

Marchenko

Jan Thorbecke, Joeri Brackenhoff,
Lele Zhang

j.w.thorbecke@tudelft.nl
j.a.brackenhoff@tudelft.nl
L.Zhang-1@tudelft.nl



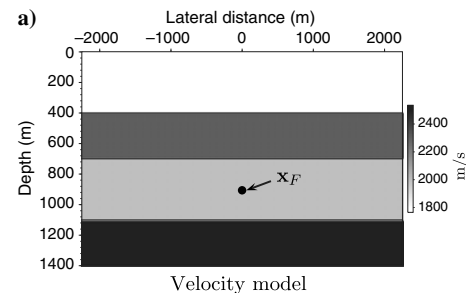
Introduction Marchenko method



Marchenko focusing functions

$$G^+(\mathbf{x}_F, \mathbf{x}_R, t) = - \int_{\partial\mathbb{D}_0} \int_{t'=-\infty}^t R(\mathbf{x}_R, \mathbf{x}, t-t') f_1^-(\mathbf{x}, \mathbf{x}_F, -t') dt' d\mathbf{x} + f_1^+(\mathbf{x}_R, \mathbf{x}_F, -t)$$

$$G^-(\mathbf{x}_F, \mathbf{x}_R, t) = \int_{\partial\mathbb{D}_0} \int_{t'=-\infty}^t R(\mathbf{x}_R, \mathbf{x}, t-t') f_1^+(\mathbf{x}, \mathbf{x}_F, t') dt' d\mathbf{x} - f_1^-(\mathbf{x}_R, \mathbf{x}_F, t)$$



Reference: Thorbecke, J., Slob, E., Brackenhoff, J., van der Neut, J., and Wapenaar, K., 2017, *Implementation of the Marchenko method*: Geophysics, Vol. 82 (6), WB29-WB45.

Marchenko focusing functions

$$\mathbf{G}^+ = -\mathbf{R}f_1^{-,*} + f_1^{+,*}$$

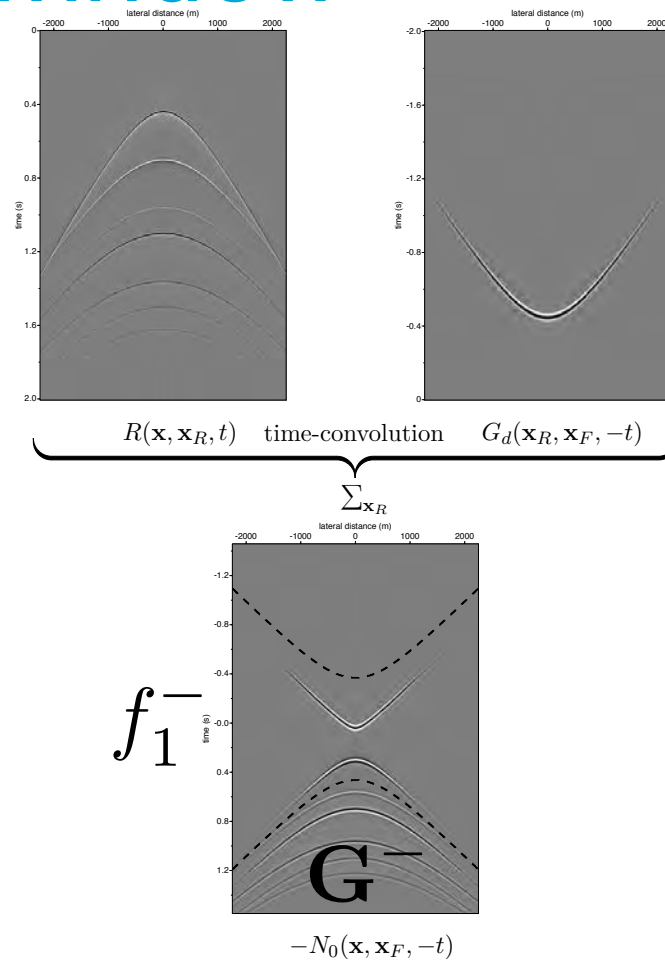
$$\mathbf{G}^- = \mathbf{R}f_1^+ - f_1^-$$

Marchenko equations

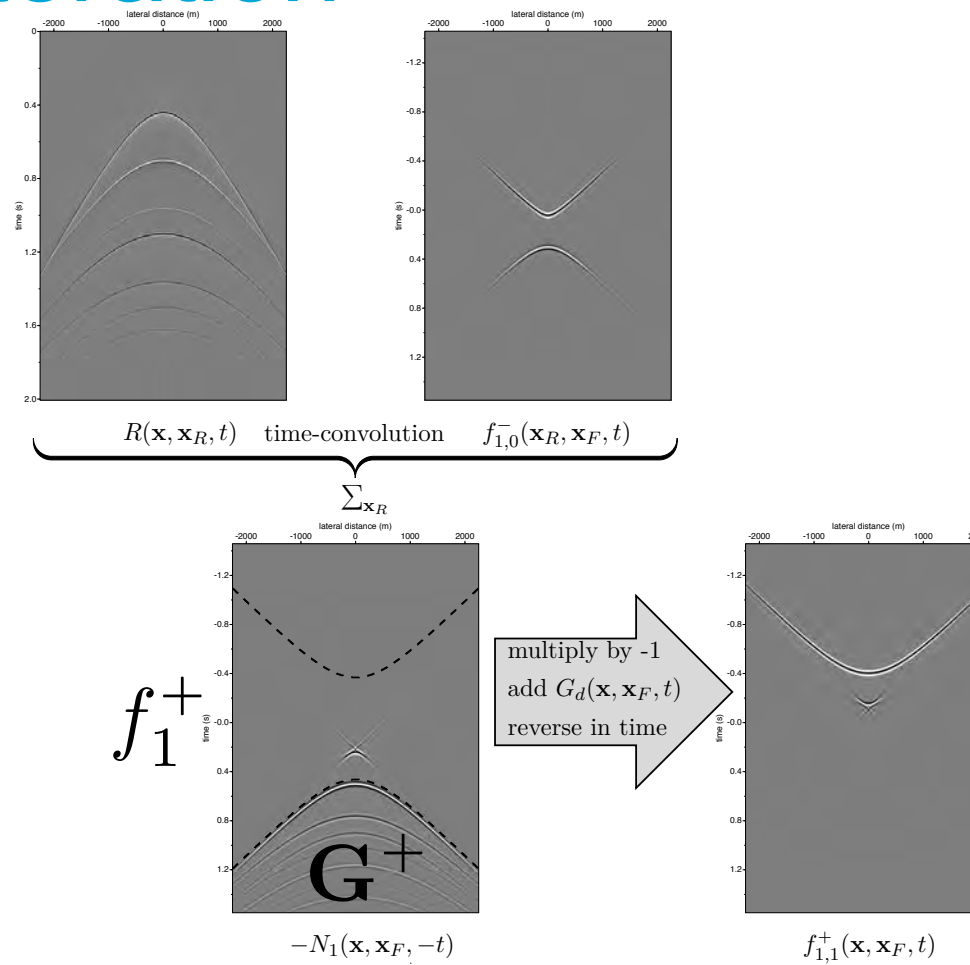
$$0 = -\theta \mathbf{R} f_1^{-,*} + f_1^{+,*}$$

$$0 = \theta \mathbf{R} f_1^{+} - f_1^{-}$$

Time window θ



First iteration



$$\Phi R N_i$$

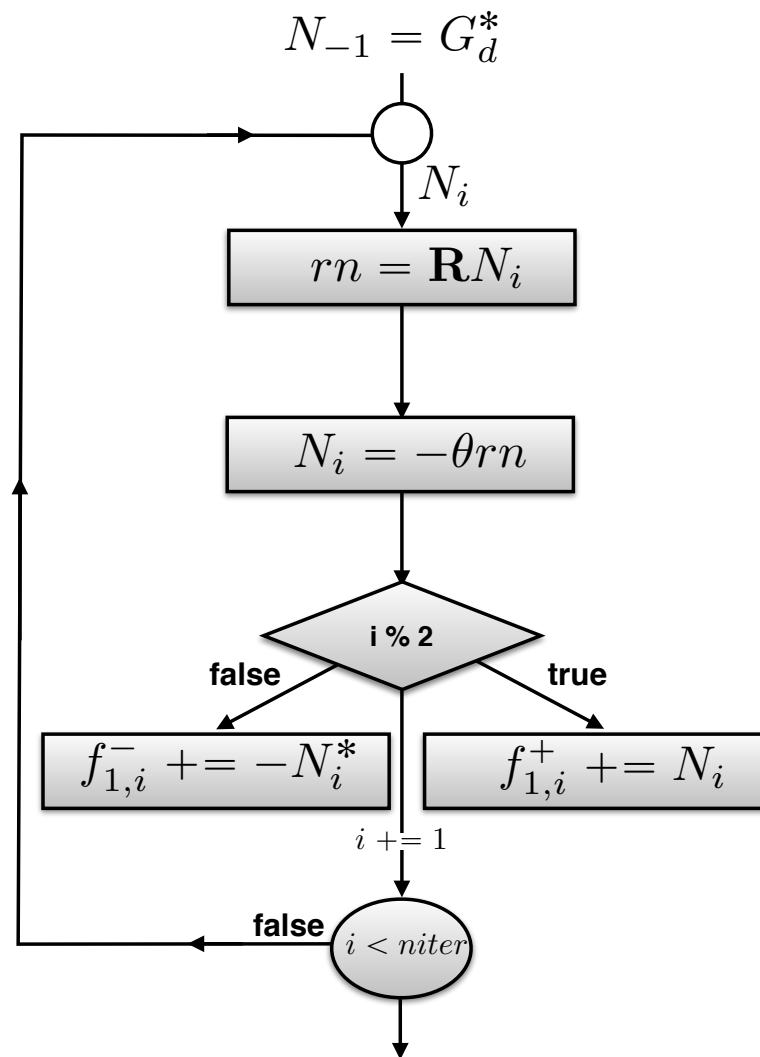
$$N_{-1}(\mathbf{x}_R, \mathbf{x}_F, -t) = G_d(\mathbf{x}, \mathbf{x}_F, -t'),$$

$$N_i(\mathbf{x}_R, \mathbf{x}_F, -t) = -\theta_t \int_{\partial \mathbb{D}_0} \int_{t'} R(\mathbf{x}_R, \mathbf{x}, t - t') N_{i-1}(\mathbf{x}, \mathbf{x}_F, t') dt' d\mathbf{x},$$

Time-domain

Time convolution

$$N_i = \Phi R N_{i-1}$$



Algorithm

Main **begin**

Reading SU-style input Data and Allocate arrays

Initialisation

$N_i(t) = f2p(t) = f1plus(t) = G_d(-t)$

$f1min(t) = pmin(t) = 0.0$

for $iter \leftarrow 0$ **to** $niter$ **do**

 synthesis(Refl, N_i , iRN) $\leftarrow R N_i$

$N_i(t) = -iRN(-t)$

$pmin(t) += iRN(t)$

 applyMute(N_i , muteW) $\leftarrow \Phi$

$f2p(t) += N_i(t)$

else if ($iter \% 2 == 0$) **then**

$f1min(t) -= N_i(-t)$

else

$f1plus(t) += N_i(t)$

end

end

$Green(t) = pmin(t) + f2p(-t)$

end



Synthesis: $R \ N_i$

```
synthesis(Refl, Ni, iRN)
begin
  iRN = 0
   $\forall l, i: \text{Fop}(l, \omega, i) = \mathcal{F} \{ \text{Ni}(l, i, t) \}$ 
  for  $k \leftarrow 0$  to  $nshots$  do
    #pragma omp parallel for
    for  $l \leftarrow 0$  to  $Nfoc$  do
      for  $\omega \leftarrow \omega_{min}$  to  $\omega_{max}$  do
        sum( $\omega$ ) = 0
        for  $i \leftarrow 0$  to  $nrecv$  do
          sum( $\omega$ ) += Refl( $k, \omega, i$ ) * Fop( $l, \omega, i$ )
        end
      end
      iRN( $l, k, t$ ) =  $\mathcal{F}^{-1} \{ \text{sum}(\omega) \}$ 
    end
  end
end
```



Synthesis: R N_i

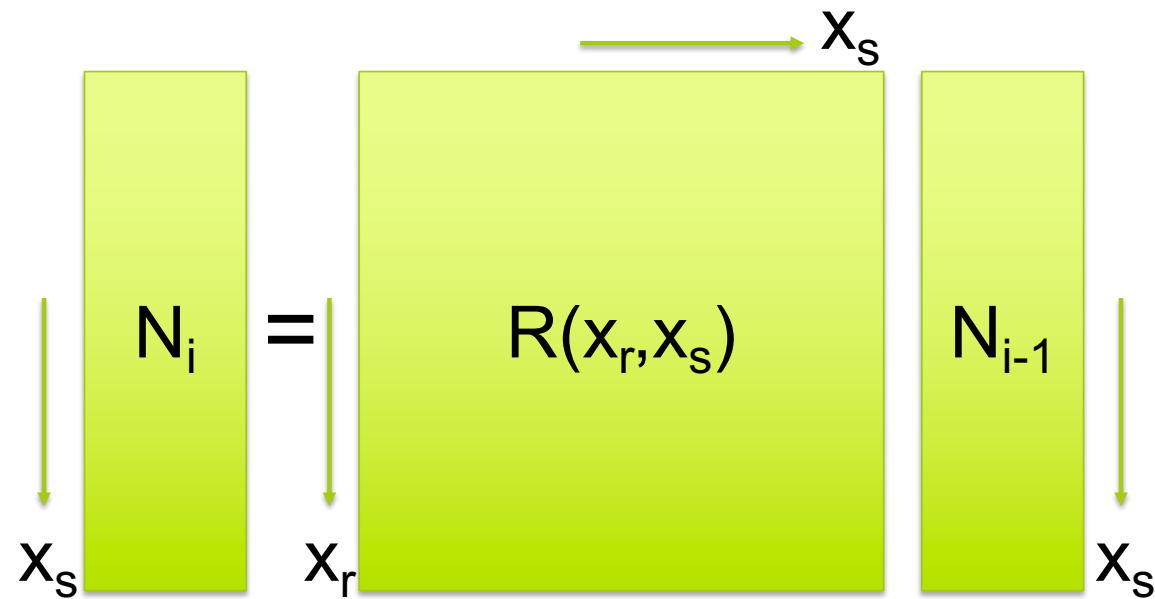
```
synthesis(Refl, Ni, iRN)
begin
  iRN = 0
   $\forall l, i: \text{Fop}(l, \omega, i) = \mathcal{F} \{ \text{Ni}(l, i, t) \}$ 
  for  $k \leftarrow 0$  to  $nshots$  do
    #pragma omp parallel for
    for  $l \leftarrow 0$  to  $Nfoc$  do
      for  $\omega \leftarrow \omega_{min}$  to  $\omega_{max}$  do
        sum( $\omega$ ) = 0
        for  $i \leftarrow 0$  to  $nrecv$  do
          sum( $\omega$ ) += Refl( $k, \omega, i$ ) * Fop( $l, \omega, i$ )
        end
      end
      iRN( $l, k, t$ ) =  $\mathcal{F}^{-1} \{ \text{sum}(\omega) \}$ 
    end
  end
end
```

FFT's

Synthesis: $R N_i$

$$N_i(x_s, t) \Rightarrow N_i(x_s, \omega)$$

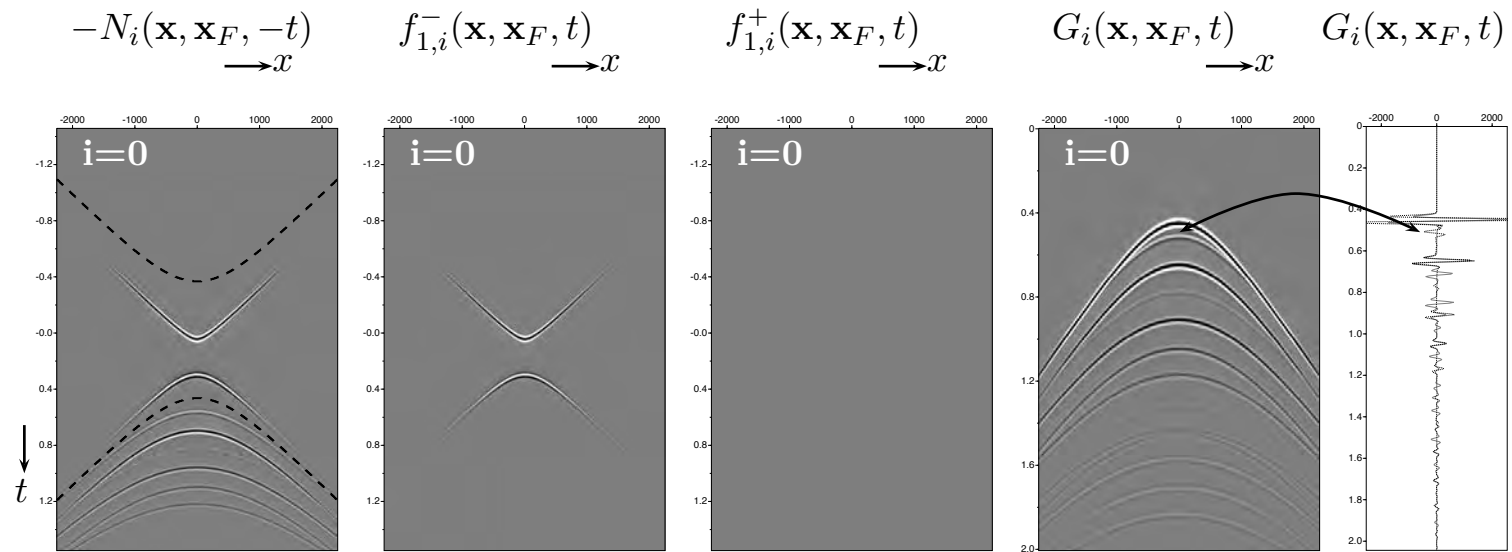
$$R(x_r, x_s, t) \Rightarrow R(x_r, x_s, \omega)$$

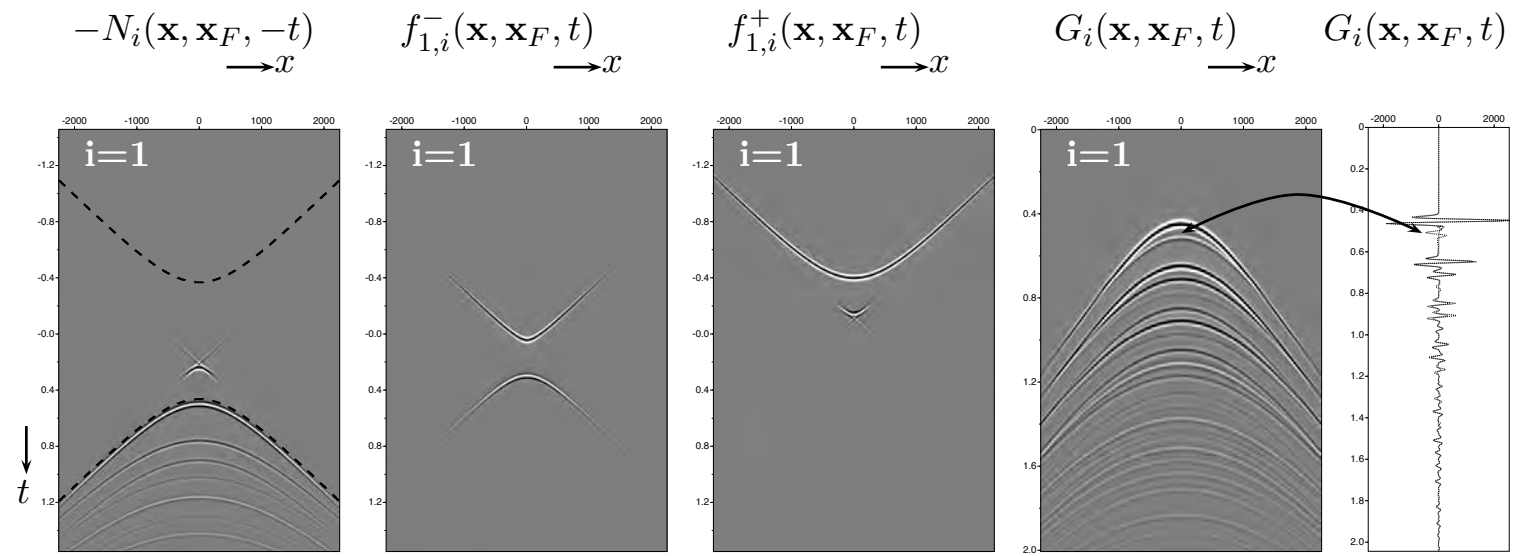


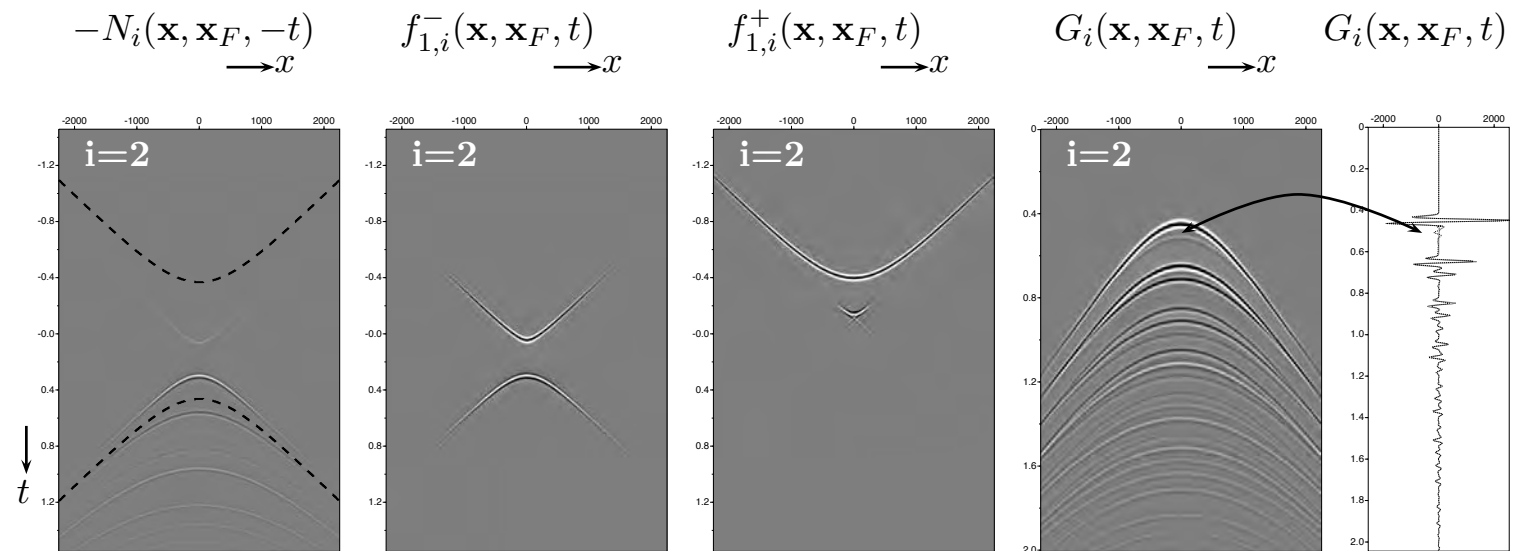
Implementation

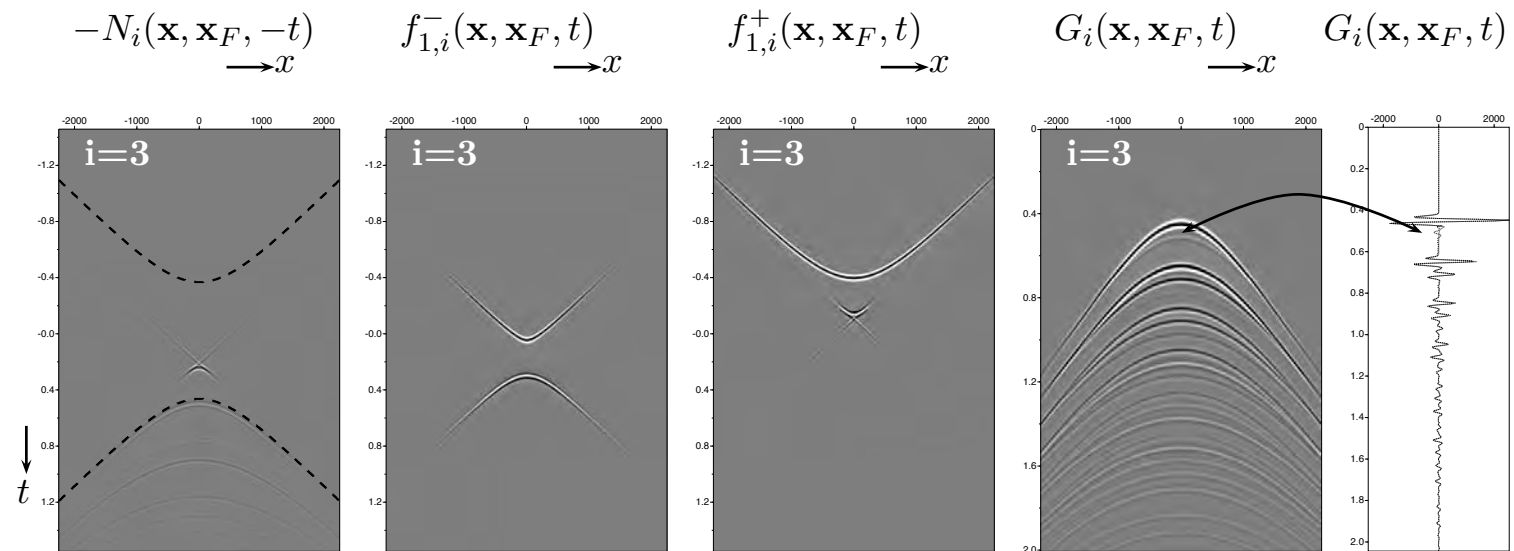
```
#pragma omp for schedule(guided,1)
for (k=0; k<nshots; k++) {
    ixsrc = NINT((xsrc[k] - fxsb)/dxs);
    for (l = 0; l < Nfoc; l++) {
        memset(&sum[0].r,0,nfreq*2*sizeof(float));
        for (j = nw_low, m = 0; j <= nw_high; j++, m++) {
            for (i = 0; i < xnx[k]; i++) {
                ix = ixrcv[k*nx+i];
                sum[j].r += Refl[k*nw*nx+m*nx+i].r*Fop[l*nw*nxs+m*nxs+ix].r -
                           Refl[k*nw*nx+m*nx+i].i*Fop[l*nw*nxs+m*nxs+ix].i;
                sum[j].i += Refl[k*nw*nx+m*nx+i].i*Fop[l*nw*nxs+m*nxs+ix].r +
                           Refl[k*nw*nx+m*nx+i].r*Fop[l*nw*nxs+m*nxs+ix].i;
            }
        }
        /* transform result back to time domain */
        cr1fft(sum, rtrace, ntfft, 1);
        /* place result at source position ixsrc; dx = receiver distance */
        for (j = 0; j < nts; j++)
            iRN[l*size+ixsrc*nts+j] += rtrace[j]*scl*dx;
    } /* end of Nfoc loop */
} /* end of parallel nshots (k) loop */
```











access to compute system

- Read MarketingPartnerNetwork agreement
 - citizens from Cuba, Iran, North Korea, Sudan, or Syria contact me.
- Fill in and sign: Training Class User Record
- You will get user name and passwd to access swan:
- ssh -Y trxx@swan.cray.com
xx: trainee number

accounts are closed, data removed, on **June 15**



swan.cray.com

- Cray XC50
- Shared resource with other users
- 120 BDW 22-core 2.2 GHz nodes
- 64 SKL 28-core 2.1 GHz nodes
- 64 SKL 20-core 2.4 GHz nodes
- node has 2 processors (40, 44 or 56 cores)
- workload manager is PBSpro



Display of results

```
suximage < result.su perc=99 &
```

or

```
supsimage < result.su perc=99 > result.eps
```

```
convert result.eps ~/result.png
```

Then copy files to your local machine

```
scp trxx@swan.cray.com:~/\*png .
```



Environment of swan

- set up environment,
create working directory,
/lus/scratch/\$USER
copy source code and data

```
source ~jan/WS15setup.sh
```



Installation Marchenko code

Latest version can be found on

git clone <https://github.com/JanThorbecke/OpenSource.git>

for this workshop copy has been prepared

```
rsync -av /lus/scratch/jan/OpenSource /lus/scratch/$USER/
```

Continue with step 2 in

OpenSource/marchenko/demo/WS15/README.1

and (re)compile the code.



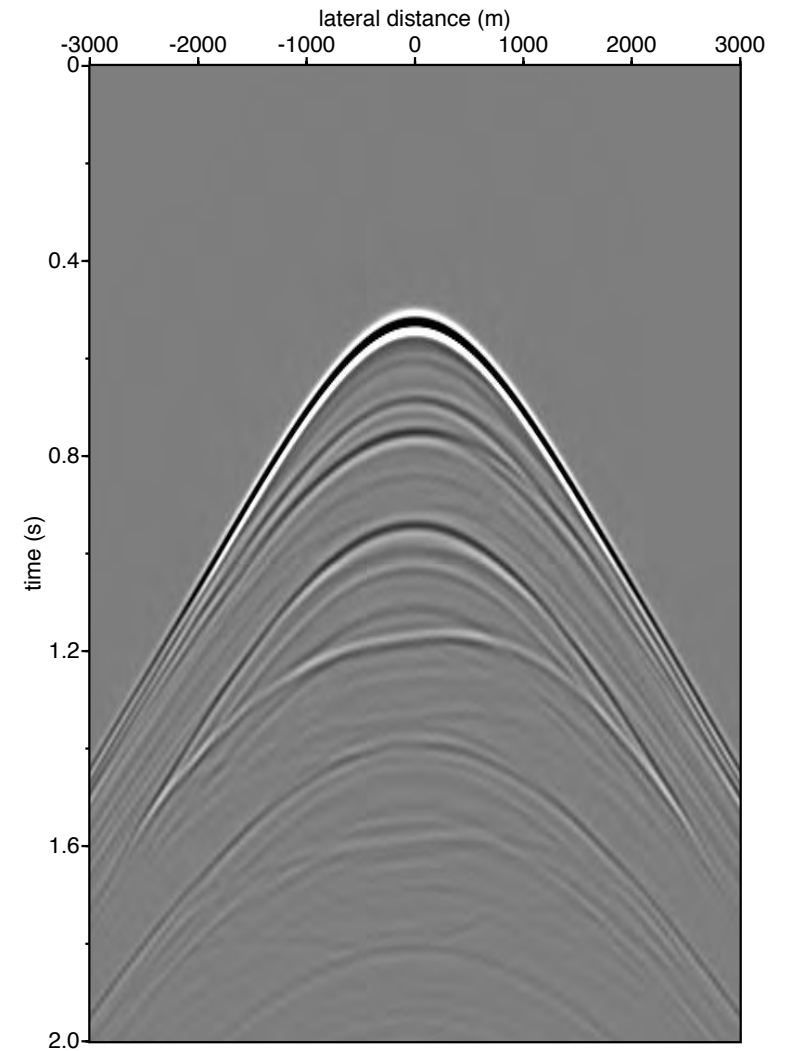
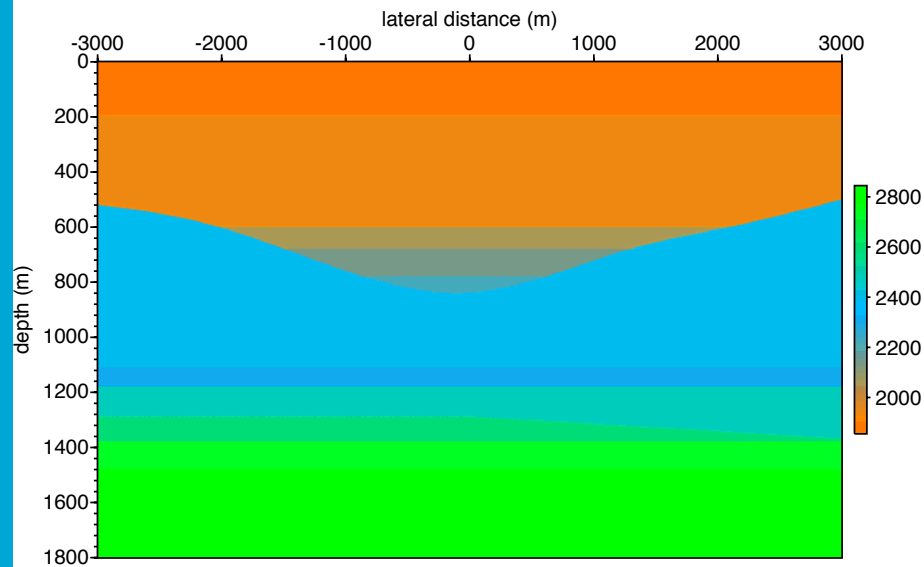
Working on demo/twoD

- demo/WS15/README.2
- Reproduce 1D examples from paper on 2D model
- Experiment with
 - Number of iterations.
 - Change amplitude of R by 2 or 0.5.
 - What wavelet is used to model R?
 - Investigate intermediate results.

45 min.

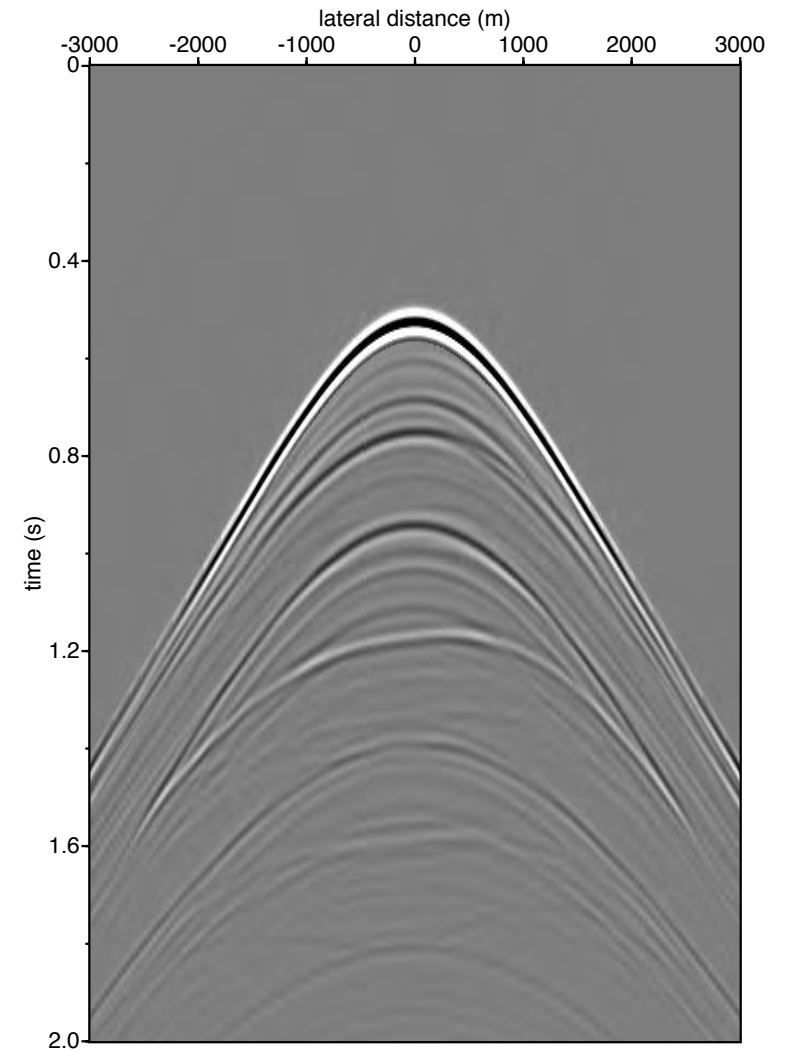
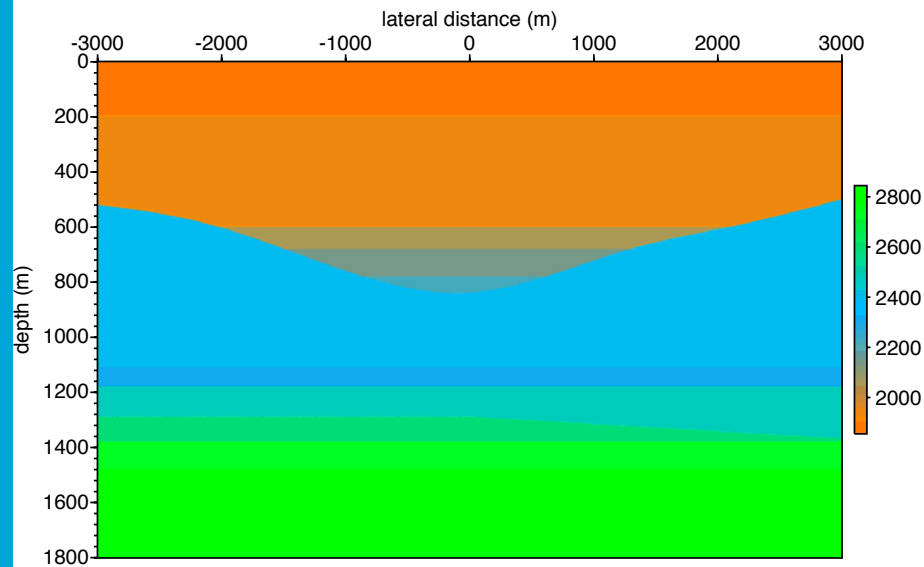


twoD figures



Reference

twoD figures



Marchenko

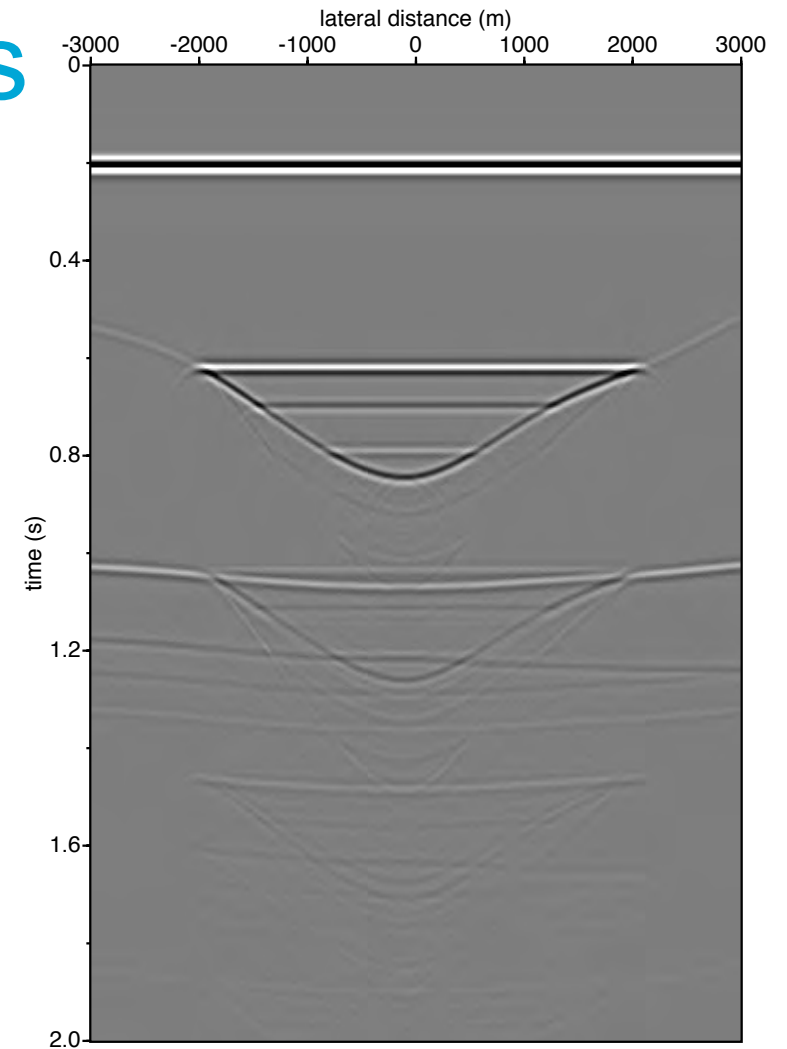
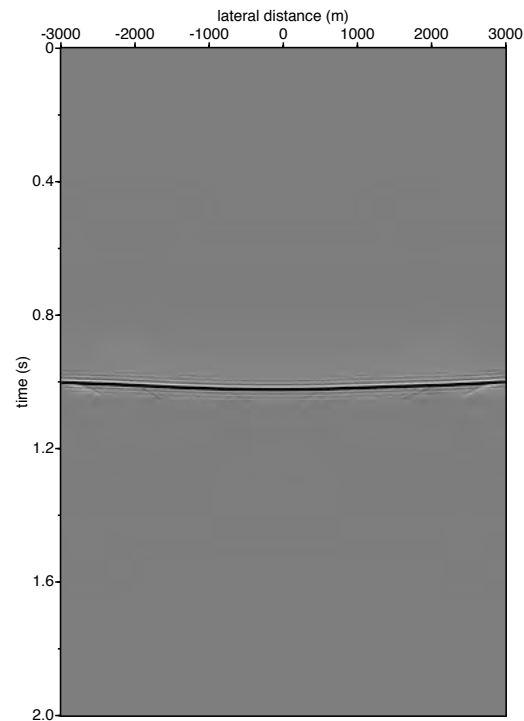
Use of plane-waves in Marchenko

- README.3
- Marchenko plane-wave method
- Based on R-data in demo/twoD
- Based on work of Giovanni Meles **15 min.**

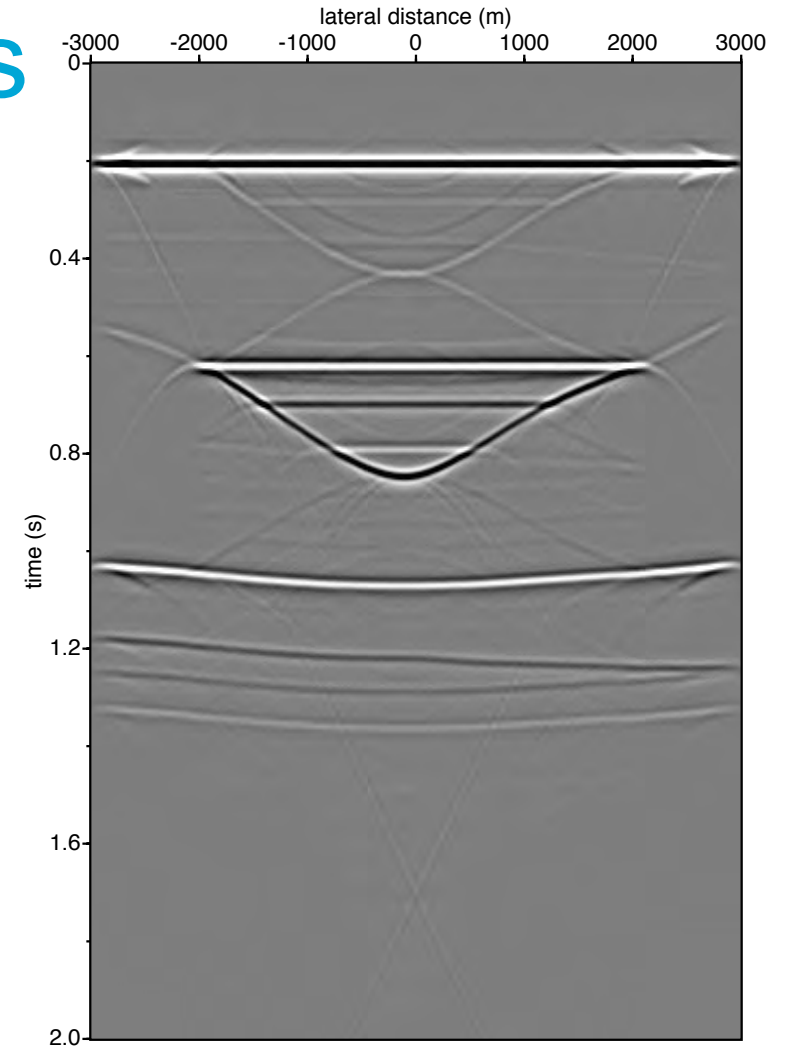
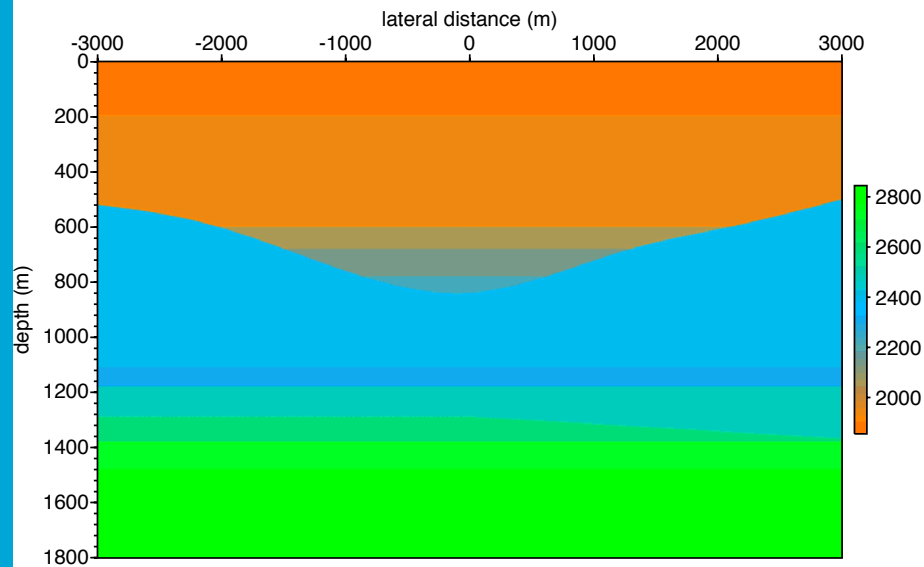
Reference: Meles, G.A., Wapenaar, K., and Thorbecke, J., 2018, *Virtual plane-wave imaging via Marchenko redatuming*: Geoph. J. Int., Vol. 214, 508-519.



