Simulating the static magnetic response of thin film superconducting devices

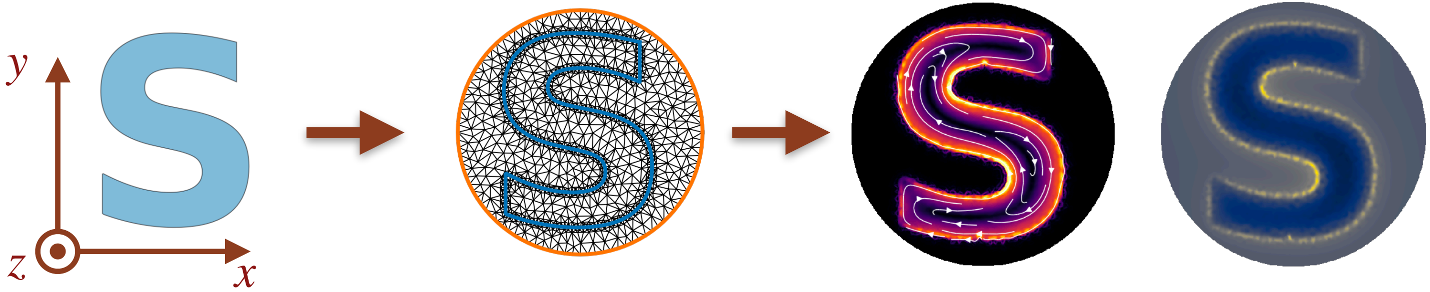
LOGAN BISHOP-VAN HORN [1], KATHRYN A. MOLER [1,2]

1. Stanford Institute for Materials and Energy Sciences, SLAC National Accelerator Laboratory, 2575 Sand Hill Road, Menlo Park, CA 94025
2. Department of Physics and Applied Physics, Stanford University, California 94305, USA

SIMES

U.S. DEPARTMENT OF ENERGY | Office of Science

Goal



2D superconducting film, penetration depth Λ, applied field $\mu_0 H_{z,\,\mathrm{applied}}$

Numerically solve 2D London equation

Extract Meissner screening currents and self-field

Physics Wish List

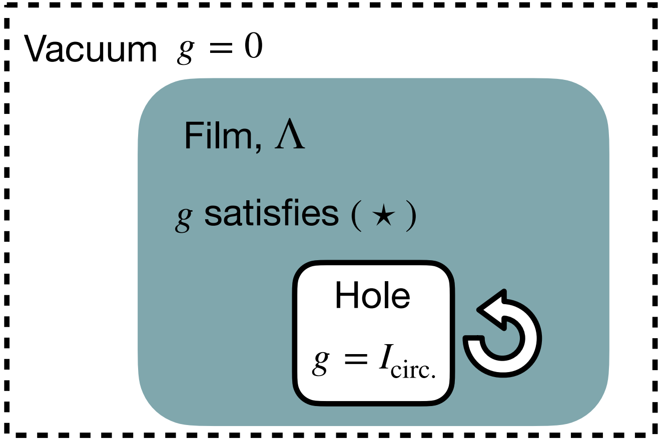
- Inhomogeneous $\Lambda(x, y)$
- Fluxoid quantization and mutual inductance in multiply-connected films
- Trapped vortices
- Stacked 2D films

Software Wish List

- User friendly
- Fast
- Open source
- Portable
- Interactive

The model

$\nabla \cdot \vec{J} = 0 \implies$ scalar stream function $g(x, y)$:
$\vec{J}(x, y) = \nabla \times (g\hat{z})$



2D London equation in terms of g:

$$\vec{H}(x, y) = \frac{\lambda^2}{d}\nabla^2 g(x, y)\hat{z} = \Lambda \nabla^2 g(x, y)\hat{z}$$

1. Brandt & Clem, PRB **69**, 184509 (2004).
2. Brandt, PRB **72**, 024529 (2005).
3. Khapaev, Supercon. Sci. Technol. (1997).
4. Kirtley, …, Supercon. Sci. Technol. (2016).

