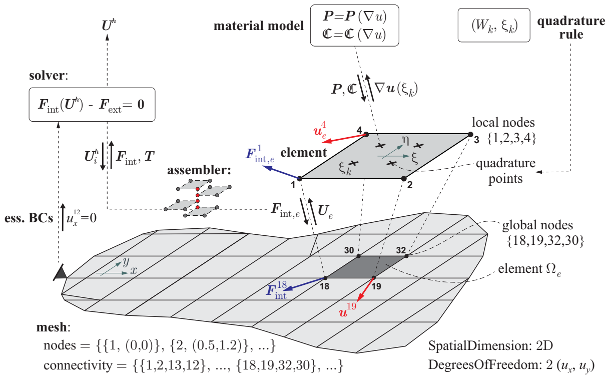
Computational Solid Mechanics

(Fall 2017)

Dennis M. Kochmann

Mechanics & Materials
Department of Mechanical and Process Engineering
ETH Zürich

course website:
www.mm.ethz.ch/teaching.html



These lecture notes are a *concise* collection of equations and comments. They are by no means a complete textbook or set of class notes that could replace lectures.

Therefore, you are strongly encouraged to take your own notes during lectures and to use this set of notes rather for reference.

Contents

1 Introduction: Continuum Mechanics and Notation

1.1 An introductory example: heat conduction

We describe a body $\Omega \subset \mathbb{R}^d$ with boundary $\partial\Omega$ as a collection of **material points**. Each point has a position $\boldsymbol{x}$ in a Cartesian coordinate system $(x_1, \dots, x_d)$ in d dimensions with origin $\boldsymbol{O}$.

Points are described by vectors defined by components in the Cartesian reference frame:

$$\boldsymbol{x} = \sum_{i=1}^{d} x_i \boldsymbol{g}_i = x_i \boldsymbol{g}_i. \tag{1.1}$$

Here and in the following we use Einstein's *summation convention* which implies summation over repeated indices. The usual index notation rules apply; e.g., the inner product is written as

$$\boldsymbol{a} \cdot \boldsymbol{b} = a_i \boldsymbol{g}_i \cdot b_j \boldsymbol{g}_j = a_i b_j \delta_{ij} = a_i b_i \qquad \text{with Kronecker's delta} \quad \delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{else.} \end{cases} \tag{1.2}$$

This is used to define the length of a vector as

$$\|\boldsymbol{a}\| = \sqrt{\boldsymbol{a} \cdot \boldsymbol{a}} = \sqrt{a_i a_i} = \sqrt{\sum_{i=1}^{d} a_i a_i}. \tag{1.3}$$

Matrix-vector multiplication becomes

$$\boldsymbol{a} = \boldsymbol{M}\boldsymbol{b} \qquad \Leftrightarrow \qquad a_i = M_{ij} b_j \qquad \text{and} \qquad (\boldsymbol{M}^{\mathrm{T}})_{ij} = M_{ji}. \tag{1.4}$$

We use **mappings** to denote fields. For example, the temperature field in a static problem is described by a mapping

$$T(\boldsymbol{x}) : \Omega \to \mathbb{R}, \tag{1.5}$$

which assigns to each point $\boldsymbol{x} \in \Omega$ a real number, the temperature. If the field is differentiable, one often introduces **kinematic variables** such as the temperature *gradient* field:

$$\boldsymbol{\beta} : \Omega \to \mathbb{R}^d \qquad \text{and} \qquad \boldsymbol{\beta} = \operatorname{grad} T = \nabla T \qquad \Leftrightarrow \qquad \beta_i = \frac{\partial T}{\partial x_i} = T_{,i}. \tag{1.6}$$

Here and in the following, we use *comma indices* to denote partial derivatives.

For every kinematic variable, there is conjugate field (often called **flux**) like the heat flux $\boldsymbol{q}$ in this thermal problem, which is also a mapping:

$$\boldsymbol{q} : \Omega \to \mathbb{R}^d. \tag{1.7}$$

The heat flux vector $\boldsymbol{q}$ assigns to each point in Ω a heat flux direction and magnitude. If we are interested, e.g., in the loss of heat through a point $\boldsymbol{x} \in \partial\Omega$ on the surface of Ω with outward unit normal $\boldsymbol{n}(\boldsymbol{x})$, then that amount of heat leaving Ω is the projection $\boldsymbol{q}(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x})$. Clearly, the components of $\boldsymbol{q} = (q_1, \dots, q_d)^{\mathrm{T}}$ imply the heat flux through surfaces perpendicular to each of the d Cartesian coordinate directions.

Next, **constitutive relations** link kinematic quantities to fluxes. For example, define the heat flux vector as $\boldsymbol{q} = \boldsymbol{q}(\boldsymbol{\beta})$. Fourier's law of heat conduction states, e.g.,

$$\boldsymbol{q} = -\boldsymbol{K}\boldsymbol{\beta} \qquad \Leftrightarrow \qquad q_i = -K_{ij}\beta_{,j}, \tag{1.8}$$

where $\boldsymbol{K}$ denotes a conductivity *tensor*. K_{ij} are the components of the conductivity tensor; they form a $d \times d$ matrix. Such a *second-order tensor* is a convenient way to store the conductivity properties in any arbitrary orientation in the form of a matrix, along with the coordinate basis in which it is defined. $\boldsymbol{K}$ provides for each direction of the temperature gradient $\boldsymbol{\beta} = \alpha \boldsymbol{n}$ the resulting normalized heat flux $\boldsymbol{q}$. To this end, one defines

$$\boldsymbol{K} = K_{ij} \boldsymbol{g}_i \otimes \boldsymbol{g}_j \qquad \Rightarrow \qquad \boldsymbol{K}\boldsymbol{n} = (K_{ij} \boldsymbol{g}_i \otimes \boldsymbol{g}_j)\boldsymbol{n} = K_{ij} \boldsymbol{g}_i (\boldsymbol{g}_j \cdot \boldsymbol{n}) = K_{ij} n_j \boldsymbol{g}_i. \tag{1.9}$$

Such a second-order tensor is hence a linear mapping of vectors onto vectors. Here, we defined the *dyadic product* (or *tensor product*), which produces a tensor according to

$$\boldsymbol{M} = \boldsymbol{a} \otimes \boldsymbol{b} \qquad \Leftrightarrow \qquad M_{ij} = a_i b_j. \tag{1.10}$$

A special tensor is the identity, which maps vectors onto themselves:

$$\boldsymbol{I} = \delta_{ij}\, \boldsymbol{g}_i \otimes \boldsymbol{g}_j. \tag{1.11}$$

To solve a thermal problem, we also need **balance laws**. Here, *conservation of energy* may be used, which states the change of **internal energy** E in a body over time t balances the inward and outward flux of energy and the energy being produced inside the body (by some heat source density ρs). Mathematically, this implies

$$\frac{\mathrm{d}}{\mathrm{d}t} E = \int_\Omega \rho s \,\mathrm{d}V - \int_{\partial\Omega} \boldsymbol{q} \cdot \boldsymbol{n} \,\mathrm{d}S. \tag{1.12}$$

We can use the **divergence theorem** to write

$$\int_{\partial\Omega} \boldsymbol{q} \cdot \boldsymbol{n} \,\mathrm{d}S = \int_{\partial\Omega} q_i n_i \,\mathrm{d}S = \int_\Omega q_{i,i} \,\mathrm{d}S = \int_\Omega \operatorname{div} \boldsymbol{q} \,\mathrm{d}S, \tag{1.13}$$

which defines the *divergence of a vector* as the scalar quantity

$$\operatorname{div}(\cdot) = (\cdot)_{i,i}. \tag{1.14}$$

Energy is an **extensive** variable, i.e., the total energy doubles when adding to bodies of the same energy. This is in contrast to **intensive** variables, such as temperature or pressure, when adding to bosides having the same, e.g., temperature. Since energy is an extensive variable, we can introduce an energy density e and write

$$E = \int_\Omega e \,\mathrm{d}V \qquad \Rightarrow \qquad \frac{\mathrm{d}}{\mathrm{d}t} E = \int_\Omega \dot{e} \,\mathrm{d}V. \tag{1.15}$$

Here and in the following, we use dots to denote *rates*, i.e., *time derivatives*. Note that we use a *Lagrangian* description and that, so far, there is no motion or deformation involved in our discussion, so the time derivative does not affect the volume integral. Rewriting the conservation of energy now yields

$$\int_\Omega \dot{e} \,\mathrm{d}V = \int_\Omega \rho s \,\mathrm{d}V - \int_\Omega \operatorname{div} \boldsymbol{q} \,\mathrm{d}V. \tag{1.16}$$

This can be rewritten as

$$\int_\Omega (\dot{e} - \rho s + \operatorname{div} \boldsymbol{q}) \,\mathrm{d}V = 0. \tag{1.17}$$

Since conservation of energy does not only have to hold for Ω but for any subbody $\omega \subset \Omega$, we may conclude that the **local energy balance** equation is

$$\boxed{\dot{e} = \rho s - \operatorname{div} \boldsymbol{q}} \tag{1.18}$$

This is the local (i.e., pointwise) counterpart to the macroscopic energy balance and states that at each point $\boldsymbol{x} \in \Omega$ the rate of energy change ($\dot{e}$) is given by the local production of heat (ρs) minus the heat lost by outward fluxes $\boldsymbol{q}$ away from the point.

Finally, exploiting that thermally stored energy gives $e = \rho c_v T$ (with constant mass density ρ and specific heat capacity c_v), and we insert Fourier's law of heat conduction to overall arrive at

$$\int_\Omega \rho c_v \dot{T}\,\mathrm{d}V = \int_\Omega \rho s\,\mathrm{d}V - \int_\Omega \operatorname{div}(-\boldsymbol{K}\boldsymbol{\beta})\,\mathrm{d}S = \int_\Omega \rho s\,\mathrm{d}V - \int_\Omega (-K_{ij}T_{,j})_{,i}\,\mathrm{d}S. \tag{1.19}$$

Note that the final term requires the use of the product rule since

$$(-K_{ij}T_{,j})_{,i} = -K_{ij,i}T_{,j} - K_{ij}T_{,ij}. \tag{1.20}$$

Let us assume a **homogeneous** body with $\boldsymbol{K}(\boldsymbol{x}) = \boldsymbol{K} = \text{const}$. Further, let us rewrite the above under a single integral:

$$\int_\Omega \left(\rho c_v \dot{T} - \rho s - K_{ij}T_{,ij}\right)\,\mathrm{d}V = 0. \tag{1.21}$$

Again, by extension of energy conservation to arbitrary subbodies, we conclude the local energy balance equation

$$\rho c_v \dot{T} = K_{ij}T_{,ij} + \rho s \tag{1.22}$$

This is the heat equation in its *anisotropic form.* In the special case of **isotropy** (i.e., the conductivity is the same in all directions), we obtain the *Laplacian* since

$$K_{ij} = \kappa\delta_{ij} \qquad \Rightarrow \qquad K_{ij}T_{,ij} = \kappa\delta_{ij}T_{,ij} = \kappa T_{,ii} = \kappa\,\Delta T \qquad \text{with} \qquad \Delta(\cdot) = (\cdot)_{,ii}, \tag{1.23}$$

and we arrive at the well-known **heat equation** with sources:

$$\boxed{\rho c_v \dot{T} = \kappa\,\Delta T + \rho s} \tag{1.24}$$

Whenever we consider a *static* problem, we assume that the body is in equilibrium and the temperature field is constant, which reduces the above to *Poisson's equation*, viz.

$$\kappa\,\Delta T = -\rho s. \tag{1.25}$$

1.2 A more advanced example: mechanical equilibrium

The mechanics of solids (and fluids) generally describes deformable bodies. To this end, we label each material point by its position $\boldsymbol{X}$ in a **reference configuration** (e.g., the configuration at time $t = 0$). The current position $\boldsymbol{x}$, by contrast, is a function of $\boldsymbol{X}$ and of time t: $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t)$. Fields in the reference and **current configuration** are generally referred to by upper- and lower-case characters (and the same applies to indices), e.g.,

$$\boldsymbol{x} = x_i \boldsymbol{g}_i, \qquad \boldsymbol{X} = X_I \boldsymbol{G}_I. \tag{1.26}$$

Note that, to avoid complication, we will work with Cartesian coordinate systems only (see the tensor notes, e.g., for curvilinear coordinates).

The mechanics of a deformable body undergoing finite deformations is generally described by a **deformation mapping**

$$\boldsymbol{\varphi}(\boldsymbol{X}, t) : \Omega \times \mathbb{R} \to \mathbb{R}^d \qquad \text{such that} \qquad \boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t). \tag{1.27}$$

Since it depends on time, we can take time derivatives to arrive at the **velocity** and **acceleration** fields, respectively:

$$\boldsymbol{V}(\boldsymbol{X}, t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(\boldsymbol{X}, t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\varphi}(\boldsymbol{X}, t), \qquad \boldsymbol{A}(\boldsymbol{X}, t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{V}(\boldsymbol{X}, t) = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{\varphi}(\boldsymbol{X}, t). \tag{1.28}$$

Note that those are **Lagrangian** fields (one could also write those as functions of the current position $\boldsymbol{x}$, which results in the **Eulerian** counterparts; this is usually done in fluid mechanics).

Like in the thermal problem, we introduce *kinematics* by defining the **deformation gradient**

$$\boldsymbol{F} = \operatorname{Grad}\boldsymbol{\varphi} \qquad \Leftrightarrow \qquad F_{iJ} = \frac{\partial \varphi_i}{\partial X_J} = \varphi_{i,J}. \tag{1.29}$$

Note that this is a second-order, *two-point tensor* defined across both configurations. If one uses the same coordinate frame for the undeformed and deformed configurations, one may alternative introduce the **displacement field**

$$\boldsymbol{u}(\boldsymbol{X}) = \boldsymbol{x} - \boldsymbol{X} \qquad \text{and} \qquad \boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}(\boldsymbol{X}), \tag{1.30}$$

so that

$$\boldsymbol{F} = \operatorname{Grad}\boldsymbol{\varphi} = \operatorname{Grad}(\boldsymbol{X} + \boldsymbol{u}) = \boldsymbol{I} + \operatorname{Grad}\boldsymbol{u} \qquad \Leftrightarrow \qquad F_{iJ} = \delta_{iJ} + u_{i,J}. \tag{1.31}$$

Note that in case of *no deformation*, we have $\boldsymbol{x} = \boldsymbol{X}$ so that $\boldsymbol{F} = \boldsymbol{I}$ (and *not* $\boldsymbol{F} = \boldsymbol{0}$).

$\boldsymbol{F}$ contains plenty of information about the local deformation. For example, the **volume change** at a point is given by

$$\frac{\mathrm{d}v}{\mathrm{d}V} = J = \det \boldsymbol{F}, \tag{1.32}$$

and for physical reasons we must have $J > 0$ (this ensures that the deformation mapping is injective; i.e., no two material points are mapped onto the same point in the current configuration). No volume change implies $J = 1$.

Similarly, the **stretch** in a direction defined by a unit vector $\boldsymbol{N}$ is determined by

$$\lambda(\boldsymbol{N}) = \frac{\mathrm{d}s}{\mathrm{d}S} = \sqrt{\boldsymbol{N} \cdot \boldsymbol{C}\boldsymbol{N}} \qquad \text{with} \qquad \boldsymbol{C} = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} \tag{1.33}$$

the **right Cauchy-Green tensor**. As for $\boldsymbol{F}$, an undeformed point has $\boldsymbol{C} = \boldsymbol{I}$.

Next, we need a constitutive law that links the deformation gradient to a "flux", which in mechanical problems we refer to as the **stress**. Heat flux may be defined as energy flow per oriented area,

$$q_i = \frac{\mathrm{d}Q}{\mathrm{d}A_i} \qquad \text{with} \qquad \mathrm{d}\boldsymbol{A} = \boldsymbol{N}\,\mathrm{d}A \quad \text{and} \quad \|\boldsymbol{N}\| = 1. \tag{1.34}$$

Analogously, stresses are force vector per oriented area:

$$P_{iJ} = \frac{\mathrm{d}F_i}{\mathrm{d}A_J}, \tag{1.35}$$

which defines the so-called **First Piola-Kirchhoff** (1st PK) **stress tensor**. This is a second-order tensor that captures the force components on any oriented area into all d coordinate directions. The resulting **traction** vector on a particular *infinitesimal* area with unit normal vector $\boldsymbol{N}$ is

$$\boldsymbol{T} = \boldsymbol{P}\boldsymbol{N} \qquad \Leftrightarrow \qquad T_i = P_{iJ}N_J. \tag{1.36}$$

Notice that the thus defined tractions satisfy Newton's law of action and reaction since

$$\boldsymbol{T}(\boldsymbol{N}) = \boldsymbol{P}\boldsymbol{N} = -\boldsymbol{P}(-\boldsymbol{N}) = -\boldsymbol{T}(-\boldsymbol{N}), \tag{1.37}$$

which we know well from inner forces in undergraduate mechanics. The total force acting on a surface A is hence

$$\boldsymbol{F}_{\text{tot}} = \int_A \boldsymbol{T}\,\mathrm{d}S. \tag{1.38}$$

For a mechanical problem, the relevant balance laws are *conservation of linear momentum* and of *anguluar momentum*. As before, one can formulate those as macroscopic balance laws. For example, macroscopic linear momentum balance is nothing but the well-known equation $\boldsymbol{F}_{\text{tot}} = M\boldsymbol{A}$ (*sum of all forces equals mass times mean acceleration*). To derive the local balance law of a continuous body, note that external forces include both surface tractions $\boldsymbol{T}$ and *body forces* $\rho_0\boldsymbol{B}$ – using ρ_0 to denote the reference mass density. Overall, we obtain

$$\int_{\partial\Omega} \boldsymbol{T}\,\mathrm{d}S + \int_\Omega \rho_0\boldsymbol{B}\,\mathrm{d}V = \int_\Omega \rho_0\boldsymbol{A}\,\mathrm{d}V \qquad \Leftrightarrow \qquad \int_{\partial\Omega} \boldsymbol{P}\boldsymbol{N}\,\mathrm{d}S + \int_\Omega \rho_0\boldsymbol{B}\,\mathrm{d}V = \int_\Omega \rho_0\boldsymbol{A}\,\mathrm{d}V. \tag{1.39}$$

Practicing the divergence theorem once more, we see that

$$\int_{\partial\Omega} P_{iJ}N_J\,\mathrm{d}S = \int_\Omega P_{iJ,J}\,\mathrm{d}V \qquad \Rightarrow \qquad (\operatorname{Div}\boldsymbol{P})_i = P_{iJ,J}, \tag{1.40}$$

which defines the *divergence of a second-order tensor*, which is a vector. Note that we use a *capitol operator* "Div" as opposed to "div" to indicate differentiation with respect to the undeformed coordinates.

When we again exploit that the above balance law must hold for all subbodies $\omega \subset \Omega$, we arrive at the *local statement of linear momentum balance*:

$$\boxed{\operatorname{Div}\boldsymbol{P} + \rho_0\boldsymbol{B} = \rho_0\boldsymbol{A}} \qquad \Leftrightarrow \qquad \boxed{P_{iJ,J} + \rho_0 B_i = \rho_0 A_i} \tag{1.41}$$

Note that the special case of **quasistatics** assumes that inertial effects are negligible, so one solves the quasistatic linear momentum balance $\operatorname{Div}\boldsymbol{P} + \rho_0\boldsymbol{B} = \boldsymbol{0}$. Except for gravity or electro/magnetomechanics, body forces also vanish in most cases, so that one simply arrives at $\operatorname{Div}\boldsymbol{P} = \boldsymbol{0}$.

It is important to recall that stresses in finite deformations are not unique but we generally have different types of stress tensors. Above, we introduced the **first Piola-Kirchhoff stress tensor $\boldsymbol{P}$**, which implies *actual force per undeformed area*. Similarly, one can define the **Cauchy stress tensor $\boldsymbol{\sigma}$**, which denotes *actual force per deformed area*. The definition and link are given by

$$\sigma_{ij} = \frac{\mathrm{d}F_i}{\mathrm{d}a_j}, \qquad \boldsymbol{\sigma} = \frac{1}{J}\boldsymbol{P}\boldsymbol{F}^{\mathrm{T}}. \tag{1.42}$$

Later on, it will be helpful to link stresses and deformation to energy. If the stored mechanical energy is characterized by the **strain energy density** $W = W(\boldsymbol{F})$, then one can show that

$$\boldsymbol{P} = \frac{\partial W}{\partial \boldsymbol{F}} \qquad \Leftrightarrow \qquad P_{iJ} = \frac{\partial W}{\partial F_{iJ}}. \tag{1.43}$$

Without knowing much about tensor analysis, we may interpret the above as a derivative of the energy density with respect to each component of $\boldsymbol{F}$, yielding the corresponding component of $\boldsymbol{P}$. This concept can also be extended to introduce a *fourth-order tensor*, the **incremental tangent modulus tensor**

$$\mathbb{C} = \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{F}} \qquad \Leftrightarrow \qquad \mathbb{C}_{iJkL} = \frac{\partial P_{iJ}}{\partial F_{kL}}, \tag{1.44}$$

for which each component of $\boldsymbol{P}$ is differentiated with respect to each component of $\boldsymbol{F}$. For further information on tensor analysis, see the tensor notes. Without further discussion, notice that a *stress-free* state implies that the energy attains an extremum.

Note that the dependence of W on $\boldsymbol{F}$ (and consequently the consitutive law between $\boldsymbol{P}$ and $\boldsymbol{F}$) is generally strongly nonlinear, which limits opportunities for closed-form analytical solutions. Also, for *material frame indifference* we must in fact have $W = W(\boldsymbol{C})$, but that is a technical detail of minor importance here.

Examples: Switching between symbolic and index notation is often convenient (especially when taking derivatives) and should be practiced. Consider the following examples:

- $$\boldsymbol{a} = \boldsymbol{T}\boldsymbol{b} \qquad \Leftrightarrow \qquad a_i = T_{ij}b_j \tag{1.45}$$
- $$\boldsymbol{a} = \boldsymbol{T}^{\mathrm{T}} \qquad \Leftrightarrow \qquad a_i = T_{ji}b_j \tag{1.46}$$
- $$\operatorname{tr}(\boldsymbol{a}\otimes\boldsymbol{b}) \qquad \Leftrightarrow \qquad \operatorname{tr}[a_ib_j] = a_ib_i = \boldsymbol{a}\cdot\boldsymbol{b} = b_ia_i = \operatorname{tr}(\boldsymbol{b}\otimes\boldsymbol{a}) \tag{1.47}$$
- $$\operatorname{tr}(\boldsymbol{R}^{\mathrm{T}}\boldsymbol{T}) = \operatorname{tr}[R_{ji}T_{jk}] = R_{ji}T_{ji} = \boldsymbol{R}\cdot\boldsymbol{T} \tag{1.48}$$
- $$\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{a}} = \left[\frac{\partial a_i}{\partial a_j}\right] = [\delta_{ij}] = \boldsymbol{I} \tag{1.49}$$
- $$\frac{\partial \operatorname{tr}\boldsymbol{T}}{\partial \boldsymbol{T}} = \frac{\partial T_{kk}}{\partial \boldsymbol{T}} = \left[\frac{\partial T_{kk}}{\partial T_{ij}}\right] = [\delta_{ki}\delta_{kj}] = [\delta_{ij}] = \boldsymbol{I} \tag{1.50}$$
- $$\frac{\partial\sqrt{\operatorname{tr}\boldsymbol{T}}}{\partial \boldsymbol{T}} = \frac{\partial\sqrt{\operatorname{tr}\boldsymbol{T}}}{\partial \operatorname{tr}\boldsymbol{T}}\frac{\partial \operatorname{tr}\boldsymbol{T}}{\partial \boldsymbol{T}} = \frac{\boldsymbol{I}}{2\sqrt{\operatorname{tr}\boldsymbol{T}}} \tag{1.51}$$
- $$\frac{\partial\lambda(\boldsymbol{N})}{\partial \boldsymbol{F}} = \frac{\partial\sqrt{\boldsymbol{N}\cdot\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\boldsymbol{N}}}{\partial \boldsymbol{F}} = \frac{1}{2\lambda(\boldsymbol{N})}\left[\frac{\partial N_KF_{jK}F_{jM}N_M}{\partial F_{iJ}}\right] \tag{1.52}$$
$$= \frac{1}{2\lambda(\boldsymbol{N})}\left[N_K\delta_{ij}\delta_{JK}F_{jM}N_M + N_KF_{jK}\delta_{ij}\delta_{JM}N_M\right] = \frac{[F_{iM}N_MN_J]}{\lambda(\boldsymbol{N})} \tag{1.53}$$
$$= \boldsymbol{F}\boldsymbol{M}\otimes\boldsymbol{N}/\lambda(\boldsymbol{N}) \tag{1.54}$$

Examples: Consider a special form of the compressible Neo-Hookean material model defined by

$$W(\boldsymbol{F}) = \frac{\mu}{2}(\operatorname{tr}\overline{\boldsymbol{C}} - 3) + \frac{\kappa}{2}(J-1)^2, \qquad \overline{\boldsymbol{C}} = \overline{\boldsymbol{F}}^{\mathrm{T}}\overline{\boldsymbol{F}} \qquad \text{and} \qquad \overline{\boldsymbol{F}} = \frac{\boldsymbol{F}}{J^{-1/3}}. \tag{1.55}$$

The first Piola-Kirchhoff stress tensor is computed as

$$\begin{aligned}
\boldsymbol{P} = \frac{\partial W}{\partial \boldsymbol{F}} &= \frac{\partial}{\partial \boldsymbol{F}}\left[\frac{\mu}{2}\left(\frac{\operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}}{J^{2/3}} - 3\right) + \frac{\kappa}{2}(J-1)^2\right], \qquad J = \det\boldsymbol{F} \\
&= \frac{\mu}{2}\left(\frac{1}{J^{2/3}}\frac{\partial \operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}}{\partial \boldsymbol{F}} + \operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\,\frac{\partial}{\partial \boldsymbol{F}}\frac{1}{J^{2/3}}\right) + 2\kappa(J-1)\frac{\partial J}{\partial \boldsymbol{F}} \\
&= \frac{\mu}{2}\left(\frac{2\boldsymbol{F}}{J^{2/3}} - \frac{2\operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}}{3J^{5/3}}J\boldsymbol{F}^{-T}\right) + \kappa(J-1)J\boldsymbol{F}^{-T} \\
&= \frac{\mu}{J^{2/3}}\boldsymbol{F} + \left[\kappa(J-1)J - \frac{\mu}{3J^{2/3}}\operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}\right]\boldsymbol{F}^{-T},
\end{aligned} \tag{1.56}$$

where we used

$$\frac{\partial J}{\partial \boldsymbol{F}} = \operatorname{cof}\boldsymbol{F} = J\,\boldsymbol{F}^{-\mathrm{T}} \qquad \text{and} \qquad \frac{\partial \operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}}{\partial \boldsymbol{F}} = 2\boldsymbol{F}. \tag{1.57}$$

Using index notation, the above identity is written as

$$P_{iJ} = \frac{\partial W}{\partial F_{iJ}} = \frac{\mu}{J^{2/3}}F_{iJ} + \left[\kappa(J-1)J - \frac{\mu}{3J^{2/3}}I_1\right]F^{-1}_{Ji}, \qquad I_1 = \operatorname{tr}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} = F_{kL}F_{kL}. \tag{1.58}$$

The incremental stiffness tensor requires taking second derivatives:

$$\begin{aligned}
\mathbb{C}_{iJkL} &= \frac{\partial P_{iJ}}{\partial F_{kL}} \\
&= \mu\left(F_{iJ}\frac{\partial}{\partial F_{kL}}\frac{1}{J^{2/3}} + \frac{1}{J^{2/3}}\frac{\partial F_{iJ}}{\partial F_{kL}}\right) + \kappa(2J-1)\frac{\partial J}{\partial F_{kL}}F^{-1}_{Ji} + \kappa(J-1)J\frac{\partial F^{-1}_{Ji}}{\partial F_{kL}} \\
&\quad - \frac{\mu}{3}\left[\frac{\partial J^{-2/3}}{\partial F_{kL}}I_1F^{-1}_{Ji} + \frac{1}{J^{2/3}}\frac{\partial I_1}{\partial F_{kL}}F^{-1}_{Ji} + \frac{I_1}{J^{2/3}}\frac{\partial F^{-1}_{Ji}}{\partial F_{kL}}\right] \\
&= \frac{\mu}{J^{2/3}}\delta_{ik}\delta_{JL} - \frac{2\mu}{3J^{2/3}}\left(F_{iJ}F^{-1}_{Lk} + F^{-1}_{Ji}F_{kL}\right) \\
&\quad + \left(\frac{2\mu}{9J^{2/3}}I_1 + \kappa(2J-1)J\right)F^{-1}_{Lk}F^{-1}_{Ji} + \left(\frac{\mu}{3J^{2/3}}I_1 - \kappa(J-1)J\right)F^{-1}_{Jk}F^{1}_{Li},
\end{aligned} \tag{1.59}$$

where we rearranged the last equation by grouping terms and simplifying, and we used

$$\frac{\partial F^{-1}_{Ji}}{\partial F_{kL}} = -F^{-1}_{Jk}F^{1}_{Li}. \tag{1.60}$$

To check the final answers, we verify that each side of the equation has the exact same free indices appearing only once.

