计算地球物理第八讲

地震成像和高性能计算

中国地质大学（武汉）
地球物理与空间信息学院
汪宇锋
Email: wangyufeng@cug.edu.cn
Website: https://yufengwa.github.io/CUG-homepage

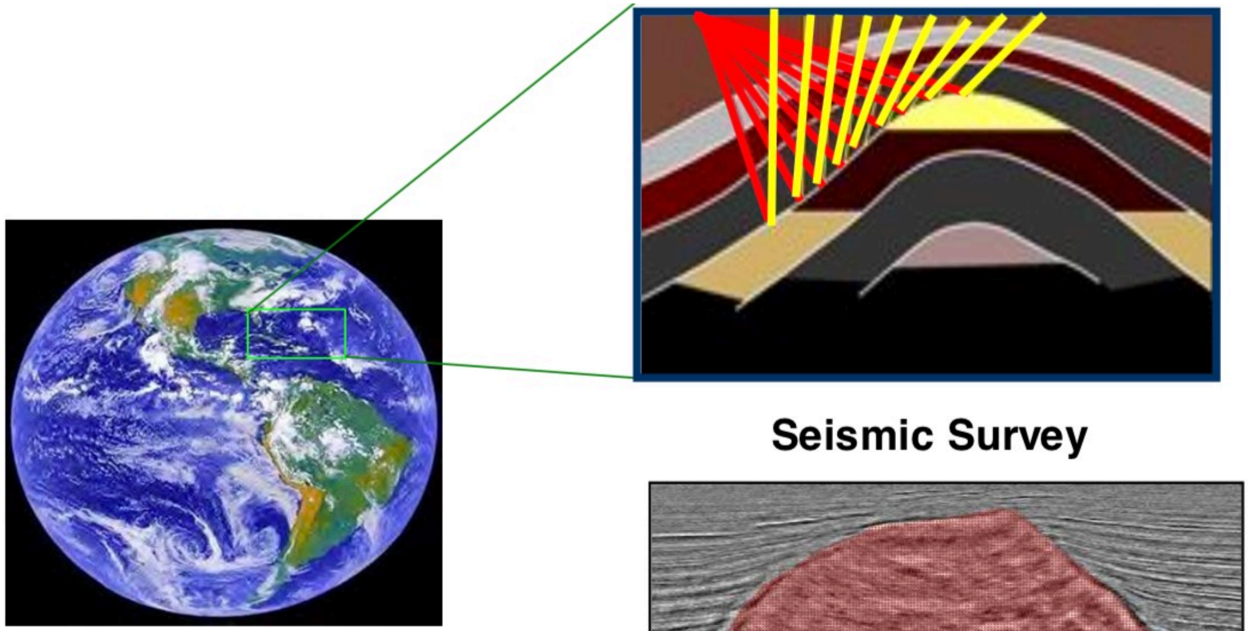
2021.12.27

提纲

1. **地震成像**
2. 高性能计算
3. 高性能地震成像实例
4. 地球物理开源研究

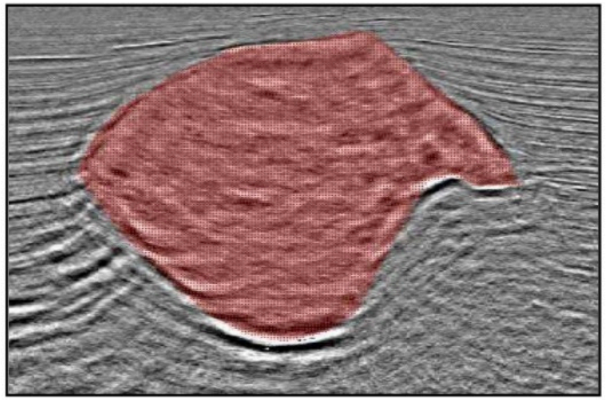
地震成像

Geo-Scientists



Planet Earth

Seismic Survey

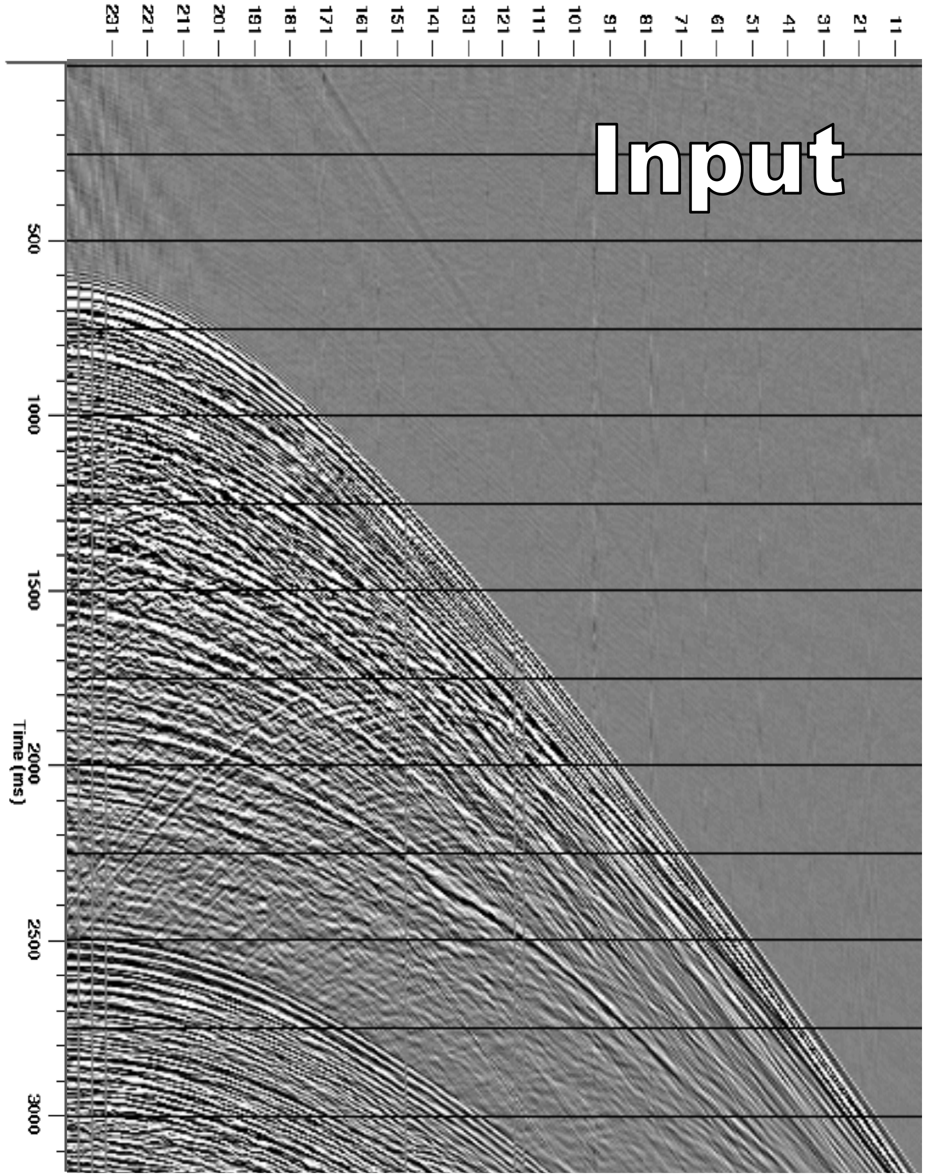


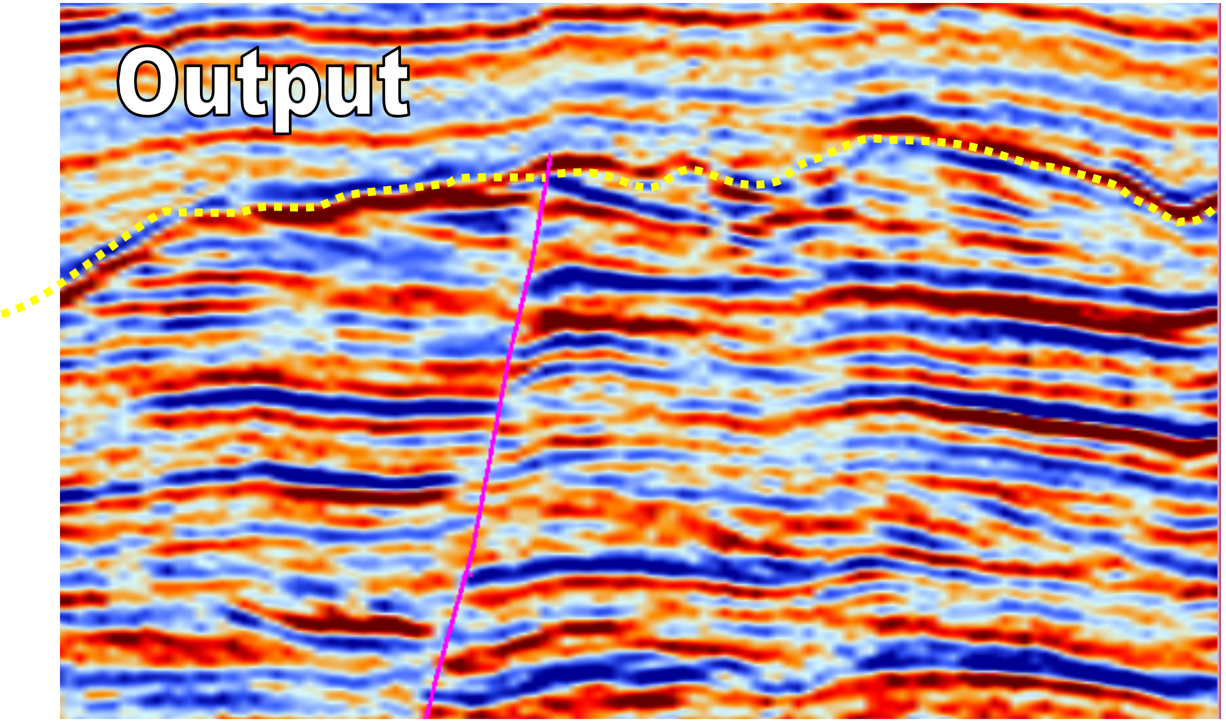
Subsurface
Image

To Understand and Image Earth at higher & higher Resolution

Credited by Richa Rastogi, 2011

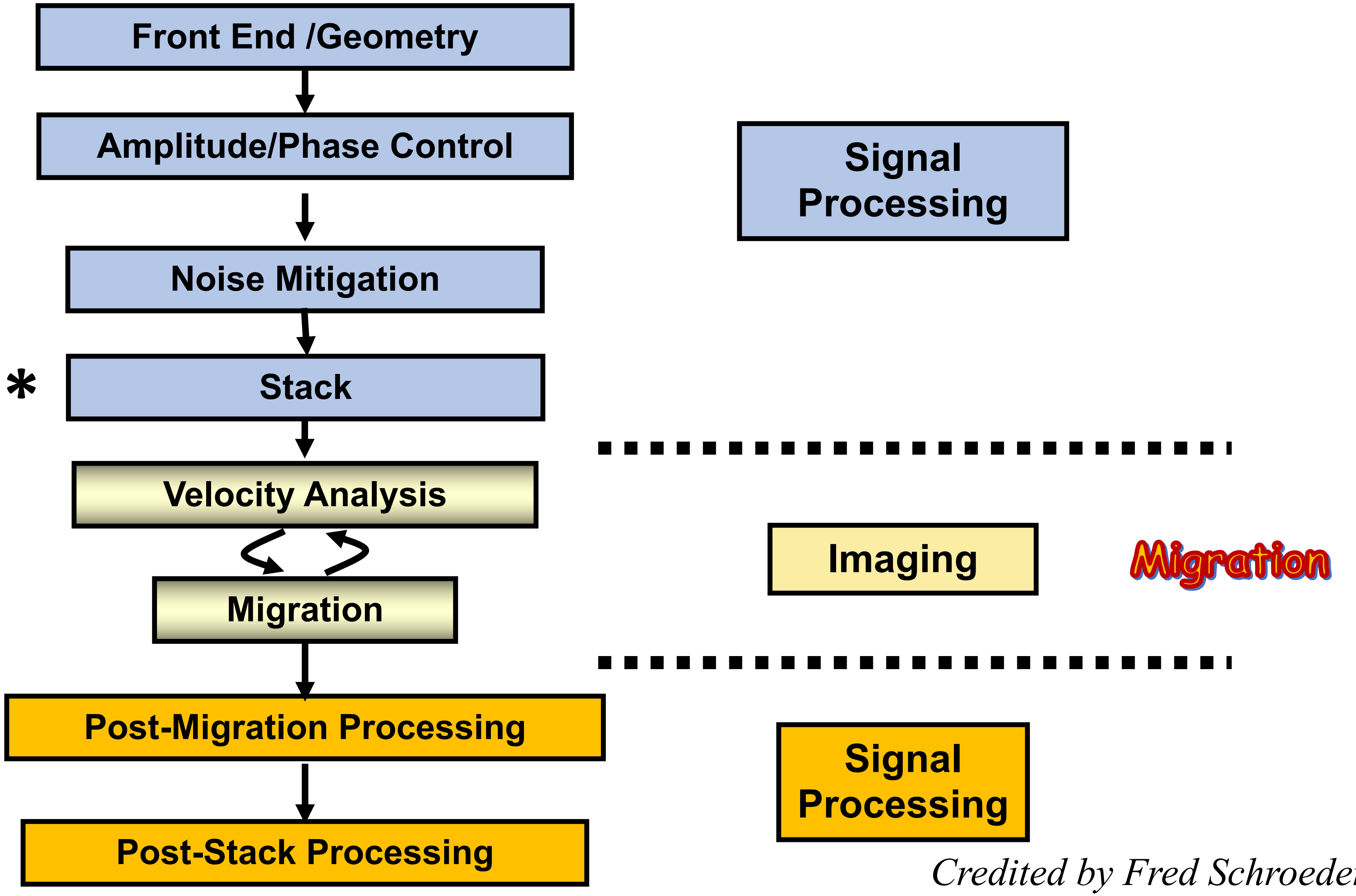
地震成像





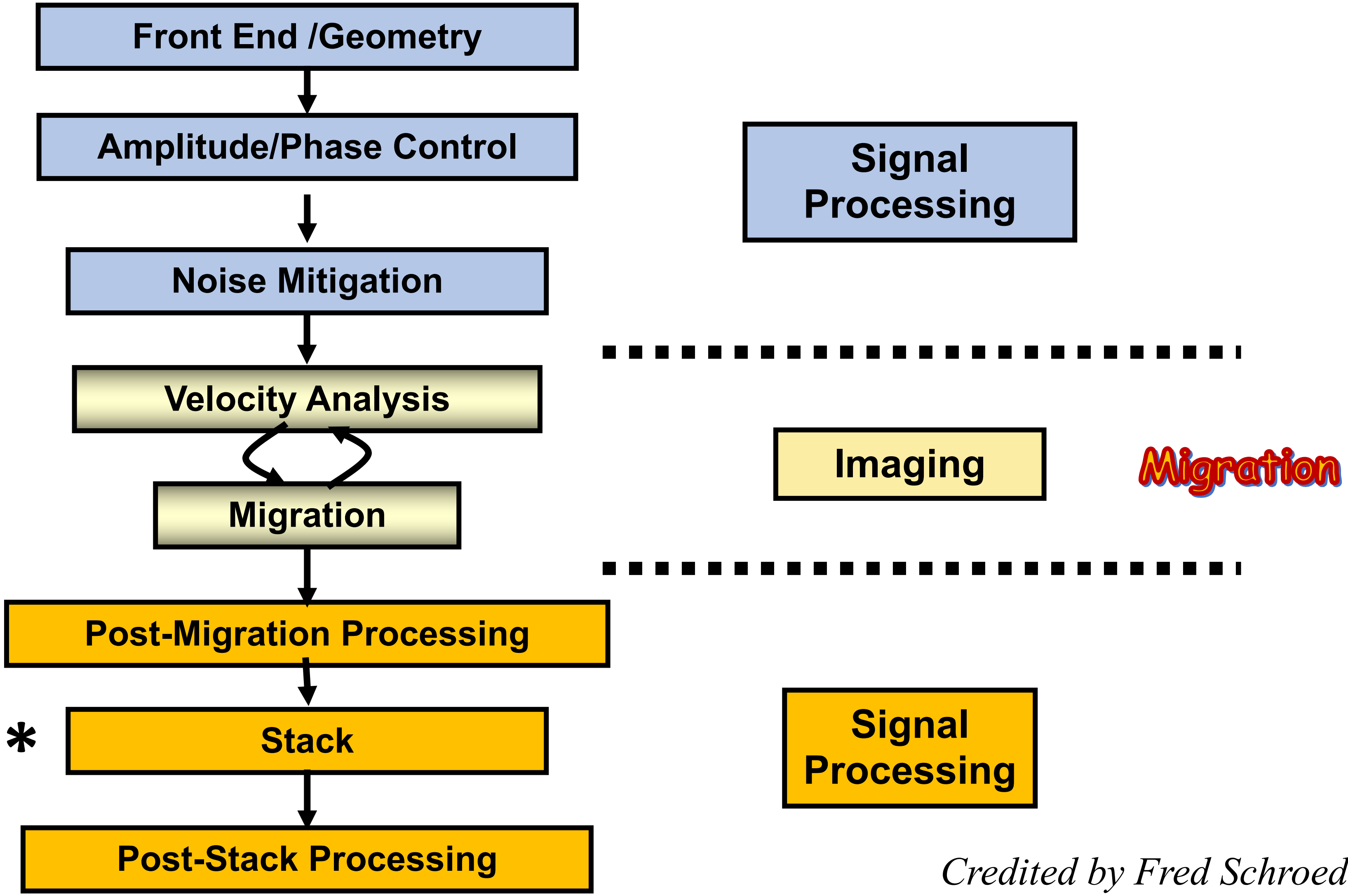
Credited by Fred Schroeder, 2017

叠后偏移成像



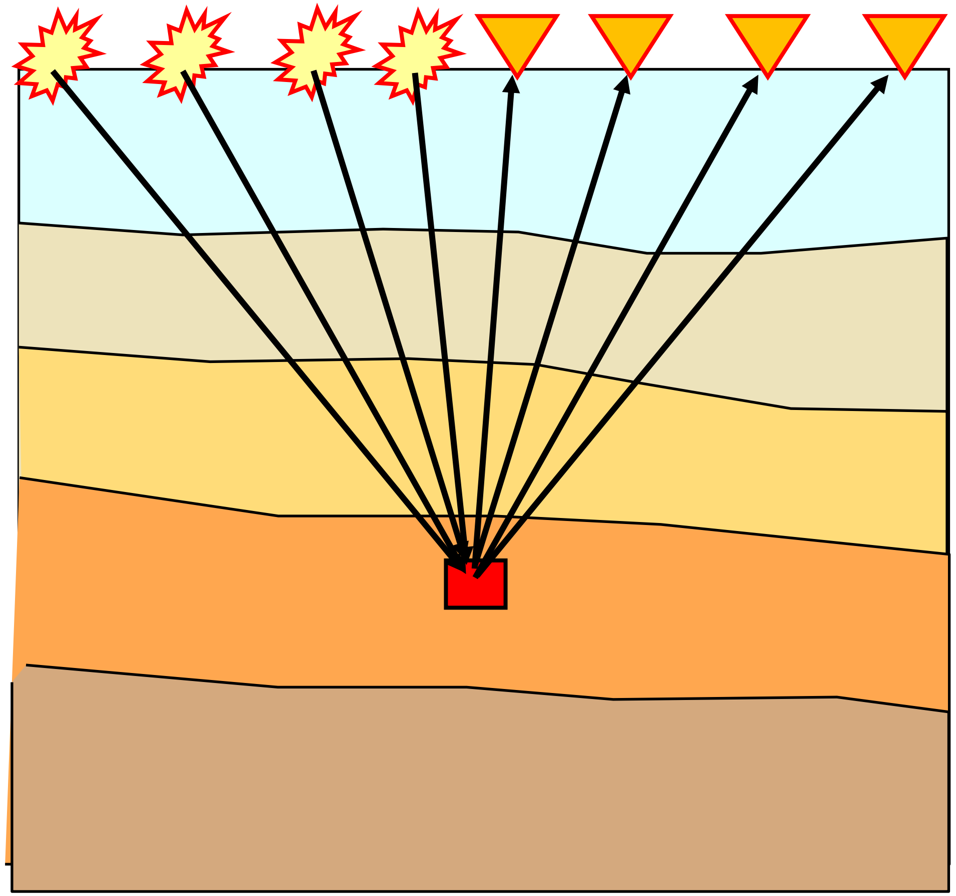
Credited by Fred Schroeder, 2017

疊前偏移成像

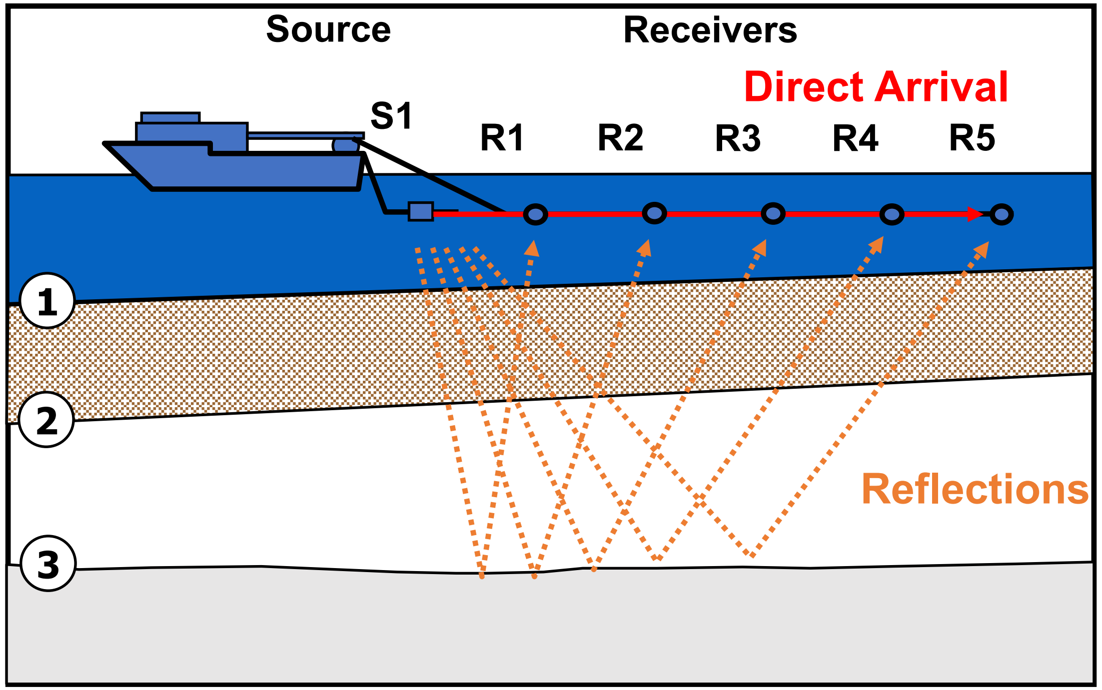


Credited by Fred Schroeder, 2017

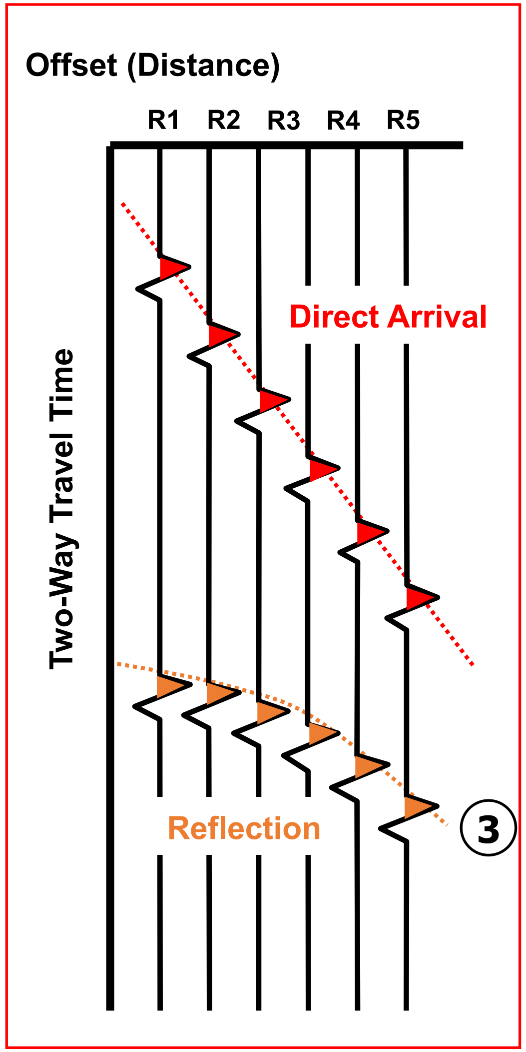
地震叠加



地震叠加

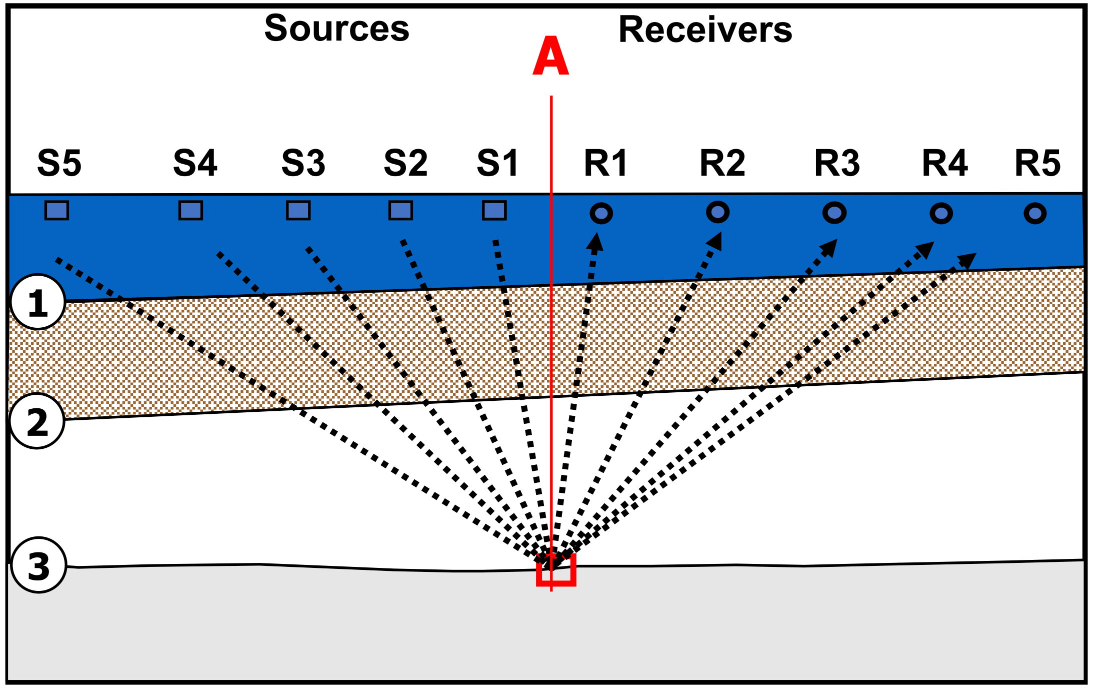


Source
Receivers
Direct Arrival
S1
R1
R2
R3
R4
R5
1
2
3
Reflections

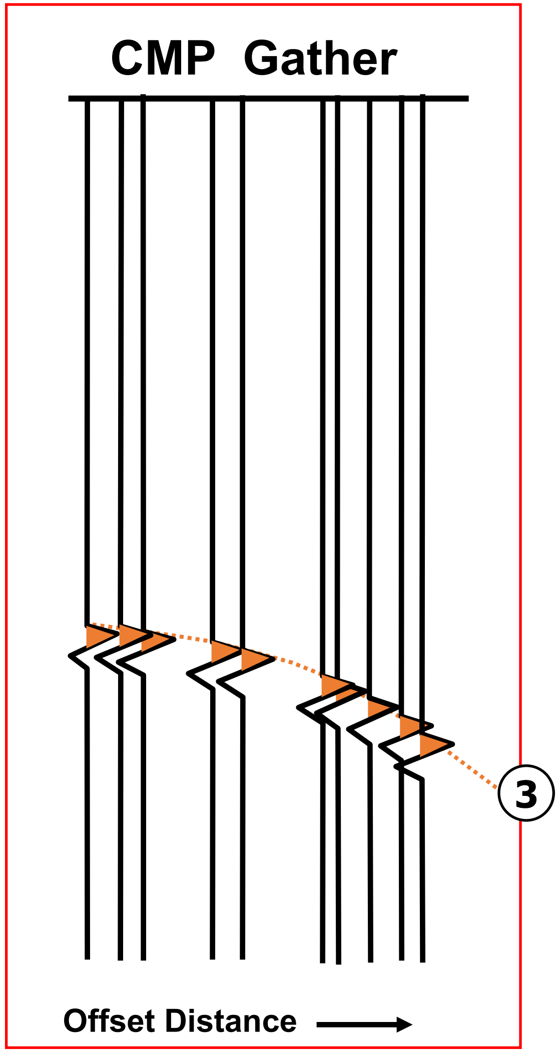


Offset (Distance)
R1 R2 R3 R4 R5
Two-Way Travel Time
Direct Arrival
