Summary of d-Dimensional Operators

- Review d-dimensional interpolation/integration/differentiation for d=1, 2, 3.

Assume:

- u, $v \in X^N$, the basic interpolation space on $\hat{\Omega} := [-1,1]^d$, with no boundary-condition restrictions.
- X^N is spanned by tensor-product polynomials of degree N.
- Basis functions in each direction are Lagrange polynomials based on the GLL points, ξ_j, $j = 0, \ldots, N$, on $[-1,1]$.

Interpolation Summary

$$\text{1D:} \quad u(x) = \sum_{j=0}^{N} l_j(x)\, u_j \tag{1}$$

$$\text{2D:} \quad u(\mathbf{x}) = \sum_{j=0}^{N} \sum_{i=0}^{N} l_j(y)\, l_i(x) u_{ij} \tag{2}$$

$$\text{3D:} \quad u(\mathbf{x}) = \sum_{k=0}^{N} \sum_{j=0}^{N} \sum_{i=0}^{N} l_k(z) l_j(x)\, l_i(x) u_{ijk} \tag{3}$$

Quadrature Summary

In variational-based numerical methods, we frequently need to evaluate the L^2 inner-product,

$$(v, u) \;:=\; \int_\Omega v(\mathbf{x})\, u(\mathbf{x})\, dV. \tag{4}$$

We define the discrete GLL equivalent to (4) as

$$\text{1D:}\;\; (v,u)_N \;=\; \sum_{k=0}^{N} v_k \rho_k u_k \;=\; \underline{v}^T \bar{B} \underline{u} \tag{5}$$

$$\text{2D:}\;\; (v,u)_N \;=\; \sum_{l=0}^{N}\sum_{k=0}^{N} v_{kl} \rho_k \rho_l u_{kl} \;=\; \underline{v}^T \bar{B} \underline{u} \tag{6}$$

$$\text{3D:}\;\; (v,u)_N \;=\; \sum_{m=0}^{N}\sum_{l=0}^{N}\sum_{k=0}^{N} v_{klm} \rho_k \rho_l \rho_m u_{klm} \;=\; \underline{v}^T \bar{B} \underline{u}, \tag{7}$$

where the mass matrix in each case is defined as a tensor-product of the 1D mass matrices comprising the GLL quadrature weights. With $\hat{B}$:=diag(ρ_k), the d-dimensional mass matrices are

$$\text{1D:}\;\; \bar{B} \;=\; \hat{B} \tag{8}$$

$$\text{2D:}\;\; \bar{B} \;=\; \hat{B} \otimes \hat{B} \tag{9}$$

$$\text{3D:}\;\; \bar{B} \;=\; \hat{B} \otimes \hat{B} \otimes \hat{B}, \tag{10}$$

which are diagonal. As an example, the diagonal mass matrix in 2D is

$$\bar{B} \;:=\; \hat{B} \otimes \hat{B} \;=\; \begin{bmatrix} \rho_0\rho_0 & & & \\ & \rho_1\rho_0 & & \\ & & \ddots & \\ & & & \rho_N\rho_N \end{bmatrix}. \tag{11}$$

“Consistent” Quadrature

- For $v, u \in X^N$, the integrand in (4) is a polynomial of degree $2N$. Gauss-Lobatto-Legendre quadrature (GLL) on $(N+1)$ points is exact for integrands in $\mathbb{P}_{2N-1}$.
- *Gauss-Legendre* (GL) quadrature on $(N+1)$ points, however, is exact for integrands in $\mathbb{P}_{2N+1}$.
- Let η_k, $k = 0, \ldots, N$ be the GL points on [-1,1] and ω_k be the corresponding quadrature weights.
- Define the interpolation matrix that maps from the GLL to the GL points,

$$\tilde{J}_{kj} \;:=\; l_j(\eta^k), \tag{12}$$

  and the corresponding mass matrix B^M :=diag(ω_k).
- If $\underline{\tilde{u}} := \tilde{J}\underline{u}$ and $\underline{\tilde{v}} := \tilde{J}\underline{v}$, the L^2 inner product in 1D is

$$(v, u) \;=\; \underline{\tilde{v}}^T B^M \underline{\tilde{u}} \;=\; \underline{v}^T \tilde{B} \underline{u}, \tag{13}$$

  where $\tilde{B} := \tilde{J}^T B^M \tilde{J}$.
- The form (13) carries over to all space dimensions, with the consistent (or exact, or full) mass matrix given by

$$\text{1D: } \bar{B} \;=\; \tilde{B} \tag{14}$$

$$\text{2D: } \bar{B} \;=\; \tilde{B} \otimes \tilde{B} \tag{15}$$

$$\text{3D: } \bar{B} \;=\; \tilde{B} \otimes \tilde{B} \otimes \tilde{B}, \tag{16}$$

  which are ***full***.

- When applying the full $\bar{B}$ we of course use the tensor product form rather than explicitly forming the matrix.
- In the case of the diagonal matrix, however, it is often more convenient to just form $\bar{B}$ explicitly, particularly when there is a nontrivial Jacobian to consider in the context of deformed domains.
- In this latter case, the diagonal property of the matrix holds, whereas the tensor-product form does not. (In this case, one has to map to the Mth-order quadrature points and evaluate the product of the integrand, the Jacobian, and the weights on the elevated mesh.)
- Both forward application and inversion are *much simpler* with the (high-order!) diagonal mass matrix than with the full mass matrix.
- For sufficiently large N (e.g., $N \geq 5$), GLL quadrature (4) is often adequate as the quadrature error is on par with the approximation error.

- A notable case in which the consistent mass matrix is imperative is in evaluation of the advection operator, where full quadrature is required to preserve skew-symmetry and the imaginary character of the eigenvalues.
  - Use of elevated quadrature (sometimes referred to as over-integration) in this context addresses the question of aliasing / de-aliasing that is important for numerical stability in advection-dominated applications.
  - The phenomenon of aliasing is often associated with the nonlinear terms in the NS equations, but it is more appropiately associated with *advection* by nonconstant flow fields, which can be either linear (in the case of temperature) or nonlinear (in the case of NS).