Biot-Savart in terms of g:

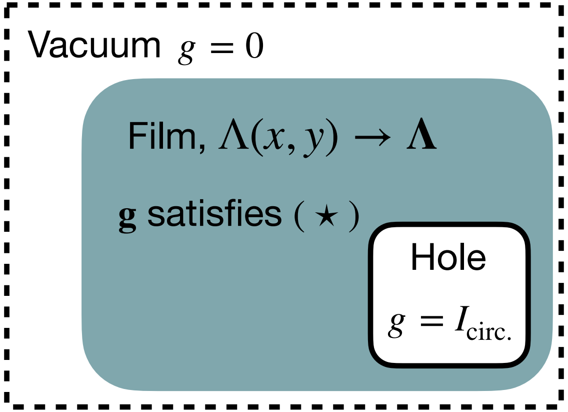
Applied field — Screening field — Total field

$$H_{z,\,\text{applied}}(\vec{r}) + \int_{\text{film}} Q_z(\vec{r}, \vec{r}')g(\vec{r}')\,\mathrm{d}^2 r' = \Lambda \nabla^2 g(\vec{r}) \qquad (\star)$$

Numerical implementation

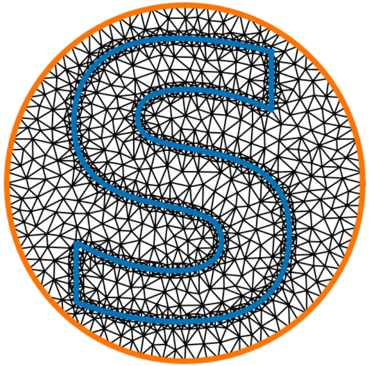
Applied field | Screening field | Total field

$$H_{z,\,\mathrm{applied}}(\vec{r}) + \int_{\mathrm{film}} Q_z(\vec{r},\vec{r}\,')g(\vec{r}\,')\,\mathrm{d}^2r' = \Lambda\,\nabla^2 g(\vec{r})$$



Discretize film and surrounding vacuum

- Delaunay triangulation —> n vertices with areas $\mathbf{w}$
- Dipole kernel Q_z —> dense $n \times n$ floating point matrix $\mathbf{Q}$
- Laplace operator ∇^2 —> sparse $n \times n$ floating point matrix $\mathbf{L}$

