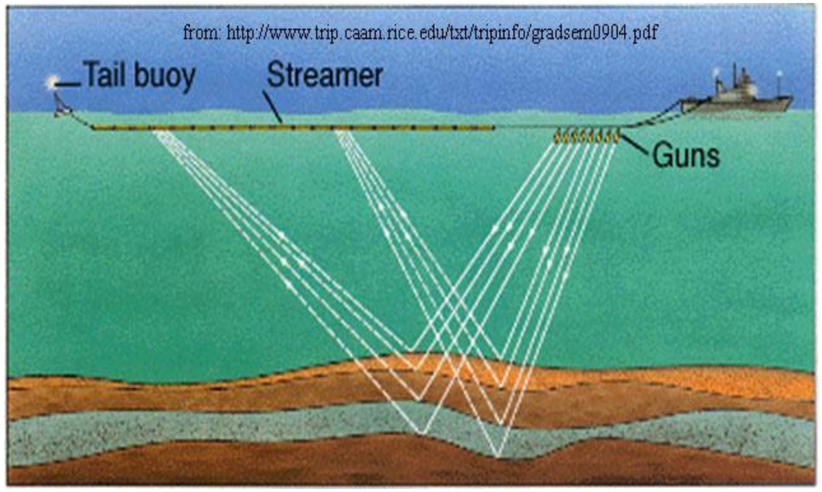
High-performance Implementation of 3D Seismic Target-oriented Imaging

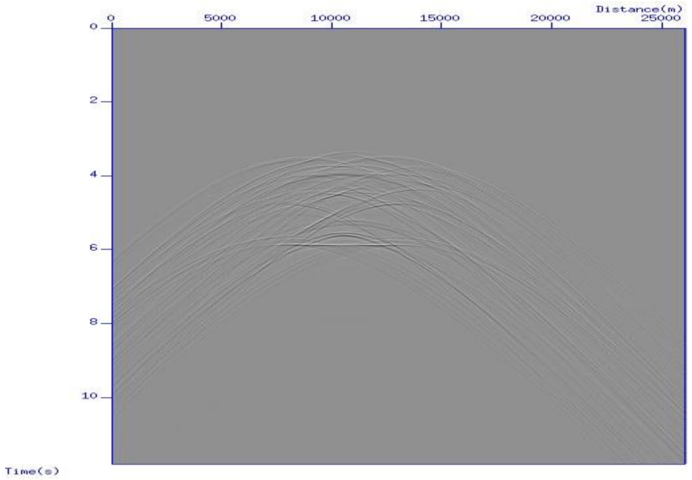
Maxim Protasov*, Institute of Petroleum Geology and Geophysics SB RAS, Novosibirsk, Russia

Seismic exploration

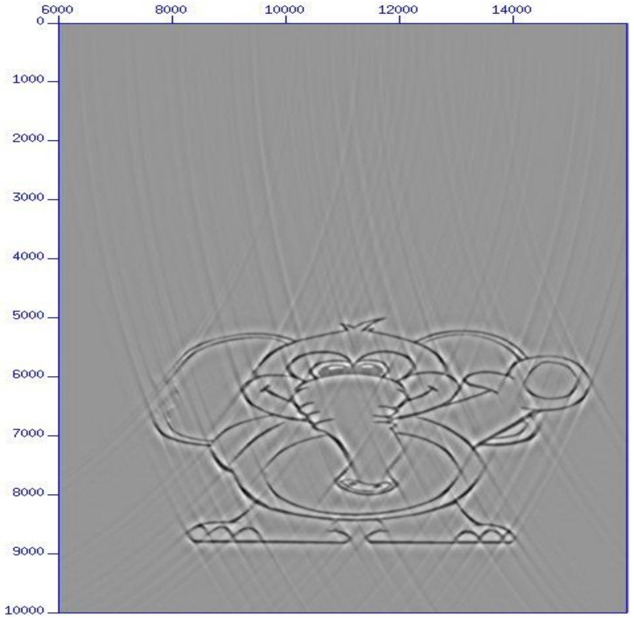


**Data size is rather big:
100 Gbytes – 10 Tbytes**

**Seismic data
In
"elephant"
model**



Seismic imaging



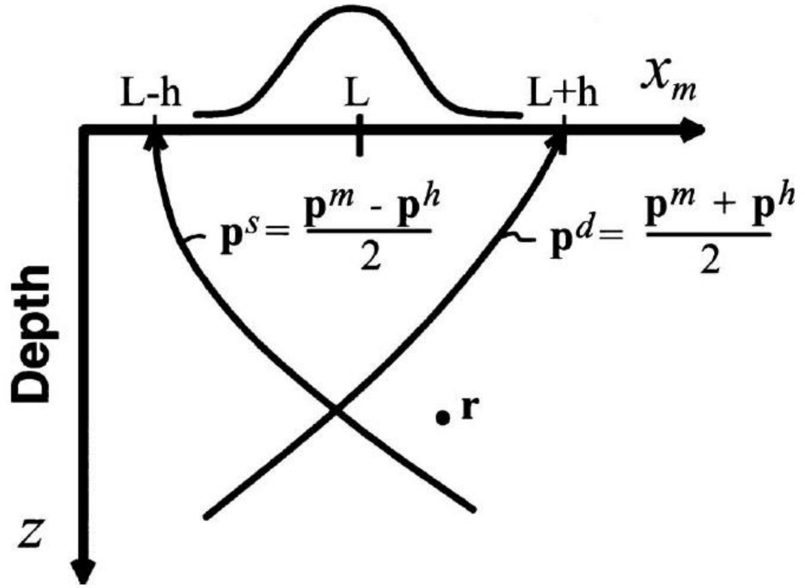
Image

Practical needs: fast computation of high quality images!

Beam Imaging for 3D SeismicData: imaging condition in acquisition point coordinates

Hill N. R. Prestack Gaussian-beam depth migration // Geophysics. – 2001. – V.66. – P. 1240–1250.

$$I_{\mathbf{h}}(\mathbf{r}) \approx \frac{2i\omega}{\pi} C_0 \sum_{\mathbf{L}} \int d\omega \iint dp_x^d \, dp_y^d \times \iint \frac{dp_x^s \, dp_y^s}{p_z^s} u_{GB}^*(\mathbf{r}; \mathbf{L}+\mathbf{h}, \mathbf{p}^d; \omega) \times u_{GB}^*(\mathbf{r}; \mathbf{L}-\mathbf{h}, \mathbf{p}^s; \omega) D_{\mathbf{h}}(\mathbf{L}, \mathbf{p}^d + \mathbf{p}^s, \omega). \quad (16)$$



M.I. Protasov, V.A. Tcheverda, A.P. Pravduhin. 3D true-amplitude anisotropic elastic Gaussian beam depth migration of 3D irregular data // Journal of seismic exploration. – 2019. – V.28(2) . – P.121-146.

$$f_h(\bar{x}, h_{x0}, h_{y0}) \approx \int dx_m dy_m dh_x dh_y dx_{m0} dy_{m0} \cdot BelDet \cdot \boxed{\mathrm{T}^m_{gbt}(x_m, y_m; \bar{x}; x_{m0}, y_{m0})} \cdot \boxed{\vec{\mathrm{T}}^r_{gbt}(h_x, h_y; \bar{x}; h_{x0}, h_{y0})} \cdot \vec{\varphi}(x_m, y_m; h_x, h_y; t = t^s_{gb}(x_s, y_s) + t^r_{gb}(x_r, y_r));$$

