Improving MPI scalability of multifrontal direct solver for 3D Helmholtz equation with data compression

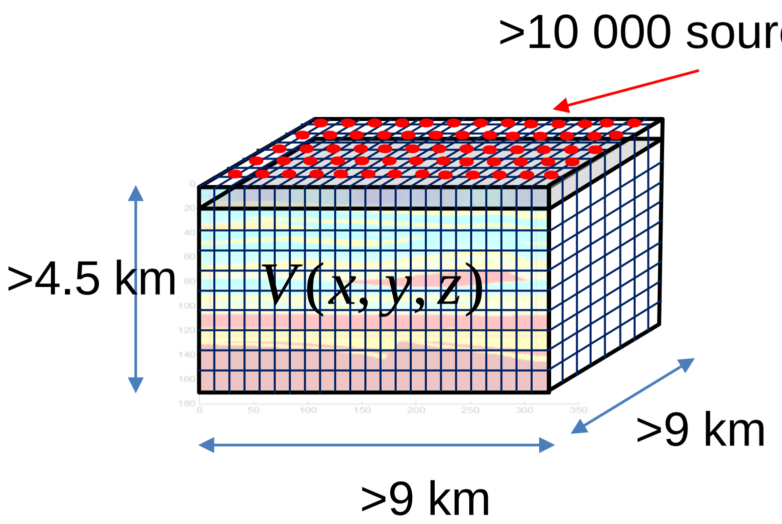
Sergey Solovyev and Victor Kostin,
Institute of Petroleum Geology and Geophysics SB RAS, Russia

Statement of problem

Solve the Helmholtz problem

$$u = ?$$

$$\Delta u + \frac{(2\pi\nu)^2}{V^2} u = f$$



- ✓ Velocity model *V*= 1000м/с ... 8000м/с
- ✓ Frequency 1,…, 16Гц

- ✓ Parallelepipedal grid, step is ~30m
- ✓ Perfect Matching Layer (PML)
- ✓ Finite difference approximation

Solve the symmetric complex sparse SLAE

$$AX = B, \quad X = \{x_1, \ldots, x_{nrhs}\}, \quad \dim(A) = n \times n$$

n> **20***10^6, *nrhs*>**10^4**

Direct solver outline

Given system of linear equations

$$AX = B.$$

- Decompose the matrix

$$A = L \cdot D \cdot L^t$$

- Solve two systems of linear equations with triangular coefficient matrices

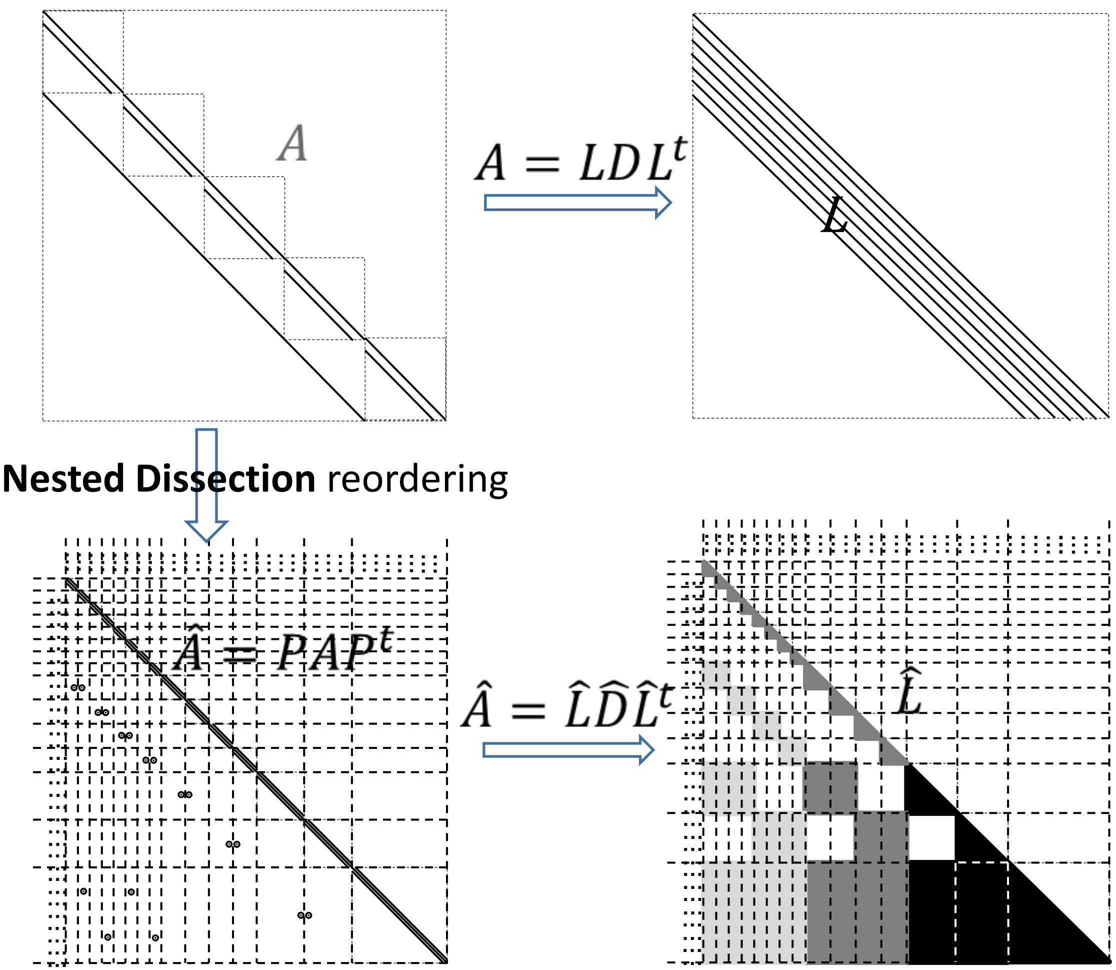
$$LV = B,$$
$$DL^tX = V$$

- Compression by using Hierarchically Semi Separable (HSS) formats and Low-Rank approximation help to ***reduce memory consumptions (and flops count)*** but lead to an approximate factorization

