AMRNodeElliptic Software Package
Node-Centered AMR for Elliptic Problems

P. McCorquodale

Applied Numerical Algorithms Group
NERSC Division
Lawrence Berkeley National Laboratory
Berkeley, CA

April 21, 2003

Disclaimer

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California, The views and opinion authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, or The Regents of the University of California.

Contents

Chapter 1

Nodal Discretization

1.1 AMR Spatial Discretization

For each level in the adaptive mesh refinement hierarchy, the underlying discretization of space is given as lattice points $(i_0, \ldots, i_{\mathbf{D}-1}) = \boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}}$, where $\boldsymbol{i}$ corresponds to a point of space $\boldsymbol{x}_0 + h \cdot \boldsymbol{i}$, with $\boldsymbol{x}_0 \in \mathbb{R}^{\mathbf{D}}$ a fixed origin of coordinates for all levels, and h the mesh spacing at the particular level.

The problem domain is discretized using node-centered grids. A grid $\Gamma \subset \mathbb{Z}^{\mathbf{D}}$ consists of the lattice points within a rectangular region bounded by two points $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{Z}^{\mathbf{D}}$, where $\boldsymbol{p} = (p_0, \ldots, p_{\mathbf{D}-1})$ is the lower corner of Γ and $\boldsymbol{q} = (q_0, \ldots, q_{\mathbf{D}-1})$ is the upper corner. Thus $\Gamma = \{\boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}} : \boldsymbol{p} \leq \boldsymbol{i} \leq \boldsymbol{q}\}$ where the notation $\boldsymbol{a} \leq \boldsymbol{b}$ means each entry of $\boldsymbol{a}$ is less than or equal to the corresponding entry of $\boldsymbol{b}$.

We will find it useful to define a number of operators on points and subsets of $\mathbb{Z}^{\mathbf{D}}$. We denote by $\Gamma + \boldsymbol{i}$ the translation of a set Γ by a point $\boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}}$. We define a coarsening operator $\mathcal{C}_r : \mathbb{Z}^{\mathbf{D}} \to \mathbb{Z}^{\mathbf{D}}$, where r is a positive integer, by

$$\mathcal{C}_r(\boldsymbol{i}) = (\left\lfloor \frac{i_0}{r} \right\rfloor, ..., \left\lfloor \frac{i_{\mathbf{D}-1}}{r} \right\rfloor).$$

The coarsening operator acting on subsets of $\mathbb{Z}^{\mathbf{D}}$ can be extended in a natural way to a node-centered grid: if $\Gamma = \{\boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}} : \boldsymbol{p} \leq \boldsymbol{i} \leq \boldsymbol{q}\}$ then $\mathcal{C}_r(\Gamma) = \{\boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}} : \mathcal{C}_r(\boldsymbol{p}) \leq \boldsymbol{i} \leq \mathcal{C}_r(\boldsymbol{q})\}$.

We extend this discretization of space to represent a nested hierarchy of grids that discretize the same continuous spatial domain. We assume that our problem domain can be discretized by a nested hierarchy of grids $\Gamma_N^0, \ldots, \Gamma_N^{l_{max}}$, with $\Gamma_N^{l+1} = \mathcal{C}_{n_{ref}^l}(\Gamma_N^l)$, where n_{ref}^l is the refinement ratio between levels l and $l + 1$.

AMR calculations are performed on a hierarchy of node-centered meshes $\Omega_N^l \subset \Gamma_N^l$, with $\Omega_N^l \supset \mathcal{C}_{n_{ref}^l}(\Omega_N^{l+1})$. Typically, Ω_N^l is decomposed into a union of node-centered grids, $\mathcal{R}(\Omega_N^l)$, such that any two distinct $\Gamma, \Gamma' \in \mathcal{R}(\Omega_N^l)$ intersect only on grid faces.

For a node-centered union of rectangles Ω_N at a fixed level of refinement, we define

the *interior* nodes $\Omega_{N,int}$ as those for which all neighbors along coordinate axes are also in Ω_N. That is, $\Omega_{N,int} = \Omega_N \cap \bigcap\limits_{d=0,\ldots,\mathbf{D}-1} ((\Omega_N + \boldsymbol{e}^d) \cap (\Omega_N - \boldsymbol{e}^d))$.

A discretized dependent variable in AMR is a *level array*

$$\varphi^l : \Omega_N^l \to \mathbb{R}^m.$$

We denote by $\varphi_{\boldsymbol{i}} \in \mathbb{R}^m$ the value of φ at node $\boldsymbol{i} \in \Omega_N^l$.

1.2   Incorporating Geometry

If the domain Ω is not rectangular, we need a subroutine that tells us, for a given grid, which of its nodes are in Ω. For the purposes of operator evaluation, the subroutine should also tell which nodes are reachable from their neighbors along lines parallel to the coordinate axes, where "reachable" in this context means that the connecting segment lies entirely in the domain.

Specifically, we have a Chombo Fortran subroutine

```
subroutine reachablenodes(CHF_CONST_REAL[dx], CHF_FIA[mask])
```

where dx is the mesh spacing and mask is a node-centered BaseFab<int> with $2\mathbf{D}+1$ components. At any node index $\boldsymbol{i}$ of the underlying Box, the corresponding point in space is $\boldsymbol{x} = \boldsymbol{x}_0 + (\boldsymbol{i} - \frac{1}{2}\boldsymbol{u}) \cdot \mathrm{dx}$, and the components of mask are set to represent reachability of the neighboring nodes, as follows (see Figure 1.1):

- component 0 is set to 1 if $\boldsymbol{x} \in \Omega$, and 0 otherwise.

- for each $d = 0, \ldots, \mathbf{D} - 1$, component $2d + 1$ is set to 1 if the entire segment $[\boldsymbol{x}, \boldsymbol{x} + \boldsymbol{e}^d \cdot \mathrm{dx}] \subset \Omega$, and 0 otherwise.

- for each $d = 0, \ldots, \mathbf{D} - 1$, component $2d + 2$ is set to 1 if the entire segment $[\boldsymbol{x}, \boldsymbol{x} - \boldsymbol{e}^d \cdot \mathrm{dx}] \subset \Omega$, and 0 otherwise.

The solver may call reachablenodes with any mesh spacing dx equal to a power of 2 multiplied by the finest mesh spacing in the hierarchy.

1.3   The Class NodeFArrayBox

The class NodeFArrayBox holds node-centered floating-point data for a single grid, but the constructor takes a cell-centered Box: NodeFArrayBox(bx, nComp) assigns storage for nComp floating-point components on each node surrounding the cells of bx.

This class is a wrapper for a node-centered FArrayBox. The purpose of introducing a new class with a constructor based on the cell-centered grid, instead of just using a node-centered FArrayBox, is that we can represent a level array $\phi : \Omega_N \to \mathbb{R}^m$ in the
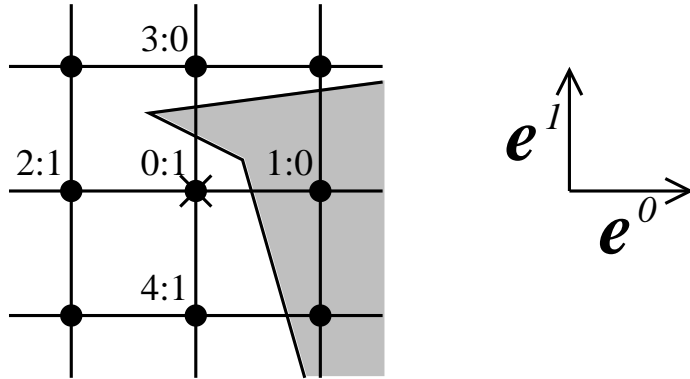
Figure 1.1: Components of the `mask` array at the center node indicated by $\times$, in two dimensions. The notation $i : v$ beside a node indicates that at node $\times$, component i has value v, as determined by the reachability of the labeled node from $\times$. The shaded area is outside the domain. We have $v = 1$ if the node is reachable from $\times$, and $v = 0$ if it is not. Note that the point at top center is considered to be *not* reachable from $\times$, even though it is in the domain, because the connecting segment does not lie entirely in the domain.

templated data holder `LevelData<NodeFArrayBox>`. This structure requires that the underlying `BoxLayout` be a `DisjointBoxLayout`. Since adjacent grids share nodes but not cells (see Figure 1.2), we must use the cell-centered grids in the constructor.

If a node belongs to more than one grid, then the values in the `LevelData<NodeFArrayBox>` at that node must be the same in each grid in which it occurs.
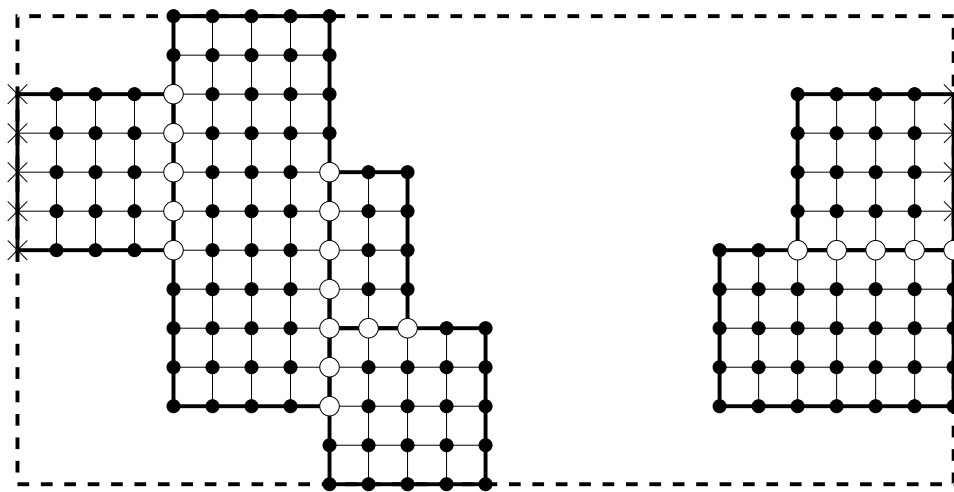
Figure 1.2: An illustration of a `DisjointBoxLayout` consisting of six grids within a problem domain delineated by dashed lines. Abutting grids share nodes indicated by hollow circles. Nodes marked $\times$ are also shared by abutting grids if the domain is periodic in the horizontal direction.

Chapter 2

Multilevel Laplacian Operator

2.1 No-Geometry Case

From a formal numerical analysis standpoint, a solution on an adaptive mesh hierarchy $\{\Omega_N^l\}_{l=0}^{l_{max}}$ approximates the exact solution to the PDE only on interior nodes, and only on those interior nodes that are not covered by interior nodes at a finer level. These are defined as the *valid* nodes of Ω_N^l (see Figure 2.1):

$$\Omega_{N,valid}^l = \Omega_{N,int}^l - \mathcal{C}_{n_{ref}^l}(\Omega_{N,int}^{l+1}).$$
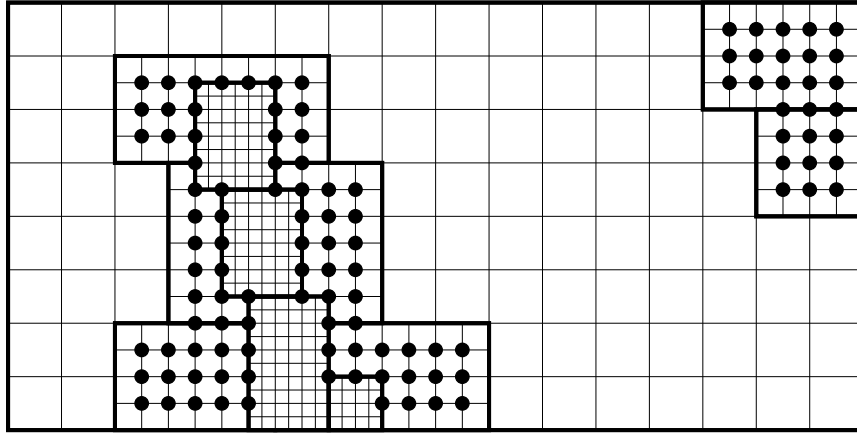
Figure 2.1: A grid hierarchy with three levels of refinement. The valid nodes at the middle level are indicated by •.

A composite array φ^{comp} is a collection of discrete values defined on the valid nodes at each of the levels of refinement.

$$\varphi^{comp} = \{\varphi^{l,valid}\}_{l=0}^{l_{max}} \ , \ \varphi^{l,valid} : \Omega_{N,valid}^l \to \mathbb{R}^m$$

The discrete Laplacian operator Δ^h on a level with mesh spacing h is defined on the valid nodes of that level, as

$$(\Delta^h \varphi^{N,l})_{\boldsymbol{i}} = \sum_{d=0}^{\mathbf{D}-1} (\Delta_d^h \varphi^{N,l})_{\boldsymbol{i}} \tag{2.1}$$

where Δ_d^h is a second-derivative operator in dimension d,

$$(\Delta_d^h \varphi^{N,l})_{\boldsymbol{i}} = \frac{\varphi_{\boldsymbol{i}-\boldsymbol{e}^d}^{N,l} - 2\varphi_{\boldsymbol{i}}^{N,l} + \varphi_{\boldsymbol{i}+\boldsymbol{e}^d}^{N,l}}{h^2}. \tag{2.2}$$

Then (2.1)–(2.2) is a second-order approximation to the exact Laplacian Δ:

$$(\Delta^h \varphi^{N,l})_{\boldsymbol{i}} = (\Delta \varphi^{N,l})_{\boldsymbol{i}} + O(h^2).$$
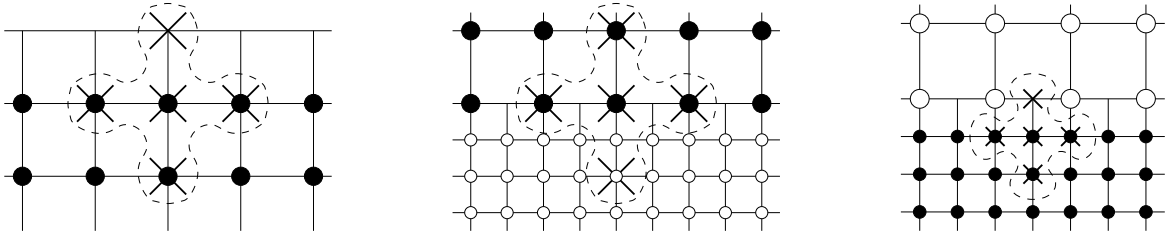
Figure 2.2: Cases in which data at non-valid nodes is needed in computing the discrete Laplacian at a valid node. Nodes of the stencil are indicated by $\times$.
Left: Stencil reaches physical boundary (at top point) — then use the physical boundary conditions.
Center: Stencil is partially covered by the next finer level (at bottom point) — then project valid data from the finer level.
Right: Stencil intersects boundary with coarser level (at top point) — then interpolate from coarse nodes along the boundary.

There are three cases in which $\boldsymbol{i}$ is a valid node at level l, but $\boldsymbol{j} = \boldsymbol{i} \pm \boldsymbol{e}^d$ is not (see Figure 2.2).

- The node $\boldsymbol{j}$ is on the physical boundary. Then we take the value of $\varphi_{\boldsymbol{j}}^{N,l}$ from the physical boundary conditions.

- The node $\boldsymbol{j}$ is covered by a grid at finer level $l+1$. Then we project the valid data from the finer level, $\varphi_{\boldsymbol{j}}^{N,l} = \varphi_{n_{ref}^l \boldsymbol{j}}^{N,l+1}$.

- The node $\boldsymbol{j}$ is on the boundary with coarser level $l-1$. Then we interpolate from the coarser level, as described in chapter 4.

2.2 Operators on Non-Rectangular Domains

When the domain Ω is not rectangular, the stencil of the discrete Laplacian operator at a valid node may include points that are not in the domain. Then we need a subroutine to compute the discrete Laplacian Δ^h on a level with mesh spacing h.

The coefficients are computed in a Chombo Fortran subroutine

```
subroutine nodalcoefficients(CHF_CONST_REAL[dx], CHF_FRA[coeffs])
```

where dx is the mesh spacing and coeffs is a node-centered FArrayBox with $2\mathbf{D} + 2$ components. At any node index $\boldsymbol{i}$ of the underlying Box, the corresponding point in space is $\boldsymbol{x} = \boldsymbol{x}_0 + (\boldsymbol{i} - \frac{1}{2}\boldsymbol{u}) \cdot \mathrm{dx}$, and the components of coeffs are set to the coefficients of the stencil for node $\boldsymbol{i}$, as follows:

- component 0 is the coefficient of the value at $\boldsymbol{x}$.

- for each $d = 0, \ldots, \mathbf{D} - 1$, component $2d + 1$ is the coefficient of the value at $\boldsymbol{x} + \boldsymbol{e}^d \cdot \mathrm{dx}$.

- for each $d = 0, \ldots, \mathbf{D} - 1$, component $2d + 2$ is the coefficient of the value at $\boldsymbol{x} - \boldsymbol{e}^d \cdot \mathrm{dx}$.

- component $2\mathbf{D} + 1$ is the constant coefficient.

Thus if $c_{\boldsymbol{i}}^0, \ldots c_{\boldsymbol{i}}^{2\mathbf{D}+1}$ are the coefficients for node $\boldsymbol{i}$ at a level with mesh spacing h, then

$$(\Delta^h \varphi^{N,l})_{\boldsymbol{i}} = c_{\boldsymbol{i}}^0 \cdot \varphi_{\boldsymbol{i}}^{N,l} + \sum_{d=0,\ldots,\mathbf{D}-1} \left(c_{\boldsymbol{i}}^{2d+1} \cdot \varphi_{\boldsymbol{i}+\boldsymbol{e}^d}^{N,l} + c_{\boldsymbol{i}}^{2d+2} \cdot \varphi_{\boldsymbol{i}-\boldsymbol{e}^d}^{N,l} \right) + c_{\boldsymbol{i}}^{2\mathbf{D}+1}.$$

If all of the nodes of the stencil for $\boldsymbol{i}$ are in the domain, then the coefficients are $c_{\boldsymbol{i}}^0 = -2/h^2$; $c_{\boldsymbol{i}}^{2d+1} = c_{\boldsymbol{i}}^{2d+2} = 1/h^2$ for $d = 0, \ldots, \mathbf{D} - 1$; and $c_{\boldsymbol{i}}^{2\mathbf{D}+1} = 0$.