Equal beam patterns for all points

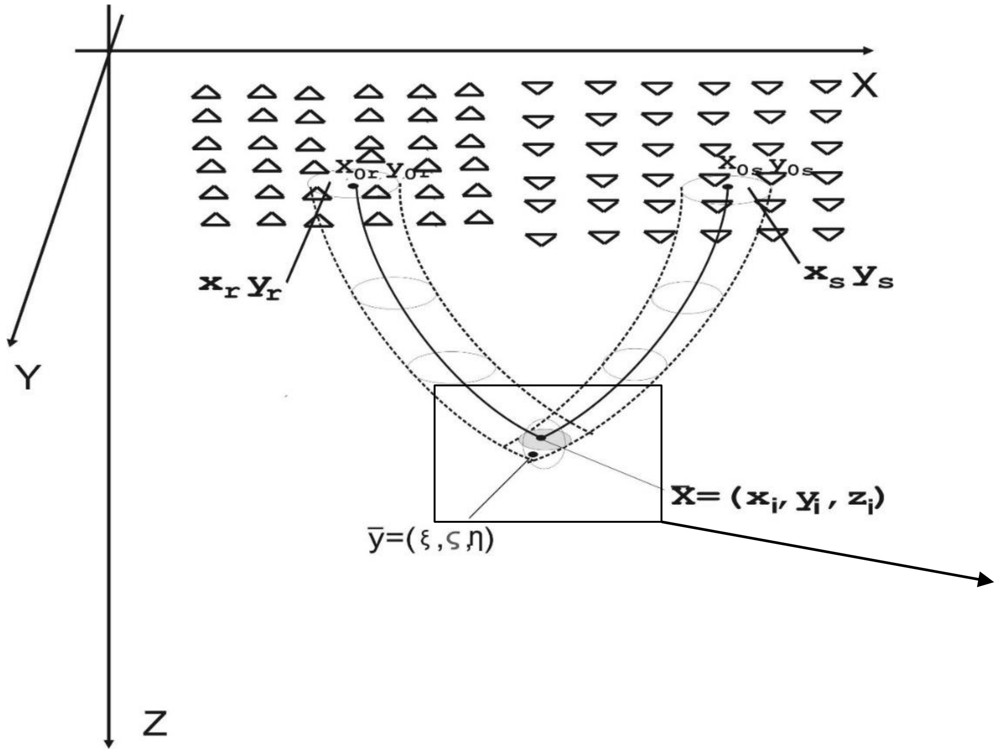
Beam Imaging for 3D Seismic Data: imaging condition in structural angles

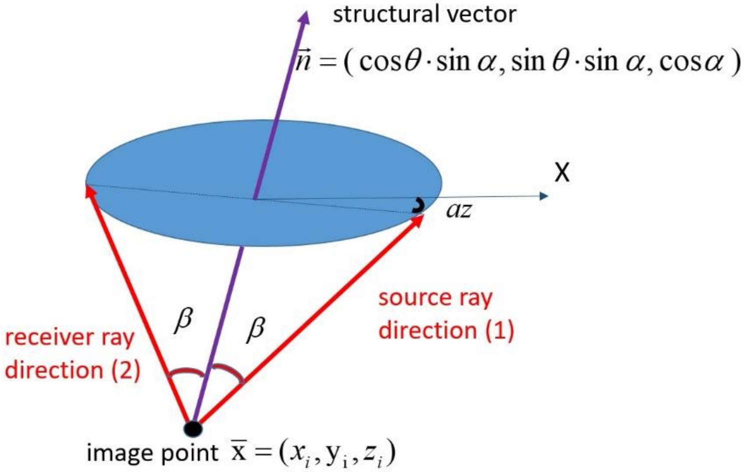
α – *structure dip*

θ – *structure azimuth*

β – *ray opening*

az – *ray azimuth*



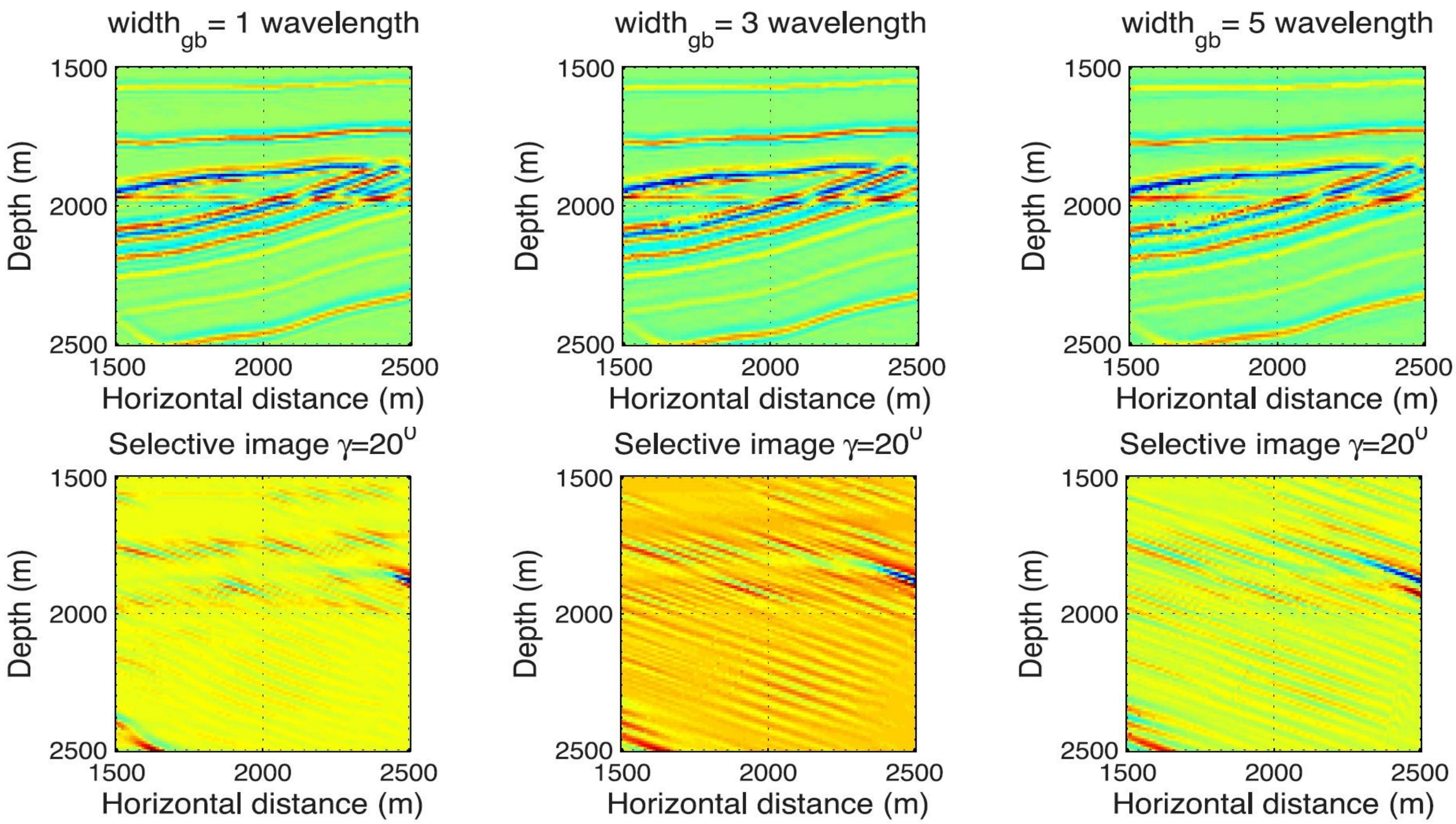


$$\widetilde{f}(\bar{x};\beta,az) = \int dx_s dy_s dx_r dy_r d\alpha d\theta d\omega \cdot k(\alpha,\beta)^3 \cdot \cos(\theta) \cdot$$

$$\cdot \boxed{\mathrm{T}^{s}_{gb}(x_s, y_s; \bar{x}; \omega; \alpha, \theta, \mathrm{az}, \beta)} \cdot \boxed{\vec{\mathrm{T}}^{r}_{gb}(x_r, y_r; \bar{x}; \omega; \alpha, \theta, az, \beta)} \cdot \vec{\varphi}(x_r, y_r; x_s, y_s; \omega);$$