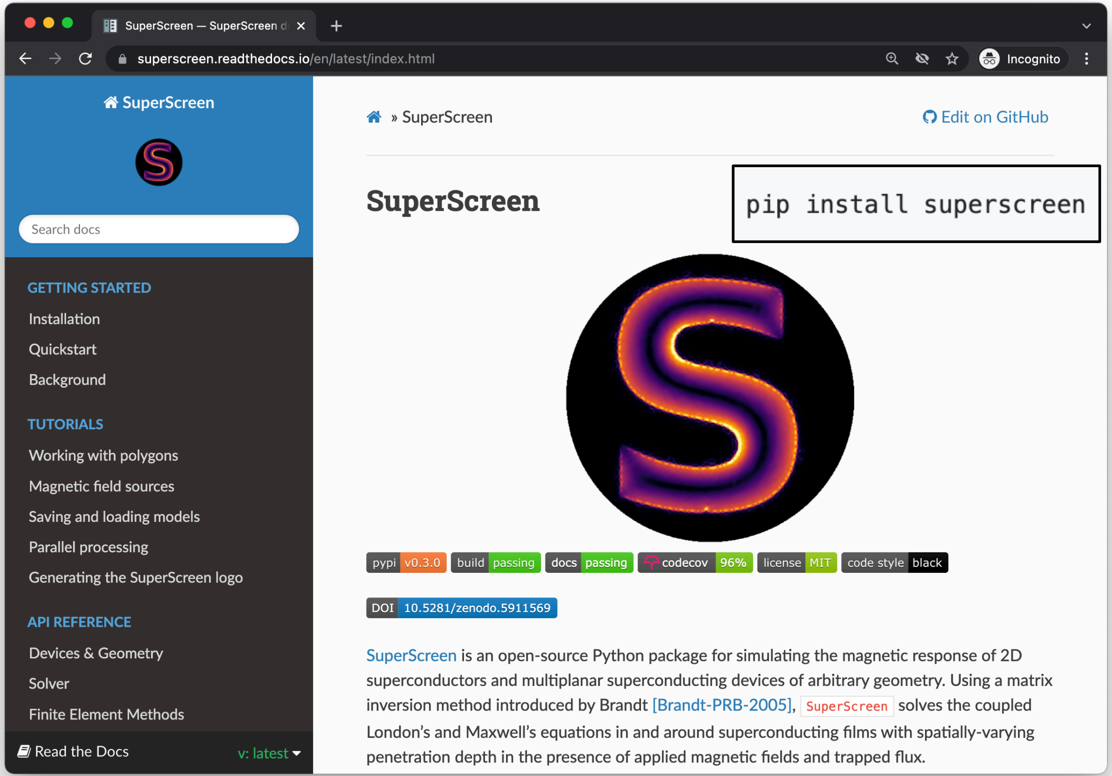


Solve linear system for unknown vector $\mathbf{g}$ inside the film:

$$\mathbf{h}_{z,\,\mathrm{applied}} = -\left(\mathbf{Q}\,.\,\mathbf{w}^T - \mathbf{L}\,.\,\mathbf{\Lambda}^T\right)\mathbf{g} \quad (\star)$$

1. Brandt, PRB **72**, 024529 (2005).
2. Kirtley, …, Supercon. Sci. Technol. (2016).

Open source software implementation



superscreen.readthedocs.io, github.com/loganbvh/superscreen

Example: Superconducting ring with a slit

```python
import superscreen as sc
from superscreen.geometry import circle, box
```

Import the package

```python
# Define the device geometry.
length_units = "um"
ro = 3  # outer radius
ri = 1  # inner radius
slit_width = 0.25
Lambda = 1  # effective penetration depth
# circle() and box() generate arrays of polygon (x, y) coordinates.
ring = circle(ro)
hole = circle(ri)
slit = box(slit_width, 1.5 * (ro - ri), center=(0, -(ro + ri) / 2))
# Define the Polygon representing the superconductor.
layer = sc.Layer("base", Lambda=Lambda)
film = sc.Polygon.from_difference(
    [ring, slit, hole], name="ring_with_slit", layer="base"
)
bounding_box = sc.Polygon("bounding_box", layer="base", points=circle(1.2 * ro))
# Create a Device and generate and plot the computational mesh.
device = sc.Device(
    film.name,
    layers=[layer],
    films=[film],
    abstract_regions=[bounding_box],
    length_units=length_units,
)
```

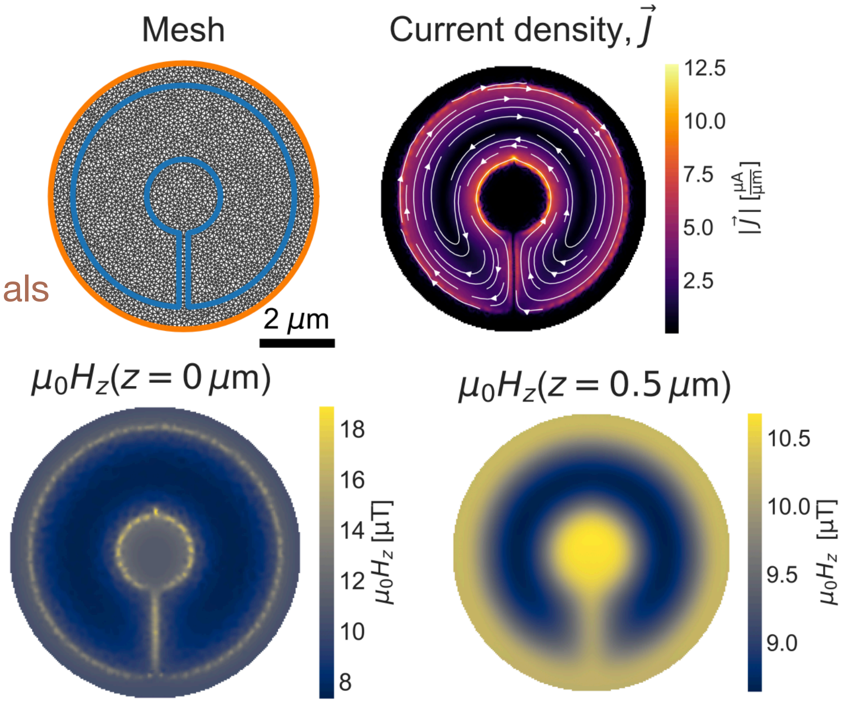
Define geometry and materials

```python
device.make_mesh(min_points=3500, optimesh_steps=None)
device.plot(mesh=True)
# Calculate the device's response to a uniform applied field.
applied_field = sc.sources.ConstantField(10)
solution = sc.solve(device, applied_field=applied_field, field_units="uT")[-1]
```

