CBMS Conference on Fast Direct Solvers

Dartmouth College

June 23 – June 27, 2014

Lecture 7: The Hierarchical Poincaré-Steklov scheme

This lecture is merely a brief intro to connect the HPS scheme to the other lectures. For details, see Adrianna Gillman's lecture on Wednesday afternoon.

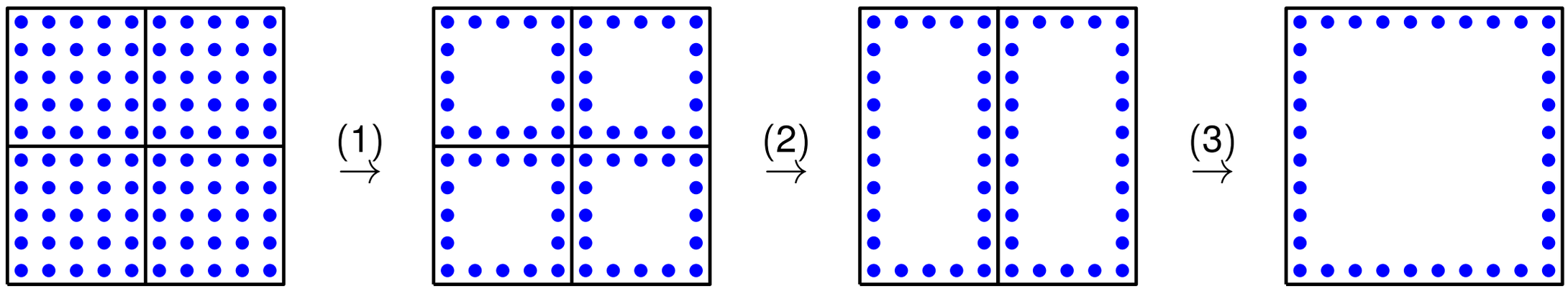
Gunnar Martinsson

The University of Colorado at Boulder

Research support by:

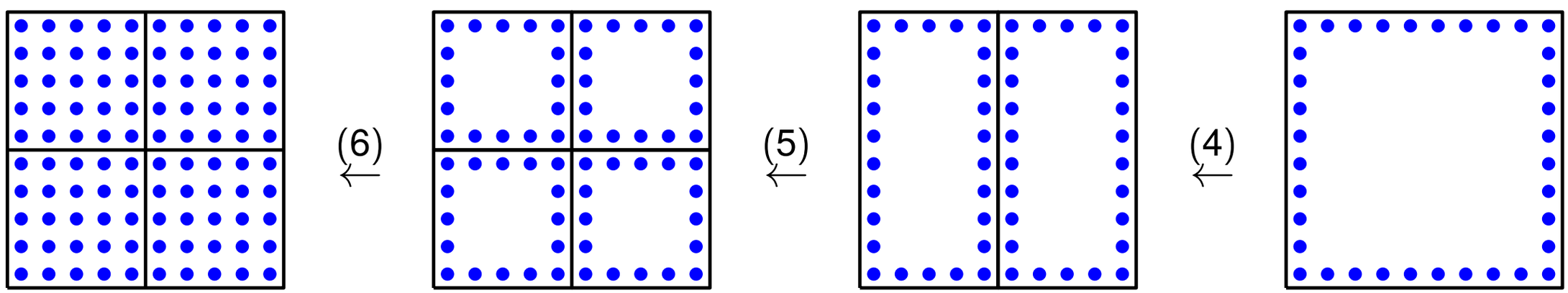
In Lecture 6, we described a direct solver for sparse matrices arising from FD/FEM discretization of an elliptic PDE, with the following pattern:

Upwards pass — build all solution operators:



The original grid. *Leaves reduced.* *After merge.* *After merge.*

Downwards pass — solve for a particular data function (*very* fast!):



Full solution. *Solve.* *Solve.* *Top level solve.*

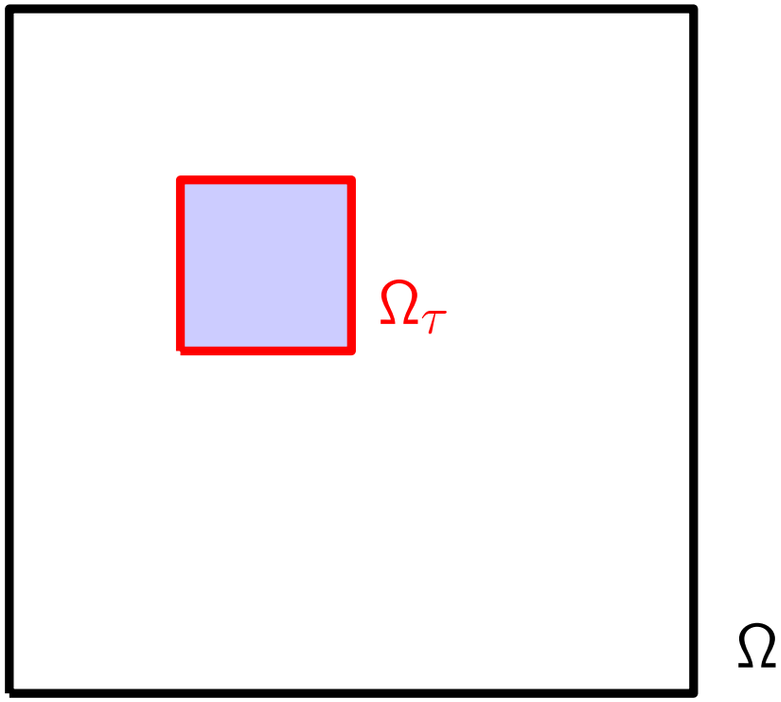
In the upwards pass, we construct for each box a local solution operator that serves as some rough analog of a Dirichlet-to-Neumann operator.

We will next build a direct solver that *explicitly* builds an approximate DtN operator.

Consider a domain Ω on which we are given a PDE

$$(1) \qquad \begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega, \end{cases}$$

where $b = b(\boldsymbol{x})$ is a given function (smooth, non-negative).



Let $\Omega_\tau \subset \Omega$, and suppose that we somehow know the value of u on $\partial\Omega_\tau$

$$f_\tau = u|_{\partial\Omega_\tau}.$$

Now, the restriction of (1) to Ω_τ,

$$(2) \qquad \begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega_\tau \\ u(\boldsymbol{x}) = f_\tau(\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega_\tau, \end{cases}$$

has a unique solution, so given f_τ, we can determine $u|_{\Omega_\tau}$, and then also $u_n|_{\partial\Omega_\tau}$.
We define the *Dirichlet-to-Neumann map* T_τ as the unique map

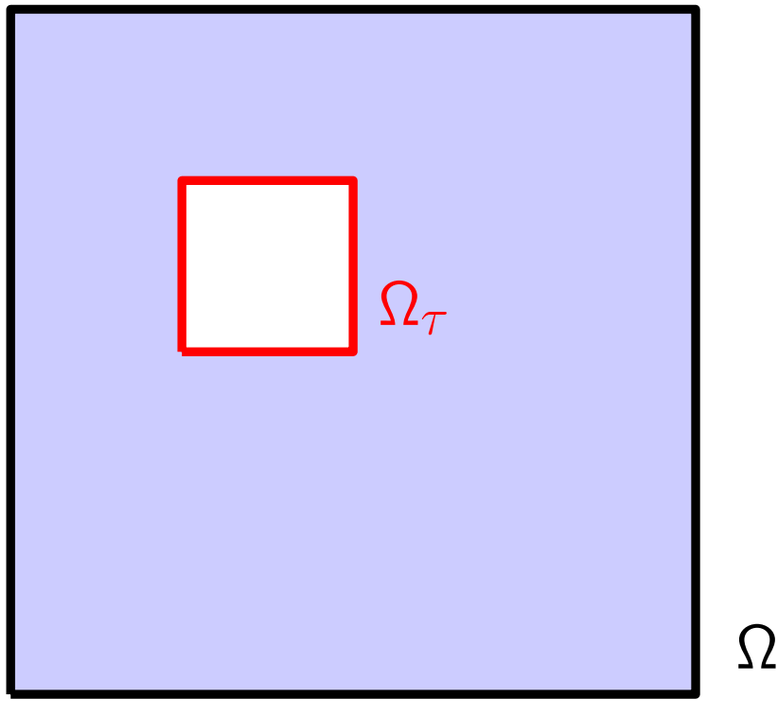
$$T_\tau : \; u|_{\partial\Omega_\tau} \mapsto u_n|_{\partial\Omega_\tau}, \qquad \text{subject to } -\Delta u + b\, u = 0 \text{ in } \Omega_\tau.$$

The map T_τ *encodes everything you need to know about Ω_τ to solve the PDE on $\Omega \backslash \Omega_\tau$.*

Consider a domain Ω on which we are given a PDE

$$(3) \qquad \begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega, \end{cases}$$

where $b = b(\boldsymbol{x})$ is a given function (smooth, non-negative).