Different beam patterns for all points

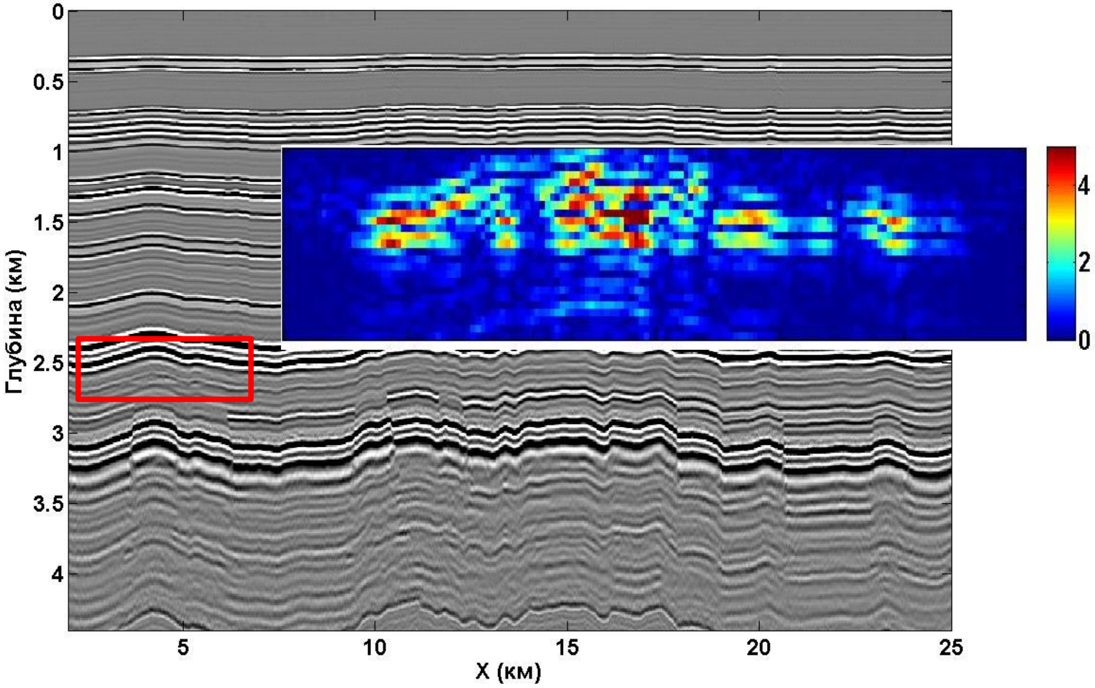
The impact of the beam kernel on selective images

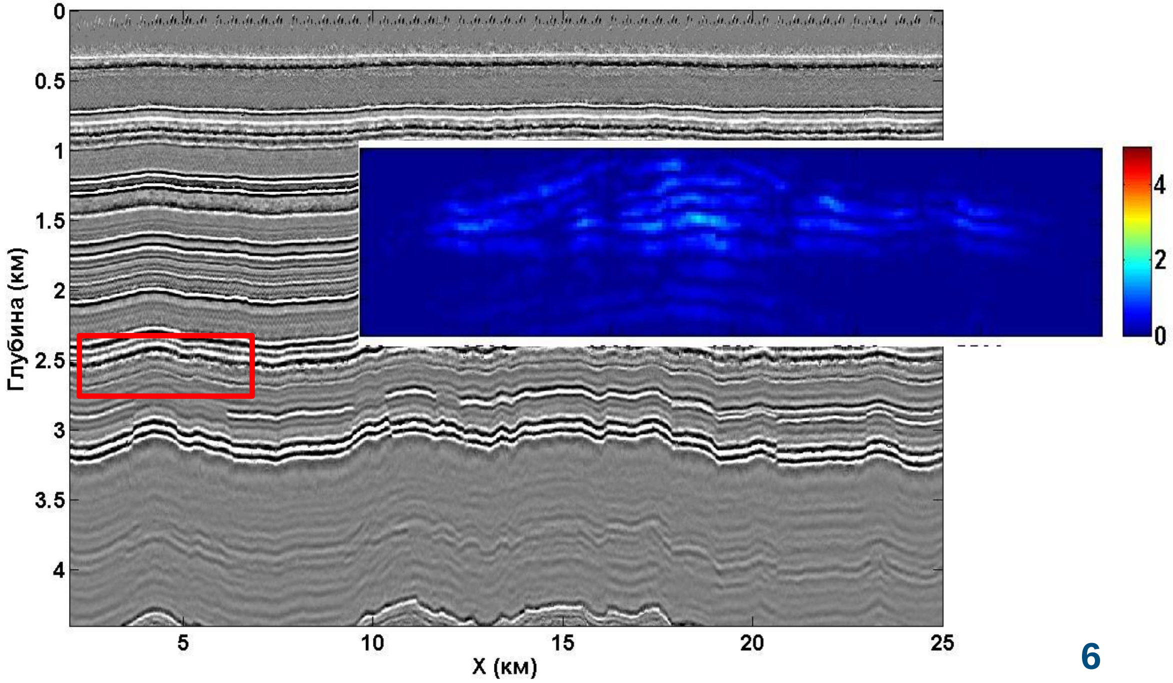


Beam Kernel is defined by beams (its width) in the image points

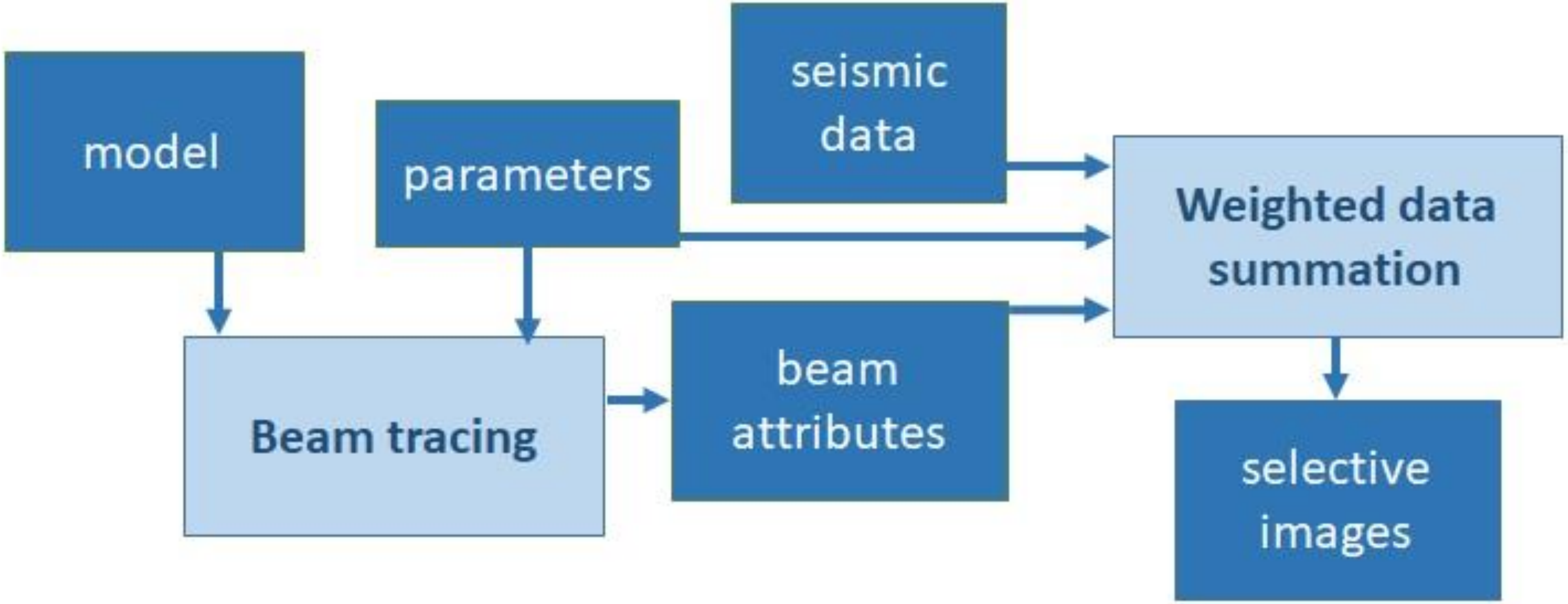
Computational time comparison for midpoint-offset realization and structural angle realization.

