HIGH-PERFORMANCE RADIATIVE TRANSPORT EQUATION SOLVER

M. A. Badri, P. Jolivet, and Y. Favennec

10th FreeFem++ days
December 13

RADIATIVE TRANSFER

- neutron transport
- combustion
- optical tomography
- laser radiation...

Propagation of photons

- absorption
- scattering
- emission

INTEGRO-DIFFERENTIAL EQUATION

Spectral radiance

- space coordinates (x, y, z)
- solid angles (θ, ψ)
- wavelength λ
- time t

$$\implies \text{7D problem}$$

$$\frac{1}{c}\frac{\partial I_\lambda}{\partial t} + \vec{s} \cdot \nabla I_\lambda + (\kappa + \sigma) I_\lambda = \sigma \oint_\Omega \varphi \cdot I_\lambda \, \mathrm{d}\Omega + \kappa B_\lambda$$

STEADY-STATE MONOCHROMATIC RTE

$$\left(\vec{s} \cdot \nabla + (\kappa + \sigma)\right) I = \sigma \oint_{\Omega} \varphi \cdot I \, \mathrm{d}\Omega + \kappa B$$

- $\vec{s} = \begin{bmatrix} \sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix}^T$
- κ absorption coefficient
- σ scattering coefficient
- φ phase scattering function
- B black body emissivity function

STEADY-STATE MONOCHROMATIC RTE

absorption/scattering

$$\left(\vec{s}\cdot\nabla + \underbrace{(\kappa+\sigma)}\right) I = \sigma \oint_{\Omega} \varphi \cdot I \, \mathrm{d}\Omega + \kappa B$$

- $\vec{s} = \left[\sin\theta\cos\psi \quad \sin\theta\sin\psi \quad \cos\theta\right]^T$
- κ absorption coefficient
- σ scattering coefficient
- φ phase scattering function
- B black body emissivity function

STEADY-STATE MONOCHROMATIC RTE

$$\left(\vec{s} \cdot \nabla + \overbrace{(\kappa + \sigma)}^{\text{absorption/scattering}}\right) I = \overbrace{\sigma \oint_{\Omega} \varphi \cdot I \, \mathrm{d}\Omega}^{\text{scattering}} + \kappa B$$

- $\vec{s} = \begin{bmatrix} \sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix}^T$
- κ absorption coefficient
- σ scattering coefficient
- φ phase scattering function
- B black body emissivity function

STEADY-STATE MONOCHROMATIC RTE

$$\left(\vec{s} \cdot \nabla + \underbrace{(\kappa + \sigma)}_{\text{absorption/scattering}}\right) I = \underbrace{\sigma \oint_{\Omega} \varphi \cdot I \,\mathrm{d}\Omega}_{\text{scattering}} + \underbrace{\kappa B}_{\text{emission}}$$

- $\vec{s} = \begin{bmatrix} \sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix}^T$
- κ absorption coefficient
- σ scattering coefficient
- φ phase scattering function
- B black body emissivity function

NUMERICAL SCHEMES IN SPACE

Deterministic methods

- FVM (element-wise conservativity)
- FEM (flexibility)

Stochastic methods

- MC

NUMERICAL SCHEMES IN SPACE

Deterministic methods

- FVM (element-wise conservativity)
- FEM (flexibility)

Stochastic methods

- MC

$\Longrightarrow$ FEM with SUPG stabilization for the convection

NUMERICAL SCHEMES IN SPACE

Deterministic methods

- FVM (element-wise conservativity)
- FEM (flexibility)

Stochastic methods

- MC

$\Longrightarrow$ FEM with SUPG stabilization for the convection

DG also possible [Le Hardy, Favennec, Rousseau 2016]

NUMERICAL SCHEMES IN ANGLES

Discretization of the unit sphere

- quadrature rules
- surrogate angular mesh

NUMERICAL SCHEMES IN ANGLES

Discretization of the unit sphere

- quadrature rules
- surrogate angular mesh

Semi-discretized RTE

$\forall m \in [\![1; N_d]\!]$:

$$\left(\vec{s}_m \cdot \nabla + (\kappa + \sigma)\right) I_m = \sigma \sum_{n=1}^{N_d} \omega_n \varphi_{m,n} \cdot I_n + \kappa B$$

NUMERICAL SCHEMES IN ANGLES

Discretization of the unit sphere

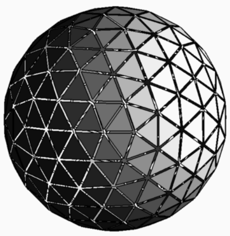
- quadrature rules
- surrogate angular mesh

Semi-discretized RTE

$\forall m \in [\![1; N_d]\!]$:

$$(\vec{s}_m \cdot \nabla + (\kappa + \sigma))\, I_m = \sigma \sum_{n=1}^{N_d} \omega_n \varphi_{m,n} \cdot I_n + \kappa B$$

$N_d = 80$



$N_d = 320$

ANGULAR DECOMPOSITION

VARIATIONAL FORMULATION

Multiplying by a test function, $\forall m \in [\![1; N_d]\!]$,

$$\int_{\mathcal{T}} \Bigg[\left(\vec{s}_m \cdot \nabla + (\kappa + \sigma)\right) I_m - \sigma \sum_{n=1}^{N_d} \omega_n \varphi_{m,n} \cdot I_n - \kappa B \Bigg] \left(v + \gamma \vec{s}_m \cdot \nabla v\right) = 0$$

VARIATIONAL FORMULATION

Multiplying by a test function, $\forall m \in [\![1; N_d]\!]$,

$$\int_{\mathcal{T}} \Bigg[\left(\vec{s}_m \cdot \nabla + (\kappa + \sigma)\right) I_m - \sigma \sum_{n=1}^{N_d} \omega_n \varphi_{m,n} \cdot I_n - \kappa B \Bigg] (v + \gamma \vec{s}_m \cdot \nabla v) = 0$$

Compact form, $\forall m \in [\![1; N_d]\!]$,

$$a_m^{(d)}(I_m, v) + \sum_{n=1}^{N_d} a_{m,n}(I_n, v) = L_m(v)$$

MATRICIAL FORMULATION

Given a FE space $\mathcal{V}_h$ with N_h d.o.f.

