Applications of Curvelets/ Surfacelets to seismic data processing

Evgeniy Lebed

SLIM
Seismic Laboratory for Imaging and Modeling

Outline

- Wavefield reconstruction
  - physical domain restriction
  - frequency domain restriction
- Wavefield separation
  - primary-multiple separation
  - Bayesian perspective

Signal Recovery

- Forward model

$$y = RMf_0 + n$$

- Analysis / Synthesis operators

$$S, S^H$$

- Inverse problem: $\tilde{f} = S^H \tilde{x}$ where

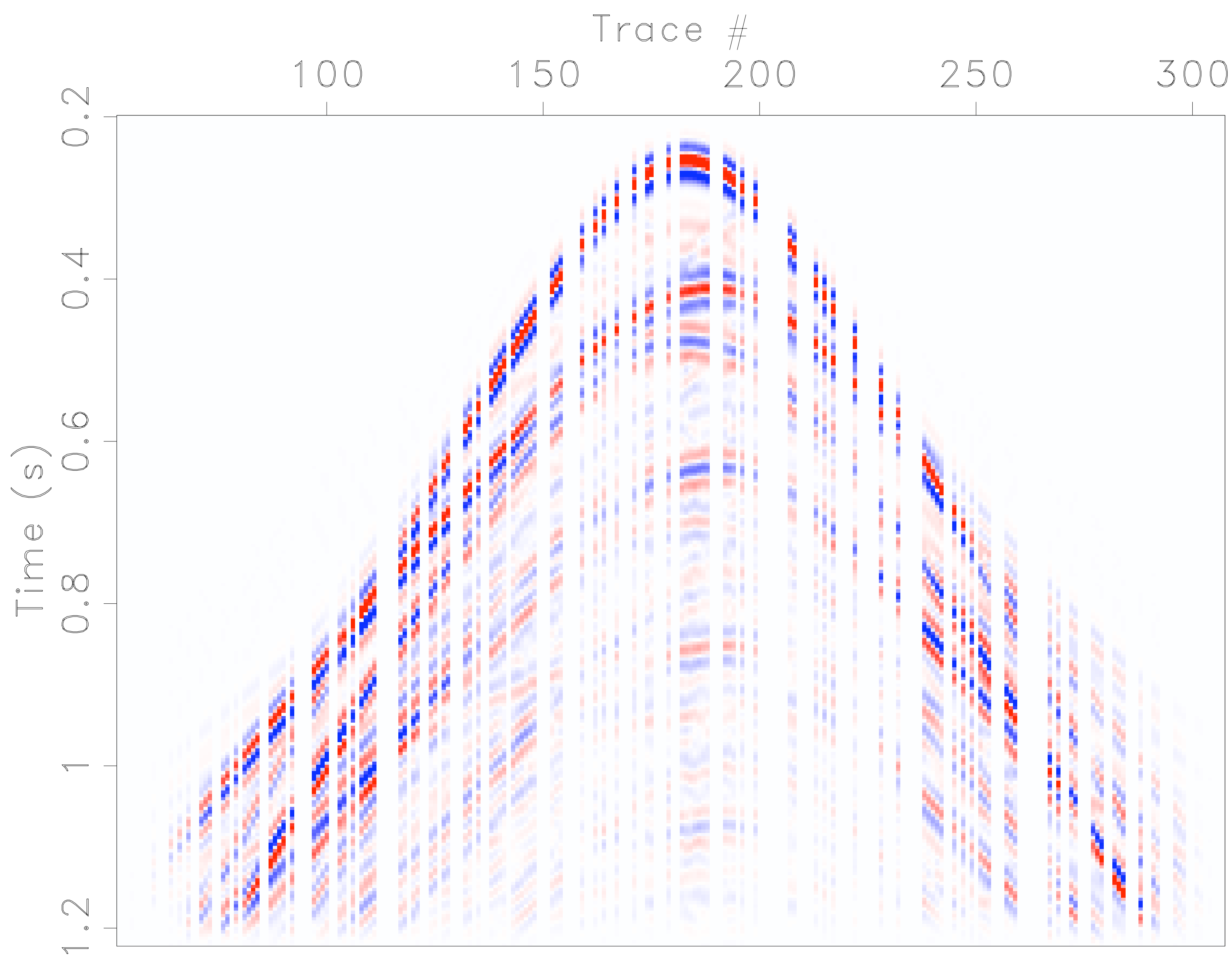
$$\tilde{x} = \arg\min_x \|x\|_1 \text{ such that } \|y - \underbrace{RMS^H}_{A} x\|_2 \leq \epsilon$$

Solver

$\mathbf{x}_0 :=$ initial guess
$\lambda_0 :=$ initial Lagrange multiplier
while $\mathbf{r} > \epsilon$
$\quad \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda_k)$
$\quad \lambda_{k+1} = \alpha_k \lambda_k$ with $0 < \alpha_k < 1$
end while.

