

Chapter 9 – Data before stack

- `$ cd /scratch/yourusername`
- `$ mkdir Temp5`
- `$ cp /cwpscratch/Data5/seismic.segy
Temp5`
- `$ cd Temp5`
- `$ segyread tape=seismic.segy verbose=1
| segyclean > seismic.su`

The data

- `$ ls -l seismic.su`
- `$ surange < seismic.su`
capture this into a file
called: Notes
- Look at the first 1000 traces
`$ suwind count=1000 <`
`seismic.su | suximage`
`perc=99 &`

Common Shot, Common midpoint,...??

- `$ sugethw < seismic.su sx gx offset eps
cdp`

The fields that change the
slowest will tell you the type of gather....

- `$ suwind count=12000 skip=0 | suxmovie
n2=1200 loop=1 sleep=10000
title=" &g th panel of
1200 traces" &`

Viewing the entire dataset

- `$ suwind < seismic.su count=12000
skip=12000 | suxmovie sleep=10000
loop=1 perc=99
title=" 12000 + %g x
1200 traces"`

Increase skip in increments of 12000 to view successive parts of the data.

Things to look for

- Do you have data?
- Are there data missing?
- Multiples at near offsets?
- Arrivals with different moveouts?
- Arrivals with monotonic moveout?
- Strong horizons through the data?

Viewing a specific shot gather

- `$ suwind key=ep min=200 < seismic.su
max=200 >
shot.ep=200.suor...`
- `$ suwind key=ep min=200 max=200
< seismic.su skip=10000
> shot.ep=200.su`
- `$ suxwigb < shot.ep=200.su perc=99
title=" shot 200 "`
- `$ suximage < shot.ep=200.su perc=99
title=" shot 200 "`

Source-receiver chart

- ```
$ suchart < seismic.su |
 xgraph n=120120
 linewidth=0 label1="sx"
 label2="gx"
 marksize=2 mark=8
 title=" sx gx chart " &
Plotting header values is an easy way to
see missing traces and bad header
values.
```

# Gaining the data

- \$ sugain
- \$ sugain < shot.ep=200.su tpow=1.0  
| suxwiggb perc=99

Try different values of  
tpow= with and without perc=99 set.

- \$ sugain tpow=1.5 < shot.ep=200.su |  
suxwigb  
also try
- \$ sugain tpow=1.5 | suxgraph  
to see the decay with time of  
the traces
- \$ sugain tpow=1.0 gpow=.5 | suxgraph
- \$ sugain tpow=2.0 gpow=.5 qclip=.99 |  
suxgraph (and so forth....)

# Common offset gathers

- `$ suwind < seismic.su key=offset  
min=-262 max=-262 |  
sugain jon=1 |  
suximage  
title="offset=-262" &`
- Repeat for offsets -1012 and -3237

# CMP (CDP) gathers

- ```
$ susort cdp offset  
          | sugain jon=1 >  
gain.jon=1.cdp.su
```

(use your

parameters instead of jon=1, note that the output filenames are chosen to record the processing steps applied.)

Stacking Chart

- `$ sugethw sx gx offset ep cdp
< gain.jon=1.cdp.su`
- `$ suchart < gain.jon=1.cdp.su
key1=cdp key2=offset |
xgraph n=120120
linewidth=0 mark=8
marksize=2 label1="cdp"
label2="offset" &`

Single CMP gather

- \$ suwind < gain.jon=1.cdp.su key=cdp
count=120 min=265
max=265 >
gain.jon=1.cdp=265.su
- \$ suxwigb < gain.jon=1.cdp=265.su
- \$ suxwigb < gain.jon=1.cdp=265.su
key=offset

Brute stacks

- \$ sustack
- \$ sustack < gain.jon=1.cdp.su
| suximage perc=99
title=" brute stack no
NMO" & Usually some
approximate correction for NMO is applied
before stack, even in a brute stack.

Better Brute Stacks

- `$ sunmo vnmo=1500 <`
`gain.jon=1.cdp.su | sustack`
`| suximage`
`perc=99`
`title=" brute stack vnmo=1500"`

Better Brute Stacks

- `$ sunmo vnmo=1500 <`
`gain.jon=1.cdp.su | sustack`
`| suximage`
`perc=99`
`title=" brute stack vnmo=1500"`

Water bottom multiples are accentuated.