Blue area is $\Omega \backslash \Omega_\tau$.

Suppose we are given T_τ for the sub-domain Ω_τ.
Then in $\Omega \backslash \Omega_\tau$, the solution to the boundary value problem

$$(4) \qquad \begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega \backslash \Omega_\tau \\ u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega, \\ u_n(\boldsymbol{x}) = [T_\tau u](\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega_\tau, \end{cases}$$

is exactly the same as the solution to (3).

We can split the problem of solving (3) into two parts:

1. Solve the PDE on Ω_τ.
2. Solve the PDE on Ω_τ^{c}.

The DtN map allows us to “glue” the two solutions together.

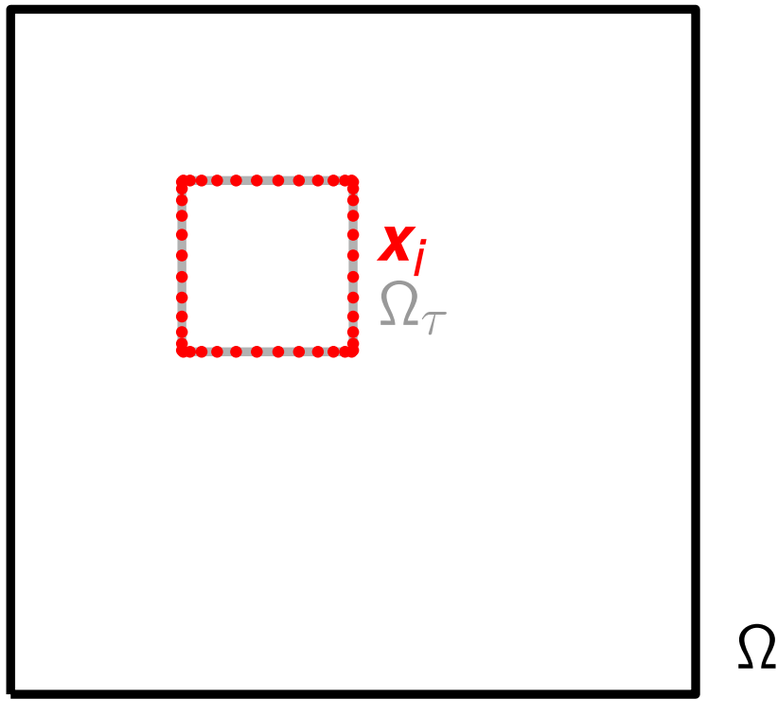
Consider a domain Ω on which we are given a PDE

$$(5) \qquad \begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega \\ u(\boldsymbol{x}) = f(\boldsymbol{x}) & \boldsymbol{x} \in \partial\Omega, \end{cases}$$

where $b = b(\boldsymbol{x})$ is a given function (smooth, non-negative).

Let T_τ be the DtN map for a subdomain $\Omega_\tau \subset \Omega$,

$$T_\tau : \; u|_{\partial\Omega_\tau} \mapsto u_n|_{\partial\Omega_\tau}, \qquad \text{subject to } -\Delta u + b\, u = 0 \text{ in } \Omega_\tau.$$



Representing functions on $\partial\Omega_\tau$ numerically:

- Place p *Legendre nodes* on each side of Ω_τ, to get points $\{\boldsymbol{x}_i\}_{i=1}^{4p}$.
- Represent a function w on $\partial\Omega_\tau$ via the vector $\mathbf{w} \in \mathbb{R}^{4p}$ of *tabulated values* $\mathbf{w}(i) = w(\boldsymbol{x}_i)$.

Representing the DtN map T_τ numerically:

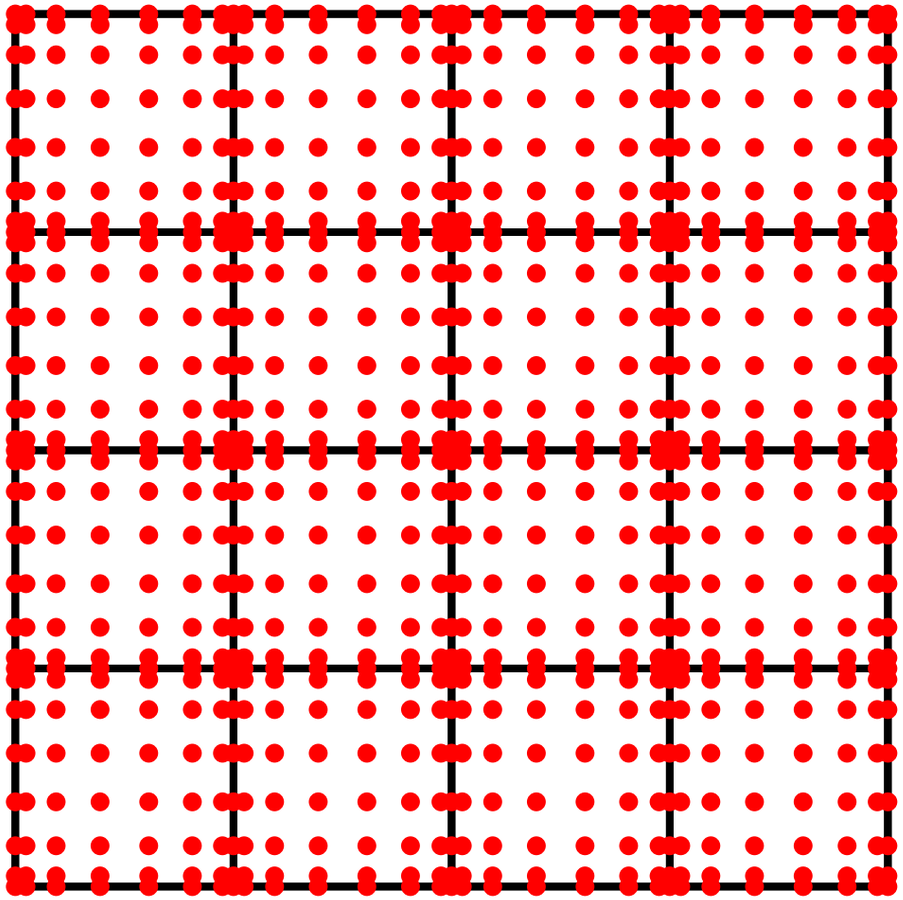
- Let $\mathbf{T}_\tau$ denote the $4p \times 4p$ matrix that takes a vector $\mathbf{u}_\tau$ of tabulated Dirichlet data on $\partial\Omega_\tau$ and maps it to the corresponding vector $\mathbf{v}_\tau$ of tabulated Neumann data.

Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Pre-process: Put down a spectral composite grid on Ω (Chebyshev nodes):



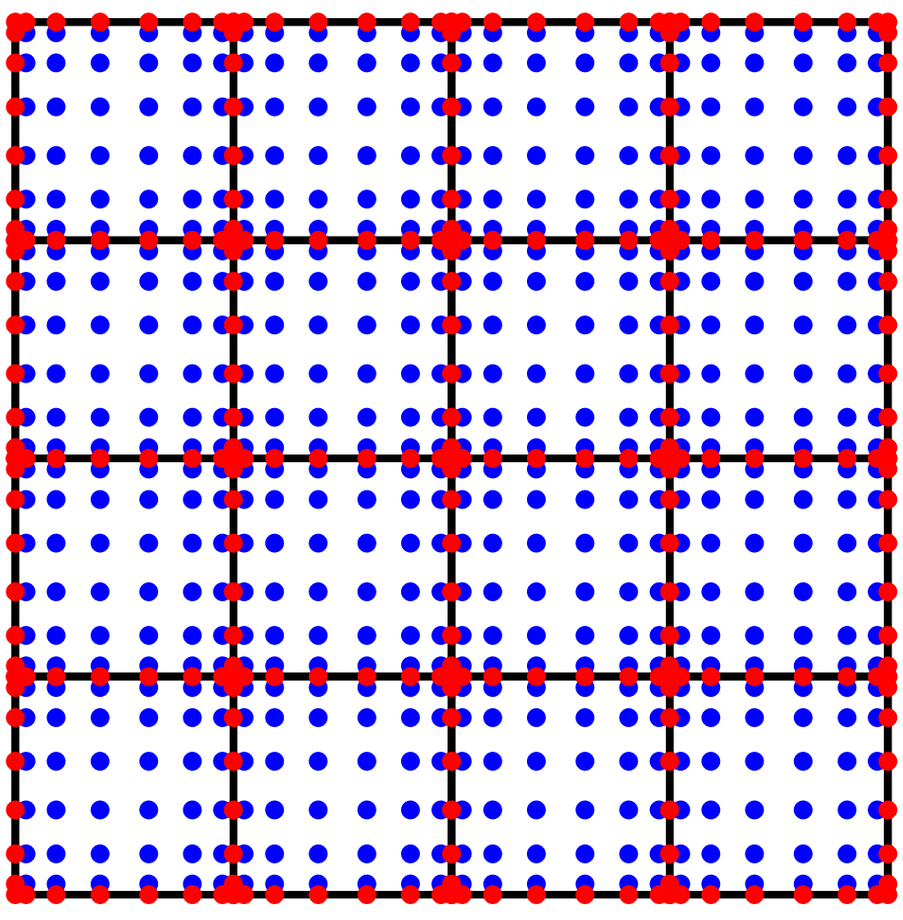
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Process leaves: Eliminate the interior (blue) nodes.

Technically, we compute the Dirichlet-to-Neumann operator via a local spectral computation.



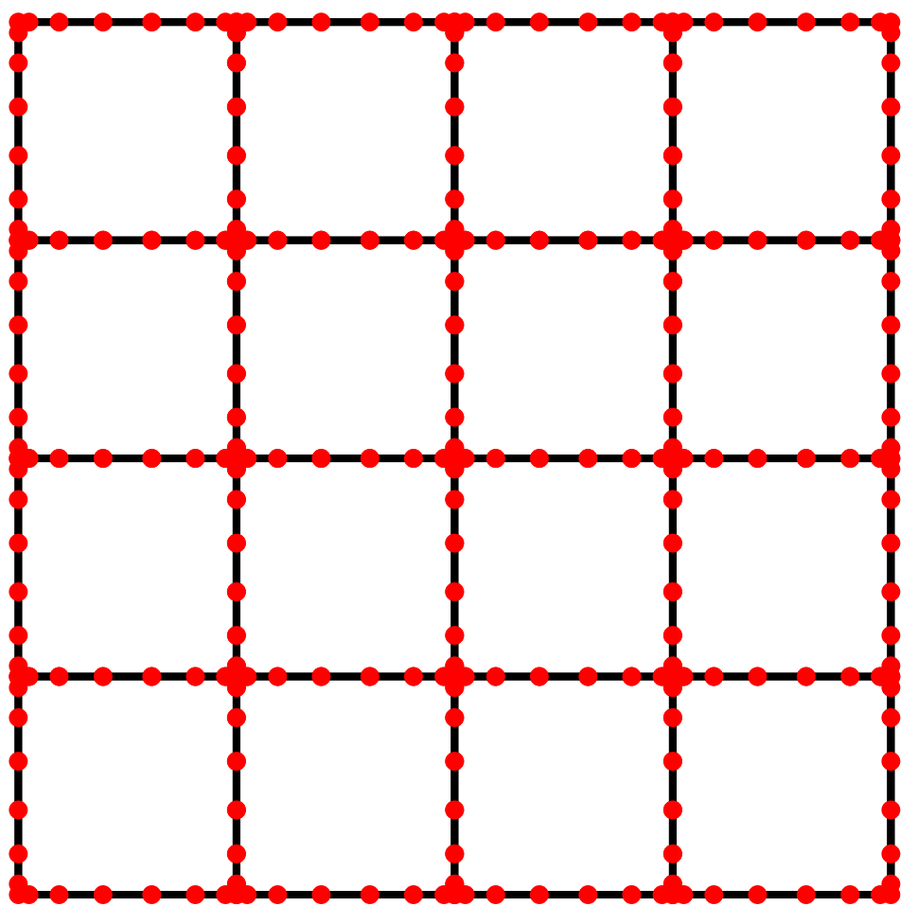
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Process leaves: Eliminate the interior (blue) nodes.

Technically, we compute the Dirichlet-to-Neumann operator via a local spectral computation.

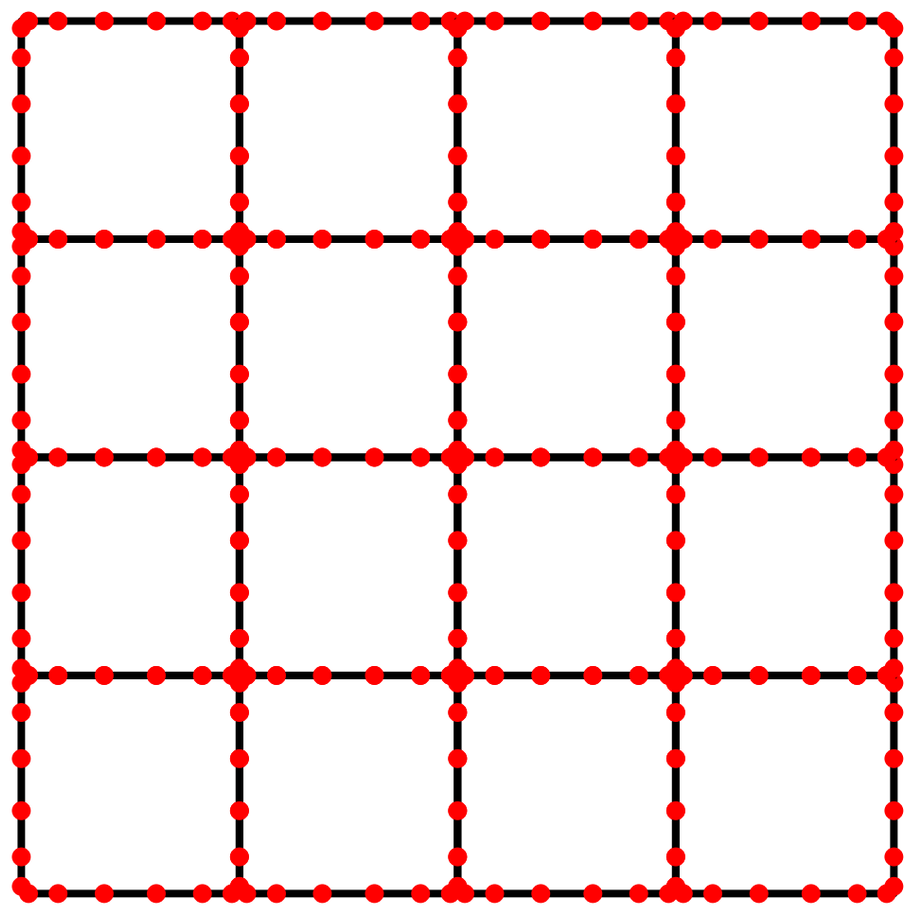


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Process leaves: Retabulate from Chebyshev to *Legendre nodes* on boundaries.