Differentiation Summary

Here we consider differentiation that maps u from the GLL points to its derivative on the same points.

$$\text{1D:}\quad \underline{u}_x = \hat{D}\underline{u} \tag{17}$$

$$\text{2D:}\quad \underline{u}_x = (\hat{I} \otimes \hat{D})\underline{u} = \hat{D}U \tag{18}$$

$$\underline{u}_y = (\hat{D} \otimes \hat{I})\underline{u} = U\hat{D}^T \tag{19}$$

$$\text{3D:}\quad \underline{u}_x = (\hat{I} \otimes \hat{I} \otimes \hat{D})\underline{u} \tag{20}$$

$$\underline{u}_y = (\hat{I} \otimes \hat{D} \otimes \hat{I})\underline{u} \tag{21}$$

$$\underline{u}_z = (\hat{D} \otimes \hat{I} \otimes \hat{I})\underline{u} \tag{22}$$

D-Dimensional Operators on $\Omega \neq \hat{\Omega}$

- We'd like to solve problems on $\Omega = [0 : L_x] \times [0 : L_y]$.
- We can derive all the necessary tools in 1D and then apply tensor product rules.

- Let $x \in \Omega = [0 : L_x]$ and $r \in \hat{\Omega} = [-1, 1]$.

$$x = \frac{L_x}{2}(1 + r), \qquad \frac{dx}{dr} = \frac{L_x}{2} =: \textit{Jacobian, } \mathcal{J}, \tag{23}$$

$$r = -1 + \frac{2}{L_x}x, \qquad \frac{dr}{dx} = \frac{2}{L_x}. \tag{24}$$

- Integration for $v(x), u(x) \in X^N$:

$$\begin{aligned}
(v, u) &:= \int_0^{L_x} v(x)\, u(x)\, dx && (25)\\
&= \int_{-1}^{1} v(x(r))\, u(x(r)) \frac{dx}{dr}\, dr && (26)\\
&= \int_{-1}^{1} v\, u \mathcal{J}\, dr && (27)\\
&= \frac{L_x}{2} \int_{-1}^{1} v\, u\, dr \qquad \text{(Jacobian is constant.)}, && (28)
\end{aligned}$$

- Quadrature for $v(x), u(x) \in X^N$:

$$\begin{aligned}
(v, u)_N &= \sum_{k=0}^{N} v_k u_k \mathcal{J}_k \rho_k && (29)\\
&= \underline{v}^T \left(\frac{L_x}{2}\, \hat{B} \right) \underline{u}. && (30)\\
&= \underline{v}^T\, \bar{B}_x\, \underline{u}, && (31)
\end{aligned}$$

with

$$\bar{B}_x := \frac{L_x}{2} \hat{B}. \qquad (32)$$

- Differentiation *(chain rule)*:

$$\frac{du}{dx} = \frac{du}{dr}\frac{dr}{dx} \tag{33}$$

$$= \frac{2}{L_x}\frac{du}{dr}. \tag{34}$$

- Differentiation at nodal points,

$$\underline{u}_x = \frac{2}{L_x}\hat{D}\underline{u} = D_x\underline{u}, \tag{35}$$

with

$$D_x := \frac{2}{L_x}\hat{D}. \tag{36}$$

- Bilinear form for 1D Poisson:

$$
\begin{aligned}
a(v,u) \;=\; (v_x,u_x)_N \;&=\; (D_x\underline{v})^T\,\bar{B}_x(D_x\underline{u})^T && (37)\\
&=\; \left(\frac{2}{L_x}\hat{D}\underline{v}\right)^T \left(\frac{L_x}{2}\hat{B}\right)\,\left(\frac{2}{L_x}\hat{D}\underline{u}\right) && (38)\\
&=\; \underline{v}^T\,\frac{2}{L_x}\hat{D}^T\,\hat{B}\,\hat{D}\,\underline{u} && (39)\\
&=\; \underline{v}^T\,\bar{A}_x\underline{u} && (40)
\end{aligned}
$$

- Bilinear form for 2D Poisson:

$$
\begin{aligned}
a(v,u) &= (v_x, u_x)_N \;+\; (v_y, u_y)_N && (41)\\
&= \underline{v}^T \left(\bar{B}_y \otimes \bar{A}_x \;+\; \bar{A}_y \otimes \bar{B}_x \right) \underline{u}. && (42)\\
&= \underline{v}^T \left(\frac{L_y}{L_x} \hat{B} \otimes \hat{A} \;+\; \frac{L_x}{L_y} \hat{A} \otimes \hat{B} \right) \underline{u}. && (43)
\end{aligned}
$$

- RHS form for 2D Poisson:

$$
\begin{aligned}
(v,f)_N &= \underline{v}^T \left(\bar{B}_y \otimes \bar{B}_x \right) \underline{f} && (44)\\
&= \underline{v}^T \frac{L_x L_y}{4} \left(\hat{B} \otimes \hat{B} \right) \underline{f}. && (45)
\end{aligned}
$$

- To get the form for $v, u \in X_0^N \subset X^N$, apply $R^T = R_y^T \otimes R_x^T$ to v and u.
- In this restricted space, we have to prolongate $\underline{v}$ to $\underline{\bar{v}}$ and $\underline{u}$ to $\underline{\bar{u}}$.

- Thus, the bilinear form for 2D Poisson is

$$a(v, u) \;=\; \underline{v}^T \left(B_y \otimes A_x \;+\; A_y \otimes B_x \right) \underline{u}. \;=\; \underline{v}^T \left(R_y \otimes R_x \right) (\bar{B}_y \otimes \bar{B}_x) \underline{f},$$

with

$$A_x \;=\; R_x \frac{2}{L_x} \hat{D}^T \hat{B} D h R_x^T \qquad\qquad B_x \;=\; R_x \frac{L_x}{2} \hat{B} R_x^T \tag{47}$$

$$A_y \;=\; R_y \frac{2}{L_y} \hat{D}^T \hat{B} D h R_y^T \qquad\qquad B_y \;=\; R_y \frac{L_y}{2} \hat{B} R_y^T \tag{48}$$

- Defining

$$A \;=\; \left(B_y \otimes A_x \;+\; A_y \otimes B_x \right), \tag{49}$$

$$\bar{B} \;=\; \left(\bar{B}_y \otimes \bar{B}_x \right), \tag{50}$$

we have

$$A\underline{u} \;=\; R\, \bar{B}\, \underline{f}. \tag{51}$$

WRT in 2D: Poisson Example

- Consider $-\nabla^2 u = f$ on $[0, L_x] \times [0, L_y]$ with $u|_{\partial\Omega} = 0$.