NMO=2300

- `$ sunmo vnmo=2300 <`
`gain.jon=1.cdp.su | sustack`
`| suximage`
`perc=99`
`title=" brute stack vnmo=2300"`

Note the part of the image that appears clearer.

NMO=1500,1800,2300

- \$ sunmo vnmo=1500,1800,2300
tnmo=0.0,1.0,2.0 <
gain.jon=1.cdp.su | sustack
suximage perc=99 title=" |
brute stack vnmo=1500,1800,2300"

Even a crude guess of NMO velocities improves the brute stack image. Multiple suppression is necessary.

Chapt 10 – Velocity analysis, Semblance analysis/Noise suppression.

- `$ cd /scratch/yourusername/Temp5`
- `$ suvelan`
- `$ suvelan nv=150 fv=1450 dv=15
< gain.jon=1.cdp=265.su
| suximage d2=15
f2=1450 waterbottom and
pegleg multiples contaminate
the data.`

Radon ($\tau - \rho$) Transform

- ```
$ suplane ntr=120 nt=256 |
 sushw key=offset a=0 b=10
 |
 sfilter
 |
 f=0,5,50,60
> suplanedata.su
```
- ```
$ suximage < suplanedata.su
```

Radon ($\tau - p$) Transform

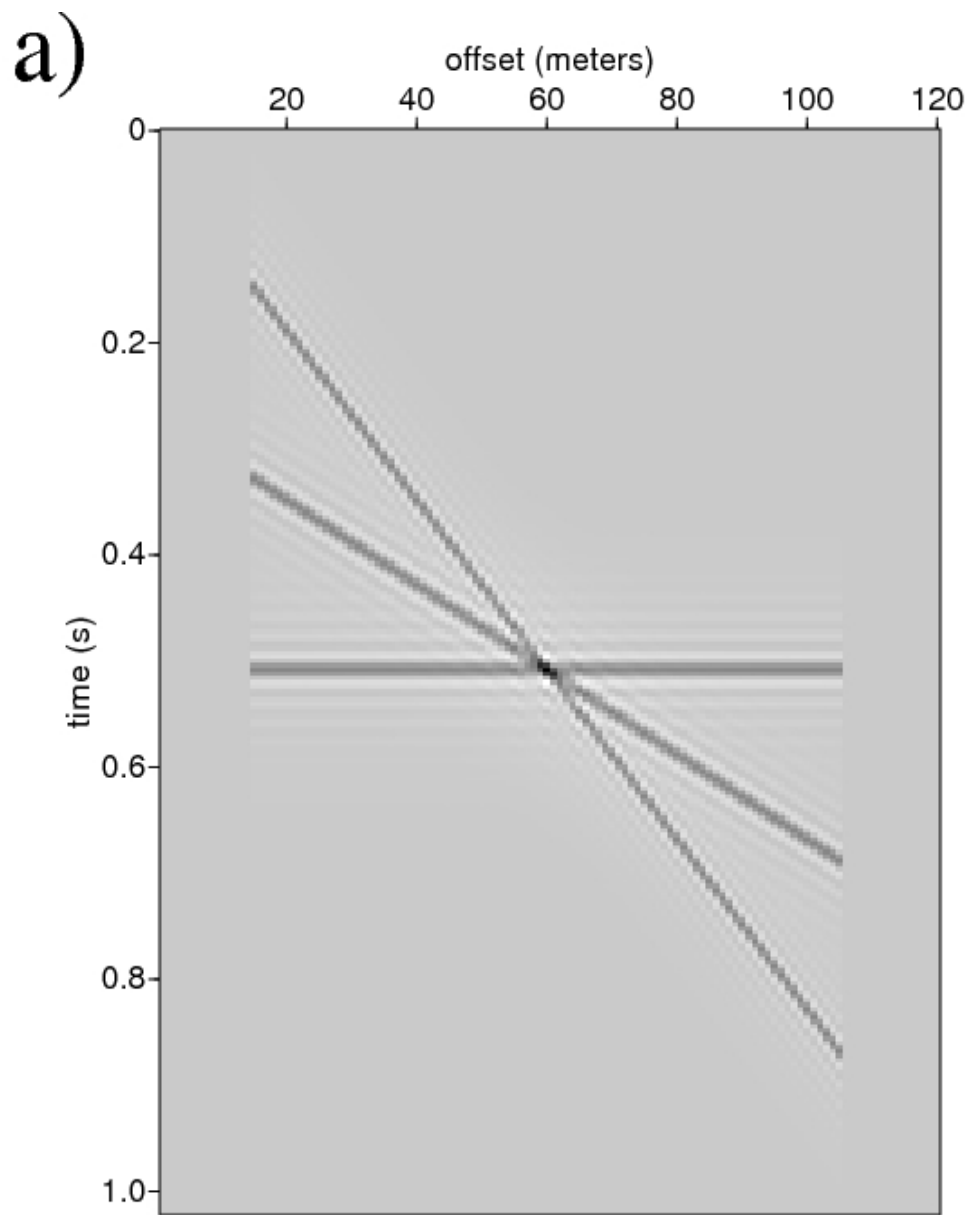
- `$ suradon`
- `$ suradon < suplanedata.su igopt=3
 interp=4 choose=0
depthref=1000 interoff=0
offref=1190
pmin=-1000 pmax=1000
 > radon.su`
- `$ suximage < radon.su perc=99
 label1="tau (s)" label2="p -
slope" &`