1.3 A special case: linearized kinematics

Whenever only small deformation is expected, the above framework can be significantly simplified by using **linearized kinematics**. To this end, we assume that $\|\mathrm{Grad}\,\boldsymbol{u}\| \ll 1$ ("*small strains*"). Note that in this case it does not make a significant difference if we differentiate with respect to x_i or X_I, so that one generally uses only lower-case indices for simplicity.

In small strains, the *displacement field* is the key field to be determined (rather than the deformation mapping), i.e., we seek $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$.

Recall that

$$\boldsymbol{C} = \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}} = (\boldsymbol{I} + \mathrm{Grad}\,\boldsymbol{u})(\boldsymbol{I} + \mathrm{Grad}\,\boldsymbol{u}^{\mathrm{T}}) = \boldsymbol{I} + \mathrm{Grad}\,\boldsymbol{u} + (\mathrm{Grad}\,\boldsymbol{u})^{\mathrm{T}} + (\mathrm{Grad}\,\boldsymbol{u})(\mathrm{Grad}\,\boldsymbol{u})^{\mathrm{T}}. \quad (1.61)$$

Now, the final term is dropped by a scaling argument ($\|\mathrm{Grad}\,\boldsymbol{u}\| \ll 1$). Therefore, we may introduce a kinematic relation like in the thermal problem:

$$\boldsymbol{\beta} = \mathrm{Grad}\,\boldsymbol{u}, \quad (1.62)$$

and all important local deformation information is encoded in $\boldsymbol{\beta}$. Like a temperature gradient causes heat flux, a displacement gradient causes stresses (if displacements are constant everywhere, the body is undergoing **rigid body translation** and does not produce any stresses).

To make sure we also do not pick up **rigid body rotation**, one introduces the infinitesimal **strain tensor**

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left[\mathrm{Grad}\,\boldsymbol{u} + (\mathrm{Grad}\,\boldsymbol{u})^{\mathrm{T}}\right] = \frac{1}{2}\left(\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^{\mathrm{T}}\right) \qquad \Leftrightarrow \qquad \varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right). \quad (1.63)$$

Notice that, unlike in finite deformations, *no deformation* implies $\boldsymbol{\varepsilon} = \boldsymbol{0}$. Furthermore, by definition $\boldsymbol{\varepsilon}$ is symmetric since $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^{\mathrm{T}}$ (not like $\boldsymbol{F}$ which is asymmetric). The same applies for $\boldsymbol{\sigma}$ and $\boldsymbol{P}$ which are, respectively, symmetric and asymmetric.

As before, local deformation metrics are encoded into $\boldsymbol{\varepsilon}$. For example, volumetric deformation is characterized by the *trace* of $\boldsymbol{\varepsilon}$, viz.

$$\frac{\mathrm{d}v}{\mathrm{d}V} = 1 + \mathrm{tr}\,\boldsymbol{\varepsilon} = 1 + \varepsilon_{ii}, \quad (1.64)$$

while stretches in the three coordinate directions are given by $\varepsilon_{(ii)}$ (parentheses implying no summation over i) and angle changes are identified as $\gamma_{ij} = 2\varepsilon_{ij}$ with $i \neq j$.

In linearized kinematics, all three stress tensors coincide and one commonly uses only the Cauchy stress tensor $\boldsymbol{\sigma}$ to define the constitutive relation $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon})$. In the simplest case of **linear elasticity**, those are linearly linked via

$$\boldsymbol{\sigma} = \mathbb{C}\,\boldsymbol{\varepsilon} \qquad \Leftrightarrow \qquad \sigma_{ij} = \mathbb{C}_{ijkl}\varepsilon_{kl} \quad (1.65)$$

with a fourth-order **elasticity tensor** $\mathbb{C}$ linking each component of $\boldsymbol{\sigma}$ to those of $\boldsymbol{\varepsilon}$. Alternatively, we can again encode the constitutive response in a *strain energy density* $W = W(\boldsymbol{\varepsilon})$, which, e.g., for the case of linear elasticity reads

$$W = \frac{1}{2}\boldsymbol{\varepsilon} \cdot \mathbb{C}\boldsymbol{\varepsilon} \qquad \Leftrightarrow \qquad W = \frac{1}{2}\varepsilon_{ij}\mathbb{C}_{ijkl}\varepsilon_{kl}, \quad (1.66)$$

so that

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\varepsilon}}, \qquad \mathbb{C} = \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} = \frac{\partial^2 W}{\partial \boldsymbol{\varepsilon}\,\partial \boldsymbol{\varepsilon}}. \quad (1.67)$$

When taking derivatives with respect to $\boldsymbol{\varepsilon}$, caution is required since $\boldsymbol{\varepsilon}$ is symmetric, so $\varepsilon_{ij} = \varepsilon_{kl}$ and, consequently, derivatives with respect to ε_{ij} must also take into account those terms containing ε_{ji} (for $i \neq j$). Therefore, the derivative should always be computed as

$$\boxed{\frac{\partial}{\partial \varepsilon_{ij}} = \frac{1}{2}\left(\frac{\partial}{\partial \varepsilon_{ij}} + \frac{\partial}{\partial \varepsilon_{ji}}\right).} \tag{1.68}$$

Alternatively, one may also use $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^{\mathrm{T}}$ and simply replace $\boldsymbol{\varepsilon} = \frac{1}{2}(\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^{\mathrm{T}})$ before differentiating.

The traction vector on a surface with unit normal $\boldsymbol{n}$ now becomes $\boldsymbol{t} = \boldsymbol{\sigma}\boldsymbol{n}$.

The local statement of linear momentum balance in linearized kinematics is

$$\boxed{\operatorname{div}\boldsymbol{\sigma} + \rho\boldsymbol{b} = \rho\boldsymbol{a}} \qquad \Leftrightarrow \qquad \boxed{\sigma_{ij,j} + \rho b_i = \rho a_i} \tag{1.69}$$

where the small-strain versions of density, body force density and acceleration field were introduced as ρ, $\boldsymbol{b}$ and $\boldsymbol{a} = \ddot{\boldsymbol{u}}$, respectively.

In small strains, we can insert the kinematic and linear elastic constitutive relations as well as the definition of the acceleration field into linear momentum balance to obtain

$$(\mathbb{C}_{ijkl}\varepsilon_{kl})_{,j} + \rho b_i = \rho \ddot{u}_i \qquad \Leftrightarrow \qquad (\mathbb{C}_{ijkl}u_{k,l})_{,j} + \rho b_i = \rho \ddot{u}_i \tag{1.70}$$

and in case of a *homogeneous* body with $\mathbb{C}(\boldsymbol{x}) = \mathbb{C} = \text{const.}$ we finally arrive at

$$\mathbb{C}_{ijkl}u_{k,lj} + \rho b_i = \rho \ddot{u}_i, \tag{1.71}$$

which is known as **Navier's equation** to be solved for the unknown field $\boldsymbol{u}(\boldsymbol{x},t)$.

Finally, the following will be helpful when implementing material models in our code. Note that we may use the relations

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \qquad \text{and} \qquad F_{iJ} = \delta_{iJ} + u_{i,J} \tag{1.72}$$

to write (using the chain rule)

$$\frac{\partial W}{\partial u_{i,J}} = \frac{\partial W}{\partial F_{kL}}\frac{\partial F_{kL}}{\partial u_{i,J}} = P_{kL}\frac{\partial}{\partial u_{i,J}}(\delta_{kL} + u_{k,L}) = P_{kL}\delta_{ik}\delta_{JL} = P_{iJ} \tag{1.73}$$

and (exploiting the symmetry of $\boldsymbol{\sigma}$)

$$\frac{\partial W}{\partial u_{i,j}} = \frac{\partial W}{\partial \varepsilon_{kl}}\frac{\partial \varepsilon_{kl}}{\partial u_{i,j}} = \sigma_{kl}\frac{\partial}{\partial u_{i,j}}\frac{1}{2}(u_{k,l} + u_{l,k}) = \frac{1}{2}\sigma_{kl}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) = \frac{1}{2}(\sigma_{ij} + \sigma_{ji}) = \sigma_{ij}. \tag{1.74}$$

Therefore, we can use the alternative relations for our stresses

$$\boxed{P_{iJ} = \frac{\partial W}{\partial u_{i,J}}} \qquad \text{and} \qquad \boxed{\sigma_{ij} = \frac{\partial W}{\partial u_{i,j}}.} \tag{1.75}$$

The beauty in those relations is that the stress tensor definition is now *identical irrespective of whether we are working in linearized or finite kinematics.*

1.4 Summary and Looking Ahead

So far, we have seen how partial differential equations govern the thermal and mechanical behavior of solid bodies (and, of course, those two can be coupled as well to describe the thermo-mechanical behavior of deformable bodies). In order to solve a problem, we need an **initial boundary value problem** (IBVP), which furnishes the above equations with appropriate **boundary conditions** (BCs) and **initial conditions** (ICs).

To this end, we subdivide the boundary $\partial\Omega$ of a body Ω into

- $\partial\Omega_D \equiv$ **Dirichlet boundary**, prescribing the *primary field* ($\boldsymbol{\varphi}$, T, etc.):

$$\text{e.g.} \qquad \boldsymbol{u}(\boldsymbol{x},t) = \hat{\boldsymbol{u}}(\boldsymbol{x},t) \text{ on } \partial\Omega_D \qquad \text{or} \qquad T(\boldsymbol{x},t) = \hat{T}(\boldsymbol{x},t) \text{ on } \partial\Omega_D. \tag{1.76}$$

- $\partial\Omega_N \equiv$ **Neumann boundary**, prescribing *derivatives of the primary field* ($\boldsymbol{F}$, $\boldsymbol{\beta}$, etc.):

$$\text{e.g.} \qquad \boldsymbol{t}(\boldsymbol{x},t) = \boldsymbol{\sigma}(\boldsymbol{x},t)\boldsymbol{n}(\boldsymbol{x},t) = \hat{\boldsymbol{t}}(\boldsymbol{x},t) \text{ on } \partial\Omega_N \qquad \text{or} \qquad q(\boldsymbol{x},t) = \hat{q}(\boldsymbol{x},t) \text{ on } \partial\Omega_N, \tag{1.77}$$

Note that we may generally assume that

$$\partial\Omega_D \cup \partial\Omega_N = \partial\Omega \qquad \text{and in most problems also} \qquad \partial\Omega_D \cap \partial\Omega_N = \varnothing. \tag{1.78}$$

In addition, all time-dependent problems require **initial conditions**, e.g.,

$$\begin{aligned} & T(\boldsymbol{x},0) = T_0(\boldsymbol{x}) \quad \forall\ \boldsymbol{X} \in \Omega, \\ \text{or} \qquad & \boldsymbol{\varphi}(\boldsymbol{X},0) = \boldsymbol{x}_0(\boldsymbol{X}) \quad \text{and} \quad \boldsymbol{V}(\boldsymbol{X},0) = \boldsymbol{V}_0(\boldsymbol{X}) \quad \forall\ \boldsymbol{X} \in \Omega. \end{aligned} \tag{1.79}$$

The number of required BCs/ICs depends on the **order of a PDE**, e.g.,

$$\rho c_v \dot{T} = \operatorname{div}(\boldsymbol{K} \operatorname{grad} T) + \rho s \tag{1.80}$$

is first-order in time and therefore requires one IC, e.g., $T(\boldsymbol{x},0) = T_0(\boldsymbol{x})$. It is second-order in space and hence requires BCs along all $\partial\Omega$ (e.g., two conditions per x and y coordinates).

In summary, we will have governing PDEs supplemented by ICs and BCs, as required (e.g., quasistatic problems, of course, do not require any initial conditions). Those need to be solved for the primary fields (e.g., temperature T or displacements $\boldsymbol{u}$ or the deformation mapping $\boldsymbol{\varphi}$).

Unfortunately, analytical solutions are hardly ever available – except for relatively simple problems involving

- simple geometries,
- simple material behavior,
- simple ICs/BCs.

For realistic geometries, materials and/or ICs/BCs, one usually requires numerical techniques to obtain *approximate solutions*.

2 Numerical Methods

To numerically solve such ODEs/PDEs, we generally have so-called direct and indirect methods.

Direct methods aim to solve the governing equations directly; for example, using **finite differences** (FD). Consider the isotropic heat equation discussed before, which for brevity we write in 1D as (absorbing the coefficient into k and r).

$$\dot{T} = k\,T_{,xx} + r. \tag{2.1}$$

Introduce a regular $(\Delta x, \Delta t)$-grid with $T_i^\alpha = T(x_i, t^\alpha)$ and use **Taylor expansions**, e.g., in space:

$$T(x_{i+1}, t^\alpha) = T_{i+1}^\alpha = T_i^\alpha + \Delta x\, \frac{\partial T}{\partial x}\bigg|_{x_i,t^\alpha} + \frac{(\Delta x)^2}{2}\, \frac{\partial^2 T}{\partial x^2}\bigg|_{x_i,t^\alpha} + \frac{(\Delta x)^3}{3!}\, \frac{\partial^3 T}{\partial x^3}\bigg|_{x_i,t^\alpha} + O(\Delta x^4) \tag{2.2}$$

$$T(x_{i-1}, t^\alpha) = T_{i-1}^\alpha = T_i^\alpha - \Delta x\, \frac{\partial T}{\partial x}\bigg|_{x_i,t^\alpha} + \frac{(\Delta x)^2}{2}\, \frac{\partial^2 T}{\partial x^2}\bigg|_{x_i,t^\alpha} - \frac{(\Delta x)^3}{3!}\, \frac{\partial^3 T}{\partial x^3}\bigg|_{x_i,t^\alpha} + O(\Delta x^4) \tag{2.3}$$

Addition of the two equations gives:

$$T_{i+1}^\alpha + T_{i-1}^\alpha = 2T_i^\alpha + (\Delta x)^2\, \frac{\partial^2 T}{\partial x^2}\bigg|_{x_i,t^\alpha} + O(\Delta x^4) \quad \Rightarrow \quad \boxed{\frac{\partial^2 T}{\partial x^2}(x_i, t^\alpha) = \frac{T_{i+1}^\alpha - 2T_i^\alpha + T_{i-1}^\alpha}{(\Delta x)^2} + O(\Delta x^2)} \tag{2.4}$$

This is the **second-order central difference** approximation.

Analogously, Taylor expansion in time and subtraction of the two equations gives:

$$T_i^{\alpha+1} - T_i^{\alpha-1} = 2\Delta t\, \frac{\partial T}{\partial t}\bigg|_{x_i,t^\alpha} + O(\Delta t^3) \quad \Rightarrow \quad \boxed{\frac{\partial T}{\partial t}(x_i, t^\alpha) = \frac{T_i^{\alpha+1} - T_i^{\alpha-1}}{2\Delta t} + O(\Delta t^2)} \tag{2.5}$$

which is the **first-order central difference** approximation. Many other such finite-difference approximations of derivatives can be obtained in a smiliar fashion. For example, a simpler first-order stencil is obtained from the first Taylor equation (2.2) alone:

$$T_i^{\alpha+1} - T_i^\alpha = \Delta t\, \frac{\partial T}{\partial t}\bigg|_{x_i,t^\alpha} + O(\Delta t^2) \quad \Rightarrow \quad \boxed{\frac{\partial T}{\partial t}(x_i, t^\alpha) = \frac{T_i^{\alpha+1} - T_i^\alpha}{\Delta t} + O(\Delta t)} \tag{2.6}$$

which is often referred to as the first-order **forward-Euler** approximation. Analogously, we can use the second Taylor equation (2.3) to obtain the **backward-Euler** approximation

$$T_i^\alpha - T_i^{\alpha-1} = \Delta t\, \frac{\partial T}{\partial t}\bigg|_{x_i,t^\alpha} + O(\Delta t^2) \quad \Rightarrow \quad \boxed{\frac{\partial T}{\partial t}(x_i, t^\alpha) = \frac{T_i^\alpha - T_i^{\alpha-1}}{\Delta t} + O(\Delta t)} \tag{2.7}$$

In order to numerically solve a PDE directly, we choose suitable finite-difference approximations for all appearing derivatives. For example, using the second-order central-difference approximation for the spatil and the forward-Euler approximation for the temporal derivative in the heat equation, the *discretized governing equation* becomes

$$\frac{T_i^{\alpha+1} - T_i^\alpha}{\Delta t} = k\frac{T_{i+1}^\alpha - 2T_i^\alpha + T_{i-1}^\alpha}{(\Delta x)^2} + r(x_i, t^\alpha) + O(\Delta t, \Delta x^2), \tag{2.8}$$

which in the limit $\Delta t, \Delta x \to 0$ is expected to converge towards the same solution as the governing equation (this is the requirement of *consistency* of the discretized equation).

Note that for known values T_i^α at the current time, the above equation can easily be solved for $T_i^{\alpha+1}$ at the new time. In fact, the right-hand side does not involve $T_i^{\alpha+1}$, which is why this finite-difference scheme is **explicit**.

By contrast, when using the backward-Euler approximation, we obtain

$$\frac{T_i^\alpha - T_i^{\alpha-1}}{\Delta t} = k\frac{T_{i+1}^\alpha - 2T_i^\alpha + T_{i-1}^\alpha}{(\Delta x)^2} + r(x_i, t^\alpha) + O(\Delta t, \Delta x^2), \tag{2.9}$$

which is a linear system to be solved for $T_i^{\alpha+1}$ at the new time step and is therefore an **implicit** scheme.

Numerical solution can be interpreted via **stencils**, which may also reveal the required BCs/ICs.

Problems associated with direct methods include:

- a *regular grid* is required (which is fine for many fluidic mechanics problems but oftentimes problematic for complex solid geometries).
- variables are *defined only at grid points*, hence the error is minimized only at grid points (and we have no information about what happens between grid points; both the primary fields and their errors are undefined between grid points). This can be problematic when seeking approximate solutions that are "*globally optimal*". Also, how to apply BCs/ICs in between grid points, how about moving BCs?
- *stability/efficiency issues* (probably known from fundamental computational mechanics classes: CFL-condition, von Neumann stability analysis, etc.). In a nutshell, the choice of Δt and Δx is not arbitrary but – aside from accuracy concerns – the stability of, especially explicit, finite-difference schemes dictates maximum step widths to be used.

As an **alternative**, **indirect methods** do not solve the ODEs/PDEs directly but search for optimal approximations, e.g., $\boldsymbol{u}^h(\boldsymbol{x}) \approx \boldsymbol{u}(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$, including BCs and ICs. To do so, we need to discuss a lot more.

Particular **questions** to be addressed include:

- How do we choose $\boldsymbol{u}^h(\boldsymbol{x})$? For example, globally or locally defined functions? Which choices minimize the error?
- What is "*optimal*"? How do we quantify between error approximation and exact solution?
- What trial functions shall we use? For example, polynomial or Fourier series, piecewise defined or maybe even piece-constant?

We need a couple concepts to address those questions. Note that in the following, we will formulate most concepts in 1D with analogous generalizations possible for higher dimensions unless specifically mentioned.

3 Variational Calculus

3.1 Functionals

In order to understand the big picture of indirect methods, let us first discuss the energetics of deformable bodies and, in particular, introduce the concepts of functionals and variational calculus.

A **functional** is special type of mapping which maps from a function space $\mathcal{U}$ to $\mathbb{R}$:

$$I : u \in \mathcal{U} \to I[u] \in \mathbb{R}. \tag{3.1}$$

Oftentimes, functionals impose constraints on the function space $\mathcal{U}$, usually differentiability/integrability constrainst. As an example, consider

$$I[u] = \int_0^1 u^2(x)\,\mathrm{d}x, \tag{3.2}$$

which is a *functional* requiring that u is square-integrable.

It will be important to restrict the space of admissible functions when seeking for "*physically reasonable*" approximations $u^h(x)$. For example, we may want to restrict how smooth a function is and whether or not it is allowed to have any poles or discontinuities, etc.

To this end, we quickly introduce the **Sobolev space**

$$H^k(\Omega) = \left\{u : \Omega \to \mathbb{R} \quad \text{such that} \quad \|u\|_{H^k(\Omega)} < \infty\right\}, \tag{3.3}$$

with the **Sobolev norm** (in 1D)

$$\|u\|^2_{H^k(\Omega)} = \int_\Omega u(x)^2\,\mathrm{d}x + \int_\Omega \left[u'(x)\right]^2\,\mathrm{d}x + \ldots + \int_\Omega \left[u^{(k)}(x)\right]^2\,\mathrm{d}x. \tag{3.4}$$

Consequently, $H^k(\Omega)$ denotes the space of all functions whose derivatives up to kth order are **square-integrable**. The above example in (3.2), e.g., requires $u \in \mathcal{U} \subset H^0(0,1)$. That is functions $u(x)$ must be square-integrable on the interval $(0,1)$.

Our classical example will be the energy of a mechanical system which depends on the displacement field $\boldsymbol{u}(\boldsymbol{x})$ and defines an energy $I \in \mathbb{R}$. Consider, e.g., the 1D strain energy of a bar of length L and with constant Young's modulus E and cross-sectional area A:

$$I[u] = \int_0^L \frac{E}{2}\left[u_{,x}(x)\right]^2 A\,\mathrm{d}x, \tag{3.5}$$

where we used that $W = \frac{E}{2}\varepsilon^2$ and $\varepsilon = u_{,x}$ with $u = u(x)$. Here, we generally may want to impose the restriction $u \in H^1(0,L)$, unless dealing with discontinuities such as cracks.

Functionals are to be distinguished from *functions* such as $f(\boldsymbol{x}) = \sqrt{x_1^2 + x_2^2} = |\boldsymbol{x}|$ with $f : \mathbb{R}^2 \to \mathbb{R}_0^+$. Unlike a *function* which is a mapping from $\mathbb{R}^d \to \mathbb{R}$, a *functional*'s domain is generally a function space $\mathcal{U}$ (e.g., all polynomial functions up to a certain degree, or all continuously differentiable functions, or all piecewise polynomial functions).

