Augmented Partial Factorization: efficient computation of the generalized scattering matrix

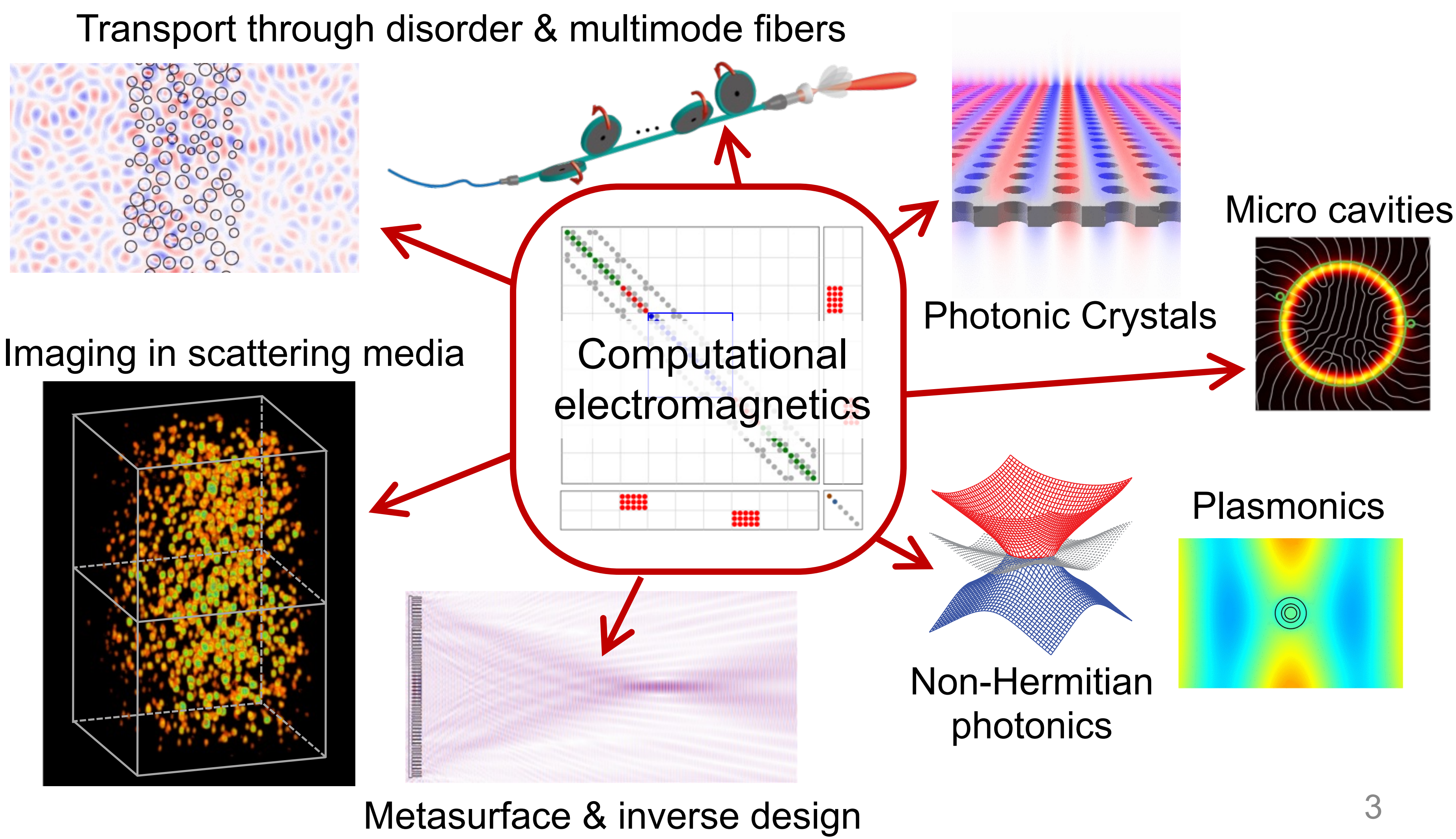
Chia Wei (Wade) Hsu

Ming Hsieh Department of Electrical and Computer Engineering
University of Southern California

Optics in Complex Systems Group @ USC

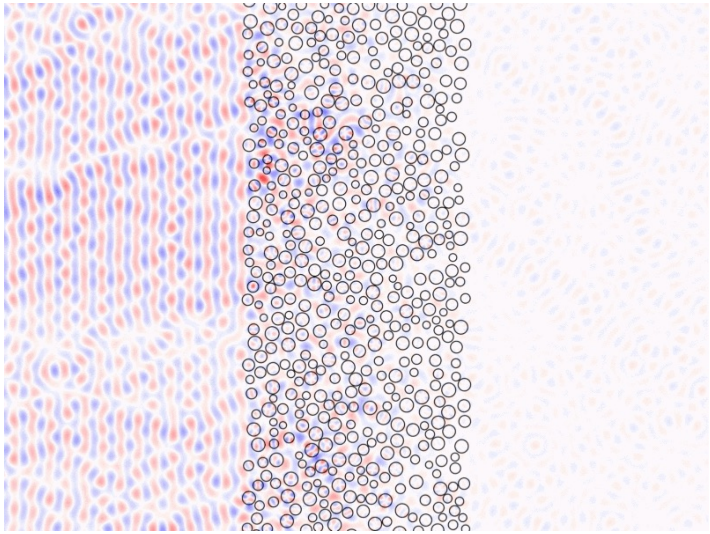
Optics in Complex Systems Group @ USC

Optical systems that couple *many degrees of freedom*
(spatial/angular, temporal/spectral, etc)

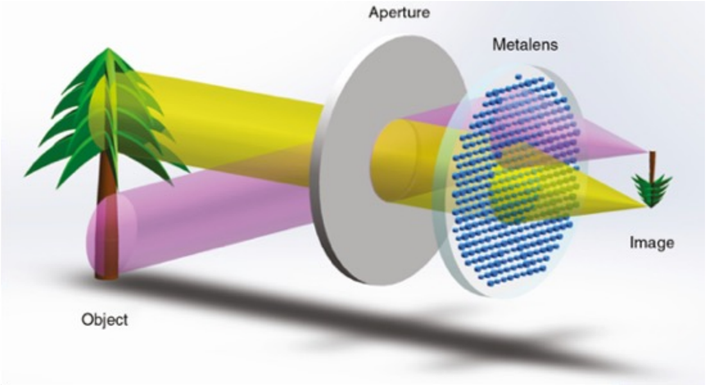


Multi-channel optical systems

Disordered Media

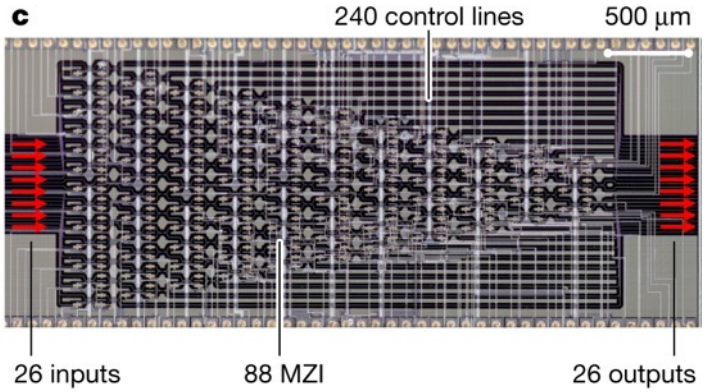


Metasurfaces



Engelberg et al, Nanophotonics (2020)

Photonic Circuits



N. Harris et al, Nature Photonics (2017)

Multi-mode Fibers

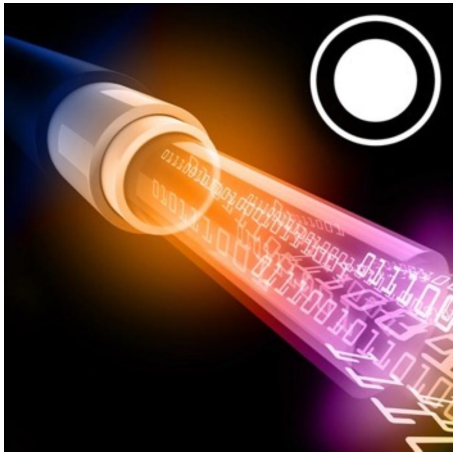
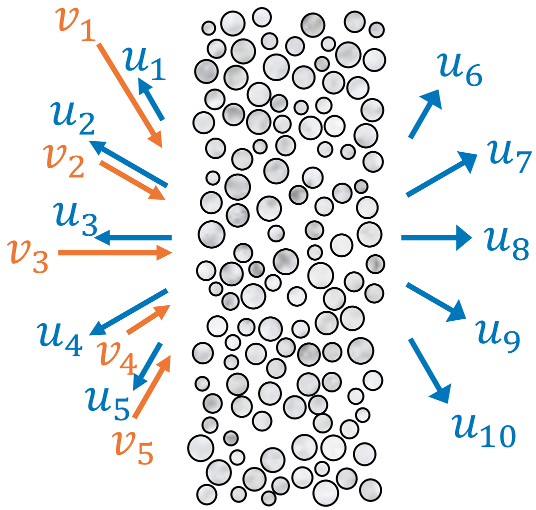


Image from Coherent

The "scattering matrix"



output $\qquad\qquad\qquad\qquad$ input

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{M'} \end{bmatrix} = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,M} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ S_{M',1} & S_{M',2} & \cdots & S_{M',M} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_M \end{bmatrix}$$

$M' \times M$ ***scattering matrix*** $\mathbf{S}(\omega)$

In two-sided systems, $\mathbf{S} = \begin{bmatrix} \mathbf{r} & \mathbf{t}' \\ \mathbf{t} & \mathbf{r}' \end{bmatrix}$, where $\mathbf{r}$ = reflection matrix, $\mathbf{t}$ = transmission matrix.

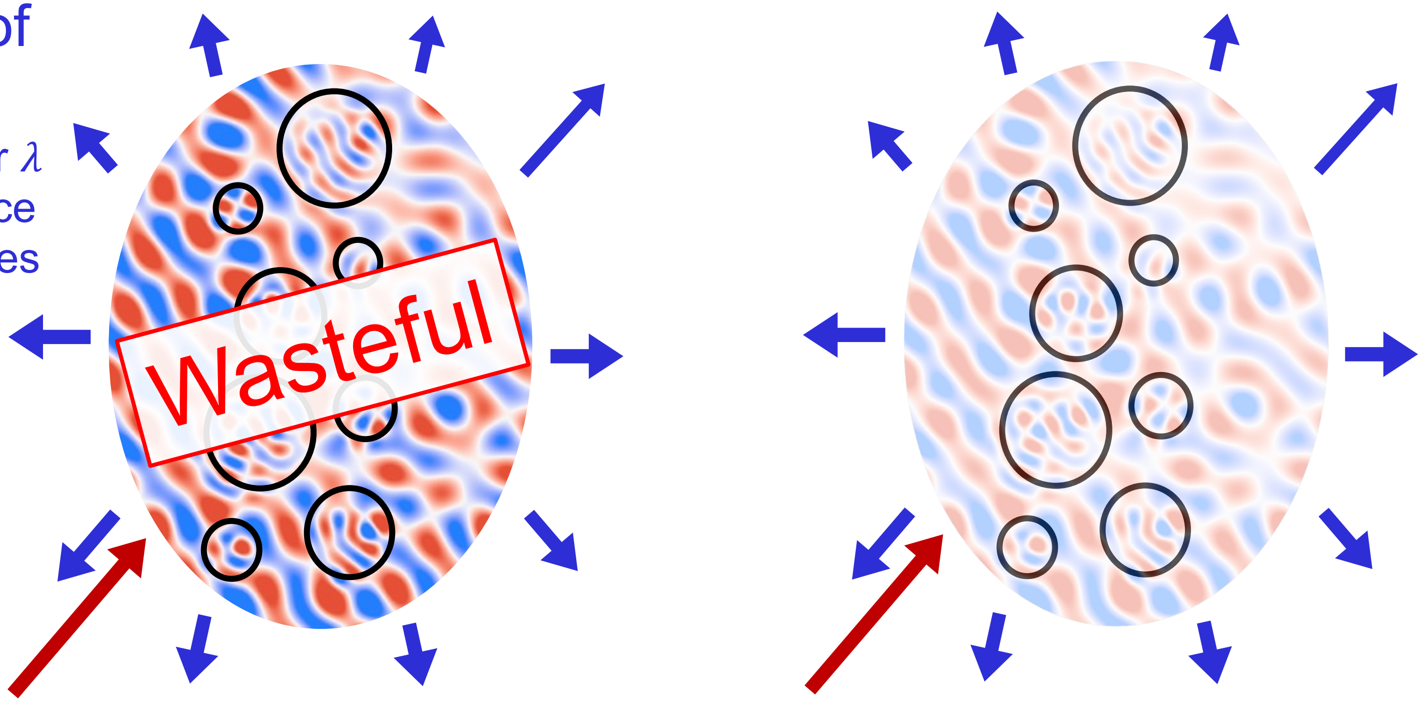
$\mathbf{S}(\omega)$ fully encapsulates the multi-channel response.

But **computing $\mathbf{S}(\omega)$ is a major challenge**.

1) Redundancy in field computation

Project onto outputs of interest
2 outputs per λ on the surface
⇒10 variables

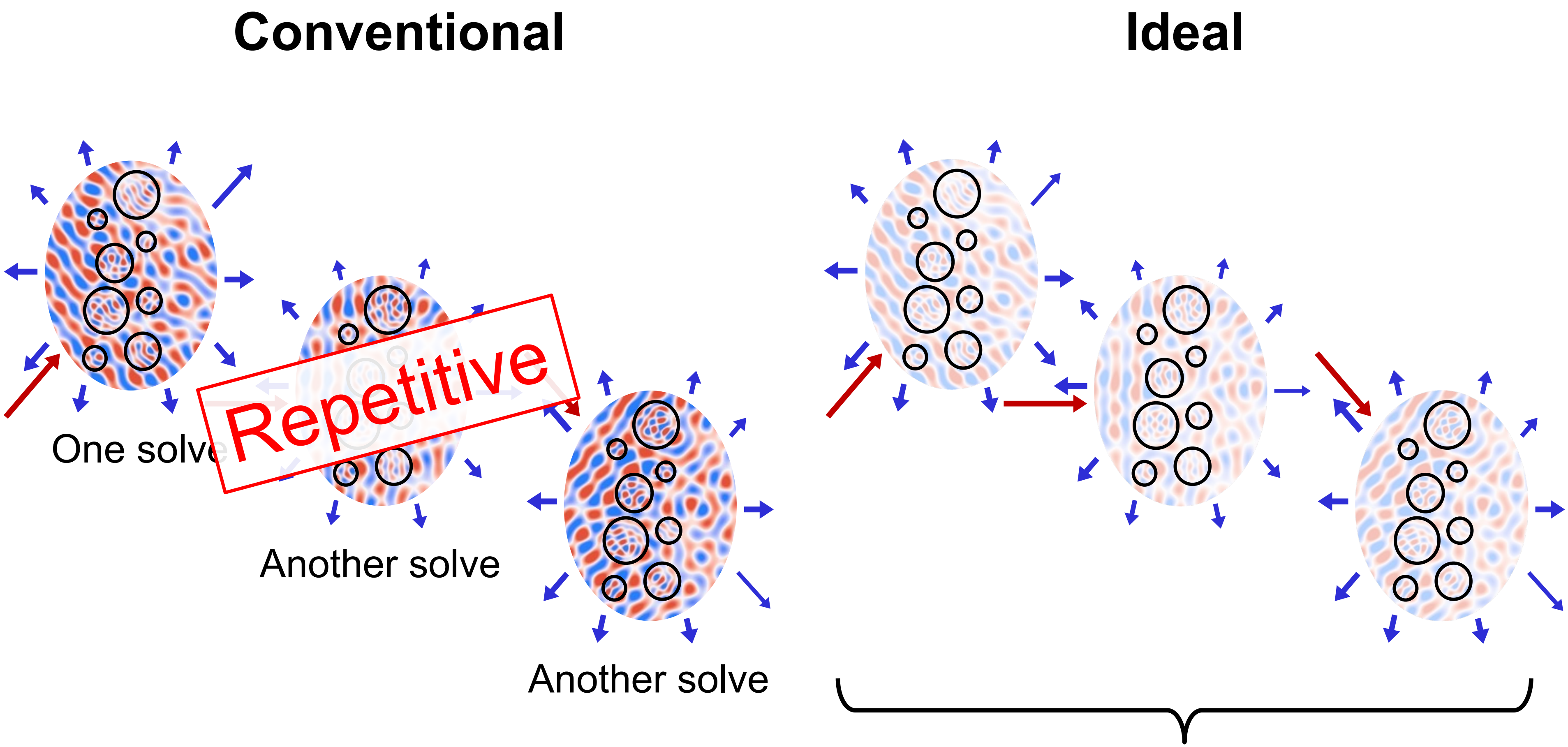
Conventional | **Ideal**



Compute full field profile
20~40 pixels per λ in the volume
⇒100,000 variables

Only compute the outputs of interest

2) Repetitions over inputs



Many steps are *repeated*
(when nrhs ≫ blocking size)

All done in a *single* computation

Outline

1. Augmented partial factorization (APF) method

2. Applications of APF:

   a) Two-photon coherent backscattering

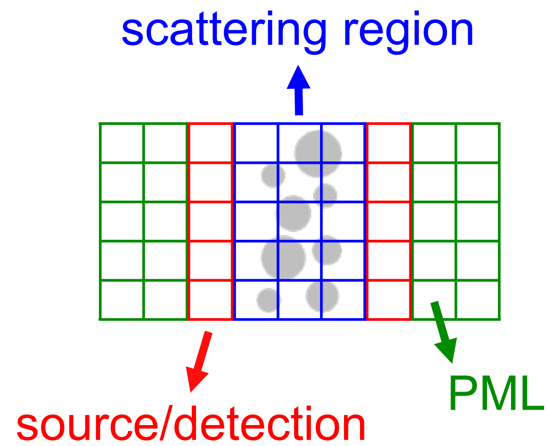
   b) Vectorial open channel in 3D

   c) Noninvasive imaging deep inside scattering media

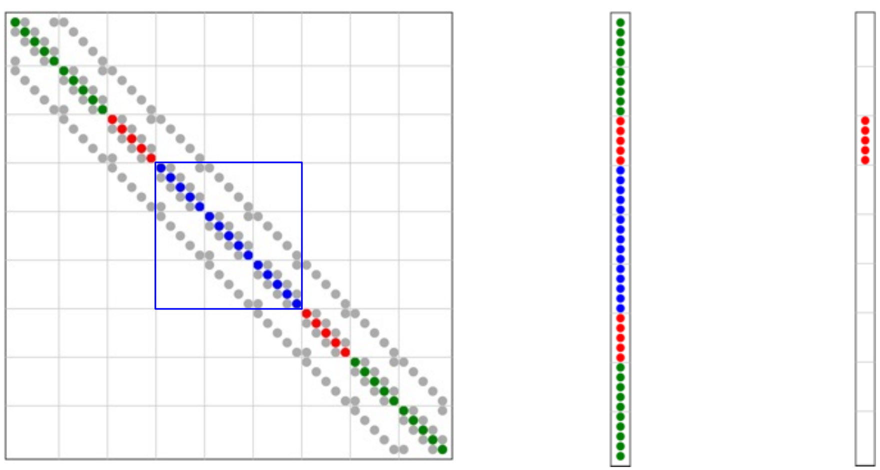
   d) Inverse design of metasurfaces

Frequency-domain response problem

Maxwell's equations in frequency domain:



$$\underbrace{\left[-\frac{\omega^2}{c^2}\varepsilon_{\mathrm{r}}(\omega,\mathbf{r}) + \nabla\times\nabla\times\right]}_{\mathbf{A}\ \text{(sparse matrix)}}\underbrace{\mathbf{E}(\mathbf{r})}_{x\ \text{(vector)}} = \underbrace{i\omega\mu_0\mathbf{J}(\mathbf{r})}_{b\ \text{(vector)}}$$