At points near an internal boundary, where not all points of the regular stencil lie in the domain, we make an approximation for $(\Delta_d^h \varphi^{N,l})_{\boldsymbol{i}}$ $(d = 0, \ldots, \mathbf{D}-1)$ that is based on the regular 3-point stencil (2.2) and extrapolating from available data to covered points of the stencil. For a point of evaluation $\boldsymbol{x}$, define $\boldsymbol{x}_L$ and $\boldsymbol{x}_R$ as follows.

- If $\boldsymbol{x} - \boldsymbol{e}^d \in \Omega$, then $\boldsymbol{x}_L = \boldsymbol{x} - \boldsymbol{e}^d$. Otherwise, $\boldsymbol{x}_L$ is the first point on the segment from $\boldsymbol{x}$ to $\boldsymbol{x} - \boldsymbol{e}^d$ that lies on the boundary of Ω.

- If $\boldsymbol{x} + \boldsymbol{e}^d \in \Omega$, then $\boldsymbol{x}_R = \boldsymbol{x} + \boldsymbol{e}^d$. Otherwise, $\boldsymbol{x}_R$ is the first point on the segment from $\boldsymbol{x}$ to $\boldsymbol{x} + \boldsymbol{e}^d$ that lies on the boundary of Ω.

We also define φ_L as the value of φ at $\boldsymbol{x}_L$ from either $\varphi_{\boldsymbol{i}-\boldsymbol{e}^d}^{N,l}$ or the boundary condition, and similarly φ_R as the value of φ at $\boldsymbol{x}_R$ from either $\varphi_{\boldsymbol{i}+\boldsymbol{e}^d}^{N,l}$ or the boundary condition.
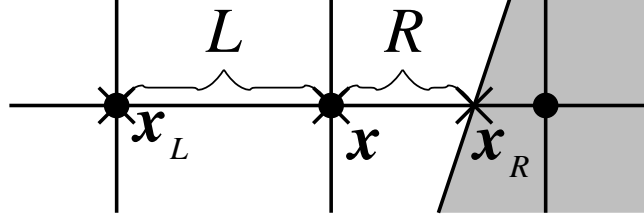
Figure 2.3: Points (marked by ×) and distances used in approximating near an internal boundary, when the shaded area is outside the domain. In this example, $\varphi_L = \varphi(\boldsymbol{x}_L) = \varphi_{\boldsymbol{i}-\boldsymbol{e}^d}^{N,l}$, and $\varphi_R = \varphi(\boldsymbol{x}_R)$ is obtained from the Dirichlet boundary condition.

The Shortley–Weller approximation is based on quadratic extrapolation from $\varphi_{\boldsymbol{i}}^{N,l}$, φ_L and φ_R to the covered grid nodes of the regular stencil. Thus

$$(\Delta_d^h \varphi^{N,l})_{\boldsymbol{i}} = \frac{2}{L \cdot (L+R)}\varphi_L - \frac{2}{L \cdot R}\varphi_{\boldsymbol{i}}^{N,l} + \frac{2}{R \cdot (L+R)}\varphi_R. \qquad (2.3)$$

Formula (2.3) provides a first-order approximation to the exact second partial derivative in dimension d, and hence to the exact Laplacian Δ:

$$(\Delta^h \varphi^{N,l})_{\boldsymbol{i}} = (\Delta \varphi^{N,l})_{\boldsymbol{i}} + O(h).$$

An alternative approximation is based on linear extrapolation to the covered grid nodes of the regular stencil. Thus

$$(\Delta_d^h \varphi^{N,l})_{\boldsymbol{i}} = \frac{1}{h \cdot L}\varphi_L - \frac{1}{h}(\frac{1}{L} + \frac{1}{R})\varphi_{\boldsymbol{i}}^{N,l} + \frac{1}{h \cdot R}\varphi_R. \qquad (2.4)$$

Formula (2.4) provides a zeroth-order approximation to the exact second partial derivative in dimension d, and hence to the exact Laplacian Δ:

$$(\Delta^h \varphi^{N,l})_{\boldsymbol{i}} = (\Delta \varphi^{N,l})_{\boldsymbol{i}} + O(1).$$

11

Chapter 3

Interlevel Transfer Operators

3.1 Averaging

This operator is used to average from a finer level down to a coarser level, or to construct averaged residuals in multigrid iterations. Averaging from level l to level $l-1$, we assume that the refinement ratio $r = n_{ref}^{l-1}$ is a power of 2. To obtain the averaged value at node $i \in \mathcal{C}_r(\Omega_{N,int}^l)$ of level $l-1$, we use the values at the level-l nodes $ri + q$, where the components of q satisfy $-r/2 \le q_d \le r/2$ for $d = 0, \ldots, \mathbf{D}-1$.

3.1.1 No-Geometry Case

For a refinement ratio of r, we define weights w_q on the points in

$$\Theta_r = \{q \in \mathbb{Z}^{\mathbf{D}} : -\frac{r}{2}u \le q \le \frac{r}{2}u\}.$$

The averaging operator down to level $l-1$ is defined at nodes $i \in \Omega_N^{l-1}$ where $\Theta_r + ri \subset \Omega_N^l$. These are the interior coarse nodes of level l. Thus for $i \in \mathcal{C}_r(\Omega_{N,int}^l)$:

$$Average(\varphi^{N,l})_i = \sum_{q \in \Theta_r} w_q \varphi_{ri+q}^{N,l} \tag{3.1}$$

where the weight at point $q = (q_0, \ldots, q_{\mathbf{D}-1}) \in \mathbb{Z}^{\mathbf{D}}$ is

$$w_q = \prod_{d=0}^{\mathbf{D}-1} \frac{1}{(1 + \delta_{|q_d|, r/2})r} \tag{3.2}$$

with δ the Kronecker delta. In words, the weights of the nodes come from the tensor product of the weights in the one-dimensional trapezoidal rule. In the one-dimensional trapezoidal rule, when there are $r+1$ points, the $r-1$ interior points are each assigned the weight $\frac{1}{r}$, and the two extreme points are each assigned the weight $\frac{1}{2r}$.
    See Figure 3.1 for two-dimensional examples.
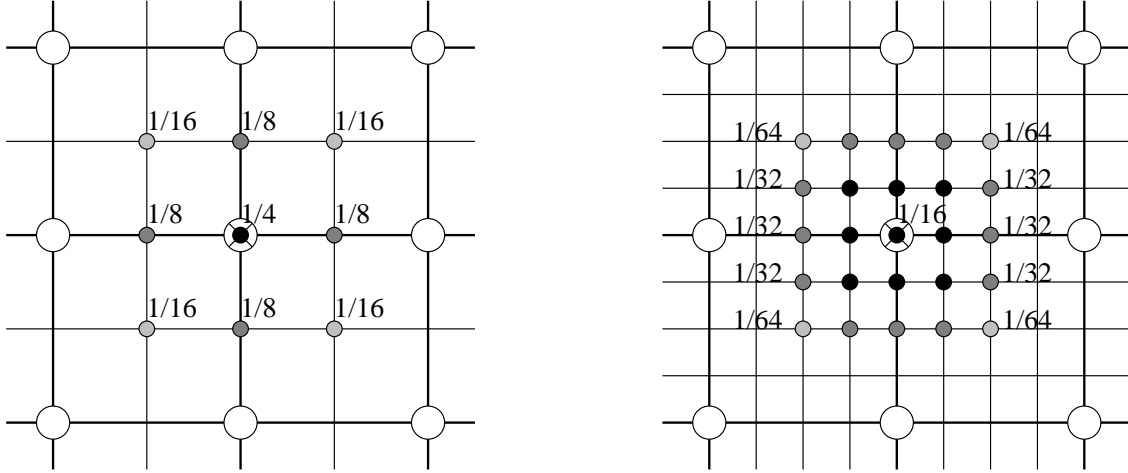
Figure 3.1: Averaging down to the center coarse node indicated by $\times$, in two dimensions. Weights are indicated at the fine nodes, with different shadings for different weights.
Left: Refinement ratio is 2. Each corner node has weight $\frac{1}{16}$, each other node on an exterior face has weight $\frac{1}{8}$, and the center node has weight $\frac{1}{4}$.
Right: Refinement ratio is 4. Each corner node has weight $\frac{1}{64}$, each other node on an exterior face has weight $\frac{1}{32}$, and each remaining node, in the interior, has weight $\frac{1}{16}$.

3.1.2   Case of Non-Rectangular Domains

It is possible that a non-rectangular domain might not contain some of the points needed in averaging by the method of section 3.1.1. Then the weights will need to be adjusted so that only points in the domain are given nonzero weights.

To simplify the procedure for non-rectangular domains, averaging is performed as a composition of averagings with refinement ratio of 2. In the remainder of this section, we assume that the refinement ratio is 2.

We have a separate weight array for each coarse node. The adjustment we make to the weights in (3.2) is that if a fine node is not reachable from the coarse node by a path lying entirely in the domain, consisting of edges parallel to the axes, without backtracking, then the weight of that fine node is set to zero. The weight of the projected coarse node on the finer level is then increased so that the total of the fine–node weights at the coarse node is 1. Thus

$$Average(\varphi^{N,l})_{\boldsymbol{i}} = \sum_{\boldsymbol{q}\in\Theta_2} w_{\boldsymbol{i},\boldsymbol{q}}\varphi^{N,l}_{2\boldsymbol{i}+\boldsymbol{q}} \tag{3.3}$$

where for $\boldsymbol{0} \neq \boldsymbol{q} \in \Theta_2$, we define the weight

$$w_{\boldsymbol{i},\boldsymbol{q}} = \begin{cases} \prod_{d=0}^{\mathbf{D}-1} \frac{1}{1+|q_d|} & \text{if } \boldsymbol{q} + 2\boldsymbol{i} \text{ is reachable from } 2\boldsymbol{i} \\ 0 & \text{otherwise} \end{cases}$$

and

$$w_{\boldsymbol{i},\boldsymbol{0}} = 1 - \sum_{\boldsymbol{0} \neq \boldsymbol{q} \in \Theta_2} w_{\boldsymbol{i},\boldsymbol{q}}.$$

As stated above, "reachable" here means that there is a path through the fine nodes that avoids backtracking, where each edge in the path is parallel to one of the axes and lies entirely in the domain. For each edge, this code calls the `reachablenodes` subroutine described in section 1.2. The two-dimensional domain subset used in Figure 1.1 is shown in Figure 3.2 with the weights on the fine nodes.
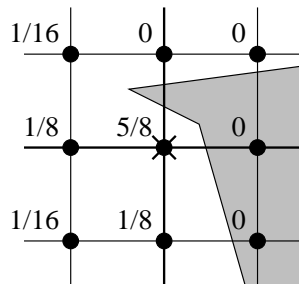
Figure 3.2: Weights of fine nodes in averaging to the coarse node indicated by $\times$. The shaded area is outside the domain. Some nodes have weight 0, even though they are in the domain, because there is no path to them from $\times$ following edges that lie entirely in the domain, without backtracking. Compare with Figure 3.1, left side.

3.2   Interpolation

This operator is primarily used in multigrid iteration to interpolate the correction from the coarser level to the next finer level. We interpolate from the neighboring coarse nodes, using a formula that is bilinear in two dimensions, trilinear in three dimensions.

In this section, we assume we are interpolating from level $l - 1$ to level l, where the refinement ratio is $r = n_{ref}^{l-1}$.

3.2.1   No-Geometry Case

The interpolated value at $\boldsymbol{i} \in \Omega^{N,l}$ is a linear combination of the values at the $2^{\mathbf{D}}$ coarse nodes in $\Omega^{N,l-1}$ that are the vertices of the box with corners at $\mathcal{C}_r(\boldsymbol{i}) = (\lfloor i_0/r \rfloor, \ldots, \lfloor i_{\mathbf{D}-1}/r \rfloor)$ and $\mathcal{C}_r(\boldsymbol{i}) + \boldsymbol{u}$. The length fractions for each dimension $d = 0, \ldots, \mathbf{D} - 1$ are

$$\varepsilon_{\boldsymbol{i}}^d(0) = 1 - \frac{i_d - r \lfloor i_d/r \rfloor}{r}$$

$$\varepsilon_{\boldsymbol{i}}^d(1) = \frac{i_d - r \lfloor i_d/r \rfloor}{r}$$

14

and these are multiplied to give the weights of the coarse nodes, offset from $\mathcal{C}_r(\boldsymbol{i})$ by $\boldsymbol{0} \leq \boldsymbol{q} \leq \boldsymbol{u}$:

$$w_{\boldsymbol{i},\boldsymbol{q}} = \prod_{d=0}^{\mathbf{D}-1} \varepsilon_{\boldsymbol{i}}^d(q_d).$$

Then the interpolation formula is

$$Interp(\varphi^{N,l-1})_{\boldsymbol{i}} = \sum_{\boldsymbol{0} \leq \boldsymbol{q} \leq \boldsymbol{u}} w_{\boldsymbol{i},\boldsymbol{q}} \cdot \varphi_{\mathcal{C}_r(\boldsymbol{i})+\boldsymbol{q}}^{N,l}.$$

See Figures 3.3 and 3.4 for illustrations in two dimensions.
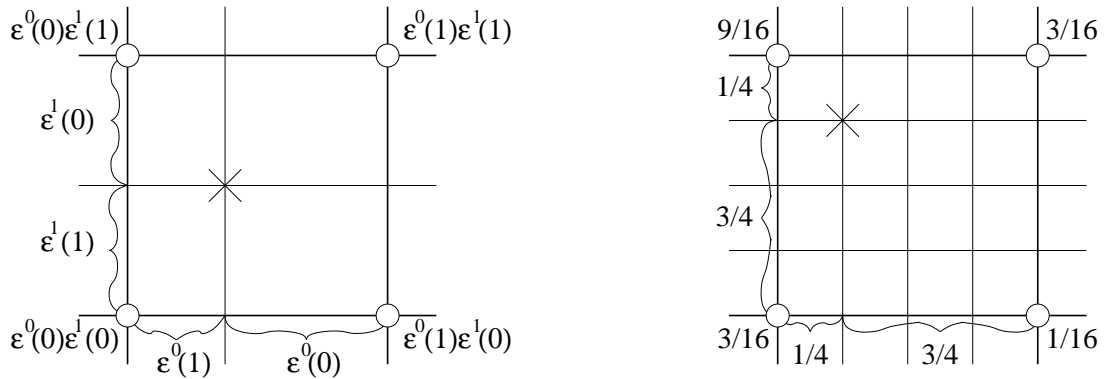
Figure 3.3: Left: Length fractions and weights of coarse nodes in bilinear interpolation in two dimensions to the fine node indicated by $\times$.
Right: An example with refinement ratio of 4.
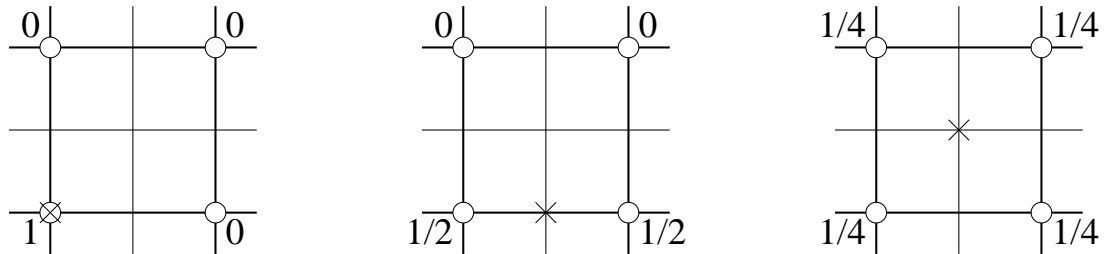
Figure 3.4: Interpolation in two dimensions to a fine node indicated by $\times$, when the refinement ratio is 2. Weights are shown on the coarse nodes in three cases.

3.2.2   Case of Non-Rectangular Domains

When the domain is not rectangular, some of the points needed in interpolating by the method of section 3.2.1 may be outside the domain. As we did with averaging in section

3.1.2, we assume that the refinement ratio is a power of 2, and to simplify the procedure, we perform interpolation as a composition of interpolations with refinement ratio of 2. In the remainder of this section, we assume a refinement ratio of 2.

The value at i is interpolated from coarse nodes in the box with corners $\lfloor i/2 \rfloor$ and $\lceil i/2 \rceil = (\lceil i_0/2 \rceil, \ldots, \lceil i_{\mathbf{D}-1}/2 \rceil)$. The number of such coarse nodes is 2 to the power of the number of odd components of i. We assign *equal* weight to each of these coarse nodes that is *reachable* from the fine node in the sense described in section 3.1.2: that is, connected by a path through fine nodes along edges parallel to the axes, with no backtracking. See Figure 3.5 for weights used in interpolating to some of the fine nodes in the two-dimensional domain subset used in Figure 1.1.
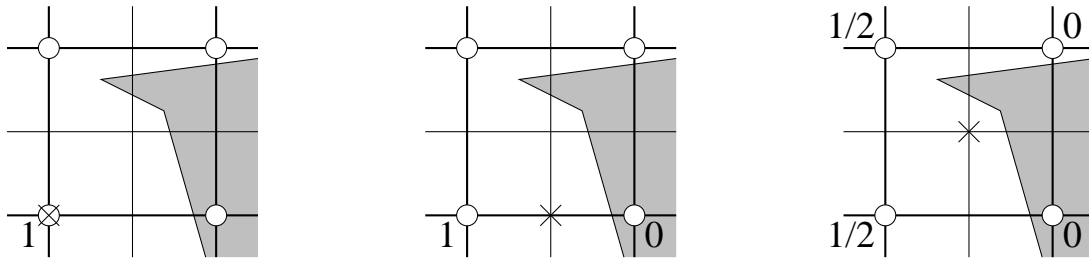
Figure 3.5: Interpolation in two dimensions to a fine node indicated by $\times$, when the refinement ratio is 2. The shaded area is outside the domain. For three different fine nodes, weights are shown on the coarse nodes used for interpolation. Compare with Figure 3.4.

Chapter 4

Coarse/Fine Boundary Interpolation

4.1 No-Geometry Case
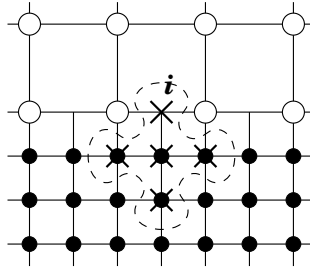
Figure 4.1: A two-dimensional example in which the stencil of the operator includes a node i on the interface with the next coarser level. This is not a *valid* node of the finer level. Valid nodes of the finer level are marked •, and valid nodes of the coarser level are marked ○.