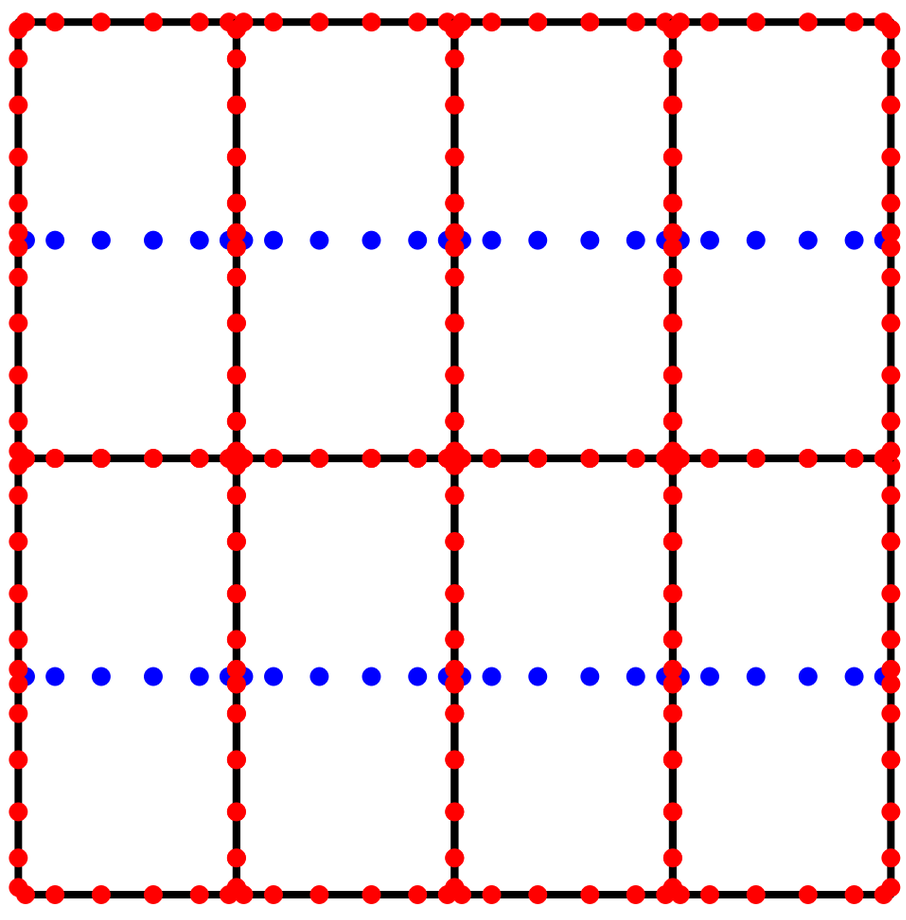
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



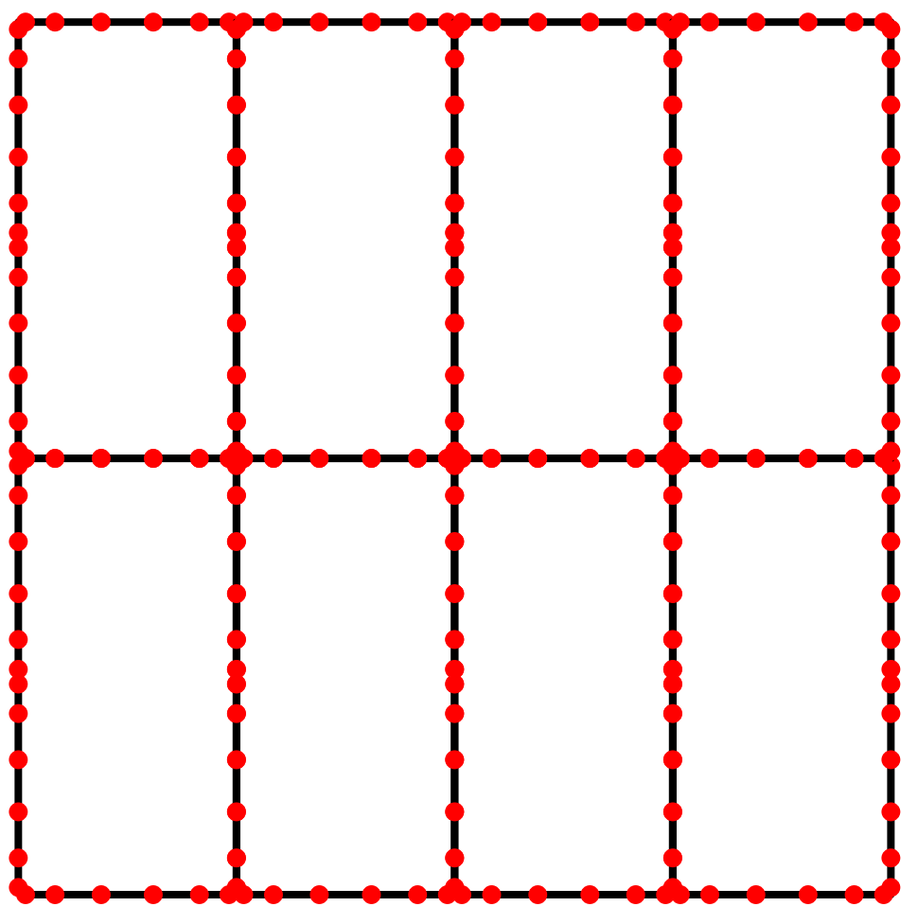
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



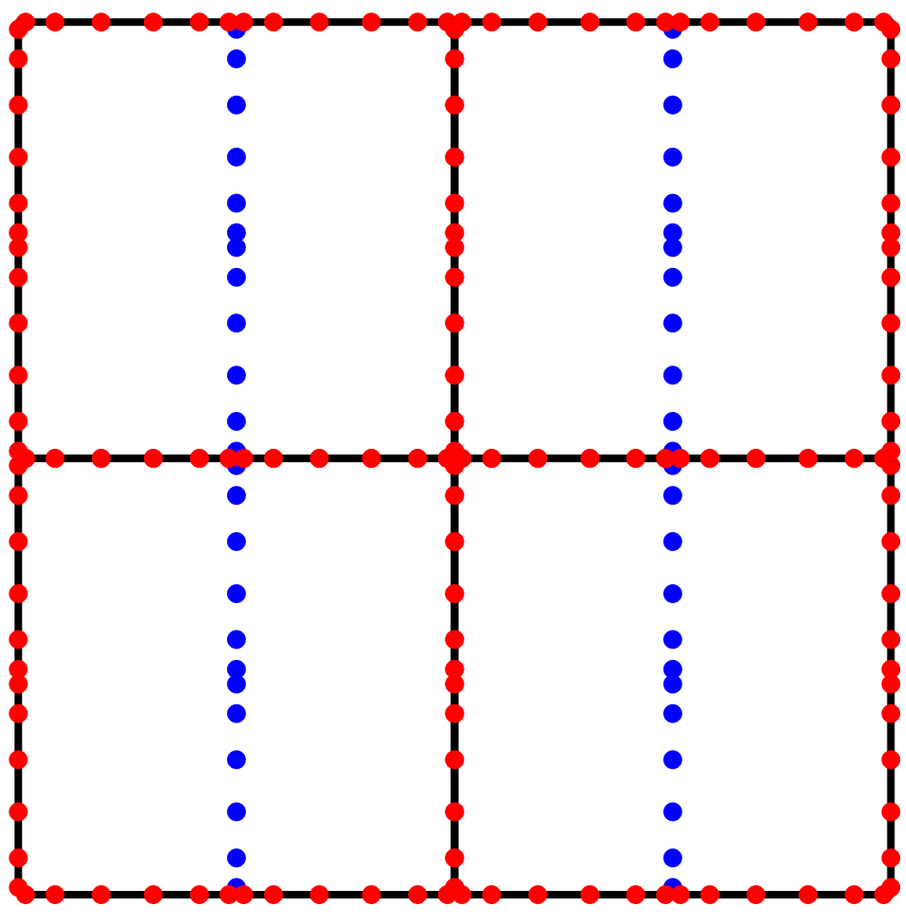
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