$$A \approx \tilde{L} \cdot \tilde{D} \cdot \tilde{L}^t$$

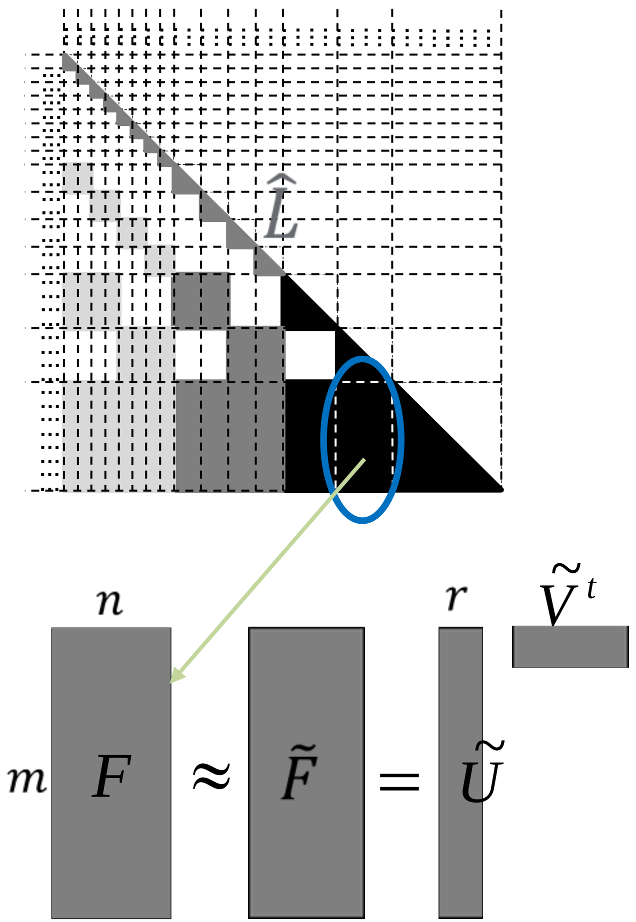
- The solution obtained with use of $\tilde{L}$ and $\tilde{D}$ instead of L and D may become inaccurate. To resolve the accuracy issue, the *iterative refinement* can be applied.
  - Provided the compression is not too aggressive, the remedy works. Otherwise, the iterations may diverge.

Sparsity of L-factor



- Straightforward LDLT factorization results in a band matrix L of $N^{\frac{5}{3}}$ nonzero elements.
- ND reordering reduces the number of nonzero elements to $N^{\frac{4}{3}}$.
- *Fill-in factors* (fractions of nonzero elements in blocks) varying from zero to one are shown in grey scale:
  - white blocks are purely zero;
  - the darker a block, the more nonzero elements it contains;
  - black blocks are (close to) dense

Low-rank approximation

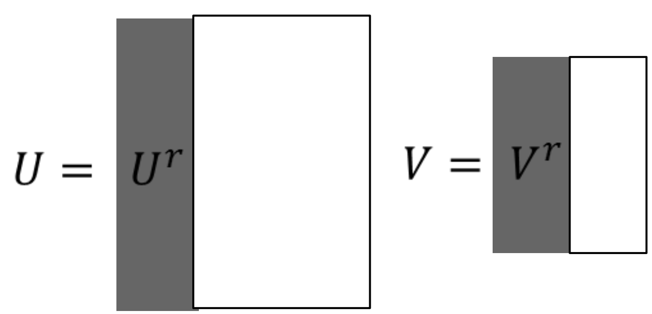


SVD-based solution

$$F = USV^t \qquad S = \begin{pmatrix} s_1 & & & & & \\ & \ddots & & & & \\ & & s_r & & & \\ & & & s_{r+1} & & \\ & & & & \ddots & \\ & & & & & s_n \end{pmatrix}$$

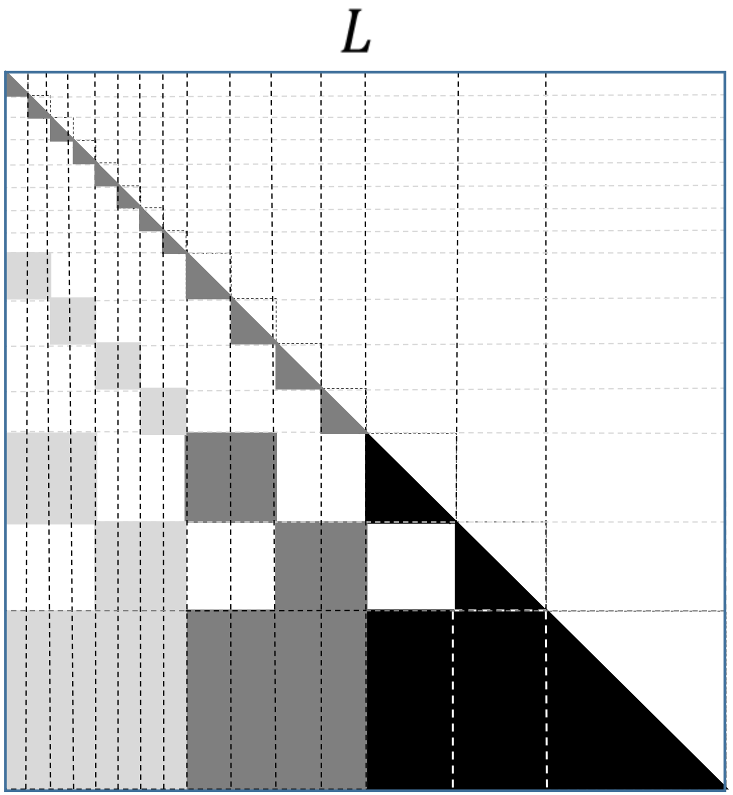
$$s_1 \geq s_1 \geq \cdots \geq s_n \geq 0$$