Reflection
3

地震叠加

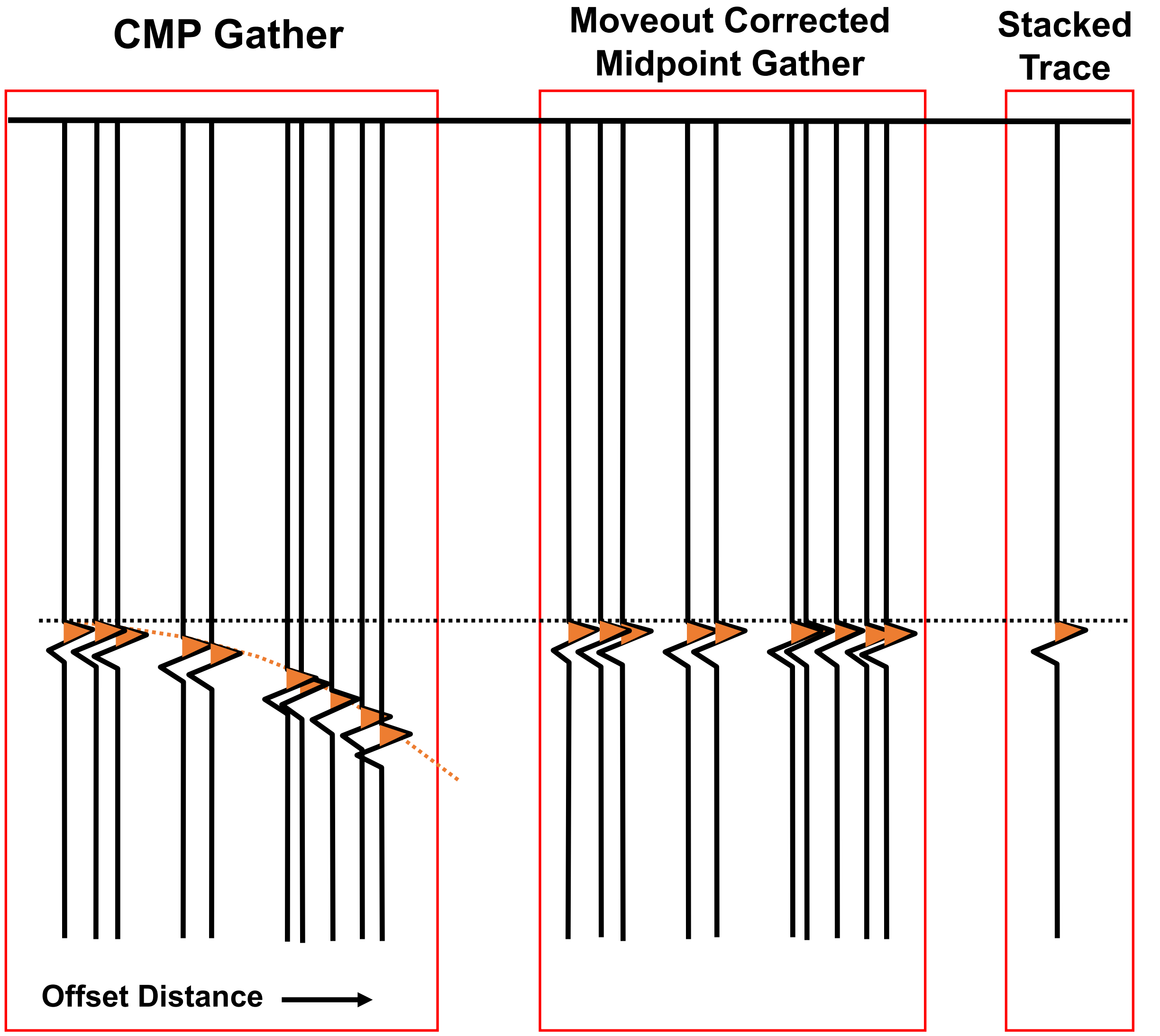


Sources
A
Receivers
S5
S4
S3
S2
S1
R1
R2
R3
R4
R5
1
2
3



CMP Gather
3
Offset Distance

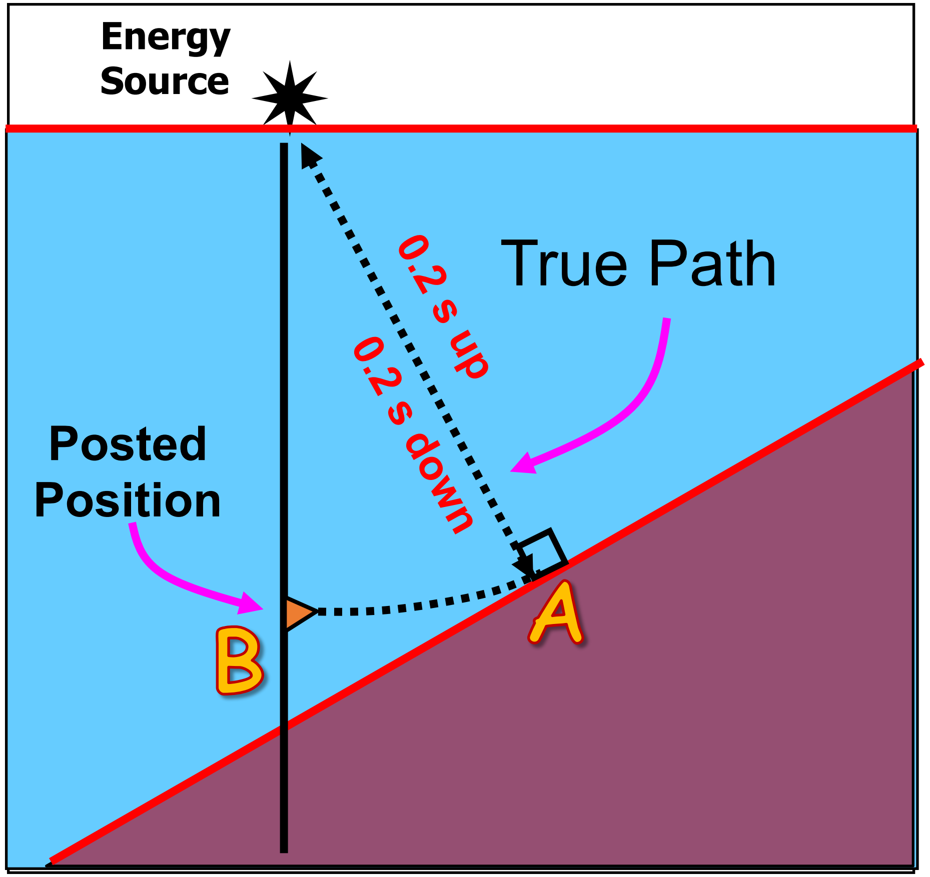
地震叠加



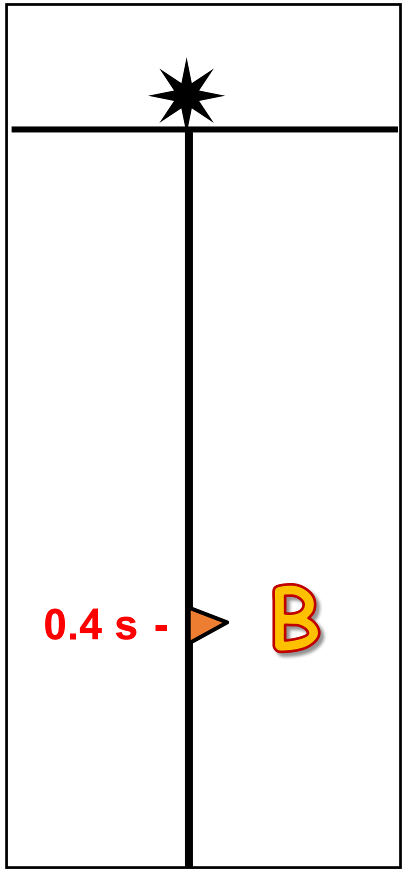
CMP Gather
Moveout Corrected
Midpoint Gather
Stacked
Trace
Offset Distance

10 Fold

地震疊加≠地震成像

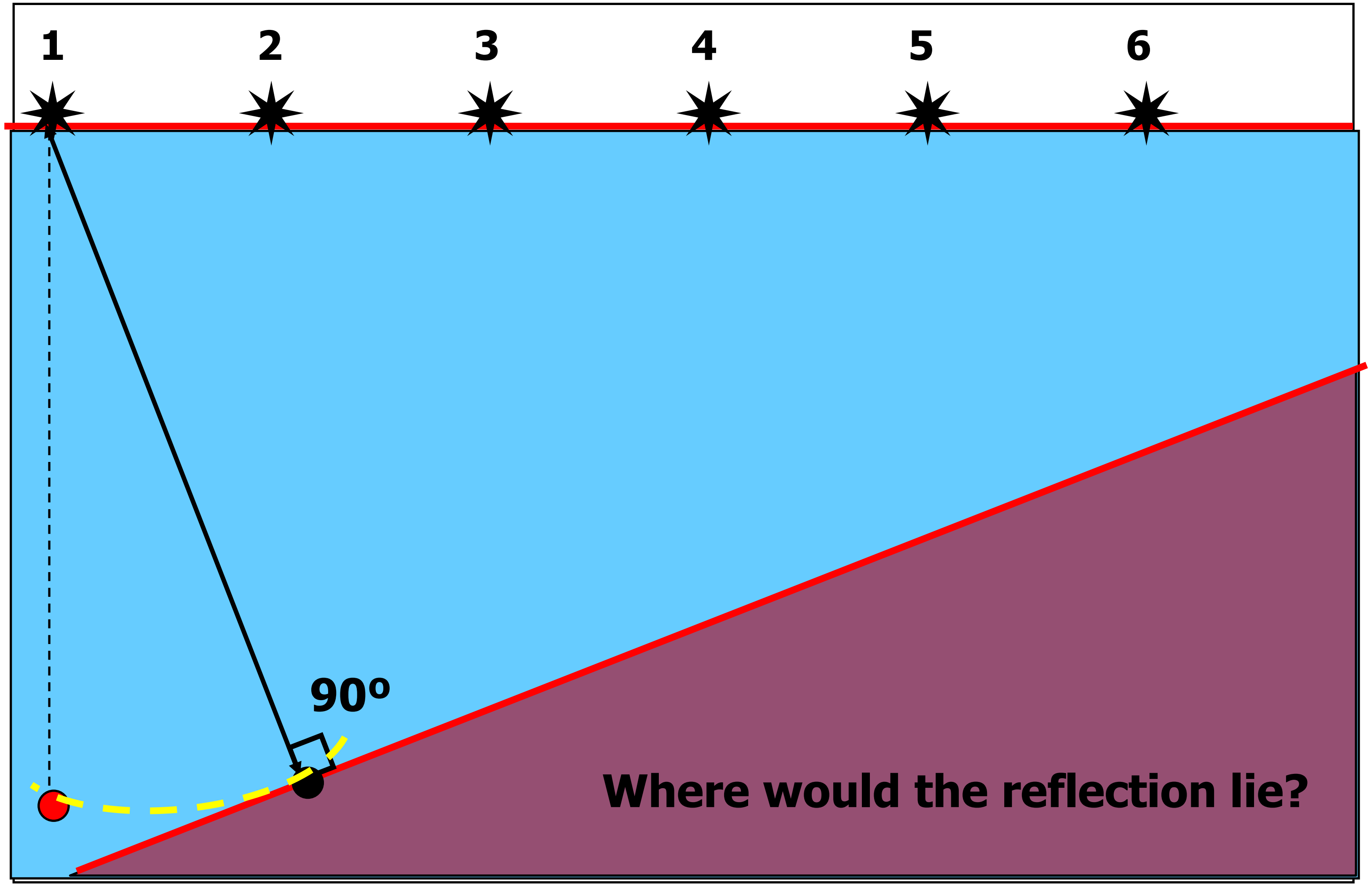


Energy
Source
True Path
0.2 s up
0.2 s down
Posted
Position
B
A

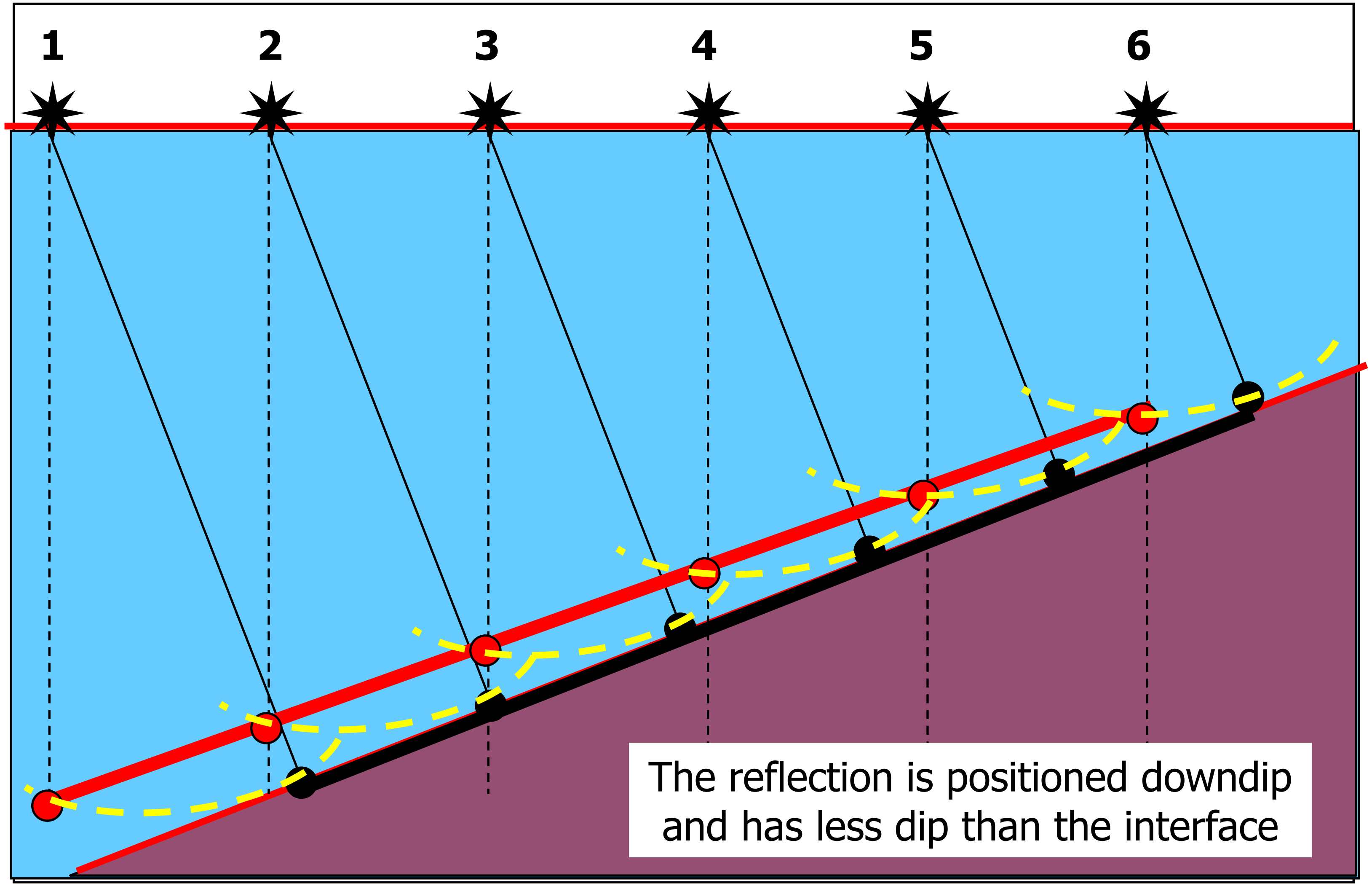


0.4 s -
B

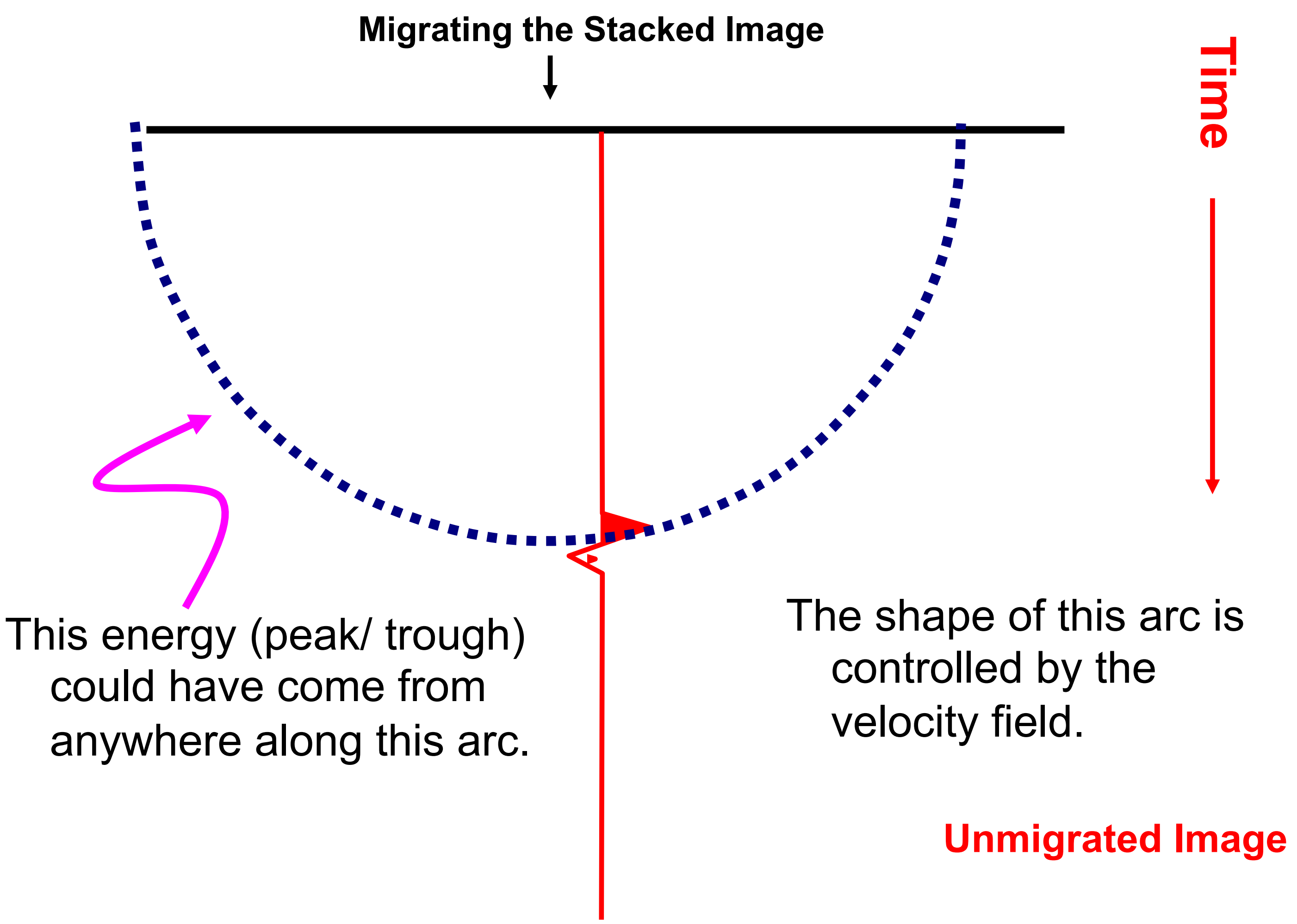
地震叠加≠地震成像



地震叠加≠地震成像

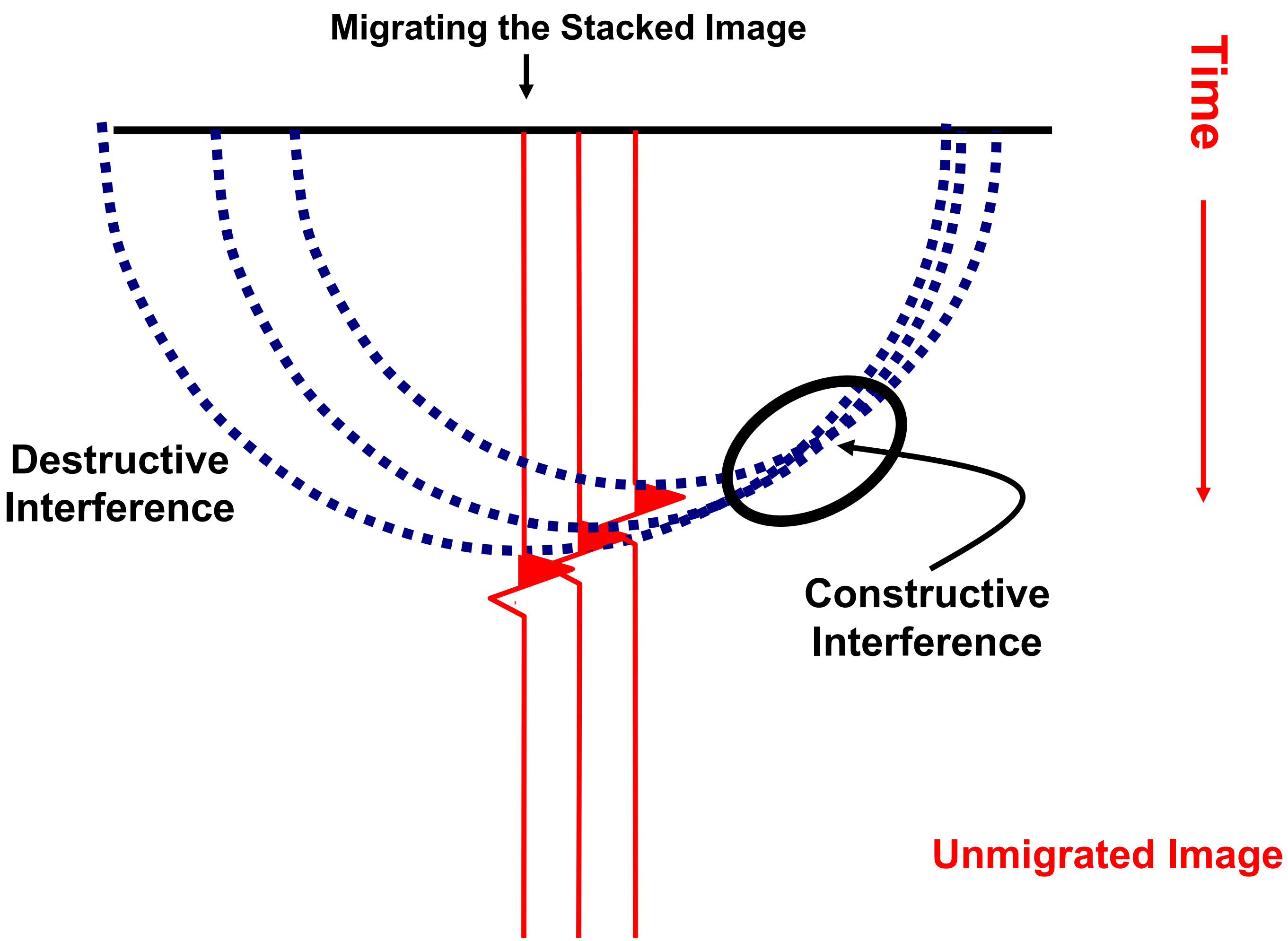

1
2
3
4
5
6
The reflection is positioned downdip and has less dip than the interface

地震叠后偏移成像



Migrating the Stacked Image
Time
This energy (peak/ trough) could have come from anywhere along this arc.
The shape of this arc is controlled by the velocity field.
Unmigrated Image

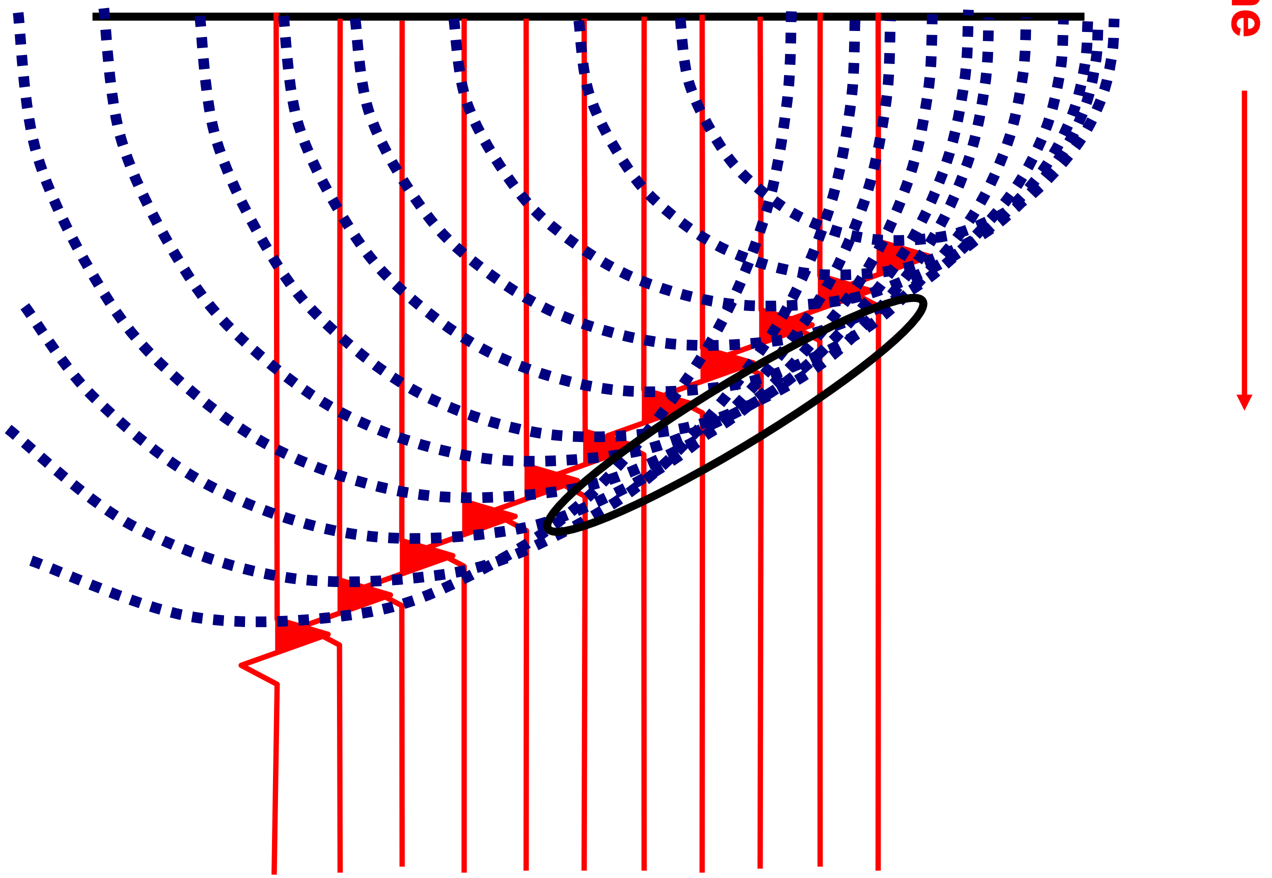
地震叠后偏移成像



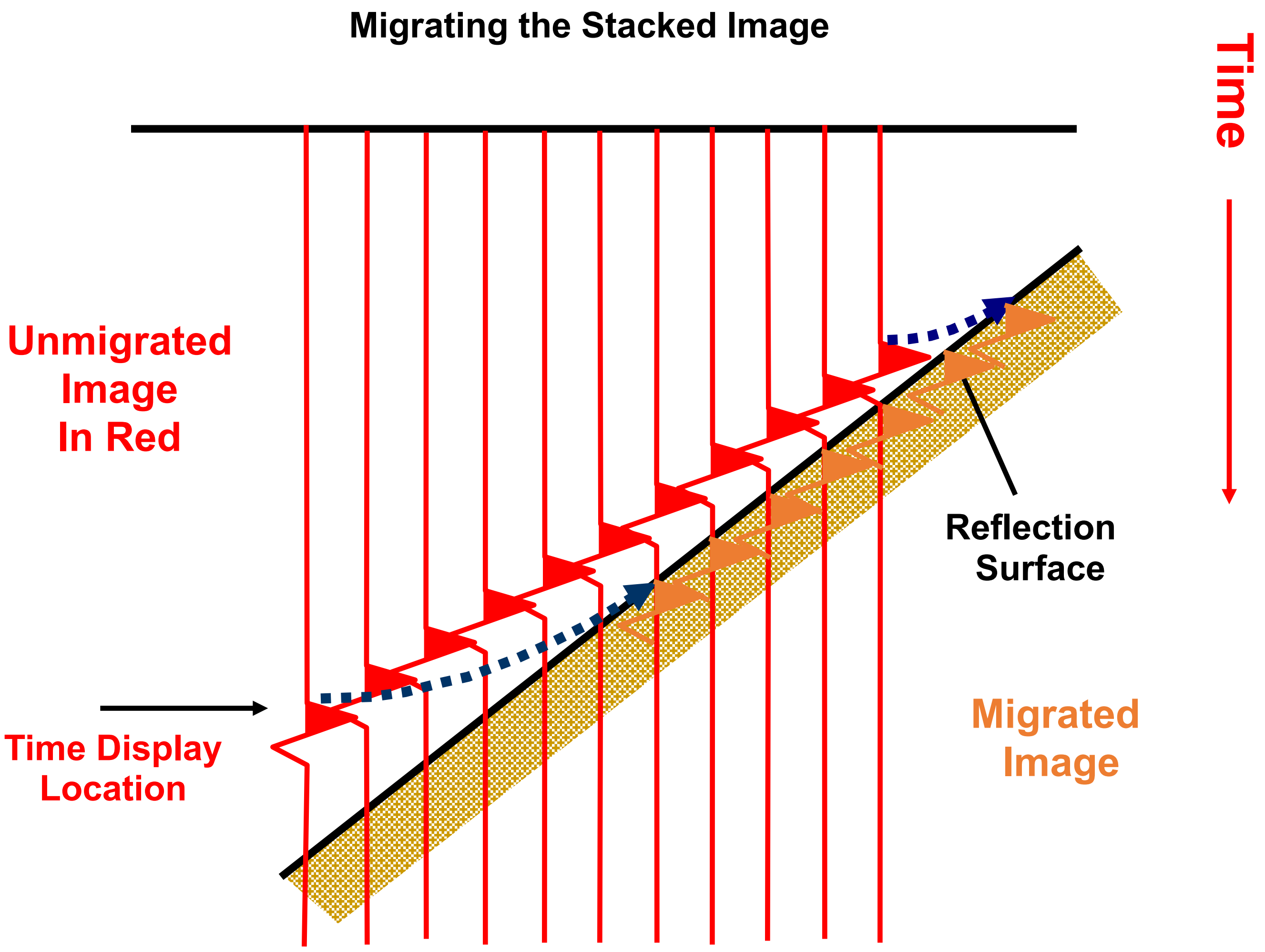
Migrating the Stacked Image
Time
Destructive Interference
Constructive Interference
Unmigrated Image