Solve the model

```python
# Visualize the solution.
# Plot the current density evaluated at each layer in the Device.
solution.plot_currents()
# Plot the magnetic field evaluated at each layer in the Device.
solution.plot_fields()
# Plot the field evaluated at any points in space.
solution.plot_field_at_positions(device.points, zs=0.5)
```

Visualize the results

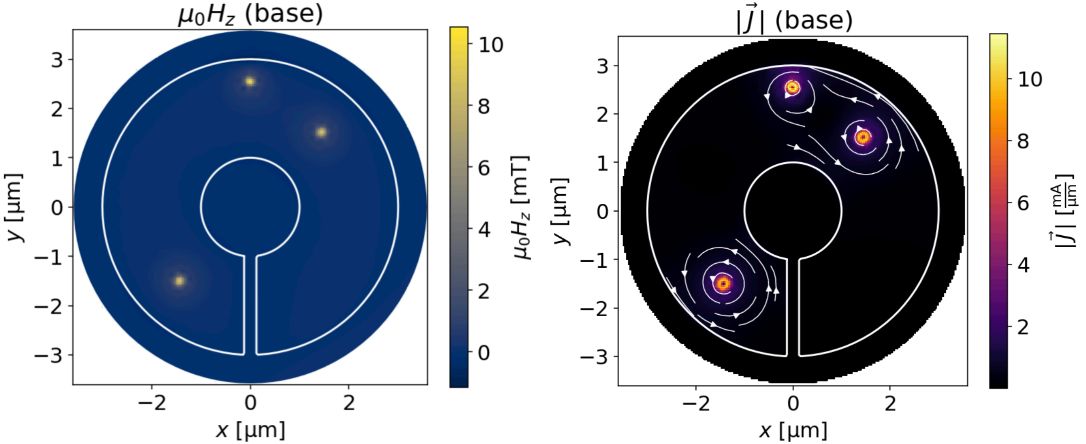


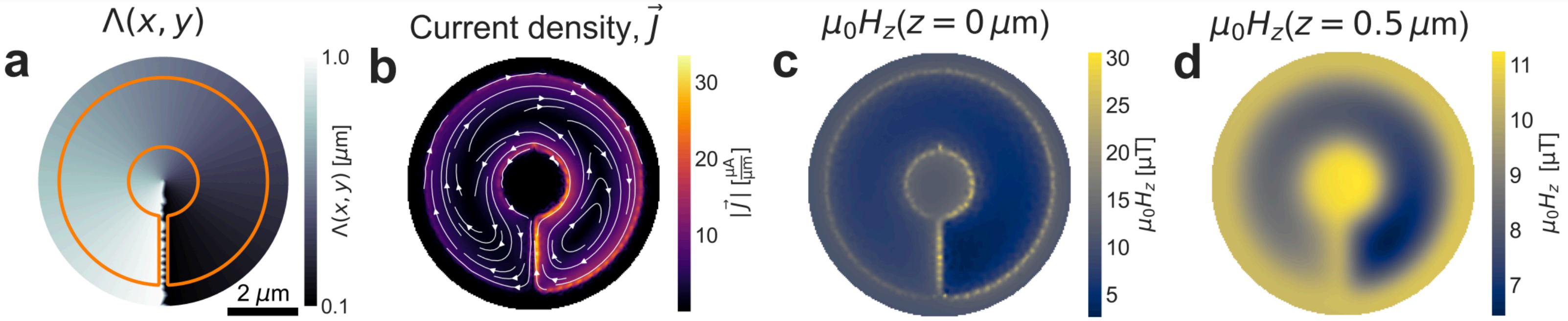
Logan Bishop-Van Horn, superscreen.readthedocs.io

Stanford University

Example: Trapped vortices, inhomogeneous $\Lambda(x, y)$

```python
vortices = [
    sc.Vortex(x=1.5, y=1.5, layer="base"),
    sc.Vortex(x=-1.5, y=-1.5, layer="base"),
    sc.Vortex(x=0, y=2.5, layer="base"),
]
```