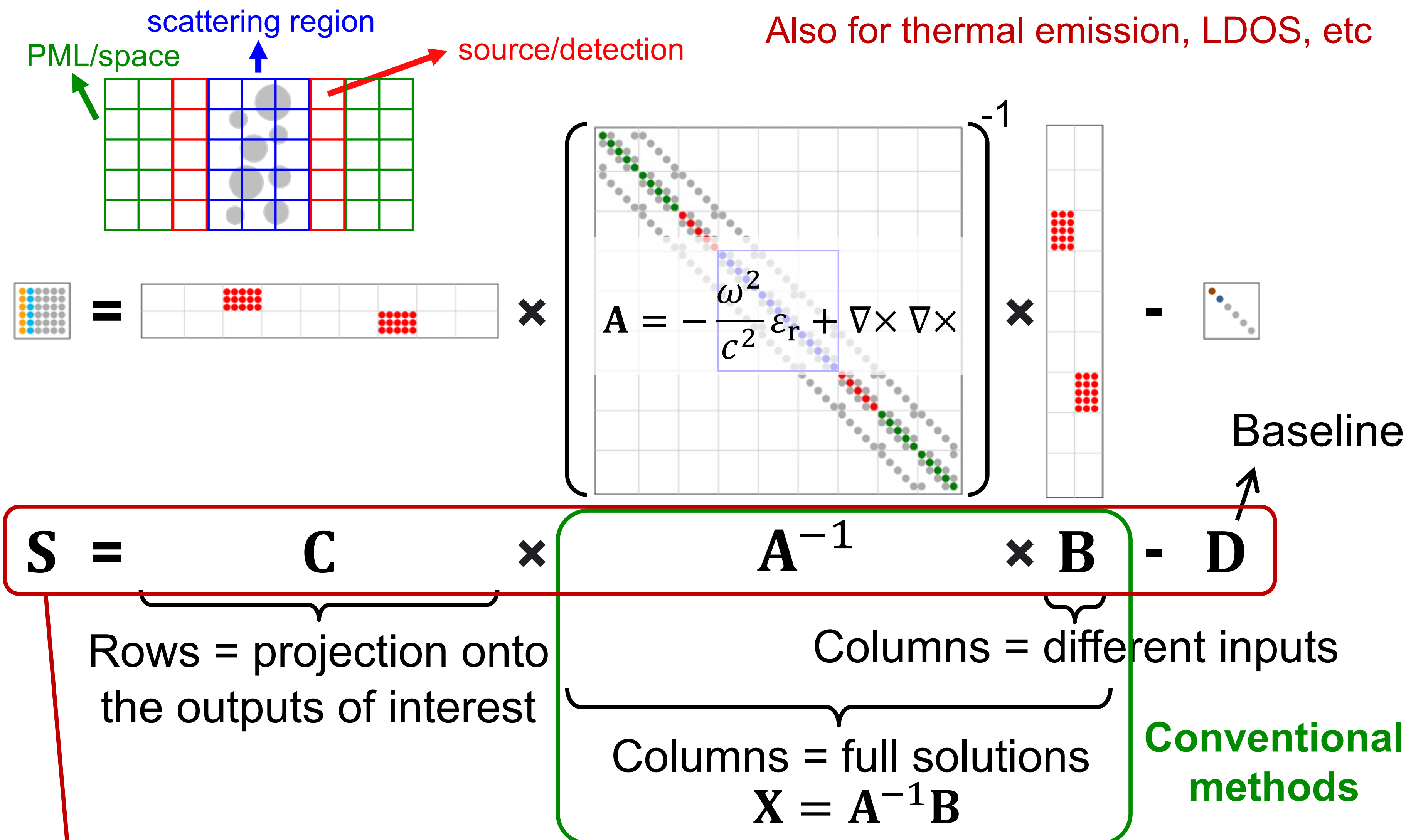
Conventional approach:

1. Factorize $\mathbf{A} = \mathbf{LU}$
2. Solve for field profile everywhere, $x = \mathbf{A}^{-1}b$.
3. Project solution x onto the outputs of interest.

Repeat for each block of inputs $\{b\}$

Sparse system of linear equations: $\mathbf{A}x = b$

Formulate the generalized scattering matrix



What we want, for *any linear-response problem*

Schur complement

Want an efficient way to evaluate $\mathbf{S} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D}$

projection of $\mathbf{A}^{-1}$ onto $\mathbf{C}$ and $\mathbf{B}$

Recall a simple problem:

$$ca^{-1}(a\,x_1 + b\,x_2 = y_1) \cdots\cdots \quad (1)$$
$$c\,x_1 + d\,x_2 = y_2 \quad \cdots\cdots \quad (2)$$

Eliminate x_1. Then solve for x_2

$$c\,x_1 + ca^{-1}b\,x_2 = ca^{-1}y_1 \quad \cdots\cdots \quad ca^{-1}(1)$$
$$-)\quad c\,x_1 + \quad d \quad x_2 = y_2 \quad \cdots\cdots \quad (2)$$
$$\boxed{(ca^{-1}b - d)}\,x_2 = ca^{-1}y_1 - y_2 \cdots\cdots \quad ca^{-1}(1) - (2)$$

Schur complement

Same procedure when we have matrices instead of scalars

Gaussian elimination ⇔ Eliminate unknowns by projecting them away

Here, want project the $\mathbf{A}^{-1}$ associated with x_1 & y_1 onto x_2 & y_2

⇒ **Augment** the system, then perform a **partial solution**

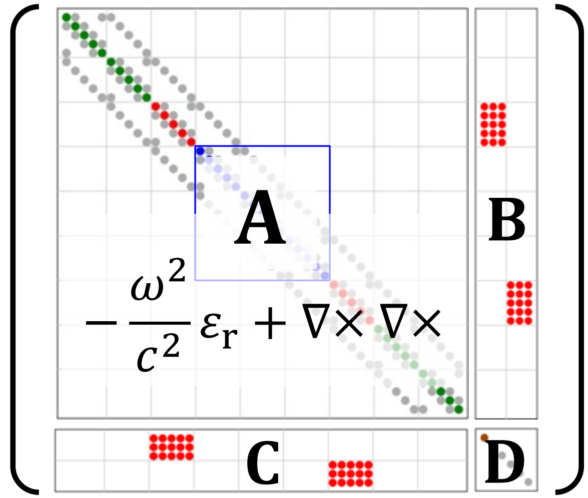
Augmented partial factorization (APF)

Want an efficient way to evaluate $\mathbf{S} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D}$

Step 1:

Build an *augmented matrix*

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} =$$



$\mathbf{B}$ = input source profiles

$\mathbf{C}$ = output projection profiles

Step 2: Use MUMPS to compute its Schur complement (through a partial LU factorization)

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{E} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{F} \\ \mathbf{0} & \mathbf{H} \end{bmatrix}$$

Schur complement

Step 3: Return $-\mathbf{H} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D} = \mathbf{S}$

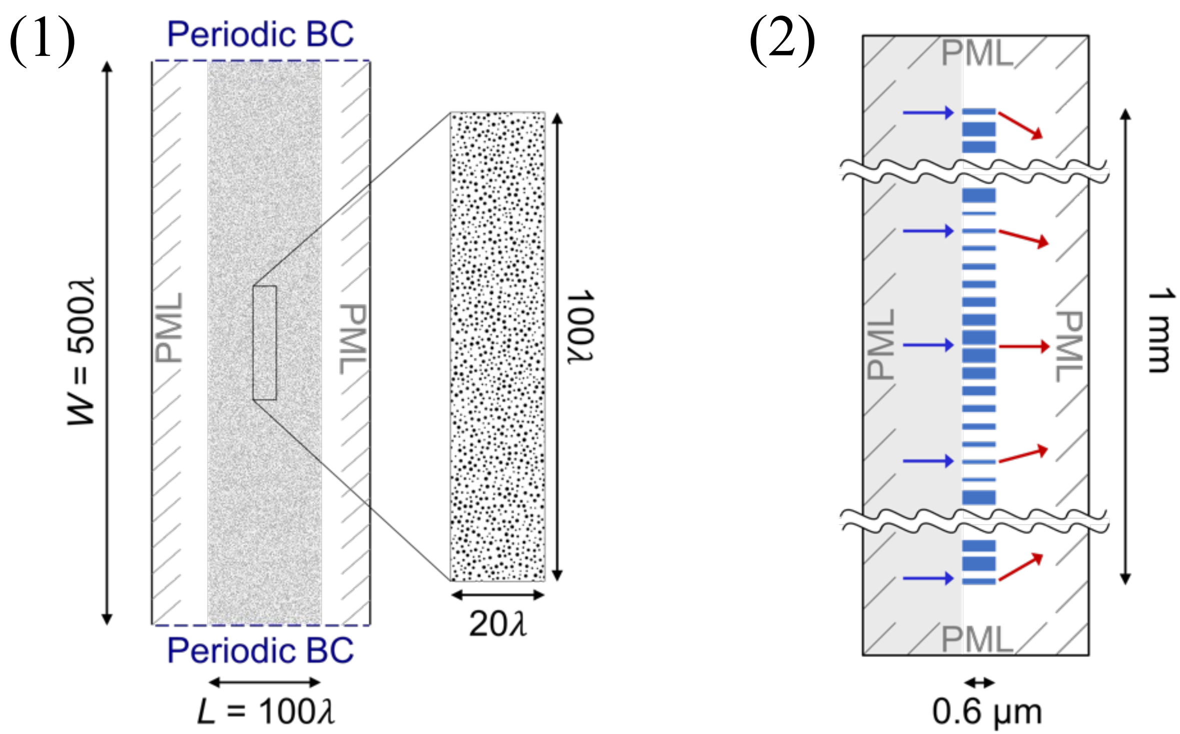
Solves for all inputs using a single factorization!

Augmented partial factorization (APF) advantages

- ✓ Full-wave solution; no approximation beyond discretization
- ✓ **Does not compute unnecessary solution, *i.e.* $\mathbf{X} = \mathbf{A}^{-1}\mathbf{B}$**
- ✓ **A *single* partial factorization solves M scattering problems with different inputs**
- ✓ Can use MUMPS ⇒ Optimized & scalable for parallel computing
- ✓ Does not need $\mathbf{L}$ and $\mathbf{U}$ factors [ICNTL(31)=1] ⇒ saves memory (up to a factor of 4)
- ✓ Uses all sparsity properties of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$
- ✓ Applicable to **any linear system**:
  - ✓ Any structure $\varepsilon_{\mathrm{r}}(\omega, \mathbf{r})$ including substrate *etc*; any dispersion
  - ✓ Any input sources & any output projections
  - ✓ Any linear PDE & any discretization scheme (finite difference, finite element, boundary element, …)
  - ✓ Any linear problem of the form $\mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B}$

Benchmarks on large-scale multi-channel systems

Implemented APF with finite difference on Yee grid



2D TM waves

Uses **MUMPS** for partial factorization

On Intel Xeon Gold 6130 (using 1 core)

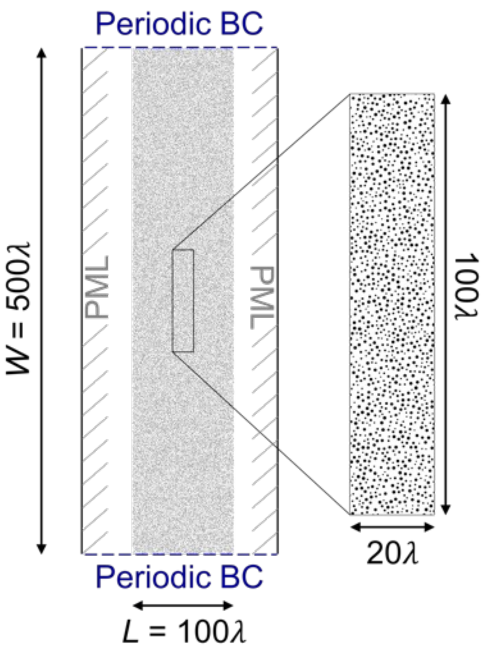
Finite Difference Frequency Domain

Compare: APF, Direct[1] & iterative[2] FDFD, RCWA[3], RGF[4]:

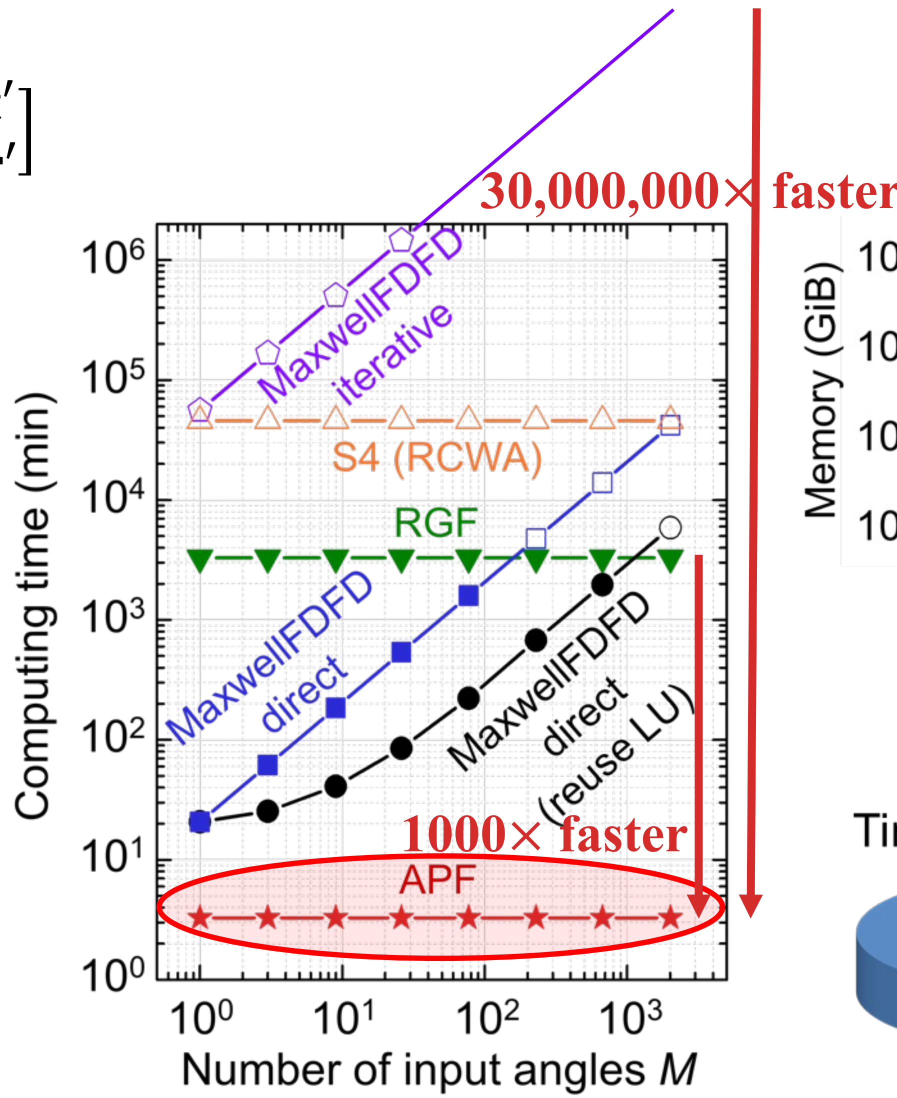
[1] MaxwellFDFD: https://github.com/wsshin/maxwellfdfd
[2] FD3D: https://github.com/wsshin/fd3d
[3] S4: https://github.com/victorliu/S4
[4] RGF: https://github.com/chiaweihsu/RGF

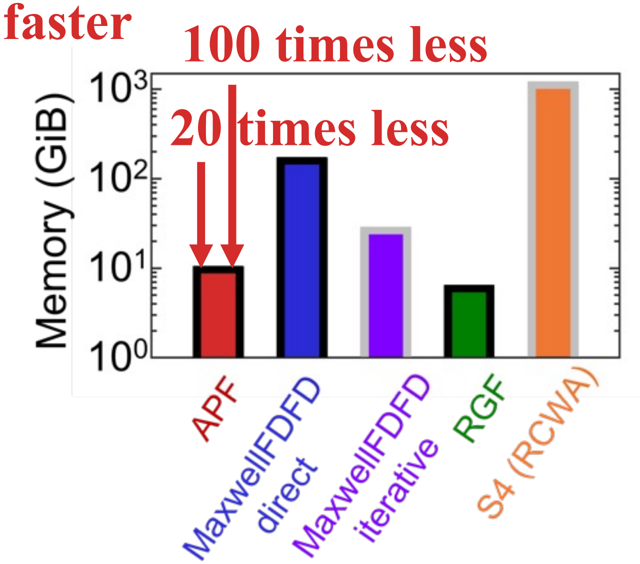
Benchmark 1: disordered media

Compute $\boldsymbol{S} = \begin{bmatrix} \mathbf{r} & \mathbf{t}' \\ \mathbf{t} & \mathbf{r}' \end{bmatrix}$

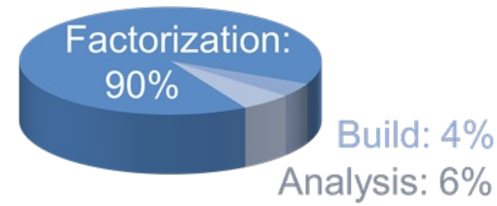


30,000 scatterers
12 million pixels
(resolution: $\Delta x = \lambda/15$)