Meaning of the Radon Transform

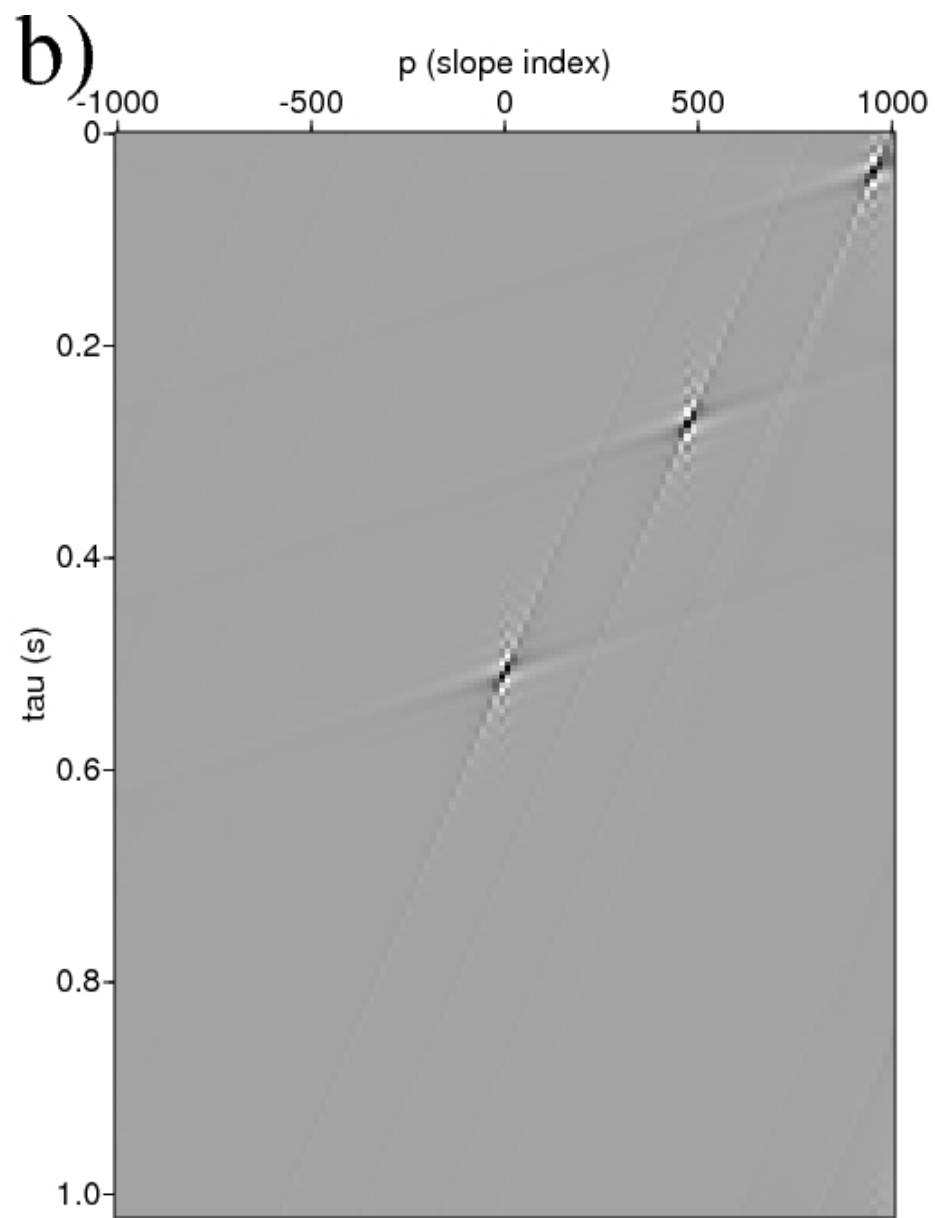
- The Radon transform takes data from the input (t, x) (time, position) domain to the output (τ, p) (time, slope) domain.
- The procedure is to sum over sloping lines and output the value of the sum at the intercept time and the given value of slope.