Data type	Data size	Computational time (1 CPU) structural angle realization	Computational time (1 CPU) midpoint-offset realization
2D data set	2 GB	40 hours	1 hour
3D data set	26 GB	4000 hours	37 hours

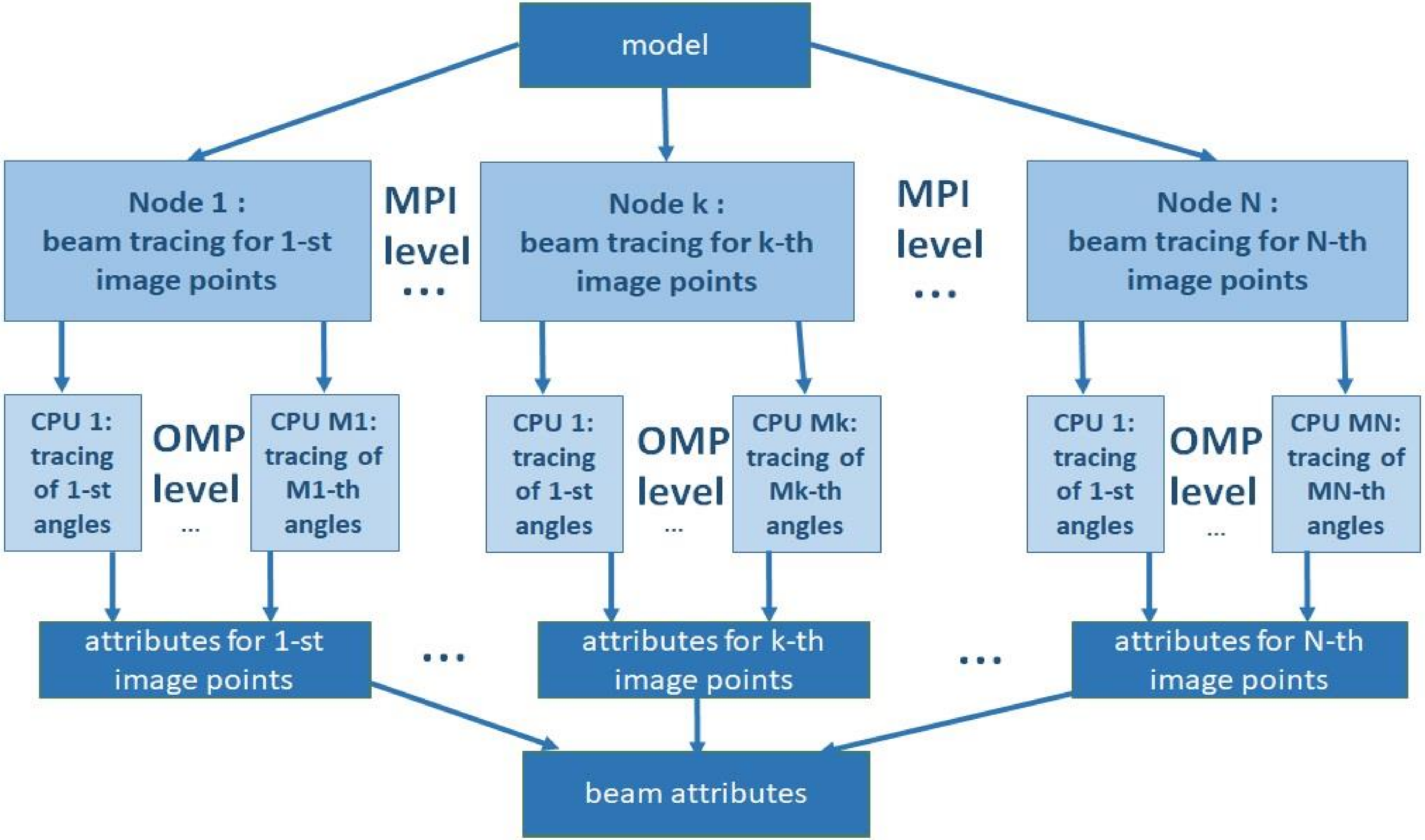




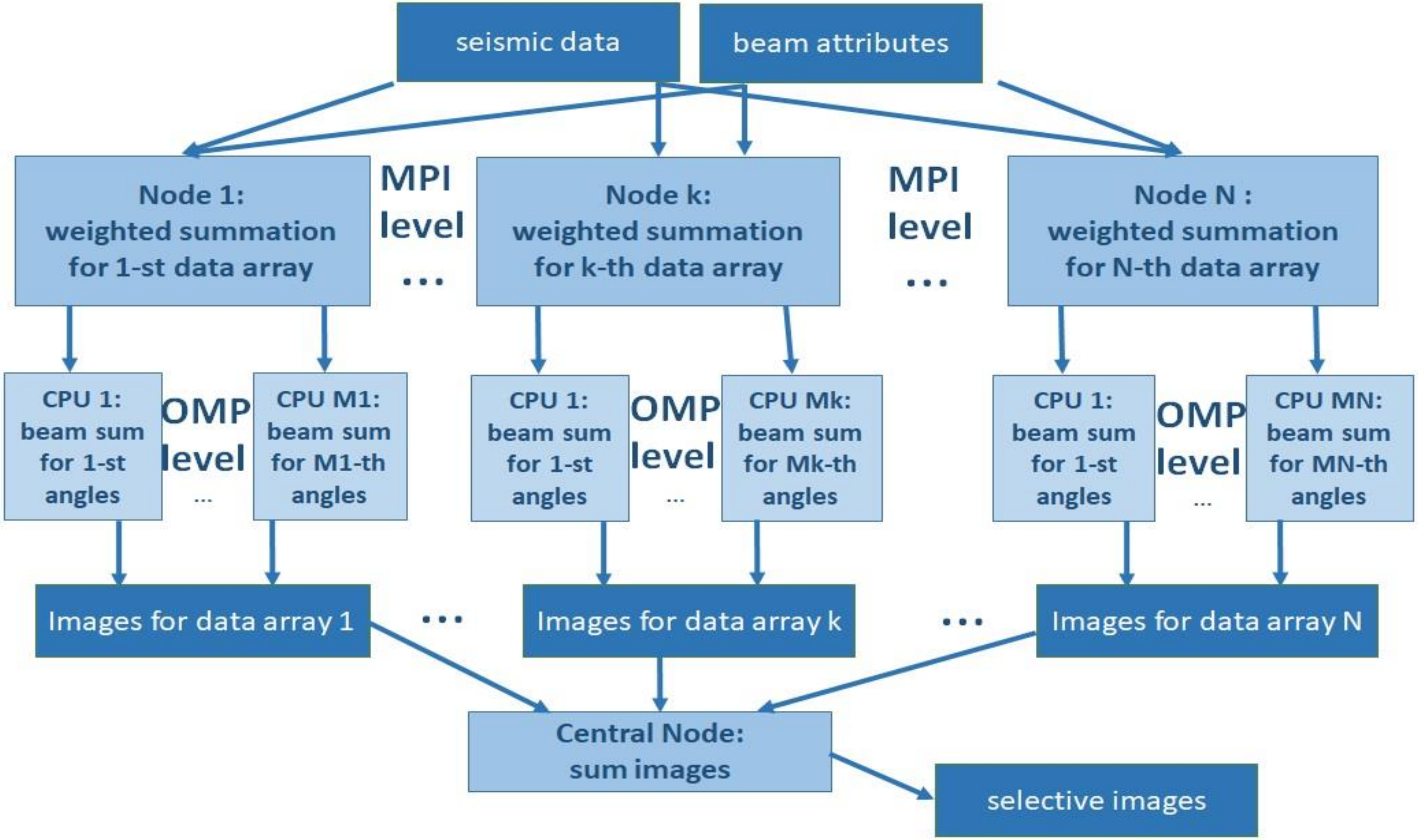
General upper-level diagram



Parallelization beam of attributes computation



Parallelization of data mapping process



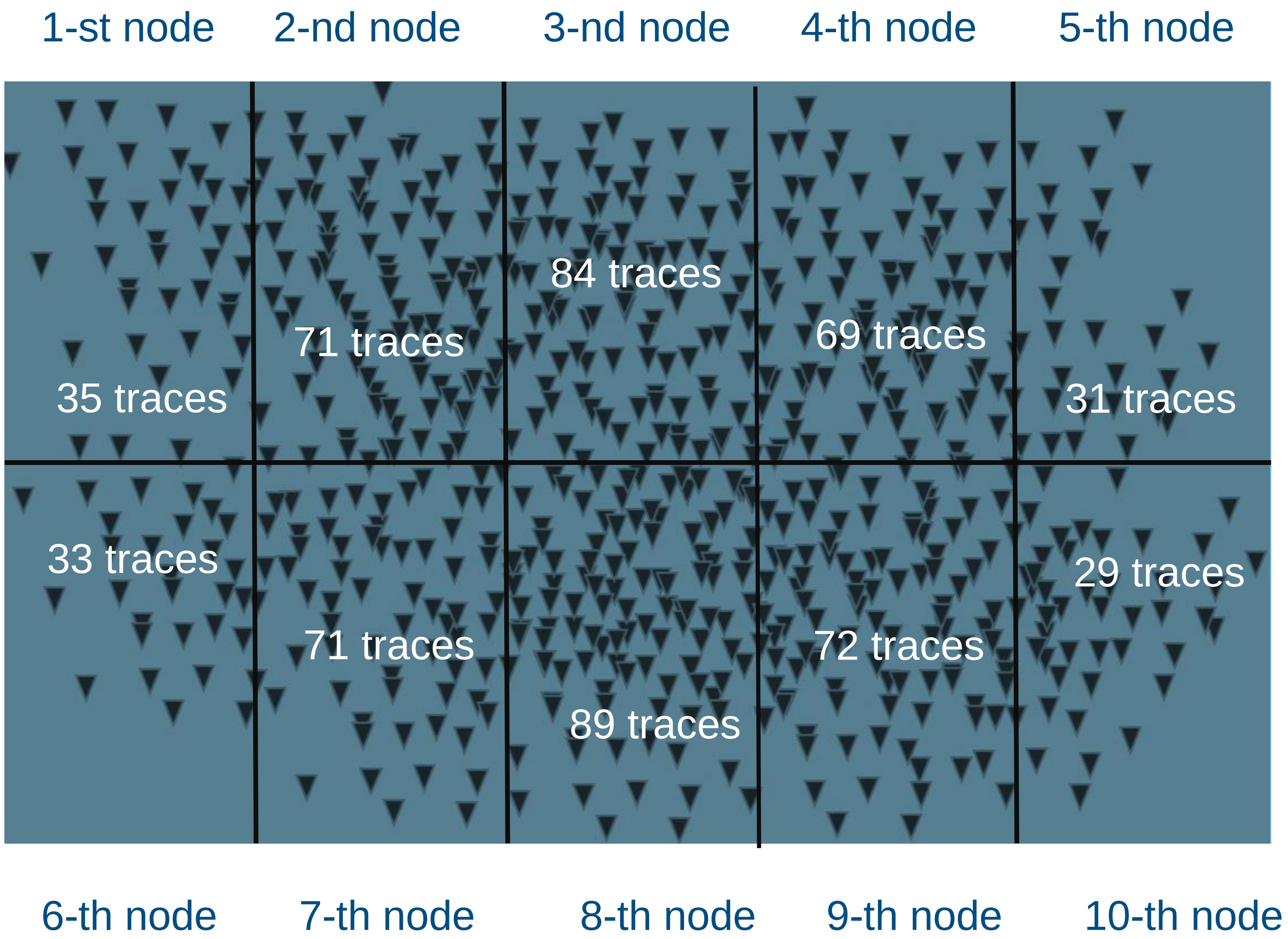