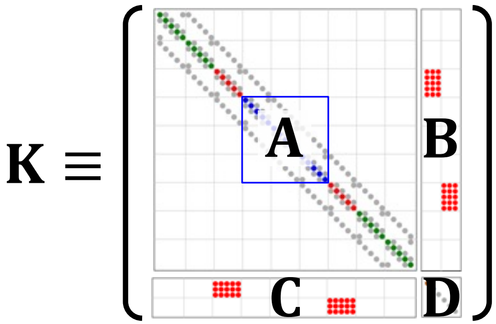


Time breakdown of APF



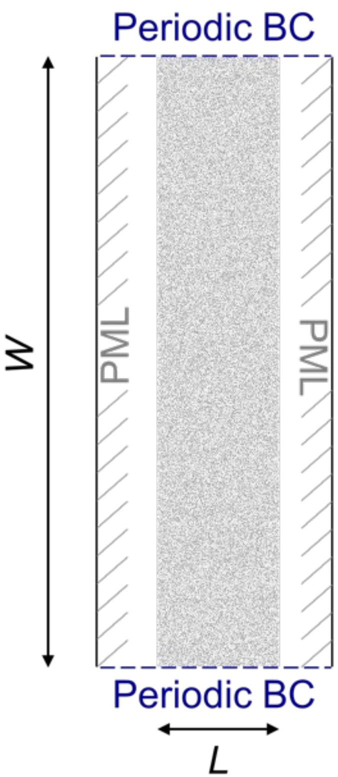
H-C Lin, Z Wang, CW Hsu, *Nature Computational Science* 2, 815 (2022)

Computing time & memory usage scaling in 2D

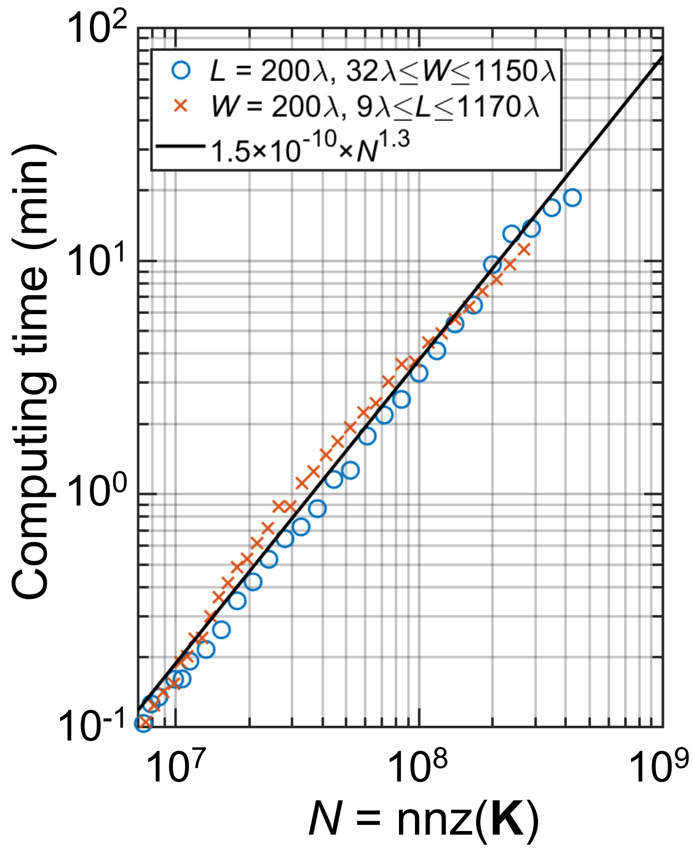


Timing & memory are governed primarily by the number of non-zero elements, $\mathrm{nnz}(\mathbf{K})$.

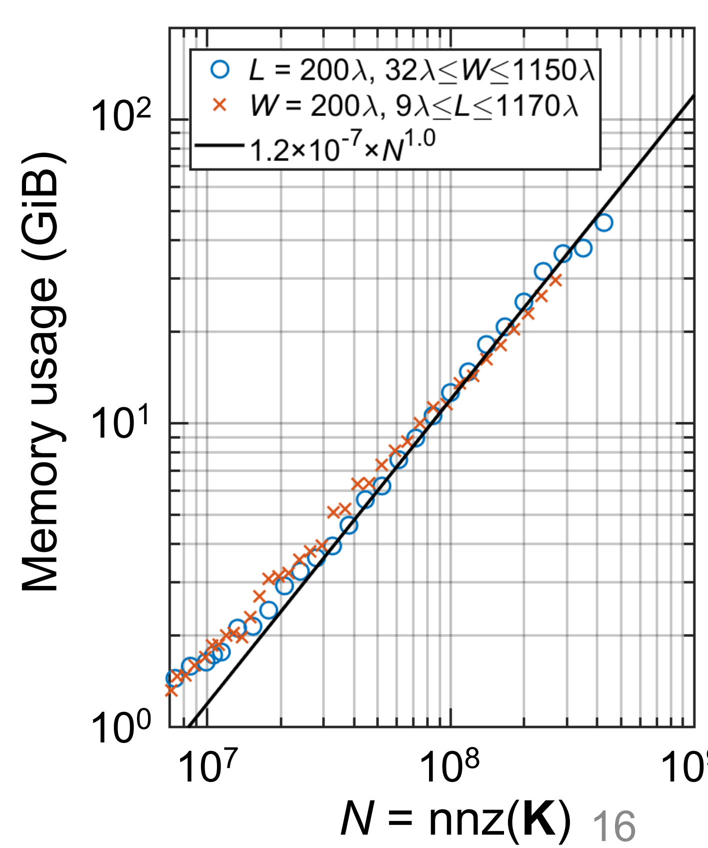
$\mathrm{nnz}(\mathbf{K}) \propto$ system size



Timing $\propto [\mathrm{nnz}(\mathbf{K})]^{1.3}$

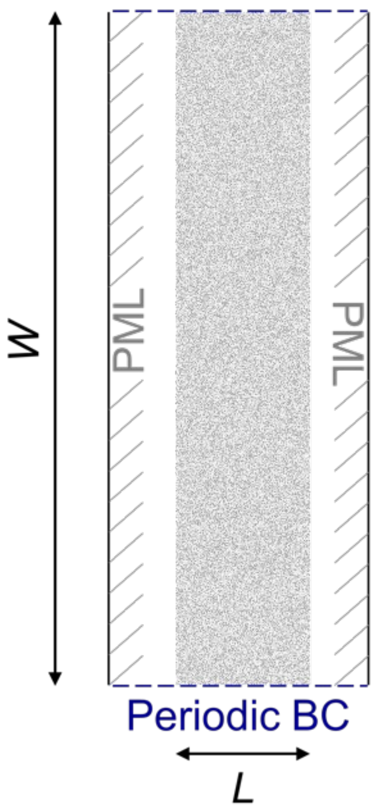


Memory $\propto \mathrm{nnz}(\mathbf{K})$

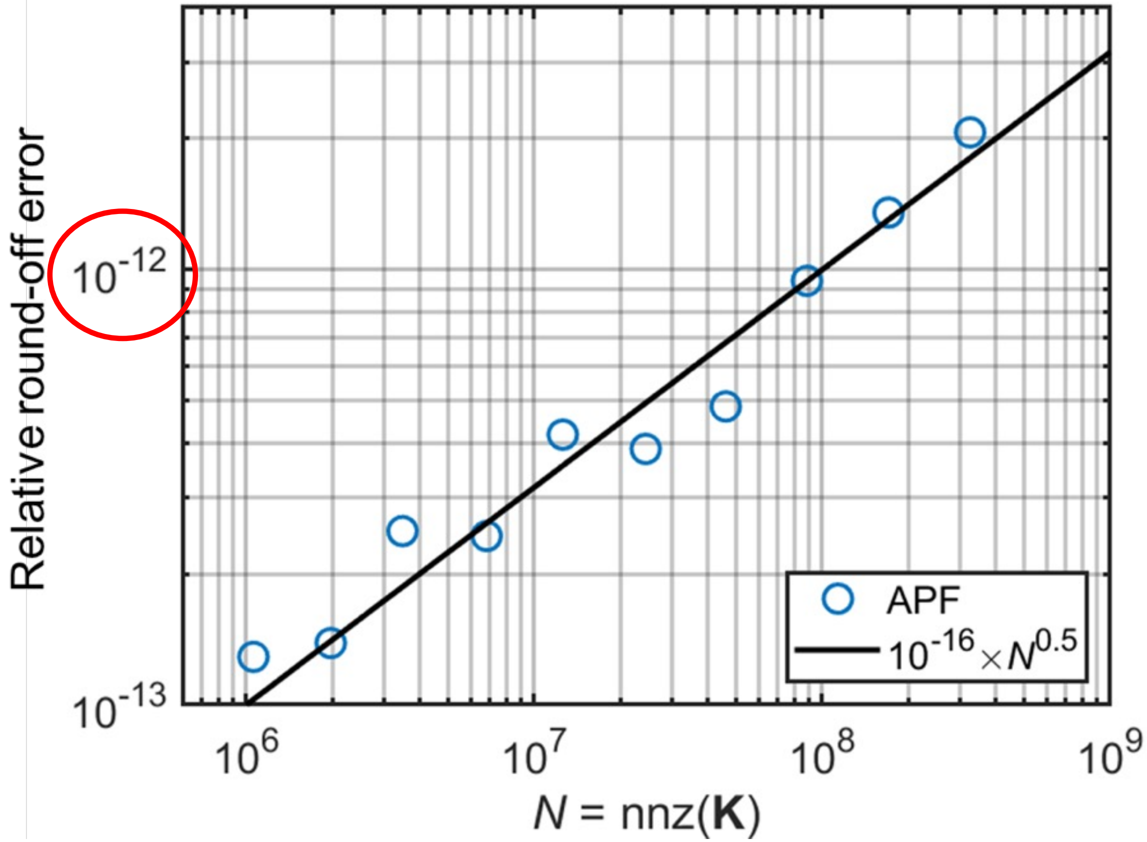


Numerical round-off error

(double-precision arithmetic)
(no iterative refinement)



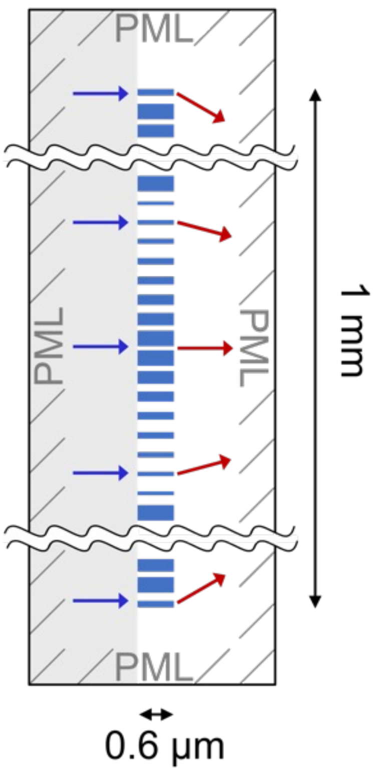
Error ~ $10^{-16}[\text{nnz}(\mathbf{K})]^{0.5}$

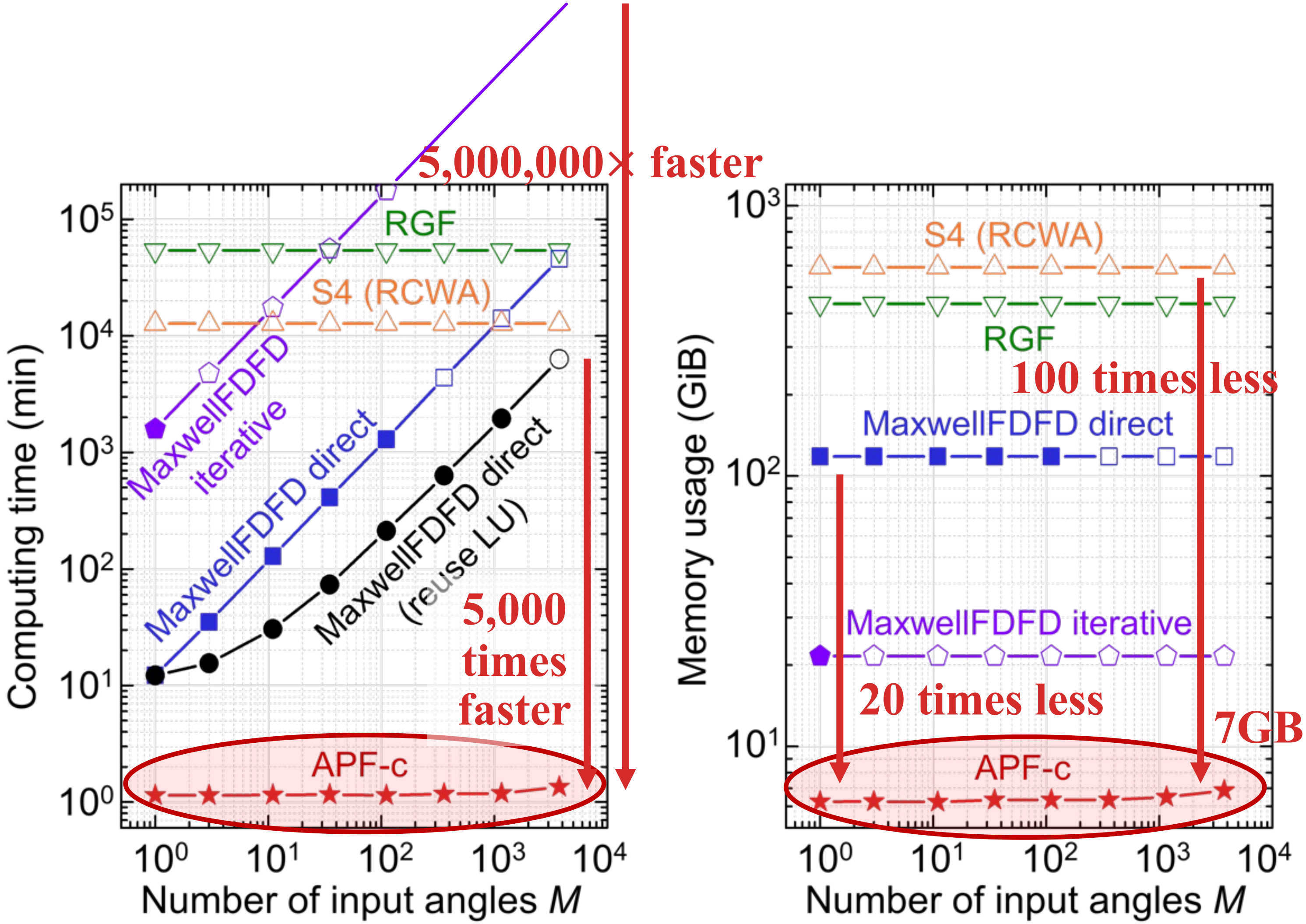


Only relevant error is from discretization

Benchmark 2: mm-scale TiO_2 metalens

Compute **t**



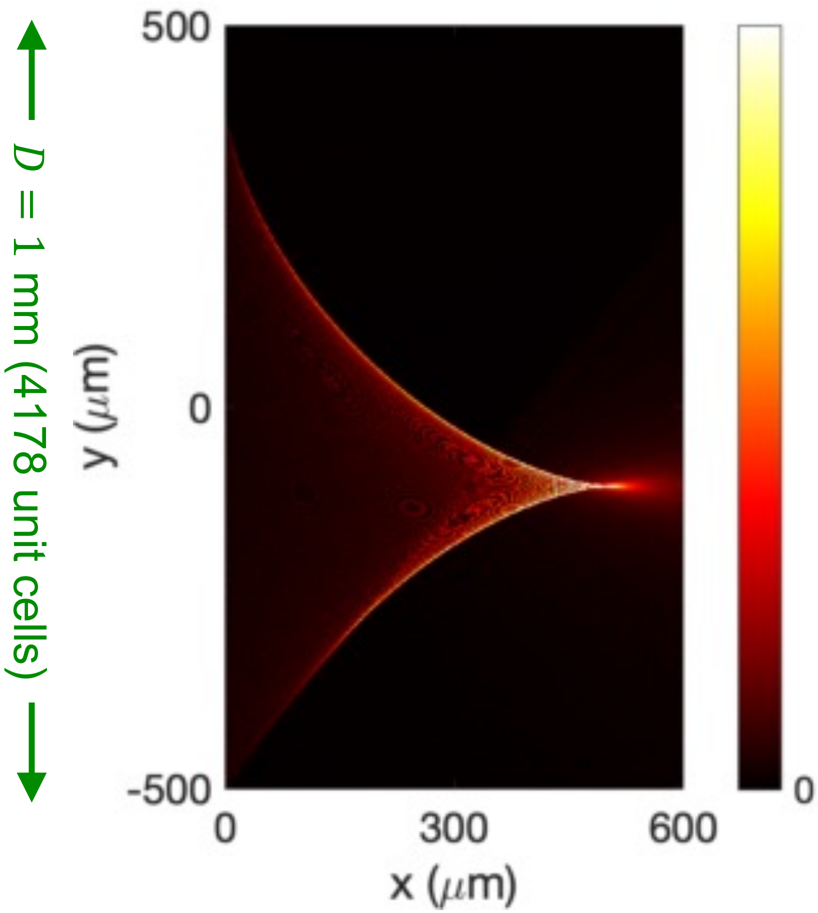


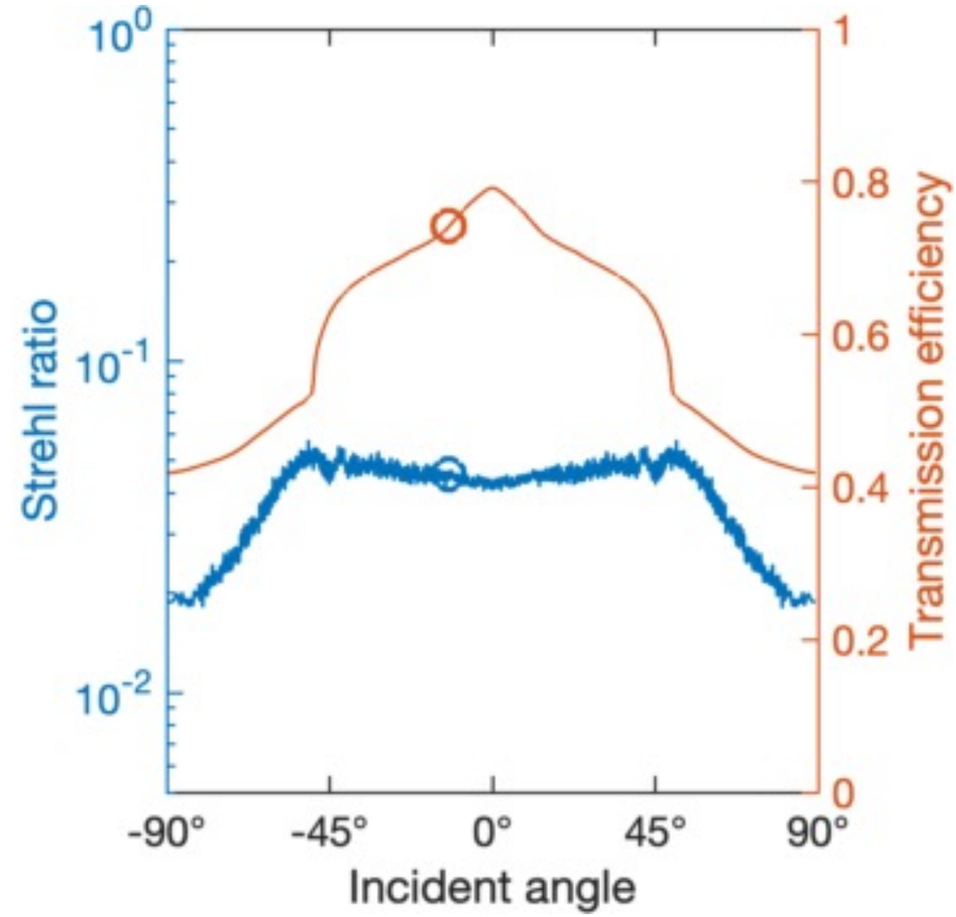
4,178 unit cells
11 million pixels
(resolution: $\Delta x = \lambda/40$; $\lambda = 532$ nm)