3.2 Variations

Consider a functional $I : \mathcal{U} \to \mathbb{R}$ such as the potential energy. Analogous to the stationarity condition of classical optimization problems, a necessary condition for an extremum of I is that the first variation of I vanishes, i.e., $\delta I[u] = 0$ (this is the **stationarity condition**). Like in functional analysis, we are going to compute the analog of a derivative to identify maxima and minima of the functional. To this end, we perturb the functional around a point by a small variation and verify if the functional increases or decreases. Complicating is the fact that a "*point*" is now in fact a function, and a variation must be a variation of that function. To this end, we define the following.

A **variation** δu is an arbitrary function that represents admissible changes of u. If $\Omega \subset \mathbb{R}^d$ is the domain of $u \in \mathcal{U}$ with boundary $\partial\Omega$, we seek solutions

$$u \in \mathcal{U} = \left\{ u \in H^k(\Omega) \;:\; u = \hat{u} \text{ on } \partial\Omega_D \right\} \tag{3.6}$$

then the variation must satisfy

$$\delta u \in \mathcal{U}_0 = \left\{ \delta u \in H^k(\Omega) \;:\; \delta u = 0 \text{ on } \partial\Omega_D \right\}. \tag{3.7}$$

k can be determined from the specific form of $I[u]$, as will be discussed later. The fact that variations $\delta \boldsymbol{u}$ must vanish on the Dirichlet boundary $\partial\Omega_D$ stems from the need for perturbations that allow the perturbed function $\boldsymbol{u} + \delta\boldsymbol{u}$ to still satisfy the Dirichlet boundary conditions.

With this, we define the **first variation** of I (analog of a first derivative) as

$$\boxed{\delta I[u] = \lim_{\epsilon\to 0} \frac{I[u+\epsilon\,\delta u] - I[u]}{\epsilon} = \frac{\mathrm{d}}{\mathrm{d}\epsilon} I[u+\epsilon\,\delta u]\bigg|_{\epsilon\to 0}} \tag{3.8}$$

and analogously higher-order variations via

$$\boxed{\delta^k I[u] = \delta\left(\delta^{k-1} I\right) \qquad \text{for} \qquad k \geq 2} \tag{3.9}$$

Note that a *Taylor expansion of a functional* I can now be written as

$$I[u+\delta u] = I[u] + \delta I[u] + \frac{1}{2!}\delta^2 I[u] + \frac{1}{3!}\delta^3 I[u] + \ldots \tag{3.10}$$

The following are helpful relations for $u, v \in \mathcal{U}$, further $I_i : \mathcal{U} \to \mathcal{V} \subset \mathbb{R}$, and constants $\alpha_i \in \mathbb{R}$:

- $\delta\left(\alpha_1 I_1 + \alpha_2 I_2\right) = \alpha_1\,\delta I_1 + \alpha_2\,\delta I_2$
- $\delta(I_1 I_2) = (\delta I_1) I_2 + I_1(\delta I_2)$
- $\delta \frac{\mathrm{d}u}{\mathrm{d}x} = \frac{\mathrm{d}}{\mathrm{d}x}\delta u$ (assuming differentiability of u, specifically $u \in C^1$)
- $\delta \int_\Omega u\,\mathrm{d}x = \int_\Omega \delta u\,\mathrm{d}x$ (assuming Ω is independent of u)
- $\delta I[u, v, \ldots] = \frac{\mathrm{d}}{\mathrm{d}\epsilon} I[u+\epsilon\,\delta u, v+\epsilon\,\delta v, \ldots]_{\varepsilon\to 0}$

Example:

Let us consider

$$I[u] = \|u\|^2_{L_2(0,1)} = \int_0^1 u^2\,\mathrm{d}x \qquad \text{so that we seek} \qquad u \in \mathcal{U} = H^0(0,1). \tag{3.11}$$

The variations follow as

$$\begin{aligned}
\delta I &= \lim_{\epsilon\to 0} \frac{\mathrm{d}}{\mathrm{d}\epsilon} \int_0^1 (u+\epsilon\,\delta u)^2\,\mathrm{d}x = \lim_{\epsilon\to 0} \int_0^1 2(u+\epsilon\,\delta u)\delta u\;\mathrm{d}x = 2\int_0^1 u\,\delta u\;\mathrm{d}x \\
\delta^2 I &= \lim_{\epsilon\to 0} \delta I[u+\epsilon\,\delta u] = \frac{\mathrm{d}}{\mathrm{d}\epsilon}\int_0^1 2(u+\epsilon\,\delta u)\,\delta u\;\mathrm{d}x = 2\int_0^1 (\delta u)^2\,\mathrm{d}x \\
\delta^k I &= 0 \qquad \text{for all} \quad k > 2.
\end{aligned} \tag{3.12}$$

Notice that

$$\begin{aligned}
I[u+\delta u] &= \int_0^1 (u+\delta u)^2\,\mathrm{d}x = \int_0^1 u^2\;\mathrm{d}x + \int_0^1 2u\,\delta u\;\mathrm{d}x + \int_0^1 (\delta u)^2\,\mathrm{d}x \\
&= I[u] + \delta I[u] + \frac{1}{2}\delta^2 I[u].
\end{aligned} \tag{3.13}$$

As a *practical hint*, note that for any integral

$$I[\boldsymbol{u}] = \int_\Omega f(u_i, u_{i,j}, \ldots)\;\mathrm{d}V \tag{3.14}$$

we can write

$$\frac{\partial I[\boldsymbol{u}+\epsilon\,\delta\boldsymbol{u}]}{\partial\epsilon} = \int_\Omega \left(\frac{\partial f}{\partial u_i}\delta u_i + \frac{\partial f}{\partial u_{i,j}}\delta u_{i,j} + \ldots\right)\mathrm{d}V \tag{3.15}$$

and omit the lengthy derivation of the first variation given above.

Here comes the reason we look into variations: some classes of partial differential equations possess a so-called **variational structure**; i.e., their solutions $u \in \mathcal{U}$ can be interpreted as extremal points over $\mathcal{U}$ of a functional $I[u]$.

3.3 Example: Hanging bar under its own weight

Consider a bar of length L (Young's modulus E, cross-sectional area A, density ρ) that is hanging from the ceiling and deformed under its own weight (gravitational acceleration g). Let us denote the 1D displacement field $u(x)$ where x runs from top to bottom of the bar. The total potential energy is thus the total strain energy of the elastic bar minus the work done by the graviational body forces:

$$I[u] = \int_0^L \frac{E}{2} u_{,x}^2(x)\,A\,\mathrm{d}x - \int_0^L \rho g\,u(x)\,A\,\mathrm{d}x \qquad \text{and} \qquad u(x) \in \mathcal{U} = \left\{u \in H^1(0,L)\;:\; u(0) = 0\right\}. \tag{3.16}$$

The first variation yields

$$\begin{aligned}
\delta I[u] &= \int_0^L E u_{,x}(x)\,\delta u_{,x}(x)\,A\,\mathrm{d}x - \int_0^L \rho g\,\delta u(x)\,A\,\mathrm{d}x \\
&= -\int_0^L \left[E u_{,xx}(x) + \rho g\right]\delta u(x)\,A\,\mathrm{d}x + EAu_{,x}(L)\,\delta u(L) - EAu_{,x}(0)\,\delta u(0) = 0,
\end{aligned} \tag{3.17}$$

where we used integration by parts of the first integral to arrive at the final form. Noting that $\delta u_x(0)$ because of boundary conditions, the last term vanishes. Finally, recall that the above variation must vanish for *all* variations $\delta u \in \mathcal{U}_0$. This implies that (3.17) implies we must have

$$E u_{,xx}(x) + \rho g = 0 \qquad \text{and} \qquad EAu_{,x}(L) = 0. \tag{3.18}$$

These are exactly the governing equations and traction-free boundary condition that the bar needs to satisfy. Hence, we have shown that minimizing (3.16) over all $u(x) \in \mathcal{U}$ is equivalent to solving (3.18) with $u(0) = 0$. That is, we have to theoretical strategy to replace the solution of a differential equation by a minimization problem.

Note that we can easily find the analytical solution to the problem by integrating (3.18) twice for $u(x)$ and inserting the boundary conditions, resulting in

$$u(x) = -\frac{\rho g}{2E} x(x - 2L). \tag{3.19}$$

By the way, this solution to the system of differential equations is called the **classical solution**.

One way to exploit the above variational structure is the so-called **Rayleigh-Ritz** approach, which introduces an approximation $u^h(x) \approx u(x)$, e.g., a polynomial series

$$u^h(x) = \sum_{i=0}^{n} c_i x^i \qquad \text{where} \qquad c_0 = 0 \text{ because of BCs} \tag{3.20}$$

with unknown coefficients $c_i \in \mathbb{R}$. Of course, any choice of ansatz functions is permissible including, e.g., Fourier series of cosine or sine terms, as long as they satisfy the essential boundary condition $u(0) = 0$ and the differentiability/integrability requirements (e.g., a piecewise linear guess for $u^h(x)$ would not be permissible as its derivatives are not square-integrable). Next, we insert $u^h(x)$ into (3.16), i.e.,

$$I[u^h] = \int_0^L \frac{E}{2} (u^h_{,x})^2(x)\, A\, \mathrm{d}x - \int_0^L \rho g\, u^h(x)\, A\, \mathrm{d}x, \tag{3.21}$$

which can be integrated to depend only on the unknown cofficients c_i. Finally, we find the solution by minimization with respect to the coefficients:

$$0 = \frac{\partial I[u^h]}{\partial c_i}, \tag{3.22}$$

which gives n equations for the n unknown coefficients. Note that the above form of the energy results in a linear system of equations for the unknown coefficients c_i, which can be solved numerically in an efficient manner.

If we use polynomial ansatz functions as in (3.20), then the exact solution (3.19) is contained in the solution space if $n \geq 2$. In fact, if one chooses $n \geq 2$ and solves for the unknown coefficients, then one obtains

$$c_1 = \frac{\rho g}{E} L \qquad c_2 = -\frac{\rho g}{2E}, \qquad c_i = 0 \quad \text{for} \quad i > 2, \tag{3.23}$$

which is the exact solution (3.19).

3.4 Example: Static Heat Conduction

As an introductory example, let us review the *static heat conduction problem* in d dimensions, defined by ($\boldsymbol{N} \in \mathbb{R}^d$ denoting the outward unit normal vector)

$$\left[\begin{array}{rl} \kappa\, \Delta T + \rho s = 0 & \text{in } \Omega, \\ T = \hat{T} & \text{on } \partial\Omega_D, \\ q = -\kappa \operatorname{grad} T \cdot \boldsymbol{n} = \hat{q} & \text{on } \partial\Omega_N, \end{array}\right. \tag{3.24}$$

and we seek solutions $T : \Omega \to \mathbb{R}$ that satisfy all of the above equations and meet the required differentiability conditions. Such solutions are called **classical solution**.

As an alternative to solving the above equations, consider the total potential energy defined by the functional $I : \mathcal{U} \to \mathbb{R}$ with

$$I[T] = \int_\Omega \left(\frac{\kappa}{2} \|\text{grad}\, T\|^2 - \rho s T \right) \text{d}V + \int_{\partial\Omega_N} \hat{q} T \,\text{d}S. \tag{3.25}$$

The specific form shows that we need to seek solutions in the space

$$\mathcal{U} = \left\{ T \in H^1(\Omega) \;:\; T = \hat{T} \text{ on } \partial\Omega_D \right\} \qquad \text{and} \qquad \mathcal{U}_0 = \left\{ \delta T \in H^1(\Omega) \;:\; \delta T = 0 \text{ on } \partial\Omega_D \right\}. \tag{3.26}$$

Let us find extremal points $T \in \mathcal{U}$ that render $I[T]$ stationary.

The first variation follows as

$$\delta I[T] = \int_\Omega \left(\frac{\kappa}{2} 2 T_{,i}\, \delta T_{,i} - \rho s\, \delta T \right) \text{d}V + \int_{\partial\Omega_N} \hat{q}\, \delta T \,\text{d}S = 0 \qquad \text{for all } \delta T \in \mathcal{U}_0. \tag{3.27}$$

Application of the divergence theorem to the first term yields

$$\int_{\partial\Omega} \kappa\, T_{,i} n_i\, \delta T \,\text{d}S - \int_\Omega \kappa\, T_{,ii}\, \delta T \,\text{d}V - \int_\Omega \rho s\, \delta T \,\text{d}V + \int_{\partial\Omega_N} \hat{q}\, \delta T \,\text{d}S = 0 \qquad \text{for all } \delta T \in \mathcal{U}_0. \tag{3.28}$$

Rearranging terms and using the fact that $\delta T = 0$ on $\partial\Omega_D$ leads to

$$-\int_\Omega (\kappa\, T_{,ii} + \rho s)\, \delta T \,\text{d}V + \int_{\partial\Omega_N} (\kappa\, T_{,i} n_i + \hat{q})\, \delta T \,\text{d}S = 0 \qquad \text{for all } \delta T \in \mathcal{U}_0. \tag{3.29}$$

This must hold for all admissible variations $\delta T \in \mathcal{U}_0$. Therefore, (3.29) is equivalent to stating

$$\kappa\, \Delta T + \rho s = 0 \text{ in } \Omega, \qquad -\kappa (\text{grad}\, T) \boldsymbol{n} = \hat{q} \text{ on } \partial\Omega_N \qquad \text{and} \qquad T = \hat{T} \text{ on } \partial\Omega_D. \tag{3.30}$$

Ergo, extremal points $T \in \mathcal{U}$ of (3.25) are guaranteed to satisfy the governing equations (3.24) and are thus classical solutions.

To see if it is a *maximizer or minimizer*, let us compute the second variation

$$\delta^2 I[T] = \int_\Omega \kappa \delta T_{,i}\, \delta T_{,i} \,\text{d}V = \int_\Omega \kappa\, \|\delta\, \text{grad}\, T\|^2 \,\text{d}V \geq 0. \tag{3.31}$$

Hence, the extremum is a minimizer, assuming that $\kappa > 0$. Otherwise, note that $\kappa < 0$ leads to solutions being (unstable) energy maxima, which implies that $\kappa > 0$ is a (necessary and sufficient) *stability condition*.

Notice that (assuming that $\kappa = \text{const.}$) we can rewrite the energy functional for short as

$$I[T] = \frac{1}{2} \mathcal{B}(T, T) - \mathcal{L}(T), \tag{3.32}$$

where we introduced the **bilinear form** $\mathcal{B}$ and the **linear form** $\mathcal{L}$ as

$$\mathcal{B}(\cdot,\cdot) = \kappa \,\langle \text{grad}\, \cdot, \text{grad}\, \cdot \rangle_\Omega \qquad \text{and} \qquad \mathcal{L}(\cdot) = \langle \rho s, \cdot \rangle_\Omega - \langle \hat{q}, \cdot \rangle_{\partial\Omega_N} \tag{3.33}$$

with the inner product operator

$$\langle f, g \rangle_\Omega = \int_\Omega f g \,\text{d}V. \tag{3.34}$$

This is in fact a recipe for a more general class of variational problems: let us consider an energy functional of the general form

$$\boxed{\begin{aligned} I[u] &= \frac{1}{2}\mathcal{B}(u,u) - \mathcal{L}_\Omega(u) - \mathcal{L}_{\partial\Omega}(u) \\ &= \frac{1}{2}\kappa \langle \operatorname{grad} u, \operatorname{grad} u\rangle_\Omega - \langle \rho s, u\rangle_\Omega - \langle \hat{q}, u\rangle_{\partial\Omega_N} \end{aligned}} \tag{3.35}$$

with $u \in \mathcal{U}$ being some (scalar- or vector-valued) mapping and ρs and $\hat{q}$ denoting, respectively, distributed body sources and surface fluxes. Now we have

$$\begin{aligned} \delta I[u] &= \mathcal{B}(u,\delta u) - \mathcal{L}(\delta u) \\ &= -\int_\Omega [\kappa \operatorname{div}(\operatorname{grad} u) + \rho s]\,\delta u \,\mathrm{d}V - \int_{\partial\Omega_N} [\hat{q} - \lambda(\operatorname{grad} u)\boldsymbol{n}]\,\delta u\,\mathrm{d}S. \end{aligned} \tag{3.36}$$

Thus, the energy density (3.35) is generally suitable for quasistatic problems of the type

$$\left[\begin{aligned} \kappa\,\Delta u + \rho s &= 0 && \text{in } \Omega \\ u &= \hat{u} && \text{on } \partial\Omega_D \\ \kappa(\operatorname{grad} u)\boldsymbol{n} &= \hat{q} && \text{on } \partial\Omega_N \end{aligned}\right. \tag{3.37}$$

Note that (3.37) describes not only heat conduction but the general form also applies to electromagnetism, elasticity (to be discussed later), and various other fields. Notice that, while (3.35) required $u \in H^1(\Omega)$ (highest derivatives are of first order), evaluating (3.37) in general requires that $u \in C^2(\Omega) \cap C^0(\overline{\Omega})$ (second derivatives are required). We will get back to this point later.

For notational purposes, let us adopt the following notation found in various textbooks on finite elements: the first variation is usually abbreviated as an operator acting on both the unknown field u and its variation δu; i.e., we write $\mathcal{G} : \mathcal{U} \times \mathcal{U}_0 \to \mathcal{V} \subset \mathbb{R}$ with

$$\boxed{\mathcal{G}(u,\delta u) = D_{\delta u} I[u] = \lim_{\epsilon\to 0} \frac{\mathrm{d}}{\mathrm{d}\epsilon} I[u + \delta u]} \tag{3.38}$$

One of the beauties of the above variational problem (3.37) is that a unique minimizer exists by the **Lax-Milgram theorem**, see Appendix E. Recall that for the linear heat problem above we already showed that the solution is a unique (global) minimizer if $\kappa > 0$.

4 The weak form

4.1 Classical and weak solutions

Consider a physical problem that is – as before – governed by the so-called **strong form**

$$\left[\begin{array}{rl} (\kappa\, u_{,i})_{,i} + s = 0 & \text{in } \Omega \\ u_i = \hat{u}_i & \text{on } \partial\Omega_D \\ \kappa u_{,i} n_i = \hat{q} & \text{on } \partial\Omega_N. \end{array}\right. \tag{4.1}$$

In order to describe the restrictions u must satisfy, let us define that a function u is of **class** $C^k(\Omega)$ with an integer $k \geq 0$ if it is k *times continuously differentiable* over Ω (i.e., u possesses derivatives up to the kth order and these derivatives are continuous functions).

Examples:

- Any kth-order polynomial $u(x)$ with $k \geq 0$ is generally $C^\infty(\mathbb{R})$.
- Consider a continuous, piecewise-linear function $u : \Omega = (0,2) \to \mathbb{R}$. Function u is $C^0(\Omega)$ but not $C^1(\Omega)$.
- The Heavyside function $H(x)$ is said to be $C^{-1}(\mathbb{R})$ since its "**zeroth derivative**" (i.e., the function itself) is not continuous.

If there are no discontinuities such as cracks, shocks, etc. (or discontinuities in the BCs/ICs) we usually assume that the classical solution fields are $C^\infty(\Omega)$, so we may take derivatives; otherwise, derivatives exist almost everywhere (**a.e.**)

It is convenient to define by $C_0^k(\Omega)$ the space of all functions contained in $C^k(\Omega)$ whose support is a *bounded subset* of Ω (i.e., $u(x) \neq 0$ only on a finite subset of Ω). Then, notice that

$$C_0^k(\Omega) \subset H_0^k(\Omega) \tag{4.2}$$

and

$$C_0^\infty(\Omega) = \bigcap_{k\geq 0} C_0^k(\Omega). \tag{4.3}$$

Going back to the problem described by the strong form (4.1) above, we need to seek solutions

$$u \in C^2(\Omega) \cap C^0(\overline{\Omega}), \tag{4.4}$$

i.e., functions u must be twice continuously differentiable within Ω and at least continuous up to the boundary $\partial\Omega$.

As we showed previously, the solution u can alternatively be found by using a variational approach, viz.

$$u = \arg\min \left\{ I[u] \ : \ u = \hat{u} \text{ on } \partial\Omega_D \right\} \tag{4.5}$$

whose stationarity condition is

$$\delta I[u] = \mathcal{G}(u, \delta u) = \mathcal{B}(u, \delta u) - \mathcal{L}(\delta u) = 0 \qquad \text{for all } \delta u \in \mathcal{U}_0(\Omega). \tag{4.6}$$

Therefore, we can reformulate that problem (without, in principle, knowing anything about variational calculus) as:

$$\boxed{\text{find } u \in \mathcal{U} \quad \text{s.t.} \quad \mathcal{G}(u,v) = \mathcal{B}(u,v) - \mathcal{L}(v) = 0 \qquad \text{for all } v \in \mathcal{U}_0(\Omega)} \tag{4.7}$$

This is called the **weak form** of the problem because we now seek solutions $u \in \mathcal{U}$ where

$$\mathcal{U} = \left\{ u \in \mathcal{H}^1(\Omega) \ : \ u = \hat{u} \text{ on } \partial\Omega_d \right\}, \tag{4.8}$$

that satisfy (4.7) for all $v \in \mathcal{U}_0(\Omega)$, and such a solution is called **weak solution**. There is one essential difference between the weak and strong form: solutions of the weak form are required to be in $H^1(\Omega)$, whereas the strong form required solutions to be in $C^2(\Omega)$. Thus, we have weakened/relaxed the conditions on the family of solutions, which is why the above is called the *weak form*.

Notice that, if v is interpreted as a virtual displacement field, then (4.7) is also referred to as the **principle of virtual work**.

Computationally, solving the weak form is usually preferable over the strong form. First, $u \in H^1(\Omega)$ is simpler to satisfy than $u \in C^2(\Omega)$ (e.g., piecewise linear interpolation is sufficient in the weak form but not in the strong form). Second, as we showed already for the *Rayleigh-Ritz* approach, the weak form boils down to solving a system of algebraic equations (rather than solving PDEs).

Let us show that we can also arrive at the weak form in an alternative fashion without the use of variational calculus. This is particularly helpful if no potential exists.

Let us take the first equation in (4.1), multiply it by some random trial function $v \in \mathcal{U}_0(\Omega)$ that vanishes on $\partial\Omega_D$, and integrate over the entire domain. The result, which must still vanish due to (4.1), is

$$0 = -\int_\Omega \left[(\kappa\, u_{,i})_{,i} + s \right] v \, \mathrm{d}V, \tag{4.9}$$

which must hold for all admissible $v \in \mathcal{U}_0(\Omega)$. This is the basis for the family of the so-called **methods of weighted residuals** (where one picks specific choices of v).

Using the divergence theorem and the fact that $v = 0$ on $\partial\Omega_D$ reduces the above to

$$\begin{aligned} 0 &= \int_\Omega \kappa\, u_{,i} v_{,i} \, \mathrm{d}V - \int_\Omega s v \, \mathrm{d}V - \int_{\partial\Omega_N} \kappa\, u_{,i} n_i v \, \mathrm{d}S \qquad \text{for all } v \in \mathcal{U}_0(\Omega) \\ &= \int_\Omega \kappa\, u_{,i} v_{,i} \, \mathrm{d}V - \int_\Omega s v \, \mathrm{d}V - \int_{\partial\Omega_N} \hat{q} v \, \mathrm{d}S \qquad \text{for all } v \in \mathcal{U}_0(\Omega), \end{aligned} \tag{4.10}$$

where we used the Neumann bounday condition $\kappa u_{,i} n_i = \hat{q}$ to transform the last term. The last equation in (4.10) is exactly identical to (4.7). In other words, we can find the weak form without the use of variational calculus, moreover, even without the existence of an energy functional by starting directly from the strong form. This is an important observation (even those problems that do not have a variational structure can thus be written in terms of a weak form). To decide whether or not a variational structure exists for a given problem, can be done by the use of Vainberg's theorem given in Appendix F.

4.2 Equivalence of strong and weak forms

We now have *two equivalent variational principles*:

Given a space

$$\mathcal{U} = \left\{ u \in \mathcal{H}^k(\Omega) \ : \ u = \hat{u} \text{ on } \partial\Omega_d \right\}, \tag{4.11}$$

a functional $I : \mathcal{U} \to \mathbb{R}$ and associated bilinear, continuous form $\mathcal{B}(\cdot,\cdot)$ defined on $\mathcal{U} \times \mathcal{U}$ and a continuous linear form $\mathcal{L}(\cdot)$ defined on $\mathcal{U}$, we seek to

$$(\mathbf{A}) \qquad \boxed{\text{find } u \in \mathcal{U} \quad \text{s.t.} \quad u = \arg\min I[u]} \tag{4.12}$$

$$(\mathbf{B}) \qquad \boxed{\text{find } u \in \mathcal{U} \quad \text{s.t.} \quad \mathcal{B}(u,v) = \mathcal{L}(v) \quad \text{for all } v \in \mathcal{U}_0} \tag{4.13}$$

And we know that the two have a unique connection since $\delta I = \mathcal{B}(u, \delta u) - \mathcal{L}(\delta u)$. Thus, we know that

$$(A) \Leftrightarrow (B) \tag{4.14}$$

with a unique solution for this particular type of problem (if it is stable, e.g., $\kappa > 0$).

4.3 Approximate solutions

The idea of numerical approaches is to find an approximate solution: we replace the space $\mathcal{U}$ by a *finite-dimensional* subspace

$$\mathcal{U}^h \subset \mathcal{U}, \tag{4.15}$$

in which we seek a solution u^h, where h stands for the discretization size.