Plane-wave results



Plane-wave results



convPlanew

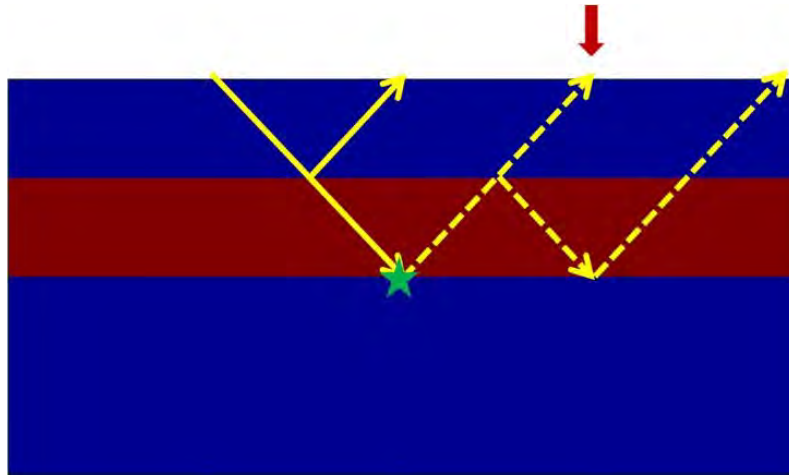
136

Extracting primaries with Marchenko

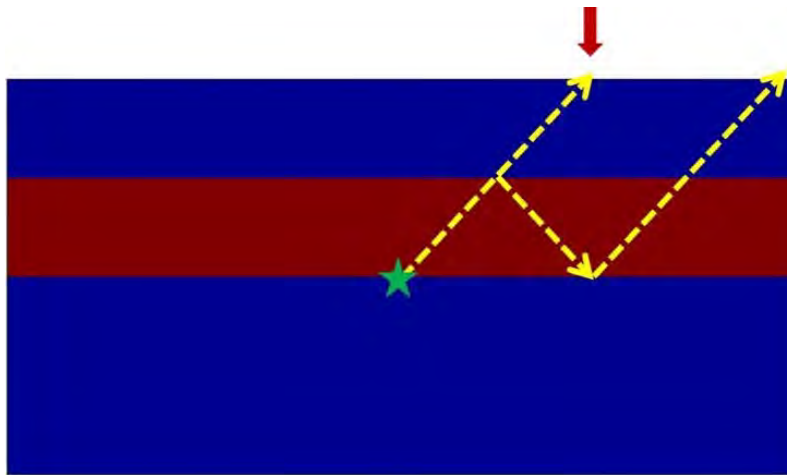


Reference: Zhang, L., Thorbecke, J., Wapenaar, K., and Slob, E., 2019, *Transmission compensated primary reflection retrieval in the data domain and consequences for imaging*: Geophysics, Vol. 84 (4), Q27-Q36.

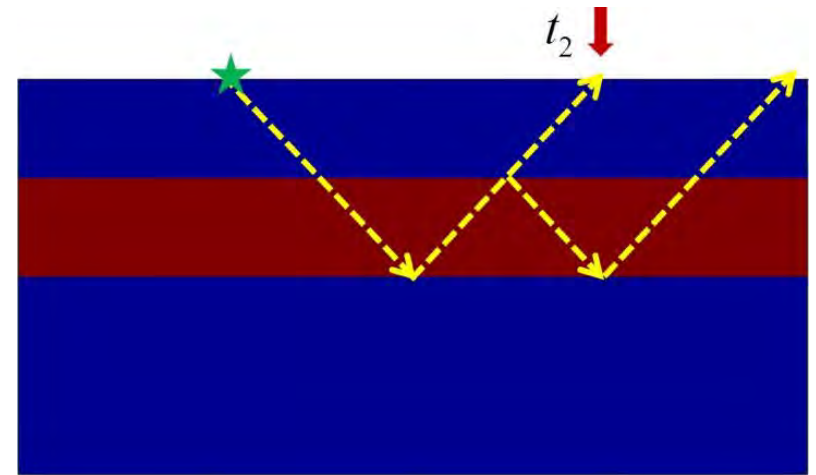
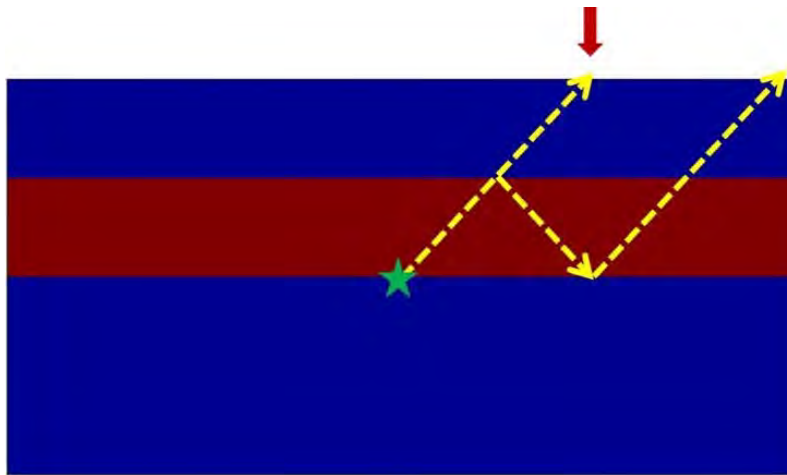
Marchenko Multiple Elimination (MME)



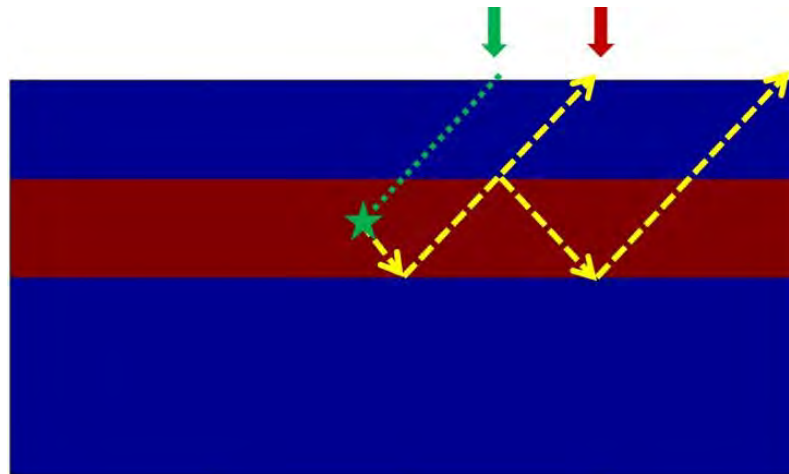
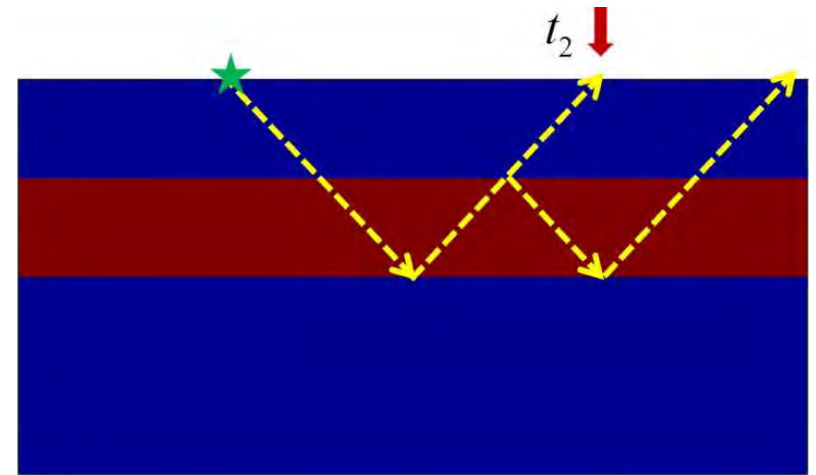
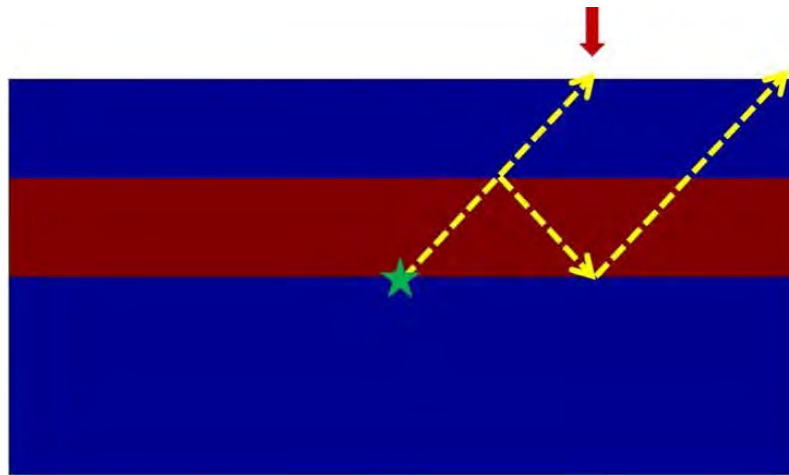
Marchenko Multiple Elimination (MME)



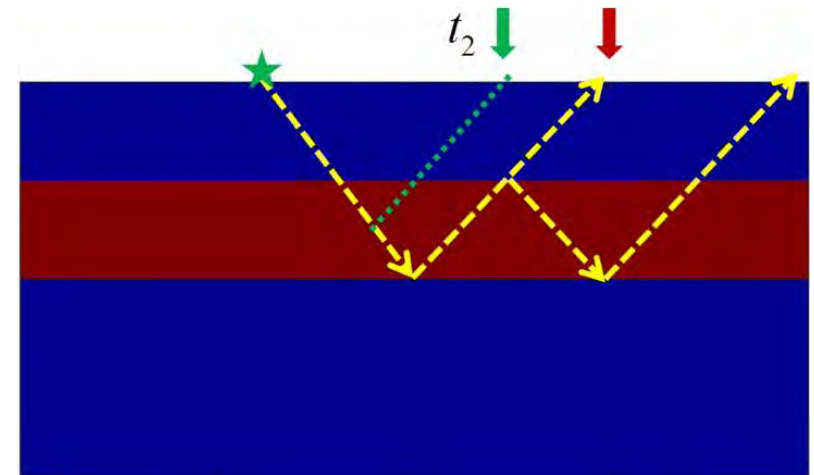
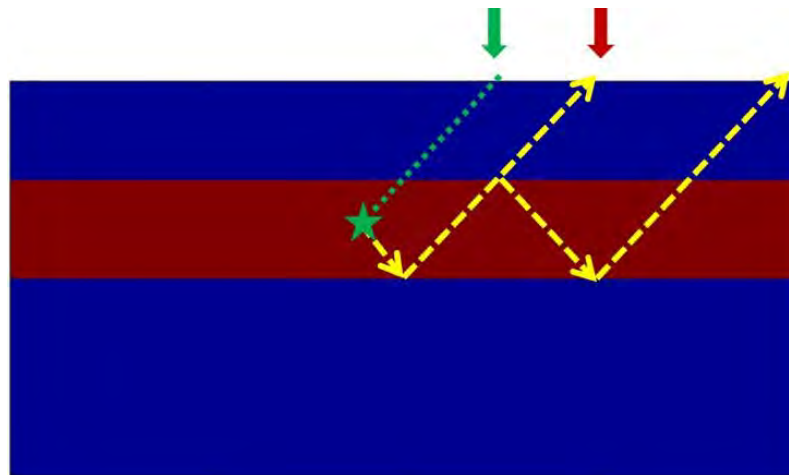
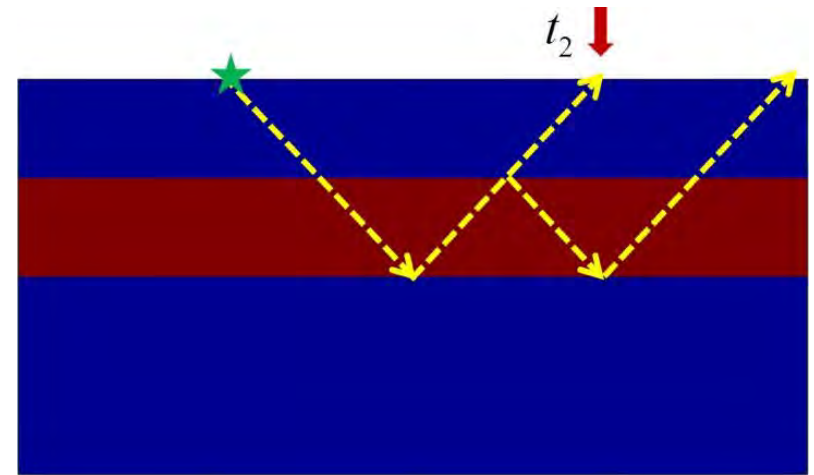
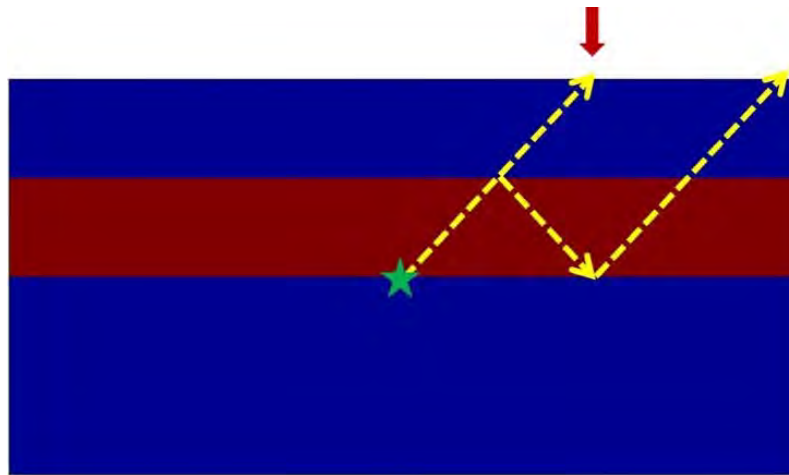
Marchenko Multiple Elimination (MME)

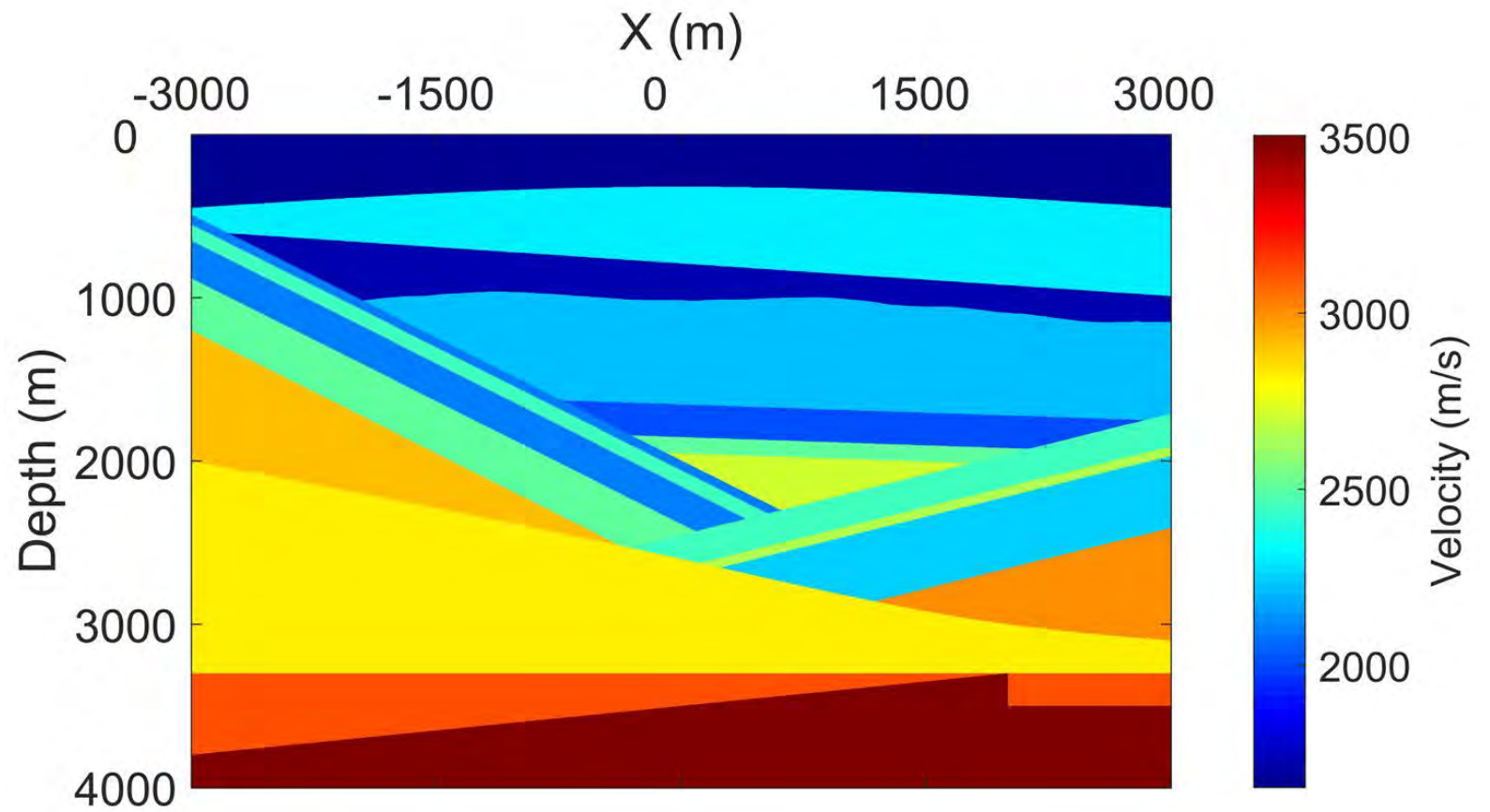


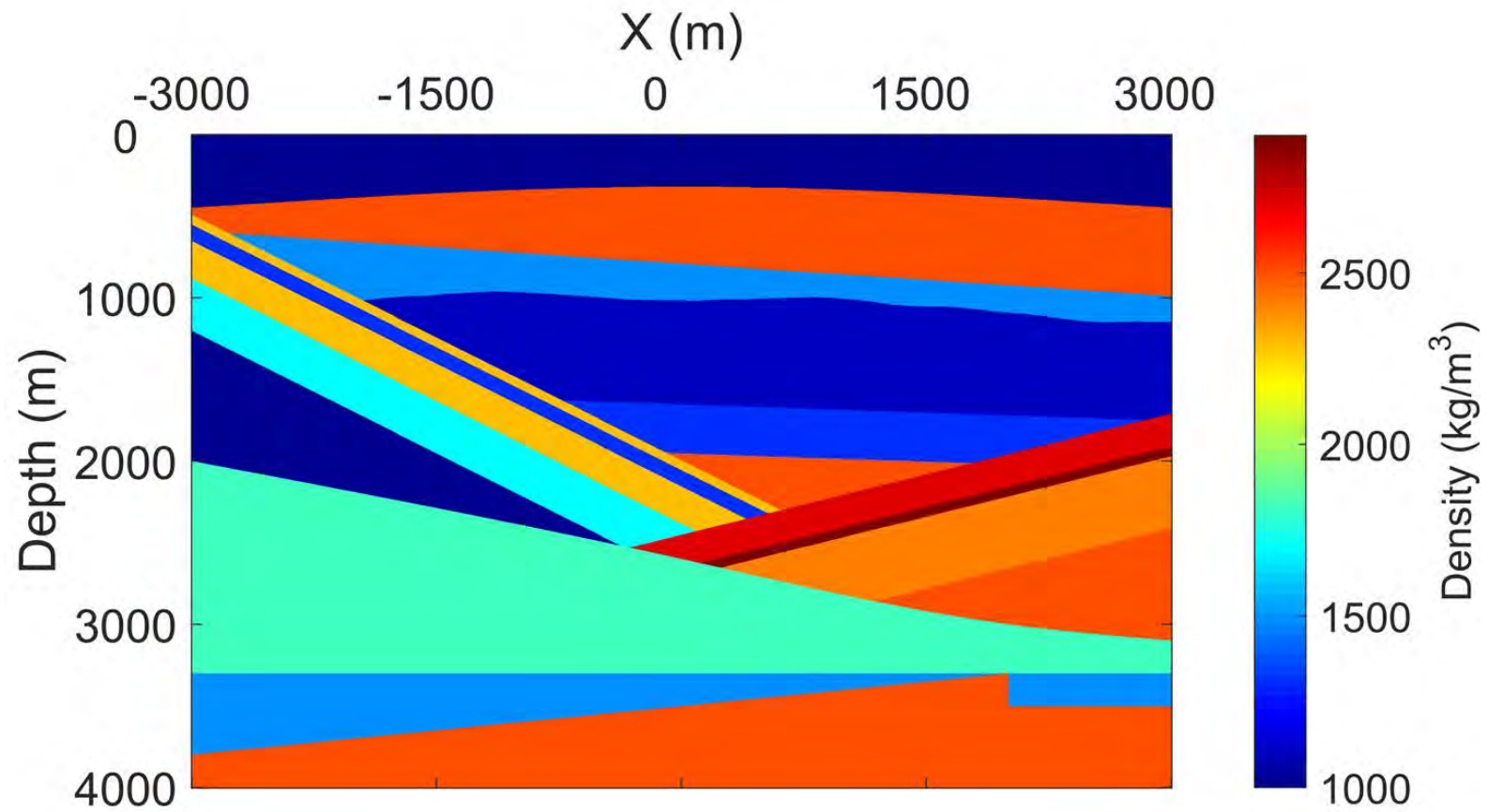
Marchenko Multiple Elimination (MME)

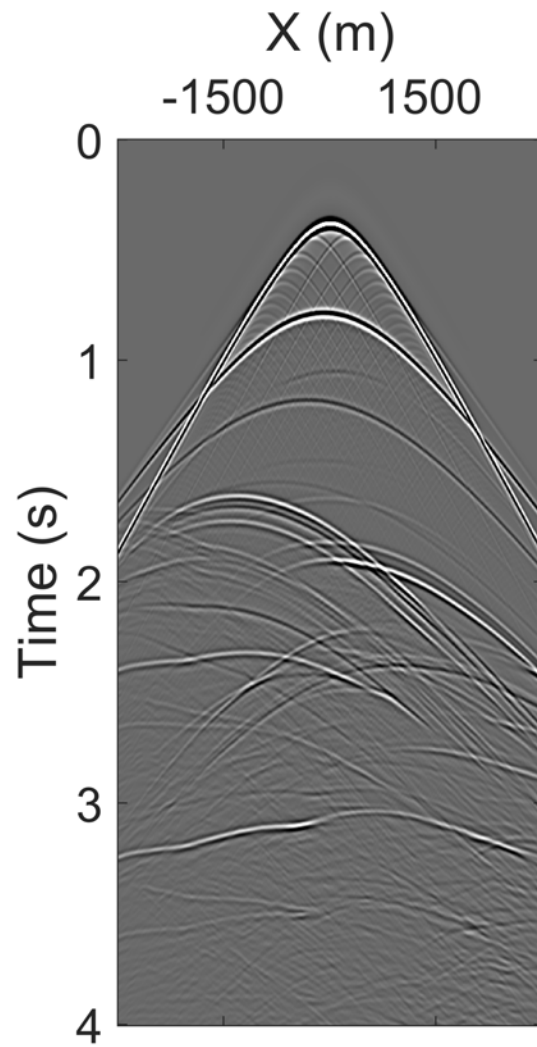


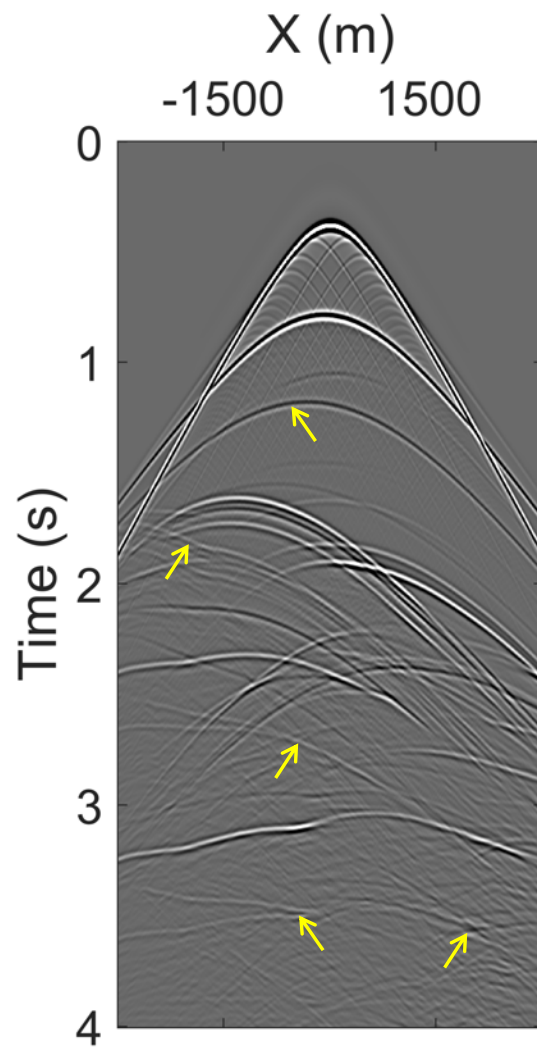
Marchenko Multiple Elimination (MME)

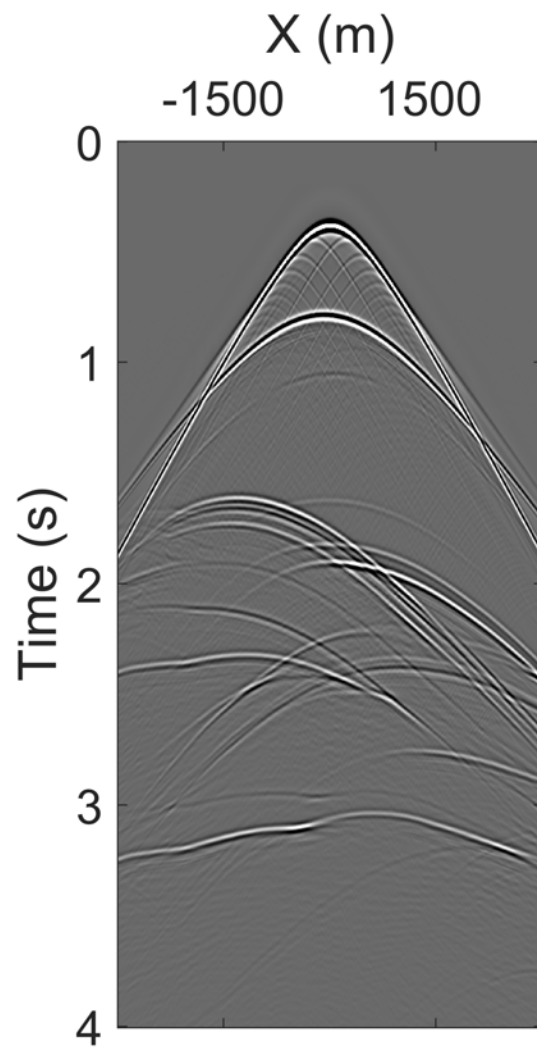


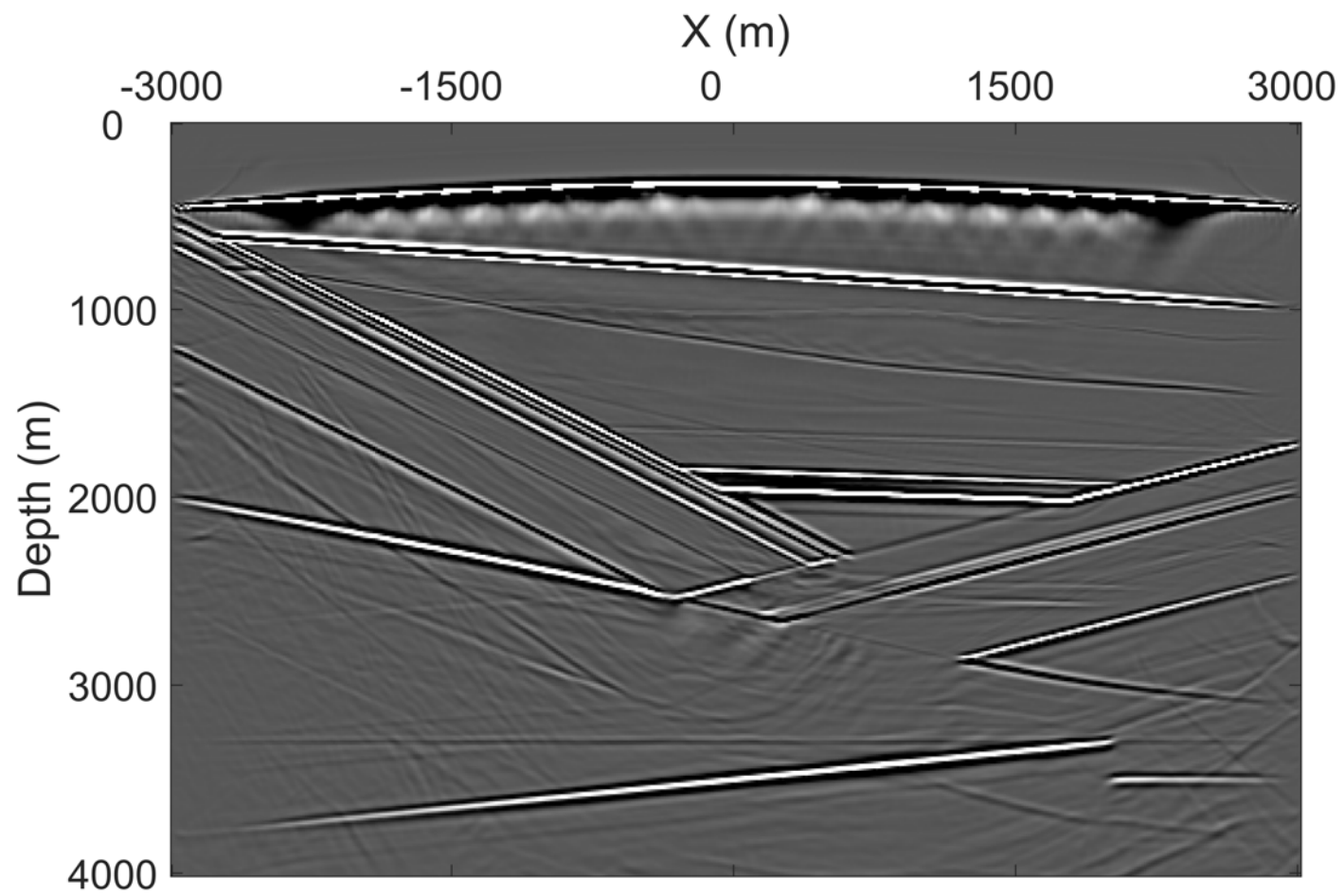


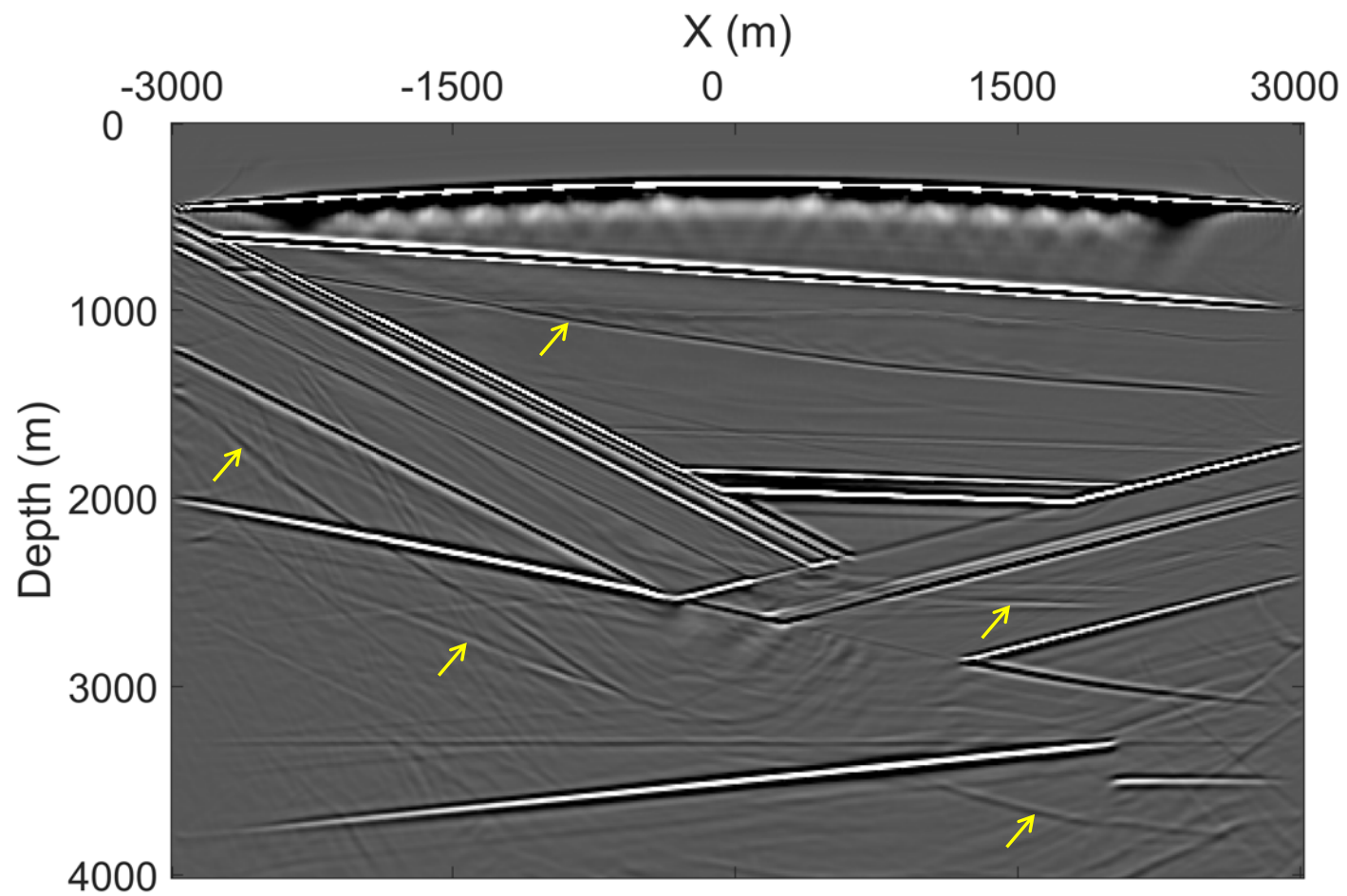


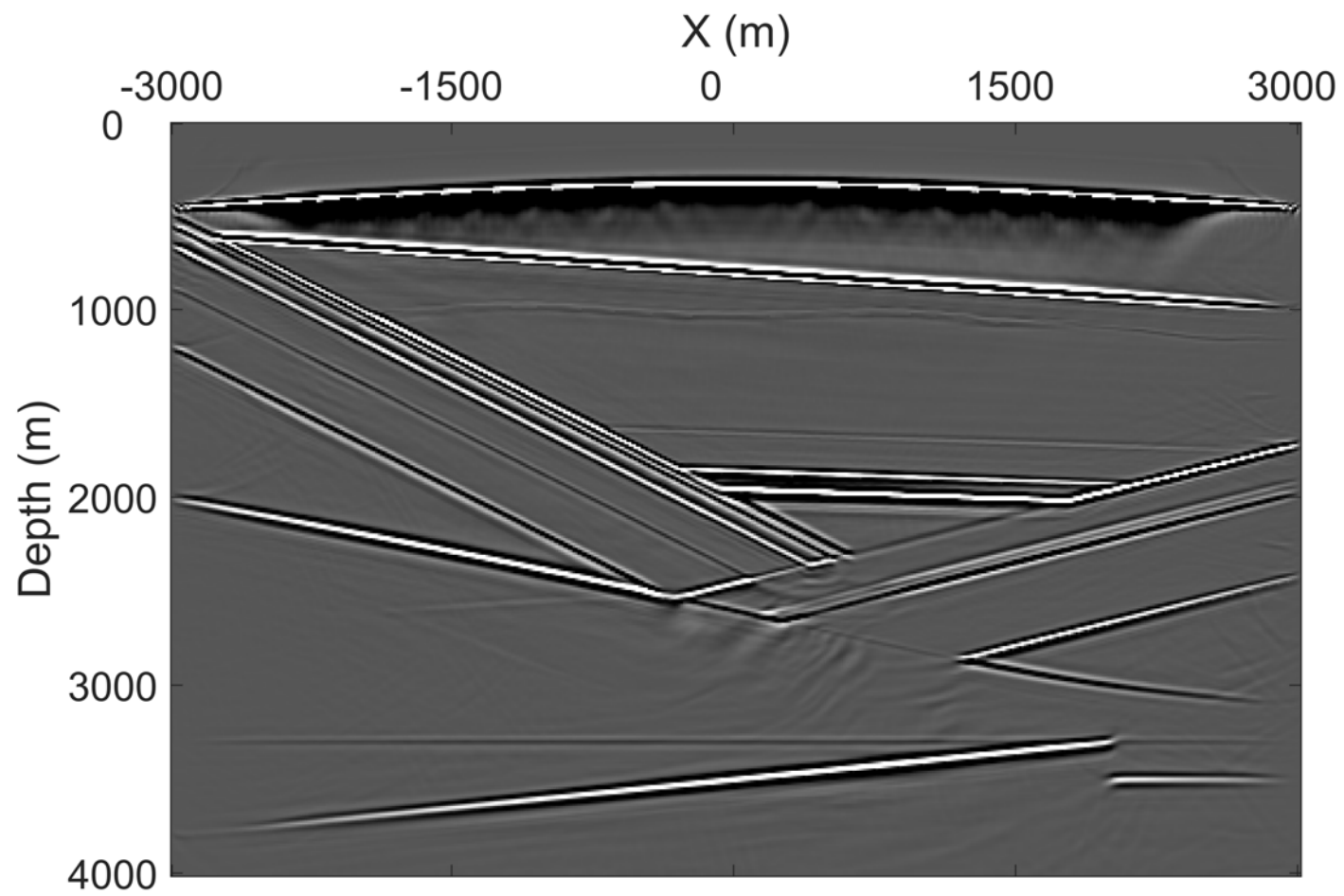












Extracting primaries with Marchenko

- README.4
- generate ‘primaries only’ for single shot
 - 2D example
 - Invisible medium

15 min.