地震叠后偏移成像

Migrating the Stacked Image

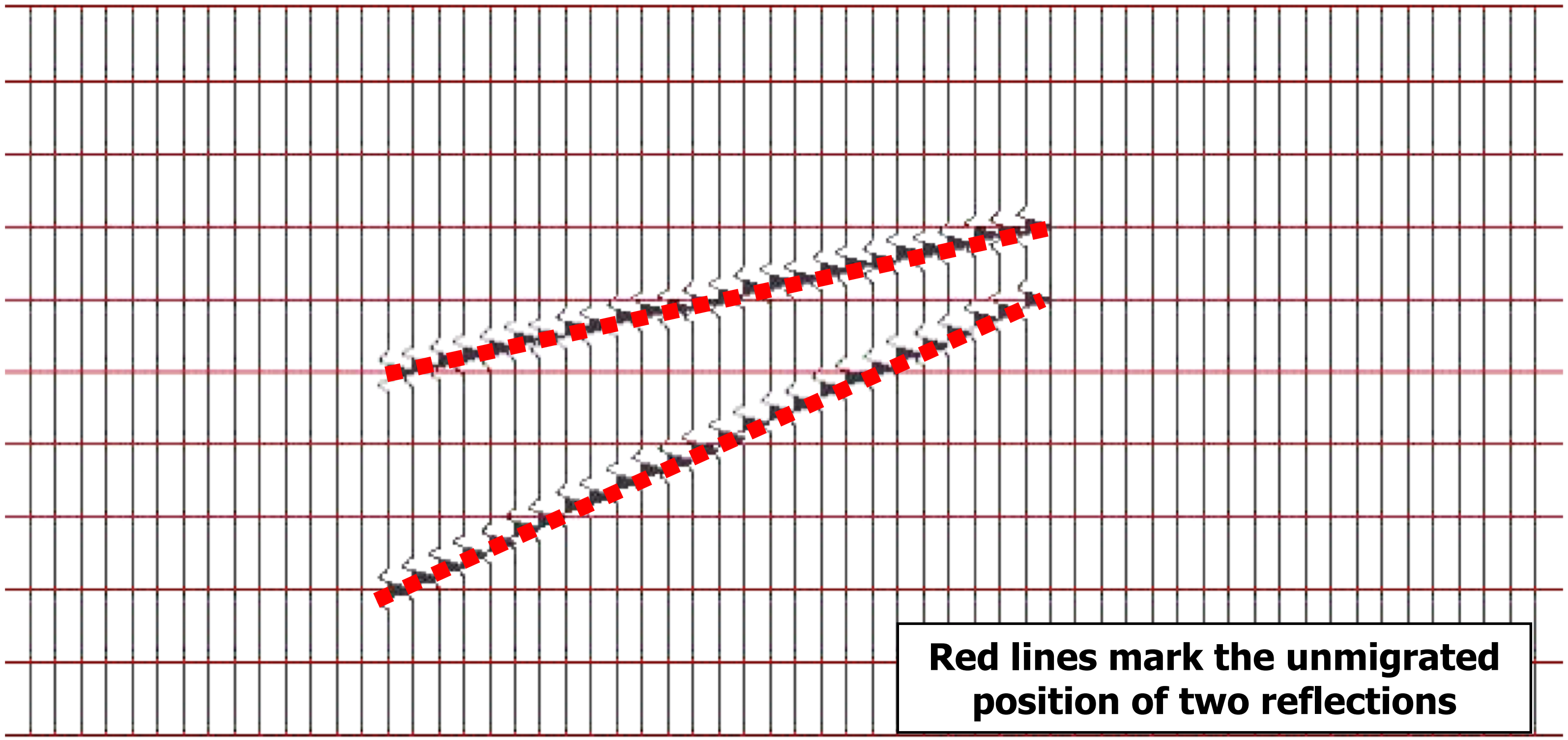


地震叠后偏移成像

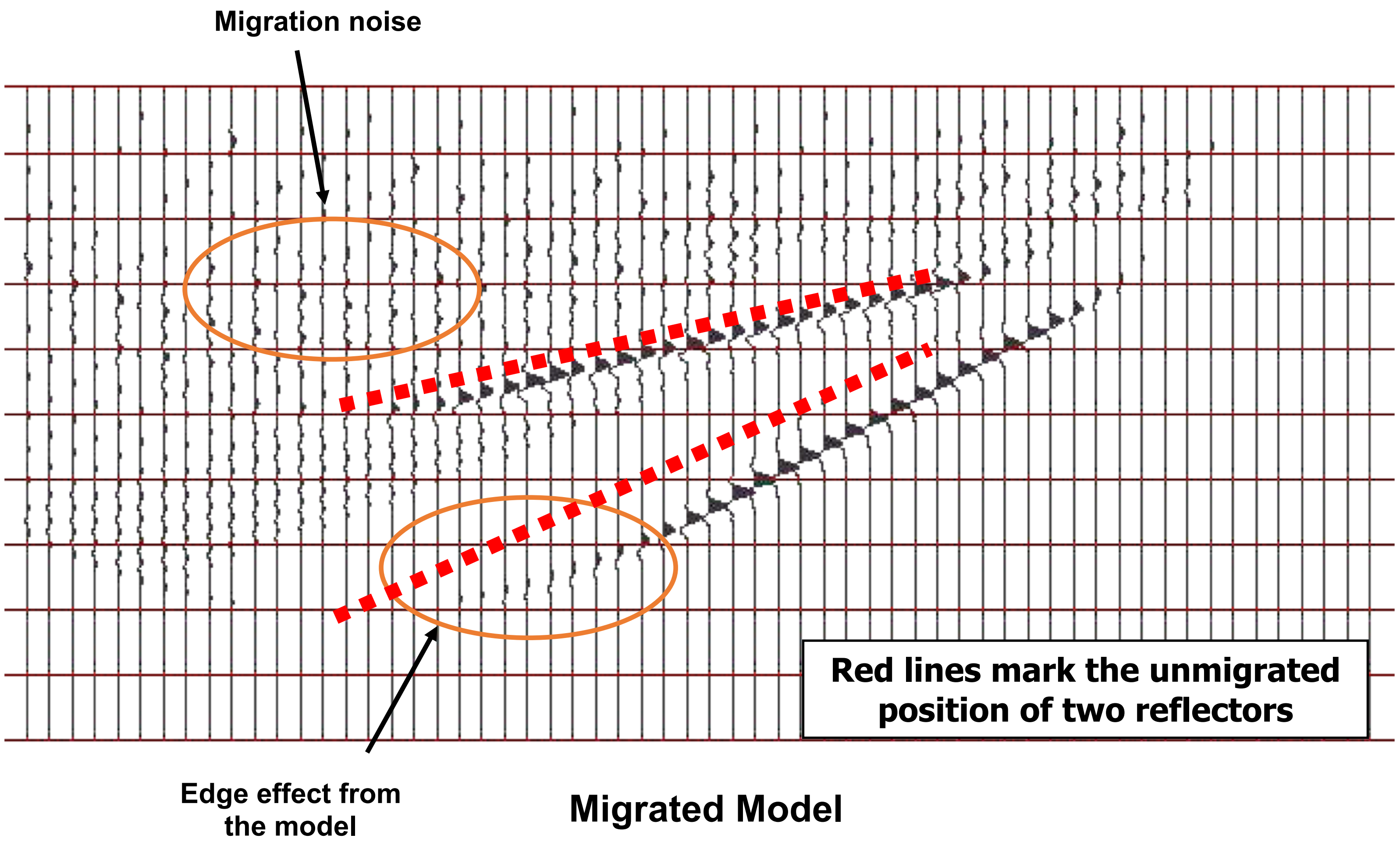


Migrating the Stacked Image
Time
Unmigrated
Image
In Red
Reflection
Surface
Time Display
Location
Migrated
Image

地震叠后偏移成像

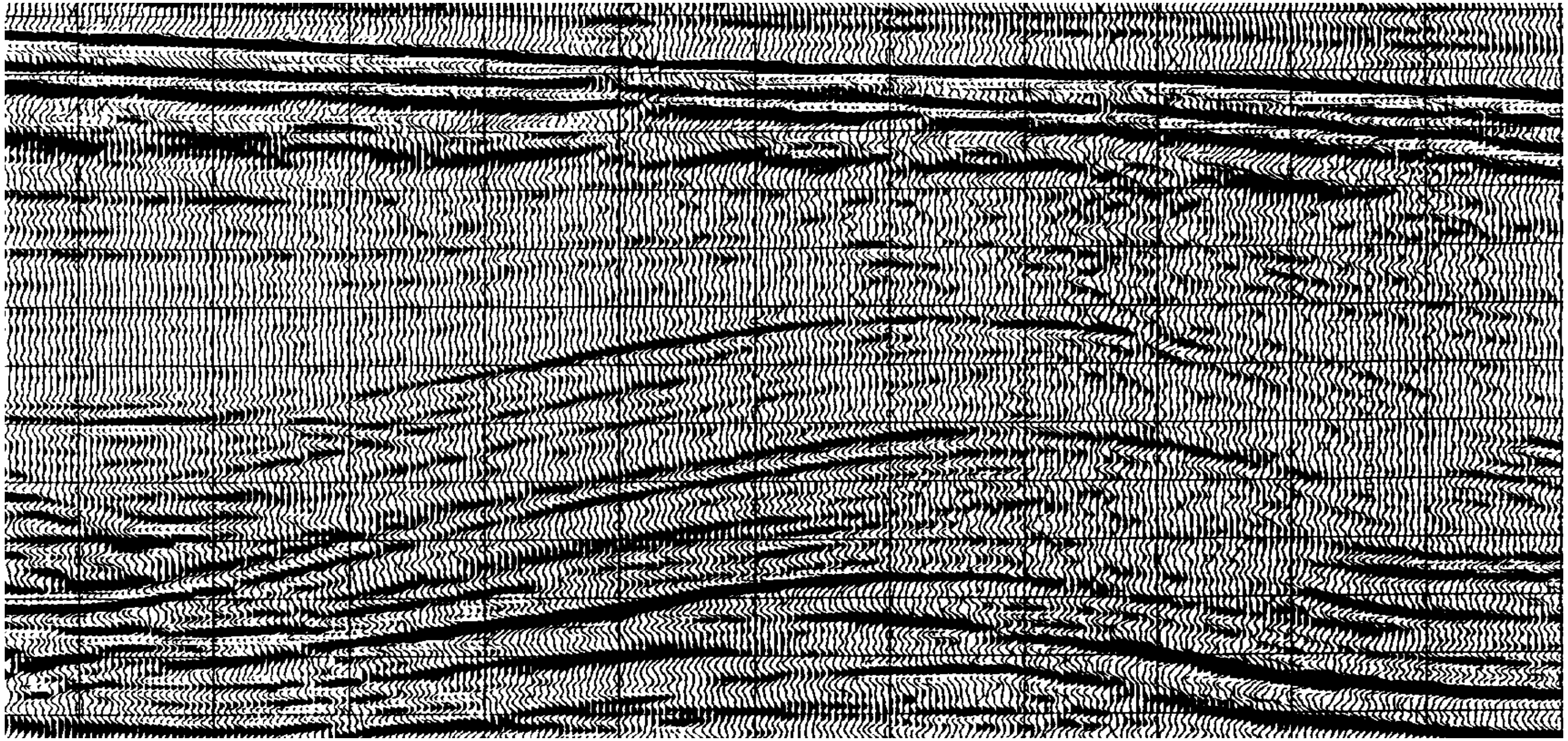


地震叠后偏移成像



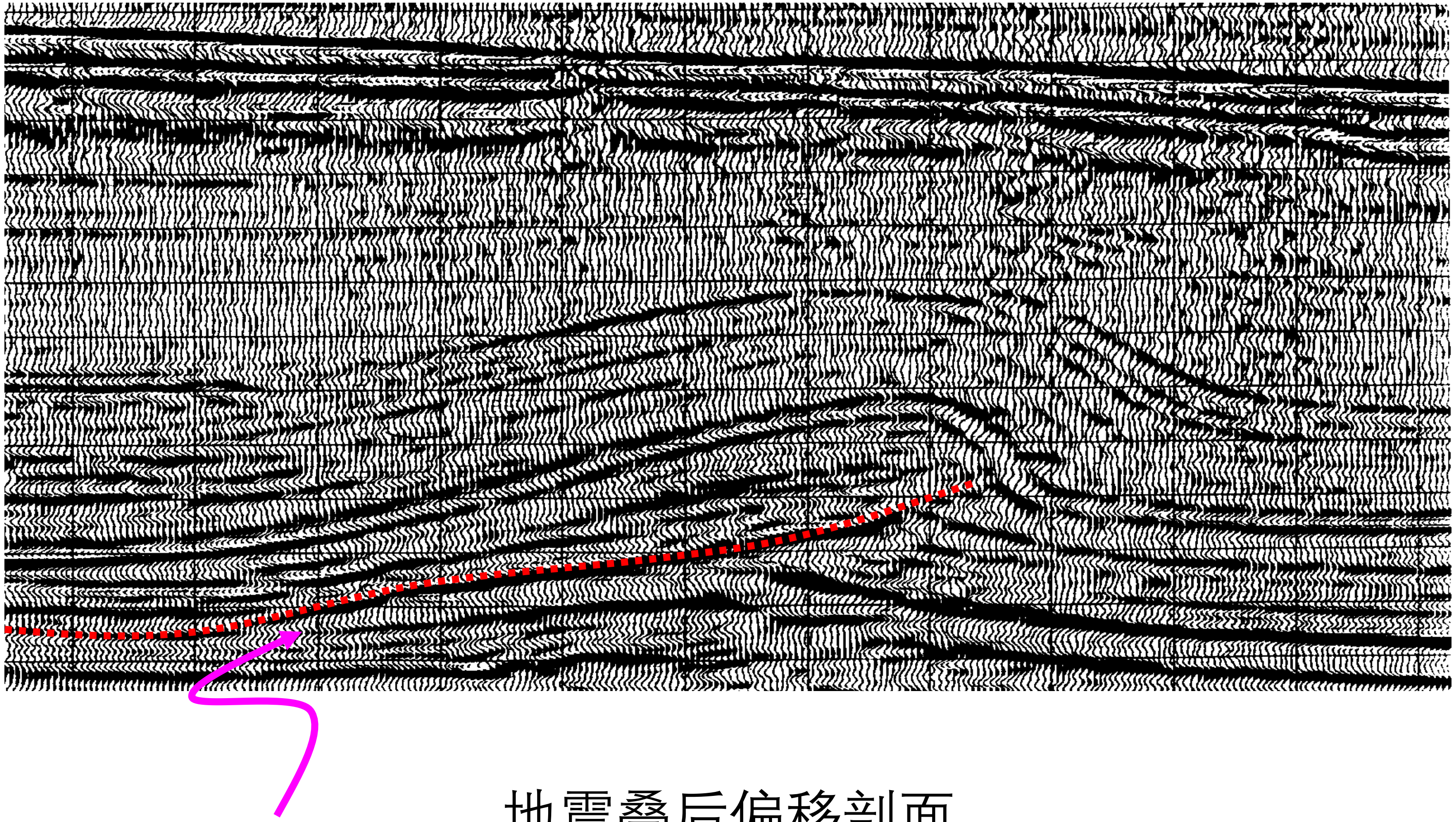
Migrated Model

地震叠后偏移成像



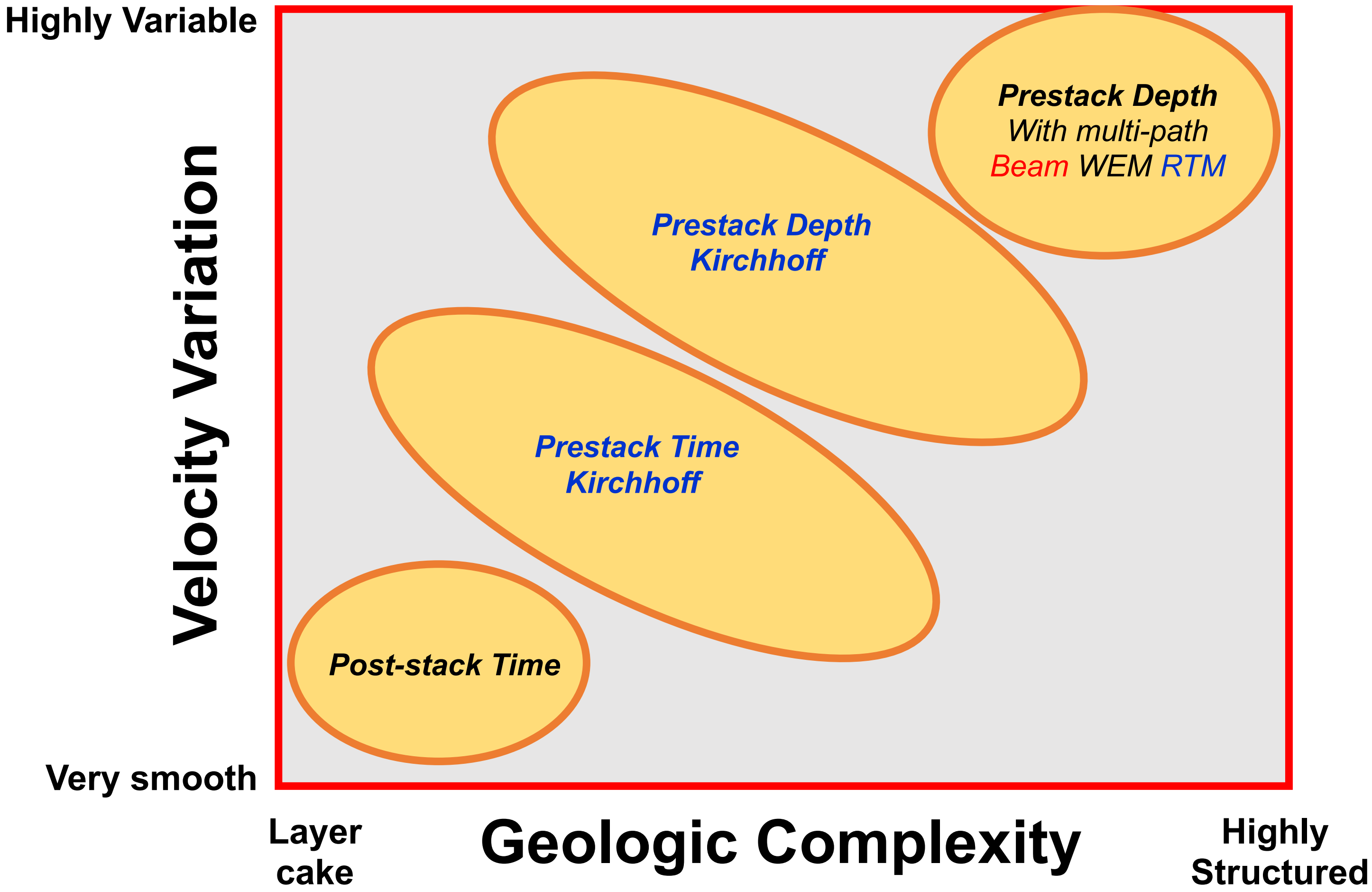
地震叠加剖面

地震叠后偏移成像



地震叠后偏移剖面

偏移成像的选择


Highly Variable
Very smooth
Velocity Variation
Prestack Depth
With multi-path
Beam WEM RTM
Prestack Depth
Kirchhoff
Prestack Time
Kirchhoff
Post-stack Time
Layer
cake
Geologic Complexity
Highly
Structured

小行星逆时偏移成像

ELSEVIER

CrossMark

Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research 55 (2015) 2149–2165

ADVANCES IN SPACE RESEARCH
(a COSPAR publication)

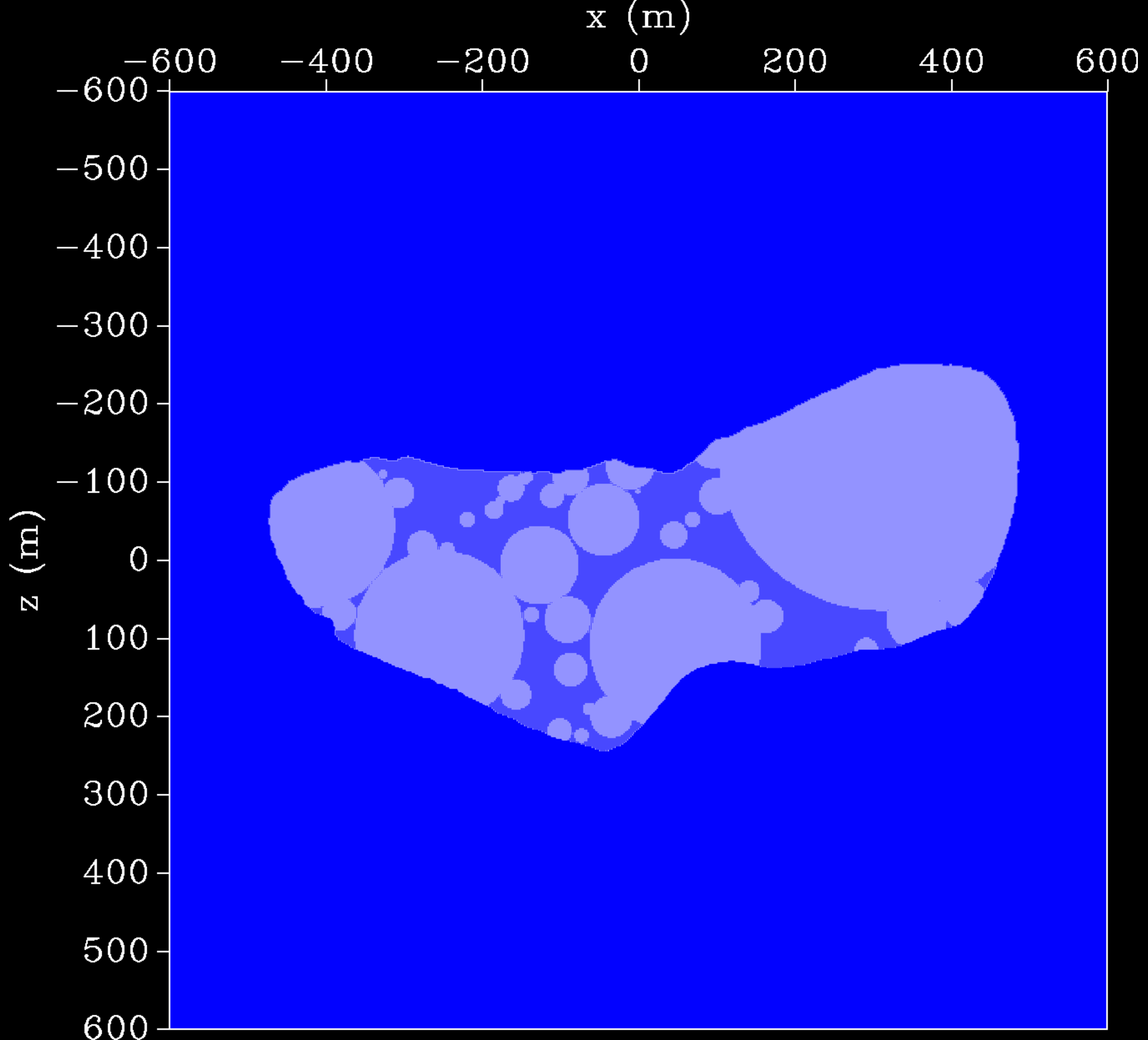
www.elsevier.com/locate/asr

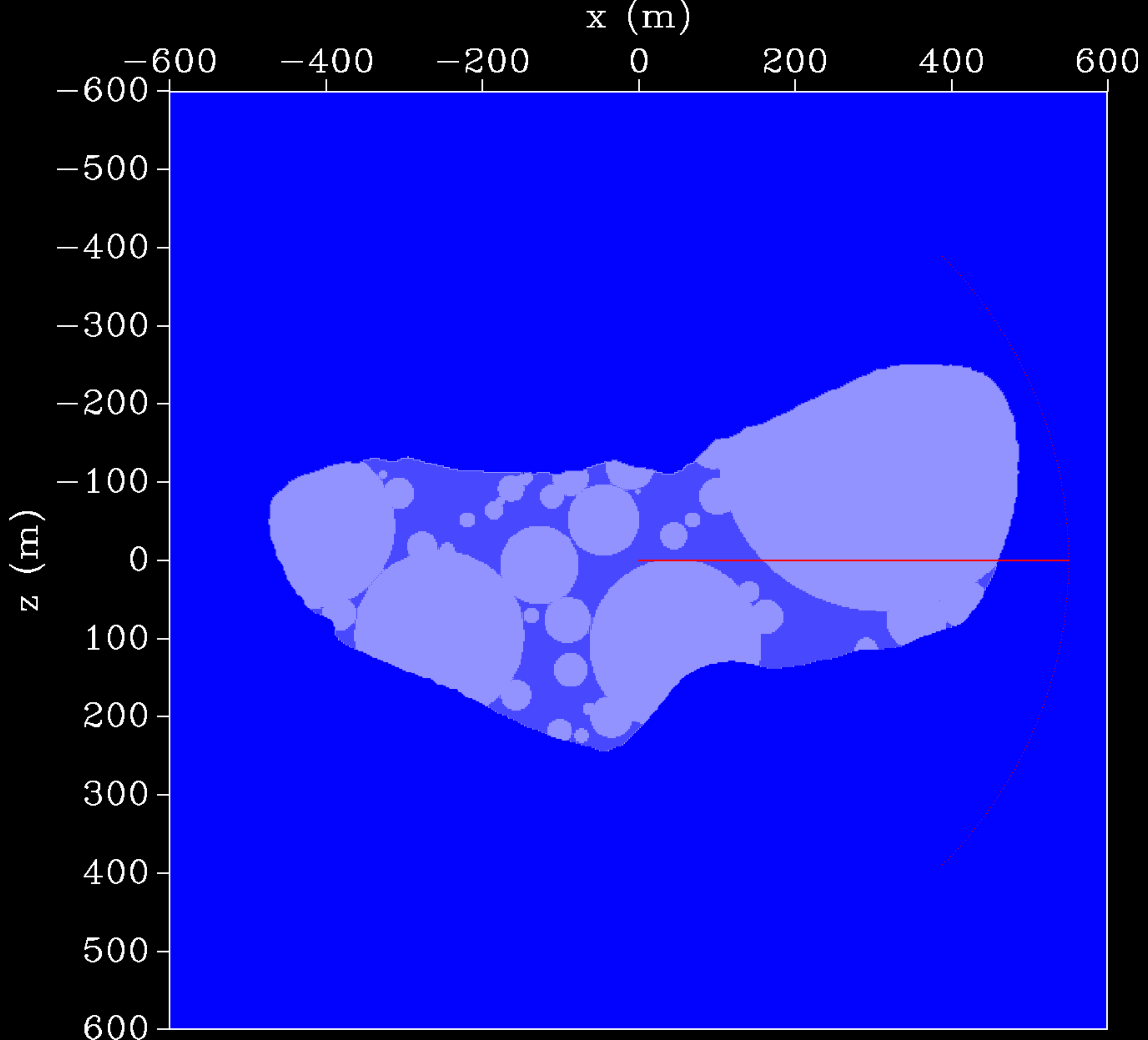
Radio reflection imaging of asteroid and comet interiors I: Acquisition and imaging theory