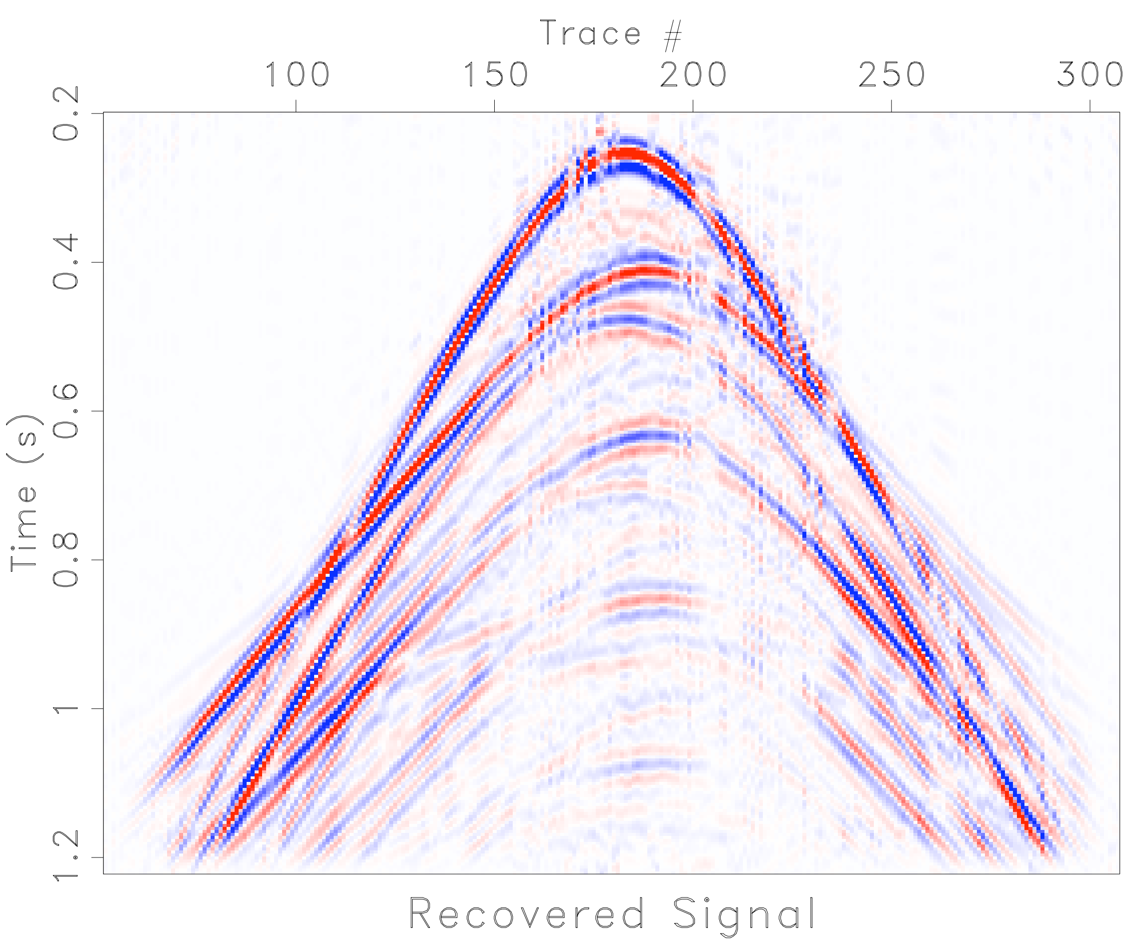
$$\mathbf{x}_{i+1} = \mathcal{S}_{\lambda_k}\left(\mathbf{x}_i + \mathbf{A}^H(\mathbf{y} - \mathbf{A}\mathbf{x}_i)\right)$$
$$\mathcal{S}_{\lambda_k}(x) := \mathrm{sign}(x) \cdot \max(|x| - \lambda_k, 0)$$