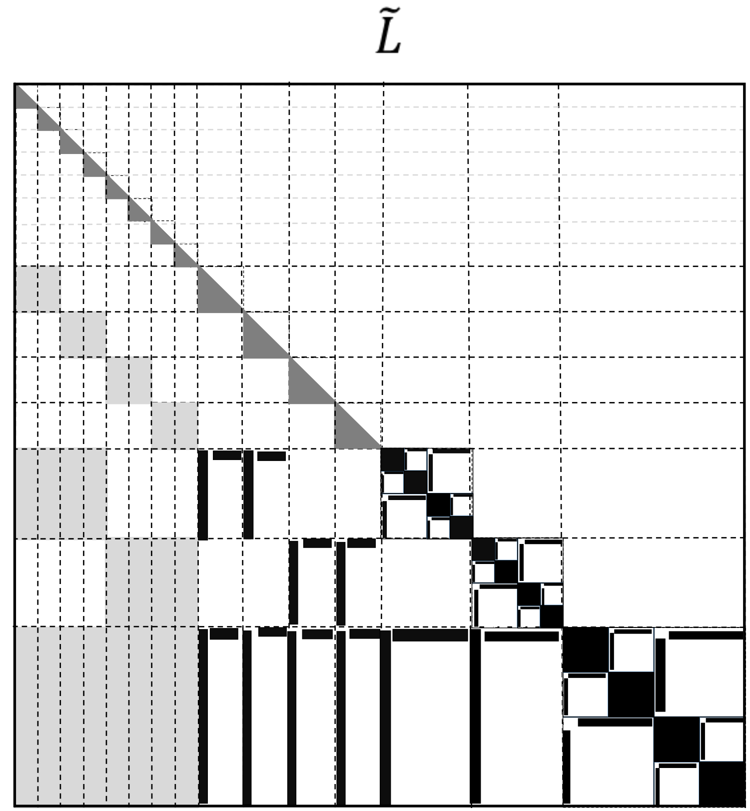
- Given threshold ε find r: $\dfrac{s_{r+1}}{s_1} < \varepsilon$
- Define $S^r = \begin{pmatrix} s_1 & & \\ & \ddots & \\ & & s_r \end{pmatrix}$,