- WRT: *For all* $v \in X_0^N$, *find* $u \in X_0^N$ *such that*

$$a(v, u) = \int_\Omega \nabla v \cdot \nabla u \, dV = \int_\Omega v \, f \, dV.$$

- Follows from *strong form* plus integration by parts:

$$-\int_\Omega v \left(\underbrace{\frac{\partial^2 u}{\partial x^2}}_{\mathcal{I}_x} + \underbrace{\frac{\partial^2 u}{\partial y^2}}_{\mathcal{I}_y} \right) dx \, dy = \int_\Omega v \, f \, dV.$$

- First integral:

$$\begin{aligned}
\mathcal{I}_x &:= -\int_0^{L_y} \left[\int_0^{L_x} v \frac{\partial^2 u}{\partial x^2} \, dx \right] dy \\
&= \int_0^{L_y} \left[\int_0^{L_x} \frac{\partial v}{\partial x} \frac{\partial u}{\partial x} \, dx - \underbrace{v(x, y) \frac{\partial u}{\partial x} \bigg|_{x=0,y}^{x=L_x,y}}_{=\,0} \right] dy \\
&= \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial x} \frac{\partial u}{\partial x} \, dx \, dy.
\end{aligned}$$

- Second integral, by symmetry:

$$\mathcal{I}_y = \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial y} \frac{\partial u}{\partial y} \, dx \, dy.$$

Evaluating Integrals Using Basis Functions

- For all $v,\, u \in X^N$,

$$
\begin{aligned}
v(x,y) &= \sum_{i=0}^{N}\sum_{j=0}^{N} v_{ij}\, \phi_i(x)\, \tilde{s}_j(y) \\
u(x,y) &= \sum_{p=0}^{N}\sum_{q=0}^{N} u_{pq}\, \phi_p(x)\, \tilde{s}_q(y).
\end{aligned}
$$

- Here, we take ϕ and $\tilde{s}$ to be the bases on $[0, L_x] \times [0, L_y]$,

$$
\begin{aligned}
\phi_i(x) &:= l_i(r(x)), \quad r(x) := -1 + \frac{2}{L_x}x, \\
\tilde{s}_j(y) &:= l_j(s(y)), \quad s(y) := -1 + \frac{2}{L_y}y,
\end{aligned}
$$

  and the l_is and l_js are the (same) Lagrange interpolating polynomials based on the Gauss-Lobatto-Legendre (GLL) points $r_j \in [-1, 1] =: \hat{\Omega}$.

- Derivative with respect to x:

$$
\begin{aligned}
\frac{\partial v}{\partial x} &= \sum_{i=0}^{N}\sum_{j=0}^{N} \phi_i'(x)\, \tilde{s}_j(y)\, v_{ij} \\
\frac{\partial u}{\partial x} &= \sum_{p=0}^{N}\sum_{q=0}^{N} \phi_p'(x)\, \tilde{s}_q(y)\, u_{pq}.
\end{aligned}
$$

- Derivative with respect to y:

$$
\begin{aligned}
\frac{\partial v}{\partial y} &= \sum_{i=0}^{N}\sum_{j=0}^{N} \phi_i(x)\, \tilde{s}_j'(y)\, v_{ij} \\
\frac{\partial u}{\partial y} &= \sum_{p=0}^{N}\sum_{q=0}^{N} \phi_p(x)\, \tilde{s}_q'(y)\, u_{pq}.
\end{aligned}
$$

- First integral,

$$
\begin{aligned}
\mathcal{I}_x &= \int_0^{L_y}\int_0^{L_x}\left(\sum_{ij} v_{ij}\,\phi_i'(x)\,\tilde{s}_j(y)v_{ij}\right)\left(\sum_{pq} u_{pq}\,\phi_p'(x)\,\tilde{s}_q(y)u_{pq}\right)dx\,dy \\
&= \sum_{pq}\sum_{ij} v_{ij}\int_0^{L_y}\tilde{s}_j(y)\,\tilde{s}_q(y)\left[\int_0^{L_x}\phi_i'(x)\,\phi_p'(x)\,dx\right]dy\,u_{pq} \\
&= \sum_{pq}\sum_{ij} v_{ij}\left(\int_0^{L_y}\tilde{s}_j(y)\,\tilde{s}_q(y)\,dy\right)\left(\int_0^{L_x}\phi_i'(x)\,\phi_p'(x)\,dx\right)u_{pq}, \\
&= \sum_{pq}\sum_{ij} v_{ij}\,\bar{B}_{y,jq}\,\bar{A}_{x,ip}\,u_{pq} \\
&= \underline{v}^T\left(\bar{B}_y\otimes\bar{A}_x\right)\underline{u}.
\end{aligned}
$$

- Second integral, by symmetry,

$$
\mathcal{I}_y = \underline{v}^T\left(\bar{A}_y\otimes\bar{B}_x\right)\underline{u}.
$$

- Following as we did in 1D, we end up with the linear system

$$
\begin{aligned}
A\underline{u} &= B\underline{f} \\
A &= B_y \otimes A_x \, + \, A_y \otimes B_x.
\end{aligned}
$$

- Let, as in 1D,

$$
\begin{aligned}
\bar{A}_x &:= \frac{2}{L_x}\hat{D}^T\hat{B}\hat{D}, \\
\left[\bar{A}_x\right]_{ip} &:= \int_0^{L_x} \phi_i'(x)\,\phi_p'(x)\,dx \\
&\equiv \sum_{k=0}^{N} \phi_i'(x_k)\left(\frac{L_x}{2}\rho_k\right)\phi_p'(x_k) \qquad \text{(GLL quadrature)} \\
&= \sum_{k=0}^{N}\left[\frac{2}{L_x}\left.\frac{dl_i}{dr}\right|_{r_k}\right]\left(\frac{L_x}{2}\rho_k\right)\left[\frac{2}{L_x}\left.\frac{dl_p}{dr}\right|_{r_k}\right] \\
&= \frac{2}{L_x}\sum_{k=0}^{N}\hat{D}_{ki}\,\rho_k\,\hat{D}_{kp} \\
&= \frac{2}{L_x}\sum_{k=0}^{N}\hat{D}_{ki}\,\hat{B}_{kk}\,\hat{D}_{kp} \\
&= \left[\frac{2}{L_x}\,\hat{D}^T\,\hat{B}\,\hat{D}\right]_{ip}.
\end{aligned}
$$

- This gives us the *bar* matrix, $\bar{A}_x$, which includes the endpoints.

