The Finite Element Method in Continuum Mechanics

T.W.J. de Geus

*Contact: goosefem.geus.me – tom@geus.me – www.geus.me

Abstract

The theory of the Finite Element Method in Continuum Mechanics is discussed in a compact way. This discussion is by no means comprehensive and one is invited to dive in more complete textbooks, but also to get one's hands dirty by implementing some (simple) examples.

Contents

1 Statics

1.1 The basic concept

The discussion starts by considering a static, solid mechanics, problem. Loosely speaking the goal is to find a deformation map, $\vec{x} = \varphi(\vec{X}, t)$, that maps a body Ω_0 to a deformed state Ω that satisfies equilibrium and the boundary conditions applied on Γ. This is illustrated in Fig. 1, whereby it is emphasized that the body can be subjected to two kinds of boundary conditions:

- *Essential* or *Dirichlet* boundary conditions on Γ_p, whereby the displacements are prescribed.

- *Natural* or *Neumann* boundary conditions on Γ_u, whereby the tractions are prescribed. Note that traction-free natural boundary conditions are also perfectly acceptable.
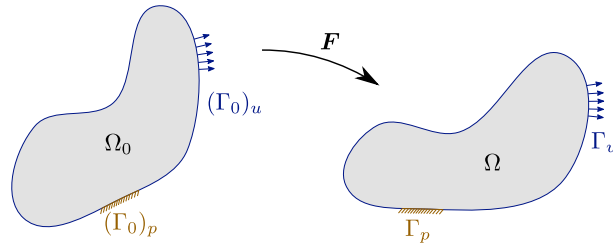
Figure 1. Generic problem definition

In practice, one is not looking for the map $\vec{x} = \varphi(\vec{X}, t)$, but for the deformation gradient $\boldsymbol{F}$, or in fact for a displacement field $\vec{u} = \vec{x} - \vec{X}$. To make things a bit more explicit, the deformation gradient is defined as follows:

$$\vec{x} = \boldsymbol{F} \cdot \vec{X} \tag{1}$$

hence

$$\boldsymbol{F} = \frac{\partial \varphi}{\partial \vec{X}} = \left(\vec{\nabla}_0 \, \vec{x} \right)^T = \boldsymbol{I} + \left(\vec{\nabla}_0 \, \vec{u} \right)^T \tag{2}$$

1.2   Momentum balance

The momentum balance reads

$$\vec{\nabla} \cdot \boldsymbol{\sigma}(\vec{x}) = \vec{0} \qquad \vec{x} \in \Omega \tag{3}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress which depends on the new position $\vec{x}$ and thus on the displacement $\vec{u}$. It has been assumed that all actions are instantaneous (no inertia) and, for simplicity, that there are no body forces. Loosely speaking the interpretation of this equation is that *the sum of all forces vanishes* everywhere in the domain Ω.

The crux of the Finite Element Method is that this non-linear differential equation is solved in a weak sense. I.e.

$$\int_{\Omega} \vec{\phi}(\vec{x}) \cdot \left[\vec{\nabla} \cdot \boldsymbol{\sigma}(\vec{x}) \right] \, \mathrm{d}\Omega = 0 \qquad \forall \, \vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{4}$$

where $\vec{\phi}$ are test functions. For reasons that become obvious below, integration by parts is applied, which results in

$$\int_{\Omega} \left[\vec{\nabla}\vec{\phi}(\vec{x}) \right] : \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega = \int_{\Omega} \vec{\nabla} \cdot \left[\vec{\phi}(\vec{x}) \cdot \boldsymbol{\sigma}(\vec{x}) \right] \, \mathrm{d}\Omega \qquad \forall \, \vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{5}$$

In going from Eq. (4) to Eq. (5), use has been made of the following chain rule

$$\vec{\nabla} \cdot \left[\vec{\phi} \cdot \boldsymbol{\sigma}^T \right] = \left[\vec{\nabla}\vec{\phi} \right] : \boldsymbol{\sigma}^T + \vec{\phi} \cdot \left[\vec{\nabla} \cdot \boldsymbol{\sigma} \right] \tag{6}$$

together with the symmetry of the Cauchy stress

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \tag{7}$$

The right-hand side of this equation can be reduced to an area integral by employing Gauss's divergence theorem:

$$\int_{\Omega} \vec{\nabla} \cdot \vec{a}(\vec{x}) \, \mathrm{d}\Omega = \int_{\Gamma} \vec{n}(\vec{x}) \cdot \vec{a}(\vec{x}) \, \mathrm{d}\Gamma \tag{8}$$

where $\vec{n}$ is the normal along the surface Γ. The result reads

$$\int_{\Omega} \left[\vec{\nabla}\vec{\phi}(\vec{x}) \right] : \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega = \int_{\Gamma} \vec{\phi}(\vec{x}) \cdot \underbrace{\vec{n}(\vec{x}) \cdot \boldsymbol{\sigma}(\vec{x})}_{\vec{t}(\vec{x})} \, \mathrm{d}\Gamma \qquad \forall \, \vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{9}$$

1.3 Discretization

The problem is now discretized using n *nodes* that are connected through *elements*, which define the discretized domain Ω_0^h. Shape functions $N_i(\vec{x})$ are used to extrapolate the nodal quantities throughout the domain Ω_0^h (and Ω^h). For example the displacement field $\vec{u}(\vec{x})$

$$\vec{u}(\vec{x}, t) \approx \vec{u}^h(\vec{x}, t) = \sum_{m=1}^{n_{\text{nodes}}} N_m(\vec{x}) \, \vec{u}_m(t) \tag{10}$$

whereby we will not further consider the time dependence (denoted with t) for now. Following standard Galerkin the test functions are interpolated in the same way, i.e.

$$\vec{\phi}(\vec{x}) \approx \vec{\phi}^h(\vec{x}) = \sum_{m=1}^{n_{\text{nodes}}} N_m(\vec{x}) \, \vec{\phi}_m \tag{11}$$

Applied to the problem sketch from Fig. 1, a discretization might look like Fig. 2. The nodes are clearly marked as circles. The lines connecting the nodes clearly mark the elements which are in this case three-node linear triangles.
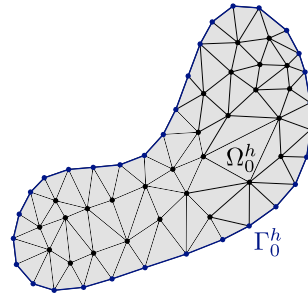
Figure 2. Example of a discretization of the problem in Fig. 1 using Finite Elements (linear triangles in this case).

