Sponsors Meeting 2014

Inverting raypath-dependent delay times to compute S-wave velocities in the near-surface

Authors: Raul Cova*
Kris Innanen

Banff, December 4th, 2014

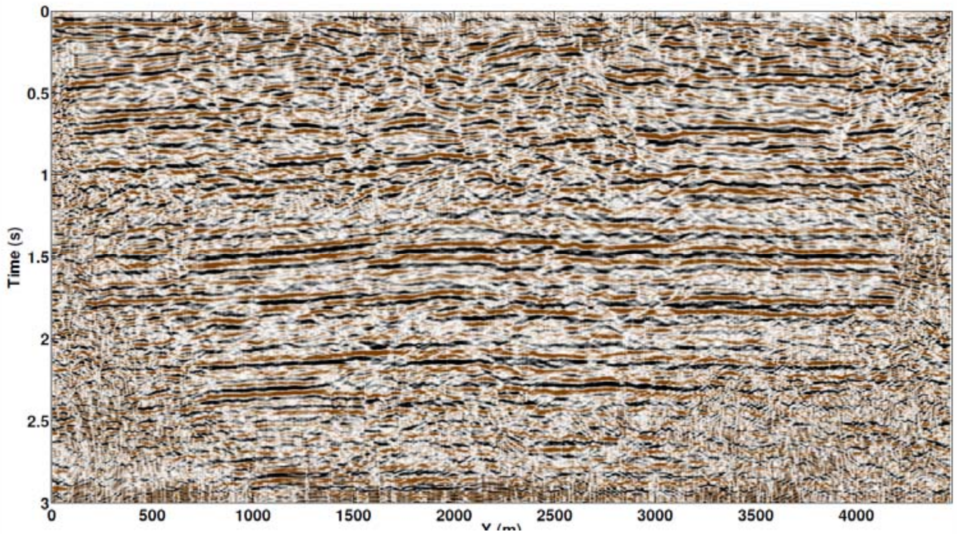
CREWES

Introduction

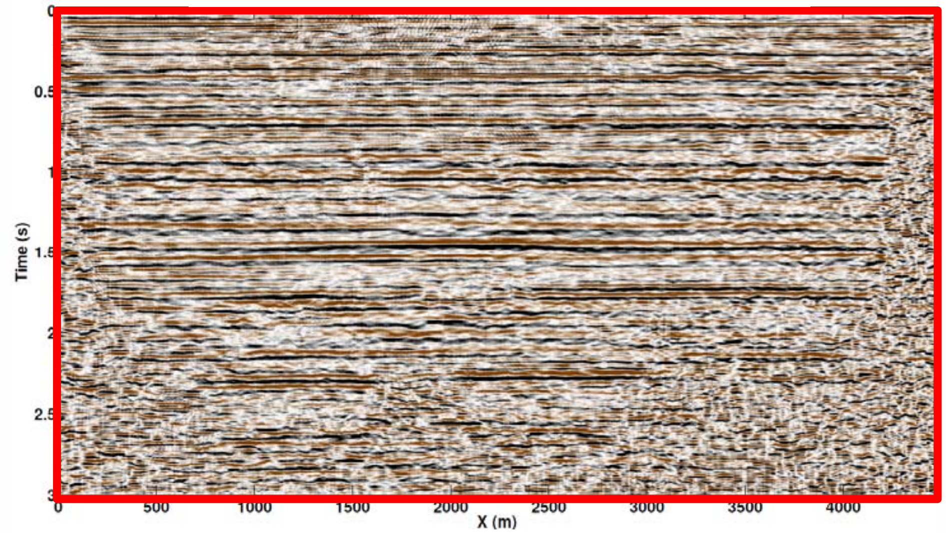
- The interferometric solution proposed by Henley (2012, 2014) retrieves the static corrections by cross correlating traces in the radial-trace domain.
- The rayparameter "p" when measured from data recorded with surface arrays is related to the emerging angle of the wavefield at the surface.
- Cova et al. (2013) showed how raypath-dependent static corrections are important to account for the non-stationary character of S-wave statics.
- Is it possible to use these cross correlation functions to characterize the near-surface? What information do we need? What type of inversion is possible?