When the stencil for evaluating the operator on $\varphi^{N,l}$ at level l includes a node i lying on the boundary with the next coarser level $l-1$ (see Figure 4.1), we must interpolate $\varphi_i^{N,l}$ from $\varphi^{N,l-1}$ at the valid coarser-level nodes.

We assume that the refinement ratio n_{ref}^{l-1} is a power of 2. Then the interpolation from level $l-1$ to level l is a composition of $\log_2(n_{ref}^{l-1})$ successive interpolations each with refinement ratio of 2.

If $n_{ref}^{l-1} > 2$ then we first interpolate from level $l-1$ to the interface with the grids of level l coarsened by $n_{ref}^{l-1}/2$. The remaining inter-level interpolations use data only on the coarse/fine interface and the physical boundary conditions, which are set at each intermediate stage (see Figure 4.2).

In the remainder of this section, we assume a refinement ratio of 2.

If the level-l node i coincides with a node at level $l-1$, then we project the value of $\varphi_{i/2}^{N,l-1}$. In other cases, in two dimensions we use either linear or quadratic interpola-
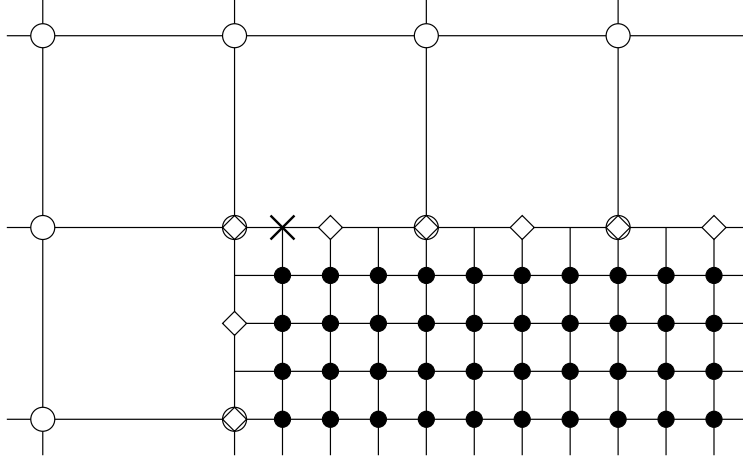
Figure 4.2: A two-dimensional example with refinement ratio of 4. We interpolate to finer-level nodes on the coarse/fine interface in two stages.
First stage: interpolate from the valid coarser-level nodes indicated by $\circ$ to interface nodes at the intermediate level, indicated by $\diamond$. Intermediate-level interface nodes that are also coarser-level nodes are indicated by $\circ$ superimposed with $\diamond$.
Second stage: interpolate from the intermediate-level interface nodes to the finer-level interface nodes. In interpolating to the node marked $\times$, we use only data at nodes indicated by $\diamond$ in this figure, and not data at other coarser-level or intermediate-level nodes.

tion along one-dimensional interfaces, and in three dimensions we use either bilinear or biquadratic interpolation along two-dimensional interfaces.

4.1.1 2-dimensional problem: 1-dimensional interface

In the two-dimensional problem, the coarse/fine interface is one-dimensional. Along the interface, the coordinates vary in only one of the two dimensions, say the first. Let (a, b) be the coordinates of a point of approximation on the interface. If h is the mesh spacing at the finer level, then the coarse nodes on the interface are at $(a \pm h, b)$, $(a \pm 3h, b)$, etc. We interpolate to the fine nodes with a function that is either piecewise linear or piecewise quadratic.

The piecewise linear interpolation function f_1 on the interface has values equal to $\varphi^{N,l-1}$ at the coarse nodes. At the fine nodes,

$$f_1(a) = \frac{f_1(a - h) + f_1(a + h)}{2}. \tag{4.1}$$

This formula gives second-order accuracy,

$$f_1(a) = \varphi^{N,l}(a, b) + O(h^2). \tag{4.2}$$

18
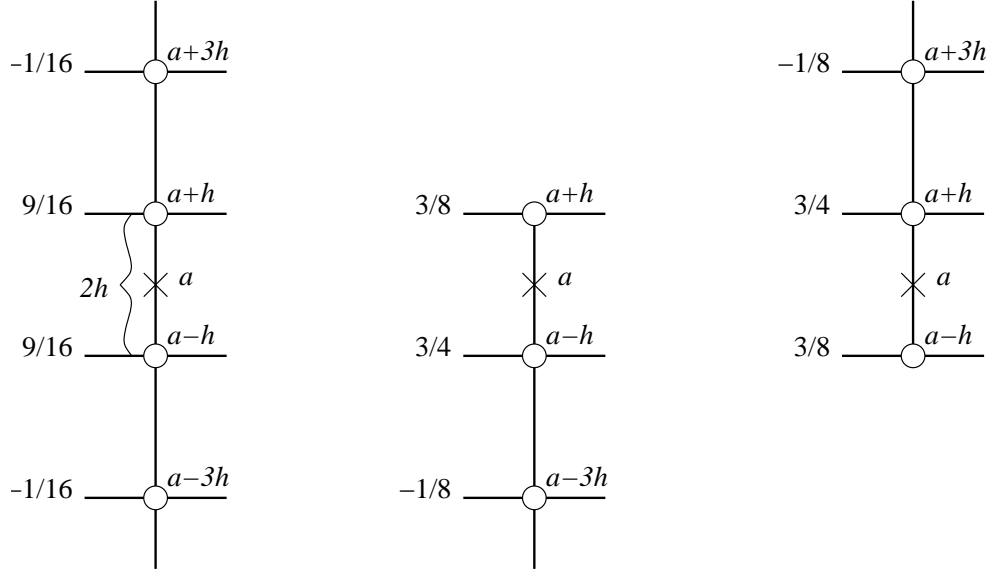
Figure 4.3: We interpolate a value at the fine node marked × from coarse nodes indicated by ○ along the one-dimensional interface. The piecewise linear interpolant $f_1(a)$ is simply the mean of the values at $a - h$ and $a + h$. The piecewise quadratic interpolant $f_2(a)$ also depends on values at $a \pm 3h$. The weight for the piecewise quadratic interpolant is shown to the left of each point.

The piecewise quadratic interpolation function f_2 on the interface also has values equal to $\varphi^{N,l-1}$ at the coarse nodes. At the fine nodes,

$$f_2(a) = f_1(a) - \frac{h^2}{2} f_2''(a) \tag{4.3}$$

where f_1 is the linear interpolant (4.1). The second derivative $f_2''(a)$ then comes from the coarse values $f_2(a - h)$, $f_2(a + h)$, and either $f_2(a - 3h)$ or $f_2(a + 3h)$ or both, using one of the following formulas (see Figure 4.3):

$$f_2''(a) = \frac{f_2(a - 3h) - f_2(a - h) - f_2(a + h) + f_2(a + 3h)}{2(2h)^2} \tag{4.4a}$$

$$f_2''(a) = \frac{f_2(a - 3h) - 2f_2(a - h) + f_2(a + h)}{(2h)^2} \tag{4.4b}$$

$$f_2''(a) = \frac{f_2(a - h) - 2f_2(a + h) + f_2(a + 3h)}{(2h)^2}. \tag{4.4c}$$

We use (4.4a) if both $(a - 3h, b)$ and $(a + 3h, b)$ are valid nodes, and one of the other formulas if not. At least one of the nodes $(a \pm 3h, b)$ will be valid, because a grid at any level, in particular the coarser level, has length at least 2 cells in each dimension. With (4.4a), we have fourth-order accuracy,

$$f_2(a) = \varphi^{N,l}(a, b) + O(h^4) \tag{4.5}$$

because this is actually cubic interpolation; with (4.4b) and (4.4c), we have third-order accuracy,

$$f_2(a) = \varphi^{N,l}(a,b) + O(h^3) \tag{4.6}$$

because this is quadratic interpolation.

4.1.2 3-dimensional problem: 2-dimensional interface

In the three-dimensional problem, the coarse/fine interface is two-dimensional, so that along the interface, the coordinates vary in only two of the three dimensions, say the first two. Let (a,b,c) be the coordinates, in level l, of a point of approximation on the interface with level $l-1$. If both a and b are even, then (a,b,c) is a coarse node, and we project the value $\varphi^{N,l-1}(a/2,b/2,c/2)$. If either a or b is even, but not both (see Figure 4.4, left side) then we interpolate along the line as in section 4.1.1. In the remainder of this section we consider the case in which both a and b are odd (see Figure 4.4, right side). Then the coarse nodes are at $(a\pm h, b\pm h, c)$, $(a\pm 3h, b\pm h, c)$, $(a\pm h, b\pm 3h, c)$, etc. We interpolate to the fine nodes with a function that is either piecewise bilinear or piecewise biquadratic.
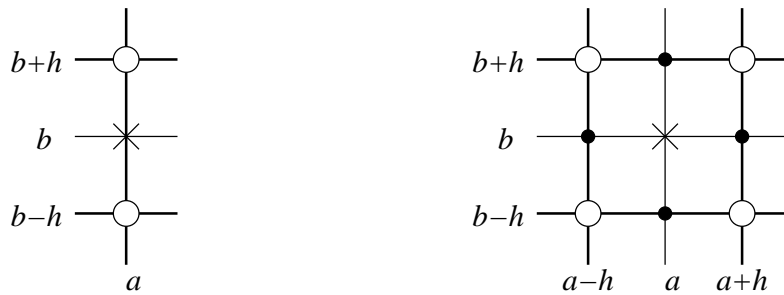
Figure 4.4: Interpolating a value at the point (a,b), marked $\times$, from coarse nodes indicated by $\bigcirc$.
Left: When a is even, interpolate along the line where the first coordinate is fixed at a.
Right: When both a and b are odd, interpolate from the four neighboring coarse nodes and their coarse neighbors. In the piecewise biquadratic interpolant, second partial derivatives are estimated at points indicated by $\bullet$.

The piecewise bilinear interpolation function f_1 on the interface has values equal to $\varphi^{N,l-1}$ at the coarse nodes. At the fine nodes,

$$f_1(a,b) = \frac{f_1(a-h,b-h) + f_1(a-h,b+h) + f_1(a+h,b-h) + f_1(a+h,b+h)}{4} \tag{4.7}$$

This formula gives second-order accuracy,

$$f_1(a,b) = \varphi^{N,l}(a,b,c) + O(h^2). \tag{4.8}$$

The piecewise biquadratic interpolation function f_2 on the interface also has values equal to $\varphi^{N,l-1}$ at the coarse nodes. At the fine nodes,

$$f_2(a, b) = f_1(a, b) - \frac{h^2}{2}\left(\frac{\partial^2 f_2}{\partial x^2}(a, b) + \frac{\partial^2 f_2}{\partial y^2}(a, b)\right). \tag{4.9}$$

where f_1 is the linear interpolant (4.7). Our value for the second derivative $\frac{\partial^2 f_2}{\partial x^2}(a, b)$ comes from the mean of estimates of $\frac{\partial^2 f_2}{\partial x^2}(a, b - h)$ and $\frac{\partial^2 f_2}{\partial x^2}(a, b + h)$. Likewise for $\frac{\partial^2 f_2}{\partial y^2}(a, b)$ from $\frac{\partial^2 f_2}{\partial y^2}(a - h, b)$ and $\frac{\partial^2 f_2}{\partial y^2}(a + h, b)$. We calculate these derivatives from points along a line, following equations (4.4) as described in Section 4.1.1. Then the formula for f_2 is third-order accurate,

$$f_2(a, b) = \varphi^{N,l}(a, b, c) + O(h^3). \tag{4.10}$$

4.2   Case of Non-Rectangular Domains

If the domain is not rectangular, then coarse/fine boundary interpolation may be more complicated. Near a fine node on the coarse/fine interface, some of the coarse-node neighbors may be outside the domain.

As in the no-geometry case, we assume a refinement ratio of 2. For higher refinement ratios, we perform a composition of interpolations with refinement ratio of 2.

Again, if the level-l node $\boldsymbol{i}$ coincides with a node at level $l-1$, then we project the value of $\varphi^{N,l-1}_{\boldsymbol{i}/2}$. In other cases, where possible we use linear or quadratic interpolation in two dimensions, or bilinear or biquadratic interpolation in three dimensions. The interpolation formulas are modified if some of the required coarse points are covered.

For non-rectangular domains, our algorithms depend on whether or not nodes are *reachable* from other nodes. As in section 1.2, "reachable" means that the segment connecting the nodes lies entirely in the domain.

4.2.1   2-dimensional problem: 1-dimensional interface

In the two-dimensional problem, the coarse/fine interface is one-dimensional. As in section 4.1.1, let (a, b) be the coordinates of a point on the interface over which the first coordinate varies. Let h be the finer-level mesh spacing. We assume that at least one of the coarse neighbors $(a - h, b)$ and $(a + h, b)$ is reachable from a. If both are reachable, then we use the formulas of section 4.1.1 and achieve the same accuracy − with the exception that quadratic interpolation is not possible if neither of the nodes $(a \pm 3h, b)$ is reachable from (a, b).

Now suppose $(a + h, b)$ is *not* reachable from (a, b), but $(a - h, b)$ is reachable. If $(a - 3h, b)$ is not reachable from (a, b), then we must approximate the value at (a, b) by the value at $(a - h, b)$, and this approximation is first-order accurate. But if $(a - 3h, b)$

is reachable from (a, b), then linear interpolation uses the one-sided formula

$$f_1(a) = \frac{3}{2}f_1(a - h) - \frac{1}{2}f_1(a - 3h)$$

which is still second-order accurate. Quadratic interpolation from one side is possible if $(a - 5h, b)$ is also reachable from (a, b):

$$f_2(a) = \frac{15}{8}f_2(a - h) - \frac{5}{4}f_2(a - 3h) + \frac{3}{8}f_2(a - 5h)$$

which is a third-order accurate formula. See Figure 4.5.
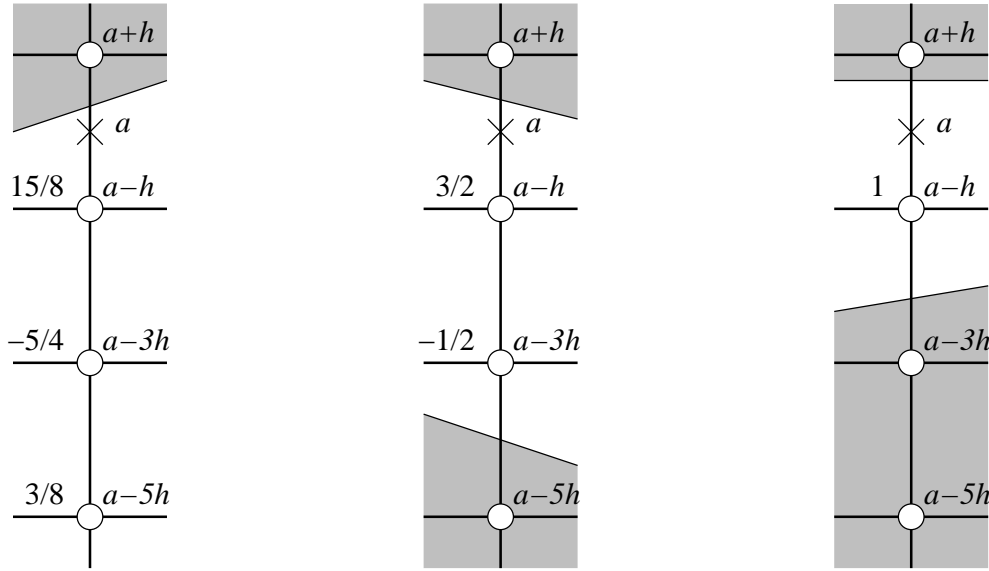
Figure 4.5: Interpolating at the fine node a marked $\times$ from coarse nodes indicated by $\bigcirc$ along the one-dimensional interface when $a + h$ is not reachable from a. We use the weights displayed at the coarse nodes, under one of the three possibilities of reachability of $a - 3h$ and $a - 5h$ from a. Interpolation is either quadratic (left), linear (center), or constant (right).

4.2.2 3-dimensional problem: 2-dimensional interface

In the three-dimensional problem, the coarse/fine interface is two-dimensional. As in section 4.1.2, let (a, b, c) be the coordinates of a point on the interface, along which the first two coordinates vary but the third is fixed. If either a or b is even then we interpolate along the line, following section 4.2.1. In the remainder of this section, we assume both a and b are odd.

Letting h be the finer-level mesh spacing, then the nearest coarse neighbors of (a, b, c) on the interface are $(a \pm h, b \pm h, c)$. We assume that at least one of these is reachable from (a, b, c). If all of them are reachable, then we can use the formulas of section 4.1.2.

If not all four of the nearest coarse neighbors on the interface are reachable from (a, b, c), then the first approximation is the mean of the values at the subset of these coarse neighbors that are in the domain, except that if the number is three, then one of them is ignored. See Figure 4.6 for the five cases: (a) one coarse neighbor; (b) two coarse neighbors, on a line parallel to an axis; (c) two coarse neighbors, not on a line parallel to an axis; (d) three coarse neighbors; (e) four coarse neighbors.

(a)　　　　　(b)　　　　　(c)　　　　　(d)　　　　　(e)
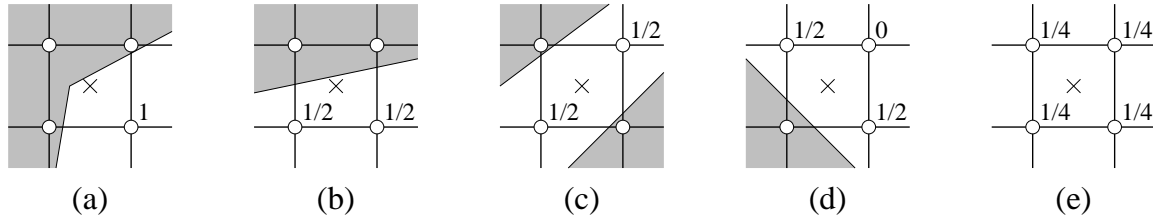
Figure 4.6: The first approximation at the center node indicated by $\times$ is a weighted average of the neighboring coarse nodes that are in the domain. Weights are indicated beside the coarse nodes.

In cases (c), (d) and (e), this first approximation is bilinear. In cases (a) and (b), in order to get a bilinear approximation we add a one-sided bilinear correction if coarse nodes neighboring the four base ones are in the domain. See Figure 4.7.

