

Let's consider the RT1 example: three interior points map to each edge with two nodes. Three parametrized values transfer via the orthogonal projection to three points along the tangent line of one edge. I now want to obtain the values of the existing 1D polynomial basis that best fit these three point values using least squares - how do I form the matrix problem and solution given my current Vandermonde matrix for the edge polynomial?

ChatGPT said:

Given your RT1 element, you have:

- Three interior points that project orthogonally onto an edge.
- A 1D polynomial basis on the edge (formed using Jacobi polynomials).
- An inverse Vandermonde matrix that maps nodal values to polynomial coefficients.

Your goal is to fit the three projected values using the 1D edge basis in a least-squares sense.

Step 1: Define the System

You are given three projected values U_i (from the RT interior) at locations s_0^i along the edge. The best-fit function in your edge polynomial basis (which consists of two nodes for RT1) is:

$$U_{\text{edge}}(s_0) = \sum_{k=0}^N c_k P_k(s_0)$$

where $P_k(s_0)$ are the 1D Jacobi basis polynomials and c_k are the coefficients we seek.

To solve this, we minimize the least-squares error:

$$\sum_{i=1}^3 \left(\sum_{k=0}^N c_k P_k(s_0^i) - U_i \right)^2.$$

This results in a linear system:

$$\mathbf{A}\mathbf{c} = \mathbf{b}$$

where:

- $A_{ik} = P_k(s_0^i)$ (evaluation of basis polynomials at the projected locations).
- c_k are the unknown coefficients.
- $b_i = U_i$ (values projected from the interior).

Step 2: Construct the Matrix System

Given:

- Three projected values:

$$\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}$$

- A 1D basis with two functions $P_0(s_0)$, $P_1(s_0)$ (since RT1 edges have two nodes).
- Three sampling locations s_0^i along the edge.

The least-squares matrix is:

$$\mathbf{A} = \begin{bmatrix} P_0(s_0^1) & P_1(s_0^1) \\ P_0(s_0^2) & P_1(s_0^2) \\ P_0(s_0^3) & P_1(s_0^3) \end{bmatrix}$$

and the right-hand side is:

$$\mathbf{b} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}.$$

The least-squares solution is:

$$\mathbf{c} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}.$$

This provides the best-fit coefficients c_k in the Jacobi polynomial basis.

Step 3: Use the Vandermonde Matrix for Nodal Values

Since you already have the **Vandermonde matrix** \mathbf{V} for the edge, which relates **nodal values** to **Jacobi coefficients**, we can compute the **nodal values** $U_{\text{edge, nodes}}$ from the fitted coefficients.

Using:

$$U_{\text{edge, nodes}} = \mathbf{V}\mathbf{c}$$

you obtain the **edge nodal values** needed for flux calculations.

Final Steps for Implementation

1. Compute the basis values $P_k(s_0^i)$ at the three projected locations.
2. Assemble the least-squares matrix \mathbf{A} and RHS \mathbf{b} .
3. Solve for the best-fit coefficients:

$$\mathbf{c} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$$

4. Convert to nodal values using the edge **Vandermonde matrix**:

$$U_{\text{edge, nodes}} = \mathbf{V}\mathbf{c}.$$

Projection of values from Interior nodes to parameterized edge

A **convenient expression** for projecting values from **(r, s) space** to a **tangential parameter** s' along an edge originating at (r_0, s_0) can be derived as follows:

General Form for Projection to a Tangential Parameter s'

Given:

- A reference point (r_0, s_0) on the edge.
- A **tangential direction vector** $\mathbf{t} = (t_r, t_s)$, which is the **unit tangent** along the edge.
- A general interior point (r, s) .

We define the **tangential projection** as:

$$s' = (r - r_0)t_r + (s - s_0)t_s$$

where:

- s' is the **scalar parameter** measuring displacement **along the edge tangent** starting from (r_0, s_0) .
- $\mathbf{t} = (t_r, t_s)$ is the **unit tangent vector** of the edge.

Alternative Expression Using a Normalized Tangent

If the edge is parameterized by an arc-length parameterization, we can normalize s' so that it lies in a canonical range (e.g., $[-1, 1]$) by dividing by the edge length L :

$$s' = \frac{(r - r_0)t_r + (s - s_0)t_s}{L}$$

where:

$$L = \sqrt{(r_1 - r_0)^2 + (s_1 - s_0)^2}$$

for an edge between endpoints (r_0, s_0) and (r_1, s_1) .