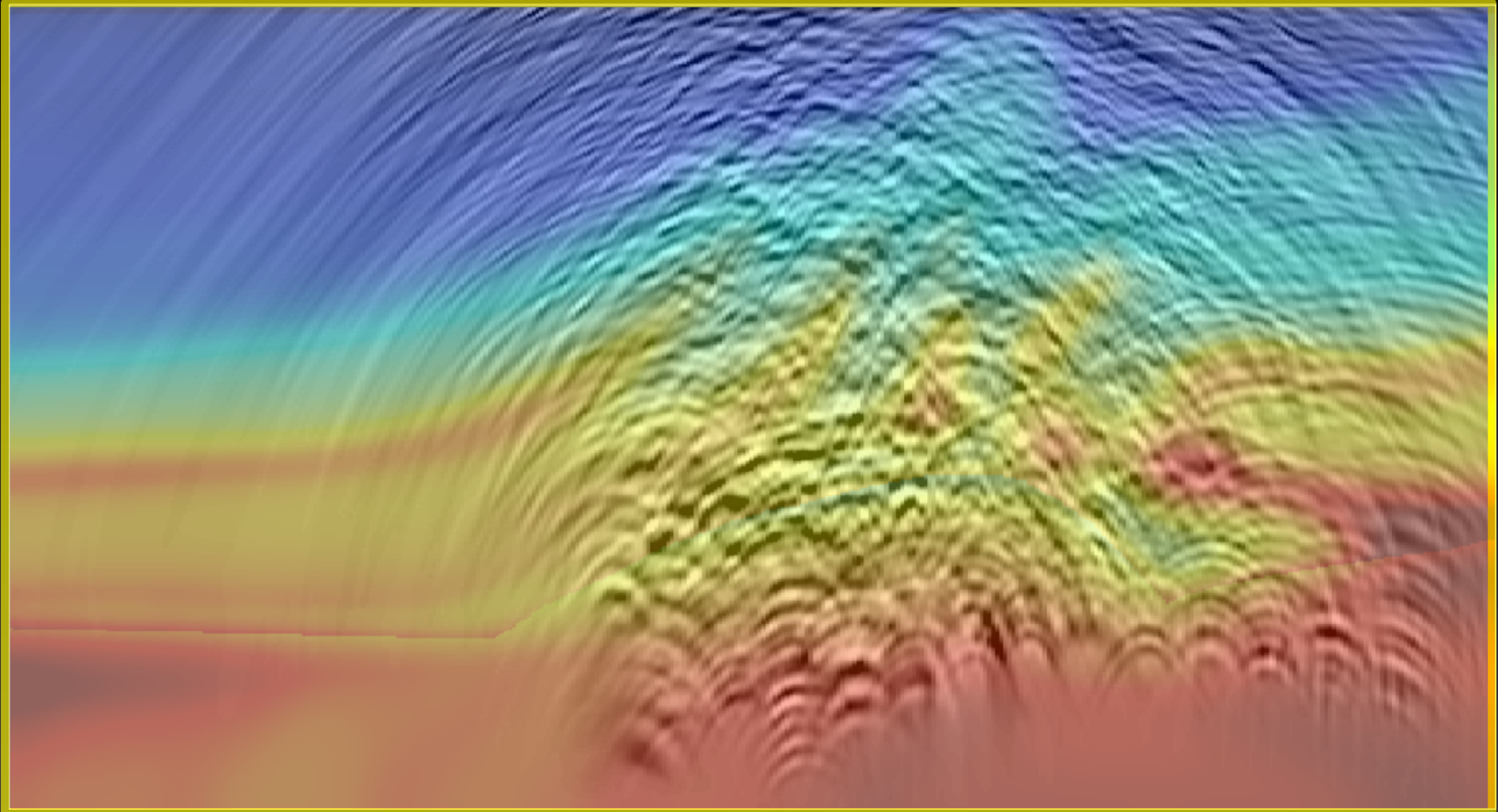
distância

profundidade



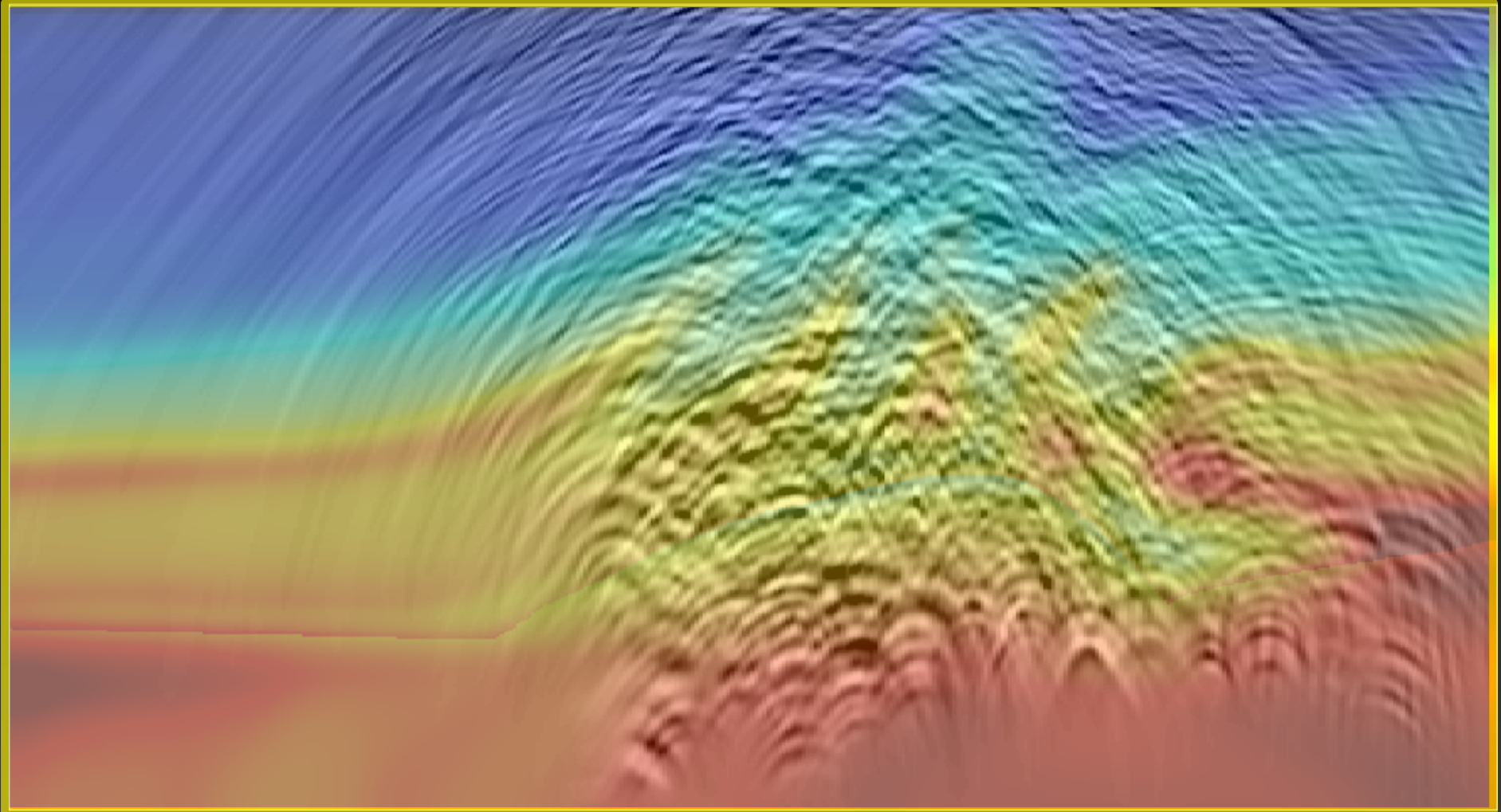
distância

profundidade



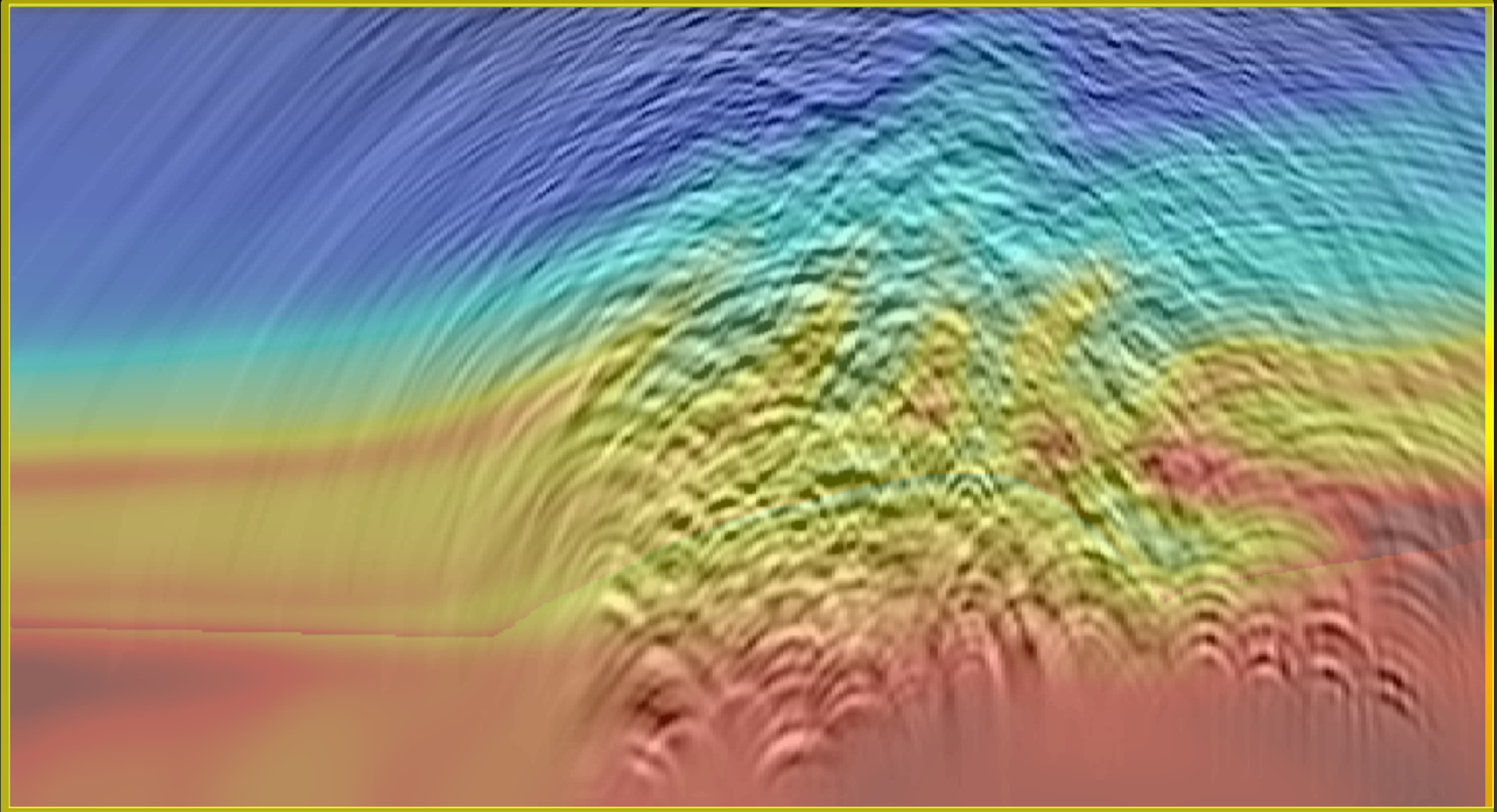
distância

profundidade



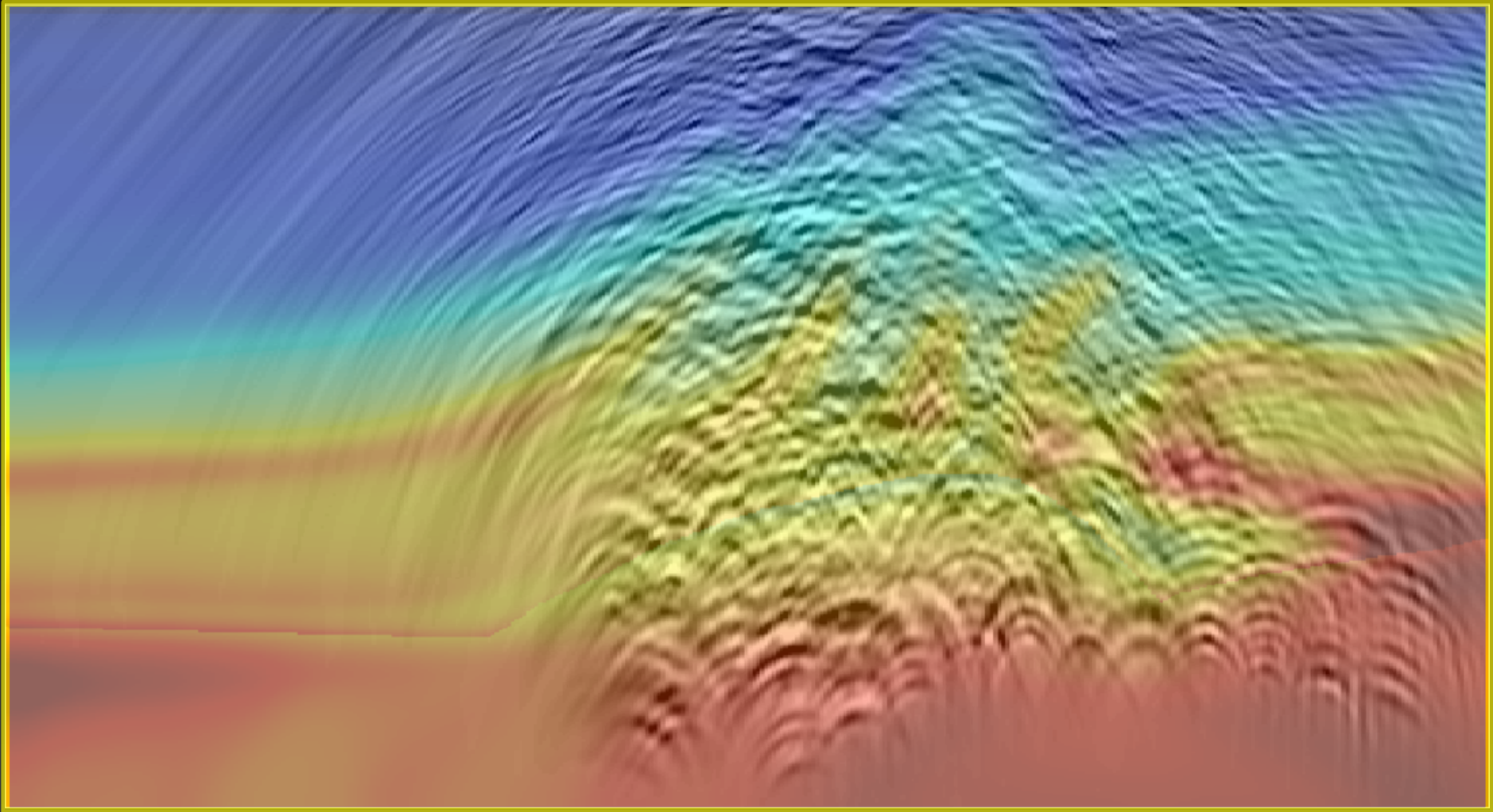
distância

profundidade

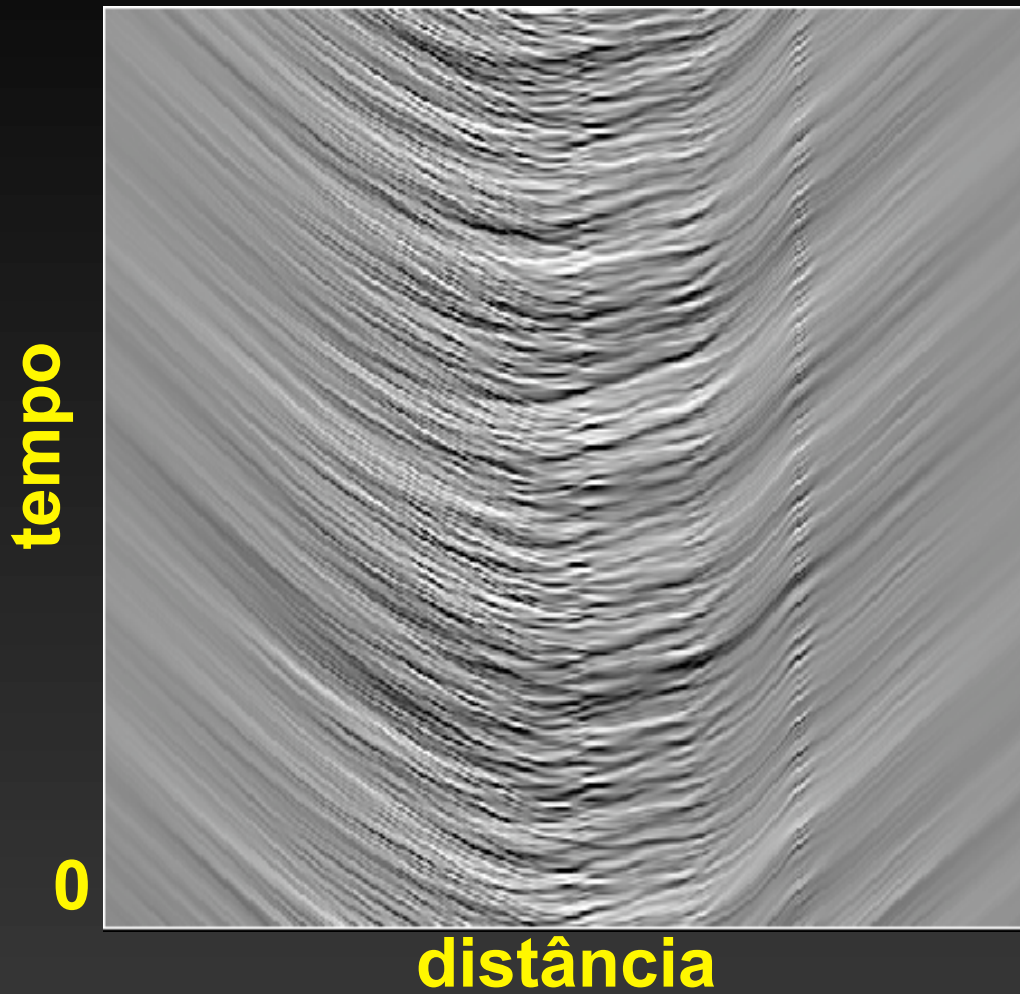


distância

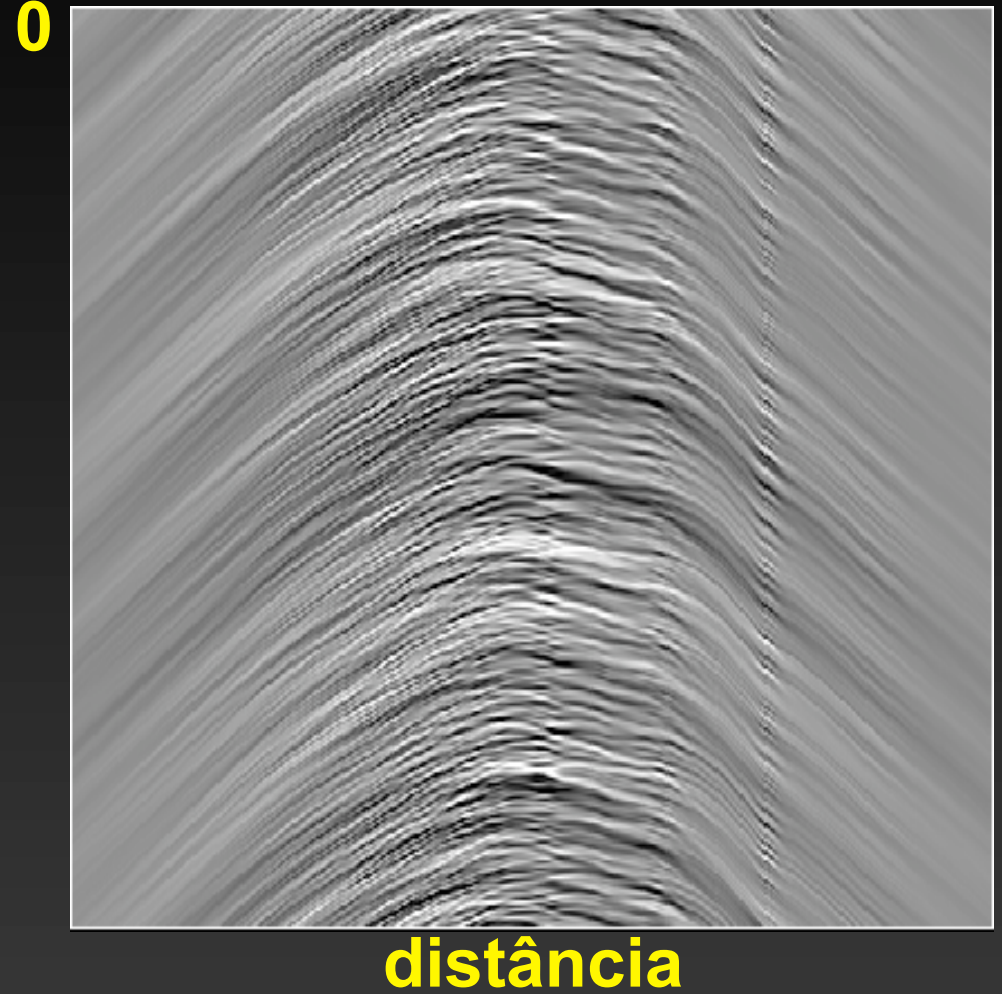
profundidade



Upgoing

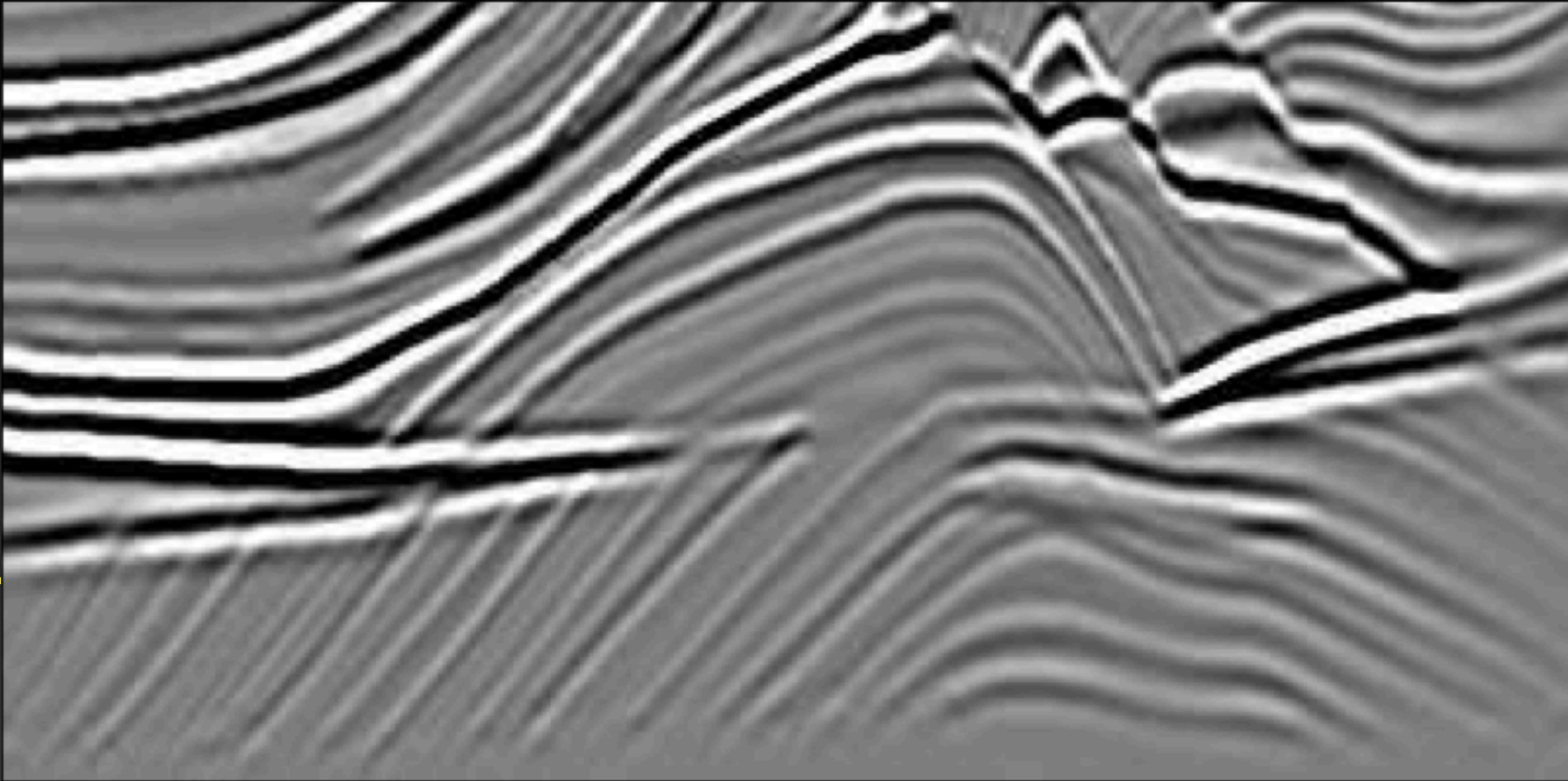


Downgoing



distância