(a)　　　　　　　　　　　　(b)
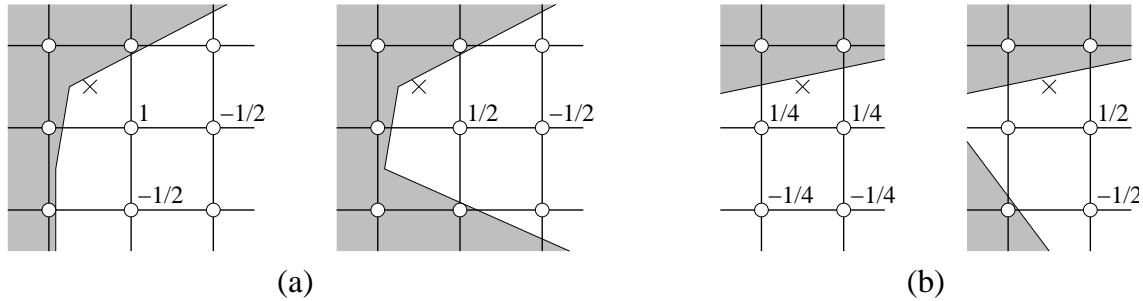
Figure 4.7: One-sided bilinear correction to approximation at center node $\times$. Weights of the correction are indicated beside the coarse nodes.
(a) If only one coarse neighbor of $\times$ is in the domain, we add a correction in each dimension where it is possible.
(b) If two coarse neighbors of $\times$ are in the domain, and they lie on a line parallel to an axis, then we add a correction in the perpendicular direction if possible.

From the bilinear approximation, we can make a biquadratic correction if there are enough coarse nodes in the domain to obtain estimates of the second derivatives. We use one of the formulas (4.4) for estimating the second derivative in each dimension, where the correction term takes the form $-\frac{h^2}{2} f_2''(a)$. In addition, if we made a one-sided correction, cases (a) or (b) above, then in each dimension where the correction was made we also add a correction of $\frac{(2h)^2}{2} f_2''(a)$.

4.3 Truncation Error of Laplacian Near Interfaces

Recall that the purpose of interpolating along the coarse/fine interface is to obtain data for a point of the stencil of the Laplacian operator (see Figure 4.1) given by (2.1)–(2.2). Because of the h^2 denominator in (2.2), the truncation error is two orders of accuracy below that of the approximation error at the interface point, and at most second-order.

Hence the second-order error in linear interpolation in two dimensions (4.2) or bilinear interpolation in three dimensions (4.8) gives rise to constant truncation error:

$$\Delta^h \varphi^{N,l} = \Delta \varphi^{N,l} + O(1).$$

Third-order error in quadratic interpolation in two dimensions (4.6) or biquadratic interpolation in three dimensions (4.10) gives rise to first-order truncation error:

$$\Delta^h \varphi^{N,l} = \Delta \varphi^{N,l} + O(h).$$

When the stencil includes interpolated points with fourth-order error in cubic interpolation in two dimensions (4.5) or bicubic interpolation in three dimensions, truncation error is second-order:

$$\Delta^h \varphi^{N,l} = \Delta \varphi^{N,l} + O(h^2).$$

Chapter 5

AMR-Multigrid Algorithm

In the pseudocode description of the algorithm, we use the operator L^{nf} which is a two-level discretization of the Laplacian: $L^{nf}(\varphi^l, \varphi^{l-1})$ is defined on the nodes $\Omega^l_{N,int}$, as described as Δ^h in chapter 2, with the assumption that there is no finer level.

5.1 Multigrid Solver on One Level

For the one-level multigrid solver, we have a recursive smoothing operator $\mathtt{mgRelax}(\varphi^f, R^f, n_c)$ on φ^f with operator L^{nf} and right-hand side R^f, assuming identically zero values on the physical and coarse/fine interface boundary conditions. The argument n_c is the depth of the V-cycle, usually chosen to be the highest number such that 2^{n_c} divides the length of every side of every grid on the level. This routine uses the average-to-coarse routine $Average$, described in section 3.1, and the coarse-to-fine interpolation routine $Interp$, described in section 3.2. It also calls the point-relaxation function $\mathtt{LevelGSRB}$ described in section 5.2, and a bottom smoother as described in section 5.4.

See Figure 5.1 for a pseudo-code description of the multigrid solver on a level.

5.2 Point Relaxation

For point relaxation we use procedure $\mathtt{LevelGSRB}(\varphi^f, R^f)$, which performs Gauss-Seidel iteration with red-black ordering, as shown in Figure 5.2. This operation uses data on one level only. Gauss-Seidel iteration increments φ^f by the residual weighted at each point by a relaxation parameter λ, such that the diagonal of the resulting operator is zero. Red-black ordering means that we relax using two passes through the domain in a checkerboard pattern: on the first pass, we relax on the points $\Omega^{BLACK}_N \subset \Omega_N$ where the sum of the coordinates is odd; on the second pass, we relax on the remaining points $\Omega^{RED}_N \subset \Omega_N$ where the sum of the coordinates is even.

$$R^f := \rho^f - L^{nf}(\varphi^f, \varphi^c)$$
$$e^f := 0$$
while $(||R^f - L^{nf}(e^f, e^c \equiv 0)|| > \epsilon||\rho^f||)$
    mgRelax(e^f, R^f, n_c)
end while
$$\varphi^f := \varphi^f + e^f$$

procedure mgRelax(φ^f, R^f, n_c)
{
    if $(n_c > 0)$ then
        $\delta^c := 0$
        for $i = 1, \ldots,$ NumSmoothDown
            LevelGSRB(φ^f, R^f)
        end for
        $R^c := Average(R^f - L^{nf}(\varphi^f, \varphi^c \equiv 0))$
        mgRelax($\delta^c, R^c, n_c - 1$)
        $\varphi^f := \varphi^f + Interp(\delta^c)$
        for $i = 1, \ldots,$ NumSmoothUp
            LevelGSRB(φ^f, R^f)
        end for
    else
        for $i = 1, \ldots,$ NumBottomGSRB
            LevelGSRB(φ^f, R^f)
        end for
        if (points > 1) then
            bottomSmooth(φ^f, R^f)
        end if
    end if
}

Figure 5.1: Pseudo-code description of the multigrid solver on a level, with the recursive relaxation procedure. Here n_c is the depth of the V-cycle.

procedure LevelGSRB(φ^f, R^f)
{

$$\varphi^f := \varphi^f + \lambda(L^{nf}(\varphi^f, \varphi^c \equiv 0) - R^f) \text{ on } \Omega_N^{BLACK}$$
$$\varphi^f := \varphi^f + \lambda(L^{nf}(\varphi^f, \varphi^c \equiv 0) - R^f) \text{ on } \Omega_N^{RED}$$

}

Figure 5.2: Gauss-Seidel relaxation with red-black ordering. Here λ is the relaxation parameter.

5.3 AMR Multigrid Algorithm

Pseudo-code for the AMR multigrid algorithm to solve $L(\varphi) = \rho$ is shown in Figure 5.4. The smoothing operator `AMRmgRelax`(φ^f, R^f, r), described in detail in Figure 5.3, performs a mini-V-cycle iteration on φ^f for the operator L^{nf} and right-hand side R^f, assuming the coarse-grid values required for the boundary conditions are identically zero. The number r is the refinement ratio. This routine uses the average-to-coarse routine $Average$, described in section 3.1, and the coarse-to-fine interpolation routine $Interp$, described in section 3.2.

5.4 Bottom Smoothers

For the bottom smoother in the multigrid algorithm (see Figure 5.1), we use either a conjugate gradient algorithm (see Figure 5.6) or a modified version of the BiCGStab algorithm (see Figure 5.5), as described in [BBC+94].

5.4.1 Preconditioner

Functionally, applyPreconditioner($\hat{\phi}, \phi$) is equivalent to solving $M\hat{\phi} = \phi$. The preconditioner initializes with the solution to $D(L)\hat{\phi} = \phi$, where $D(L)$ is the diagonal part of the operator. Then it does a few GSRB passes, as shown in Figure 5.7.

5.5 Level-by-level Multigrid Algorithm

As an alternative to the AMR multigrid algorithm, we may solve level by level, from coarsest to finest, interpolating the coarse solution to the fine solution between each level solve. See Figure 5.8.

```
procedure AMRmgRelax($\varphi^f, R^f, r$)
{
    for $i = 1, \ldots,$ NumBottomGSRB
        LevelGSRB($\varphi^f, R^f$)
    end for
    if $(r > 2)$ then
        for $i = 1, \ldots,$ NumSmoothDown
            LevelGSRB($\varphi^f, R^f$)
        end for
        $\delta^c := 0$
        $R^c := Average(R^f - L^{nf}(\varphi^f, \varphi^c \equiv 0))$
        AMRmgRelax($\delta^c, R^c, r/2$)
        $\varphi^f := \varphi^f + Interp(\delta^c)$
        for $i = 1, \ldots,$ NumSmoothUp
            LevelGSRB($\varphi^f, R^f$)
        end for
    end if
}
```

Figure 5.3: Recursive relaxation procedure.

$R := \rho - L(\varphi)$
while $(||R|| > \epsilon||\rho||)$
    AMRVCycleMG(l^{max})
    $R := \rho - L(\varphi)$
end while

procedure AMRVCycleMG(level l):
{
    if $(l = l^{max})$ then $R^l := \rho^l - L^{nf}(\varphi^l, \varphi^{l-1})$
    if $(l > 0)$ then
        $\varphi^{l,save} := \varphi^l$ on Ω_N^l
        $e^l := 0$ on Ω_N^l
        AMRmgRelax(e^l, R^l, n_{ref}^{l-1})
        $\varphi^l := \varphi^l + e^l$
        $e^{l-1} := 0$ on Ω_N^{l-1}
        $R^{l-1} := Average(R^{l-1} - L^{nf}(e^l, e^{l-1}))$ on $\mathcal{C}_{n_{ref}^{l-1}}(\Omega_N^l)$
        $R^{l-1} := \rho^{l-1} - L^{comp,l-1}(\varphi)$ on $\Omega_N^{l-1} - \mathcal{C}_{n_{ref}^{l-1}}(\Omega_N^l)$
        AMRVCycleMG($l - 1$)
        $e^l := e^l + Interp(e^{l-1})$
        $R^l := R^l - L^{nf,l}(e^l, e^{l-1})$
        $\delta e^l := 0$ on Ω_N^l
        AMRmgRelax($\delta e^l, R^l, n_{ref}^{l-1}$)
        $e^l := e^l + \delta e^l$
        $\varphi^l := \varphi^{l,save} + e^l$
    else
        solve $L^{nf}(e^0, 0) = R^0$ on Ω_N^0, by one-level multigrid algorithm.
        $\varphi^0 := \varphi^0 + e^0$
    end if
}

Figure 5.4: Pseudo-code description of the AMR multigrid algorithm.

```
procedure bottomSmooth(x, b)
{
    r^(0) := b - L^{nf}(x, 0)
    r̃ := r^(0), i := 0
    e^(0) := 0 (set correction to 0)
    while i < i_{max} and ||r^(i)|| > ε||r^(0)||
        i := i + 1
        ρ_{i-1} = r̃^T r^(i-1)
        if ρ_{i-1} = 0 then    (method fails)
            x := x + e^(i-1)
            exit
        end if
        if i = 1 then
            p^(i) := r^(i-1)
        else
            β_{i-1} := (ρ_{i-1}/ρ_{i-2}) · (α_{i-1}/ω_{i-1})
            p^(i) := r^(i-1) + β_{i-1}(p^(i-1) - ω_{i-1}v^(i-1))
        end if
        applyPreconditioner(p̂, p^(i))
        v^(i) := L^{nf}(p̂, 0)
        α_i := ρ_{i-1}/(r̃^T v^(i))
        check magnitude of α_i v^(i):  OK := (r̃^T v^(i) > (small)ρ_{i-1})
        if OK then
            r^(i) := r^(i-1) - α_i v^(i)
            e^(i) := e^(i-1) + α_i p̂
        else
            r^(i) := 0 (this will force restart from clean residual)
            e^(i) := e^(i-1)
        end if
        if ||r^(i)|| > ε||r^0|| then
            applyPreconditioner(ŝ, r^(i))
            t := L^{nf}(ŝ, 0)
            ω_i := (t^T r^(i))/(t^T t)
            e^(i) := e^(i) + ω_i ŝ
            r^(i) := r^(i) - ω_i t
        end if
        if ||r^(i)|| < ε||r^(0)|| or ω_i = 0 then
            x := x + e^(i)
            r^(i) := b - L^{nf}(x, 0) (recompute "clean" residual)
            r̃ := r^(i)
            e^(i) := 0 (reset correction to 0, since we've already incremented solution)
        end if
    end while                                30
}
```

```
procedure bottomSmooth(x, b)
{
```

$r^{(0)} := b - L^{nf}(x, 0)$

$\tilde{r} := r^{(0)}$, $i := 0$

$e^{(0)} := 0$ (set correction to 0)

while $i < i_{max}$ and $||r^{(i)}|| > \epsilon||r^{(0)}||$

 $i := i + 1$

 $\rho_{i-1} = \tilde{r}^T r^{(i-1)}$

 if $\rho_{i-1} = 0$ then   (method fails)

  $x := x + e^{(i-1)}$

  exit

 end if

 if $i = 1$ then

  $p^{(i)} := r^{(i-1)}$

 else

  $\beta_{i-1} := (\rho_{i-1}/\rho_{i-2}) \cdot (\alpha_{i-1}/\omega_{i-1})$

  $p^{(i)} := r^{(i-1)} + \beta_{i-1}(p^{(i-1)} - \omega_{i-1}v^{(i-1)})$

 end if

 applyPreconditioner($\hat{p}, p^{(i)}$)

 $v^{(i)} := L^{nf}(\hat{p}, 0)$

 $\alpha_i := \rho_{i-1}/(\tilde{r}^T v^{(i)})$

 check magnitude of $\alpha_i v^{(i)}$: OK := $(\tilde{r}^T v^{(i)} > (small)\rho_{i-1})$

 if OK then

  $r^{(i)} := r^{(i-1)} - \alpha_i v^{(i)}$

  $e^{(i)} := e^{(i-1)} + \alpha_i \hat{p}$

 else

  $r^{(i)} := 0$ (this will force restart from clean residual)

  $e^{(i)} := e^{(i-1)}$

 end if

 if $||r^{(i)}|| > \epsilon||r^0||$ then

  applyPreconditioner($\hat{s}, r^{(i)}$)

  $t := L^{nf}(\hat{s}, 0)$

  $\omega_i := (t^T r^{(i)})/(t^T t)$

  $e^{(i)} := e^{(i)} + \omega_i \hat{s}$

  $r^{(i)} := r^{(i)} - \omega_i t$

 end if

 if $||r^{(i)}|| < \epsilon||r^{(0)}||$ or $\omega_i = 0$ then

  $x := x + e^{(i)}$

  $r^{(i)} := b - L^{nf}(x, 0)$ (recompute "clean" residual)

  $\tilde{r} := r^{(i)}$

  $e^{(i)} := 0$ (reset correction to 0, since we've already incremented solution)

 end if

end while

30

```
}
```

Figure 5.5: Pseudo-code description of the BiCGStab bottom smoother.

```
procedure bottomSmooth(x, b)
{
    r^(0) := b - L^{nf}(x, 0)
    i := 0
    e^(0) := 0 (set correction to 0)
    while i < i_max and ||r^(i)|| > ε||r^(0)||
        i := i + 1
        applyPreconditioner(z, r^(i-1))
        ρ_{i-1} = z^T r^(i-1)
        if ρ_{i-1} = 0 then   (method fails)
        if i = 1 then
            p := z
        else
            p := z + (ρ_{i-1}/ρ_{i-2})p
        end if
        q := L^{nf}(p, 0)
        α_i := ρ_{i-1}/(p^T q)
        check magnitude of α_i q:  OK := (p^T q > (small)ρ_{i-1})
        if OK then
            x := x + α_i p
            r^(i) := r^(i-1) - α_i q
        else
            r^(i) := 0 (this will force restart from clean residual)
        end if
    end while
}
```

Figure 5.6: Pseudo-code description of the conjugate gradient bottom smoother.

```
procedure applyPreconditioner(φ̂, φ)
{
    solve D(L)φ̂ = φ
    LevelGSRB(φ̂, φ)
    LevelGSRB(φ̂, φ)
}
```

Figure 5.7: Pseudo-code description of applying the preconditioner.

for $l = 0, \ldots, l^{max}$
    solve $L^{nf}(\varphi^l, \varphi^{l-1}) = \rho^l$ on Ω_N^l, by one-level multigrid algorithm.
end for

Figure 5.8: Pseudo-code description of the level-by-level multigrid algorithm.

Chapter 6

Convergence Tests

6.1 Norms over Multilevel Hierarchies

Before describing the convergence tests, we define the norms we use to measure error on multilevel hierarchies. Let $f^{comp} = \{f^l\}_{l=0}^{l_{max}}$ be a composite array defined on the nodes of each level.