Reference: Zhang, L., Thorbecke, J., Wapenaar, K., and Slob, E., 2019, *Transmission compensated primary reflection retrieval in the data domain and consequences for imaging*: Geophysics, Vol. 84 (4), Q27-Q36.

Invisible medium puzzle

- Find out how many reflectors there are in the data in demo/invisible. **Only** by inspecting data:

`shotsdx4_rp.su`

and using the program

`marchenko primaries`

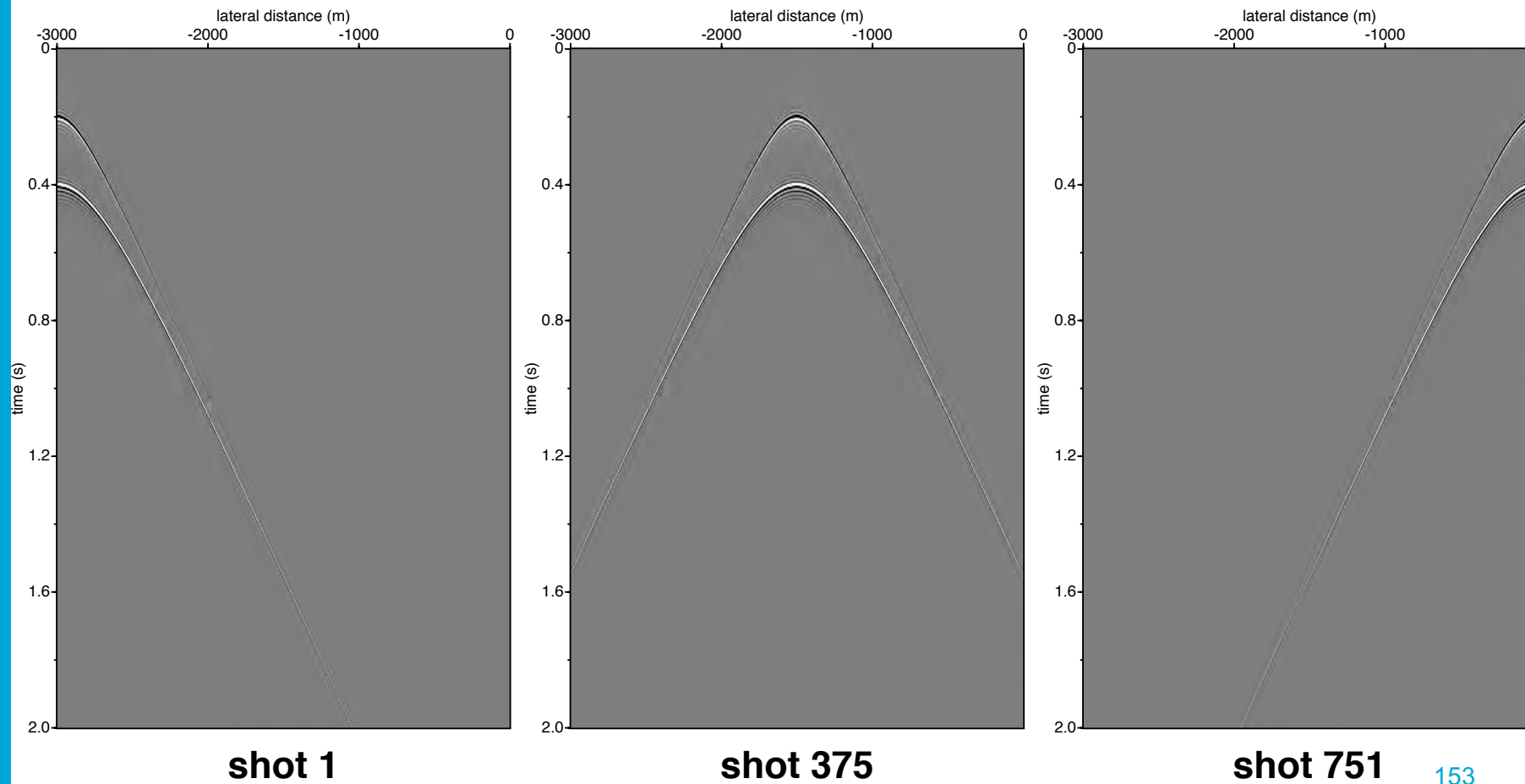
10 min.

*The solution can be found in the *.scr and README, but you are not allowed to read those.*

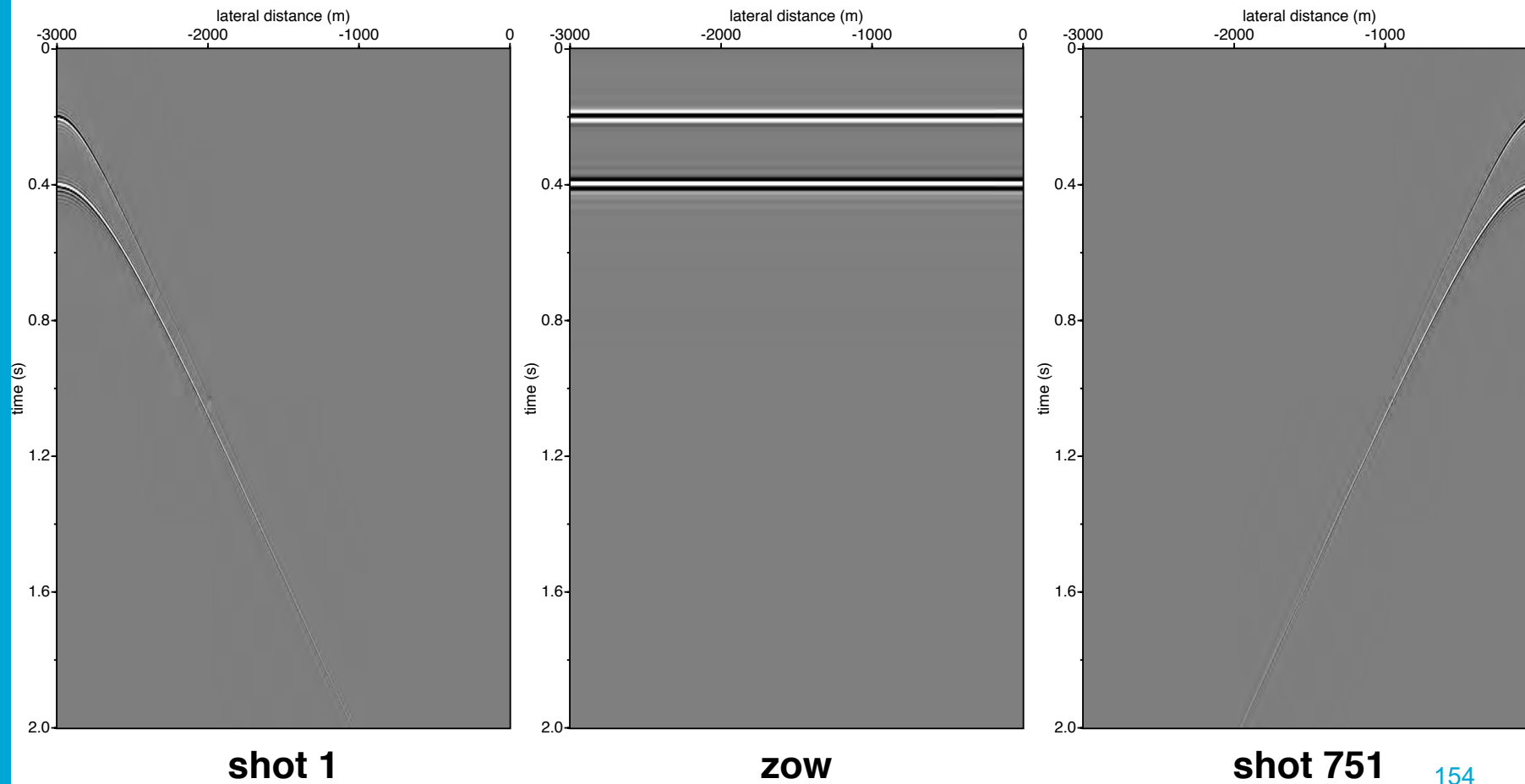
Reference: Lele Zhang, Jan Thorbecke, Kees Wapenaar, Evert Slob: *Data-driven internal multiple elimination and its consequences for imaging: A comparison of strategies*, accepted Geophysics



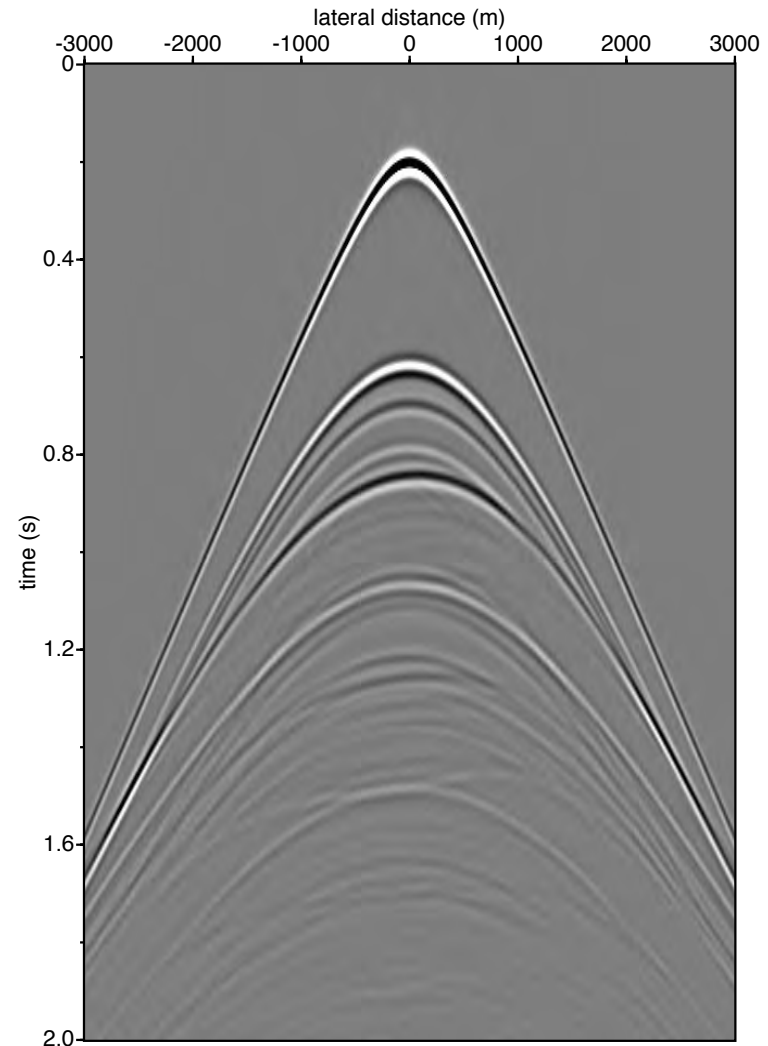
shotsdx4_rp.su



shotsdx4_rp.su

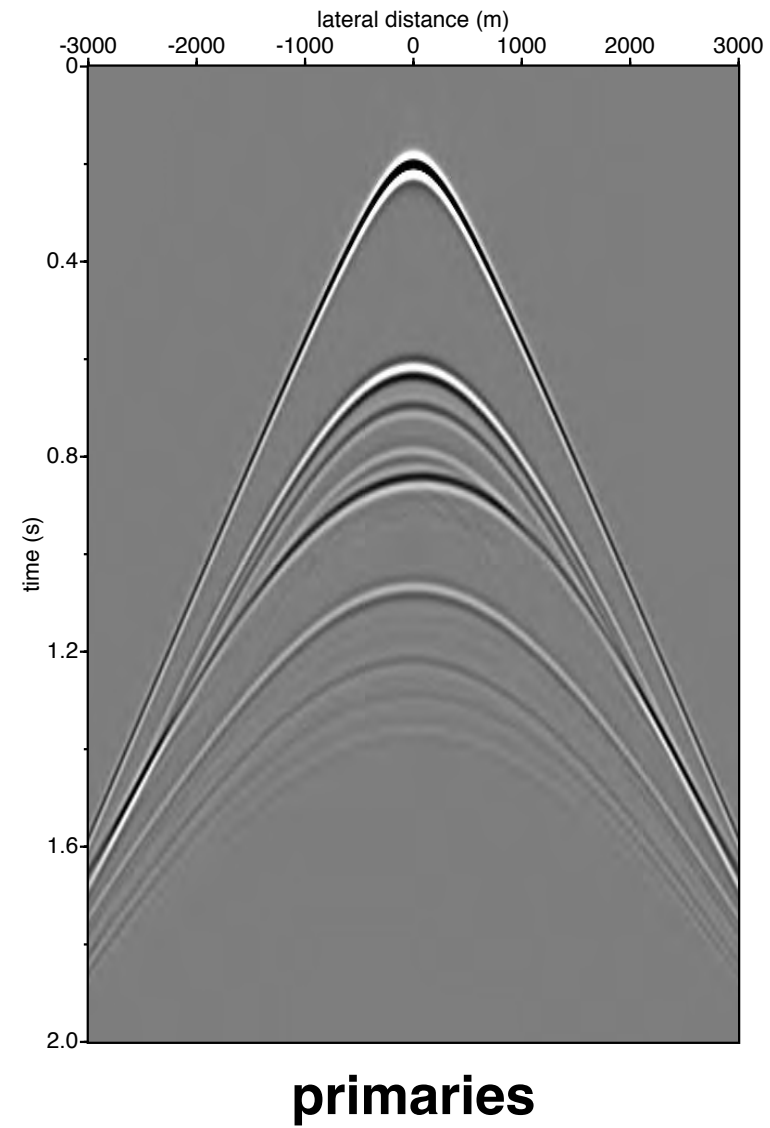


twoD figures

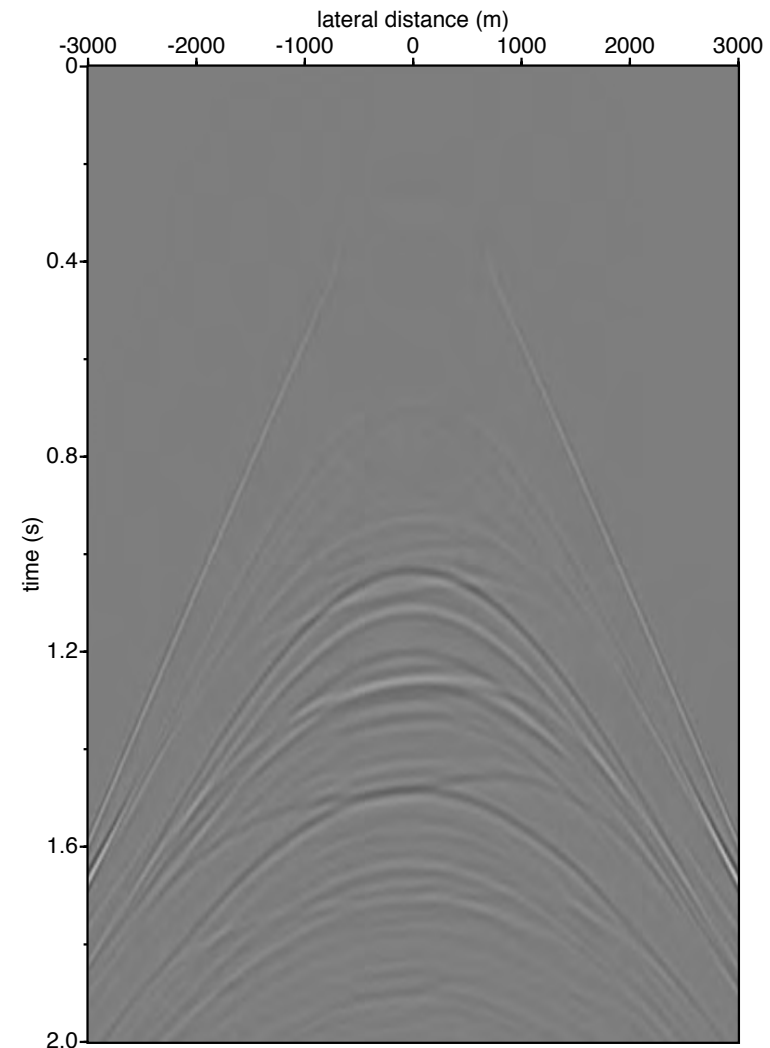


original shot

twoD figures



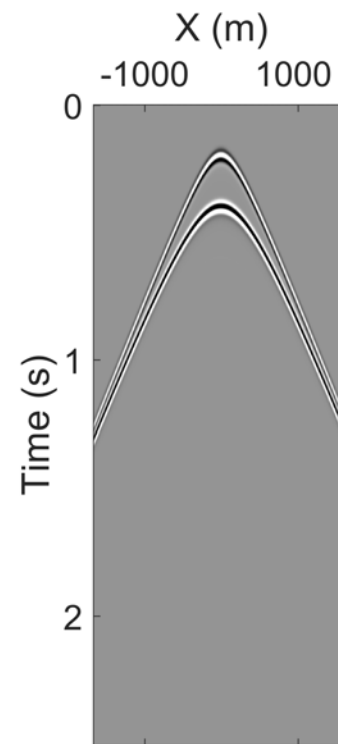
twoD figures



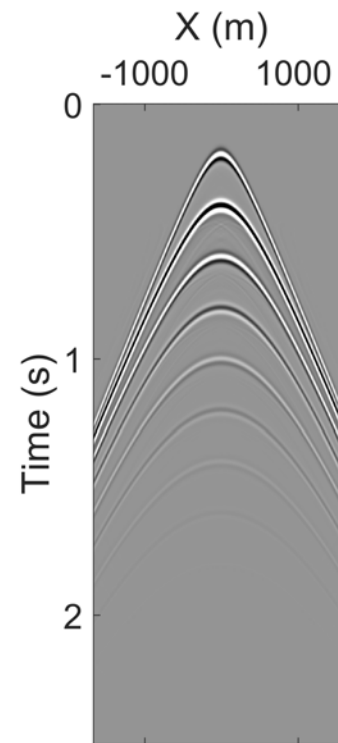
removed multiples



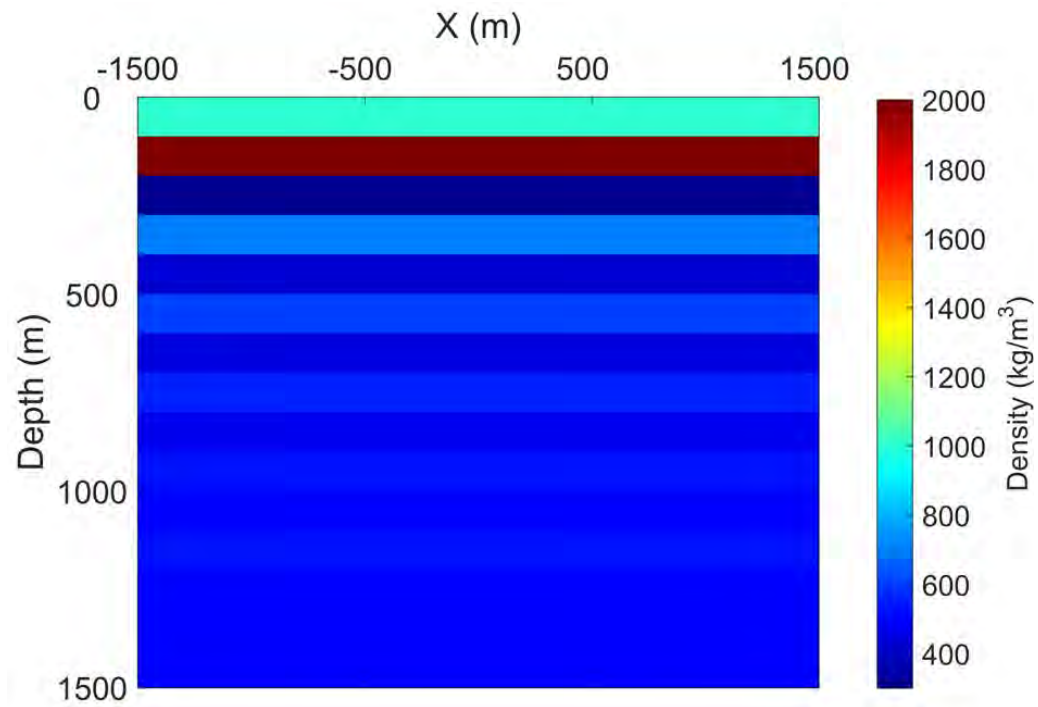
Invisible medium



Invisible medium



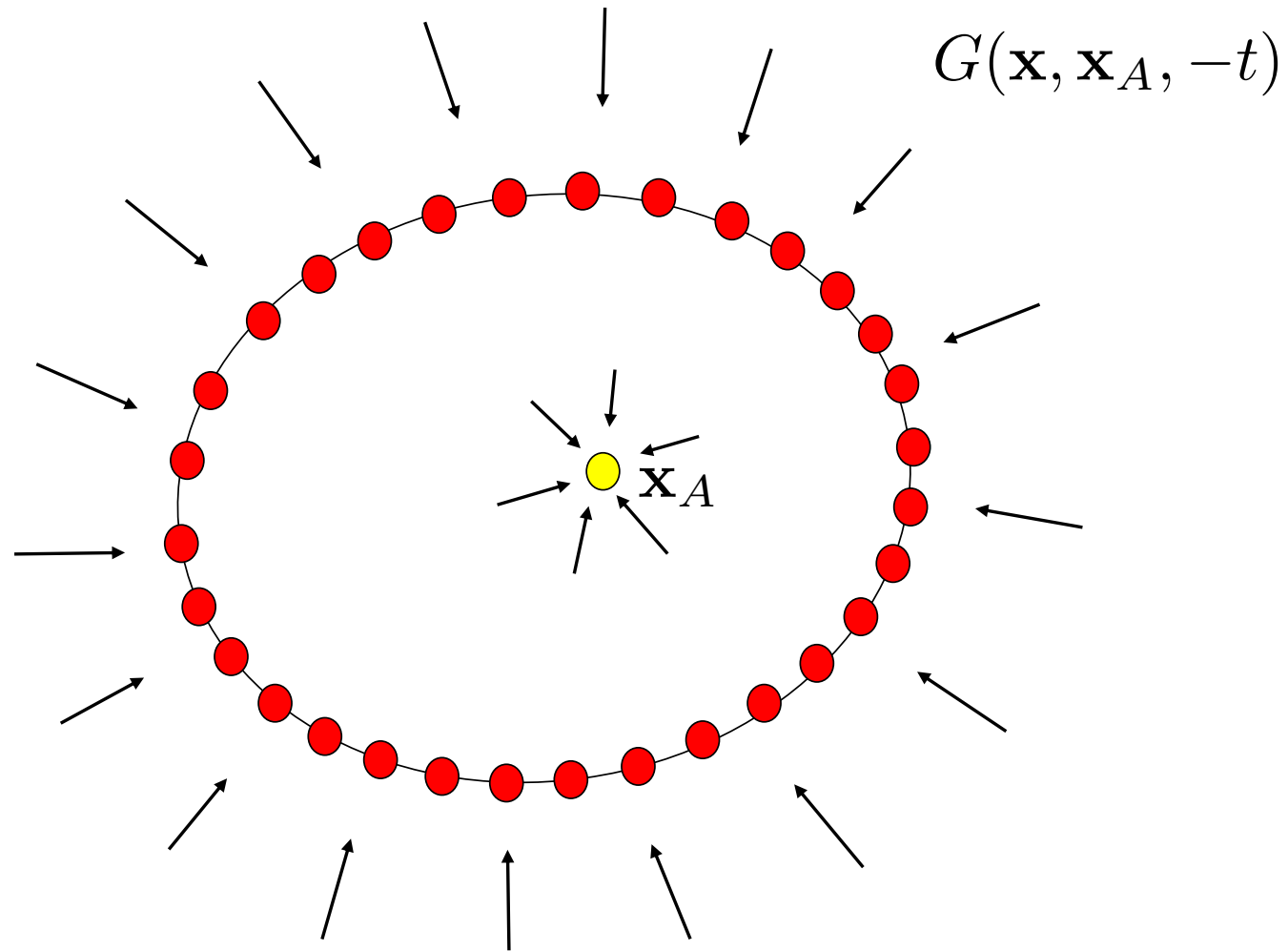
Invisible medium



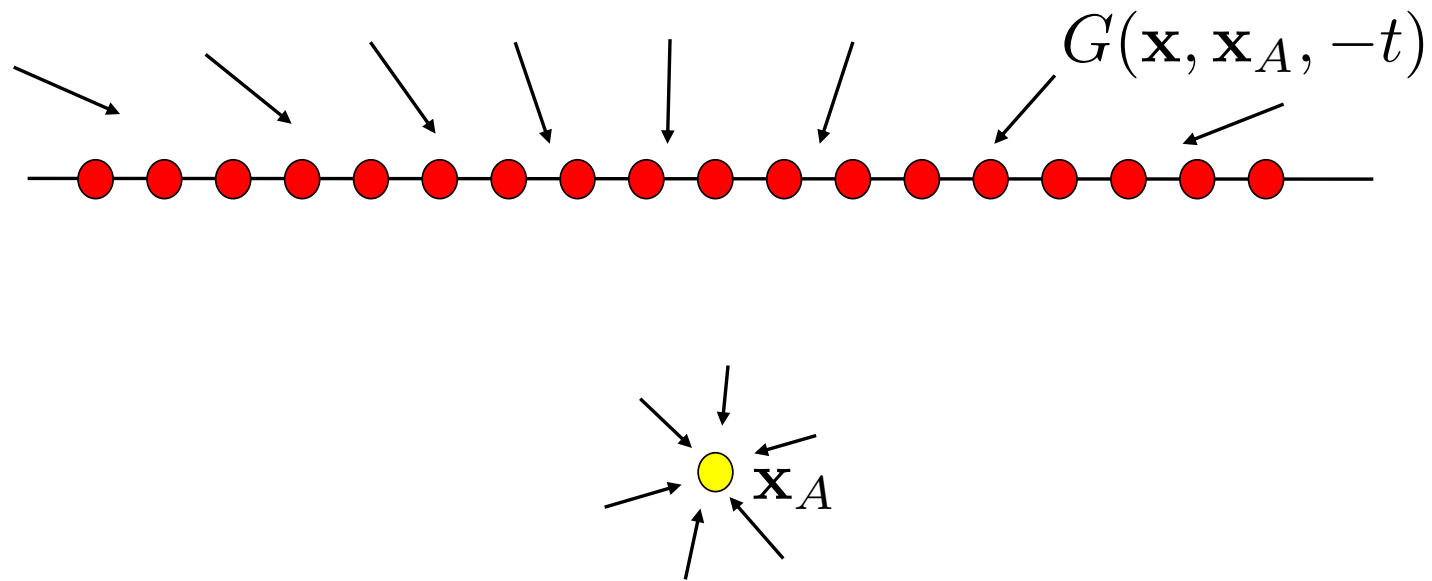
Homogeneous Green's function



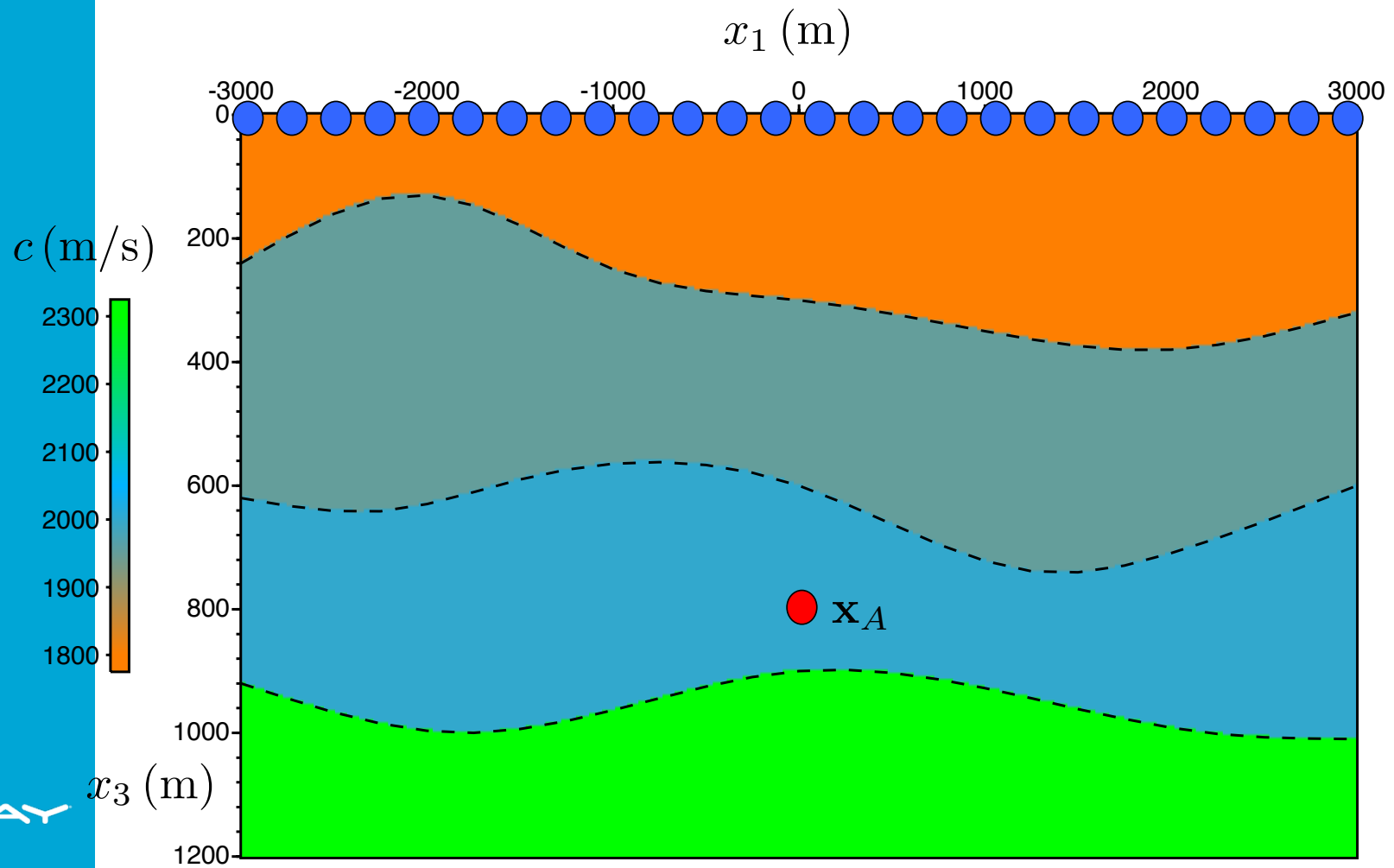
Reference: Brackenhoff, J., Thorbecke, J., and Wapenaar, K.: *Monitoring induced distributed double-couple sources using Marchenko-based virtual receivers*, Solid Earth Discuss., <https://doi.org/10.5194/se-2018-142>, in review, 2019.



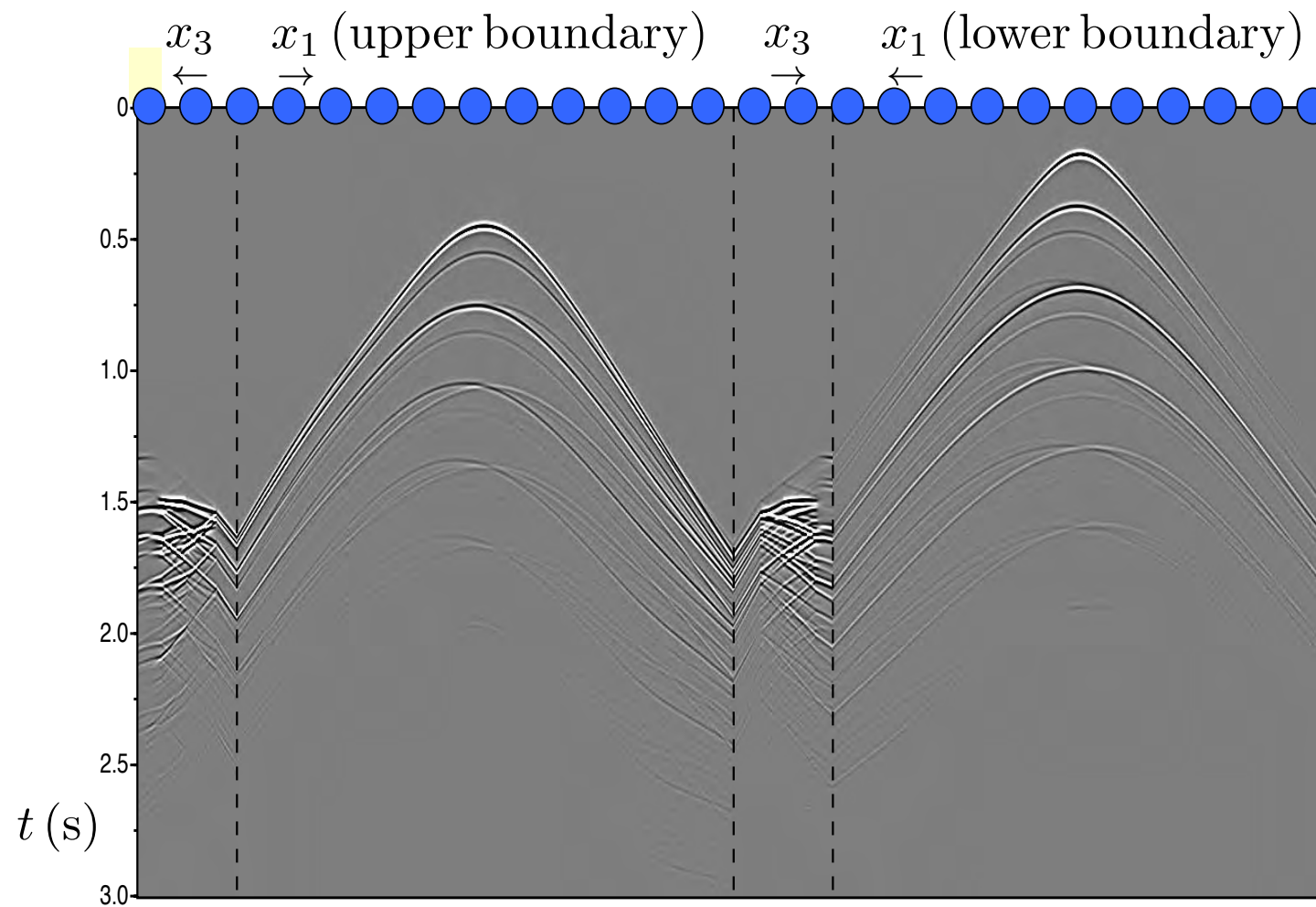
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \oint_{\partial \mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * G(\mathbf{x}, \mathbf{x}_A, -t) d^2 \mathbf{x}$$



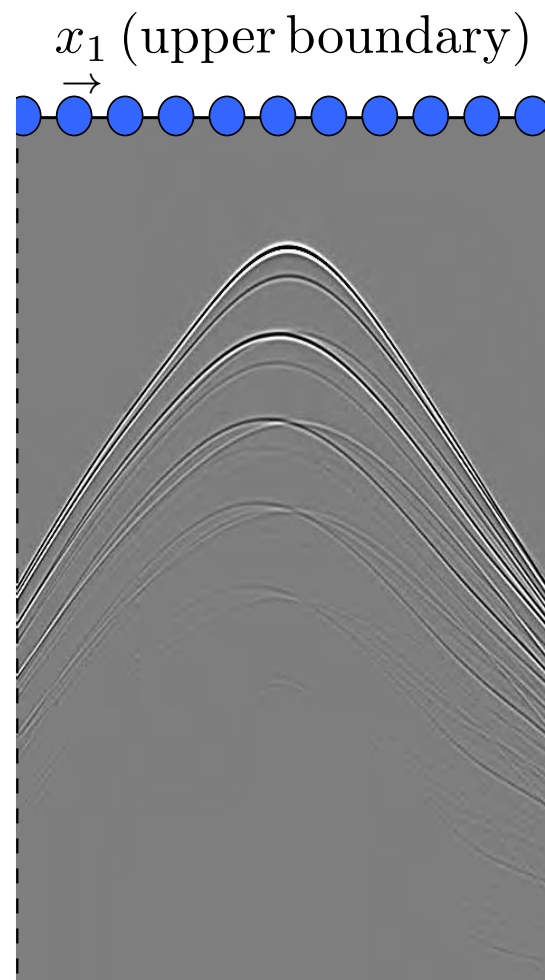
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \neq \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * G(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$