profundidade



distância

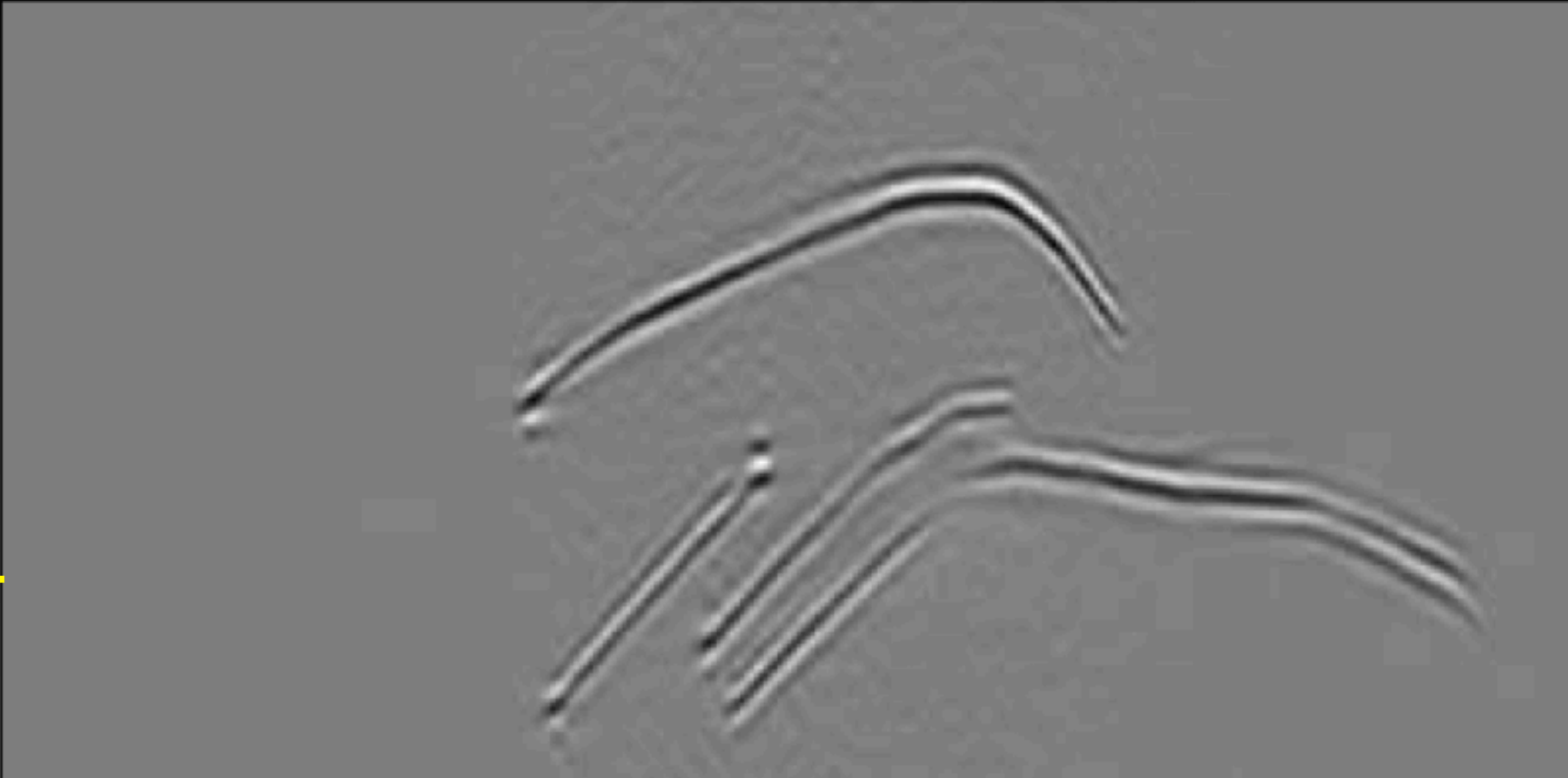
profundidade



Migração de 11 tiros areais

distância

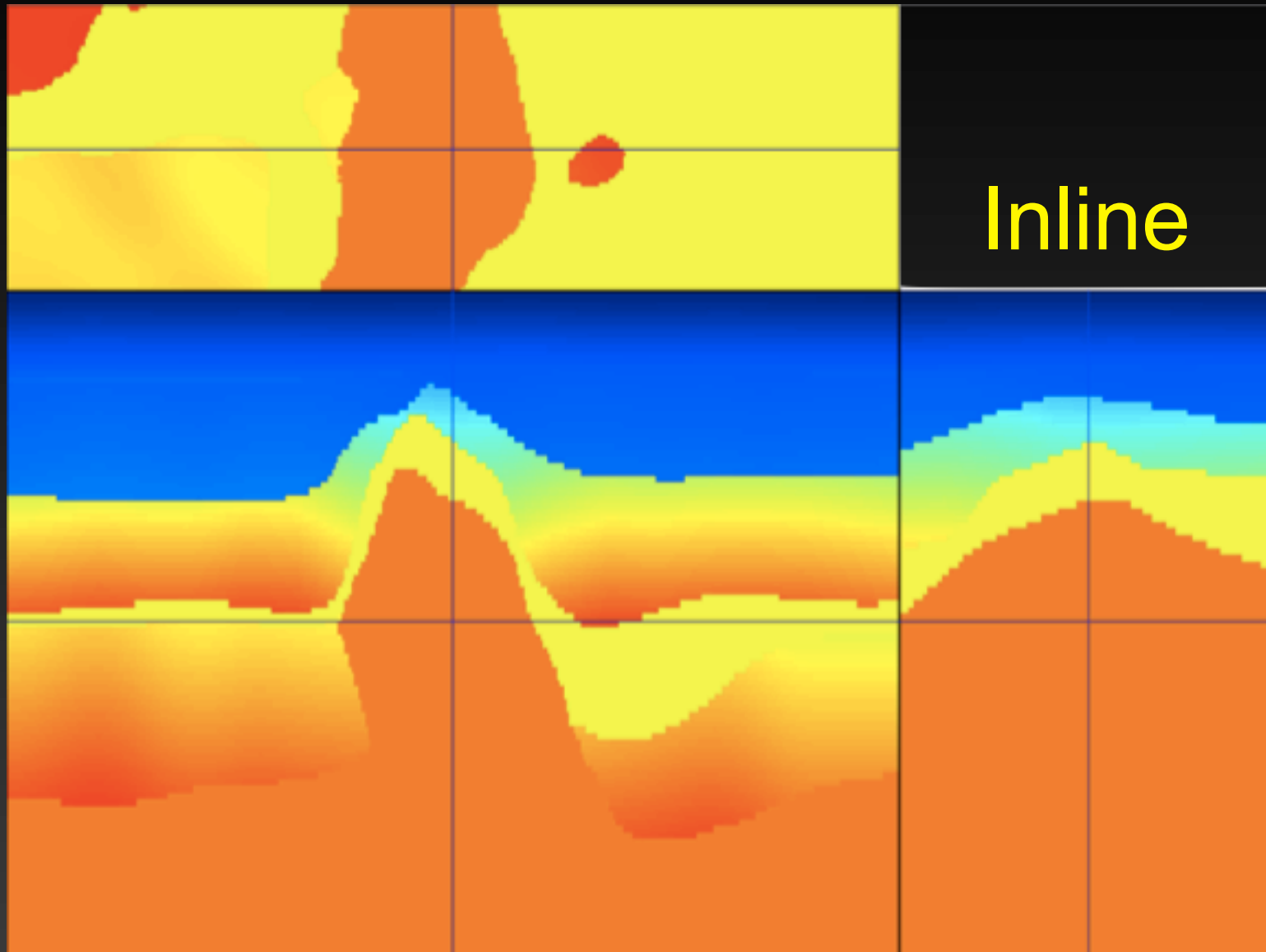
profundidade



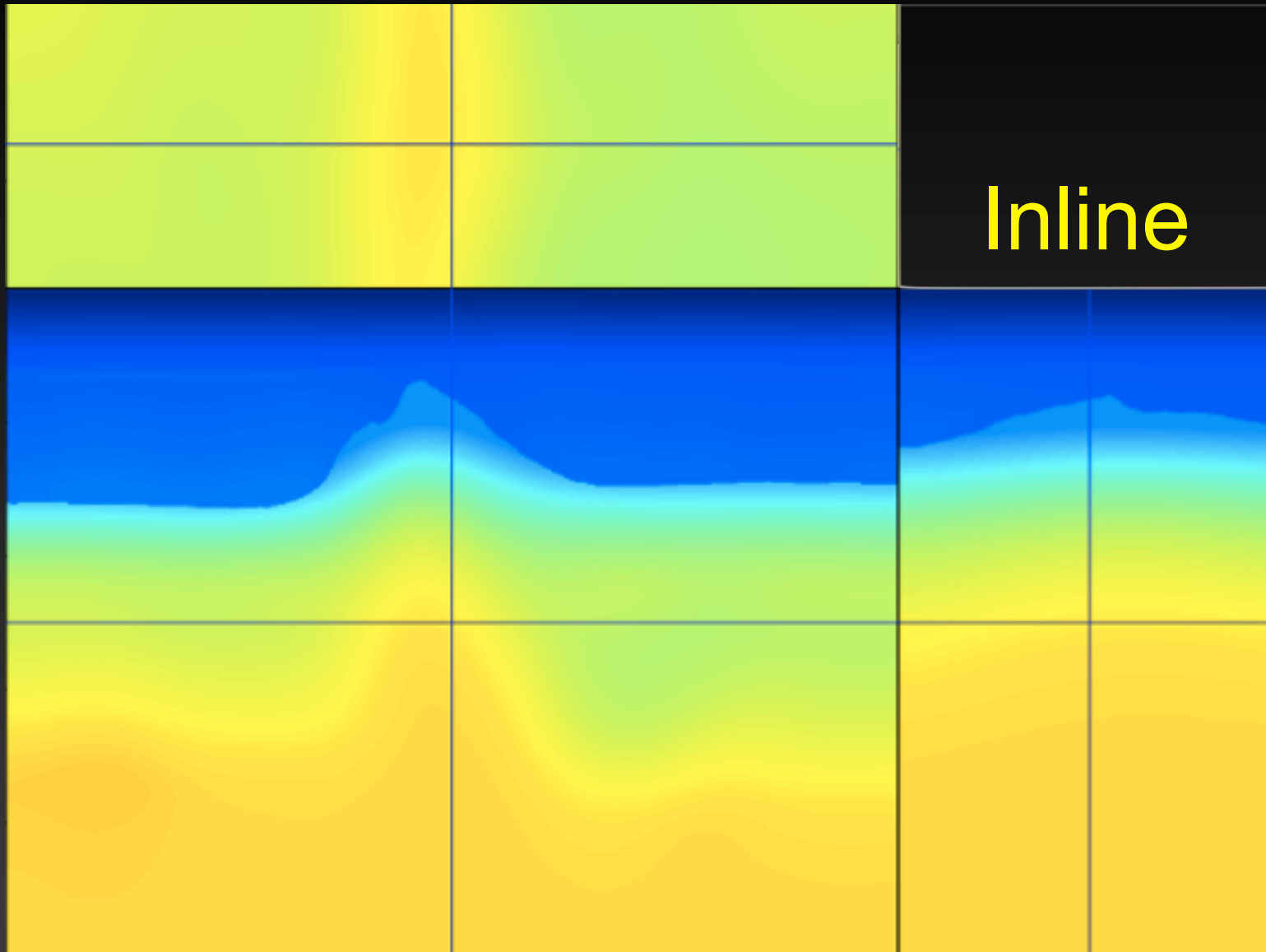
- Modelagem
- Migração
- AVM com tomografia de campos de onda (AVMTO)
- Custo e flexibilidade
- Exemplos

- **Offset máximo: 3.6 km**
- **30 dados areaais com phase-encoding**
- **Forte estruturação e corpo de sal irregular**