- To get the *interior points only*, we (re-)introduce the restriction matrix, R, and prolongation matrix, R^T.
- For the Dirichlet-Dirichlet case,

$$
R^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \underline{0}^T \\ I \\ \underline{0}^T \end{bmatrix} \in \mathbb{R}^{(N+1)\times(N-1)}.
$$

- **Q:** What is the role of R^T in a matrix-vector product?
- **Q:** What is the role of R in a matrix-vector product?

Matlab Example `wrttest.m`

```
N=5; % Polynomial Order
[Ah,Bh,Ch,Dh,z,w] =  semhat(N);
R = speye(N+1); R=R(2:(end-1),:); % Dirichlet Restriction Matrix
A = R*Ah*R';
f=1+0*z;
u = A \ (R*(Bh*f));
u=[0;u;0];
ue=.5*(1-z.*z); err = max(abs(u-ue))
```

- After executing this code we can look at R:

```
R = full(R)
R =
     0     1     0     0     0     0
     0     0     1     0     0     0
     0     0     0     1     0     0
     0     0     0     0     1     0

Ah =
   5.1667e+00  -5.7553e+00   7.5291e-01  -2.3287e-01   1.0197e-01  -3.3333e-02
  -5.7553e+00   9.1267e+00  -3.9809e+00   8.3085e-01  -3.2331e-01   1.0197e-01
   7.5291e-01  -3.9809e+00   6.0400e+00  -3.4100e+00   8.3085e-01  -2.3287e-01
  -2.3287e-01   8.3085e-01  -3.4100e+00   6.0400e+00  -3.9809e+00   7.5291e-01
   1.0197e-01  -3.2331e-01   8.3085e-01  -3.9809e+00   9.1267e+00  -5.7553e+00
  -3.3333e-02   1.0197e-01  -2.3287e-01   7.5291e-01  -5.7553e+00   5.1667e+00
>> A
A =
   9.1267e+00  -3.9809e+00   8.3085e-01  -3.2331e-01
  -3.9809e+00   6.0400e+00  -3.4100e+00   8.3085e-01
   8.3085e-01  -3.4100e+00   6.0400e+00  -3.9809e+00
  -3.2331e-01   8.3085e-01  -3.9809e+00   9.1267e+00
>> Bh
Bh =
   6.6667e-02            0            0            0            0            0
            0   3.7847e-01            0            0            0            0
            0            0   5.5486e-01            0            0            0
            0            0            0   5.5486e-01            0            0
            0            0            0            0   3.7847e-01            0
            0            0            0            0            0   6.6667e-02
>> R*Bh*R'
ans =
   3.7847e-01            0            0            0
            0   5.5486e-01            0            0
            0            0   5.5486e-01            0
            0            0            0   3.7847e-01
```

- We see that R does just what we need. It picks out the active *rows* (i.e., equations) in the ODE/PDE.

- $R^T \underline{u}$ extends $\underline{u}$ by zero, so that we can apply quadrature and/or differentiate $u(x)$ on $x_0, x_1, \ldots, x_N$, as needed.

```
>> u=[1 2 3 4 ]'

u =
     1
     2
     3
     4

>> R'*u

ans =
     0
     1
     2
     3
     4
     0
```

- **Note:** If we have *Neumann* conditions in the y-direction (say), then y-indices range from 0 to N.
- For example, LHS of the ij-th equation will be

$$\begin{aligned}
\mathcal{I} \;&=\; = \sum_{q=0}^{N} \sum_{p=1}^{N-1} \left(B_{y,jq}\, A_{x,ip} \;+\; A_{y,jq}\, B_{x,ip} \right) u_{pq} \qquad \left\{ \begin{array}{l} i = 1, \dots, N-1 \\ j = 0, \dots, N \end{array} \right. \\
&=\; \left[B_y \otimes A_x \;+\; A_y \otimes B_x \right] \underline{u} \\
&=\; A\, \underline{u}.
\end{aligned}$$

- The mass matrices, B_x, B_y, and stiffness matrices, A_x, A_y, will have the correct size (and hence, correct index range in the first expression), if we set

$$\begin{aligned}
B_x &= R_x \bar{B}_x R_x^T \qquad & A_x &= R_x \bar{A}_x R_x^T \\
B_y &= R_y \bar{B}_y R_y^T \qquad & A_y &= R_y \bar{A}_y R_y^T,
\end{aligned}$$

with

$$R_x^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \underline{0}^T \\ I \\ \underline{0}^T \end{bmatrix} \in \mathbb{R}^{(N+1)\times(N-1)},$$

$$R_y^T = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} = \begin{bmatrix} \; I \; \end{bmatrix} \in \mathbb{R}^{(N+1)\times(N+1)}.$$

- We typically refer to $\bar{A}$ and $\bar{B}$ as the *Neumann operators.*
- They are the full $(N+1) \times (N+1)$ matrices corresponding to the Neumann problem.
- In terms of the generic ("hat") matrices, they are (for x, say)
  - $\bar{A}_x \;=\; \dfrac{2}{L_x}\, \hat{A},$
  - $\bar{B}_x \;=\; \dfrac{L_x}{2}\, \hat{B}.$

Example: Solve $-\nabla^2 u = 1$ on $[0,2] \times [0,1]$

- $u(x,y) = 0$ at $x = 0, 2$.
- $\dfrac{\partial u}{\partial y} = 0$ at $y = 0, 1$.

