Event-consistent smoothing in generalized high-density velocity analysis

J. Mann E. Duveneck[1]

[1]now: SINTEF Petroleum Research, Trondheim, Norway

Wave Inversion Technology (WIT) Consortium
Geophysical Institute, University of Karlsruhe (TH)

October 12, 2004

Overview

Introduction

Smoothing algorithm

Data examples

Conclusions

Acknowledgments

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only
- ☞ human interaction required
- ☞ low temporal and spatial resolution
- ☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Mann & Duveneck

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Mann & Duveneck

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only

☞ human interaction required

☞ low temporal and spatial resolution

☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- ► (semi-)interactive, interpretative velocity picking
- ► coarse picks on selected key events, only
- ☞ human interaction required
- ☞ low temporal and spatial resolution
- ☞ pulse stretch deteriorates stack result

Thus desirable:

- ► automated approach
- ► more appropriate parameterization
- ► maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only
- ☞ human interaction required
- ☞ low temporal and spatial resolution
- ☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Introduction

Conventional stacking velocity analysis:

- (semi-)interactive, interpretative velocity picking
- coarse picks on selected key events, only
- ☞ human interaction required
- ☞ low temporal and spatial resolution
- ☞ pulse stretch deteriorates stack result

Thus desirable:

- automated approach
- more appropriate parameterization
- maximum resolution

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator

- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator

- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

Mann & Duveneck

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - much more prestack traces used
  - enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  ☞ geometrical interpretation

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - ☞ much more prestack traces used
  - ☞ enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - ☞ much more prestack traces used
  - ☞ enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - much more prestack traces used
  - enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  - geometrical interpretation

Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - ☞ much more prestack traces used
  - ☞ enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

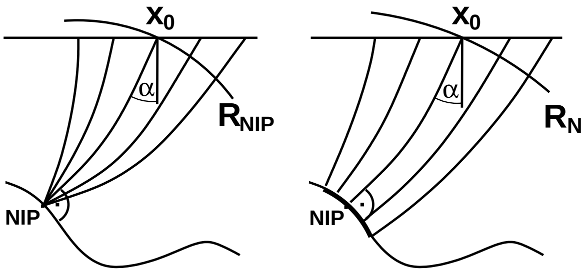
Common-Reflection-Surface (CRS) stack

Generalization of conventional approach:

- second-order approximation of traveltime
- fully automated coherence-based application
- high-density analysis
- *spatial* stacking operator
  - ☞ much more prestack traces used
  - ☞ enhanced signal/noise ratio
- additional stacking parameters related to 1. and 2. traveltime derivatives
  - ☞ geometrical interpretation

Common-Reflection-Surface stack

Geometrical interpretation of stacking parameters:

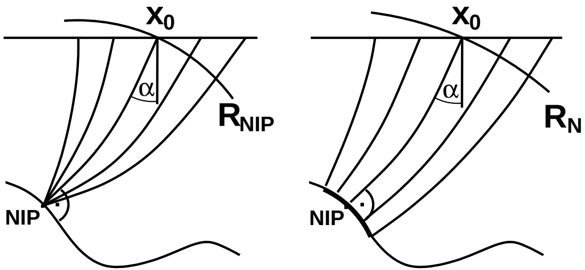


Emergence direction and curvatures of hypothetical wavefronts:

- exploding point source ☞ normal-incidence-point (NIP) wave
- exploding reflector ☞ normal (N) wave

Common-Reflection-Surface stack

Geometrical interpretation of stacking parameters:

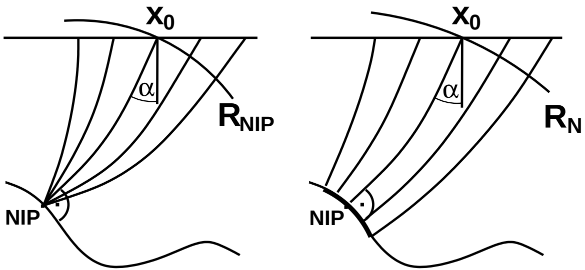


Emergence direction and curvatures of hypothetical wavefronts:

- exploding point source ☞ normal-incidence-point (NIP) wave
- exploding reflector ☞ normal (N) wave

Common-Reflection-Surface stack

Geometrical interpretation of stacking parameters:

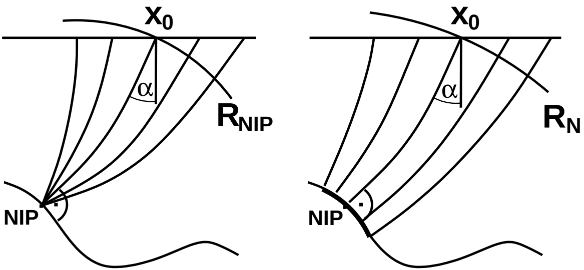


Emergence direction and curvatures of hypothetical wavefronts:

- exploding point source ☞ normal-incidence-point (NIP) wave
- exploding reflector ☞ normal (N) wave

Common-Reflection-Surface stack

Geometrical interpretation of stacking parameters:



Emergence direction and curvatures of hypothetical wavefronts:

- exploding point source ☞ normal-incidence-point (NIP) wave
- exploding reflector ☞ normal (N) wave

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

➡ smooth variation along reflection events

➡ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

➥ smooth variation along reflection events

➥ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

- ➥ smooth variation along reflection events
- ➥ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

➥ smooth variation along reflection events

➥ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

➥ smooth variation along reflection events

➥ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Stacking parameters are subject to

- fluctuations due to noise
- outliers due to failures to detect the relevant coherence maximum

Stacking parameters represent integral properties of the subsurface

➥ smooth variation along reflection events

➥ *event-consistent* smoothing along reflection events is justified!

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters
  - virtually constant along wavelet
  - smoothing also allowed along wavelet without pulse stretch

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters
  - virtually constant along wavelet
  - smoothing also allowed along wavelet without pulse stretch

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters
  - virtually constant along wavelet
  - smoothing also allowed along wavelet without pulse stretch

High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters
  - virtually constant along wavelet
  - smoothing also allowed along wavelet without pulse stretch

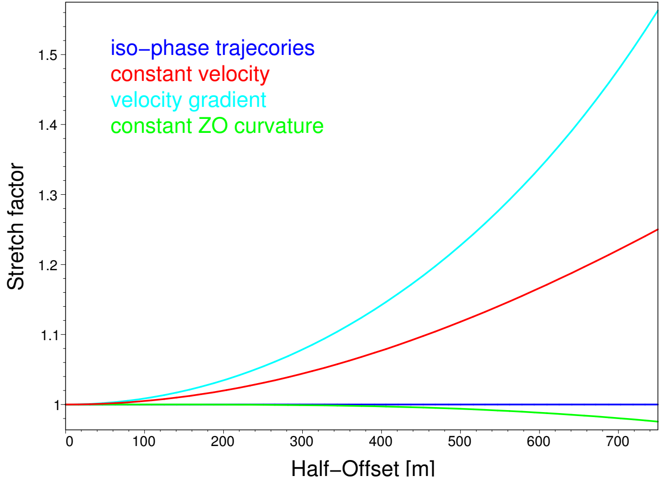
High-density analysis vs. smoothing

Bandwidth is limited. What happens along the wavelet?

- high-density stacking velocity
  - systematic variation along wavelet
  - smoothing reintroduces pulse stretch phenomenon
- CRS stacking parameters
  - virtually constant along wavelet
  - smoothing also allowed along wavelet without pulse stretch

Pulse stretch phenomenon

Smooth model: stacking velocity vs. CRS parameters



From: Mann and Höcht, 2003, *Pulse stretch effects in the context of data-driven imaging methods*, 65th Conf., Eur. Assn. Geosci. Eng.

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_N)
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_N)
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_N)
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_N)
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

Requirements:

- smoothing along reflection events justified ✔
- smoothing along wavelet justified ✔
- remaining task: ensure event consistence

CRS stack provides:

- local shape of zero-offset reflection event (α, R_{N})
- approximation of projected Fresnel zone
- coherence values as measure of reliability

☞ this allows a simple and efficient smoothing algorithm

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_{N})
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

For each zero-offset sample and each CRS parameter

- align smoothing window along reflection event using emergence angle α (optionally also R_N)
- reject samples below given coherence threshold ☞ use only reliable attributes
- reject samples with dip difference beyond threshold ☞ avoid mixing of intersecting events
- apply combined filter:
  - median filter ☞ remove outliers
  - averaging ☞ remove fluctuations
- assign result to zero-offset sample