Initial non-optimal version: **equal areas of data** for every mpi node

Non-equal number of traces!

Non-optimal node usage!

Computational efficiency 40 %!

Example: 10 data areas per 10 mpi nodes



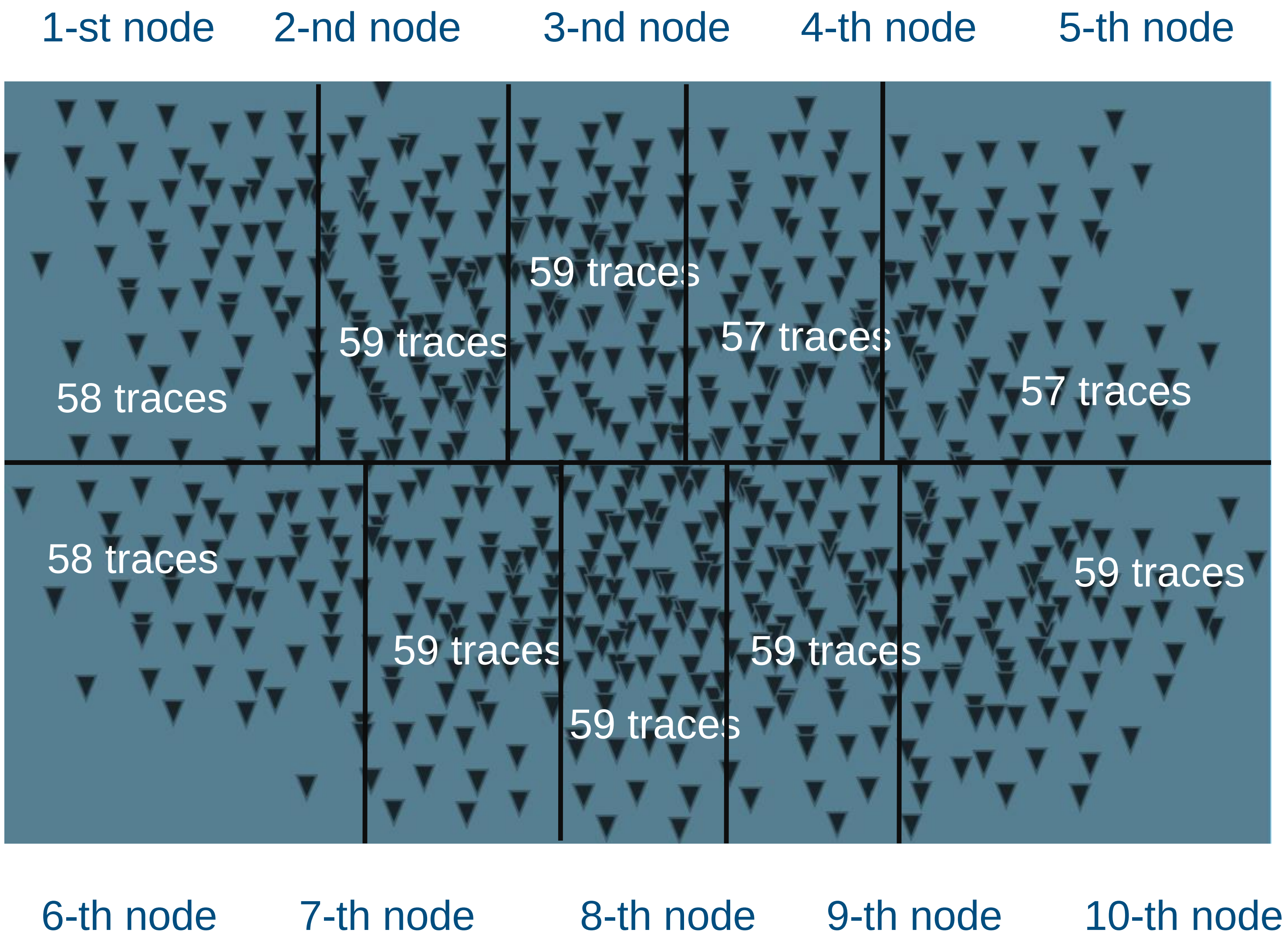
First optimization: get **equal number of data traces** for every mpi node

Example: 10 data areas per 10 mpi nodes

Equal number of traces!

More efficient node usage!

Computational efficiency 60-70 %!