Variable Coefficients:

- Recall 1D example:

$$-\frac{d}{dx}p(x)\frac{du}{dx} \quad = \quad f(x) \; + \; \text{BCs} \begin{cases} u(0) = 0 \\ u'(1) = 0 \end{cases}$$

- WRT:

$$\begin{aligned}
-\int_{-1}^{1} v \, \frac{d}{dx}p(x)\frac{du}{dx} \, dx &= \int_{-1}^{1} v \, f(x) \, dx \\
\int_{-1}^{1} \frac{dv}{dx}p(x)\frac{du}{dx} \, dx &= \int_{-1}^{1} v \, f(x) \, dx \qquad \text{(boundary term vanishes because of BCs)} \\
(D\underline{v})^T \, PBD \, \underline{u} &= \underline{v}^T B \underline{f} \\
\underline{v}^T D^T PBD \, \underline{u} &= \underline{v}^T B \underline{f} \\
A\underline{u} &= B\underline{f} \\
A &:= D^T PBD.
\end{aligned}$$

- With (*highly accurate*) GLL quadrature, it is sufficient to evaluate $p(x)$ at the GLL nodal points, x_k, $k = 0, \ldots, N$
- The matrices P and B are thus both diagonal:

$$P_{kk} \;=\; p(x_k), \quad B_{kk} \;=\; \rho_k, \quad k = 0, \ldots, N.$$

- 2D example:

$$
\begin{aligned}
-\nabla \cdot [p(x,y)\nabla u] &= f(x,y) \; + \; \text{BCs} \\
-\int\int v\,\nabla \cdot [p\nabla u] \; dx\,dy &= \int\int v\,f(x,y)\,dx\,dy \\
-\int\int v \left[\frac{\partial}{\partial x} p \frac{\partial u}{\partial x} \; + \; \frac{\partial}{\partial y} p \frac{\partial u}{\partial y} \right] \; dx\,dy &= \int\int v\,f(x,y)\,dx\,dy
\end{aligned}
$$

- First term on left, itegrated by parts

$$
\mathcal{I}_x \;=\; \int_0^{L_y} \left[\int_0^{L_x} \frac{\partial v}{\partial x} p \frac{\partial u}{\partial x}\,dx \; - \; v\,p\; \frac{\partial u}{\partial x}\Big|_{x=0}^{x=1} \right] \; dy
$$

- Let Γ_i denote the ith face of Ω,
  - i=1: left face
  - i=2: right face
  - i=3: lower face
  - i=4: upper face
- Let $\hat{\mathbf{n}}_i$ be the associated outward-pointing unit-normal on each face:
  - i=1: $\hat{\mathbf{n}}_1 = (\hat{\mathbf{n}}_{x,1}, \hat{\mathbf{n}}_{y,1}) = (-1,0)$
  - i=2: $\hat{\mathbf{n}}_2 = (\hat{\mathbf{n}}_{x,2}, \hat{\mathbf{n}}_{y,2}) = (1,0)$
  - i=3: $\hat{\mathbf{n}}_3 = (\hat{\mathbf{n}}_{x,3}, \hat{\mathbf{n}}_{y,3}) = (0,-1)$
  - i=4: $\hat{\mathbf{n}}_4 = (\hat{\mathbf{n}}_{x,4}, \hat{\mathbf{n}}_{y,4}) = (0,1)$
- We can rewrite $\mathcal{I}_x$ as:

$$
\mathcal{I}_x \;=\; \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial x} p \frac{\partial u}{\partial x}\,dx\,dy \; - \; \sum_{i=1}^{2} \int_{\Gamma_i} v\,p\,\frac{\partial u}{\partial x}\,\hat{\mathbf{n}}_{x,i}\,dy.
$$

- Similarly, $\mathcal{I}_y$ is

$$
\mathcal{I}_y \;=\; \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial y} p \frac{\partial u}{\partial y}\,dx\,dy \; - \; \sum_{i=3}^{4} \int_{\Gamma_i} v\,p\,\frac{\partial u}{\partial y}\,\hat{\mathbf{n}}_{y,i}\,dy.
$$

- Combining $\mathcal{I}_x$ and $\mathcal{I}_y$,

$$
\begin{aligned}
\mathcal{I} &= \mathcal{I}_x + \mathcal{I}_y \\
&= \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial x} p \frac{\partial u}{\partial x} \; + \; \frac{\partial v}{\partial y} p \frac{\partial u}{\partial y}\,dx\,dy \; - \; \sum_{i=1}^{4} \int_{\Gamma_i} v\,p\,\left(\frac{\partial u}{\partial x}\,\hat{\mathbf{n}}_{x,i} \; + \; v\,p\,\frac{\partial u}{\partial y}\,\hat{\mathbf{n}}_{y,i} \right)\,dS. \\
&= \int_{\Omega} \nabla v \cdot (p\nabla u)\;dV \; - \; \int_{\partial\Omega} v\,p\,\nabla u \cdot \hat{\mathbf{n}}\,dS.
\end{aligned}
$$

- If $\displaystyle\int_{\partial\Omega} v\,p\,\nabla u\cdot\hat{\mathbf{n}}\Big|_{\partial\Omega} \;=\; 0$, then u satisfies

$$\int_{\Omega} \nabla v\cdot(p\nabla u)\;dV \;\;=\;\; \int_{\Omega} v\,f\,dV \quad \text{for all } v \,\in\, X_0^N.$$

- **Note:** For any $u \in X^N$ we have the *gradient*

$$\nabla u \;\;=\;\; \begin{pmatrix} \frac{\partial u}{\partial x} \\ \\ \frac{\partial u}{\partial y} \end{pmatrix} \;=\; \begin{pmatrix} D_x \\ \\ D_y \end{pmatrix} \underline{u} \;=\; \begin{pmatrix} \hat{I}\otimes\frac{2}{L_x}\hat{D} \\ \\ \frac{2}{L_y}\hat{D}\otimes\hat{I} \end{pmatrix} \underline{u}.$$

- Example:

$$\begin{aligned}
&x = \frac{L_x}{2}(z+1), \;\; y = \frac{L_y}{2}(z+1) \\
&[X,Y] = ndgrid(x,y) \\
&U = \sin\pi X \\
&U_x = D_x U, \;\; U_y = D_y U \\
&D_x = \hat{I}\otimes\frac{2}{L_x}\hat{D}, \;\; D_y = \frac{2}{L_y}\hat{D}\otimes\hat{I}
\end{aligned}$$

- Plot U_x, U_y.