Crossline

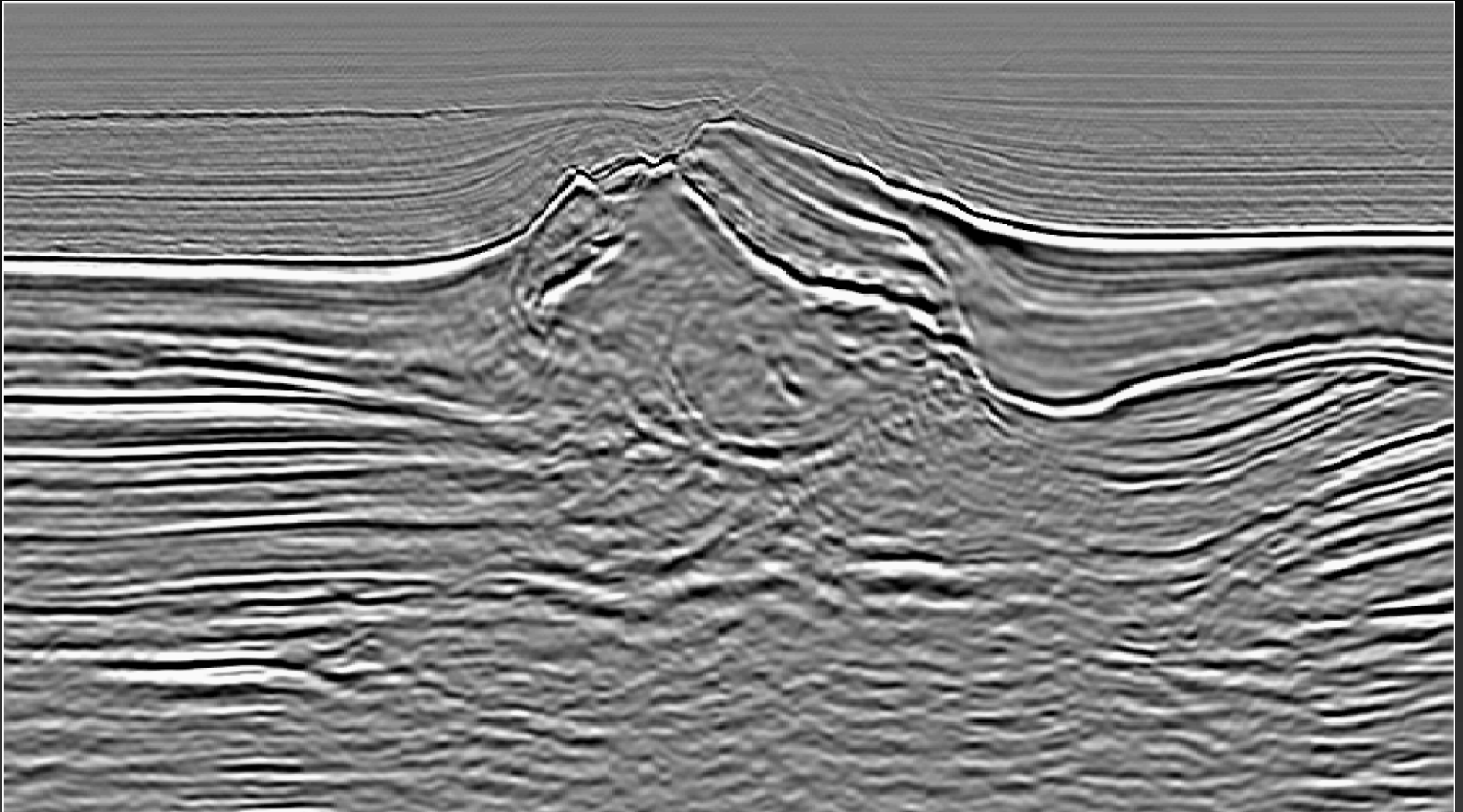


Crossline



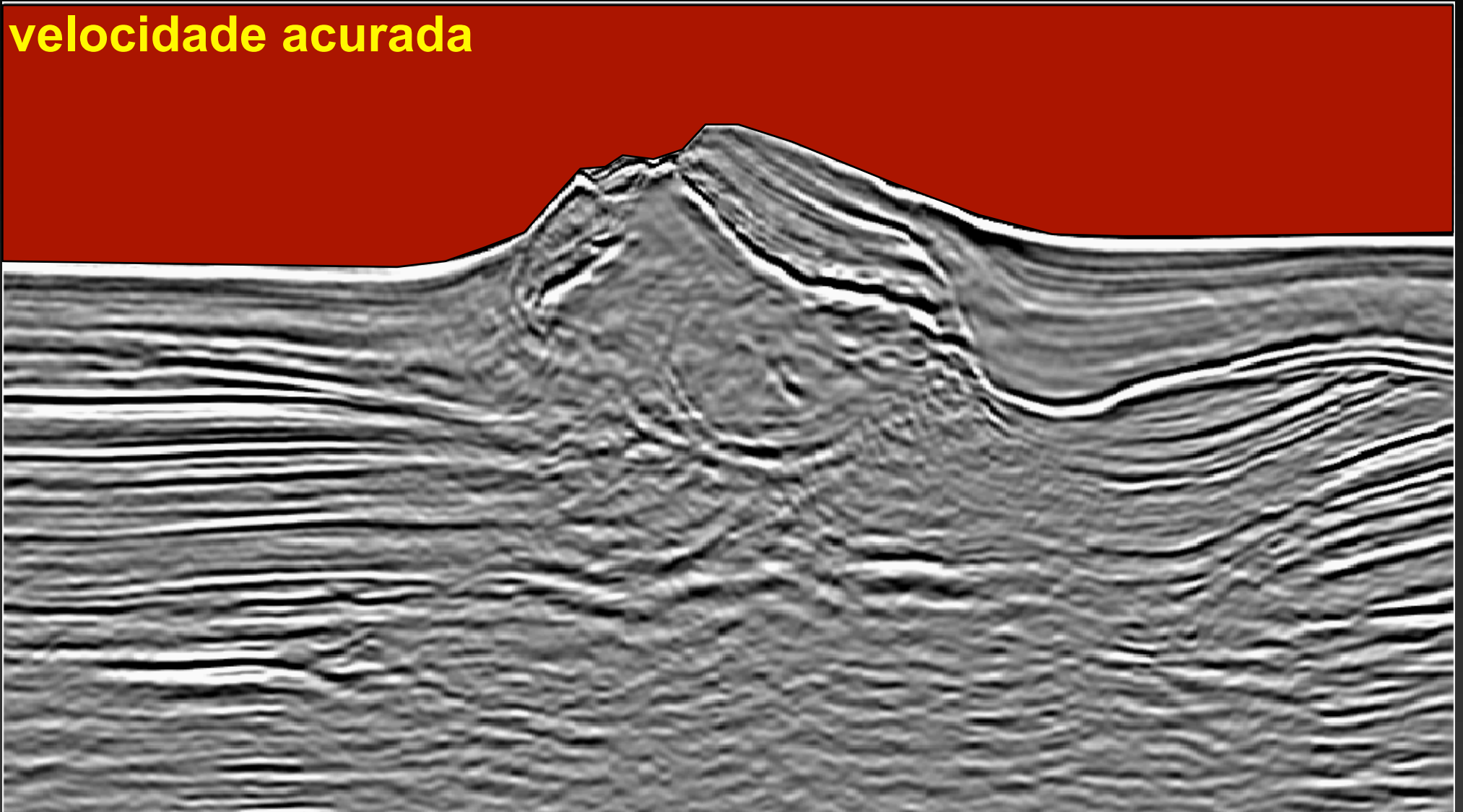
Inline

Crossline



Crossline

velocidade acurada



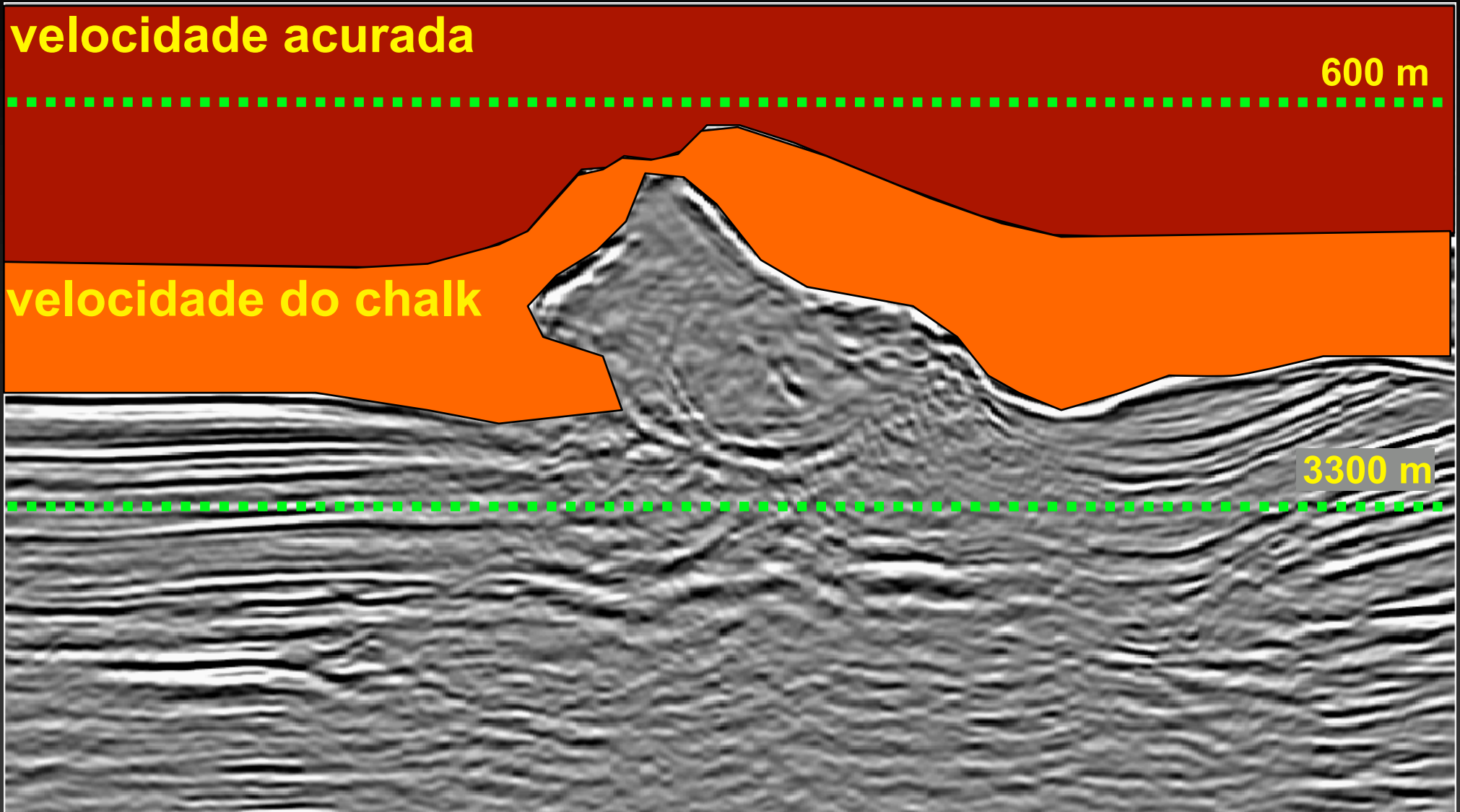
Crossline

velocidade acurada

600 m

velocidade do chalk

3300 m

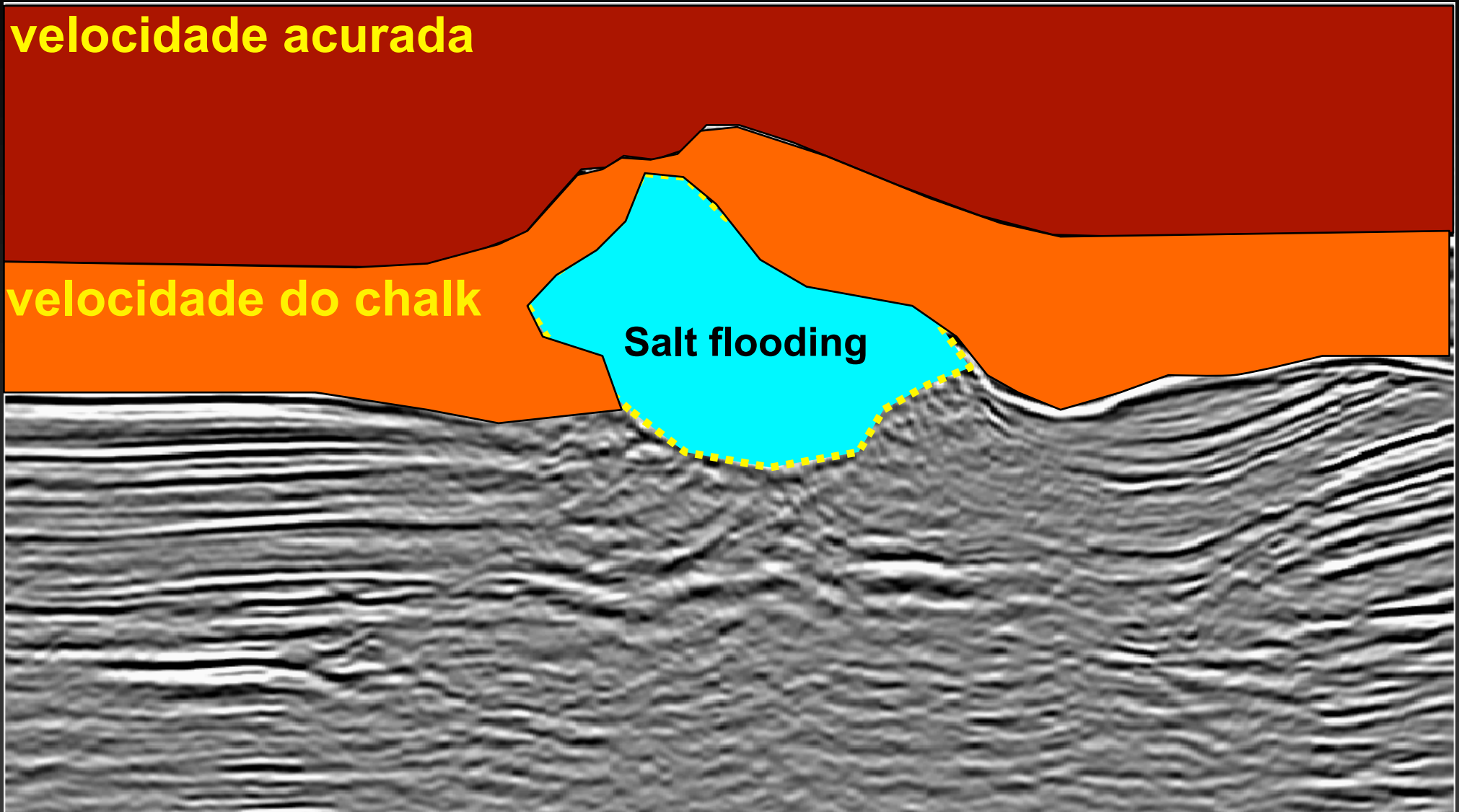


