Computing Global System Matrices

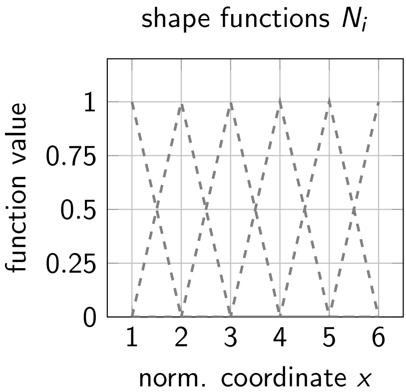
- two approaches:
  - global approach (integrate over whole domain)
  - local/element approach (sum element contributions)

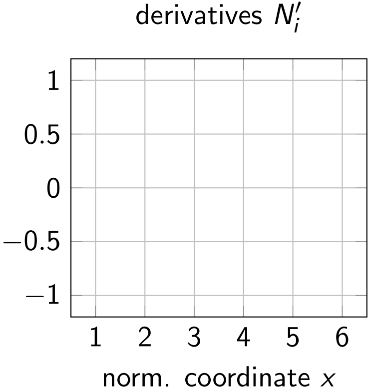
Example: compute the global stiffness matrix K

- entries of the stiffness matrix:

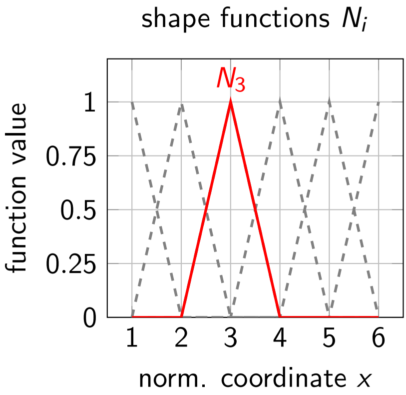
$$K_{ij} = \int_a^b N_i' N_j' \mathrm{d}x$$

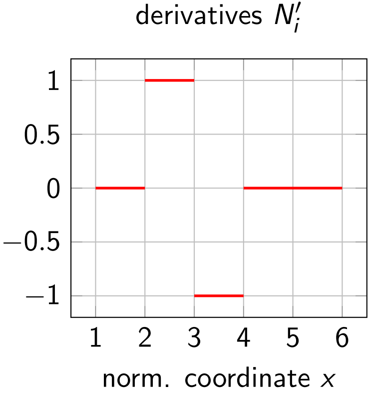
Diagonal Entry: $K_{33} = \int_a^b N_3' N_3' \mathrm{d}x$



