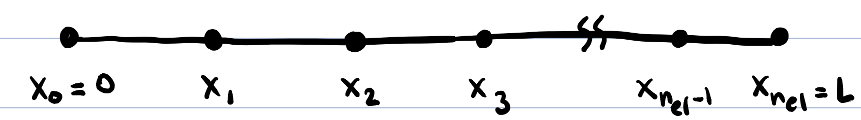
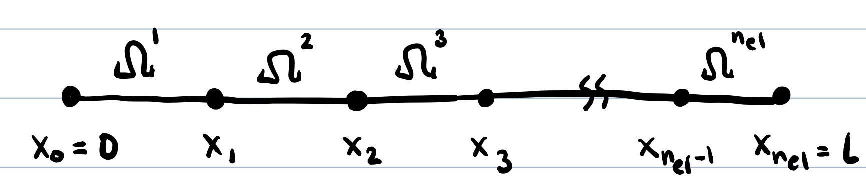
1D Model Problem: Finite Element Approximations:

As discussed in the previous lecture, the key ingredient in any Galerkin method is the selection of suitable test and trial functions. Today, we will discuss a particularly appealing selection of test and trial functions: so-called finite element functions.

The cornerstone in building a space of finite element functions is the creation of a suitable mesh. In this direction, let us partition the domain of interest $(0, L)$ into n_{el} non-overlapping subintervals (x_{e-1}, x_e) for $e = 1, \ldots, n_{el}$ such that $0 = x_0 < x_1 < \cdots < x_{n_{el}} = L$. The points $\{x_e\}_{e=0}^{n_{el}}$ are called nodes. Visually:



Each subinterval (x_{e-1}, x_e) is referred to as an element. We denote the e^{th} element (x_{e-1}, x_e) as Ω^e:



and we refer to the set of elements $\{\Omega^e\}_{e=1}^{n_{el}}$ as the finite element mesh $\mathcal{M}^h$. To simplify notation, we also denote the domain

$(0,L)$ as Ω. We define the mesh size h^e of the e^{th} element to be $x_e - x_{e-1}$, and we define the global mesh size h to be $\max \{h^e\}_{e=1}^{n_{el}}$. With the above in hand, finite element functions are then defined as piecewise polynomial functions with respect to the mesh.

In order for finite element functions to be used as test and trial functions, they must be square integrable with square integrable derivative. The following result dictates the required continuity between adjacent elements to meet such a requirement:

Theorem: A piecewise infinitely differentiable function $v : \bar{\Omega} \to \mathbb{R}$ over the mesh $\mathcal{M}^h$ belongs to $H^k(\Omega)$ if and only if $v \in C^{k-1}(\bar{\Omega})$.

To prove the above result requires heavy use of functional analysis and specifically the theory of distributions, but it has a powerful implication: finite element functions are square integrable with square integrable derivative if they are globally C^0-continuous. Thus, we are particularly interested in the following space of finite element functions, referred to as a finite element approximation space:

$$\mathcal{P}^k_{cont.}(\mathcal{M}^h) := \left\{ v^h \in C^0(\bar{\Omega}) : v^h|_{\Omega^e} \in \mathcal{P}^k(\Omega^e) \text{ for } e = 1, \dots, n_{el} \right\}$$

where k is a specified polynomial degree and $\mathcal{P}^k(\Omega^e)$ is the space of polynomials of degree k defined over the e^{th} element.

When $k=0$, $\mathcal{P}^k_{\text{Cont.}}(\mathcal{M}^h)$ consists of only globally constant functions, no matter how many elements are in the finite element mesh. Thus we are specifically interested in the setting $k \geq 1$.

Given a proper space of finite element functions to work with, we are now in a position to define a corresponding set of trial solutions and space of test functions to arrive at a finite element approximation of our 1D model problem. In particular, we select:

$$\mathcal{S}^h := \left\{ v^h \in \mathcal{P}^k_{\text{Cont.}}(\mathcal{M}^h) : v^h(0) = g_0,\ v^h(L) = g_L \right\}$$

$$\mathcal{V}^h := \left\{ v^h \in \mathcal{P}^k_{\text{Cont.}}(\mathcal{M}^h) : v^h(0) = v^h(L) = 0 \right\}$$

The space $\mathcal{S}^h$ is equal to $\mathcal{V}^h + g^h$ for any given $g^h \in \mathcal{S}^h$, so the above selections give rise to the following Bubnov-Galerkin finite element approximation of our 1D model problem:

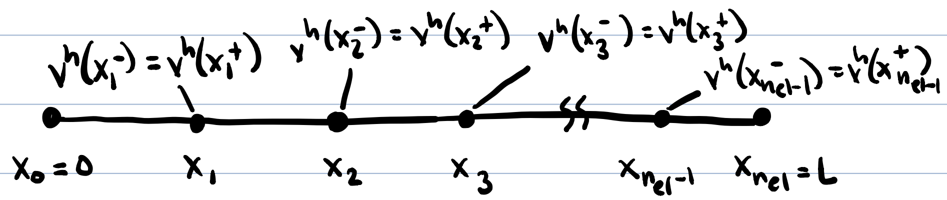
$$(G) \begin{cases} \text{Find } u^h \in \mathcal{S}^h \text{ such that:} \\ \displaystyle\int_0^L \kappa\, u^h_{,x}\, w^h_{,x}\, dx = \int_0^L f w^h dx \\ \text{for all } w^h \in \mathcal{V}^h. \end{cases}$$

To arrive at a matrix form of the above problem, however, we need to determine a suitable set of basis functions for our finite element space of test functions.

Let us begin by identifying the dimension of the space $\mathcal{P}^k_{\text{cont.}}(\mathcal{M}^h)$. The dimension of the space of polynomials of degree k over a given element is $k+1$ since we can write any polynomial in the form:

$$p(x) = \sum_{i=0}^{k} c_i x^i$$

Thus, the space of piecewise discontinuous polynomials of degree k has dimension $n_{el} * (k+1)$. To enforce continuity between elements, we need to enforce that $v^h(x_A^-) = v^h(x_A^+)$ for every $v^h \in \mathcal{P}^k_{\text{cont.}}(\mathcal{M}^h)$ and inter-element boundary $A = 1, \ldots, n_{ele}-1$:



Thus, the dimension of $\mathcal{P}^k_{\text{cont.}}(\mathcal{M}^h)$ is $(n_{el}-1)$ less than that of the space of discontinuous piecewise polynomials of degree k. Consequently:

$$\dim\left(\mathcal{P}^k_{\text{cont.}}(\mathcal{M}^h)\right) = n_{el} + 1 + (k-1) * n_{el}$$

Likewise, the dimension of $\mathcal{V}^h$ is two less than that of $\mathcal{P}^k_{\text{cont.}}(\mathcal{M}^h)$ as two boundary conditions $(v^h(0) = 0,\ v^h(L) = 0)$ are enforced on functions in $\mathcal{V}^h$, so:

$$\dim(\mathcal{V}^h) = n_{el} - 1 + (k-1) * n_{el}$$

Now we know we need to find $n = n_{el} - 1 + (k-1) * n_{el}$ basis functions.

Let us start with the easiest case, $k=1$. Then:

$$\dim\left(\mathcal{P}^1_{\text{cont.}}(\mathcal{M}^h)\right) = n_{el}+1, \quad \dim\left(\mathcal{V}^h\right) = n_{el}-1$$

Let us also find a basis first for $\mathcal{P}^1_{\text{cont.}}(\mathcal{M}^h)$. This will enable us, as you will see, to find a basis for $\mathcal{V}^h$ and to find a suitable function $g^h \in \mathcal{S}^h$ satisfying the desired nonhomogeneous boundary conditions.

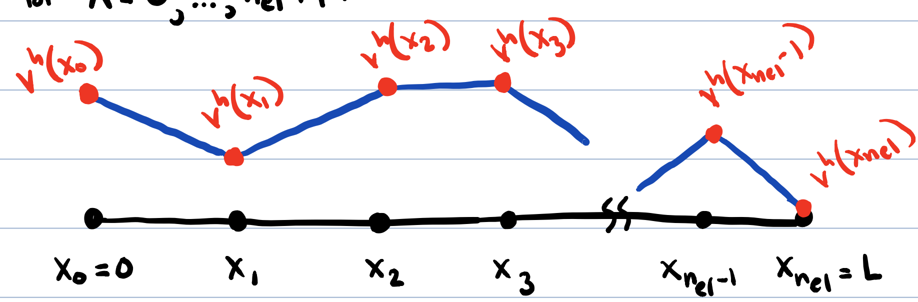
Suppose $\{N_A\}_{A=0}^{n_{el}}$ is a basis for $\mathcal{P}^1_{\text{cont.}}(\mathcal{M}^h)$. Then we can express each $v^h \in \mathcal{P}^1_{\text{cont.}}(\mathcal{M}^h)$ as:

$$v^h(x) = \sum_{A=0}^{n_{el}} v^h_A \, N_A(x)$$

where $\{v^h_A\}_{A=0}^{n_{el}}$ are the degrees of freedom for v^h corresponding to the chosen basis. Thus, to find a suitable basis, we can first choose degrees of freedom and then infer the basis. It is rather natural to choose nodal values as degrees of freedom, so suppose that:

$$v^h_A = v^h(x_A)$$

for $A = 0, \ldots, n_{el}+1$:



Then, since:

$$v^h(x_B) = \sum_{A=0}^{n_{el}} v^h(x_A)\, N_A$$

for $B = 0, \dots, n_{el}$, we have:

$$N_A(x_B) = \begin{cases} 1 & \text{if } A = B \\ 0 & \text{otherwise} \end{cases}$$

for $A = 0, \dots, n_{el}$ and $B = 0, \dots, n_{el}$. The above constitutes $n_{el}+1$ equations for each basis function, so it uniquely defines each basis function. In fact, over each element Ω^e, the A^{th} basis function is linear and thus can be written as:

$$N_A(x) = c_1 x + c_2 \qquad \text{for } x \in (x_{e-1}, x_e)$$

The unknown coefficients c_1 and c_2 are then determined by the values $N_A(x_{e-1})$ and $N_A(x_e)$ as these give rise to two equations for our two unknowns:

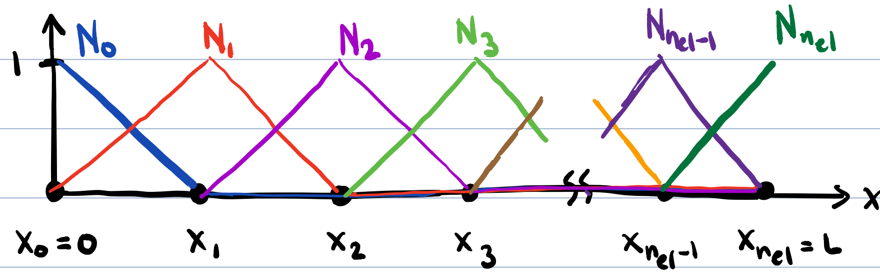
$$N_A(x_{e-1}) = c_1 x_{e-1} + c_2$$

$$N_A(x_e) = c_1 x_e + c_2$$

Solving for the form of each basis function over each element gives rise to the final expression for $A = 0, \dots, n_{el}$:

$$N_A(x) = \begin{cases} \left(\dfrac{x - x_{A-1}}{x_A - x_{A-1}}\right) & \text{for } x_{A-1} \leq x \leq x_A \\ \left(\dfrac{x_{A+1} - x}{x_{A+1} - x_A}\right) & \text{for } x_A \leq x \leq x_{A+1} \\ 0 & \text{otherwise} \end{cases}$$

Visually:



The above are commonly referred to as piecewise linear finite element basis functions or simply hat functions. With respect to the above basis functions, every function $v^h \in \mathcal{V}^h$ takes the form:

$$v^h(x) = \overset{\nearrow 0}{v^h(0)} N_0(x) + \sum_{A=1}^{n_{el}-1} v^h(x_A) N_A(x) + \overset{\nearrow 0}{v^h(L)} N_{el}(x)$$

$$= \sum_{A=1}^{n_{el}-1} v^h(x_A) N_A(x)$$

Noting that $n = n_{el} - 1$, we immediately see $\{N_A\}_{A=1}^{n}$ is a basis for $\mathcal{V}^h$! Moreover, every function $v^h \in \mathcal{S}^h$ takes the form:

$$v^h(x) = \overset{\nearrow g_0}{v^h(0)} N_0(x) + \sum_{A=1}^{n_{el}-1} v^h(x_A) N_A(x) + \overset{\nearrow g_L}{v^h(L)} N_{el}(x)$$

$$= \underbrace{\left(g_0 N_0(x) + g_L N_{el}(x)\right)}_{\in \mathcal{S}} + \underbrace{\sum_{A=1}^{n_{el}-1} v^h(x_A) N_A(x)}_{\in \mathcal{V}}$$

Thus we can select $g^h(x) = g_0 N_0(x) + g_L N_{el}(x)$! This is due to the fact that the basis functions are interpolatory at nodes.