Applied to the balance equation the following is obtained

$$\vec{\phi}_m \cdot \int_{\Omega^h} \left[\vec{\nabla} N_m(\vec{x}) \right] \cdot \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega = \vec{\phi}_m \cdot \int_{\Gamma^h} N_m(\vec{x}) \cdot \vec{t}(\vec{x}) \, \mathrm{d}\Gamma \qquad \forall \, \vec{\phi}_m \in \mathbb{R}_n^d \tag{12}$$

from which the dependency on $\vec{\phi}_m$ can be dropped:

$$\int_{\Omega^h} \left[\vec{\nabla} N_m(\vec{x}) \right] \cdot \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x}) \cdot \vec{t}(\vec{x}) \, \mathrm{d}\Gamma \tag{13}$$

This corresponds to a (non-linear) set of nodal balance equations:

$$\vec{f}_m^{\text{int}}(\vec{x}) = \vec{f}_m^{\text{ext}}(\vec{x}) \tag{14}$$

with:

- *Internal forces*

$$\vec{f}_m^{\text{int}}(\vec{x}) = \int_{\Omega^h} \left[\vec{\nabla} N_m(\vec{x}) \right] \cdot \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega \tag{15}$$

- *External forces*

$$\vec{f}_m^{\text{ext}}(\vec{x}) = \int_{\Gamma^h} N_m(\vec{x}) \cdot \vec{t}(\vec{x}) \, \mathrm{d}\Gamma \tag{16}$$

Note that this term is zero in the interior of the domain, i.e. in $\Omega^h \bigcap \Gamma^h$, while it can be zero or non-zero in Γ^h depending on the problem details. Most often it following completely from the boundary conditions of the boundary value problem that we set out to solve.

1.4 Iterative solution – small strain

A commonly used strategy to solve the non-linear system is the iterative Newton-Raphson scheme (see Appendix D). The idea is thereby to formulate an initial guess for the solution, determine possible residual forces, and use these forces to come to a better guess for the solution. This is continued until the solution has been found, i.e. when the residual vanishes.

This solution technique is discussed here in the context of small deformations. Assuming the deformations (and therefore rotations) to be small leads to the assumption that $\Omega = \Omega_0$, and thus that $\nabla = \nabla_0$. In this context one typically uses the linear strain tensor

$$\varepsilon = \tfrac{1}{2} \left[\nabla_0 \vec{u} + \left[\nabla_0 \vec{u} \right]^T \right] = \mathbb{I}_s : \left[\nabla_0 \vec{u} \right] \tag{17}$$

and some (non-linear) relationship between it and the stress as follows

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\varepsilon) \tag{18}$$

To further simplify the discussion, the boundary tractions are assumed to be some displacement independent quantity, which is *a-priori* known for all relevant boundary nodes.

The nodal equilibrium equations now reads

$$\vec{r}_m(\vec{X}, \vec{u}) = \vec{f}_m^{\text{ext}}(\vec{X}) - \vec{f}_m^{\text{int}}(\vec{X}, \vec{u}) = \vec{\underline{0}} \tag{19}$$

with

$$\vec{f}_m^{\text{int}}(\vec{X}, \vec{u}) = \int\limits_{\Omega_0^h} \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \cdot \boldsymbol{\sigma}(\vec{X}, \vec{u}) \, \mathrm{d}\Omega_0 \tag{20}$$

The displacement field $\vec{u}$ is now iteratively updated starting from some initial guess, i.e.

$$\vec{u}_{(i+1)} = \vec{u}_{(i)} + \delta\vec{u} \tag{21}$$

The update, $\delta\vec{u}$, follows from the linearization of Eq. (19). The results of which is

$$\int\limits_{\Omega_0^h} \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \cdot \mathbb{K}(\vec{X}, \vec{u}_{(i)}) \cdot \left[\vec{\nabla}_0 N_n(\vec{X}) \right] \, \mathrm{d}\Omega_0 \cdot \delta\vec{u}_n = \vec{f}_m^{\text{ext}}(\vec{X}) - \int\limits_{\Omega_0^h} \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \cdot \boldsymbol{\sigma}(\vec{X}, \vec{u}_{(i)}) \, \mathrm{d}\Omega_0 \tag{22}$$

where, at each position $\vec{X}$,

$$\mathbb{K}(\vec{u}_{(i)}) = \left. \frac{\partial\boldsymbol{\sigma}}{\partial\varepsilon} \right|_{\vec{u}_{(i)}} : \mathbb{I}_s \tag{23}$$

is the *constitutive tangent operator*.