H-C Lin, Z Wang, CW Hsu, *Nature Computational Science* 2, 815 (2022)

Example: High-NA quadratic metalens

$$\phi(\rho) = -\frac{2\pi}{\lambda}\frac{\rho^2}{2f} \quad (D = 1000\ \mu\text{m},\ f = 500\ \mu\text{m},\ \text{NA} = 0.71)$$





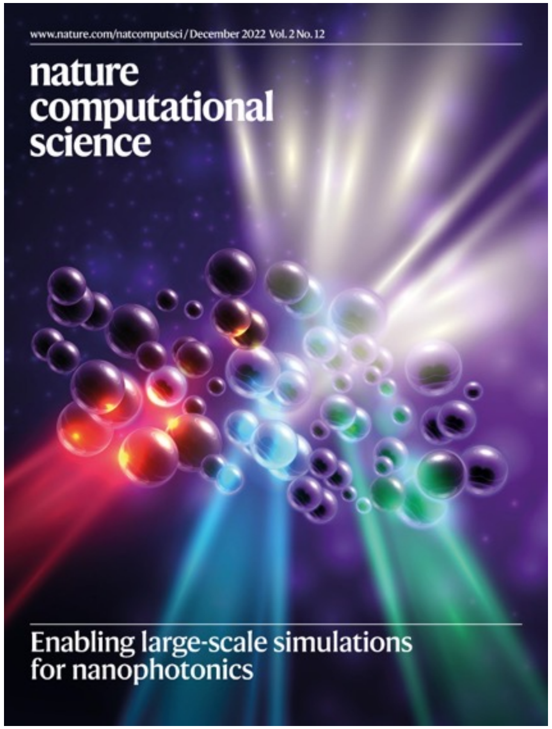
Full-wave simulation @ 3,761 incident angles
Total computing time ~ 1 minute using one core on a laptop

MESTI software

<u>M</u>axwell's <u>E</u>quations <u>S</u>olver with <u>T</u>housands of <u>I</u>nputs

https://github.com/complexphoton/MESTI.m

Uses sequential MUMPS (yes multithreading, no MPI)



- Open-source
- TE & TM polarizations in 2D
- Any $\varepsilon(x, y)$ including substrates *etc*
- Any dispersion
- Any list of input source profiles
- Any list of output projection profiles (or full solution)
- All common boundary conditions
- PML with real & imaginary coordinate stretching
- Utility functions for building inputs/outputs
- Documentation
- Examples

3D vectorial version in Julia using parallel MUMPS: coming soon!

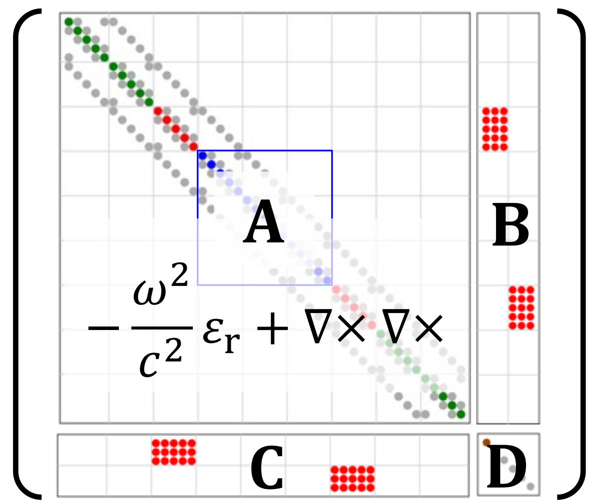
One caveat

Want to evaluate $\mathbf{S} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D}$

Step 1:

Build an *augmented matrix*

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} =$$



$\mathbf{B}$ = input source profiles

$\mathbf{C}$ = output projection profiles

Step 2: Compute its Schur complement

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{E} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{F} \\ \mathbf{0} & \mathbf{H} \end{bmatrix}$$

Schur complement

Step 3: Return $-\mathbf{H} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D} = \mathbf{S}$

What if **number of columns in $\mathbf{B}$ $\neq$ number of rows in $\mathbf{C}$?**

$\Rightarrow$ Pad zero-columns to $\mathbf{B}$ or zero-rows to $\mathbf{C}$

Very inefficient when the two numbers are very different (eg: gradient)

Wish list for MUMPS: skip Schur complement evaluation associated with zero rows/columns

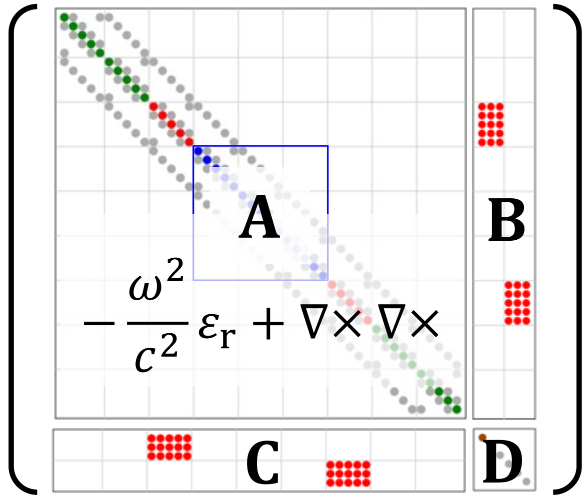
Schur complement without LU?

Want to evaluate $\mathbf{S} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D}$

Step 1:

Build an *augmented matrix*

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} =$$



$\mathbf{B}$ = input source profiles

$\mathbf{C}$ = output projection profiles

Step 2: Compute its Schur complement

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{E} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{F} \\ \mathbf{0} & \mathbf{H} \end{bmatrix}$$

Schur complement ($\mathbf{H}$)

Step 3: Return $-\mathbf{H} = \mathbf{C}\,\mathbf{A}^{-1}\,\mathbf{B} - \mathbf{D} = \mathbf{S}$

We only need the Schur complement; not the LU factors

⇒ Potential room for further acceleration?

Outline

1. Augmented partial factorization (APF) method

2. Applications of APF (all done with MESTI):

   a) Two-photon coherent backscattering

      with Yaron Bromberg @ Hebrew University
      & Arthur Goetschy @ Institut Langevin

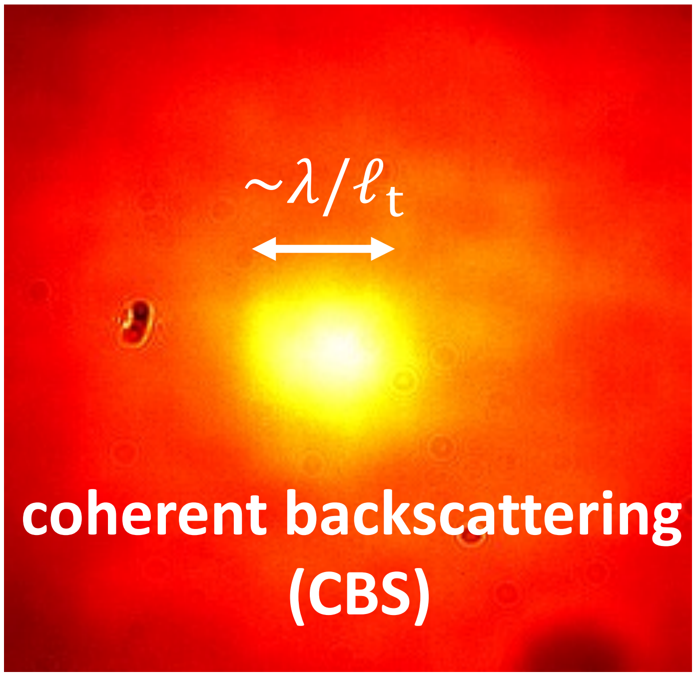
   b) Vectorial open channel in 3D

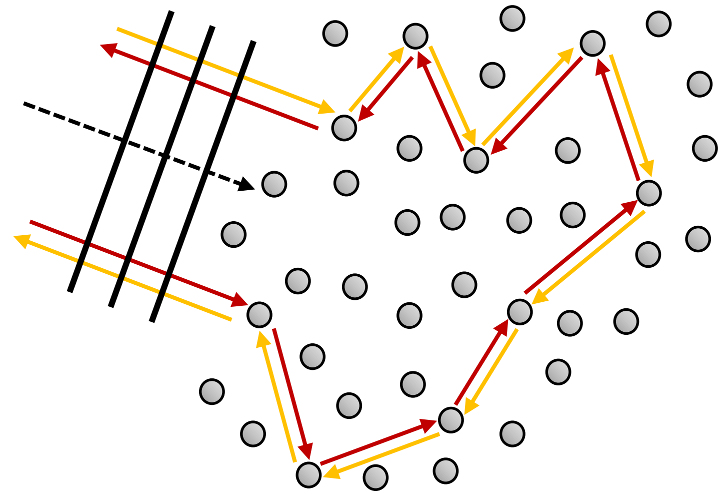
   c) Noninvasive imaging deep inside scattering media

   d) Inverse design of metasurfaces

Coherent backscattering (CBS) of classical light

Disorder averaged backscattered intensity



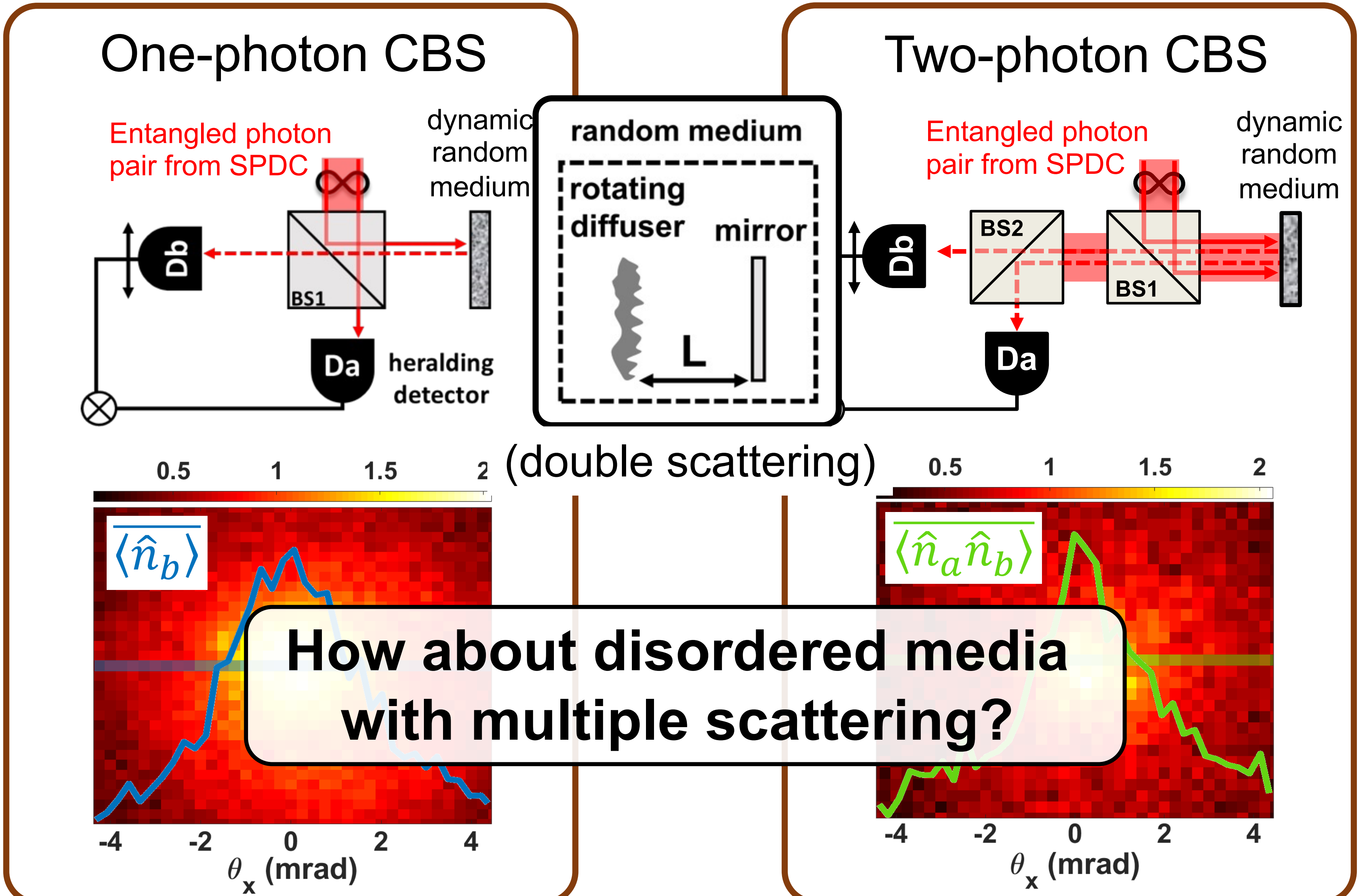


ℓ_t: transport mean free path

Akkermans, Wolf, Maynard, PRL (1986)

Does non-classical light exhibit CBS?
How does that differ from classical CBS?

Coherent backscattering of non-classical light



How about disordered media with multiple scattering?

M. Safadi, O. Lib, H.-C. Lin, CWH, A. Goetschy, and Y. Bromberg, *Nature Physics* (2023).

Theory of two-photon CBS

Entangled photon-pair input: $|\psi\rangle \propto \sum_{q'} \hat{c}^\dagger_{q'} \hat{c}^\dagger_{-q'} |0\rangle$

$\hat{c}^\dagger_q$: creation operator for input mode

Reflected output: $\hat{d}_q = \sum_{q'} r_{qq'} \hat{c}_{q'}$

$r_{qq'}$: reflection matrix

Coincidence rate $\propto \overline{\langle\psi|\!: \hat{n}_a \hat{n}_b :\!|\psi\rangle}$

$\hat{n}_q = \hat{d}^\dagger_q \hat{d}_q$

$\overline{}$ = disorder average

$\propto \overline{\left|(r^2)_{q_b,-q_a}\right|^2}$ (use reciprocity)

matrix square

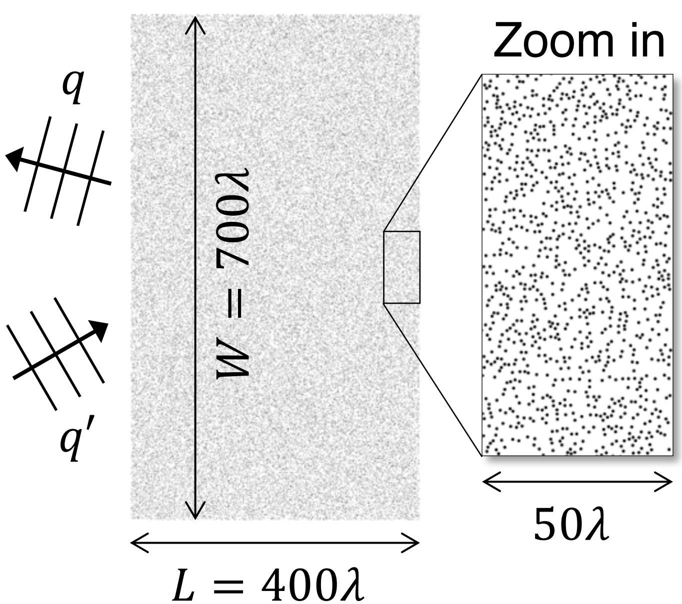
Need:

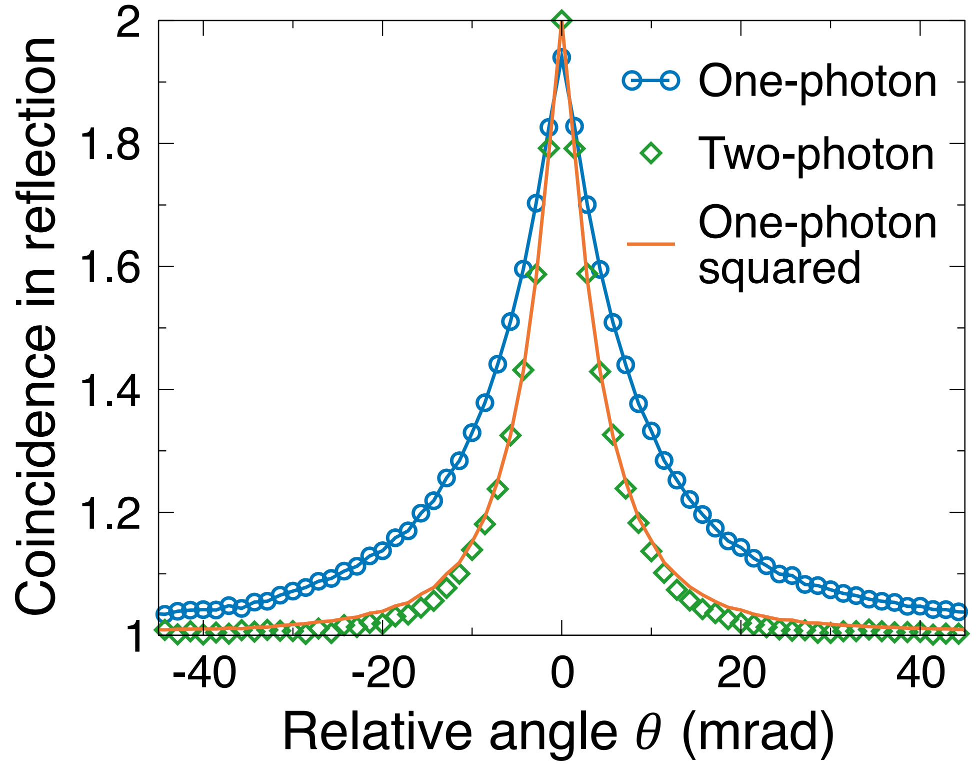
- Full reflection matrix r for all of the many input/output angles.
- Average over thousands of disorder realizations.
- System width $W \gtrsim 60\ell_t$ to resolve the two-photon CBS cone.
- System thickness $L \gg \ell_t$ to be in diffusive regime of transport.
- Need to suppress single scattering in reflection ⇒ ~~point scatterers~~
- Full-wave solution.

Very challenging for existing numerical methods…
But not with APF.