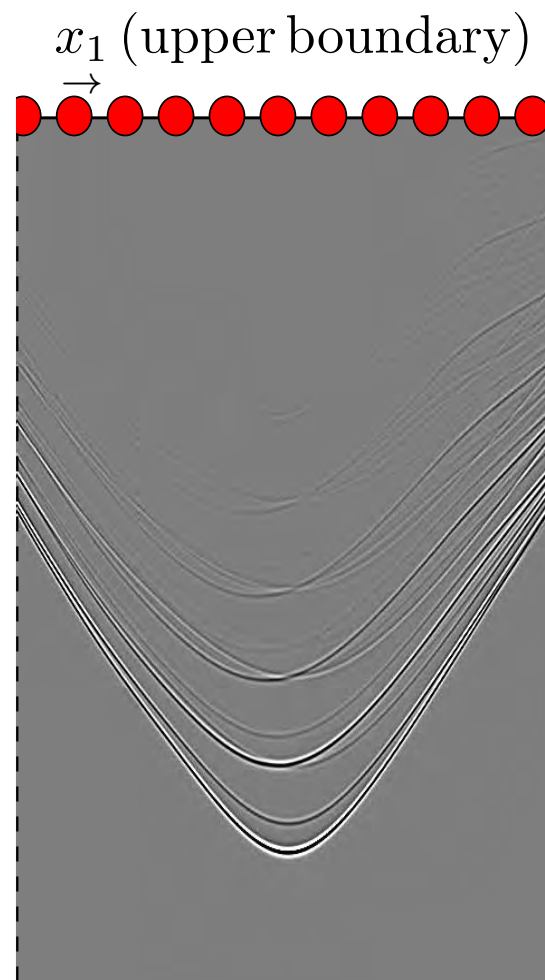
Single-sided time-reversal experiment



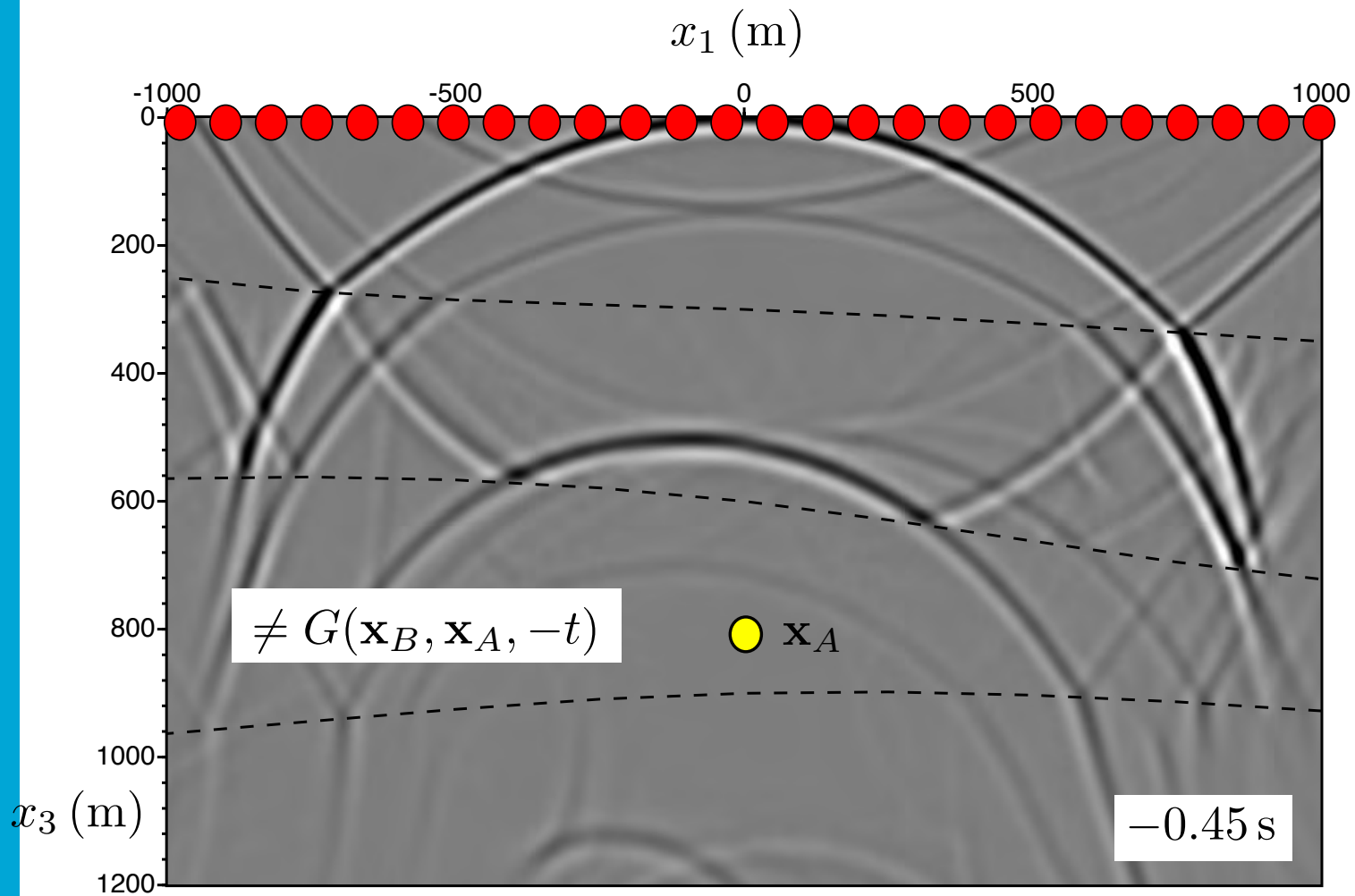
Single-sided time-reversal experiment



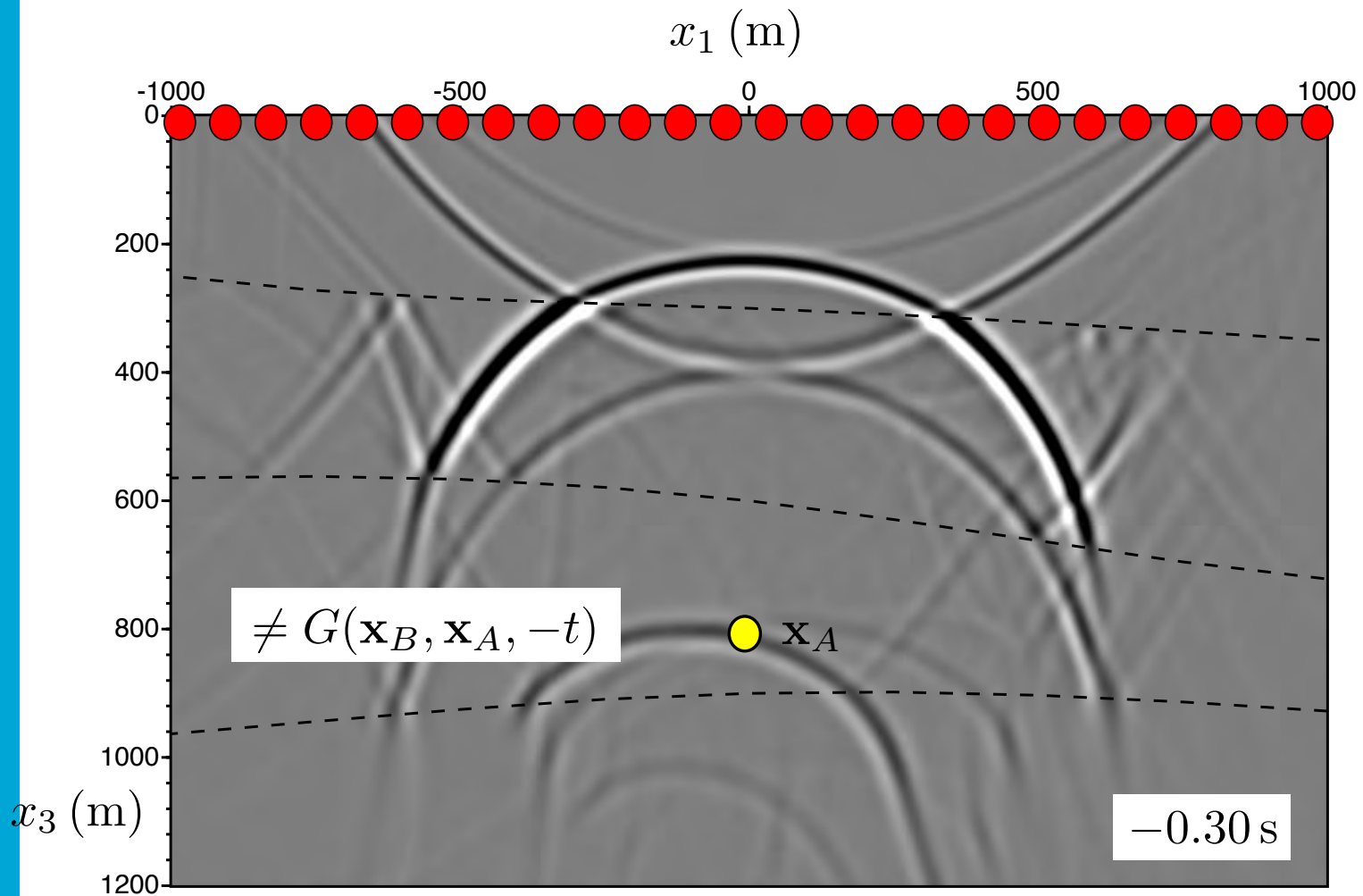
Single-sided time-reversal experiment



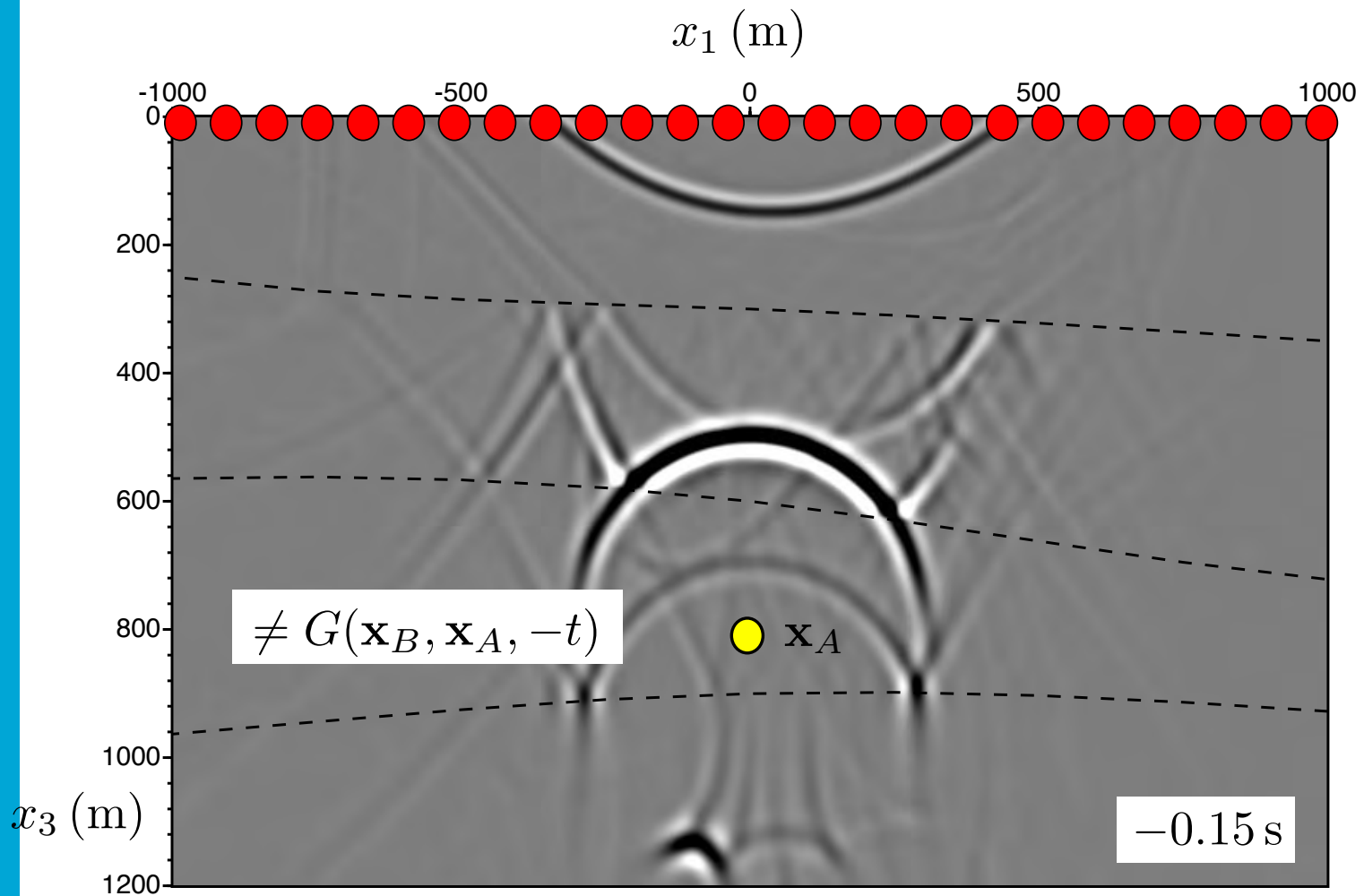
Single-sided time-reversal experiment



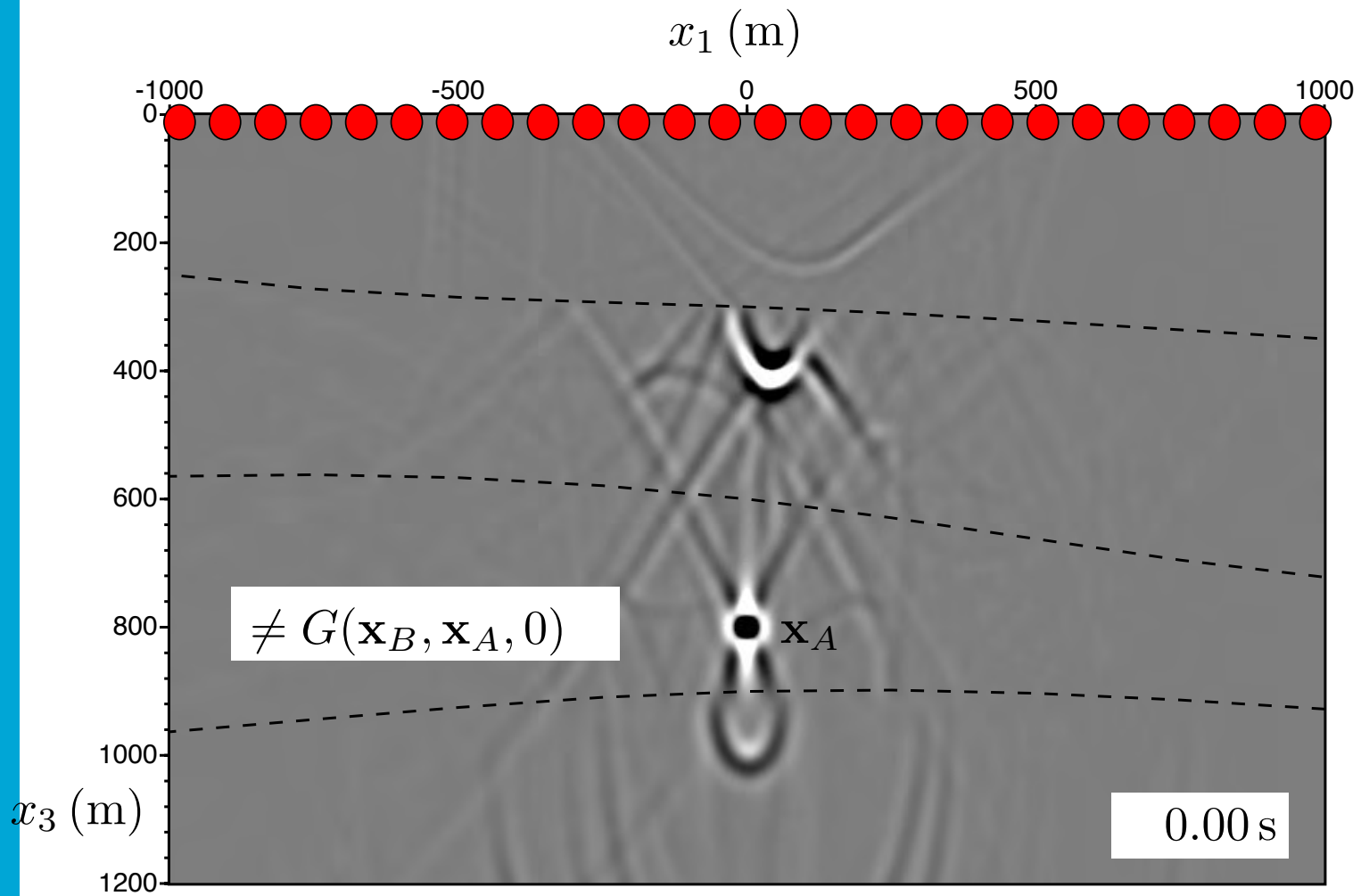
Single-sided time-reversal experiment



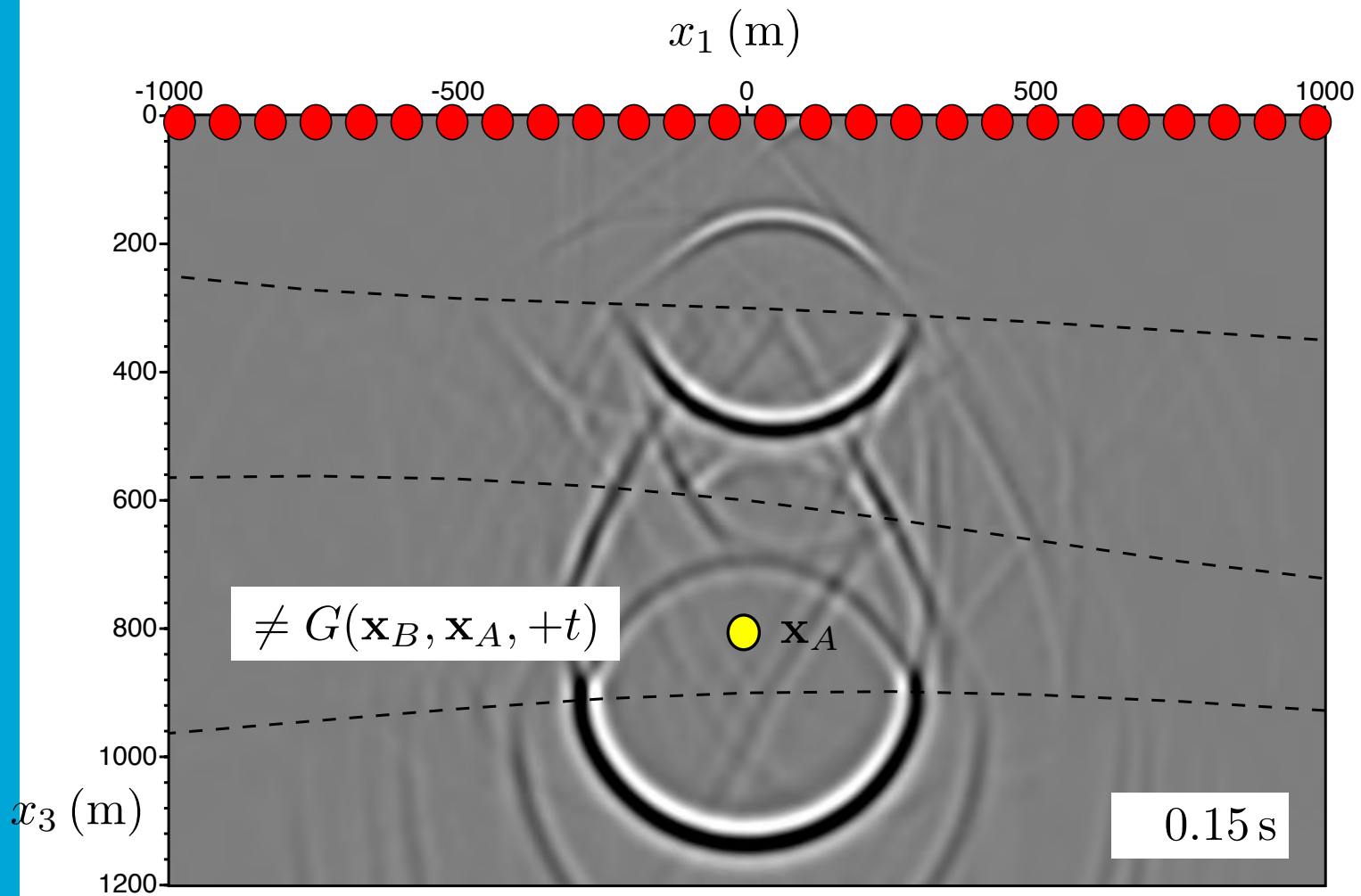
Single-sided time-reversal experiment



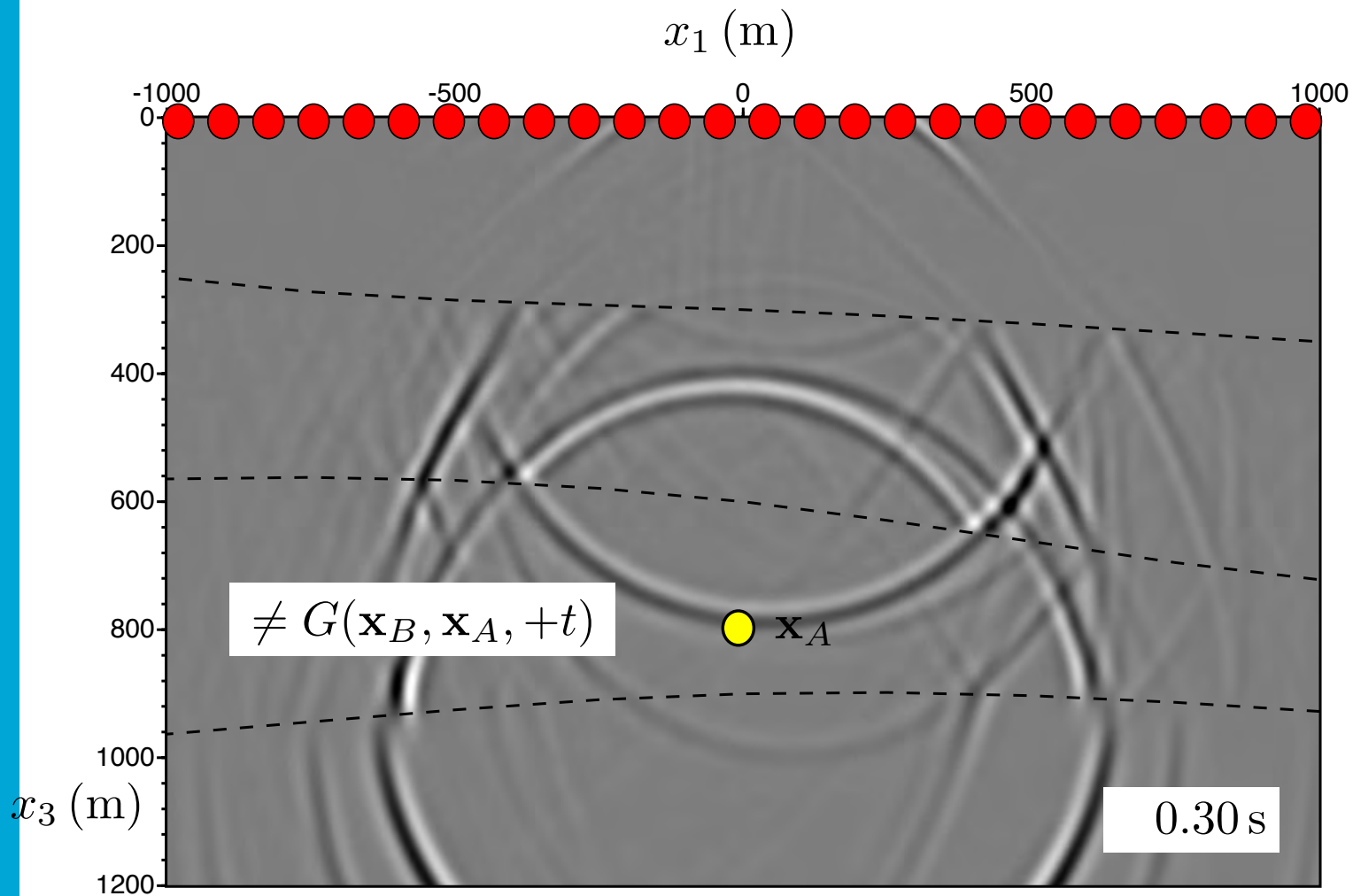
Single-sided time-reversal experiment



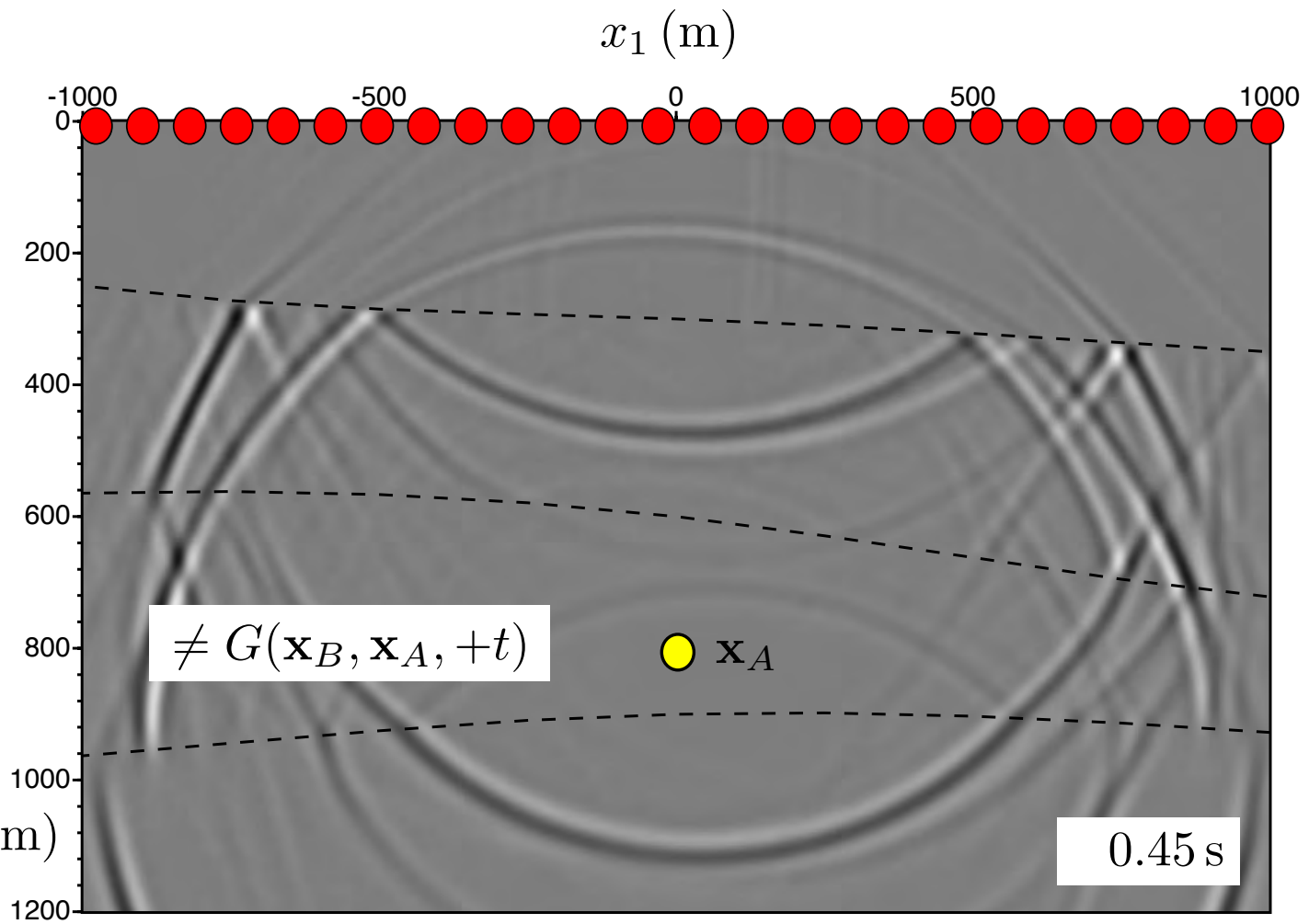
Single-sided time-reversal experiment



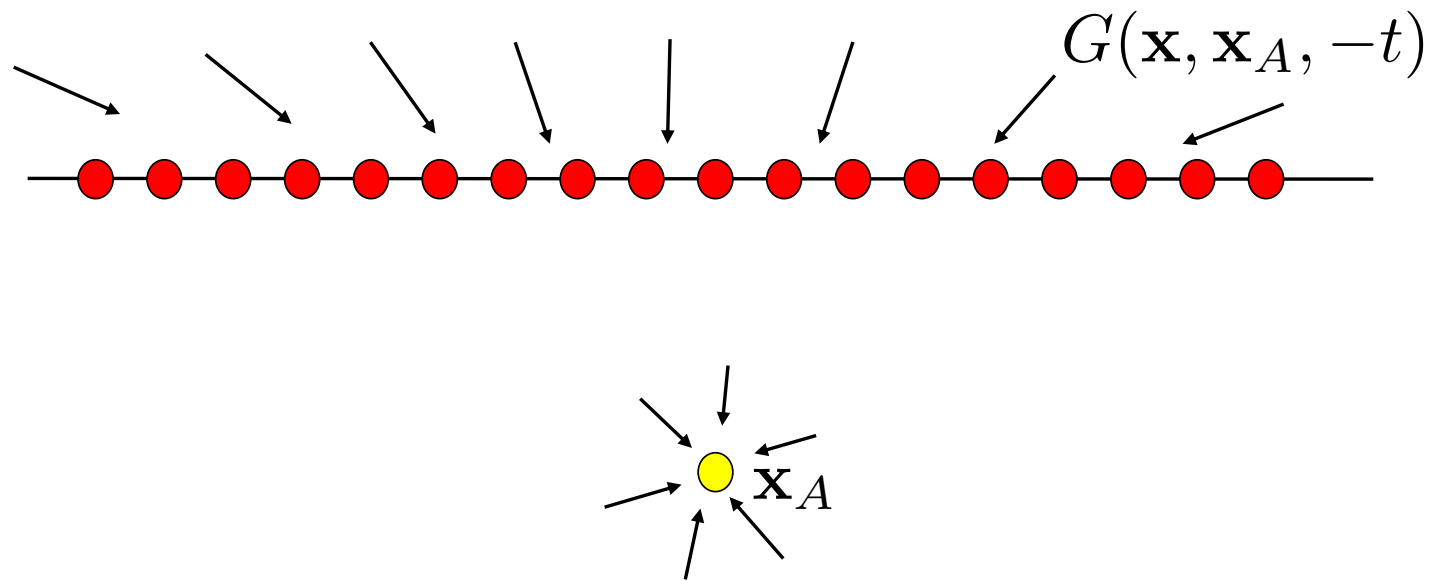
Single-sided time-reversal experiment



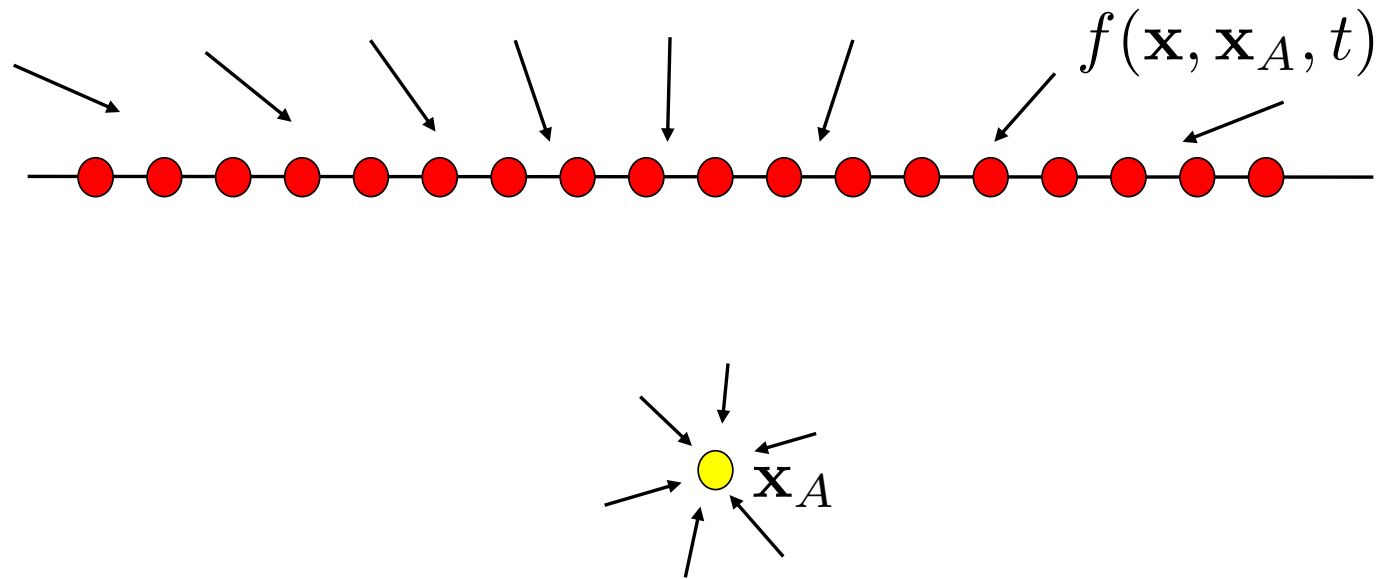
Single-sided time-reversal experiment



Single-sided time-reversal experiment

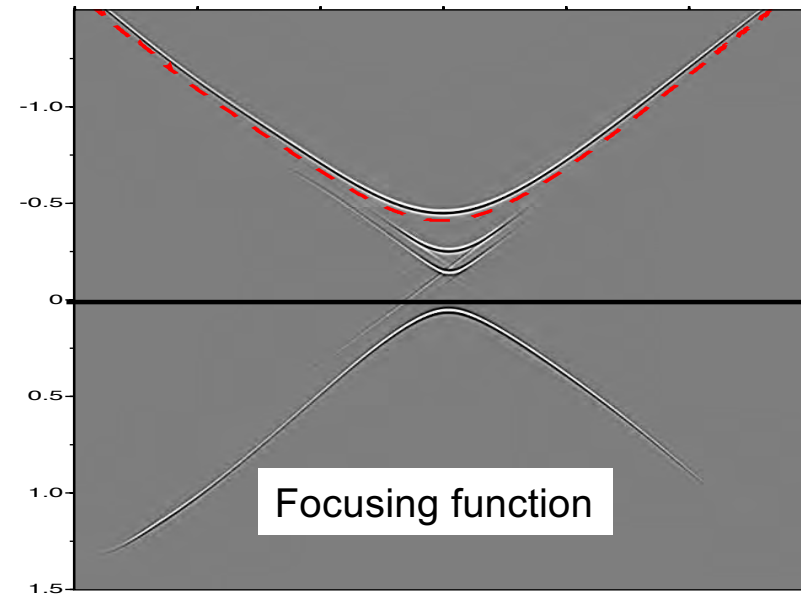
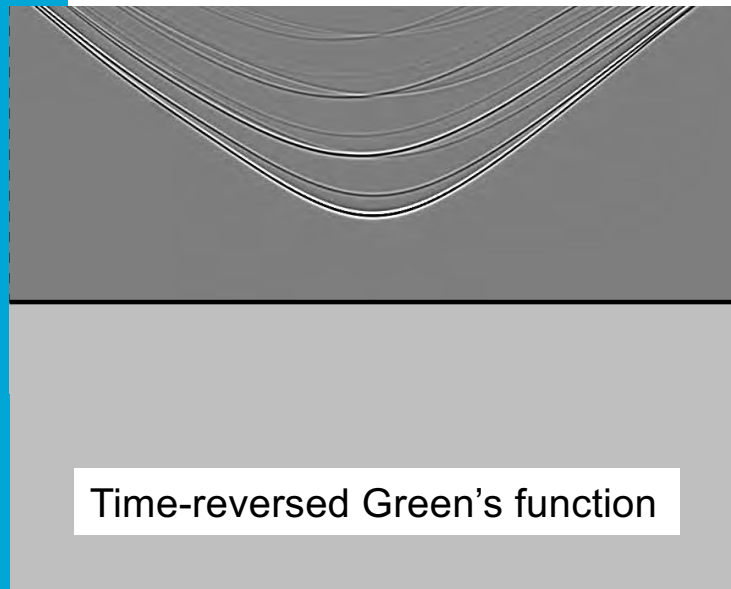


$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \neq \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * G(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$

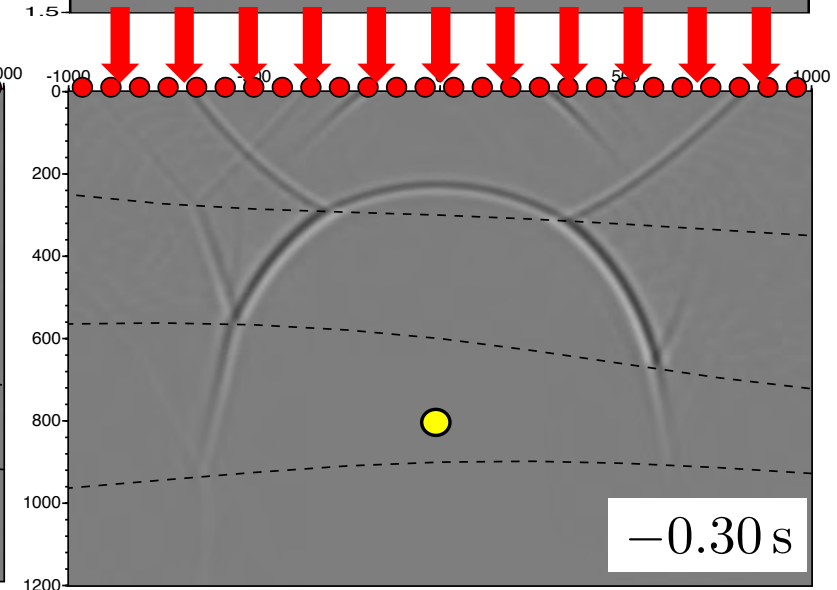
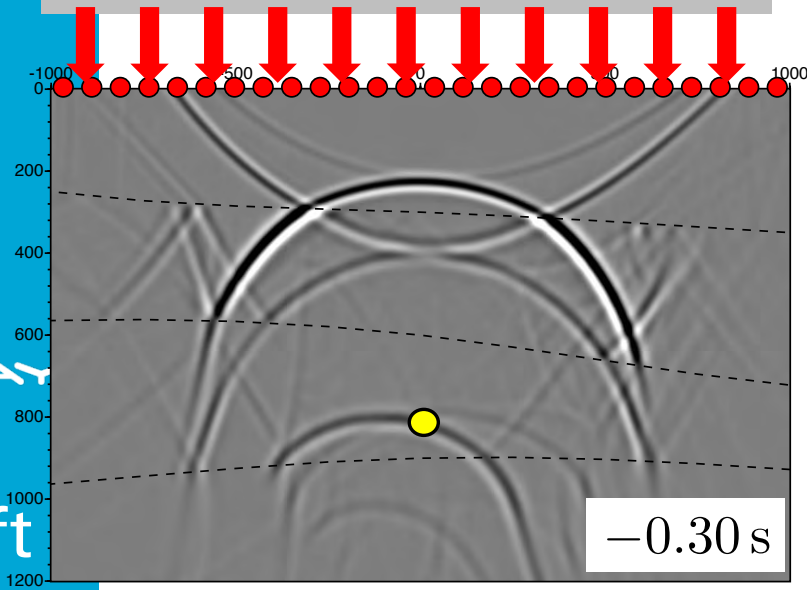
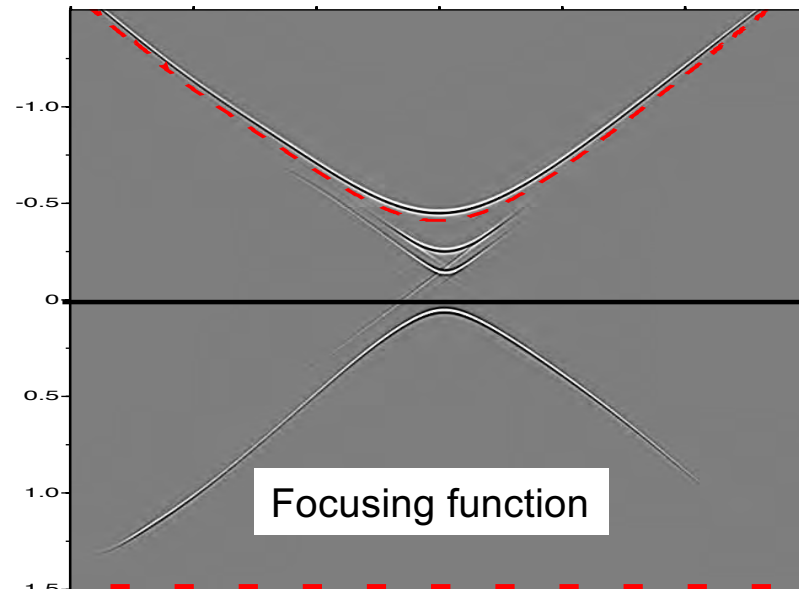
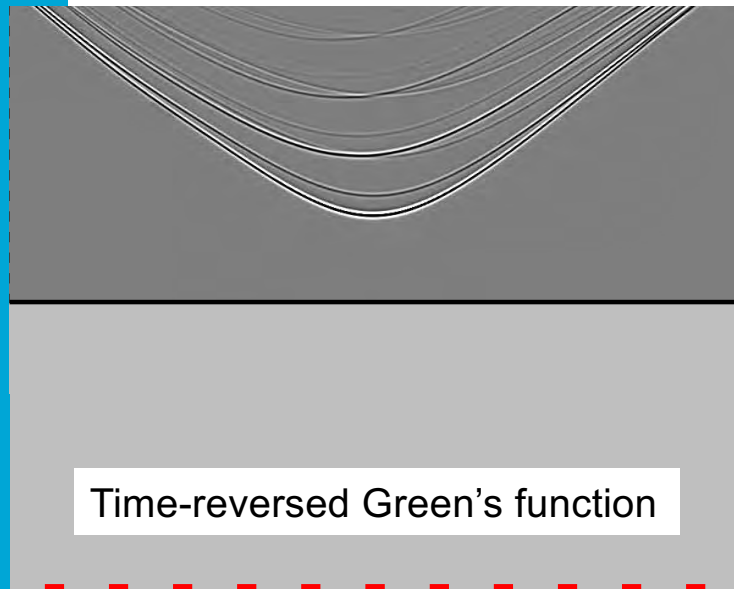