In a shorter notation, the iterative update reads:

$$\boldsymbol{K}_{mn,(i)} \cdot \delta\vec{u}_m = \vec{f}_m^{\text{ext}} - \vec{f}_{m,(i)}^{\text{int}} \tag{24}$$

with

$$\boldsymbol{K}_{mn,(i)} = \int\limits_{\Omega_0^h} \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \cdot \mathbb{K}(\vec{X}, \vec{u}_{(i)}) \cdot \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \, \mathrm{d}\Omega_0 \tag{25}$$

and

$$\vec{f}_{m,(i)}^{\text{int}} = \int\limits_{\Omega_0^h} \left[\vec{\nabla}_0 N_m(\vec{X}) \right] \cdot \boldsymbol{\sigma}(\vec{X}, \vec{u}_{(i)}) \, \mathrm{d}\Omega_0 \tag{26}$$

2 Dynamics

2.1 Momentum balance

The next step is to add inertia to the balance of momentum in Eq. (3):

$$\rho\,\vec{a} = \vec{\nabla} \cdot \boldsymbol{\sigma}(\vec{x}) \qquad \vec{x} \in \Omega \tag{27}$$

where ρ is the mass density, and second time derivative of the position $\vec{x}$ is the acceleration $\vec{a} = \ddot{\vec{u}} = \ddot{\vec{x}}$. Note that this function is the continuum equivalent of $\vec{f} = m\vec{a}$.

Like before, this equation is solved in a weak sense

$$\int_{\Omega} \rho(\vec{x})\,\vec{\phi}(\vec{x}) \cdot \vec{a}\,\mathrm{d}\Omega = \int_{\Omega} \vec{\phi}(\vec{x}) \cdot \left[\vec{\nabla} \cdot \boldsymbol{\sigma}(\vec{x}) \right] \mathrm{d}\Omega \qquad \forall\,\vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{28}$$

Integration by parts results in

$$\int_{\Omega} \rho(\vec{x})\,\vec{\phi}(\vec{x}) \cdot \vec{a}\,\mathrm{d}\Omega = \int_{\Gamma} \vec{\phi}(\vec{x}) \cdot \vec{t}(\vec{x})\,\mathrm{d}\Gamma - \int_{\Omega} \left[\vec{\nabla}\vec{\phi}(\vec{x}) \right] : \boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega \qquad \forall\,\vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{29}$$

which is discretized as before:

$$\vec{\phi}_m \cdot \int_{\Omega} \rho(\vec{x})\,N_m(\vec{x})\,N_n(\vec{x})\,\mathrm{d}\Omega\,\vec{a}_n = \vec{\phi}_m \cdot \int_{\Gamma} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma - \vec{\phi}_m \cdot \int_{\Omega} \left[\vec{\nabla}N_m(\vec{x}) \right] : \boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega \qquad \forall\,\vec{\phi}_m \in \mathbb{R}^d_n \tag{30}$$

This is independent of the test functions, hence:

$$\underbrace{\int_{\Omega} \rho(\vec{x})\,N_m(\vec{x})\,N_n(\vec{x})\,\mathrm{d}\Omega}_{M_{mn}(\vec{x})}\,\vec{a}_n = \underbrace{\int_{\Gamma} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma}_{\vec{f}^{\mathrm{ext}}_m(\vec{x})} - \underbrace{\int_{\Omega} \left[\vec{\nabla}N_m(\vec{x}) \right] : \boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega}_{\vec{f}^{\mathrm{int}}_m(\vec{x})} \tag{31}$$

This is denoted as follows

$$M_{mn}(\vec{x})\,\vec{a}_n = \vec{f}^{\mathrm{ext}}_m(\vec{x}) - \vec{f}^{\mathrm{int}}_m(\vec{x}) \tag{32}$$

Where $M_{mn}(\vec{x})$ is the *mass matrix*, $\vec{f}^{\mathrm{ext}}_m(\vec{x})$ are the *external forces*, and $\vec{f}^{\mathrm{int}}_m(\vec{x})$ are the *internal forces*.

2.2 Time discretization

To solve the second order differential equation in time, one typically also discretizes time with some finite difference based scheme. For this we also need to distinguish the velocity field $\vec{v} = \dot{\vec{u}} = \dot{\vec{x}}$.

2.2.1 Verlet

In the case that the forces are velocity independent one can use the Velocity Verlet scheme, in which energy is conserved, i.e. it is reversible in time. The scheme reads are follows:

1. Compute the velocity on a dummy time grid:

$$\vec{v}_{n+1/2} = \vec{v}_{n-1/2} + \Delta_t\,\vec{a}_n \tag{33}$$

   Note that this involves solving Eq. (31) for $\vec{a}_n$. From this it is directly obvious why the forces need to be velocity independent.

2. Update the positions

$$\vec{u}_{n+1} = \vec{u}_n + \Delta_t\,\vec{av}_{n+1/2} \tag{34}$$

2.2.2 Velocity Verlet

In the case that the forces are dependent on the velocity (i.e. when there is damping) the previous scheme cannot be used anymore. An additional approximation has to be made in this case. Note that this is not very problematic as energy is in no case conserved. The resulting scheme reads:

1. Compute the position at $t_{n+1} = t_n + \Delta_t$:

$$\vec{u}_{n+1} = \vec{u}_n + \Delta_t \vec{v}_n + \tfrac{1}{2}\Delta_t^2 \vec{a}_n \tag{35}$$

2. Estimate the velocity at $t_{n+1} = t_n + \Delta_t$ (involves solving Eq. (31)):

$$\hat{\vec{v}}_{n+1} = \vec{v}_n + \tfrac{1}{2}\Delta_t \left[\vec{a}_n + \vec{a}(\vec{u}_{n+1}, \vec{v}_n + \Delta_t \vec{a}_n, t_{n+1}) \right] \tag{36}$$

3. Correct $\hat{\vec{v}}_{n+1}$ (involves solving Eq. (31)):

$$\vec{v}_{n+1} = \vec{v}_n + \tfrac{1}{2}\Delta_t \left[\vec{a}_n + \vec{a}(\vec{u}_{n+1}, \hat{\vec{v}}_{n+1}, t_{n+1}) \right] \tag{37}$$

2.3 Approximations

For problems where the local volume is conversed (either weakly through slightly compressible elasticity, or strongly by adding an incompressibility constraint) it makes sense to assume the mass matrix constant, as any change of volume results in an equivalent change of the density. In that case

$$\int_{\Omega} \rho(\vec{x}) \, \mathrm{d}\Omega = \int_{\Omega_0} \rho(\vec{X}) \, \mathrm{d}\Omega_0 \tag{38}$$

Which results in:

$$M_{mn}(\vec{x}) = \int_{\Omega_0} \rho(\vec{X}) \, N_m(\vec{X}) \, N_n(\vec{X}) \, \mathrm{d}\Omega_0 = \text{constant} \tag{39}$$

To enhance computational efficiency, it may be a good option concentrate the mass to point masses on the nodes. This has to strong advantage that the mass matrix becomes diagonal, known as the *lumped mass matrix*. Consequently, instead of solving a linear system one just has to solve fully independent equations.

