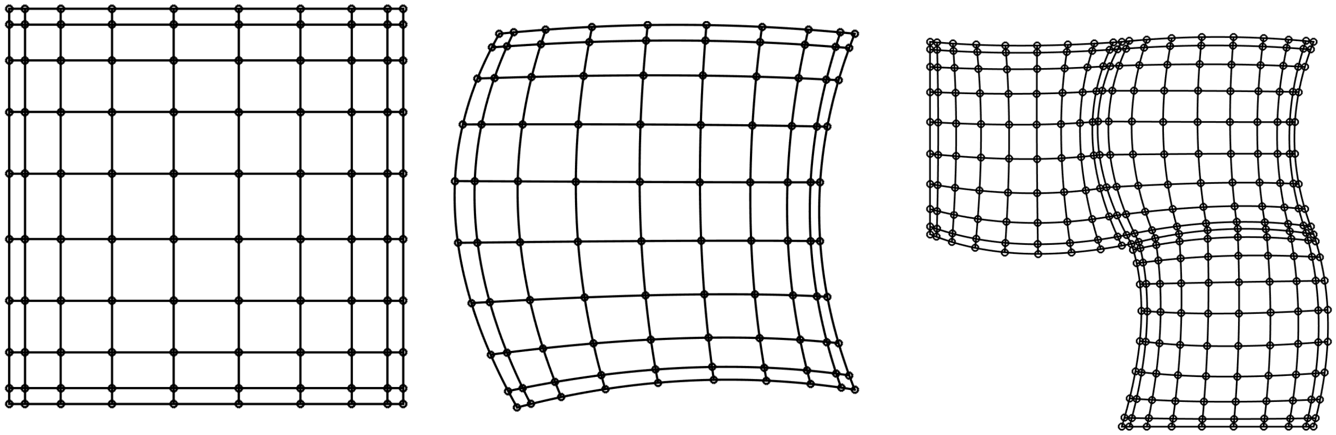


## 1 Multi-Element Case

- We've seen the undeformed single-element case already, with $\Omega = \hat{\Omega} = [-1,1]^2$, and the deformed single-element case.

- We now turn to spectral elements, where $\Omega$ consists of multiple deformed spectral subdomains,

$$\Omega \;=\; \bigcup_{e=1}^{E} \Omega^e. \tag{1}$$

- Here, the energy inner product simply becomes the integral over all elements,

$$a(v,u) \;:=\; \int_{\Omega} \nabla v \cdot \nabla u \, d\mathbf{x} \;=\; \sum_{e=1}^{E} \int_{\Omega^e} \nabla v \cdot \nabla u \, d\mathbf{x}. \tag{2}$$

- Given that we know how to integrate over one element, we now know how to integrate over all of $\Omega$ and we are almost ready to implement the variational statement for the Poisson problem,

  *Find $u \in X_0^N$ such that, for all $v \in X_0^N$,*

$$a(v,u) \;=\; (v,f). \tag{3}$$

- It will be helpful to introduce some spaces.
- Let $\mathcal{L}^2$ denote the set of square-integrable functions on $\Omega$,

$$\mathcal{L}^2 \;:=\; \left\{ v \,|\, \int_\Omega v^2 \, dV \;<\; \infty \right\}. \tag{4}$$

- Let $\mathcal{H}^1$ denote the set of functions in $\mathcal{L}^2$ whose gradient is also square-integrable,

$$\mathcal{H}^1 \;:=\; \left\{ v \,\in \mathcal{L}^2 | \int_\Omega \nabla v \cdot \nabla v \, dV \;<\; \infty \right\}. \tag{5}$$

- Practically, we think of $\mathcal{H}^1$ as the space of continuous functions, $C^0$, which is a subset of $\mathcal{H}^1$.
- We are particularly interested in an important subspace, namely, $\mathcal{H}^1_0$, which is the space of functions in $\mathcal{H}^1$ that vanish on the domain boundary wherever Dirichlet conditions are applied,