Rayparameter Domain Statics

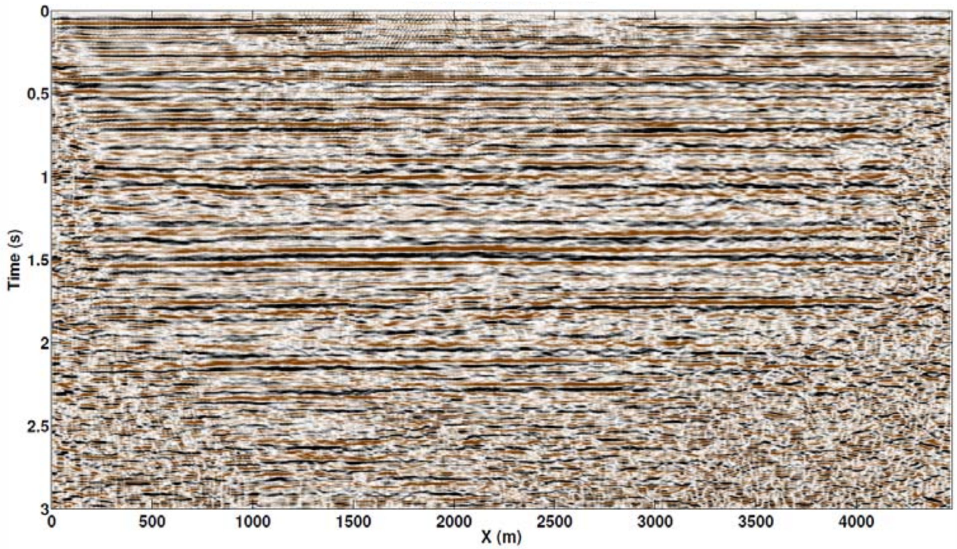
Raw ACP Stack



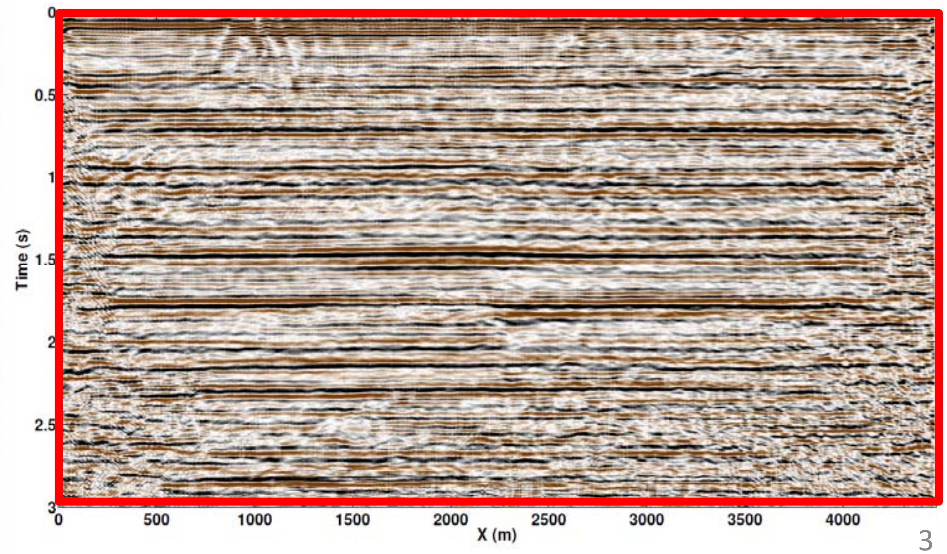
RT Domain Static Corrections



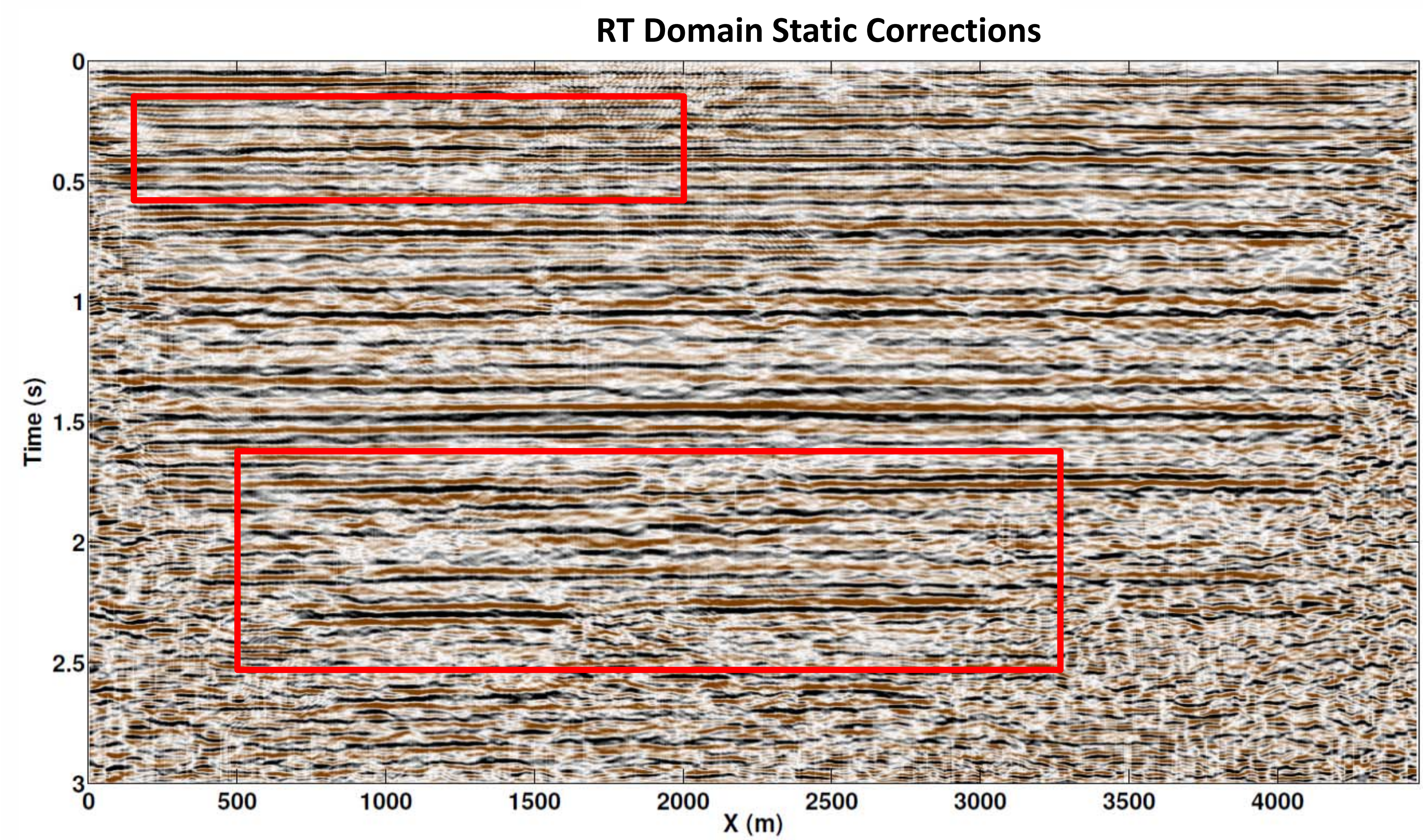
Snell-Trace Domain Static Corrections



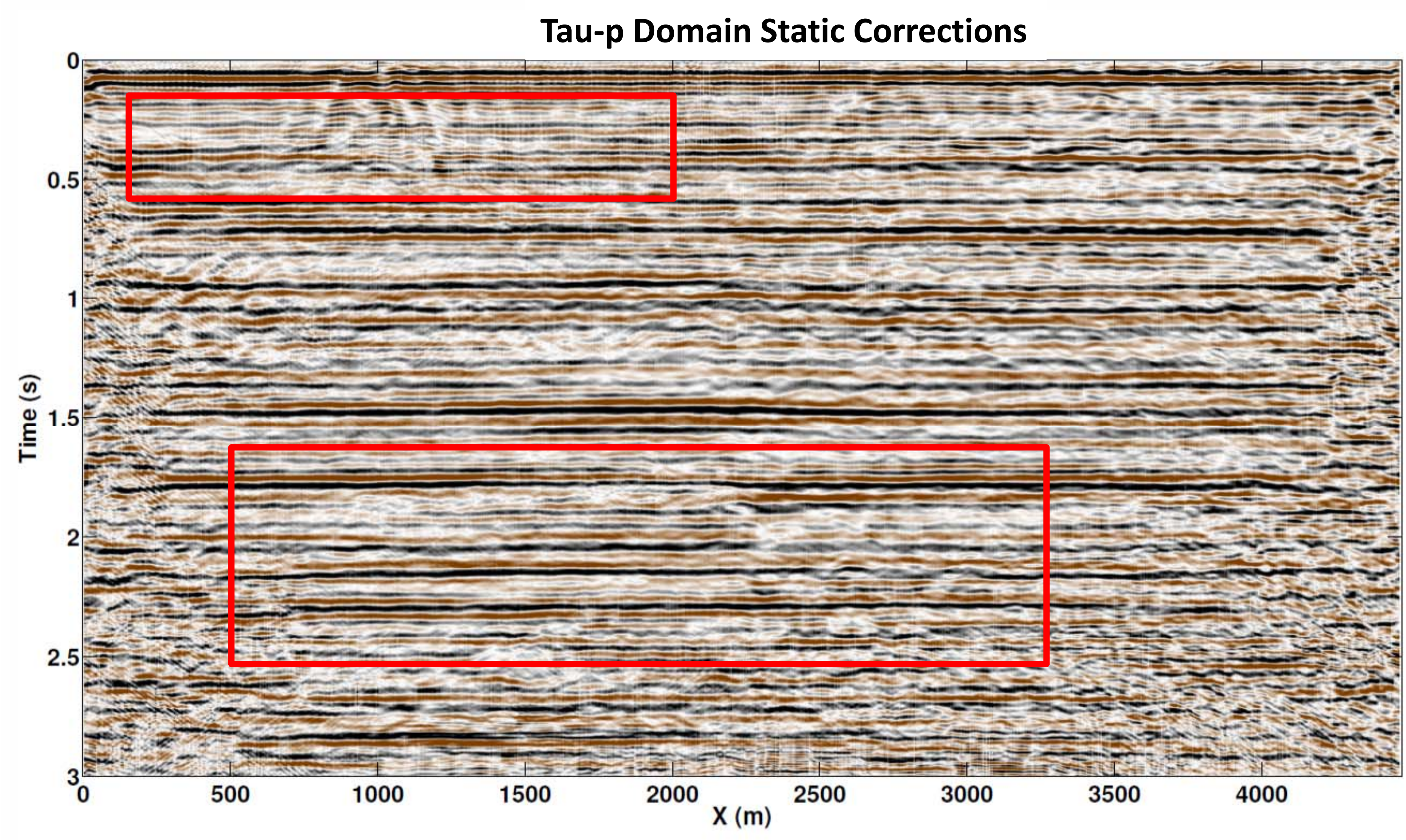
Tau-p Domain Static Corrections



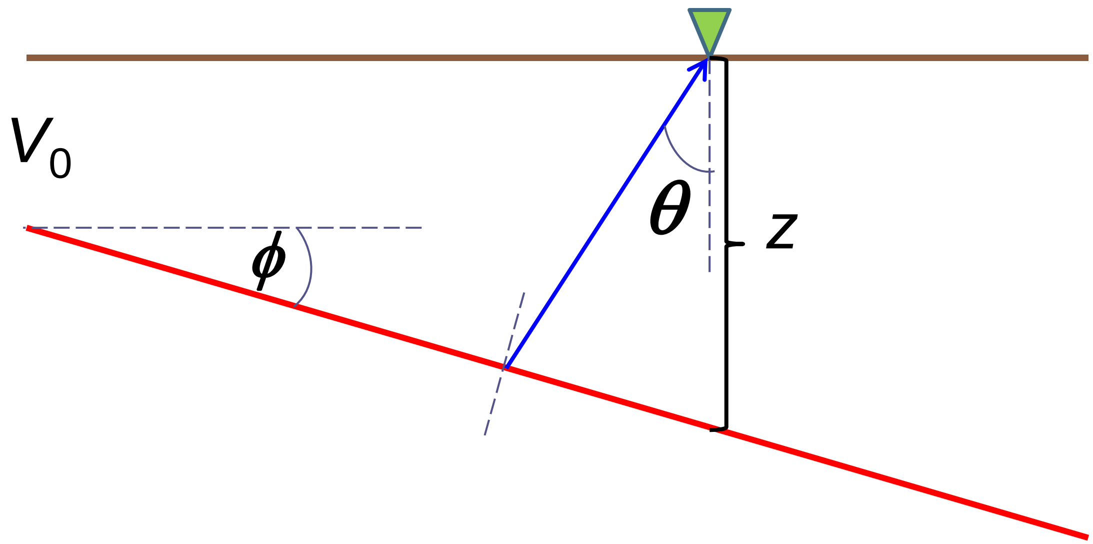
Rayparameter Domain Statics



Rayparameter Domain Statics



Traveltimes in the near-surface



Raypath angle parameterization

$$t(\theta) = \frac{z}{V_0}\frac{\cos(\phi)}{\cos(\theta - \phi)}$$

Rayparameter parameterization

$$t(p) = \frac{z}{V_0^2(q + p\tan(\phi))}$$

Horizontal (p) and vertical (q) slownesses

$$p = \frac{\sin(\theta)}{V_0} \qquad q = \frac{\cos(\theta)}{V_0}$$

$$q = \frac{1}{V_0}\sqrt{1 - V_0^2 p^2}$$

Quasi-Newton Inversion

Data Residuals — Forward modelled data — Observed data

Objective Function

$$\mathbf{\Phi(m)} = ||\delta \mathbf{d}||^2 = ||\mathbf{g(m)} - \mathbf{d_{obs}}||^2,$$