Two-photon CBS in disordered media

56,000 scatterers
28 million pixels
(resolution: $\Delta x = \lambda/10$)
1,400 angles per side





Compute 4,000 reflection matrices from 2,000 realizations
One realization takes 11 minutes using one core, using APF

M. Safadi, O. Lib, H.-C. Lin, CWH, A. Goetschy, and Y. Bromberg, *Nature Physics* (2023).

Outline

1. Augmented partial factorization (APF) method
2. Applications of APF:
   a) Two-photon coherent backscattering
   b) Vectorial open channel in 3D
   c) Noninvasive imaging deep inside scattering media
   d) Inverse design of metasurfaces

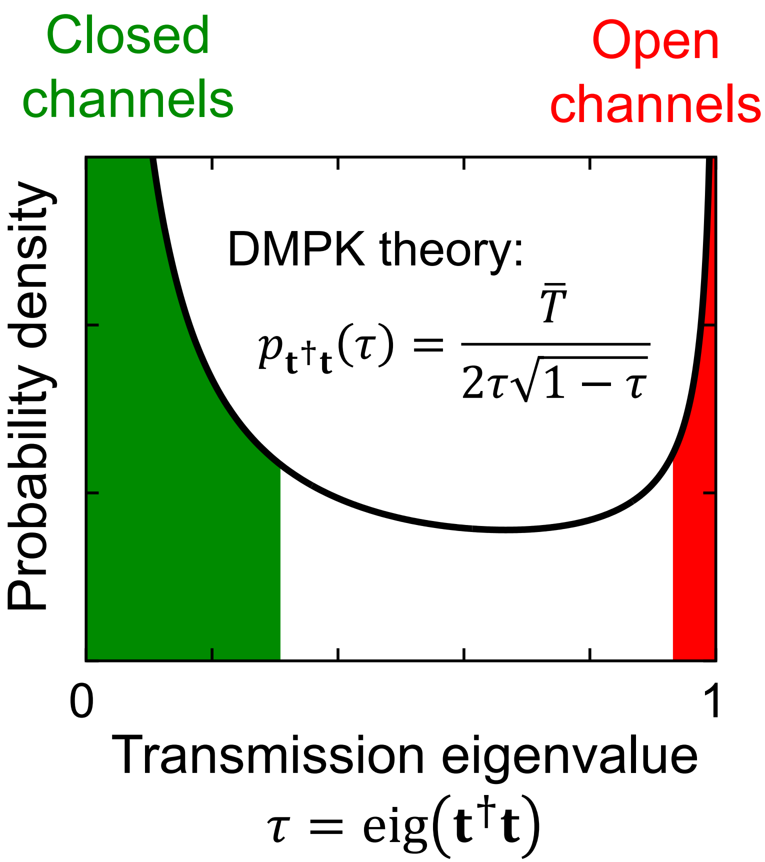
Open channels through disorder

First predicted for scalar electron waves:

Dorokhov, *Solid State Commum* (1984)

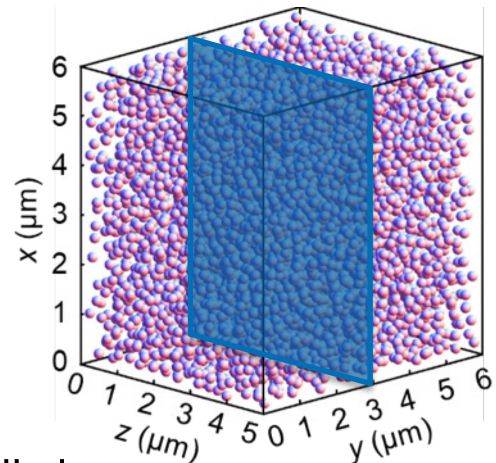
Mello, Pereyra, Kumar, *Ann Phys* (1988)

$$\mathbf{S} = \begin{bmatrix} \mathbf{r} & \mathbf{t}' \\ \mathbf{t} & \mathbf{r}' \end{bmatrix}$$ for two-sided systems



- **Shaping the wavefront of electrons is hard.**
- **Realized for scalar waves in 2D waveguides:**
  - FDTD simulation: Choi et al, *PRB* (2011)
  - Acoustic exp: Gérardin et al, *PRL* (2014)
  - Optical exp: Sarma et al, *PRL* (2016)
  - Microwave exp: Horodynski et al, *Nature* (2022)
- **Realization in 3D remains challenging:**
  - Experiments face incomplete channel control
    - Yu et al, *PRL* (2013): 7% ⇒ 65%
    - Popoff et al, *PRL* (2014): 5% ⇒ 18%
    - Bosch, PhD thesis (2020): 26% => 49%
  - Simulations take unrealistic resources (but not with APF!)

Open channel for 3D vectorial EM waves



6,000 scatterers; 2 million voxels
$\lambda = 532$ nm; $\Delta x = \lambda/10$

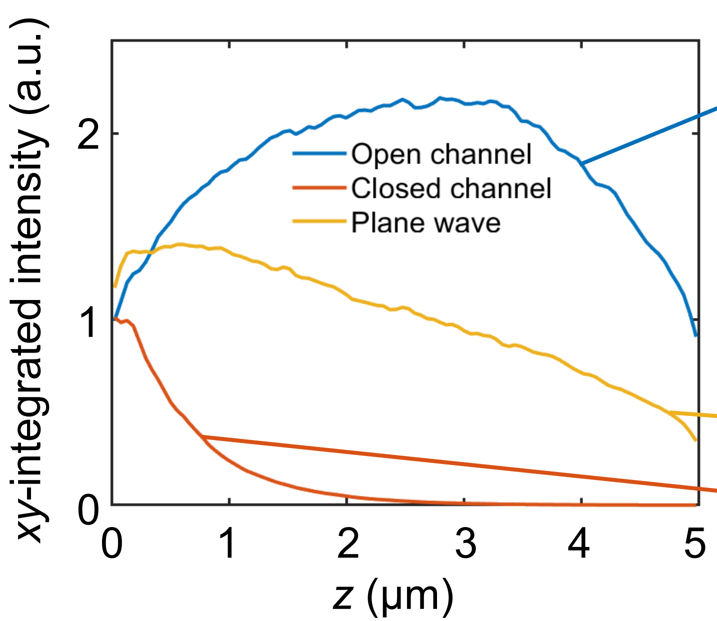
$\bar{T} = 0.12;\ g \approx 100$
826 angles per side (both polarizations)

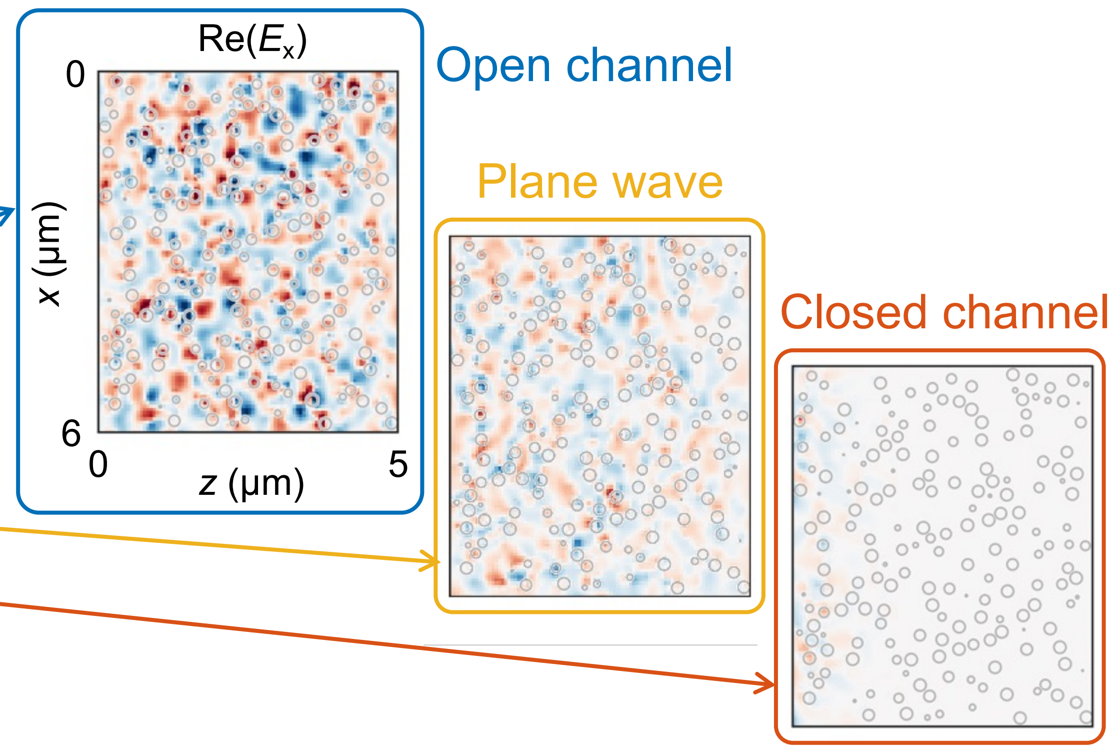
APF computation takes <u>25 minutes</u> (using one 16-core Intel Xeon Gold 6130)

Memory: 87 GB (would be 300 GB if LU is stored)

Periodic in x, y
PML in z

$n_{\text{sphere}} = 2.54$ (TiO_2)
diameter $= 0.2$ μm

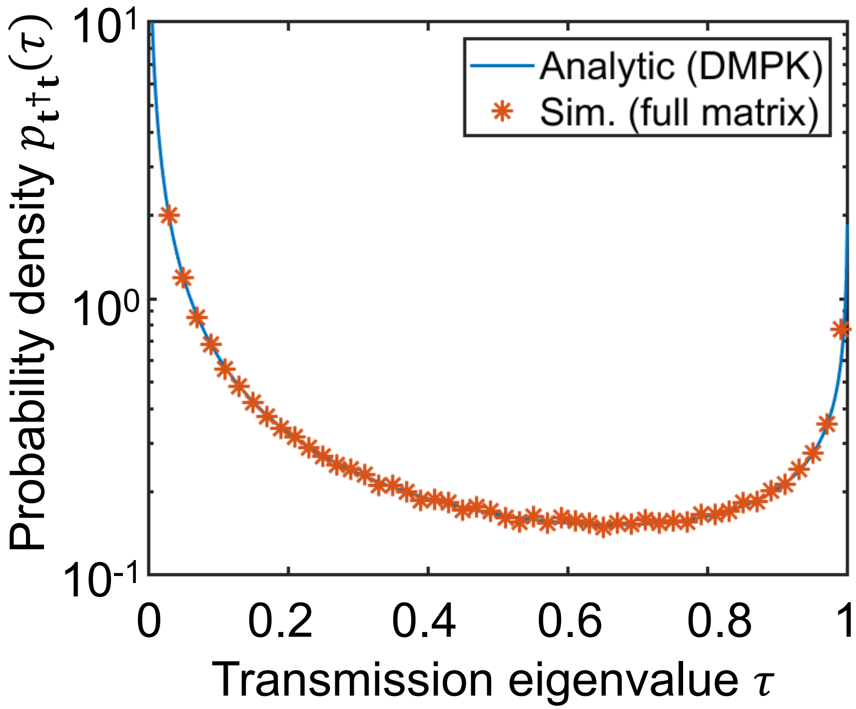




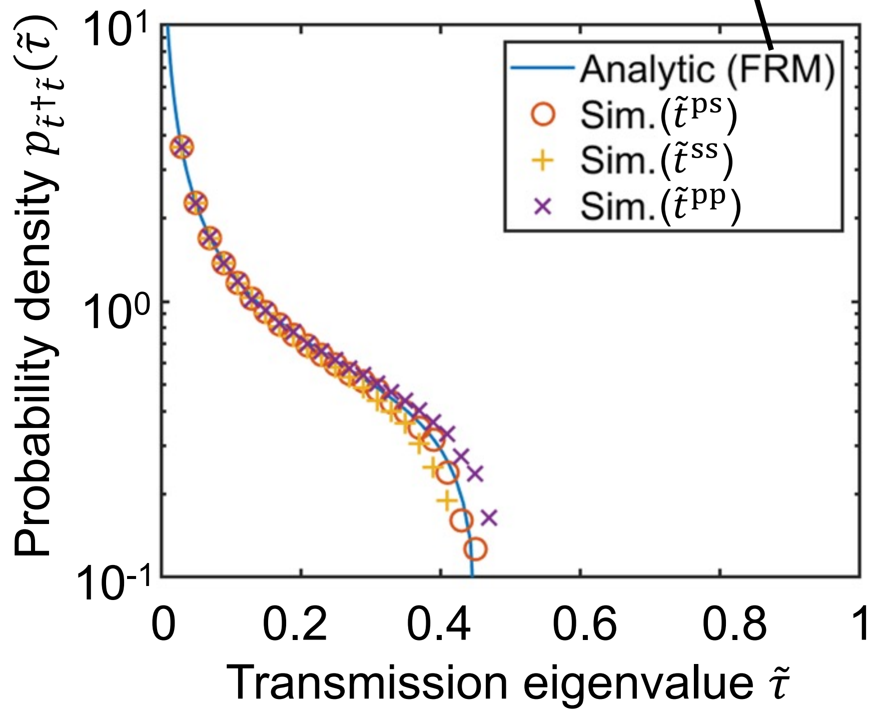
Eigenvalue distribution for 3D vectorial EM waves

Filtered random matrix (FRM) theory
A. Goetschy & A. D. Stone, PRL (2013)

Full transmission matrix



One-quarter of *t*

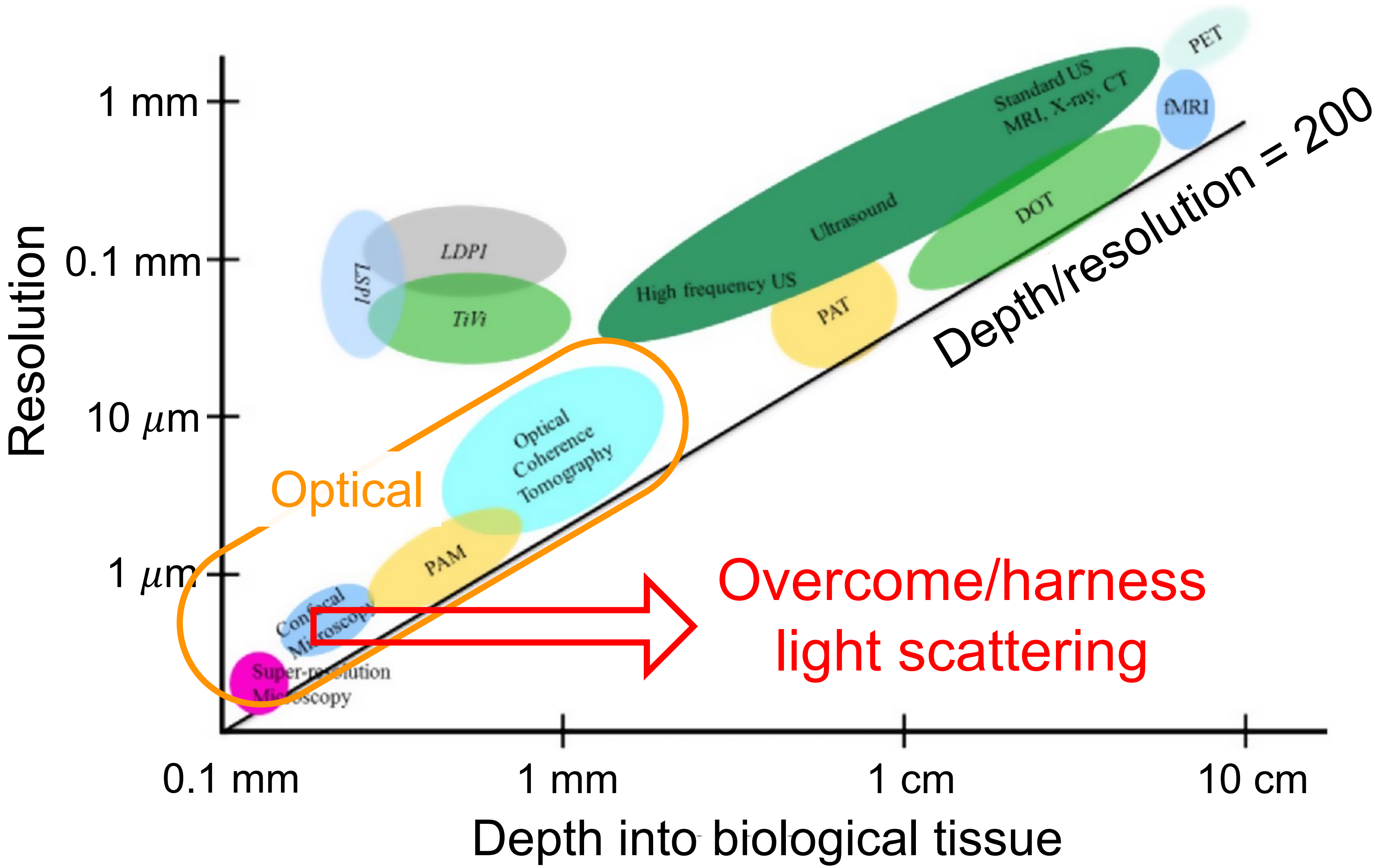


Ensemble average over 500 realizations

Outline

1. Augmented partial factorization (APF) method
2. Applications of APF:
   a) Two-photon coherent backscattering
   b) Vectorial open channel in 3D
   c) Noninvasive imaging deep inside scattering media
   d) Inverse design of metasurfaces

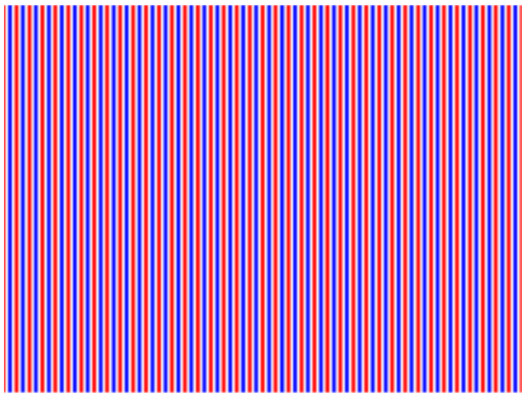
Depth-vs-resolution trade-off for deep imaging



Lai & Leahy, MICC 23, 345 (2016)

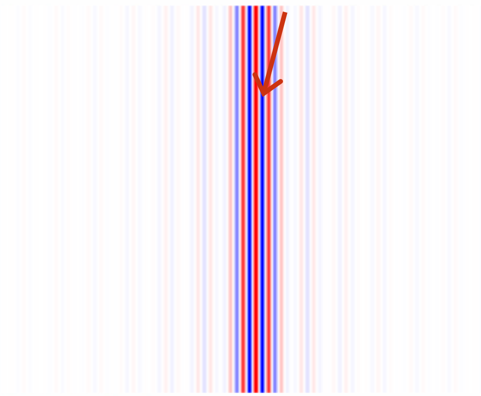
Spatiotemporal gating ⇔ summing plane waves

$$e^{i\mathbf{k}_{\text{in}}\cdot(\mathbf{r}-\mathbf{r}_0)-i\omega t}$$



$$\sum_{\omega} e^{i\mathbf{k}_{\text{in}}\cdot(\mathbf{r}-\mathbf{r}_0)-i\omega t}$$