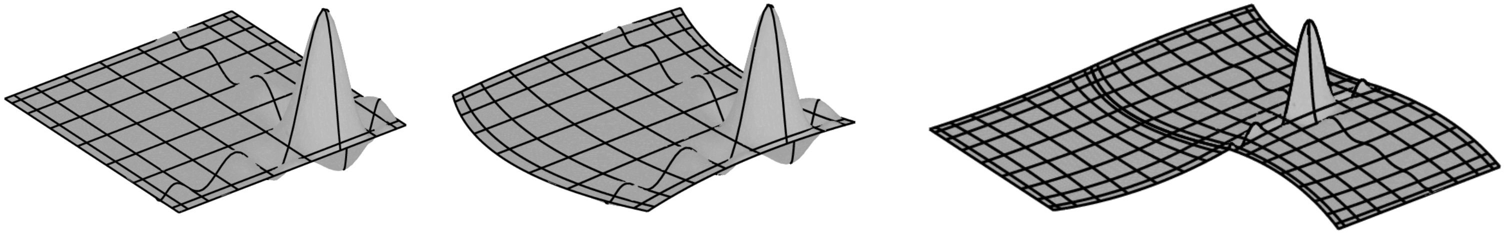
$$\mathcal{H}^1_0 \;:=\; \left\{ v \,\in \mathcal{H}^1 | \, v = 0 \text{ on } \partial\Omega_D \right\}. \tag{6}$$

- The continous weak form for the Poisson equation reads *Find* $\tilde{u} \in \mathcal{H}^1_0$ *such that, for all* $v \in \mathcal{H}^1_0$,

$$a(v, \tilde{u}) \;=\; (v, f). \tag{7}$$

- Our goal is to find a *continuous* $u(\mathbf{x}) \approx \tilde{u}$.

- As in the mono-domain case, we can think of functions in the SEM as *basis functions.*



- In particular, on element $\Omega^e$, we will have

$$v(\mathbf{x})|_{\Omega^e} = \sum_{p=0}^{N}\sum_{q=0}^{N} l_p(r)l_q(s)v^e_{pq} \in \mathbb{P}_N(r,s) \tag{8}$$

$$u(\mathbf{x})|_{\Omega^e} = \sum_{i=0}^{N}\sum_{j=0}^{N} l_i(r)l_j(s)u^e_{ij} \tag{9}$$

$$\mathbf{x}|_{\Omega^e} = \sum_{i=0}^{N}\sum_{j=0}^{N} l_i(r)l_j(s)\mathbf{x}^e_{ij}. \tag{10}$$

- Define

$$\hat{X}^N := \{v|_{\Omega^e} \in \mathbb{P}_N(r,s),\ e = 1:E\} \subset \mathcal{L}^2(\Omega) \tag{11}$$

$$X^N := \{v \in \hat{X}^N \mid v \in \mathcal{H}^1\} \subset \mathcal{H}^1(\Omega) \tag{12}$$

$$X_0^N := \{v \in X^N \mid v = 0 \text{ on } \partial\Omega_D\} \subset \mathcal{H}_0^1(\Omega) \tag{13}$$

- The key point is that functions in $X_0^N$ are

  - *representable* in our SEM basis $\quad (\in\ \hat{X}^N)$
  - *continuous* $\quad (\in\ X^N)$
  - *vanishing on* $\partial\Omega_D$ $\quad (\in\ X_0^N)$

- To cast our variational statement into matrix form we introduce the following notation.

$$\underline{u}^e \;=\; \begin{pmatrix} u^e_{00} \\ u^e_{10} \\ \vdots \\ u^e_{NN} \end{pmatrix} \quad - \quad \textit{local basis coefficients on } \Omega^e$$

$$\underline{u}_L \;=\; \begin{pmatrix} \underline{u}^1 \\ \underline{u}^2 \\ \vdots \\ \underline{u}^E \end{pmatrix} \quad - \quad \textit{unassembled set of local basis coefficients}$$

$$\underline{\bar{u}} \;=\; \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{\bar{n}} \end{pmatrix} \quad - \quad \textit{set of global (aka nodal) basis coefficients, including boundary nodes}$$

$$\underline{u} \;=\; R\underline{\bar{u}} \;=\; \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \quad - \quad \textit{interior-only nodal values.}$$

- *Note:* Generally one simply defines $\underline{u} = R\underline{\bar{u}}$.
  - It's not typical that $R = [I\ O]$.
  - We will not usually expect $\underline{u} = [\bar{u}_1\ \bar{u}_2\ \ldots\ \bar{u}_n]^T$.