- Returning to our weighted residual form, insertion of the bases and using quadrature yields

$$\begin{aligned}
\mathcal{I} &= \int_0^{L_y} \int_0^{L_x} \frac{\partial v}{\partial x} p \frac{\partial u}{\partial x} \; + \; \frac{\partial v}{\partial y} p \frac{\partial u}{\partial y} \, dx \, dy \\
&= (D_x \underline{v})^T \; P \, \bar{B} \, D_x \underline{u} \; + \; (D_y \underline{v})^T \; P \, \bar{B} \, D_y \underline{u} \\
&= \underline{v}^T \, D_x^T \, P \, \bar{B} \, D_x \underline{u} \; + \; \underline{v}^T \, D_y^T \, P \, \bar{B} \, D_y \underline{u}.
\end{aligned}$$

- Note that if the range of indices on $\underline{v}$ and $\underline{u}$ includes *all nodes*, including the boundaries, then we would have the form of our *Neumann operator*,

$$\bar{A} \;=\; D_x^T \, P \, \bar{B} \, D_x \; + \; D_y^T \, P \, \bar{B} \, D_y.$$

- To construct the A-matrix that governs the interior degrees of freedom, apply the restriction operator R to $\bar{A}$:

$$A \;=\; R \, \bar{A} \, R^T.$$

- In the case of a rectangular domain, R restricts index sets in both the x and y directions and it can be written in tensor-product form:

$$R \;=\; R_y \otimes R_x.$$

  As always, the R_x matrix is on the right, because it is acting on the fastest changing index in the $\underline{u}^T = [u_{00} \, u_{10} \, \ldots \, u_{NN}]^T$ lexicographical ordering of the unknowns.

Gradient-Based Form

- It is sometimes convenient to write $\bar{A}$ in a higher-level form.
- We can also write

$$\bar{A} \;=\; \begin{bmatrix} D_x \\ D_y \end{bmatrix}^T \begin{bmatrix} P\bar{B} & 0 \\ 0 & P\bar{B} \end{bmatrix} \begin{bmatrix} D_x \\ D_y \end{bmatrix} \;=\; \mathbf{D}^T \, \mathbf{P} \, \mathbf{B} \, \mathbf{D}.$$

- Here, we use the bold font to indicate that the matrices are working on or producing **vector fields**, i.e., inputs and/or outputs that have two components in 2D, three components in 3D, etc.
- Obviously, the **D** matrix is the discrete gradient operator introduced above.

Eigenvalue Examples

- Eigenvalues are of central importance in understanding the behavior of time-dependent problems, both from a physics standpoint and from numerical convergence and stability concerns.

- Consider the time-dependent PDE,

$$\frac{\partial \tilde{u}}{\partial t} = -\mathcal{L}\tilde{u}, \quad \tilde{u}(\mathbf{x}, t=0) = u^0(\mathbf{x}), \tag{52}$$

  where associated boundary conditions are embedded in the linear operator, $\mathcal{L}$.

- Suppose there exists a set of eigenfunctions $\tilde{s}_k(\mathbf{x})$ and associated eigenvalues such that

$$\mathcal{L}\,\tilde{s}_k = \tilde{\lambda}\,\tilde{s}_k, \tag{53}$$

  and that, for any given $\tilde{u}$ having sufficient regularity that we can find coefficients $\hat{u}_k$ such that

$$\tilde{u}(\mathbf{x}, t) = \sum_k \hat{u}_k(t)\,\tilde{s}_k(\mathbf{x}). \tag{54}$$

- Then,

$$\frac{\partial \tilde{u}}{\partial t} = \sum_k \frac{d\hat{u}_k}{dt}\,\tilde{s}_k(\mathbf{x}) = -\sum_k \hat{u}_k(t)\,\mathcal{L}\,\tilde{s}_k(\mathbf{x}) = -\sum_k \tilde{\lambda}_k \hat{u}_k(t)\,\tilde{s}_k(\mathbf{x}). \tag{55}$$

- Since we have diagonalized $\mathcal{L}$, we have a set of decoupled ODEs in time,

$$\frac{d\hat{u}_k}{dt} = -\tilde{\lambda}_k \hat{u}_k(t), \quad \hat{u}_k(0) =: \hat{u}_k^0, \tag{56}$$

  for which the solution is

$$\hat{u}_k(t) = \hat{u}_k^0\, e^{-\tilde{\lambda}_k t}. \tag{57}$$

Interpretation of Analytical Eigenvalues

- We make several remarks regarding (57).
- First, if the eigenvalues $\tilde{\lambda}_k$ are real and positive, then the solution exhibits exponential decay, with high wavenumber (i.e., large eigenvalue) components decaying much more rapidly than the low wavenumber components.
- For second-order operators in space we can anticipate from dimensional analysis that $\tilde{\lambda}_k \sim Ck^2$, which means that the solution will rapidly evolve to a multiple of the eigenmode with the smallest eigenvalue.
- Specifically, assuming $0 < \tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \cdots$, we'll find, after a relatively short time,

$$\tilde{u}(\mathbf{x}, t) \;\sim\; \hat{u}_1^0 \, e^{-\tilde{\lambda}_1 t} \, \tilde{s}_1(\mathbf{x}) \tag{58}$$

- This physical behavior is characteristic of the unsteady heat (or diffusion) equation.

- Conversely, if the eigenvalues are imaginary, the soution will exhibit *no decay*. The modulus of the coefficients,

$$|\,\hat{u}_k(t)\,| = |\,\hat{u}_k^0\,|\,|\,e^{-\tilde{\lambda}_k t}\,|, \tag{59}$$

  will be unchanged for all time.

- This physical behavior is characteristic of the advection equation with minimal or no diffusion, modulo bounary effects.

- The fact that the solution does not decay often implies that errors also do not decay. As they are generated in the course of the numerical solution, they tend to stay in the system and continually accumulate until the solution is essentially garbage.

- For this reason, many practioners apply low pass filtering or some sort of high wavenumber damping (or diffusion) to try to suppress spurious modes.

- As long as the effect of these smoothing operators is high order, there is no particular harm in suppressing noise with these approaches.

- It is possible to develop filters that are quite robust and that require minimal or no parametric tuning.

Numerical Eigenproblem

- Because of their impact on time-dependent problems, we wish to understand how the eigenvalues of our discrete system relate to their continuous counterparts.
- To make things precise, consider

$$\mathcal{L}\tilde{u} \;=\; -\frac{d^2\tilde{u}}{dx^2} \;=\; \tilde{\lambda}\,\tilde{u}, \quad \tilde{u}(0)=\tilde{u}(1)=1. \tag{60}$$

- The eigenfunctions in this case are

$$\tilde{s}_k \;=\; \sin k\pi x, \tag{61}$$

  with corresponding eigenvalues,

$$\tilde{\lambda}_k \;=\; k^2\pi^2, \quad k=1,\,2\,\ldots \tag{62}$$

- For the discrete problem, let $\tilde{u}(x) \approx u(x) \in X_0^N$ be given by the expansion

$$u(x) \;=\; \sum_{j=0}^{N} u_j l_j(x), \tag{63}$$

  with $u_0 = u_N = 0$.