An **n-dimensional space** $\mathcal{U}^h$ is defined by a set of n basis functions $\{N_1, \ldots, N_n\}$ and the approximation

$$\boxed{u^h(x) = \sum_{a=1}^{n} u^a N^a(x) \qquad \text{and} \qquad v^h(x) = \sum_{a=1}^{n} v^a N^a(x).} \tag{4.16}$$

Assume that the approximation space is chosen wisely, so the exact solution can be attained with infinite refinement; i.e., we assume that

$$\text{for all } u \in \mathcal{U} \quad \text{there exists} \quad u_h(v) \in \mathcal{U}^h \quad \text{such that} \quad \lim_{h\to 0} \|u_h(v) - u\| = 0. \tag{4.17}$$

Then we can formulate the **discrete problem**

$$(\mathbf{C}) \qquad \boxed{\text{find } u^h \in \mathcal{U}^h \quad \text{s.t.} \quad \mathcal{B}(u^h, v^h) = \mathcal{L}(v^h) \quad \text{for all } v^h \in \mathcal{U}_0^h} \tag{4.18}$$

Next, let us insert the approximations (4.16) into (4.18) to obtain:

$$\mathcal{B}\left(\sum_{a=1}^{n} u^a N^a, \sum_{b=1}^{n} v^b N^b \right) = \mathcal{L}\left(\sum_{b=1}^{n} v^b N^b \right) \qquad \text{for all } v^b \tag{4.19}$$

or, exploiting that $\mathcal{B}$ is bilinear and $\mathcal{L}$ is linear,

$$\sum_{b=1}^{n} v^b \left[\sum_{a=1}^{n} u^a \mathcal{B}\left(N^a, N^b\right) - \mathcal{L}\left(N^b\right) \right] = 0 \quad \text{for all } v^b. \tag{4.20}$$

Since this must hold for all (admissible) v^b, we conclude that

$$\boxed{\sum_{a=1}^{n} u^a \mathcal{B}\left(N^a, N^b\right) = \mathcal{L}\left(N^b\right)} \qquad \text{for } b = 1, \ldots, n. \tag{4.21}$$

This is a *linear* system to be solved for u^a $(a = 1, \ldots, n)$.

Let us define a vector of all unknown coefficients:

$$\boldsymbol{U}^h = \left\{u^1, \ldots, u^n\right\}^{\mathrm{T}}. \tag{4.22}$$

Further, we define a (symmetric) matrix $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ and vector $\boldsymbol{F} \in \mathbb{R}^n$ with components

$$K_{ab} = \mathcal{B}\left(N^a, N^b\right), \qquad F_b = \mathcal{L}\left(N^b\right). \tag{4.23}$$

Then, the linear system reads

$$\boldsymbol{K}\,\boldsymbol{U}^h = \boldsymbol{F} \qquad \Leftrightarrow \qquad K_{ba} U^h_a = F_b. \tag{4.24}$$

When we are using the same approximation space for u^h and v^h, this is the so-called the **Bubnov-Galerkin approximation**. Alternatively, one can choose different function spaces for the approximations u^h and v^h, which leads to the so-called **Petrov-Galerkin** method. The latter gains importance when solving *over/underconstrained problems* since it allows to control the number of equations by the choice of the dimension of the space of v^h.

5 The mechanical variational problem

5.1 Linearized kinematics

After all those precursors, let us analyze the mechanical variational problem and start with the simplest problem: *quasistatics in linearized kinematics*. Here, the **strong form** is

$$\left[\begin{array}{rl} \sigma_{ij,j} + \rho\, b_i = 0 & \text{in } \Omega, \\ u_i = \hat{u}_i & \text{on } \partial\Omega_D, \\ \sigma_{ij} n_j = \hat{t} & \text{on } \partial\Omega_N. \end{array}\right. \tag{5.1}$$

The associated total potential energy is

$$I[\boldsymbol{u}] = \int_\Omega W(\boldsymbol{\varepsilon})\,\mathrm{d}V - \int_\Omega \rho \boldsymbol{b}\cdot\boldsymbol{u}\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{\boldsymbol{t}}\cdot\boldsymbol{u}\,\mathrm{d}S \tag{5.2}$$

and we seek displacement field solutions

$$\boldsymbol{u} = \arg\min\left\{ I[\boldsymbol{u}] \;:\; \boldsymbol{u} = \hat{\boldsymbol{u}} \text{ on } \partial\Omega_D \right\}. \tag{5.3}$$

We compute the first variation, defining $\operatorname{sym}(\cdot) = \frac{1}{2}(\cdot + \cdot^{\mathrm{T}})$,

$$\begin{aligned} \delta I[\boldsymbol{u}] &= \int_\Omega \frac{\partial W}{\partial \varepsilon_{ij}} \operatorname{sym}(\delta u_{i,j})\,\mathrm{d}V - \int_\Omega \rho b_i \delta u_i \,\mathrm{d}V - \int_{\partial\Omega_N} \hat{t}_i \delta u_i\,\mathrm{d}S \\ &= \int_\Omega \sigma_{ij}\delta u_{i,j}\,\mathrm{d}V - \int_\Omega \rho b_i \delta u_i\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{t}_i\delta u_i\,\mathrm{d}S = 0 \qquad \forall \quad \delta\boldsymbol{u} \in \mathcal{U}_0, \end{aligned} \tag{5.4}$$

where we used $\sigma_{ij} = \partial W/\partial\varepsilon_{ij}$ and $\sigma_{ij} = \sigma_{ji}$ (by angular momentum balance). Note that application of the divergence theorem shows the equivalence of the two forms since

$$\delta I[\boldsymbol{u}] = 0 = \int_{\partial\Omega_N} (\sigma_{ij}n_j - \hat{t}_i)\delta u_i\,\mathrm{d}S - \int_\Omega (\sigma_{ij,j} + \rho b_i)\delta u_i\,\mathrm{d}V \qquad \forall \quad \delta\boldsymbol{u} \in \mathcal{U}_0. \tag{5.5}$$

We can use the first variation to define the weak form as

$$G(\boldsymbol{u},\boldsymbol{v}) = \mathcal{A}(\boldsymbol{u},\boldsymbol{v}) - \mathcal{L}(\boldsymbol{v}) = 0 \qquad \text{for all adm. } \boldsymbol{v} \tag{5.6}$$

with

$$\mathcal{A}(\boldsymbol{u},\boldsymbol{v}) = \int_\Omega \sigma_{ij}\big(\operatorname{sym}(\nabla\boldsymbol{u})\big) v_{i,j}\,\mathrm{d}V \qquad \text{and} \qquad \mathcal{L}(\boldsymbol{v}) = \int_\Omega \rho b_i v_i\,\mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i v_i\,\mathrm{d}S. \tag{5.7}$$

Notice that $\mathcal{A}(\cdot,\cdot)$ is generally *not* a bilinear operator, while $\mathcal{L}(\cdot)$ is a linear operator.

Next, we introduce the discrete weak form $\mathcal{A}(\boldsymbol{u}^h,\boldsymbol{v}^h) - \mathcal{L}(\boldsymbol{v}^h) = 0$ with the Bubnov-Galerkin approximation

$$\boldsymbol{u}^h(\boldsymbol{x}) = \sum_{a=1}^n \boldsymbol{u}^a N^a(\boldsymbol{x}) \qquad \text{and} \qquad \boldsymbol{v}^h(\boldsymbol{x}) = \sum_{a=1}^n \boldsymbol{v}^a N^a(\boldsymbol{x}), \tag{5.8}$$

so that we arrive at (in component form)

$$\sum_{a=1}^n v_i^a \left[\int_\Omega \sigma_{ij}\big(\operatorname{sym}(\nabla\boldsymbol{u}^h)\big) N^a_{,j}\,\mathrm{d}V - \int_\Omega \rho b_i N^a\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{t}_i N^a\,\mathrm{d}S \right] = 0 \qquad \text{for all adm. } \boldsymbol{v}^a \tag{5.9}$$

or

$$\boxed{\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h)-\boldsymbol{F}_{\text{ext}}=\boldsymbol{0}}\qquad\text{with}\qquad \boldsymbol{U}^h=\{\boldsymbol{u}^1,\dots,\boldsymbol{u}^n\}^{\text{T}}\tag{5.10}$$

and

$$\boxed{F^a_{\text{int},i}=\int_\Omega \sigma_{ij}(\nabla\boldsymbol{u}^h)N^a_{,j}\,\text{d}V\qquad\text{and}\qquad F^a_{\text{ext},i}=\int_\Omega \rho b_i N^a\,\text{d}V+\int_{\partial\Omega_N}\hat{t}_iN^a\,\text{d}S}\tag{5.11}$$

For the special case of **linear elasticity** we have $\sigma_{ij}=\mathbb{C}_{ijkl}u_{k,l}$ so that the *weak form* reads

$$G(\boldsymbol{u},\boldsymbol{v})=\mathcal{B}(\boldsymbol{u},\boldsymbol{v})-\mathcal{L}(\boldsymbol{v})=0\qquad\text{for all adm. }\boldsymbol{v}\tag{5.12}$$

with

$$\mathcal{B}(\boldsymbol{u},\boldsymbol{v})=\int_\Omega v_{i,j}\mathbb{C}_{ijkl}u_{k,l}\,\text{d}V\qquad\text{and}\qquad\mathcal{L}(\boldsymbol{v})=\int_\Omega\rho b_iv_i\,\text{d}V+\int_{\partial\Omega_N}\hat{t}_iv_i\,\text{d}S,\tag{5.13}$$

so $\mathcal{B}(\cdot,\cdot)$ is indeed a bilinear form. Inserting the approximate fields, (5.11) becomes

$$\begin{aligned}F^a_{\text{int},i}&=\int_\Omega\mathbb{C}_{ijkl}u^h_{k,l}N^a_{,j}\,\text{d}V=\sum_{b=1}^n\int_\Omega\mathbb{C}_{ijkl}u^b_kN^b_{,l}N^a_{,j}\,\text{d}V=\sum_{b=1}^n u^b_k\int_\Omega\mathbb{C}_{ijkl}N^a_{,j}N^b_{,l}\,\text{d}V\\&=\sum_{b=1}^nK^{ab}_{ik}u^b_k\qquad\text{with}\qquad K^{ab}_{ik}=\int_\Omega\mathbb{C}_{ijkl}N^a_{,j}N^b_{,l}\,\text{d}V\\&\Rightarrow\qquad\boldsymbol{F}_{\text{int}}=\boldsymbol{K}\,\boldsymbol{U}^h\qquad\Rightarrow\qquad\boldsymbol{U}^h=\boldsymbol{K}^{-1}\boldsymbol{F}_{\text{ext}}\quad\text{if }\det\boldsymbol{K}\neq0.\end{aligned}\tag{5.14}$$

That is, we arrive at a linear problem to be solved for the unknown coefficients $\boldsymbol{U}^h=\{\boldsymbol{u}^1,\dots,\boldsymbol{u}^n\}$.

For computational purposes, notice that vectors $\boldsymbol{U}^h$ and (internal or extrenal) $\boldsymbol{F}$, e.g., in 3D are, respectively

$$\boldsymbol{U}^h=\begin{pmatrix}\boldsymbol{u}^1\\\dots\\\boldsymbol{u}^n\end{pmatrix}=\begin{pmatrix}u^1_1\\u^1_2\\u^1_3\\\dots\\u^n_1\\u^n_2\\u^n_3\end{pmatrix},\qquad\boldsymbol{F}=\begin{pmatrix}\boldsymbol{F}^1\\\dots\\\boldsymbol{F}^n\end{pmatrix}=\begin{pmatrix}F^1_1\\F^1_2\\F^1_3\\\dots\\F^n_1\\F^n_2\\F^n_3\end{pmatrix}.\tag{5.15}$$

If we use 0-index notation like in C++ (i.e., we sum over $a=0,\dots,n-1$ instead of $a=1,\dots,n$), then

$$u^a_i\text{ is the }(d\cdot a+i)th\text{ component of vector }\boldsymbol{U}^h\text{ in }d\text{ dimensions.}\tag{5.16}$$

Similarly, we apply the same rule to the rows and columns of matrix $\boldsymbol{K}$, so that

$$K^{ab}_{ik}\text{ is the component at }(d\cdot a+i,d\cdot b+k)\text{ of matrix }\boldsymbol{K}\text{ in }d\text{ dimensions.}\tag{5.17}$$

There is a shortcut to computing the internal force vector via the **Rayleigh-Ritz** method. As introduced in Section 3.3, this technique inserts $\boldsymbol{u}^h$ directly into the total potential energy and minimizes the latter with respect to the unknown coefficients, i.e., we must solve

$$\frac{\partial I[\boldsymbol{u}^h]}{\partial\boldsymbol{u}^a}=\boldsymbol{0}\qquad\forall\quad a=1,\dots,n.\tag{5.18}$$

Note that

$$\begin{aligned}\frac{\partial I[\boldsymbol{u}^h]}{\partial u_i^a} = 0 &= \frac{\partial}{\partial u_i^a}\left[\int_\Omega W(\boldsymbol{\varepsilon}^h)\,\mathrm{d}V - \int_\Omega \rho \boldsymbol{b}\cdot\boldsymbol{u}^h\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{\boldsymbol{t}}\cdot\boldsymbol{u}^h\,\mathrm{d}S\right]\\ &= \int_\Omega \frac{\partial W}{\partial \varepsilon_{kl}}(\boldsymbol{\varepsilon}^h)\frac{\partial \varepsilon_{kl}^h}{\partial u_i^a}\,\mathrm{d}V - \int_\Omega \rho b_k \frac{\partial}{\partial u_i^a}\sum_{b=1}^n u_k^b N^b\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{t}_k\frac{\partial}{\partial u_i^a}\sum_{b=1}^n u_k^b N^b\,\mathrm{d}S.\end{aligned} \tag{5.19}$$

where

$$\varepsilon_{kl}^h = \frac{1}{2}(u_{k,l}^h + u_{l,k}^h) = \sum_{b=1}^n \frac{1}{2}(u_k^b N_{,l}^b + u_l^b N_{,k}^b) \qquad \Rightarrow \qquad \frac{\partial \varepsilon_{kl}^h}{\partial u_i^a} = \frac{1}{2}(\delta_{ik}N_{,l}^a + N_{,k}^a\delta_{li}). \tag{5.20}$$

This is equivalent to

$$\begin{aligned}0 &= \int_\Omega \sigma_{kl}(\boldsymbol{\varepsilon}^h)\frac{1}{2}(N_{,l}^a\delta_{ik} + N_{,k}^a\delta_{li})\,\mathrm{d}V - \int_\Omega \rho b_i N^a\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{t}_i N^a\,\mathrm{d}S\\ &= \int_\Omega \sigma_{il}(\boldsymbol{\varepsilon}^h)N_{,l}^a\,\mathrm{d}V - \int_\Omega \rho b_i N^a\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{t}_i N^a\,\mathrm{d}S.\end{aligned} \tag{5.21}$$

By comparison, we see immediately that this yields $F_{\text{int},i}^a$ and $F_{\text{ext},i}^a$ directly. Thus, rather than resorting to variations, we can obtain the internal and external force vectors alternatively (and oftentimes much more simply) by the Rayleigh-Ritz approach, which inserts the approximation into the potential energy and then differentiates with respect to the unknown coefficients.

Also, notice that $\boldsymbol{F}_{\text{ext}}$ is independent of the constitutive law and only depends on the applied body forces and surface tractions. That is, $\boldsymbol{F}_{\text{ext}}$ is sufficiently general for arbitrary materials (in linearized kinematics), while the computation of $\boldsymbol{F}_{\text{int}}$ depends on the particular material model.

5.2 Finite kinematics

The variational problem in finite-deformation quasistatics is quite similar:

$$I[\boldsymbol{\varphi}] = \int_\Omega W(\boldsymbol{F})\,\mathrm{d}V - \int_\Omega \rho_0 \boldsymbol{B}\cdot\boldsymbol{\varphi}\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{\boldsymbol{T}}\cdot\boldsymbol{\varphi}\,\mathrm{d}S \tag{5.22}$$

and we seek solutions

$$\boldsymbol{\varphi} \in \mathcal{U} = \left\{\boldsymbol{\varphi}\in H^1(\Omega)\ :\ \boldsymbol{\varphi} = \hat{\boldsymbol{\varphi}} \text{ on } \partial\Omega_D\right\} \qquad \text{such that} \qquad \boldsymbol{\varphi} = \arg\min I[\boldsymbol{\varphi}]. \tag{5.23}$$

In all our problems, we will assume that the undeformed and deformed coordinate systems coincide so that we write for convenience $\boldsymbol{\varphi} = \boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}$ and we thus formulate the above problem in terms of the displacement field $\boldsymbol{u}\in\mathcal{U}$, like in the linear elastic case (note that this is a "notational crime" that we adopt here for convenience).

We may then write $\boldsymbol{F} = \boldsymbol{I} + \operatorname{Grad} u$ and compute the first variation as

$$\begin{aligned}\delta I[u] &= \int_\Omega \frac{\partial W}{\partial F_{iJ}}\delta u_{i,J}\,\mathrm{d}V - \int_\Omega \rho_0 B_i \delta u_i\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{T}_i\delta u_i\,\mathrm{d}S\\ &= \int_\Omega P_{iJ}\delta u_{i,J}\,\mathrm{d}V - \int_\Omega \rho_0 B_i\delta u_i\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{T}_i\delta u_i\,\mathrm{d}S = 0,\end{aligned} \tag{5.24}$$

where we used the first Piola-Kirchhoff stress tensor $P_{iJ} = \partial W/\partial F_{iJ}$ (which is *not* symmetric).

Even though the form looks similar to (5.4), recall that $\boldsymbol{P}(\boldsymbol{F})$ involves in a generally nonlinear relation between $\boldsymbol{P}$ and the displacement gradient $\operatorname{Grad}\boldsymbol{u}$. Therefore, the finite-deformation variational problem does *not* involve a bi*linear* form (even if the material is elastic).

As before, we produce a discrete approximation, e.g., with the **Bubnov-Galerkin** approximation

$$\boldsymbol{u}^h(\boldsymbol{X}) = \sum_{a=1}^{n} \boldsymbol{u}^a N^a(\boldsymbol{X}) \qquad \text{and} \qquad \boldsymbol{v}^h(\boldsymbol{X}) = \sum_{a=1}^{n} \boldsymbol{v}^a N^a(\boldsymbol{X}), \tag{5.25}$$

so that we again arrive at

$$\boxed{\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}} = \boldsymbol{0}} \tag{5.26}$$

where now, by comparison,

$$\boxed{F^a_{\text{int},i} = \int_\Omega P_{iJ}(\nabla \boldsymbol{u}) N^a_{,J} \,\mathrm{d}V \qquad \text{and} \qquad F^a_{\text{ext},i} = \int_\Omega \rho_0 B_i N^a \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{T}_i N^a \,\mathrm{d}S} \tag{5.27}$$

In a nutshell, the linearized and finite elastic variational problems result in the same system of equations (5.26). For the special case of linear elasticity, that system is linear. Otherwise the problem is nonlinear and requires an iterative solution method.

Note that in both formulations of linearized and finite kinematics we assumed that $\hat{\boldsymbol{T}} = \text{const.}$, i.e., that the *externally applied forces are constant* and do not depend on deformation. Especially in finite deformations this is oftentimes not the case; e.g., consider pressure loading $\hat{T} = p\boldsymbol{n}$ where $\boldsymbol{n}$ is the *deformed* surface normal and $\boldsymbol{n}\,\mathrm{d}s = J\boldsymbol{F}^{-\mathrm{T}}\boldsymbol{N}\,\mathrm{d}S$ by the Piola transform. In such cases, the above variational form does not apply and one needs to revise the external force terms appropriately. For example, for pressure loading we know the work done by pressure is pv with the deformed volume v, so we may use $I_{\text{ext}} = \int_\Omega pJ\,\mathrm{d}V$, which must replace the traction term in the above total potential energy (and the derivatives follow analogously).

5.3 Thermal problem revisited

For completeness, let us revisit the quasistatic thermal problem in an analogous fashion: we seek temperature values $\boldsymbol{T}^h = \{T_1, \ldots, T_n\}$ such that

$$\boxed{\boldsymbol{Q}_{\text{int}}(\boldsymbol{T}^h) - \boldsymbol{Q}_{\text{ext}} = \boldsymbol{0}} \tag{5.28}$$

with

$$\boxed{Q^a_{\text{int}} = \int_\Omega q_i(\nabla T) N^a_{,i} \,\mathrm{d}V \qquad \text{and} \qquad Q^a_{\text{ext}} = \int_\Omega \rho s N^a \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{q}\, N^a \,\mathrm{d}S} \tag{5.29}$$

where $q_i = \partial W / \partial T_{,i}$.

For *linear heat conduction*, $\boldsymbol{q} = \kappa \,\mathrm{grad}\, T$, we obtain a linear system of equations since then

$$Q^a_{\text{int}} = \sum_{b=1}^{n} T^b \int_\Omega \kappa\, N^a_{,i} N^b_{,i} \,\mathrm{d}V = \sum_{b=1}^{n} K^{ab} T^b \qquad \text{with} \qquad K^{ab} = \int_\Omega \kappa\, N^a_{,i} N^b_{,i} \,\mathrm{d}V. \tag{5.30}$$

(Notice that for convenience we defined the flux vector $\boldsymbol{q}$ without the negative sign, which results in the analogous forms as in linear elasticity.)

5.4 A simple example: nonlinear springs

Consider an axial spring that undergoes large deformation, i.e., its two end points move from $(\boldsymbol{X}^0, \boldsymbol{X}^1)$ to $(\boldsymbol{x}^0, \boldsymbol{x}^1)$ and $\boldsymbol{x}^i = \boldsymbol{X}^i + \boldsymbol{u}^i$. (Note that we use 0 an 1 instead of 1 and 2 to comply with C++ standard indexing which starts with 0.)

The bar stores strain energy upon stretching with an energy density $W = W(\varepsilon)$ where ε is the axial bar strain:

$$\varepsilon = \frac{l-L}{L}, \qquad \text{where} \qquad \boldsymbol{l} = \boldsymbol{x}^1 - \boldsymbol{x}^0, \quad \boldsymbol{L} = \boldsymbol{X}^1 - \boldsymbol{X}^0, \quad \text{and} \quad l = |\boldsymbol{l}|, \quad L = |\boldsymbol{L}|. \tag{5.31}$$

The total energy of a bar with cross-sectional area A and initial length L is therefore

$$I = A\,L\,W(\varepsilon). \tag{5.32}$$

Without even defining interpolation functions, we can use the Rayleigh-Ritz shortcut to calculate the resulting *internal force* on node 0 as

$$\begin{aligned}\boldsymbol{F}^0_{\text{int}} &= \frac{\partial I}{\partial \boldsymbol{u}^0} = A\,L\frac{\partial W}{\partial \varepsilon}\frac{\partial}{\partial \boldsymbol{u}^0}\frac{l-L}{L} \\ &= A\,\sigma(\varepsilon)\frac{\partial}{\partial \boldsymbol{u}^0}\sqrt{(\boldsymbol{X}^1 + \boldsymbol{u}^1 - \boldsymbol{X}^0 - \boldsymbol{u}^0)\cdot(\boldsymbol{X}^1 + \boldsymbol{u}^1 - \boldsymbol{X}^0 - \boldsymbol{u}^0)} \\ &= -A\,\sigma(\varepsilon)\frac{\boldsymbol{l}}{l},\end{aligned} \tag{5.33}$$

where $\sigma(\varepsilon) = \partial W/\partial\varepsilon$ is the axial stress in the bar. Analogously, the force on node 1 becomes

$$\boldsymbol{F}^1_{\text{int}} = -\boldsymbol{F}^0_{\text{int}} = A\,\sigma(\varepsilon)\frac{\boldsymbol{l}}{l}. \tag{5.34}$$

As expected, the force points along the (deformed) axis of the spring end points, and the forces on the two end points are of the same magnitude but of opposite sign.

Note that we did not specify whether or not the spring is linear elastic; i.e., the specific choice of $W(\varepsilon)$ will determine the behavior of the spring and can be chosen to be quadratic, i.e. $W(\varepsilon) = \frac{k}{2}\varepsilon^2$, which results in a linear spring, but can also be more complex.

As a final remark, the assumption of a constant strain ε along the spring length tacitly implies that we assume a linear displacement profile along the spring. That is, the above formulation is equivalent to assuming linear shape functions N^0 and N^1 so that

$$\boldsymbol{u}^h(\boldsymbol{X}) = \boldsymbol{u}^0 N^0(\boldsymbol{X}) + \boldsymbol{u}^1 N^1(\boldsymbol{X}). \tag{5.35}$$

In our implementation, we will define two classes – one *material model* that defines the material point relations $W = W(\varepsilon)$ and $\sigma = \sigma(\varepsilon)$ as well as an nonlinear bar/spring *element* that defines I and $\{\boldsymbol{F}^0_{\text{int}}, \boldsymbol{F}^1_{\text{int}}\}$. Notice that the element will require the material model to compute its required quantities.

6 Interpolation spaces

So far, we have assumed approximations of the type

$$u^h(x) = \sum_{a=1}^{n} u^a N^a(x), \tag{6.1}$$

but we have not chosen particular spaces $\mathcal{U}^h$ for the interpolation or **shape functions** $N^a(x)$.