A Nomenclature

- Vector

$$\vec{u} = \sum_i u_i \vec{e}_i \tag{40}$$

  for example for Cartesian coordinates

$$\vec{u} = u_x \vec{e}_x + u_y \vec{e}_y + u_z \vec{e}_z \tag{41}$$

- Second-order tensor

$$\boldsymbol{A} = \sum_i \sum_j A_{ij} \vec{e}_i \vec{e}_j \tag{42}$$

- Fourth-order tensor

$$\mathbb{A} = \sum_i \sum_j \sum_k \sum_l A_{ijkl} \vec{e}_i \vec{e}_j \vec{e}_k \vec{e}_l \tag{43}$$

- Divergence

$$\vec{\nabla} \cdot \boldsymbol{\sigma} = \frac{\partial \sigma_{ij}}{\partial x_i} \tag{44}$$

- Double tensor contraction

$$C = \boldsymbol{A} : \boldsymbol{B} = A_{ij} B_{ji} \tag{45}$$

B Shape functions

In the Finite Element Method a geometry is discretized using nodes. The nodes are grouped in elements which define the domain Ω^h. The crux of the method is that nodal quantities, for example $\vec{u}_i$, are interpolated throughout the discretized domain Ω^h using shape functions $N_i(\vec{x})$. The shape functions are polynomials that spanned by the nodes, such that they constitute a partition of unity and satisfy $N_i(\vec{x}_j) = \delta_{ij}$, i.e. it is one in the node i, and zero in all other nodes. This implies that, while each shape function is globally supported, $N_i(\vec{x}) \neq 0$ only in the elements containing node i. For all other elements it is zero (even in between the nodes). This again implies that $\partial N_i / \partial \vec{x}$ is discontinuous across element boundaries.

For a one-dimensional problem comprising four linear elements and five nodes the shape functions are sketched in Fig. 3(a). To emphasize the global support of each shape function the shape function of node 2 is isolated in Fig. 3(b). As can be seen, node 2 is only non-zero in elements 1 and 2, while it is zero everywhere else. To evaluate for example f_2, the internal force one node 2, the integration can be limited to these two elements:

$$f_2 = \int\limits_{\Omega^{(1)}} \frac{\mathrm{d}N_2^{(1)}}{\mathrm{d}x}\, \sigma(x)\, \mathrm{d}\Omega^{(1)} + \int\limits_{\Omega^{(2)}} \frac{\mathrm{d}N_2^{(2)}}{\mathrm{d}x}\, \sigma(x)\, \mathrm{d}\Omega^{(2)} \tag{46}$$

(a) All shape functions $N_i(x)$, each with a different color.

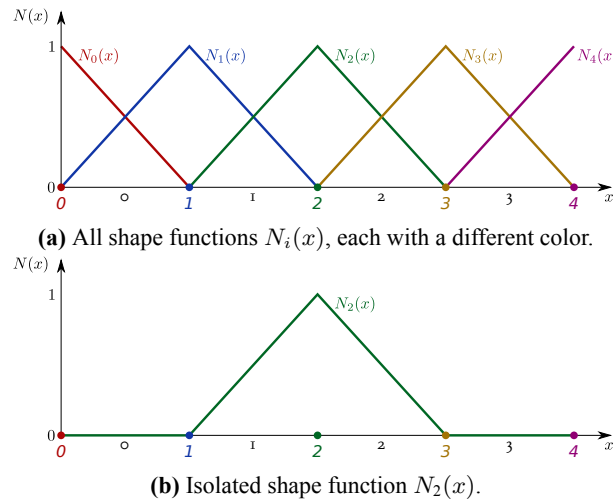**(b)** Isolated shape function $N_2(x)$.

Figure 3. (a) Shape functions for a one-dimensional domain discretized using four two node, linear, element. (b) Isolated shape functions for node 2. The node numbers are displayed using a unique color, while the element numbers are shown in black in a different font, in between the nodes.

By now it should be clear that the above allows us assemble $\underline{f}$ element-by-element. For the example in Fig. 3 this is graphically shown in Fig. 4.

where the indices show that the *shape functions* are evaluated compared to some generic element definition.

C Iso-parametric transformation and quadrature

A very important concept in the Finite Element Method is the iso-parametric transformation. It allows an arbitrarily shaped element with volume Ω^e to be mapped onto a generic *iso-parametric element* of constant volume Q, see Fig. 5. Using this mapping it is easy to perform interpolation and numerical quadrature using a generic implementation.