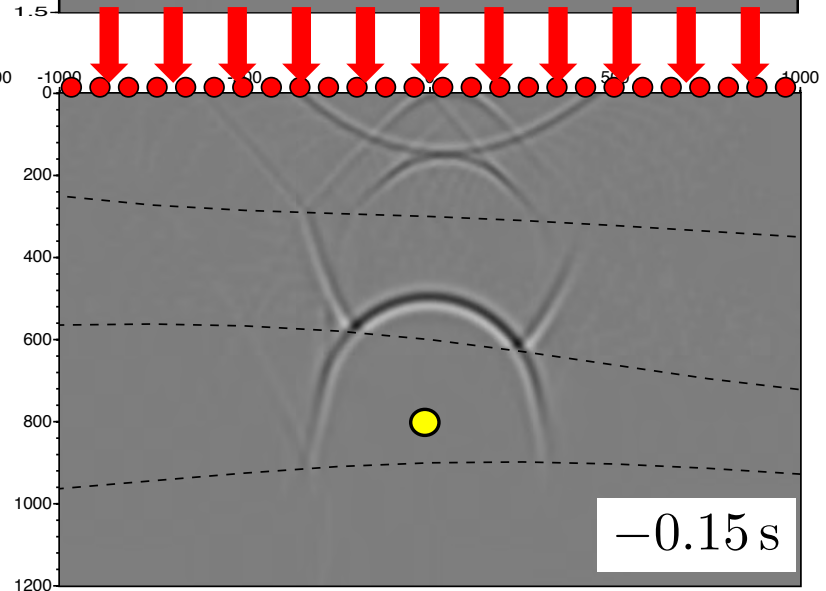
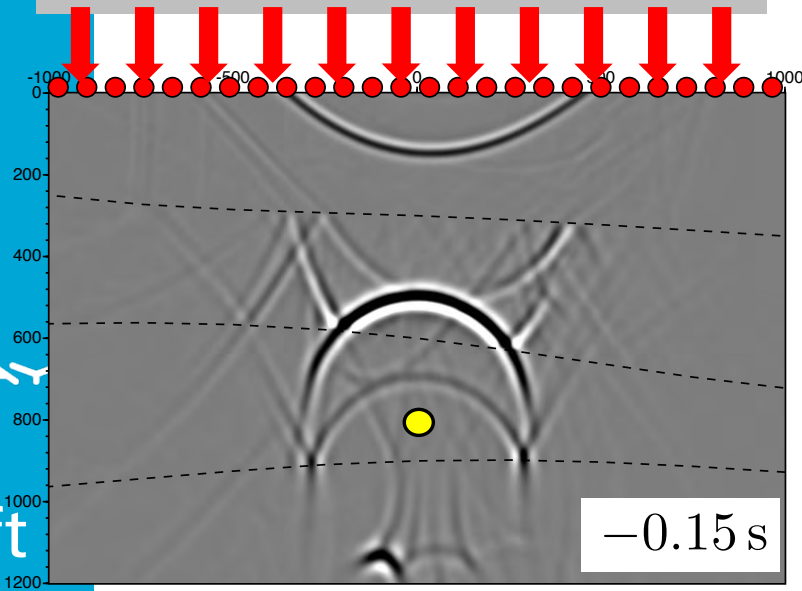
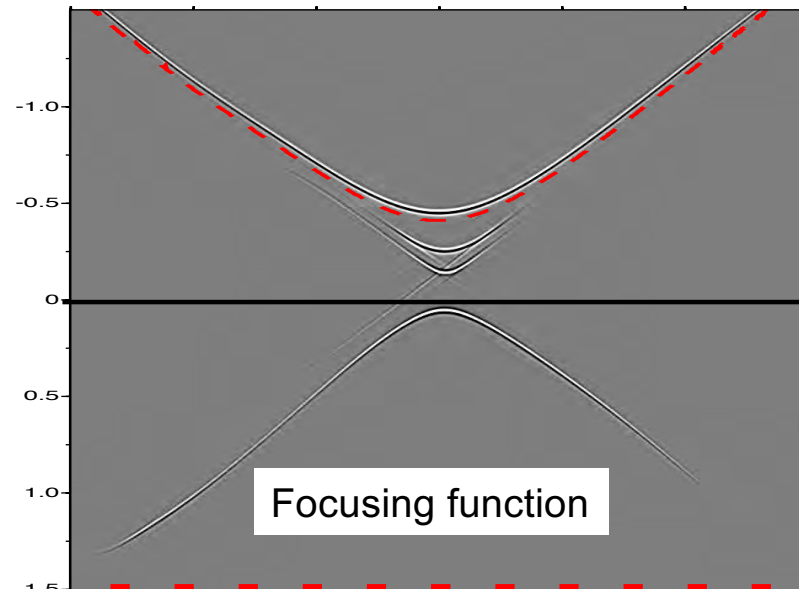
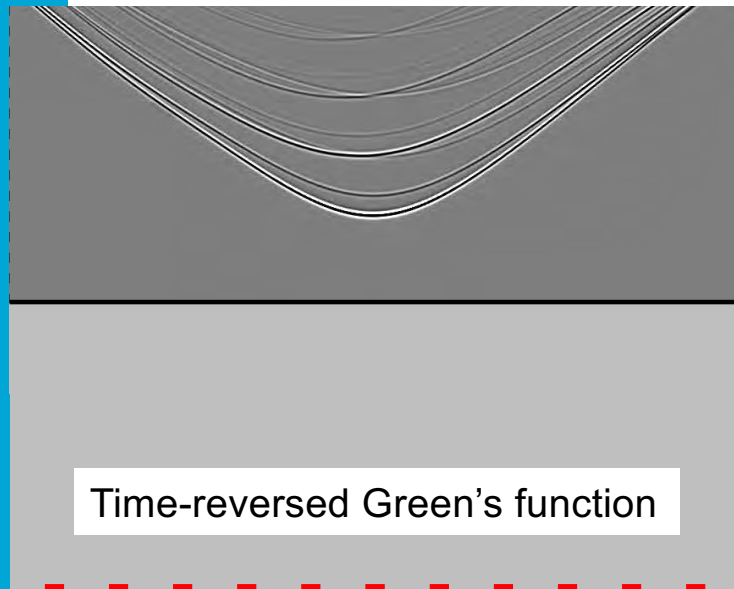
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * f(\mathbf{x}, \mathbf{x}_A, t) d^2\mathbf{x} \\ + \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, -t) * f(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$

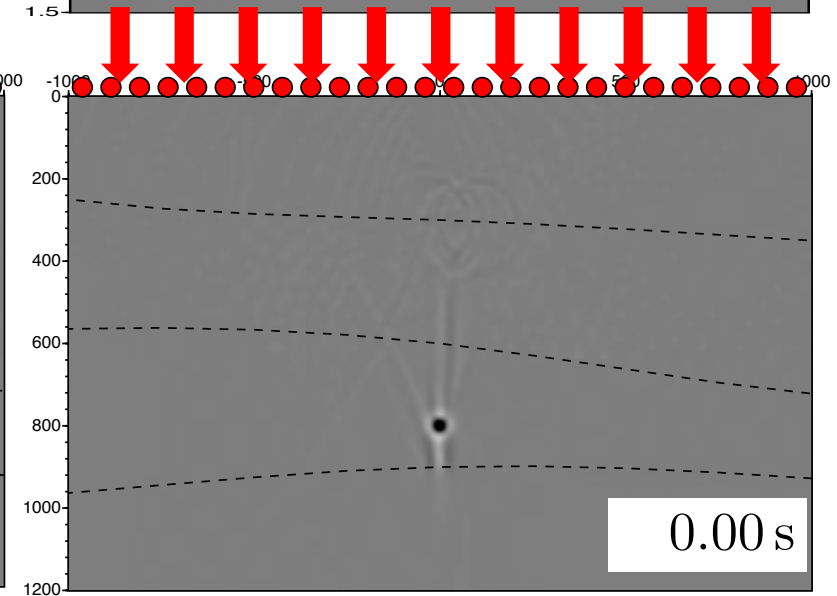
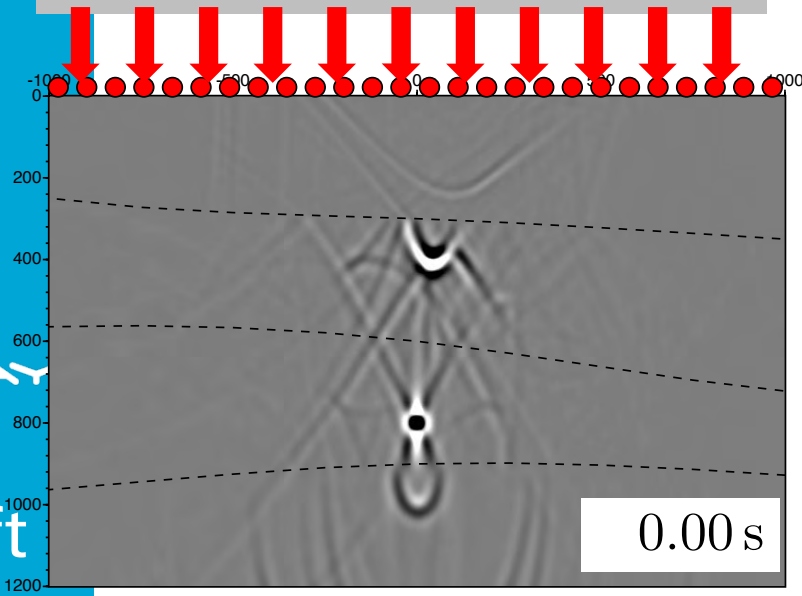
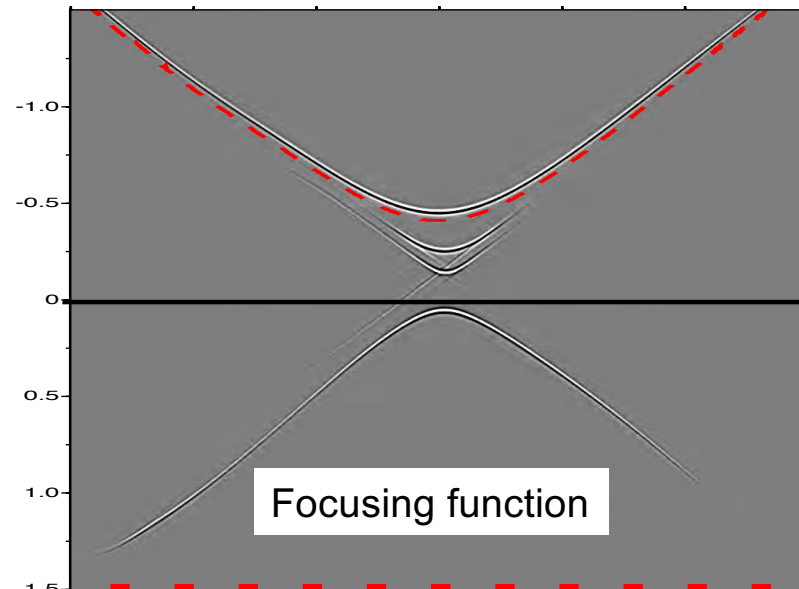
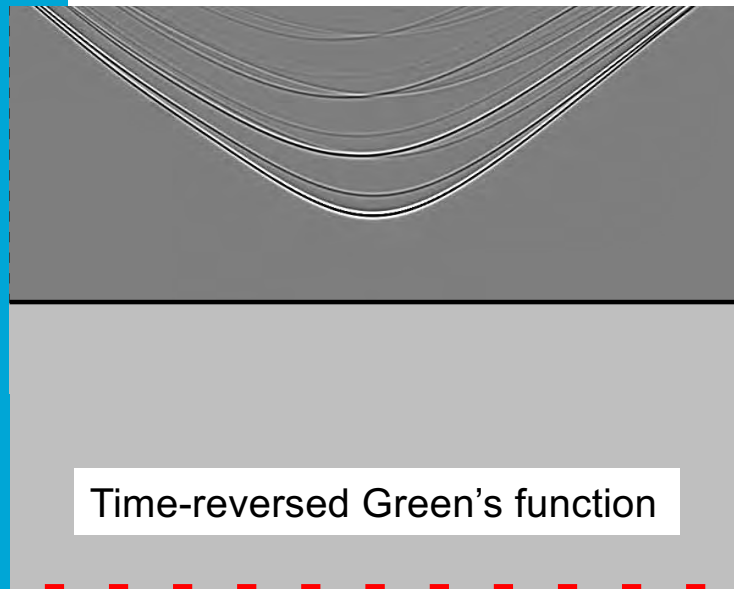
References: Wapenaar et al., *Green's function retrieval from reflection data*, JASA 2014;
Virtual sources and their responses, Geophys. Prosp. 2017

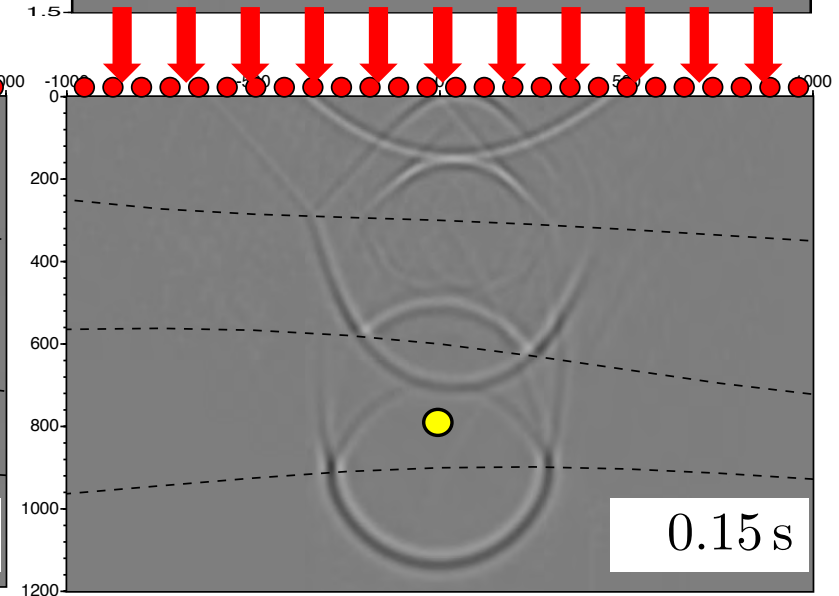
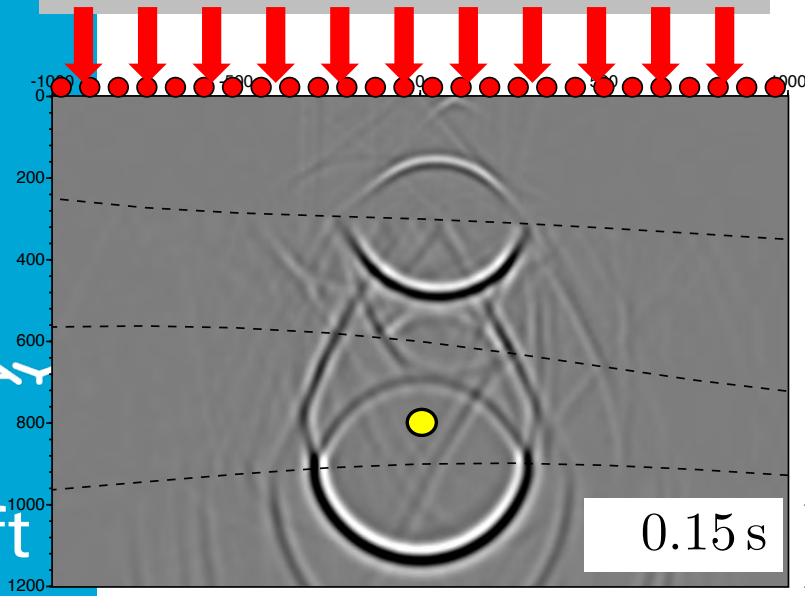
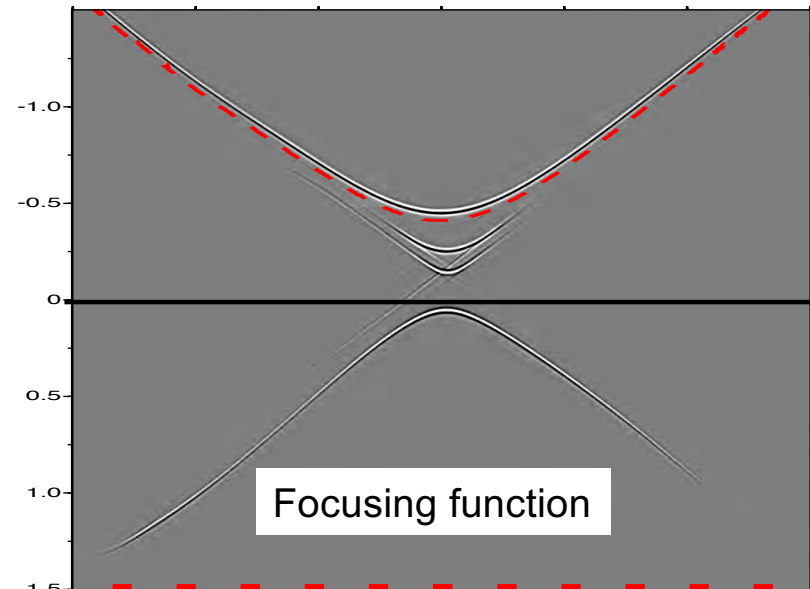
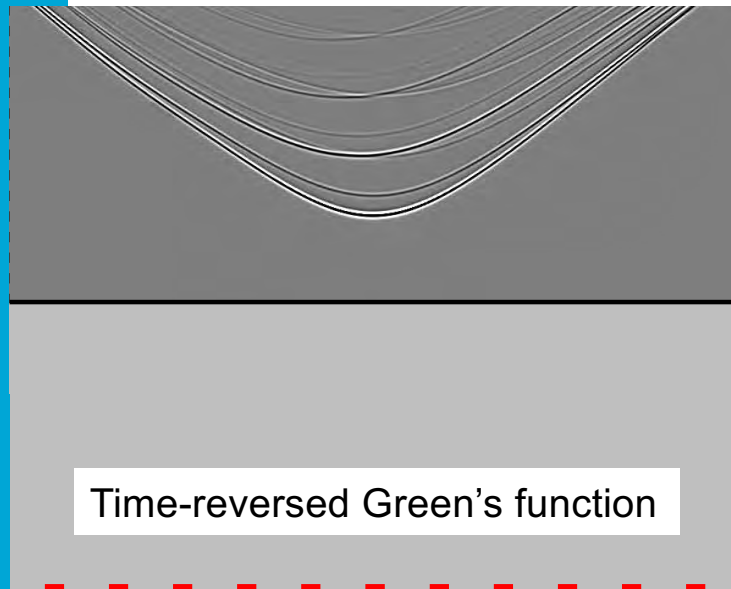


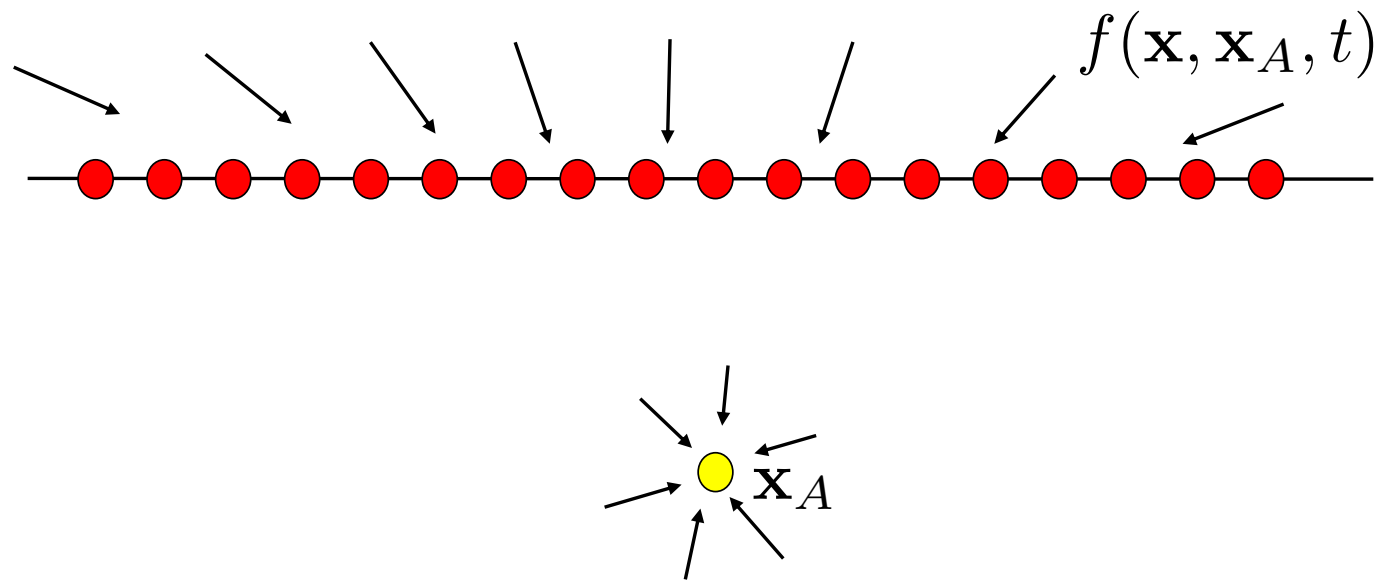
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * f(\mathbf{x}, \mathbf{x}_A, t) d^2\mathbf{x}$$



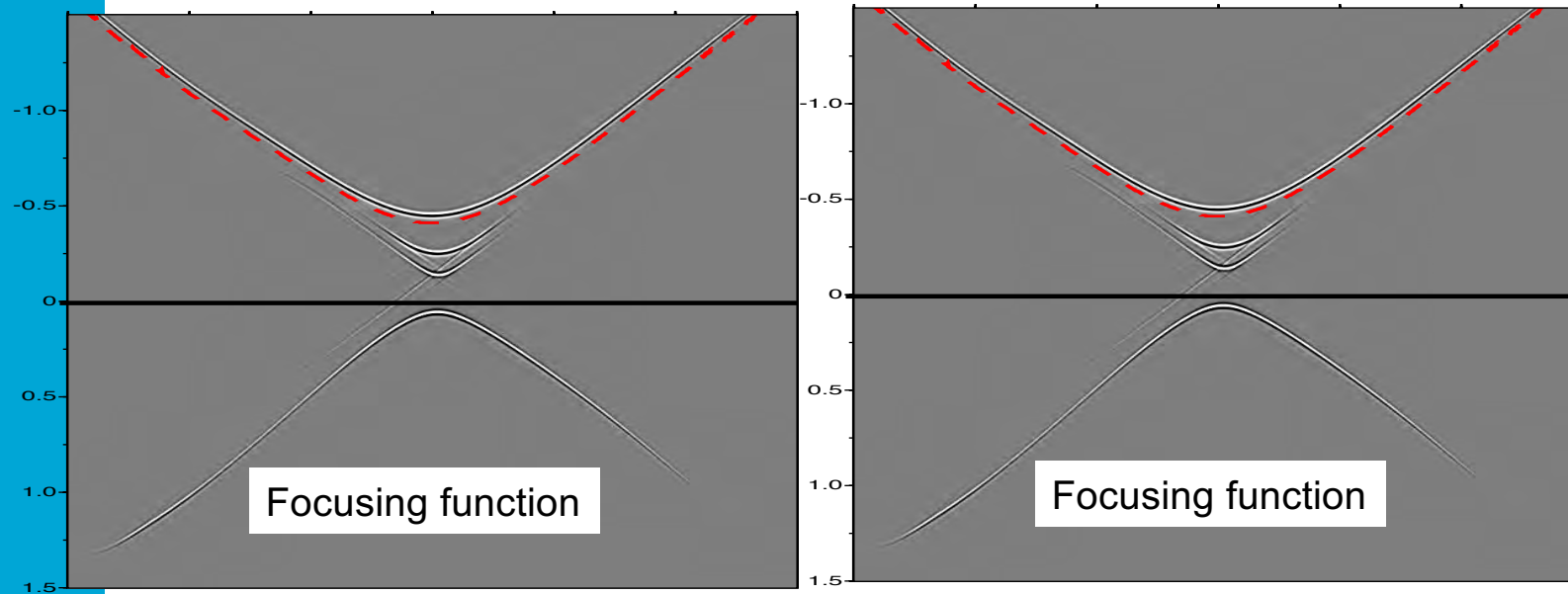




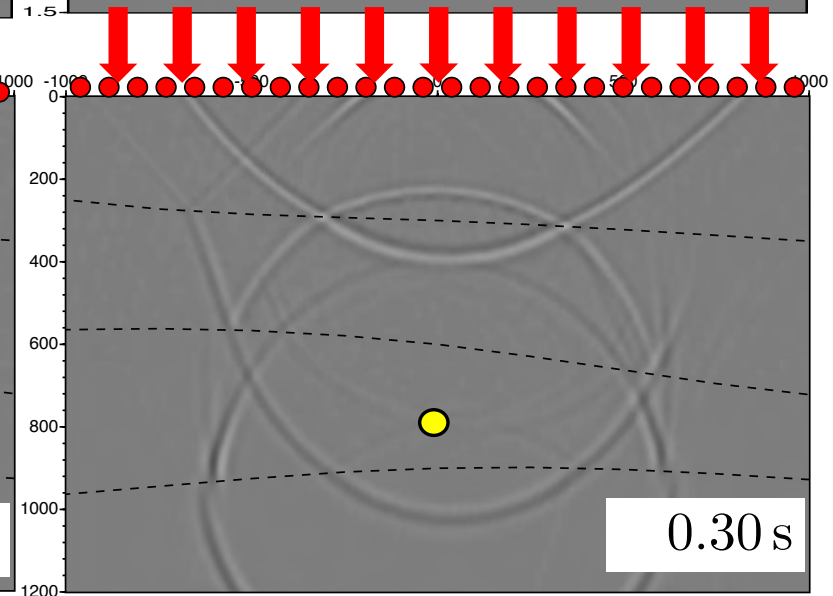
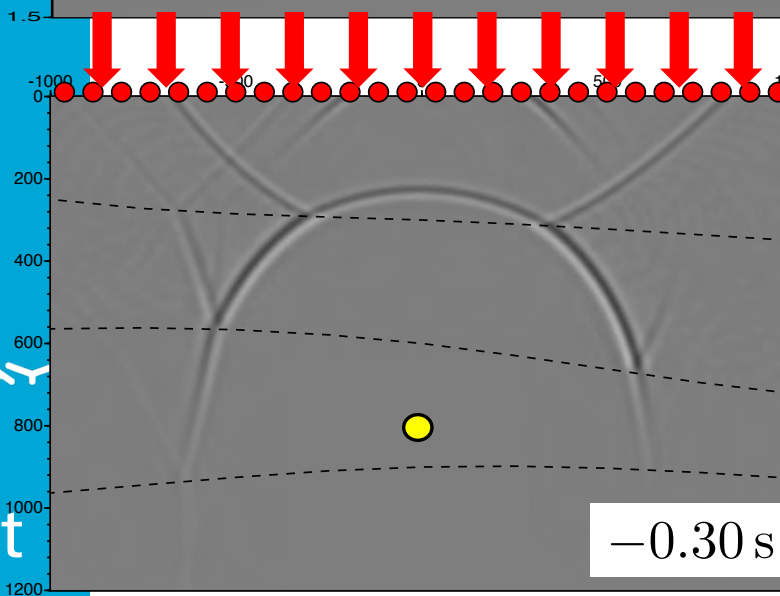
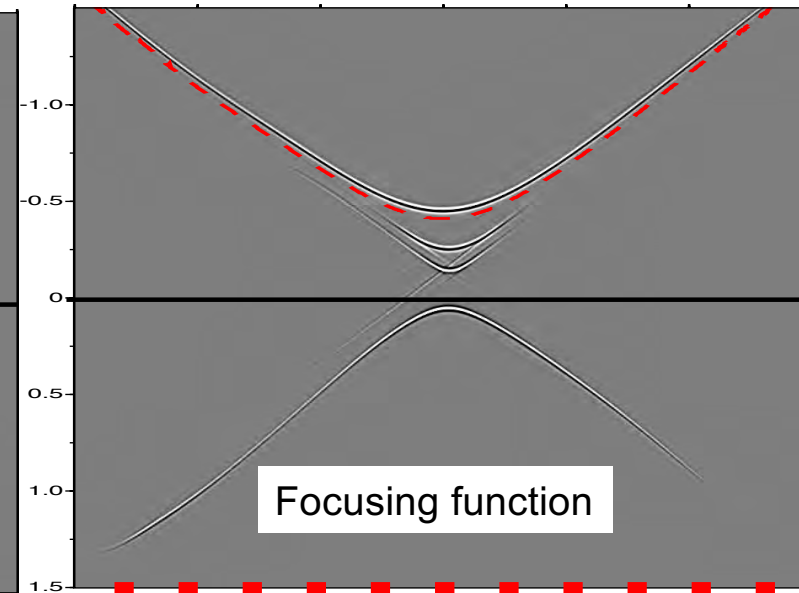
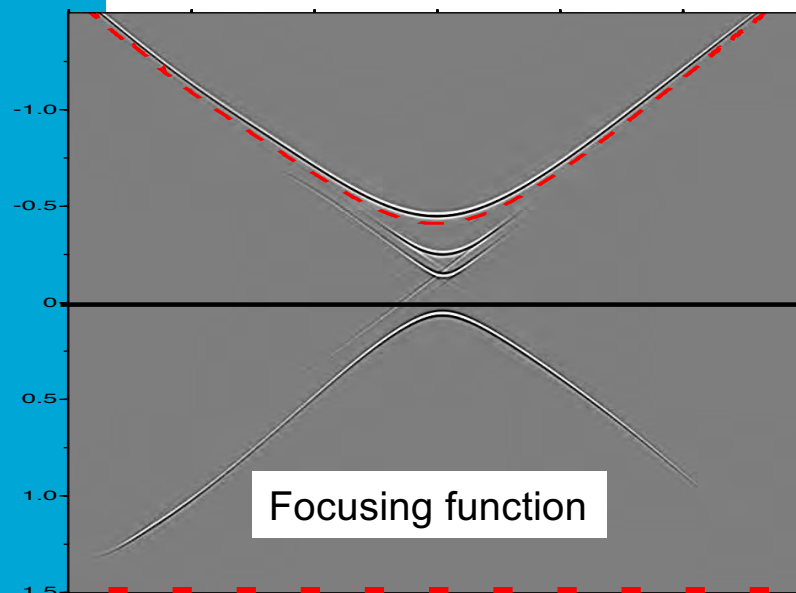


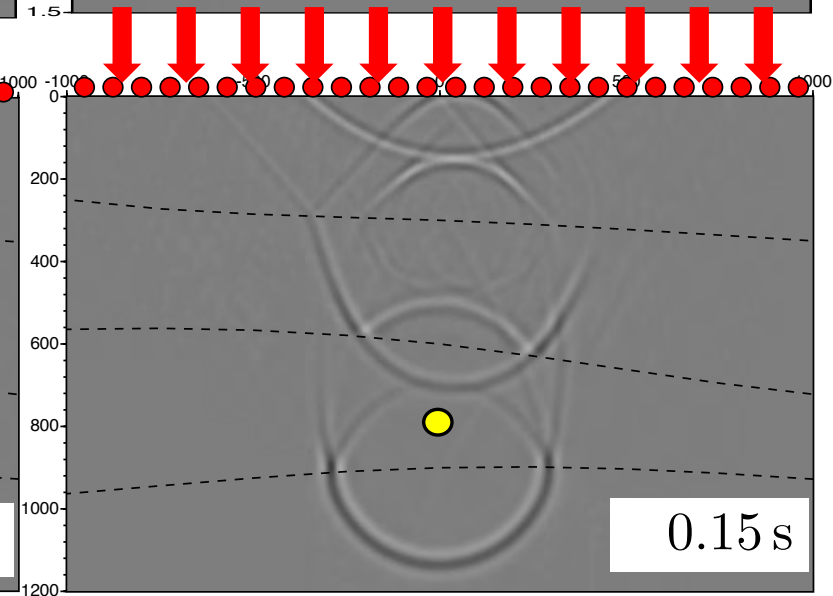
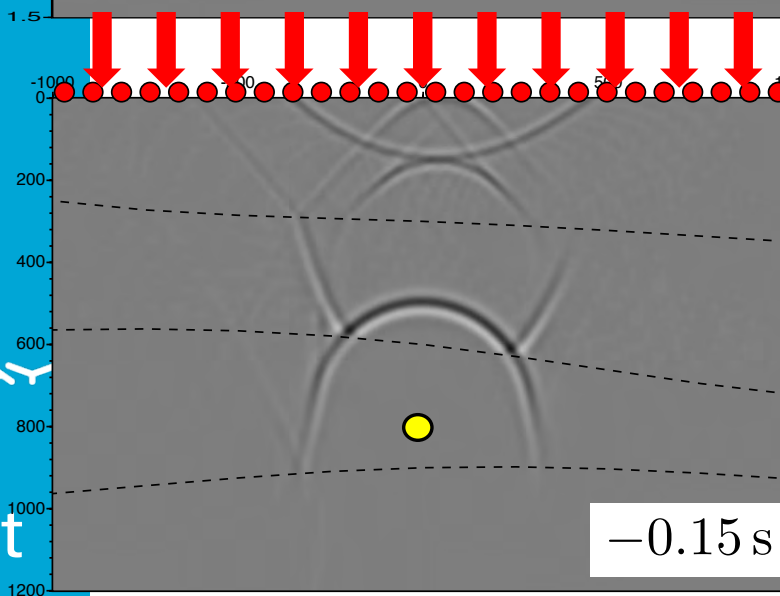
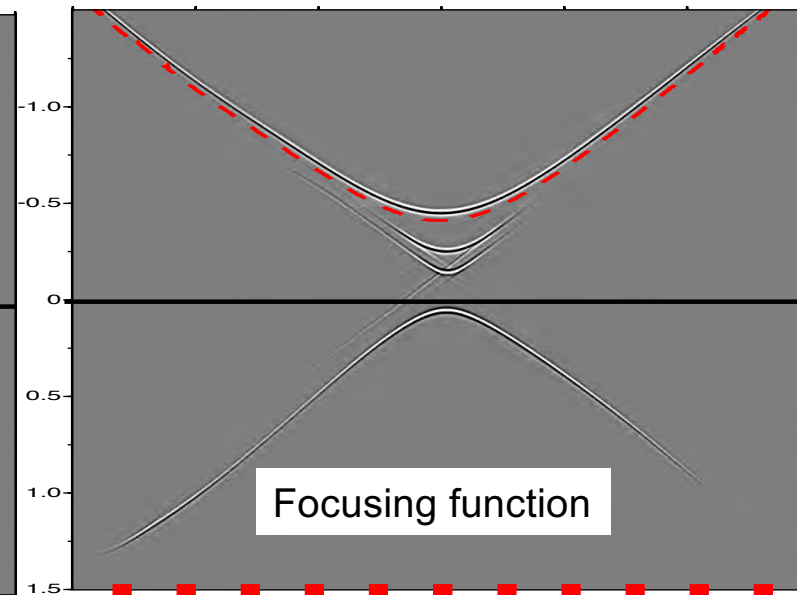
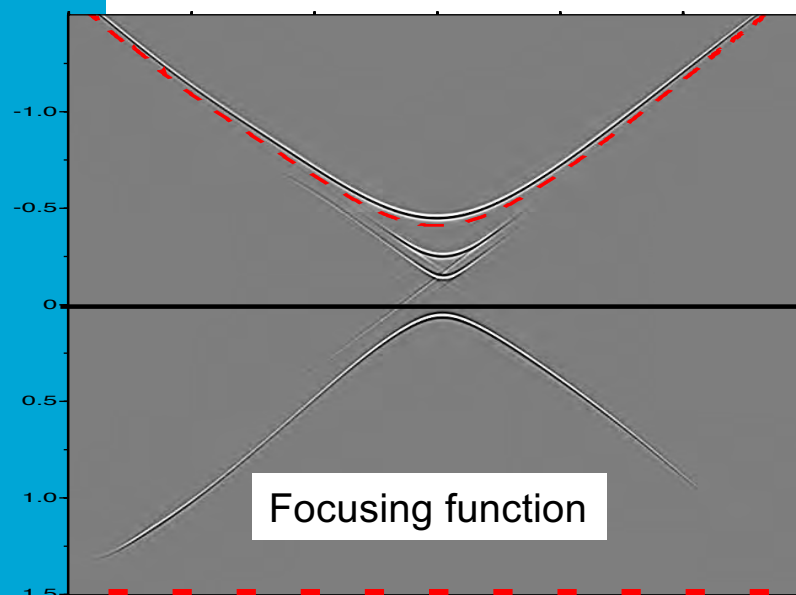


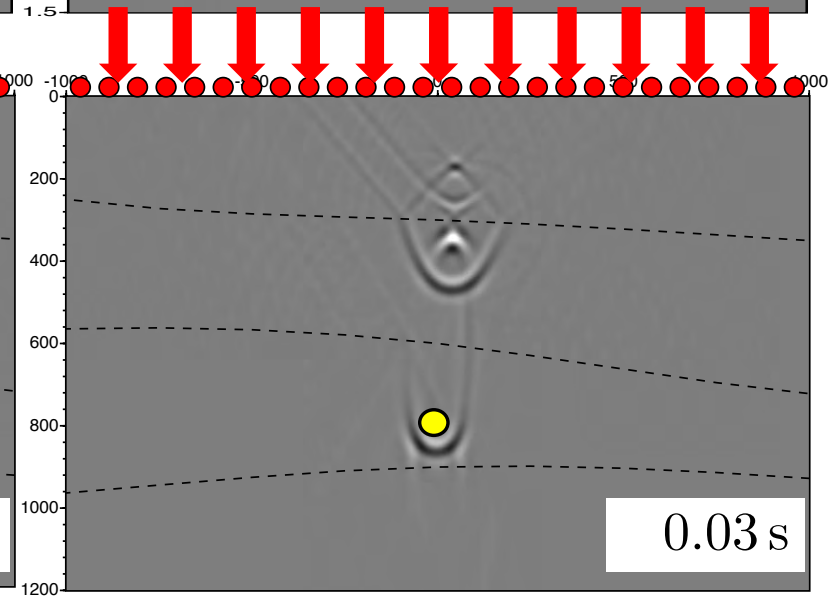
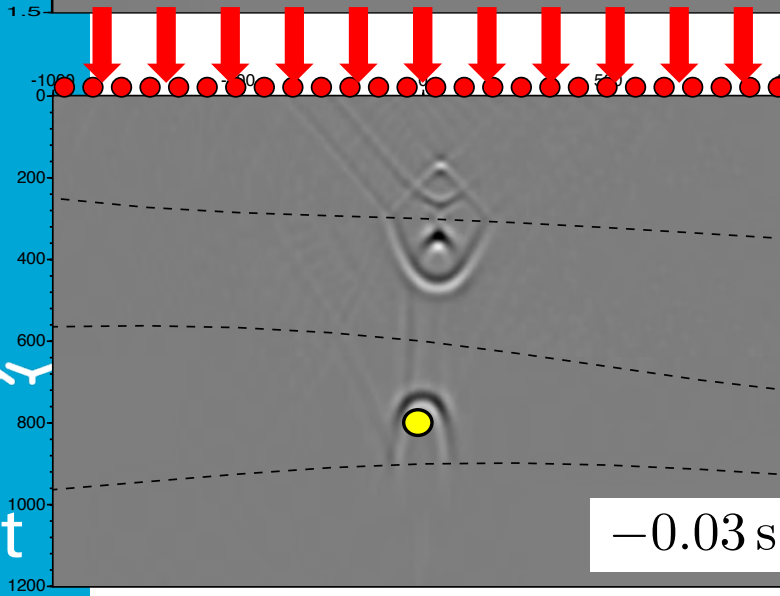
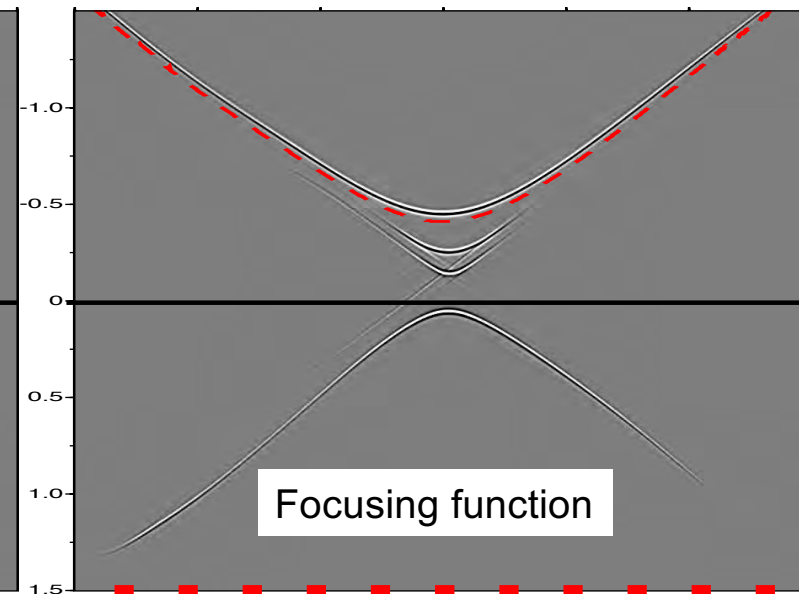
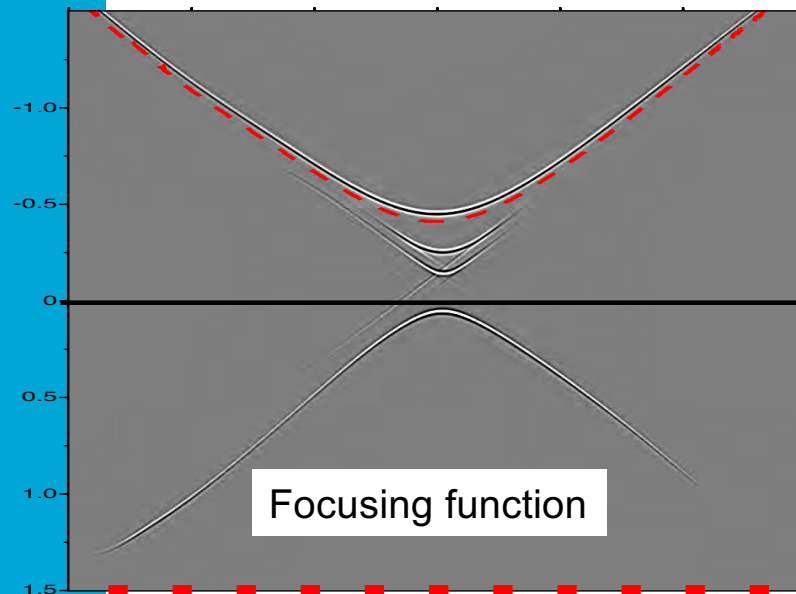
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * f(\mathbf{x}, \mathbf{x}_A, t) d^2\mathbf{x} \\ + \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, -t) * f(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$