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  - ☞ current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  ☞ current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  - current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  - current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  ☞ current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  ☞ current research

Smoothing algorithm

Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  ☞ current research

Smoothing algorithm

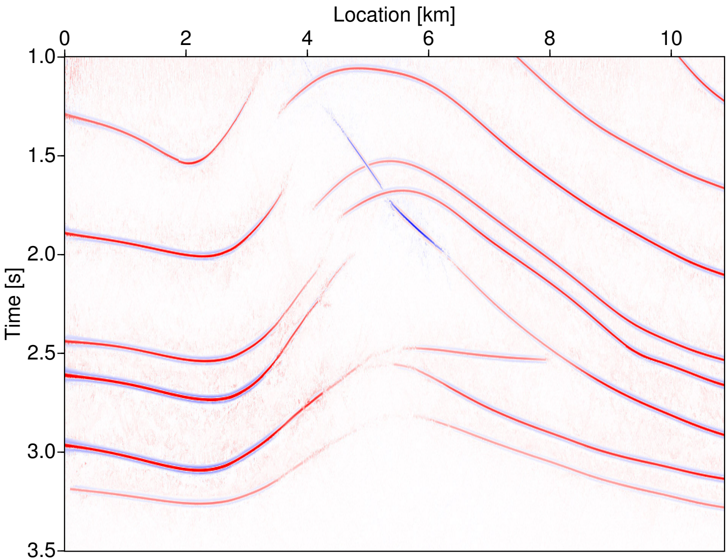
Size of smoothing window:

- as small as possible, as large as required
- temporal extension $\leq$ wavelet length
- lateral extension $\ll$ projected Fresnel zone, either fixed or a fraction of approximate Fresnel zone given by CRS parameters

Smoothing in the 3D case:

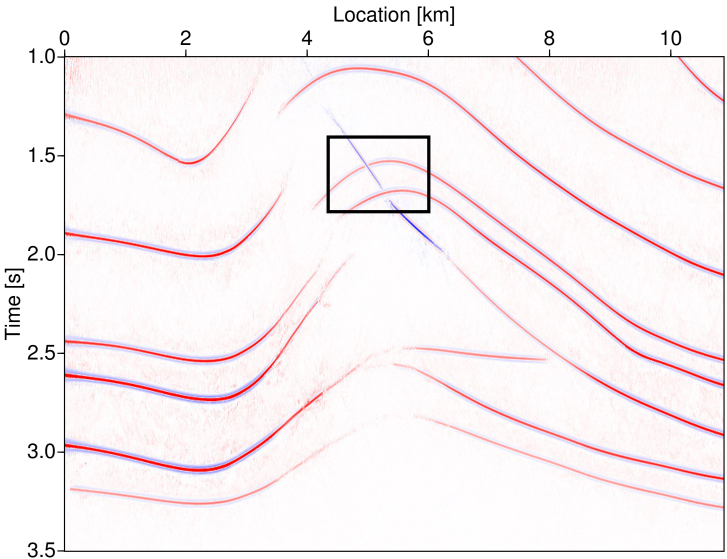
- smoothing window is a small volume
- same selection criteria as in 2D
- combined filter has to be generalized for curvature matrices and slowness vectors
  ☞ current research