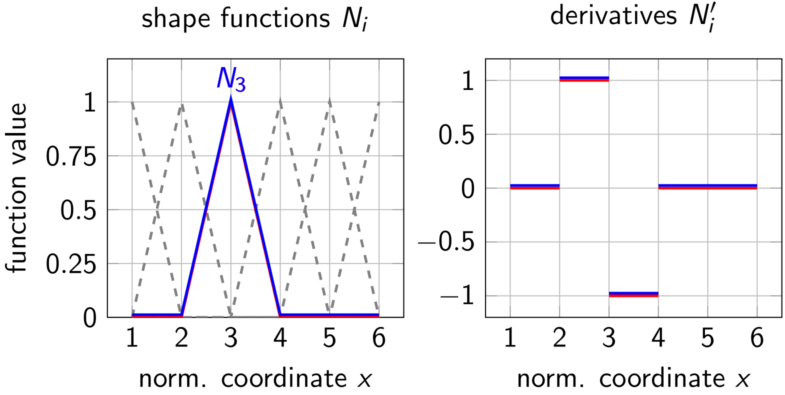


Diagonal Entry: $K_{33} = \int_a^b N_3' N_3' \mathrm{d}x$

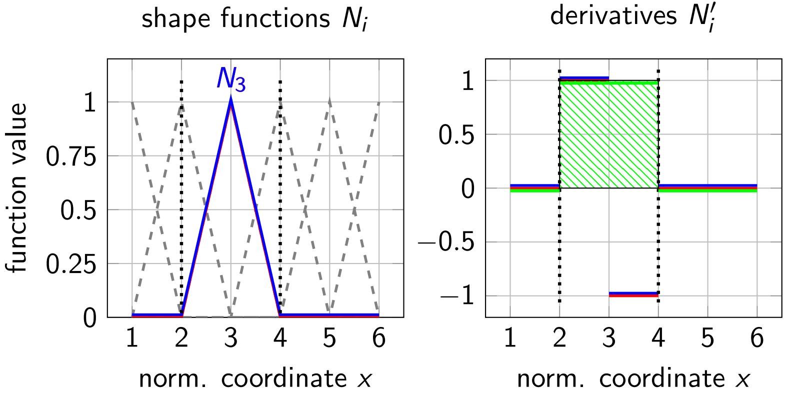




Diagonal Entry: $K_{33} = \int_a^b N_3' N_3' \mathrm{d}x$

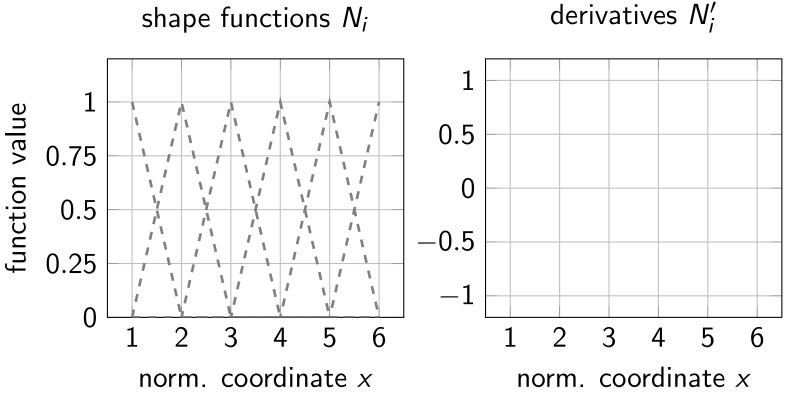


Diagonal Entry: $K_{33} = \int_a^b N_3' N_3' \mathrm{d}x$

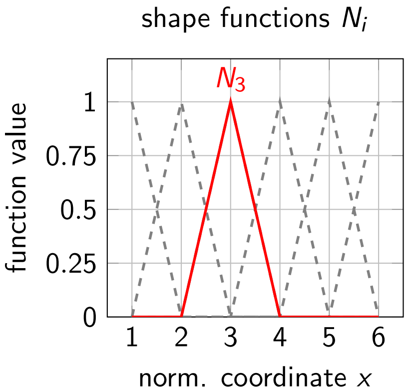


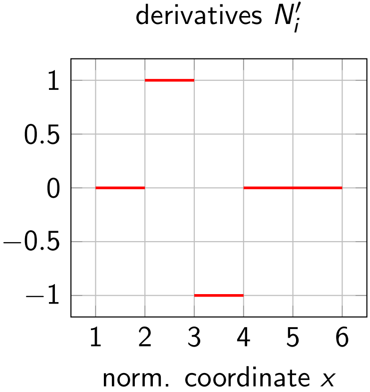
- elements 2 and 3 are involved
- $K_{33} = 2$