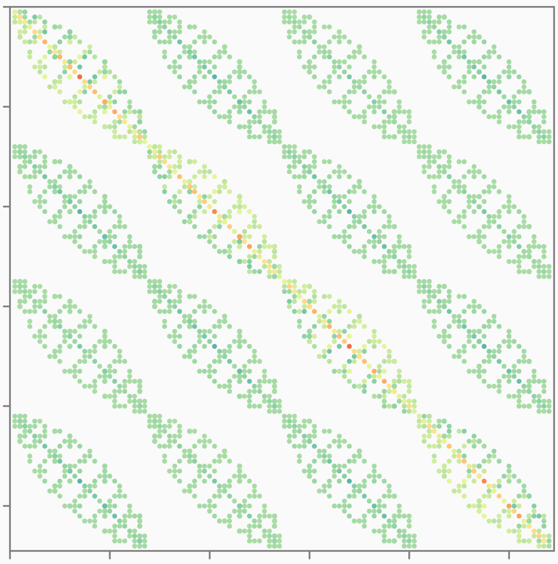
$$\begin{bmatrix} A_1^{(d)} + A_{1,1} & \cdots & A_{1,N_d} \\ \vdots & \ddots & \vdots \\ A_{N_d,1} & \cdots & A_{N_d}^{(d)} + A_{N_d,N_d} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{N_d} \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_{N_d} \end{bmatrix}$$

MATRICIAL FORMULATION

Given a FE space $\mathcal{V}_h$ with N_h d.o.f.

$$\begin{bmatrix} A_1^{(d)} + A_{1,1} & \cdots & A_{1,N_d} \\ \vdots & \ddots & \vdots \\ A_{N_d,1} & \cdots & A_{N_d}^{(d)} + A_{N_d,N_d} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{N_d} \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_{N_d} \end{bmatrix}$$

$\Longrightarrow$ global linear system of order $N_d \cdot N_h$



SPATIAL DECOMPOSITION

VECTORIAL VARIATIONAL FORMULATION

Given a FE space $\mathcal{V}_h^{N_d} = \mathcal{V}_h \times \mathcal{V}_h \times \cdots \times \mathcal{V}_h$,

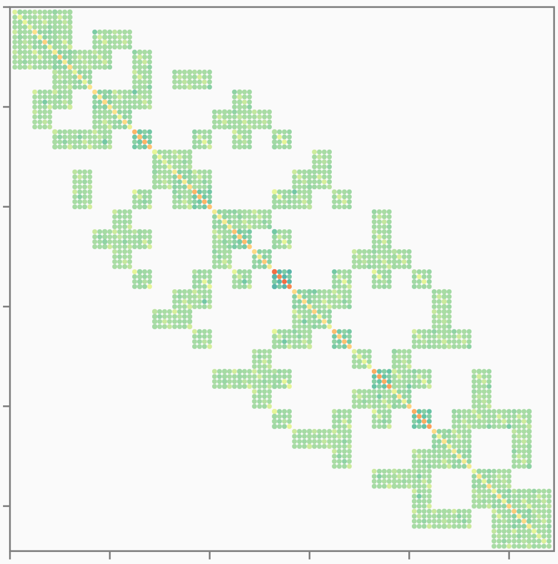
$$\int_{\mathcal{T}} \left[\vec{\mathbb{S}}^{\dagger} \nabla \mathbb{I} + (\kappa + \sigma)\mathbb{I} - \Phi : \mathbb{I} \right] \odot \left(\mathbb{V} + \gamma \vec{\mathbb{S}}^{\dagger} \nabla \mathbb{V} \right) = 0$$

with

$$\Phi = \sigma \begin{bmatrix} \omega_1 \varphi_{1,1} & \omega_2 \varphi_{1,2} & & \omega_{N_d} \varphi_{1,N_d} \\ \omega_1 \varphi_{2,1} & \omega_2 \varphi_{2,2} & & \omega_{N_d} \varphi_{2,N_d} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_1 \varphi_{N_d,1} & \omega_2 \varphi_{N_d,2} & & \omega_{N_d} \varphi_{N_d,N_d} \end{bmatrix}$$