Filtering in the (tau, p) domain

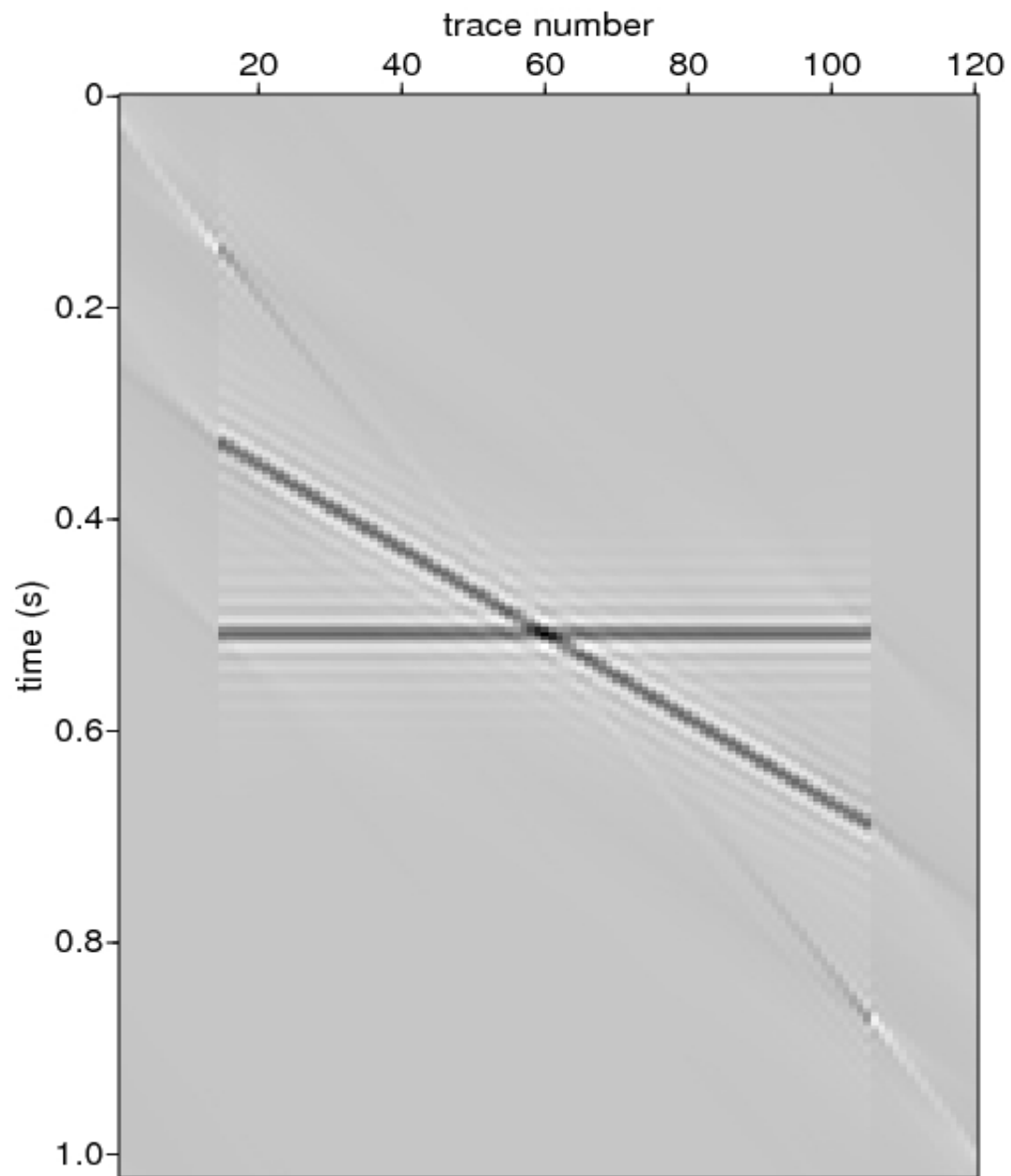
- `$ suradon < suplanedata.su igopt=3
 interp=4 choose=1
depthref=1000 pmula=600
pmulb=600 interoff=0
offref=1190 pmin=-1000
 pmax=1000 > filtered.su`
- `$ suximage < radon.su`
 Selecting “choose=1” turns suradon into a filtering program. Everything to the right of `p=600` is smoothly suppressed.