- We have $N-1$ uknowns and need $N-1$ equations, which we can obtain through our standard variational projection technique.

- We seek $u \in X_0^N$ such that for all $v \in X_0^N$,

$$a(v,u) \;=\; \lambda\,(v,u). \tag{64}$$

- The discrete form is thus

$$A\underline{u} \;=\; \lambda B\underline{u}, \tag{65}$$

  which is a *generalized eigenvalue problem.*

- In matlab, the solution is obtained as

```
[S,D] = eig(A,B); lam=diag(D);
[lam,ind] = sort(lam);   %% Sort eigenvalues
S=S(:,ind);              %% Sort eigenvectors
```

- Here, $S = [\underline{s}_1\ \underline{s}_2\ \dots\ \underline{s}_{N-1}]$ is the matrix of eigenvectors, sorted such that $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{N-1}$.

- Because the low-wavenumber eigenvectors are *smooth*, we can anticipate that they will be well represented in X_0^N for $N > \gamma k$, where $\gamma > 1$ is an order-unity constant.

- We would thus also expect that $\tilde{\lambda}_k \sim \lambda_k$ for $N > \gamma k$.

Eigenvalue Exercises

1. Use the Legendre spectral method utilities provided to explore the assertions made in the last two bullets.

2. Can you (numerically) find an asymptotic value of γ as $(N, k) \longrightarrow \infty$?

3. For $N = 250$, plot on a loglog scale $\tilde{\lambda}_k$ vs. k and λ_k vs. k.

4. At what value of k/N do they appear to diverge?

5. For $N = 250$, plot on a loglog scale e_k vs. k, where

$$e_k \;:=\; \frac{|\tilde{\lambda}_k - \lambda_k|}{\tilde{\lambda}_k}. \tag{66}$$

   - What is the value of k/N when $e_k \approx .001$?
   - What is the value e_k when $k = 0.9\times$ the value found in the preceding question? (This gives some idea of how rapidly the approximation converges one the eigenfunction is resolved.)

6. Use your code to answer the following for $N \longrightarrow \infty$.

   (a) Identify the constants γ and ν in the asymptotic relationship,

$$\lambda_{\max} \;\sim\; \gamma N^{\nu}. \tag{67}$$

   (b) Identify the constants γ and ν in the asymptotic relationship,

$$\lambda_{\max} \;\sim\; \gamma \Delta x_{\min}^{\nu}. \tag{68}$$

   (c) Identify the constants γ and ν in the asymptotic relationship,

$$\Delta x_{\min} \;\sim\; \gamma N^{\nu}. \tag{69}$$

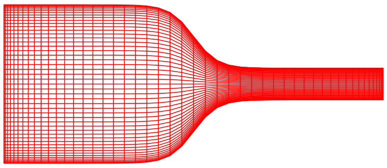
   (d) Let σ_k be the eigenvalue of A (not $B^{-1}A$). Identify the constants γ and ν in the asymptotic relationship,

$$\sigma_{\max} \;\sim\; \gamma N^{\nu}. \tag{70}$$

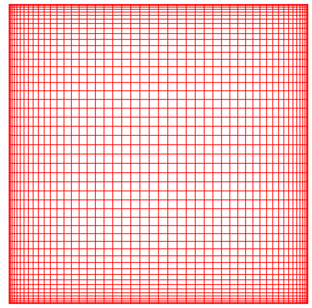
7. Repeat (a)–(c) for the preceding question for the matrix $B^{-1}C$, where $C = Q^T \hat{B}\hat{C}Q$, $B = Q^T \hat{B} Q$, and Q is the periodic prolongation matrix introduced earlier.

8. Here, $B^{-1}C$ is the 1D advection operator on a periodic domain. Plot $|\lambda_k|$ for this case and plot the error. (What are $\tilde{s}_k$ and $\tilde{\lambda}_k$ for this problem?)

Deformed Geometries



Ω



$\hat{\Omega}$

- If Ω is deformed,

$$
\begin{aligned}
\mathbf{x}(r,s) &= \sum_{ij} \mathbf{x}_{ij}\, l_i(r)\, l_j(s) \\
&= \sum_{j=0}^{N} \sum_{i=0}^{N} \mathbf{x}_{ij}\, l_i(r)\, l_j(s).
\end{aligned}
$$

- Derivative is computed using the *chain rule*

$$
\begin{aligned}
[\underline{w}]_{pq} &= \left.\frac{\partial u}{\partial x}\right|_{r_p s_q} = \frac{\partial u}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial u}{\partial s}\frac{\partial s}{\partial x} \\
&= (r_x)_{pq}(\underline{u}_r)_{pq} + (s_x)_{pq}(\underline{u}_s)_{pq} \\
\underline{w} &= [r_x](\hat{I} \otimes \hat{D})\underline{u} + [s_x](\hat{D} \otimes \hat{I})\underline{u}.
\end{aligned}
$$

- The full gradient (*vector field*) is given by,

$$
\begin{aligned}
\underline{\mathbf{w}} = \nabla \underline{u} &= \underbrace{\begin{bmatrix} [r_x] & [s_x] \\ [r_y] & [s_y] \end{bmatrix}}_{\mathbf{R_x}} \begin{bmatrix} D_r \\ D_s \end{bmatrix} \underline{u} \\
&= \mathbf{R_x} \mathbf{D}_r\, \underline{u}.
\end{aligned}
$$

- Here, we use **bold font** to indicate a *vector field* or a matrix that operates on and/or produces a vector field.