$$\mathbf{m_i} = \mathbf{m_{i-1}} + \delta \mathbf{m_i},$$

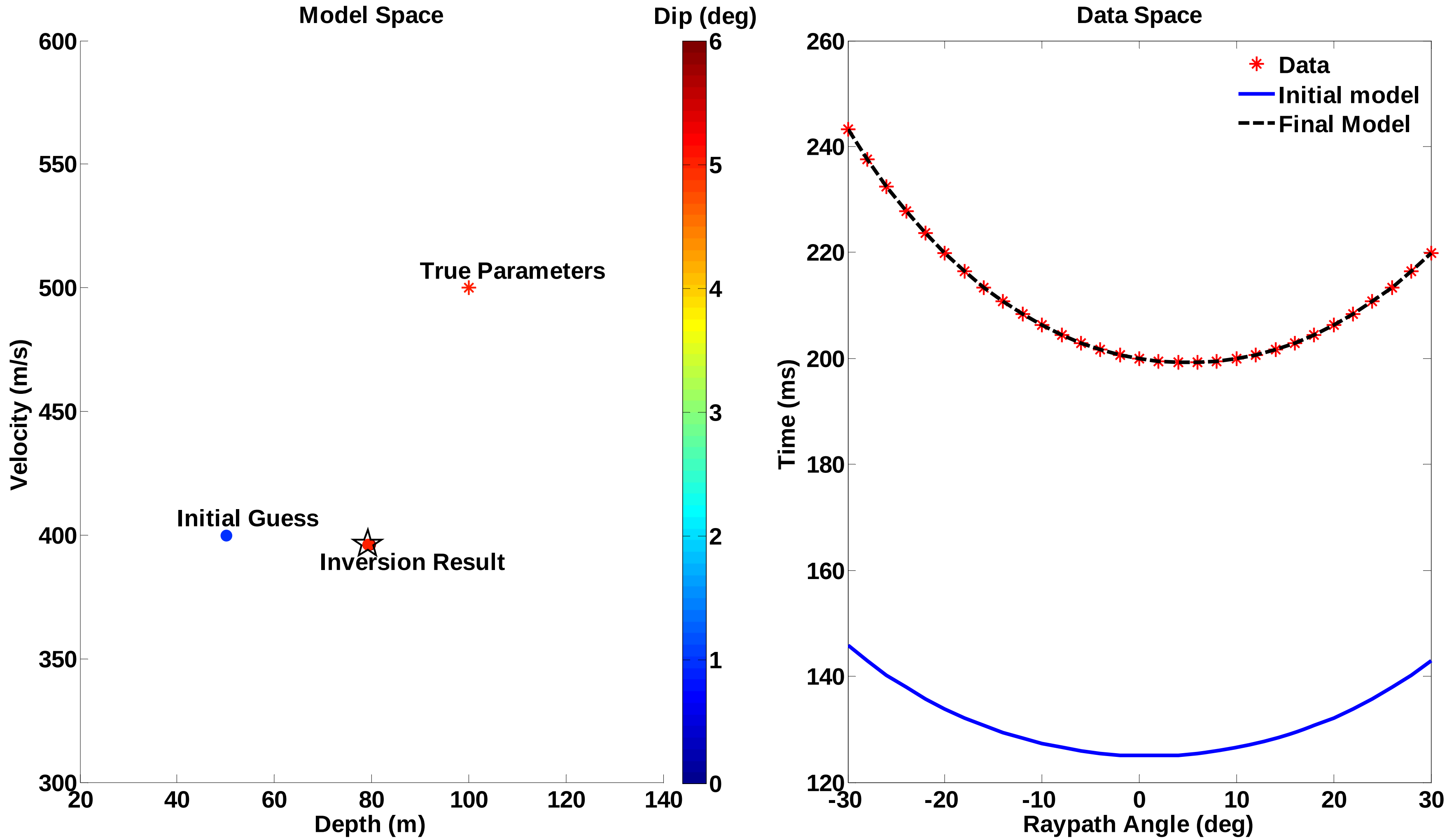
← Model Update

$$\delta \mathbf{m} = \left[\mathbf{J(m)}^\dagger \mathbf{J(m)}\right]^{-1} \mathbf{J(m)}^\dagger \delta \mathbf{d}.$$

Jacobian

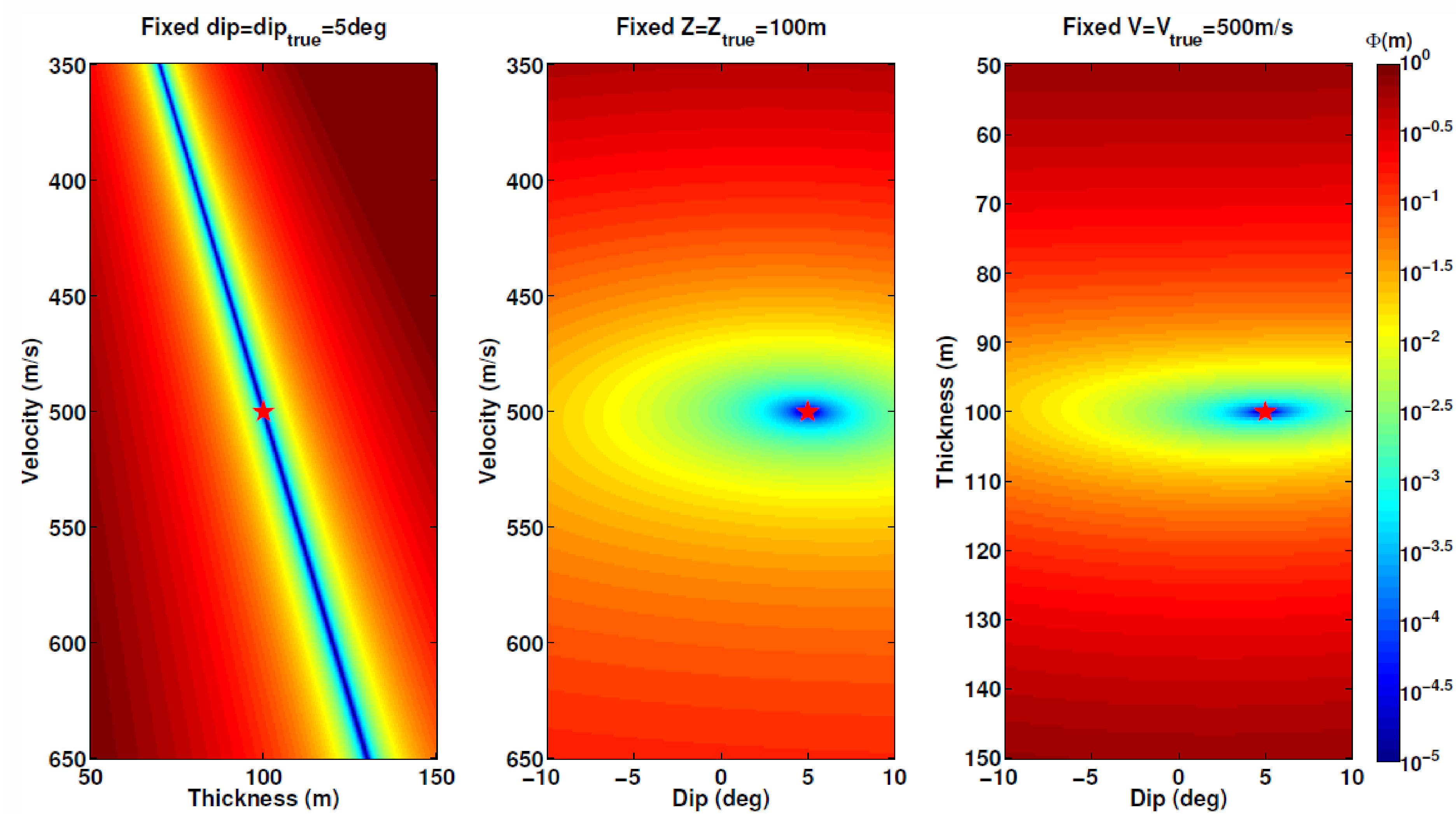
$$\mathbf{J(m)} = \left[\frac{\partial \mathbf{g(m)}}{\partial \mathbf{m}}\right] = \left[\frac{\partial g(m)}{\partial z}, \quad \frac{\partial g(m)}{\partial V_0}, \quad \frac{\partial g(m)}{\partial \phi}\right].$$

Inversion Results in the Raypath Angle Domain



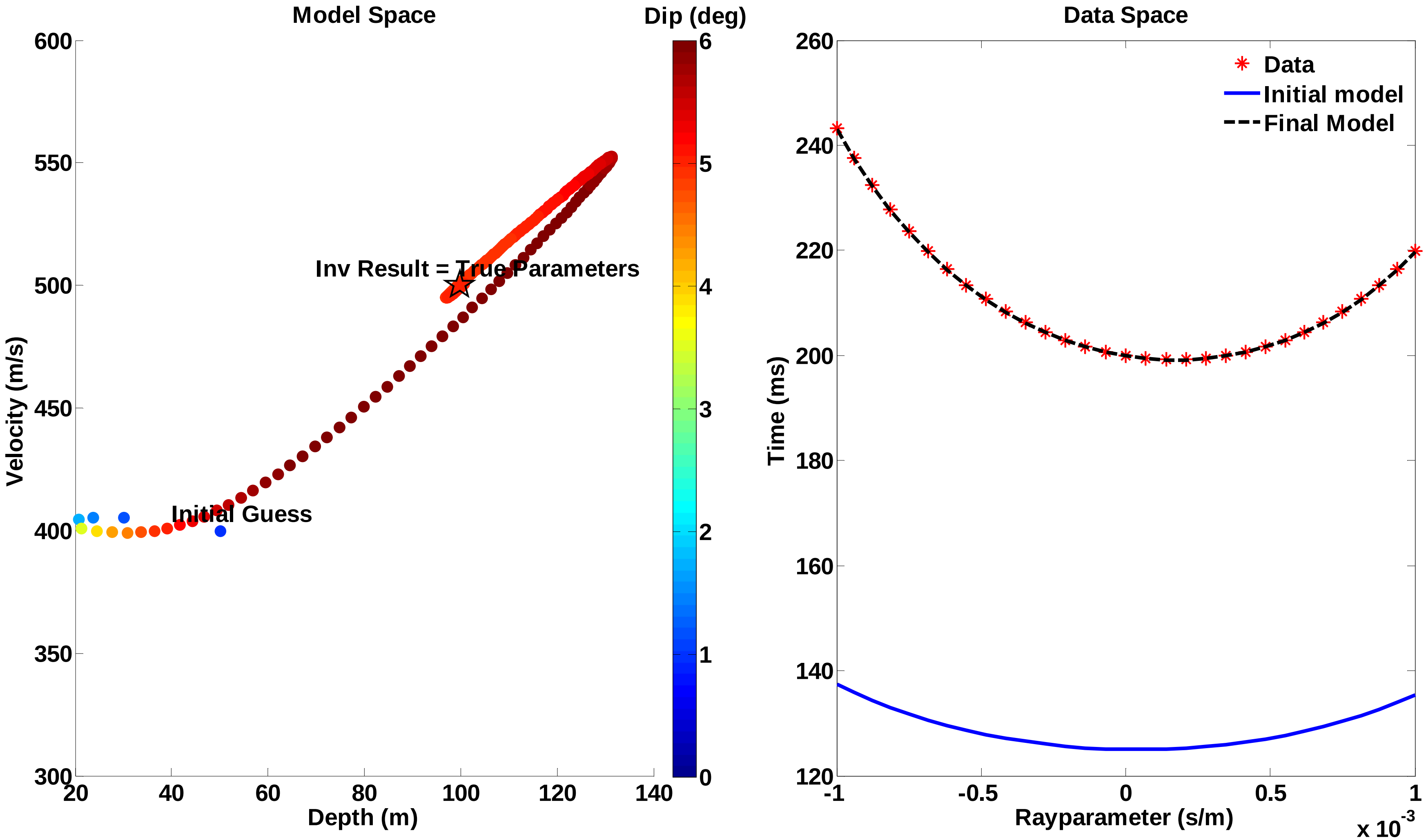
Only the actual dip is successfully retrieved

Objective Function (Raypath Angle Parameterization)