Suplane Data

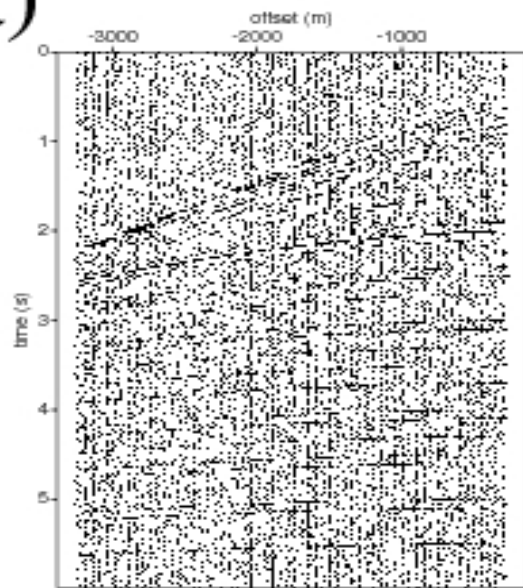


Suplane data---Radon domain



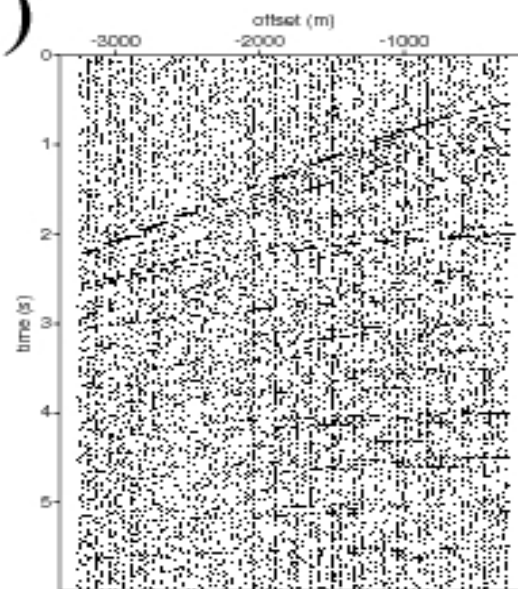
Steeply dipping plane removed

a)



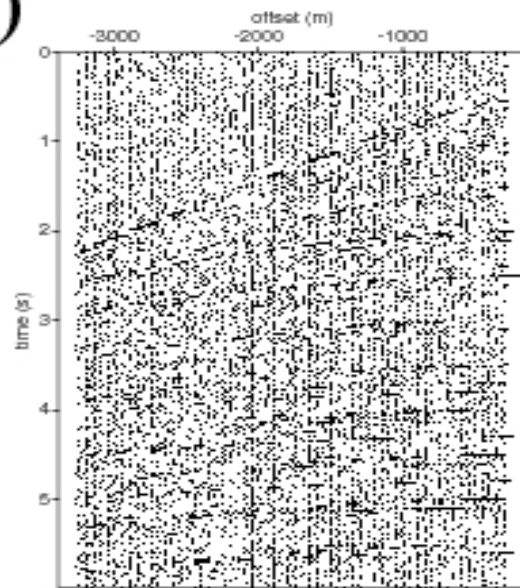
Synthetic data (no multiples)

b)