Paul Sava [a,*], Detchai Ittharat [a], Robert Grimm [b], David Stillman [b]

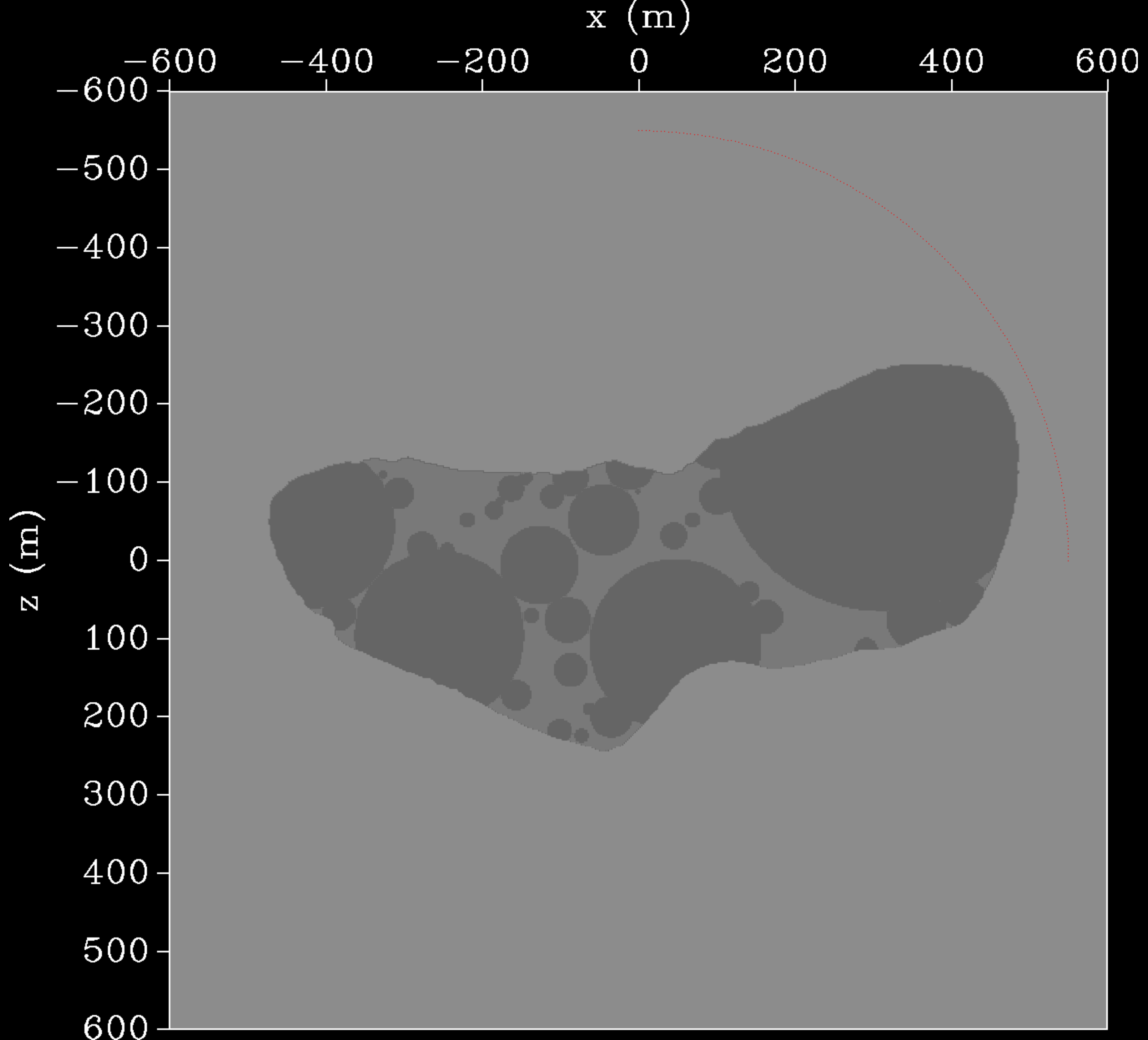
[a] *Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois Street, Golden, CO 80401, USA*
[b] *Planetary Science Directorate, Southwest Research Institute, 1050 Walnut St. #300, Boulder, CO 80302, USA*

Received 3 February 2014; received in revised form 15 October 2014; accepted 15 October 2014
Available online 18 November 2014

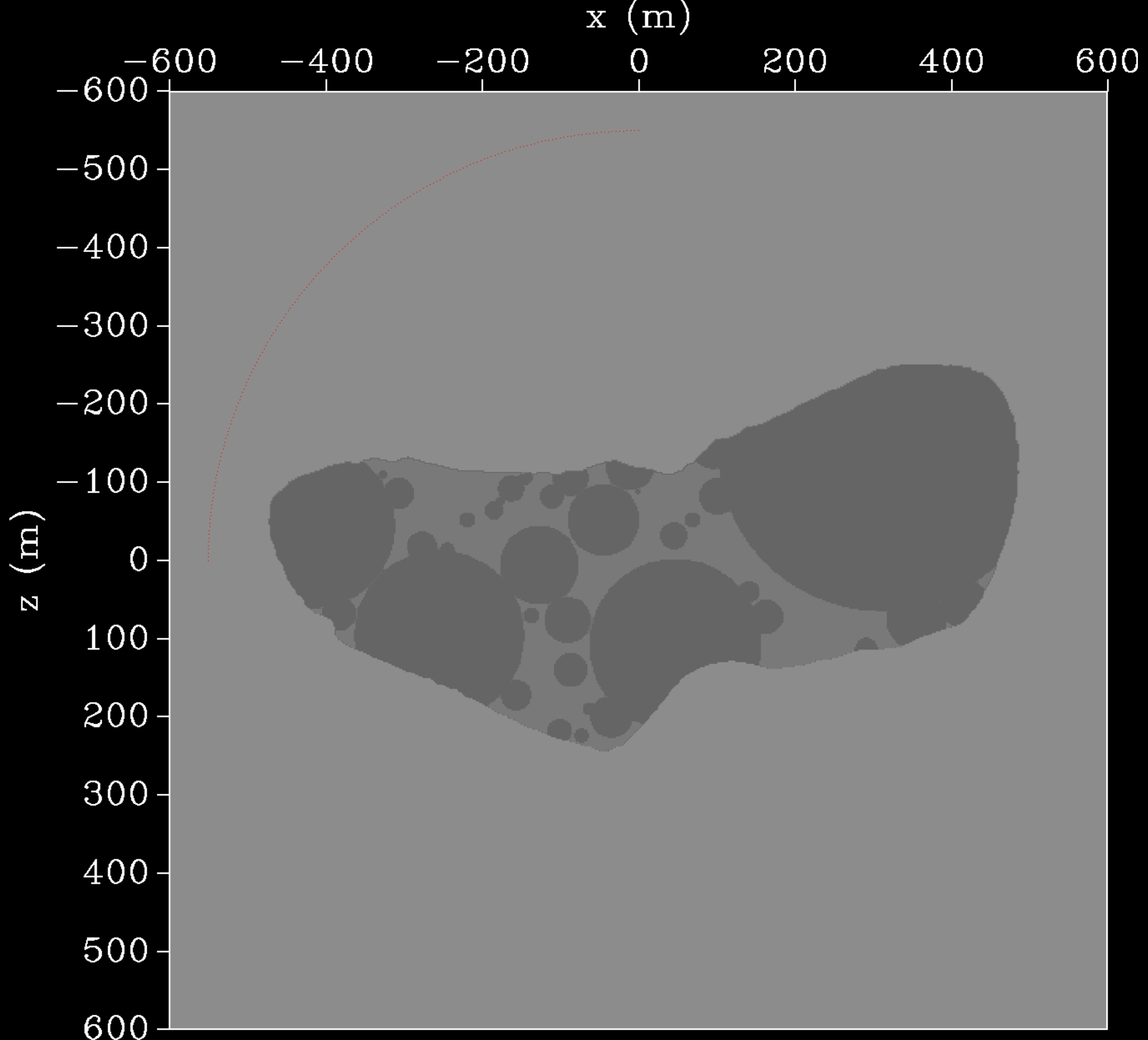

x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600



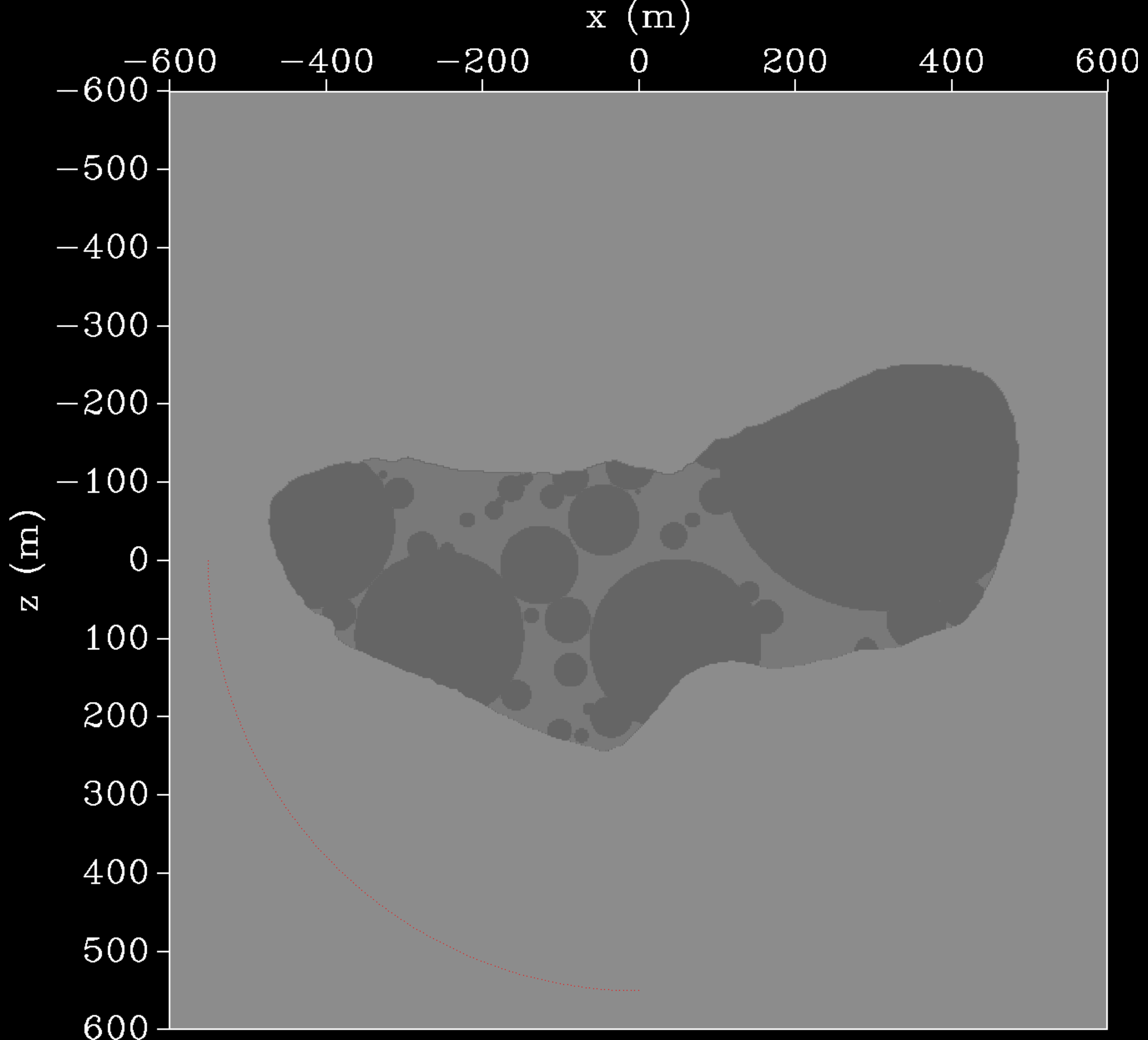
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

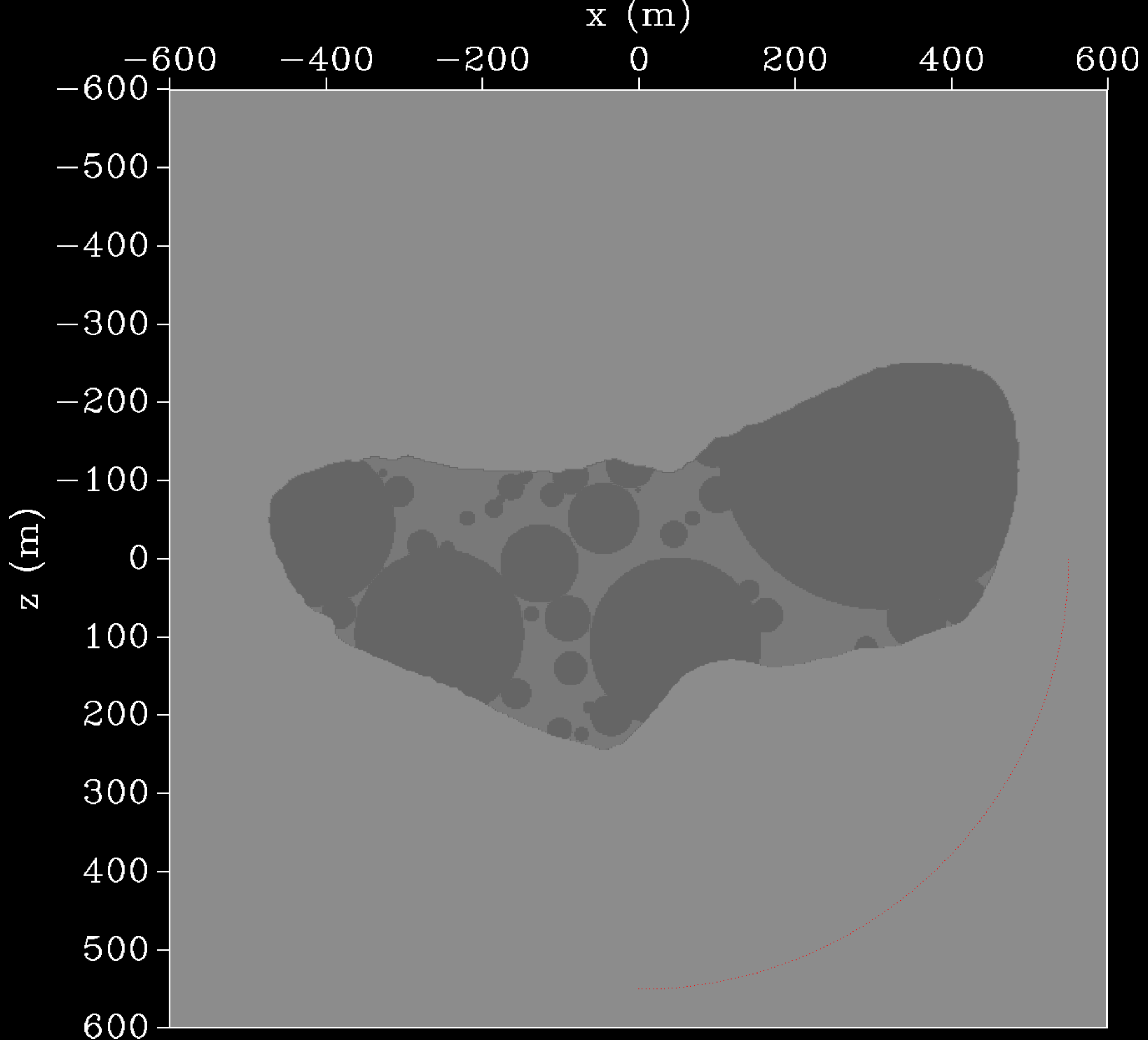


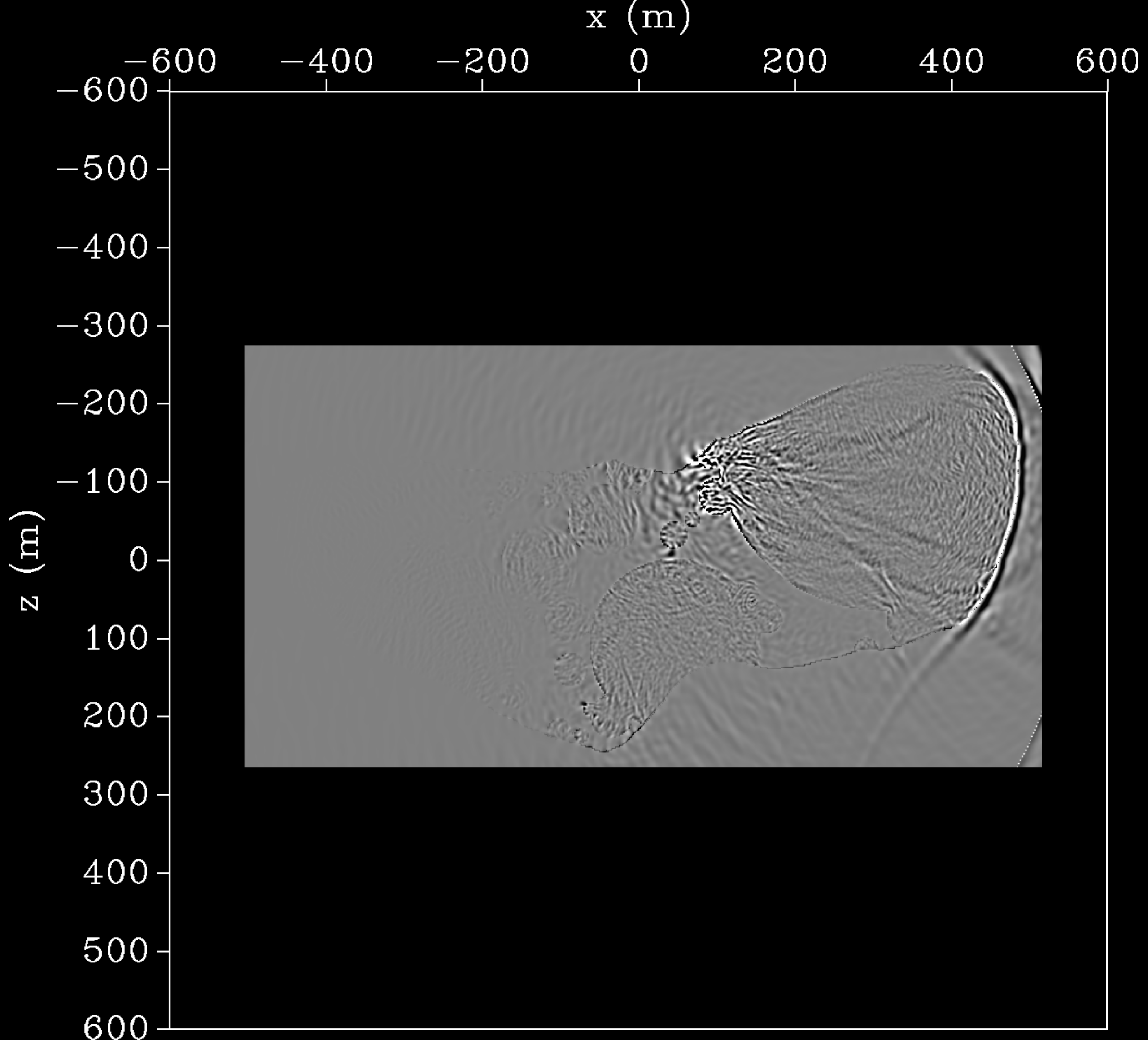
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600



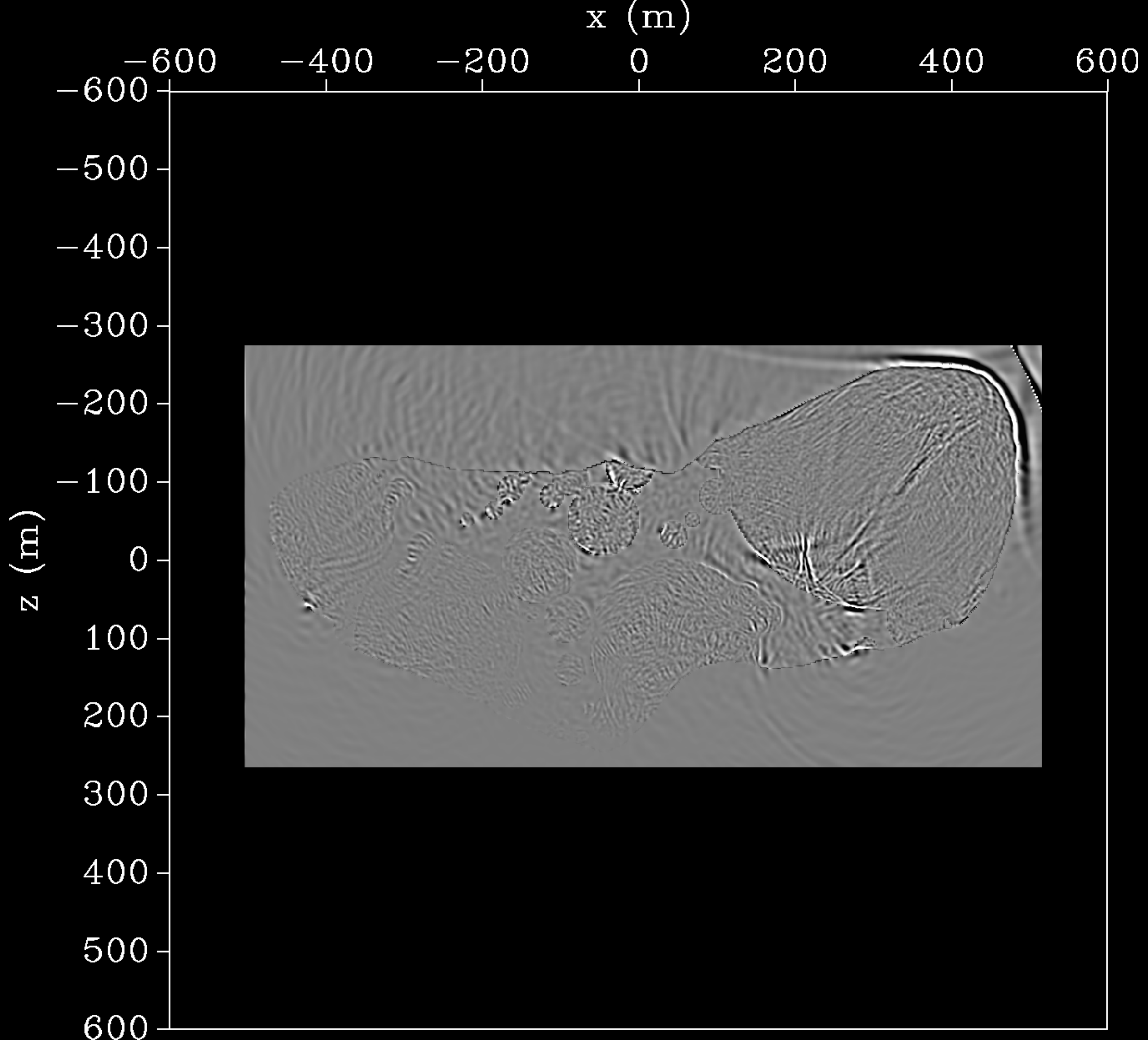
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