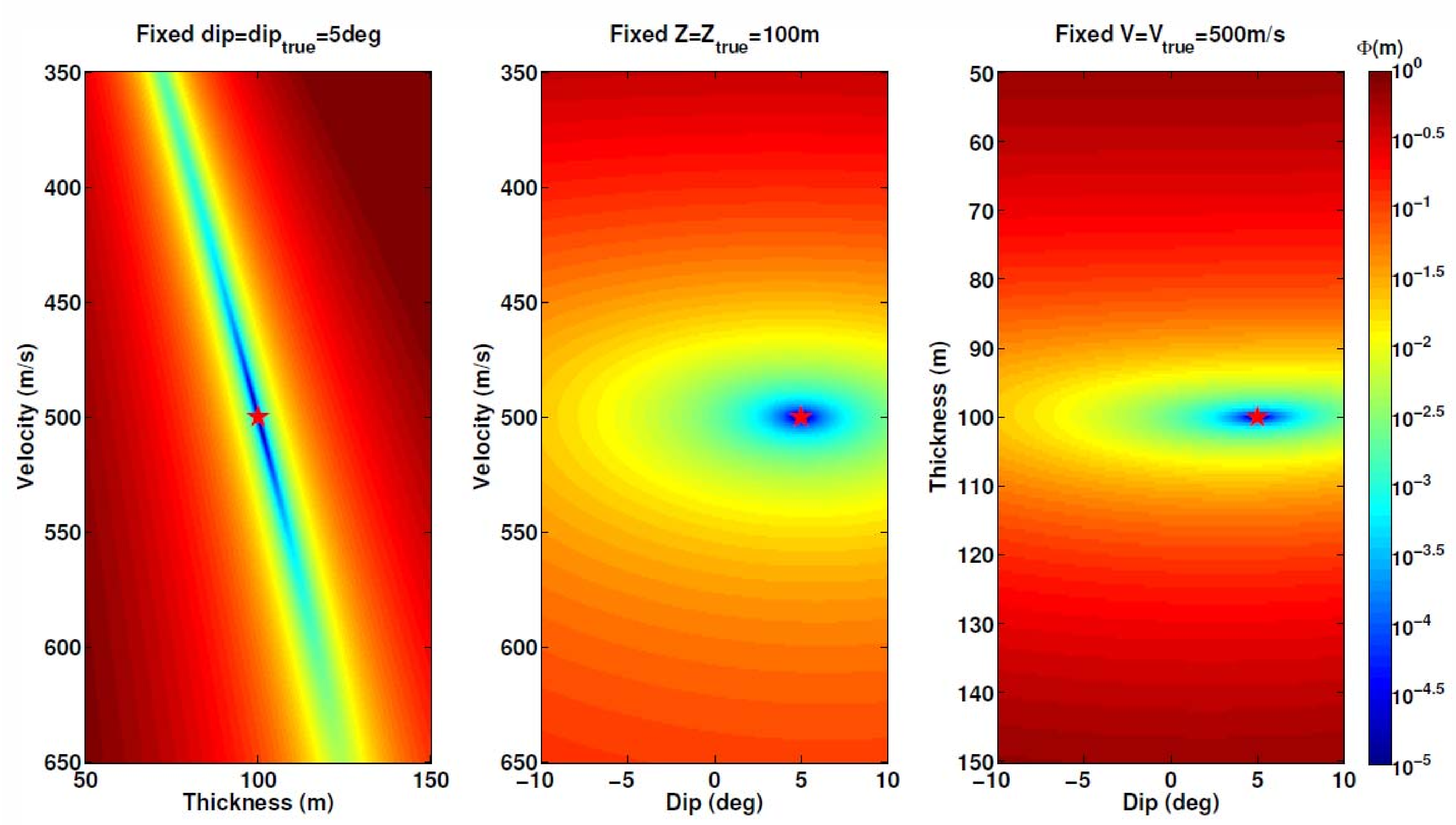
There is not a well defined minimum in the objective function for a fixed dip value

Inversion Results in the Rayparameter Domain



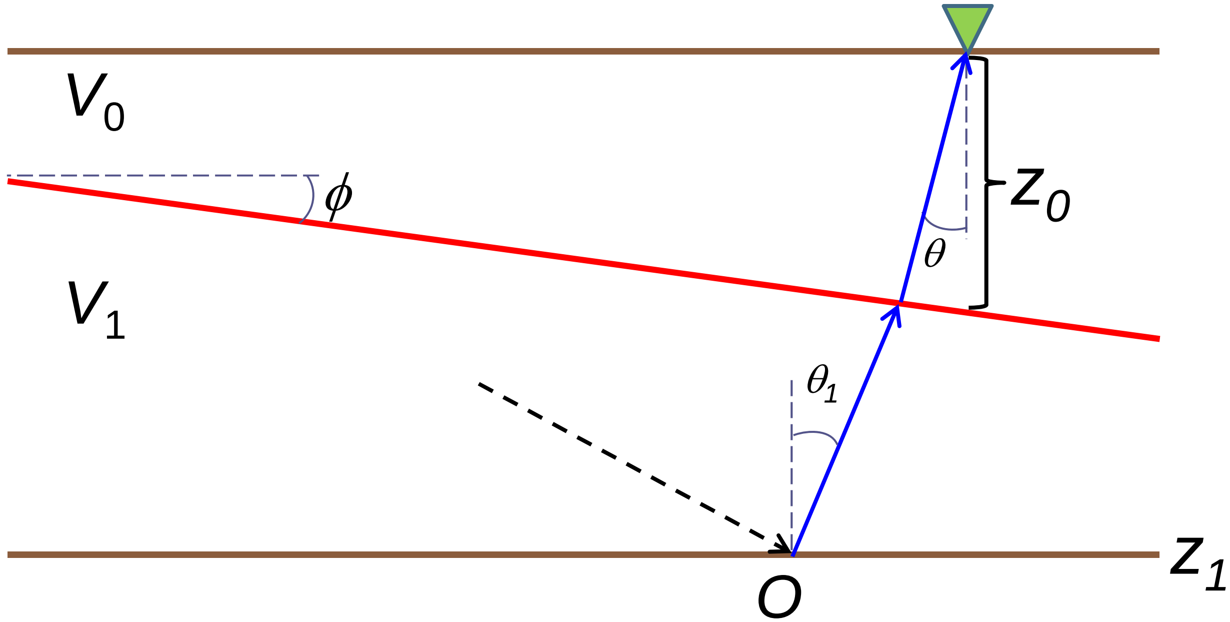
True model parameters successfully recovered.

Objective Function (Rayparameter Parameterization)



The objective function now displays a well defined minimum for a fixed dip value

Receiver side traveltimes



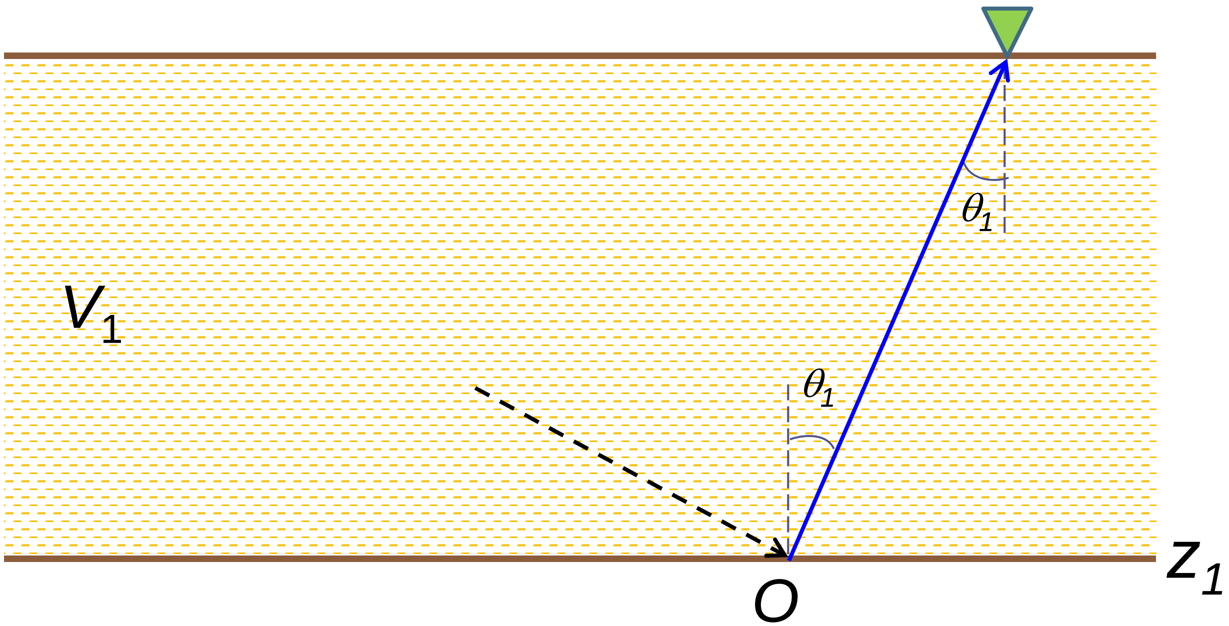
$$t(\theta) = \frac{Z_1}{V_1\cos(\theta_1)} + \frac{Z_0\cos(\phi)}{V_0\cos(\theta-\phi)}\left(1 - \frac{V_0\cos(\theta)}{V_1\cos(\theta_1)}\right)$$