Crossline

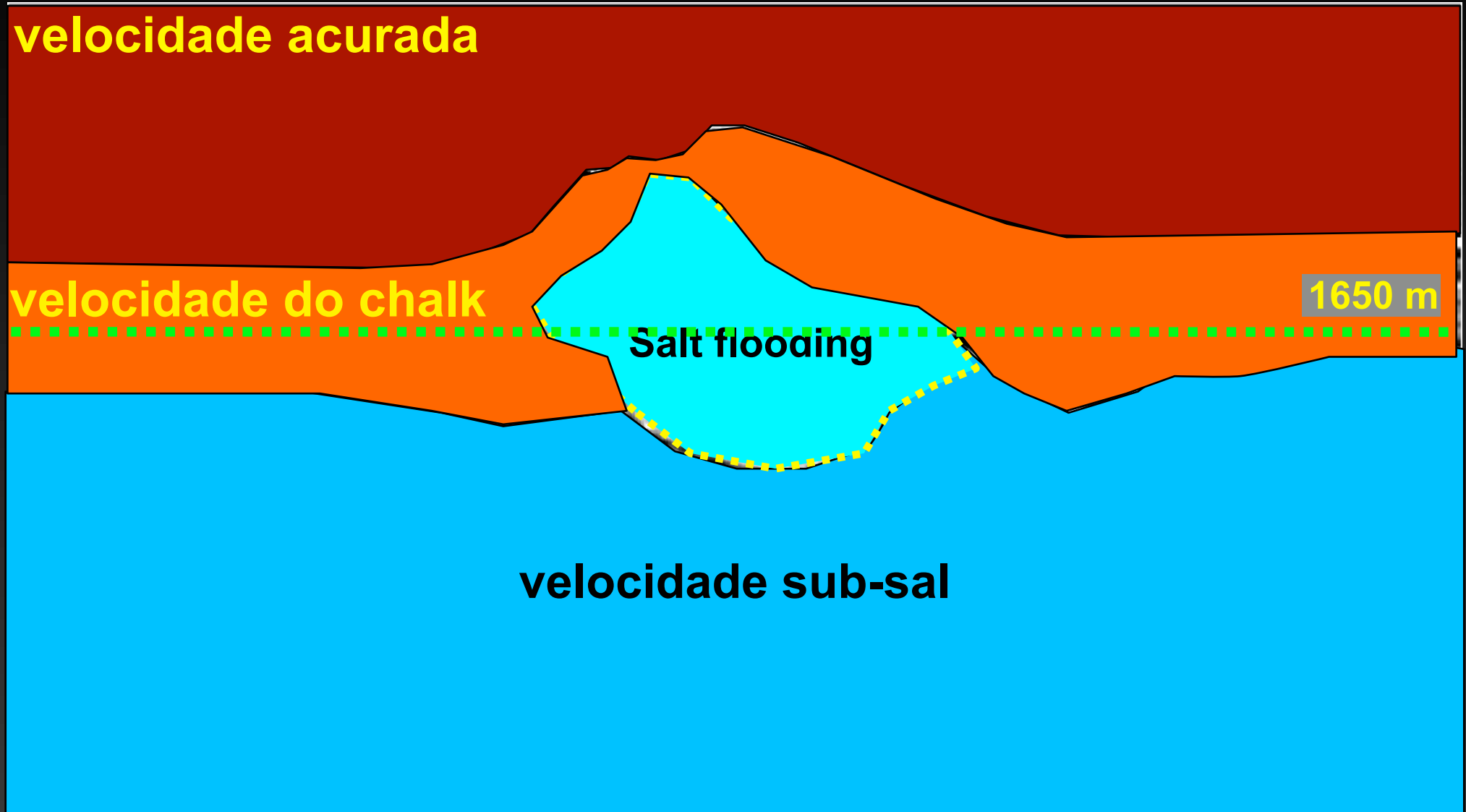
velocidade acurada

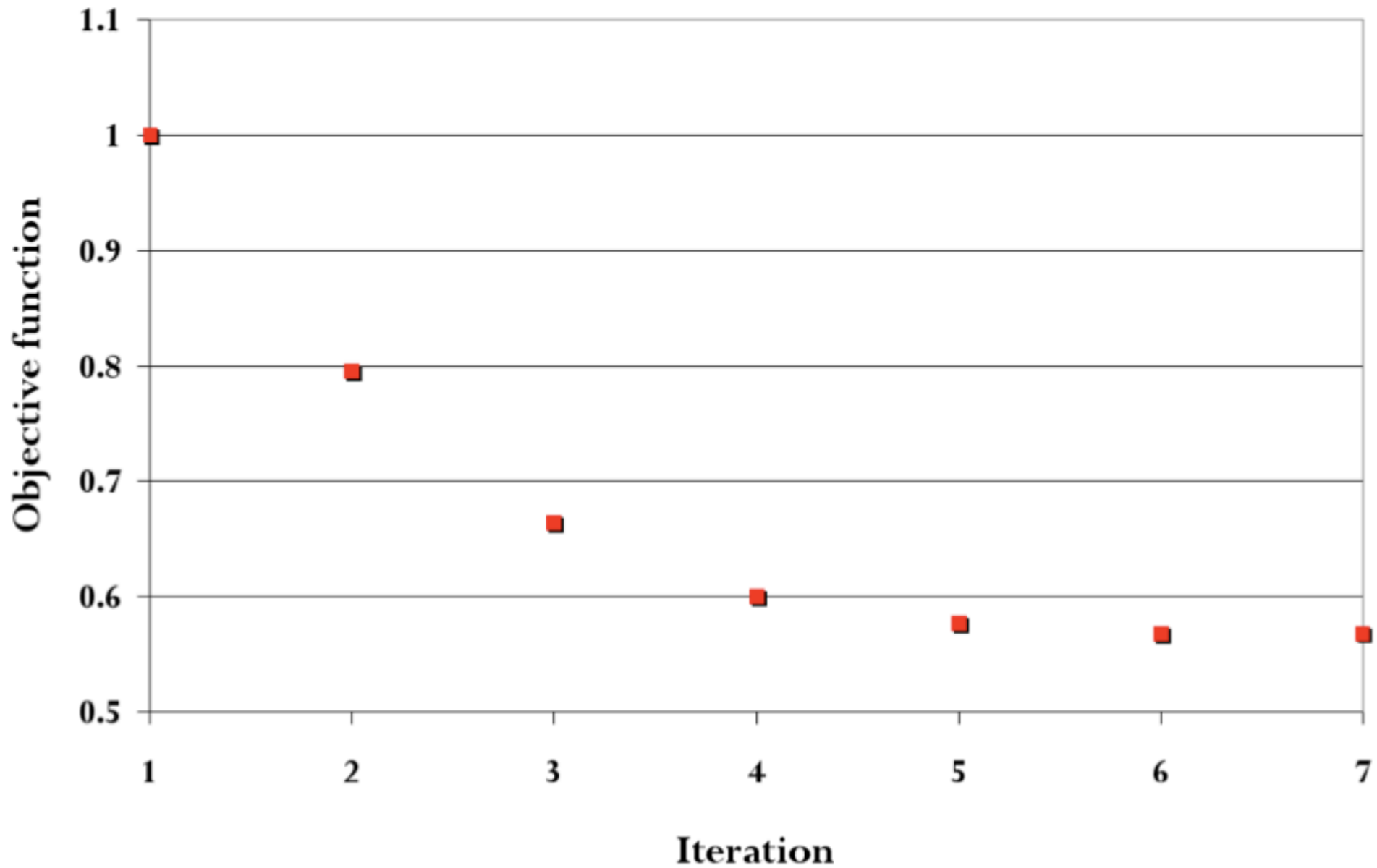
velocidade do chalk

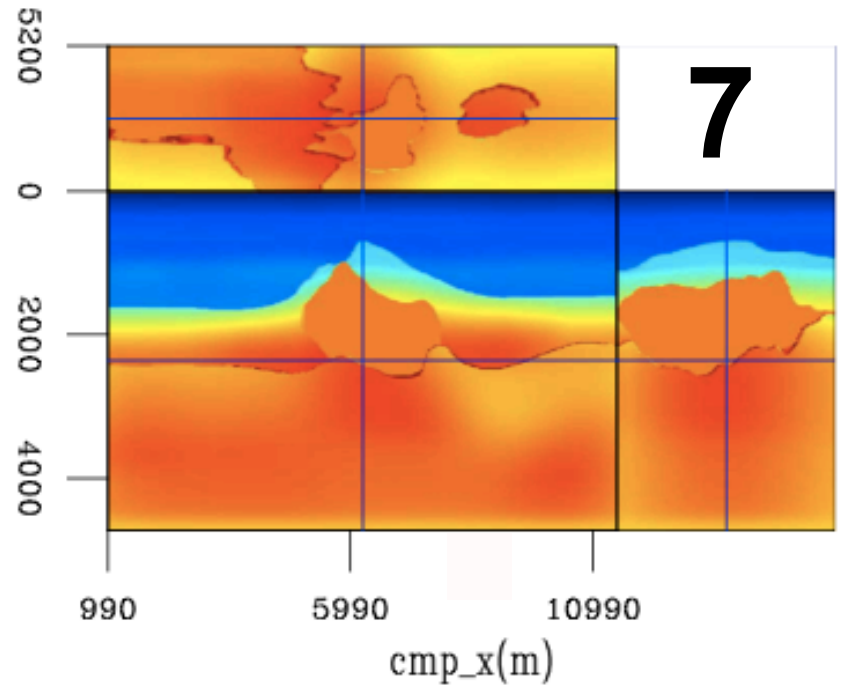
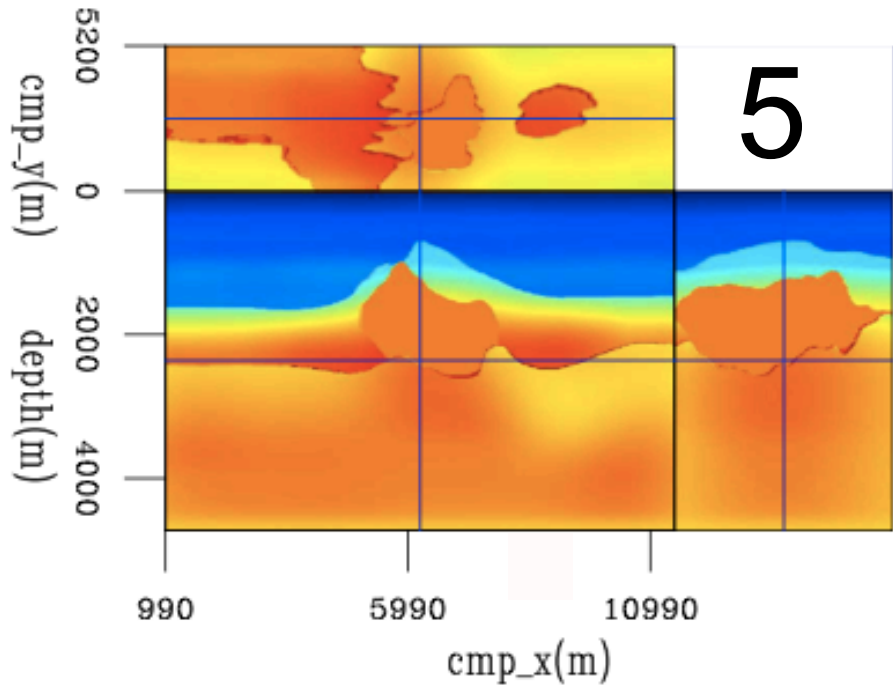
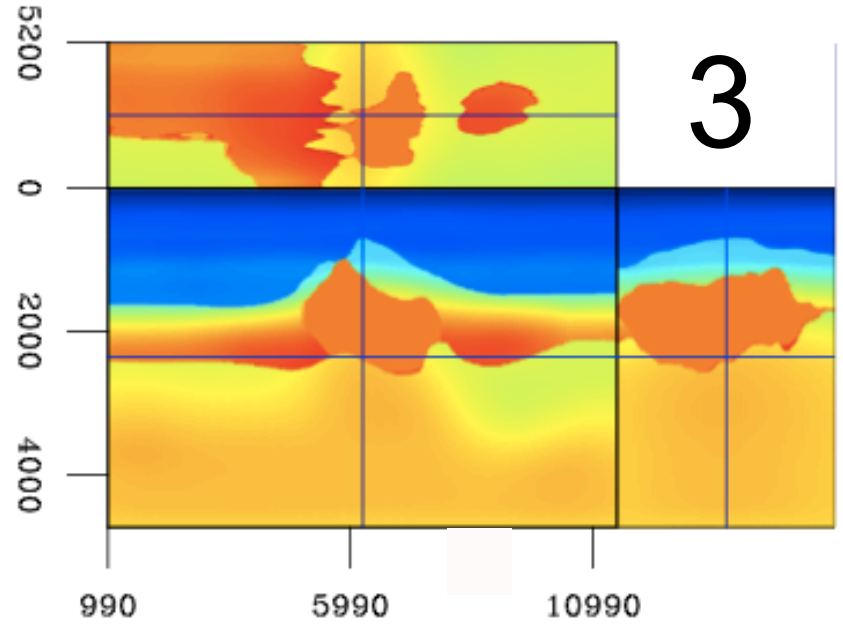
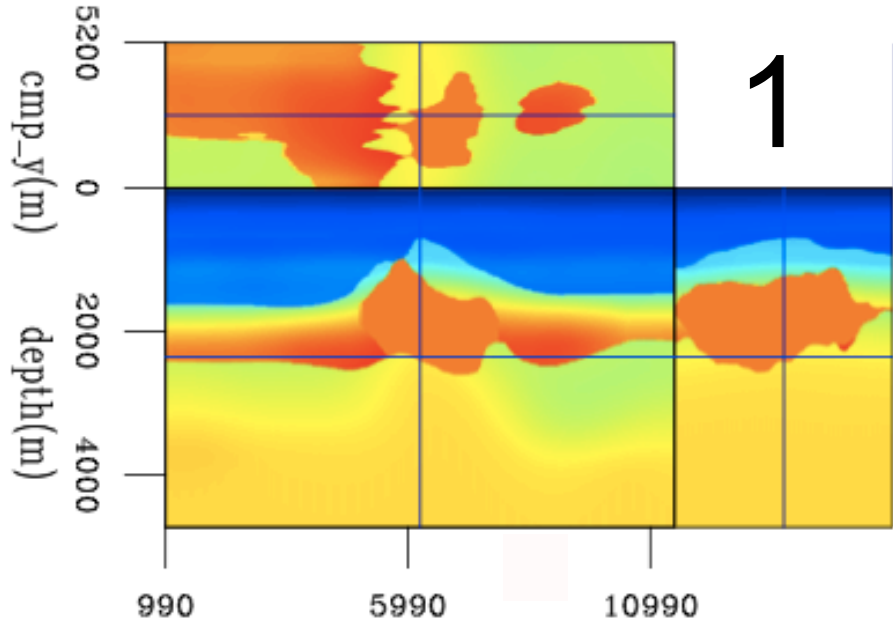
Salt flooding