- With these definitions and (as a reminder),

$$\begin{aligned} v(\mathbf{x})|_{\Omega^e} &= \sum_{p=0}^{N}\sum_{q=0}^{N} l_p(r)l_q(s)v^e_{pq} \in \mathbb{P}_N(r,s) \\ u(\mathbf{x})|_{\Omega^e} &= \sum_{i=0}^{N}\sum_{j=0}^{N} l_i(r)l_j(s)u^e_{ij} \\ \mathbf{x}|_{\Omega^e} &= \sum_{i=0}^{N}\sum_{j=0}^{N} l_i(r)l_j(s)\mathbf{x}^e_{ij}, \end{aligned}$$

  our unknowns are the local basis coefficients, $\underline{u}_L$.

- Given these, we can reconstruct approximation solutions on each $\Omega^e$ and, hence, on all of $\Omega$.

- For *any* $v,\, u \in \hat{X}^N$,

$$\begin{aligned} a(v,u) &= \sum_{e=1}^{E} \int_{\Omega^e} \nabla v \cdot \nabla u \, d\mathbf{x} && (14) \\ &= \sum_{e=1}^{E} (\underline{v}^e)^T A^e \underline{u}^e && (15) \\ &= \underbrace{\begin{pmatrix} \underline{v}^1 \\ \underline{v}^2 \\ \vdots \\ \underline{v}^e \\ \vdots \\ \underline{v}^E \end{pmatrix}^T}_{\underline{v}_L^T} \underbrace{\begin{pmatrix} A^1 & & & & & \\ & A^2 & & & & \\ & & \ddots & & & \\ & & & A^e & & \\ & & & & \ddots & \\ & & & & & A^E \end{pmatrix}}_{A_L} \underbrace{\begin{pmatrix} \underline{u}^1 \\ \underline{u}^2 \\ \vdots \\ \underline{u}^e \\ \vdots \\ \underline{u}^E \end{pmatrix}}_{\underline{u}_L} && (16) \\ &= \underline{v}_L^T A_L \, \underline{u}_L . && (17) \end{aligned}$$

- $A_L$ is the *unassembled* system matrix

- $A^e$ is the *local* system matrix, which we've seen in the monodomain case,

$$A^e = \frac{2}{L_x^e}\left(\hat{B}\otimes\hat{A}\right) + \frac{2}{L_y^e}\left(\hat{A}\otimes\hat{B}\right) \quad \text{if } \Omega^e = L_x \times L_y \text{ rectangle} \tag{18}$$

$$A^e = \mathbf{D}^T\mathbf{G}^e\mathbf{D} \quad \text{for general geometries,} \tag{19}$$

$$\mathbf{D} = \begin{pmatrix} \hat{I}\otimes\hat{D} \\ \hat{D}\otimes\hat{I} \end{pmatrix}, \qquad G_{ij}^e\Big|_{\xi_p\xi_q} = \sum_{k=1}^{d}\left(\frac{\partial r_i}{\partial x_k^e}\frac{\partial r_j}{\partial x_k^e}\right)\Bigg|_{\xi_p\xi_q} \tag{20}$$

- In a similar fashion, we have for all $v,\ f \in \hat{X}^N$,

$$(v,f) = \underline{v}_L^T\, B_L\, \underline{f}_L \tag{21}$$

$$B_L = \text{block-diag}(B^e) \qquad\qquad (\textit{diagonal matrix}) \tag{22}$$

$$B^e = \frac{L_x^e L_y^e}{4}\left(\hat{B}\otimes\hat{B}\right) \quad \text{if } \Omega^e = L_x \times L_y \text{ rectangle} \tag{23}$$

$$B^e = \mathcal{J}^e\left(\hat{B}\otimes\hat{B}\right) \qquad \text{for general geometries.} \tag{24}$$

- The key point is that we already know how to evaluate all of the operators.