The mapping between the generic domain Q and the physical domain Ω^e is as follows

$$\vec{x}(\vec{\xi}) = N_m^e\, \underline{x}_m^e \tag{47}$$

where the column $\underline{x}^e$ contains the position vectors of the element nodes in the actual coordinates (i.e. spanning Ω^e). In order to perform the quadrature on Q we must also map the gradient operator:

$$\vec{\nabla}_\xi = \vec{e}_i \frac{\partial}{\partial \xi_i} = \vec{e}_i \frac{\partial x_j(\vec{\xi})}{\partial \xi_i} \frac{\partial}{\partial x_j} = \vec{e}_i \frac{\partial x_j(\vec{\xi})}{\partial \xi_i} \vec{e}_j \cdot \vec{e}_k \frac{\partial}{\partial x_k} = \left[\, \vec{\nabla}_\xi\, \vec{x}(\vec{\xi})\, \right] \cdot \vec{\nabla} = \boldsymbol{J}(\vec{\xi}) \cdot \vec{\nabla} \tag{48}$$
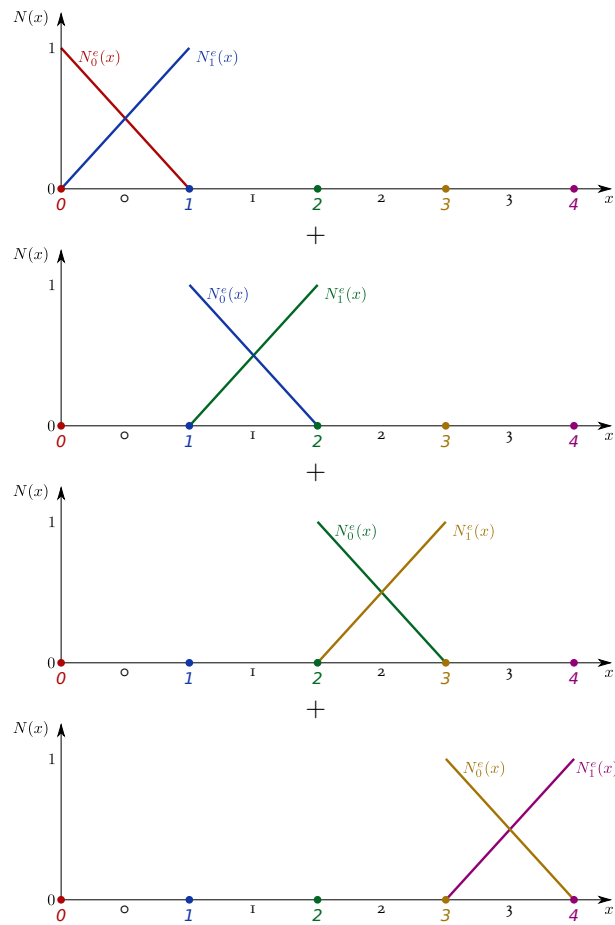
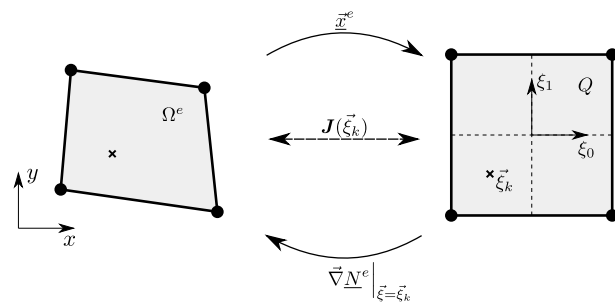**Figure 4.** Element-by-element assembly of the problem in Fig. 3.

Figure 5. Iso-parameteric transformation for a four node, bi-linear, quadrilateral element.

i.e.

$$J_{ij} = \frac{\partial x_i}{\partial \xi_j} \tag{49}$$

from which the inverse relationship trivially follows

$$\vec{\nabla} = \boldsymbol{J}^{-1}(\vec{\xi}) \cdot \vec{\nabla}_\xi \tag{50}$$

where $\boldsymbol{J}$ is the Jacobian which can be obtained from the gradient of the shape functions with respect to the iso-parametric coordinates and the nodal coordinates in the actual coordinates through Eq. (47):

$$\boldsymbol{J}(\vec{\xi}) = \vec{\nabla}_\xi \, \vec{x}(\vec{\xi}) = \vec{\nabla}_\xi \left[N_m^e \, \vec{x}_m^e \right] = \left[\vec{\nabla}_\xi N_m^e \right] \vec{x}_m^e \tag{51}$$

The gradient of the shape functions with respect to the actual coordinates can now be computed though

$$\vec{\nabla} N_m^e = \boldsymbol{J}^{-1}(\vec{\xi}) \cdot \left[\vec{\nabla}_\xi N_m^e \right] \tag{52}$$

The change of variables should also be accounted for in the integration:

$$\int_\Omega (\ldots) \, \mathrm{d}\Omega = \iiint (\ldots) \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z = \iiint (\ldots) \left| \frac{\partial (x, y, z)}{\partial (\xi, \eta, \mu)} \right| \mathrm{d}\xi \, \mathrm{d}\eta \, \mathrm{d}\mu \tag{53}$$

where the determinant of the Jacobian

$$\frac{\partial (x, y, z)}{\partial (\xi, \eta, \mu)} = \begin{vmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial x}{\partial \eta} & \dfrac{\partial x}{\partial \mu} \\ \dfrac{\partial y}{\partial \xi} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial y}{\partial \mu} \\ \dfrac{\partial z}{\partial \xi} & \dfrac{\partial z}{\partial \eta} & \dfrac{\partial z}{\partial \mu} \end{vmatrix} \tag{54}$$

We can now make use of the Jacobian that we already obtained

$$\int_\Omega (\ldots) \, \mathrm{d}\Omega = \iiint (\ldots) \det\left(\boldsymbol{J}(\vec{\xi})\right) \, \mathrm{d}\xi \, \mathrm{d}\eta \, \mathrm{d}\mu = \int_Q (\ldots) \det\left(\boldsymbol{J}(\vec{\xi})\right) \, \mathrm{d}Q \tag{55}$$

Note that another way to obtain this result is using

$$\mathrm{d}\Omega = \mathrm{d}\vec{x}_0 \times \mathrm{d}\vec{x}_1 \cdot \mathrm{d}\vec{x}_2 = \left[\mathrm{d}\vec{x}_0 \cdot \boldsymbol{J}(\vec{\xi}) \right] \times \left[\mathrm{d}\vec{x}_1 \cdot \boldsymbol{J}(\vec{\xi}) \right] \cdot \left[\mathrm{d}\vec{x}_2 \cdot \boldsymbol{J}(\vec{\xi}) \right] = \det\left(\boldsymbol{J}(\vec{\xi})\right) \mathrm{d}\vec{\xi}_0 \times \mathrm{d}\vec{\xi}_1 \cdot \mathrm{d}\vec{\xi}_2 \tag{56}$$

$$= \det\left(\boldsymbol{J}(\vec{\xi})\right) \mathrm{d}Q \tag{57}$$

Numerical quadrature can be formulated (exactly) on the master element. It corresponds to taking the weighted sum of the integrand evaluated at specific *quadrature points* (or *integration points*).