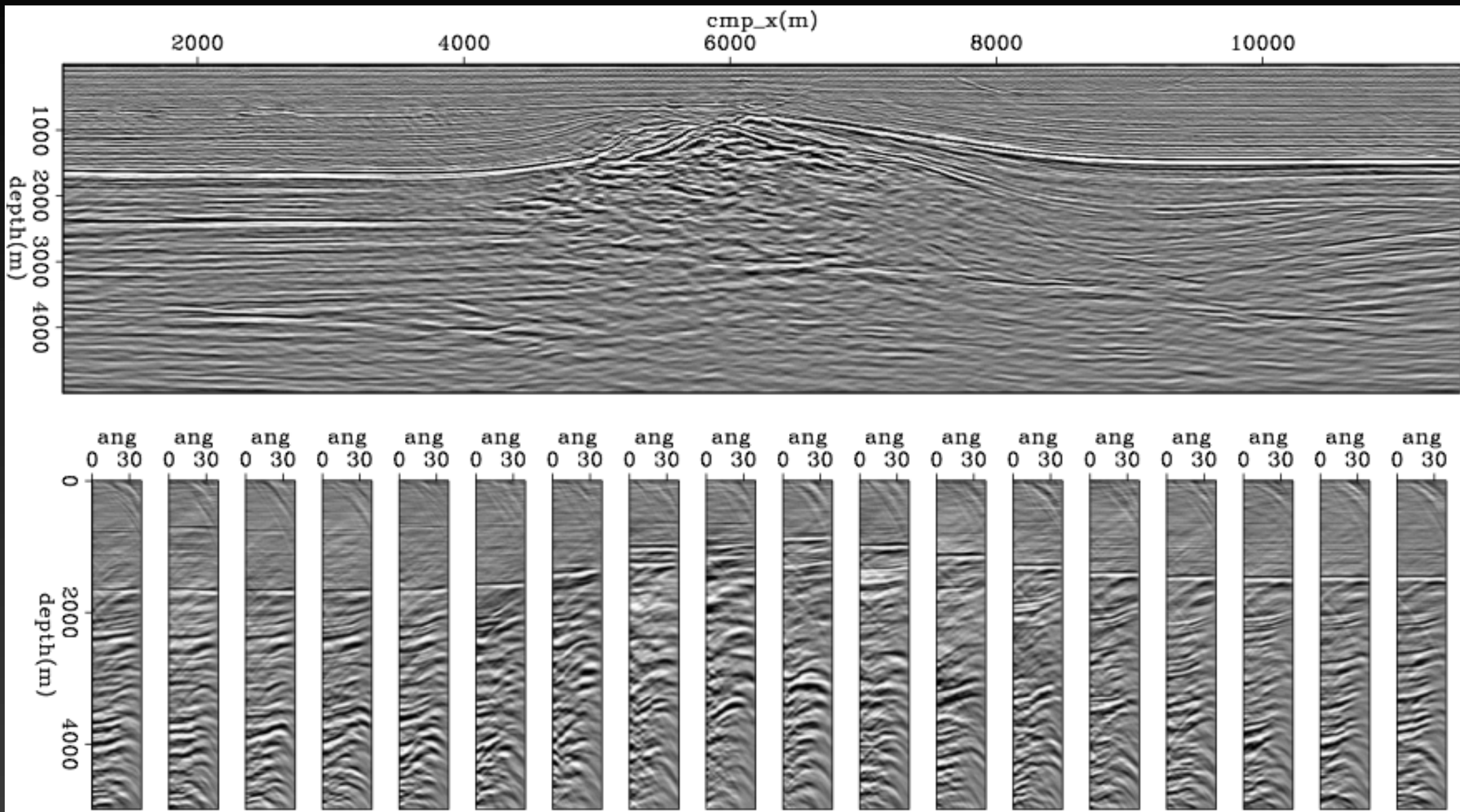


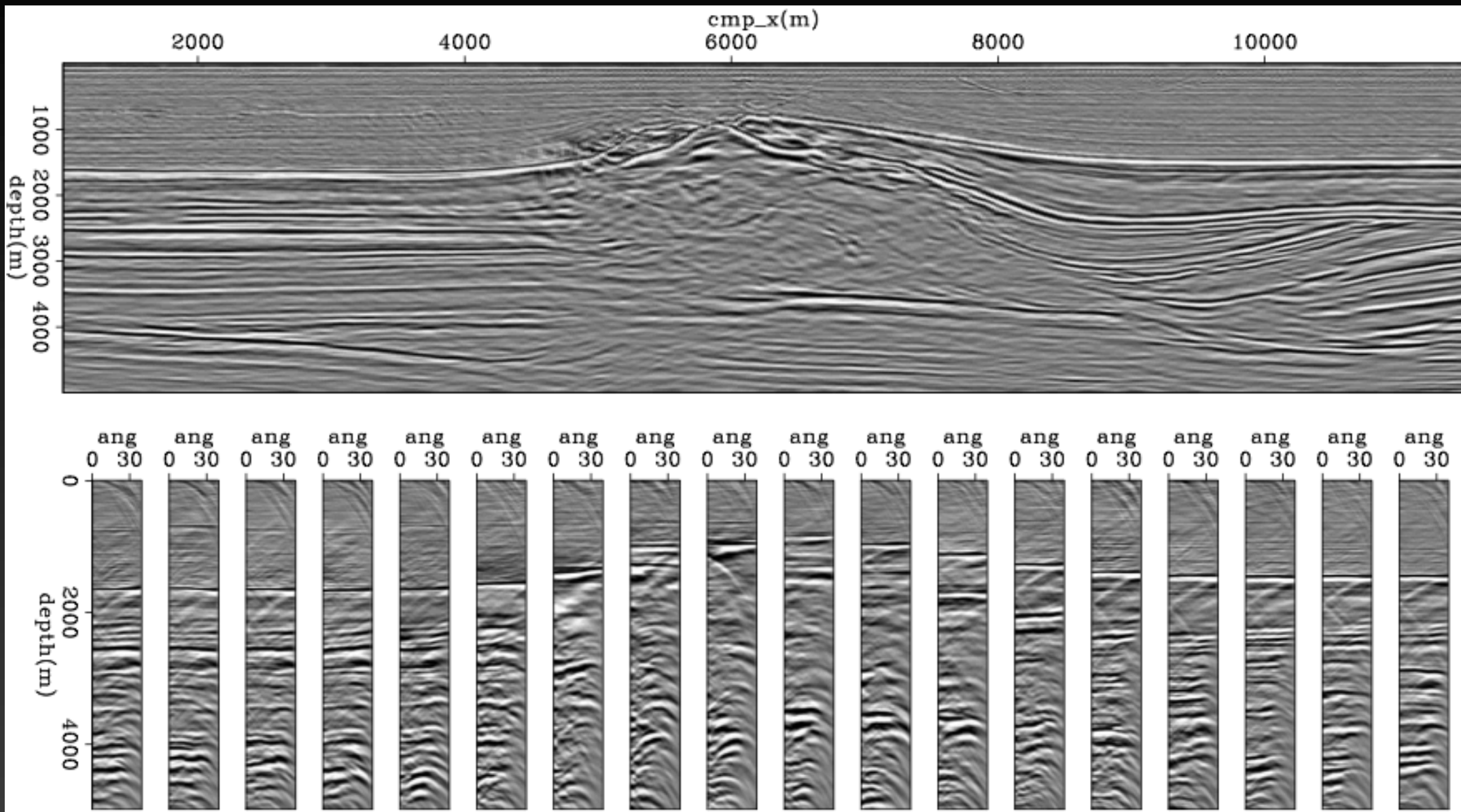
Crossline

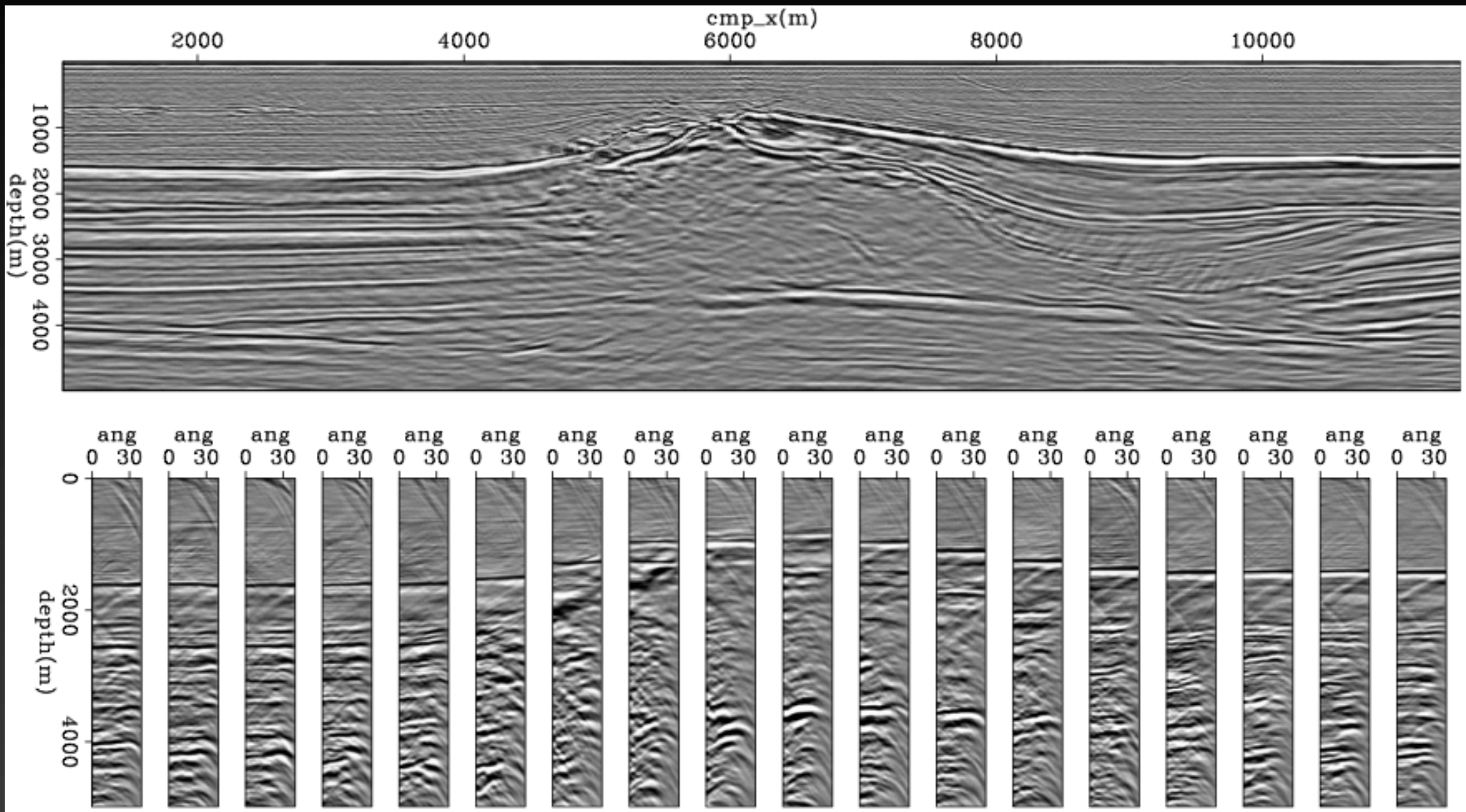




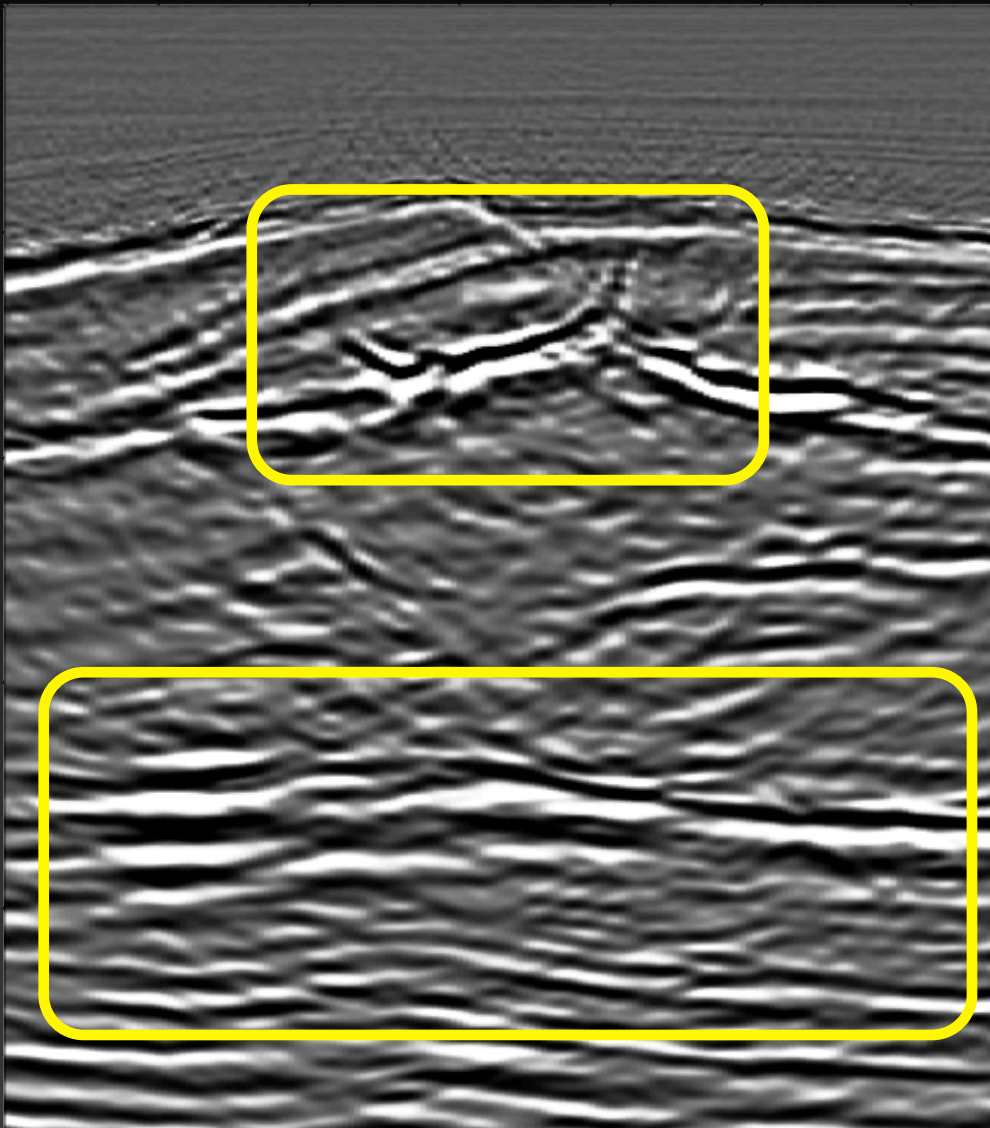




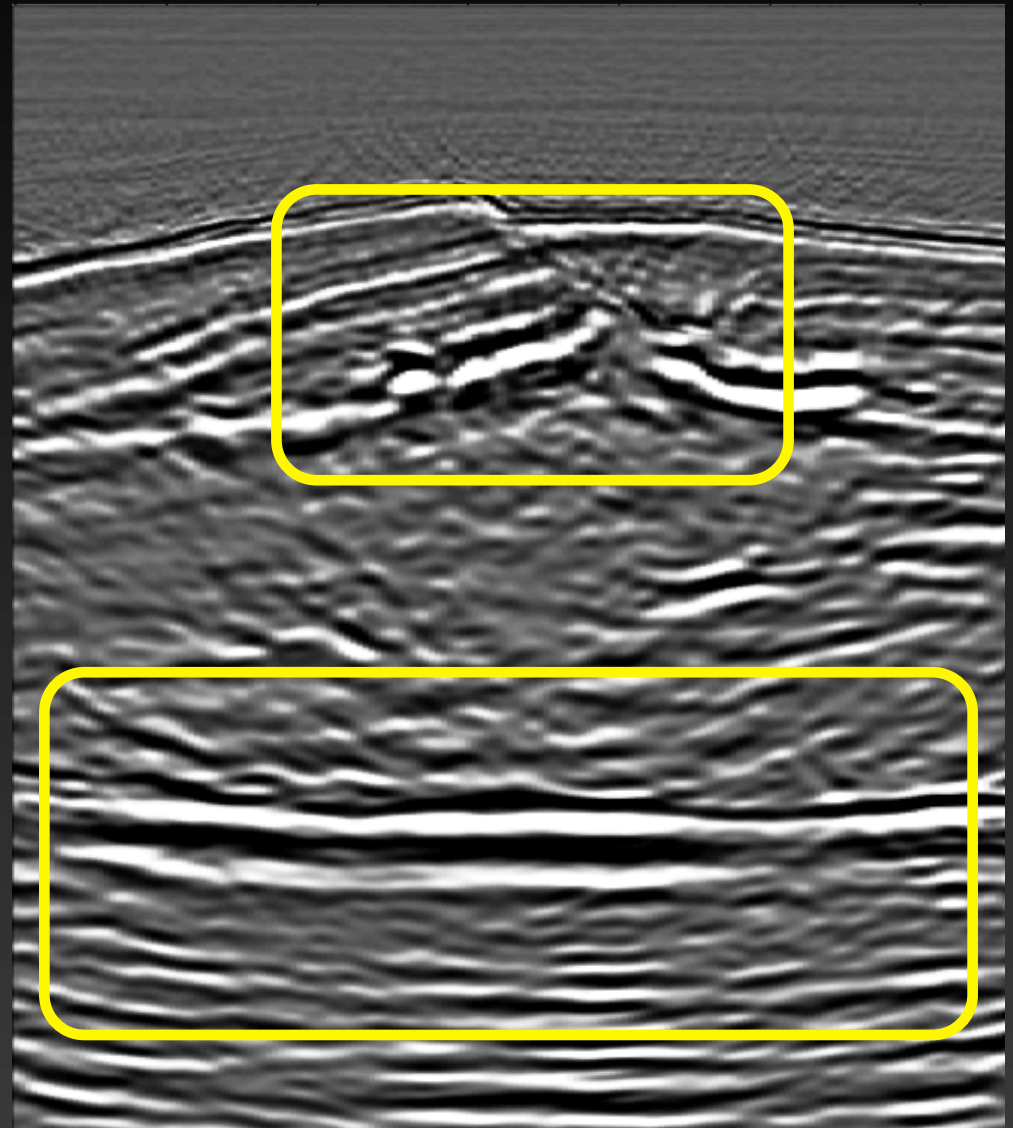




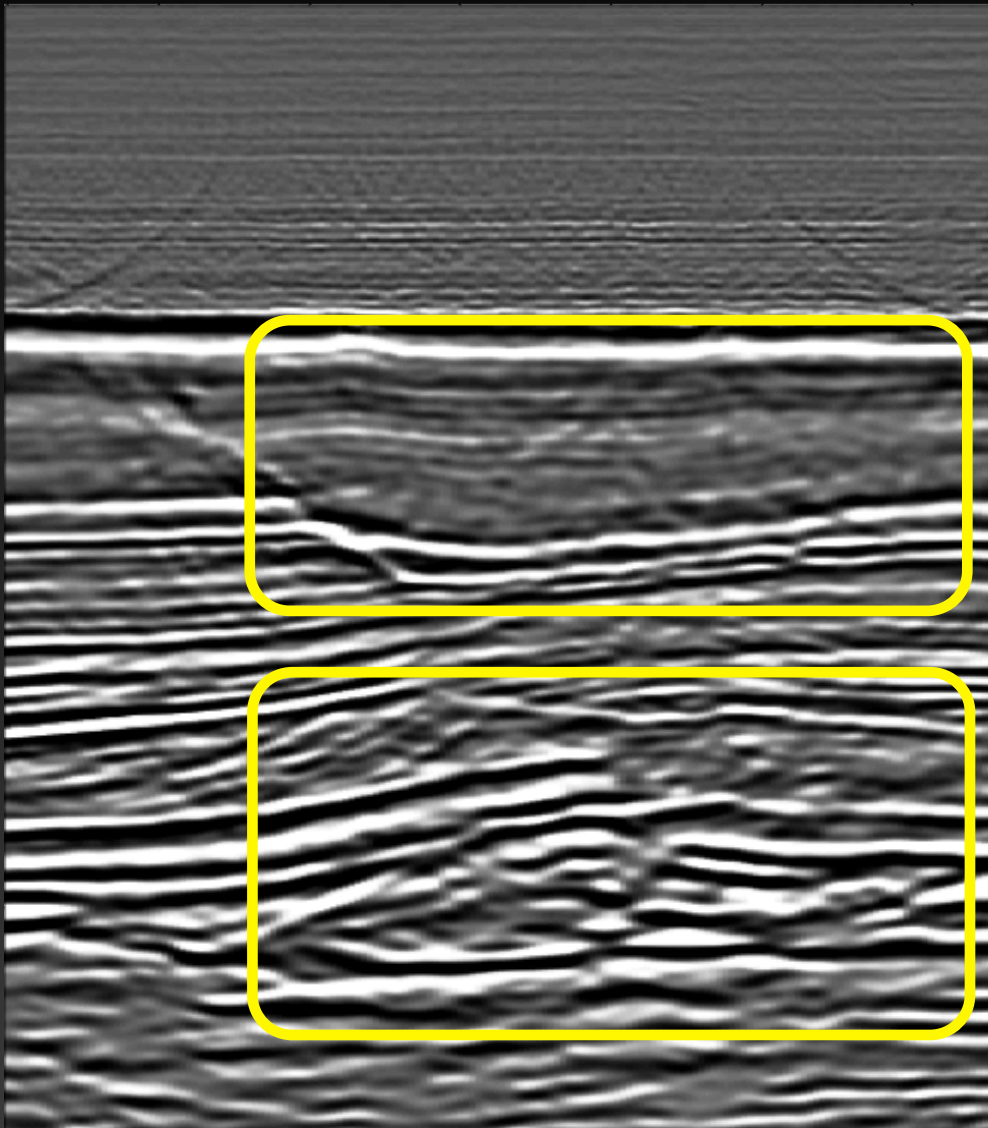
Original



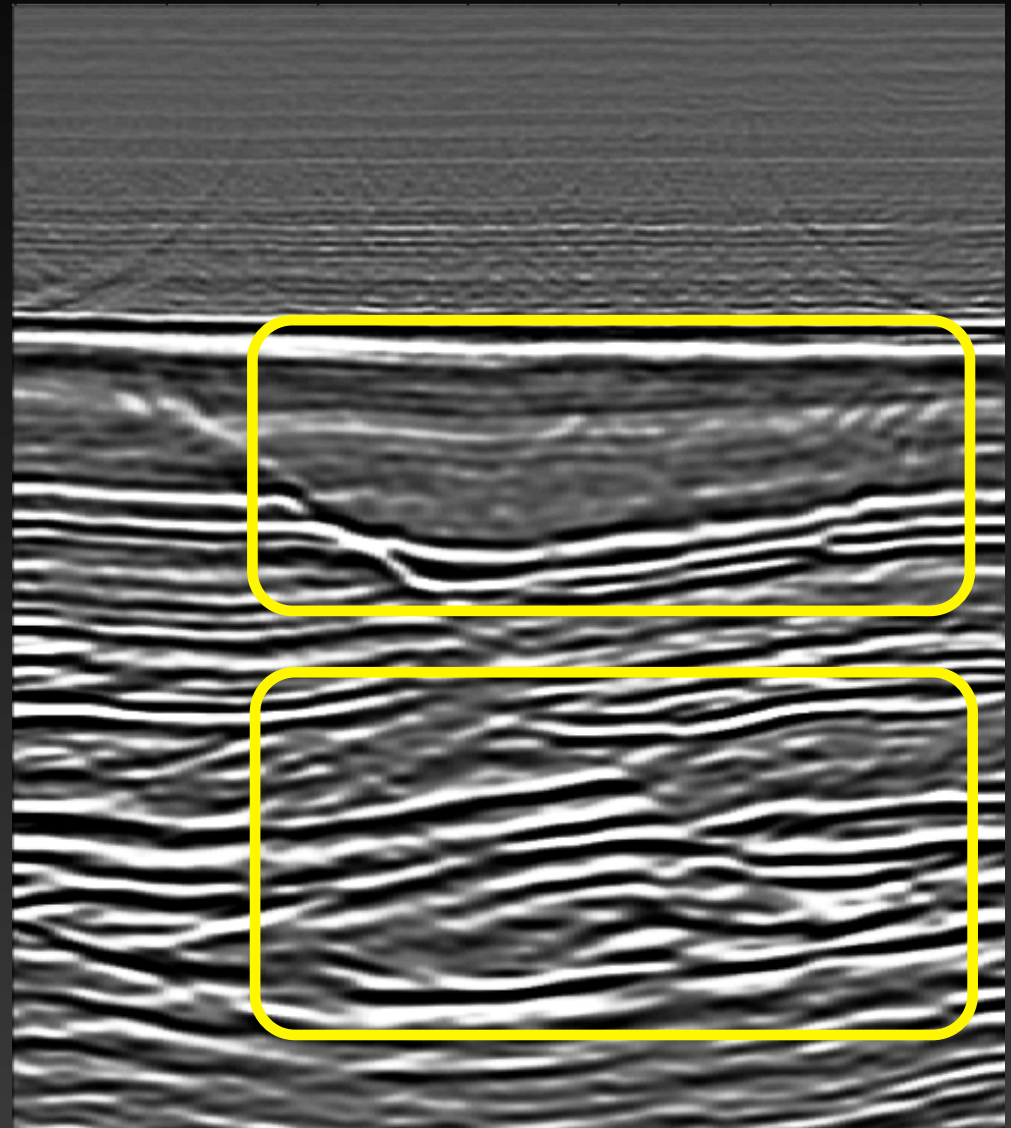
Final