The L_∞ norm of f^l is defined as the maximum value over the valid nodes at level l:

$$||f^l||_\infty = \max_{\boldsymbol{i} \in \Omega^l_{N,valid}} |f^l_{\boldsymbol{i}}|.$$

The composite L_∞ norm is the maximum over all levels:

$$||f^{comp}||_\infty = \max_{l=0}^{l_{max}} ||f^l||_\infty.$$

For finite p we define the L_p norm of f^l by integrating $|f^l|^p$ over the valid nodes. If the mesh spacing on level l is h_l, then

$$||f^l||_p = (h_l^{\mathbf{D}} \sum_{\boldsymbol{i} \in \Omega^l_{N,valid}} |f^l_{\boldsymbol{i}}|^p)^{1/p}.$$

The composite L_p norm is the p-th root of the sum of p-th powers of the norms of each level:

$$||f^{comp}||_p = (\sum_{l=0}^{l_{max}} ||f^l||_p^p)^{1/p} = (\sum_{l=0}^{l_{max}} h_l^{\mathbf{D}} \sum_{\boldsymbol{i} \in \Omega^l_{N,valid}} |f^l_{\boldsymbol{i}}|^p)^{1/p}.$$

6.2 Operator Tests, No-Geometry Case

We perform some tests over two-level grid hierarchies in two and three dimensions. In two dimensions, the hierarchy is shown in Figure 6.1 and has a refinement ratio of four between the levels. The three-dimensional grid hierarchy is shown in Figure 6.2 and has a refinement ratio of two between the levels. In both cases, the coarser-level domain has unit length in each dimension, and is centered at the origin.
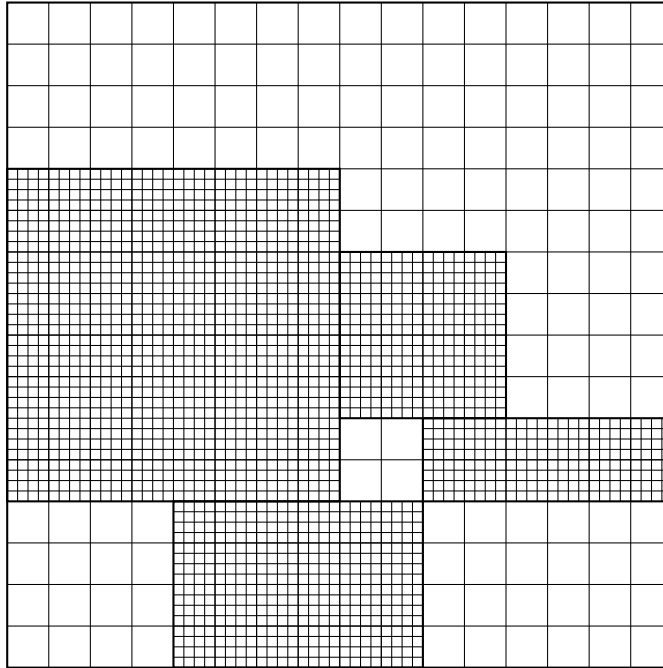
Figure 6.1: Grid configuration for two-dimensional operator tests. There are two levels of refinement, with a refinement ratio of four between them. In this illustration, we show one 16×16 grid at the coarser level, and four grids at the finer level. We also use fully refined versions of this same set of grids, partitioned so that the maximum length of any grid in any dimension is 32.
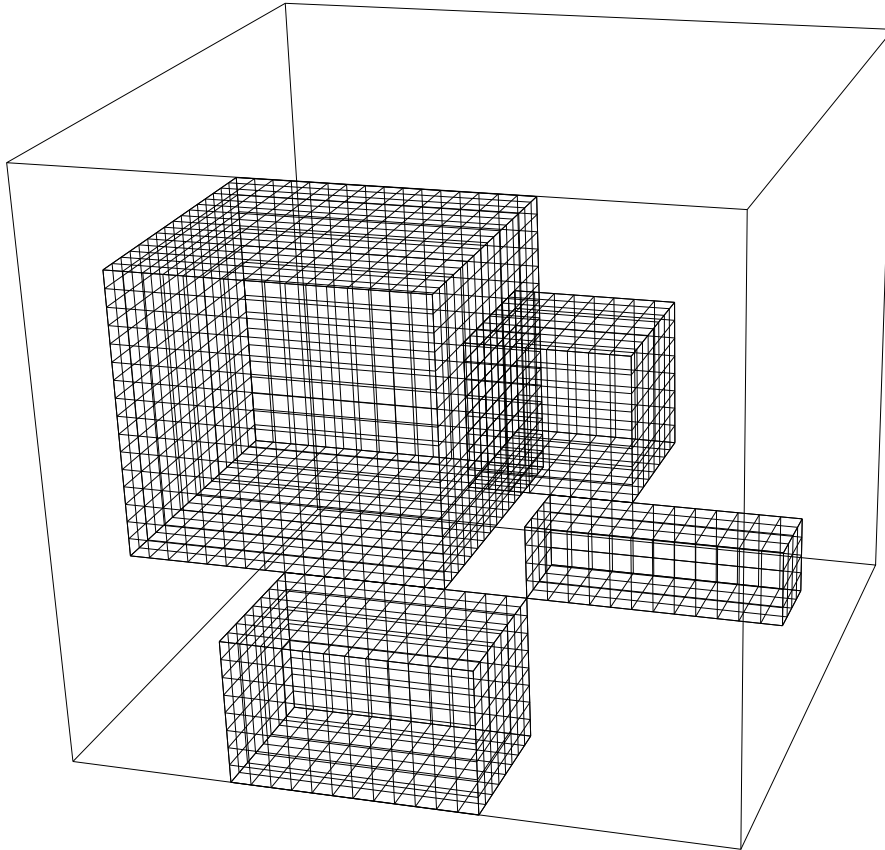
Figure 6.2: Grid configuration for three-dimensional operator tests. There are two levels of refinement. The edges of the coarser-level grid and the face cells of the finer-level grids are shown. Note that the projection of the finer-level grids on the the (horizontal) xy-plane yields the configuration of Figure 6.1. The refinement ratio between the levels is two. In this illustration, the coarser-level domain contains one $16 \times 16 \times 16$ grid, and there are four grids at the finer level where the domain is $32 \times 32 \times 32$. We also use fully refined versions of this same set of grids, partitioned so that the maximum length of any grid in any dimension is 32.

6.2.1 Laplacian of quadratic

We first test the discrete Laplacian operator $\Delta^h \varphi$ by checking that it evaluates the correct answer if φ is quadratic over a hierarchy. Since φ is quadratic, its third derivative is identically zero, so the truncation error of the operator is also zero in this special case, whether the coarse/fine interpolation is (bi)quadratic or (bi)linear. In two dimensions, we initialize the solution to be the quadratic

$$\varphi(x, y) = x^2 + y^2 \tag{6.1}$$

over the grid hierarchy in Figure 6.1, and we evaluate the Laplacian $\Delta^h \varphi$. At all of the valid nodes, we find $\Delta^h \varphi = 4$. In three dimensions, we initialize the solution to be the quadratic

$$\varphi(x, y, z) = x^2 + y^2 + z^2 \tag{6.2}$$

over the grid hierarchy shown in Figure 6.2, and we evaluate the Laplacian $\Delta^h \varphi$. At all of the valid nodes, we find $\Delta^h \varphi = 6$.

6.2.2 Gaussian example

In our next tests, we initialize φ to be the Gaussian function

$$\varphi(x, y) = \exp(-\frac{x^2 + y^2}{4\sigma^2}) \tag{6.3}$$

in two dimensions or

$$\varphi(x, y, z) = \exp(-\frac{x^2 + y^2 + z^2}{4\sigma^2}) \tag{6.4}$$

in three dimensions, where $\sigma = \frac{1}{8}$.

6.2.3 Trigonometric polynomial on semi-periodic domain

For examples on periodic domains, we choose the same grids in Figures 6.1 and 6.2, with periodicity in the x direction. That is, the domain wraps around from $x = 1/2$ to $x = -1/2$.

We initialize φ to be the trigonometric polynomial

$$\varphi(x, y) = \frac{\sin(2\pi x) + \sin(2\pi y)}{-4\pi^2} \tag{6.5}$$

in two dimensions or

$$\varphi(x, y, z) = \frac{\sin(2\pi x) + \sin(2\pi y) + \sin(2\pi z)}{-4\pi^2} \tag{6.6}$$

in three dimensions.

36

6.2.4 Tests performed

We compute the discrete Laplacian $\Delta^h \varphi$ at two different levels of refinement and take the difference with the exact Laplacian, $\Delta \varphi$. Using three different norms (L_1, L_2, and L_∞), we compute the order of convergence of the operator with grid refinement. If φ_h denotes the composite array with mesh spacing of h at the base level, then the order of convergence p is given by

$$p = \frac{1}{\ln(2)} \ln\left(\frac{||\Delta^{2h} \varphi_{2h} - \Delta \varphi_{2h}||}{||\Delta^h \varphi_h - \Delta \varphi_h||}\right). \tag{6.7}$$

We also test the convergence of the norms of the difference between the calculated values of $\Delta^{2h} \varphi_{2h}$ and the projection of those calculated on the same set of grids refined by two, $P_2(\Delta^h \varphi_h)$. Using three different mesh spacings, h, $2h$, and $4h$, the order of convergence p is

$$p = \frac{1}{\ln(2)} \ln\left(\frac{||\Delta^{4h} \varphi_{4h} - P_2(\Delta^{2h} \varphi_{2h})||}{||\Delta^{2h} \varphi_{2h} - P_2(\Delta^h \varphi_h)||}\right). \tag{6.8}$$

Finally, we test with both linear and quadratic interpolation along the coarse/fine boundary interace in two dimensions, and both bilinear and biquadratic interpolation along the interface in three dimensions.

Summarizing our tests, we vary the following.

- Dimensions: two (example with refinement ratio of four) or three (example with refinement ratio of two).

- Mesh spacings.

- Norms: L_1, L_2, and L_∞.

- Error: difference with exact error, $\Delta^h \varphi_h - \Delta \varphi_h$, or with projected value, $\Delta^{2h} \varphi_{2h} - P_2(\Delta^h \varphi_h)$.

- Degree of coarse/fine interpolation: quadratic (biquadratic) or linear (bilinear).

6.2.5 Interpretation of results

Tables 6.1–6.2 and 6.5–6.6 show that when using quadratic interpolation in two dimensions or biquadratic interpolation in three dimensions, the discrete Laplacian operator converges to first order in L_∞ norm and to second order in L_1 norm. The L_2 norm of the error is $O(h^{3/2})$ in two dimensions and $O(h^2)$ in three dimensions. These results are consistent with the operator error being $O(h^2)$ everywhere except on a one-dimensional set on which the error is $O(h)$, as shown in Table 6.9. In three dimensions, the set of points with data interpolated biquadratically is one-dimensional instead of two-dimensional because on the interior of the coarse/fine interfaces, we interpolate bicubically.

D	deg	norm	h	$\|\|\Delta^h \varphi_h - \Delta\varphi_h\|\|$	$\|\|\Delta^{2h} \varphi_{2h} - \Delta\varphi_{2h}\|\|$	p
2	2	L_∞	1/128	2.99939E-01	6.28624E-01	1.07
2	2	L_∞	1/256	1.46318E-01	2.99939E-01	1.04
2	2	L_∞	1/512	7.22411E-02	1.46318E-01	1.02
2	2	L_∞	1/1024	3.58907E-02	7.22411E-02	1.01
2	2	L_1	1/128	1.39222E-03	5.95619E-03	2.10
2	2	L_1	1/256	3.36113E-04	1.39222E-03	2.05
2	2	L_1	1/512	8.25601E-05	3.36113E-04	2.03
2	2	L_1	1/1024	2.04562E-05	8.25601E-05	2.01
2	2	L_2	1/128	3.39859E-03	1.35755E-02	2.00
2	2	L_2	1/256	9.29003E-04	3.39859E-03	1.87
2	2	L_2	1/512	2.75872E-04	9.29003E-04	1.75
2	2	L_2	1/1024	8.74869E-05	2.75872E-04	1.66

D	deg	norm	h	$\|\|\Delta^{2h} \varphi_{2h} - P_2(\Delta^h \varphi_h)\|\|$	$\|\|\Delta^{4h} \varphi_{4h} - P_2(\Delta^{2h} \varphi_{2h})\|\|$	p
2	2	L_∞	1/128	6.28710E-01	1.36572E+00	1.12
2	2	L_∞	1/256	2.99961E-01	6.28710E-01	1.07
2	2	L_∞	1/512	1.46323E-01	2.99961E-01	1.04
2	2	L_∞	1/1024	7.22425E-02	1.46323E-01	1.02
2	2	L_1	1/128	4.60020E-03	2.13511E-02	2.21
2	2	L_1	1/256	1.06385E-03	4.60020E-03	2.11
2	2	L_1	1/512	2.55320E-04	1.06385E-03	2.06
2	2	L_1	1/1024	6.25220E-05	2.55320E-04	2.03
2	2	L_2	1/128	1.19733E-02	5.39635E-02	2.17
2	2	L_2	1/256	3.02249E-03	1.19733E-02	1.99
2	2	L_2	1/512	8.46416E-04	3.02249E-03	1.84
2	2	L_2	1/1024	2.58986E-04	8.46416E-04	1.71

Table 6.1: Convergence test of discrete Laplacian operator on Gaussian (6.3) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and quadratic (deg = 2) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$||\Delta^h \varphi_h - \Delta \varphi_h||$	$||\Delta^{2h} \varphi_{2h} - \Delta \varphi_{2h}||$	p
2	2	L_∞	1/128	4.90485E-02	9.78139E-02	1.00
2	2	L_∞	1/256	2.45404E-02	4.90485E-02	1.00
2	2	L_∞	1/512	1.22718E-02	2.45404E-02	1.00
2	2	L_∞	1/1024	6.13602E-03	1.22718E-02	1.00
2	2	L_1	1/128	1.04271E-04	4.38903E-04	2.07
2	2	L_1	1/256	2.53882E-05	1.04271E-04	2.04
2	2	L_1	1/512	6.26122E-06	2.53882E-05	2.02
2	2	L_1	1/1024	1.55466E-06	6.26122E-06	2.01
2	2	L_2	1/128	3.84463E-04	1.52148E-03	1.98
2	2	L_2	1/256	9.86578E-05	3.84463E-04	1.96
2	2	L_2	1/512	2.59415E-05	9.86578E-05	1.93
2	2	L_2	1/1024	7.09558E-06	2.59415E-05	1.87

D	deg	norm	h	$||\Delta^{2h} \varphi_{2h} - P_2(\Delta^h \varphi_h)||$	$||\Delta^{4h} \varphi_{4h} - P_2(\Delta^{2h} \varphi_{2h})||$	p
2	2	L_∞	1/128	9.78007E-02	1.93217E-01	0.98
2	2	L_∞	1/256	4.90453E-02	9.78007E-02	1.00
2	2	L_∞	1/512	2.45396E-02	4.90453E-02	1.00
2	2	L_∞	1/1024	1.22716E-02	2.45396E-02	1.00
2	2	L_1	1/128	3.40011E-04	1.52611E-03	2.17
2	2	L_1	1/256	7.97738E-05	3.40011E-04	2.09
2	2	L_1	1/512	1.92927E-05	7.97738E-05	2.05
2	2	L_1	1/1024	4.74099E-06	1.92927E-05	2.02
2	2	L_2	1/128	1.46500E-03	5.82888E-03	1.99
2	2	L_2	1/256	3.70663E-04	1.46500E-03	1.98
2	2	L_2	1/512	9.53200E-05	3.70663E-04	1.96
2	2	L_2	1/1024	2.51519E-05	9.53200E-05	1.92

Table 6.2: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.5) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and quadratic (deg = 2) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$\|\|\Delta^h\varphi_h - \Delta\varphi_h\|\|$	$\|\|\Delta^{2h}\varphi_{2h} - \Delta\varphi_{2h}\|\|$	p
2	1	L_∞	1/128	4.97998E+01	4.96696E+01	0.00
2	1	L_∞	1/256	4.98324E+01	4.97998E+01	0.00
2	1	L_∞	1/512	4.98405E+01	4.98324E+01	0.00
2	1	L_∞	1/1024	4.98426E+01	4.98405E+01	0.00
2	1	L_1	1/128	5.20676E-02	1.06302E-01	1.03
2	1	L_1	1/256	2.57803E-02	5.20676E-02	1.01
2	1	L_1	1/512	1.28269E-02	2.57803E-02	1.01
2	1	L_1	1/1024	6.39808E-03	1.28269E-02	1.00
2	1	L_2	1/128	9.56406E-01	1.34852E+00	0.50
2	1	L_2	1/256	6.77208E-01	9.56406E-01	0.50
2	1	L_2	1/512	4.79170E-01	6.77208E-01	0.50
2	1	L_2	1/1024	3.38932E-01	4.79170E-01	0.50

D	deg	norm	h	$\|\|\Delta^{2h}\varphi_{2h} - P_2(\Delta^h\varphi_h)\|\|$	$\|\|\Delta^{4h}\varphi_{4h} - P_2(\Delta^{2h}\varphi_{2h})\|\|$	p
2	1	L_∞	1/128	4.96705E+01	4.91548E+01	-0.02
2	1	L_∞	1/256	4.98000E+01	4.96705E+01	0.00
2	1	L_∞	1/512	4.98324E+01	4.98000E+01	0.00
2	1	L_∞	1/1024	4.98405E+01	4.98324E+01	0.00
2	1	L_1	1/128	1.04946E-01	2.16124E-01	1.04
2	1	L_1	1/256	5.17393E-02	1.04946E-01	1.02
2	1	L_1	1/512	2.56995E-02	5.17393E-02	1.01
2	1	L_1	1/1024	1.28068E-02	2.56995E-02	1.00
2	1	L_2	1/128	1.34852E+00	1.89397E+00	0.49
2	1	L_2	1/256	9.56407E-01	1.34852E+00	0.50
2	1	L_2	1/512	6.77208E-01	9.56407E-01	0.50
2	1	L_2	1/1024	4.79170E-01	6.77208E-01	0.50

Table 6.3: Convergence test of discrete Laplacian operator on Gaussian (6.3) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and linear (deg = 1) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$||\Delta^h\varphi_h - \Delta\varphi_h||$	$||\Delta^{2h}\varphi_{2h} - \Delta\varphi_{2h}||$	p
2	1	L_∞	1/128	2.53827E+00	2.51451E+00	-0.01
2	1	L_∞	1/256	2.54964E+00	2.53827E+00	-0.01
2	1	L_∞	1/512	2.55519E+00	2.54964E+00	0.00
2	1	L_∞	1/1024	2.55794E+00	2.55519E+00	0.00
2	1	L_1	1/128	5.24667E-03	1.06590E-02	1.02
2	1	L_1	1/256	2.60286E-03	5.24667E-03	1.01
2	1	L_1	1/512	1.29634E-03	2.60286E-03	1.01
2	1	L_1	1/1024	6.46904E-04	1.29634E-03	1.00
2	1	L_2	1/128	8.26120E-02	1.16683E-01	0.50
2	1	L_2	1/256	5.84471E-02	8.26120E-02	0.50
2	1	L_2	1/512	4.13385E-02	5.84471E-02	0.50
2	1	L_2	1/1024	2.92342E-02	4.13385E-02	0.50

D	deg	norm	h	$||\Delta^{2h}\varphi_{2h} - P_2(\Delta^h\varphi_h)||$	$||\Delta^{4h}\varphi_{4h} - P_2(\Delta^{2h}\varphi_{2h})||$	p
2	1	L_∞	1/128	2.51453E+00	2.46303E+00	-0.03
2	1	L_∞	1/256	2.53828E+00	2.51453E+00	-0.01
2	1	L_∞	1/512	2.54964E+00	2.53828E+00	-0.01
2	1	L_∞	1/1024	2.55519E+00	2.54964E+00	0.00
2	1	L_1	1/128	1.05602E-02	2.15927E-02	1.03
2	1	L_1	1/256	5.22217E-03	1.05602E-02	1.02
2	1	L_1	1/512	2.59676E-03	5.22217E-03	1.01
2	1	L_1	1/1024	1.29482E-03	2.59676E-03	1.00
2	1	L_2	1/128	1.16683E-01	1.64475E-01	0.50
2	1	L_2	1/256	8.26120E-02	1.16683E-01	0.50
2	1	L_2	1/512	5.84471E-02	8.26120E-02	0.50
2	1	L_2	1/1024	4.13385E-02	5.84471E-02	0.50