In general, there are two possible choices:

- **global** shape functions that are defined everywhere in Ω, i.e., $|\operatorname{supp} N^a| \sim |\Omega|$, e.g., polynomials $N^a(x) = x^{a-1}$ or trigonometric polynomials $N^a(x) = \cos\big(\pi(a-1)x\big)$.
- **local** shape functions that are defined only locally: $|\operatorname{supp} N^a| \ll |\Omega|$, e.g., picewise linear shape functions,

where we introduced the **support** of a continuous function u defined on $\Omega \in \mathbb{R}^d$ as the (closure in Ω of the) set of all points where $u(x) \neq 0$, i.e.,

$$\operatorname{supp} u = \big\{x \in \Omega \;:\; u(x) \neq 0\big\} \tag{6.2}$$

This means that $u(x) = 0$ for $x \in \Omega \setminus \operatorname{supp} u$.

For any set of shape functions, the following **shape function properties** must be satisfied:

(1) for any $x \in \Omega$ there is at least one a with $1 \leq a \leq n$ such that $N^a(x) \neq 0$ (i.e., the *whole domain must be covered*; otherwise, there is no approximation at all at certain points)

(2) all N^a should allow to satisfy the *Dirichlet boundary conditions* if required.

(3) *linear independence* of the shape functions:

$$\sum_{a=1}^{n} u^a N^a = 0 \qquad \Leftrightarrow \qquad u^a = 0 \text{ for all } a = 1, \ldots, n. \tag{6.3}$$

In other words, given any function $u^h \in \mathcal{U}^h$, there exists a unique set of parameters $\{u^1, \ldots, u^n\}$ such that

$$u^h = \sum_{a=1}^{n} u^a N^a. \tag{6.4}$$

Then, functions $\{N^1, \ldots, N^n\}$ are a **basis** of $\mathcal{U}^h$. Linear independence is important since it avoids ill-posed problems.

For example, take $\mathcal{U}^h = \mathbb{P}_2$ and $\{N^1, N^2, N^3\} = \{1, x, x^2\}$ so that $u^h = u^1N^1 + u^2N^2 + u^3N^3$. Hence, if, e.g., $u^h = a + bx + x^2$ then we immediately conclude that $u^1 = a$, $u^2 = b$, $u^3 = c$ uniquely.

Otherwise, i.e., if there existed a set $\{\alpha^1, \ldots, \alpha^n\} \neq 0$ such that $\sum_{a=1}^{n} \alpha^a N^a = 0$, then this set of parameters could be added on top of any solution $\{u^1, \ldots, u^n\}$ such that

$$u^h = \sum_{a=1}^{n} u^a N^a = \sum_{a=1}^{n} (u^a + \alpha^a) N^a, \tag{6.5}$$

which means both $\{u^a\}$ and $\{u^a + \alpha^a\}$ are solutions (hence the problem is not well-posed).

(4) The shape functions N^a must satisfy the *differentiability/integrability requirements* of the weak form (this depends on the problem to be solved and will be discussed later).

(5) The shape functions must possess "*sufficient approximation power*". In other words, consider $u^h \in \mathcal{U}^h \subset \mathcal{U}$: we should ensure that $u^h = \sum_{a=1}^{n} u^a N^a \to u$ as $n \to \infty$.

Condition (5) is a crucial one. It tells us that for an approximation to converge, we must pick an approximate function space that gives the solution "a chance to converge". For example, assume you aim to approximate a high-order polynomial $u \in \mathbb{P}_n$ (with $n \gg 1$) by an approximation u^h using shape functions $\{1, x, x^2, x^3, \ldots, x^n\}$. This is expected to converge as $n \to \infty$, because the coefficients of u will approach the coefficients of u^h. But choosing shape functions $\{1, x, x^3, \ldots, x^n\}$ (notice the x^2-term is omitted) will never converge as $n \to \infty$. Polynomials do satisfy this requirement by the following theorem.

Weierstrass approximation theorem: Given a continuous function $f : [a,b] \subset \mathbb{R} \to \mathbb{R}$ and any scalar $\epsilon > 0$, then there exists a polynomial

$$p_n(x) \in \mathbb{P}_\infty \qquad \text{such that} \qquad |f(x) - p_n(x)| < \epsilon \quad \text{for all } x \in [a,b]. \tag{6.6}$$

This means every continuous function u can be approximated by a polynomial function to within any level of accuracy.

Therefore, $\{N^i\} = \{1, x, x^2, x^3, \ldots\}$, i.e., the polynomials in $\mathbb{R}$, is a suitable choice for the shape functions that satisfy the **completeness property** (and we have shown their linear independence).

Note that, as discussed above, one can*not* omit any intermediate-order terms from the set

$$\{1, x, x^2, x^3, \ldots\}. \tag{6.7}$$

If one omits a term, e.g., take $\{1, x^2, x^3, \ldots\}$, then if $u^h \in \mathcal{U}^h = \mathbb{P}_n$ then there is no set $\{u^1, \ldots, u^n\}$ such that $u^h = \sum_{i=1}^{n} u^i N^i$.

As an extension, the Weierstrass approximation theorem also applies to **trigonometric polynomials** (cf. Fourier series).

completeness in higher dimensions:

A polynomial approximation in $\mathbb{R}^d$ is **complete up to order q**, if it contains independently all monomials $\boldsymbol{x}^\alpha$ with $|\alpha| = \alpha_1 + \ldots + \alpha_d \le q$, i.e., using multi-indices we write

$$u^h = \sum_{\beta=0}^{q} \sum_{|\alpha|=\beta} c_\alpha \boldsymbol{x}^\alpha. \tag{6.8}$$

What does this mean in practice?

1D: $\{1, x, x^2, x^3, \ldots, x^q\}$ so that a polynomial of order q contains $q+1$ monomials

2D: $q = 0$: $\{1\}$
$q = 1$: $\{1, x_1, x_2\}$
$q = 2$: $\{1, x_1, x_2, x_1^2, x_1 x_2, x_2^2\}$
$q = 3$: $\{1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, x_1^2 x_2, x_1 x_2^2, x_2^3\}$
...

The number of independent monomials in 2D is hence $(q+1)(q+2)/2$.

7 The Finite Element Method

Motivation: we would like to define shape functions that are *local* and admit a simple way to enforce *Dirichlet BCs*.

Idea: we introduce a discretization $\mathcal{T}_h$ that splits Ω into subdomains Ω_e, the so-called **elements**, such that

$$\Omega_e \subset \Omega, \qquad \Omega = \bigcup_e \Omega_e, \qquad \partial\Omega \subseteq \bigcup_e \partial\Omega_e. \tag{7.1}$$

$\mathcal{T}_h$ is defined by the collection of **nodes** and **elements** and is called a **mesh**.

Mathematically (and computationally), a **finite element** is an object that has

(i) a FE subdomian Ω_e.
(ii) a (linear) space of shape functions N^i (restricted to Ω_e, i.e. $\operatorname{supp} N^i = \overline{\Omega}_e$).
(iii) a set of **degrees of freedom** (dofs), viz. the u^a associated with those N^i.

The **Finite Element Method** (FEM) defines continuous, piecewise-polynomial shape functions such that

$$\boxed{N^i(x_j) = \delta_{ij} \qquad \text{for all } i,j \in \{1,\dots,n\}} \tag{7.2}$$

This is the defining relation that determines the shape functions. Notice that if we evaluate the approximation $u^h(x)$ at one of the nodes x_j, then

$$u^h(x_j) = \sum_{a=1}^n u^a N^a(x_j) = \sum_{a=1}^n u^a \delta_{aj} = u^j. \tag{7.3}$$

That is, the coefficient u^j can now be identified as the value of approximate function u^h at node j. This makes for a very beneficial physical interpretation of the (yet to be determined) shape function coefficients.

Let us check the *requirements for shape functions*:

(1) is automatically satisfied: if $x \in \Omega$ then $x \in \Omega_e$ for some e, then there are $N^i(x) \neq 0$
(2) can be satisfied by fixing degrees of freedom of the boundary nodes (*errors possible*)
(3) Assume, by contradiction, that $u^h(x) = 0$ for all $x \in \Omega$ while some $u^a \neq 0$. Now, evaluate at a node x_j:

$$0 = u^h(x_j) = \sum_{a=1}^n u^a N^a(x_j) = u^j \qquad \Rightarrow \qquad u^j = 0, \tag{7.4}$$

which contradicts the assumption that some $u_a^h \neq 0$. Thus, we have *linear independence*.
(4) *Integrability/differentiability* requirements depend on the variational problem to be solved and must be ensured. For example, for mechanics we have $\mathcal{U}^h, \mathcal{V}^h \in H^1$, i.e., first derivatives must be square-integrable. Note that this guarantees that displacements (0th derivatives) are continuous and thus *compatible* (no jumps in displacements).
(5) *Completeness* requires $u^h \to u$ (and thus $\mathcal{U}^h \to \mathcal{U}$) to within desirable accuracy. In the FE method, one enriches $\mathcal{U}^h$ by, e.g.,

- **h-refinement**: refining the discretization $\mathcal{T}_h$ while keeping the polynomial order fixed.
- **p-refinement**: increasing the polynomial interpolation order within a fixed discretization $\mathcal{T}_h$.

- **hp-refinement**: combination of the two above.
- **r-refinement**: repositioning of nodes while keeping discretization/interpolation fixed.

A note on ensuring *sufficient approximation power*: consider the exact solution $u(x)$ at a point $x \in \Omega$ so

$$u(x+h) = u(x) + h\,u'(x) + \frac{1}{2}h^2 u''(x) + \ldots + \frac{1}{q!}h^q u^{(q)}(x) + O(h^{q+1}) \tag{7.5}$$

Assume that $\mathcal{U}^h$ contains all polynomials complete up to degree q (i.e., $u^h \in \mathbb{P}_q$), then there exists

$$u^h \in \mathcal{U}^h \qquad \text{such that} \qquad u(x) = u^h(x) + O(h^{q+1}). \tag{7.6}$$

Let p be the highest derivative in the weak form, then

$$\frac{\mathrm{d}^p u}{\mathrm{d}x^p} = \frac{\mathrm{d}^p u^h}{\mathrm{d}x^p} + O(h^{q+1-p}). \tag{7.7}$$

For the solution to converge as $h \to 0$ we need $q + 1 - p \geq 1$ so that we have at least order $O(h)$. Thus we must ensure that

$$\boxed{q \geq p} \tag{7.8}$$

8 Finite element spaces: polynomial shape functions in 1D

Let us start with the simplest of all choices: *continuous, piecewise-polynomial interpolation functions*. Note that we need $q \geq 1$ since $p = 1$ for the mechanical/thermal/electromagnetic variational problem; i.e., we need *at least linear interpolation* within elements.

8.1 One dimension

Simplest example: 2-node bar element

Interpolation with element dofs $\{u_e^1, u_e^2\}$ so that

$$u_e^h(x) = N_e^1(x) u_e^1 + N_e^2(x) u_e^2 \tag{8.1}$$

and we must have $u_e^h(0) = u_e^1$ and $u_e^h(\Delta x) = u_e^2$.

This gives the **element shape functions**:

$$N_e^1(x) = 1 - \frac{x}{\Delta x}, \qquad N_e^2(x) = \frac{x}{\Delta x}. \tag{8.2}$$

Note that the interpolation space uses $\{1, x\}$ which is complete up to $q = 1$.

8.2 Example: linear elastic bar in 1D

Recall that for a linear bar element we obtained

$$N_e^1(x) = 1 - \frac{x}{\Delta x}, \quad N_e^2(x) = \frac{x}{\Delta x} \qquad \Rightarrow \qquad N_{e,x}^1(x) = -\frac{1}{\Delta x}, \quad N_{e,x}^2(x) = \frac{1}{\Delta x}. \tag{8.3}$$

so that the (only non-zero) axial strain inside the bar element is *constant* since:

$$\varepsilon_{xx}^h = u_{,x}^h(x) = \sum_{i=1}^n u_e^i \, N_{e,x}^i(x) = u_e^1 N_{e,x}^1(x) + u_e^2 N_{e,x}^2(x) = \frac{u_e^2 - u_e^1}{\Delta x} = \text{const}. \tag{8.4}$$

If we assume a linear elastic bar, so $\sigma = E\,\varepsilon$, we obtain from (5.11)

$$F_{\text{int}}^a = \int_\Omega \sigma_{xx}(\varepsilon_{xx}^h) N_{,x}^a \, \mathrm{d}V = \int_0^{\Delta x} E \frac{u_e^2 - u_e^1}{\Delta x} N_{,x}^a A \, \mathrm{d}x = \frac{EA}{\Delta x}(u_e^2 - u_e^1) \int_0^{\Delta x} N_{,x}^a \, \mathrm{d}x, \tag{8.5}$$

and inserting the constant shape function derivatives finally yields

$$F_{\text{int}}^1 = \frac{EA}{\Delta x}(u_e^2 - u_e^1)\left(-\frac{1}{\Delta x}\right)\Delta x = -\frac{EA}{\Delta x}(u_e^2 - u_e^1) \qquad \text{and} \qquad F_{\text{int}}^2 = -F_{\text{int}}^1. \tag{8.6}$$

The stiffness matrix follows as

$$\left[\frac{\partial F_{\text{int}}^i}{\partial u_e^j}\right] = \frac{EA}{\Delta x}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \tag{8.7}$$

which has the typical form known from 1D assemblies of linear springs with stiffness $k = EA/\Delta x$.

Lagrangian interpolation of higher order

Interpolation up to degree q, i.e., $\{1, x, x^2, \ldots, x^q\}$ so that

$$u_e^h(x) = \sum_{a=1}^{q+1} N_e^a(x) u_e^a = a_0 + a_1 x + a_2 x^2 + \ldots + a_q x^q. \tag{8.8}$$

In general, shape functions can be determined by solving the $q+1$ equations

$$u_e^h(x_i) = u^i \qquad \text{for all} \qquad i = 1, \ldots, q+1 \tag{8.9}$$

for the $q+1$ coefficients a_i ($i = 0, \ldots, q$). Then, rearranging the resulting polynomial allows to extract the shape functions $N_e^a(x)$ by comparison of the coefficients of u_e^a.

Alternatively, we can solve

$$N_e^a(x_i) = \delta_{ai} \qquad \text{for all nodes } i = 1, \ldots, q+1. \tag{8.10}$$

The solution is quite intuitive:

$$\boxed{N_e^a(x) = \frac{(x - x_1) \cdot \ldots \cdot (x - x_{a-1}) \cdot (x - x_{a+1}) \cdot \ldots \cdot (x - x_{q+1})}{(x_a - x_1) \cdot \ldots \cdot (x_a - x_{a-1}) \cdot (x_a - x_{a+1}) \cdot \ldots \cdot (x_a - x_{q+1})}} \tag{8.11}$$

One can readily verify that $N_e^a(x_i) = \delta_{ai}$. These are called **Lagrange polynomials**.

Alternative: hierarchical interpolation

We can also construct higher-order interpolations based on lower-order shape functions. For example, start with a 2-node bar:

$$N_e^1(x) = 1 - \frac{x}{\Delta x}, \qquad N_e^2(x) = \frac{x}{\Delta x}. \tag{8.12}$$

Let us enrich the interpolation to reach $q = 2$:

$$u^h(x) = N_e^1(x) u_e^1 + N_e^2(x) u_e^2 + \widetilde{N}_e^3(x) \alpha_e \tag{8.13}$$

with

$$\widetilde{N}_e^3(x) = a_0 + a_1 x + a_2 x^2. \tag{8.14}$$

We need to find the coefficients a_i. Note that we must have

$$\widetilde{N}_e^3(0) = \widetilde{N}_e^3(\Delta x) = 0 \qquad \Rightarrow \qquad \widetilde{N}_e^3(x) = c \frac{x}{\Delta x}\left(1 - \frac{x}{\Delta x}\right) \tag{8.15}$$

with some constant $c \neq 0$.

Note that α_e does not have to be continuous across elements and can hence be determined *locally* (i.e., given u_e^1 and u_e^2, α_e can be determined internally for each element, which allows for *condensation* of the α_e-dof).

Example of higher-order interpolation: 2-node beam element

Linear elastic Euler-Bernoulli beams are a most common structural element. From the variational form (or the strong form, $EIw^{(4)}(x) = q(x)$ for statics, which is of 4th order in the deflection $w(x)$) we know that $p = 2$. Therefore, we must have $q \geq 2$, and $w \in H^2(\Omega)$.

The simplest admissible interpolation is based on $\{1, x, x^2, x^3\}$, so

$$w_e^h(x) = c_0 + c_1 x + c_2 x^2 + c_3 x^3. \tag{8.16}$$

We need *four dofs*, so we pick *two nodes* and assign to each node a deflection w and angle $\theta = w'$:

$$w_e^h(x) = \sum_{i=1}^{2} \left[N_e^{i1}(x) w_e^i + N_e^{i2}(x) \theta_e^i \right] \tag{8.17}$$

and we must have

$$w_e^h(0) = w_e^1, \qquad w_e^h(\Delta x) = w_e^1, \qquad (w_e^h)'(0) = \theta_e^1, \qquad (w_e^h)'(\Delta x) = \theta_e^1. \tag{8.18}$$

The resulting shape functions are known as **Hermitian polynomials** (and can be found in textbooks and the notes).

Note that this is only one possible choice; we could also define alternative nodes and nodal values. However, the above choice ensures that *both deflection and angle are continuous* across elements.

8.3 Higher dimensions

Problem: in higher dimensions it will be cumbersome to define polynomial shape functions on the actual shape of the element, unless one uses regular structured meshes (e.g, grids). In general, all elements have different shapes and it is beneficial to define shape functions independent of the specific element shape.

To this end, we introduce a (bijective) **isoparametric mapping** ϕ from a *reference domain* (with reference coordinates $\boldsymbol{\xi} = \{\xi, \eta, \zeta\}$) onto the *physical domain* (with coordinates $\boldsymbol{x} = \{x, y, z\}$) of an element e:

$$\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{\xi}), \qquad \text{i.e. in 3D:} \quad x = x(\xi, \eta, \zeta), \quad y = y(\xi, \eta, \zeta), \quad z = z(\xi, \eta, \zeta). \tag{8.19}$$

For simplicity, we reuse the interpolation concepts from before:

$$\begin{aligned} &\text{so far:} \quad \boldsymbol{u}_e^h = \sum_{i=1}^{n} N_e^i(\boldsymbol{x}) \boldsymbol{u}_e^i \\ &\text{now:} \quad \boldsymbol{u}_e^h = \sum_{i=1}^{n} N_e^i(\boldsymbol{\xi}) \boldsymbol{u}_e^i \qquad \text{and} \qquad \boldsymbol{x} = \sum_{i=1}^{m} \tilde{N}_e^i(\boldsymbol{\xi}) \boldsymbol{x}_e^i, \end{aligned} \tag{8.20}$$

where we have three options for the mapping:

(i) **isoparametric**: $n = m$ and $N_e^i = \tilde{N}_e^i$ (same interpolation)
(ii) **subparametric**: $n > m$ (lower-order interpolation of positions)
(iii) **superparametric**: $n < m$ (higher-order interpolation of positions)

The strategy is now to define $N_e^i(\boldsymbol{\xi})$ in the reference configuration. Let us first discuss this concept for simplicial elements in the next section, before proceeding to more general classes of polynomial element interpolations.

9 Simplicial elements

A **simplex** of order k is a k-dimensional polytope which is the convex hull of its $k+1$ vertices.

In plain English, a simplex of order k is a convex body made up of $k+1$ nodes:

- in 1D: a 2-node bar, interpolation basis is $\{1, x\}$
- in 2D: a 3-node triangle (T3), interpolation basis is $\{1, x, y\}$
- in 3D: a 4-node tetrahedron (T4), interpolation basis is $\{1, x, y, z\}$

Overall, this shows that interpolation is of degree $q = 1$ (linear) for all simplicial elements.

One usually uses special shape functions for simplices, based on barycentric coordinates.

9.1 Linear Triangle (T3)

In 2D, the 3-node triangular element uses the **barycentric coordinates**

$$l_e^i(\boldsymbol{x}) = \frac{A_e^i(\boldsymbol{x})}{A_e}, \tag{9.1}$$

where $A_e = |\Omega_e|$ is the total triangle area, and A_e^i is the sub-area opposite from node i, so that $\sum_{i=1}^3 A_e^i = A$.

It is an easy check to see that $0 \le l_e^i \le 1$ and $\sum_{i=1}^3 l_e^i(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \Omega_e$, so the l_e^i qualify as shape functions for the T3 element.

For convenience we use $r = l_e^1$ and $s = l_e^2$ as *reference coordinates*, so that the shape functions become

$$N_e^1(r,s) = r, \qquad N_e^2(r,s) = s, \qquad N_e^3(r,s) = 1 - r - s. \tag{9.2}$$

The isoparametric mapping in 2D now means

$$x = \sum_{i=1}^3 N_e^i(r,s)\, x_e^i, \qquad y = \sum_{i=1}^3 N_e^i(r,s)\, y_e^i. \tag{9.3}$$

There is one particular **difficulty with isoparametric elements**:

To compute force vectors, etc., we need shape function derivatives $N_{,x}^i$ and $N_{,y}^i$ but the shape functions were defined as $N^i(r,s)$, so only $N_{,r}^i$ and $N_{,s}^i$ are known.

Let us use $x = x(r,s)$ and $y = y(r,s)$ so the chain rule gives

$$\begin{pmatrix} u_{,r} \\ u_{,s} \end{pmatrix} = \begin{pmatrix} u_{,x}x_{,r} + u_{,y}y_{,r} \\ u_{,x}x_{,s} + u_{,y}y_{,s} \end{pmatrix} = \begin{pmatrix} x_{,r} & y_{,r} \\ x_{,s} & y_{,s} \end{pmatrix} \begin{pmatrix} u_{,x} \\ u_{,y} \end{pmatrix} = \boldsymbol{J} \begin{pmatrix} u_{,x} \\ u_{,y} \end{pmatrix} \tag{9.4}$$

with the **Jacobian** matrix

$$\boldsymbol{J} = \begin{pmatrix} x_{,r} & y_{,r} \\ x_{,s} & y_{,s} \end{pmatrix} \tag{9.5}$$

so that

$$J = \det \boldsymbol{J} = \frac{\partial x}{\partial r}\frac{\partial y}{\partial s} - \frac{\partial x}{\partial s}\frac{\partial y}{\partial r}. \tag{9.6}$$

Note that, like for the deformation mapping, for the isoparametric mapping to be invertible we need to have $J > 0$. This implies that *elements cannot be distorted* (inverted or non-convex).

Using our isoparametric mapping, we obtain

$$\boldsymbol{J} = \begin{pmatrix} \sum_{i=1}^{3} N_{e,r}^{i} x_e^i & \sum_{i=1}^{3} N_{e,r}^{i} y_e^i \\ \sum_{i=1}^{3} N_{e,s}^{i} x_e^i & \sum_{i=1}^{3} N_{e,s}^{i} y_e^i \end{pmatrix}. \tag{9.7}$$

This solves the problem. As discussed, we need to have $J > 0$ so that we can invert $\boldsymbol{J}$ to arrive at (by the **inverse function theorem**)

$$\begin{pmatrix} u_{,x} \\ u_{,y} \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} u_{,r} \\ u_{,s} \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} \sum_{i=1}^{3} N_{e,r}^{i} u_e^i \\ \sum_{i=1}^{3} N_{e,s}^{i} u_e^i \end{pmatrix} \qquad \text{but also} \qquad \begin{pmatrix} u_{,x} \\ u_{,y} \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{3} N_{e,x}^{i} u_e^i \\ \sum_{i=1}^{3} N_{e,y}^{i} u_e^i \end{pmatrix}. \tag{9.8}$$

By equating these two and comparing the coefficients of u_e^i we thus obtain

$$\begin{pmatrix} N_{e,x}^i \\ N_{e,y}^i \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} N_{e,r}^i \\ N_{e,s}^i \end{pmatrix} \qquad \text{and more generally:} \qquad \boxed{\nabla_{\boldsymbol{x}} N_e^i = \boldsymbol{J}^{-1} \nabla_{\boldsymbol{\xi}} N_e^i} \tag{9.9}$$

with reference coordinates $\boldsymbol{\xi} = \{r, s\}$. This is generally applicable *for any isoparametric mapping.*

By inserting the above shape functions into the Jacobian matrix we find that

$$\boldsymbol{J} = \begin{pmatrix} x_{,r} & y_{,r} \\ x_{,s} & y_{,s} \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{3} N_{e,r}^{i} x_e^i & \sum_{i=1}^{3} N_{e,r}^{i} y_e^i \\ \sum_{i=1}^{3} N_{e,s}^{i} x_e^i & \sum_{i=1}^{3} N_{e,s}^{i} y_e^i \end{pmatrix} = \begin{pmatrix} x_e^1 - x_e^3 & y_e^1 - y_e^3 \\ x_e^2 - x_3^e & y_e^2 - y_e^3 \end{pmatrix} \tag{9.10}$$

and

$$J = \det \boldsymbol{J} = (x_e^1 - x_e^3)(y_e^2 - y_e^3) - (x_e^2 - x_3^e)(y_e^1 - y_e^3) = 2A_e. \tag{9.11}$$

Notice that $\boldsymbol{J}$ and hence J are constant and do not depend on (r, s).

Further, we have

$$\begin{pmatrix} N_{e,x}^i \\ N_{e,y}^i \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} N_{e,r}^i \\ N_{e,s}^i \end{pmatrix}. \tag{9.12}$$

From (12.1) we see that all shape function derivatives are constant (-1, $+1$, or 0), so that we may conclude that

$$N_{e,x}^i = \text{const.}, \qquad N_{e,y}^i = \text{const.} \qquad \text{and also} \qquad \mathrm{d}A = J\,\mathrm{d}r\,\mathrm{d}s = 2A_e\,\mathrm{d}r\,\mathrm{d}s. \tag{9.13}$$

The constant shape function derivatives indicate that *all strain components are constant* within the element since $\boldsymbol{\varepsilon} = \operatorname{sym}(\nabla \boldsymbol{u})$. This is why the 3-node triangle element is also called **Constant Strain Triangle** or **CST**. This also has the important consequence that integration of force vectors or stiffness matrices can easily be performed *exactly* by a *single quadrature point* since the integrands are constant across the element.