Logan Bishop-Van Horn, superscreen.readthedocs.io

Example: Fluxoid states

Fluxoid quantization:

$$\Phi^f = \overbrace{\int_S \mu_0 H_z(\vec{r})\,\mathrm{d}^2r}^{\text{"flux part"}} + \overbrace{\oint_{\partial S} \mu_0 \Lambda(\vec{r})\,\vec{J}(\vec{r})\cdot \mathrm{d}\vec{r}}^{\text{"supercurrent part"}} = n\Phi_0, \quad n \in \mathbb{Z}$$

$\Lambda = 0.25\,\mu\mathrm{m}$, $\mu_0 H_{z,\,\mathrm{applied}} = 1\,\mathrm{mT}$

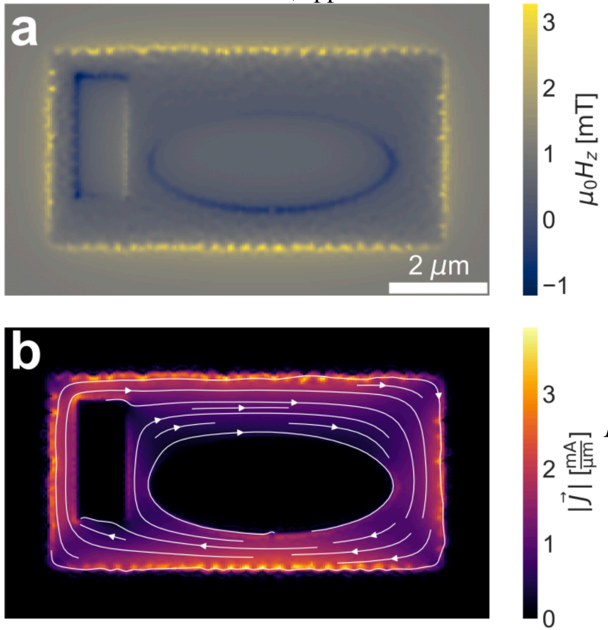
Singly-connected films, $N_h = 0$ holes

- Fluxoid quantization satisfied by solutions to 2D London equation with $n = 0$

Multiply-connected films, $N_h > 0$ holes

- Adjust circulating currents $\{I_h\}$ via gradient descent to realize desired fluxoid state $\{\Phi^f_h\}$
- Can also compute mutual inductance matrix:

$$M_{ij} = \frac{\Phi^f_{\text{hole } i}}{I_{\text{circ. } j}}$$



$\Phi^f_{\text{rectangle}} = 0\,\Phi_0$

$\Phi^f_{\text{ellipse}} = 0\,\Phi_0$

$I^{\text{rectangle}}_{\text{circ.}} = -1.07\,\mathrm{mA}$

$I^{\text{ellipse}}_{\text{circ.}} = -1.59\,\mathrm{mA}$

Model 2D superconductors

- Create complex geometries and solve for their magnetic response in a few lines of code
- Generate publication-quality visualizations
- Run on a laptop, HPC cluster, or anything in between

Share simulations with the research community

- Publish interactive Jupyter notebooks to allow others to learn from and reproduce your results

Additional Features

- Built-in magnetic field sources: distribution of dipoles, monopoles, Pearl vortices, 2D current distribution
- Robustly save/load results to/from disk
- Extensive online documentation
- Automated unit test suite

Limitations

- 2D only: $\lambda_{\mathrm{London}} > d$
- Only circulating currents, no “terminal currents”
- Slow convergence + memory-intensive for structures with many layers

Use Cases

- Inductance extraction for 2D superconducting (e.g. VdW) devices
- Modeling of magnetic microscopy, including scanning SQUID magnetometry + susceptometry