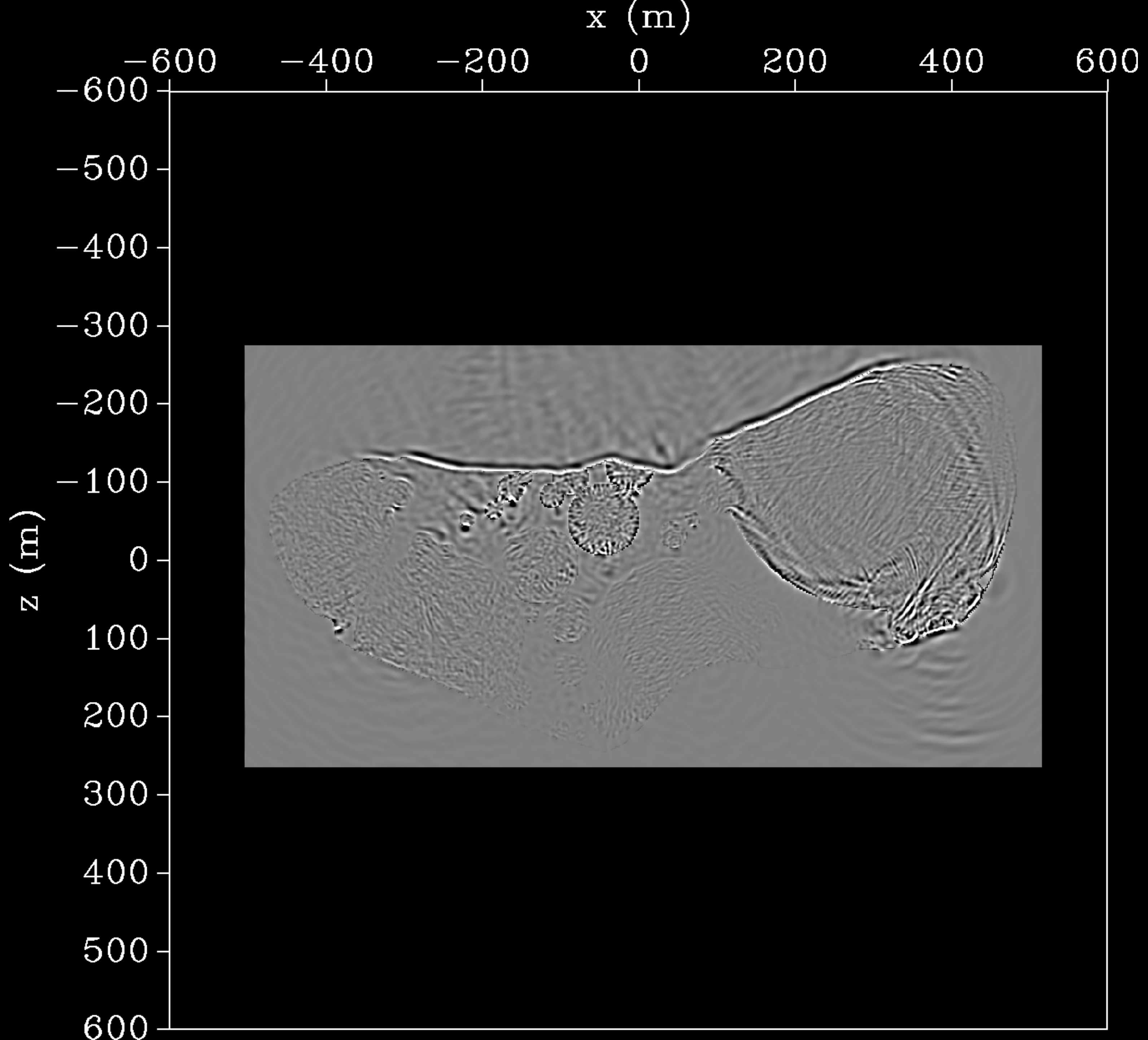

x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

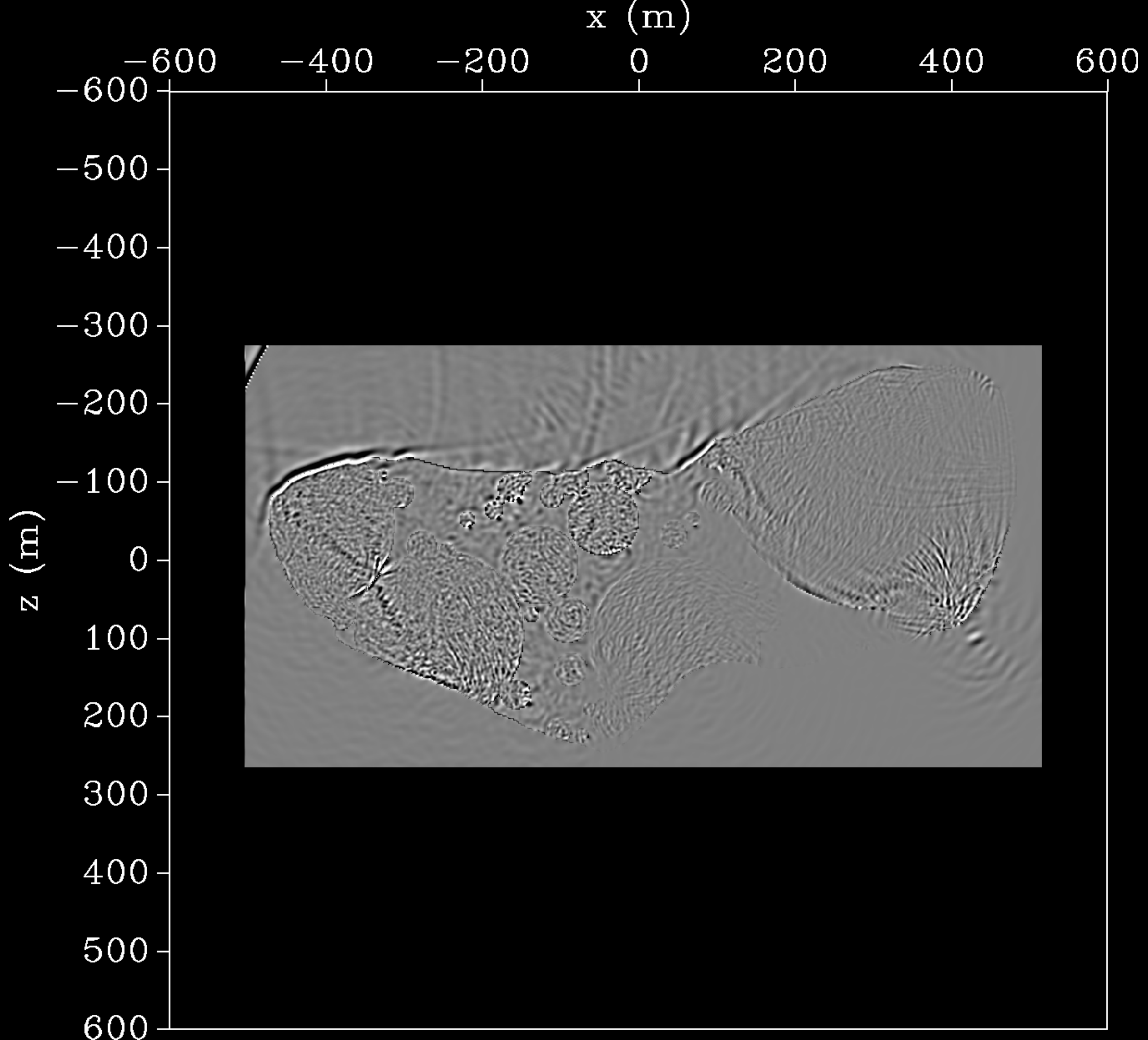


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

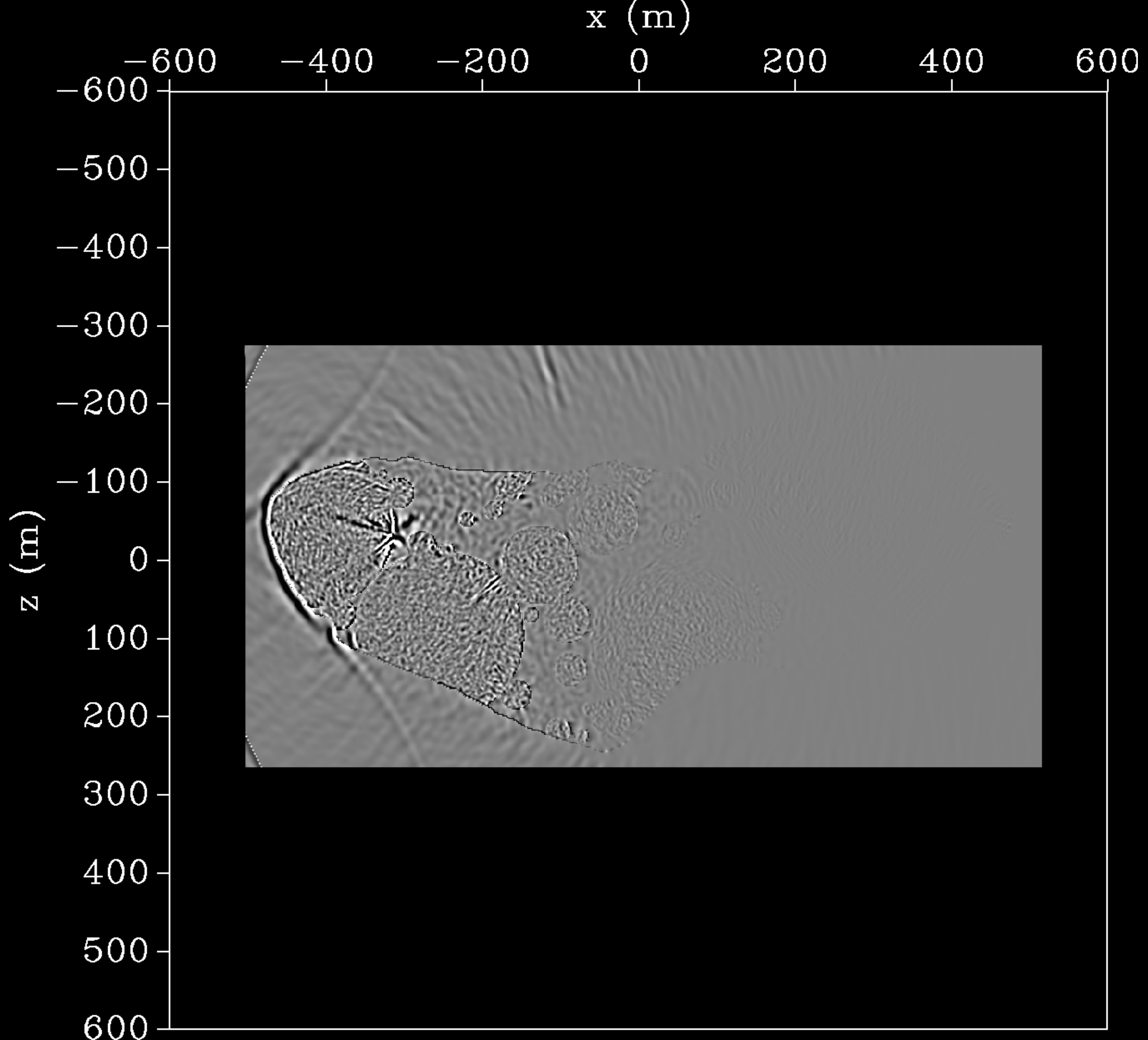

x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

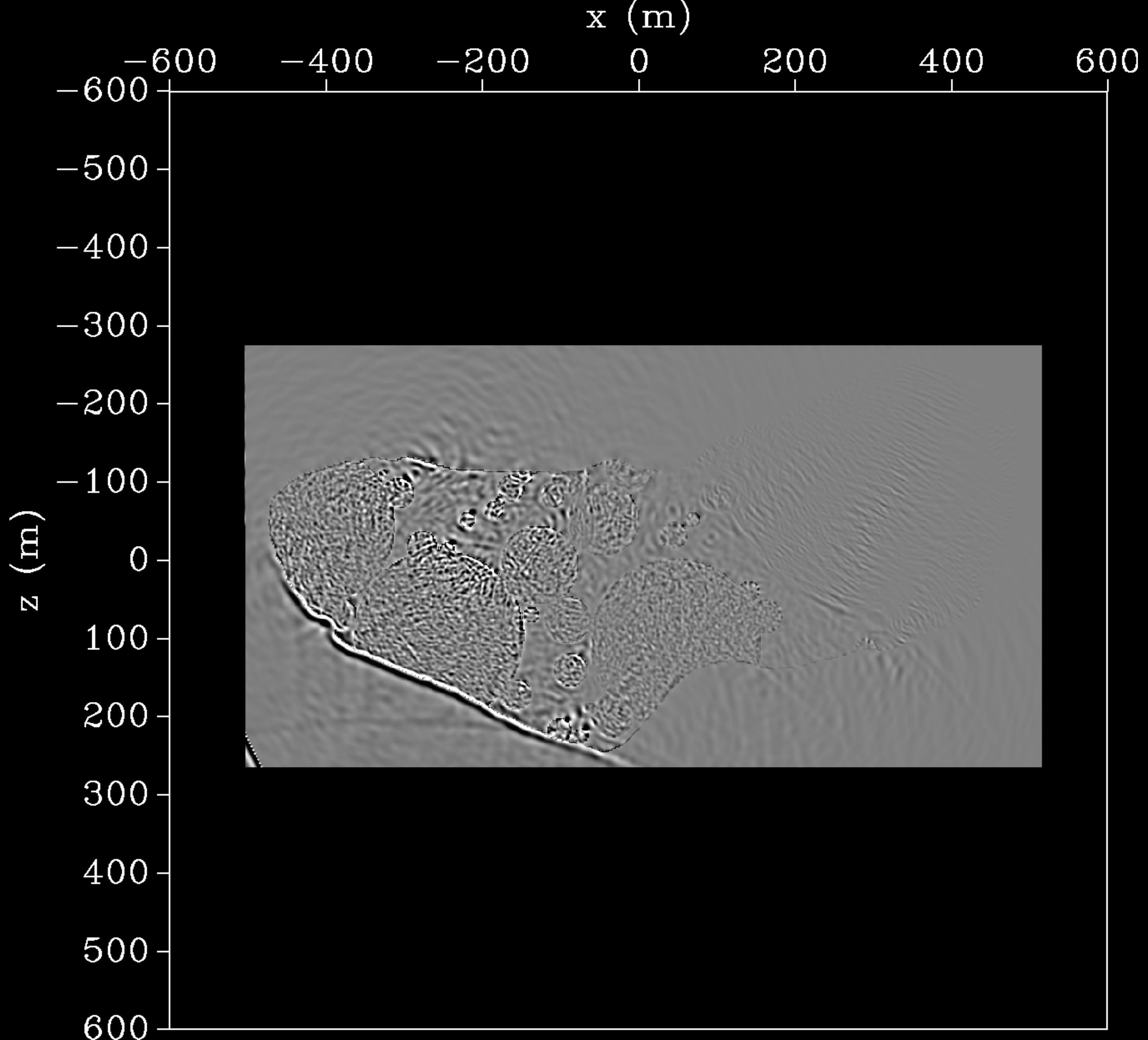


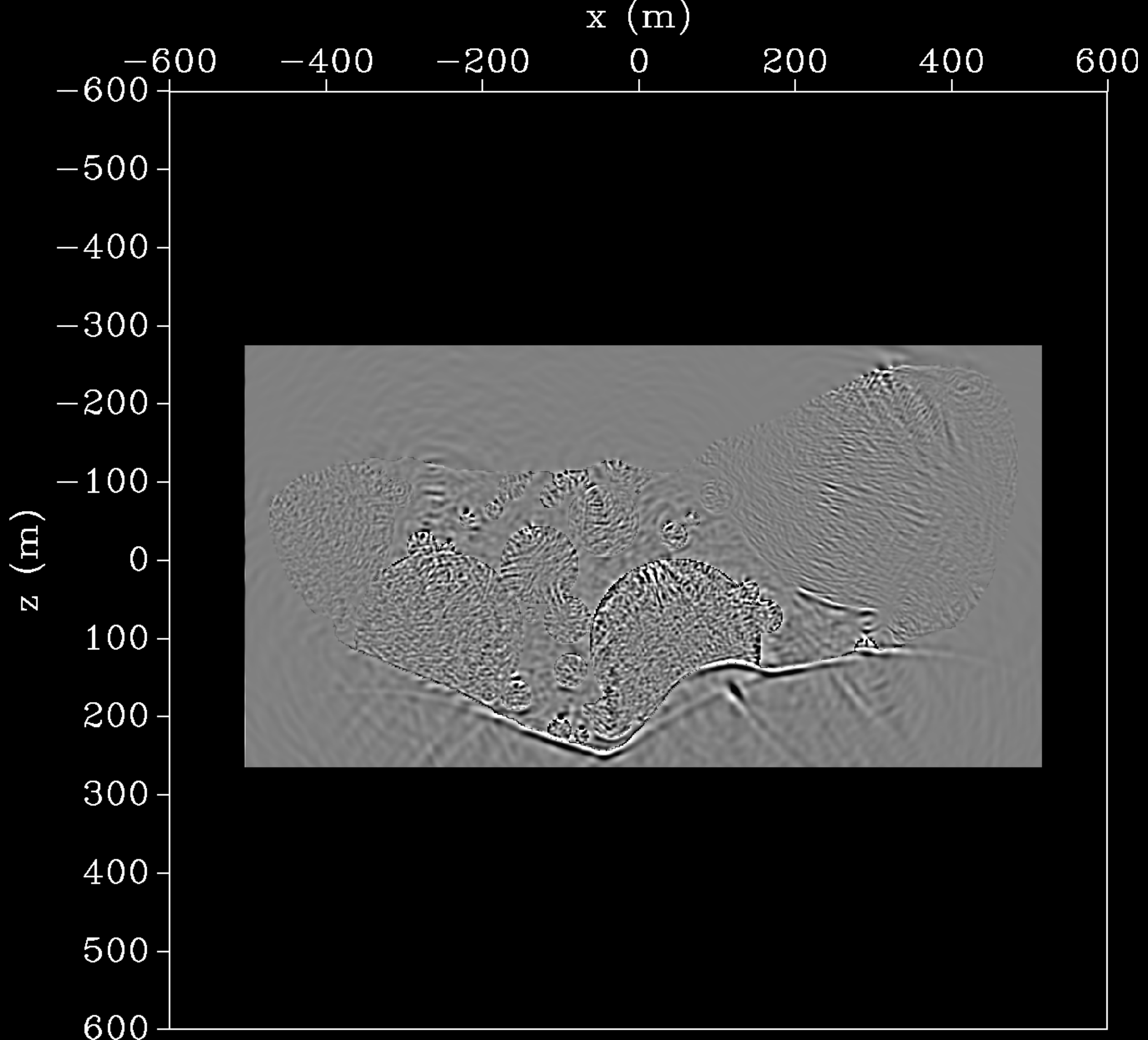
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600



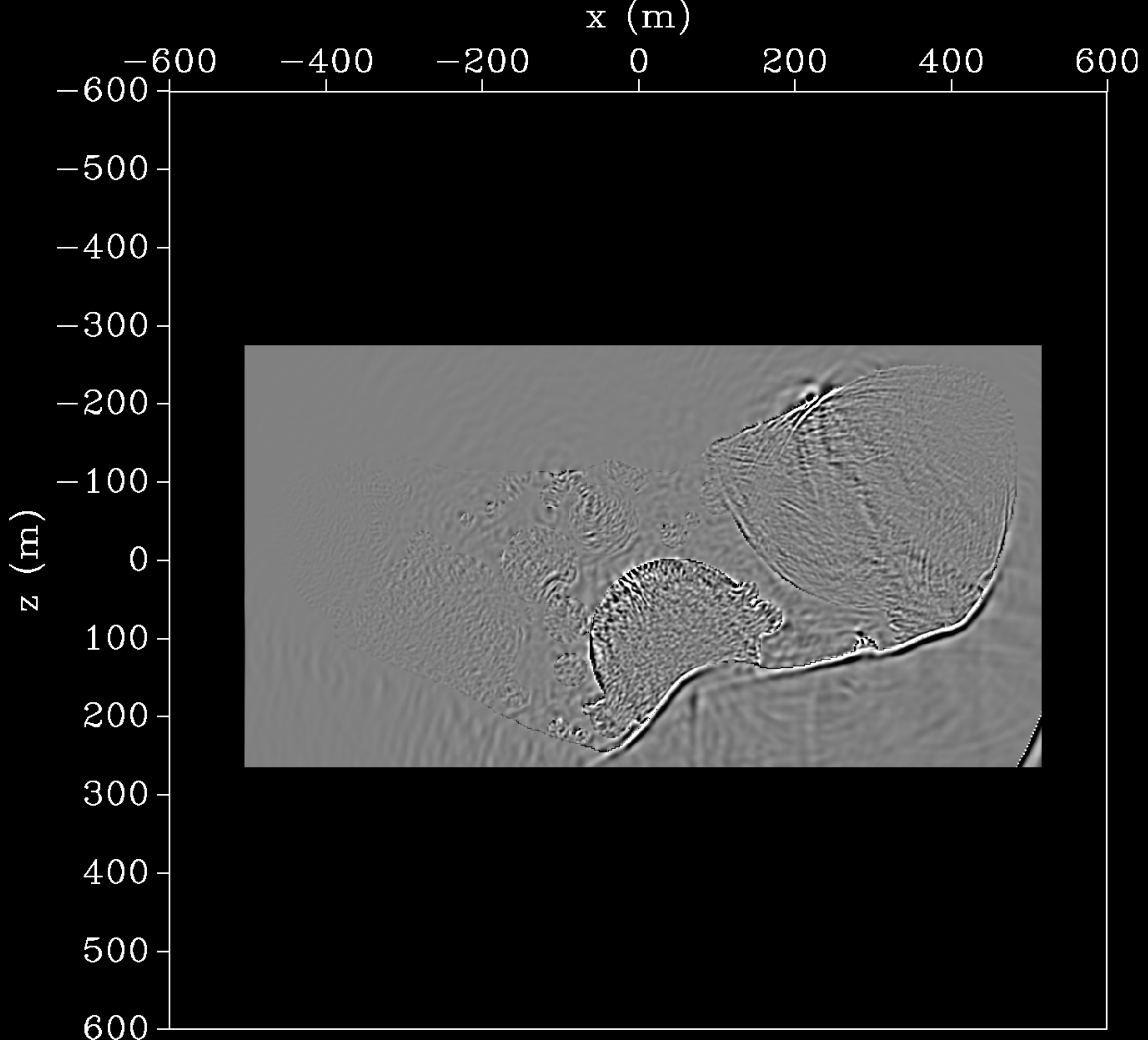
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

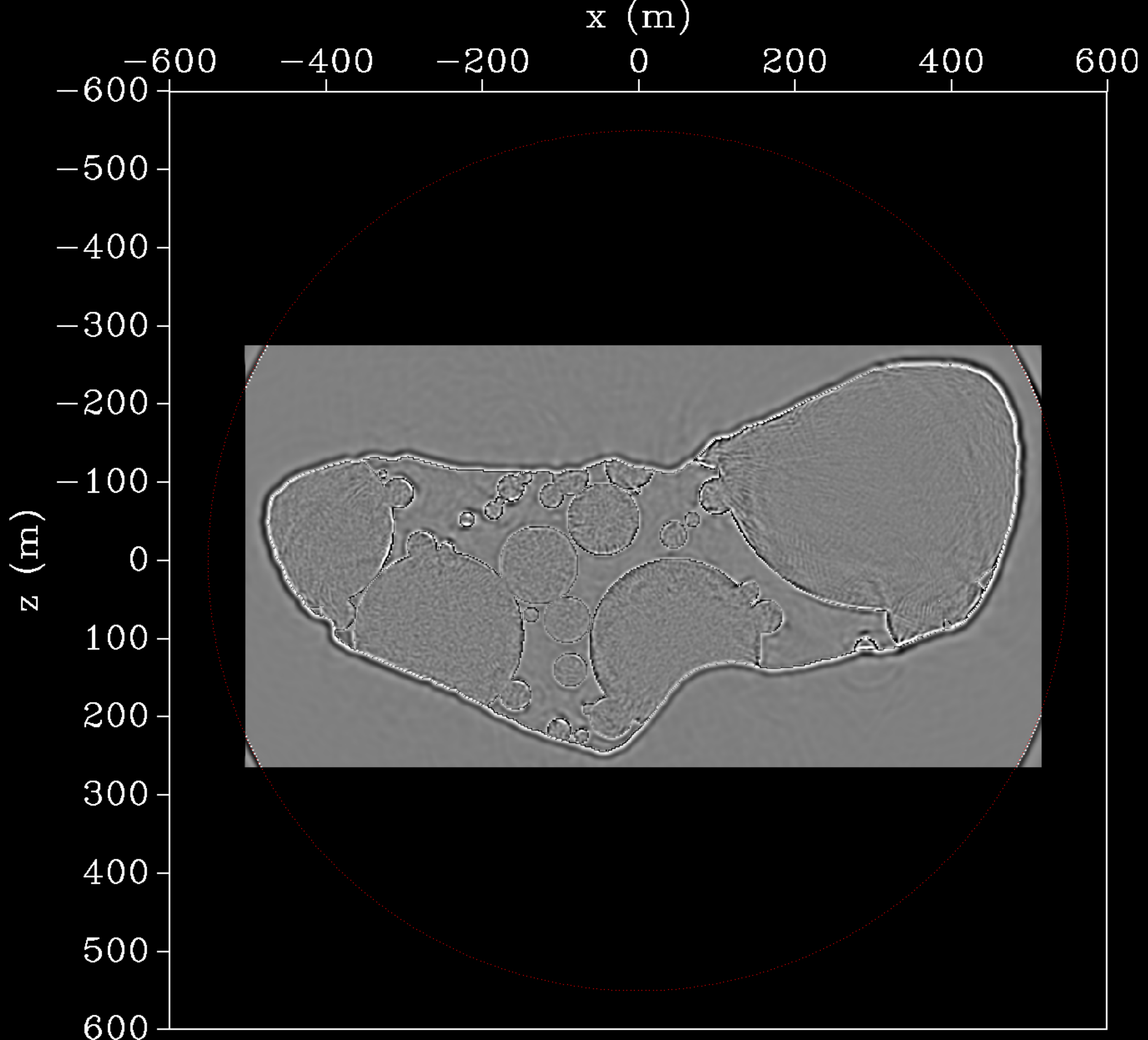

x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

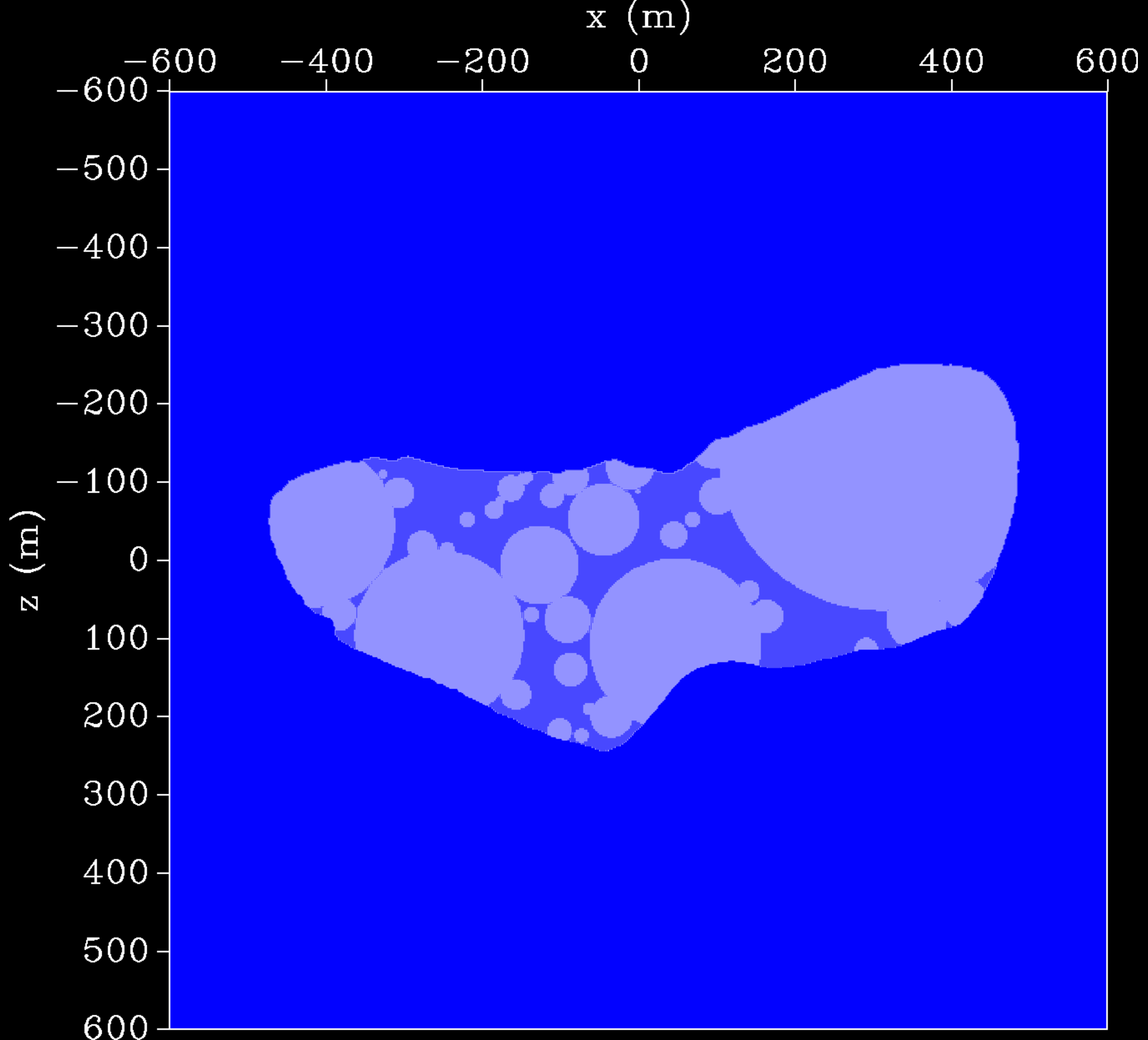


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600



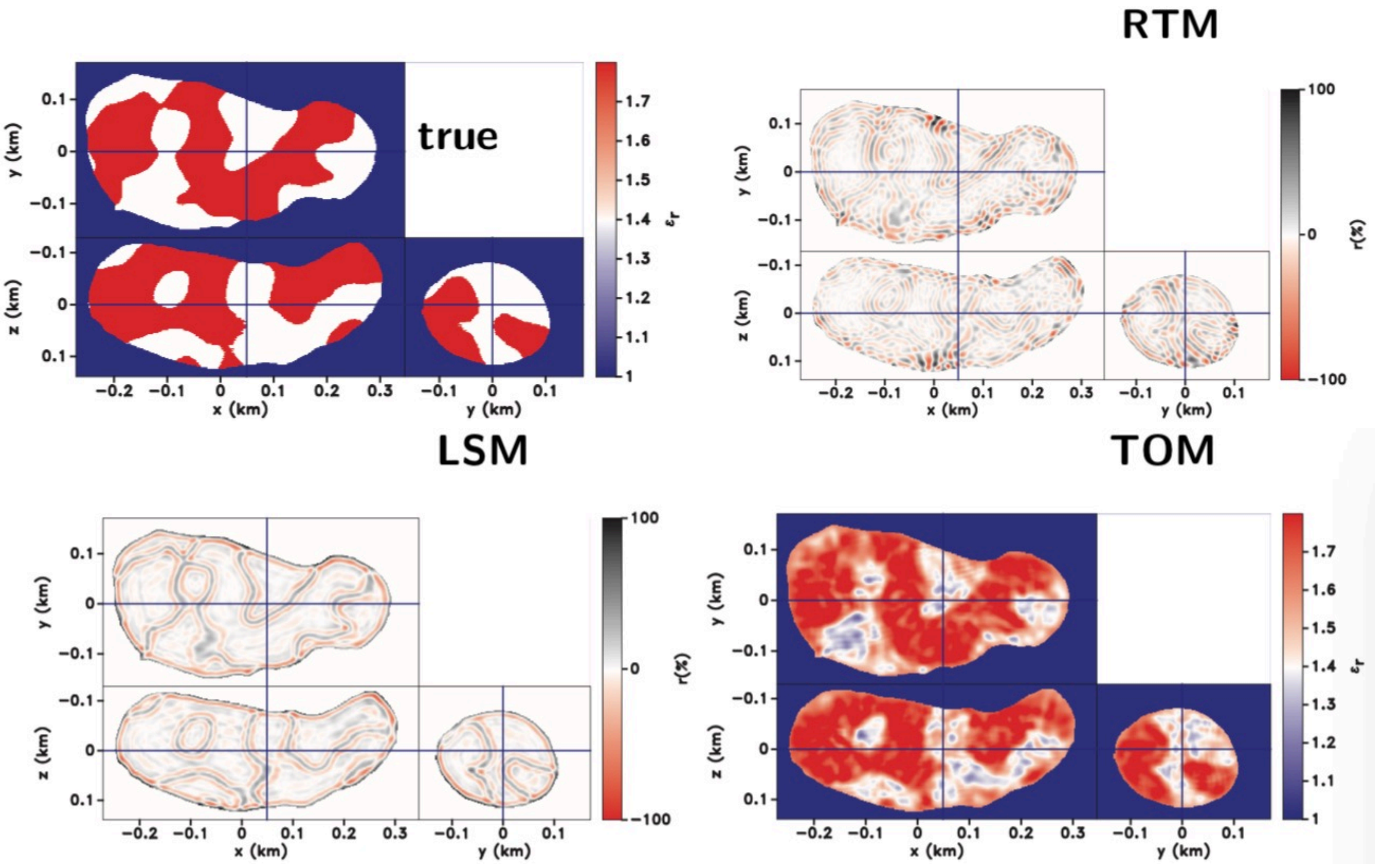
x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600


x (m)
−600
−400
−200
0
200
400
600
z (m)
−600
−500
−400
−300
−200
−100
0
100
200
300
400
500
600