Table 6.4: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.5) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and linear (deg $= 1$) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

$\mathbf{D}$	deg	norm	h	$\|\Delta^h \varphi_h - \Delta\varphi_h\|$	$\|\Delta^{2h} \varphi_{2h} - \Delta\varphi_{2h}\|$	p
3	2	L_∞	1/32	4.88845E-01	1.87290E+00	1.94
3	2	L_∞	1/64	2.34887E-01	4.88845E-01	1.06
3	2	L_∞	1/128	1.14431E-01	2.34887E-01	1.04
3	2	L_1	1/32	2.20173E-02	9.32968E-02	2.08
3	2	L_1	1/64	5.29526E-03	2.20173E-02	2.06
3	2	L_1	1/128	1.29560E-03	5.29526E-03	2.03
3	2	L_2	1/32	4.04914E-02	1.79846E-01	2.15
3	2	L_2	1/64	9.54041E-03	4.04914E-02	2.09
3	2	L_2	1/128	2.31020E-03	9.54041E-03	2.05

$\mathbf{D}$	deg	norm	h	$\|\Delta^{2h} \varphi_{2h} - P_2(\Delta^h \varphi_h)\|$	$\|\Delta^{4h} \varphi_{4h} - P_2(\Delta^{2h} \varphi_{2h})\|$	p
3	2	L_∞	1/32	1.95947E+00	7.07055E+00	1.85
3	2	L_∞	1/64	4.89721E-01	1.95947E+00	2.00
3	2	L_∞	1/128	2.35139E-01	4.89721E-01	1.06
3	2	L_1	1/32	7.16858E-02	3.21741E-01	2.17
3	2	L_1	1/64	1.67692E-02	7.16858E-02	2.10
3	2	L_1	1/128	4.00633E-03	1.67692E-02	2.07
3	2	L_2	1/32	1.39949E-01	6.72440E-01	2.26
3	2	L_2	1/64	3.12095E-02	1.39949E-01	2.16
3	2	L_2	1/128	7.28422E-03	3.12095E-02	2.10

Table 6.5: Convergence test of discrete Laplacian operator on Gaussian (6.4) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and biquadratic (deg $= 2$) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$\|\Delta^h \varphi_h - \Delta\varphi_h\|$	$\|\Delta^{2h}\varphi_{2h} - \Delta\varphi_{2h}\|$	p
3	2	L_∞	1/32	1.09348E-01	2.33653E-01	1.10
3	2	L_∞	1/64	5.20965E-02	1.09348E-01	1.07
3	2	L_∞	1/128	2.53218E-02	5.20965E-02	1.04
3	2	L_1	1/32	2.80423E-03	1.11387E-02	1.99
3	2	L_1	1/64	6.99859E-04	2.80423E-03	2.00
3	2	L_1	1/128	1.74598E-04	6.99859E-04	2.00
3	2	L_2	1/32	3.93319E-03	1.56701E-02	1.99
3	2	L_2	1/64	9.81668E-04	3.93319E-03	2.00
3	2	L_2	1/128	2.44991E-04	9.81668E-04	2.00

D	deg	norm	h	$\|\Delta^{2h}\varphi_{2h} - P_2(\Delta^h\varphi_h)\|$	$\|\Delta^{4h}\varphi_{4h} - P_2(\Delta^{2h}\varphi_{2h})\|$	p
3	2	L_∞	1/32	2.35071E-01	7.76380E-01	1.72
3	2	L_∞	1/64	1.09728E-01	2.35071E-01	1.10
3	2	L_∞	1/128	5.21943E-02	1.09728E-01	1.07
3	2	L_1	1/32	8.52419E-03	3.22905E-02	1.92
3	2	L_1	1/64	2.12814E-03	8.52419E-03	2.00
3	2	L_1	1/128	5.28239E-04	2.12814E-03	2.01
3	2	L_2	1/32	1.25655E-02	4.87623E-02	1.96
3	2	L_2	1/64	3.14705E-03	1.25655E-02	2.00
3	2	L_2	1/128	7.83819E-04	3.14705E-03	2.01

Table 6.6: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.6) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and biquadratic (deg $= 2$) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$\|\|\Delta^h\varphi_h - \Delta\varphi_h\|\|$	$\|\|\Delta^{2h}\varphi_{2h} - \Delta\varphi_{2h}\|\|$	p
3	1	L_∞	1/32	3.28267E+01	3.51650E+01	0.10
3	1	L_∞	1/64	3.11139E+01	3.28267E+01	0.08
3	1	L_∞	1/128	3.01457E+01	3.11139E+01	0.05
3	1	L_1	1/32	8.13179E-02	2.00611E-01	1.30
3	1	L_1	1/64	3.64497E-02	8.13179E-02	1.16
3	1	L_1	1/128	1.72252E-02	3.64497E-02	1.08
3	1	L_2	1/32	7.09253E-01	1.05989E+00	0.58
3	1	L_2	1/64	4.94085E-01	7.09253E-01	0.52
3	1	L_2	1/128	3.47614E-01	4.94085E-01	0.51

D	deg	norm	h	$\|\|\Delta^{2h}\varphi_{2h} - P_2(\Delta^h\varphi_h)\|\|$	$\|\|\Delta^{4h}\varphi_{4h} - P_2(\Delta^{2h}\varphi_{2h})\|\|$	p
3	1	L_∞	1/32	3.52516E+01	3.50928E+01	-0.01
3	1	L_∞	1/64	3.28432E+01	3.52516E+01	0.10
3	1	L_∞	1/128	3.11173E+01	3.28432E+01	0.08
3	1	L_1	1/32	1.79018E-01	4.85710E-01	1.44
3	1	L_1	1/64	7.60713E-02	1.79018E-01	1.23
3	1	L_1	1/128	3.51612E-02	7.60713E-02	1.11
3	1	L_2	1/32	1.05795E+00	1.81204E+00	0.78
3	1	L_2	1/64	7.09454E-01	1.05795E+00	0.58
3	1	L_2	1/128	4.94157E-01	7.09454E-01	0.52

Table 6.7: Convergence test of discrete Laplacian operator on Gaussian (6.4) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and bilinear (deg $= 1$) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	deg	norm	h	$\|\|\Delta^h\varphi_h - \Delta\varphi_h\|\|$	$\|\|\Delta^{2h}\varphi_{2h} - \Delta\varphi_{2h}\|\|$	p
3	1	L_∞	1/32	2.98076E+00	2.92347E+00	-0.03
3	1	L_∞	1/64	2.99518E+00	2.98076E+00	-0.01
3	1	L_∞	1/128	2.99880E+00	2.99518E+00	0.00
3	1	L_1	1/32	1.08923E-02	2.52229E-02	1.21
3	1	L_1	1/64	4.98999E-03	1.08923E-02	1.13
3	1	L_1	1/128	2.37785E-03	4.98999E-03	1.07
3	1	L_2	1/32	6.73793E-02	9.82547E-02	0.54
3	1	L_2	1/64	4.69739E-02	6.73793E-02	0.52
3	1	L_2	1/128	3.29774E-02	4.69739E-02	0.51

D	deg	norm	h	$\|\|\Delta^{2h}\varphi_{2h} - P_2(\Delta^h\varphi_h)\|\|$	$\|\|\Delta^{4h}\varphi_{4h} - P_2(\Delta^{2h}\varphi_{2h})\|\|$	p
3	1	L_∞	1/32	2.92584E+00	2.70966E+00	-0.11
3	1	L_∞	1/64	2.98136E+00	2.92584E+00	-0.03
3	1	L_∞	1/128	2.99533E+00	2.98136E+00	-0.01
3	1	L_1	1/32	2.26160E-02	5.25943E-02	1.22
3	1	L_1	1/64	1.02172E-02	2.26160E-02	1.15
3	1	L_1	1/128	4.81851E-03	1.02172E-02	1.08
3	1	L_2	1/32	9.79474E-02	1.49947E-01	0.61
3	1	L_2	1/64	6.73646E-02	9.79474E-02	0.54
3	1	L_2	1/128	4.69750E-02	6.73646E-02	0.52

Table 6.8: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.6) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and bilinear (deg $= 1$) coarse/fine interpolation. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by either (6.7) or (6.8). The other parameters of the convergence test are described in the text.

D	coarse/fine interpolation	operator error	on set of size	norm of operator error L_∞	L_1	L_2
2	quadratic, $O(h^3)$	$O(h)$	$O(h)$	$O(h)$	$O(h^2)$	$O(h^{3/2})$
2	linear, $O(h^2)$	$O(1)$	$O(h)$	$O(1)$	$O(h)$	$O(h^{1/2})$
3	biquadratic, $O(h^3)$	$O(h)$	$O(h^2)$	$O(h)$	$O(h^2)$	$O(h^2)$
3	bilinear, $O(h^2)$	$O(1)$	$O(h)$	$O(1)$	$O(h)$	$O(h^{1/2})$

Table 6.9: The error in the discrete Laplacian operator is $O(h^2)$, except where the operator is computed from interpolated data. At these points, two orders of accuracy are lost from the data by computing the discrete operator. This table shows the dimensionality $\mathbf{D}$, the error in the interpolated data points, the resulting error in the operator, and the volume of the set of points on which interpolated data with this order of accuracy are used — the operator error being $O(h^2)$ on the remaining points. The table also shows the norm of the error with the three norms.

These tables also show the advantage of quadratic over linear interpolation (in two dimensions) and biquadratic over bilinear interpolation (in three dimensions). For the linear/bilinear cases, Tables 6.3–6.4 and 6.7–6.8 show that the error in the discrete Laplacian operator is constant in L_∞ norm and converges linearly in L_1 norm. The L_2 norm of the error is $O(h^{1/2})$. These results are consistent with the operator error being $O(h^2)$ everywhere except on a set of codimension one on which the error is $O(1)$, as shown in Table 6.9.

6.3   Solver Tests, No-Geometry Case

We initialize the right-hand side ρ of the solver to be a function for which we can compute the analytical solution φ in Poisson's equation

$$\Delta\varphi = \rho.$$

We compute the solution at different base refinements and compare the solutions to the exact solutions. The boundary conditions of the computed solution are inhomogeneous Dirichlet. The value at the boundary is set to the analytic solution at that location. The difference between the computed and exact solutions should converge to zero with grid refinement, to second order if coarse/fine interpolation is (bi)quadratic, and to first order if (bi)linear. We also initialize φ with the exact solution and compare $\Delta^h\varphi$ with the exact ρ.

6.3.1   Polynomial example

We initialize the charge density ρ to be a function of r,

$$\rho(r) \;=\; \begin{cases} \rho_0(2(\frac{r}{R_0})^3 - 3(\frac{r}{R_0})^2 + 1), & \text{if } r < R_0; \\ 0, & \text{if } r \geq R_0 \;. \end{cases} \tag{6.9}$$

Then for the two-dimensional problem, we set boundary conditions for the exact solution

$$\varphi(r) \;=\; \begin{cases} \rho_0 r^2(\frac{1}{4} - \frac{3}{16}(\frac{r}{R_0})^2 + \frac{2}{25}(\frac{r}{R_0})^3), & \text{if } r < R_0; \\ \frac{3\rho_0 R_0^2}{20}(\ln(\frac{r}{R_0}) + \frac{19}{20}), & \text{if } r \geq R_0 \;. \end{cases} \tag{6.10}$$

For the three-dimensional problem, we set boundary conditions for the exact solution

$$\varphi(r) \;=\; \begin{cases} \rho_0 r^2(\frac{1}{6} - \frac{3}{20}(\frac{r}{R_0})^2 + \frac{1}{15}(\frac{r}{R_0})^3) & \text{if } r < R_0; \\ \rho_0 R_0^2(\frac{3}{20} - \frac{1}{15}\frac{R_0}{r}) & \text{if } r \geq R_0 \;. \end{cases} \tag{6.11}$$

We use the two–level grid hierarchies shown in Figures 6.1 and 6.2, on domains that have unit length in each direction and are centered at the origin. In (6.9), we set $R_0 = \frac{1}{2}$ and $\rho_0 = \frac{3}{4}$.

6.3.2   Trigonometric polynomial on semi-periodic domain

For examples on periodic domains, we choose the same grids in Figures 6.1 and 6.2, with periodicity in the x direction, meaning that the domain wraps around from $x = 1/2$ to $x = -1/2$.

We initialize the charge density ρ to be the trigonometric polynomial

$$\rho(r) = \sin(2\pi x) + \sin(2\pi y) \tag{6.12}$$

in two dimensions or

$$\rho(r) = \sin(2\pi x) + \sin(2\pi y) + \sin(2\pi z) \tag{6.13}$$

in three dimensions.

Then for the two-dimensional problem, we set boundary conditions for the exact solution

$$\varphi(x, y) = \frac{\sin(2\pi x) + \sin(2\pi y)}{-4\pi^2} \tag{6.14}$$

in two dimensions or

$$\varphi(x, y, z) = \frac{\sin(2\pi x) + \sin(2\pi y) + \sin(2\pi z)}{-4\pi^2} \tag{6.15}$$

in three dimensions.

6.3.3 Results

In Tables 6.10–6.17 we show the order of convergence of the solution U_h with mesh spacing h to the exact solution U_e:

$$p = \frac{1}{\ln(2)} \ln\left(\frac{||U_{2h} - U_e||}{||U_h - U_e||}\right). \tag{6.16}$$

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	L_∞	1/128	3.71549E-07	1.54797E-06	2.06
2	2	L_∞	1/256	9.36432E-08	3.71549E-07	1.99
2	2	L_∞	1/512	2.35204E-08	9.36432E-08	1.99
2	2	L_∞	1/1024	5.89332E-09	2.35204E-08	2.00
2	2	L_1	1/128	1.35213E-07	5.62979E-07	2.06
2	2	L_1	1/256	3.31677E-08	1.35213E-07	2.03
2	2	L_1	1/512	8.21767E-09	3.31677E-08	2.01
2	2	L_1	1/1024	2.04532E-09	8.21767E-09	2.01
2	2	L_2	1/128	1.66213E-07	6.91554E-07	2.06
2	2	L_2	1/256	4.08389E-08	1.66213E-07	2.03
2	2	L_2	1/512	1.01300E-08	4.08389E-08	2.01
2	2	L_2	1/1024	2.52297E-09	1.01300E-08	2.01

Table 6.10: Convergence test of Poisson solver on polynomial example with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and quadratic (deg = 2) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.10). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	L_∞	1/128	1.11486E-05	4.48981E-05	2.01
2	2	L_∞	1/256	2.77719E-06	1.11486E-05	2.01
2	2	L_∞	1/512	6.93039E-07	2.77719E-06	2.00
2	2	L_∞	1/1024	1.73098E-07	6.93039E-07	2.00
2	2	L_1	1/128	3.89967E-06	1.58442E-05	2.02
2	2	L_1	1/256	9.66897E-07	3.89967E-06	2.01
2	2	L_1	1/512	2.40699E-07	9.66897E-07	2.01
2	2	L_1	1/1024	6.00452E-08	2.40699E-07	2.00
2	2	L_2	1/128	4.90515E-06	1.98496E-05	2.02
2	2	L_2	1/256	1.21874E-06	4.90515E-06	2.01
2	2	L_2	1/512	3.03718E-07	1.21874E-06	2.00
2	2	L_2	1/1024	7.58069E-08	3.03718E-07	2.00

Table 6.11: Convergence test of Poisson solver on trigonometric polynomial with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and quadratic (deg = 2) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.14). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	L_∞	1/32	1.49905E-05	6.81452E-05	2.18
3	2	L_∞	1/64	3.54130E-06	1.49905E-05	2.08
3	2	L_∞	1/128	8.61632E-07	3.54130E-06	2.04
3	2	L_1	1/32	2.23346E-06	9.77178E-06	2.13
3	2	L_1	1/64	5.33136E-07	2.23346E-06	2.07
3	2	L_1	1/128	1.30171E-07	5.33136E-07	2.03
3	2	L_2	1/32	3.69778E-06	1.67465E-05	2.18
3	2	L_2	1/64	8.67701E-07	3.69778E-06	2.09
3	2	L_2	1/128	2.10033E-07	8.67701E-07	2.05

Table 6.12: Convergence test of Poisson solver on polynomial example with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and biquadratic (deg = 2) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.11). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	L_∞	1/32	1.91237E-04	7.68912E-04	2.01
3	2	L_∞	1/64	4.78046E-05	1.91237E-04	2.00
3	2	L_∞	1/128	1.19318E-05	4.78046E-05	2.00
3	2	L_1	1/32	4.50721E-05	1.86469E-04	2.05
3	2	L_1	1/64	1.10234E-05	4.50721E-05	2.03
3	2	L_1	1/128	2.72211E-06	1.10234E-05	2.02
3	2	L_2	1/32	5.99624E-05	2.47669E-04	2.05
3	2	L_2	1/64	1.47278E-05	5.99624E-05	2.03
3	2	L_2	1/128	3.64781E-06	1.47278E-05	2.01

Table 6.13: Convergence test of Poisson solver on trigonometric polynomial with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and biquadratic (deg = 2) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.15). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	1	L_∞	1/128	3.91186E-05	7.88447E-05	1.01
2	1	L_∞	1/256	1.94866E-05	3.91186E-05	1.01
2	1	L_∞	1/512	9.72555E-06	1.94866E-05	1.00
2	1	L_∞	1/1024	4.85859E-06	9.72555E-06	1.00
2	1	L_1	1/128	1.25249E-05	2.53626E-05	1.02
2	1	L_1	1/256	6.21855E-06	1.25249E-05	1.01
2	1	L_1	1/512	3.09799E-06	6.21855E-06	1.01
2	1	L_1	1/1024	1.54620E-06	3.09799E-06	1.00
2	1	L_2	1/128	1.63519E-05	3.28192E-05	1.01
2	1	L_2	1/256	8.15521E-06	1.63519E-05	1.00
2	1	L_2	1/512	4.07198E-06	8.15521E-06	1.00
2	1	L_2	1/1024	2.03461E-06	4.07198E-06	1.00