The practical implications, for example for the internal force, are as follows:

$$\vec{f}_m^e = \int_{\Omega^e} \left[\vec{\nabla} N_m \right] \cdot \boldsymbol{\sigma}(\vec{x}) \, \mathrm{d}\Omega = \int_Q \left[\vec{\nabla} N_m \right] \cdot \boldsymbol{\sigma}(\vec{x}) \, \det\left(\boldsymbol{J}(\vec{\xi})\right) \mathrm{d}Q \tag{58}$$

$$= \sum_k^{n_k} w_k \left[\vec{\nabla} N_m \right]_{\vec{\xi} = \vec{\xi}_k} \cdot \boldsymbol{\sigma}\left(\vec{x}(\vec{\xi}_k)\right) \, \det\left(\boldsymbol{J}(\vec{\xi}_k)\right) \tag{59}$$

Total Lagrange

To obtain

$$\vec{x}(\vec{\xi}), \, \vec{\nabla}_0, \text{ and } \int_{\Omega_0} . \, \mathrm{d}\Omega \tag{60}$$

simply replace $\vec{x}_m^e$ with $\vec{X}_m^e$ in Eq. (47). For this reason the same element implementation can be used in small strain and finite strain (total Lagrange and updated Lagrange), by providing either $\vec{X}_m^e$ or $\vec{X}_m^e + \vec{u}_m^e$ as input.

D   Newton-Raphson in one dimension

The objective is to find x such that

$$r(x) = 0 \tag{61}$$

An initial guess is made for x which is (hopefully) iteratively improved. This iterative value is denoted using $x_{(i)}$. It is updated by making use of the following Taylor expansion

$$r\big(x_{(i+1)}\big) = r\big(x_{(i)}\big) + \left.\frac{\mathrm{d}r}{\mathrm{d}x}\right|_{x=x_{(i)}} \delta x + \mathcal{O}\big(\delta x^2\big) \approx 0 \tag{62}$$

where

$$\delta x = x_{(i+1)} - x_{(i)} \tag{63}$$

The value of δx can be obtained by neglecting higher order terms:

$$r\big(x_{(i)}\big) + \left.\frac{\mathrm{d}r}{\mathrm{d}x}\right|_{x=x_{(i)}} \delta x = 0 \tag{64}$$

From which it follows that

$$\delta x = - \left[\left.\frac{\mathrm{d}r}{\mathrm{d}x}\right|_{x=x_{(i)}}\right]^{-1} r\big(x_{(i)}\big) \tag{65}$$

Thereafter, the improved 'guess' of the solution is

$$x_{(i+1)} = x_{(i)} + \delta x \tag{66}$$

These steps are then repeated, until the solution has been reached within a certain accuracy ϵ:

$$\big|r\big(x_{(i+1)}\big)\big| < \epsilon \tag{67}$$

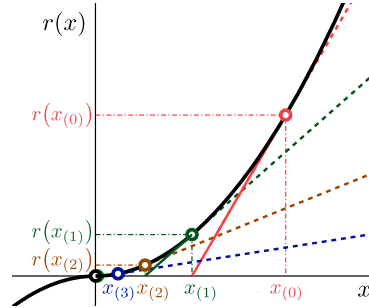The iterative scheme is well understood from the illustration in Fig. 6.

Figure 6. Schematic of the Newton-Raphson iterations in a one-dimensional problem.

E   B-matrix

E.1   Discretization

(9)

$$\int_{\Omega} \big[\vec{\nabla}\vec{\phi}(\vec{x})\big] : \boldsymbol{\sigma}(\vec{x})\, \mathrm{d}\Omega = \int_{\Gamma} \vec{\phi}(\vec{x}) \cdot \vec{t}(\vec{x})\, \mathrm{d}\Gamma \qquad \forall\, \vec{\phi}(\vec{x}) \in \mathbb{R}^d \tag{68}$$

The discretization from Eq. (11) can be written more precisely as

$$\phi_x(\vec{x})\vec{e}_x + \phi_y(\vec{x})\vec{e}_y + \phi_z(\vec{x})\vec{e}_z \approx \sum_{m=1}^{n_{\text{nodes}}} N_m(\vec{x})\,\phi_{m,x}\vec{e}_x + N_m(\vec{x})\,\phi_{m,y}\vec{e}_y + N_m(\vec{x})\,\phi_{m,z}\vec{e}_z \tag{69}$$

$$\vec{\nabla}\vec{\phi}(\vec{x}) \approx \sum_{m=1}^{n_{\text{nodes}}} \left(\vec{e}_x\frac{\partial}{\partial x} + \vec{e}_y\frac{\partial}{\partial y} + \vec{e}_z\frac{\partial}{\partial z}\right)\left(N_m(\vec{x})\,\phi_{m,x}\vec{e}_x + N_m(\vec{x})\,\phi_{m,y}\vec{e}_y + N_m(\vec{x})\,\phi_{m,z}\vec{e}_z\right) \tag{70}$$

$$= \phi_{m,x}\frac{\partial N_m}{\partial x}\vec{e}_x\vec{e}_x + \phi_{m,y}\frac{\partial N_m}{\partial x}\vec{e}_x\vec{e}_y + \phi_{m,z}\frac{\partial N_m}{\partial x}\vec{e}_x\vec{e}_z \tag{71}$$

$$+ \phi_{m,x}\frac{\partial N_m}{\partial y}\vec{e}_y\vec{e}_x + \phi_{m,y}\frac{\partial N_m}{\partial y}\vec{e}_y\vec{e}_y + \phi_{m,z}\frac{\partial N_m}{\partial y}\vec{e}_y\vec{e}_z \tag{72}$$

$$+ \phi_{m,x}\frac{\partial N_m}{\partial z}\vec{e}_z\vec{e}_x + \phi_{m,y}\frac{\partial N_m}{\partial z}\vec{e}_z\vec{e}_y + \phi_{m,z}\frac{\partial N_m}{\partial z}\vec{e}_z\vec{e}_z \tag{73}$$