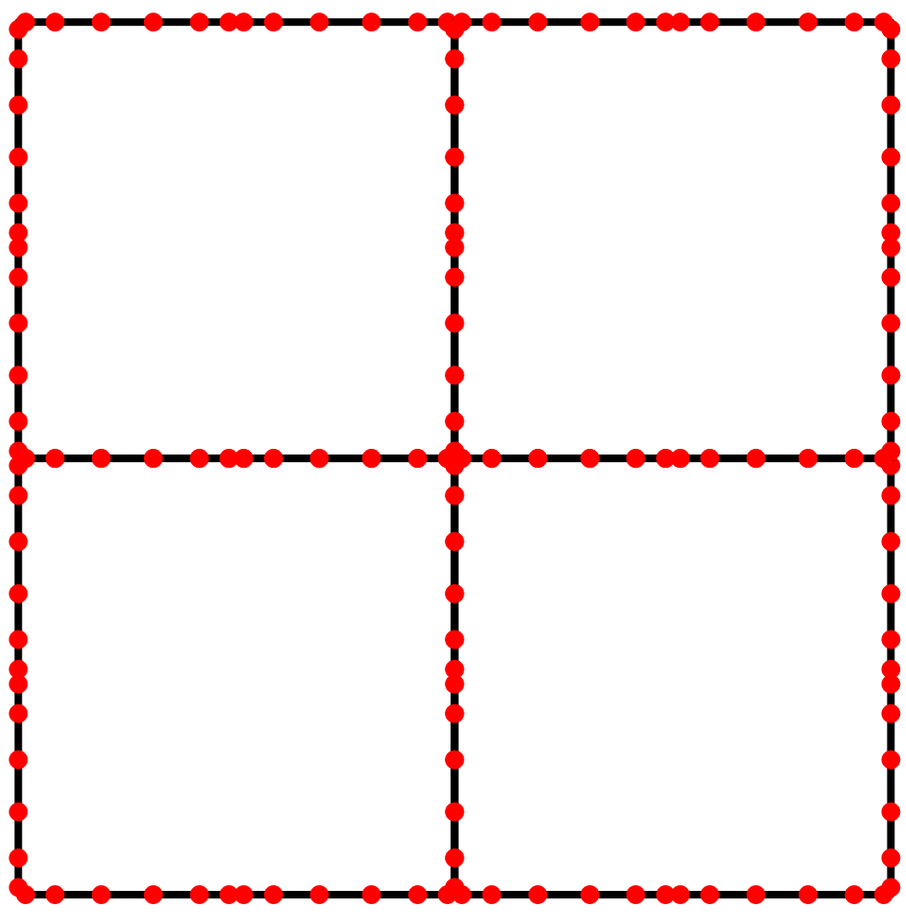
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



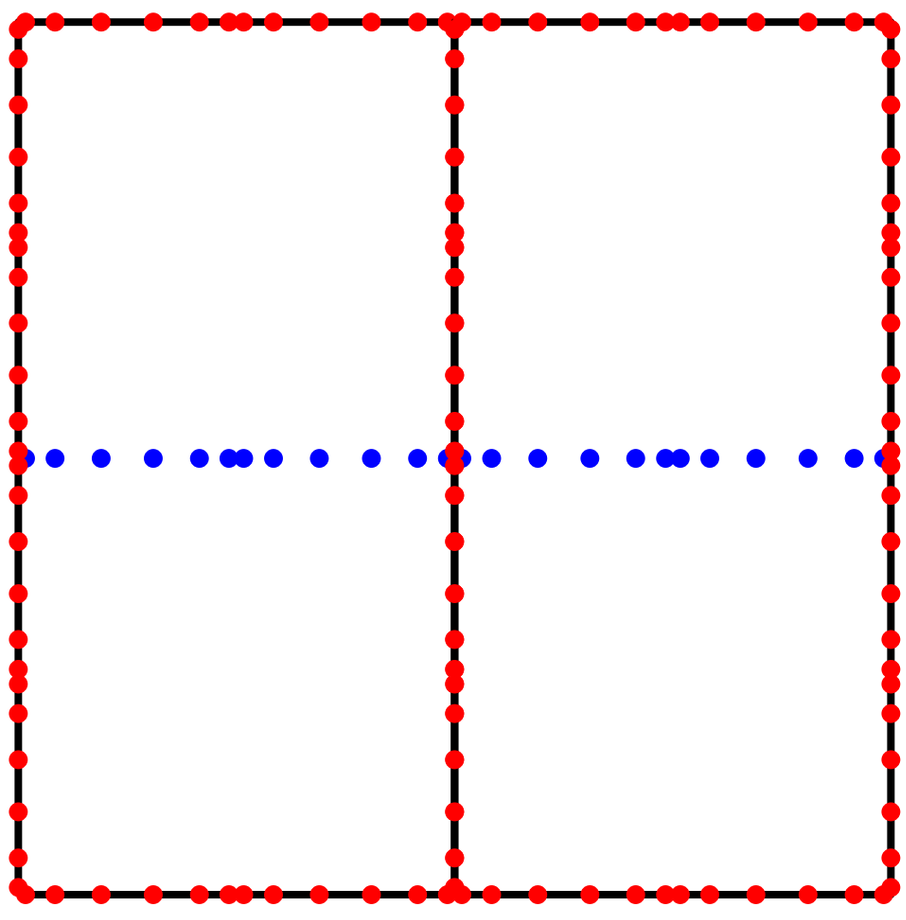
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



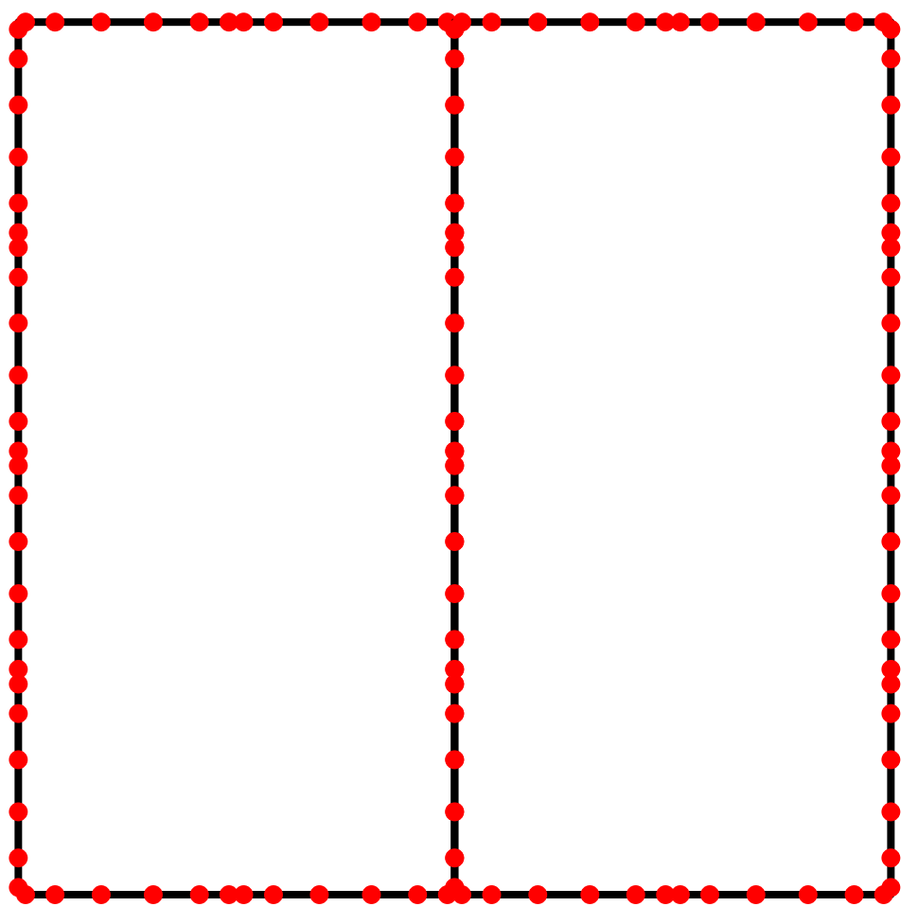
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