Future Work

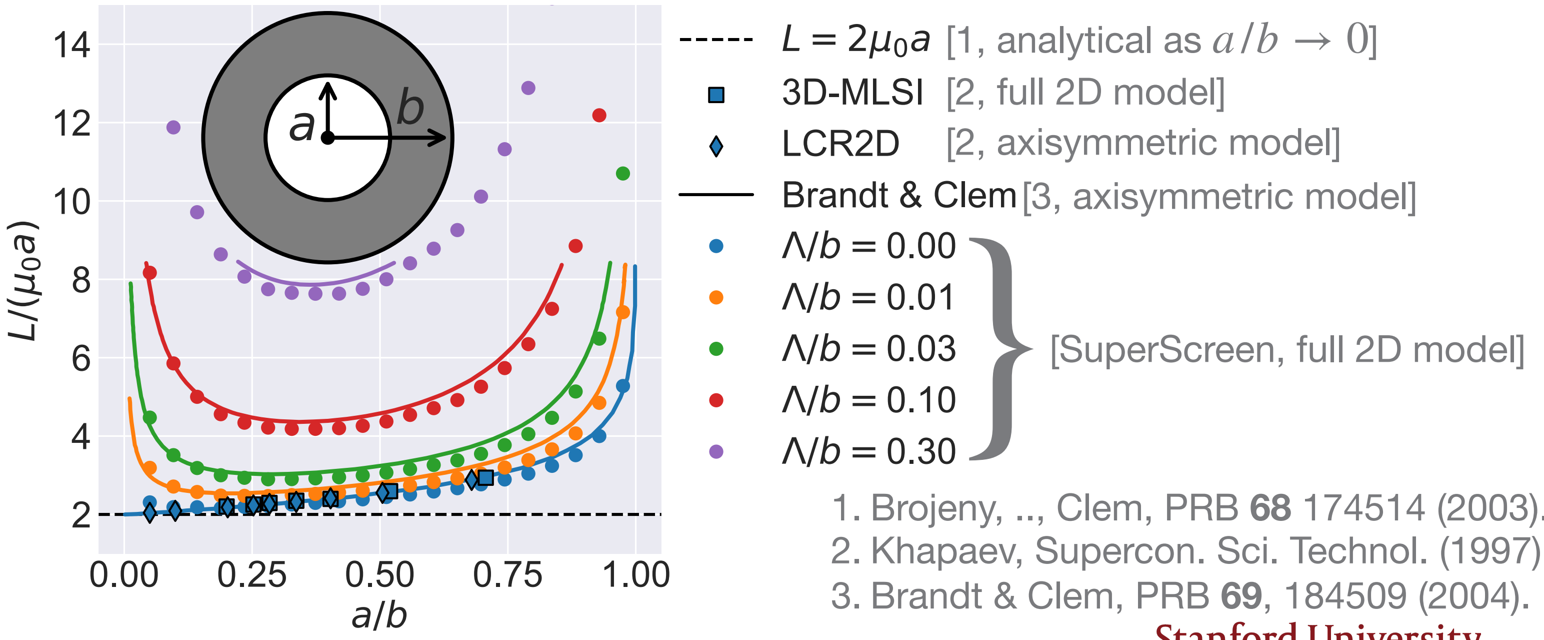
- GPU-acceleration
- Automated or adaptive mesh refinement
- Integration with CAD software/file formats

Acknowledgements

- John Kirtley: MATLAB implementation for modeling scanning SQUID microscopy:
  - Supercond. Sci. Technol. **29** (2016) 124001.
  - Rev. Sci. Instrum. 87, 093702 (2016).
- John Kirtley, Yusuke Iguchi: useful comments, discussions