- In fact, that is the central operation for *matrix-free* forms, which are needed for timestepping and for iterative solvers.

- There is one more novel item that we need, and it relates to ensuring that our solution is continuous, $u \in X^N \subset \hat{X}^N$.

- The easiest way to ensure that $u \in X^N \subset \mathcal{H}^1$ is to require that $u(\mathbf{x})$ is continuous.
- Thus, for any two elements $\Omega^e$ and $\Omega^{\hat{e}}$,

$$\mathbf{x}^e_{ij} = \mathbf{x}^{\hat{e}}_{\hat{\imath},\hat{\jmath}} \implies u^e_{ij} = u^{\hat{e}}_{\hat{\imath},\hat{\jmath}}. \tag{25}$$

- Imposition of function continuity follows just as it did with the periodic restriction/prolongation operator.
- There, we let $Q = R^T$ be the operator that mapped from the *necessary* unknown coefficients to the *representation* coefficients.
- At that point, we had some discussion as to whether the operator should be called $R^T$ or $Q$, with the former being associated with boundary conditions and the latter associated with function continuity.
- Here, we will need both, distinct, operators.

- Let $Q$ be a rectangular $N_L \times \bar{n}$ Boolean matrix a single "1" on each of $N_L = E(N+1)^d$ rows.

- The number of columns of $Q$ is equal to the number of unique vertices.

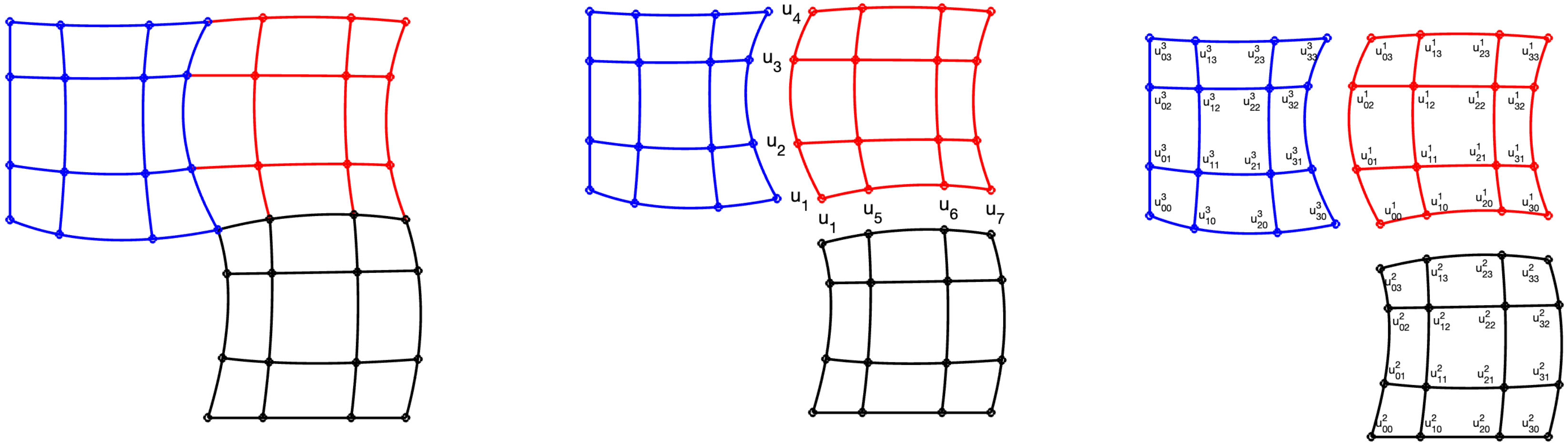
- Application of $Q$ to a vector, $\underline{\bar{u}}$, yields a local vector,

$$\underline{u}_L \;=\; Q\underline{\bar{u}}, \tag{26}$$

  such that entries in $\underline{u}_L$ are *copies* of data in $\underline{\bar{u}}$.