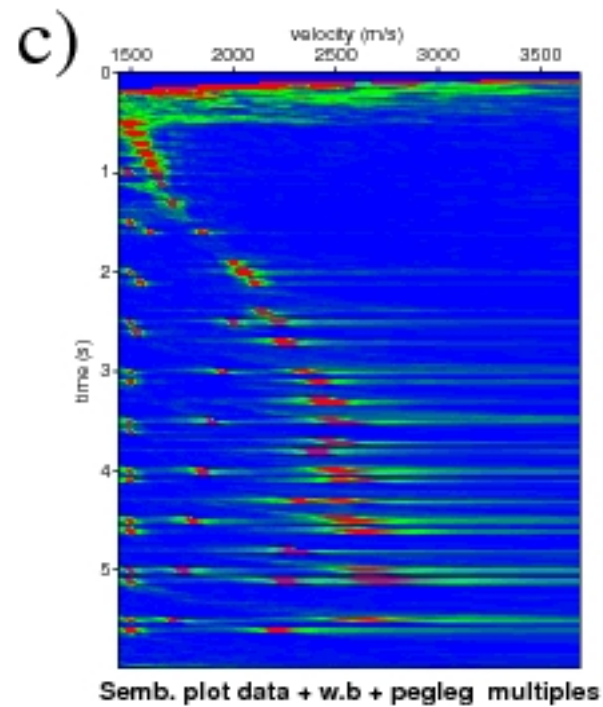
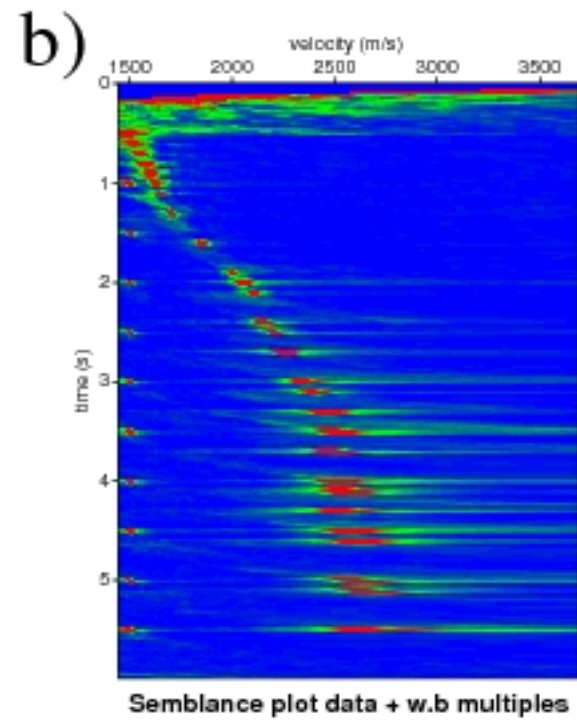
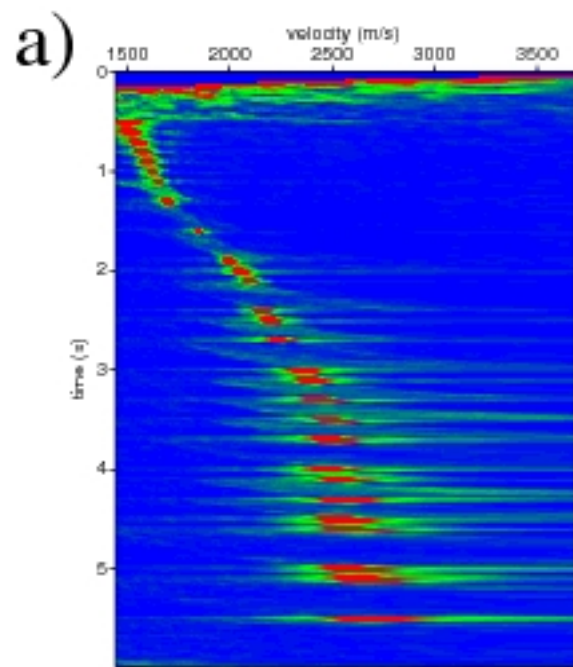


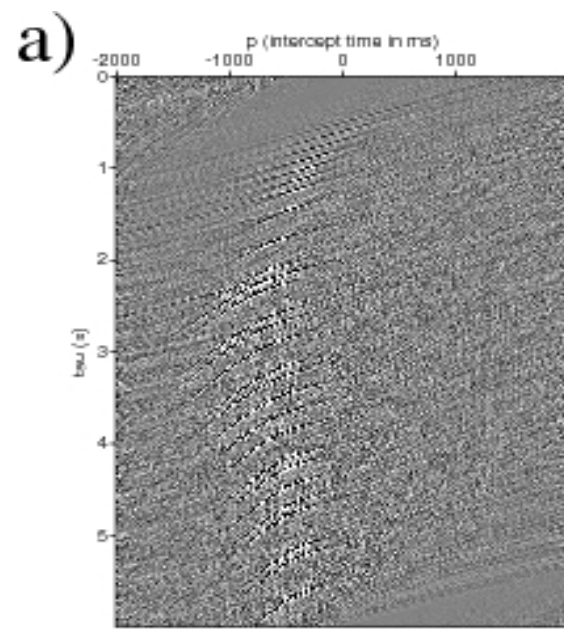
Synthetic data + w.b. multiples

c)