Signal Recovery - physical domain restriction

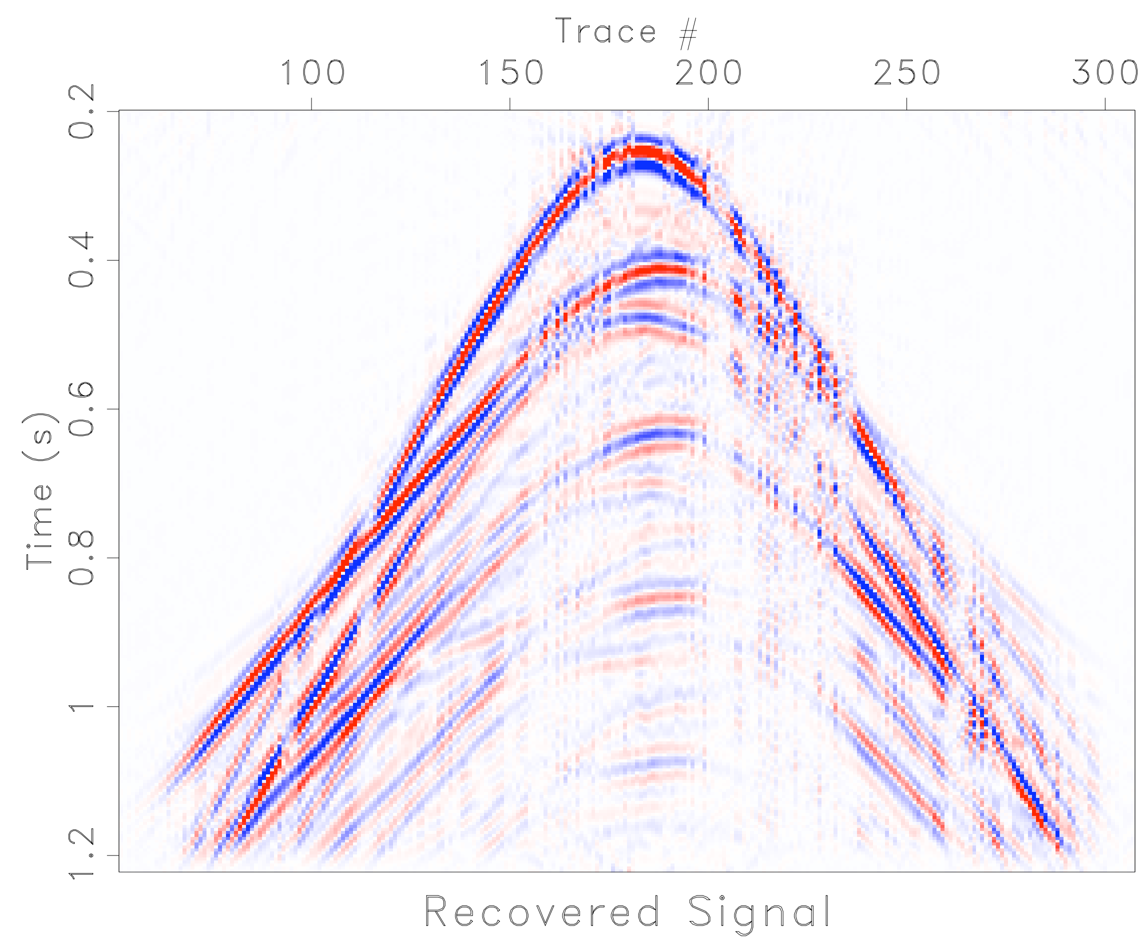


Data in x−t domain

Signal Recovery - physical domain restriction

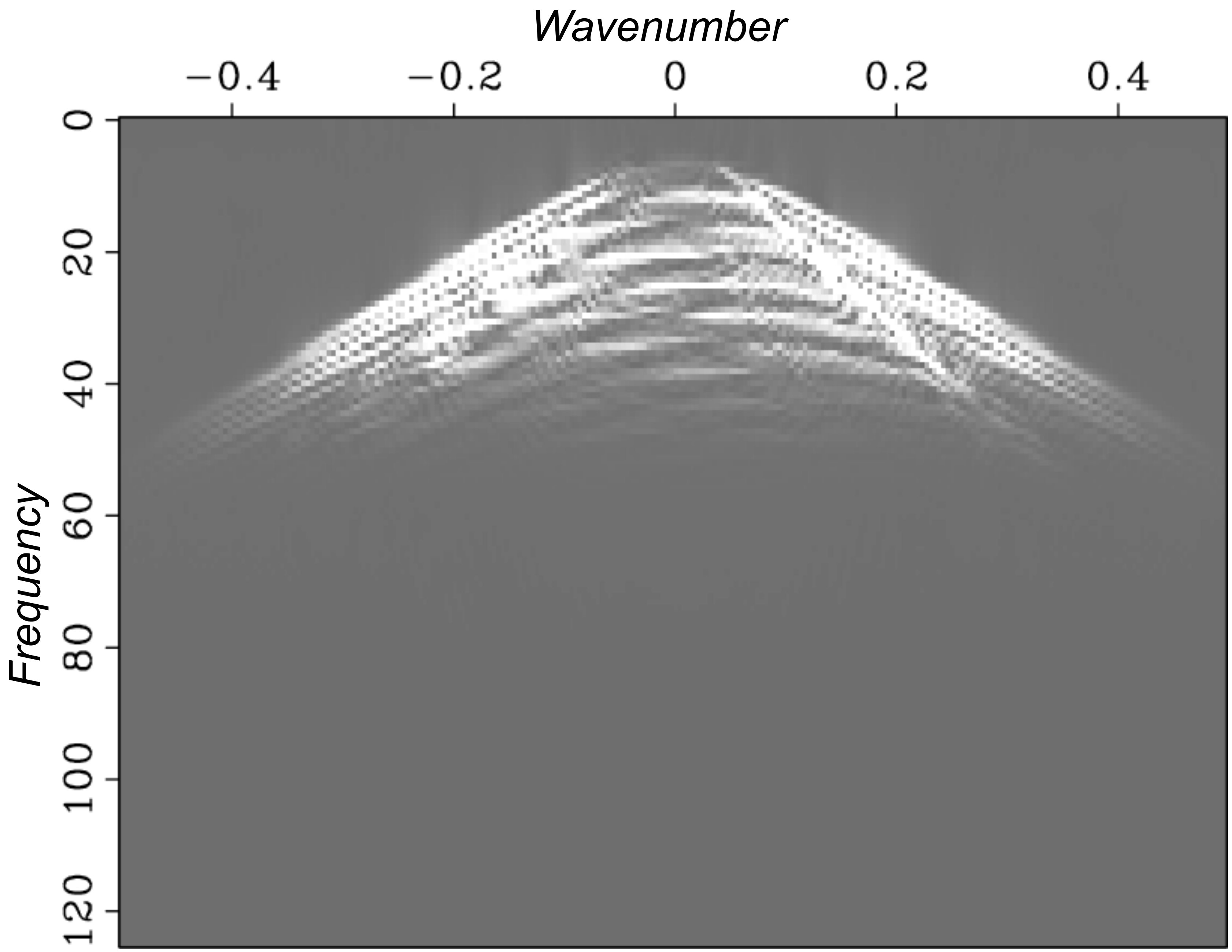