Schematic example



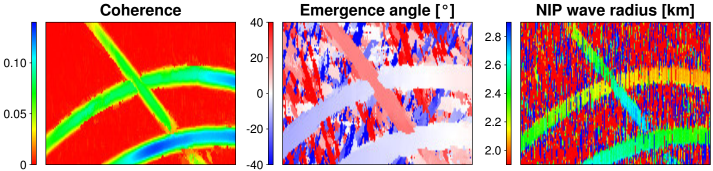
CRS stack section

Schematic example

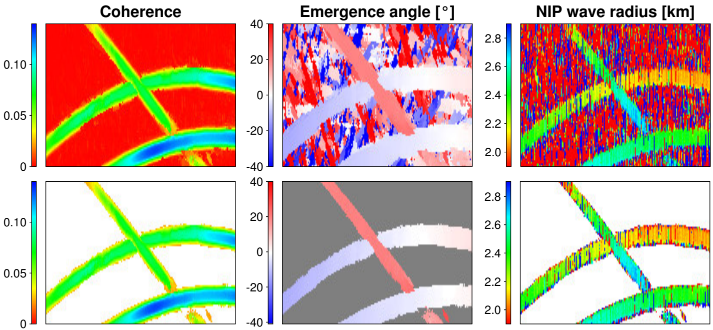


CRS stack section

Schematic example

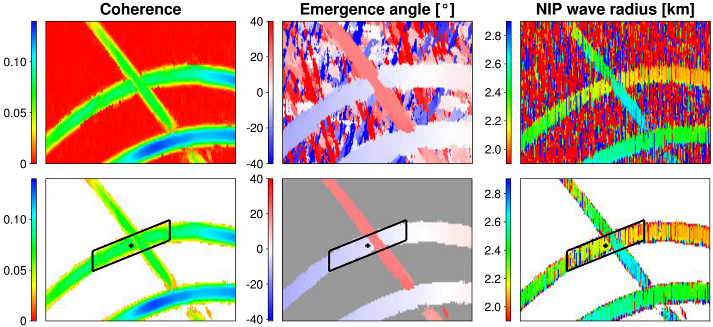


Schematic example



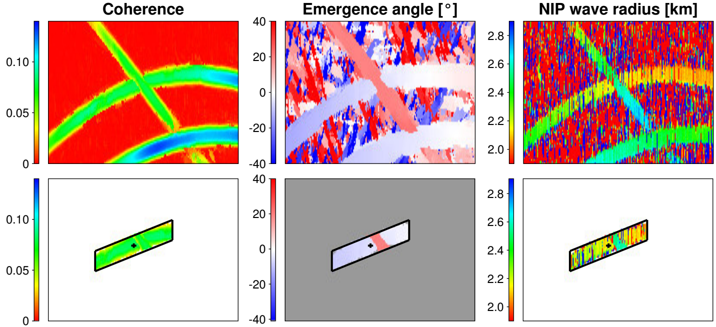
Coherence-based mask applied
(for visualization, only)

Schematic example



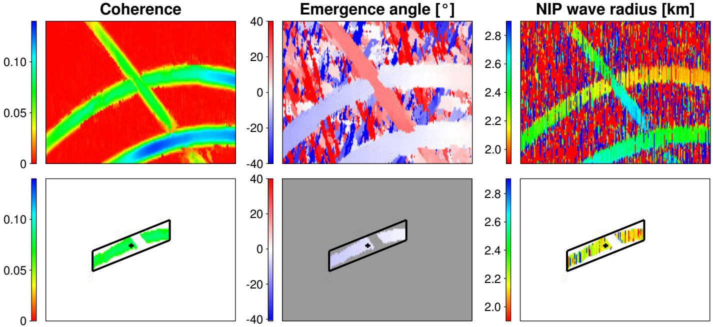
Smoothing window aligned with reflection event

Schematic example



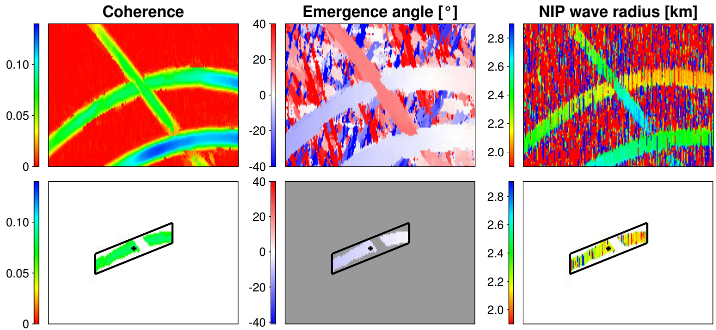
Select all samples in window

Schematic example



Apply coherence threshold and
dip difference threshold

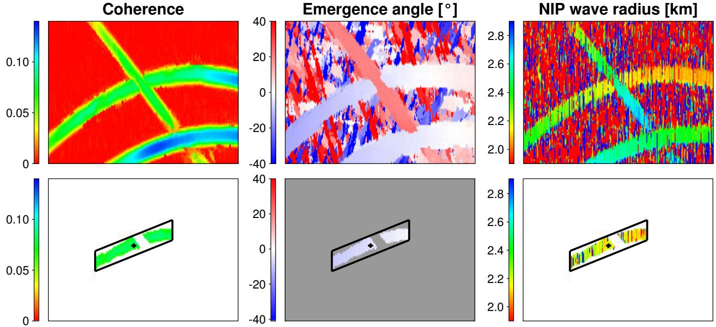
Schematic example



Smoothing:

- Sort remaining samples by magnitude

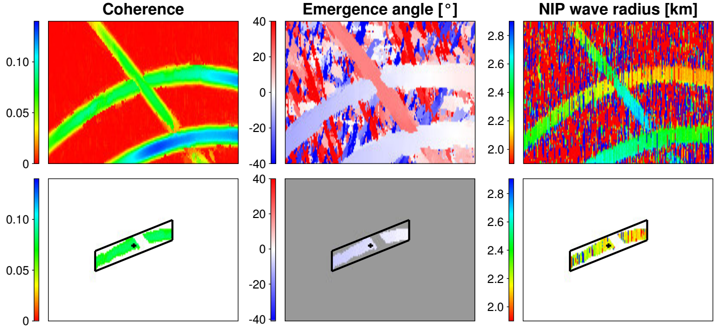
Schematic example



Smoothing:

- Sort remaining samples by magnitude
- Average given fraction of samples around median

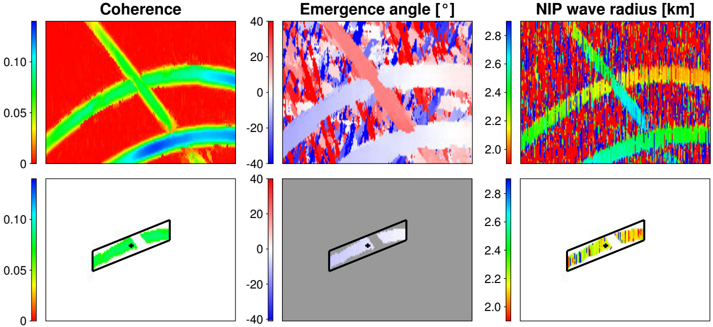
Schematic example



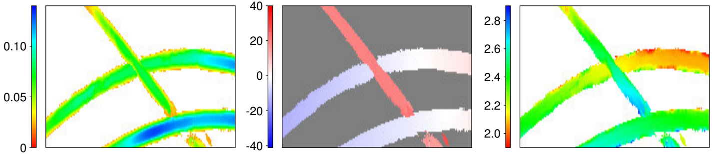
Smoothing:

- Sort remaining samples by magnitude
- Average given fraction of samples around median
- Assign result to considered ZO location

Schematic example



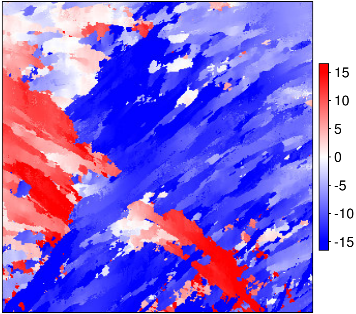
Repeated for all location ☞ smoothed attribute sections



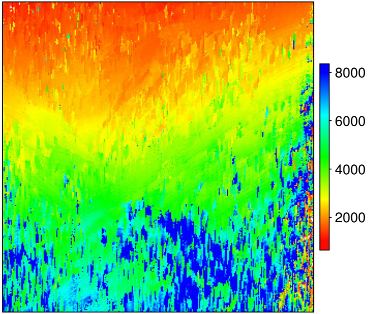
74th Annual Meeting
SEG, Denver 2004
Mann & Duveneck

Real data example

CRS parameter sections (detail)



Emergence angle [°]

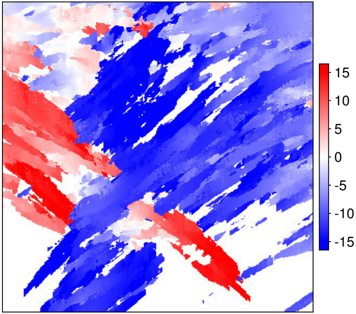


NIP wave radius [m]

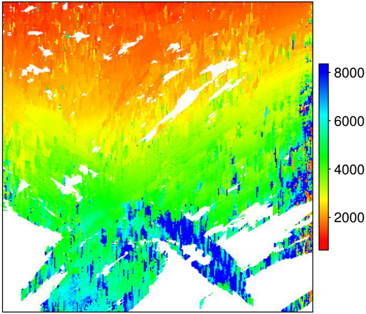
Original parameters as obtained by CRS stack
(no coherence mask applied)