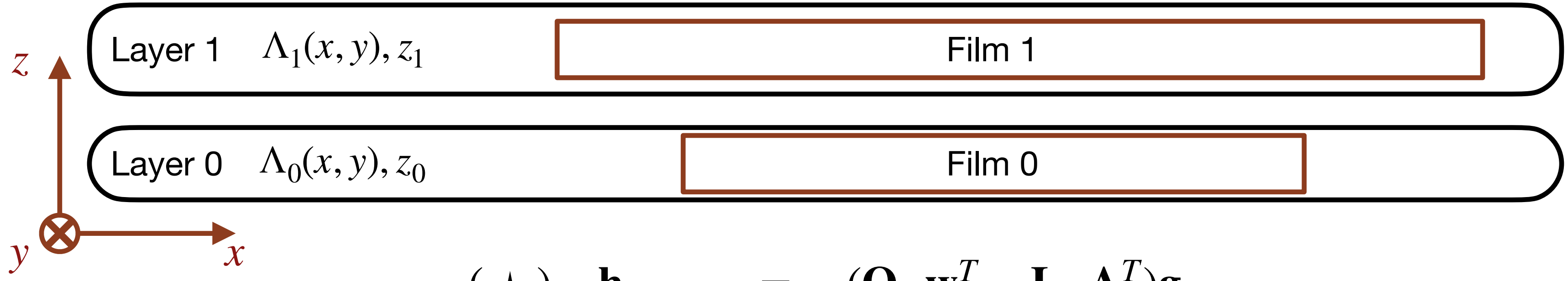
Example: Self-inductance of a flat ring

$$L = \frac{\Phi^f}{I_{\mathrm{circ.}}} = \frac{\overbrace{\int_S \mu_0 H_z(\vec{r})\,\mathrm{d}^2r}^{\text{"flux part"}} + \overbrace{\oint_{\partial S} \mu_0 \Lambda(\vec{r})\,\vec{J}(\vec{r})\cdot\mathrm{d}\vec{r}}^{\text{"supercurrent part"}}}{I_{\mathrm{circ.}}} = L_{\mathrm{geo.}} + L_{\mathrm{kin.}}$$



1. Brojeny, .., Clem, PRB **68** 174514 (2003).
2. Khapaev, Supercon. Sci. Technol. (1997).
3. Brandt & Clem, PRB **69**, 184509 (2004).

Numerical implementation: Multiple layers

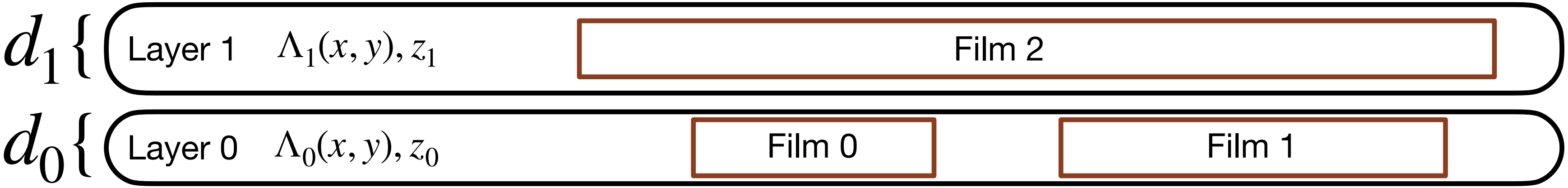


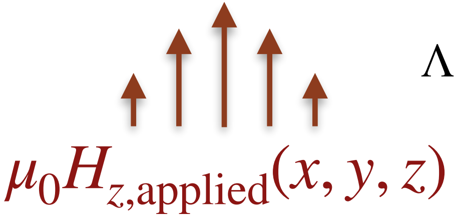
$$(\star) \quad \mathbf{h}_{z,\,\mathrm{applied}} = -(\mathbf{Q}\,.\,\mathbf{w}^T - \mathbf{L}\,.\,\mathbf{\Lambda}^T)\mathbf{g}$$

- Solve ($\star$) for each layer ℓ to obtain stream function $\mathbf{g}_\ell$
- For each layer ℓ, add to $\mathbf{h}_{z,\,\mathrm{applied}}$ the field due to $\mathbf{g}_k$ for all layers $k \neq \ell$
- Re-solve ($\star$) with updated applied field
- Repeat until solution converges

1. Brandt, PRB **72**, 024529 (2005).
2. Kirtley, …, Supercon. Sci. Technol. (2016).

Problem statement





$$\Lambda = \frac{\lambda_{\mathrm{London}}^2}{d} = \frac{\Lambda_{\mathrm{Pearl}}}{2}$$

Inputs

- $\Lambda_\ell(x, y)$, z_ℓ for each **layer** ℓ
- $x - y$ geometry for each **film** in each layer ℓ
- Applied field, $\mu_0 H_{z,\,\mathrm{applied}}(x, y, z)$

Outputs

- Sheet current density $\vec{J}_\ell(x, y)$ in each layer ℓ
- Total magnetic field $\mu_0 \vec{H}(x, y, z)$ anywhere in space

Assumptions

- Layers are 2D, $\lambda_{\mathrm{London}} > d$, and obey the London equation $\nabla \times \vec{J} = -\vec{H}/\Lambda$