- By having redundant values stored along shared edges, we know that the function representation associated with $\underline{u}_L = Q\underline{\bar{u}}$ is *continuous* ($\in\; X^N$).

- An example is illustrated in the figures above and below.



$u_4$
$u_3$
$u_2$
$u_1$
$u_1$
$u_5$
$u_6$
$u_7$
$u^3_{03}$ $u^3_{13}$ $u^3_{23}$ $u^3_{33}$
$u^3_{02}$ $u^3_{12}$ $u^3_{22}$ $u^3_{32}$
$u^3_{01}$ $u^3_{11}$ $u^3_{21}$ $u^3_{31}$
$u^3_{00}$ $u^3_{10}$ $u^3_{20}$ $u^3_{30}$
$u^1_{03}$ $u^1_{13}$ $u^1_{23}$ $u^1_{33}$
$u^1_{02}$ $u^1_{12}$ $u^1_{22}$ $u^1_{32}$
$u^1_{01}$ $u^1_{11}$ $u^1_{21}$ $u^1_{31}$
$u^1_{00}$ $u^1_{10}$ $u^1_{20}$ $u^1_{30}$
$u^2_{03}$ $u^2_{13}$ $u^2_{23}$ $u^2_{33}$
$u^2_{02}$ $u^2_{12}$ $u^2_{22}$ $u^2_{32}$
$u^2_{01}$ $u^2_{11}$ $u^2_{21}$ $u^2_{31}$
$u^2_{00}$ $u^2_{10}$ $u^2_{20}$ $u^2_{30}$

- **Q:** How many columns does $Q$ have in this case?
  How many rows?

- As in the monodomain case, if $u(\mathbf{x}) \in X_0^N$, there exists a rectangular restriction matrix $R$ such that $\underline{\bar{u}} = R^T \underline{u}$ will be zero for all nodes $\mathbf{x}_i \in \partial\Omega_D$.

- The full Laplacian matrix for the SEM is thus,

$$A = R \underbrace{Q^T A_L Q}_{\bar{A}} R^T \tag{27}$$

- We refer to $A_L$ as the *unassembled* Laplacian matrix and $\bar{A}$ as the *assembled Neumann operator*.

- It has a null space corresponding to the constant function.

- In a similar fashion, the mass matrix and discrete convection operator are

$$B = R \underbrace{Q^T B_L Q}_{\bar{B}} R^T, \qquad C \;=\; R \underbrace{Q^T C_L Q}_{\bar{C}} R^T, \tag{28}$$

  where $B_L =$ block-diagonal$(B^e)$ and $C_L =$ block-diagonal$(C^e)$, with $B^e$ and $C^e$ the element-based counterparts to (**??**) and (**??**), respectively.

- We remark that the factor $QR^T$ on the right of each of (27)–(28) results from $u \in X^N$ $(Q)$ and $u \in X_0^N \subset X^N$ $(R^T)$, whereas the prefactor $RQ^T$ results from the test functions $v$ being in the same space.

- The governing mathematics/physics dictates the contents of the block-diagonal matrices, $A_L$, $B_L$, and $C_L$, which are sandwiched between these operators.

- It is thus relatively easy to construct new operators provided that one understands the constraints on spaces for the test $(v)$ and trial $(u)$ functions.

- Since we know how to *apply* the operators, we also know how to *solve* the relevant systems given that we will use iterative methods.

- It is somewhat remarkable that the SEM works, i.e.., that we get exponential convergence even though our basis functions are only $C^0$ continuous.

- Why is the convergence so fast?

- It has to do with the fact that, for smooth solutions, our *approximation* is converging exponentially fast on *each* local subdomain.

- High-order continuity is not requisite.