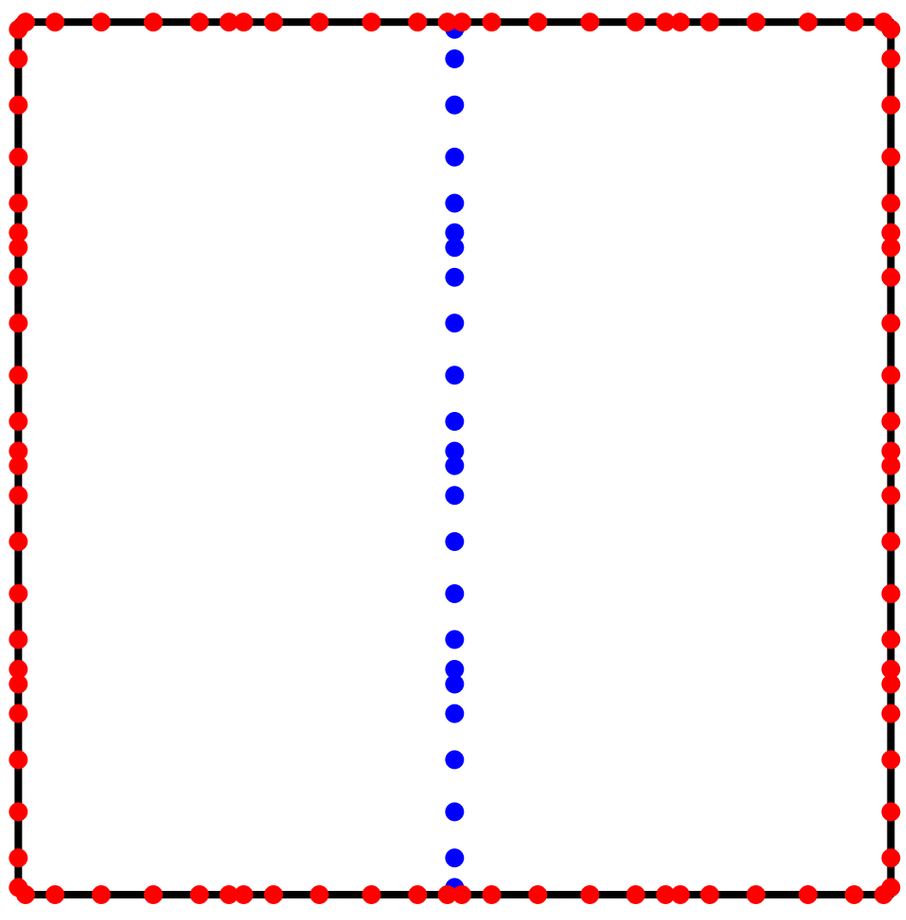
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.



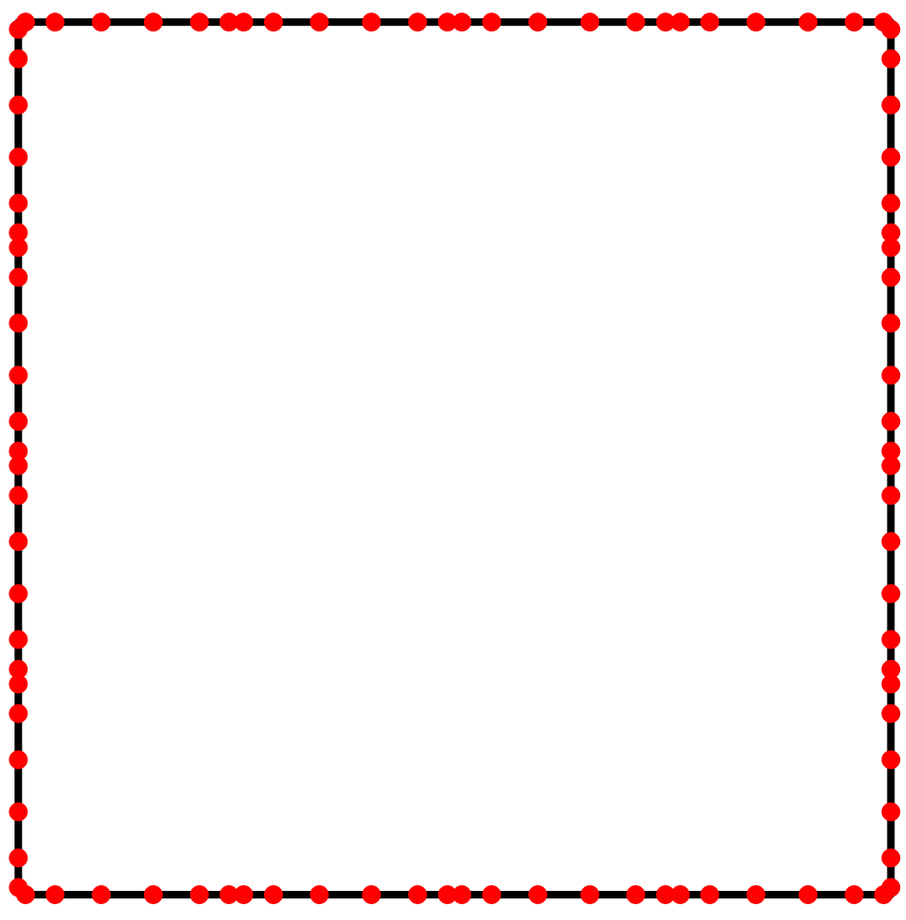
Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Upwards sweep: Merge boxes by pairs and eliminate the interior (blue) nodes.

To do this, use the computed DtN operators to enforce continuity of u and du/dn across interior boundaries. Compute the DtN operator for the larger box.

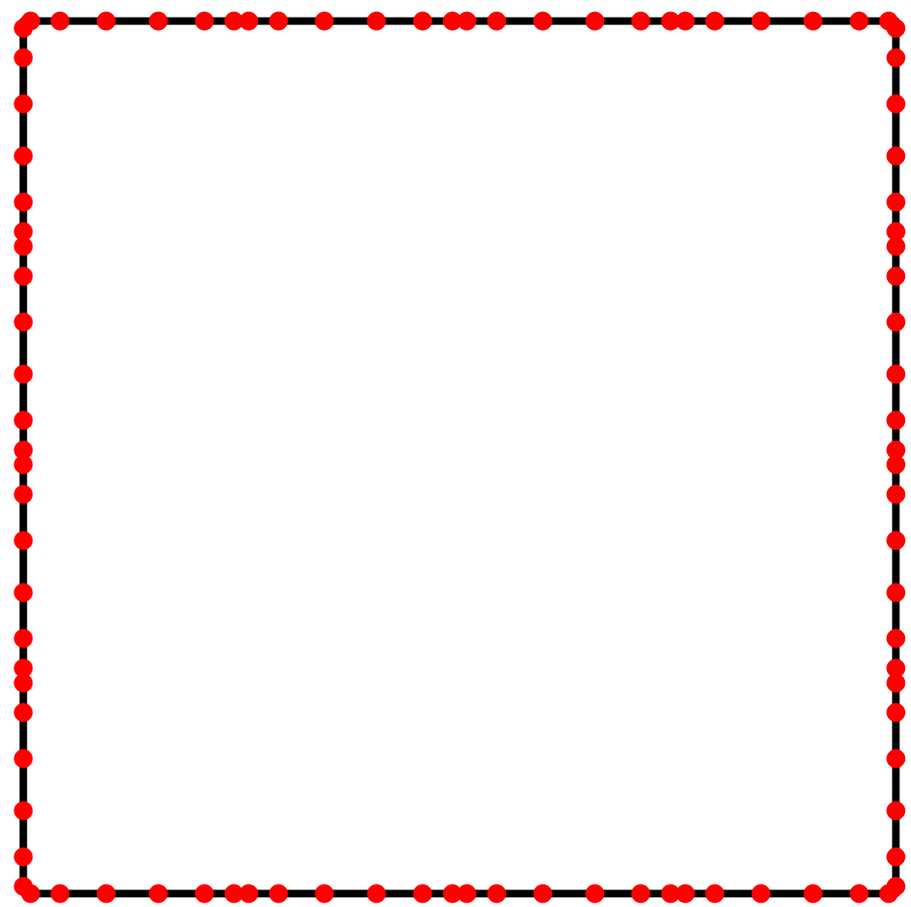


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Top level solve: Invert the DtN operator for the top level box.