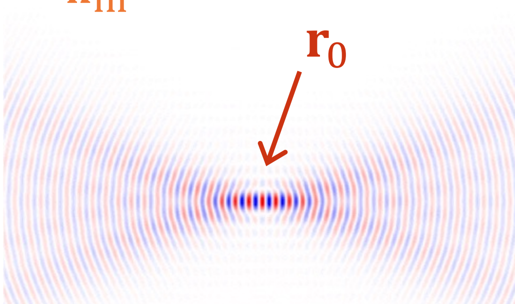
arrive at $\mathbf{r}_0$ at time $t = 0$



time gating

$$\sum_{\mathbf{k}_{\text{in}}} e^{i\mathbf{k}_{\text{in}}\cdot(\mathbf{r}-\mathbf{r}_0)-i\omega t}$$

$\mathbf{r}_0$



input spatial gating

$$\sum_{\omega}\sum_{\mathbf{k}_{\text{in}}} e^{i\mathbf{k}_{\text{in}}\cdot(\mathbf{r}-\mathbf{r}_0)-i\omega t}$$

arrive at $\mathbf{r}_0$ at time $t = 0$

spatio-temporal gating

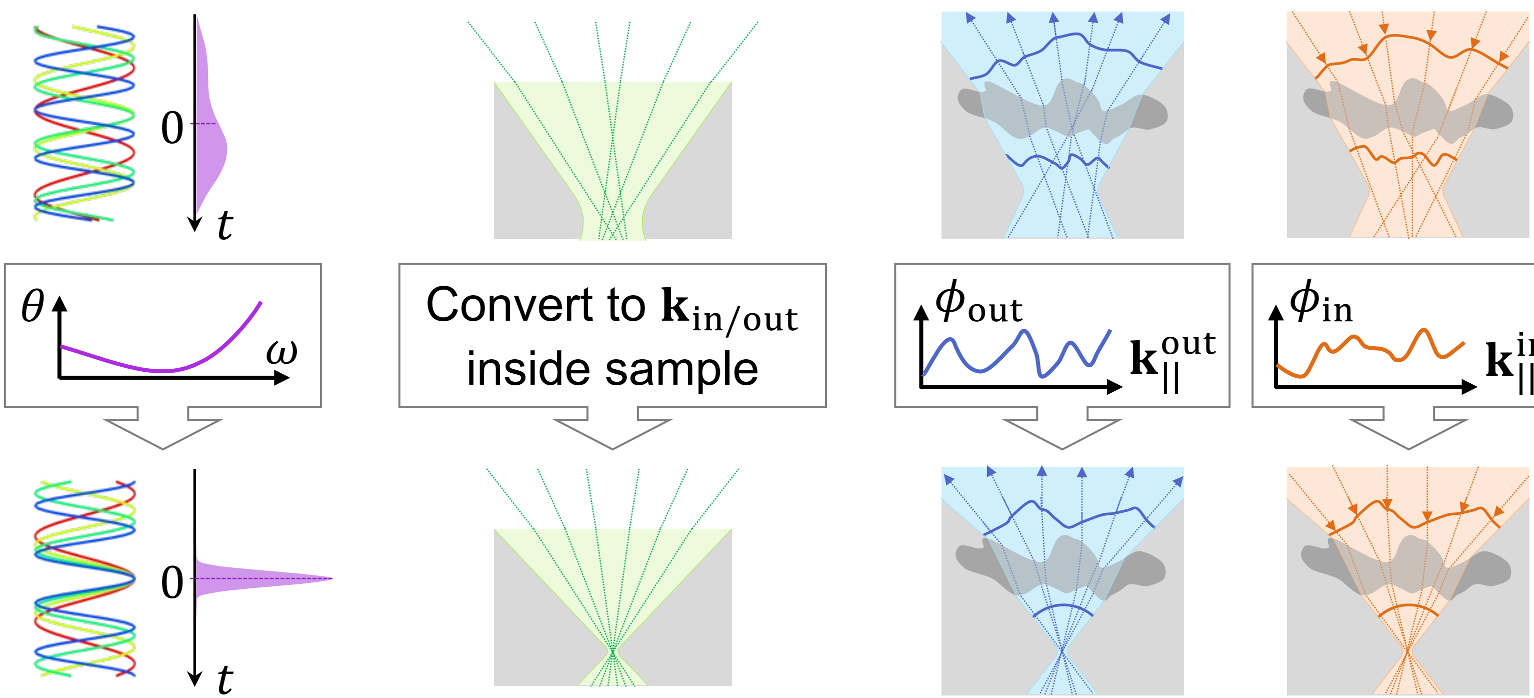
Scattering matrix tomography (SMT)

Time gating | Output spatial gating | Input spatial gating

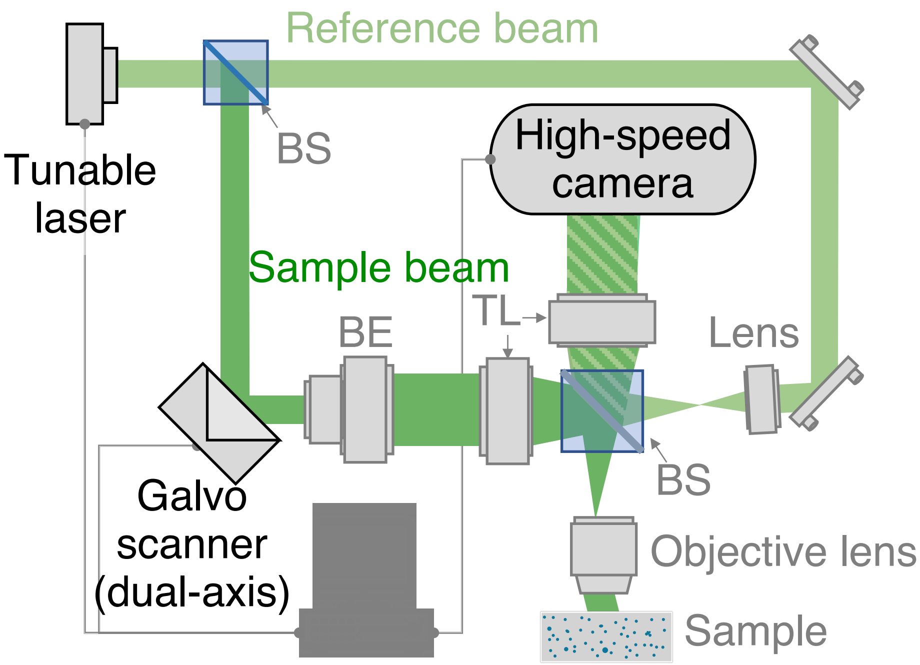
$$I_{\text{SMT}}(\mathbf{r}) = \left|\sum_{\omega} e^{i\theta(\omega)} \sum_{\mathbf{k}_{\text{out}}} e^{i\mathbf{k}_{\text{out}}\cdot\mathbf{r}+i\phi_{\text{out}}(\mathbf{k}_{\text{out}})} \sum_{\mathbf{k}_{\text{in}}} e^{-i\mathbf{k}_{\text{in}}\cdot\mathbf{r}+i\phi_{\text{in}}(\mathbf{k}_{\text{in}})} S(\mathbf{k}_{\text{out}},\mathbf{k}_{\text{in}},\omega)\right|^2$$

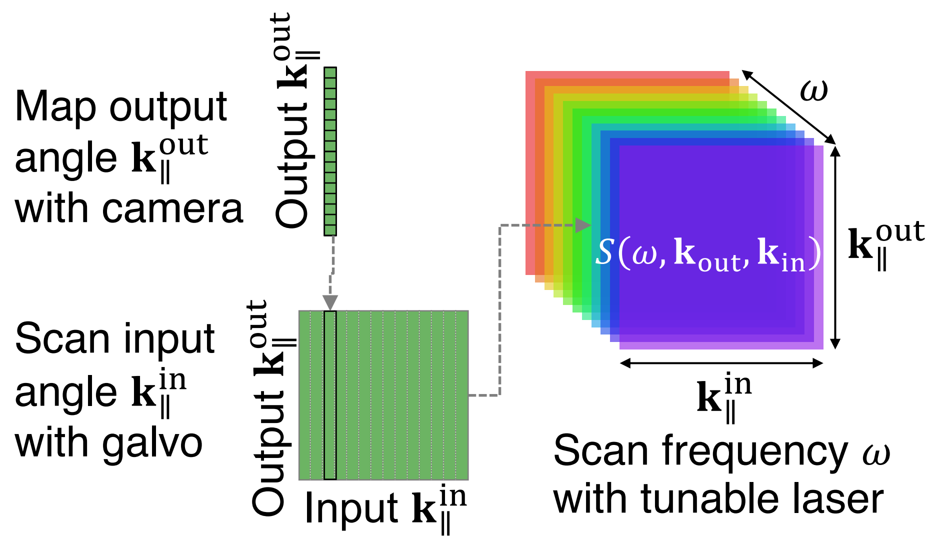
Scan $\mathbf{r}$ digitally ⇒ noninvasive image of the scattering amplitude

Digital pulse compression | Index mismatch correction | Wavefront corrections for output & input, syst & sample



Hyper-spectral reflection matrix measurement



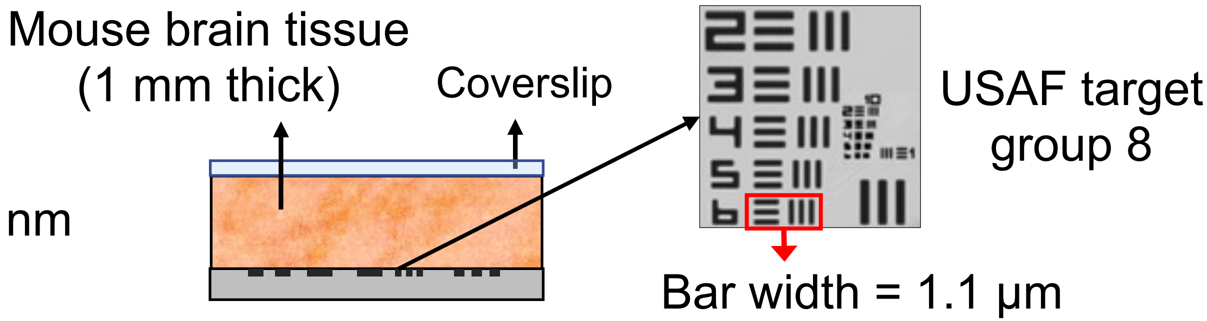


~200 frequencies ω (720-950 nm)
~4000 output angles $\mathbf{k}_{\mathrm{out}}$
~3000 input angles $\mathbf{k}_{\mathrm{in}}$

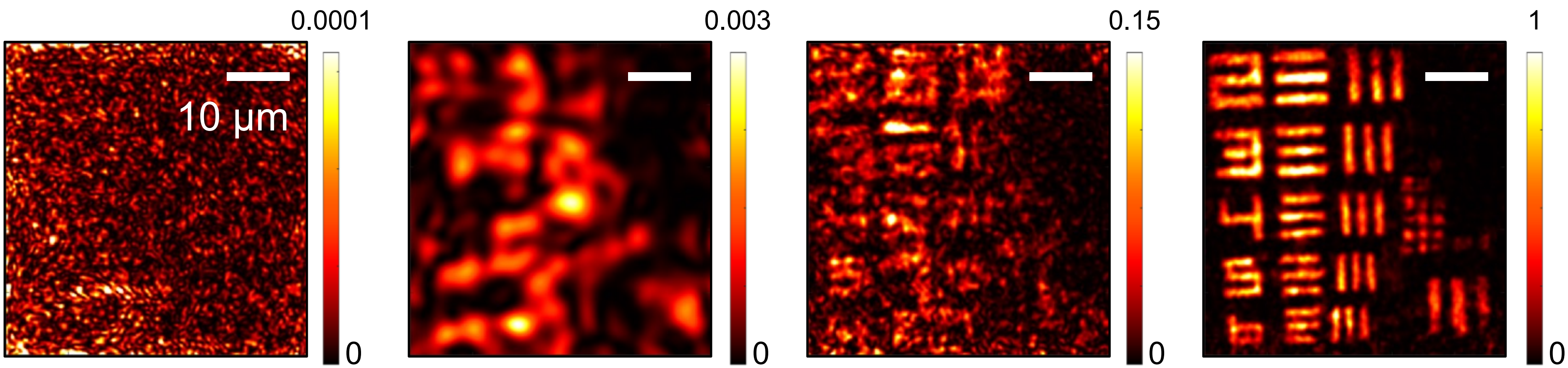
$$I_{\mathrm{SMT}}(\mathbf{r}) = \left| \sum_{\omega} e^{i\,\theta(\omega)} \sum_{\mathbf{k}_{\mathrm{out}}} e^{i\mathbf{k}_{\mathrm{out}}\cdot\mathbf{r} + i\,\phi_{\mathrm{out}}(\mathbf{k}_{\mathrm{out}})} \sum_{\mathbf{k}_{\mathrm{in}}} e^{-i\mathbf{k}_{\mathrm{in}}\cdot\mathbf{r} + i\,\phi_{\mathrm{in}}(\mathbf{k}_{\mathrm{in}})}\, S(\mathbf{k}_{\mathrm{out}}, \mathbf{k}_{\mathrm{in}}, \omega) \right|^2$$

Y. Zhang *et al*, arXiv:2306.08793

Imaging through brain tissue



λ = 720-950 nm
NA = 0.5



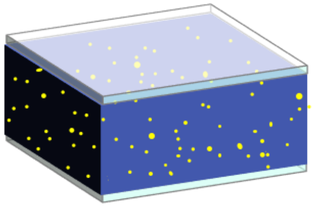
Reflectance confocal microscopy (**RCM**) | Optical coherence tomography (**OCT**) | Optical coherence microscopy (**OCM**) | **SMT**

Correct for dispersion (system + sample) and input aberration of the optical system

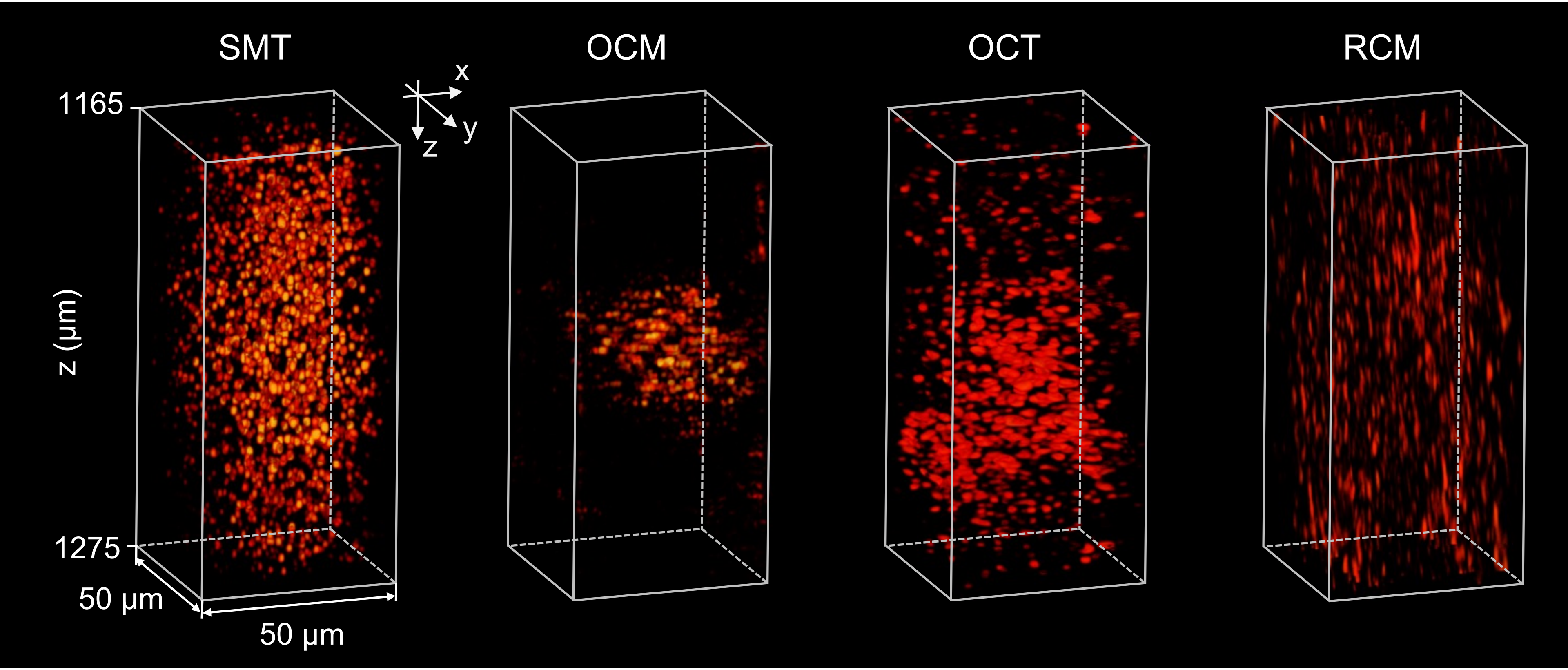
Also corrects for index mismatch & scattering from the sample