Traveltime without near surface effects

Near-surface traveltime

Traveltime lost in medium 1

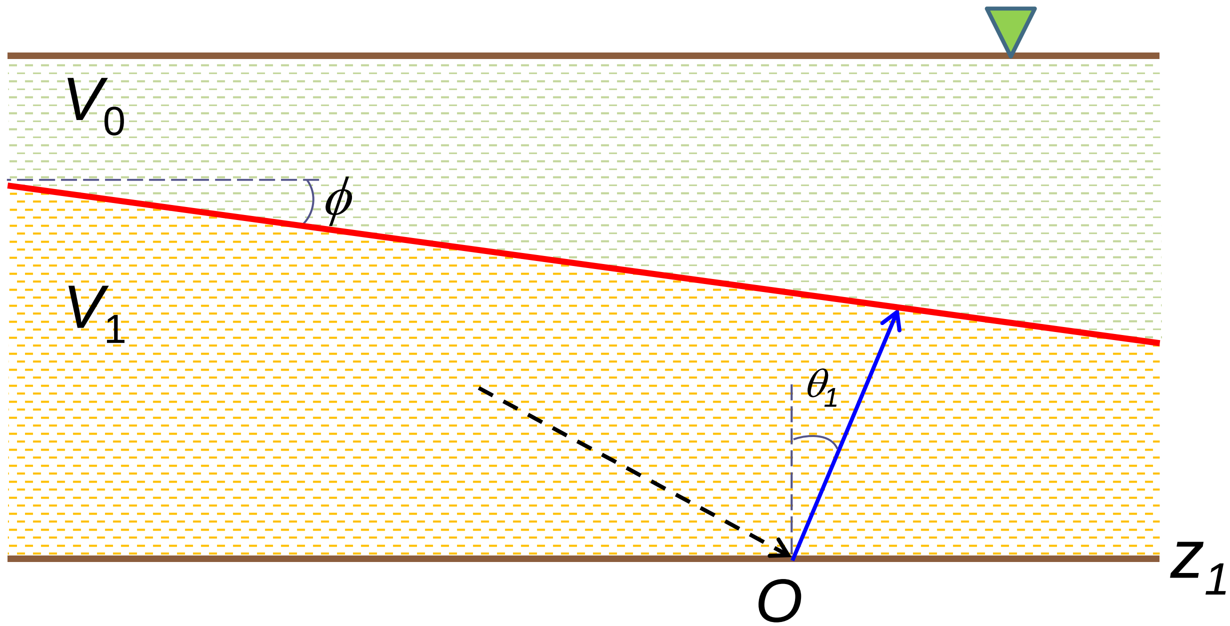
Receiver side traveltimes



$$t(\theta) = \frac{Z_1}{V_1 \cos(\theta_1)} + \frac{Z_0 \cos(\phi)}{V_0 \cos(\theta - \phi)} \left(1 - \frac{V_0 \cos(\theta)}{V_1 \cos(\theta_1)}\right)$$

Traveltime without near surface effects

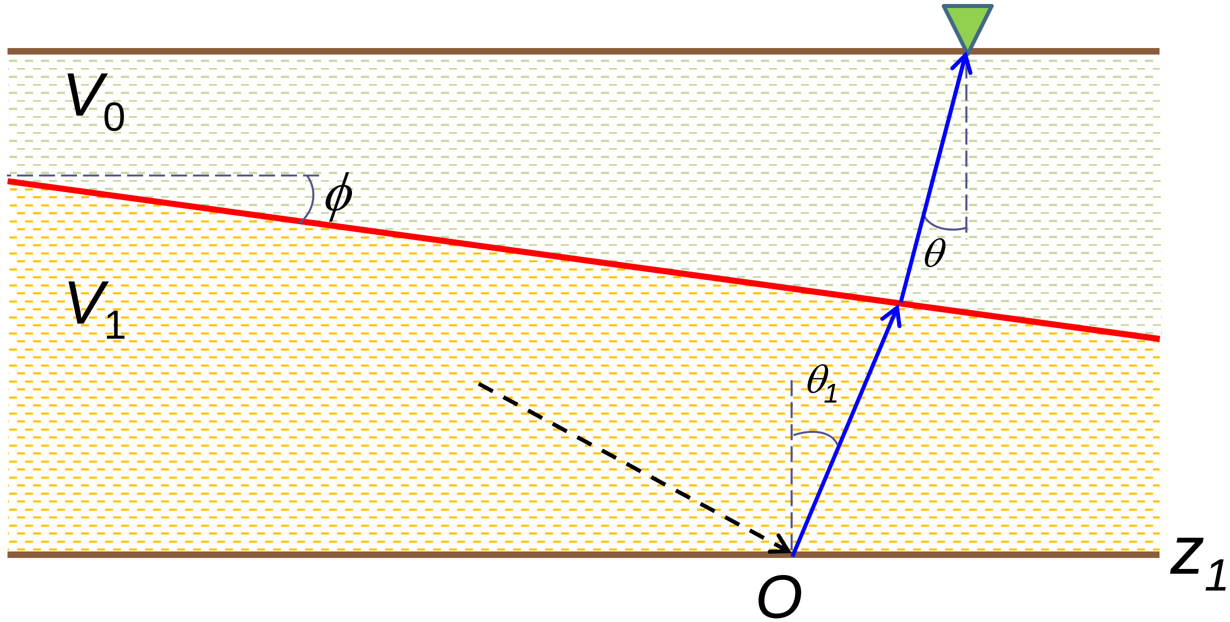
Receiver side traveltimes



$$t(\theta) = \frac{Z_1}{V_1\cos(\theta_1)} + \frac{Z_0\cos(\phi)}{V_0\cos(\theta-\phi)}\left(1 - \frac{V_0\cos(\theta)}{V_1\cos(\theta_1)}\right)$$

Traveltime lost in medium 1

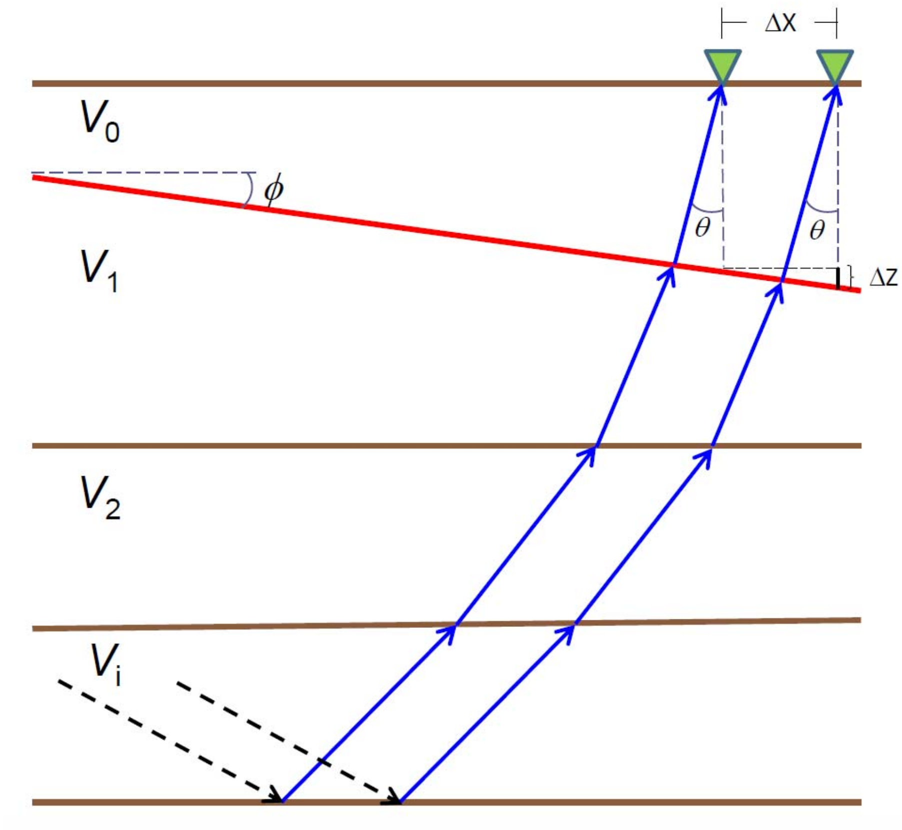
Receiver side traveltimes



$$t(\theta) = \frac{Z_1}{V_1 \cos(\theta_1)} + \frac{Z_0 \cos(\phi)}{V_0 \cos(\theta - \phi)} \left(1 - \frac{V_0 \cos(\theta)}{V_1 \cos(\theta_1)}\right)$$

Near-surface traveltime

Receiver side traveltime Differences



Raypath angle parameterization:

$$\Delta t(\theta) = \frac{\Delta X \sin(\phi)}{V_0 \cos(\theta - \phi)} \left(1 - \frac{V_0 \cos(\theta)}{V_1 \cos(\theta_1)}\right)$$

$$\Delta Z = \Delta X \tan(\phi)$$

Rayparameter parameterization:

$$\Delta t(p) = \frac{\Delta X \tan(\phi)}{V_0^2 (q_0 + p_0 \tan(\phi))} \left(1 - \frac{V_0^2 q_0}{V_1^2 q_1}\right)$$

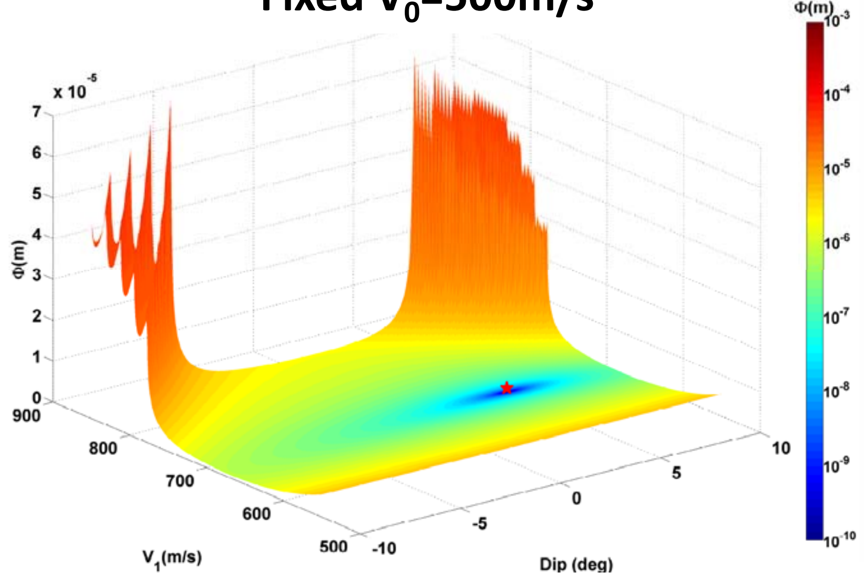
$$\cos(\theta_1) = \left[1 - (V_1 P)^2\right]^{1/2} \cos(\phi) - P \sin(\phi)$$

$$P = p_0 \cos(\phi) - q_0 \sin(\phi)$$

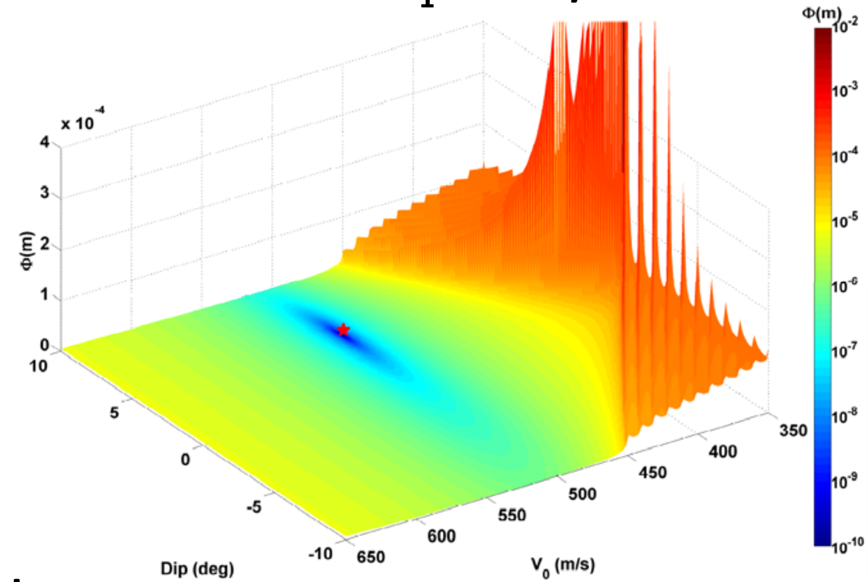
Introduction of the raypath angle θ_1 makes the problem highly non linear