Curvelets

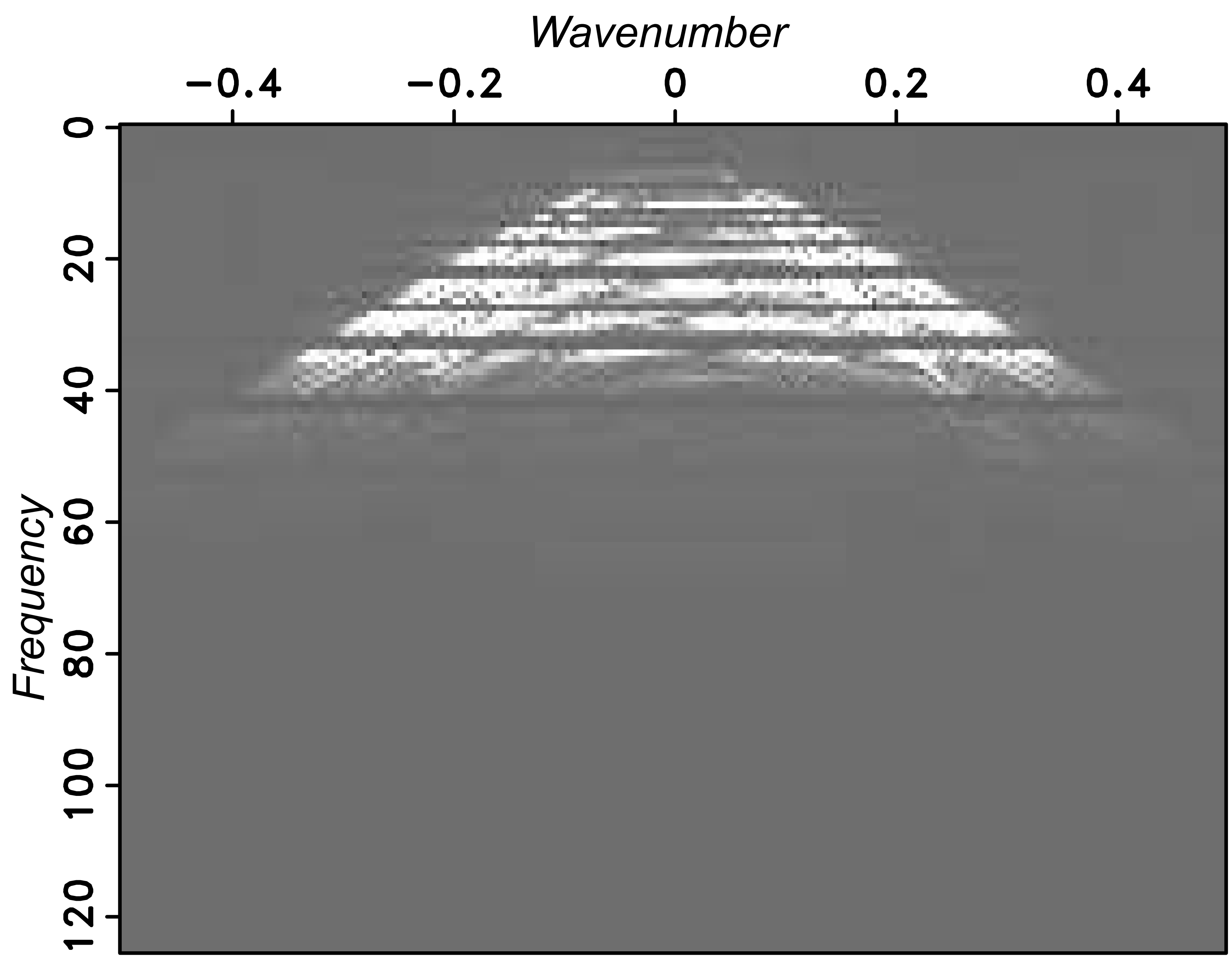


Surfacelets

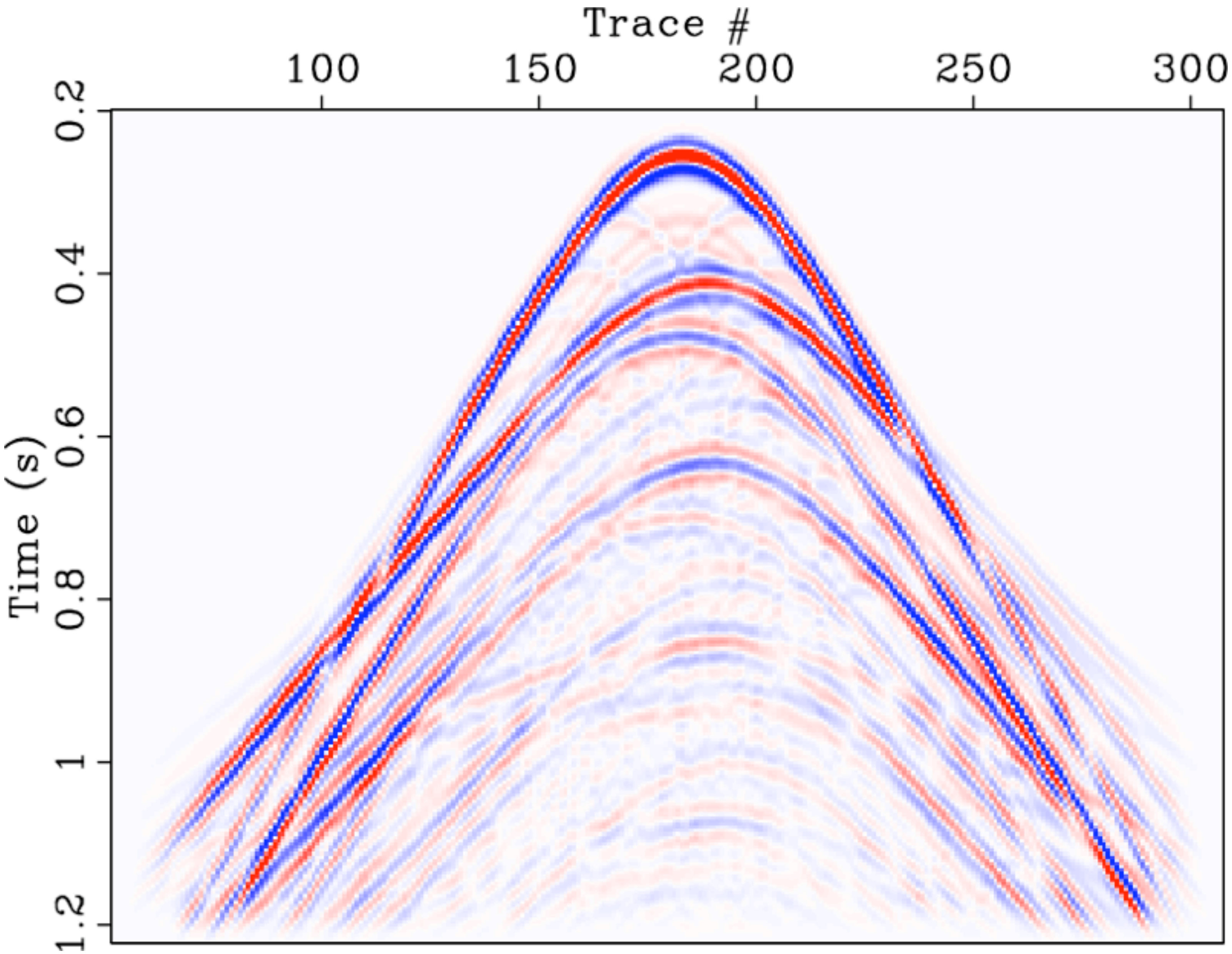
Signal Recovery - frequency domain restriction