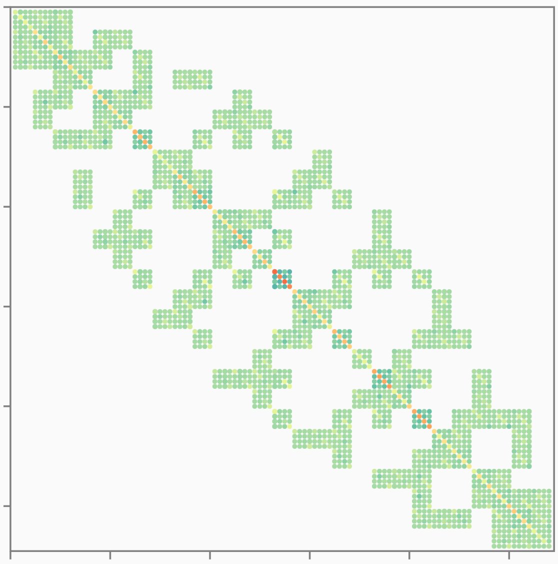
MATRICIAL FORMULATION

$\Longrightarrow$ global linear system of order $N_d \cdot N_h$



MATRICIAL FORMULATION

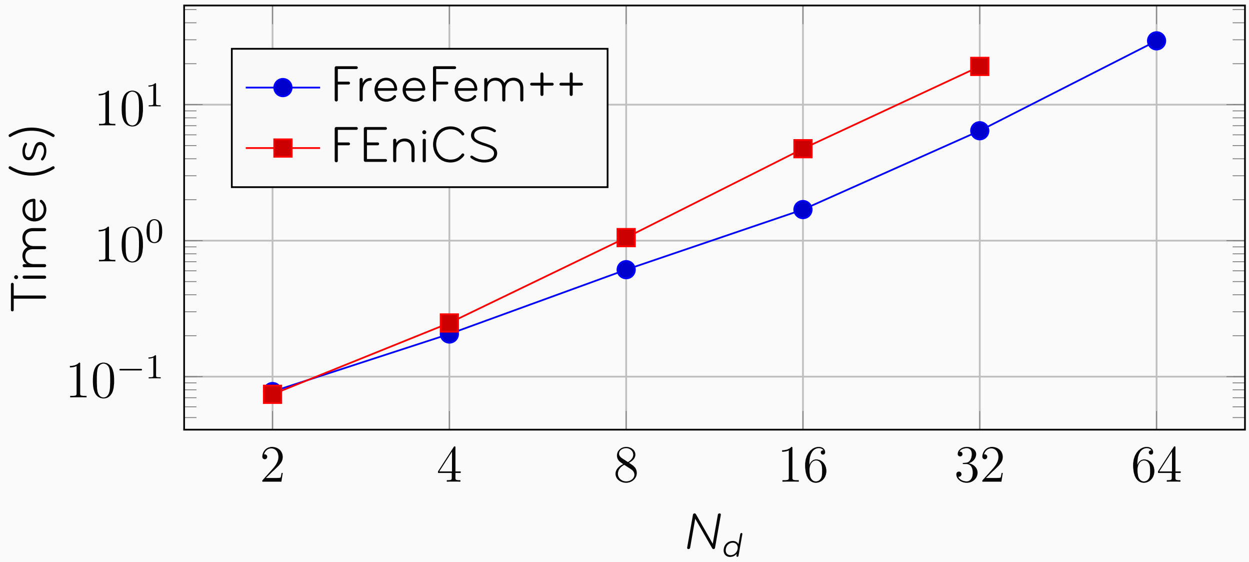
$\Longrightarrow$ global linear system of order $N_d \cdot N_h$



- interleaved degrees of freedom between components
- sparse matrix with dense blocks of order N_d
- number of nonzeros grows quadratically with N_d

MATRIX-FREE SOLVER

ASSEMBLY WITH GENERAL PURPOSE DSL



Fixed spatial mesh with 6,768 triangles

$\Longrightarrow$ curse of dimensionality

MATRIX-FREE STRATEGY

Ideas

- don't deal with a huge variational formulation
- only assemble elementary matrices
- compute the global matrix–vector product using tensors

MATRIX-FREE STRATEGY

Ideas

- don't deal with a huge variational formulation
- only assemble elementary matrices
- compute the global matrix–vector product using tensors

Pros

- memory gains
- cheaper assembly
- more efficient matrix–vector products

MATRIX-FREE STRATEGY

Ideas

- don't deal with a huge variational formulation
- only assemble elementary matrices
- compute the global matrix–vector product using tensors

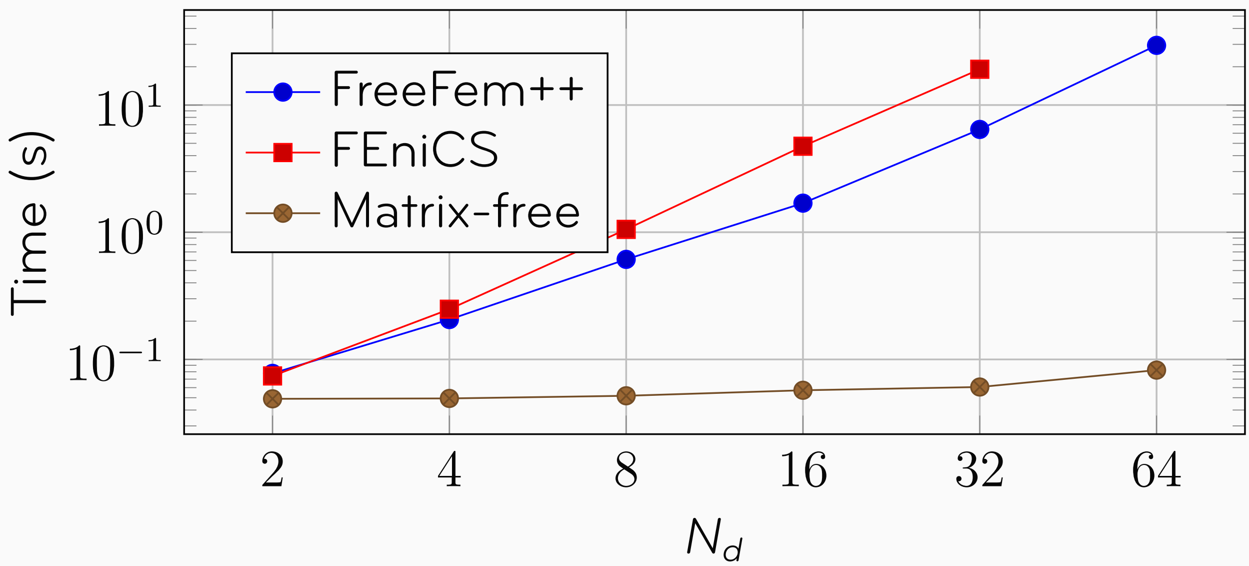
Pros

- memory gains
- cheaper assembly
- more efficient matrix–vector products

Cons

- no off the shelf preconditioner

PERFORMANCE OF THE ASSEMBLY



Fixed spatial mesh with 6,768 triangles

PERFORMANCE OF THE MATRIX–VECTOR PRODUCT

N_d	$\mathrm{GFLOP}_{\mathrm{MR}}$
80	6
128	14
320	90
512	229
1,280	1,434
2,048	3,670