Table 6.14: Convergence test of Poisson solver on polynomial example with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and linear (deg $= 1$) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.10). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	1	L_∞	1/128	5.24215E-05	1.09359E-04	1.06
2	1	L_∞	1/256	2.56060E-05	5.24215E-05	1.03
2	1	L_∞	1/512	1.33853E-05	2.56060E-05	0.94
2	1	L_∞	1/1024	6.85865E-06	1.33853E-05	0.96
2	1	L_1	1/128	1.42906E-05	2.79043E-05	0.97
2	1	L_1	1/256	7.55767E-06	1.42906E-05	0.92
2	1	L_1	1/512	3.89860E-06	7.55767E-06	0.95
2	1	L_1	1/1024	1.98108E-06	3.89860E-06	0.98
2	1	L_2	1/128	1.80384E-05	3.53822E-05	0.97
2	1	L_2	1/256	9.51594E-06	1.80384E-05	0.92
2	1	L_2	1/512	4.92608E-06	9.51594E-06	0.95
2	1	L_2	1/1024	2.51000E-06	4.92608E-06	0.97

Table 6.15: Convergence test of Poisson solver on trigonometric polynomial with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and linear (deg $= 1$) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.14). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$||U_h - U_e||$	$||U_{2h} - U_e||$	p
3	1	L_∞	1/32	1.20183E-04	2.32466E-04	0.95
3	1	L_∞	1/64	6.21735E-05	1.20183E-04	0.95
3	1	L_∞	1/128	3.15018E-05	6.21735E-05	0.98
3	1	L_1	1/32	2.06800E-05	3.42476E-05	0.73
3	1	L_1	1/64	1.10046E-05	2.06800E-05	0.91
3	1	L_1	1/128	5.64018E-06	1.10046E-05	0.96
3	1	L_2	1/32	3.04811E-05	4.91818E-05	0.69
3	1	L_2	1/64	1.65058E-05	3.04811E-05	0.88
3	1	L_2	1/128	8.53615E-06	1.65058E-05	0.95

Table 6.16: Convergence test of Poisson solver on polynomial example with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and bilinear (deg $=$ 1) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.11). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	deg	norm	h	$||U_h - U_e||$	$||U_{2h} - U_e||$	p
3	1	L_∞	1/32	4.19926E-04	7.86925E-04	0.91
3	1	L_∞	1/64	2.43219E-04	4.19926E-04	0.79
3	1	L_∞	1/128	1.31656E-04	2.43219E-04	0.89
3	1	L_1	1/32	7.63277E-05	1.90602E-04	1.32
3	1	L_1	1/64	3.54366E-05	7.63277E-05	1.11
3	1	L_1	1/128	1.71978E-05	3.54366E-05	1.04
3	1	L_2	1/32	9.71736E-05	2.56948E-04	1.40
3	1	L_2	1/64	4.89655E-05	9.71736E-05	0.99
3	1	L_2	1/128	2.56318E-05	4.89655E-05	0.93

Table 6.17: Convergence test of Poisson solver on trigonometric polynomial with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and bilinear (deg $=$ 1) coarse/fine interpolation. We compare the solution at different refinements to U_e, the exact solution in (6.15). Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

$\mathbf{D}$	deg	norm	h	$\|\Delta^h U_e - \rho_h\|$	$\|\Delta^{2h} U_e - \rho_{2h}\|$	p
2	2	L_∞	1/128	2.45308E-03	4.75624E-03	0.96
2	2	L_∞	1/256	1.20846E-03	2.45308E-03	1.02
2	2	L_∞	1/512	5.99774E-04	1.20846E-03	1.01
2	2	L_∞	1/1024	2.98781E-04	5.99774E-04	1.01
2	2	L_1	1/128	1.75358E-05	7.26963E-05	2.05
2	2	L_1	1/256	4.30312E-06	1.75358E-05	2.03
2	2	L_1	1/512	1.06609E-06	4.30312E-06	2.01
2	2	L_1	1/1024	2.65295E-07	1.06609E-06	2.01
2	2	L_2	1/128	3.37694E-05	1.28541E-04	1.93
2	2	L_2	1/256	9.40996E-06	3.37694E-05	1.84
2	2	L_2	1/512	2.81870E-06	9.40996E-06	1.74
2	2	L_2	1/1024	8.94368E-07	2.81870E-06	1.66

Table 6.18: Convergence test of discrete Laplacian operator on polynomial example (6.10) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and quadratic (deg $= 2$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

Tables 6.10–6.13 show that the solution error converges quadratically to zero in all three norms when using quadratic or biquadratic coarse/fine interpolation. These results are consistent with uniform second-order convergence of the solution across the domain.

Tables 6.14–6.17 show that the solution error converges only linearly to zero in all three norms when using linear or bilinear coarse/fine interpolation. These results are consistent with uniform first-order convergence of the solution across the domain.

Tables 6.18–6.25 show the convergence of the error in the discrete Laplacian operator. The results for order of convergence agree with those for the examples in section 6.2. With quadratic/biquadratic interpolation, the operator error converges to first order in L_∞ norm and to second order in L_1 norm, while the L_2 norm of the error is $O(h^{3/2})$ in two dimensions and $O(h^2)$ in three dimensions. With linear/bilinear interpolation, the operator error is constant in L_∞ norm and converges to first order in L_1 norm, while the L_2 norm of the error is $O(h^{1/2})$. These results are consistent with the error analysis of Table 6.9.

D	deg	norm	h	$\|\|\Delta^h U_e - \rho_h\|\|$	$\|\|\Delta^{2h} U_e - \rho_{2h}\|\|$	p
2	2	L_∞	1/128	4.90485E-02	9.78139E-02	1.00
2	2	L_∞	1/256	2.45404E-02	4.90485E-02	1.00
2	2	L_∞	1/512	1.22718E-02	2.45404E-02	1.00
2	2	L_∞	1/1024	6.13602E-03	1.22718E-02	1.00
2	2	L_1	1/128	1.04271E-04	4.38903E-04	2.07
2	2	L_1	1/256	2.53882E-05	1.04271E-04	2.04
2	2	L_1	1/512	6.26122E-06	2.53882E-05	2.02
2	2	L_1	1/1024	1.55466E-06	6.26122E-06	2.01
2	2	L_2	1/128	3.84463E-04	1.52148E-03	1.98
2	2	L_2	1/256	9.86578E-05	3.84463E-04	1.96
2	2	L_2	1/512	2.59415E-05	9.86578E-05	1.93
2	2	L_2	1/1024	7.09558E-06	2.59415E-05	1.87

Table 6.19: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.14) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and quadratic (deg $= 2$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$\|\|\Delta^h U_e - \rho_h\|\|$	$\|\|\Delta^{2h} U_e - \rho_{2h}\|\|$	p
3	2	L_∞	1/32	5.01139E-03	9.33079E-03	0.90
3	2	L_∞	1/64	2.59449E-03	5.01139E-03	0.95
3	2	L_∞	1/128	1.27999E-03	2.59449E-03	1.02
3	2	L_1	1/32	2.68607E-04	1.09871E-03	2.03
3	2	L_1	1/64	6.63975E-05	2.68607E-04	2.02
3	2	L_1	1/128	1.65126E-05	6.63975E-05	2.01
3	2	L_2	1/32	3.84822E-04	1.58720E-03	2.04
3	2	L_2	1/64	9.46695E-05	3.84822E-04	2.02
3	2	L_2	1/128	2.35043E-05	9.46695E-05	2.01

Table 6.20: Convergence test of discrete Laplacian operator on polynomial example (6.11) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and bi-quadratic (deg $= 2$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$\|\|\Delta^h U_e - \rho_h\|\|$	$\|\|\Delta^{2h} U_e - \rho_{2h}\|\|$	p
3	2	L_∞	1/32	1.09348E-01	2.33653E-01	1.10
3	2	L_∞	1/64	5.20965E-02	1.09348E-01	1.07
3	2	L_∞	1/128	2.53218E-02	5.20965E-02	1.04
3	2	L_1	1/32	2.80423E-03	1.11387E-02	1.99
3	2	L_1	1/64	6.99859E-04	2.80423E-03	2.00
3	2	L_1	1/128	1.74598E-04	6.99859E-04	2.00
3	2	L_2	1/32	3.93319E-03	1.56701E-02	1.99
3	2	L_2	1/64	9.81668E-04	3.93319E-03	2.00
3	2	L_2	1/128	2.44991E-04	9.81668E-04	2.00

Table 6.21: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.15) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and biquadratic (deg $= 2$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$\|\|\Delta^h U_e - \rho_h\|\|$	$\|\|\Delta^{2h} U_e - \rho_{2h}\|\|$	p
2	1	L_∞	1/128	6.88865E-01	6.88272E-01	0.00
2	1	L_∞	1/256	6.89013E-01	6.88865E-01	0.00
2	1	L_∞	1/512	6.89050E-01	6.89013E-01	0.00
2	1	L_∞	1/1024	6.89059E-01	6.89050E-01	0.00
2	1	L_1	1/128	1.02557E-03	2.08522E-03	1.02
2	1	L_1	1/256	5.08584E-04	1.02557E-03	1.01
2	1	L_1	1/512	2.53243E-04	5.08584E-04	1.01
2	1	L_1	1/1024	1.26362E-04	2.53243E-04	1.00
2	1	L_2	1/128	1.91226E-02	2.71016E-02	0.50
2	1	L_2	1/256	1.35074E-02	1.91226E-02	0.50
2	1	L_2	1/512	9.54612E-03	1.35074E-02	0.50
2	1	L_2	1/1024	6.74834E-03	9.54612E-03	0.50

Table 6.22: Convergence test of discrete Laplacian operator on polynomial example (6.10) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, and linear (deg $= 1$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$||\Delta^h U_e - \rho_h||$	$||\Delta^{2h} U_e - \rho_{2h}||$	p
2	1	L_∞	1/128	2.53827E+00	2.51451E+00	-0.01
2	1	L_∞	1/256	2.54964E+00	2.53827E+00	-0.01
2	1	L_∞	1/512	2.55519E+00	2.54964E+00	0.00
2	1	L_∞	1/1024	2.55794E+00	2.55519E+00	0.00
2	1	L_1	1/128	5.24667E-03	1.06590E-02	1.02
2	1	L_1	1/256	2.60286E-03	5.24667E-03	1.01
2	1	L_1	1/512	1.29634E-03	2.60286E-03	1.01
2	1	L_1	1/1024	6.46904E-04	1.29634E-03	1.00
2	1	L_2	1/128	8.26120E-02	1.16683E-01	0.50
2	1	L_2	1/256	5.84471E-02	8.26120E-02	0.50
2	1	L_2	1/512	4.13385E-02	5.84471E-02	0.50
2	1	L_2	1/1024	2.92342E-02	4.13385E-02	0.50

Table 6.23: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.14) with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.1, periodic in the x direction, and linear (deg $= 1$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$||\Delta^h U_e - \rho_h||$	$||\Delta^{2h} U_e - \rho_{2h}||$	p
3	1	L_∞	1/32	3.99153E-01	3.95059E-01	-0.01
3	1	L_∞	1/64	4.24970E-01	3.99153E-01	-0.09
3	1	L_∞	1/128	4.37564E-01	4.24970E-01	-0.04
3	1	L_1	1/32	1.41976E-03	3.27223E-03	1.20
3	1	L_1	1/64	6.54851E-04	1.41976E-03	1.12
3	1	L_1	1/128	3.13588E-04	6.54851E-04	1.06
3	1	L_2	1/32	1.14019E-02	1.74821E-02	0.62
3	1	L_2	1/64	7.75064E-03	1.14019E-02	0.56
3	1	L_2	1/128	5.37544E-03	7.75064E-03	0.53

Table 6.24: Convergence test of discrete Laplacian operator on polynomial example (6.11) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, and bilinear (deg $= 1$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

D	deg	norm	h	$\lVert \Delta^h U_e - \rho_h \rVert$	$\lVert \Delta^{2h} U_e - \rho_{2h} \rVert$	p
3	1	L_∞	1/32	2.98076E+00	2.92347E+00	-0.03
3	1	L_∞	1/64	2.99518E+00	2.98076E+00	-0.01
3	1	L_∞	1/128	2.99880E+00	2.99518E+00	0.00
3	1	L_1	1/32	1.08923E-02	2.52229E-02	1.21
3	1	L_1	1/64	4.98999E-03	1.08923E-02	1.13
3	1	L_1	1/128	2.37785E-03	4.98999E-03	1.07
3	1	L_2	1/32	6.73793E-02	9.82547E-02	0.54
3	1	L_2	1/64	4.69739E-02	6.73793E-02	0.52
3	1	L_2	1/128	3.29774E-02	4.69739E-02	0.51

Table 6.25: Convergence test of discrete Laplacian operator on trigonometric polynomial (6.15) with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy shown in Figure 6.2, periodic in the x direction, and bilinear (deg $= 1$) coarse/fine interpolation. We compare the discrete operator evaluated at different refinements of U_e, the exact solution, to ρ, the right-hand side. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.7).

6.4   Solver Tests, With Geometry

In our tests of convergence on a non-rectangular domain, we use the following physical example. The domain lies between a cube and a sphere nested inside and concentric with it (or a square and a circle in two dimensions), as illustrated in Figure 6.3. We solve for the field within this domain due to a point charge lying inside the spherical region, from Laplace's equation

$$\Delta \varphi = 0$$

with inhomogeneous Dirichlet boundary conditions for φ enforced on the sphere and on the faces of the cube.

Assuming a coordinate system with the surrounding cube having unit length in each dimension and centered at the origin, the point charge is located at $\mathbf{q} = (0.02, -0.05, -0.01)$ in the three-dimensional problem, or $\mathbf{q} = (0.02, -0.05)$ in the two-dimensional problem. The radius of the sphere is $r = 1/\sqrt{35} = 0.169$. We can compare our answers with the exact solutions,

$$\varphi(\mathbf{x}) = \log \lVert \mathbf{x} - \mathbf{q} \rVert_2 \tag{6.17}$$

in two dimensions, or

$$\varphi(\mathbf{x}) = \frac{1}{\lVert \mathbf{x} - \mathbf{q} \rVert_2} \tag{6.18}$$
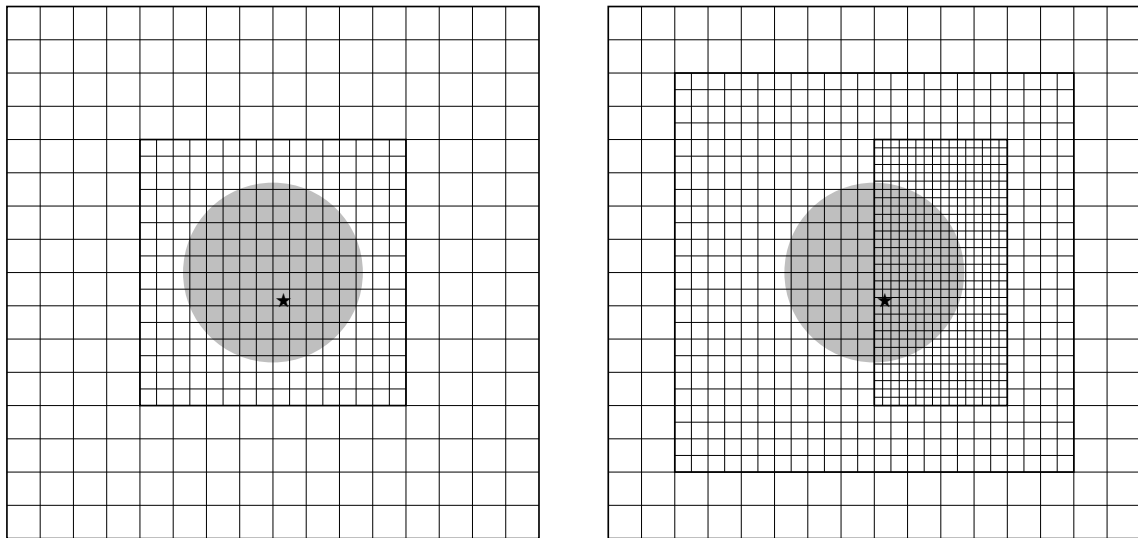
Figure 6.3: In our examples, we solve for the field outside the shaded region due to a point charge at the position marked $\star$. We solve on nested hierarchies of grids in the configurations shown in two dimensions with either two levels (left) or three levels (right) of refinement. The three-dimensional grid configurations are such that their projections on two of the three axis planes are exactly the two-dimensional grid configurations shown here. We also use fully refined versions of these same sets of grids, partitioned so that the maximum length of any grid in any dimension is 32.

in three dimensions.

In Tables 6.26–6.29 we show the solution error on the two-level or three-level grid hierarchies in two or three dimensions, with (bi)quadratic interpolation at coarse/fine interfaces, and either the Shortley–Weller approximation or linear extrapolation to compute the derivative near internal boundaries, as discussed in section 2.2. The solution error converges quadratically to zero in all three norms. These results are consistent with uniform second-order convergence of the solution across the domain.

In Tables 6.30–6.33 we show the solution error on the same grid hierarchies, but with (bi)linear interpolation at coarse/fine interfaces. The solution error converges linearly to zero in all three norms. These results are consistent with uniform first-order convergence of the solution across the domain.