In case of higher-order triangular elements, the following relation is helpful for evaluating integrals:

$$\int_{\Omega_e} r^\alpha s^\beta \,\mathrm{d}A = \frac{\alpha!\,\beta!}{(\alpha + \beta + 2)!} 2A. \tag{9.14}$$

9.2 Extension to three dimensions:

The extension to three dimensions is straight-forward and results in the **4-node tetrahedron** (**constant strain tetrahedron**) with reference coordinates (r,s,t) and shape functions

$$N_e^1(r,s,t)=r,\qquad N_e^2(r,s,t)=s,\qquad N_e^3(r,s,t)=t,\qquad N_e^4(r,s,t)=1-r-s-t. \tag{9.15}$$

Like in 2D, strains are constant in the linear tetrahedron, and $\mathrm{d}V=6V\,\mathrm{d}r\,\mathrm{d}s\,\mathrm{d}t$.

Note the following important relation that is analogous to (9.14),

$$\int_{\Omega_e} r^\alpha s^\beta t^\gamma\,\mathrm{d}A=\frac{\alpha!\,\beta!\,\gamma!}{(\alpha+\beta+\gamma+2)!}2A. \tag{9.16}$$

9.3 Finite element implementation

When using the above simplicial elements, the finite element implementation can make use of the isoparametric mapping. Specifically, we showed that

$$N_{e,j}^i=J_{j\xi}^{-1}N_{e,\xi}^i=\text{const.}\qquad\text{and}\qquad J_e=\det\boldsymbol{J}=2A_e. \tag{9.17}$$

We further note that strains are constant within elements since

$$u_{i,j}^h=\sum_{a=1}^n u_i^a N_{e,j}^a=\sum_{a=1}^n u_i^a J_{j\xi}^{-1}N_{e,\xi}^a=\text{const.}\qquad\Rightarrow\qquad \sigma_{ij}=\sigma_{ij}(\nabla\boldsymbol{u}^h)=\text{const.} \tag{9.18}$$

The key integrals to be evaluated (e.g., in 2D with an element thickness t) now become

$$F_{\text{int},i}^a=\int_{\Omega_e}\sigma_{ij}N_{e,j}^a\,\mathrm{d}V=\int_{\Omega_e}\sigma_{ij}N_{e,j}^a\,t\,\mathrm{d}A=\sigma_{ij}J_{j\xi}^{-1}N_{e,\xi}^a t\int_{\Omega_e}\mathrm{d}A=\sigma_{ij}J_{j\xi}^{-1}N_{e,\xi}^a\frac{J_e t}{2}. \tag{9.19}$$

Note that if we pre-compute the constant quantities in the element constructor, e.g.,

$$w_e=\frac{J_e t}{2}=\text{const.}\qquad\text{and}\qquad N_{e,j}^a=J_{j\xi}^{-1}N_{e,\xi}^a=\text{const.}, \tag{9.20}$$

then the above reduces to

$$F_{\text{int},i}^a=\sigma_{ij}(\nabla\boldsymbol{u}^h)N_{e,j}^a w_e. \tag{9.21}$$

Similarly, we compute the element energy

$$I_e=\int_{\Omega_e}W(\nabla\boldsymbol{u}^h)\,\mathrm{d}V=\int_{\Omega_e}W(\nabla\boldsymbol{u}^h)\,t\,\mathrm{d}A=W(\nabla\boldsymbol{u}^h)w_e. \tag{9.22}$$

9.4 Higher-order triangles and tetrahedra:

One can also define higher-order triangular and tetrahedral elements.

For example, the **quadratic triangle** (T6) element has six nodes so that the interpolation function space is $\{1,x,y,x^2,xy,y^2\}$ and therefore complete up to second order. Per convention, nodes 1-3 are the corners while 4-6 are at edge midpoints, and counting is counter-clockwise as before.

With the same reference coordinates (r,s) as for the T3 element, the shape functions can be found as

$$\begin{aligned} &N_e^1(r,s) = r(2r-1), && N_e^2(r,s) = s(2s-1), && N_e^3(r,s) = t(2t-1), \\ &N_e^4(r,s) = 4rs, && N_e^5(r,s) = 4st, && N_e^6(r,s) = 4rt, \end{aligned} \tag{9.23}$$

where $t = 1 - r - s$.

Since shape function derivatives are linear, strains (and thus stresses) also vary linearly within the element which is therefore known as the **linear strain triangle** (LST). The quadratic interpolation implies that element edges of isoparametric elements can be curved.

Analogously, the **quadratic tetrahedron** (T10) has four corner nodes and six nodes at edge midpoints.

10 The bilinear quadrilateral element

The above simplicial element is simple to implement and its constant strains admit an exact integration of energy, stresses, and stiffness. However, the constant fields within CST elements may not be ideal. As an alternative in 2D, let us discuss the **4-node bilinear quadrilateral** element (also known as **Q4**).

By definition, the reference configuration is $\Omega_e = [-1,1]^2$ and node numbering starts in the bottom left corner and is counterclockwise.

Shape functions must satisfy $N_e^i(\xi_j,\eta_j) = \delta_{ij}$ and can be obtained from the 2-node bar which has:

$$\text{2-node bar:} \qquad \bar{N}_e^1(\xi) = \frac{1}{2}(1-\xi), \quad \bar{N}_e^2(\xi) = \frac{1}{2}(1+\xi). \tag{10.1}$$

It can easily be verified that $\bar{N}_e^1(-1) = 1$, $\bar{N}_e^1(1) = 0$, and $\bar{N}_e^2(-1) = 0$, $\bar{N}_e^2(1) = 1$.

The 2D element shape functions thus follow from combining the above in the ξ and η directions:

$$\begin{aligned}
\text{Q4 element:} \qquad N_e^1(\xi,\eta) &= \bar{N}_e^1(\xi)\bar{N}_e^1(\eta) = \frac{1}{4}(1-\xi)(1-\eta),\\
N_e^2(\xi,\eta) &= \bar{N}_e^2(\xi)\bar{N}_e^1(\eta) = \frac{1}{4}(1+\xi)(1-\eta),\\
N_e^3(\xi,\eta) &= \bar{N}_e^2(\xi)\bar{N}_e^2(\eta) = \frac{1}{4}(1+\xi)(1+\eta),\\
N_e^4(\xi,\eta) &= \bar{N}_e^1(\xi)\bar{N}_e^2(\eta) = \frac{1}{4}(1-\xi)(1+\eta).
\end{aligned} \tag{10.2}$$

One can easily verify that $N_e^i(\xi_j,\eta_j) = \delta_{ij}$ and $\sum_{i=1}^4 N_e^i(\xi,\eta) = 1$ for all (ξ,η).

The isoparametric mapping here implies

$$x = \sum_{i=1}^4 N_e^i(\xi,\eta)\, x_e^i, \qquad y = \sum_{i=1}^4 N_e^i(\xi,\eta)\, y_e^i. \tag{10.3}$$

Notice that this implies *straight edges* (in the reference configuration) *remain straight* (in the actual mesh). For example, take the bottom edge ($\eta = -1$), the interpolation along this edge is $x = \sum_{i=1}^4 N_e^i(\xi,-1)x_e^i$ and $y = \sum_{i=1}^4 N_e^i(\xi,-1)y_e^i$, which are both linear in ξ. Therefore, this element has straight edges in physical space.

Remarks:

- *completeness* up to only $q = 1$ is given by $\{1,\xi,\eta,\xi\eta\}$. This means we must be able to represent solutions

  $$u^h(x,y) = c_0 + c_1 x + c_2 y \tag{10.4}$$

  exactly. Check:

  $$\begin{aligned}
  u^h(x,y) &= \sum_{i=1}^4 N_e^i u_e^i = \sum_{i=1}^4 N_e^i u^h(x_i,y_i) = \sum_{i=1}^4 N_e^i(c_0 + c_1 x_i + c_2 y_i)\\
  &= \left(\sum_{i=1}^4 N_e^i\right) c_0 + c_1 \sum_{i=1}^4 N_e^i x_i + c_2 \sum_{i=1}^4 N_e^i y_i = c_0 + c_1 x + c_2 y.
  \end{aligned} \tag{10.5}$$

- *integrability*: Note that this interpolation scheme ensures that u^h is continuous across elements. To see this, notice that on any element edge only those two shape functions are non-zero whose nodes are on that edge while the others are zero.

As for the simplicial element, computing shape function derivatives requires the use of the inverse function theorem.

In analogy to the simplicial element, we use $x = x(\xi, \eta)$ and $y = y(\xi, \eta)$ so the inverse function theorem yields

$$\begin{pmatrix} N^i_{e,x} \\ N^i_{e,y} \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} N^i_{e,\xi} \\ N^i_{e,\eta} \end{pmatrix} \qquad \text{and more generally:} \qquad \boxed{\nabla_{\boldsymbol{x}} N^i_e = \boldsymbol{J}^{-1} \nabla_{\boldsymbol{\xi}} N^i_e} \tag{10.6}$$

As a simple **example**, recall the 2-node bar element whose shape functions we computed as

$$N^1_e(x) = 1 - \frac{x}{\Delta x}, \qquad N^2_e(x) = \frac{x}{\Delta x}. \tag{10.7}$$

For a reference bar element with nodes at $\xi = \pm 1$, the analogous shape functions in 1D read

$$N^1_e(\xi) = \frac{1-\xi}{2}, \qquad N^2_e(\xi) = \frac{1+\xi}{2}. \tag{10.8}$$

Applying the above relations to the 1D problem gives

$$J = \frac{\partial x}{\partial \xi} = \frac{\partial N^1_e}{\partial \xi} x^1_e + \frac{\partial N^2_e}{\partial \xi} x^2_e = \frac{x^2_e - x^1_e}{2} = \frac{\Delta x}{2}. \tag{10.9}$$

This confirms that indeed

$$N^i_{e,x} = J^{-1} N^i_{e,\xi} = \frac{2}{\Delta x} N^i_{e,\xi}. \tag{10.10}$$

A useful relation to evaluate area integrals in the following is ($\boldsymbol{e}_1, \boldsymbol{e}_2$ being reference unit vectors)

$$\mathrm{d}\boldsymbol{A} = \mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{y} = \left(\frac{\partial x}{\partial \xi}\mathrm{d}\xi\, \boldsymbol{e}_1 + \frac{\partial x}{\partial \eta}\mathrm{d}\eta\, \boldsymbol{e}_2\right) \times \left(\frac{\partial y}{\partial \xi}\mathrm{d}\xi\, \boldsymbol{e}_1 + \frac{\partial y}{\partial \eta}\mathrm{d}\eta\, \boldsymbol{e}_2\right) = J\,\mathrm{d}\xi\,\mathrm{d}\eta\,\boldsymbol{e}_1 \times \boldsymbol{e}_2 \tag{10.11}$$

so that

$$\boxed{\mathrm{d}A = |\mathrm{d}\boldsymbol{A}| = J\,\mathrm{d}\xi\,\mathrm{d}\eta} \tag{10.12}$$

Extension to higher-order elements:

The **9-node quadratic quadrilateral** (**Q9**) derives its shape functions from applying the 1D shape functions of the 3-node bar (using Lagrangian interpolation) to the 2D case. For example,

$$N_1(\xi, \eta_= \frac{\xi(1-\xi)\eta(1-\eta)}{4}, \tag{10.13}$$

so that overall the interpolation includes monomials

$$\{1, \xi, \eta, \xi\eta, \xi^2, \eta^2, \xi^2\eta, \xi\eta^2, \xi^2\eta^2\}, \tag{10.14}$$

which is complete up to order $q = 2$ (quadratic).

Since the above elements includes way more polynomial terms than required for quadratic interpolation, one can construct elements with less nodes, e.g., the **8-node quadratic quadrilateral** (**Q8**) also known as **serendipity element**.

Shape functions are constructed as follows:

$$\begin{aligned} N_5(\xi,\eta) &= \frac{(1-\eta)(1-\xi^2)}{4}, \\ N_6(\xi,\eta) &= \frac{(1-\eta^2)(1+\xi)}{4}, \\ N_1(\xi,\eta) &= \frac{(1-\eta)(1-\xi)}{4} - \frac{1}{2}(N_5+N_8), \quad \text{etc.} \end{aligned} \tag{10.15}$$

Extension to three dimensions:

The same procedure can be applied to 3D, resulting in the **8-node brick element**. The reference coordinates (ξ,η,ζ) are linked to the physical coordinates (x,y,z) via the shape functions

$$N_e^1(\xi,\eta,\zeta) = \frac{1}{8}(1-\xi)(1-\eta)(1-\zeta), \quad \dots \tag{10.16}$$

again with

$$\boxed{\nabla_{\boldsymbol{x}} N_e^i = \boldsymbol{J}^{-1}\,\nabla_{\boldsymbol{\xi}} N_e^i} \qquad \text{and} \qquad \boxed{\mathrm{d}V = J\,\mathrm{d}\xi\,\mathrm{d}\eta\,\mathrm{d}\zeta} \tag{10.17}$$

11 Numerical quadrature

The finite element method frequently requires computing integrals, e.g., for the internal/external force vectors. Since these cannot be integrated analytically in general (with exceptions like the simplicial elements), we need **numerical integration** schemes.

Consider the integral

$$I[u] = \int_a^b u(x)\,\mathrm{d}x. \tag{11.1}$$

For convenience, let us introduce the **shift**

$$\xi = 2\frac{x-a}{b-a} - 1 \qquad \Rightarrow \qquad \xi(x=a) = -1, \quad \xi(x=b) = 1, \qquad \mathrm{d}\xi = \frac{2\,\mathrm{d}x}{b-a} \tag{11.2}$$

and $x = \frac{\xi+1}{2}(b-a)+a$ so that

$$I[u] = \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi \qquad \text{where} \qquad f(\xi) = \frac{b-a}{2}u\big(x(\xi)\big). \tag{11.3}$$

The goal is now to approximate the integral in (11.3) numerically.

11.1 Example: Riemann sums

Consider a *partition* with $n+1$ nodes:

$$P = \{[\xi_0,\xi_1],[\xi_1,\xi_2],\dots,[\xi_{n-1},\xi_n]\} \qquad \text{such that} \qquad -1 = \xi_0 < \xi_1 < \xi_2 < \dots < \xi_n = 1 \tag{11.4}$$

The **Riemann sum** makes use of the approximation

$$I \approx S = \sum_{i=1}^{n} f(\xi_i^*)(\xi_i - \xi_{i-1}) \qquad \text{with} \qquad \xi_{i-1} \le \xi_i^* \le \xi_i. \tag{11.5}$$

Different choices of ξ_i^*:

(i) **left Riemann sum**: $\xi_i^* = \xi_{i-1}$:
(ii) **right Riemann sum**: $\xi_i^* = \xi_i$
(iii) **middle Riemann sum**: $\xi_i^* = \frac{1}{2}(\xi_{i-1}+\xi_i)$
(iv) **trapezoidal sum**: average of left and right
(v) **upper Riemann sum**: ξ_i^* s.t. $g(\xi_i^*) = \sup_{\xi\in[\xi_{i-1},\xi_i]} g(\xi)$
(vi) **lower Riemann sum**: ξ_i^* s.t. $g(\xi_i^*) = \inf_{\xi\in[\xi_{i-1},\xi_i]} g(\xi)$

More refined formulations can be found in the so-called **Newton-Cotes** formulas (the trapezoidal rule is the Newton-Cotes formula of degree 1).

11.2 Gauss quadrature

A much more efficient alternative are **quadrature rules** (also called **cubature rules** in 3D) which are best suited for polynomial functions:

$$I[u] = \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi \approx \sum_{i=0}^{n_{QP}-1} W_i\, f(\xi_i). \tag{11.6}$$

Now, we need to choose n_{QP}, $\{W_0, \dots, W_{n_{QP}-1}\}$ and $\{\xi_0, \dots, \xi_{n_{QP}-1}\}$. The choice should depend on the function to be integrated (more specifically, on its *smoothness*). Note that most functions of interest will be of polynomial type.

We say a quadrature rule is **exact of order** q if it integrates exactly all polynomial functions $g \in \mathbb{P}_q([-1,1])$. **Gauss quadrature** generally chooses n_{QP} quadrature points and associated weights such that the quadrature rule is exact of order $q = 2n_{QP} - 1$.

11.2.1 Gauss-Legendre quadrature

Gauss-Legendre quadrature selects the nodes and weights such that the first $2n_{QP}$ *moments* are computed exactly:

$$\mu_k = \int_{-1}^{1} \xi^k \, \mathrm{d}\xi \overset{!}{=} \sum_{i=0}^{n_{QP}-1} W_i \, \xi_i^k, \qquad k = 0, 1, \dots, 2n_{QP} - 1. \tag{11.7}$$

These are $2n_{QP}$ equations for the $2n_{QP}$ free parameters (W_i, ξ_i) for $i = 0, \dots, n_{QP} - 1$. The equations are generally nonlinear and thus hard to solve analytically.

Let us compute the Gauss-Legendre weights and points for the lowest few orders in 1D:

- *a single quadrature point*, i.e., $n_{QP} = 1$:

  Two equations for two unknowns:

  $$\int_{-1}^{1} \xi^0 \, \mathrm{d}\xi = 2 = W_0 \, \xi_0^0 = W_0 \qquad \text{and} \qquad \int_{-1}^{1} \xi^1 \, \mathrm{d}\xi = 0 = W_0 \, \xi_0^1 = W_0 \xi_0 \tag{11.8}$$

  So that the first-order quadrature rule is given by

  $$\boxed{W_0 = 2, \quad \xi_0 = 0} \tag{11.9}$$

  Since linear functions are integrated exactly, this quadrature rule is *exact to order* $q = 1$.

- *two quadrature points*, i.e., $n_{QP} = 2$:

  In close analogy, we now have

  $$\begin{aligned} \int_{-1}^{1} \xi^0 \, \mathrm{d}\xi &= 2 = W_0 + W_1, & \int_{-1}^{1} \xi^1 \, \mathrm{d}\xi &= 0 = W_0 \, \xi_0 + W_1 \, \xi_1, \\ \int_{-1}^{1} \xi^2 \, \mathrm{d}\xi &= \frac{1}{3} = W_0 \, \xi_0^2 + W_1 \, \xi_1^2, & \int_{-1}^{1} \xi^3 \, \mathrm{d}\xi &= 0 = W_0 \, \xi_0^3 + W_1 \, \xi_1^3. \end{aligned} \tag{11.10}$$

  A simple solution can be found for symmetric quadrature points $\xi_0 = -\xi_1$:

  $$\boxed{W_0 = W_1 = 1, \quad \xi_0 = -\frac{1}{\sqrt{3}}, \quad \xi_1 = \frac{1}{\sqrt{3}}.} \tag{11.11}$$

  This quadrature rule is *exact to order* $q = 3$ (cubic polynomials are integrated exactly).

- *higher-order quadrature rules*:

  Quadrature weights and points for arbitrary order can be obtained in analogous fashion and, most importantly, can be found in numerous look-up tables (see notes and textbooks). However, there is a better, systematic way to compute Gauss-Legendre quadrature weights and points.

Note that monomials $\{1, \xi, \xi^2, \xi^3, \ldots\}$, although complete, are not orthogonal basis functions. We can turn them into orthogonal polynomials $P_n(\xi)$ by, e.g., the **Gram-Schmidt orthogonalization** procedure. To this end, let us start with

$$P_0(\xi) = 1 \tag{11.12}$$

and obtain the next basis function by starting with the linear momonial ξ and computing

$$P_1(\xi) = \xi - \frac{\langle 1, \xi \rangle}{\langle 1, 1 \rangle} 1 = \xi, \tag{11.13}$$

where we used the inner product

$$\langle u, v \rangle = \int_{-1}^{1} u(\xi) v(\xi) \, \mathrm{d}\xi. \tag{11.14}$$

Similarly, the next higher basis function is obtained by starting from ξ^2, so that

$$P_2(\xi) = \xi^2 - \frac{\langle \xi, \xi^2 \rangle}{\langle \xi, \xi \rangle} \xi - \frac{\langle 1, \xi^2 \rangle}{\langle 1, 1 \rangle} 1 = \xi^2 - \frac{1}{3}. \tag{11.15}$$

Analogously, one finds

$$P_3(\xi) = \xi^3 - \frac{3}{5}\xi. \tag{11.16}$$

By continuing analogously, we create a *countably infinite set of orthogonal basis functions* $P_n(\xi)$ such that

$$\int_{-1}^{1} P_n(\xi) P_m(\xi) \, \mathrm{d}\xi = 0 \qquad \text{if } n \neq m. \tag{11.17}$$

These polynomials are known as **Legendre polynomials**. Note that they are defined only up to a constant, so one can renormalize them, which is commonly done by enforcing that $P_n(1) = 1$ for all n. The result is the well known Legendre polynomials which can alternatively be defined via

$$\boxed{P_n(\xi) = \frac{1}{2^n n!} \frac{\mathrm{d}^n}{\mathrm{d}\xi^n} \left[(\xi^2 - 1)^n \right]} \tag{11.18}$$

These polynomials have another interesting feature, viz. by orthogonality with $P_0(\xi) = 1$ we know that

$$\int_{-1}^{1} P_n(\xi) \, \mathrm{d}\xi = \langle 1, P_n \rangle = \begin{cases} 2, & \text{if } n = 0, \\ 0, & \text{else.} \end{cases} \tag{11.19}$$

$P_n(\xi)$ has exactly n roots in the interval $[-1, 1]$. Also, for $n \neq 0$ we know that

$$\begin{aligned} P_n(\xi) &= -P_n(-\xi) \qquad \text{for odd } n, \\ P_n(\xi) &= P_n(-\xi) \qquad \text{for even } n. \end{aligned} \tag{11.20}$$

Moreover, $P_n(0) = 0$ for odd n.

With this new set of basis functions, we can define the Gauss-Legendre quadrature rule to enforce

$$\int_{-1}^{1} P_k(\xi) \, \mathrm{d}\xi \overset{!}{=} \sum_{i=0}^{n_{QP}-1} W_i \, P_k(\xi_i), \qquad k = 0, 1, \ldots, 2n_{QP} - 1. \tag{11.21}$$

If $n_{QP}=1$, then the solution is simple because the above equations simplify to

$$W_0 = 2 \qquad \text{and} \qquad 0 = W_0 P_1(\xi_0). \tag{11.22}$$

Therefore, the weight is, as before, $W_0 = 2$ and the quadrature point is the root of $P_1(\xi)$, viz. $\xi_0 = 0$.

If $n_{QP}=2$, then the four equations to be solved are

$$\begin{aligned} W_0 + W_1 = 2 \qquad &\text{and} \qquad 0 = W_0 P_1(\xi_0) + W_1 P_1(\xi_1), \\ 0 = W_0 P_2(\xi_0) + W_1 P_2(\xi_1) \qquad &\text{and} \qquad 0 = W_0 P_3(\xi_0) + W_1 P_3(\xi_1). \end{aligned} \tag{11.23}$$

By analogy, we choose the quadrature points to be the roots of $P_2(\xi)$, so that

$$P_2(\xi_0) = P_2(\xi_1) = 0 \qquad \Rightarrow \qquad \xi_0 = \frac{1}{\sqrt{3}}, \quad \xi_1 = -\frac{1}{\sqrt{3}}. \tag{11.24}$$

Using $P_n(\xi) = -P_n(-\xi)$, the above equations reduce to

$$W_0 = W_1 = 1. \tag{11.25}$$

Further, note that $P_n(0) = 0$ for odd n. Therefore, the same procedure can be continued as follows. For an arbitrary number of quadrature points, n_{QP}, the **Gauss-Legendre quadrature** points and associated weights are defined by

$$\boxed{P_{n_{QP}}(\xi_i) = 0, \qquad w_i = \frac{2}{(1-\xi_i^2)[P'_{n_{QP}}(\xi_i)]^2} \qquad i = 0, \ldots, n_{QP}-1.} \tag{11.26}$$

As a check, take, e.g., $n_{QP} = 1$ so that $P_1(\xi) = \xi$ with root $\xi_0 = 0$. The weight is computed as

$$w_0 = \frac{2}{(1-\xi_0^2)[P'_{n_{QP}}(\xi)]^2} = 2, \tag{11.27}$$

as determined above. Similarly, for $n_{QP} = 2$ we have $P_2(\xi) = \frac{1}{2}(3\xi^2 - 1)$ with the above roots of $\pm 1/\sqrt{3}$. The associated weights are computed as

$$w_0 = \frac{2}{(1-\xi_0^2)[P_2'(\xi_0)]^2} = \frac{2}{(1-\xi_0^2)[3\xi_0]^2} = \frac{2}{\frac{2}{3}\frac{3^2}{3}} = 1 = w_1, \tag{11.28}$$

which agrees with our prior solution.

11.2.2 Other Gauss quadrature rules

Note that if, for general functions f, one can sometimes find a decomposition $f(\xi) = w(\xi)g(\xi)$ where $w(\cdot)$ is a known weighting function and $g(\xi)$ is (approxiately) polynomial, so that a more suitable quadrature rule may be found via

$$I[u] = \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi = \int_{-1}^{1} w(\xi)\,g(\xi)\,\mathrm{d}\xi \approx \sum_{i=0}^{n_{QP}-1} w(\xi_i)\,g(\xi_i). \tag{11.29}$$

Examples of such Gaussian quadrature rules include those of **Gauss-Chebyshew** type, which are obtained form a weighting function $w(\xi) = (1-\xi^2)^{-1/2}$, and the quadrature points are the roots of Chebyshew polynomials. **Gauss-Hermite** quadrature uses a weighting function

$w(\xi) = \exp(-\xi^2)$ (and the integral is taken over the entire real axis). Gauss-Legendre quadrature is included as the special case $w(\xi) = 1$.