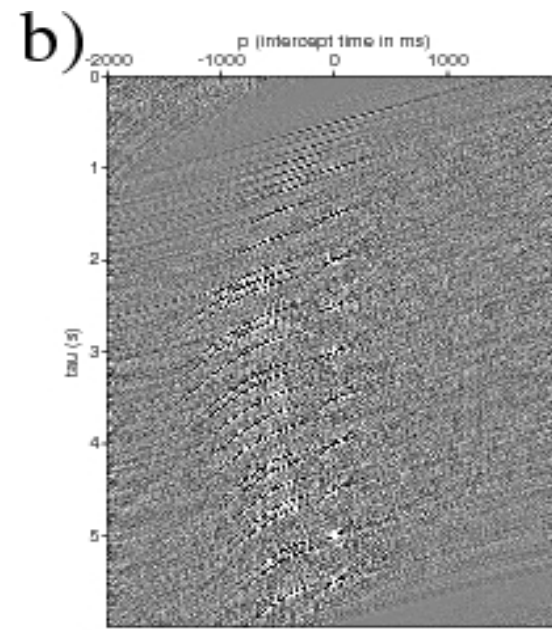


Synthetic data + w.b. + pegleg multiples

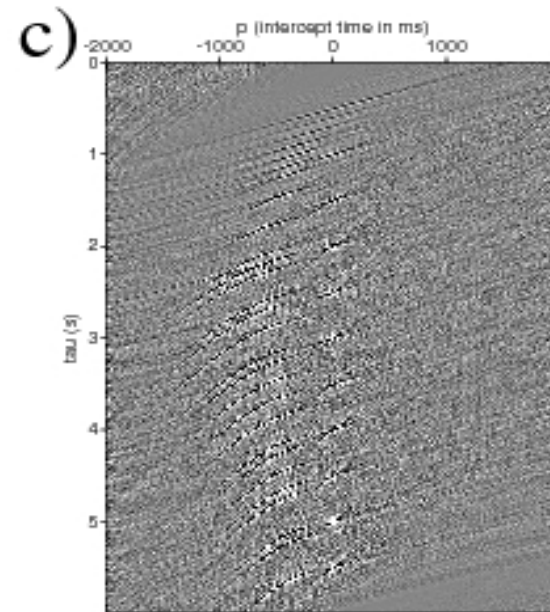




Synthetic data in the tau-p domain



Synthetic data + w.b. multiples (tau-p domain)



Synthetic data + w.b. + pegleg multiples (tau-p domain)

Spectral methods-assumptions

- Minimum phase – energy at the beginning of the waveform
- Causal – no arrival before time zero, no arrival before minimum travelttime
- White spectrum – division by zero or by small values avoided

Some spectral methods

- Simple frequency filtering
- Spectral whitening
- Spiking decon also known as “minimum phase deconvolution
- Predictive or gapped deconvolution

Frequency filtering

- `$ suspecfx < gain.jon=1.cdp=265.su
| suxgraph`
- `$ sufilter f=0,5,50,60 amps=0,1,1,0 <
gain.jon=1.cdp=265.su |
suximage perc=99
experiment with f
values and amps values (you can have
amps values > 1.`

Spectral whitening

- Divide data by shifted amplitude in frequency domain to “balance” amplitudes (phase not handled properly)
- ```
$ suftw < gain.jon=1.cdp=265.su
 w0=0 w1=1 w2=0
 sufft | suxwign
perc=99 (signal is whitened,
but so is the noise)
```

# Spiking (minimum phase) decon

- Data assumed to be minimum phase

- `$ suacor < gain.jon=1.cdp=265.su  
          ntout=101 | suxwigb  
perc=90`

(measure the width of the autocorrelation waveforms. That is the value of “maxlag” for spiking decon)

# Gapped (predictive) decon

- There is a “two for the price of one” aspect of minimum phase filtering. We can suppress repetitions in our data by determining the period of the repetition. The width of the waveform becomes “minlag” and the period of the repetition is “maxlag”. The “minlag” is the gap in gapped decon.

# More techniques

- Predictive decon in the Radon domain
- Wavelet shaping
- Removing source strength and receiver gain
- Muting
- Surface Related Multiple Elimination