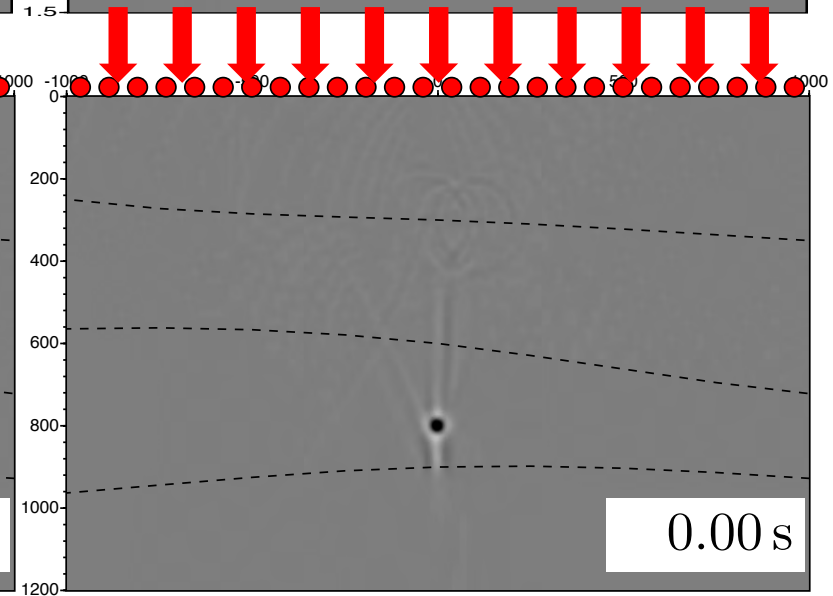
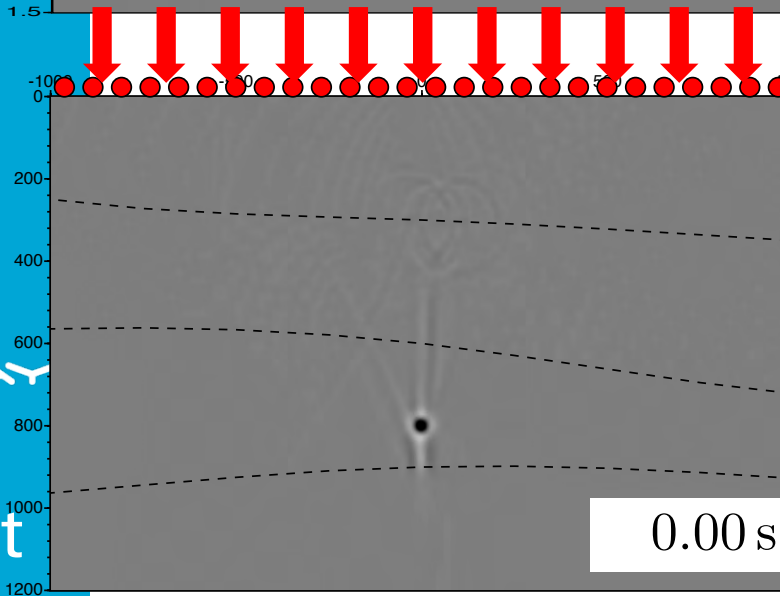
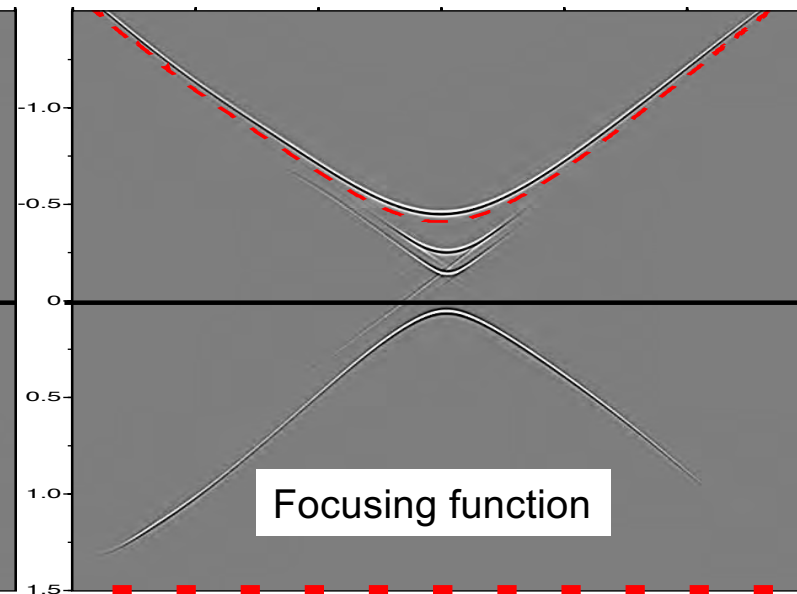
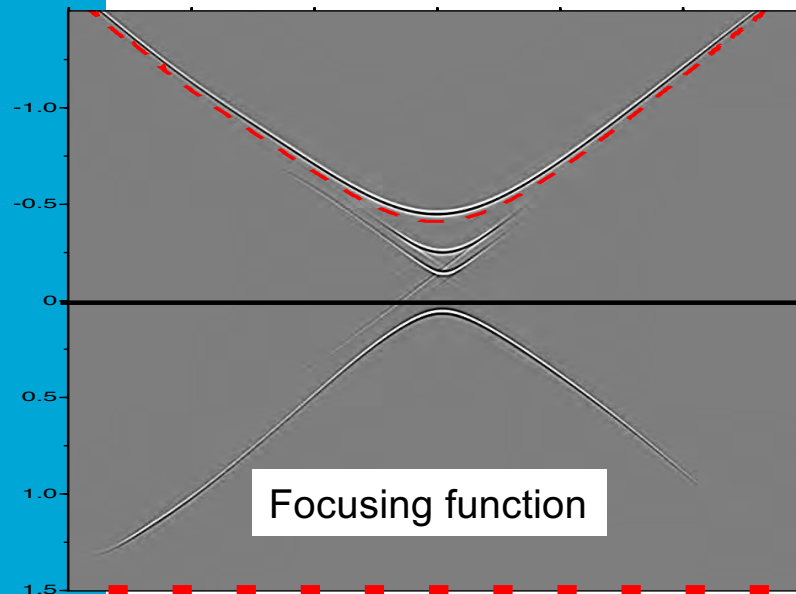


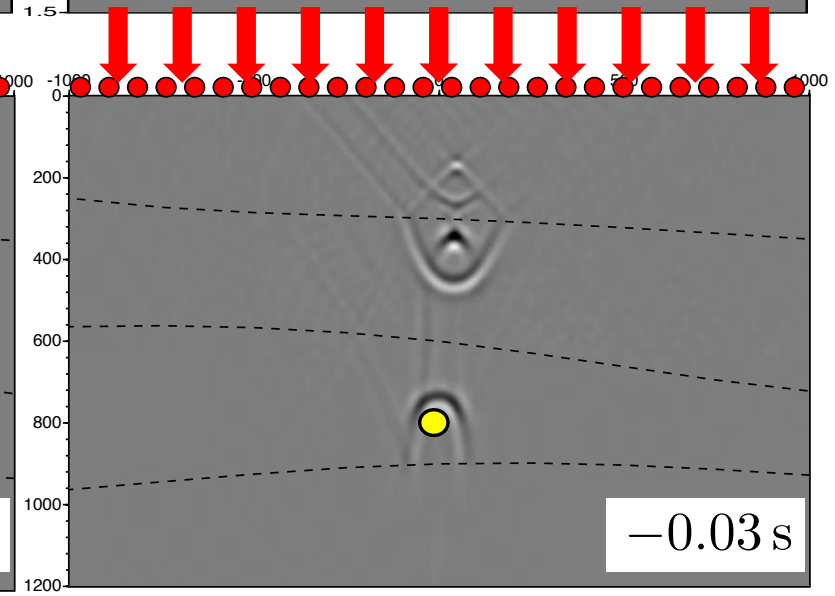
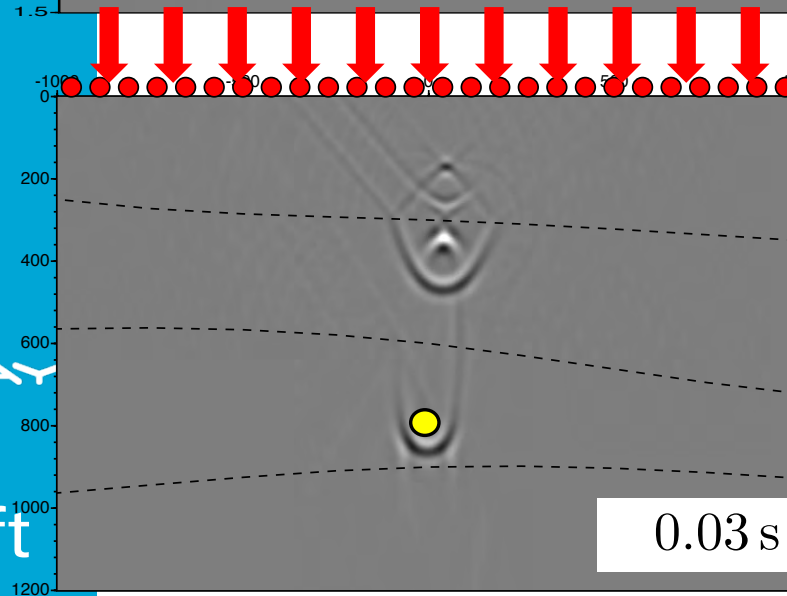
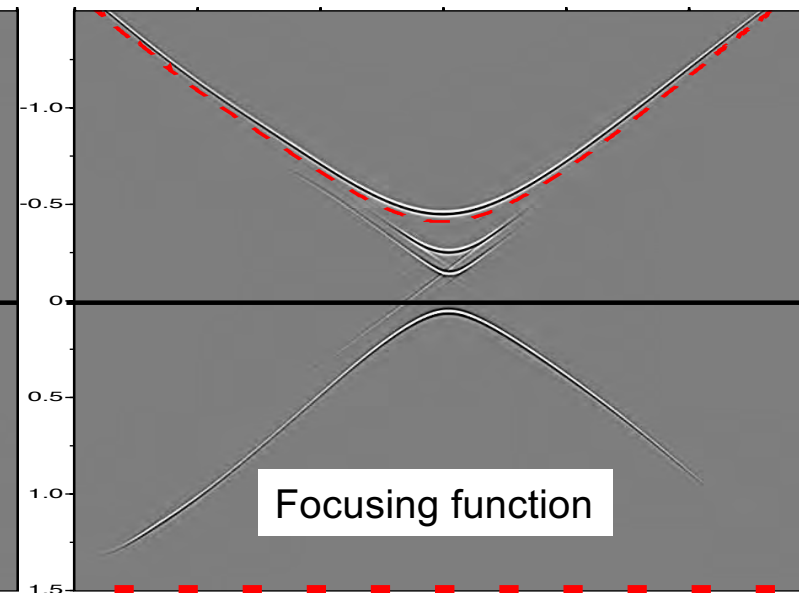
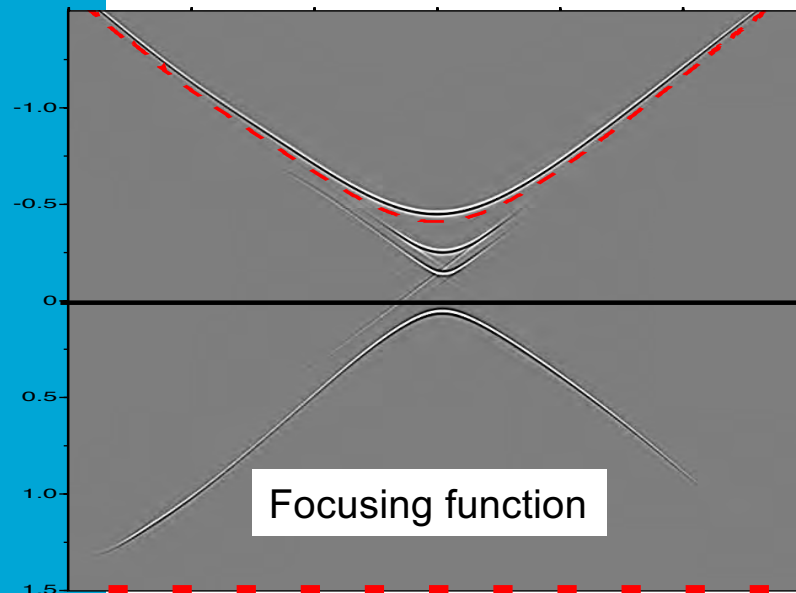
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * f(\mathbf{x}, \mathbf{x}_A, t) d^2\mathbf{x} \\ + \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, -t) * f(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$

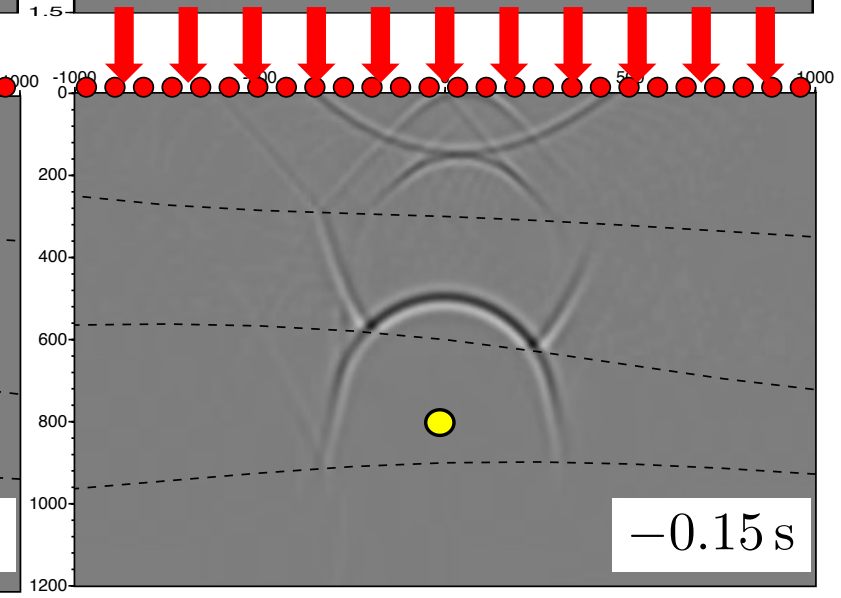
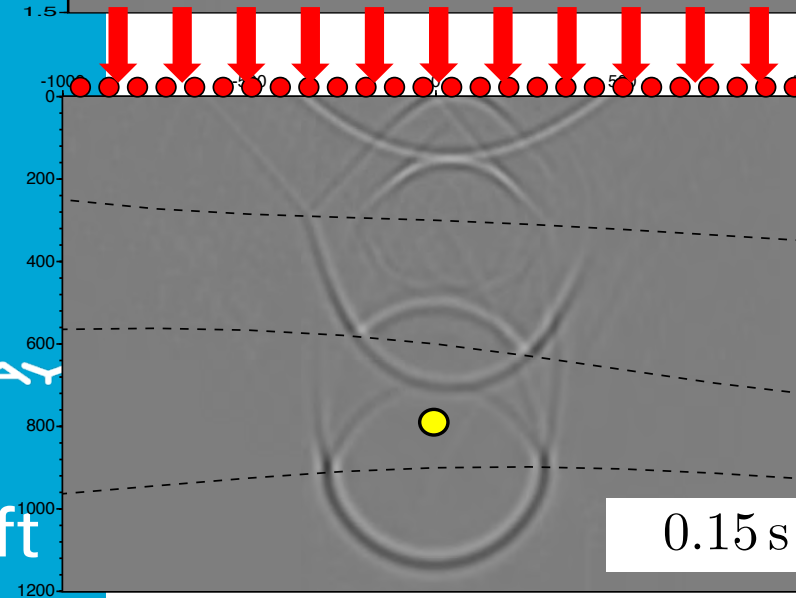
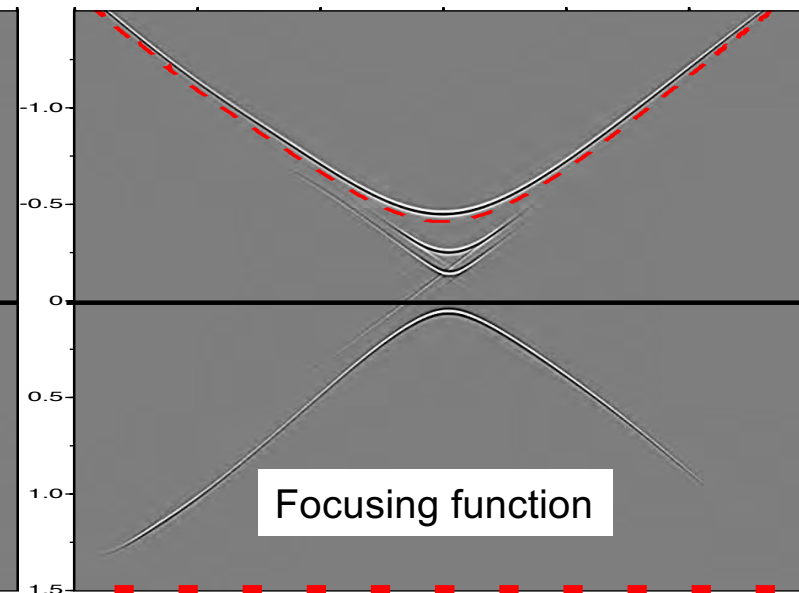
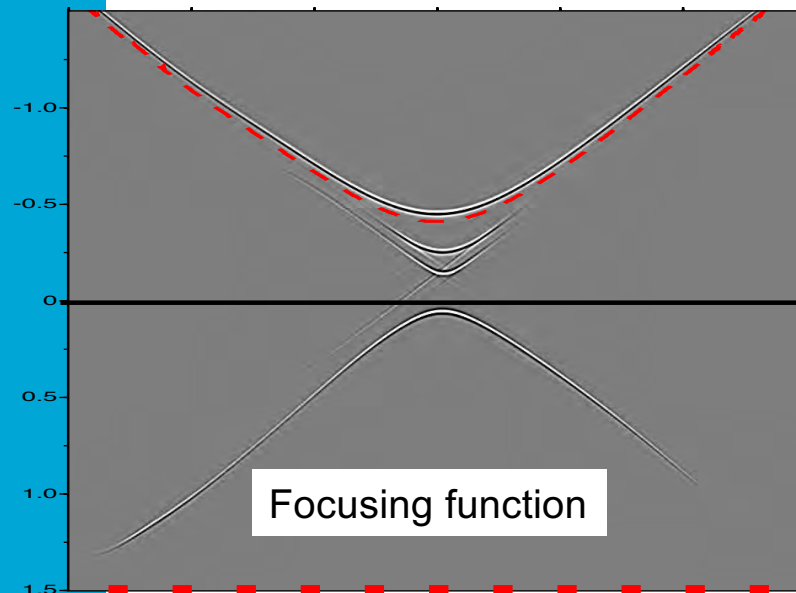


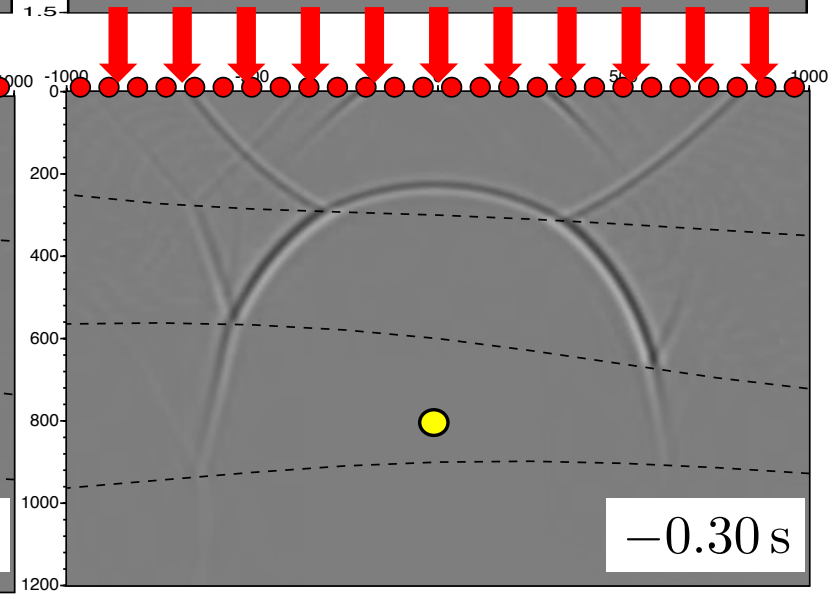
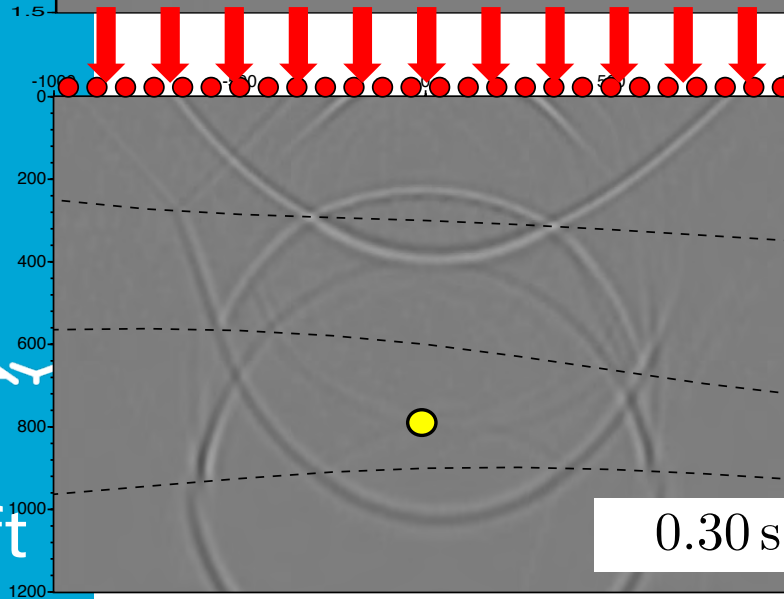
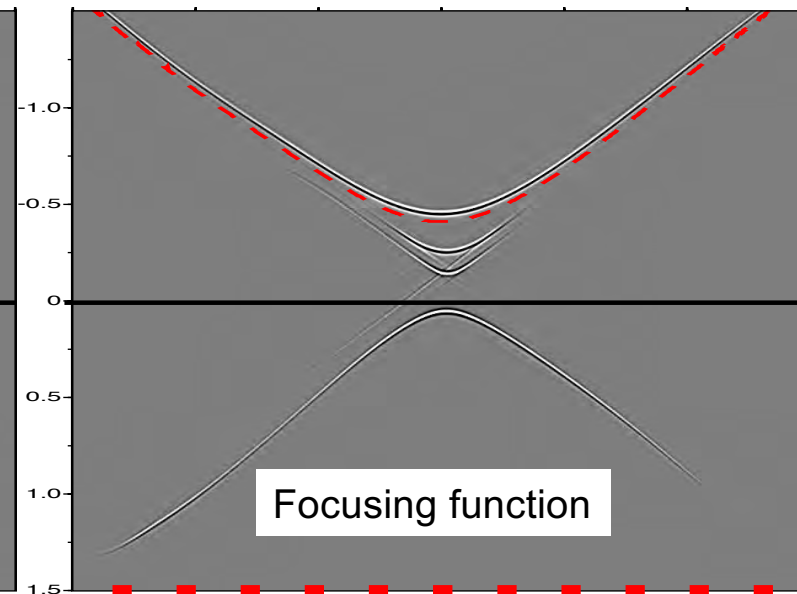
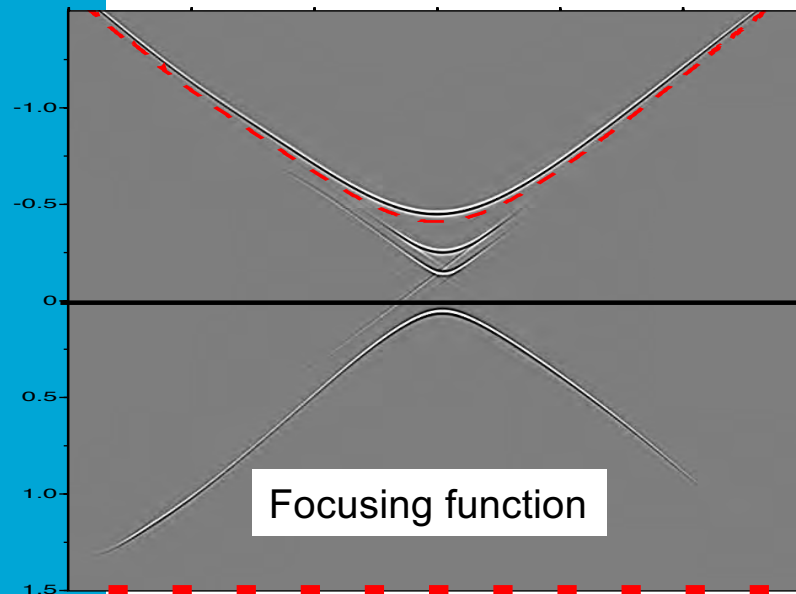


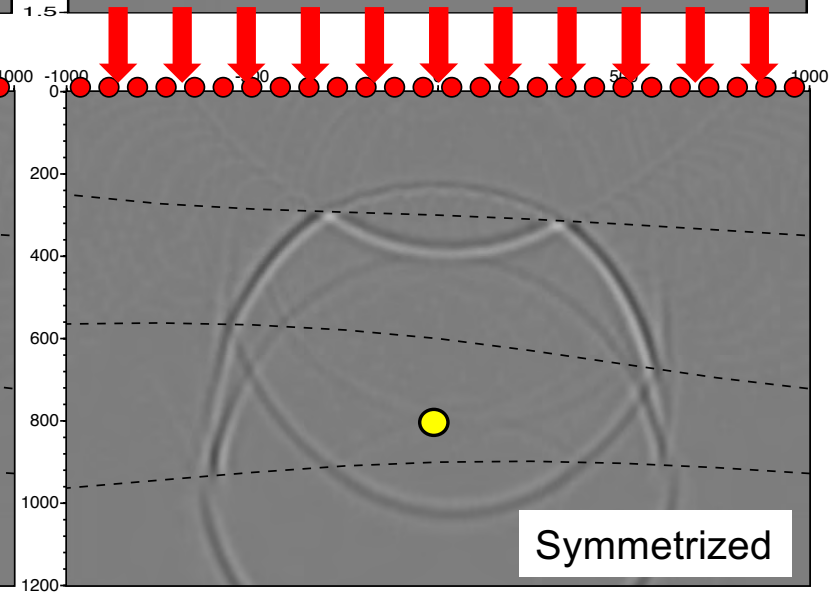
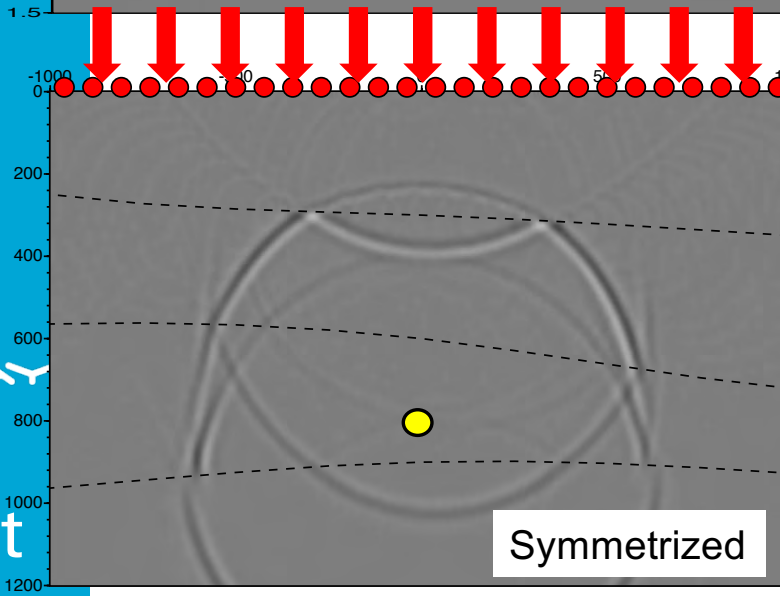
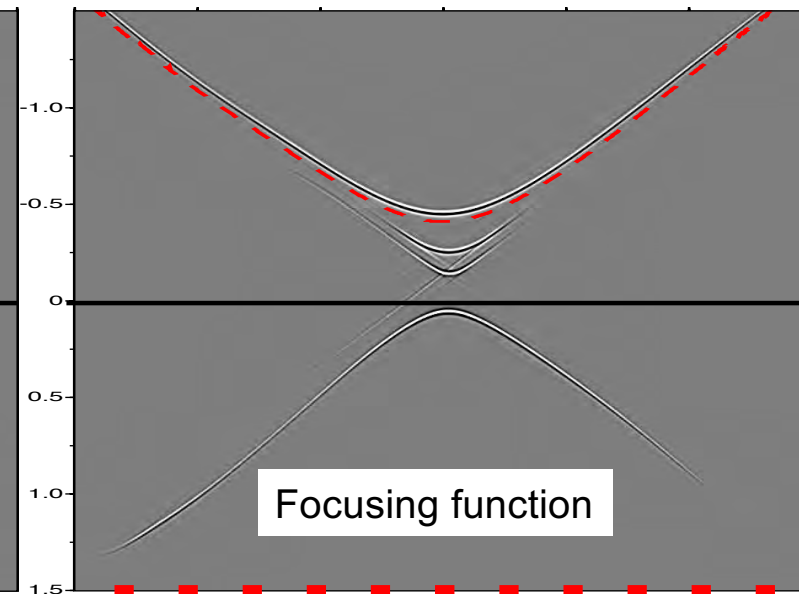
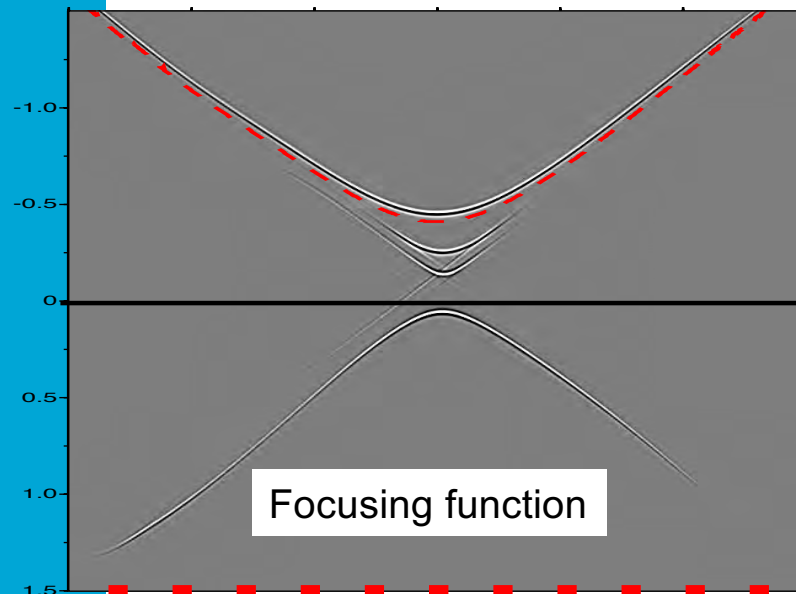