- theoretical MR complexity is absurdly high ($\approx$ 1 min/MV)

PERFORMANCE OF THE MATRIX–VECTOR PRODUCT

N_d	GFLOP_{MR}	Time_{MF} (ms)
80	6	96
128	14	162
320	90	422
512	229	756
1,280	1,434	2,980
2,048	3,670	6,607

- theoretical MR complexity is absurdly high ($\approx$ 1 min/MV)
- practical MF performance dominated by `gemm`

PERFORMANCE OF THE MATRIX–VECTOR PRODUCT

N_d	GFLOP_{MR}	Time_{MF} (ms)	GFLOP_{MF}
80	6	96	1
128	14	162	2
320	90	422	7
512	229	756	14
1,280	1,434	2,980	67
2,048	3,670	6,607	156

- theoretical MR complexity is absurdly high ($\approx$ 1 min/MV)
- practical MF performance dominated by `gemm`
- high fraction of the peak of an Intel SKL ($\approx$ 26 GFLOP/s)

MATRIX-FREE PRECONDITIONING

Implicit

- geometric MG
- element-wise preconditioning

MATRIX-FREE PRECONDITIONING

Implicit

- geometric MG
- element-wise preconditioning

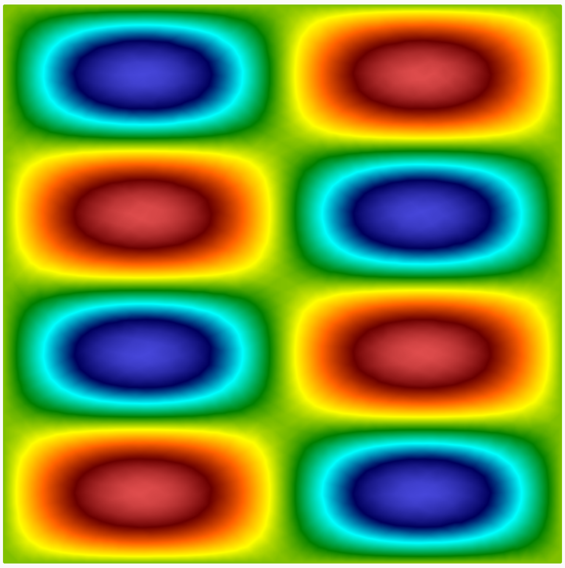
Explicit

- "pruned" operator with diagonal dense blocks

$$\Phi = \sigma \begin{bmatrix} \omega_1 \varphi_{1,1} & 0 & & 0 \\ 0 & \omega_2 \varphi_{2,2} & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & & \omega_{N_d} \varphi_{N_d,N_d} \end{bmatrix}$$