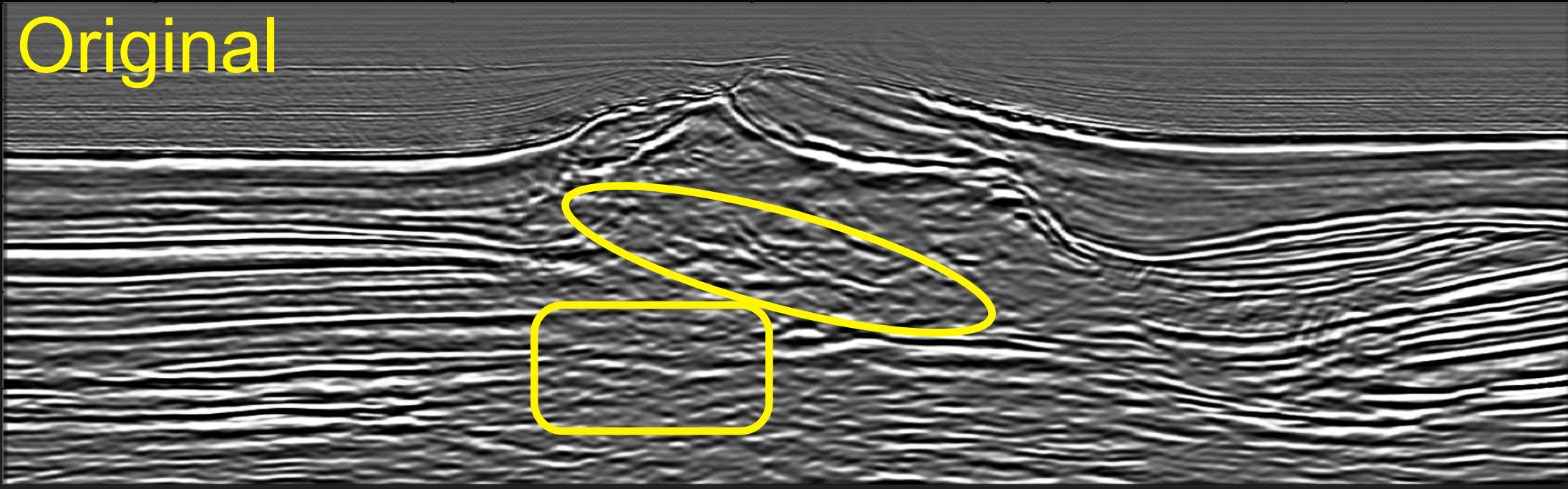
Original



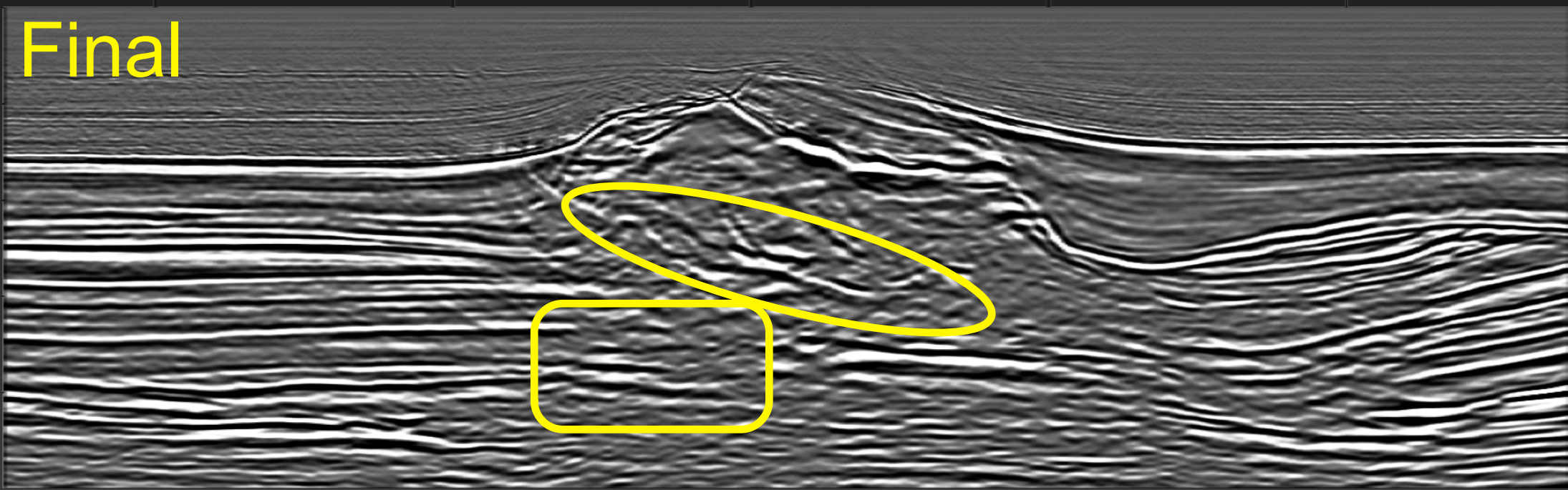
Final



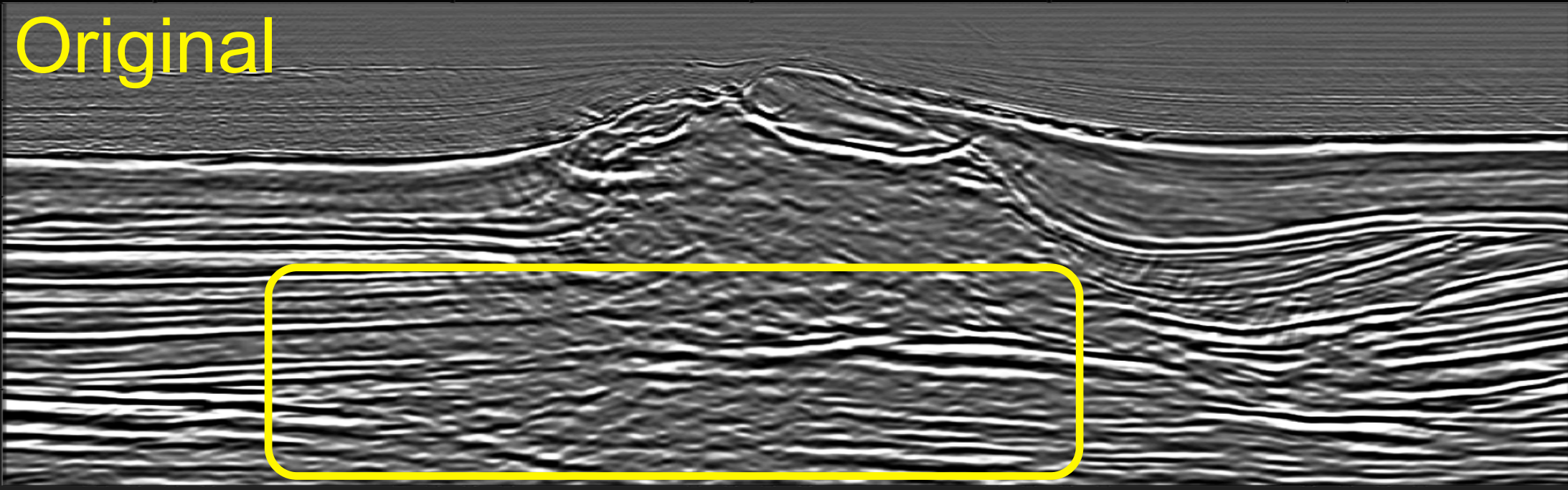
Original



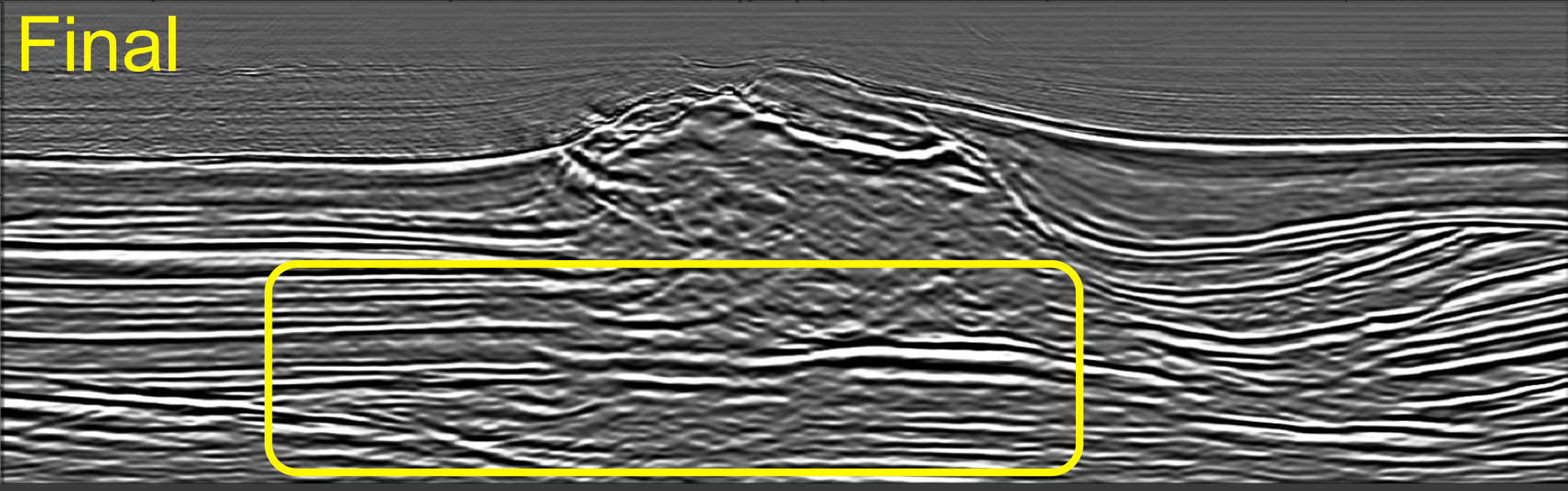
Final



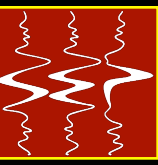
Original



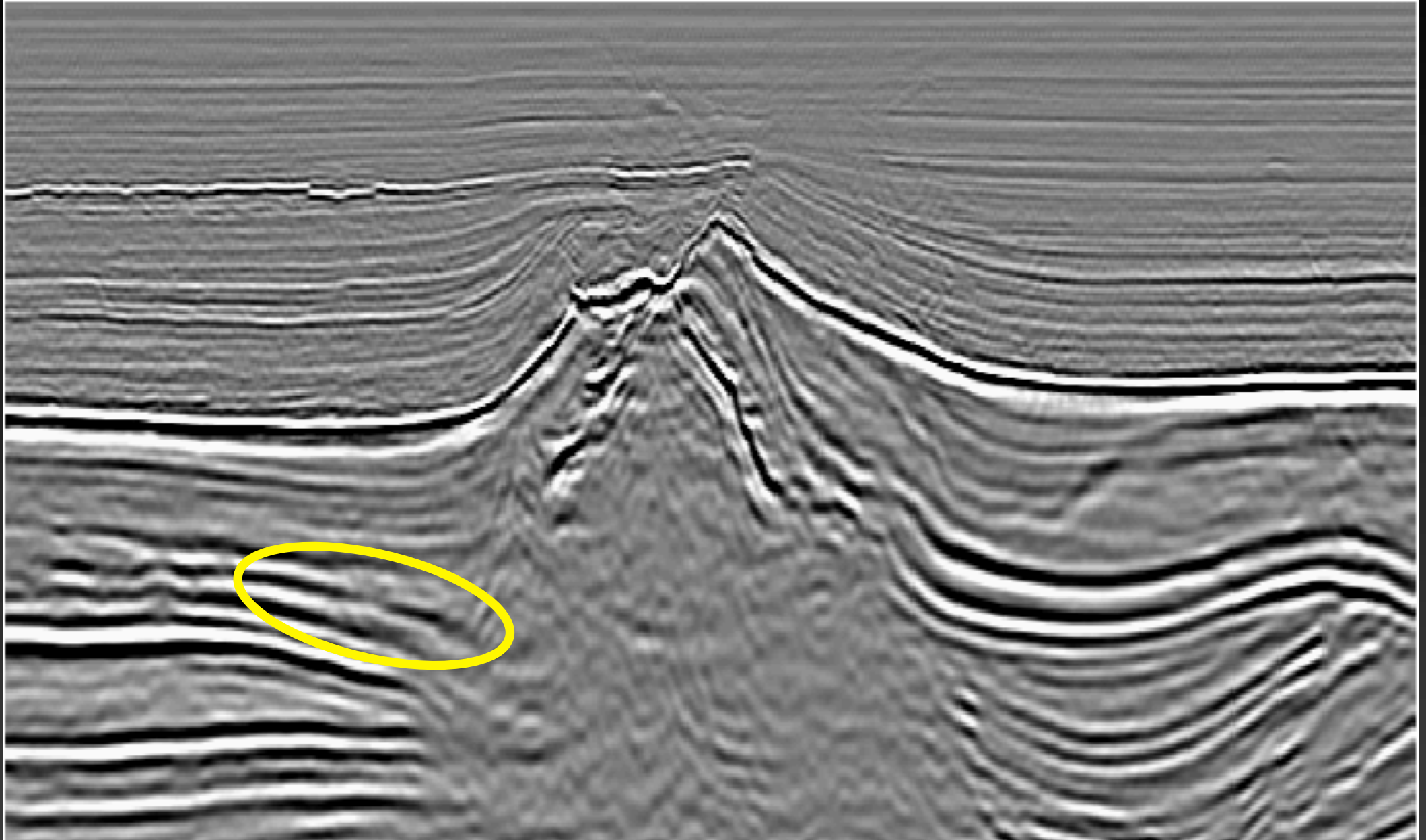
Final

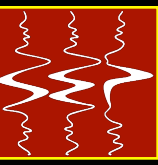


Perguntas?
Sugestões?