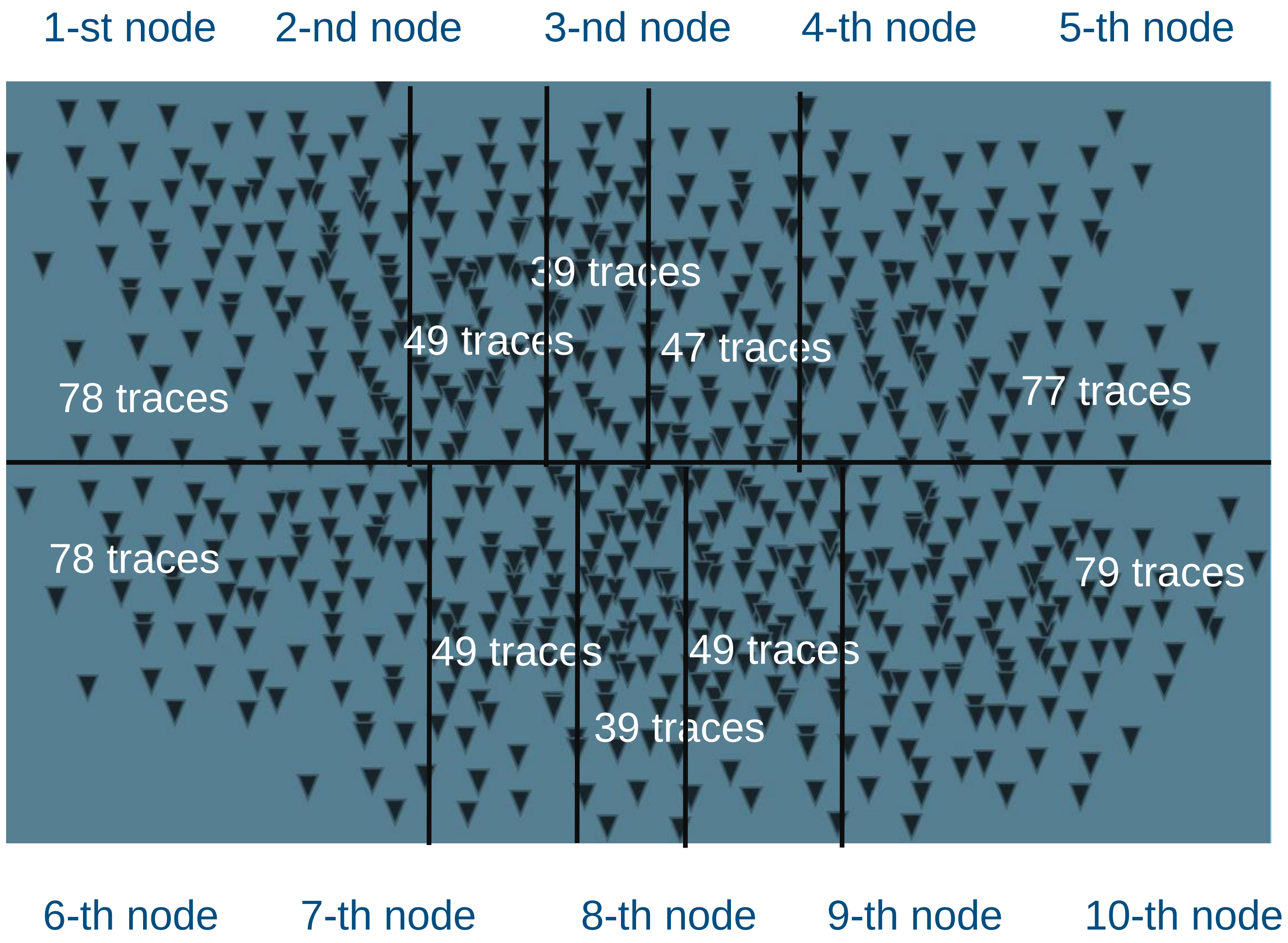
Second optimization: take into account **data illumination in each area**

Equal number of traces + data illumination!

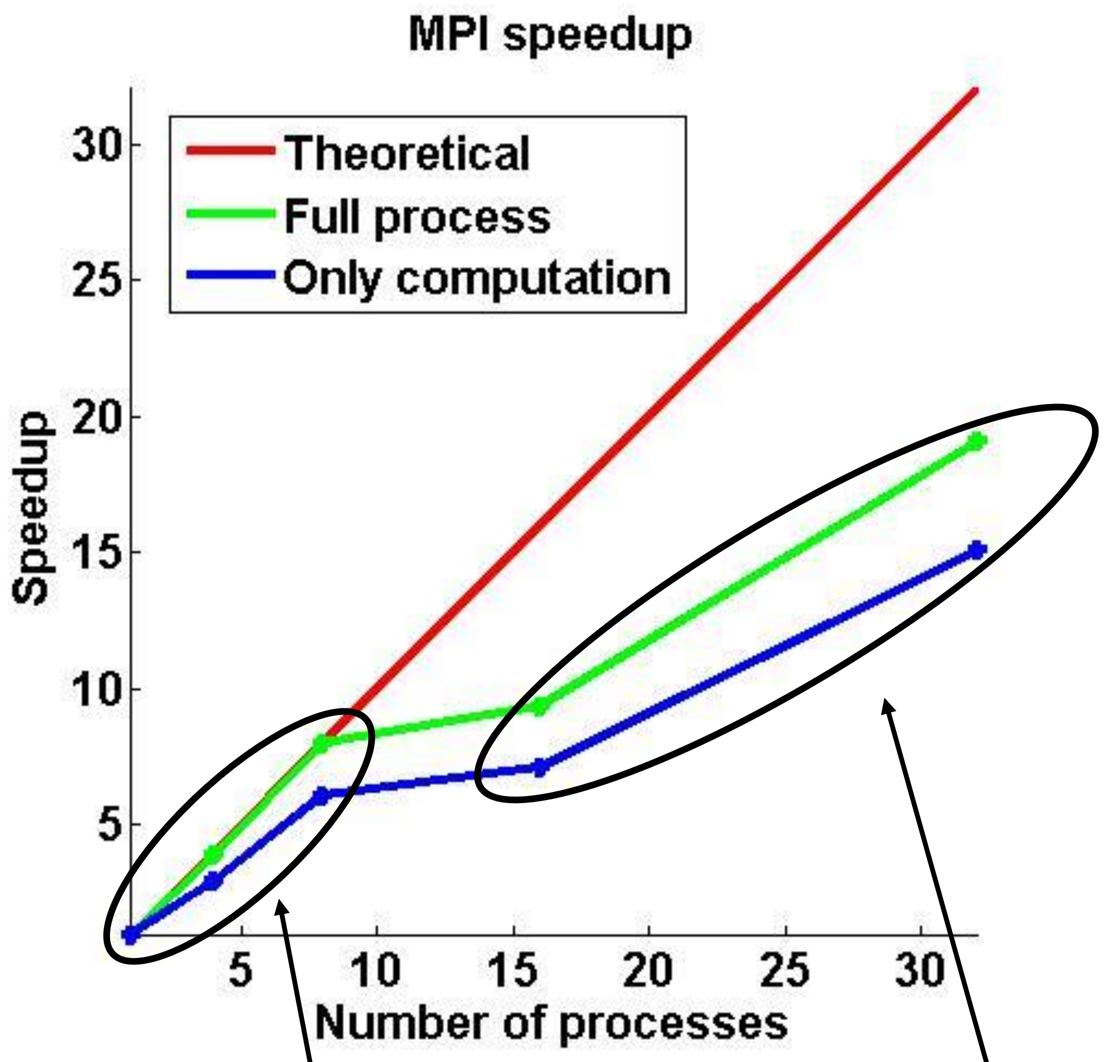
More efficient node usage!