NUMERICAL RESULTS

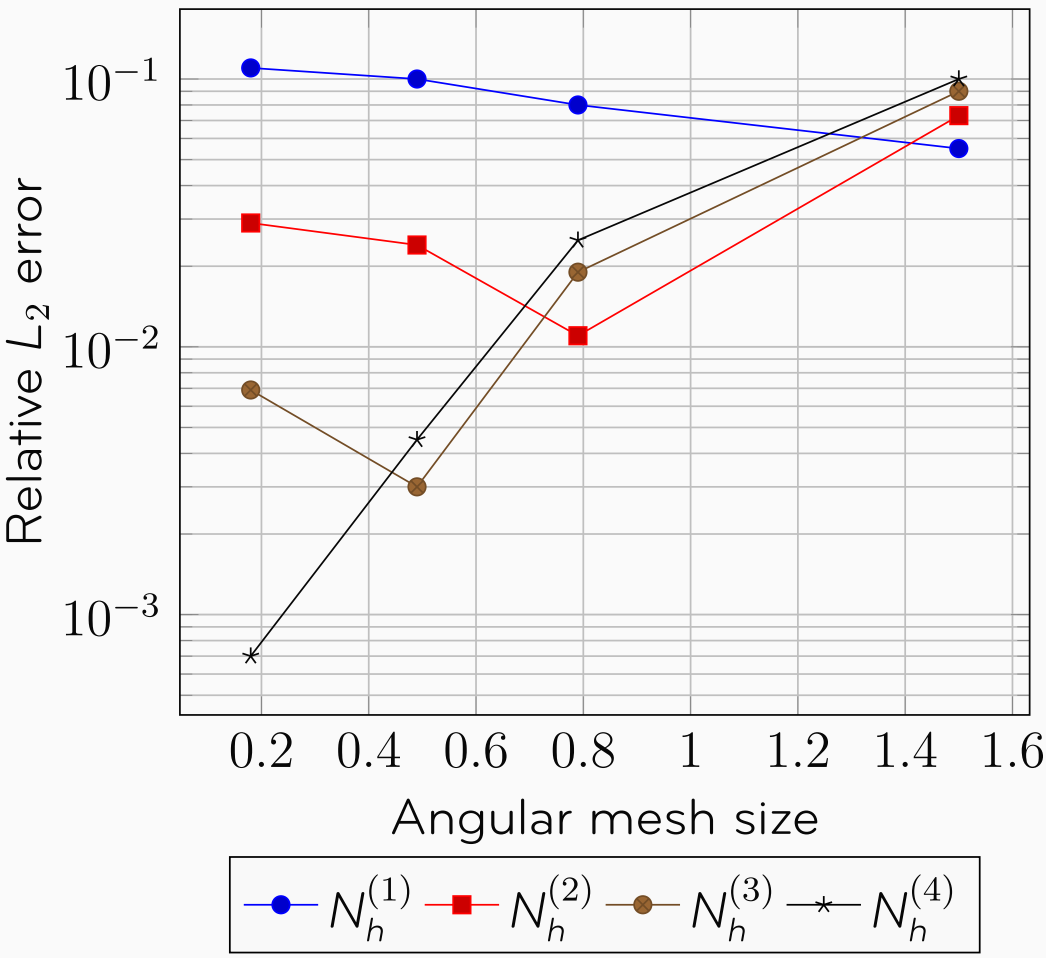
MANUFACTURED SOLUTION



$$D = 2\pi\left(1 + \sin(2\pi x)\sin(4\pi y)\right)$$

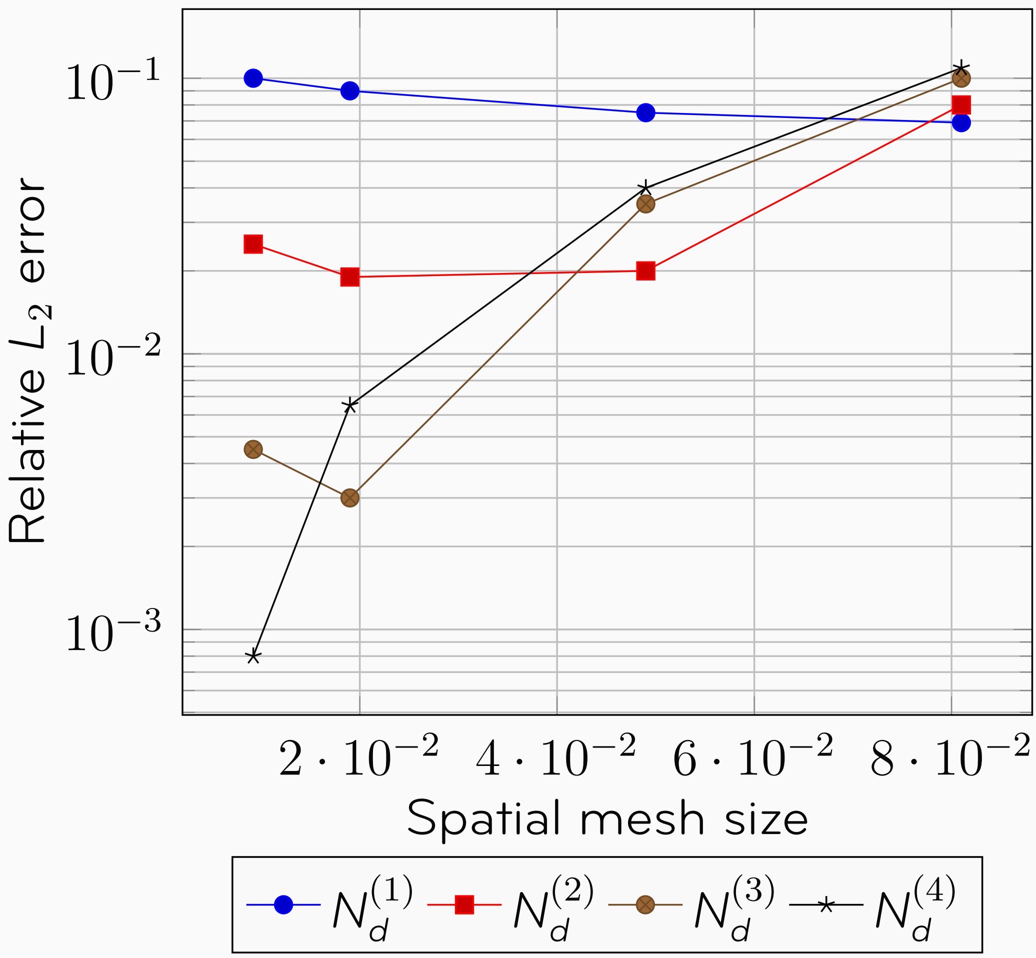
MANUFACTURED SOLUTION

Angular mesh refinement