Objective Function (Traveltime Differences Inversion)

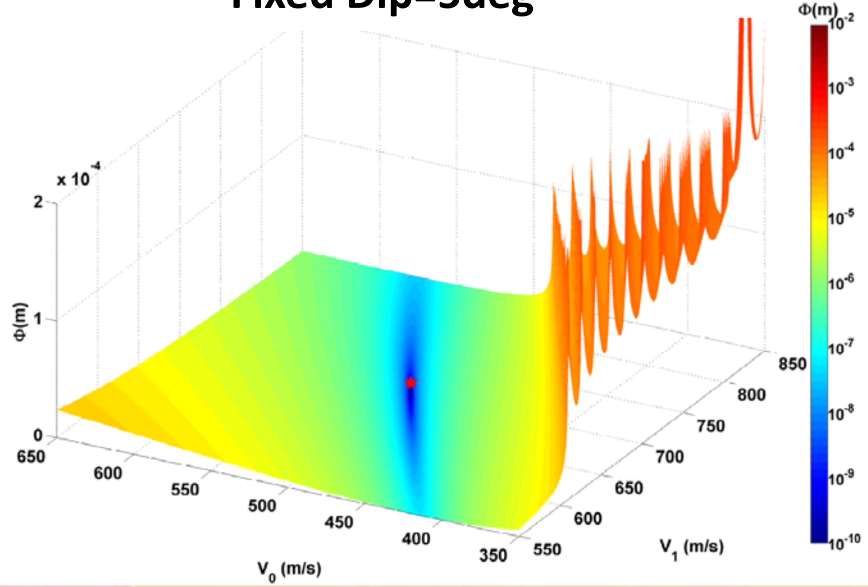
Fixed V_0=500m/s



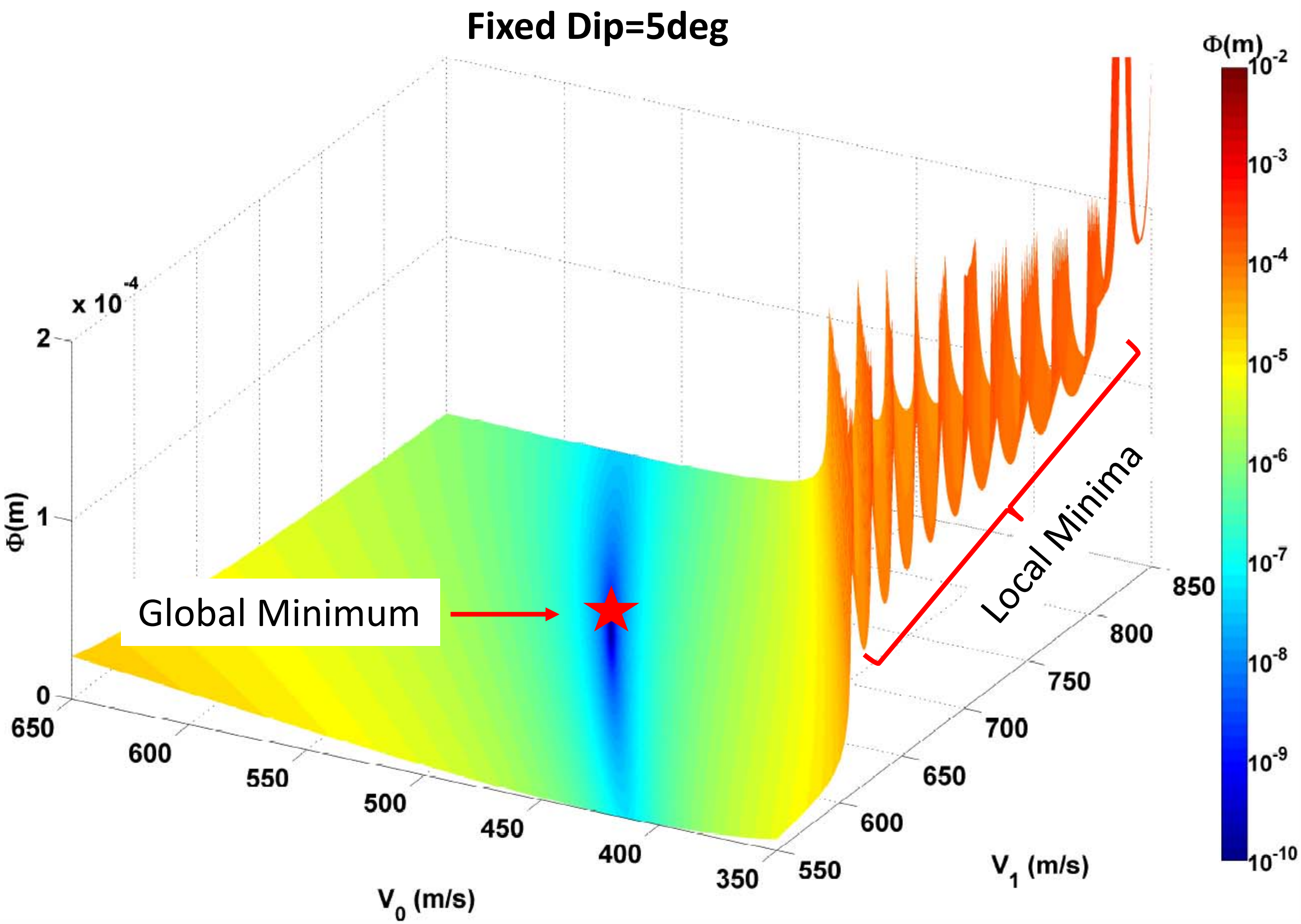
Fixed V_1=700m/s



Fixed Dip=5deg



Objective Function (Traveltime Differences Inversion)



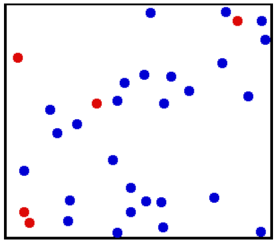
Complex "topography" of the objective function may be a problem for descent-based inversion methods.

Simulated Annealing

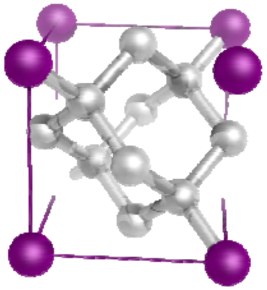
Physical annealing:

A solid material is heated past its melting point and then cooled back into a solid state.

While temperature is high atoms move randomly due to thermal motions



As temperature decrease atoms tend to fall into a regular configuration (crystal) that represents a minimum energy state



Images source: Wikipedia.org

Simulated Annealing

1. A temperature schedule that controls the algorithm is chosen: $\mathbf{T_i} = k\Phi(\mathbf{m_0})\left(\frac{\mathbf{N}-\mathbf{i}}{\mathbf{N}}\right)^2$

2. At each iteration new parameter values (m_{i+1}) are drawn from a Gaussian distribution and the objective function $\Phi(m_{i+1})$ is evaluated.

3. Decide:

if, $\Phi(m_{i+1}) \leq \Phi(m_i)$

Always accept the new model parameters

else,

compute $A = \exp\left(-\frac{\Phi(m_{i+1}) - \Phi(m_i)}{T}\right)$ and pick a random value (r) between 0 and 1.