Signal Recovery - frequency domain restriction

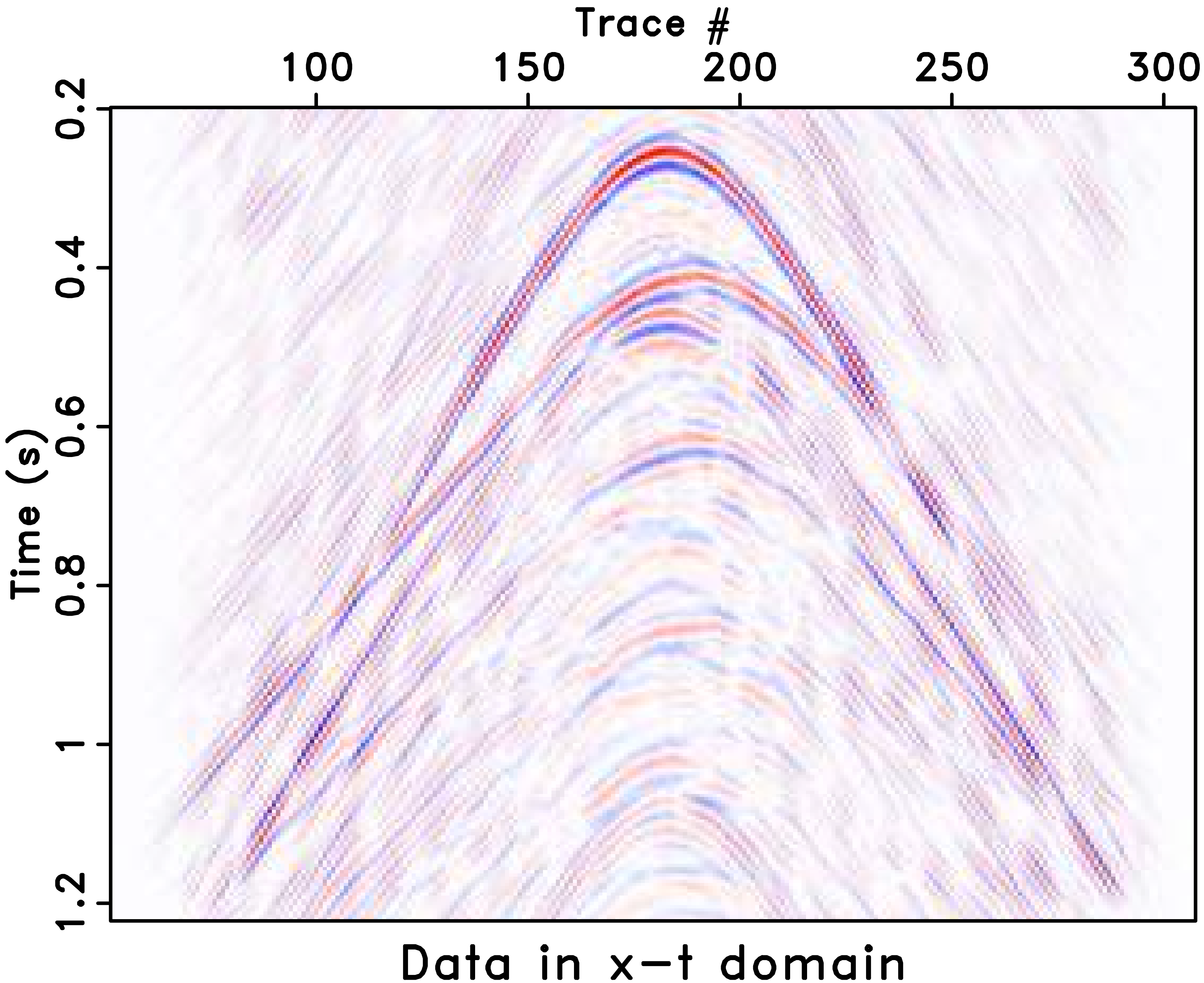


Signal Recovery - frequency domain restriction



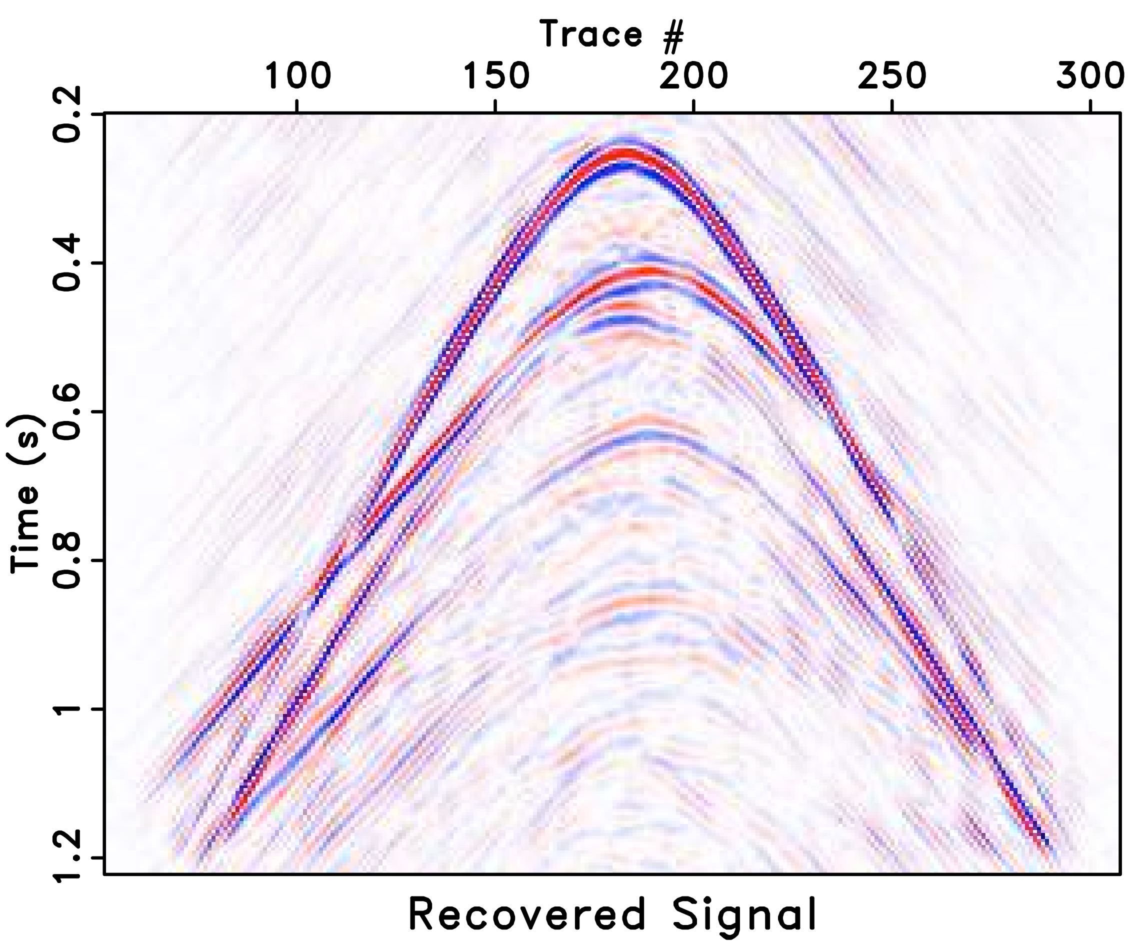
SLIM
Seismic Laboratory for Imaging and Modeling