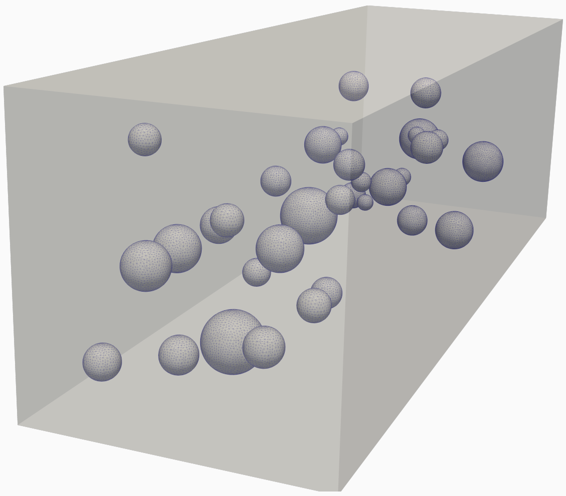
MANUFACTURED SOLUTION

Spatial mesh refinement

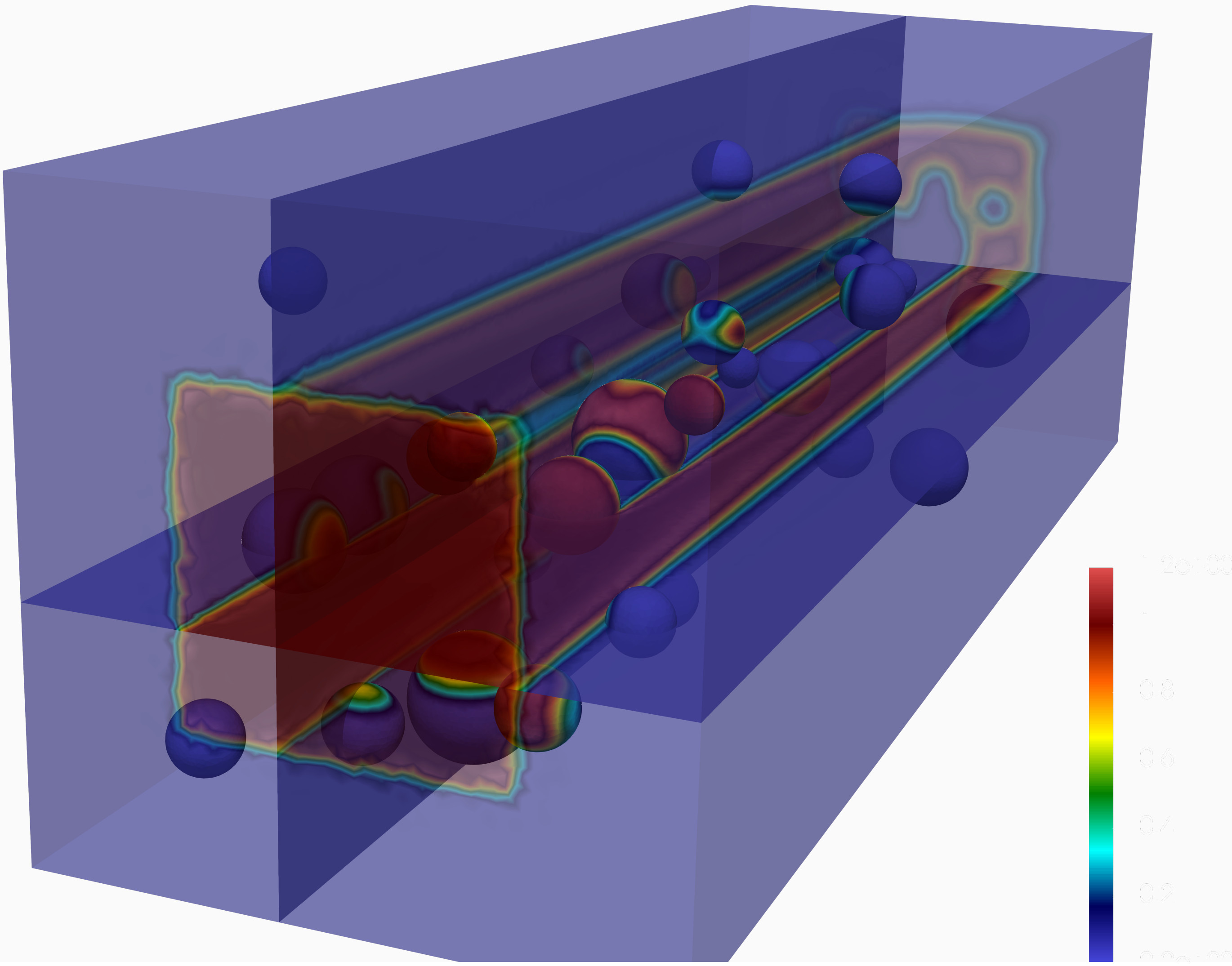


LARGE-SCALE EXPERIMENTS

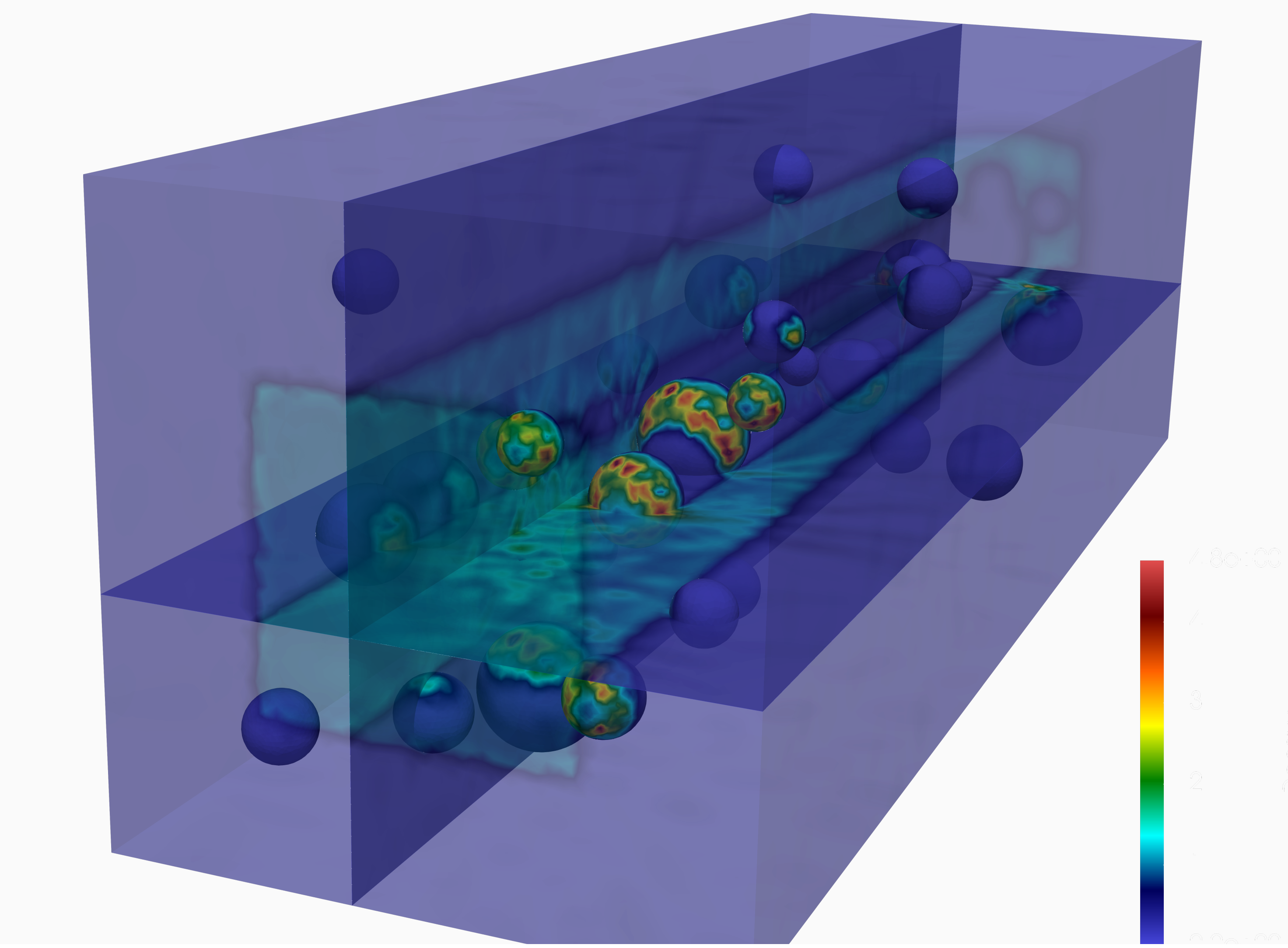
- Henyey–Greenstein phase scattering function φ
- transparent ($\kappa = \sigma = 10^{-6}$) material with absorbing and scattering inclusions
- MKL PARDISO used as a subdomain solver
- three million tetrahedra, $N_d = 2{,}048$