$$\tilde{U} = U^r$$
$$\tilde{V} = V^r \cdot S^r$$

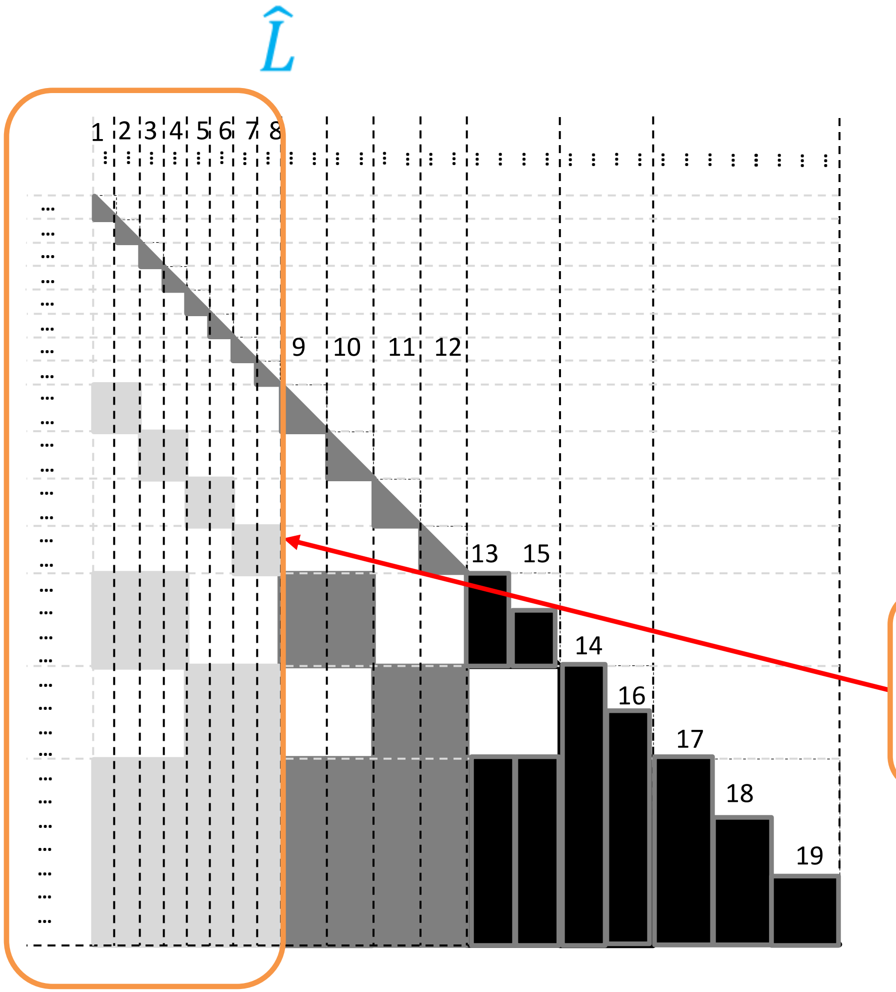
Compressed matrix structure

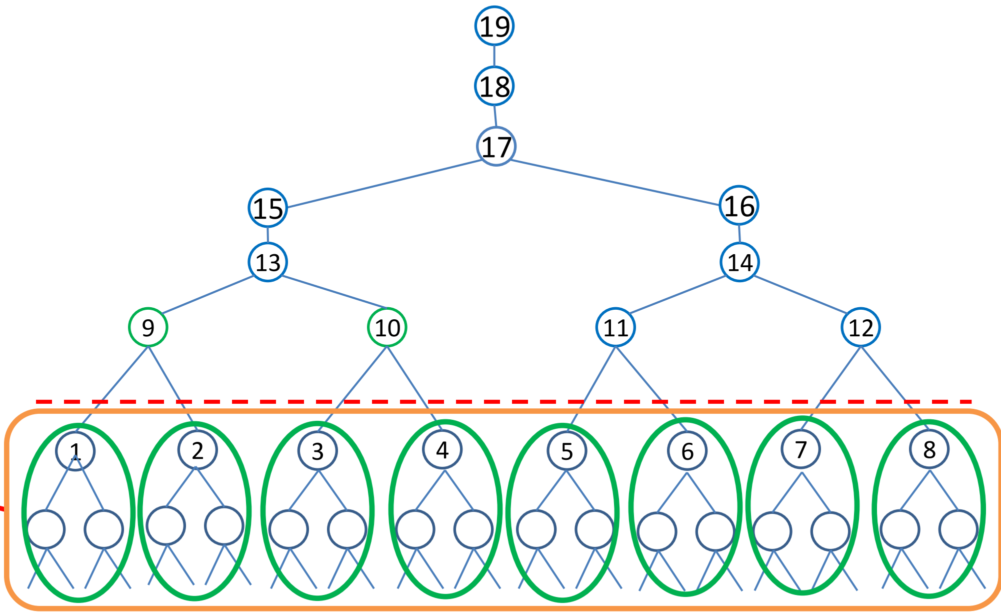




- ✓ Factorization time (5x speed up)
- ✓ Memory usage (5x compression)

Cluster implementation



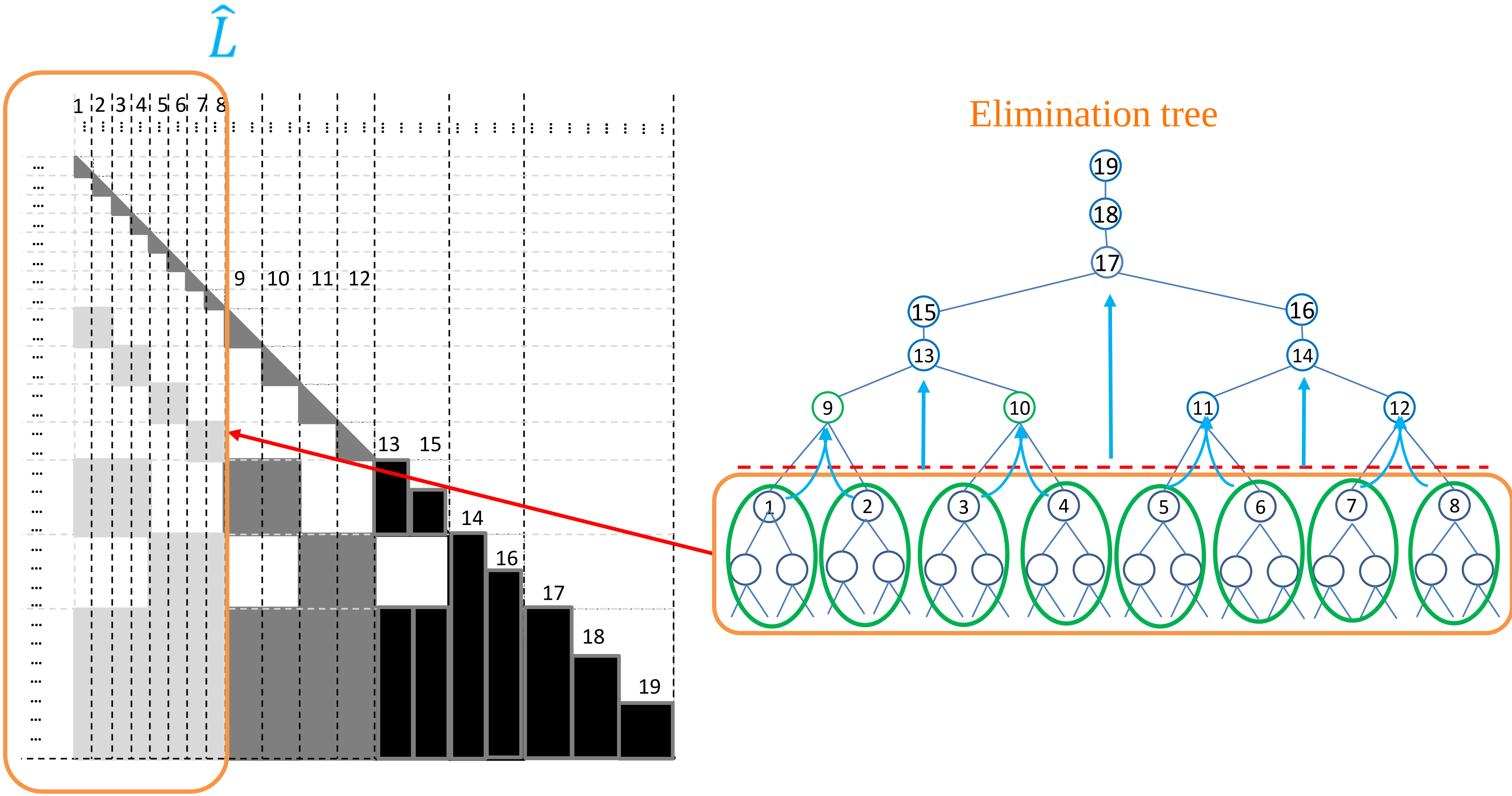


Parallel computations on 8 cluster nodes:

- ✓ Low-Rank compression
- ✓ Factorization

8 cluster nodes

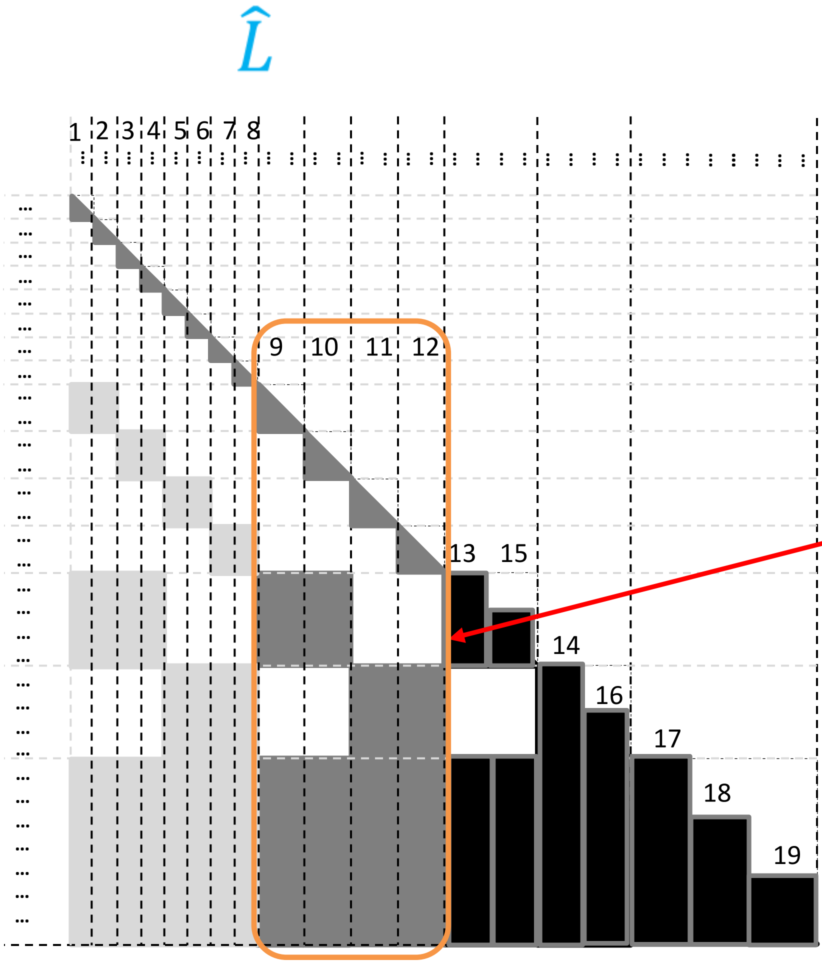
Cluster implementation

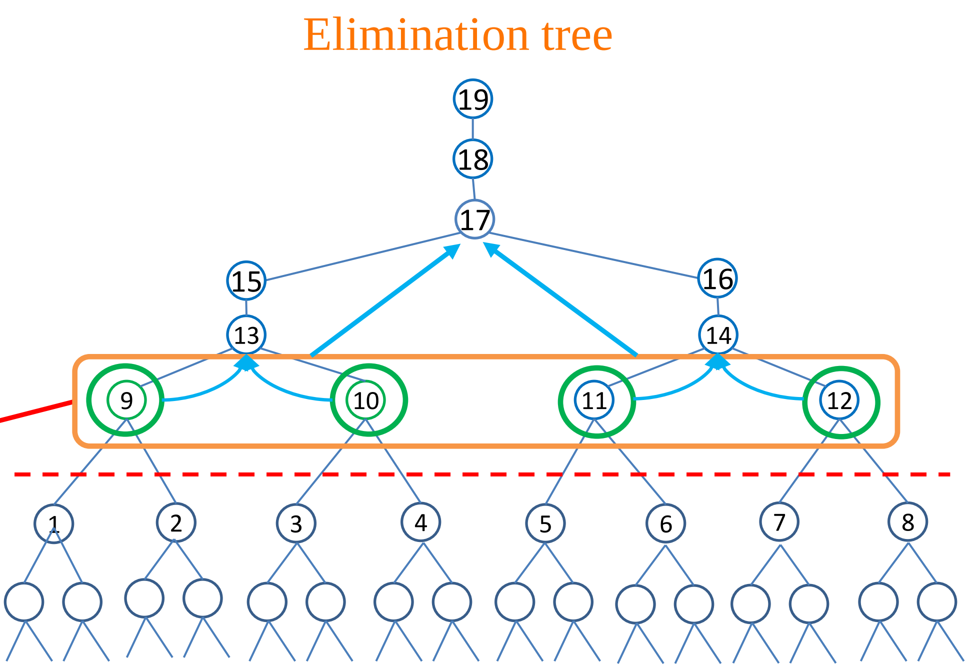


Parallel computations on 8 cluster nodes:

- ✓ Compute Schur complement
- ✓ Low-Rank compression

Cluster implementation

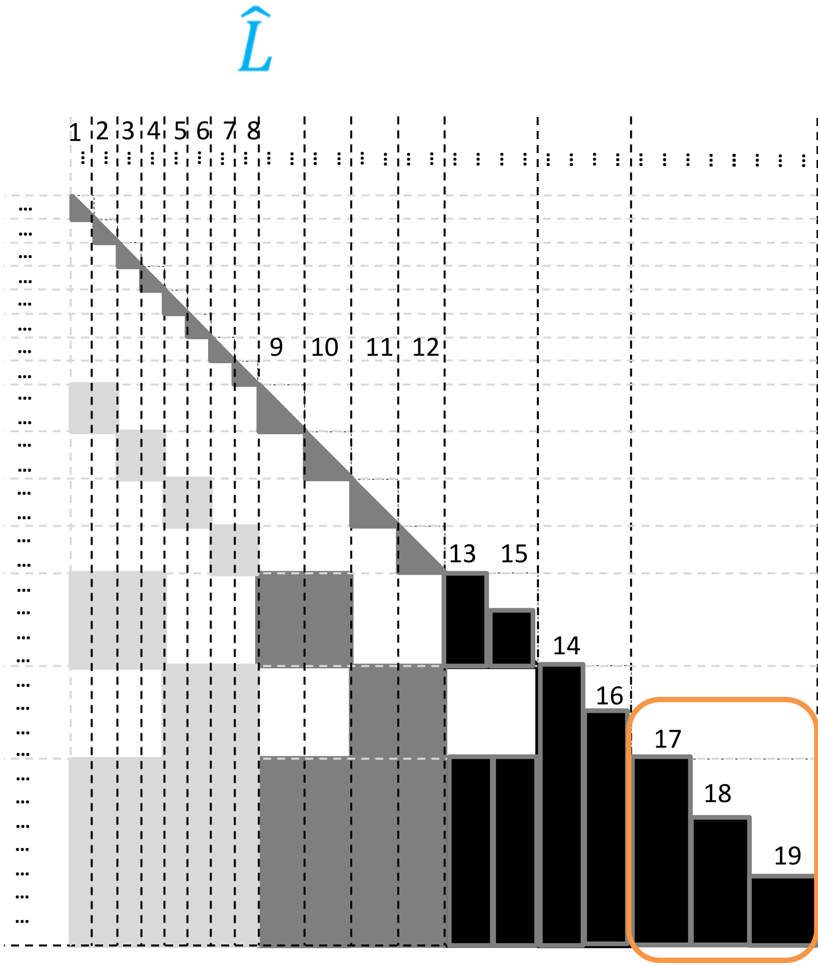


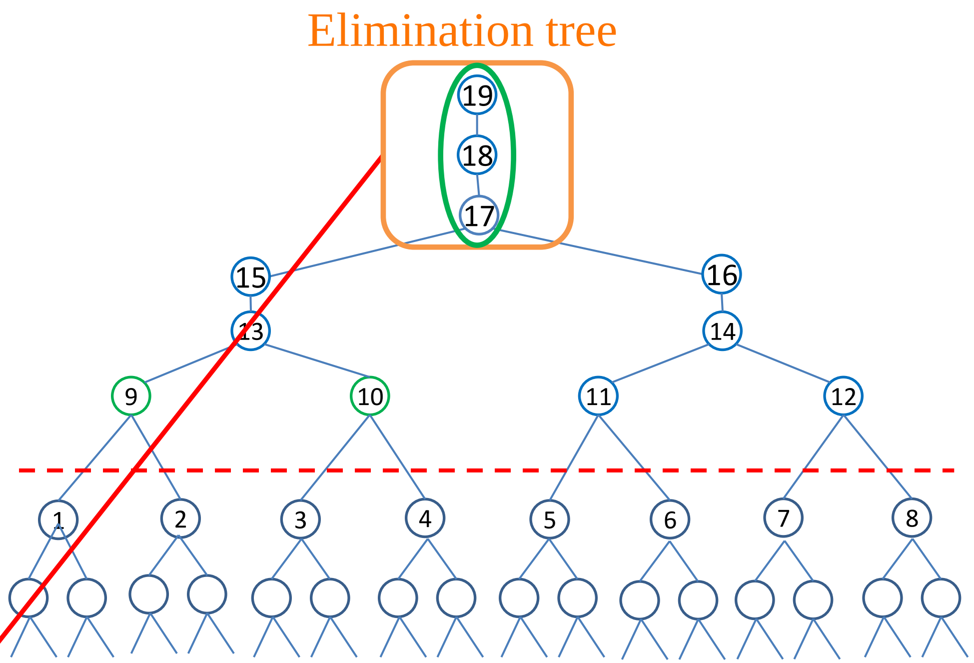


Parallel computations on 4 cluster nodes:

- ✓ ~~Compute Schur complement~~
- ✓ ~~Low-Rank compression~~
- ✓ Factorization

Cluster implementation





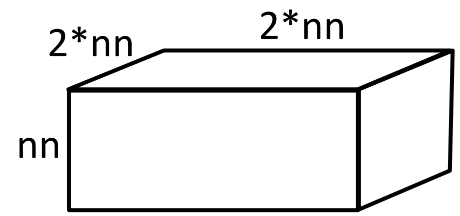
Computations on one cluster node:

✓ ~~Compute Schur complement~~

✓ ~~Low-Rank compression~~

✓ Factorization

Numerical experiments, tests descriptions



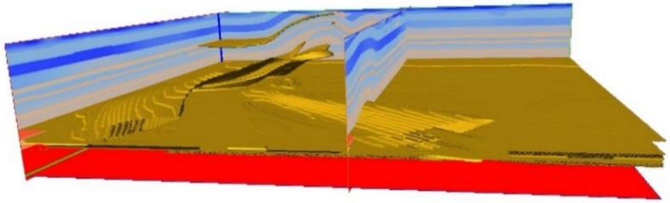
- ✓ Geometry: 3D domain ~nn*2nn*2nn
- ✓ Spatial step: const=h in each direction
- ✓ PML width=10points
- ✓ Eps_lowrank=**10^(-4.5)**

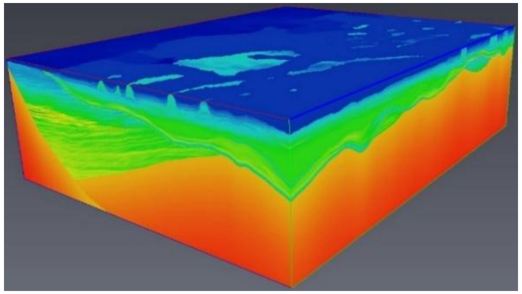
Homogeneous models

- various constant velocity models
- constant frequency

Heterogeneous models

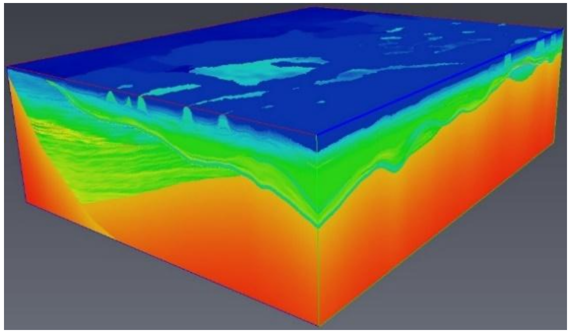
- Real high-contrast velocity models
- various frequencies





Real velocity model

- ✓ 3D domain – **30км* 39км* 11.5км**
- ✓ Velocity model **1043m/s** до **7628m/s**
- ✓ Frequency **2Hz**
- ✓ PML: **10** grid points
- ✓ Grid step h=50m
- ✓ Low-rank threshold is $\mathbf{10^{-4.5}}$



eqs=$\mathbf{123 \cdot 10^6}$:

- ✓ Factorization time:
  - **1h40m** (16 nodes)
  - **37m** (128 nodes)
- ✓ Solve time (per **128** sources):
  - **~3m** (16 nodes)
  - **~50s** (128 nodes)

Numerical results were obtained on Shaheen II:

(Intel® Xeon® CPU E5-2698 v3 @2.3 GHz, 128 GB RAM)

Factorization performance, compressibility factors

Table 1: Homogeneous medium, fixed frequency.