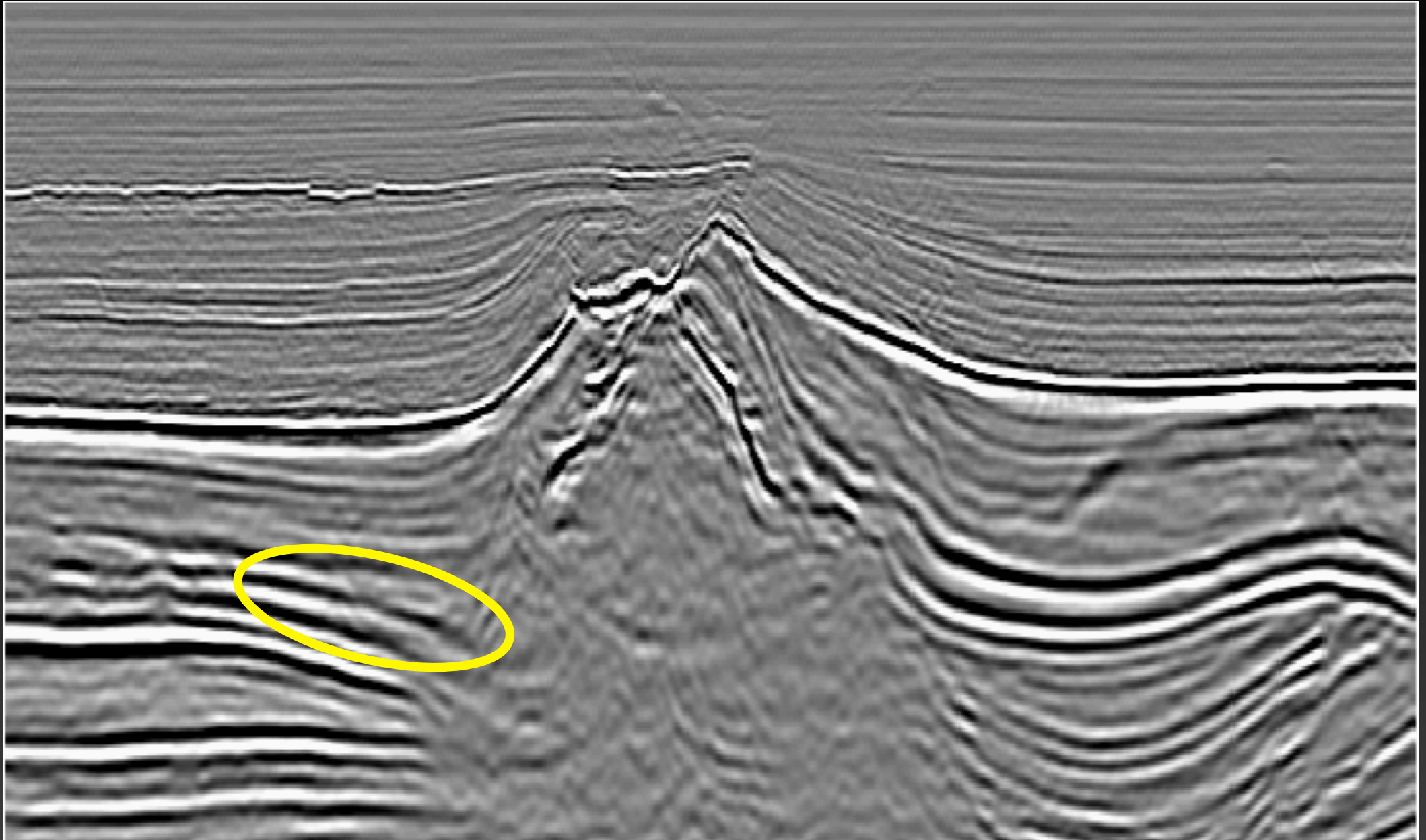


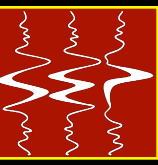
Before



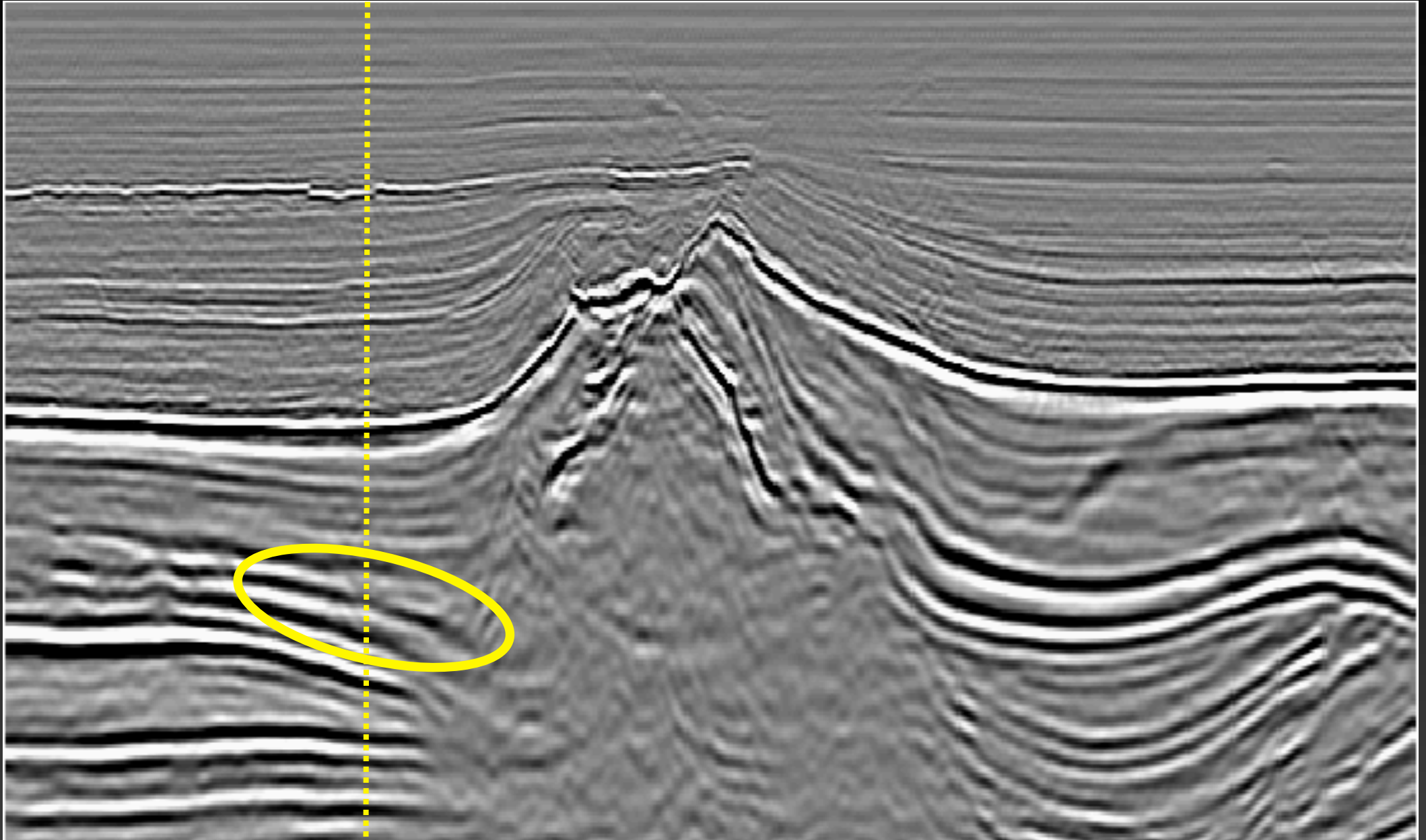


After



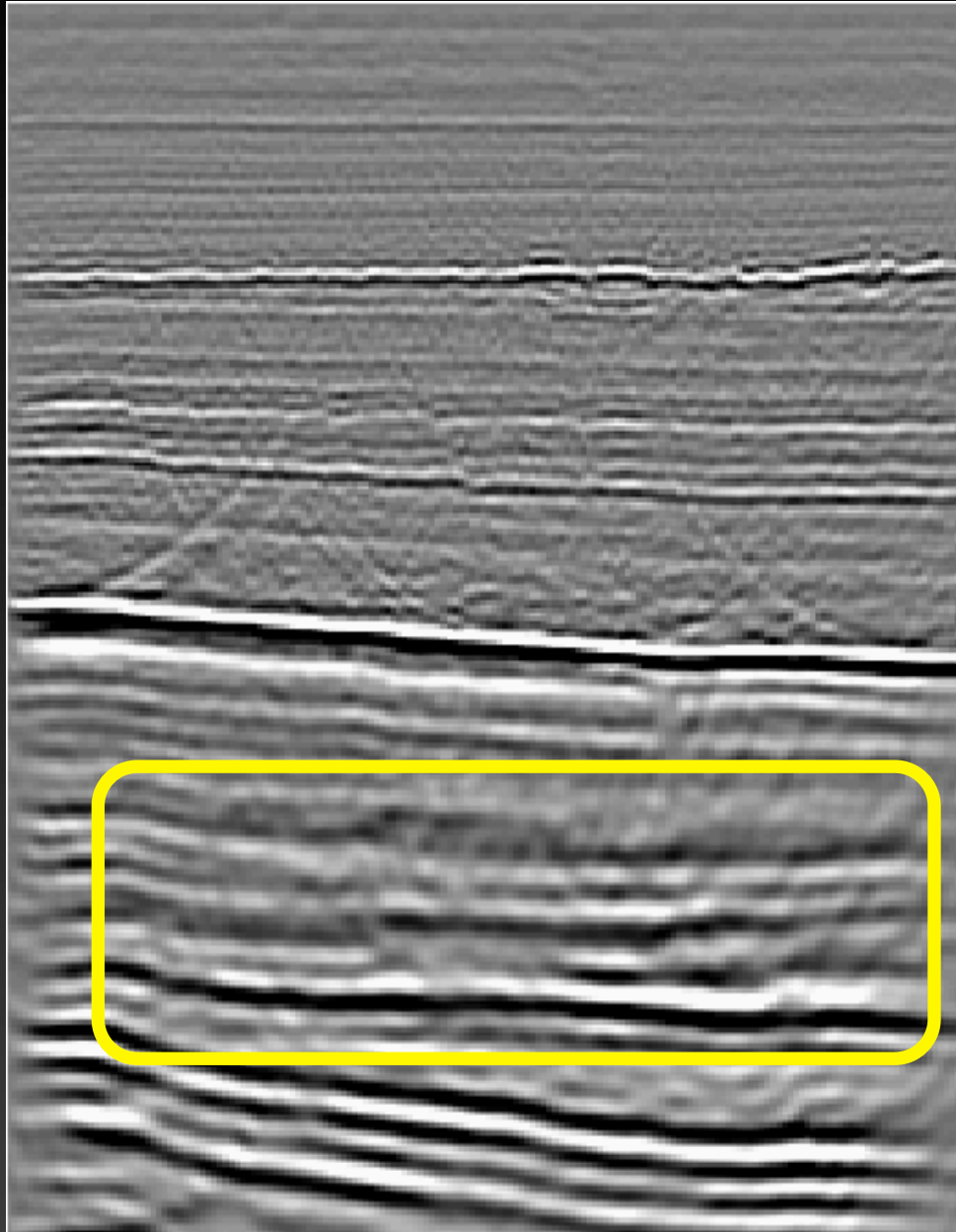


After

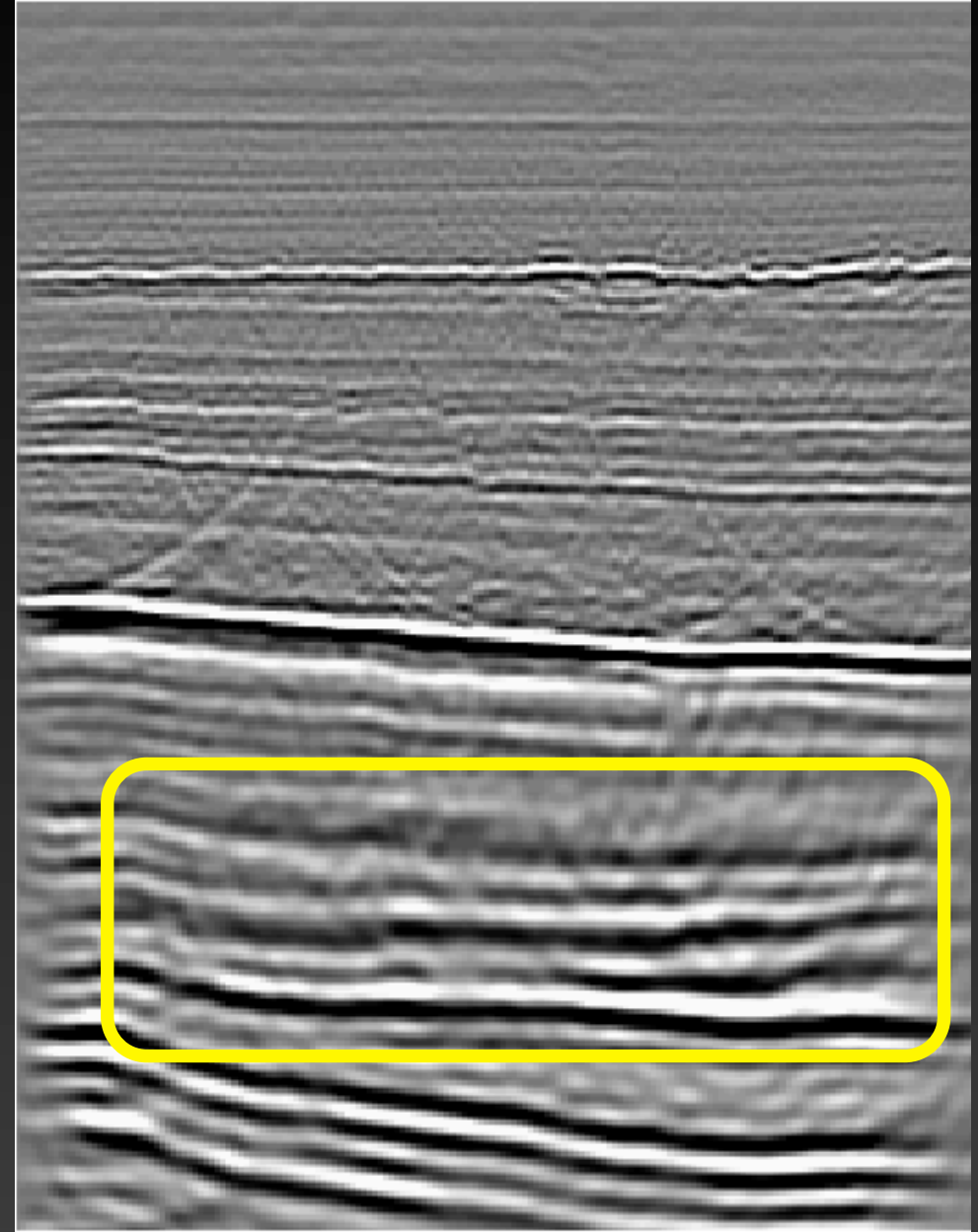




Before



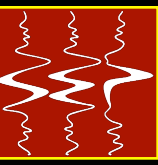
After



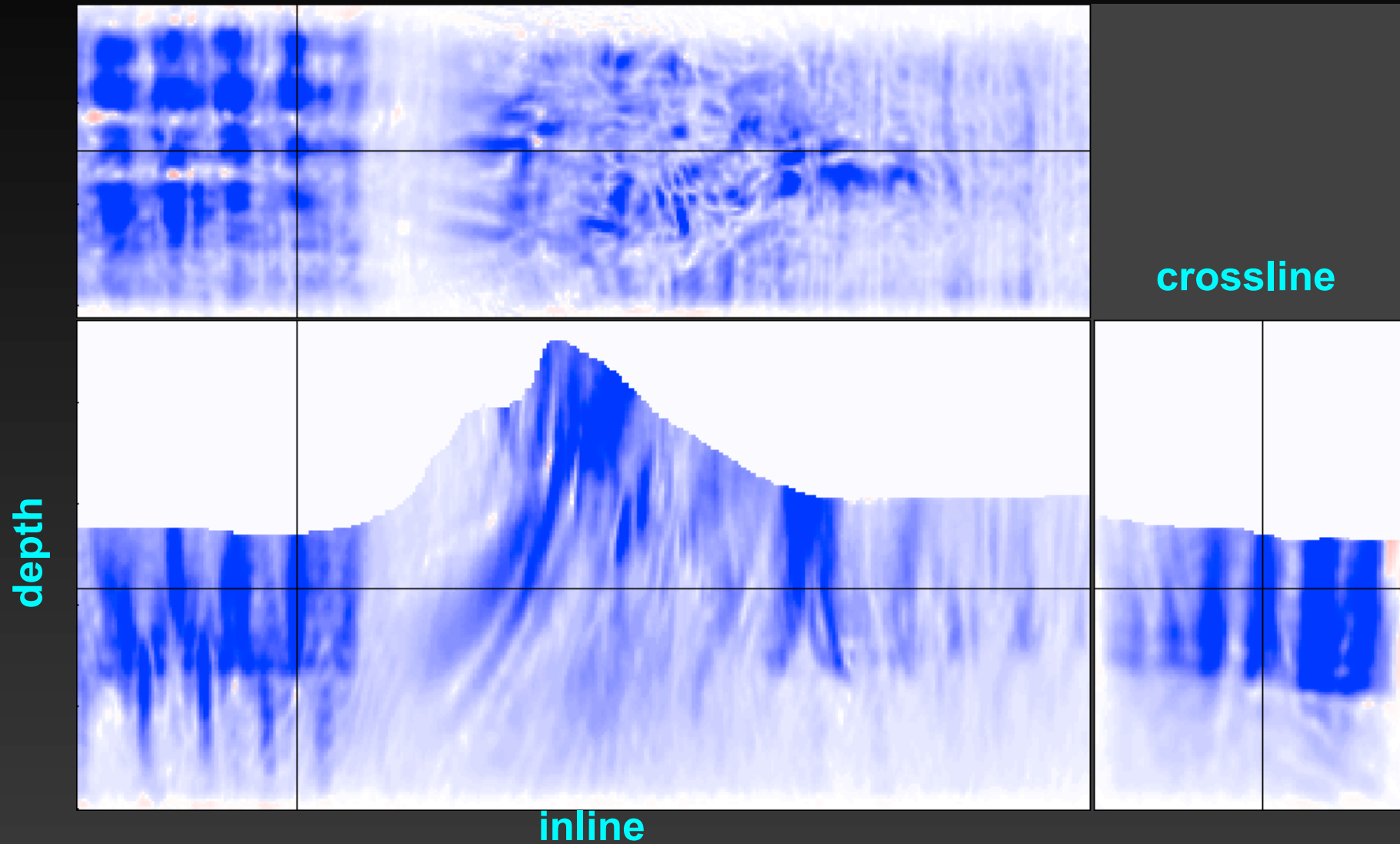


- **Non-linear conjugate gradient**
- **Maximum of 10% velocity variation between iterations**
- **Two function evaluations per iteration**
- **CEES: 30 Dual Nehalem 5520, 24Gb RAM**
 - **Objective function: 30 min**
 - **Gradient: 60 min**

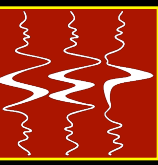
Potential for grid-tomography



Gradient with 1 pair of ISPEW

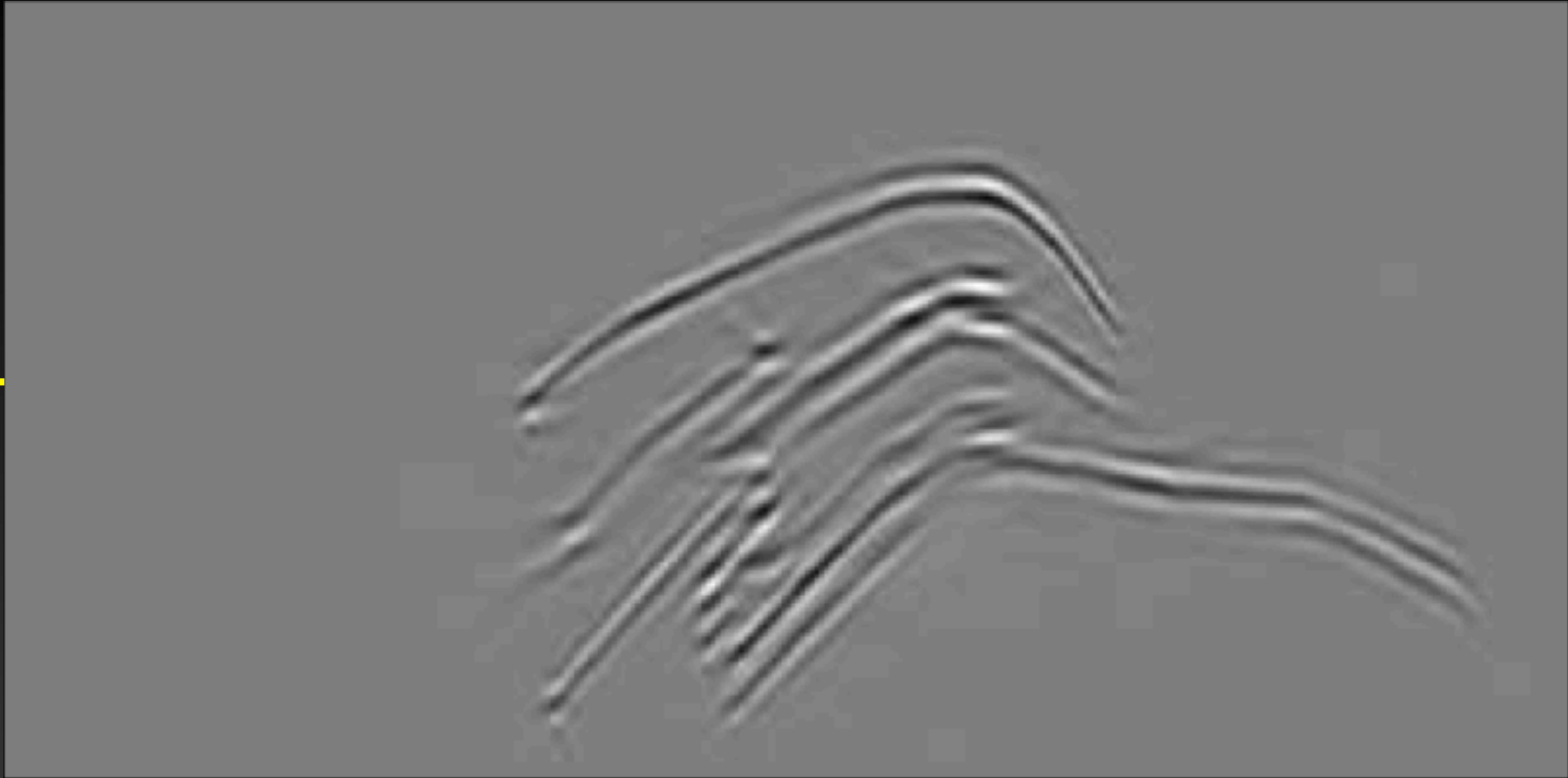


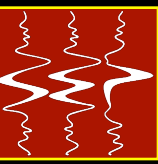
35 PERM migrated image



distance

depth





Source wavefield