At this stage, it useful to pause and identify particular properties that piecewise linear finite element basis functions happen to satisfy:

1. They are interpolatory. That is:

$$N_A(x_B) = \begin{cases} 1 & \text{if } A = B \\ 0 & \text{if } A \neq B \end{cases}$$

2. They are locally supported. That is:

$$N_A(x) = 0 \quad \text{if } x \notin (x_{A-1}, x_{A+1})$$

3. They form a partition of unity. That is:

$$\sum_{A=0}^{n_{el}} N_A(x) = 1 \quad \text{for } x \in [0, L]$$

4. They are non-negative. That is:

$$N_A(x) \geq 0 \quad \text{for } x \in [0, L]$$

As already demonstrated above, the interpolatory nature of piecewise linear finite element basis functions dramatically simplifies the imposition of boundary conditions. The local support of piecewise linear finite element basis functions gives rise to sparse stiffness matrices. In particular, the stiffness matrix associated with a linear finite element method is tridiagonal:

$$K_{AB} = 0 \quad \text{if} \quad |A-B| > 1$$

This sparsity can be exploited in linear system solution. The local support of piecewise linear finite element basis functions can also be exploited in the formation of the stiffness and forcing vector, a property that will be explained in detail in a future lecture.

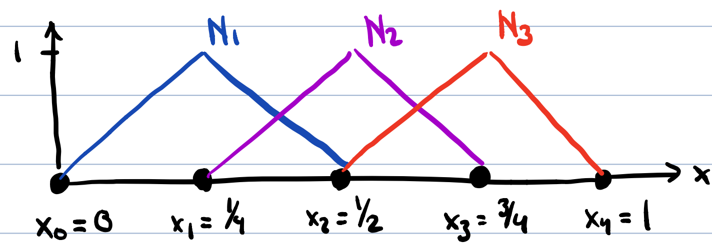
A Bubnov-Galerkin method employing piecewise linear finite element basis functions is commonly referred to as a piecewise linear finite element method. Such a method is the simplest finite element method in practice.

EXAMPLE

To illustrate how we can use piecewise linear finite element functions in practice to solve a problem, consider again the simple case when:

$$\kappa = 1, \; f = 1, \; L = 1, \; g_0 = 0, \; g_L = 0$$

and suppose a Bubnov-Galerkin method is used to approximate the model problem of interest using piecewise linear finite element functions defined on a mesh of four equally sized elements:



The Galerkin finite element solution then takes the form:

$$u^h(x) = d_1 N_1(x) + d_2 N_2(x) + d_3 N_3(x)$$

and the finite element degrees of freedom d_1, d_2, and d_3 may be found via the solution of:

$$\underline{\underline{K}}\, \underline{d} = \underline{f}$$

where:

$$K_{AB} = \int_0^1 N_{A,x} N_{B,x}\, dx$$

$$F_A = \int_0^1 N_A\, dx$$

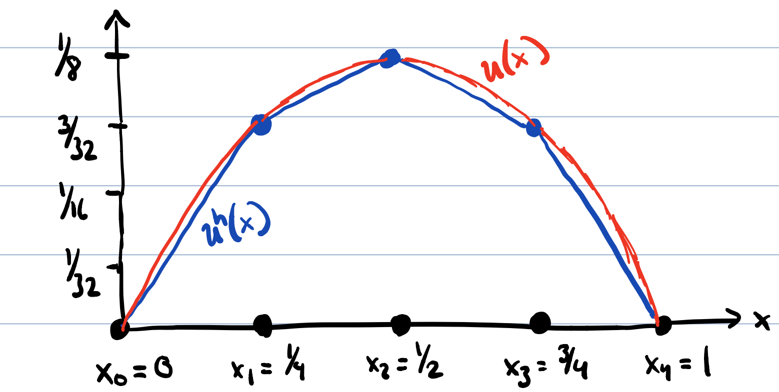
Straightforward calculations show:

$$\underline{\underline{K}} = \begin{bmatrix} 8 & -4 & 0 \\ -4 & 8 & -4 \\ 0 & -4 & 8 \end{bmatrix} \qquad \underline{F} = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}$$

so:

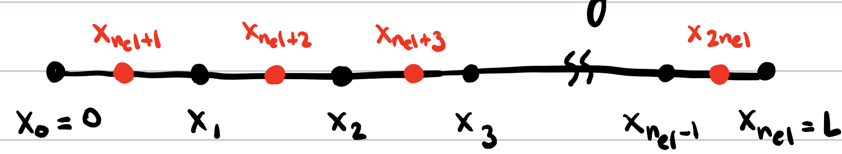
$$\underline{d} = \underline{\underline{K}}^{-1} \underline{F} = \begin{bmatrix} 3/32 \\ 1/8 \\ 3/32 \end{bmatrix}$$

Plotting the Galerkin finite element solution against the exact solution $u(x) = \frac{1}{2} x(1-x)$ reveals the finite element solution is exact at the finite element mesh nodes:

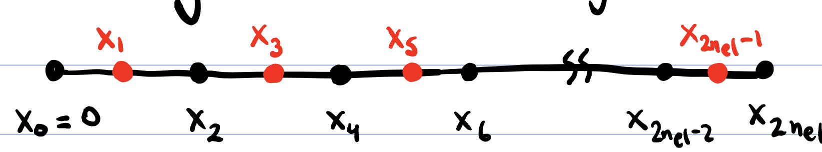


We continue now by identifying finite element basis functions of degree $k=2$. As in the setting of degree $k=1$, we first choose degrees of freedom and then infer the corresponding basis functions.

The dimension of the finite element approximation space $\mathcal{P}^2_{\text{cont.}}(\mathcal{M}^h)$ is $2n_{el}+1$, n_{el} more than the space $\mathcal{P}^1_{\text{cont.}}(\mathcal{M}^h)$. Thus the nodal values at interelement boundaries do not give enough degrees of freedom for a finite element function $v^h \in \mathcal{P}^2_{\text{cont.}}(\mathcal{M}^h)$. We need n_{el} more. To overcome this issue, we introduce an additional node in the middle of each element. Visually:



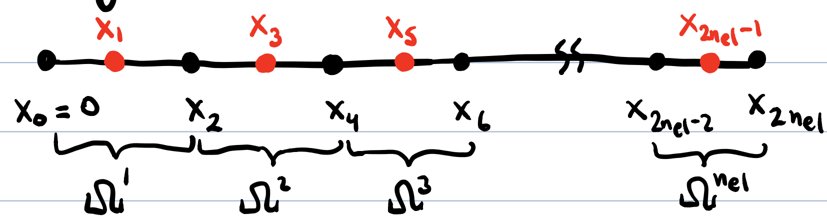
It is customary to renumber the nodes so they are ordered:



Then the e^{th} element is:

$$\Omega^e = (x_{2e-2}, x_{2e})$$

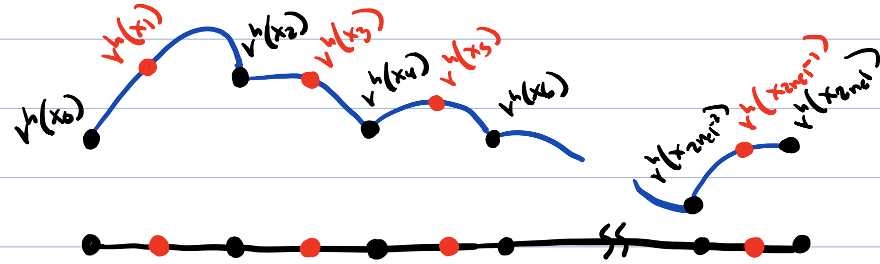
Visually:



Given the increased number of nodes, we then choose the nodal values:

$$\{v^h(x_A)\}_{A=0}^{2n_{el}}$$

as degrees of freedom:



The basis functions $\{N_A\}_{A=0}^{2n_{el}}$ corresponding to these degrees of freedom satisfy:

$$N_A(x_B) = \begin{cases} 1 & \text{if } A = B \\ 0 & \text{otherwise} \end{cases}$$

just like piecewise linear finite element basis functions. Also just like piecewise linear finite element basis functions, we can determine the exact form of each basis function by isolating to individual elements. For example, over the $e^{\underline{th}}$ element, the $A^{\underline{th}}$ basis function is quadratic and thus can be expressed as:

$$N_A(x) = c_1 x^2 + c_2 x + c_3$$

The three unknown coefficients can be determined by noting the $A^{\underline{th}}$ basis function satisfies:

$$N_A(x_{2e-2}) = \begin{cases} 1 & \text{if } A = 2e-2 \\ 0 & \text{otherwise} \end{cases}$$

$$N_A(x_{2e}) = \begin{cases} 1 & \text{if } A = 2e \\ 0 & \text{otherwise} \end{cases}$$

at the ends of the element and:

$$N_A(x_{2e-1}) = \begin{cases} 1 & \text{if } A = 2e-1 \\ 0 & \text{otherwise} \end{cases}$$