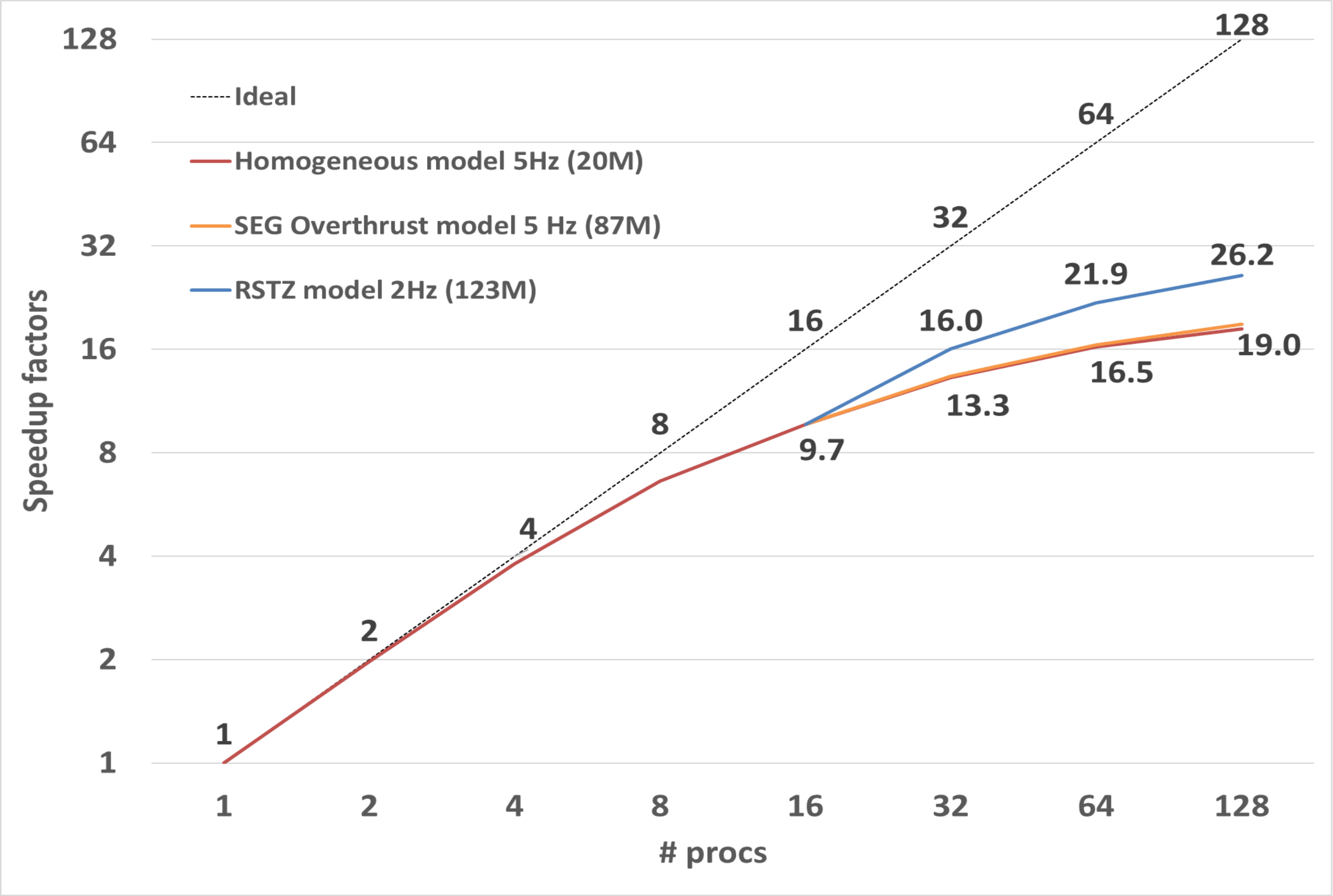
Table 1: Homogeneous medium (data for υ=2 Hz, $\varepsilon=10^{-4}$)				
Sound velocity, m/s.	1000	2000	4000	8000
Points per wavelength	10	20	40	80
Compressibility factor	4.1	5.3	5.8	6.0
Factorization time, s.	8 458	4 347	3 537	3 276

Table 2: RSTZ model. Notice the compressibility factor gets worth with increase of frequency, and factorization time respectively increases.

Table 2: RSZT model ($\varepsilon = 3 \cdot 10^{-5}$)				
ν (Hz)	1	2	4	8
Compressibility factor	5.9	5.8	5.3	4.1
Factorization time (s) on	3 427	3 638	4 578	9 949