Metrics

- What is $\left.\dfrac{\partial r}{\partial x}\right|_{r_p s_q}$?

$$\left.\frac{\partial \underline{\mathbf{x}}}{\partial r}\right|_{pq} = \sum_{ij} \mathbf{x}_{ij} \left.\frac{dl_i}{dr}\right|_p l_j(s_q) = D_r \underline{\mathbf{x}} = (\hat{I} \otimes \hat{D})\underline{\mathbf{x}}$$

$$\left.\frac{\partial \underline{\mathbf{x}}}{\partial s}\right|_{pq} = \sum_{ij} \mathbf{x}_{ij}\, l_i(r_p) \left.\frac{dl_j}{ds}\right|_q = D_s \underline{\mathbf{x}} = (\hat{D} \otimes \hat{I})\underline{\mathbf{x}}.$$

- *Chain Rule:*

$$\frac{\partial u}{\partial x} = \frac{\partial u}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial u}{\partial s}\frac{\partial s}{\partial x}.$$

- Set $u = x$:

$$\begin{aligned}
1 = \frac{\partial x}{\partial x} &= \frac{\partial x}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial x}{\partial s}\frac{\partial s}{\partial x} \\
0 = \frac{\partial y}{\partial x} &= \frac{\partial y}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial y}{\partial s}\frac{\partial s}{\partial x} \\
0 = \frac{\partial x}{\partial y} &= \frac{\partial x}{\partial r}\frac{\partial r}{\partial y} + \frac{\partial x}{\partial s}\frac{\partial s}{\partial y} \\
1 = \frac{\partial y}{\partial y} &= \frac{\partial y}{\partial r}\frac{\partial r}{\partial y} + \frac{\partial y}{\partial s}\frac{\partial s}{\partial y}.
\end{aligned}$$

- At *each point*, (r_p, s_q):

$$\begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} \\ \frac{\partial s}{\partial x} & \frac{\partial s}{\partial y} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

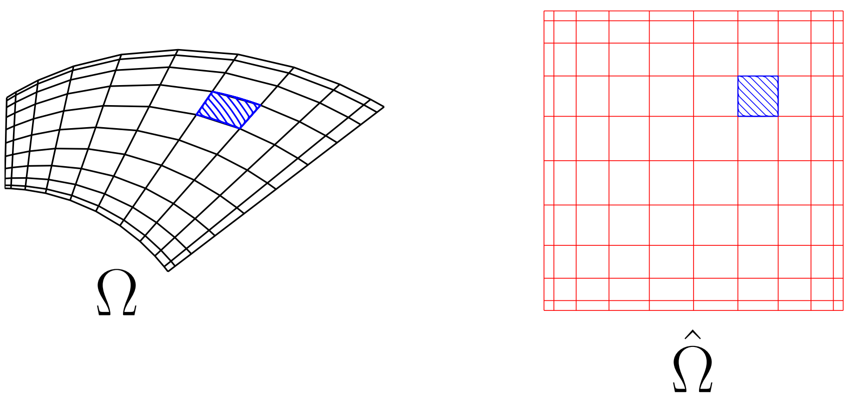
- Thus we find the inverse metrics, $\frac{\partial r_i}{\partial x_j}$, by inverting the 2×2 matrix of *computable* derivatives $\frac{\partial x_i}{\partial r_j}$, at each gridpoint (r_p, s_q).

Integration in Ω

- Similarly, we must transform

$$\int_{\Omega} f\, dA \;=\; \int_{\hat{\Omega}} f\, \mathcal{J}\, dr\, ds.$$

- Here, $\mathcal{J}$ is the *Jacobian*:
  - It is a scalar field corresponding to the amount of area in Ω that is associated with a unit area in $\hat{\Omega}$.
- An example is shown in the figure below.



- The blue area is given by the *cross product*,

$$\begin{aligned} dA &= \frac{\partial \mathbf{x}}{\partial r} dr \;\times\; \frac{\partial \mathbf{x}}{\partial s} ds \;=\; \mathcal{J}\, dr\, ds \\ &= \left[\frac{\partial x}{\partial r}\frac{\partial y}{\partial s} \;-\; \frac{\partial x}{\partial s}\frac{\partial y}{\partial r} \right] dr\, ds \\ &= \left| \begin{array}{cc} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} \end{array} \right| dr\, ds. \end{aligned}$$

- That is, $\mathcal{J}_{pq}$ is the determinant of the 2×2 metric tensor at each quadrature point.

- **Mass Matrix:** We note that the mass matrix in the deformed geometry is *diagonal*, evaluated at the GLL points:

$$\bar{B} \;=\; \mathrm{diag}(\rho_p \rho_q \mathcal{J}_{pq}) \;=\; \mathcal{J}(\hat{B} \otimes \hat{B}),$$

  if we view $\mathcal{J}$ as a *diagonal matrix,* following standard practice with GLL quadrature.

Bilinear Form in Deformed Coordinates

- As usual, we evaluate the bilinear form,

$$
\begin{aligned}
a(v,u) &= \int_\Omega \nabla v \cdot \nabla u \, dA \\
&= [\mathbf{D}\underline{v}]^T \, (\mathbf{B}) \, [\mathbf{D}\underline{u}] \\
&= [\mathbf{R_x}\mathbf{D}_r\underline{v}]^T \, (\mathbf{B}) \, [\mathbf{R_x}\mathbf{D}_r\underline{u}]
\end{aligned}
$$

$$
= \underline{v}^T \begin{pmatrix} D_r \\ D_s \end{pmatrix}^T \begin{bmatrix} [\underline{r}_x] & [\underline{s}_x] \\ [\underline{r}_y] & [\underline{s}_y] \end{bmatrix}^T \begin{pmatrix} \mathcal{J}(\hat{B} \otimes \hat{B}) & 0 \\ 0 & \mathcal{J}(\hat{B} \otimes \hat{B}) \end{pmatrix} \begin{bmatrix} [\underline{r}_x] & [\underline{s}_x] \\ [\underline{r}_y] & [\underline{s}_y] \end{bmatrix} \begin{pmatrix} D_r \\ D_s \end{pmatrix} \underline{u}.
$$

$$
= \underline{v}^T \mathbf{D}_r^T \mathbf{G} \mathbf{D} \underline{u}.
$$

- In d space dimensions,

$$
G_{ij} \;:=\; \left(\sum_{k=1}^{d} \frac{\partial r_i}{\partial x_k} \frac{\partial r_j}{\partial x_k} \right) \bar{B}, \quad i = 1, \ldots, d, \;\; j = 1, \ldots, d, \quad d = 2 \text{ or } 3.
$$

- Example, $d = 2$:

$$
G \;=\; \begin{bmatrix} G_{rr} & G_{rs} \\ G_{sr} & G_{ss} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}.
$$

- **Note:** $G_{ij} = G_{ji}$.
- *Each* G_{ij} is a *diagonal matrix* with $(N+1)^2$ (or, $(N+1)^d$) entries.