If $A > r$

The new solution is accepted despite it leads to a higher value of $\Phi(m)$

else,

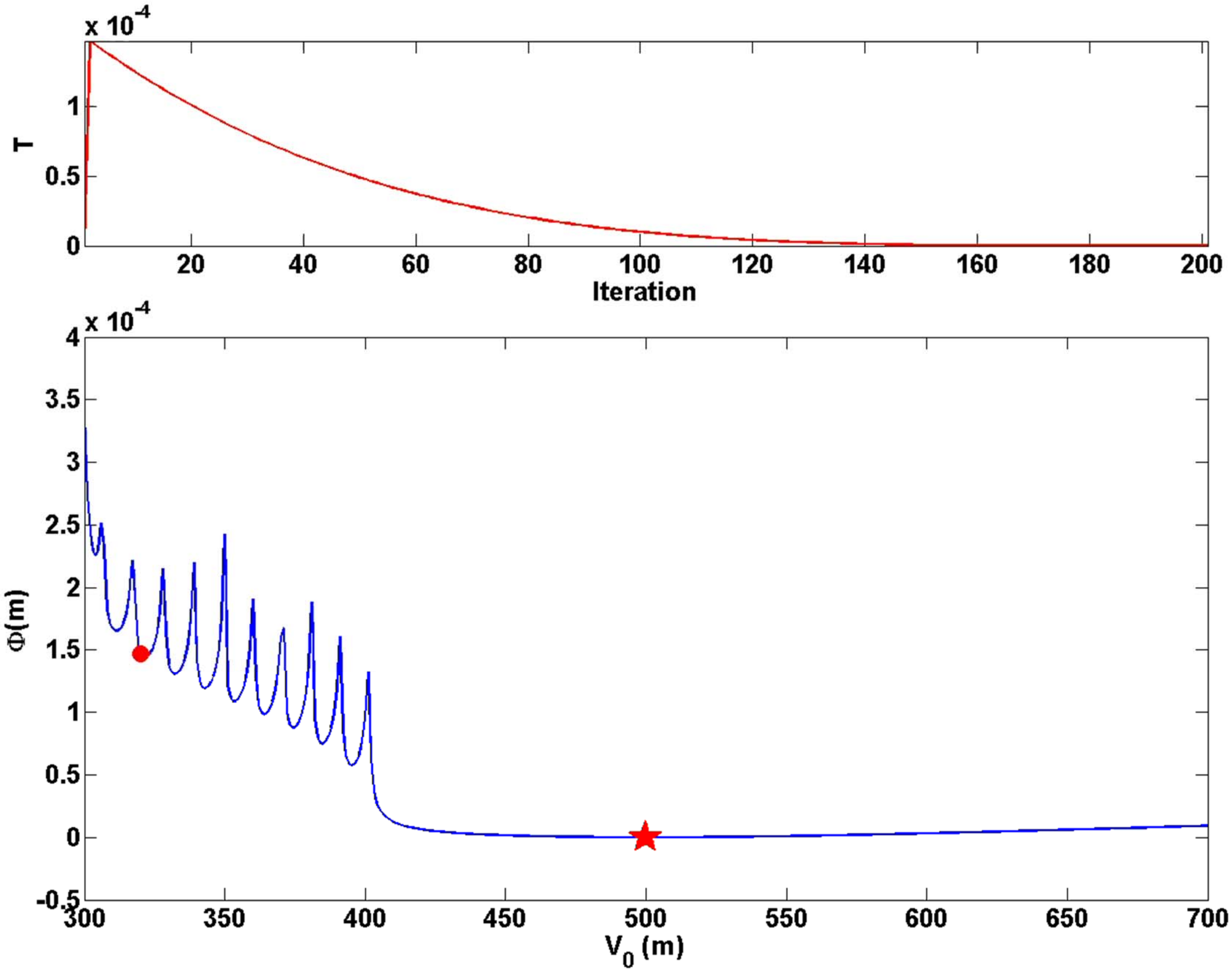
The new solution is rejected

end

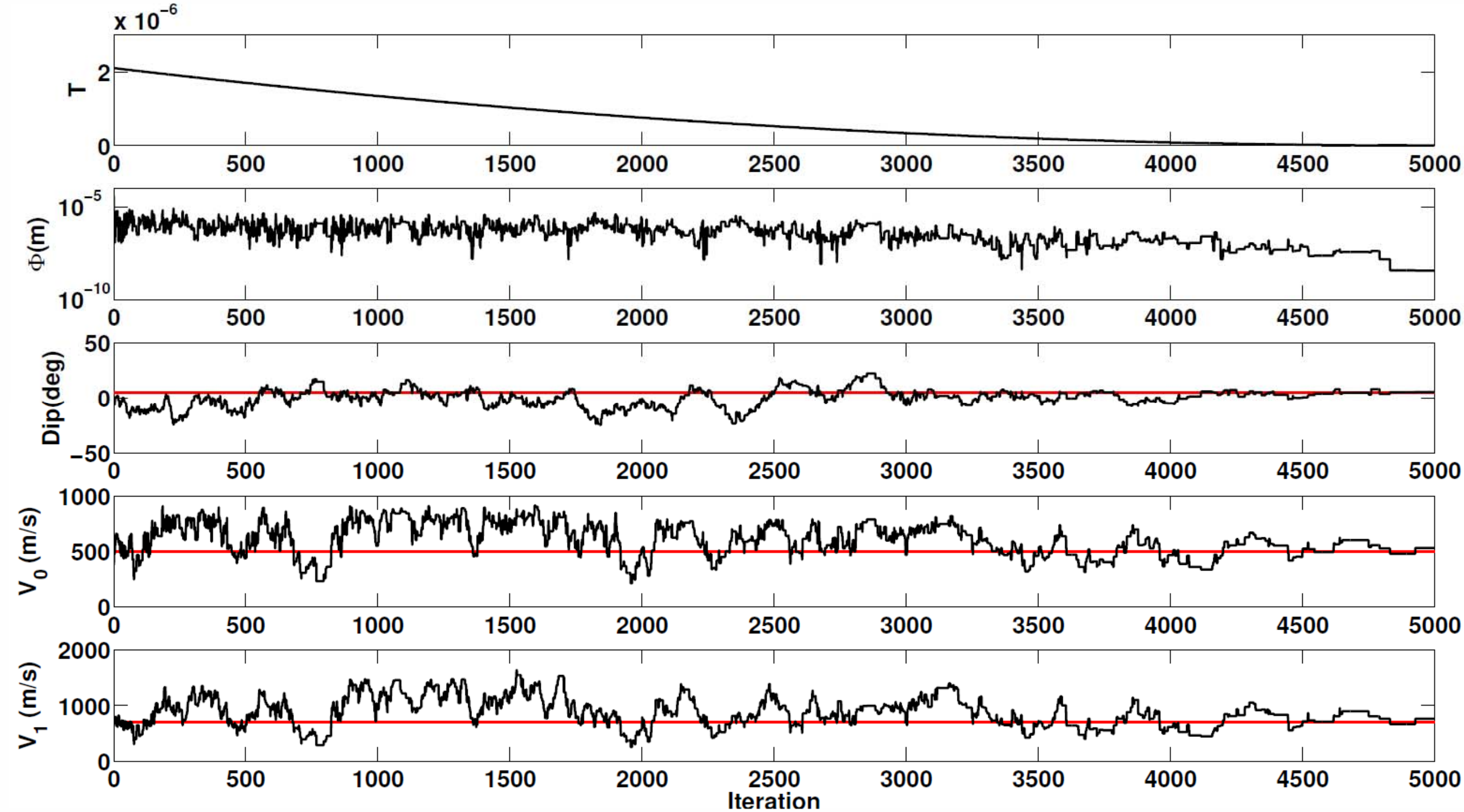
end

4. Update model parameter and iterate until freezing point is reached

Simulated Annealing

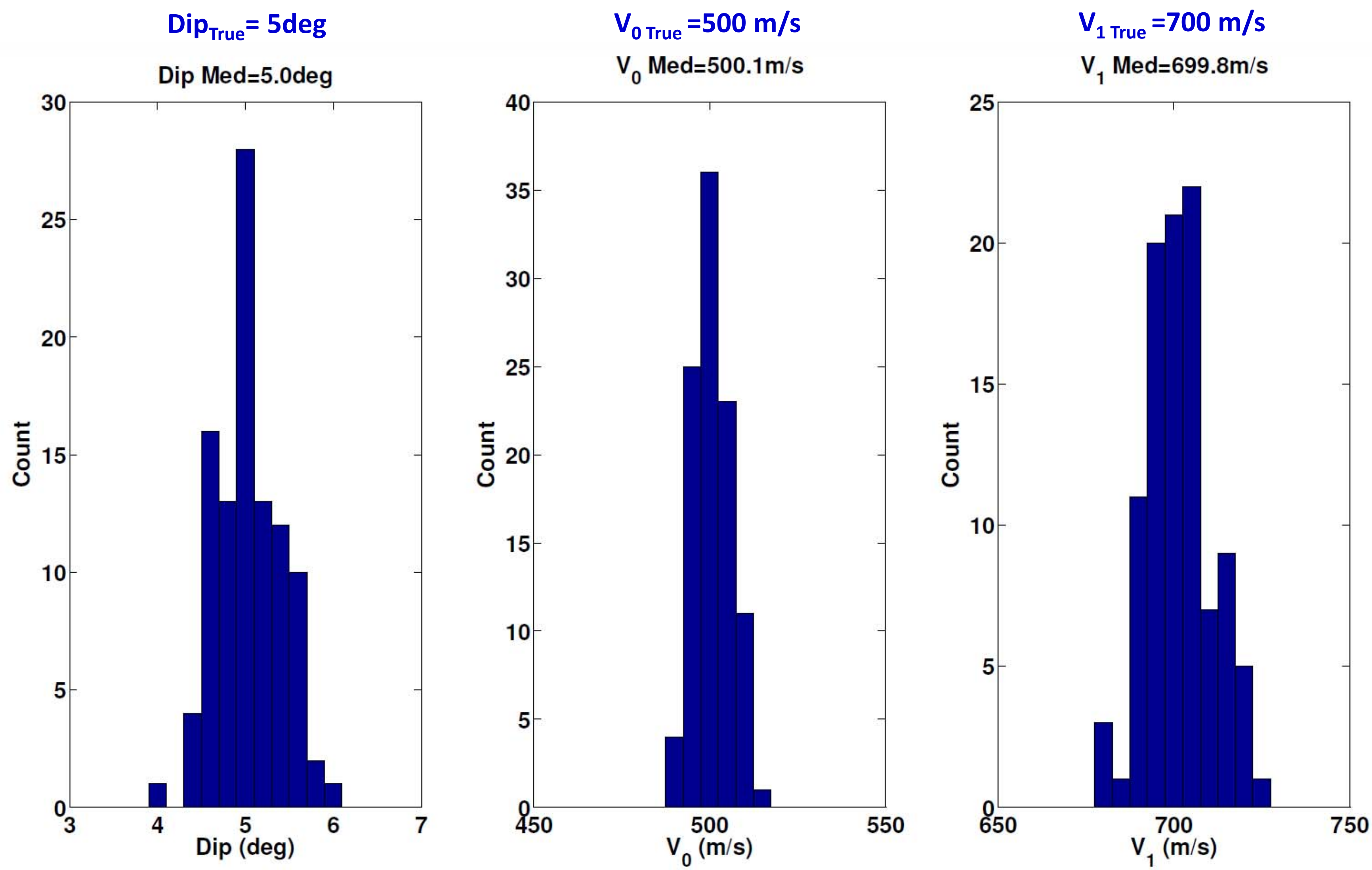


SA Results



As the temperatures approach zero the trial parameters converge toward the true parameters of the model

SA Statistics



CREWES

Conclusions

- The inversion of traveltimes in the rayparameter domain helped to constrain the inversion results.
- A quasi-Newton non-linear inversion successfully solved the initial problem.
- The SA algorithm used to invert reflection traveltime differences, proved to be effective in recovering the true parameters of the model.
- Traveltime differences can be retrieved from seismic data by using interferometric principles.