Another popular alternative (less for FE though) is **Gauss-Lobatto quadrature** which includes the interval end points as quadrature points and is accurate for polynomials up to degree $2n_{QP} - 3$, viz.

$$I[u] = \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi \approx \frac{2}{n_{QP}(n_{QP}-1)}\big[f(-1)+f(1)\big] + \sum_{i=2}^{n_{QP}-1} W_i\, f(\xi_i). \tag{11.30}$$

11.3 Higher dimensions

Like the polynomial shape functions, the above quadrature rules can easily be extended to 2D and 3D, e.g.,

$$\begin{aligned}\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta)\,\mathrm{d}\xi\,\mathrm{d}\eta &= \int_{-1}^{1}\left[\sum_{i=0}^{N-1} W_i\, f(\xi_i,\eta)\right]\mathrm{d}\eta = \sum_{j=0}^{N-1} W_j\left[\sum_{i=0}^{N-1} W_i\, f(\xi_i,\eta_j)\right]\\ &= \sum_{k=0}^{n_{QP}-1} W_k^*\, f(\xi_k,\eta_k)\end{aligned} \tag{11.31}$$

with the combined weights $W_k^* = W_i W_j$ and points $(\xi_k,\eta_k) = (\xi_i,\eta_j)$ obtained from the individual quadrature rules in each direction. By symmetry we choose $N = \sqrt{n_{QP}}$ so that $N^2 = n_{QP}$.

For example, consider the **Q4 element**. By reusing the 1D Gauss-Legendre weights and points, we now have:

- *first-order quadrature* ($q = 1$), as in 1D, has only a single quadrature point ($n_{QP} = 1$):

$$\boxed{W_0 = 1 \qquad \text{and} \qquad (\xi_0,\eta_0) = (0,0)} \tag{11.32}$$

  Bilinear functions (at most linear in ξ and η) are integrated exactly with this rule.

- *third-order quadrature* ($q = 3$), now has four quadrature points ($n_{QP} = 2^2 = 4$):

$$\boxed{W_0 = W_1 = W_2 = W_3 = 1 \qquad \text{and} \qquad (\xi_i,\eta_i) = \left(\pm\frac{1}{\sqrt{3}}, \pm\frac{1}{\sqrt{3}}\right)} \tag{11.33}$$

  Bicubic polynomial functions (at most cubic in ξ and η) are integrated exactly.

Similarly, the **brick element** in 3D uses Gauss-Legendre quadrature as follows:

- *first-order quadrature* ($q = 1$) still has a single quadrature point ($n_{QP} = 1$):

$$\boxed{W_0 = 1 \qquad \text{and} \qquad (\xi_0,\eta_0) = (0,0)} \tag{11.34}$$

- *third-order quadrature* ($q = 3$), now has four quadrature points ($n_{QP} = 2^3 = 8$):

$$\boxed{W_i = 1 \qquad \text{and} \qquad (\xi_i,\eta_i,\zeta_i) = \left(\pm\frac{1}{\sqrt{3}}, \pm\frac{1}{\sqrt{3}}, \pm\frac{1}{\sqrt{3}}\right).} \tag{11.35}$$

11.4 Finite element implementation

The key integrals to be evaluated numerically are (e.g., in 2D)

$$I^e = \int_{\Omega_e} W(\nabla \boldsymbol{u}^h)\,\mathrm{d}V = \int_{-1}^{1}\int_{-1}^{1} W\big(\nabla \boldsymbol{u}^h(\boldsymbol{\xi})\big)\,J(\boldsymbol{\xi})t_e\,\mathrm{d}\xi\,\mathrm{d}\eta \approx \sum_{k=0}^{n_{QP}-1} W_k\, W\big(\nabla \boldsymbol{u}^h(\boldsymbol{\xi}_k)\big)\,J(\boldsymbol{\xi}_k)t_e,$$

$$F^a_{\text{int},i} = \int_{\Omega_e} \sigma_{ij} N^a_{,j}\,\mathrm{d}V = \int_{-1}^{1}\int_{-1}^{1} \sigma_{ij}(\boldsymbol{\xi}) N^a_{,j}(\boldsymbol{\xi})\,J(\boldsymbol{\xi})t_e\,\mathrm{d}\xi\,\mathrm{d}\eta \approx \sum_{k=0}^{n_{QP}-1} W_k\,\sigma_{ij}(\boldsymbol{\xi}_k) N^a_{,j}(\boldsymbol{\xi}_k)\,J(\boldsymbol{\xi}_k)t_e \tag{11.36}$$

and the tangent matrix

$$\begin{aligned} T^{ab}_{ik} &= \int_{\Omega_e} \mathbb{C}_{ijkl} N^a_{,j} N^b_{,l}\,\mathrm{d}V = \int_{-1}^{1}\int_{-1}^{1} \mathbb{C}_{ijkl}(\boldsymbol{\xi}) N^a_{,j}(\boldsymbol{\xi}) N^b_{,l}(\boldsymbol{\xi}) J(\boldsymbol{\xi}) t_e\,\mathrm{d}\xi\,\mathrm{d}\eta \\ &\approx \sum_{k=0}^{n_{QP}-1} W_k\,\mathbb{C}_{ijkl}(\boldsymbol{\xi}_k) N^a_{,j}(\boldsymbol{\xi}_k) N^b_{,l}(\boldsymbol{\xi}_k) J(\boldsymbol{\xi}_k) t_e \end{aligned} \tag{11.37}$$

Notice for implementation purposes that $\sigma_{ij} N^a_{,j}$ is a simple matrix-vector multiplication so that, using the isoparametric mapping of Section 8.3,

$$\boldsymbol{F}^a_{\text{int}} \approx \sum_{k=0}^{n_{QP}-1} W_k J(\boldsymbol{\xi}_k) t_e\,\boldsymbol{\sigma}(\boldsymbol{\xi}_k) \nabla_{\boldsymbol{x}} N^a(\boldsymbol{\xi}_k) \qquad \text{with} \qquad \nabla_{\boldsymbol{x}} N^a = \boldsymbol{J}^{-1} = \nabla_{\boldsymbol{\xi}} N^a. \tag{11.38}$$

11.5 Quadrature error estimates

Using numerical quadrature to approximate an exact integral introduces a **quadrature error**, which is bounded as follows. For a function $f(\xi) \in C^{2n_{QP}}(\Omega)$ we have that (without proof here)

$$\left| \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi - \sum_{k=0}^{n_{QP}-1} W_k\, f(\xi_k) \right| \le C h^{2n_{QP}} \max_{\xi\in[-1,1]} \left| f^{(2n_{QP})}(\xi) \right| \tag{11.39}$$

with a constant $C > 0$. This shows that, as can be expected, the quadrature error decreases with decreasing mesh size h and with increasing smoothness of function f. The rate of convergence under h-refinement also depends on the smoothness of the function.

The exact error depends on the chosen quadrature rule. For example, for Gauss-Legendre quadrature an error estimate is given by

$$e = \frac{2^{2n_{QP}+1}(n_{QP}!)^4}{(2n_{QP}+1)[(2n_{QP})!]^3} \max_{\xi\in[-1,1]} \left\| f^{(2n_{QP})}(\xi) \right\|. \tag{11.40}$$

11.6 Quadrature rules for simplicial elements:

As discussed above, simplicial elements (bar, triangle, tetrahedron) produce *constant strains* and thus *constant stresses* within elements. Hence, a **single quadrature point** at an **arbitrary location** inside the element is sufficient. Usually one chooses the point to be located at the element center, which gives

$$W_0 = 1, \qquad r_0 = s_0 = \frac{1}{3} \text{ (in 2D)} \qquad \text{and} \qquad r_0 = s_0 = t_0 = \frac{1}{4} \text{ (in 3D)}. \tag{11.41}$$

If higher-order quadrature rules are required (e.g., for triangular and tetrahedral elements of higher interpolation order as discussed next), the same concepts as for Gauss-Legendre quadrature can be applied here, resulting in **simplicial quadrature** rules whose weights and quadrature point locations can be found in look-up tables.

11.7 Which quadrature rule to use?

Stresses, shape function derivatives, and Jacobians are not necessarily smooth polynomials. Thus, rather than finding the exact integration order for each element and constitutive model, we introduce a minimum required integration order for a particular element type.

Our minimum requirement is that an undistorted elastic element is integrated exactly. Thus, we define **full integration** as the order needed to integrate an *undistorted, homogeneous, linear elastic element exactly*. An element is undistorted if element angles are preserved or, in other words, if $J = \text{const}$.

For **example**, the 4-node quadrilateral (Q4) is undistorted if the physical element has the shape of a rectangle (side lengths a and b), so that

$$J = \frac{ab}{4} = \text{const}. \tag{11.42}$$

Then, for a linear elastic Q4 element we have

$$F^a_{\text{int},i} = \int_{-1}^{1}\int_{-1}^{1} \mathbb{C}_{ijkl}\,\varepsilon^h_{kl}(\boldsymbol{\xi})N^a_{,j}(\boldsymbol{\xi})\,\frac{ab}{4}t_e\,\mathrm{d}\xi\,\mathrm{d}\eta = \frac{abt_e}{4}\mathbb{C}_{ijkl}\int_{-1}^{1}\int_{-1}^{1}\varepsilon^h_{kl}(\boldsymbol{\xi})N^a_{,j}(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta, \tag{11.43}$$

where $\boldsymbol{\varepsilon}^h = \text{sym}(\nabla_{\boldsymbol{x}}\boldsymbol{u}^h)$ is at most linear (since the interpolation of $\boldsymbol{u}^h$ is bilinear), $\nabla_{\boldsymbol{x}}N^a$ is also at most linear for the same reason. Overall, the integrand is at most a quadratic polynomial, so that we need integration order $q \geq 2$.

Recall that in 1D we showed that $q = 2n_{QP} - 1$, so that full integration of the Q4 element in 2D requires $n_{QP} = 2 \times 2 = 4$ quadrature points.

Analogously, full integration of the quadratic 2D elements Q8/Q9 requires $n_{QP} = 3^2 = 9$ quadrature points. Full integration of the 8-node brick element requires $n_{QP} = 8$ quadrature points.

For simplicial elements, we showed that a single quadrature point ($n_{QP} = 1$) is always sufficient to integrate exactly, since stresses and strains within elements are constant. For the quadratic triangle (T6), full integration requires order $q = 2$, which corresponds to *three quadrature points.*

Note that not only does full integration guarantee that the internal force vector of an undistorted, elastic element is integrated exactly. By reviewing the element energy and the element tangent matrix, we make the same observation (i.e., those are integrated exactly as well):

$$\begin{aligned} I_e &= \int_{\Omega_e} W\,\mathrm{d}V = \frac{abt_e}{4}\frac{1}{2}\mathbb{C}_{ijkl}\int_{-1}^{1}\int_{-1}^{1}\varepsilon^h_{ij}(\boldsymbol{\xi})\varepsilon^h_{kl}(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta, \\ T^{ab}_{ij} &= \frac{abt_e}{4}\mathbb{C}_{ijkl}\int_{-1}^{1}\int_{-1}^{1}N^a_{,j}(\boldsymbol{\xi})N^b_{,l}(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta. \end{aligned} \tag{11.44}$$

Using an integration rule less than full integration is called **under-integration**; the opposite is called **over-integration**. Which integration order to use depends very much on the element, material model, etc. Sometimes under-integration can be beneficial (e.g., to avoid locking). We will discuss some of these issues later.

12 Generalization and implementation of the simplicial elements

For a general simplicial element in d dimensions having $n = d+1$ nodes, we have shape functions

$$N_e^1(r_1,\dots,r_d) = r_1, \qquad N_e^2(r_1,\dots,r_d) = r_2, \qquad \dots \qquad N_e^n(r_1,\dots,r_d) = 1 - \sum_i^d r_i, \quad (12.1)$$

where $\boldsymbol{\xi} = \{r_1,\dots,r_d\}$ denote the d *barycentric coordinates.* Then, the Jacobian $\boldsymbol{J}$ is given by

$$J_{ij} = \frac{\partial X_j}{\partial r_i} = \sum_{a=1}^n X_j^a \frac{\partial N_e^a}{\partial r_i} \qquad \text{or} \qquad \boldsymbol{J} = \sum_{a=1}^n \nabla_{\boldsymbol{\xi}} N_e^a \otimes \boldsymbol{X}^a, \qquad J = \det \boldsymbol{J}, \quad (12.2)$$

so that shape function derivatives in *physical coordinates* follow as

$$\begin{pmatrix} N^a_{e,X_1} \\ \dots \\ N^a_{e,X_d} \end{pmatrix} = \boldsymbol{J}^{-1} \begin{pmatrix} N^a_{e,r_1} \\ \dots \\ N^a_{e,r_d} \end{pmatrix} \qquad \text{or} \qquad \nabla_{\boldsymbol{X}} N_e^a = \boldsymbol{J}^{-1} \nabla_{\boldsymbol{\xi}} N_e^a. \quad (12.3)$$

Using numerical quadrature with weights W_k and points $\boldsymbol{\xi}_k$ (in *reference* coordinates), the element energy is given by

$$I_e \approx \sum_{k=1}^{n_{QP}} W_k\, W\left(\boldsymbol{F}(\boldsymbol{\xi}_k)\right) J(\boldsymbol{\xi}_k)\, t_e. \quad (12.4)$$

Nodal forces are approximated by

$$F^a_{\text{int},i} \approx \sum_{k=1}^{n_{QP}} W_k\, P_{iJ}\left(\boldsymbol{F}(\boldsymbol{\xi}_k)\right) N^a_{,J}(\boldsymbol{\xi}_k) J(\boldsymbol{\xi}_k)\, t_e. \quad (12.5)$$

Finally, the element stiffness matrix components become

$$T^{ab}_{il} \approx \sum_{k=1}^{n_{QP}} W_k\, \mathbb{C}_{iJlL}\left(\boldsymbol{F}(\boldsymbol{\xi}_k)\right) N^a_{,J}(\boldsymbol{\xi}_k) N^b_{,L}(\boldsymbol{\xi}_k) J(\boldsymbol{\xi}_k)\, t_e. \quad (12.6)$$

Here, t_e is an element constant (e.g., the cross-sectional area A in 1D, the thickness t in 2D, and simply 1 in 3D).

The deformation gradient, $\boldsymbol{F} = \boldsymbol{I} + \nabla_{\boldsymbol{X}} \boldsymbol{u}$, and strain tensor, $\boldsymbol{\varepsilon} = \text{sym}(\nabla_{\boldsymbol{X}} \boldsymbol{u})$, can be obtained directly from $\nabla_{\boldsymbol{X}} \boldsymbol{u}$. Also, recall that

$$\nabla_{\boldsymbol{X}} \boldsymbol{u}(\boldsymbol{\xi}) = \sum_{a=1}^n \boldsymbol{u}_e^a \otimes \nabla_{\boldsymbol{X}} N^a(\boldsymbol{\xi}). \quad (12.7)$$

Note that, in *linearized kinematics*, the above equations hold analogously with P_{iJ} replaced by the Cauchy stress tensor, upper-case coordinates $\boldsymbol{X}$ by lower-case $\boldsymbol{x}$, etc.

13 Assembly

So far, we have defined **local** element vectors and matrices. The solution of any problem requires the assembly of **global** vectors and matrices. Specifically, recall that shape functions associated with a node a are non-zero only in elements adjacent to node a (this is the principle of *locality*). When computing, e.g., the total energy of a body discretized into n_e elements, we exploit that

$$I = \int_\Omega W(\nabla \boldsymbol{u}^h)\,\mathrm{d}V = \sum_{e=1}^{n_e} \int_{\Omega_e} W(\nabla \boldsymbol{u}^h)\,\mathrm{d}V = \sum_{e=1}^{n_e} I^e. \tag{13.1}$$

When computing force vectors and incremental stiffness matrices, the situation is more complex, which is why we introduce the **assembly operator**, viz.

$$\boldsymbol{F}_{\text{int}} = \mathop{\mathcal{A}}_{e=1}^{n_e} \boldsymbol{F}_{\text{int},e}, \qquad \boldsymbol{T}_{\text{int}} = \mathop{\mathcal{A}}_{e=1}^{n_e} \boldsymbol{T}_{\text{int},e}, \tag{13.2}$$

which loops over all n_e elements e and *adds* their respective contributions to the global quantities. This requires careful book-keeping to keep track of the correspondence between local and global node numbering (and is the reason our implemented elements store the node IDs).

Similarly, an inverse assignment operator extracts element quantities from a global vector:

$$\boldsymbol{U}_e^h = \mathop{\mathcal{A}}_{e}^{-1}(\boldsymbol{U}^h). \tag{13.3}$$

For practical purposes, it is helpful to recall that entry U_i^a is located at position $a \cdot d + i$ in the assembled vector $\boldsymbol{U}^h$ (using C++ notation with 0-indexed lists, in d dimensions). For example, consider a 2D problem using 2-node bar elements. If an element connecting nodes 1 and 3 computes the nodal force vectors $\boldsymbol{F}_{\text{int},e}^1$ and $\boldsymbol{F}_{\text{int},e}^3$, then they are to be added into the global force vector as

$$\boldsymbol{F}_{\text{int}} = \begin{pmatrix} \cdot \\ \cdot \\ F_{\text{int},e,1}^1 \\ F_{\text{int},e,2}^1 \\ \cdot \\ \cdot \\ F_{\text{int},e,1}^3 \\ F_{\text{int},e,2}^3 \\ \cdot \\ \cdot \end{pmatrix} \tag{13.4}$$

Analogously, if the element connecting nodes 0 and 2 computes a stiffness matrix $\boldsymbol{K}_e^{02}$, then its components must be added onto the global tangent stiffness matrix as

$$\boldsymbol{T} = \begin{pmatrix} K_{11}^{02} & K_{12}^{02} & \cdot & \cdot & K_{13}^{02} & K_{14}^{02} & \cdot & \cdot \\ K_{21}^{02} & K_{22}^{02} & \cdot & \cdot & K_{23}^{02} & K_{24}^{02} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ K_{31}^{02} & K_{32}^{02} & \cdot & \cdot & K_{33}^{02} & K_{34}^{02} & \cdot & \cdot \\ K_{41}^{02} & K_{42}^{02} & \cdot & \cdot & K_{43}^{02} & K_{44}^{02} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix} \tag{13.5}$$

Note that because nodes in FEM have non-zero shape functions only in adjacent elements, the above matrix is generally sparse (which comes with significant computational advantages).

14 Overview: Numerical Implementation

The above theoretical concepts are implemented in our c++ finite element code, whose general structure is summarized in the following. Details can be extracted from the source files. The code structure is schematically shown in Fig. 1.

material model:

The material model computes

- $W = W(\nabla \boldsymbol{u})$
- $P_{iJ} = P_{iJ}(\nabla \boldsymbol{u})$ or $\sigma_{ij} = \sigma_{ij}(\nabla \boldsymbol{u})$
- $\mathbb{C}_{iJkL} = \mathbb{C}_{iJkL}(\nabla \boldsymbol{u})$ or $\mathbb{C}_{ijkl}(\nabla \boldsymbol{u})$

Depending on the (finite/linearized) model, the strain is

$$\boldsymbol{F} = \boldsymbol{I} + \nabla \boldsymbol{u} \qquad \text{or}$$
$$\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathrm{T}})$$

Also, internal variables must be updated[1]:

$$\boldsymbol{z}_{\alpha+1} = \arg\inf \mathcal{F}_{\boldsymbol{z}_\alpha}(\nabla \boldsymbol{u})$$

The `MaterialModel` class implements:

- `computeEnergy`$(\nabla \boldsymbol{u}, \boldsymbol{z}_\alpha, t) \rightarrow W^*$
- `computeStresses`$(\nabla \boldsymbol{u}, \boldsymbol{z}_\alpha, t) \rightarrow P_{iJ}$ or σ_{ij}
- `computeTangentMatrix`$(\nabla \boldsymbol{u}, \boldsymbol{z}_\alpha, t) \rightarrow \mathbb{C}_{iJkL}$

The respective strain tensor is provided by

- `computeStrain`$(\nabla \boldsymbol{u}) \rightarrow F_{iJ}$ or ε_{ij}

which can, of course, also be called within the element to conveniently compute stresses, etc.

Internal variables are updated by[1]

- `updateInternalVariables`$(\nabla \boldsymbol{u}, \boldsymbol{z}_\alpha, t)$
  $\rightarrow \boldsymbol{z}_{\alpha+1}$ (if no update, simply return $\boldsymbol{z}_\alpha$)

element:

The element computes

- $I_e \cong \sum_{k=1}^{n_{QP}} W_k\, W\big(\nabla \boldsymbol{u}(\boldsymbol{\xi}_k)\big) J(\boldsymbol{\xi}_k)$
- $(F_{\mathrm{int},e})_i^a \cong \sum_{k=1}^{n_{QP}} W_k\, P_{iJ}(\boldsymbol{\xi}_k) N^a_{,J}(\boldsymbol{\xi}_k) J(\boldsymbol{\xi}_k)$
- $(T_{\mathrm{int},e})_{ik}^{ab} \cong \sum_{k=1}^{n_{QP}} W_k\, \mathbb{C}_{iJkL}(\boldsymbol{\xi}_k) N^a_{,J}(\boldsymbol{\xi}_k) N^b_{,L}(\boldsymbol{\xi}_k) \times J(\boldsymbol{\xi}_k)$

In addition, internal variables must be updated[1]:

$$\boldsymbol{z}_{\alpha+1} = \arg\inf \mathcal{F}_{\boldsymbol{z}_\alpha}(\nabla \boldsymbol{u}, t)$$

The `Element` class implements:

- `computeEnergy`$(\boldsymbol{U}_e^h, t) \rightarrow I_e$
  where $\boldsymbol{U}_e^h = \{\boldsymbol{u}_e^1, \ldots, \boldsymbol{u}_e^n\}$
- `computeForces`$(\boldsymbol{U}_e^h, t) \rightarrow \{\boldsymbol{F}_{\mathrm{int},e}^1, \ldots, \boldsymbol{F}_{\mathrm{int},e}^n\}$
- `computeStiffnessMatrix`$(\boldsymbol{U}_e^h, t) \rightarrow (T_{\mathrm{int},e})_{ik}^{ab}$

Note that the element has a `MaterialModel`, which is used to compute W, P_{iJ}, and $\mathbb{C}_{iJkL}$ from $\nabla \boldsymbol{u}$, $\boldsymbol{z}_\alpha$, and time t.

The element stores and updates[1] $\boldsymbol{z}$:

- `updateInternalVariables`$(\boldsymbol{U}_e^h, t)$:
  update $\boldsymbol{z}_\alpha \leftarrow \boldsymbol{z}_{\alpha+1}$

by calling the `MaterialModel`.

[1] Internal variables will be discussed later in the context of inelasticity. They are included here for completeness.

assembler:

The assembly procedure calculates

- $I(\boldsymbol{U}^h) = \mathop{\mathcal{A}}\limits_{e=1}^{n_e} I_e(\boldsymbol{U}_e^h)$
- $\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) = \mathop{\mathcal{A}}\limits_{e=1}^{n_e} \boldsymbol{F}_{\text{int},e}(\boldsymbol{U}_e^h)$
- $\boldsymbol{T}_{\text{int}}(\boldsymbol{U}^h) = \mathop{\mathcal{A}}\limits_{e=1}^{n_e} \boldsymbol{T}_{\text{int},e}(\boldsymbol{U}_e^h)$

These are the *global* quantities derived from *local* element quantities.

The `Assembler` class implements:

- `assembleEnergy`$(\boldsymbol{U}^h, t)$ → I
  where $\boldsymbol{U}^h = \{\boldsymbol{u}^1, \dots, \boldsymbol{u}^n\}$
- `assembleForces`$(\boldsymbol{U}^h, t)$ → $\boldsymbol{F}_{\text{int}} - \boldsymbol{F}_{\text{ext}}$
- `assembleStiffnessMatrix`$(\boldsymbol{U}^h, t)$ → $\boldsymbol{T}_{\text{int}}$

The assembler calls all elements to request their contributions, then assembles those into the global matrices.

The assembler also implements[1]

- `updateInternalVariables`$(\boldsymbol{U}^h, t)$

which asks each element to update its internal variables (usually called at the end of a converged solver step; this will be discussed later).

Finally, note that material models and elements use **vector notation** for all stiffness matrices.