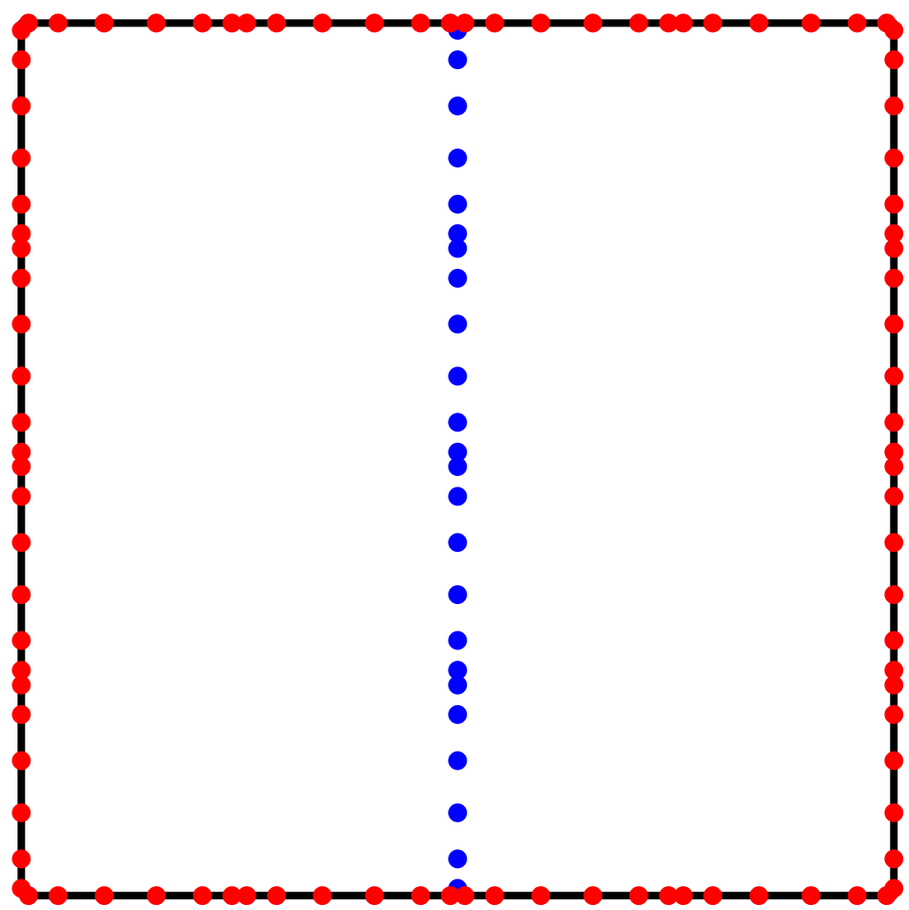


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Downwards sweep: We know u on the red nodes. We can use the computed DtN operators to reconstruct u on the blue nodes.

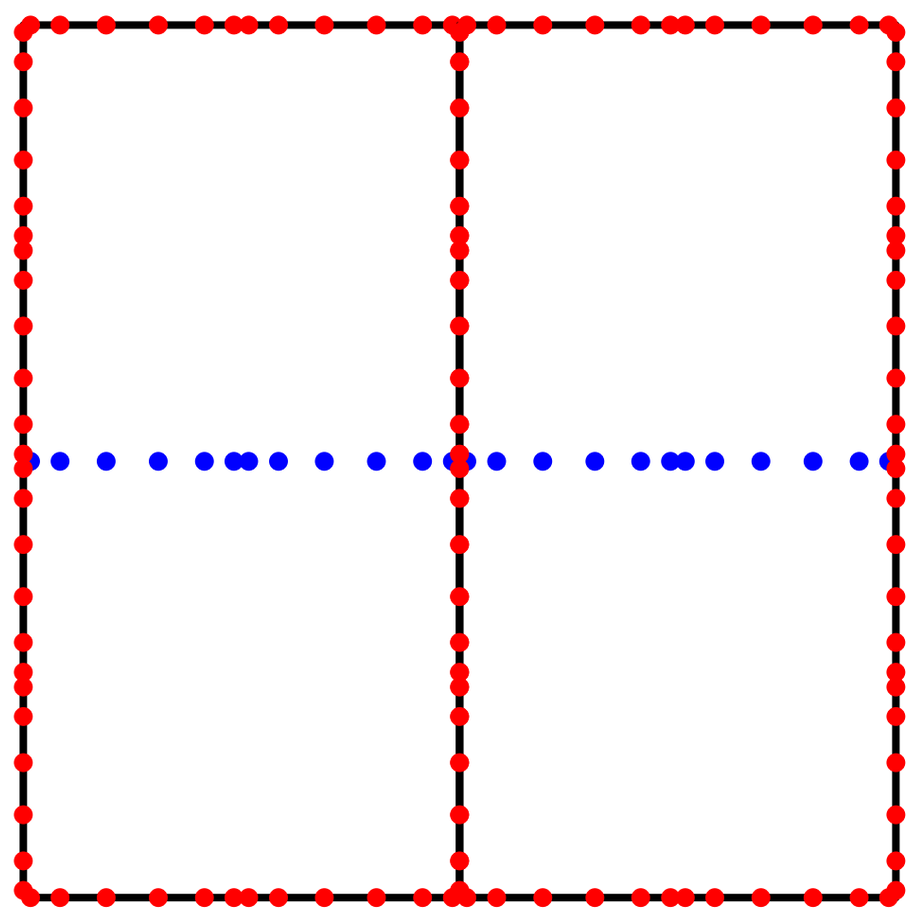


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Downwards sweep: We know u on the red nodes. We can use the computed DtN operators to reconstruct u on the blue nodes.

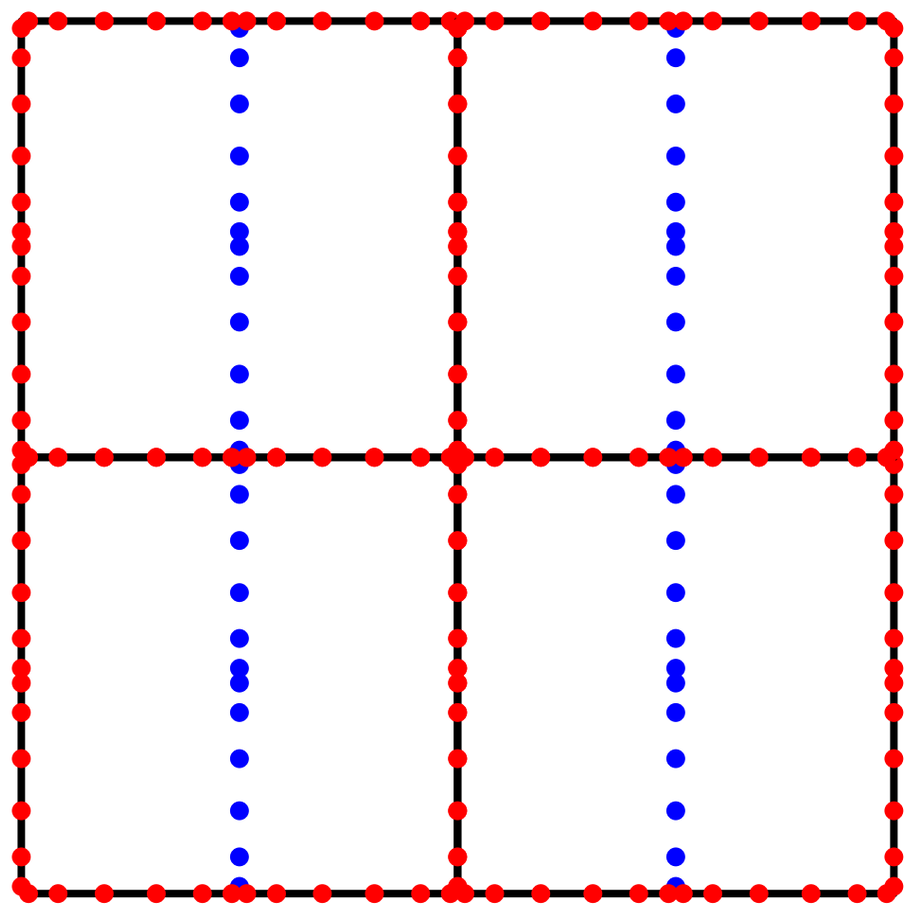


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Downwards sweep: We know u on the red nodes. We can use the computed DtN operators to reconstruct u on the blue nodes.