小行星逆时偏移成像

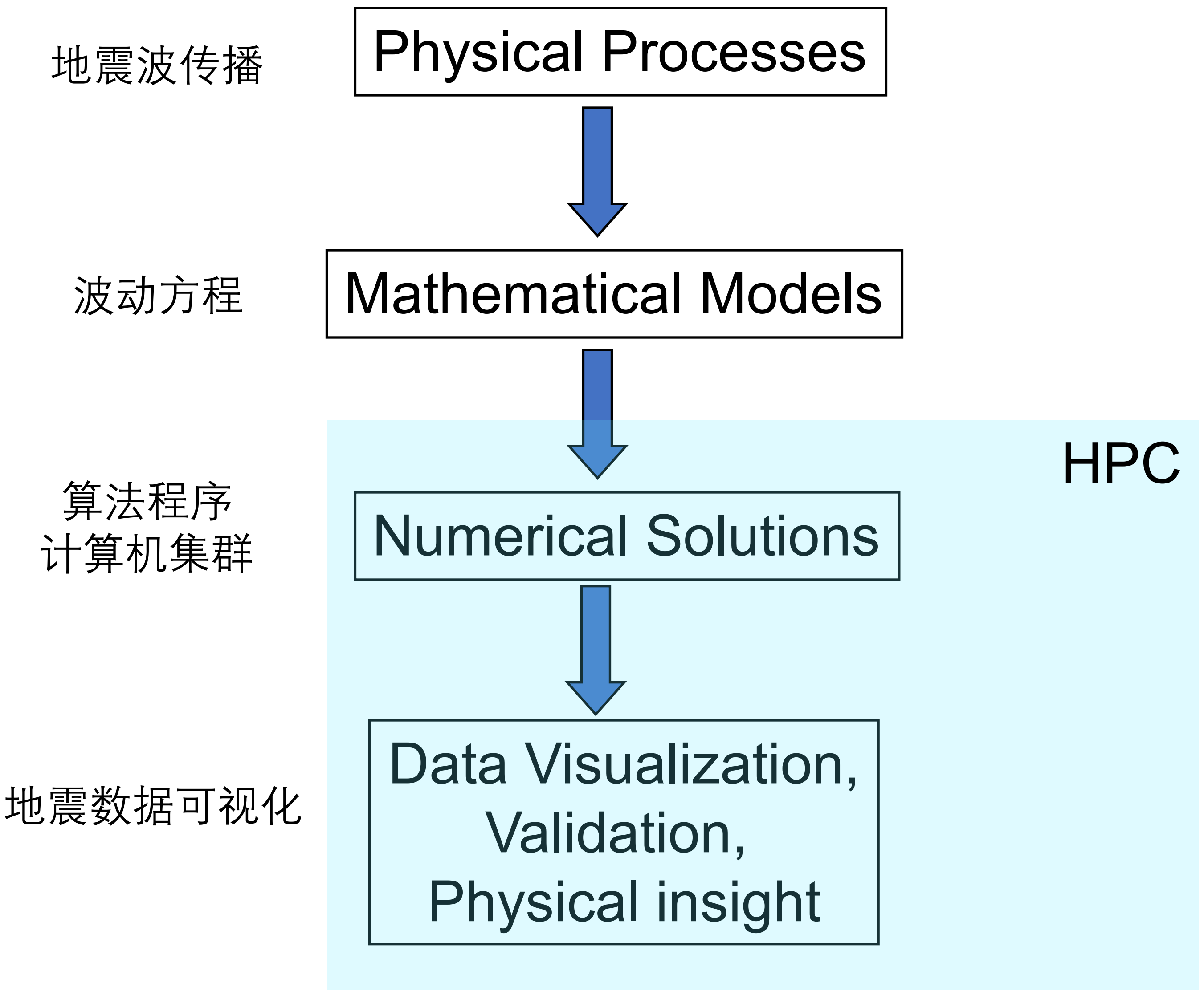
Credited by Paul Sava, 2018



提纲

1. 地震成像
2. **高性能计算**
3. 高性能地震成像实例
4. 地球物理开源研究

高性能计算



地震波传播
Physical Processes
波动方程
Mathematical Models
HPC
算法程序
计算机集群
Numerical Solutions
地震数据可视化
Data Visualization,
Validation,
Physical insight

高性能计算

Credited by Richa Rastogi, 2011

Problems	Algorithms	Parallelization	Solution	Methods
Modeling	*Solving wave equation (diff. Eq.s)*	*Dividing Problem space*	*Finite Difference Schemes with Boundary conditions*	• *2D Elastic* • *2D & 3D Acoustic*
Migration	*Solving wave equation (diff. Eq.s, integral equation)*	*Dividing data to be processed*	• *Frequency,* • *Wave number* • *Time domain*	• *Kirchhoff* • *W-X* • *PSPI*
Inversion	*Global and local optimsation & Seismic Data simulation: Forward modelling*	*Dividing search space as well as data*	• *GA and GLI* • *Forward Modeling (EI,RM,AM,* • *EM,Acoustic Modeling, TT tomography, 2D Depth migration)*	• *Real/Binary Coded GA* • *Damp. factor optimization* • *Tomographic inversion* • *Migration Vel. inversion*

高性能计算

富岳是富士通与日本理化学研究所共同开发的超级电脑。作为“京”的后继机型于2014年开始研发，于2021年正式激活。其运算能力为 442 PFLOPS，耗电为30百万瓦至40百万瓦。

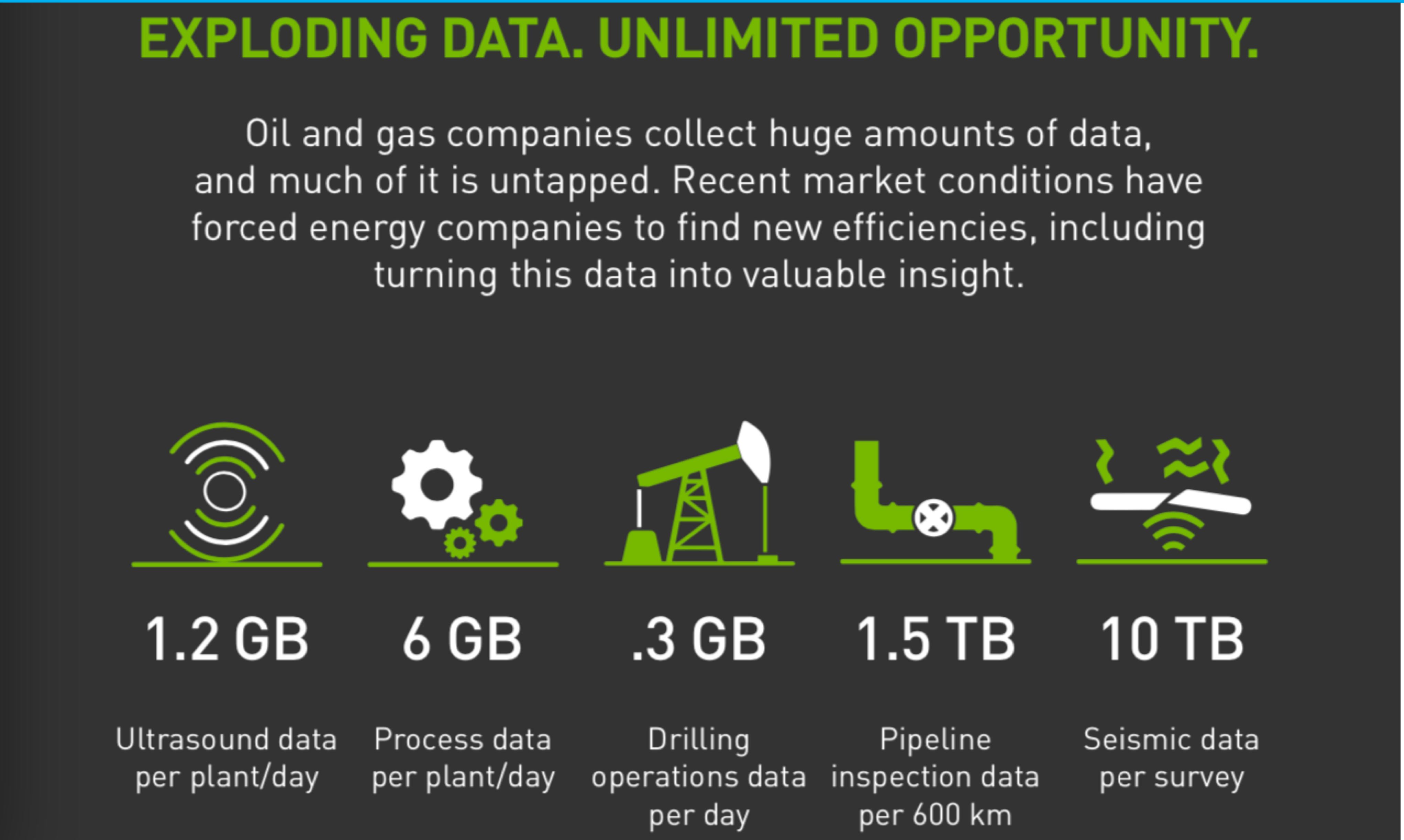
Credited by https://www.top500.org

高性能计算

Performance and cost comparison chart against computers ranked #1 in TOP500

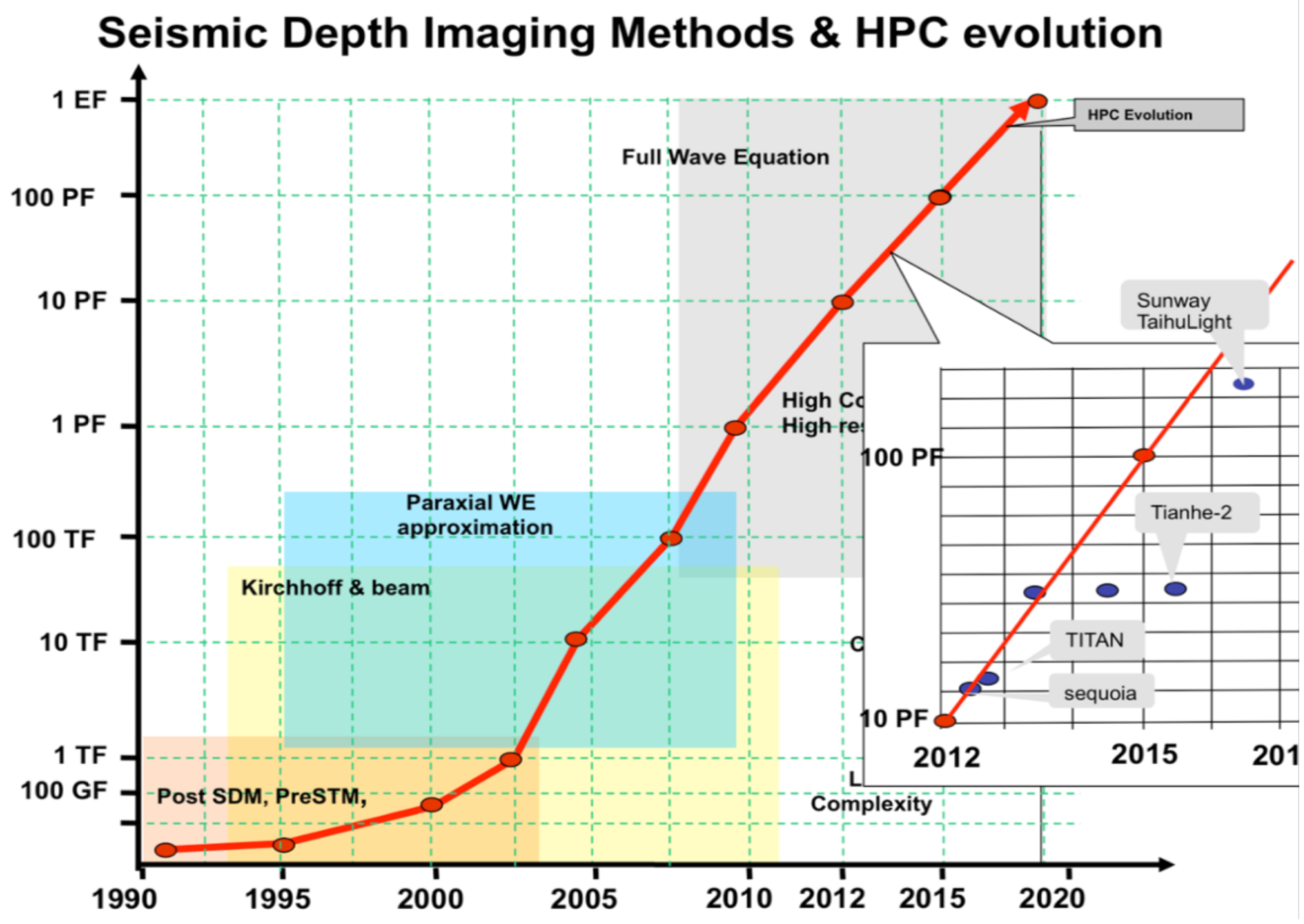
Name	Start year	End year	Performance (PFLOPS)[note 1]	Cost (million USD) (not inflation adjusted)	TOP500 ranking	CPU/GPU vendor	CPU	OS
Fugaku	2020	-	442[33]	1213[3][note 2]	June 2020 to June 2021 1st	Fujitsu	A64FX	Custom Linux-based kernel
Summit	2018	-	148	300[34]	June 2018 to November 2019 1st	IBM, NVIDIA	POWER9, Tesla	Linux (RedHat)
Sierra	2018	-	94		November 2018 to November 2019 2nd			
Sunway TaihuLight	2016	-	93	280[35]	June 2016 to November 2017 1st	NRCPC	Sunway SW26010	Linux (Raise)
K	2011	2019	10	1045[36]	June 2011 – November 2011 1st	Fujitsu	SPARC64 VIIIfx	Linux

高性能计算



Credited by NVIDIA

高性能计算



Credited by Calandra et al., 2017, Rice Oil & Gas HPC Conference

Forward and Adjoint Simulations of Seismic Wave Propagation on Emerging Large-Scale GPU Architectures

Max Rietmann*, Peter Messmer†, Tarje Nissen-Meyer‡, Daniel Peter§, Piero Basini‡, Dimitri Komatitsch¶, Olaf Schenk*, Jeroen Tromp§, Lapo Boschi‡, and Domenico Giardini‡

*USI Lugano, Institute of Computational Science
Via Giuseppe Buffi 13
CH-6900 Lugano, Switzerland
Email: [max.rietmann, olaf.schenk]@usi.ch

†NVIDIA Corp.
Zurich, Switzerland
Email: pmessmer@nvidia.com

§Princeton University,
Dept. of Geosciences, Guyot Hall
Princeton, New Jersey 08544, USA
Email: [dpeter, jtromp]@princeton.edu

‡ETH Zurich, Institute of Geophysics
Sonneggstrasse 5, CH-8092 Zurich, Switzerland
Email: tarjen@ethz.ch,
piero.basini@gmail.com,
lapo@erdw.ethz.ch,
domenico.giardini@sed.ethz.ch

¶CNRS Marseille
Laboratory of Mechanics and Acoustics
31 chemin Joseph Aiguier
13402 Marseille cedex 20, France
Email: komatitsch@lma.cnrs-mrs.fr

高性能计算

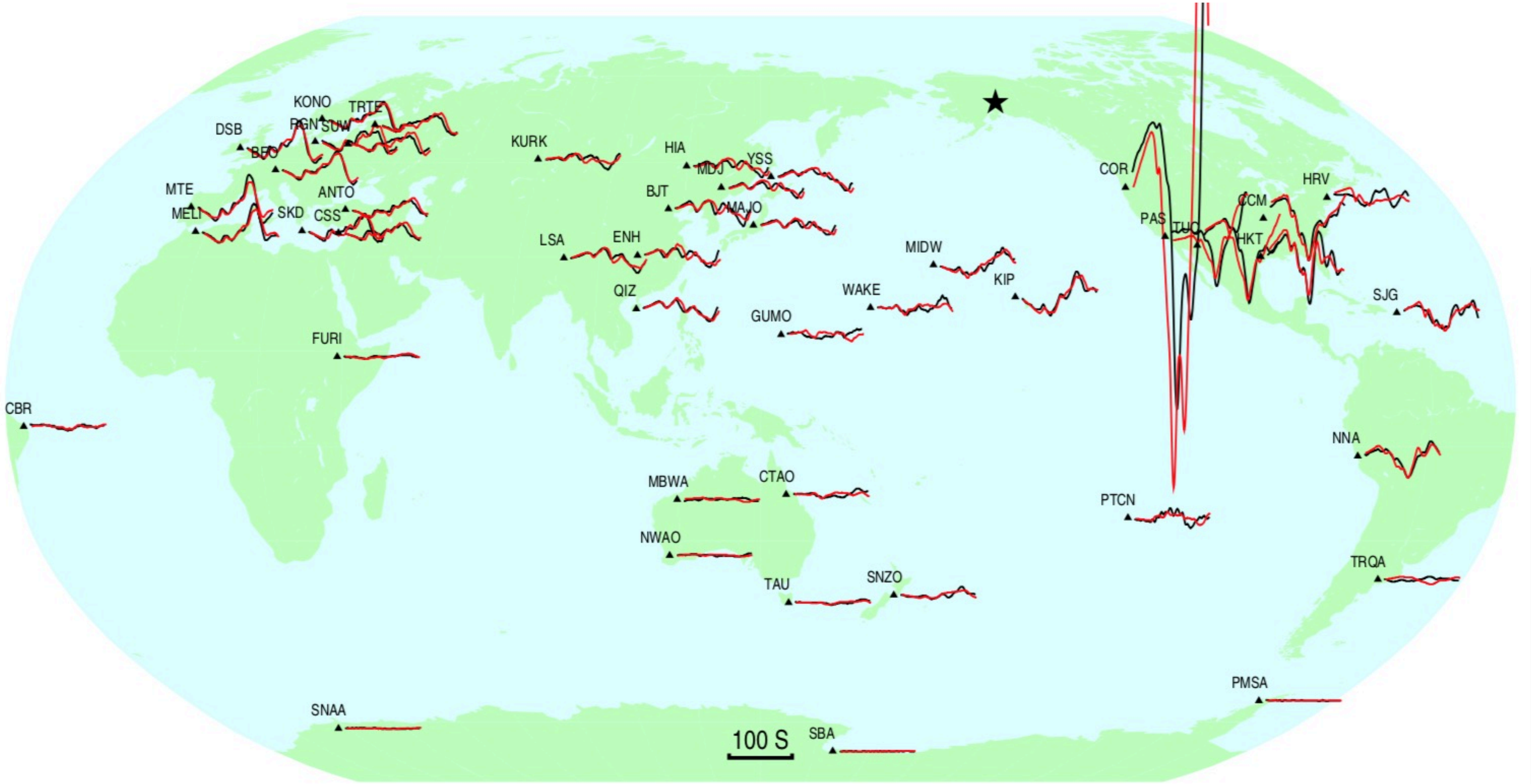
ACM Gordon Bell Prize Winner, 2003

A 14.6 billion degrees of freedom, 5 teraflops, 2.5 terabyte earthquake simulation on the Earth Simulator

Dimitri Komatitsch
Caltech
Pasadena, California, USA

Seiji Tsuboi*
IFREE, JAMSTEC
Yokohama, Japan

Chen Ji & Jeroen Tromp
Caltech
Pasadena, California, USA



高性能计算

ACM Gordon Bell Prize Winner, 2017

18.9-Pflops Nonlinear Earthquake Simulation on Sunway TaihuLight: Enabling Depiction of 18-Hz and 8-Meter Scenarios

Haohuan Fu
Tsinghua University
National Supercomputing Center in Wuxi, China
haohuan@tsinghua.edu.cn

Conghui He
Tsinghua University
National Supercomputing Center in Wuxi, China
hch13@mails.tsinghua.edu.cn

Bingwei Chen
Tsinghua University
National Supercomputing Center in Wuxi, China
cbw15@mails.tsinghua.edu.cn

Zekun Yin
Shandong University
zekun.yin@mail.sdu.edu.cn

Zhenguo Zhang
Southern University of Science and Technology, China
zhangzg@sustc.edu.cn

Wenqiang Zhang
University of Science and Technology of China
wqzhang1@mail.ustc.edu.cn

Tingjian Zhang
Shandong University
sdubeyhhhh@gmail.com

Wei Xue
Tsinghua University
National Supercomputing Center in Wuxi, China
xuewei@tsinghua.edu.cn

Weiguo Liu
Shandong University
weiguo.liu@sdu.edu.cn

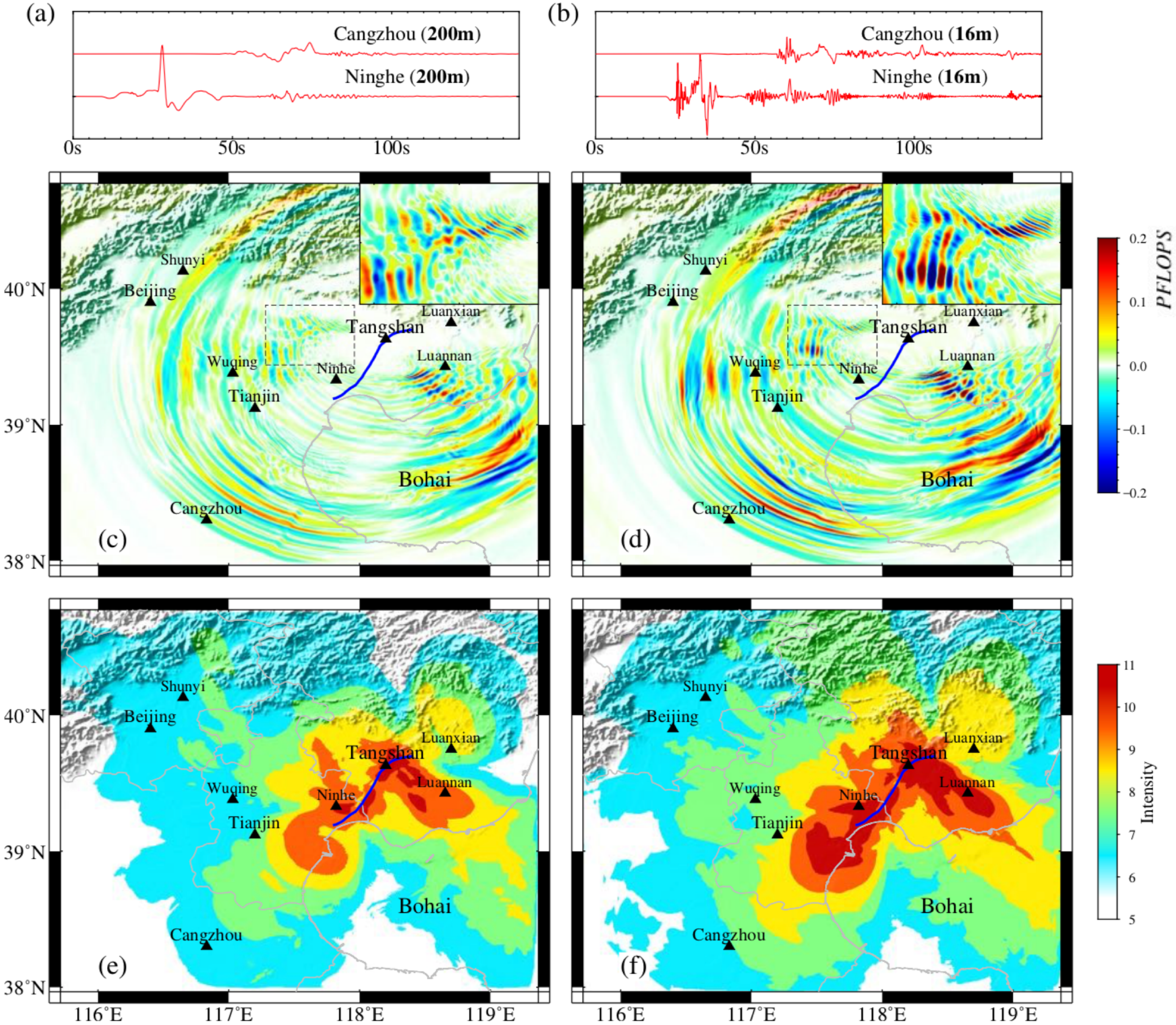
Wanwang Yin
National Research Center of Parallel Computer Engineering and Technology, China
yinwanwang@gmail.com

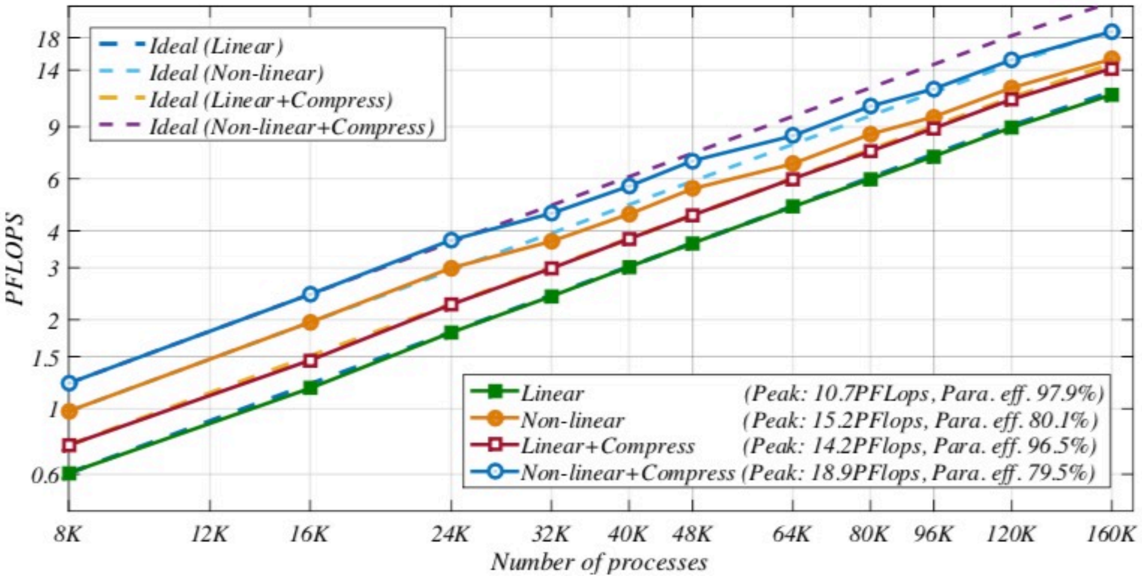
Guangwen Yang
Tsinghua University
National Supercomputing Center in Wuxi, China
ygw@tsinghua.edu.cn

Xiaofei Chen
Southern University of Science and Technology, China
chenxf@sustc.edu.cn

高性能计算

ACM Gordon Bell Prize Winner, 2017





18.9-Pflops non-linear earthquake simulation on Sunway TaihuLight (15% of the peak), using 10,400,000 cores, for 18-Hz and 8-meter scenarios. Size: 320 km by 312 km by 40 km.