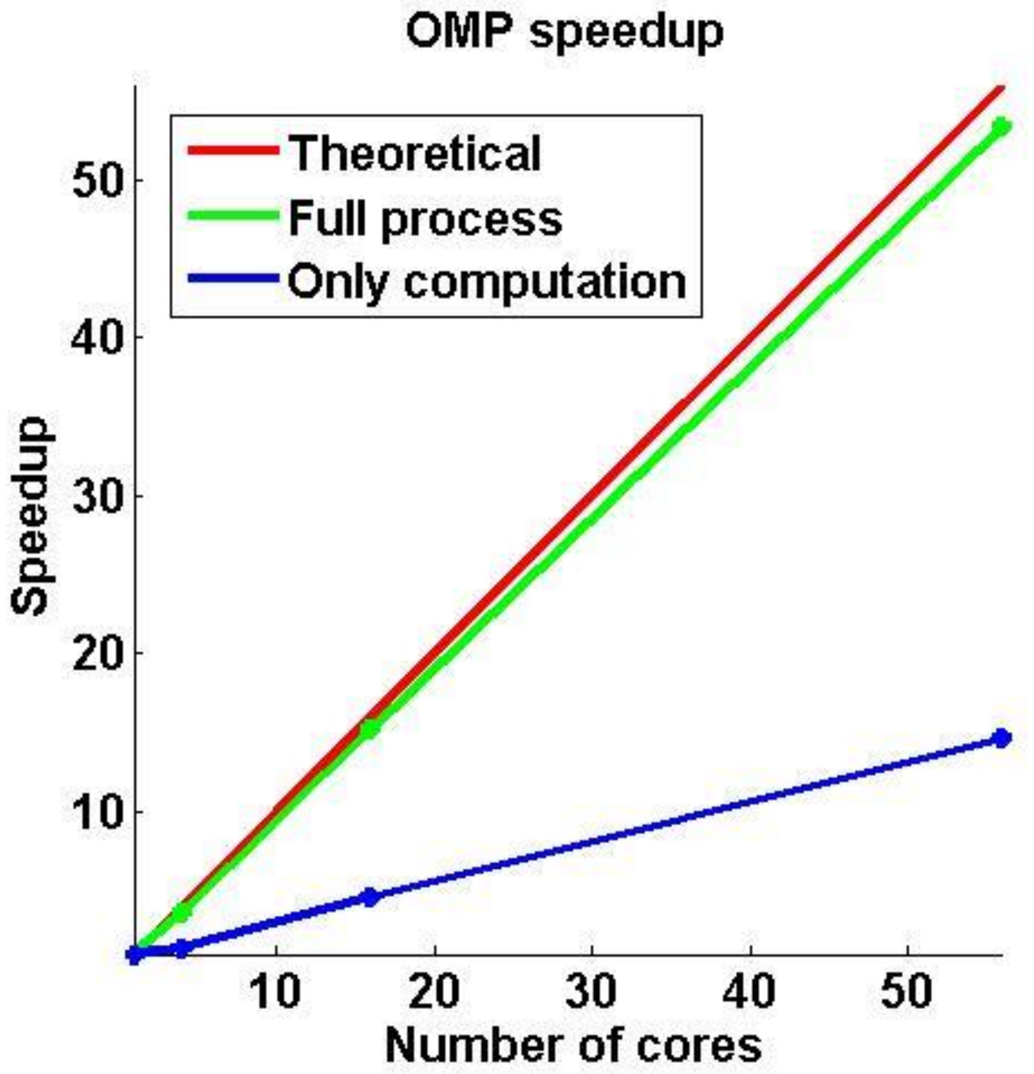
Computational efficiency 80-90 %!