$$\begin{cases} \left(\frac{\partial}{\partial z} - i\Lambda \right) \tilde{P}_d(\mathbf{x}, \mathbf{r}_m, \omega) & = \tilde{I}_d(\mathbf{x}, \mathbf{h}, \mathbf{r}_m, \omega) \\ \tilde{P}_d(x, y, z = z_{\max}, \mathbf{r}_m, \omega) & = 0 \end{cases}$$

Receiver wavefield

$$\begin{cases} \left(\frac{\partial}{\partial z} + i\Lambda \right) \tilde{P}_u(\mathbf{x}, \mathbf{r}_m, \omega) & = \tilde{I}_u(\mathbf{x}, \mathbf{h}, \mathbf{r}_m, \omega) \\ \tilde{P}_u(x, y, z = z_{\max}, \mathbf{r}_m, \omega) & = 0 \end{cases}$$

Λ = SSR operator

\tilde{P}_d = phase encoded source wavefield

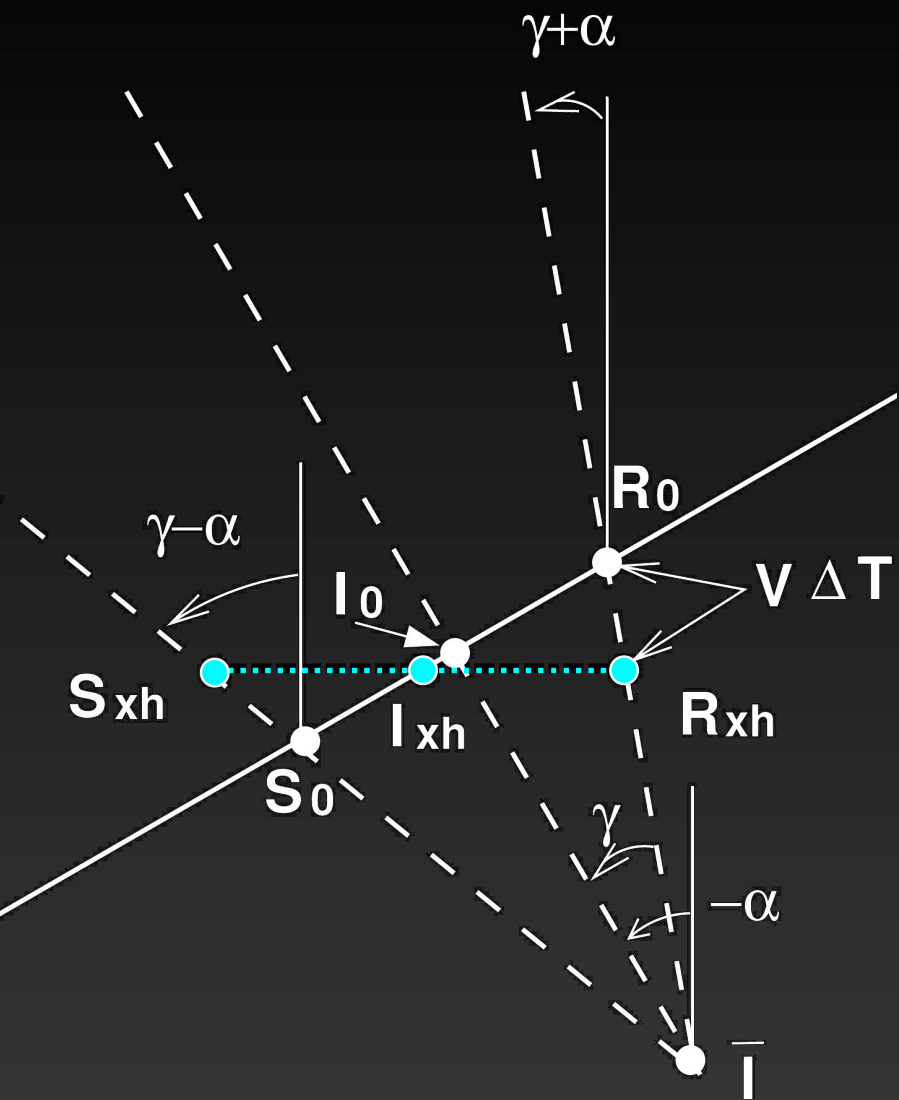
\tilde{I}_d = source wavefield initial condition

\mathbf{r}_m = index of random realizations

\tilde{P}_u = phase encoded receiver wavefield

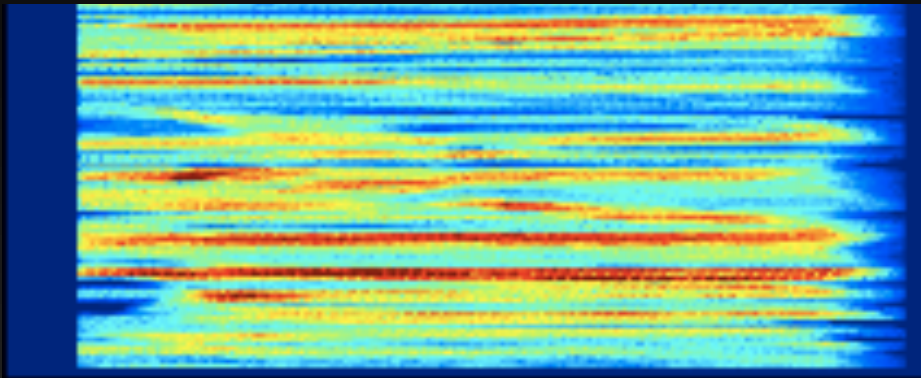
\tilde{I}_u = receiver wavefield initial condition

Image-space phase-encoded wavefields

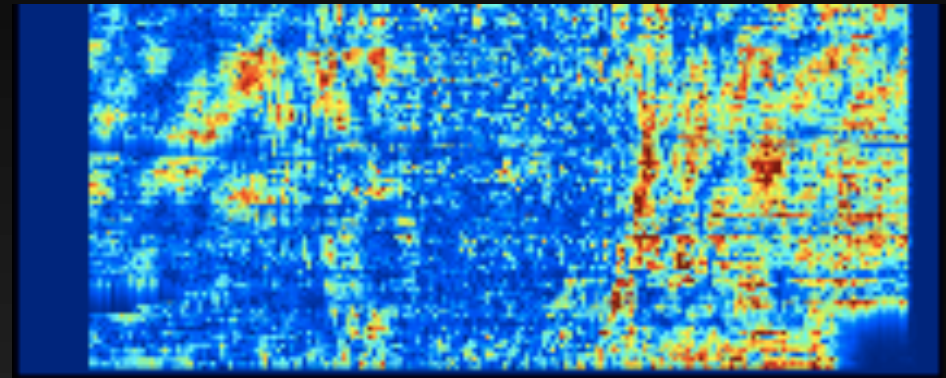


cobertura
crossline

inline



Amplitude
crossline



gradiente
crossline

inline