Flops: floating-point operations per second

1 Pflops = 10^{15} flops

高性能计算

ACM Gordon Bell Prize Winner, 2017

“超算的计算资源十分宝贵，一天下来电费就要好几十万元，所以我们年轻人都是连轴转，最多轮换着休息一会儿。机器在那里，不用就浪费了。”

—— 付昊桓

提纲

1. 地震成像
2. 高性能计算
3. **高性能地震成像实例**
4. 地球物理开源研究

粘滞声波地震成像实例

Constant-Q viscoacoustic wave equation with DFLs was first proposed by Zhu and Harris (2014) as follows

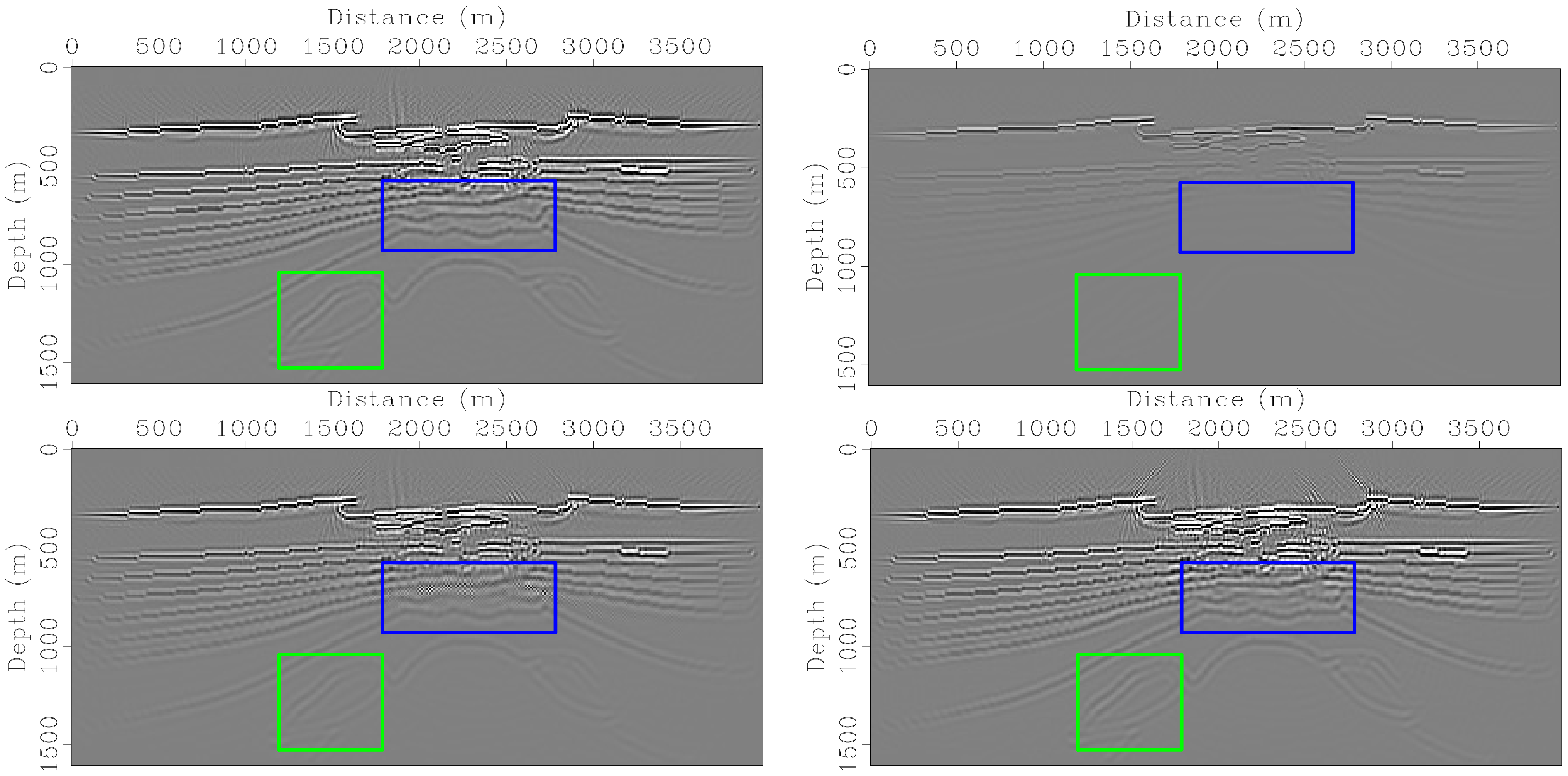
$$\begin{cases} \dfrac{1}{c^2(\mathbf{x})}\dfrac{\partial^2 p}{\partial t^2}(\mathbf{x},t) - \eta(-\nabla^2)^{\gamma+1} p(\mathbf{x},t) - \tau\dfrac{\partial}{\partial t}(-\nabla^2)^{\gamma+1/2} p(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_s) f(t), \\ p(\mathbf{x},t) = \dfrac{\partial p}{\partial t}(\mathbf{x},t) = 0, \qquad \mathbf{x}\in\Omega, t<0. \end{cases}$$

The compensated wave equation with DFLs is obtained by reversing the absorption term in sign but leaving the dispersion parameter unchanged.

$$\begin{cases} \dfrac{1}{c^2(\mathbf{x})}\dfrac{\partial^2 p_s}{\partial t^2}(\mathbf{x},t) - \eta(-\nabla^2)^{\gamma+1} p_s(\mathbf{x},t) + \tau\dfrac{\partial}{\partial t}(-\nabla^2)^{\gamma+1/2} p_s(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_s) f(t), \\ p_s(\mathbf{x},t) = \dfrac{\partial p_s}{\partial t}(\mathbf{x},t) = 0, \qquad \mathbf{x}\in\Omega, t<0. \end{cases}$$

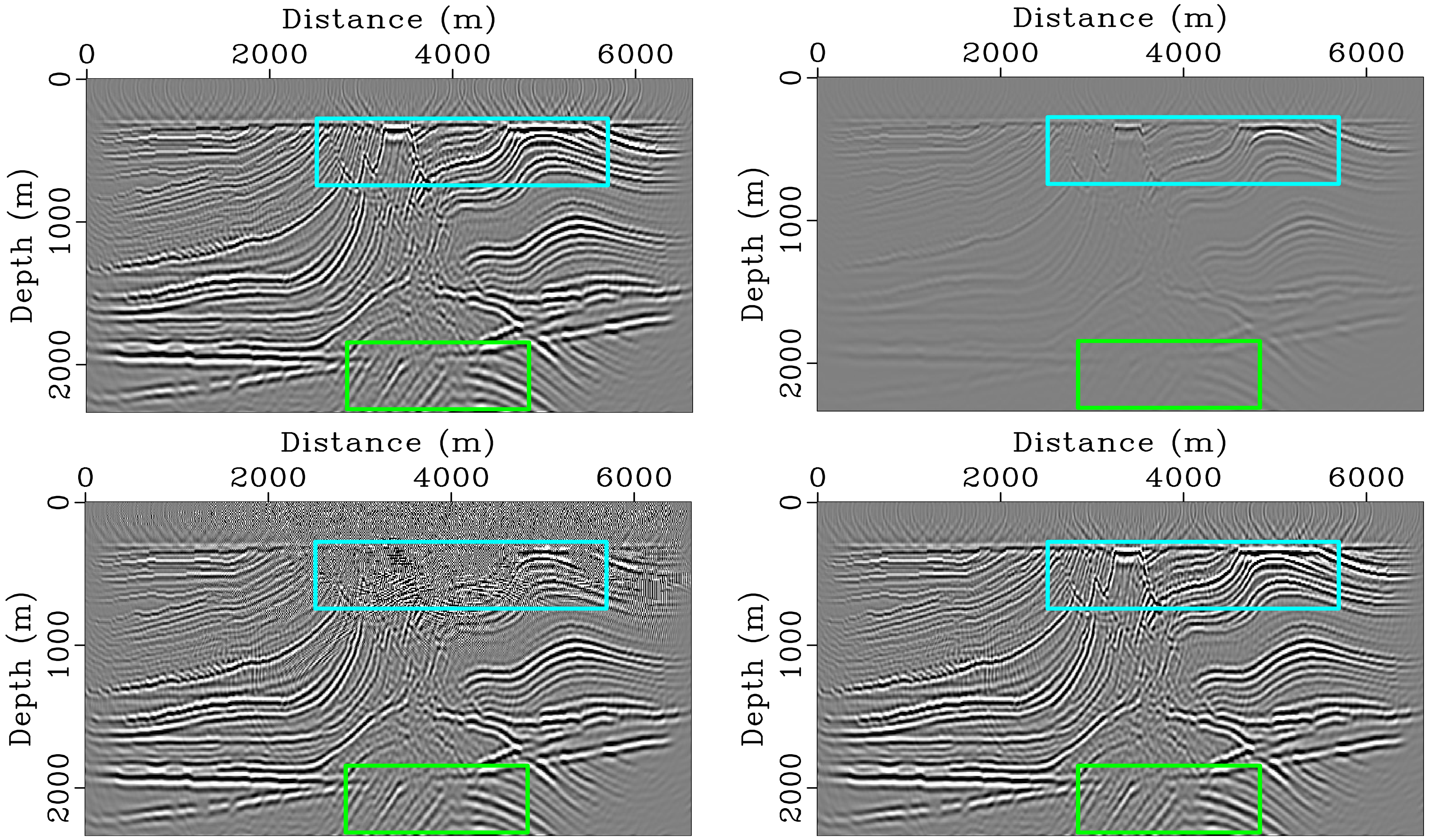
[Wang et al., 2018, Geophysics]

粘滞声波地震成像实例



[Wang et al., 2018, Geophysics]

粘滞声波地震成像实例



[Wang et al., 2018, Geophysics]

粘滞弹性波地震成像实例

The constant-*Q* viscoelastic wave equation with DFLs in a first-order matrix form:

$$\partial_t \mathbf{u}^{\Diamond} = \mathbf{H}\mathbf{u}^{\Diamond} + \mathbf{f},$$
$$\mathbf{u}^{\Diamond} = (v_x, v_z, \sigma_{xx}, \sigma_{zz}, \sigma_{xz})^T$$
$$\mathbf{f} = (f_x, f_z, 0, 0, 0)^T$$

$$\mathbf{H} = \begin{pmatrix} 0 & 0 & 1/\rho\partial_x & 0 & 1/\rho\partial_z \\ 0 & 0 & 0 & 1/\rho\partial_z & 1/\rho\partial_x \\ (\bar{\lambda}+2\bar{\mu})\partial_x & \bar{\lambda}\partial_z & 0 & 0 & 0 \\ \bar{\lambda}\partial_x & (\bar{\lambda}+2\bar{\mu})\partial_z & 0 & 0 & 0 \\ \bar{\mu}\partial_z & \bar{\mu}\partial_x & 0 & 0 & 0 \end{pmatrix}$$

$$\bar{\mu} = \eta_s \mathbf{D}_s + \tau_s \mathbf{A}_s \partial_t,$$
$$\bar{\lambda} = \left(\eta_p \mathbf{D}_p + \tau_p \mathbf{A}_p \partial_t\right) - 2\left(\eta_s \mathbf{D}_s + \tau_s \mathbf{A}_s \partial_t\right).$$

There are two fractional Laplacians in these coefficients, i.e.,

$$\begin{cases} \mathbf{D}_m = (-\nabla^2)^{\gamma_m}, \\ \mathbf{A}_m = (-\nabla^2)^{\gamma_m - 1/2}. \end{cases}$$

We conduct P- and S-wavefield decomposition

$$\partial_t \mathbf{u}^{\circ} = \mathbf{W}\mathbf{u}^{\bullet},$$
$$\mathbf{u}^{\circ} = (\sigma_p, v_{xp}, v_{zp}, \sigma_s, v_{xs}, v_{zs})^T$$
$$\mathbf{u}^{\bullet} = (v_x, v_z, \sigma_p, \sigma_s)^T$$

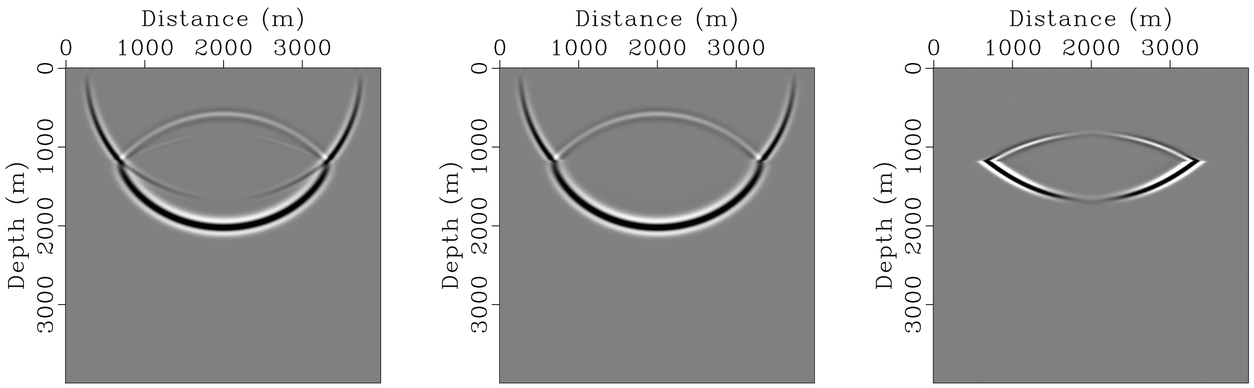
$$\mathbf{W} = \begin{pmatrix} (\bar{\lambda}+2\bar{\mu})\partial_x & \bar{\lambda}\partial_z & 0 & 0 \\ 0 & 0 & 1/\rho\partial_x & 0 \\ 0 & 0 & 1/\rho\partial_z & 0 \\ \bar{\mu}\partial_z & -\bar{\mu}\partial_x & 0 & 0 \\ 0 & 0 & 0 & 1/\rho\partial_z \\ 0 & 0 & 0 & -1/\rho\partial_x \end{pmatrix}.$$

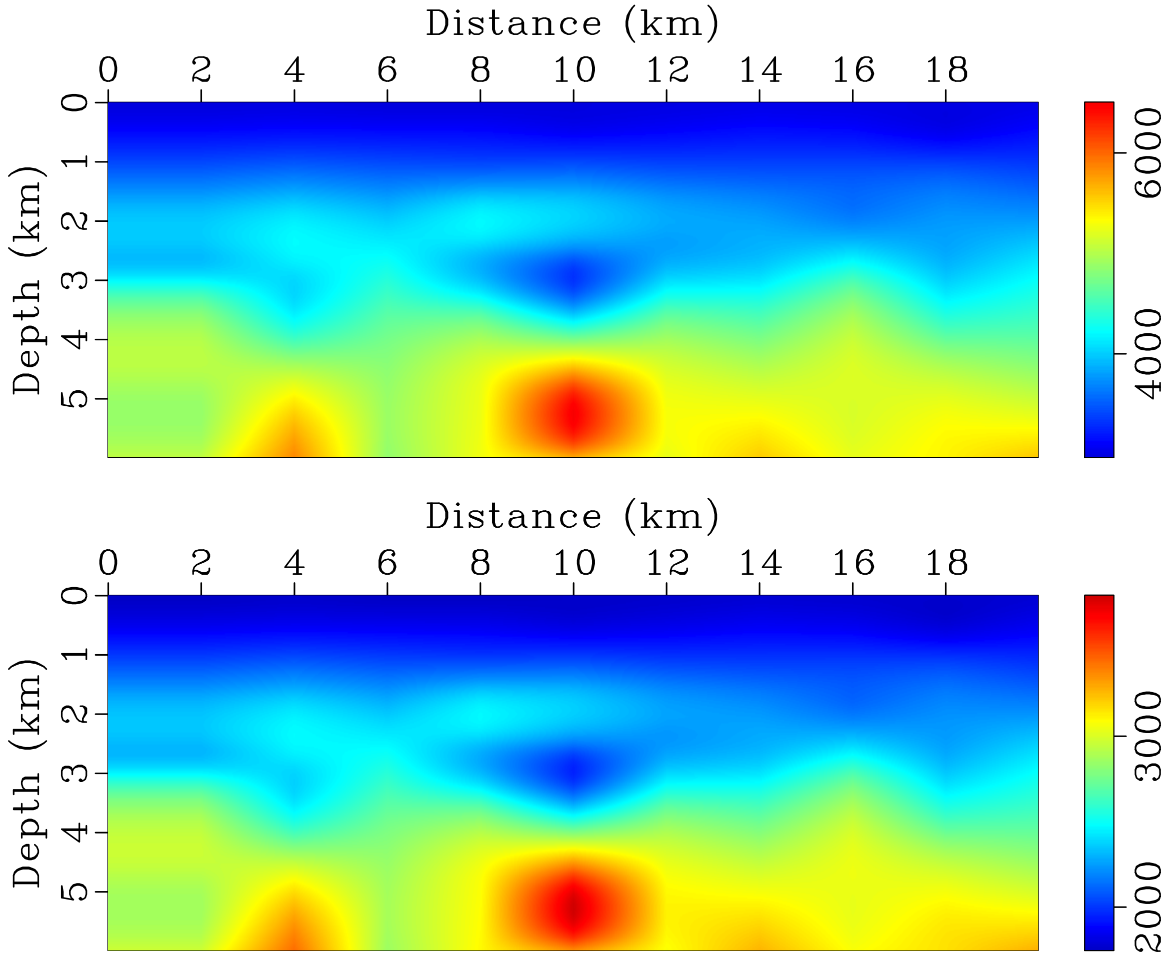
[Wang et al., 2019, Geophysics]

粘滞弹性波地震成像实例

We adopt source-normalized crosscorrelation imaging conditions for vector-based viscoelastic imaging

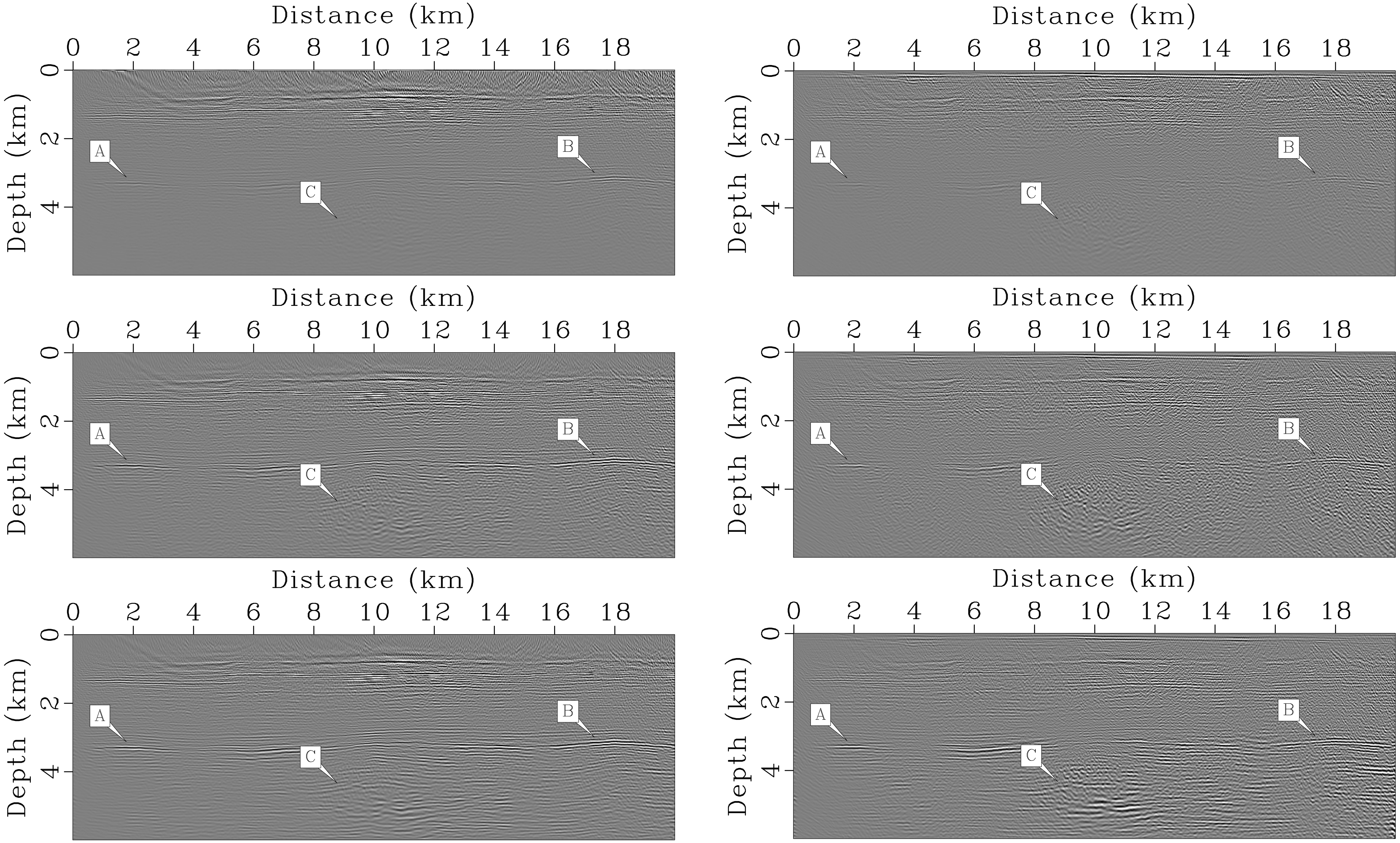
$$\begin{cases} I_{pp}(\mathbf{x}) = \dfrac{\int_0^T \mathbf{v}_p^{S^a}(\mathbf{x},t)\cdot \mathbf{v}_p^{R^c}(\mathbf{x},t)}{\int_0^T \mathbf{v}_p^{S^a}(\mathbf{x},t)\cdot \mathbf{v}_p^{S^a}(\mathbf{x},t)}\,dt, \\ I_{ps}(\mathbf{x}) = \dfrac{\int_0^T \mathbf{v}_p^{S^a}(\mathbf{x},t)\cdot \mathbf{v}_s^{R^c}(\mathbf{x},t)}{\int_0^T \mathbf{v}_p^{S^a}(\mathbf{x},t)\cdot \mathbf{v}_p^{S^a}(\mathbf{x},t)}\,dt. \end{cases}$$





[Wang et al., 2019, Geophysics]

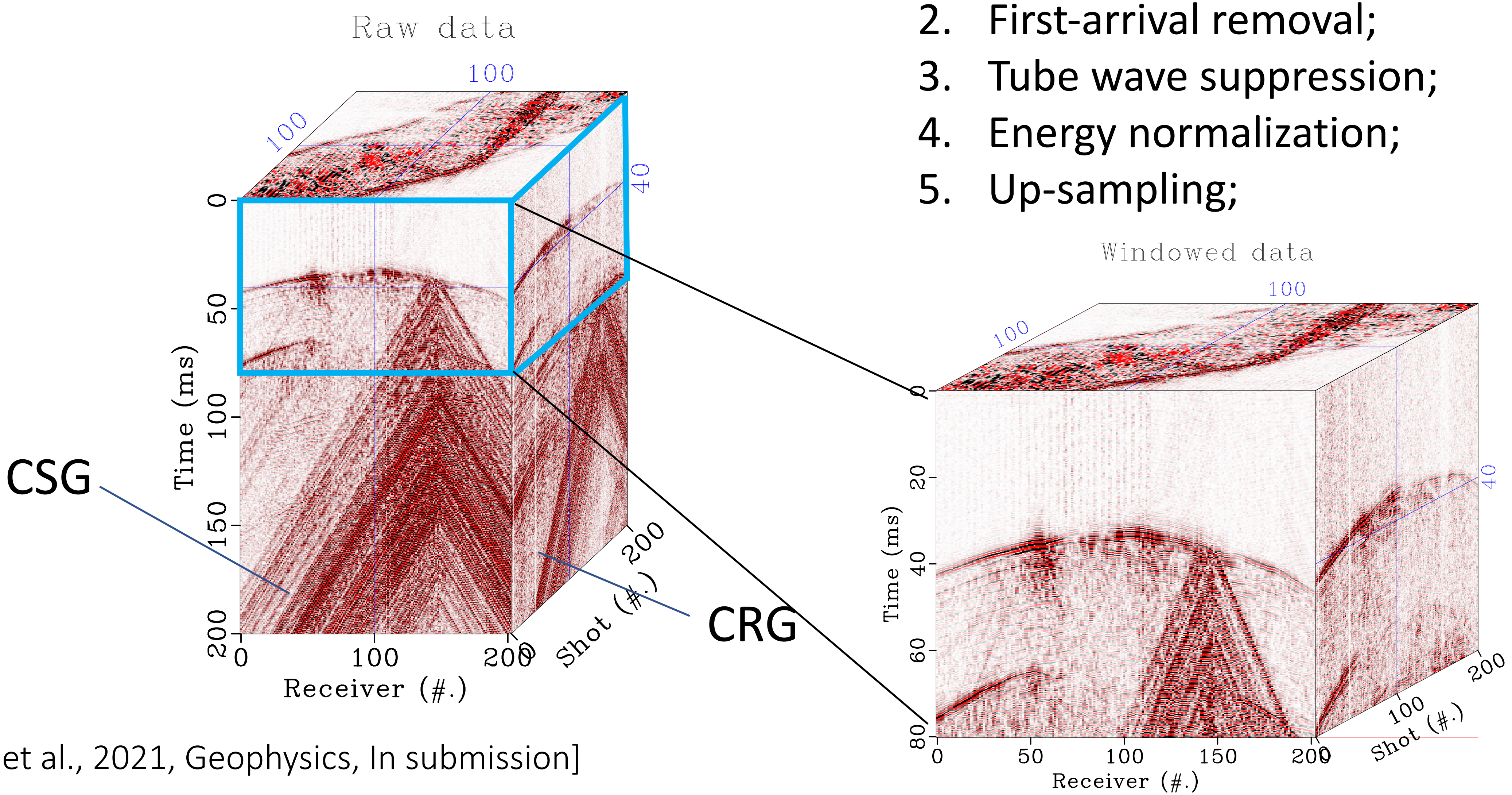
粘滞弹性波地震成像实例



[Wang et al., 2019, Geophysics]

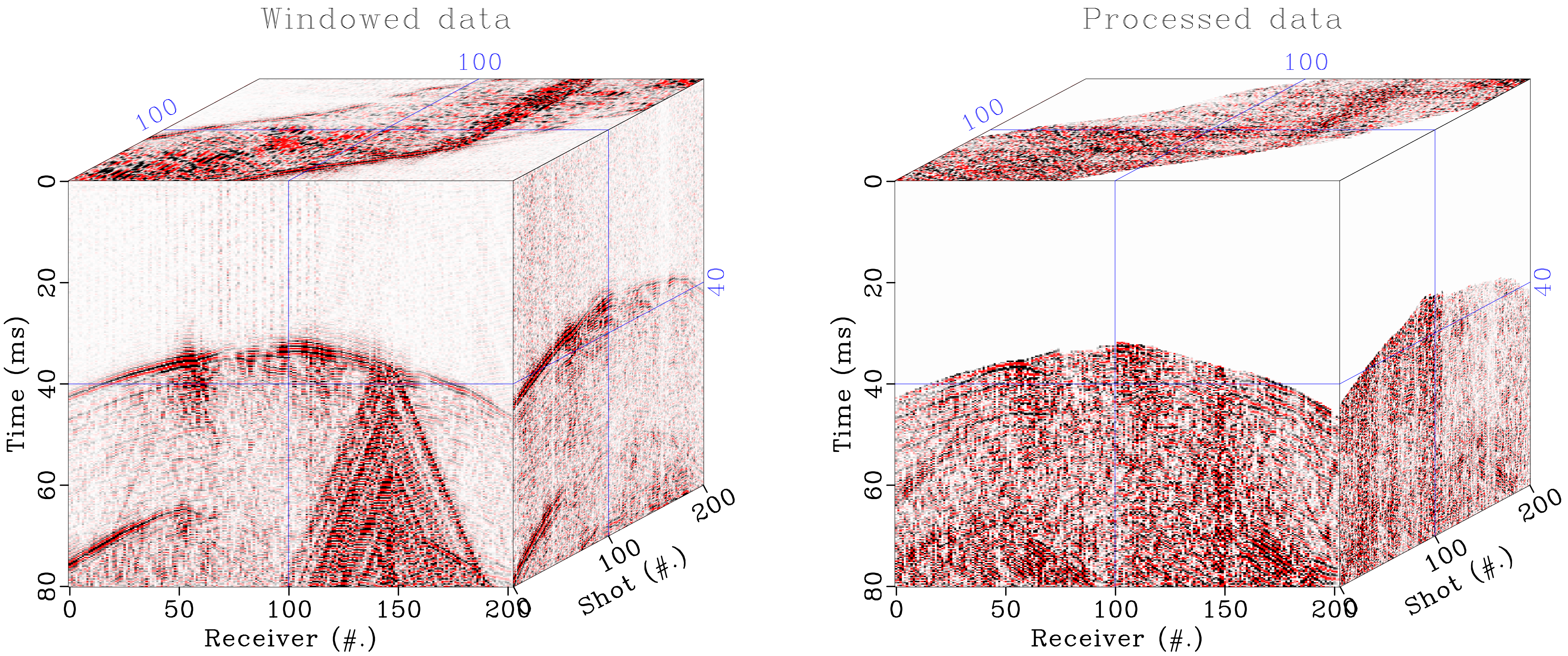
井间地震成像实例

1. Windowing;
2. First-arrival removal;
3. Tube wave suppression;
4. Energy normalization;
5. Up-sampling;



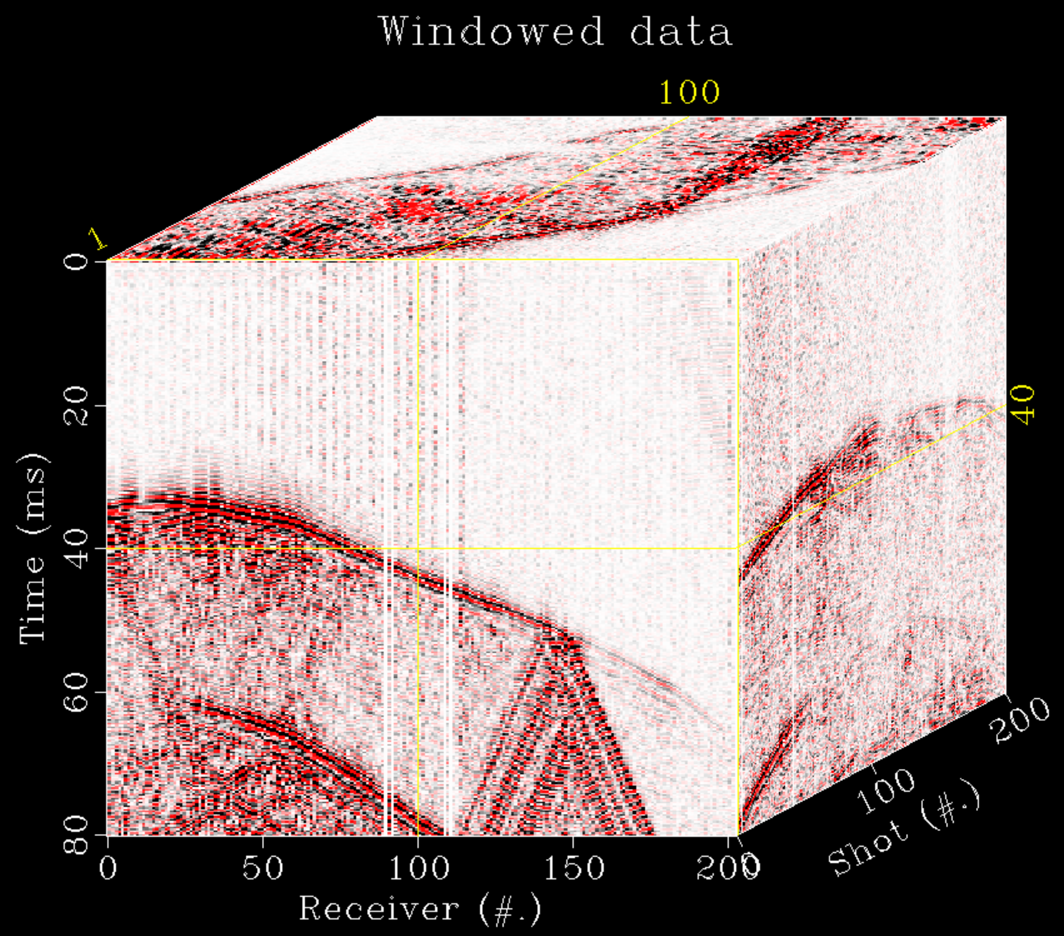
[Wang et al., 2021, Geophysics, In submission]