Example: 10 data areas per 10 mpi nodes



Weak scaling

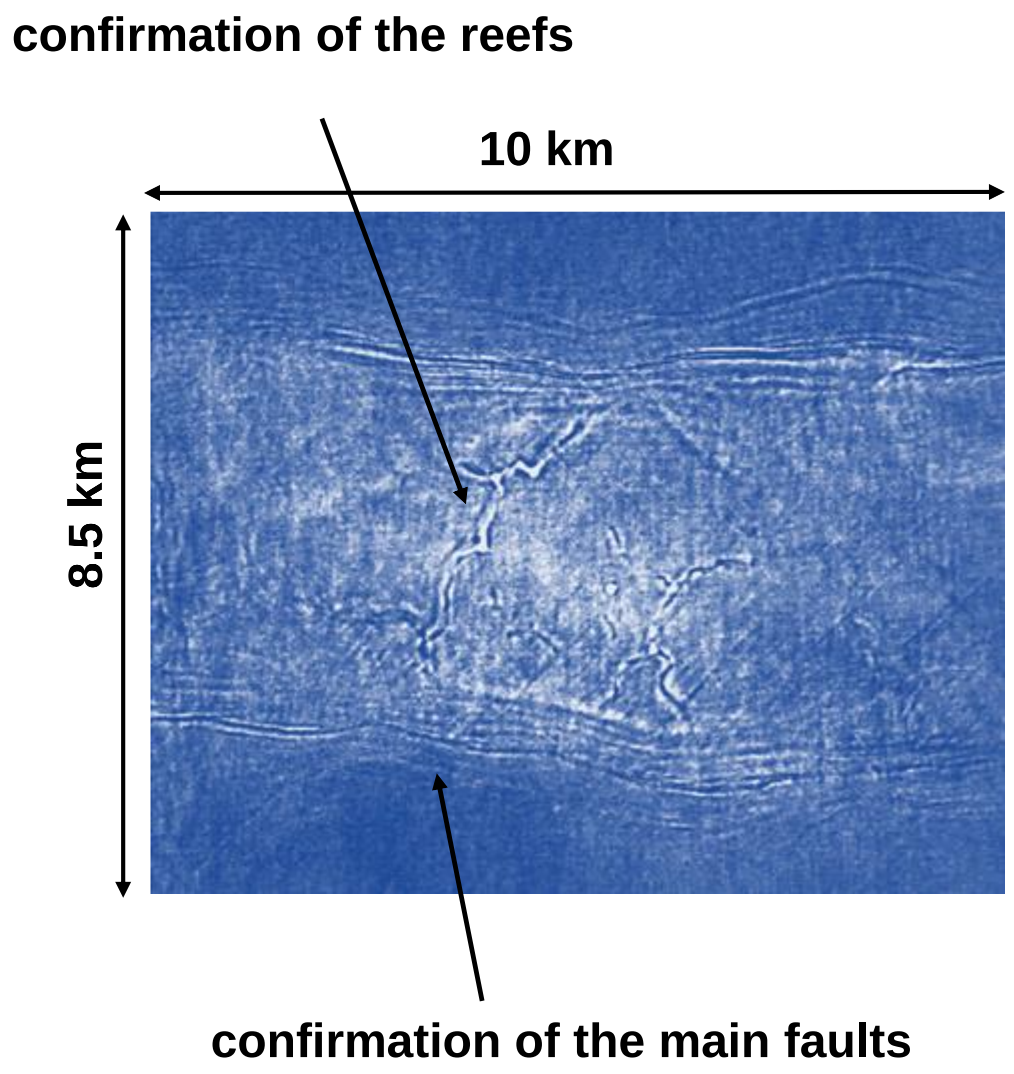


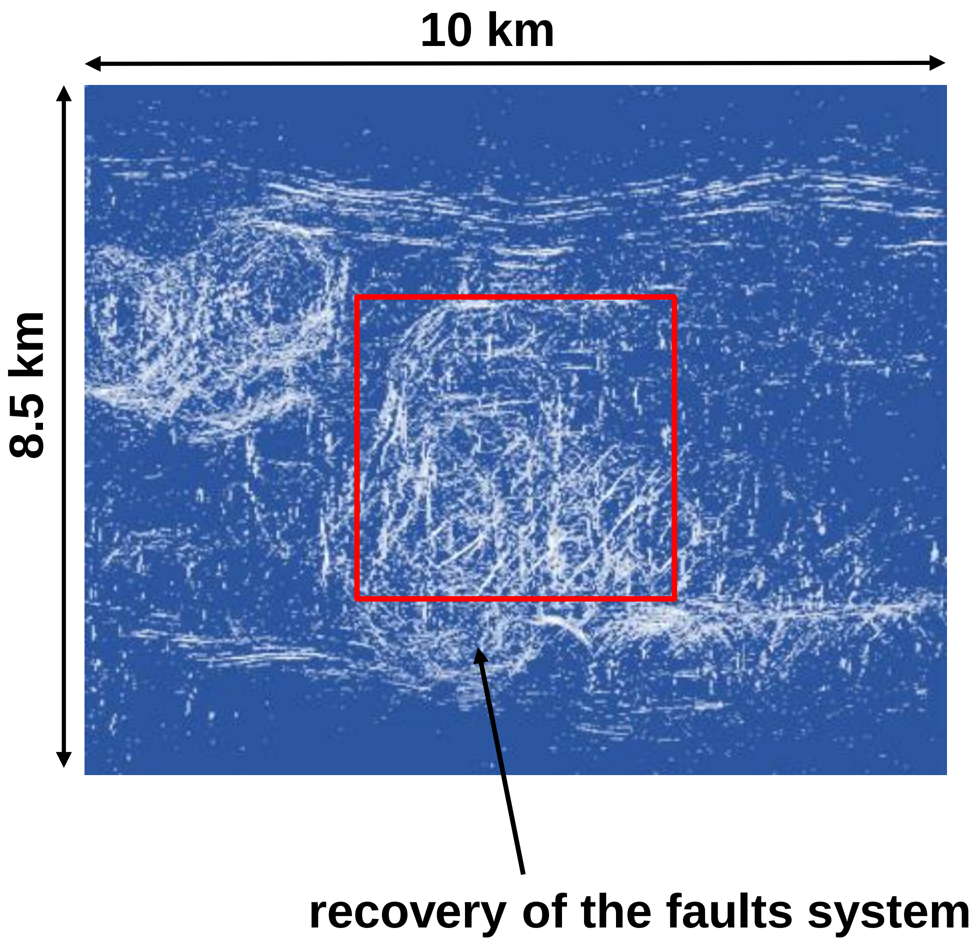


`trace_min/trace_max=0.7`

`trace_min/trace_max=0.9`

3D seismic data from north of Siberia: image results

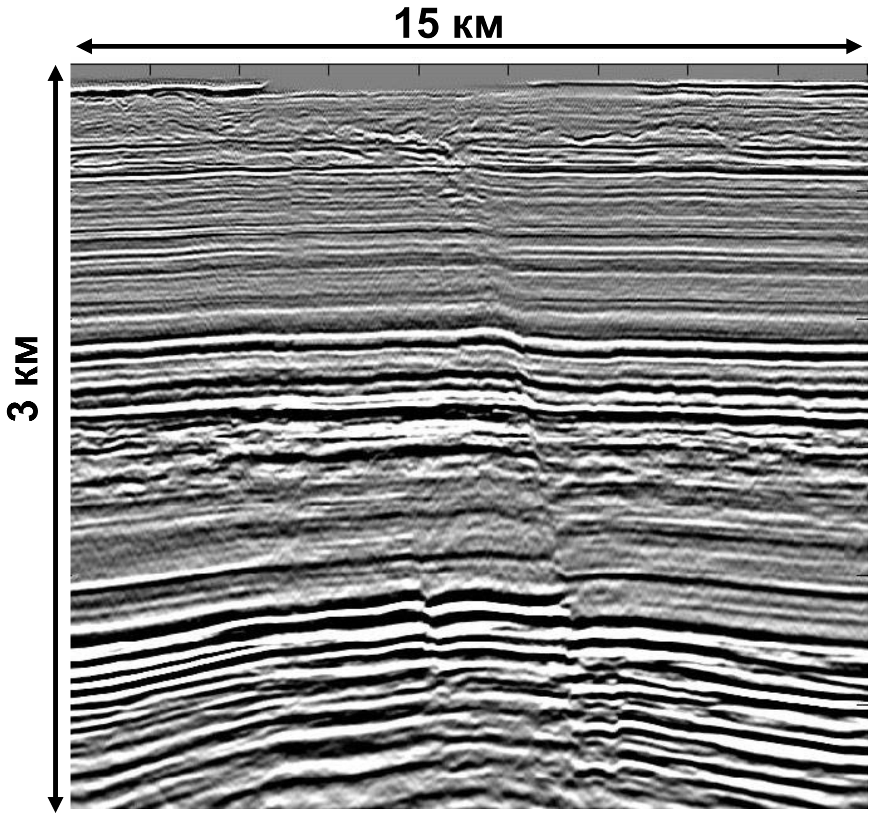




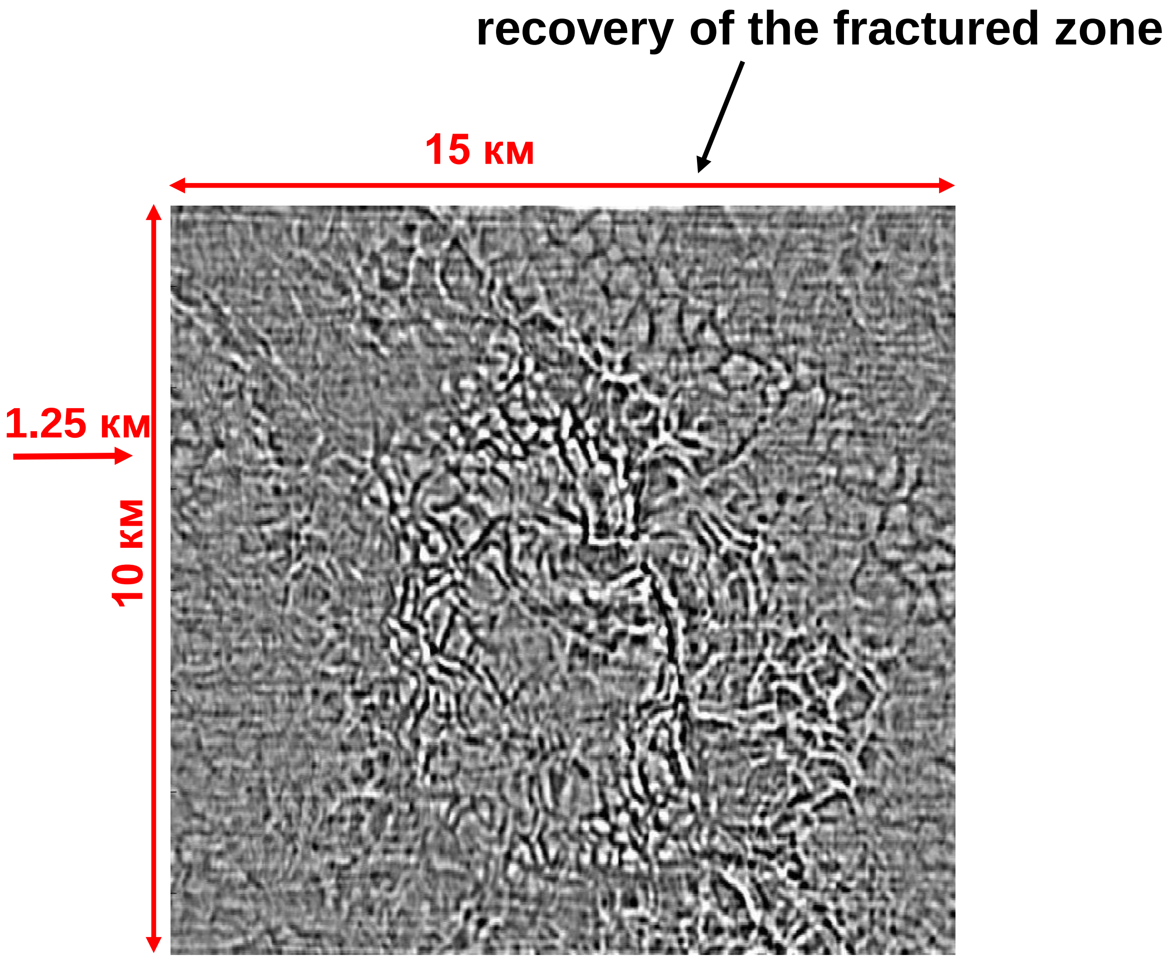
3D seismic data from north of Siberia: computational times

Image size nx×ny×nz×na	Data size	Computational resources	Computational time
450×400×2×8	23 GB	2 Nodes × 24 CPU cores	≈ 2.3 hours
200×1×125×24	44 GB	4 Nodes × 24 CPU cores	≈ 0.6 hours
270×250×1×24	44 GB	4 Nodes × 24 CPU cores	≈ 1.2 hours

3D seismic data from Kara Sea: image results



reflection imaging



diffraction imaging

3D seismic data from Kara Sea: computational times

Image size nx×ny×nz × na	Data size	Computational resources	Computational time
300×200×1 × 24	592 GB	16 Nodes × 24 CPU cores	≈ 20 hours
300×200×1 × 108	592 GB	32 Nodes × 54 CPU cores	≈ 20 hours
300×1×401 × 56	592 GB	32 Nodes × 56 CPU cores	≈ 6 hours

Conclusions

1. Parallel implementation of the 3D seismic target-oriented imaging algorithm is developed.

2. The implementation of target-oriented imaging utilizes MPI and OpenMP technologies providing two-level parallelization.

3. Created version of target-oriented migration provides superior quality images comparing to conventional beam imaging, especially in terms of the diffraction imaging results.

4. Field data examples demonstrate that the developed parallel implementation of target-oriented imaging can process large enough volumes of 3D seismic data in production mode.

The presented work is funded by RSF, project number 21-71-20002.

The numerical results of the work were obtained using computational resources of Peter the Great Saint-Petersburg Polytechnic University Supercomputing Center (scc.spbstu.ru).

Thank you for attention!

Questions?