Y. Zhang *et al*, arXiv:2306.08793

Volumetric SMT imaging



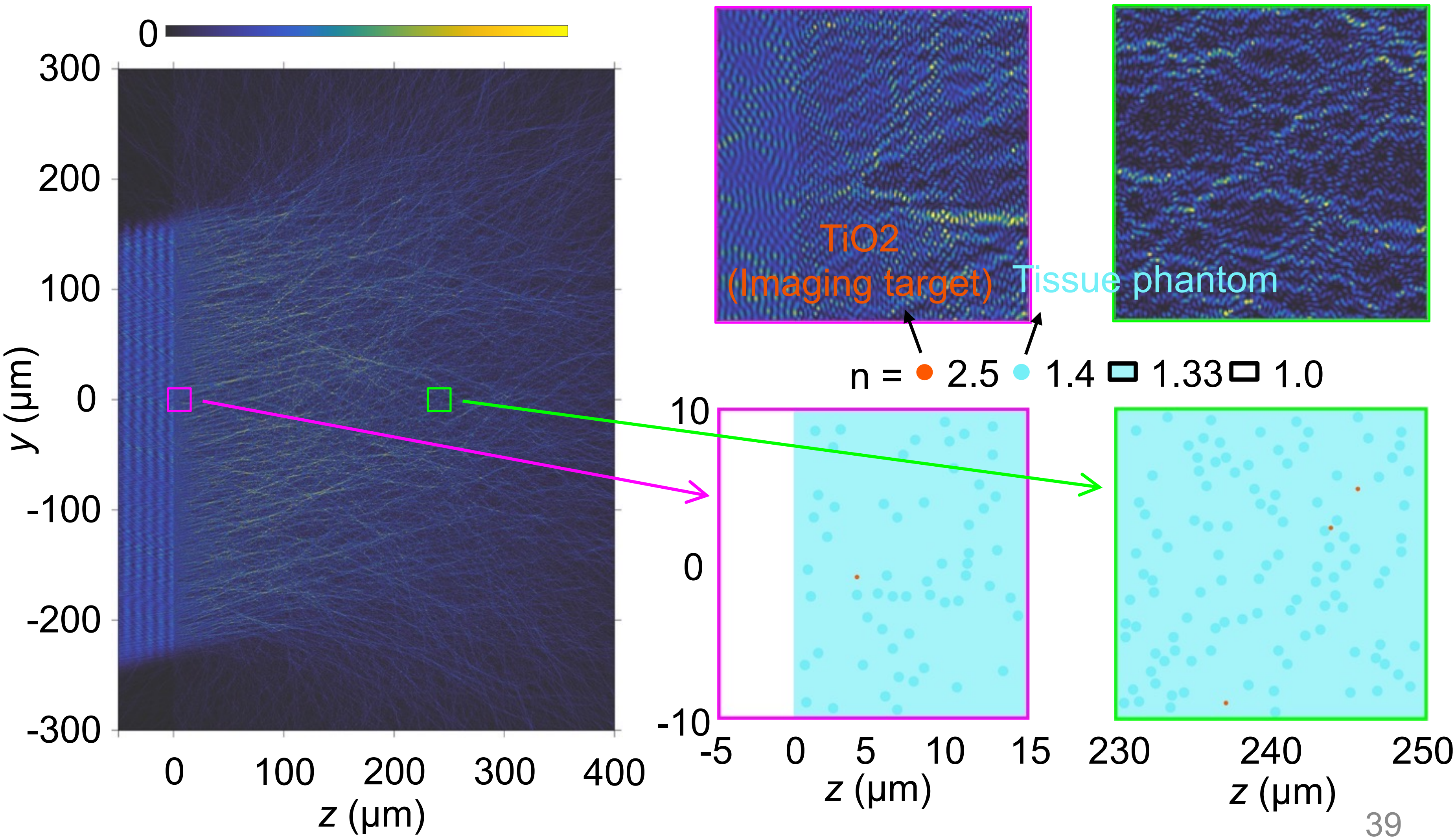
TiO_2 nanoparticles (500-nm diameter) in PDMS
Transport mean free path: 1 mm



Y. Zhang *et al*, arXiv:2306.08793

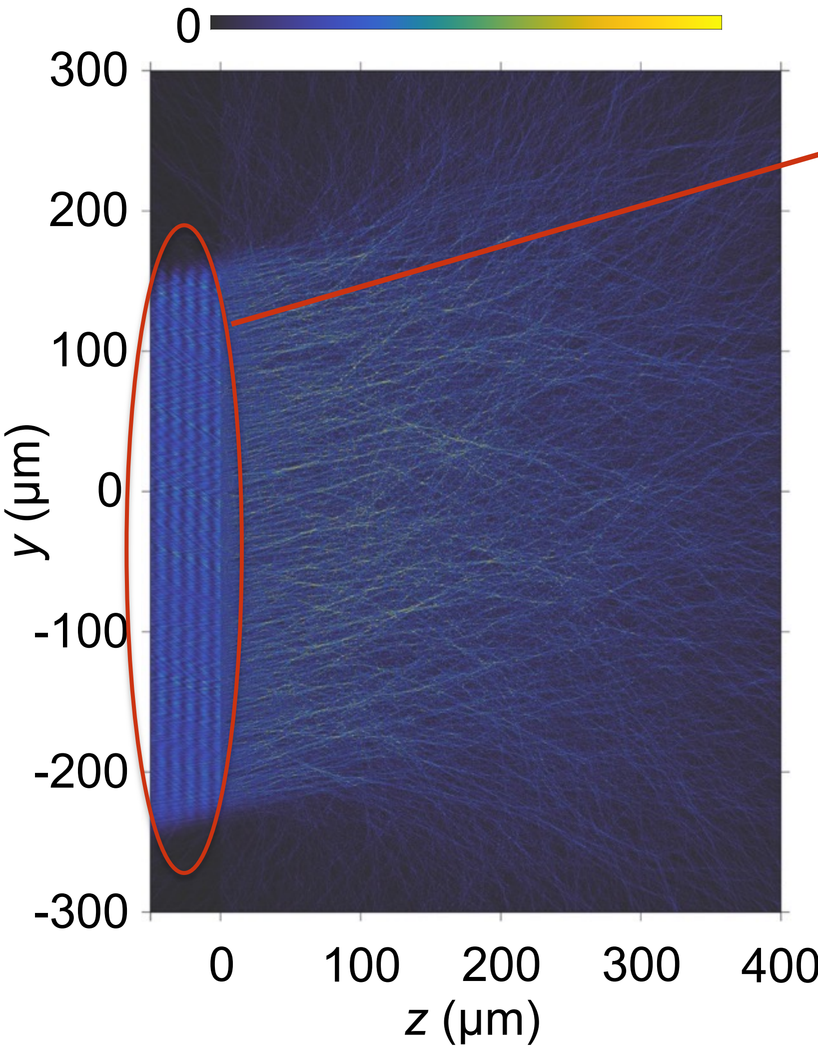
Numerical experiment with full-wave simulations

TiO_2 nanoparticles (300 nm diameter) in tissue phantom
$(\ell_s = 44\ \mu m,\ \ell_t = 340\ \mu m)$



Numerical experiment with full-wave simulations

TiO_2 nanoparticles (300 nm diameter) in tissue phantom
$(\ell_s = 44\ \mu\text{m},\ \ell_t = 340\ \mu\text{m})$



Project reflected wave onto plane waves at different angles

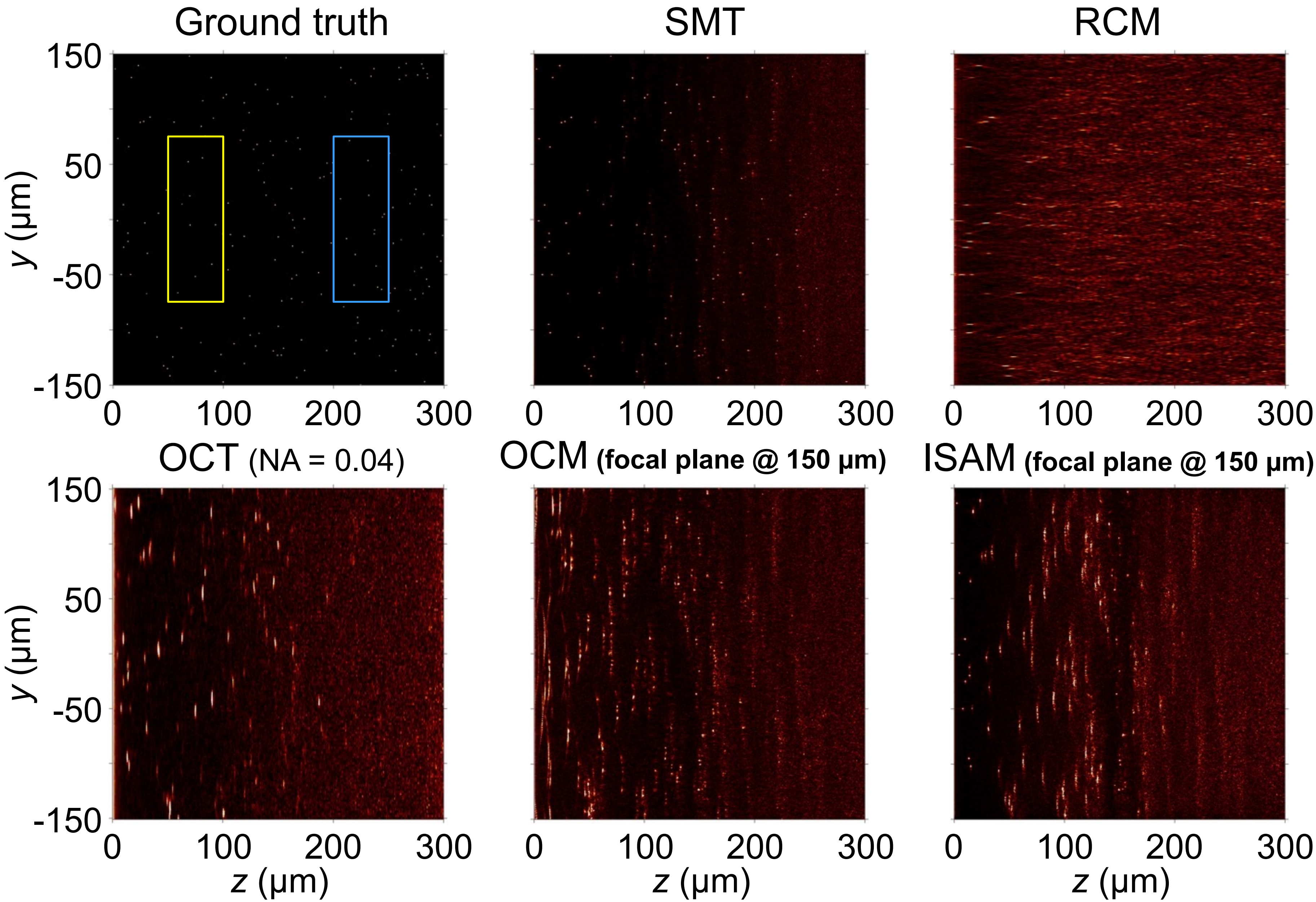
⇨ One column of $R(\mathbf{k}_{\text{out}}, \mathbf{k}_{\text{in}}, \omega)$

Use MESTI to compute $R(\mathbf{k}_{\text{out}}, \mathbf{k}_{\text{in}}, \omega)$
(simulation time: 4 minutes per wavelength)

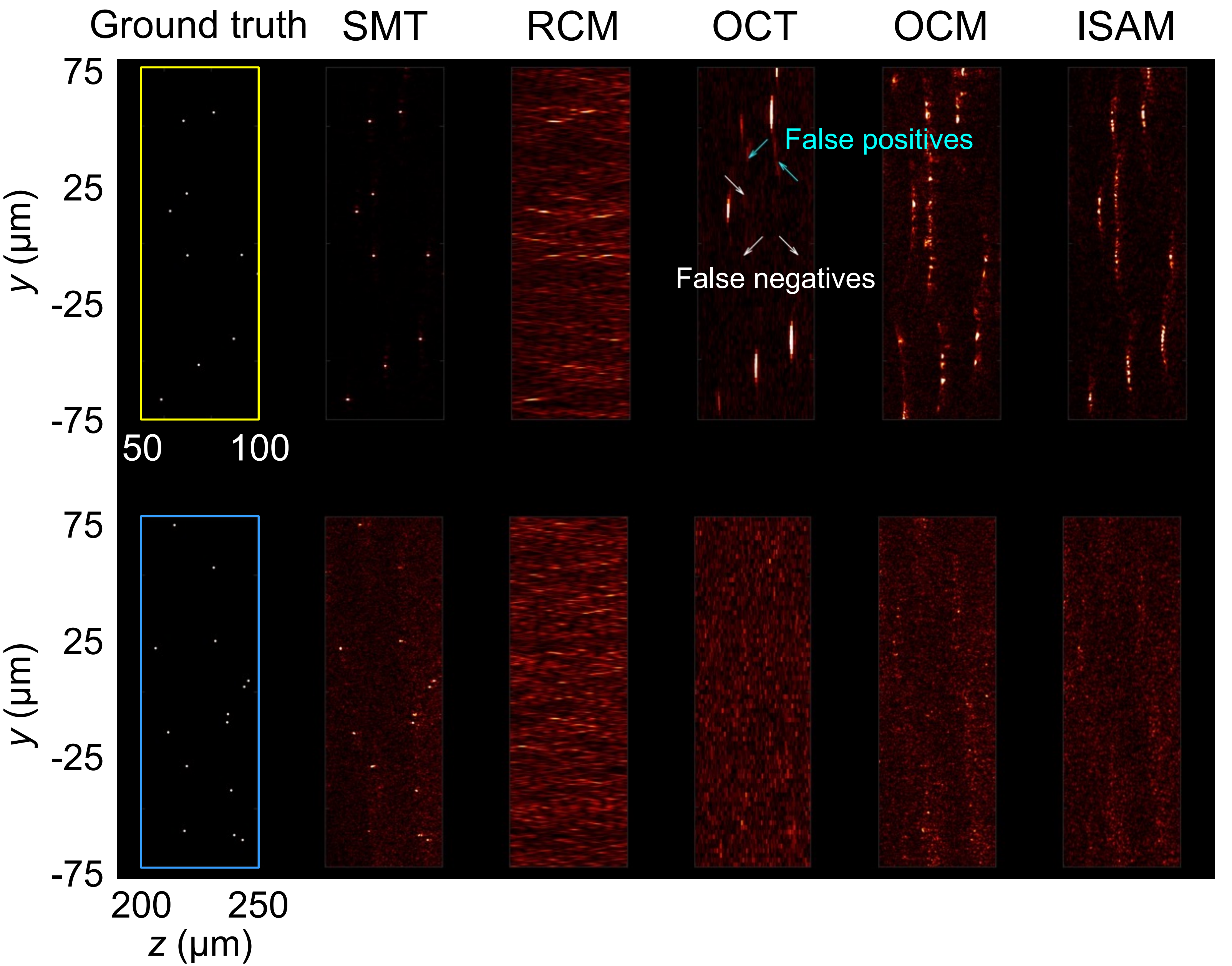
Compute $R(\mathbf{k}_{\text{out}}, \mathbf{k}_{\text{in}}, \omega)$ with

- 600 wavelengths within $\lambda \in [700,1000]$ nm
- $\text{NA}_{\text{out}} = \text{NA}_{\text{in}} = 0.5$ (**600~900 angles each**)

SMT from full-wave simulations



Comparing methods with zoom-in



Outline

1. Augmented partial factorization (APF) method
2. Applications of APF (all done with MESTI):
   a) Two-photon coherent backscattering
   b) Vectorial open channel in 3D
   c) Noninvasive imaging deep inside scattering media
   d) Inverse design of metasurfaces

Inverse design

Desired properties & constraints

Figure of merit f

Find parameters P_{opt} that maximize f and yield the desired properties

System parameters $P = \{p_1, \dots, p_K\}$

Efficient optimization requires the **gradient** $\vec{\nabla}_P f = \left\{\frac{\partial f}{\partial p_1}, \dots, \frac{\partial f}{\partial p_K}\right\}$

Adjoint method:

1 input: 1 forward simulation + 1 adjoint simulation $\Rightarrow \vec{\nabla}_P f$

M inputs: *M* forward simulation + *M* adjoint simulation $\Rightarrow \vec{\nabla}_P f$

Gradient computation using APF

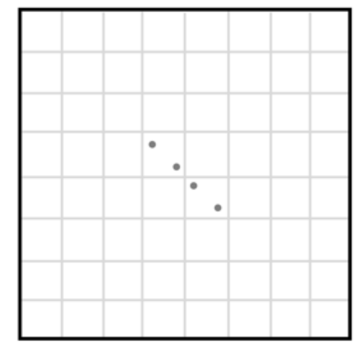
Figure of Merit (FoM): $f[\mathbf{S}(P), P]$

$\mathbf{S}$ --- scattering matrix $\qquad \mathbf{S} = \mathbf{C}\mathbf{A}^{-1}\mathbf{B} - \mathbf{D}$

P --- parameters to be optimized

$$\frac{df}{dp_k} = \frac{\partial f}{\partial p_k} + \sum_{n,m} 2\mathrm{Re}\left(\frac{\partial f}{\partial S_{nm}} \frac{\partial S_{nm}}{\partial p_k}\right)$$

Low-rank matrix $\partial \mathbf{A}/\partial p_k$

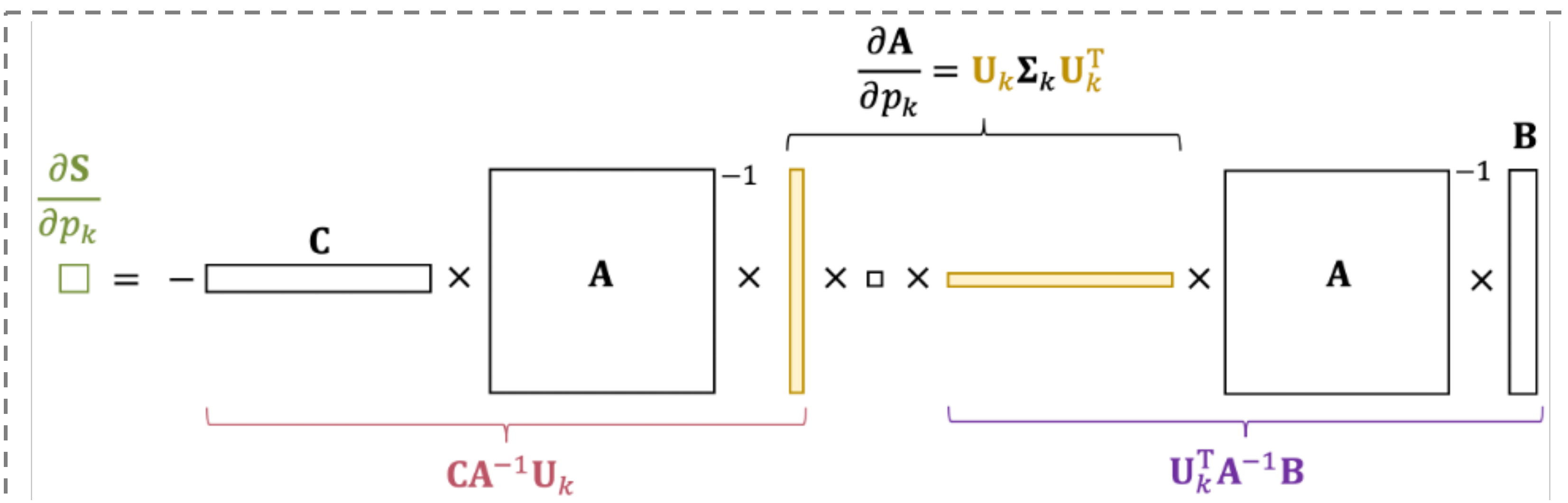


For a single parameter p_k:

$$\frac{\partial \mathbf{S}}{\partial p_k} = \mathbf{C}\frac{\partial \mathbf{A}^{-1}}{\partial p_k}\mathbf{B}$$

$$= -\mathbf{C}\mathbf{A}^{-1}\frac{\partial \mathbf{A}}{\partial p_k}\mathbf{A}^{-1}\mathbf{B}$$

$$= -\mathbf{C}\mathbf{A}^{-1}\mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{U}_k^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}$$



Gradient computation using APF

Figure of Merit (FoM): $f[\mathbf{S}(P), P]$

$\mathbf{S}$ --- scattering matrix $\quad \mathbf{S} = \mathbf{C}\mathbf{A}^{-1}\mathbf{B} - \mathbf{D}$