Off-Diagonal Entry: $K_{34} = \int_a^b N_3' N_4' \mathrm{d}x$

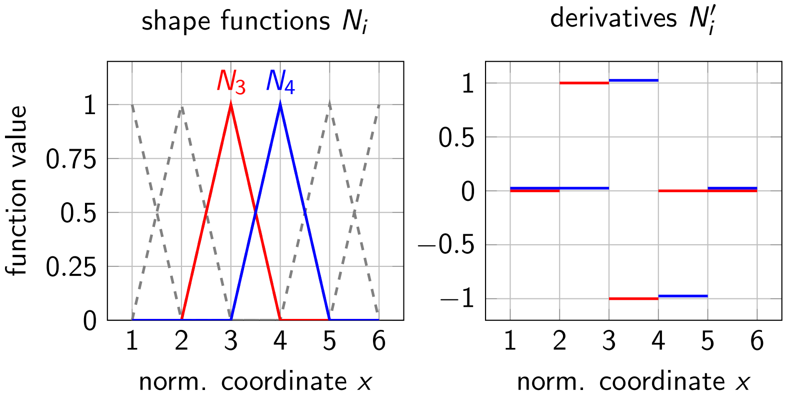


Off-Diagonal Entry: $K_{34} = \int_a^b N_3' N_4' \mathrm{d}x$

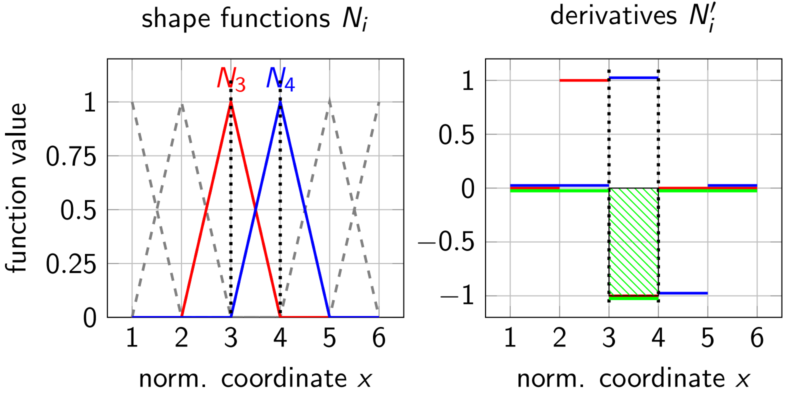




Off-Diagonal Entry: $K_{34} = \int_a^b N_3' N_4' \mathrm{d}x$



Off-Diagonal Entry: $K_{34} = \int_a^b N_3' N_4' \mathrm{d}x$



- element 3 is involved
- $K_{34} = -1$

Stiffness Matrix: $K_{ij} = \int_a^b N_i' N_j' \mathrm{d}x$

- in general:
  - $K_{ij} \neq 0$ only where N_i and N_j **overlap**
  - leads to **sparse** matrices

$$
\begin{bmatrix}
\bullet & \bullet & & & & \\
\bullet & \bullet & \bullet & & & \\
& \bullet & \bullet & \bullet & & \\
& & \bullet & \bullet & \bullet & \\
& & & \bullet & \bullet & \bullet \\
& & & & \bullet & \bullet
\end{bmatrix}
$$

Element Stiffness Matrix

- issues with global approach:
  - N_i' is not continuous at the element boundaries
  - we have to differentiate and integrate **piecewise**
  - N_i needs to be defined per element
    - complicated for 2D and 3D
    - preferably use *reference element* (later exercise class)

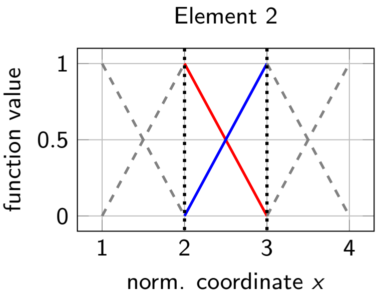
Element Stiffness Matrix

- issues with global approach:
  - N_i' is not continuous at the element boundaries
  - we have to differentiate and integrate **piecewise**
  - N_i needs to be defined per element
    - complicated for 2D and 3D
    - preferably use *reference element* (later exercise class)
- solution: compute entries element-wise
  - split the integral into **element contributions**

$$K_{ij} = \int_{\Omega} N_i' N_j' \mathrm{d}x = \sum_e \underbrace{\int_{\Omega_e} N_i' N_j' \mathrm{d}x}_{k_{ij}^e}$$

  - k_{ij}^e is calculated only on each element domain Ω_e
  - N_i, N_j are continuously differentiable functions on Ω_e
  - mapping between element and *reference element* possible

Reduced Element Stiffness Matrix



- element domain $\Omega_e = [x_l, x_r]$:
  - $N_1^e(x) = (x_r - x)/h$
  - $N_2^e(x) = (x - x_l)/h$
- where $h = x_r - x_l$ (element size)

- for each element e, k_{ij}^e has only 4 non-zero entries.
  - **reduce** to a 2×2–matrix with elements

$$k_{ab}^e = \int_{x_l}^{x_r} N_a^{e\prime} N_b^{e\prime} \mathrm{d}x$$

  - assemble: add k_{ab}^e to the correct K_{ij}

Assembling Procedure

1. initialize $\mathrm{K} = 0$ of size $N \times N$
2. for each element e
   2.1 compute: $\mathrm{k}^\mathrm{e} = [k^e_{ab}]$ of size 2×2
   2.2 map: $(a, b) \to (i, j)$
   2.3 add: $K_{ij} = K_{ij} + k^e_{ab}$

- the map $(a, b) \to (i, j)$ is given by `mesh.nodesOfElem`

Assembling Procedure

1. initialize $\mathrm{K} = 0$ of size $N \times N$
2. for each element e
   2.1 compute: $\mathrm{k}^e = [k^e_{ab}]$ of size 2×2
   2.2 map: $(a, b) \to (i, j)$
   2.3 add: $K_{ij} = K_{ij} + k^e_{ab}$

- the map $(a, b) \to (i, j)$ is given by `mesh.nodesOfElem`

Example: assemble element 2 into global 4×4–matrix

- get indices of global matrix: `ij = mesh.nodesOfElem(2,:)`
  - yields `ij = [2, 3]` (elem. 2 is connected to node 2 and 3)
- add $K_{ij} = K_{ij} + k^2_{ab}$ (superscript 2: second element) as:

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} \\ K_{21} & \mathrm{K}_{22} + \mathrm{k}^2_{11} & \mathrm{K}_{23} + \mathrm{k}^2_{12} & K_{24} \\ K_{31} & \mathrm{K}_{32} + \mathrm{k}^2_{21} & \mathrm{K}_{33} + \mathrm{k}^2_{22} & K_{34} \\ K_{41} & K_{42} & K_{43} & K_{44} \end{bmatrix}$$