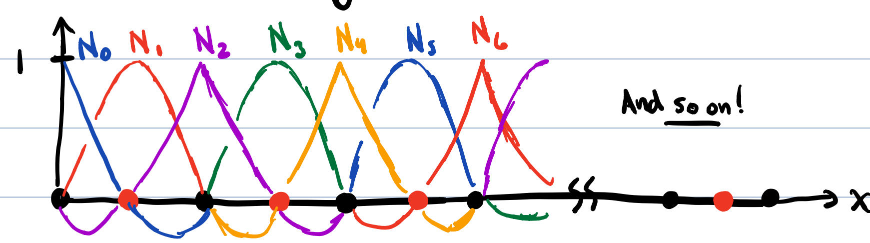
at the node in the middle of the element. Solving for the form of each basis function over each element gives rise to the expression:

$$N_{2A}(x) = \begin{cases} \left(\dfrac{x - x_{2A-2}}{x_{2A} - x_{2A-2}}\right)\left(\dfrac{x - x_{2A-1}}{x_{2A} - x_{2A-2}}\right) & \text{for } x_{2A-2} \leq x \leq x_{2A} \\ \left(\dfrac{x_{2A+2} - x}{x_{2A+2} - x_{2A}}\right)\left(\dfrac{x_{2A+1} - x}{x_{2A+1} - x_{2A}}\right) & \text{for } x_{2A} \leq x \leq x_{2A+2} \\ 0 & \text{otherwise} \end{cases}$$

for $A = 0, \ldots, n_{el}$ and:

$$N_{2A-1}(x) = \begin{cases} \left(\dfrac{x - x_{2A-2}}{x_{2A-1} - x_{2A-2}}\right)\left(\dfrac{x_{2A} - x}{x_{2A} - x_{2A-1}}\right) & \text{for } x_{2A-2} \leq x \leq x_{2A} \\ 0 & \text{otherwise} \end{cases}$$

for $A = 1, \ldots, n_{el}$. Visually:



And so on!

The above are referred to as piecewise quadratic finite element basis functions or, more specifically, Lagrange piecewise quadratic finite element basis functions. There are other flavors of piecewise quadratic finite element basis functions including Bernstein and hierarchical quadratic finite element basis functions. Lagrange quadratic finite element basis functions are unique in that they interpolate both element boundary nodes and element interior nodes.

Note every quadratic finite element test function $v^h \in \mathcal{V}^h$ takes the form:

$$v^h(x) = \cancelto{0}{v^h(0)} N_0(x) + \sum_{A=1}^{2n_{el}-1} v^h(x) N_A(x) + \cancelto{0}{v^h(L)} N_{2n_{el}}(x)$$

so $\{N_A\}_{A=1}^{2n_{el}-1}$ is a basis for $\mathcal{V}^h$. Likewise, every quadratic trial solution $v^h \in \mathcal{S}^h$ takes the form:

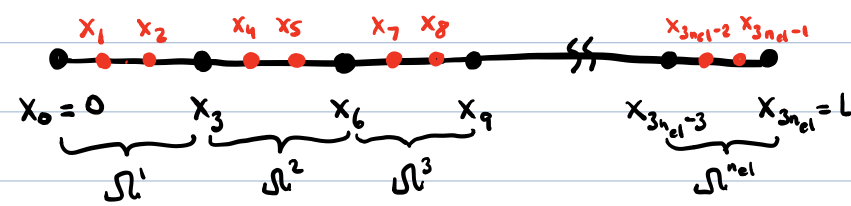
$$v^h(x) = \underbrace{v^h(0)}_{\nearrow g_0} N_0(x) + \sum_{A=1}^{2n_{el}-1} v^h(x) N_A(x) + \underbrace{v^h(L)}_{\nearrow g_L} N_{2n_{el}}(x)$$

$$= \underbrace{\left(g_0 N_0(x) + g_L N_{2n_{el}}(x) \right)}_{\in \mathcal{S}} + \underbrace{\sum_{A=1}^{2n_{el}-1} v^h(x) N_A(x)}_{\in \mathcal{V}}$$

so we can select $g^h(x) = g_0 N_0(x) + g_L N_{2n_{el}}(x)$. As already mentioned, Lagrange quadratic finite element basis functions are interpolatory just like linear finite element basis functions. They also have local support and form a partition of unity just like linear finite element basis functions. However, they are not pointwise non-negative, unlike linear finite element basis functions.