- If the solution is $C^\infty$, we will converge exponentially fast to that solution.

- Note that our derivatives are *not* continuous.

- The jump, however, goes to zero exponentially fast with $N$.

- Some comments about data structures.

- We generally like to keep things in *local representation*, $\underline{u}_L$, which allows us to trivially evaluate gradients, integrals, etc.

- This form is convenient for parallel computing.

- It does change, however, how one implements a matrix-vector product of the form $\underline{w} = A\underline{u}$.

- Assuming that $\underline{u}_L$ denotes a function that is already in $X_0^N$, we would modify the matrix-vector product as

$$\begin{aligned} \underline{w} &= A\underline{u} = RQ^T A_L Q R^T \underline{u} && (29) \\ \underline{w}_L &= QR^T R Q^T A_L Q R^T \underline{u} && (30) \\ &= QR^T R Q^T A_L \underline{u}_L && (31) \end{aligned}$$

- Note that $M := R^T R$ is a diagonal *mask* matrix comprising 1s and 0s on the diagonal, with 0 for any points on $\partial\Omega_D$.
- Its action is to zero out any Dirichlet values.
- We can also zero out when the data is in its *local* representation by defining a diagonal matrix $M_L \in \mathbb{R}^{n_L \times n_L}$, where $n_L = E(N+1)^d$.

- The matrix-vector product then becomes

$$\begin{aligned} \underline{w}_L &= QR^T RQ^T A_L QR^T \underline{u} && (32) \\ &= M_L\, QQ^T A_L \underline{u}_L. && (33) \end{aligned}$$

- $QQ^T$ ensures that "$A\underline{u}$" is in $\mathcal{H}^1$ and $M_L$ assures that $\underline{w}_L$ is in $\mathcal{H}^1_0$, which is what we require.

- It is important to recognize that we can apply conjugate gradients to vectors in $\mathcal{R}(M_L)$ because it is a *projector.*

- As long as the rhs is in $\mathcal{R}(M_L)$, all of the Krylov search directions will also be in $\mathcal{R}(M_L)$ and the iteration will converge.

$$\underline{u}^e = \begin{pmatrix} u^e_{0,0} \\ u^e_{1,0} \\ \vdots \\ u^e_{N,N} \end{pmatrix}$$

$$\underline{u}_L = \begin{pmatrix} \underline{u}^1 \\ \vdots \\ \underline{u}^e \\ \vdots \\ \underline{u}^E \end{pmatrix} = QR^T\underline{u}$$

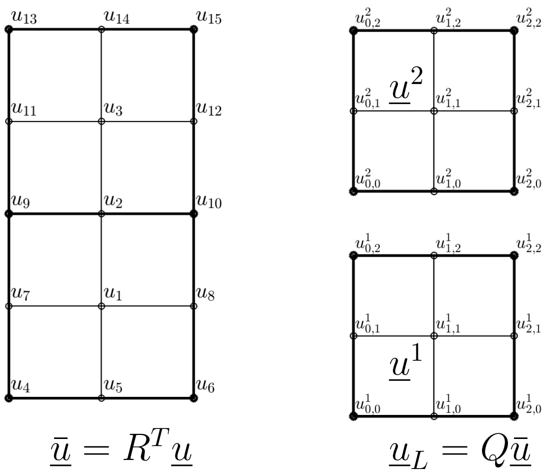

$u_{13}$ $u_{14}$ $u_{15}$
$u_{11}$ $u_3$ $u_{12}$
$u_9$ $u_2$ $u_{10}$
$u_7$ $u_1$ $u_8$
$u_4$ $u_5$ $u_6$
$\bar{\underline{u}} = R^T\underline{u}$
$u^2_{0,2}$ $u^2_{1,2}$ $u^2_{2,2}$
$u^2_{0,1}$ $\underline{u}^2$ $u^2_{1,1}$ $u^2_{2,1}$
$u^2_{0,0}$ $u^2_{1,0}$ $u^2_{2,0}$
$u^1_{0,2}$ $u^1_{1,2}$ $u^1_{2,2}$
$u^1_{0,1}$ $u^1_{1,1}$ $u^1_{2,1}$
$\underline{u}^1$
$u^1_{0,0}$ $u^1_{1,0}$ $u^1_{2,0}$
$\underline{u}_L = Q\bar{\underline{u}}$