Acknowledgements

- David Henley
- NSERC (Grant CRDPJ 379744-08)
- CREWES sponsors
- CREWES staff and students.

Thanks!!!

CREWES

Sensitivity Matrix (Raypath Angle Parameterization)

$$\mathrm{J}(\mathrm{m}) = \left[\frac{\partial \mathrm{g}(\mathrm{m})}{\partial \mathrm{z}}, \quad \frac{\partial \mathrm{g}(\mathrm{m})}{\partial \mathrm{V}_0}, \quad \frac{\partial \mathrm{g}(\mathrm{m})}{\partial \phi}\right].$$

$$\frac{\partial t(\theta)}{\partial \phi} = -\frac{z}{V_0}\frac{\sin(\phi)}{\cos^2(\theta-\phi)}$$

$$\frac{\partial t(\theta)}{\partial z} = \frac{1}{V_0}\frac{\cos(\phi)}{\cos(\theta-\phi)}$$

$$\frac{\partial t(\theta)}{\partial V_0} = -\frac{z}{V_0^2}\frac{\cos(\phi)}{\cos(\theta-\phi)} = -\frac{z}{V_0}\frac{\partial t(\theta)}{\partial z}$$

The derivatives respect to the thickness and the velocity are linearly related

CREWES

Sensitivity Matrix (Rayparameter Parameterization)

$$\frac{\partial t(\theta)}{\partial \phi} = \frac{z}{V_0^2 \cos(\phi)^2} \frac{1}{(q + p\tan(\phi))}$$

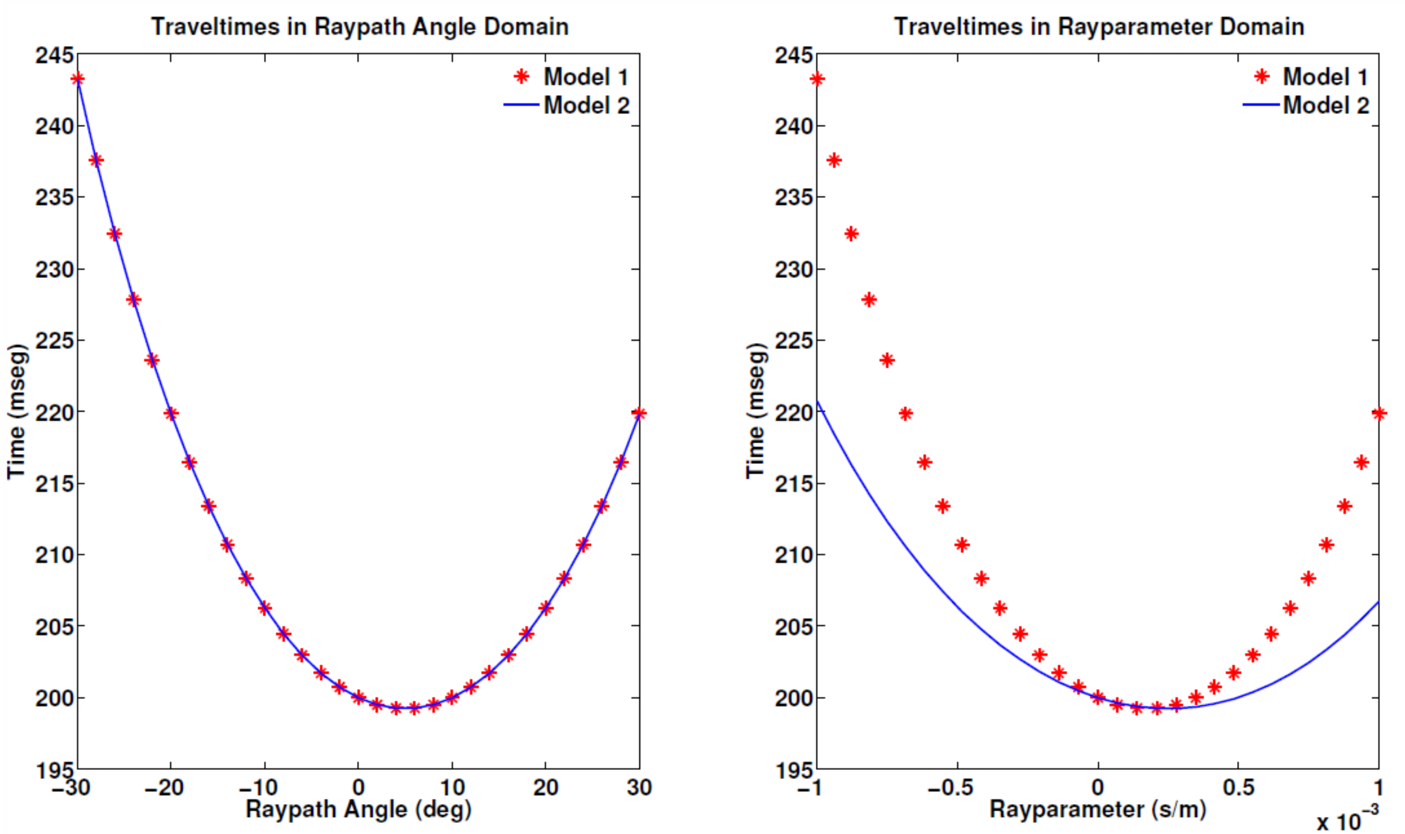
$$\frac{\partial t(\theta)}{\partial V_0} = \frac{z}{qV_0^5} \frac{[1 - 2qV_0^2(q + p\tan(\phi))]}{(q + p\tan(\phi))^2}$$

$$\frac{\partial t(\theta)}{\partial z} = \frac{1}{V_0^2} \frac{1}{(q + p\tan(\phi))}$$

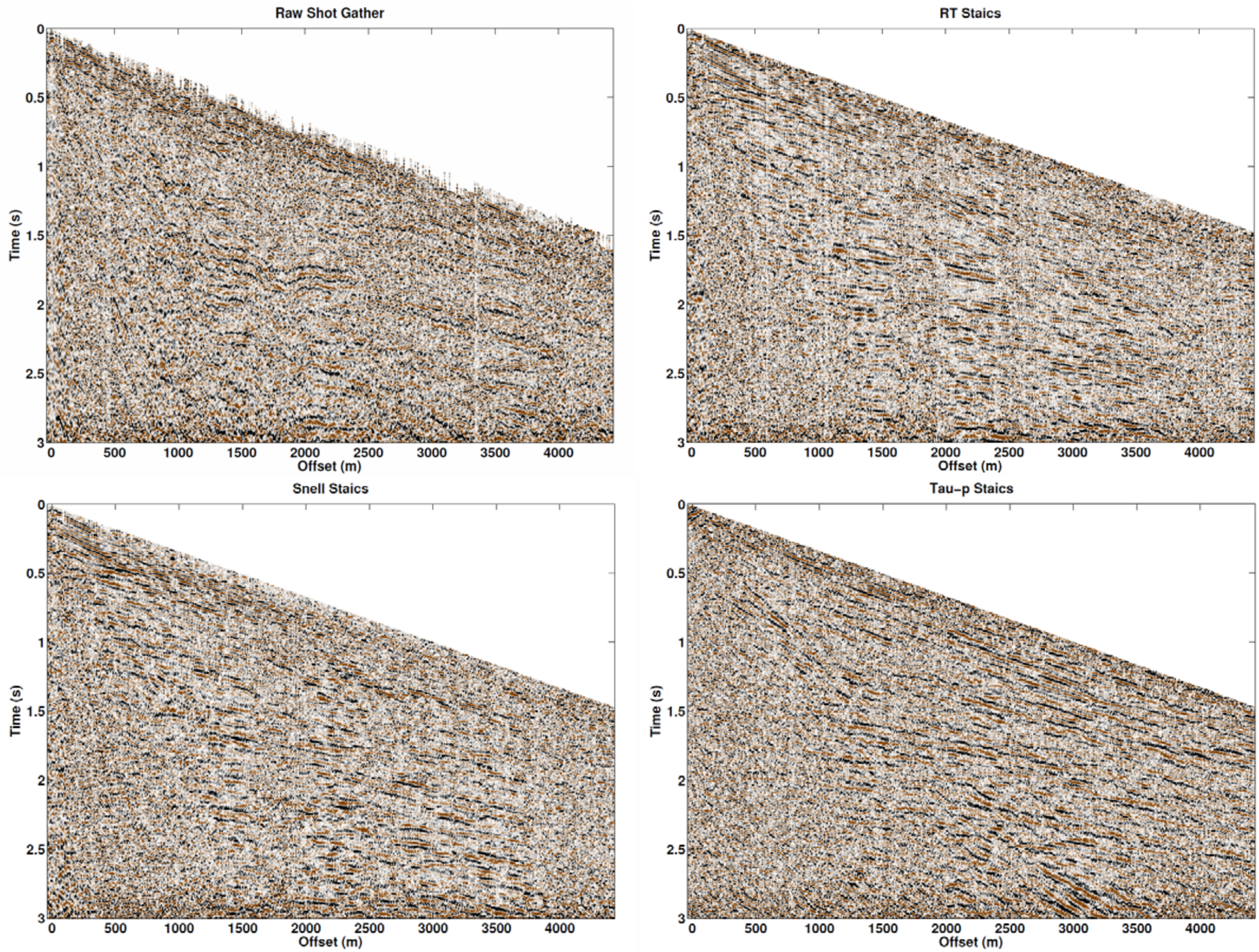
There is no linear relationship between the derivatives

Traveltimes Modelling

	Z (m)	V (m/s)	Dip (deg)
Model 1	100	500	5
Model 2	75	450	5



Ambiguities in the traveltimes can be solved in the rayparameter domain


Raw Shot Gather
RT Staics
Snell Staics
Tau–p Staics
Time (s)
Offset (m)
0
0.5
1
1.5
2
2.5
3
500
1000
1500
2000
2500
3000
3500
4000