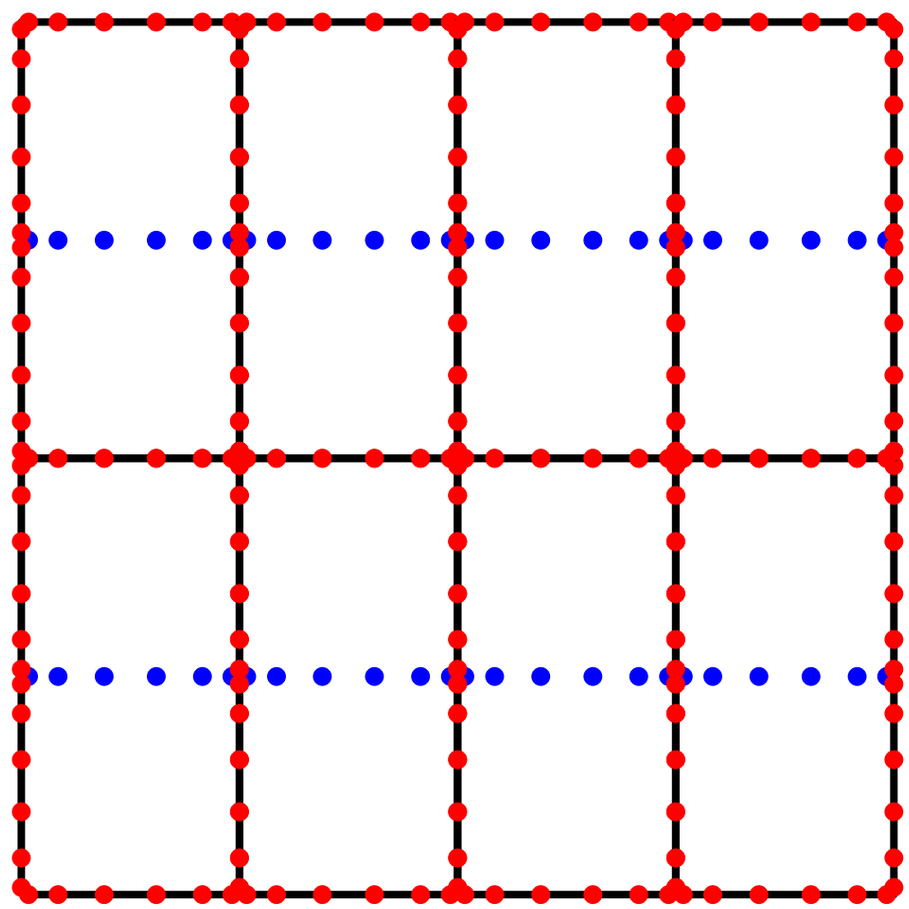


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0, 1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Downwards sweep: We know u on the red nodes. We can use the computed DtN operators to reconstruct u on the blue nodes.

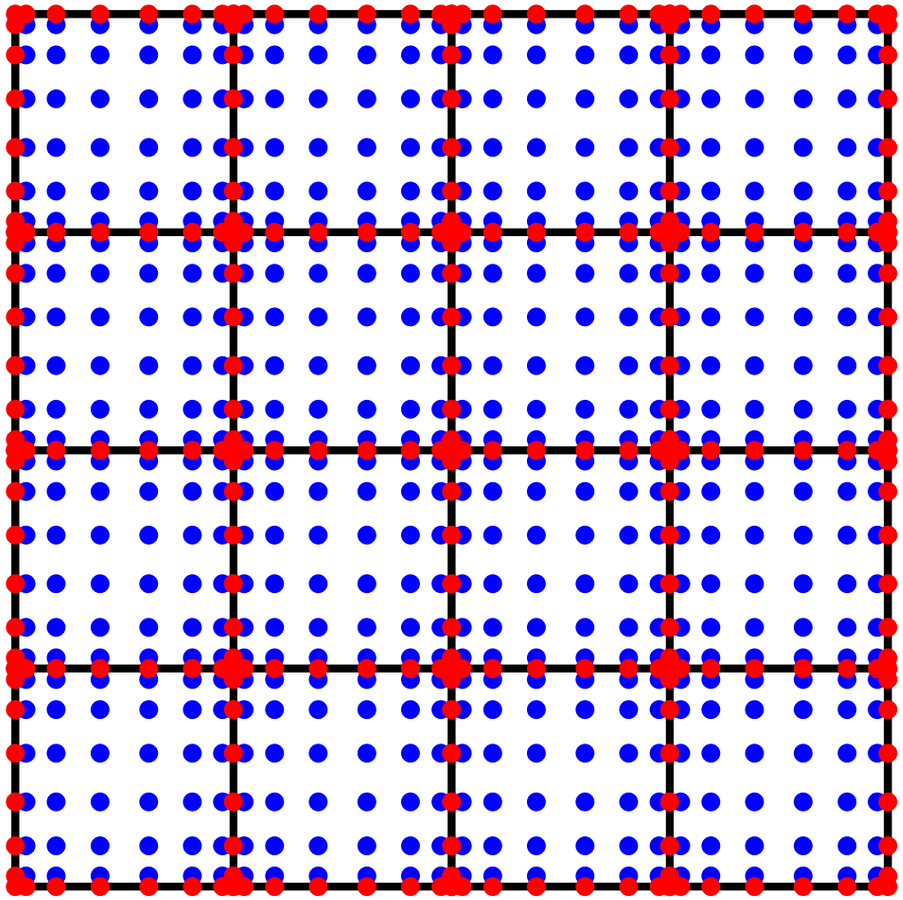


Model problem: Given f and b, find u such that

$$\begin{cases} -\Delta u(\boldsymbol{x}) + b(\boldsymbol{x})\, u(\boldsymbol{x}) = 0, & \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) = f(\boldsymbol{x}), & \boldsymbol{x} \in \Gamma, \end{cases}$$

where $\Omega = [0,1]^2$ is the unit square and $\Gamma = \partial\Omega$. We assume u is smooth.

Downwards sweep: We know u on the red nodes. We can use the computed DtN operators to reconstruct u on the blue nodes.



Notes:

- Adrianna Gillman's afternoon lecture will describe how to perform the "leaf computation", the "merge operation", and the "solve stage".
- Very high order methods can be implemented at minimal extra cost (compared to a low order scheme). Here, the "divider" has width 1, regardless of the order. In contrast, in classical nested dissection, the dividers get thicker, and the dense "frontal matrices" get larger, as the order increases.
- We used the Dirichlet-to-Neumann operator in this presentation, but analogous schemes can be constructed for other Poincaré-Steklov operators.
- Generalization to Helmholtz works very well, but the possibility of *resonances* requires some modifications.
- The scheme works conceptually the same in 3D.
- Acceleration to $O(N)$ complexity is analogous to what we described for nested dissection — exploit that DtN matrices are *rank structured.*
- The scheme can be used to rapidly construct and apply the time evolution operator associated with *parabolic and hyperbolic PDEs.*