Figure 1: Illustration of $\underline{u}_L = Q\bar{\underline{u}}$=$QR^T\underline{u}$ for $(E,N)$=$(2,2)$: (center) $\underline{u}$=$[u_1\ u_2\ u_3]^T$ is the set of active (interior) basis coefficients; $\bar{\underline{u}}$=$[u_1\ \dots\ u_{15}]^T$=$R^T\underline{u}$ is the set of all global basis coefficients, extended by zero to the domain boundary; (right) $\underline{u}_L$=$Q\bar{\underline{u}}$ is the set of local basis coefficients represented element-by-element (i.e., as $\underline{u}^e$, $e$=$1,\dots,E$). For any $u \in X^N_0 \subset \mathcal{H}^1_0$, all representations contain the same information.

- The easiest way to ensure that $u \in X^N \subset \mathcal{H}^1$ is to require that $u(\mathbf{x})$ is continuous.
- For any two elements $\Omega^e$ and $\Omega^{\hat{e}}$,

$$\mathbf{x}^e_{ij} = \mathbf{x}^{\hat{e}}_{\hat{\imath},\hat{\jmath}} \implies u^e_{ij} = u^{\hat{e}}_{\hat{\imath},\hat{\jmath}}. \tag{34}$$

- Imposition of continuity for functions in $X^N$ is accomplished by assigning a global numbering, $i_g = g^e_{i,j}$ for each vertex $i,j \in \{0,\dots,N\}^2$ in each element, $e = 1,\dots,E$.
- Vertex pairs (or triplets, etc.) satisfying (34) are assigned the same global index, $i_g$.
- For example, in Fig. 1, we have $i_g = 2$ for $\mathbf{x}^1_{1,2}$ and $\mathbf{x}^2_{1,0}$.
- The global index is $g^1_{1,2} = g^2_{1,0}$=2.

- From the global numbering, one can construct a Boolean matrix $Q$ such that the set of basis coefficients $\underline{u}_L = Q\underline{\bar{u}}$ represents a $C^0$-continuous function $u(\mathbf{x})$, where $\underline{\bar{u}}$ is the vector of globally- (i.e., uniquely) numbered coefficients, including the boundary values.

- Assume that there are $\bar{n}$ contiguously-numbered global vertices, $i_g \in \bar{\mathcal{I}} := \{1, \dots, \bar{n}\}$, and let $k = 1 + i + (N+1)j + (N+1)^2(e-1) \in \{1, \dots, E(N+1)^2\}$ represent a lexicographical indexing of the local degrees-of-freedom (dofs), $\{u^e_{ij}\}$.

- Then the $k$th row of $Q$ will be $\underline{\hat{e}}^T_{i_g}$, where $i_g = g^e_{ij}$ and $\underline{\hat{e}}_{i_g}$ is the $i_g$th column of the $\bar{n} \times \bar{n}$ identity matrix.

- Note that the action of $Q$ is to *copy* (scatter) values of a global vector to their local counterparts.

- (In Fig. 1, compare the center panel, $\underline{\bar{u}}$, to the right panel, $\underline{u}_L = Q\underline{\bar{u}}$.)

- Conversely, application of $Q^T$ *sums* (gathers) local contributions to their global counterparts.

- Thus, the action of $QQ^T$ is equivalent to an exchange-and-sum of values among shared dofs (sometimes referred to as direct-stiffness summation [Strang & Fix, '73].

- This gather-scatter operation is the communication intensive phase of parallel operator evaluation.

- Note that, in contrast to high-order finite difference stencils, the SEM/FEM stencil depth is unity—one only needs to pass surface data between subdomain interfaces, rather than multiple layers of data.

- In this sense, SEM operator evaluation is *communication minimal.*