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	2	S–W	L_∞	1/32	9.73844E-04	4.44521E-03	2.19
2	2	2	S–W	L_∞	1/64	2.35225E-04	9.73844E-04	2.05
2	2	2	S–W	L_∞	1/128	5.76281E-05	2.35225E-04	2.03
2	2	2	S–W	L_∞	1/256	1.42413E-05	5.76281E-05	2.02
2	2	2	S–W	L_∞	1/512	3.54359E-06	1.42413E-05	2.01
2	2	2	S–W	L_1	1/32	1.40614E-04	6.11903E-04	2.12
2	2	2	S–W	L_1	1/64	3.36654E-05	1.40614E-04	2.06
2	2	2	S–W	L_1	1/128	8.27132E-06	3.36654E-05	2.03
2	2	2	S–W	L_1	1/256	2.04687E-06	8.27132E-06	2.01
2	2	2	S–W	L_1	1/512	5.09495E-07	2.04687E-06	2.01
2	2	2	S–W	L_2	1/32	2.07108E-04	9.27218E-04	2.16
2	2	2	S–W	L_2	1/64	4.93872E-05	2.07108E-04	2.07
2	2	2	S–W	L_2	1/128	1.21237E-05	4.93872E-05	2.03
2	2	2	S–W	L_2	1/256	2.99974E-06	1.21237E-05	2.01
2	2	2	S–W	L_2	1/512	7.46776E-07	2.99974E-06	2.01

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	2	linear	L_∞	1/32	1.57254E-03	3.40825E-03	1.12
2	2	2	linear	L_∞	1/64	4.57024E-04	1.57254E-03	1.78
2	2	2	linear	L_∞	1/128	1.09987E-04	4.57024E-04	2.05
2	2	2	linear	L_∞	1/256	3.04461E-05	1.09987E-04	1.85
2	2	2	linear	L_∞	1/512	7.69995E-06	3.04461E-05	1.98
2	2	2	linear	L_1	1/32	1.53771E-04	6.28844E-04	2.03
2	2	2	linear	L_1	1/64	3.69547E-05	1.53771E-04	2.06
2	2	2	linear	L_1	1/128	8.71767E-06	3.69547E-05	2.08
2	2	2	linear	L_1	1/256	2.21476E-06	8.71767E-06	1.98
2	2	2	linear	L_1	1/512	5.37367E-07	2.21476E-06	2.04
2	2	2	linear	L_2	1/32	2.15568E-04	8.77324E-04	2.02
2	2	2	linear	L_2	1/64	5.19275E-05	2.15568E-04	2.05
2	2	2	linear	L_2	1/128	1.20342E-05	5.19275E-05	2.11
2	2	2	linear	L_2	1/256	3.09112E-06	1.20342E-05	1.96
2	2	2	linear	L_2	1/512	7.42570E-07	3.09112E-06	2.06

Table 6.26: Convergence test of solution from Poisson solver with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.3, quadratic (deg = 2) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	3	2	S–W	L_∞	1/32	2.90628E-04	1.31271E-03	2.18
2	3	2	S–W	L_∞	1/64	6.90617E-05	2.90628E-04	2.07
2	3	2	S–W	L_∞	1/128	1.68905E-05	6.90617E-05	2.03
2	3	2	S–W	L_∞	1/256	4.16657E-06	1.68905E-05	2.02
2	3	2	S–W	L_∞	1/512	1.03590E-06	4.16657E-06	2.01
2	3	2	S–W	L_1	1/32	6.24714E-05	2.70186E-04	2.11
2	3	2	S–W	L_1	1/64	1.48516E-05	6.24714E-05	2.07
2	3	2	S–W	L_1	1/128	3.63267E-06	1.48516E-05	2.03
2	3	2	S–W	L_1	1/256	8.97397E-07	3.63267E-06	2.02
2	3	2	S–W	L_1	1/512	2.23125E-07	8.97397E-07	2.01
2	3	2	S–W	L_2	1/32	8.55212E-05	3.73703E-04	2.13
2	3	2	S–W	L_2	1/64	2.03233E-05	8.55212E-05	2.07
2	3	2	S–W	L_2	1/128	4.98116E-06	2.03233E-05	2.03
2	3	2	S–W	L_2	1/256	1.23206E-06	4.98116E-06	2.02
2	3	2	S–W	L_2	1/512	3.06585E-07	1.23206E-06	2.01

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	3	2	linear	L_∞	1/32	1.07990E-03	3.46799E-03	1.68
2	3	2	linear	L_∞	1/64	3.73431E-04	1.07990E-03	1.53
2	3	2	linear	L_∞	1/128	7.07797E-05	3.73431E-04	2.40
2	3	2	linear	L_∞	1/256	2.70732E-05	7.07797E-05	1.39
2	3	2	linear	L_∞	1/512	6.11590E-06	2.70732E-05	2.15
2	3	2	linear	L_1	1/32	8.00561E-05	3.32044E-04	2.05
2	3	2	linear	L_1	1/64	1.92086E-05	8.00561E-05	2.06
2	3	2	linear	L_1	1/128	4.41078E-06	1.92086E-05	2.12
2	3	2	linear	L_1	1/256	1.13220E-06	4.41078E-06	1.96
2	3	2	linear	L_1	1/512	2.68771E-07	1.13220E-06	2.07
2	3	2	linear	L_2	1/32	1.16620E-04	4.67352E-04	2.00
2	3	2	linear	L_2	1/64	2.83191E-05	1.16620E-04	2.04
2	3	2	linear	L_2	1/128	6.31035E-06	2.83191E-05	2.17
2	3	2	linear	L_2	1/256	1.65507E-06	6.31035E-06	1.93
2	3	2	linear	L_2	1/512	3.85755E-07	1.65507E-06	2.10

Table 6.27: Convergence test of solution from Poisson solver with the two-dimensional ($\mathbf{D} = 2$) three-level grid hierarchy shown in Figure 6.3, quadratic (deg = 2) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	2	S–W	L_∞	1/16	6.14956E-02	1.36680E-01	1.15
3	2	2	S–W	L_∞	1/32	1.28625E-02	6.14956E-02	2.26
3	2	2	S–W	L_∞	1/64	2.95191E-03	1.28625E-02	2.12
3	2	2	S–W	L_1	1/16	2.11123E-03	9.18361E-03	2.12
3	2	2	S–W	L_1	1/32	4.81696E-04	2.11123E-03	2.13
3	2	2	S–W	L_1	1/64	1.15904E-04	4.81696E-04	2.06
3	2	2	S–W	L_2	1/16	4.59751E-03	1.73702E-02	1.92
3	2	2	S–W	L_2	1/32	1.00240E-03	4.59751E-03	2.20
3	2	2	S–W	L_2	1/64	2.39005E-04	1.00240E-03	2.07

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	2	linear	L_∞	1/16	4.55179E-02	1.88341E-01	2.05
3	2	2	linear	L_∞	1/32	2.46091E-02	4.55179E-02	0.89
3	2	2	linear	L_∞	1/64	7.13979E-03	2.46091E-02	1.79
3	2	2	linear	L_1	1/16	2.30059E-03	1.30019E-02	2.50
3	2	2	linear	L_1	1/32	5.65412E-04	2.30059E-03	2.02
3	2	2	linear	L_1	1/64	1.38440E-04	5.65412E-04	2.03
3	2	2	linear	L_2	1/16	4.07855E-03	2.19075E-02	2.43
3	2	2	linear	L_2	1/32	9.57652E-04	4.07855E-03	2.09
3	2	2	linear	L_2	1/64	2.32183E-04	9.57652E-04	2.04

Table 6.28: Convergence test of solution from Poisson solver with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy described in Figure 6.3, biquadratic (deg = 2) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|\|U_h - U_e\|\|$	$\|\|U_{2h} - U_e\|\|$	p
3	3	2	S–W	L_∞	1/16	2.65224E-02	7.29967E-02	1.46
3	3	2	S–W	L_∞	1/32	8.27441E-03	2.65224E-02	1.68
3	3	2	S–W	L_∞	1/64	2.24033E-03	8.27441E-03	1.88
3	3	2	S–W	L_1	1/16	9.67464E-04	4.47553E-03	2.21
3	3	2	S–W	L_1	1/32	2.21810E-04	9.67464E-04	2.12
3	3	2	S–W	L_1	1/64	5.29570E-05	2.21810E-04	2.07
3	3	2	S–W	L_2	1/16	1.71964E-03	8.56805E-03	2.32
3	3	2	S–W	L_2	1/32	3.91017E-04	1.71964E-03	2.14
3	3	2	S–W	L_2	1/64	9.37494E-05	3.91017E-04	2.06

D	levels	deg	deriv	norm	h	$\|\|U_h - U_e\|\|$	$\|\|U_{2h} - U_e\|\|$	p
3	3	2	linear	L_∞	1/16	4.96743E-02	1.63454E-01	1.72
3	3	2	linear	L_∞	1/32	1.89876E-02	4.96743E-02	1.39
3	3	2	linear	L_∞	1/64	6.00576E-03	1.89876E-02	1.66
3	3	2	linear	L_1	1/16	1.26120E-03	6.37969E-03	2.34
3	3	2	linear	L_1	1/32	3.13440E-04	1.26120E-03	2.01
3	3	2	linear	L_1	1/64	7.57937E-05	3.13440E-04	2.05
3	3	2	linear	L_2	1/16	2.10255E-03	1.12886E-02	2.42
3	3	2	linear	L_2	1/32	5.51658E-04	2.10255E-03	1.93
3	3	2	linear	L_2	1/64	1.32848E-04	5.51658E-04	2.05

Table 6.29: Convergence test of solution from Poisson solver with the three-dimensional ($\mathbf{D} = 3$) three-level grid hierarchy described in Figure 6.3, biquadratic (deg = 2) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	1	S–W	L_∞	1/32	1.93841E-03	4.10973E-03	1.08
2	2	1	S–W	L_∞	1/64	1.04052E-03	1.93841E-03	0.90
2	2	1	S–W	L_∞	1/128	5.50668E-04	1.04052E-03	0.92
2	2	1	S–W	L_∞	1/256	2.83014E-04	5.50668E-04	0.96
2	2	1	S–W	L_∞	1/512	1.43334E-04	2.83014E-04	0.98
2	2	1	S–W	L_1	1/32	3.64472E-04	7.86963E-04	1.11
2	2	1	S–W	L_1	1/64	2.08342E-04	3.64472E-04	0.81
2	2	1	S–W	L_1	1/128	1.09886E-04	2.08342E-04	0.92
2	2	1	S–W	L_1	1/256	5.62656E-05	1.09886E-04	0.97
2	2	1	S–W	L_1	1/512	2.84493E-05	5.62656E-05	0.98
2	2	1	S–W	L_2	1/32	5.13819E-04	1.12287E-03	1.13
2	2	1	S–W	L_2	1/64	2.79849E-04	5.13819E-04	0.88
2	2	1	S–W	L_2	1/128	1.48011E-04	2.79849E-04	0.92
2	2	1	S–W	L_2	1/256	7.61958E-05	1.48011E-04	0.96
2	2	1	S–W	L_2	1/512	3.86595E-05	7.61958E-05	0.98

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	2	1	linear	L_∞	1/32	2.46572E-03	5.16868E-03	1.07
2	2	1	linear	L_∞	1/64	1.19729E-03	2.46572E-03	1.04
2	2	1	linear	L_∞	1/128	5.81155E-04	1.19729E-03	1.04
2	2	1	linear	L_∞	1/256	2.91844E-04	5.81155E-04	0.99
2	2	1	linear	L_∞	1/512	1.45185E-04	2.91844E-04	1.01
2	2	1	linear	L_1	1/32	4.72402E-04	9.47369E-04	1.00
2	2	1	linear	L_1	1/64	2.35521E-04	4.72402E-04	1.00
2	2	1	linear	L_1	1/128	1.15486E-04	2.35521E-04	1.03
2	2	1	linear	L_1	1/256	5.78447E-05	1.15486E-04	1.00
2	2	1	linear	L_1	1/512	2.87967E-05	5.78447E-05	1.01
2	2	1	linear	L_2	1/32	6.52228E-04	1.40974E-03	1.11
2	2	1	linear	L_2	1/64	3.16475E-04	6.52228E-04	1.04
2	2	1	linear	L_2	1/128	1.55525E-04	3.16475E-04	1.02
2	2	1	linear	L_2	1/256	7.83115E-05	1.55525E-04	0.99
2	2	1	linear	L_2	1/512	3.91222E-05	7.83115E-05	1.00

Table 6.30: Convergence test of solution from Poisson solver with the two-dimensional ($\mathbf{D} = 2$) two-level grid hierarchy shown in Figure 6.3, linear (deg = 1) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	3	1	S–W	L_∞	1/32	1.42850E-03	2.99625E-03	1.07
2	3	1	S–W	L_∞	1/64	7.10728E-04	1.42850E-03	1.01
2	3	1	S–W	L_∞	1/128	3.52346E-04	7.10728E-04	1.01
2	3	1	S–W	L_∞	1/256	1.75116E-04	3.52346E-04	1.01
2	3	1	S–W	L_∞	1/512	8.72580E-05	1.75116E-04	1.00
2	3	1	S–W	L_1	1/32	4.56828E-04	9.55886E-04	1.07
2	3	1	S–W	L_1	1/64	2.20841E-04	4.56828E-04	1.05
2	3	1	S–W	L_1	1/128	1.08038E-04	2.20841E-04	1.03
2	3	1	S–W	L_1	1/256	5.33268E-05	1.08038E-04	1.02
2	3	1	S–W	L_1	1/512	2.64886E-05	5.33268E-05	1.01
2	3	1	S–W	L_2	1/32	5.74100E-04	1.21862E-03	1.09
2	3	1	S–W	L_2	1/64	2.80744E-04	5.74100E-04	1.03
2	3	1	S–W	L_2	1/128	1.38692E-04	2.80744E-04	1.02
2	3	1	S–W	L_2	1/256	6.88673E-05	1.38692E-04	1.01
2	3	1	S–W	L_2	1/512	3.43191E-05	6.88673E-05	1.00

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
2	3	1	linear	L_∞	1/32	1.48681E-03	4.63644E-03	1.64
2	3	1	linear	L_∞	1/64	7.20245E-04	1.48681E-03	1.05
2	3	1	linear	L_∞	1/128	3.54764E-04	7.20245E-04	1.02
2	3	1	linear	L_∞	1/256	1.75666E-04	3.54764E-04	1.01
2	3	1	linear	L_∞	1/512	8.73932E-05	1.75666E-04	1.01
2	3	1	linear	L_1	1/32	5.17284E-04	1.20203E-03	1.22
2	3	1	linear	L_1	1/64	2.35301E-04	5.17284E-04	1.14
2	3	1	linear	L_1	1/128	1.11144E-04	2.35301E-04	1.08
2	3	1	linear	L_1	1/256	5.41408E-05	1.11144E-04	1.04
2	3	1	linear	L_1	1/512	2.66707E-05	5.41408E-05	1.02
2	3	1	linear	L_2	1/32	6.38078E-04	1.49702E-03	1.23
2	3	1	linear	L_2	1/64	2.95370E-04	6.38078E-04	1.11
2	3	1	linear	L_2	1/128	1.41811E-04	2.95370E-04	1.06
2	3	1	linear	L_2	1/256	6.96679E-05	1.41811E-04	1.03
2	3	1	linear	L_2	1/512	3.44977E-05	6.96679E-05	1.01

Table 6.31: Convergence test of solution from Poisson solver with the two-dimensional ($\mathbf{D} = 2$) three-level grid hierarchy shown in Figure 6.3, linear (deg = 1) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	1	S–W	L_∞	1/16	5.00943E-02	1.34004E-01	1.42
3	2	1	S–W	L_∞	1/32	1.38932E-02	5.00943E-02	1.85
3	2	1	S–W	L_∞	1/64	8.05347E-03	1.38932E-02	0.79
3	2	1	S–W	L_1	1/16	3.33804E-03	1.10461E-02	1.73
3	2	1	S–W	L_1	1/32	1.73776E-03	3.33804E-03	0.94
3	2	1	S–W	L_1	1/64	9.81944E-04	1.73776E-03	0.82
3	2	1	S–W	L_2	1/16	5.43462E-03	1.99433E-02	1.88
3	2	1	S–W	L_2	1/32	2.64239E-03	5.43462E-03	1.04
3	2	1	S–W	L_2	1/64	1.44245E-03	2.64239E-03	0.87

D	levels	deg	deriv	norm	h	$\|U_h - U_e\|$	$\|U_{2h} - U_e\|$	p
3	2	1	linear	L_∞	1/16	4.85218E-02	2.10950E-01	2.12
3	2	1	linear	L_∞	1/32	2.67532E-02	4.85218E-02	0.86
3	2	1	linear	L_∞	1/64	9.48032E-03	2.67532E-02	1.50
3	2	1	linear	L_1	1/16	4.25807E-03	1.53695E-02	1.85
3	2	1	linear	L_1	1/32	2.17625E-03	4.25807E-03	0.97
3	2	1	linear	L_1	1/64	1.09445E-03	2.17625E-03	0.99
3	2	1	linear	L_2	1/16	6.95016E-03	2.71364E-02	1.97
3	2	1	linear	L_2	1/32	3.32538E-03	6.95016E-03	1.06
3	2	1	linear	L_2	1/64	1.61891E-03	3.32538E-03	1.04

Table 6.32: Convergence test of solution from Poisson solver with the three-dimensional ($\mathbf{D} = 3$) two-level grid hierarchy described in Figure 6.3, bilinear (deg = 1) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

D	levels	deg	deriv	norm	h	$\|\|U_h - U_e\|\|$	$\|\|U_{2h} - U_e\|\|$	p
3	3	1	S–W	L_∞	1/16	5.23181E-02	6.12688E-02	0.23
3	3	1	S–W	L_∞	1/32	1.33302E-02	5.23181E-02	1.97
3	3	1	S–W	L_∞	1/64	5.07895E-03	1.33302E-02	1.39
3	3	1	S–W	L_1	1/16	3.75659E-03	8.49343E-03	1.18
3	3	1	S–W	L_1	1/32	1.79256E-03	3.75659E-03	1.07
3	3	1	S–W	L_1	1/64	8.66731E-04	1.79256E-03	1.05
3	3	1	S–W	L_2	1/16	5.00354E-03	1.18890E-02	1.25
3	3	1	S–W	L_2	1/32	2.35013E-03	5.00354E-03	1.09
3	3	1	S–W	L_2	1/64	1.14896E-03	2.35013E-03	1.03

D	levels	deg	deriv	norm	h	$\|\|U_h - U_e\|\|$	$\|\|U_{2h} - U_e\|\|$	p
3	3	1	linear	L_∞	1/16	8.05650E-02	1.72357E-01	1.10
3	3	1	linear	L_∞	1/32	2.22017E-02	8.05650E-02	1.86
3	3	1	linear	L_∞	1/64	7.74776E-03	2.22017E-02	1.52
3	3	1	linear	L_1	1/16	4.69344E-03	1.25343E-02	1.42
3	3	1	linear	L_1	1/32	2.04888E-03	4.69344E-03	1.20
3	3	1	linear	L_1	1/64	9.28839E-04	2.04888E-03	1.14
3	3	1	linear	L_2	1/16	6.27555E-03	1.83465E-02	1.55
3	3	1	linear	L_2	1/32	2.67609E-03	6.27555E-03	1.23
3	3	1	linear	L_2	1/64	1.22368E-03	2.67609E-03	1.13

Table 6.33: Convergence test of solution from Poisson solver with the three-dimensional ($\mathbf{D} = 3$) three-level grid hierarchy described in Figure 6.3, bilinear (deg $= 1$) coarse/fine interpolation, and either (top) Shortley–Weller approximation or (bottom) linear extrapolation for derivatives at internal boundaries. We compare the solution at different refinements to U_e, the exact solution. Here h denotes the grid spacing at the base level, and p is the order of convergence of the scheme calculated by (6.16).

Bibliography

[BBC$^+$94] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, Philadelphia, PA, 1994.