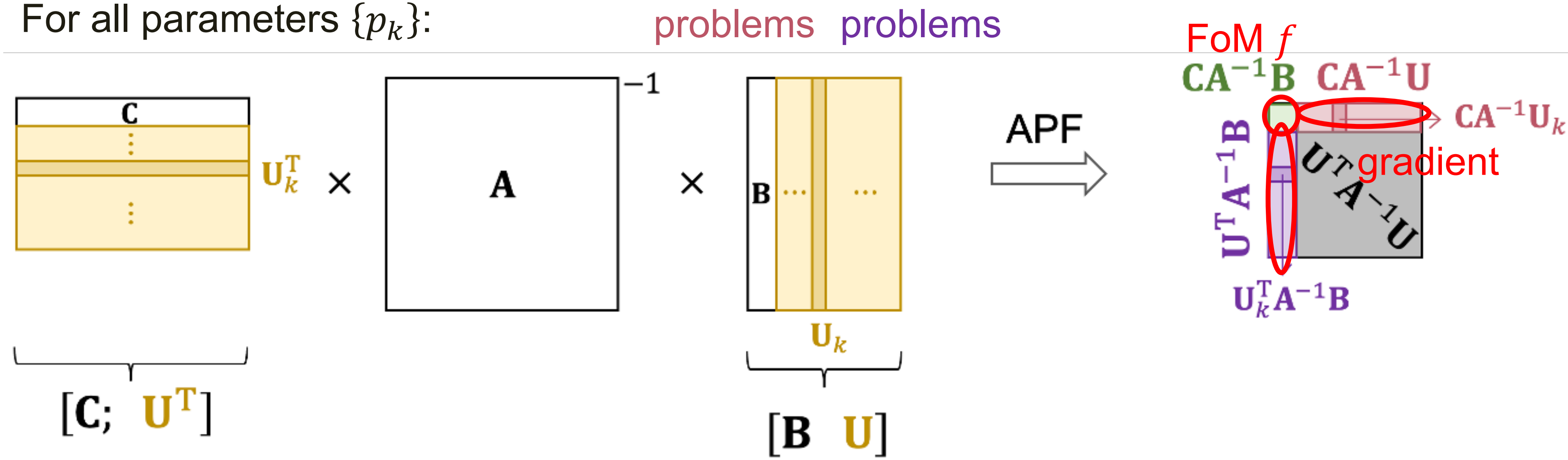
P --- parameters to be optimized

$$\frac{df}{dp_k} = \frac{\partial f}{\partial p_k} + \sum_{n,m} 2\mathrm{Re}\left(\frac{\partial f}{\partial S_{nm}} \frac{\partial S_{nm}}{\partial p_k}\right)$$

For a single parameter p_k: $\dfrac{\partial \mathbf{S}}{\partial p_k} = -\mathbf{C}\mathbf{A}^{-1}\mathbf{U}_k \boldsymbol{\Sigma}_k \mathbf{U}_k^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}$

Adjoint problems ($\mathbf{C}\mathbf{A}^{-1}\mathbf{U}_k$) Forward problems ($\mathbf{U}_k^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B}$)

For all parameters $\{p_k\}$:



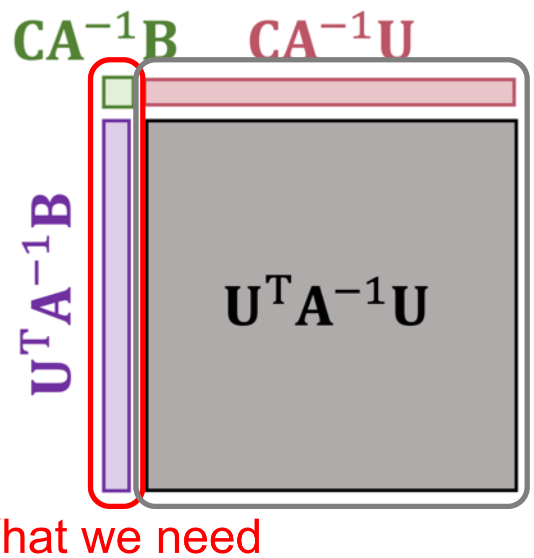
A single APF computation yields the multi-channel FoM and its gradient

S. Li, H.-C. Lin, CWH, arXiv:2306.09257

Redundancy in APF… and a partial remedy

Compute using APF

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{U}^{\mathrm{T}} \end{bmatrix} \mathbf{A}^{-1} [\mathbf{B} \quad \mathbf{U}]$$



What we DO NOT need

Partial remedy:

Divide matrices $\mathbf{U}$ and $\mathbf{U}^{\mathrm{T}}$ into N_{sub} subsets

If $N_{\mathrm{sub}} = 3$:

$\mathbf{U} = [\mathbf{U}_{(1)}, \mathbf{U}_{(2)}, \mathbf{U}_{(3)}]$

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{U}_{(1)}^{\mathrm{T}} \end{bmatrix} \mathbf{A}^{-1} [\mathbf{B} \quad \mathbf{U}_{(1)}]$$

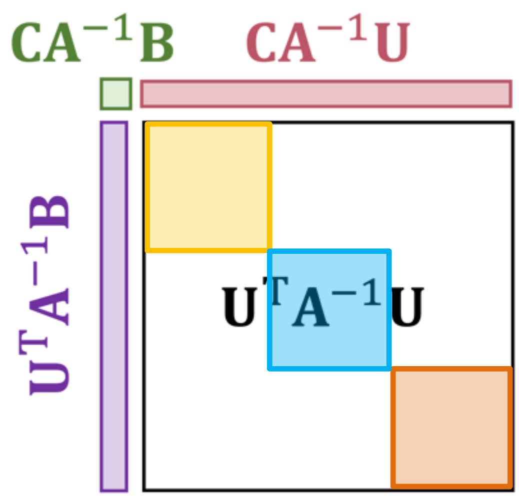
APF #1

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{U}_{(2)}^{\mathrm{T}} \end{bmatrix} \mathbf{A}^{-1} [\mathbf{B} \quad \mathbf{U}_{(2)}]$$

APF #2

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{U}_{(3)}^{\mathrm{T}} \end{bmatrix} \mathbf{A}^{-1} [\mathbf{B} \quad \mathbf{U}_{(3)}]$$

APF #3



Divide one large APF computation into N_{sub} sub-APF computations

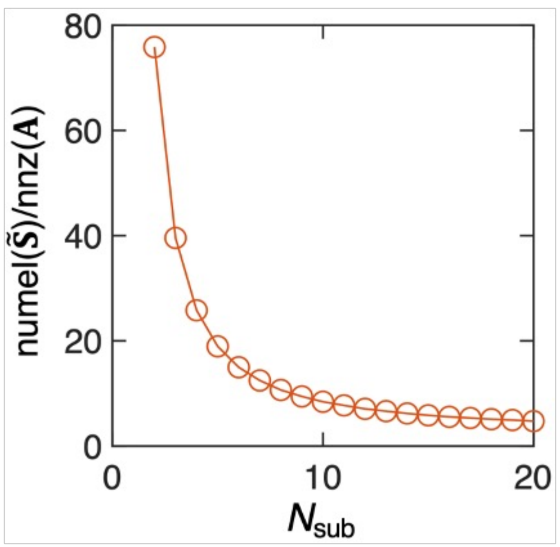
Less redundant…
But more repetitive

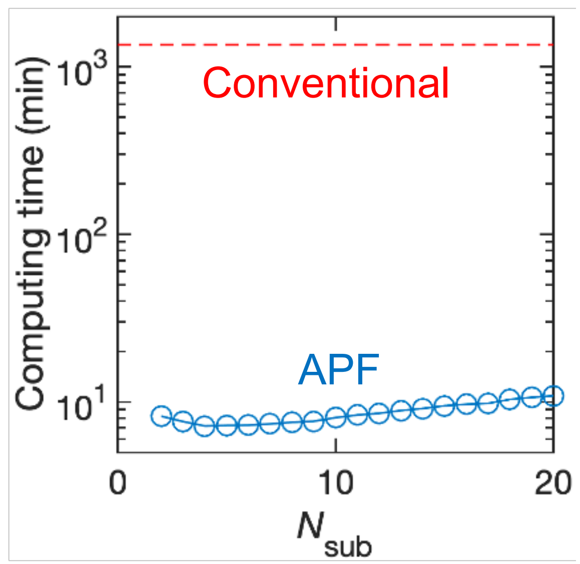
Dependence on N_{sub}

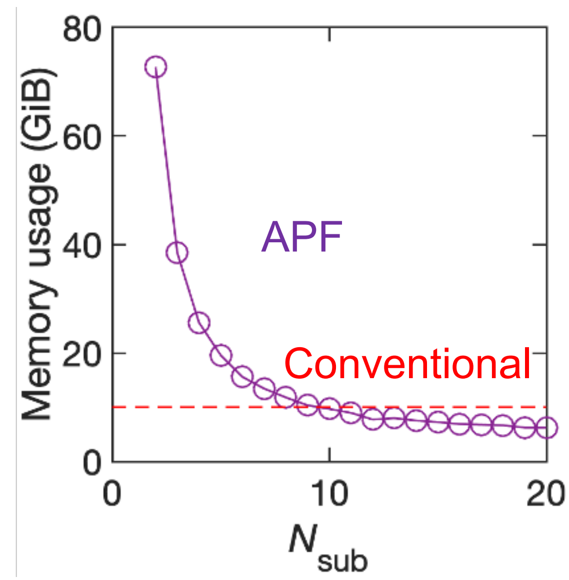
Divide one large APF computation into N_{sub} sub-APF computations

$$\tilde{\mathbf{S}} = \begin{bmatrix} \mathbf{C}\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}\mathbf{A}^{-1}\mathbf{U}_{(n)} \\ \mathbf{U}_{(n)}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B} & \mathbf{U}_{(n)}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{U}_{(n)} \end{bmatrix}$$

Metasurface with 1200 ridges:

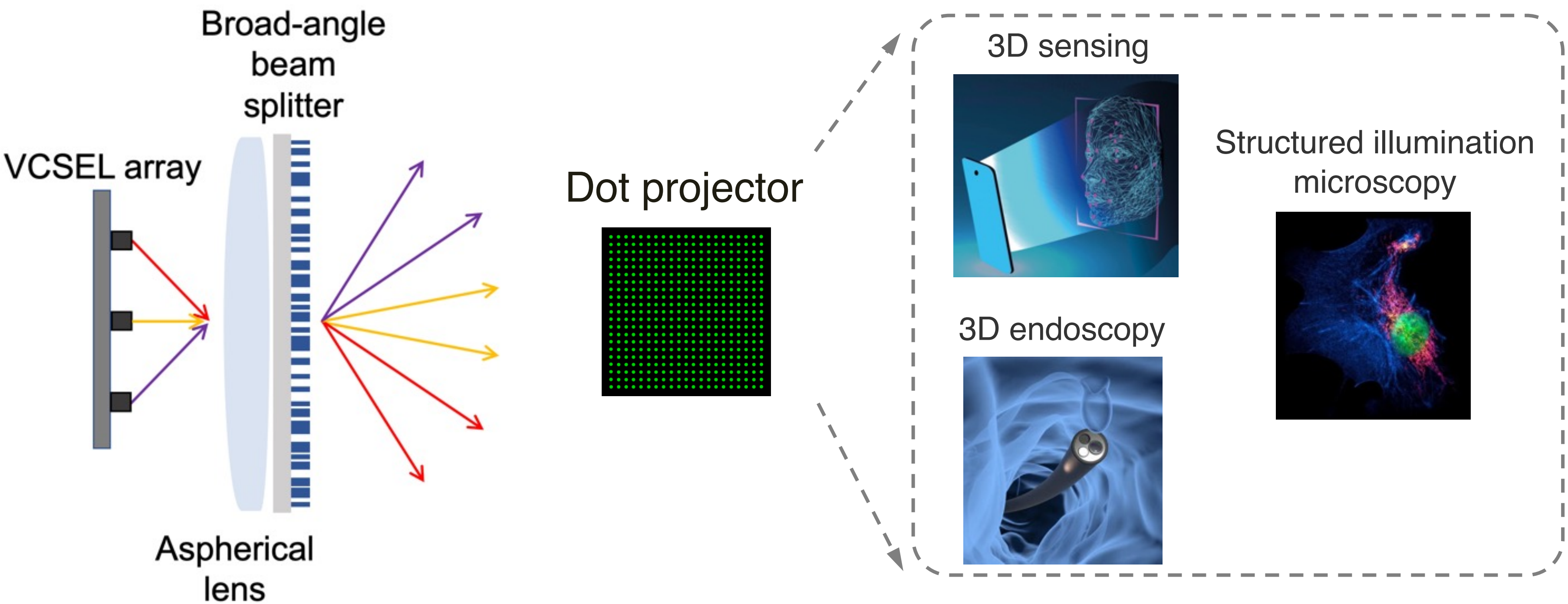






Conventional adjoint method: $2M_{\text{in}}$ simulations (M_{in} forward, M_{in} adjoint)

Optimize a broad-angle metasurface beam splitter

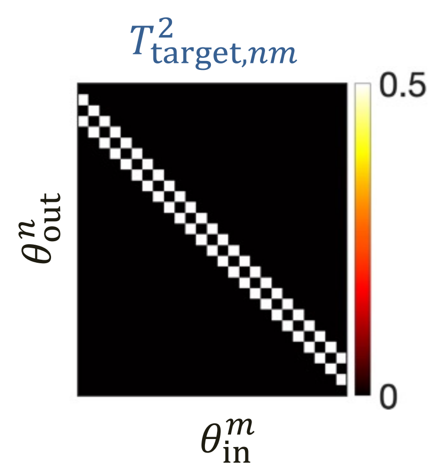


(VCSEL: Vertical-Cavity Surface-Emitting Laser)

Figure of Merit (FoM):

$$f(\mathbf{T}, P) = \sum_{n=1}^{M_{\text{out}}} \sum_{m=1}^{M_{\text{in}}} \left| |T_{nm}(P)|^2 - T_{\text{target},nm}^2 \right|^2$$

Transmission matrix: $\mathbf{T} = T_{nm} = T(\theta_{\text{out}}^n, \theta_{\text{in}}^m)$



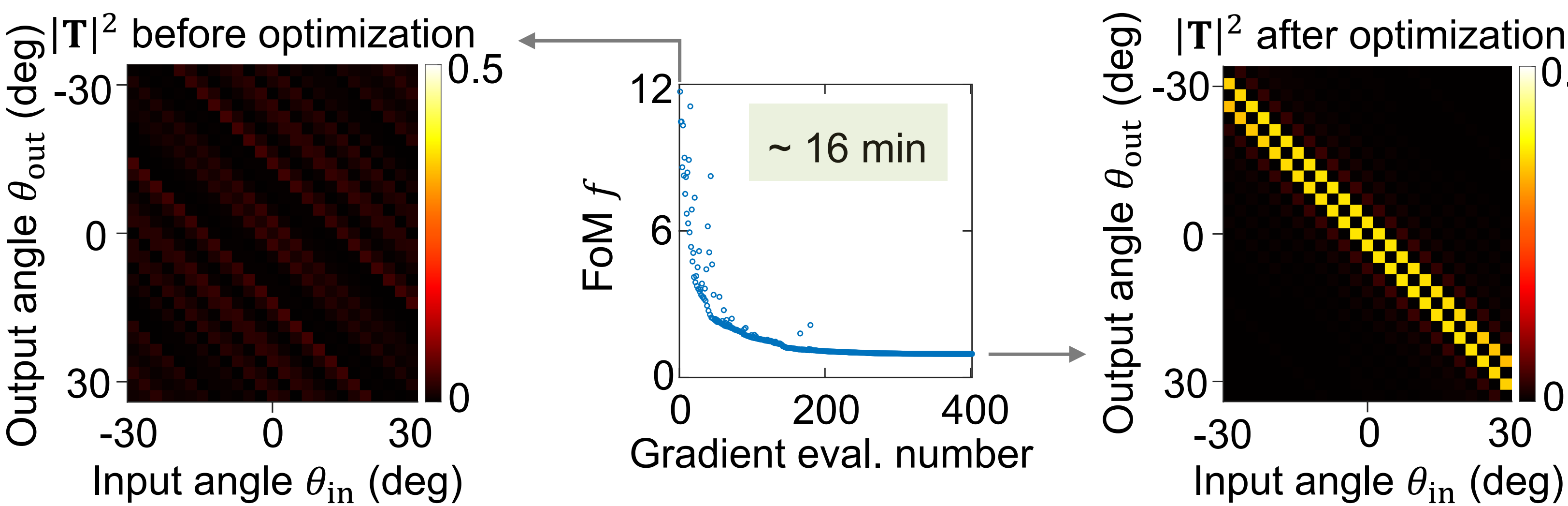
S. Li, H.-C. Lin, CWH, arXiv:2306.09257

Optimize a broad-angle metasurface beam splitter

- α-Si ridges sitting on a silica-substrate. Wavelength = 940 nm
- Parameters P = {edge positions}
- Angular range = 60°, 25 input angles, 51 output angles
- Optimized with the SLSQP algorithm in NLopt package
- Best result over 1000 randomly generated initial guesses

Before optimization:

After optimization:



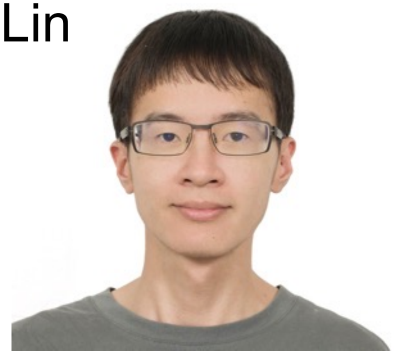
S. Li, H.-C. Lin, CWH, arXiv:2306.09257

Summary

1. Augmented partial factorization (APF) method
   - *Bypass unnecessary computation & Avoid repetition*
     ⇒ Fast computation of $\mathbf{C}\,\mathbf{A}^{-1}\mathbf{B}$
   - Enabled by the Schur complement feature of MUMPS
2. Applications of APF (all done with MESTI):
   a) Two-photon coherent backscattering
   b) Vectorial open channel in 3D
   c) Noninvasive imaging deep inside scattering media
   d) Inverse design of metasurfaces

USC | Advanced Research Computing
Enabling scientific breakthroughs at scale
NSF
Chan Zuckerberg Initiative
SONY

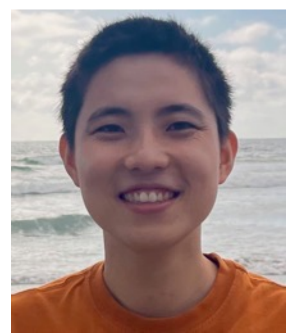
Augmented partial factorization (APF) solver:
Ho-Chun Lin



Two-photon coherent backscattering:
Hebrew Univ: Mamoon Safadi, Ohad Lib, Yaron Bromberg
Institut Langevin: Arthur Goetschy
USC: Ho-Chun Lin

Vectorial open channel in 3D:
Ho-Chun Lin

Imaging inside scattering media:
Yiwen Zhang, Zeyu Wang, Minh Dinh



Metasurfaces inverse design:
Shiyu Li

Thank you, MUMPS developers!