Which we can denote in short as

$$\vec{\nabla}\vec{\phi}(\vec{x}) \approx \sum_{m=1}^{n_{\text{nodes}}} \mathcal{B}_m(\vec{x})\cdot\vec{\phi}_m \tag{74}$$

$$\mathcal{B}_{m,xxx} = \frac{\partial N_m}{\partial x} \qquad \mathcal{B}_{m,yxx} = \frac{\partial N_m}{\partial y} \qquad \mathcal{B}_{m,zxx} = \frac{\partial N_m}{\partial z}$$

$$\mathcal{B}_{m,xyy} = \frac{\partial N_m}{\partial x} \qquad \mathcal{B}_{m,yyy} = \frac{\partial N_m}{\partial y} \qquad \mathcal{B}_{m,zyy} = \frac{\partial N_m}{\partial z} \tag{75}$$

$$\mathcal{B}_{m,xzz} = \frac{\partial N_m}{\partial x} \qquad \mathcal{B}_{m,yzz} = \frac{\partial N_m}{\partial y} \qquad \mathcal{B}_{m,zzz} = \frac{\partial N_m}{\partial z} \tag{76}$$

Note that similarly

$$\vec{\nabla}\vec{u}(\vec{x}) \approx \sum_{m=1}^{n_{\text{nodes}}} \mathcal{B}_m(\vec{x})\cdot\vec{u}_m \tag{77}$$

$$\int_{\Omega^h} \left(\mathcal{B}_m\cdot\vec{\phi}_m\right):\boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x})\,\vec{\phi}_m\cdot\vec{t}(\vec{x})\,\mathrm{d}\Gamma \qquad \forall\,\vec{\phi}_m \in \mathbb{R}_n^d \tag{78}$$

$$\vec{\phi}_m\cdot\int_{\Omega^h} \mathcal{B}_m^T:\boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega = \vec{\phi}_m\cdot\int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \qquad \forall\,\vec{\phi}_m \in \mathbb{R}_n^d \tag{79}$$

$$\int_{\Omega^h} \mathcal{B}_m^T:\boldsymbol{\sigma}(\vec{x})\,\mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \tag{80}$$

E.2 Linear elasticity

$$\int_{\Omega_0^h} \tilde{\mathcal{B}}_m^T:\left(\mathbb{C}:\boldsymbol{\varepsilon}(\vec{x})\right)\mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \tag{81}$$

$$\int_{\Omega_0^h} \tilde{\mathcal{B}}_m^T:\left(\mathbb{C}:\vec{\nabla}\vec{u}(\vec{x})\right)\mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \tag{82}$$

$$\int_{\Omega_0^h} \tilde{\mathcal{B}}_m^T:\left(\mathbb{C}:(\tilde{\mathcal{B}}_n\cdot\vec{u}_n)\right)\mathrm{d}\Omega = \int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \tag{83}$$

$$\int_{\Omega_0^h} \tilde{\mathcal{B}}_m^T:\left(\mathbb{C}:\tilde{\mathcal{B}}_n\right)\mathrm{d}\Omega\cdot\vec{u}_n = \int_{\Gamma^h} N_m(\vec{x})\,\vec{t}(\vec{x})\,\mathrm{d}\Gamma \tag{84}$$
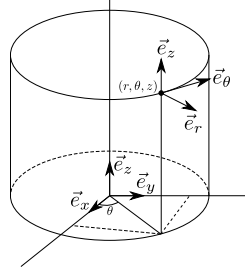
F   Cylindrical coordinates

F.1   ??

Figure 7. Caption here

$$x = r \cos\theta \qquad y = r \sin\theta \qquad z = z \tag{85}$$

$$r = \sqrt{x^2 + y^2} \qquad \theta = \arctan\left(y/x\right) \qquad z = z \tag{86}$$

$$\begin{aligned}
\vec{e}_r &= \cos\theta\ \vec{e}_x + \sin\theta\ \vec{e}_y \\
\vec{e}_\theta &= -\sin\theta\ \vec{e}_x + \cos\theta\ \vec{e}_y \\
\vec{e}_r &= \vec{e}_z
\end{aligned} \tag{87}$$

$$\vec{\nabla} = \vec{e}_r \frac{\partial}{\partial r} + \frac{1}{r}\vec{e}_\theta \frac{\partial}{\partial \theta} + \vec{e}_z \frac{\partial}{\partial z} \tag{88}$$

$$\begin{aligned}
\frac{\partial \vec{e}_r}{\partial r} = 0 \qquad & \frac{\partial \vec{e}_\theta}{\partial r} = 0 \qquad && \frac{\partial \vec{e}_z}{\partial r} = 0 \\
\frac{\partial \vec{e}_r}{\partial \theta} = \vec{e}_\theta \qquad & \frac{\partial \vec{e}_\theta}{\partial \theta} = -\vec{e}_r \qquad && \frac{\partial \vec{e}_z}{\partial \theta} = 0 \\
\frac{\partial \vec{e}_r}{\partial z} = 0 \qquad & \frac{\partial \vec{e}_\theta}{\partial z} = 0 \qquad && \frac{\partial \vec{e}_z}{\partial z} = 0
\end{aligned} \tag{89}$$