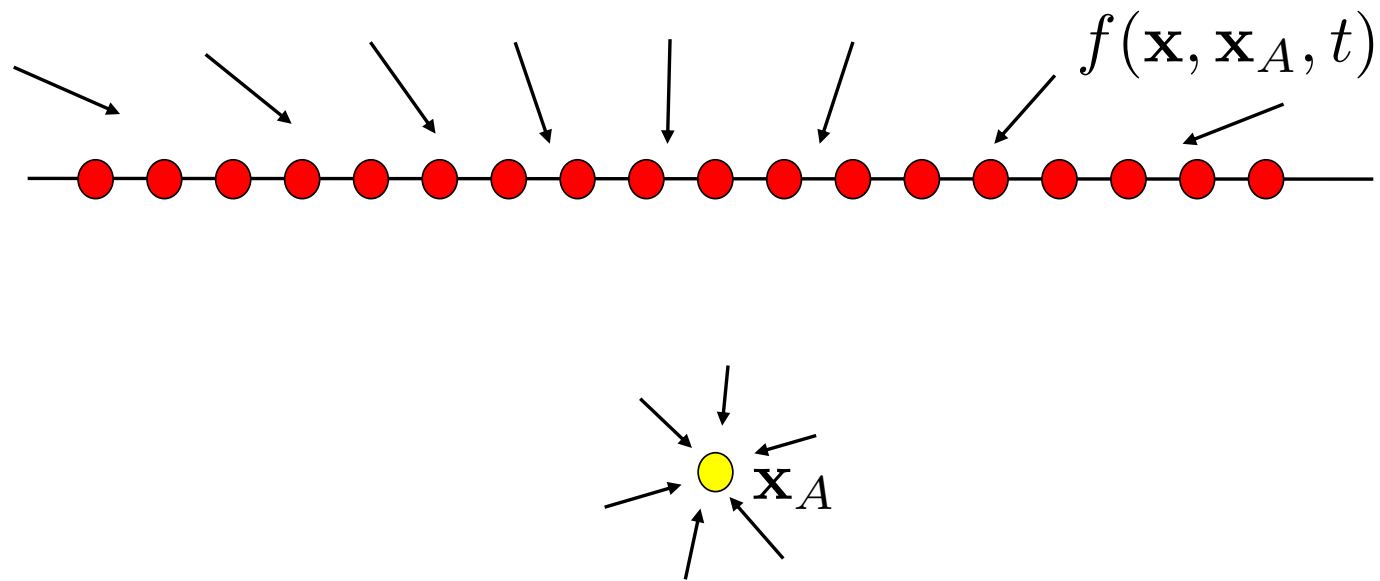




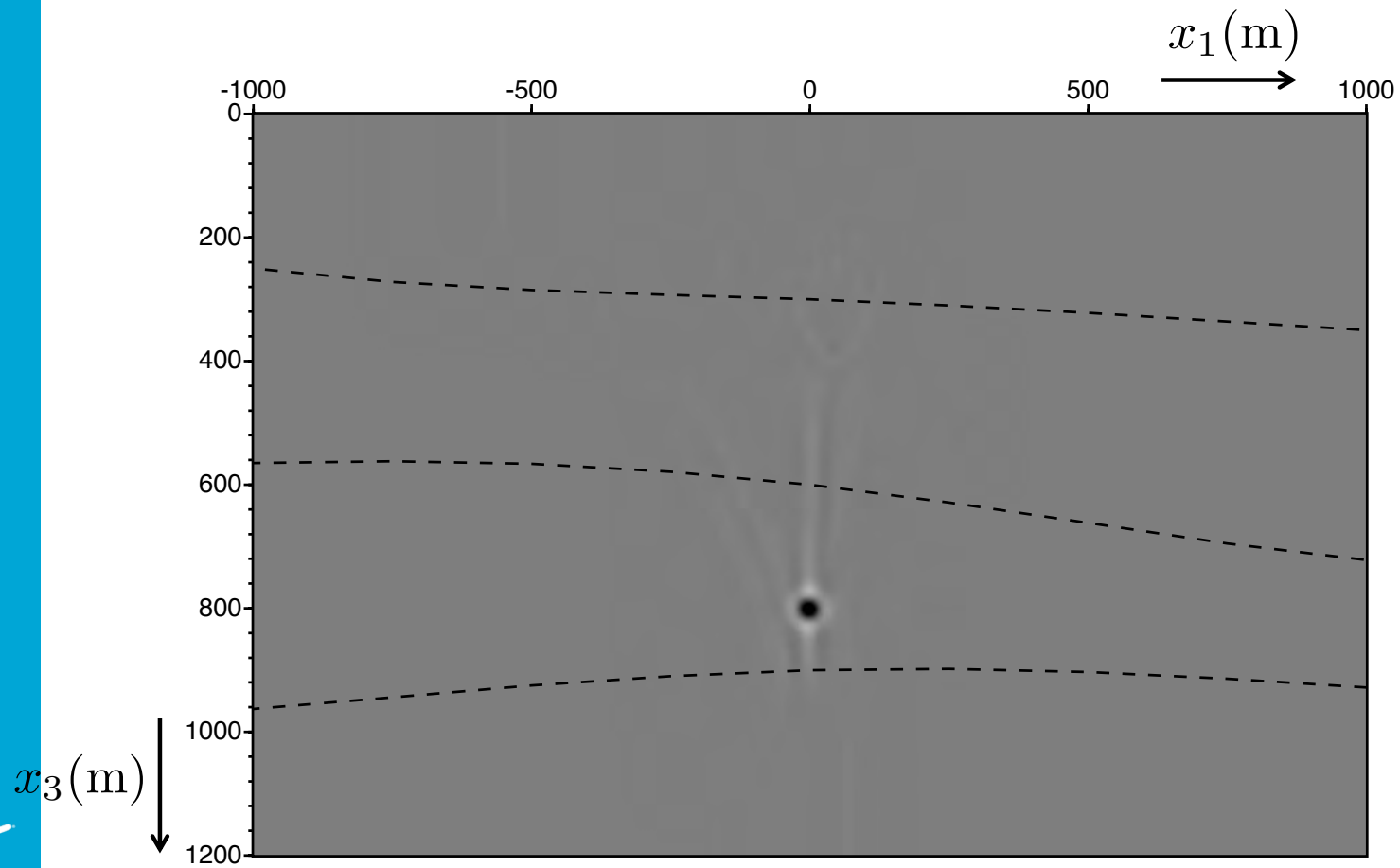


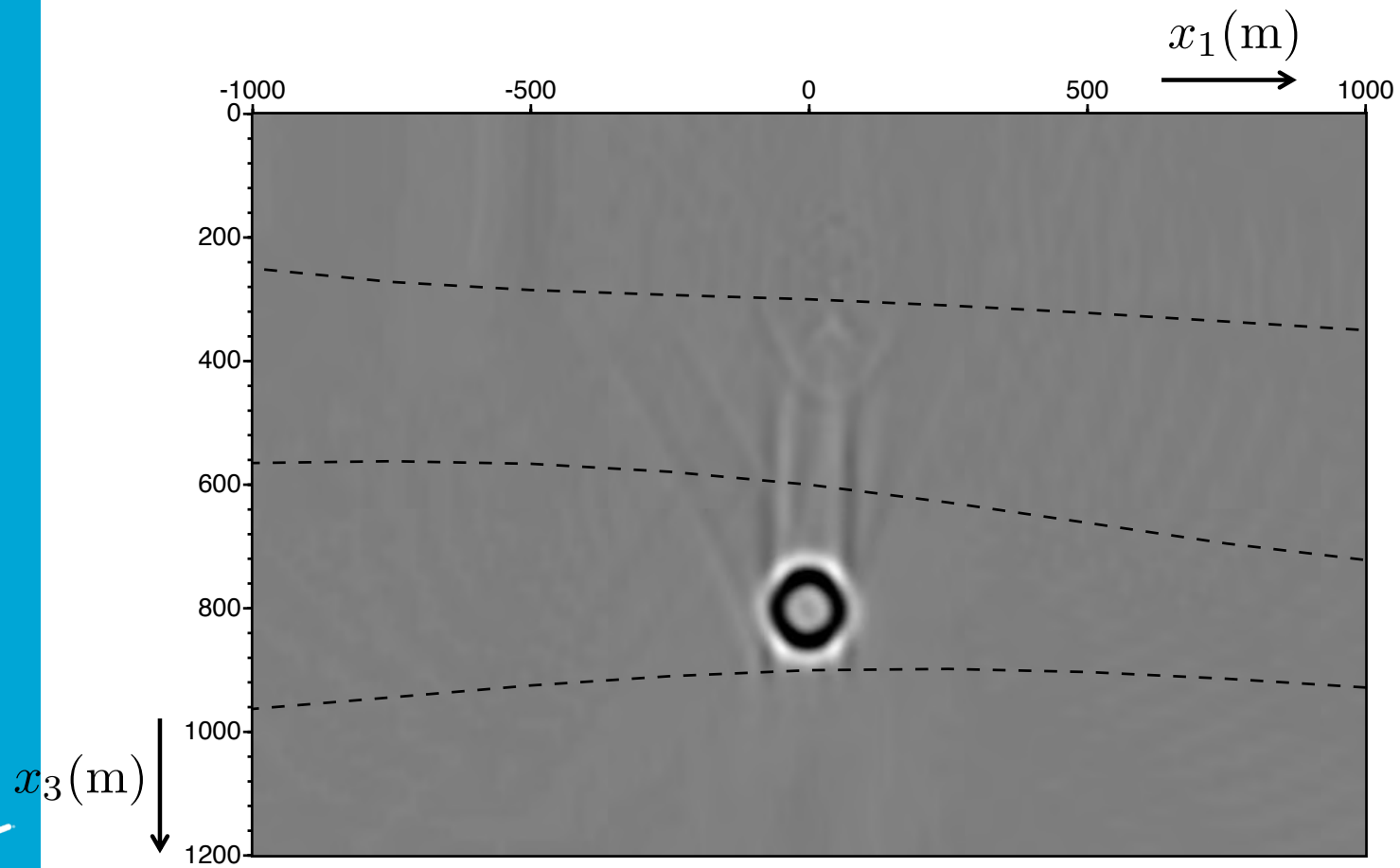


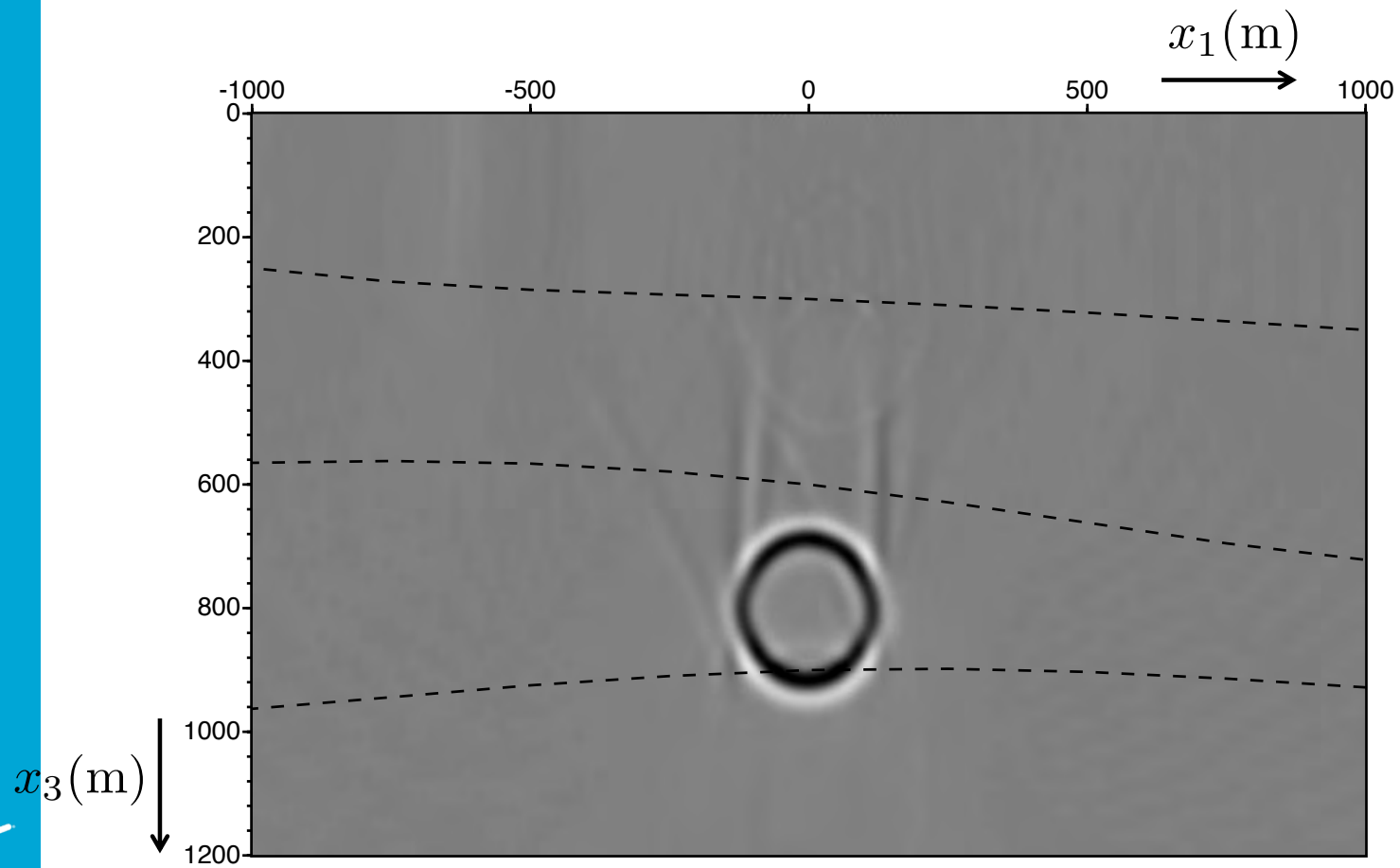


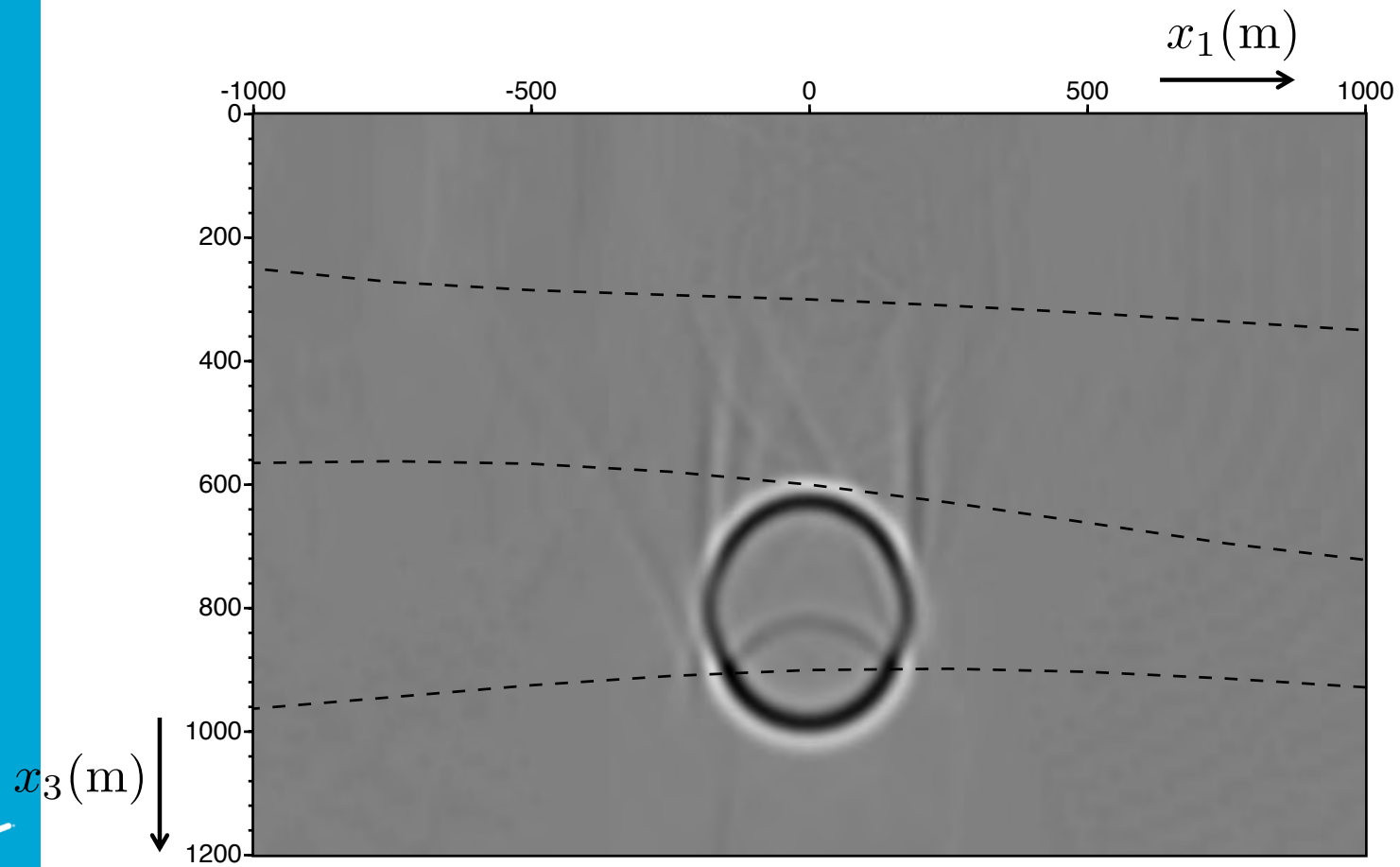


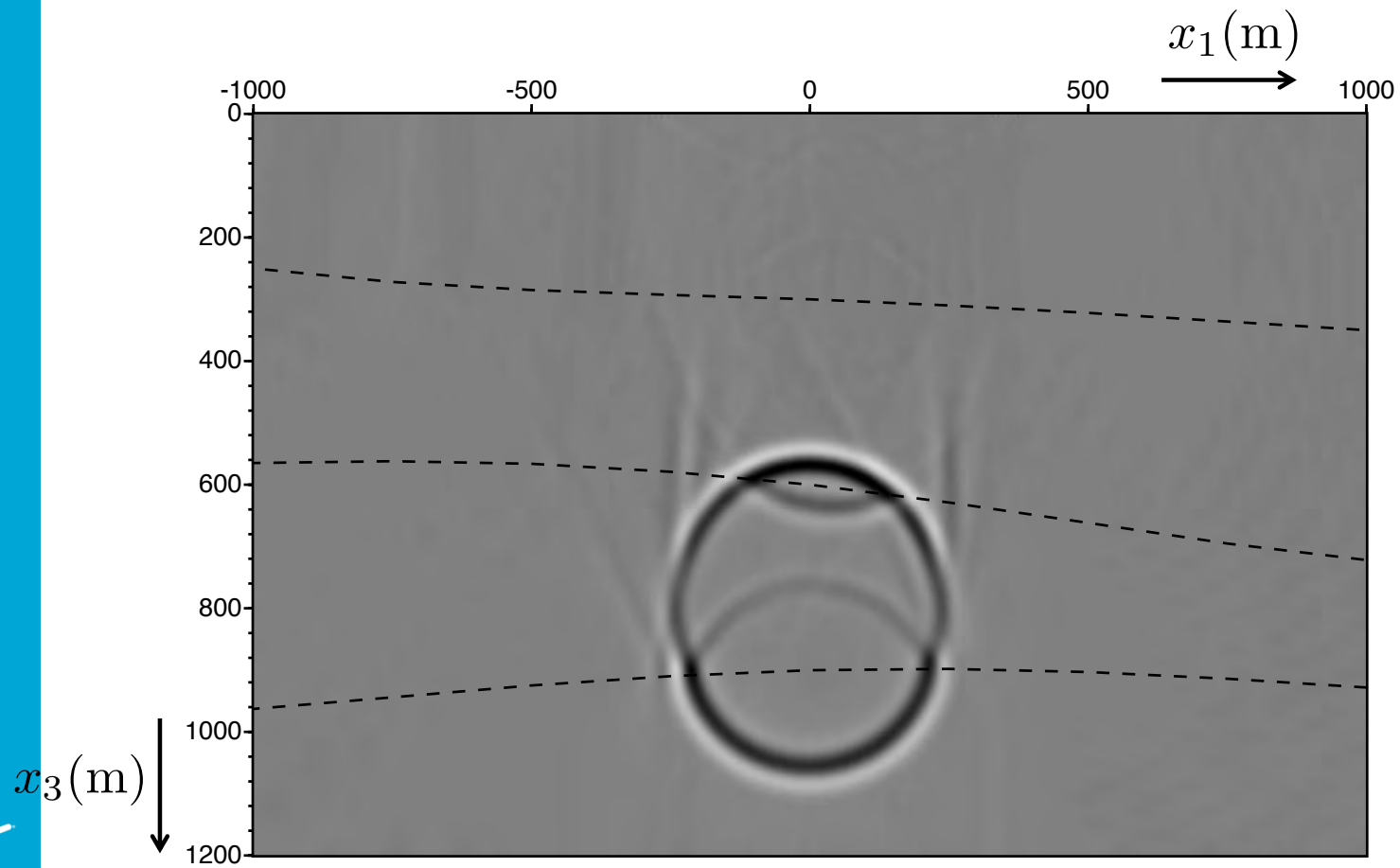
$$G(\mathbf{x}_B, \mathbf{x}_A, t) + G(\mathbf{x}_B, \mathbf{x}_A, -t) \propto \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, t) * f(\mathbf{x}, \mathbf{x}_A, t) d^2\mathbf{x} \\ + \int_{\partial\mathbb{D}} G(\mathbf{x}_B, \mathbf{x}, -t) * f(\mathbf{x}, \mathbf{x}_A, -t) d^2\mathbf{x}$$

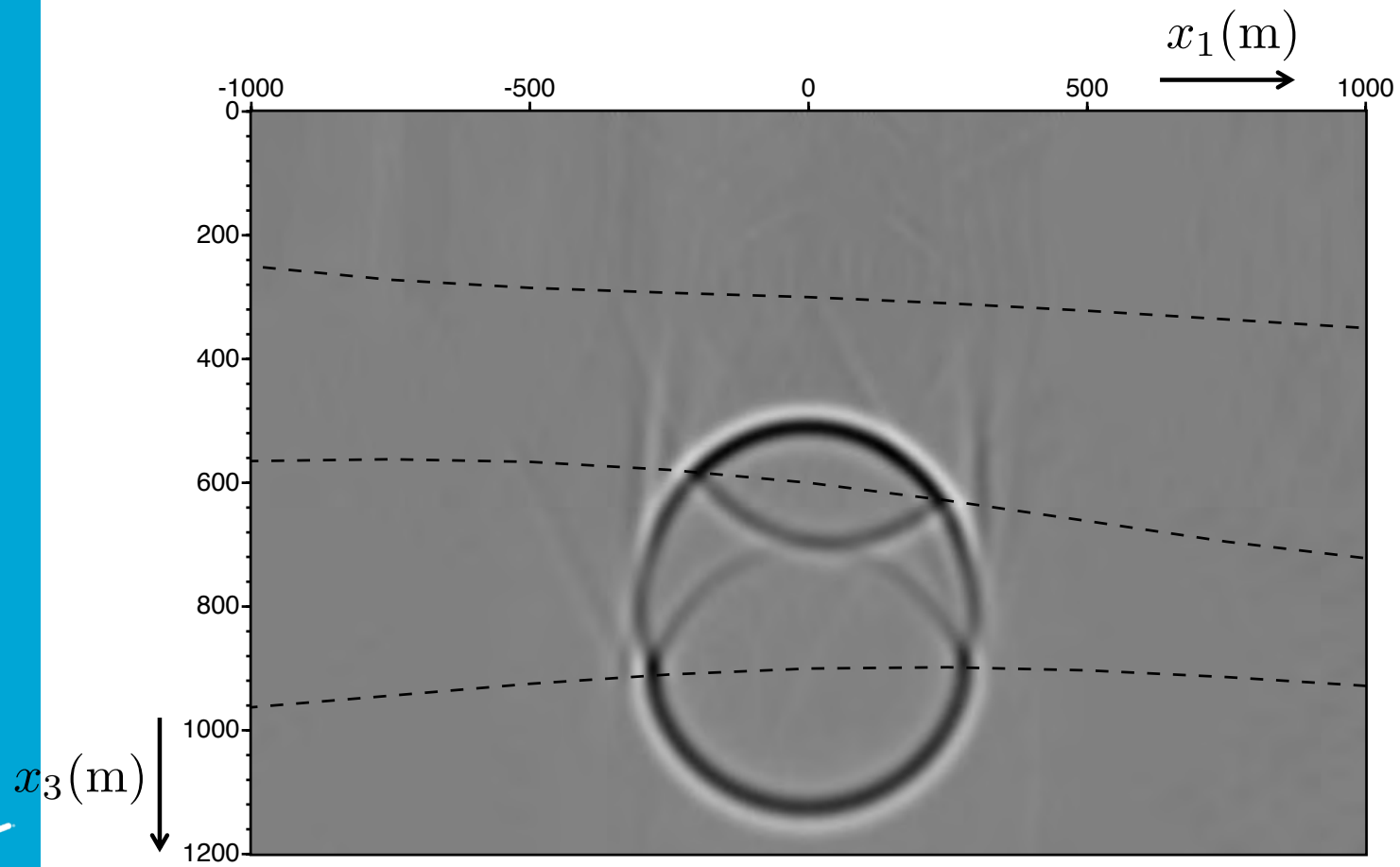


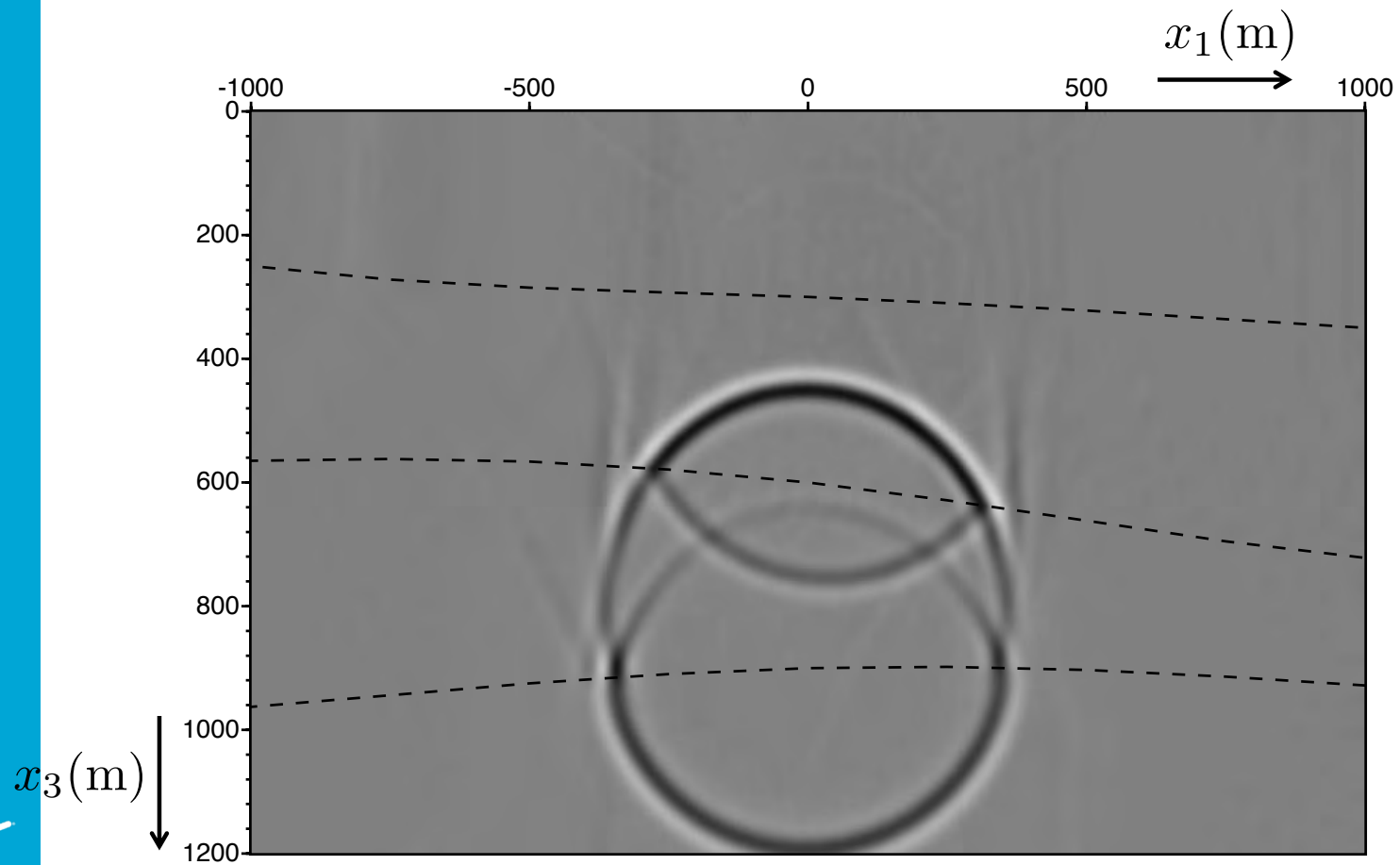


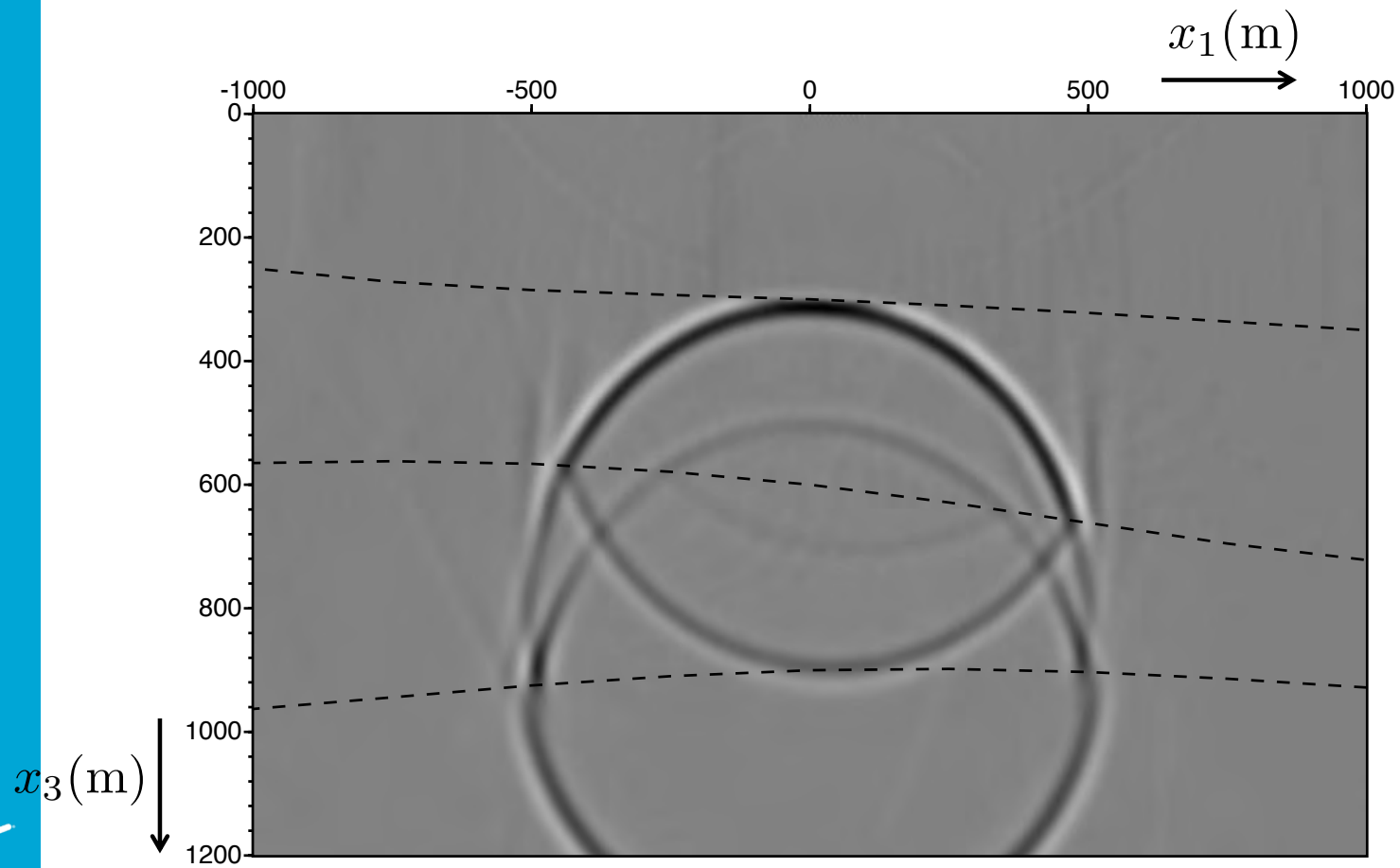




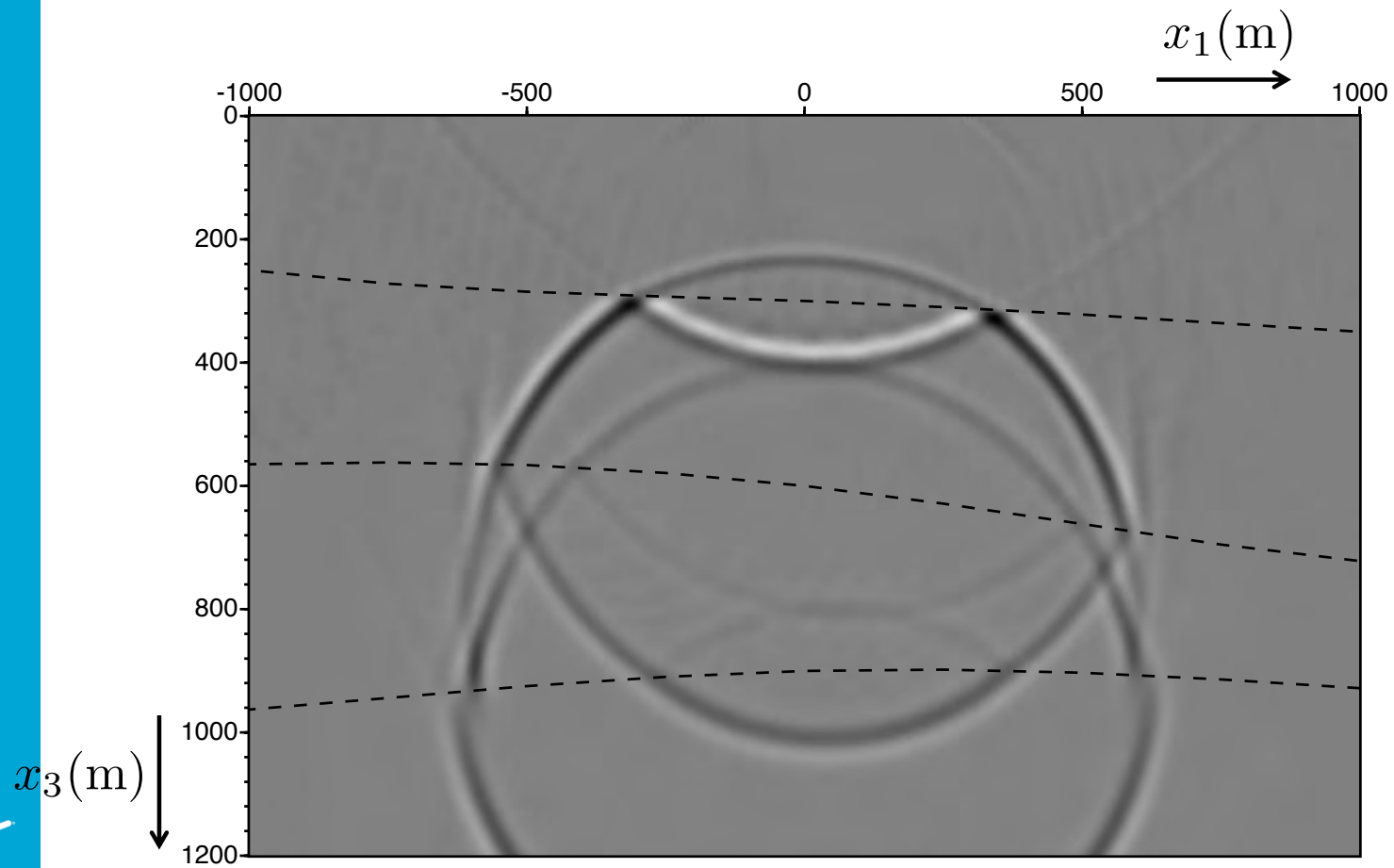




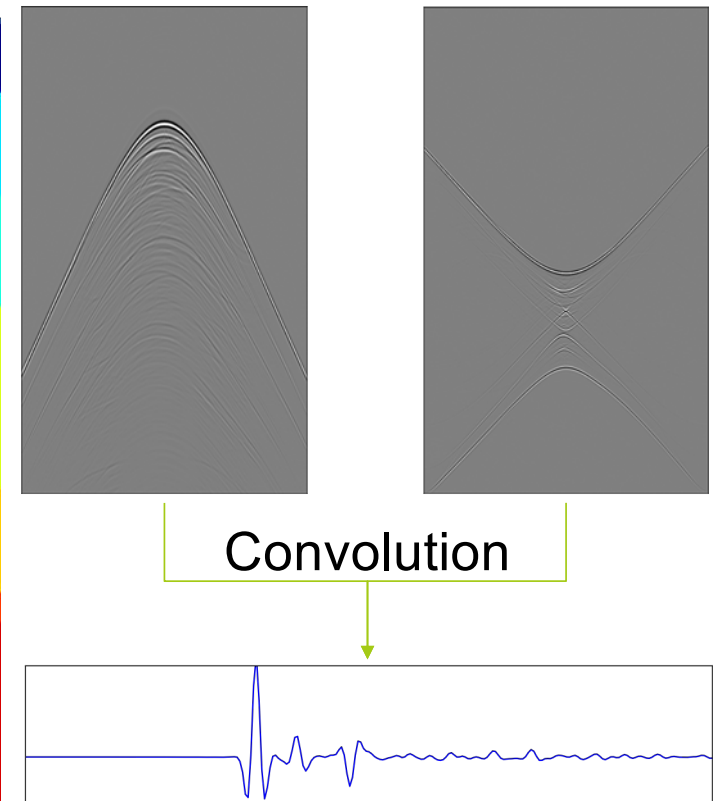
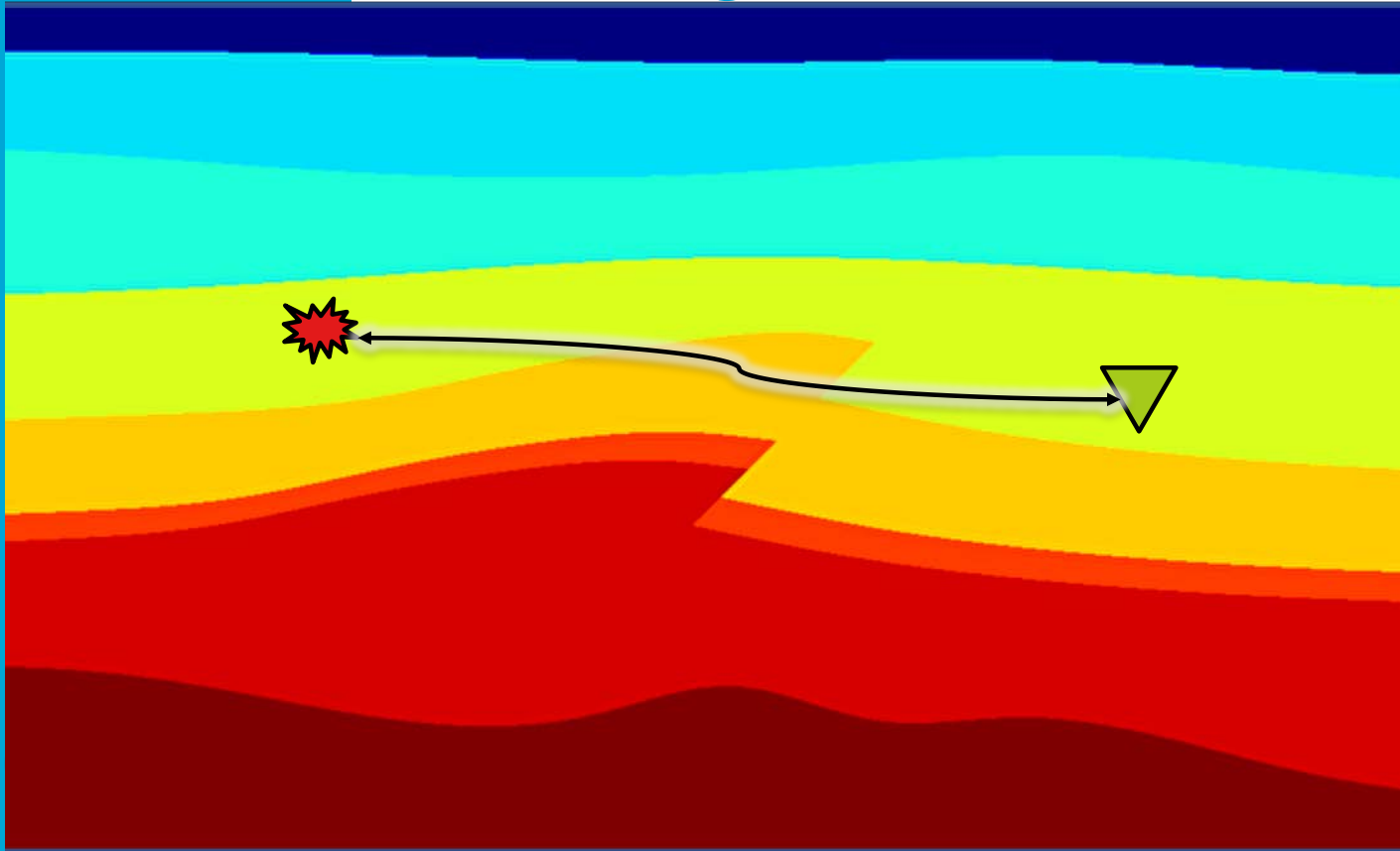




$t = 0.25\text{s}$



Homogeneous Green's function



Homogeneous Green's function

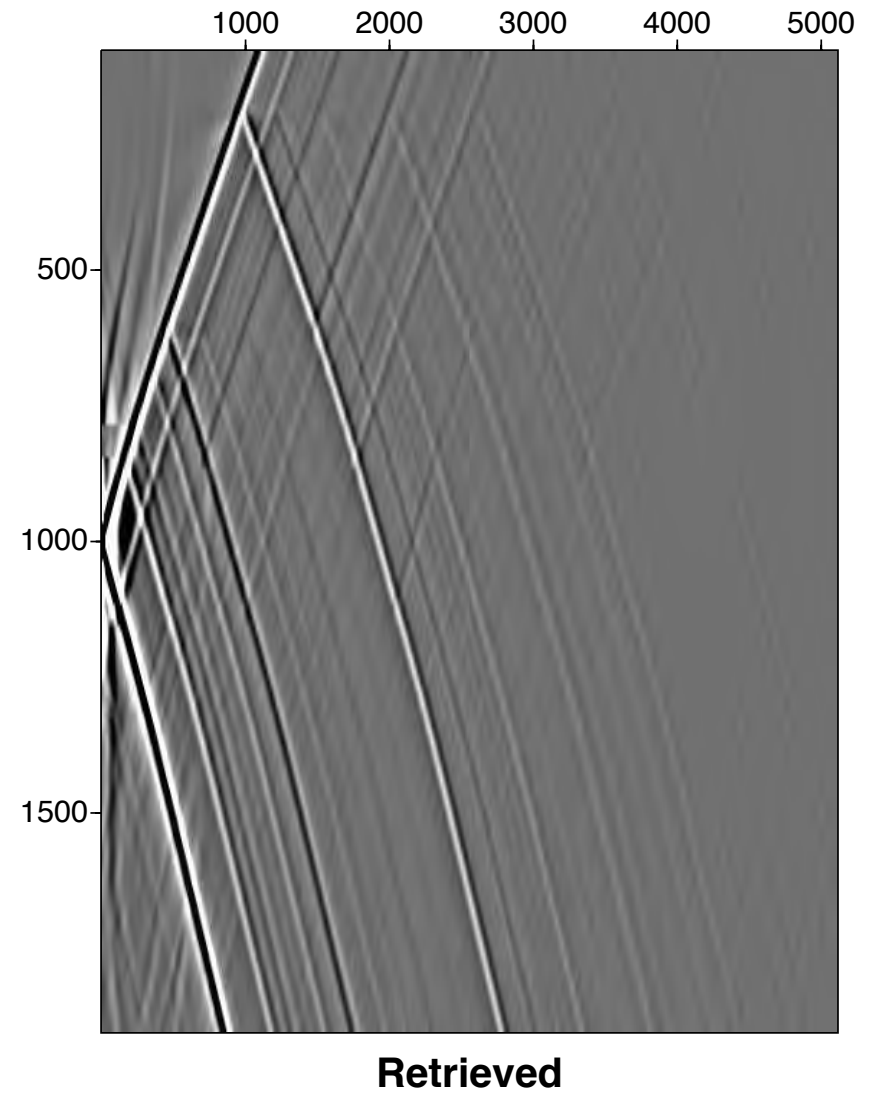
- README.5
- Generate snapshots for homogeneous Green's function



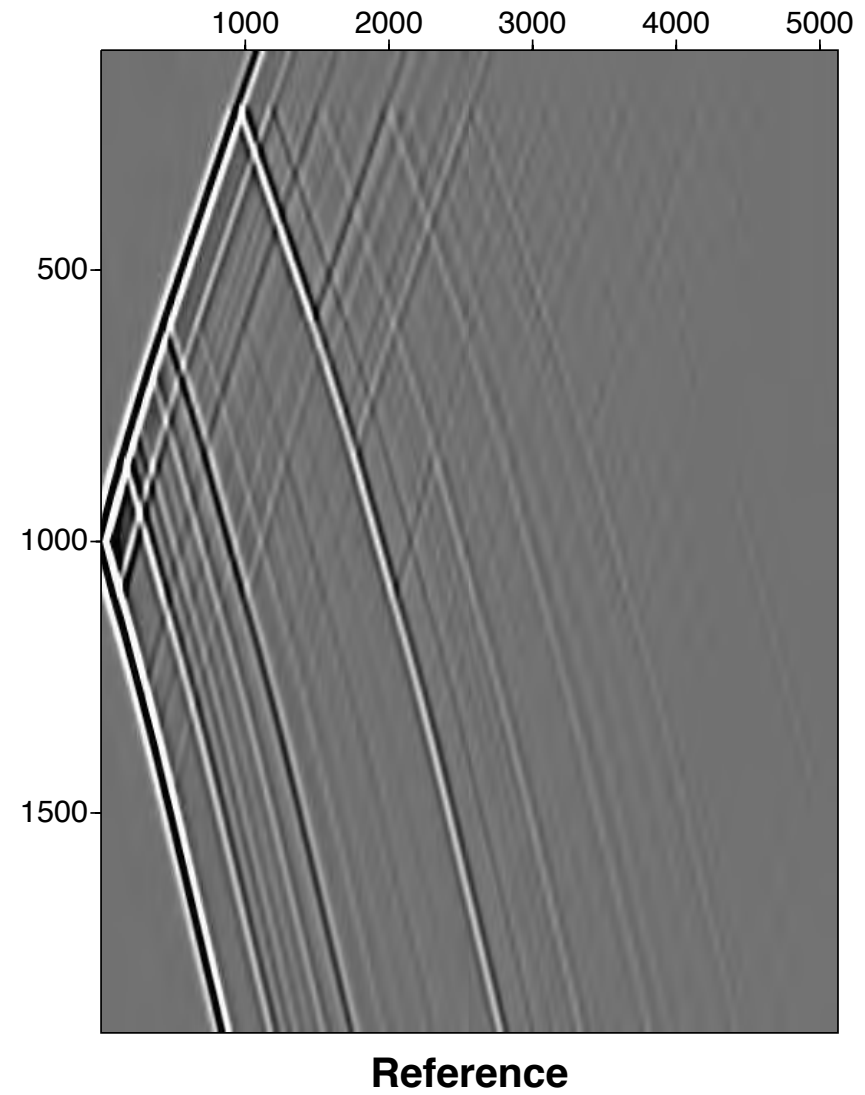
20 min.