A Bubnov-Galerkin method employing piecewise quadratic finite element basis functions is, unsurprisingly, commonly referred to as a piecewise quadratic finite element method. Piecewise quadratic finite element methods are more complicated than piecewise linear finite element methods, but they yield a higher rate of convergence to sufficiently smooth weak solutions, a fact that will be discussed in the next lecture.

We finish by identifying finite element basis functions of degree $k=3$. As in the quadratic setting, we add nodes to each element, but now we add two nodes per element rather than one before re-ordering nodes:



We then again construct an interpolatory basis associated with nodal degrees of freedom. This basis takes the form:

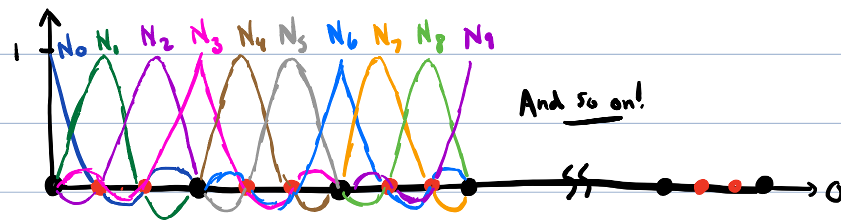
$$N_{3A}(x) = \begin{cases} \left(\dfrac{x - x_{3A-3}}{x_{3A} - x_{3A-3}}\right)\left(\dfrac{x - x_{3A-2}}{x_{3A} - x_{3A-2}}\right)\left(\dfrac{x - x_{3A-1}}{x_{3A} - x_{3A-1}}\right) & \text{for } x_{3A-3} \leq x \leq x_{3A} \\ \left(\dfrac{x_{3A+3} - x}{x_{3A+3} - x_{3A}}\right)\left(\dfrac{x_{3A+2} - x}{x_{3A+2} - x_{3A}}\right)\left(\dfrac{x_{3A+1} - x}{x_{3A+1} - x_{3A}}\right) & \text{for } x_{3A} \leq x \leq x_{3A+3} \\ 0 & \text{otherwise} \end{cases}$$

for $A = 0, \dots, n_{el}$ and:

$$N_{3A-1}(x) = \begin{cases} \left(\dfrac{x - x_{3A-3}}{x_{3A-1} - x_{3A-3}}\right)\left(\dfrac{x - x_{3A-2}}{x_{3A-1} - x_{3A-2}}\right)\left(\dfrac{x_{3A} - x}{x_{3A} - x_{3A-1}}\right) & \text{for } x_{3A-3} \leq x \leq x_{3A} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{3A-2}(x) = \begin{cases} \left(\dfrac{x - x_{3A-3}}{x_{3A-2} - x_{3A-3}}\right)\left(\dfrac{x_{3A-1} - x}{x_{3A-1} - x_{3A-2}}\right)\left(\dfrac{x_{3A} - x}{x_{3A} - x_{3A-2}}\right) & \text{for } x_{3A-3} \leq x \leq x_{3A} \\ 0 & \text{otherwise} \end{cases}$$

for $A = 1, \ldots, N_{el}$. Visually:



The above are referred to as Lagrange piecewise cubic finite element basis functions. Their definition is impacted by the position of the nodes in the element interiors. It is common to equispace the points, but it is more stable to place them at so-called Gauss-Lobatto integration points. A Bubnov-Galerkin method employing such basis functions is called a piecewise cubic finite element method, and piecewise cubic finite element methods return even higher convergence rates than piecewise quadratic finite element methods.

While in this lecture, we have discussed how to construct piecewise linear, quadratic, and cubic finite element methods, we have not discussed how to efficiently implement such methods on a computer. Such an implementation should exploit the local or element definition of finite element basis functions. We will discuss such an implementation shortly. First, however, we will discuss the accuracy of Galerkin finite element solutions using Céa's lemma, a powerful theoretical result from functional analysis.