Signal Recovery - frequency domain restriction

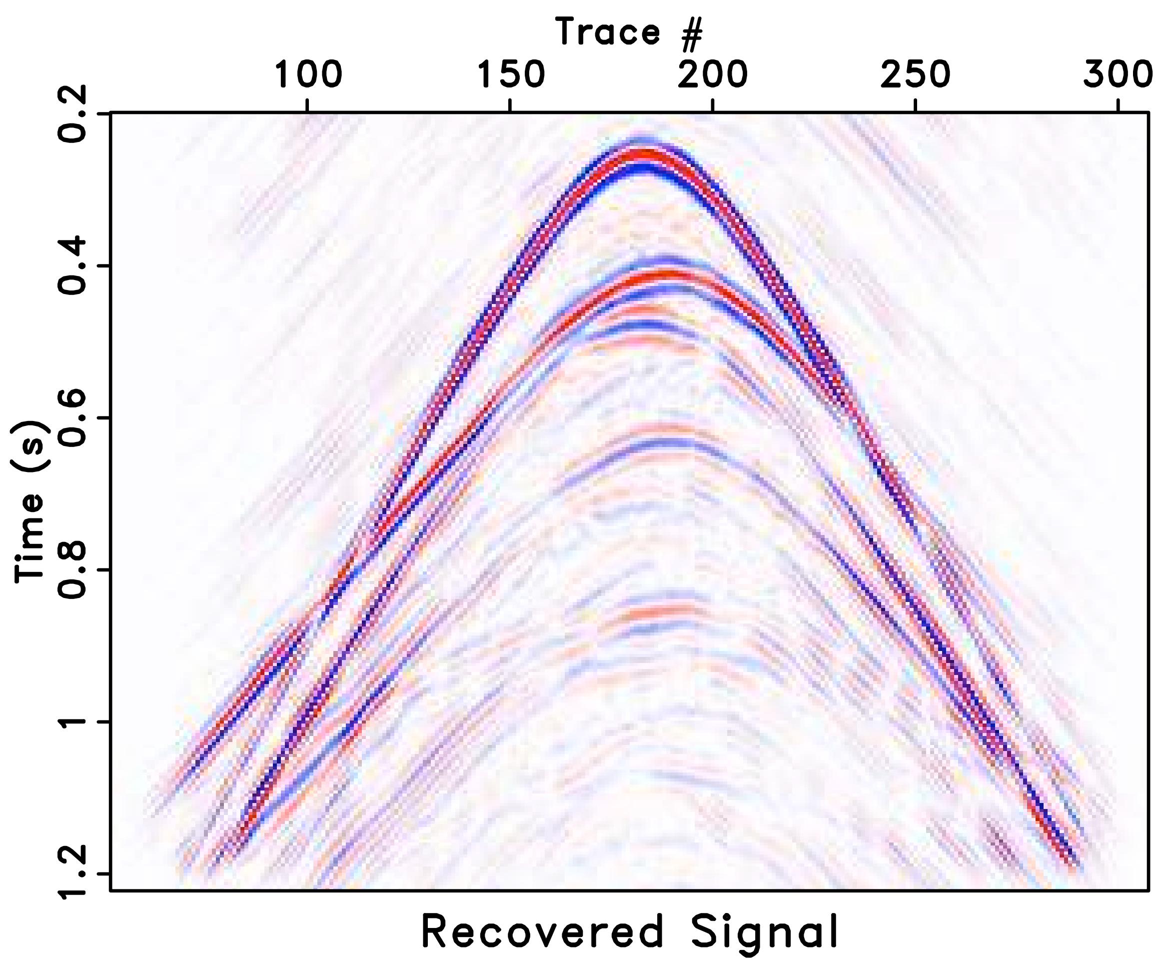


SLIM
Seismic Laboratory for Imaging and Modeling

Signal Recovery - frequency domain restriction



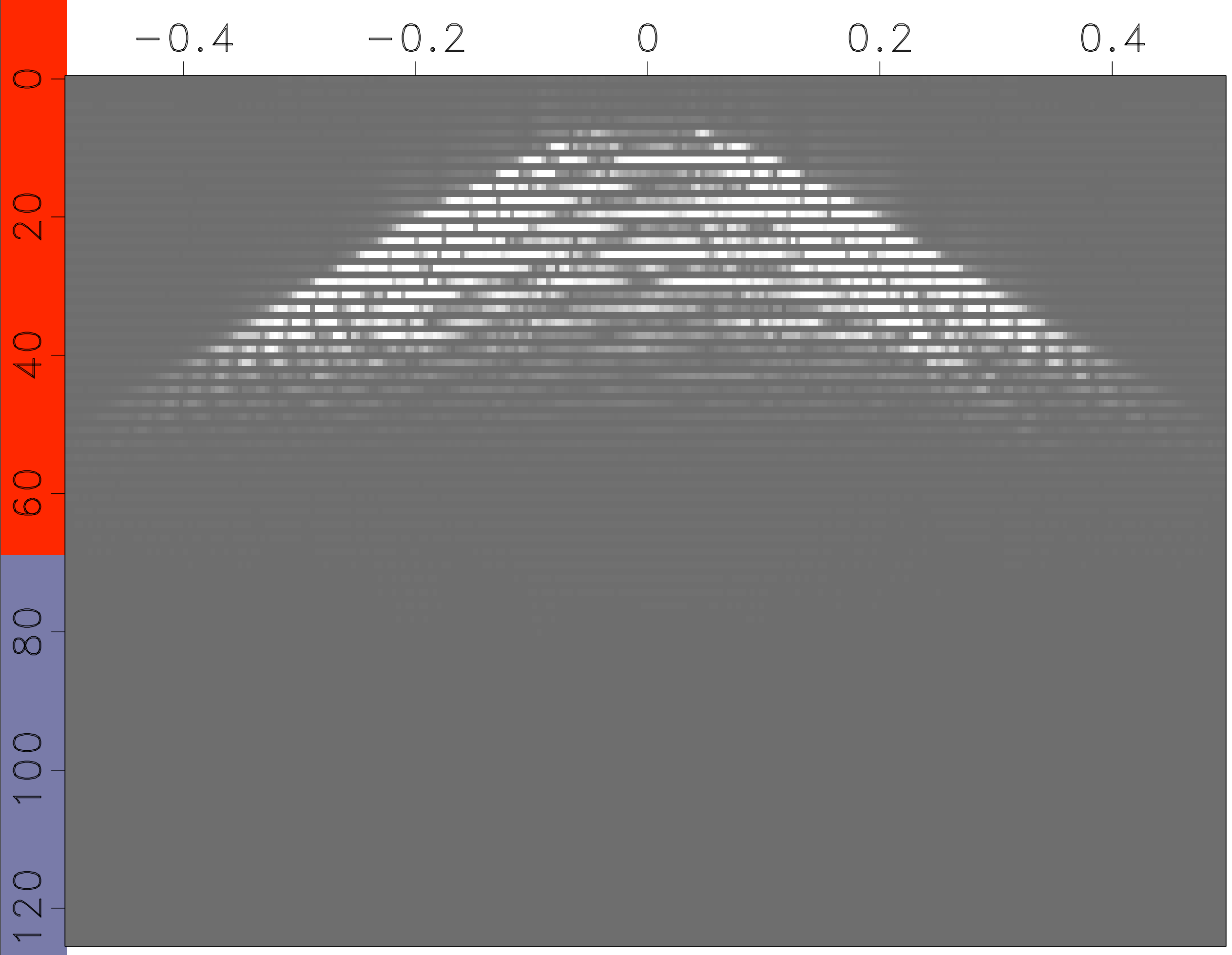
Curvelets

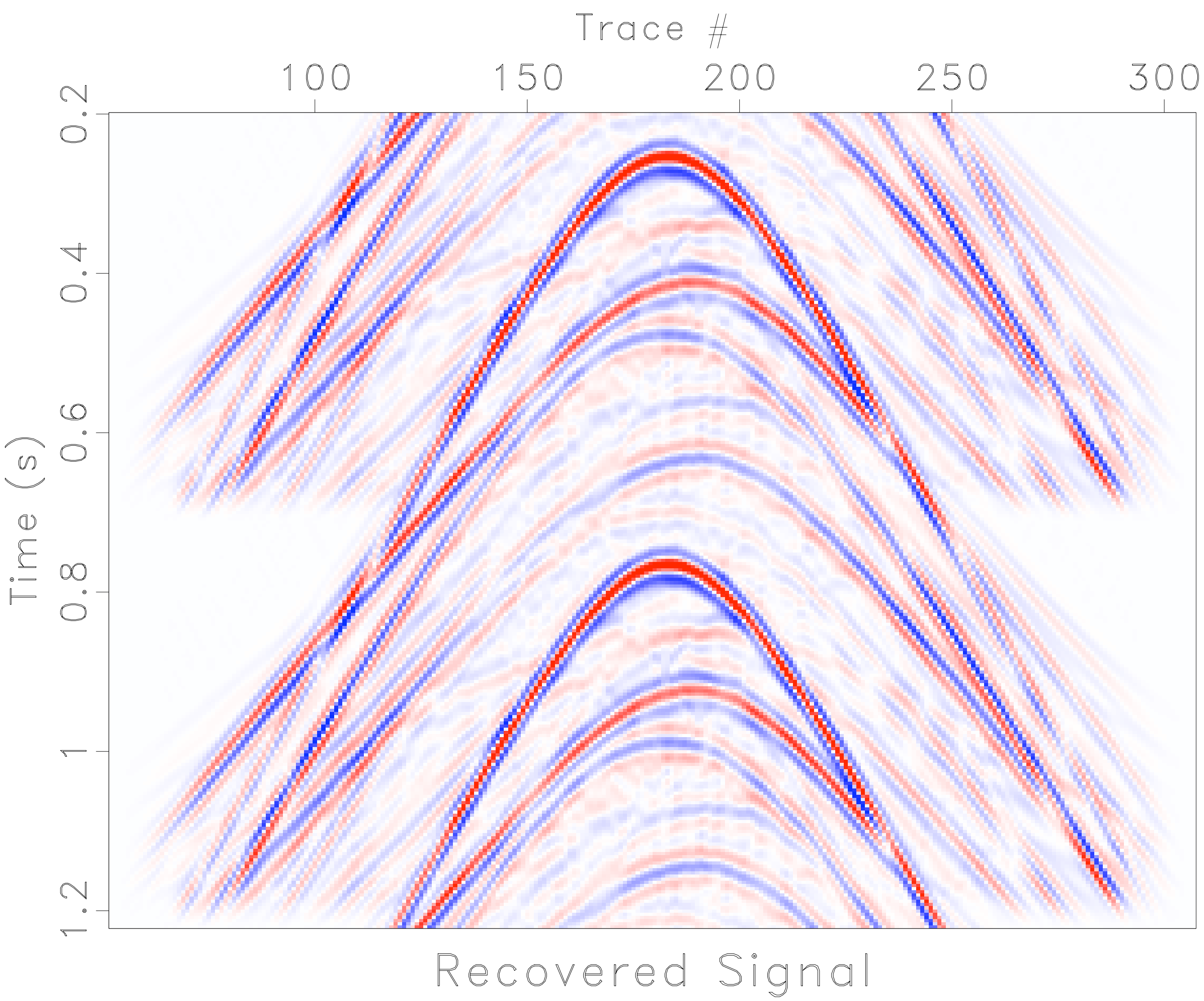


Surfacelets

The importance of irregular subsampling

- regular subsampling example





Primary-Multiple separation Bayesian Perspective

forward model

$$\mathbf{b} = \mathbf{s}_1 + \mathbf{s}_2 + \mathbf{n} \quad \text{(total data)}$$

$$\mathbf{b}_1 = \mathbf{A}\mathbf{x}_1 + \mathbf{n}_1 \quad \text{(predicted primaries)}$$

$$\mathbf{b}_2 = \mathbf{A}\mathbf{x}_2 + \mathbf{n}_2 \quad \text{(predicted multiples)}$$

$\mathbf{x}_1$ curv/surf coefficients of primaries

$\mathbf{x}_2$ curv/surf coefficients of multiples

$\mathbf{A}$ inverse curv/surf transform

inverse problem

$$\mathbf{P_w}: \begin{cases} \tilde{\mathbf{x}} = \arg\min_{\mathbf{x}} \lambda_1 \|\mathbf{x}_1\|_{1,\mathbf{w}_1} + \lambda_2 \|\mathbf{x}_2\|_{1,\mathbf{w}_2} + \\ \|\mathbf{A}\mathbf{x}_2 - \mathbf{b}_2\|_2^2 + \eta\|\mathbf{A}(\mathbf{x}_1 + \mathbf{x}_2) - \mathbf{b}\|_2^2 \\ \tilde{\mathbf{s}}_1 = \mathbf{A}\tilde{\mathbf{x}}_1 \quad \text{and} \quad \tilde{\mathbf{s}}_2 = \mathbf{A}\tilde{\mathbf{x}}_2. \end{cases}$$