Reference: Brackenhoff, J., Thorbecke, J., and Wapenaar, K.: *Monitoring induced distributed double-couple sources using Marchenko-based virtual receivers*, Solid Earth Discuss., <https://doi.org/10.5194/se-2018-142>, in review, 2019.

demo/twoD



demo/twoD



Group Discussion



SAGA data examples

- Homogeneous Green's function movie directed by Joeri.
- Primaries extraction featured by Lele



Reference: Brackenhoff, J., Thorbecke, J., and Wapenaar, K.: *Monitoring induced distributed double-couple sources using Marchenko-based virtual receivers*, Solid Earth Discuss., <https://doi.org/10.5194/se-2018-142>, in review, 2019.

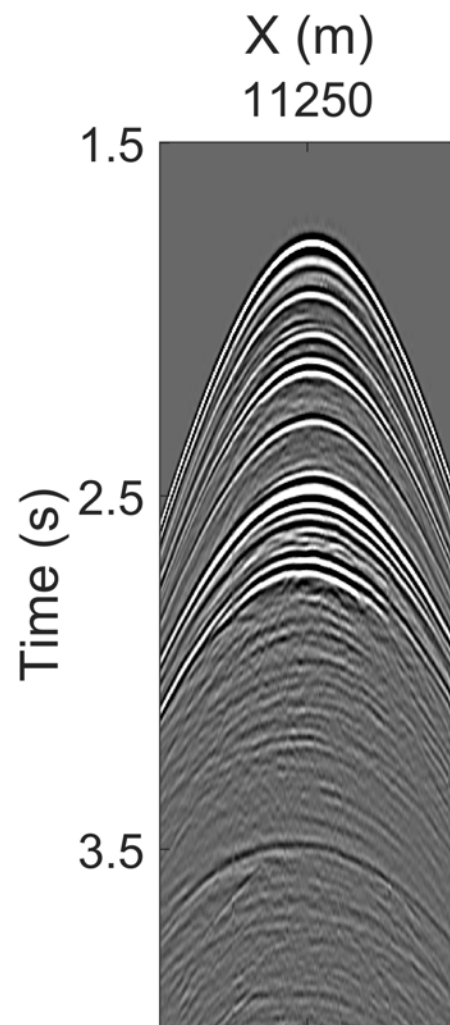
Field data

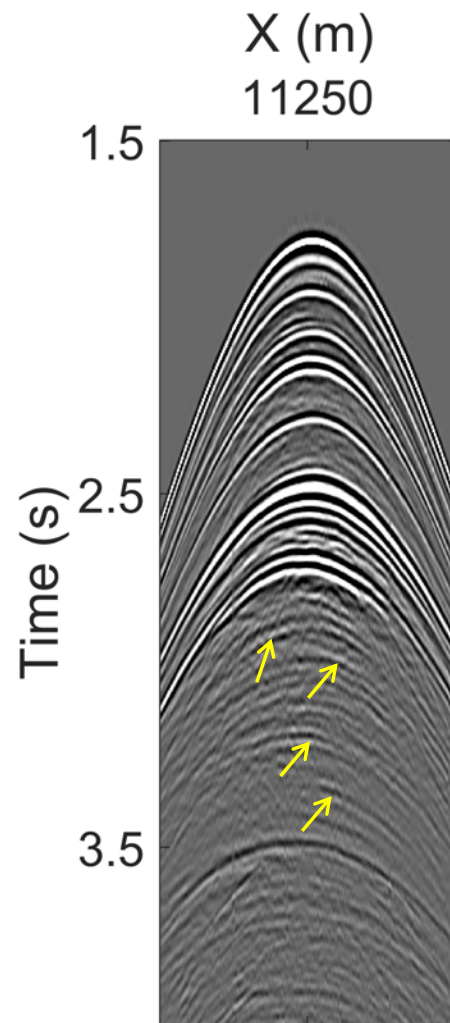
- Data recorded in a marine environment
- Acoustic scheme applied
- No free-surface multiples included
 - EPSI: wavelet deconvolved
- Q-compensation and scalar correction applied

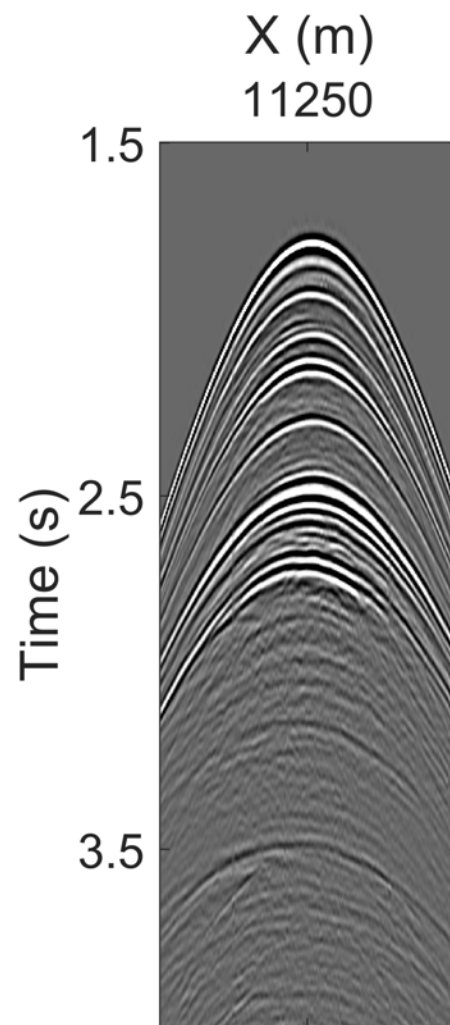


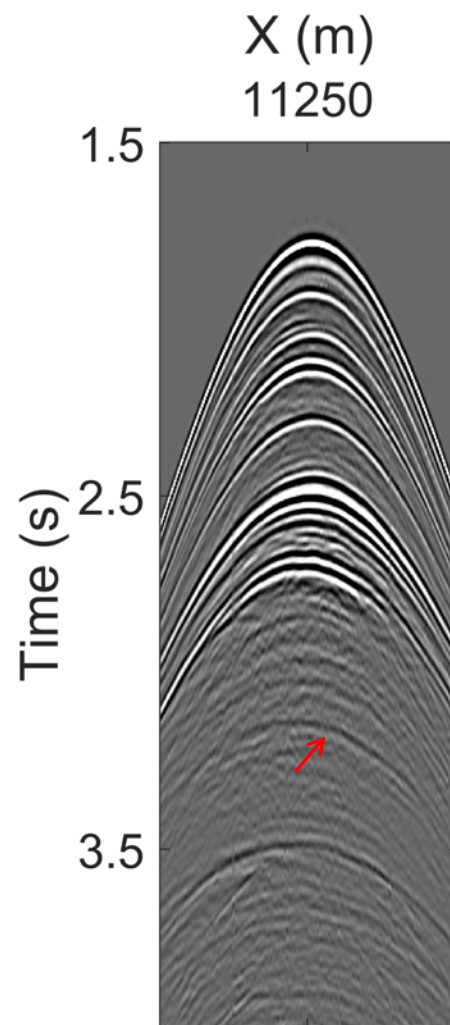
Primaries

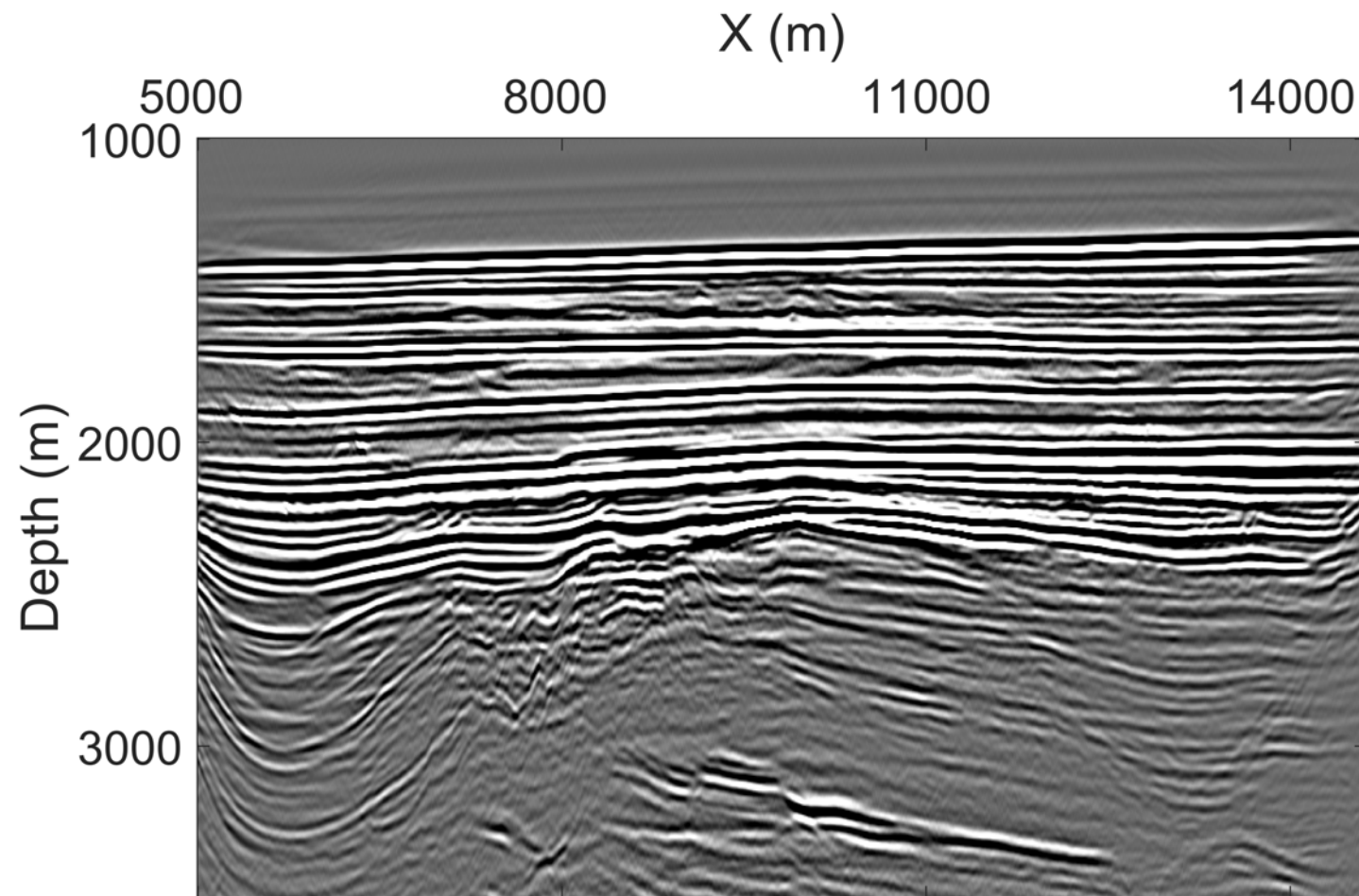


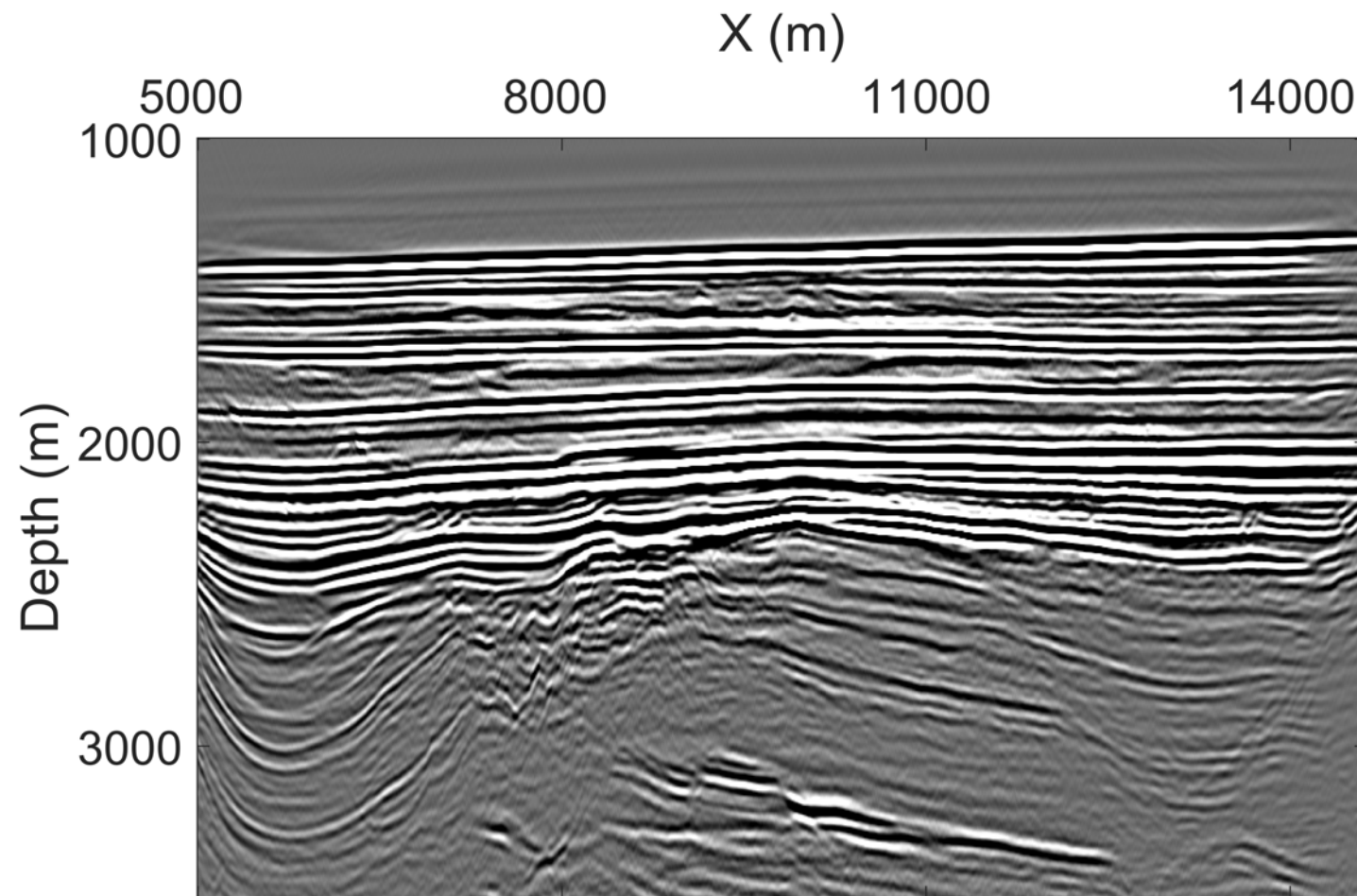


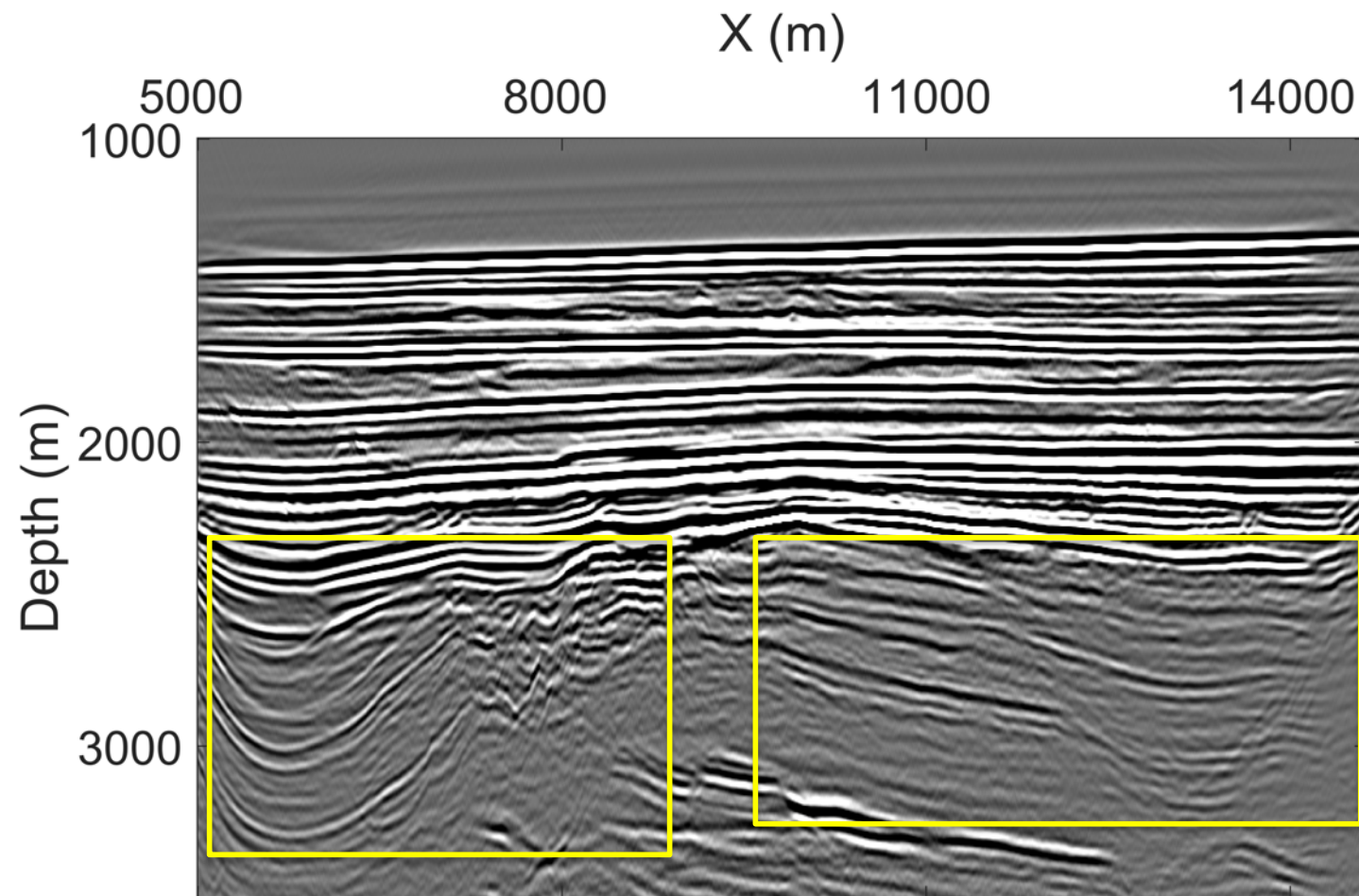


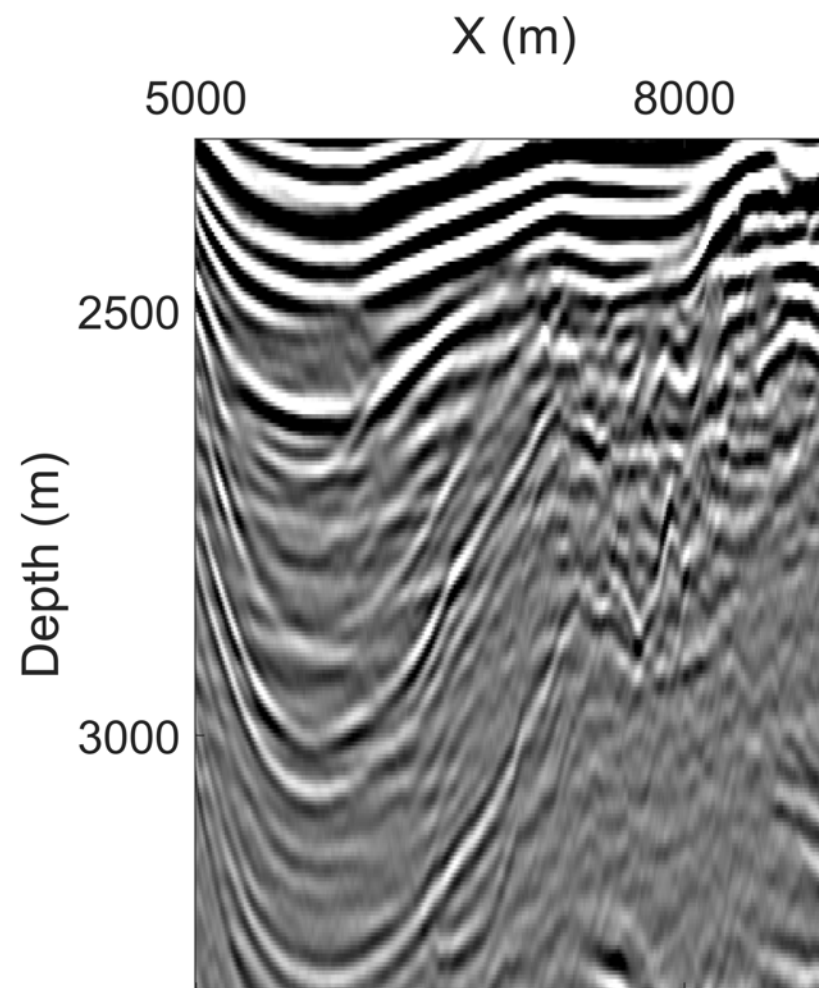


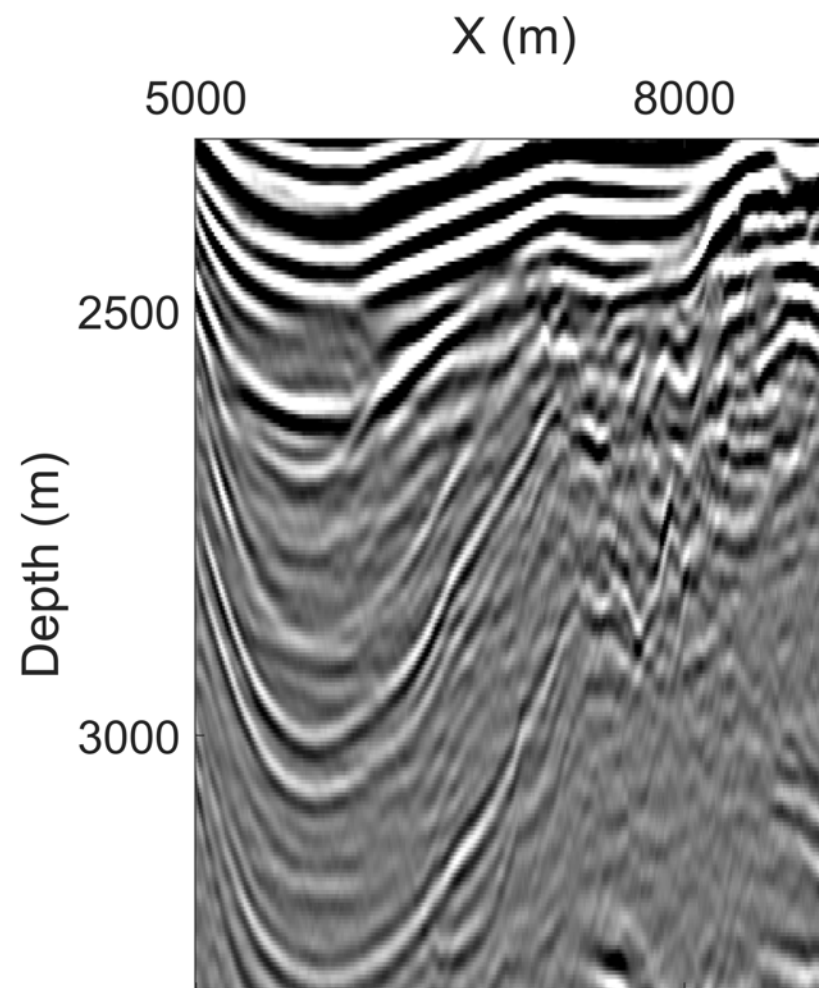


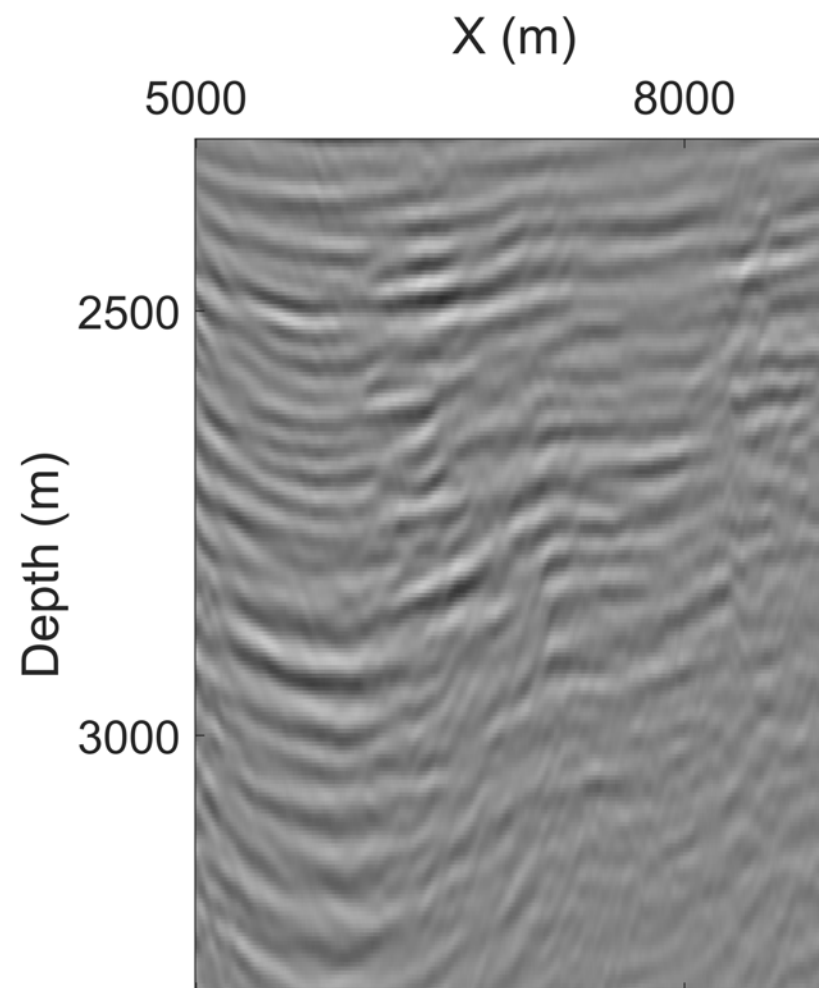


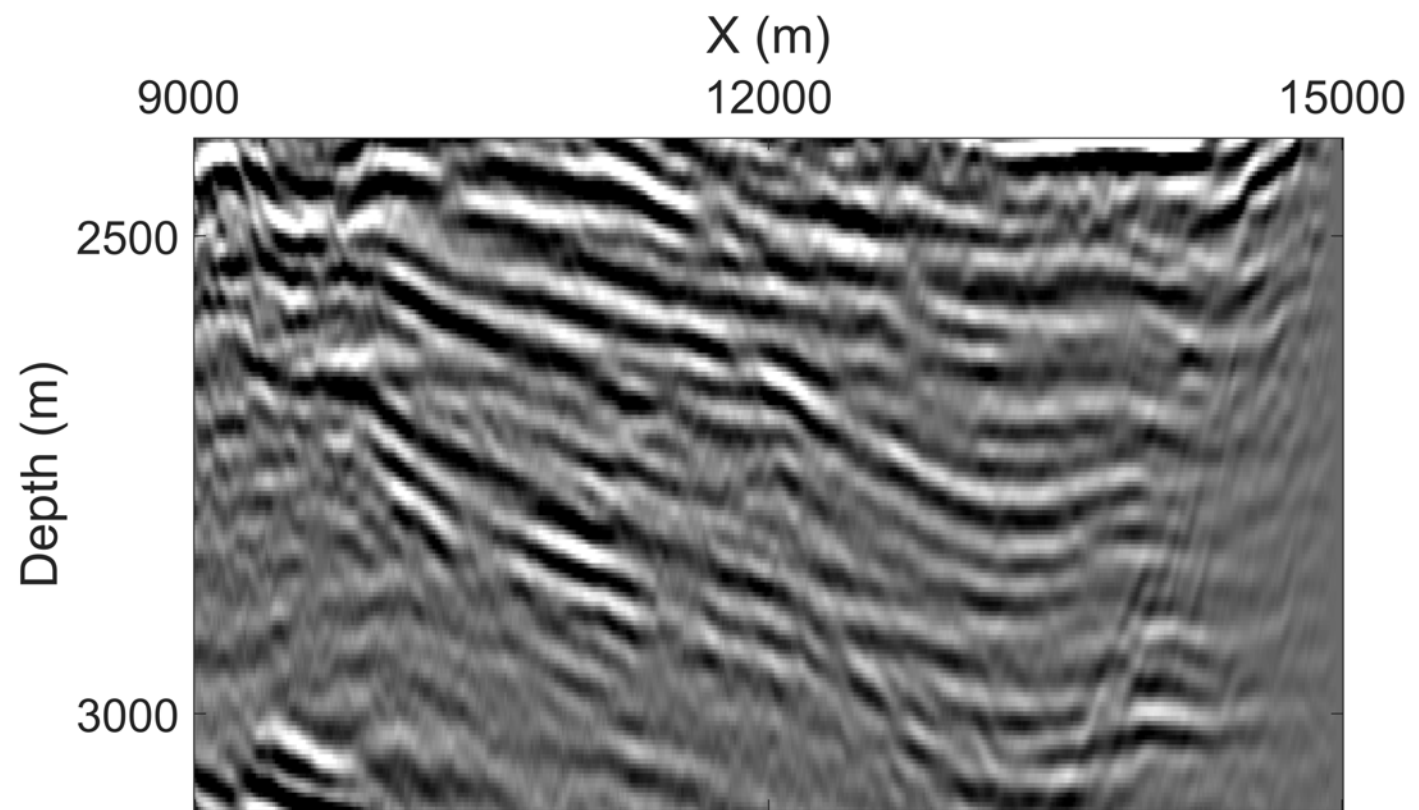


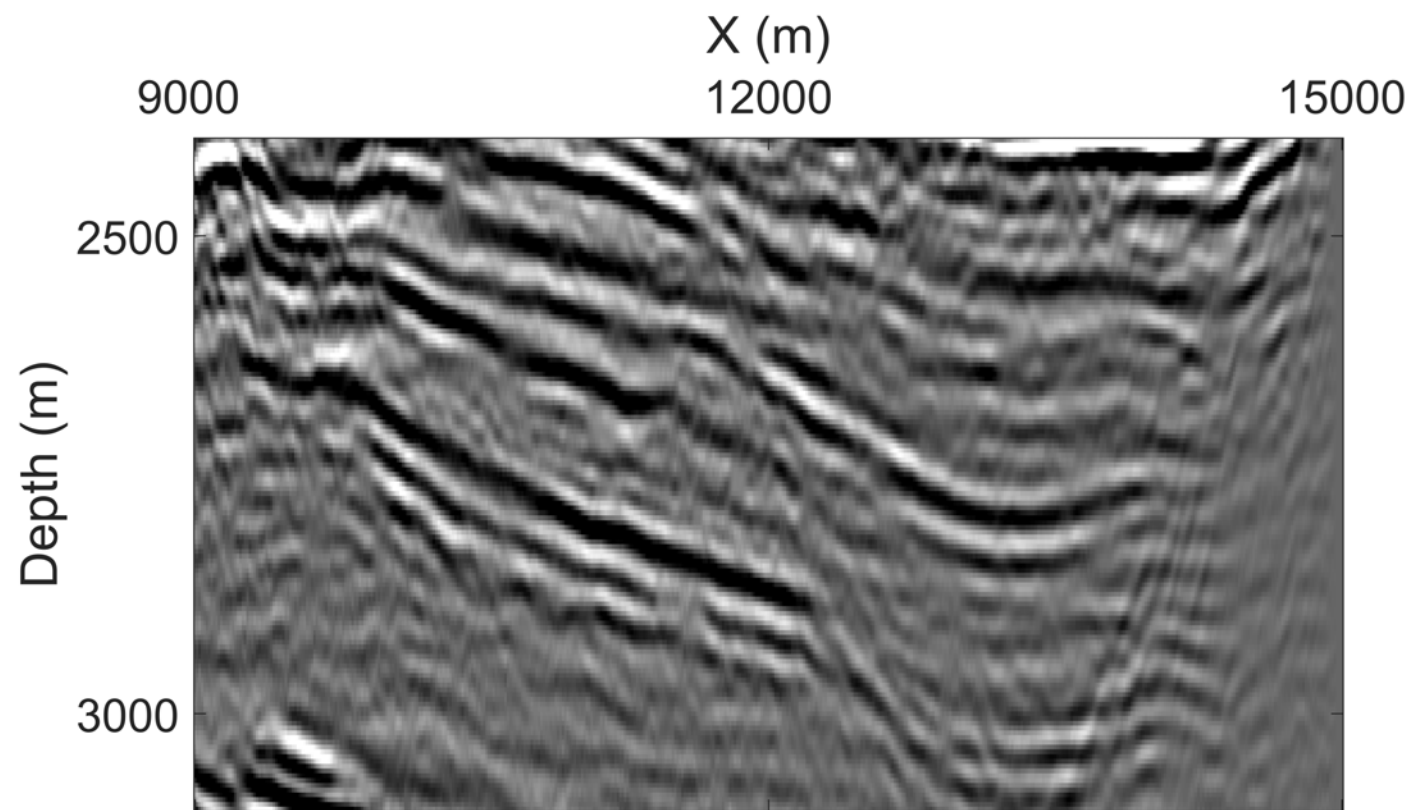


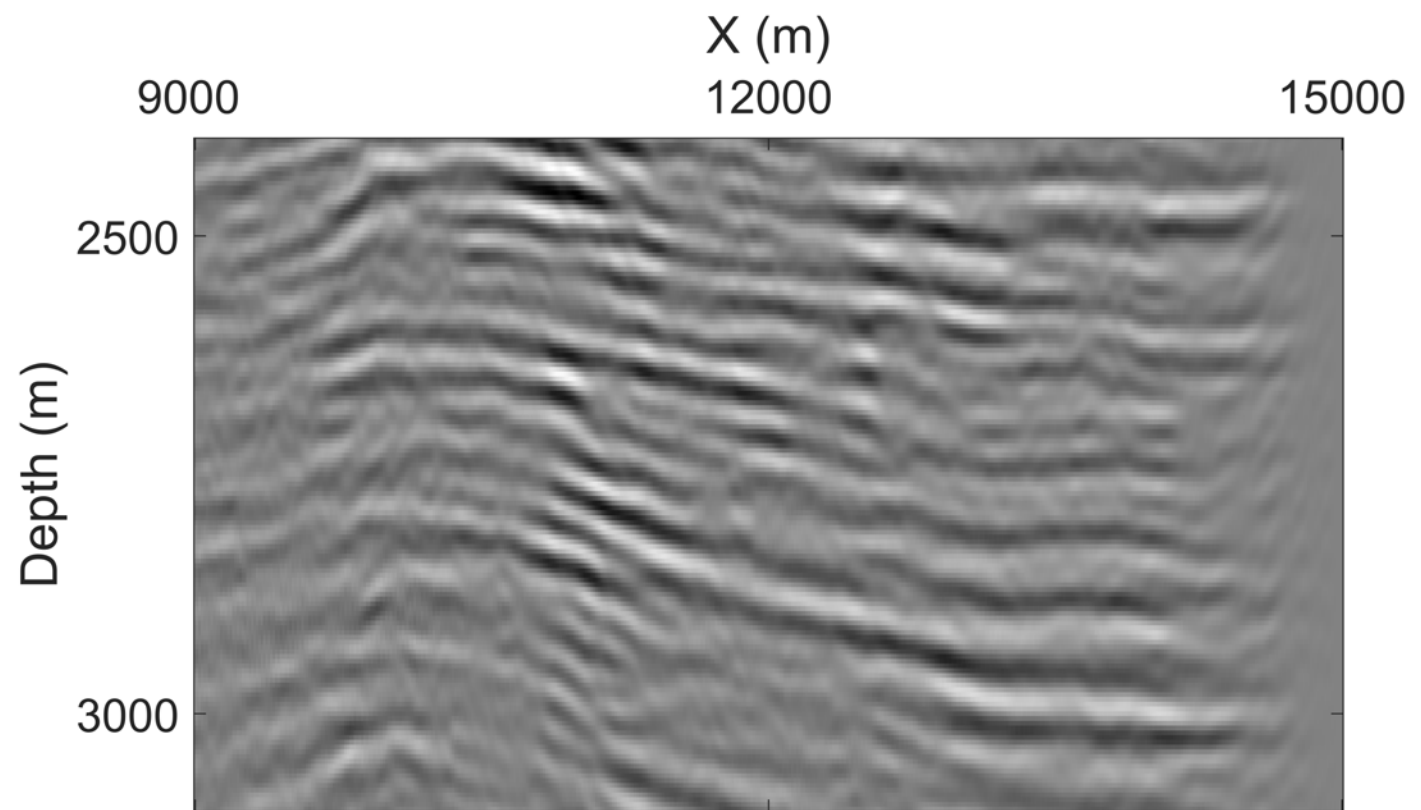








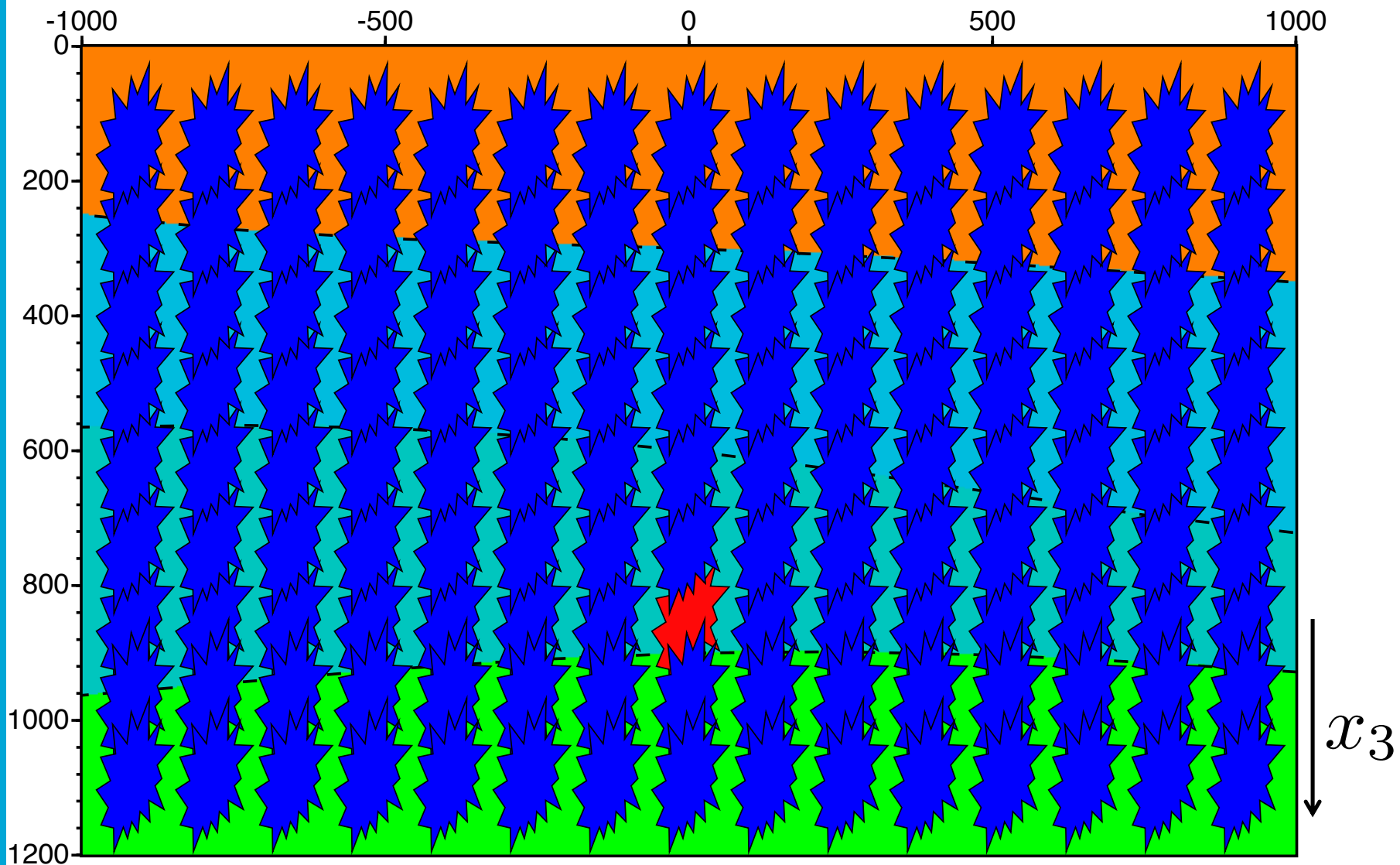




Homogeneous Green's function



Reference: Wapenaar, K., Brackenhoff, J., Thorbecke, J., van der Neut, J., Slob, E., and Verschuur, E., 2018, *Virtual acoustics in inhomogeneous media with single-sided access*: Scientific Reports, Vol. 8, 2497..



Creation of virtual source *and* virtual receivers 224