井间地震成像实例

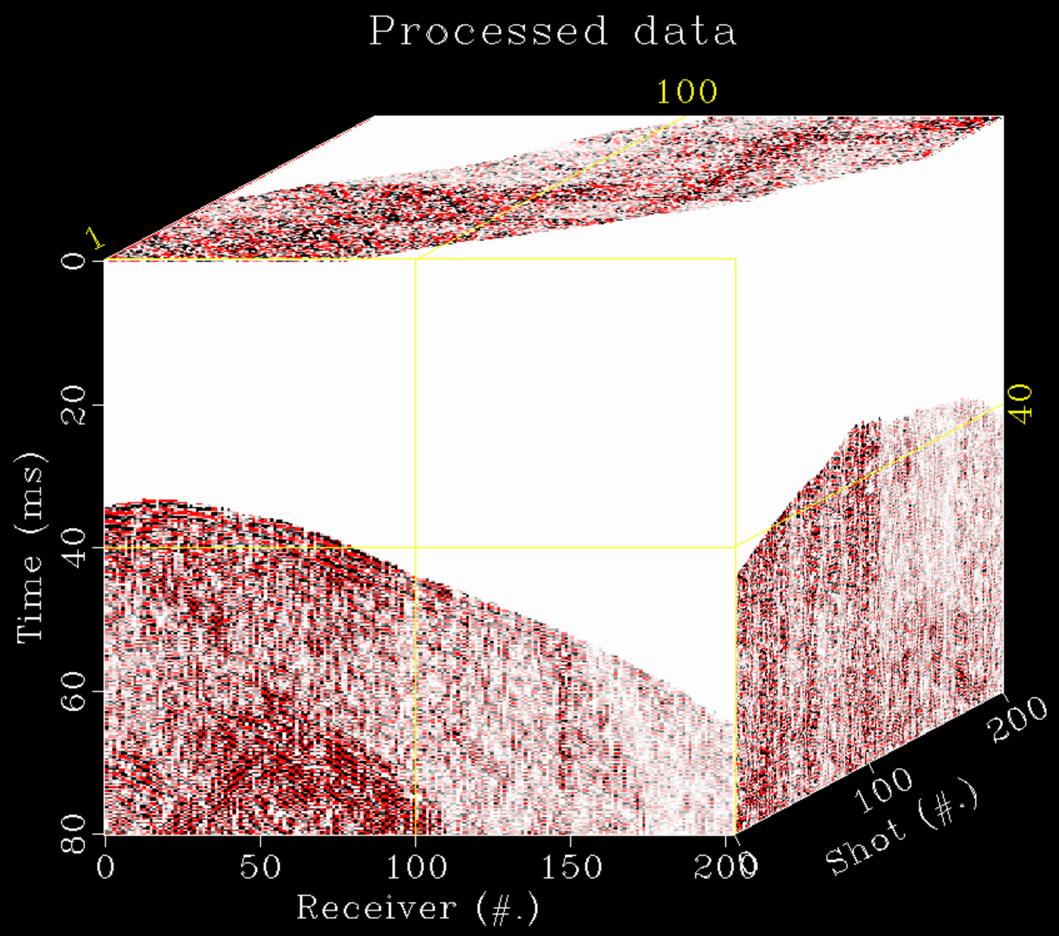


[Wang et al., 2021, Geophysics, In submission]

井间地震成像实例

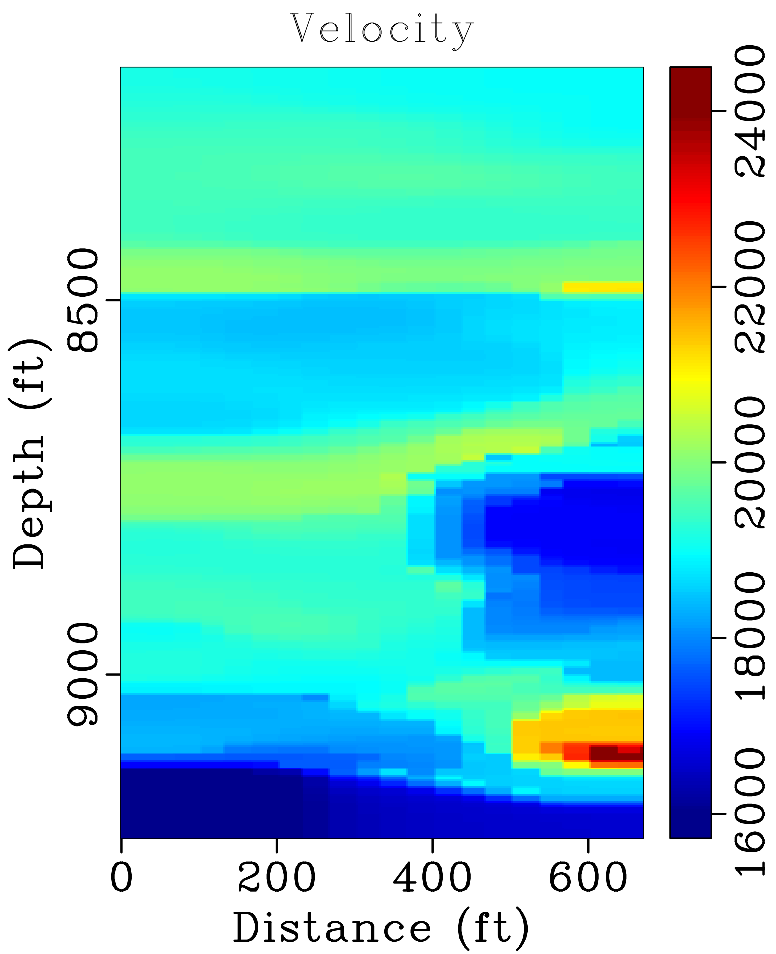


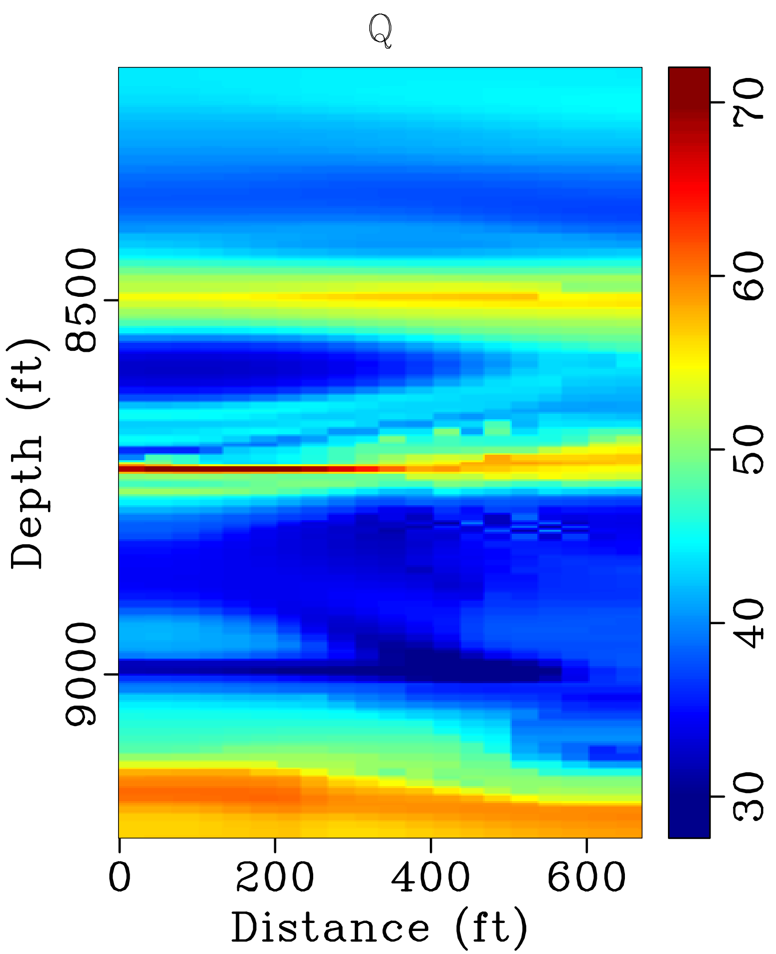
Windowed data
100
1
40
Time (ms)
0
20
40
60
80
Receiver (#.)
0
50
100
150
200
Shot (#.)
100
200



Processed data
100
1
40
Time (ms)
0
20
40
60
80
Receiver (#.)
0
50
100
150
200
Shot (#.)
100
200

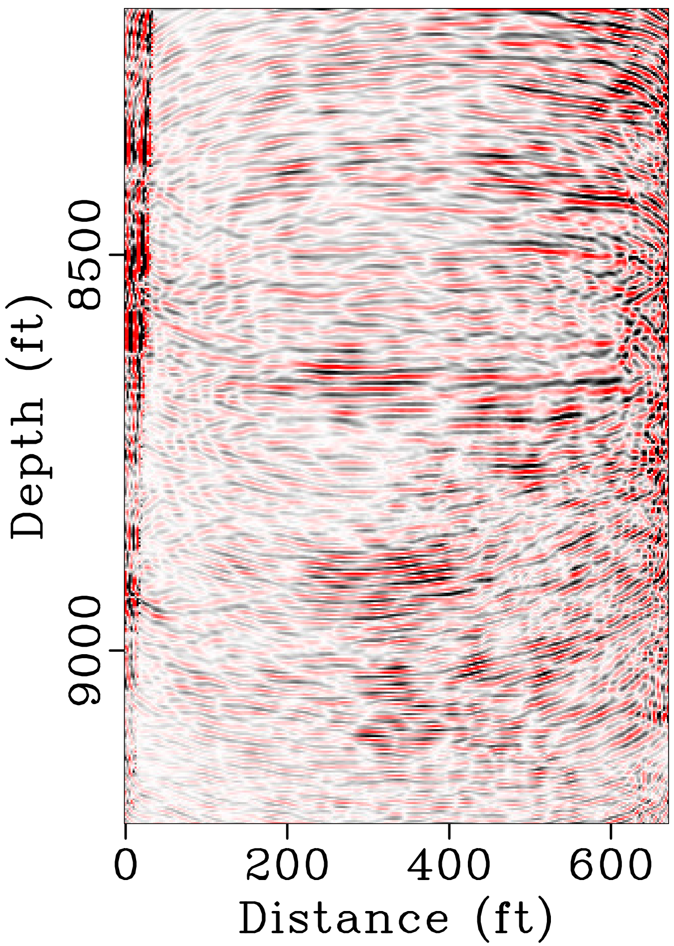
井间地震成像实例

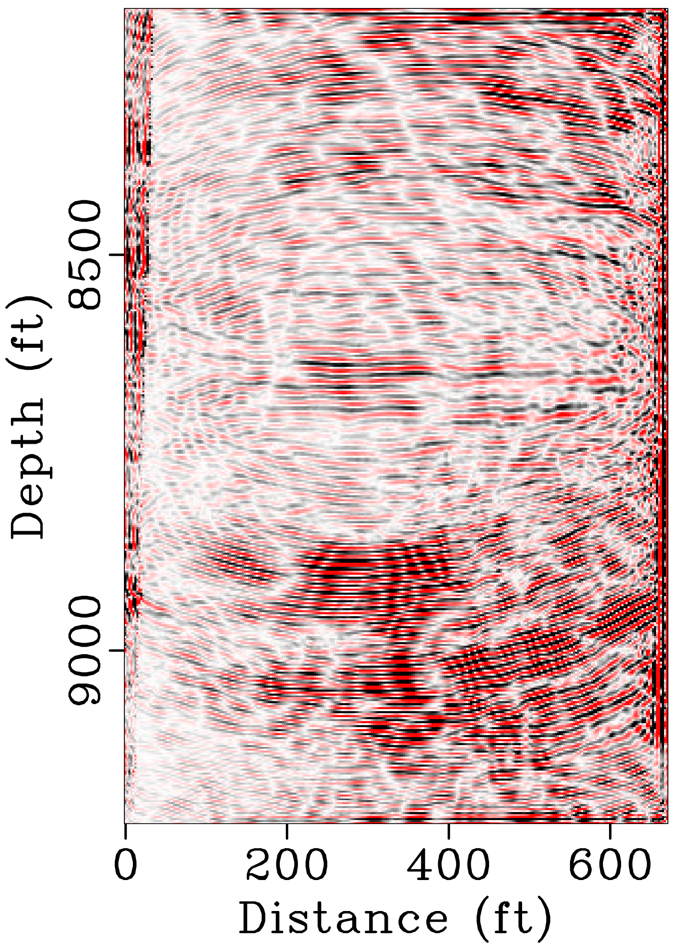




[Wang et al., 2021, Geophysics, In submission]

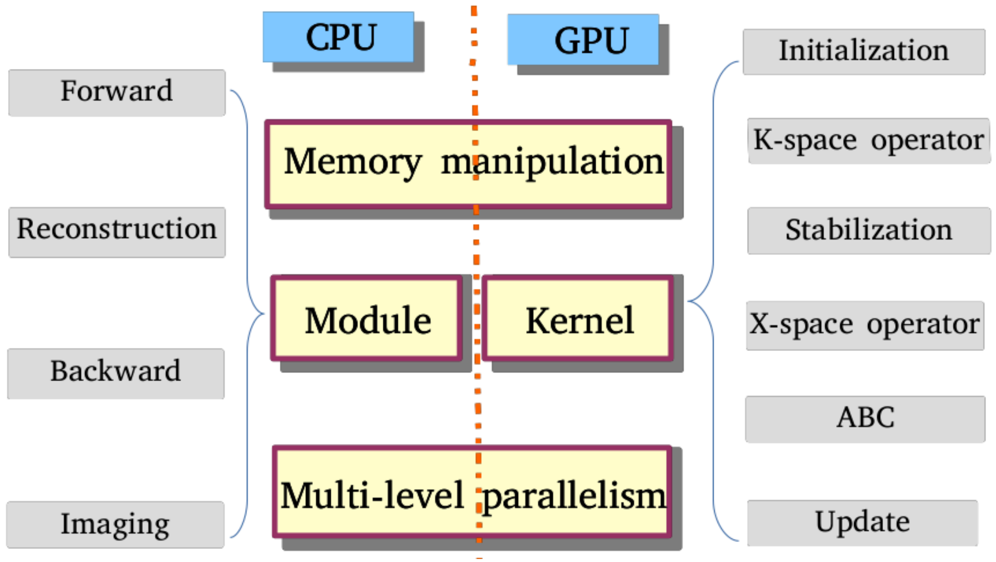
井间地震成像实例





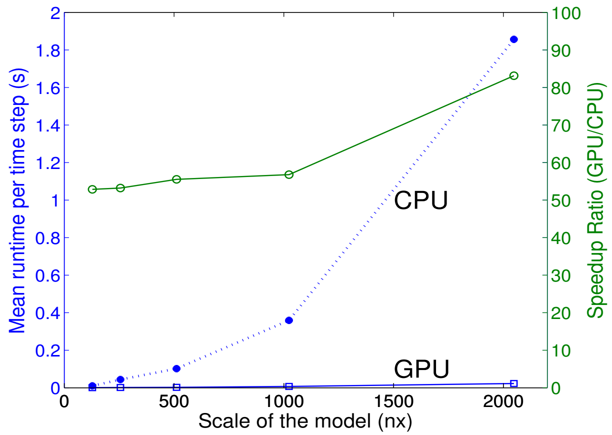
[Wang et al., 2021, Geophysics, In submission]

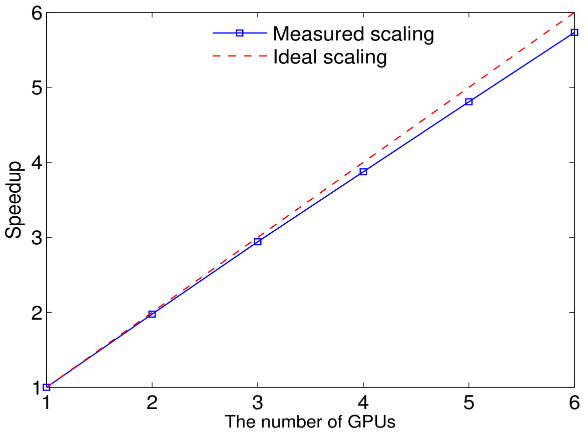
cuQ-RTM



Model scale (grids)	128 × 128	256 × 256	512 × 512	1024 × 1024	2048 × 2048
CPU runtime (ms)	9.7170	43.5925	101.3938	359.0682	1855.8382
GPU runtime (ms)	0.1839	0.8195	1.8262	6.3267	22.3263
Speedup ratio	52.8385	53.1940	55.5217	56.7544	83.1234

The number of GPUs	1	2	3	4	5	6
Manipulational runtime (s)	7.62	10.07	10.52	11.02	11.41	11.92
Computational runtime (s)	2639.29	1329.40	889.82	672.78	539.21	449.97
Total runtime (s)	2646.91	1339.50	900.34	683.80	550.62	461.89
Speedup ratio	1.0000	1.9761	2.9399	3.8709	4.8071	5.7306





[Wang et al., 2019, Geophysics]

cuQ-RTM

```
extern "C"
void cuda_visco_PSM_2d_forward(...)
{
    Multistream plans[GPU_N];
    for(i=0;i<GPU_N;i++)
    {
        cudaSetDevice(i);
        cudaStreamCreate(&plans[i].stream); // create stream for CUFFT
        cufftSetStream(plan[i].PLAN_FORWARD,plans[i].stream);
        cufftSetStream(plan[i].PLAN_BACKWARD,plans[i].stream);
        // copy memory from the host to the device
        cudaMemcpyAsync(...,cudaMemcpyHostToDevice,plans[i].stream);
        // wavefield initialization
        cuda_kernel_initialization<<<dimGrid,dimBlock,0,plans[i].stream>>>(...);
    }
    for(it=0;it<nt;it++)  // forward modeling
    {
        for(i=0;i<GPU_N;i++)
        {
            cudaSetDevice(i);
            cufftExecC2C(...,CUFFT_FORWARD); // CUFFT_FORWARD
            // k_space forward modeling
            cuda_kernel_visco_PSM_2d_forward_k_space<<<...>>>(...);
            cufftExecC2C(...,CUFFT_INVERSE); // CUFFT_INVERSE
            // x_space forward modeling
            cuda_kernel_visco_PSM_2d_forward_x_space<<<...>>>(...);
            cuda_kernel_MTF_2nd<<<...>>>(...); // MTF ABC
            cuda_kernel_checkpoints_Out<<<...>>>(...); // record checkpoints
            cuda_kernel_update<<<...>>>(...); // wavefield unpdate
        }
    }
    for(i=0;i<GPU_N;i++)
    {
        cudaSetDevice(i);
        // output seismograms
        cudaMemcpyAsync(...,cudaMemcpyDeviceToHost,plans[i].stream);
        cudaDeviceSynchronize(); // synchronize
        cudaStreamDestroy(plans[i].stream); // destroy stream for CUFFT
    }
}
```

```
extern "C"
void cuda_visco_PSM_2d_backward(...)
{
    Multistream plans[GPU_N];
    for(i=0;i<GPU_N;i++)
    {
        cudaSetDevice(i);
        cudaStreamCreate(&plans[i].stream); // create stream for CUFFT
        cufftSetStream(plan[i].PLAN_FORWARD,plans[i].stream);
        cufftSetStream(plan[i].PLAN_BACKWARD,plans[i].stream);
        cudaMemcpyAsync(...,cudaMemcpyHostToDevice,plans[i].stream);
        cuda_kernel_initialization<<<dimGrid,dimBlock,0,plans[i].stream>>>(...);
    }
    for(it=nt-3;it>=0;it--)
    {
        for(i=0;i<GPU_N;i++)
        {
            cudaSetDevice(i);
            if(Sto_Rec==0) // wavefield reconstruction
            {
                cufftExecC2C(...,CUFFT_FORWARD); // CUFFT_FORWARD
                cuda_kernel_visco_PSM_2d_reconstruction_k_space<<<...>>>(...);
                cufftExecC2C(...,CUFFT_INVERSE); // CUFFT_INVERSE
                cuda_kernel_AdaSta<<<...>>>(...); // stabilization
                cuda_kernel_visco_PSM_2d_reconstruction_x_space<<<...>>>(...);
                cuda_kernel_MTF_2nd<<<...>>>(...); // MTF ABC
                cuda_kernel_checkpoints_In<<<...>>>(...); // read checkpoints
                cuda_kernel_update<<<...>>>(...); // wavefield unpdate
            }
            cufftExecC2C(...,CUFFT_FORWARD); // CUFFT_FORWARD
            // k_space backward modeling
            cuda_kernel_visco_PSM_2d_backward_k_space<<<...>>>(...);
            cufftExecC2C(...,CUFFT_INVERSE); // CUFFT_INVERSE
            cuda_kernel_AdaSta<<<...>>>(...); // stabilization
            // x_space backward modeling
            cuda_kernel_visco_PSM_2d_backward_x_space<<<...>>>(...);
            cuda_kernel_MTF_2nd<<<...>>>(...); // MTF ABC
            cuda_kernel_image<<<...>>>(...); // imaging
            cuda_kernel_update<<<...>>>(...); // wavefield unpdate
        }
    }
    for(i=0;i<GPU_N;i++)
    {
        cudaSetDevice(i);
        cudaMemcpyAsync(...,cudaMemcpyDeviceToHost,plans[i].stream);
        cudaDeviceSynchronize(); // synchronize
        cudaStreamDestroy(plans[i].stream); // destroy stream for CUFFT
    }
}
```

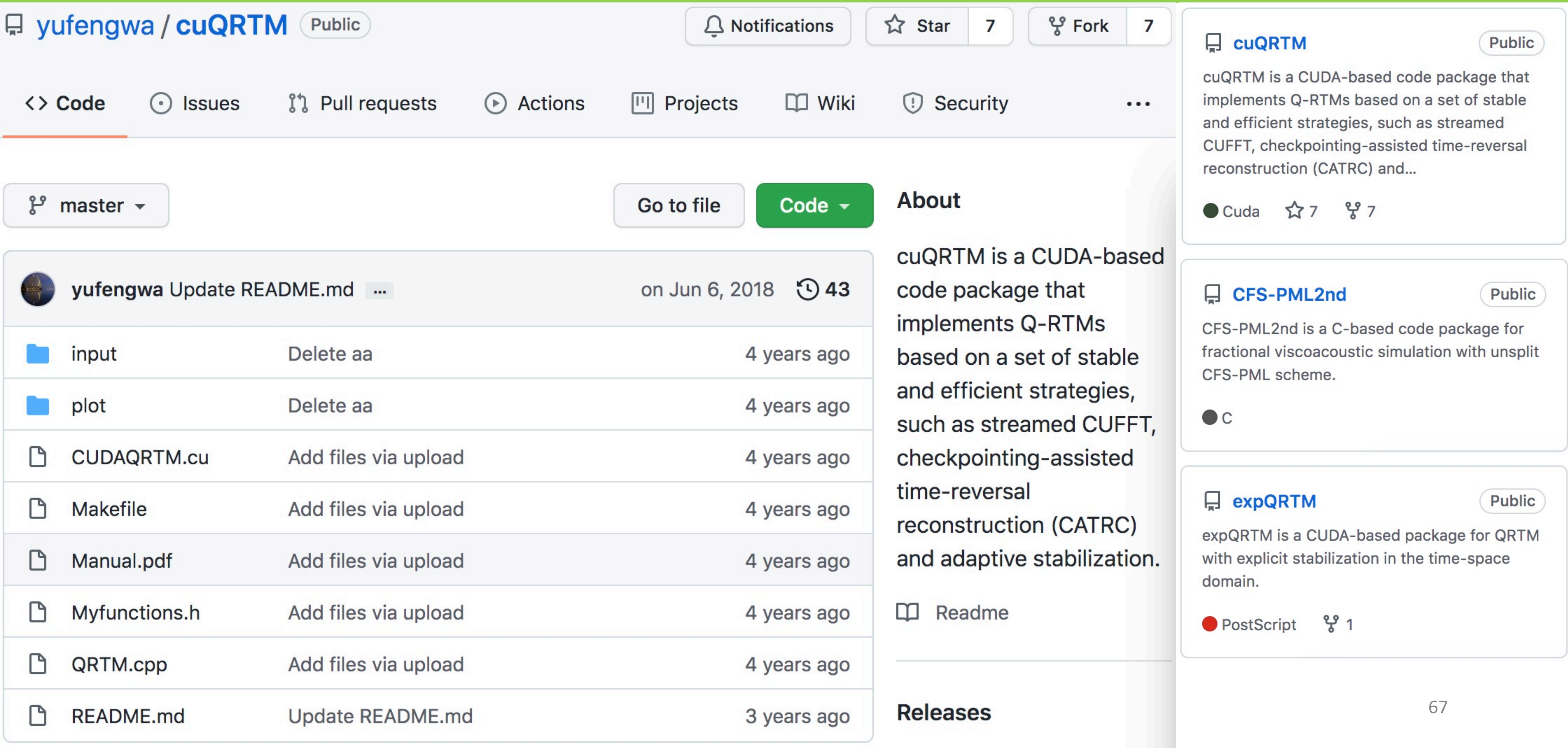
[Wang et al., 2019, Geophysics]

提纲

1. 地震成像
2. 高性能计算
3. 高性能地震成像实例
4. **地球物理开源研究**

开源研究

https://github.com/yufengwa/cuQRTM



开源研究

https://en.wikipedia.org/wiki/Comparison_of_free_geophysics_software

Name	Description	Originator	License	Platforms	Language	Notes
Madagascar[1]	Multidimensional data analysis including seismic processing	Sergey Fomel and others	GPL	Cross-platform	C, C++, Python	Programming Interfaces to Fortran 77, Fortran 90, Python, Java, MATLAB, Julia.
FreeUSP[2]	Seismic processing	Amoco, BP	Custom	Unix, Linux	Fortran 77, C	Includes OBS Node Survey Simulator, BP Anisotropic Velocity-Analysis Benchmark
FreeDDS[3]	Seismic processing	Amoco, BP	Custom	Unix, Linux	Fortran 77, C	Generic format (SEGY, SU, SEPlib, USP)
PSEIS-OSS[4]	Parallel seismic processing	Randy Selzler	GPL	Unix, Linux	Fortran 77, C	Generic format (SEGY, SU, SEPlib, USP, Madagascar)
CWP/SU (Seismic Unix)[5]	Seismic processing	Stanford, Colorado School of Mines	BSD-style	Unix, Linux, OS X, Cygwin 32	C, some Fortran 77	World's most widely used (82 countries and territories).
CPSeis[6]	Seismic processing	ConocoPhillips	MIT	Cross-platform	Fortran 90, C++	
SPARC[7]	Seismic processing	ARCO	GPL	Cray	Fortran	
SEPlib[8]	Seismic processing	Stanford University	Custom	Cross-platform	Fortran, C	
GeBR[9]	Seismic processing	Various contributors, Brazil	GPL	Linux	C	Interfaces with SU, Madagascar, etc.
OpenSeaSeis[10]	Seismic processing	Bjorn Olofsson, now property of the Colorado School of Mines	BSD-style	Linux, Windows, OS X	C, C++, Java	Interfaces with SU data. Reads most common SEGY and a subset of SEGD formats.

Madagascar

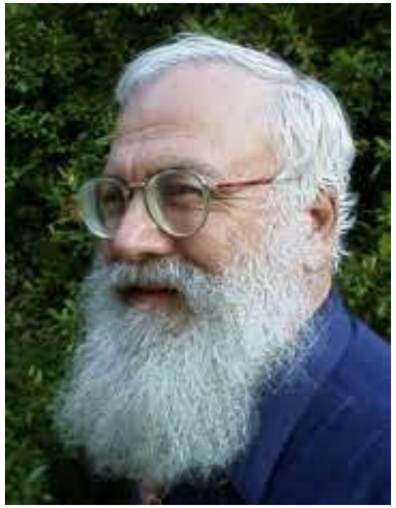
MADAGASCAR

Initial release	June 12, 2006
Stable release	3.1 / December 5, 2020; 9 months ago
Preview release	Through Subversion
Written in	C, C++, Python
Operating system	Posix
Type	Geophysical software
License	GNU General Public License
Website	www.ahay.org

开源研究

An article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures.

Jon Claerbout



科学出版物中关于计算科学的文章并不是学术本身，它只是学术的广告，而真正的学术则是包括完整的软件开发环境和生成结果图形的程序集。

谢谢大家！