32 cluster nodes

Factorization scalability

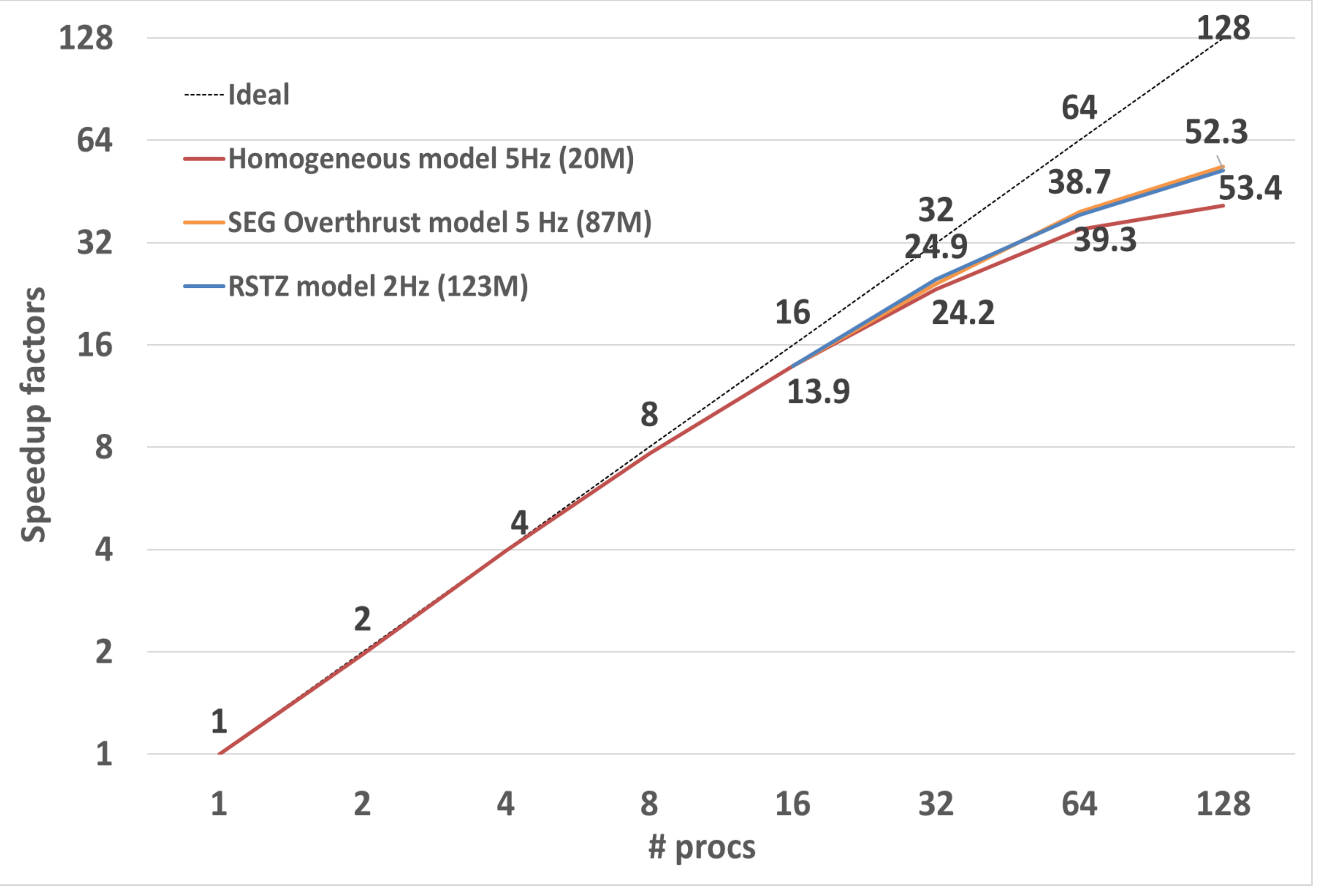


One MPI proc per node used.

Scalabilities are evaluated as factorization time ratios $s_n = \frac{t_1}{t_n}$ for one and n procs.

HW: Shaheen II @ KAUST (2× Intel® Xeon® CPU E5-2698 v3 @2.3 GHz per cluster node, 128 GB RAM).

Solving step scalability



- Data shown for 128 RHS vectors
- For RSTZ model, on 128 processes, solving step for one RHS vector takes 0.4 sec per vector.

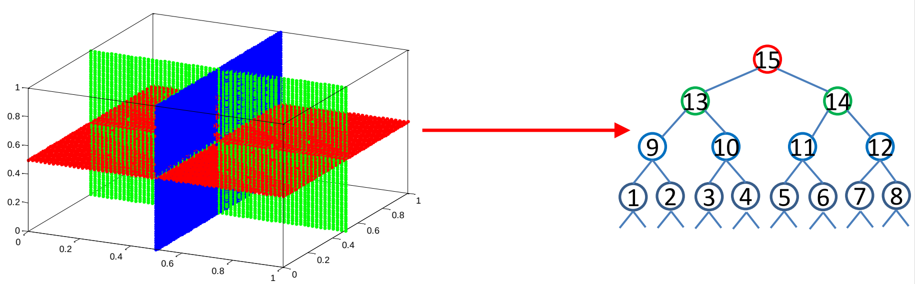
Scalability issue

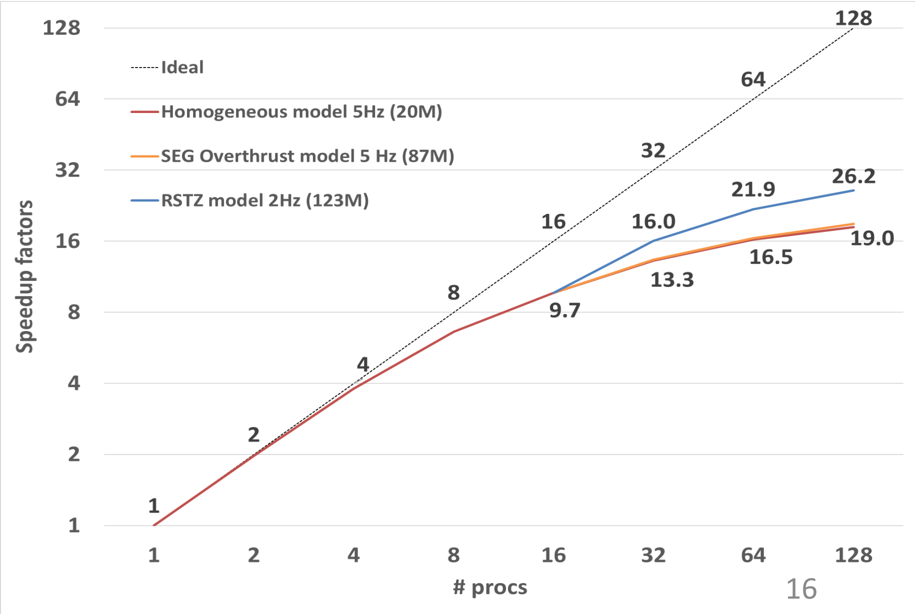
Reasons of poor factorization scalability on many nodes (>8):

✓ Weak parallelization of factorization the top-level nodes

✓ Different factorization jobs of low-level nodes

✓ High optimization the single-node version of solver => high performance of low-level nodes





Thank you for attention!