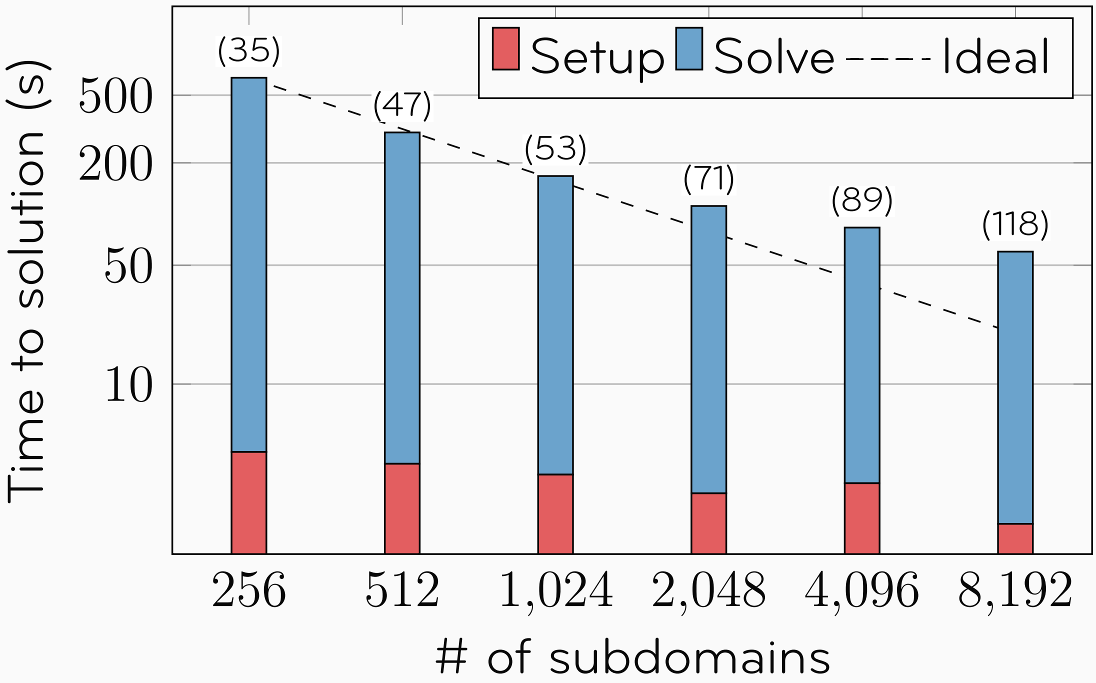
WITHOUT REFLECTION



WITH REFLECTIONS

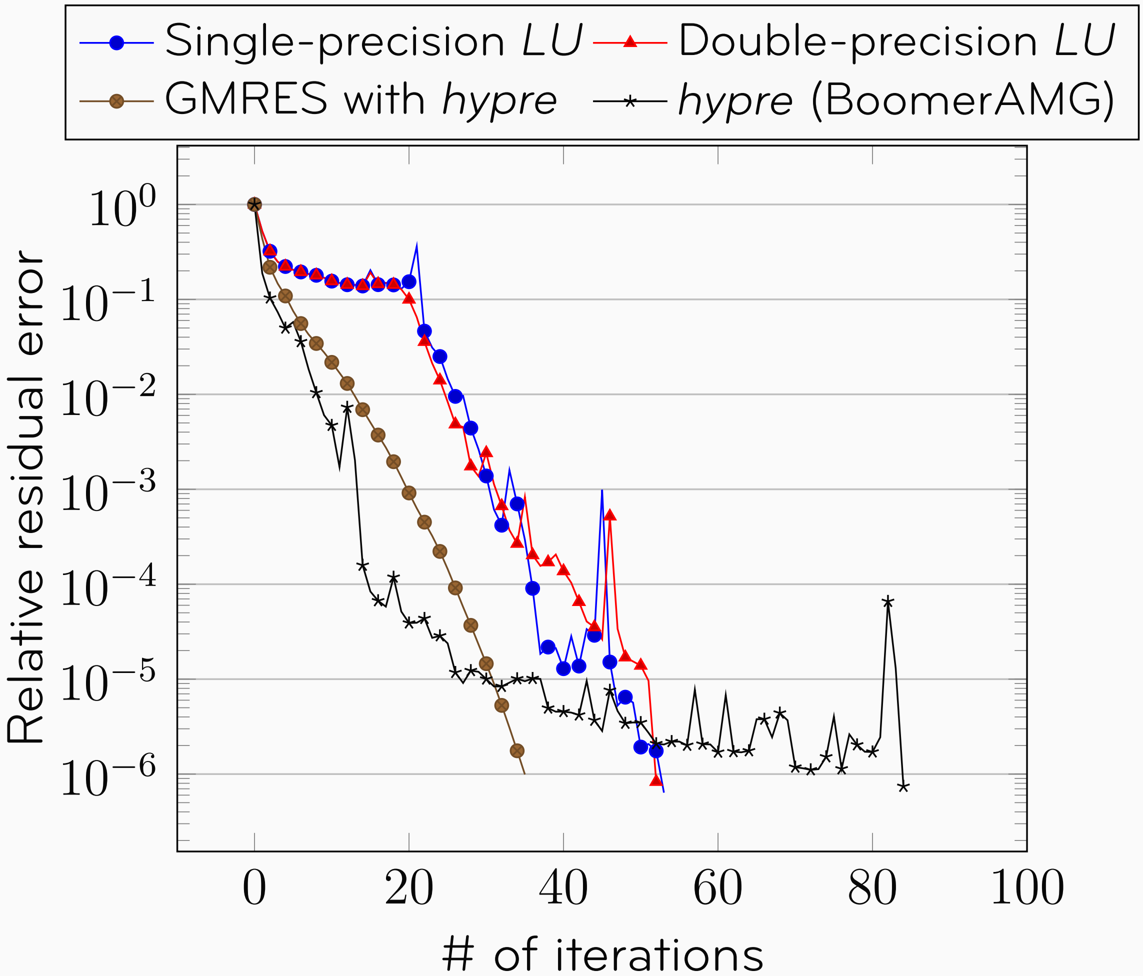


SCALING



1.1 billion unknowns, 36×10^{12} nnz

CHOOSING THE CORRECT PRECONDITIONER



TUNING BOOMERAMG

- default PETSc parameters
- block structure

TUNING BOOMERAMG

- default PETSc parameters
- block structure

- l_1 forward/backward Gauss–Seidel
- aggressive coarsening
- angular domain decomposition

$$\Phi = \sigma \begin{bmatrix} \omega_1 \varphi_{1,1} & & \\ & \ddots & \\ & & \omega_{N_d} \varphi_{N_d,N_d} \end{bmatrix}$$

TUNING BOOMERAMG

- default PETSc parameters
- block structure

- l_1 forward/backward Gauss–Seidel
- aggressive coarsening
- angular domain decomposition

$$\Phi_1 = \sigma \begin{bmatrix} \omega_1 \varphi_{1,1} & & \\ & \ddots & \\ & & \omega_{\frac{N_d}{2}} \varphi_{\frac{N_d}{2},\frac{N_d}{2}} \end{bmatrix}$$

$$\Phi_2 = \sigma \begin{bmatrix} \omega_{\frac{N_d}{2}+1} \varphi_{\frac{N_d}{2}+1,\frac{N_d}{2}+1} & & \\ & \ddots & \\ & & \omega_{N_d} \varphi_{N_d,N_d} \end{bmatrix}$$

BOOMERAMG STATISTICS

```
             nonzero         entries per row
lev   rows   entries  sparse  min  max   avg
=============================================
 0 66204390 986172550  0.000     6   33  14.9
 1 5984174 180411300  0.000     3   80  30.1
 2  496183 19823983  0.000     2   92  40.0
 3   39630  1625090  0.001     3   89  41.0
 4    2739    87273  0.012     4   78  31.9
 5     164     1774  0.066     4   21  10.8
 6      19       37  0.102     1    3   1.9

     Complexity:    grid = 1.098527
                operator = 1.204781
                  memory = 1.328484
```

LARGEST FREEFEM++ SIMULATION?

- 40×10^6 tetrahedra and $N_d = 5{,}120$

LARGEST FREEFEM++ SIMULATION?

- 40×10^6 tetrahedra and $N_d = 5{,}120$
- 33×10^9 unknowns (PETSc with 64 bit integers)

LARGEST FREEFEM++ SIMULATION?

- 40×10^6 tetrahedra and $N_d = 5{,}120$
- 33×10^9 unknowns (PETSc with 64 bit integers)
- 2.5×10^{15} nonzero entries

LARGEST FREEFEM++ SIMULATION?

- 40×10^6 tetrahedra and $N_d = 5{,}120$
- 33×10^9 unknowns (PETSc with 64 bit integers)
- 2.5×10^{15} nonzero entries
- 20 PB of memory to store the matrix (impossible)

LARGEST FREEFEM++ SIMULATION?

- 40×10^6 tetrahedra and $N_d = 5{,}120$
- 33×10^9 unknowns (PETSc with 64 bit integers)
- 2.5×10^{15} nonzero entries
- 20 PB of memory to store the matrix (impossible)
- 330 seconds on 24,576 processes of Irene@CEA

CLOSING REMARKS

- large-scale RTE solver using FreeFem++ and PETSc
- mixed matrix-free/matrix-ready method

CLOSING REMARKS

- large-scale RTE solver using FreeFem++ and PETSc
- mixed matrix-free/matrix-ready method

Thank you!