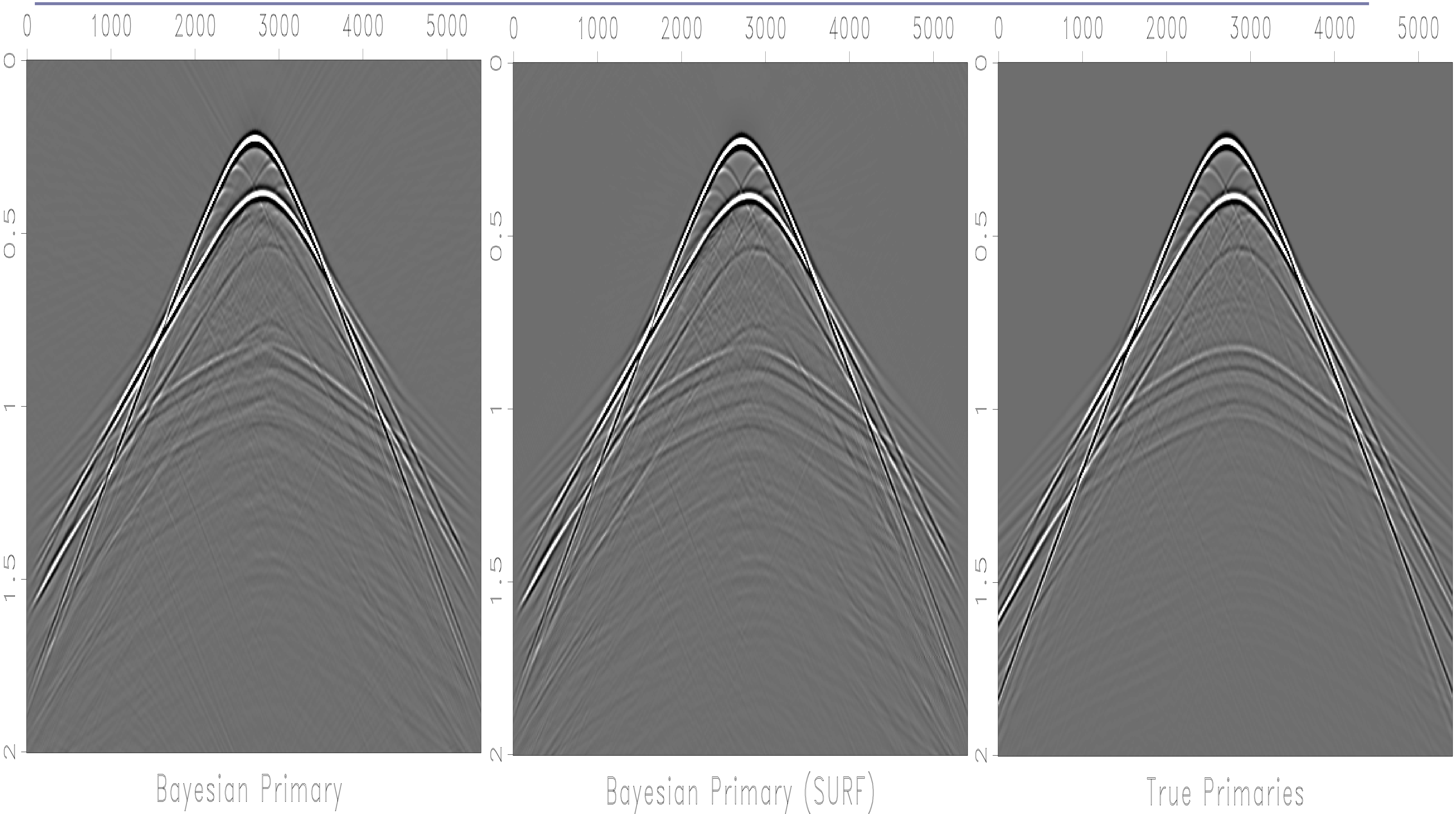
solver

$$\mathbf{x}_1^{n+1} = \mathbf{T}_{\frac{\lambda_1 \mathbf{w}_1}{2\eta}} \left[\mathbf{A}^T\mathbf{b}_2 - \mathbf{A}^T\mathbf{A}\mathbf{x}_2^n + \mathbf{A}^T\mathbf{b}_1 - \mathbf{A}^T\mathbf{A}\mathbf{x}_1^n + \mathbf{x}_1^n \right]$$

$$\mathbf{x}_2^{n+1} = \mathbf{T}_{\frac{\lambda_2 \mathbf{w}_2}{2(1+\eta)}} \left[\mathbf{A}^T\mathbf{b}_2 - \mathbf{A}^T\mathbf{A}\mathbf{x}_2^n + \mathbf{x}_2^n + \frac{\eta}{\eta+1}\left(\mathbf{A}^T\mathbf{b}_1 - \mathbf{A}^T\mathbf{A}\mathbf{x}_1^n\right) \right]$$

(Saab et al.,2007)

Primary-Multiple separation Bayesian Perspective



SLIM
Seismic Laboratory for Imaging and Modeling

Summary

	Curvelets SNR	Surfacelets SNR
Physical Restriction	6.4	4.2
Frequency Restriction	5.4	6.8
PMS Bayesian	11.6	12.2

SLIMpy note

- Interchangeability between curvelet / surfacelet operators is simple!

```
def main( data, mask, output,
        transparams=[4,16,1], angconst=[90,90], coutput=None,
        thrparams=[.01,.99], solverparams=[10,5] ):

    # define curvelet matrix (analysis/decomposition)
    C = fdct2(data.getSpace(), *transparams)

    # define thresholding weights as norms of columns of curvelet
    # synthesis matrix
    ThrWeights = C.norm()

    # define angular constraint in the curvelet domain
    AngWeights = C.minvelconst(ang=angconst)

    # define picking operator
    P = pickingoper(data.getSpace(),mask)

    # define global operator
    A = CompoundOperator([P,C.adj(),AngWeights])

    # define the threshold scheme to pass to the lanweber solver
    thresh = logcooling(thrparams[0],thrparams[1],ThrWeights=ThrWeights)

    # define the solver to use
    solver = GenThreshLandweber(solverparams[0],solverparams[1],thresh=thresh)

    # run the interpolation
    x = solver.solve(A,data)

    # return solution in the transform domain if wanted
    if coutput:
        x.setName(os.path.abspath(coutput))

    # compute solution in the (t,x) domain
    final = (C.adj() * x)

    # return solution in the (t,x) domain
    final.setName(output)

    End()
```

```
def main( data, mask, output,
        surf_k="4,4,3,2,1", surf_pyr=5, angconst=[90,90], coutput=None,
        thrparams=[.01,.99], solverparams=[10,5] ):

    # define surfacelet matrix (analysis/decomposition)
    C =  surf(data.getSpace(),K=surf_k,Pyr_Level=surf_pyr)

    # define thresholding weights as norms of columns of surfacelet
    # synthesis matrix
    ThrWeights = C.norm()

    # define angular constraint in the curvelet domain
    AngWeights = C.minvelconst(ang=angconst)

    # define picking operator
    P = pickingoper(data.getSpace(),mask)

    # define global operator
    A = CompoundOperator([P,C.adj(),AngWeights])

    # define the threshold scheme to pass to the lanweber solver
    thresh = logcooling(thrparams[0],thrparams[1],ThrWeights=ThrWeights)

    # define the solver to use
    solver = GenThreshLandweber(solverparams[0],solverparams[1],thresh=thresh)

    # run the interpolation
    x = solver.solve(A,data)

    # return solution in the transform domain if wanted
    if coutput:
        x.setName(os.path.abspath(coutput))

    # compute solution in the (t,x) domain
    final = (C.adj() * x)

    # return solution in the (t,x) domain
    final.setName(output)

    End()
```

Conclusions

- Wavefield reconstruction
- Irregular subsampling is key!
  - Physical domain restriction
    - Curvelets SNR: higher | Surfacelets SNR: lower
  - Frequency domain restriction
    - Curvelets SNR: lower | Surfacelets SNR: higher
- Bayesian wavefield separation
    - Curvelets SNR: lower | Surfacelets SNR: higher

Acknowledgments

- This presentation was carried out as part of the SINBAD project with financial support, secured through ITF, from the following organizations: BG, BP, Chevron, ExxonMobil, and Shell. SINBAD is part of the collaborative research & development (CRD) grant number 334810-05 funded by the Natural Science and Engineering Research Council (NSERC).