We begin by making Eq. (11) more specific:

$$\phi_r(\vec{x})\vec{e}_r + \phi_\theta(\vec{x})\vec{e}_\theta + \phi_z(\vec{x})\vec{e}_z \approx \sum_{i=1}^{n} N_i(\vec{x})\phi_{i,r}\vec{e}_r + N_i(\vec{x})\phi_{i,\theta}\vec{e}_\theta + N_i(\vec{x})\phi_{i,z}\vec{e}_z \tag{90}$$

We can also evaluate the discretized gradient

$$\vec{\nabla}\vec{\phi}(\vec{x}) \approx \sum_{i=1}^{n} \left(\vec{e}_r \frac{\partial}{\partial r} + \frac{1}{r}\vec{e}_\theta \frac{\partial}{\partial \theta} + \vec{e}_z \frac{\partial}{\partial z} \right) \left(N_i(\vec{x})\phi_{i,r}\vec{e}_r + N_i(\vec{x})\phi_{i,\theta}\vec{e}_\theta + N_i(\vec{x})\phi_{i,z}\vec{e}_z \right) \tag{91}$$

$$\begin{aligned}
= \sum_{i=1}^{n} &\phi_{i,r} \frac{\partial N_i}{\partial r}\vec{e}_r\vec{e}_r + \phi_{i,\theta} \frac{\partial N_i}{\partial r}\vec{e}_r\vec{e}_\theta + \phi_{i,z} \frac{\partial N_i}{\partial r}\vec{e}_r\vec{e}_z \\
&+ \phi_{i,r} \frac{1}{r}\frac{\partial N_i}{\partial \theta}\vec{e}_\theta\vec{e}_r + \phi_{i,r}\frac{1}{r}N_i(\vec{x})\vec{e}_\theta\vec{e}_\theta + \phi_{i,\theta}\frac{1}{r}\frac{\partial N_i}{\partial \theta}\vec{e}_\theta\vec{e}_\theta - \phi_{i,\theta}\frac{1}{r}N_i(\vec{x})\vec{e}_r\vec{e}_\theta + \phi_{i,z}\frac{1}{r}\frac{\partial N_i}{\partial \theta}\vec{e}_\theta\vec{e}_z \\
&+ \phi_{i,r} \frac{\partial N_i}{\partial z}\vec{e}_z\vec{e}_r + \phi_{i,\theta} \frac{\partial N_i}{\partial z}\vec{e}_z\vec{e}_\theta + \phi_{i,z} \frac{\partial N_i}{\partial z}\vec{e}_z\vec{e}_z
\end{aligned} \tag{92}$$

Which we can denote in short as

$$\vec{\nabla}\vec{\phi}(\vec{x}) \approx \sum_{i=1}^{n} \mathcal{B}_i(\vec{x}) \cdot \vec{\phi}_i \tag{93}$$

where the non-zero components of the third order $\mathcal{B}_i$ read

$$
\begin{aligned}
&\mathcal{B}_{i,rrr} = \frac{\partial N_i}{\partial r} &&\mathcal{B}_{i,\theta rr} = \frac{1}{r}\frac{\partial N_i}{\partial \theta} &&\mathcal{B}_{i,zrr} = \frac{\partial N_i}{\partial z} \\
&\mathcal{B}_{i,r\theta\theta} = -\frac{1}{r}N_i(\vec{x}) &&\mathcal{B}_{i,\theta\theta r} = \frac{1}{r}N_i(\vec{x}) \\
&\mathcal{B}_{i,r\theta\theta} = \frac{\partial N_i}{\partial r} &&\mathcal{B}_{i,\theta\theta\theta} = \frac{1}{r}\frac{\partial N_i}{\partial \theta} &&\mathcal{B}_{i,z\theta\theta} = \frac{\partial N_i}{\partial z}
\end{aligned} \tag{94}
$$

$$
\begin{aligned}
&\mathcal{B}_{i,rzz} = \frac{\partial N_i}{\partial r} &&\mathcal{B}_{i,\theta zz} = \frac{1}{r}\frac{\partial N_i}{\partial \theta} &&\mathcal{B}_{i,zzz} = \frac{\partial N_i}{\partial z}
\end{aligned} \tag{95}
$$

$$\vec{\nabla}_\xi = \boldsymbol{J} \cdot \vec{\nabla} \tag{96}$$

$$\vec{e}_\xi \frac{\partial}{\partial \xi} + \vec{e}_\eta \frac{\partial}{\partial \eta} + \vec{e}_\mu \frac{\partial}{\partial \mu} = \boldsymbol{J} \cdot \left(\vec{e}_r \frac{\partial}{\partial r} + \frac{1}{r}\vec{e}_\theta \frac{\partial}{\partial \theta} + \vec{e}_z \frac{\partial}{\partial z} \right) \tag{97}$$

$$
\begin{aligned}
\boldsymbol{J} = &\ \vec{e}_\xi \frac{\partial r}{\partial \xi}\vec{e}_r + \vec{e}_\xi r \frac{\partial \theta}{\partial \xi}\vec{e}_\theta + \vec{e}_\xi \frac{\partial z}{\partial \xi}\vec{e}_z \\
&+ \vec{e}_\eta \frac{\partial r}{\partial \eta}\vec{e}_r + \vec{e}_\eta r \frac{\partial \theta}{\partial \eta}\vec{e}_\theta + \vec{e}_\eta \frac{\partial z}{\partial \eta}\vec{e}_z \\
&+ \vec{e}_\mu \frac{\partial r}{\partial \mu}\vec{e}_r + \vec{e}_\mu r \frac{\partial \theta}{\partial \mu}\vec{e}_\theta + \vec{e}_\mu \frac{\partial z}{\partial \mu}\vec{e}_z
\end{aligned} \tag{98}
$$

F.2 Axisymmetric