Real data example

CRS parameter sections (detail)



Emergence angle [°]

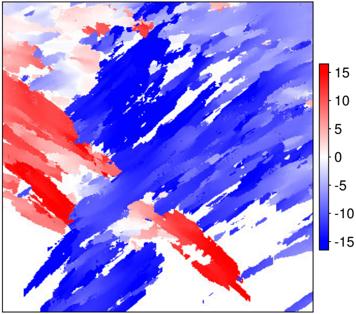


NIP wave radius [m]

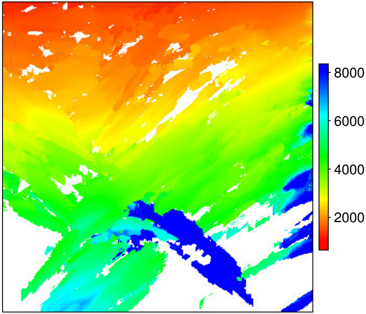
Original parameters as obtained by CRS stack
(coherence mask applied for display, only)

Real data example

CRS parameter sections (detail)



Emergence angle [°]

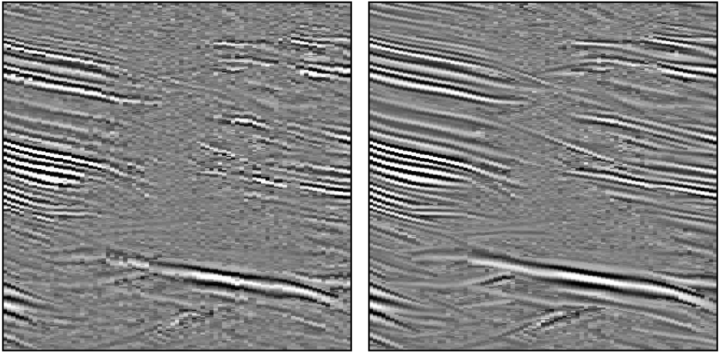


NIP wave radius [m]

Smoothed parameters
(coherence mask applied for display, only)

Real data examples

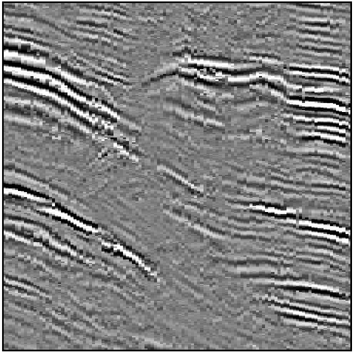
CRS stack sections (detail I)

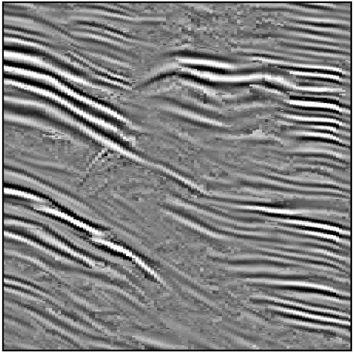


Stack with original vs. stack with smoothed parameters

Real data examples

CRS stack sections (detail II)





Stack with original vs. stack with smoothed parameters

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

➥ improved quality of stacked section

➥ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

↪ improved quality of stacked section

↪ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Conclusions

Smoothing algorithm:

- event-consistent smoothing
- based on CRS stacking parameters and coherence
- removes outliers
- removes fluctuations
- preserves kinematic properties of reflection events
- avoids mixing of intersecting events

↪ improved quality of stacked section

↪ more physical CRS stacking parameter sections for various applications like macromodel determination etc.

Acknowledgments

This work was kindly supported by the sponsors of the Wave Inversion Technology (WIT) Consortium, Karlsruhe, Germany

Related presentations

Session SP 4, Thursday morning:

SP 4.4 A seismic reflection imaging workflow based on the common-reflection-surface (CRS) stack: theoretical background and case study

SP 4.5 CRS imaging and tomography versus PreSDM: a case history in overthrust geology

SP 4.6 CRS stack and redatuming for rugged surface topography: a synthetic data example

SP 4.8 3D focusing operator estimation using sparse data