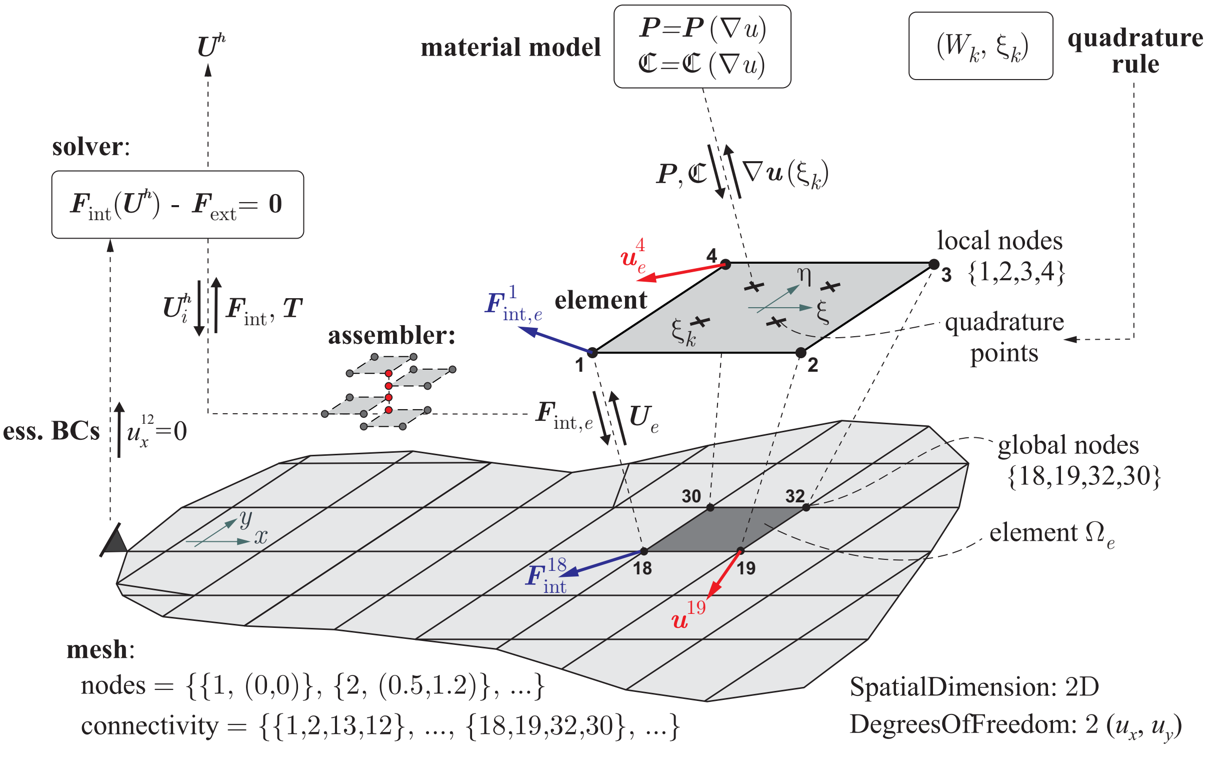


Figure 1: Illustration of the overall code structure.

15 Iterative solvers

In the linear elastic or thermal problem, the solution is simple to obtain from a linear system of equations, as discussed before. In case of nonlinear problems, an iterative solution method is required. Here, we discuss a few common examples.

The problem to be solved has the general form

$$\boldsymbol{f}(\boldsymbol{U}^h) = \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}} = \boldsymbol{0}. \tag{15.1}$$

All iterative solvers start with an initial guess $\boldsymbol{U}_0^h$, which is then corrected in a multitude of ways to find

$$\boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \Delta\boldsymbol{U}_n^h. \tag{15.2}$$

The iterative scheme converges if $\Delta\boldsymbol{U}_n^h \to 0$ as $n \to \infty$, or equivalently $\boldsymbol{f}(\boldsymbol{U}_n^h) \to \boldsymbol{0}$ as $n \to \infty$.

15.1 Netwon-Raphson (NR) method

The **Newton-Raphson method** (introduced by Newton in 1669 and generalized by Raphson in 1690) starts with a Taylor expansion:

$$\begin{aligned} 0 &= \boldsymbol{f}(\boldsymbol{U}_{n+1}^h) = \boldsymbol{f}(\boldsymbol{U}_n^h + \Delta\boldsymbol{U}^n) \\ &= \boldsymbol{f}(\boldsymbol{U}_n^h) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{U}}(\boldsymbol{U}_n^h)\,\Delta\boldsymbol{U}_n^h + O(\Delta\boldsymbol{U}_n^h). \end{aligned} \tag{15.3}$$

If we neglect higher-order terms, then the above can be solved for the increment

$$\Delta\boldsymbol{U}_n^h = -\left[\boldsymbol{T}(\boldsymbol{U}_n^h)\right]^{-1}\boldsymbol{f}(\boldsymbol{U}_n^h) \qquad \text{with} \qquad \boldsymbol{T}(\boldsymbol{U}_n^h) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{U}}(\boldsymbol{U}_n^h) \tag{15.4}$$

being the **tangent matrix**.

For the mechanical problem, this gives (e.g., in finite deformations)

$$\begin{aligned} T_{ik}^{ab}(\boldsymbol{U}_n^h) = \frac{\partial F_i^a}{\partial U_k^b}(\boldsymbol{U}_n^h) &= \frac{\partial}{\partial U_k^b}\left(\int_\Omega P_{iJ}N_{,J}^a\,\mathrm{d}V - F_{\text{ext},i}^a\right) \\ &= \int_\Omega \frac{\partial P_{iJ}}{\partial F_{kL}} N_{,L}^b N_{,J}^a\,\mathrm{d}V - \frac{\partial F_{\text{ext},i}^a}{\partial U_k^b} \\ &= \int_\Omega \mathbb{C}_{iJkL} N_{,L}^b N_{,J}^a\,\mathrm{d}V - \frac{\partial F_{\text{ext},i}^a}{\partial U_k^b}, \end{aligned} \tag{15.5}$$

where $\mathbb{C}_{iJkL}$ is the incremental stiffness tensor (in linearized kinematics, the expression is the same with $\mathbb{C}_{ijkl}$ the linearized stiffness tensor).

Note that the solver requires that $\det\boldsymbol{T} \neq 0$, which is guaranteed in linear elasticity if no rigid body mode exists (i.e., the linearized system has no zero-energy mode so that $\boldsymbol{U}^h \cdot \boldsymbol{T}\boldsymbol{U}^h \neq 0$ for all admissible $\boldsymbol{U}^h$). The Newton-Raphson solver displays *quadratic convergence*.

Note that, if the problem is linear as in linear elasticity, the solver converges in one step since

$$\begin{aligned} \boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \Delta\boldsymbol{U}_n^h &= \boldsymbol{U}_n^h - \boldsymbol{K}^{-1}\left[\boldsymbol{F}_{\text{int}} - \boldsymbol{F}_{\text{ext}}\right] \\ &= \boldsymbol{U}_n^h - \boldsymbol{K}^{-1}\left[\boldsymbol{K}\boldsymbol{U}_n^h - \boldsymbol{F}_{\text{ext}}\right] \\ &= \boldsymbol{U}_n^h - \boldsymbol{U}_n^h + \boldsymbol{K}^{-1}\boldsymbol{F}_{\text{ext}} \\ &= \boldsymbol{K}^{-1}\boldsymbol{F}_{\text{ext}}. \end{aligned} \tag{15.6}$$

15.2 Damped Newton-Raphson (dNR) method

A slight modification of the Newton-Raphson method, the **damped Newton-Raphson method** is beneficial, e.g., when the NR method tends to overshoot (e.g., in case of oscillatory energy landscapes or multiple minima such as in finite-deformation elasticity).

The iterative scheme is identical to the classical NR method except that

$$\boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \alpha\,\Delta\boldsymbol{U}_n^h \qquad \text{with} \qquad \alpha \in (0,1). \tag{15.7}$$

The damping parameter α can be chosen constant or adjusted based on convergence.

15.3 Quasi-Newton (QN) method

The **Quasi-Newton method** is the same as the classical NR method with the exception that one does not use the actual tangent matrix $\boldsymbol{T}$ for computational simplicity or efficiency.

Motivation is thus to avoid the computation of $\boldsymbol{T}(\boldsymbol{U}^h)$ and its inversion at each iteration step. Instead one uses a matrix $\boldsymbol{B}_n$ and updates its inverse directly.

The general algorithm is as follows:

(1) start with an initial guess $\boldsymbol{U}_0^h$ and $\boldsymbol{B}_0 = \boldsymbol{T}(\boldsymbol{U}_0^h)$
(2) compute $\Delta\boldsymbol{U}_n^h = -\boldsymbol{B}_n^{-1}\boldsymbol{f}(\boldsymbol{U}_n^h)$ and $\boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \Delta\boldsymbol{U}_n^h$ and

$$\boldsymbol{B}_{n+1}^{-1} = \boldsymbol{B}_n^{-1} - \frac{\left(\boldsymbol{B}_n^{-1}\boldsymbol{z}_n - \Delta\boldsymbol{U}_n^h\right) \otimes \Delta\boldsymbol{U}_n^h \boldsymbol{B}_n^{-1}}{\Delta\boldsymbol{U}_n^h \cdot \boldsymbol{B}_n^{-1}\boldsymbol{z}_n} \qquad \text{with} \qquad \boldsymbol{z}_n = \boldsymbol{f}(\boldsymbol{U}_{n+1}^h) - f(\boldsymbol{U}_n^h). \tag{15.8}$$

We omit the full derivation of the update for $\boldsymbol{B}_{n+1}$ here for brevity. The idea is that $\boldsymbol{B}_{n+1}^{-1}$ and $\boldsymbol{B}_n^{-1}$ are approximately rank-one-connected using the *Sherman-Morrison formula*. The added benefit is that not only does $\boldsymbol{T}$ not have to be recomputed exactly but also can the inversion or linear solver be skipped since the updated inverse is computed explicitly.

15.4 Line search method

The **line search method** can be used as an improvement for other nonlinear iterative solvers. Similar to the Quasi-Newton schemes, updates are made according to

$$\boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \beta\,\Delta\boldsymbol{U}_n^h, \tag{15.9}$$

where now β is not a constant but chosen such that $\boldsymbol{f}(\boldsymbol{U}_{n+1}^h) = \boldsymbol{0}$. For example, we can find β from solving

$$\Delta\boldsymbol{U}_n^h \cdot \boldsymbol{f}(\boldsymbol{U}_n^h + \beta\,\Delta\boldsymbol{U}_n^h) = 0. \tag{15.10}$$

This is generally a nonlinear but *scalar* problem that can be solved by bisection, regula falsi, secant, and other methods.

Notice that (15.10) is in fact the stationarity condition of the minimization problem

$$\beta = \arg\inf \left\| \boldsymbol{f}(\boldsymbol{U}_n^h + \beta\,\Delta\boldsymbol{U}_n^h) \right\|^2, \tag{15.11}$$

which is the motivation for the *nonlinear least-squares method* described below.

15.5 Gradient flow method

Although not with a proper physical meaning, the **gradient flow method** (also known as *gradient descent*) has become popular as an iterative solver for quasistatic problems.

The idea is to replace the equation

$$\mathbf{0} = \boldsymbol{f}(\boldsymbol{U}^h_{n+1}) \tag{15.12}$$

by a dynamic evolution equation:

$$\boldsymbol{C}\dot{\boldsymbol{U}}^h_{n+1/2} = -\boldsymbol{f}(\boldsymbol{U}^h_n) \qquad \text{and} \qquad \boldsymbol{U}^h_{n+1} = \boldsymbol{U}^h_n + \Delta t\,\dot{\boldsymbol{U}}^h_{n+1/2} \tag{15.13}$$

with, e.g., $\boldsymbol{C} = c\boldsymbol{I}$ with $c > 0$. It is obvious that as $\boldsymbol{f} \to \mathbf{0}$ we have $\dot{\boldsymbol{U}}^h_{n+1/2} \to \mathbf{0}$ and thus the method converges. Although there is no guarantee to reach an extremum, the method is popular because it does not require a tangent matrix and is quite robust.

For example, using a simple backward-Euler discretization for the time derivative and $\boldsymbol{C} = c\boldsymbol{I}$, we obtain

$$\boldsymbol{U}^h_{n+1} = \boldsymbol{U}^h_n - \frac{1}{c}\boldsymbol{f}(\boldsymbol{U}^h_n). \tag{15.14}$$

15.6 Nonlinear Least Squares

The family of methods based on **nonlinear least squares** aim to minimize

$$r(\boldsymbol{U}^h) = \left\|\boldsymbol{f}(\boldsymbol{U}^h)\right\|^2 = \boldsymbol{f}(\boldsymbol{U}^h)\cdot\boldsymbol{f}(\boldsymbol{U}^h) \qquad \text{so we solve} \qquad \frac{\partial r}{\partial \boldsymbol{U}^h} = \mathbf{0}. \tag{15.15}$$

This approach is helpful, e.g., in case of over-constrained systems. Application of Newton-Raphson to this nonlinear system of equations leads to

$$\begin{aligned}\Delta\boldsymbol{U}^h &= -\left[\frac{\partial}{\partial\boldsymbol{U}^h}\frac{\partial r}{\partial\boldsymbol{U}^h}\right]^{-1}_{\boldsymbol{U}^h_n}\frac{\partial r}{\partial\boldsymbol{U}^h}(\boldsymbol{U}^h_n)\\ &= -\left[\boldsymbol{T}^{\mathrm{T}}(U^h_n)\frac{\partial f}{\partial\boldsymbol{U}^h}(U^h_n)\boldsymbol{f}(U^h_n) + \frac{\partial\boldsymbol{T}^{\mathrm{T}}}{\partial\boldsymbol{U}^h}(U^h_n)\boldsymbol{f}(U^h_n)\right]^{-1}\boldsymbol{T}^{\mathrm{T}}(U^h_n)\,\boldsymbol{f}(U^h_n).\end{aligned} \tag{15.16}$$

If updates are small, we can neglect the second term in brackets (which requires higher derivatives than what is commonly computed in FEM), which gives

$$\Delta\boldsymbol{U}^h = -\left[\boldsymbol{T}^{\mathrm{T}}(U^h_n)\boldsymbol{T}(U^h_n)\right]^{-1}\boldsymbol{T}^{\mathrm{T}}(U^h_n)\,\boldsymbol{f}(U^h_n). \tag{15.17}$$

This is known as the **Gauss-Newton method**. Note that this reduces to Newton-Raphson for standard problems (i.e., those with as many equations as unknowns). However, Gauss-Newton can also be applied to overdetermined systems (i.e., more equations than unknowns).

15.7 Conjugate Gradient (CG) method

The **conjugate gradient** method follows the idea of iterating into the direction of steepest descent in order to minimize the total potential energy (as a variation, it can also be applied to the nonlinear least squares problem).

Here, the update is

$$\boldsymbol{U}_{n+1}^h = \boldsymbol{U}_n^h + \alpha_n \boldsymbol{S}_n, \tag{15.18}$$

where both the direction $\boldsymbol{S}_n$ and increment α_n are determined in an optimal way as follows.

The conjugate direction is updated according to

$$\boldsymbol{S}_n = -\boldsymbol{f}(\boldsymbol{U}_n^h) + \beta_n \boldsymbol{S}_{n-1} \tag{15.19}$$

with β computed from the current solution $\boldsymbol{U}_n^h$ and the previous solution $\boldsymbol{U}_n^h$ according to one of several options (Polak-Ribière, Fletcher-Reeves, etc.)

Then, the scalar increment α_n is obtained from a *line search* to find

$$\alpha_n = \arg\min r\left(\boldsymbol{U}_n^h + \alpha\, \boldsymbol{S}_n\right). \tag{15.20}$$

A benefit of the conjugate gradient technique is that, as for gradient flow, *no tangent matrix is required.*

A variation of this scheme, originally developed for atomistics but also applicable to the FE method (and oftentimes converging faster than CG) is the so-called **Fast Inertial Relaxation Engine** (FIRE).

16 Boundary conditions

16.1 Neumann boundary conditions

Recall that the nonlinear system of equations to solve,

$$\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}} = \boldsymbol{0}, \tag{16.1}$$

requires computation of the external force vector. The latter includes both *body forces* and *surface tractions.*

Now that we have introduced isoparametric mappings and numerical quadrature rules, we can apply those to the external force terms.

Body forces $\rho \boldsymbol{b}$ produce the external force on node a in direction i, e.g., in 2D for a single element

$$F_{\text{ext},i}^a = \int_{\Omega_e} \rho b_i N^a \,\mathrm{d}V = \int_{-1}^{1}\int_{-1}^{1} \rho b_i(\boldsymbol{\xi}) N_e^a(\boldsymbol{\xi})\, J(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta \approx \sum_{k=0}^{n_{QP}-1} W_k\, \rho b_i(\boldsymbol{\xi}_k) N_e^a(\boldsymbol{\xi}_k)\, J(\boldsymbol{\xi}_k). \tag{16.2}$$

Surface tractions $\hat{\boldsymbol{t}}$ result in an external force on node a in direction i, again in 2D and for a single element,

$$F_{\text{ext},i}^a = \int_{\partial\Omega_{N,e}} \hat{t}_i N^a \,\mathrm{d}V = \int_{-1}^{1} \hat{t}_i(\boldsymbol{\xi}) N_e^a(\boldsymbol{\xi})\, J(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta \approx \sum_{k=0}^{n_{QP}-1} W_k\, \hat{t}_i(\boldsymbol{\xi}_k) N_e^a(\boldsymbol{\xi}_k)\, J(\boldsymbol{\xi}_k). \tag{16.3}$$

Note that the surface traction term integrates over the boundary of the element, so in d dimensions we can use a quadrature rule for $d-1$ dimensions (e.g., for a 2D element we use 1D quadrature on the element edges).

Implementation:

The variational formulation allows us to implement **external force elements** just like regular finite elements: we have defined their energy,

$$I_e = -\int_{\Omega_e} \rho \boldsymbol{b}\cdot\boldsymbol{u}\,\mathrm{d}V - \int_{\Omega_e} \hat{\boldsymbol{t}}\cdot\boldsymbol{u}\,\mathrm{d}S. \tag{16.4}$$

The corresponding force vectors $\boldsymbol{F}_{\text{ext}}$ are given above, and the tangent matrices are obtained from

$$\boldsymbol{T} = \frac{\partial \boldsymbol{F}_{\text{ext}}}{\partial \boldsymbol{U}_e^h}. \tag{16.5}$$

Notice that if $\hat{\boldsymbol{t}}$ and $\rho\boldsymbol{b}$ are independent of displacements (as in most cases), we have $\boldsymbol{T} = \boldsymbol{0}$. An exception is, e.g., the application of *pressure* to the boundary *in finite deformations* (since the resulting force depends on the surface area which, in turn, depends on the sought deformation).

16.2 Examples of external forces

Constant force:

Consider a constant force $\boldsymbol{P}$ applied to a particular node i at deformed position $\boldsymbol{x}_i$. The element energy of this external force vector is

$$I_e = -\boldsymbol{P}\cdot\boldsymbol{u}^i, \tag{16.6}$$

so that the resulting force vector is

$$\boldsymbol{F}^a = \frac{\partial I_e}{\partial \boldsymbol{u}^a} = -\boldsymbol{P}\,\delta_{ia}, \tag{16.7}$$

i.e., an external force is applied only to node i. The stiffness matrix vanishes since

$$\boldsymbol{T}^{ab} = \frac{\partial \boldsymbol{F}^a}{\partial \boldsymbol{u}^b} = \boldsymbol{0}. \tag{16.8}$$

Linear spring:

Next, consider a linear elastic spring (stiffness k) attached to a node i (deformed position $\boldsymbol{x}_i$, undeformed position $\boldsymbol{X}_i$) and anchored at a constant position $\boldsymbol{x}_0 = \boldsymbol{X}_0$. The element energy in this case is simply the spring energy

$$I_e = \frac{k}{2}\left(\,\|\boldsymbol{x}_i - \boldsymbol{x}_0\| - \|\boldsymbol{X}_i - \boldsymbol{X}_0\|\,\right)^2, \tag{16.9}$$

and force vector and stiffness matrix follow by differentiation.

Indenter:

When simulating indentation tests, it is oftentimes convenient to apply the indenter forces via an external potential rather than by modeling contact. Consider a spherical (in 3D) or circular (in 2D) indenter of radius R whose center is located at the constant, known point $\boldsymbol{x}_0$. Here, one may use potentials of the type

$$I_e = C\left[\,\|\boldsymbol{x}_0 - \boldsymbol{x}_i\| - R\,\right]_-^n \tag{16.10}$$

with a force constant $C > 0$ and integer exponent $n \geq 2$ (a common choice is $n = 3$). The bracket $[\cdot]_- = \min(\cdot, 0)$ implies that the energy is only non-zero if the term in bracket is negative (i.e., if the point enters the indenter radius. Again, forces and stiffness matrix follow by differentiation.

Pressure in linearized kinematics:

Applying a constant pressure p to a surface can be accomplished rather easily in linearized kinematics via (16.3). Specifically, we have for a single element surface

$$I_e = -\int_{\partial\Omega_e} \hat{\boldsymbol{t}}\cdot\boldsymbol{u}^h\,\mathrm{d}S = -\int_{\partial\Omega_e}(-p)\boldsymbol{n}\cdot\sum_{a=1}^{n}\boldsymbol{u}_e^a N_e^a\,\mathrm{d}S = \sum_{a=1}^{n}\int_{\partial\Omega_e} p\boldsymbol{n}\cdot\boldsymbol{u}_e^a N_e^a\,\mathrm{d}S. \tag{16.11}$$

Integration can be carried out using numerical quadrature on the element boundary (the element boundary normal $\boldsymbol{n}$ can be computed from the nodal locations). Again, forces and stiffness matrix follow by differentiation (forces are constant, and the stiffness matrix vanishes).

Pressure in finite kinematics:

Applying a constant pressure p to a surface in finite kinematics is more complicated as the element boundary undergoes finite changes during deformation, resulting in nodal forces that depend on deformation. Here, we start with the work done by the pressure (which we assume *constant*), viz.

$$\begin{aligned} I_e &= -pv = -\int_{\varphi(\Omega_e)} p\,\mathrm{d}v = -\int_{\varphi(\Omega_e)} p\frac{1}{d}x_{i,i}\,\mathrm{d}v \\ &= -\frac{p}{d}\int_{\partial\varphi(\Omega_e)} \varphi_i n_i\,\mathrm{d}s = -\frac{p}{d}\int_{\partial\Omega_e} \varphi_i J F_{Ji}^{-1} N_J\,\mathrm{d}S, \end{aligned} \tag{16.12}$$

where we used that $x_{i,i} = \delta_{ii} = d$ in d dimensions as well as the Piola transform $n_i\,\mathrm{d}s = JF_{Ji}^{-1}N_J\,\mathrm{d}S$. Thus the numerical integration uses

$$I_e = -\frac{p}{d}\int_{\partial\Omega_e} \boldsymbol{u}^h \cdot \boldsymbol{F}^{-\mathrm{T}} \boldsymbol{N} J\,\mathrm{d}S, \tag{16.13}$$

which can be evaluated by numerical quadrature. As before, forces and stiffness matrix follow by differentiation but are non-zero and rather complex.

16.3 Dirichlet boundary conditions

Essential boundary conditions require us to replace individual equations in the nonlinear system by $\boldsymbol{u}^a = \hat{\boldsymbol{u}}^a$. This is accomplished computationally usually in one of two ways.

Substitution:

First, brute-force substitution is a simple method: one replaces the respective equation for Δu_i^a in the linearized system $\boldsymbol{T}\,\Delta\boldsymbol{U}^h = \boldsymbol{F}$ by $\Delta u_i^a = \Delta\hat{u}_i^a$; e.g.,

$$\begin{pmatrix} T_{11} & T_{12} & \dots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 1 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & T_{55} \end{pmatrix} \begin{pmatrix} \Delta u^1 \\ \Delta u^2 \\ \Delta u^3 \\ \Delta u^4 \\ \Delta u^5 \end{pmatrix} = \begin{pmatrix} F^1 \\ F^2 \\ \Delta\hat{u}^3 \\ F^4 \\ F^5 \end{pmatrix} \tag{16.14}$$

For example, when using iterative solvers, one may want to choose the initial guess as $\hat{\boldsymbol{u}}^a$ and subsequently enforce $\Delta\boldsymbol{u}^a = \boldsymbol{0}$ in iterations by using the above condensation method.

Constraints:

The same method discussed above can also impose other types of boundary conditions, e.g., *periodic boundary conditions* of the type

$$\boldsymbol{u}^+ = \boldsymbol{u}^- \tag{16.15}$$

for some opposite nodes $(+,-)$. Also, constraints of the general type

$$f(\boldsymbol{u}^i, \boldsymbol{u}^j, \dots) = 0 \tag{16.16}$$

can be implemented in a similar fashion (e.g., rigid links between nodes).

In case of linear constraints on boundary nodes, it is sometimes convenient to introduce a *reduced* set of degrees of freedom $\boldsymbol{U}^h_{\text{red}}$, which contains only the free degrees of freedom, from which the full set of nodal degrees of freedom can be obtained via

$$\boldsymbol{U}^h = \boldsymbol{A}\boldsymbol{U}^h_{\text{red}}, \tag{16.17}$$

where $\boldsymbol{A}$ is a matrix defined by the constraints. For example, consider a node a being constrained to move along a particular direction $\boldsymbol{n} \in \mathbb{R}^d$. That is, its degrees of freedom $\boldsymbol{u}^a = \{u^a_1, u^a_2\}$ in 2D are reduced[2] to $\boldsymbol{u}^a = \chi^a \boldsymbol{n}$ with the only unknown being $\chi^a \in \mathbb{R}$. E.g., in 2D we have

$$\boldsymbol{U}^h = \begin{pmatrix} \ldots & & & & & & \\ \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & n_1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & n_2 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & n_3 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot \\ & & & & & & \ldots \end{pmatrix} \begin{pmatrix} \vdots \\ u^{a-1}_1 \\ u^{a-1}_2 \\ \chi^a \\ u^{a+1}_1 \\ u^{a+1}_2 \\ \vdots \end{pmatrix} = \boldsymbol{A}\,\boldsymbol{U}^h_{\text{red}} \tag{16.18}$$

The governing equation now becomes

$$\boldsymbol{F}_{\text{int}}\left(\boldsymbol{A}\boldsymbol{U}^h_{\text{red}}\right) - \boldsymbol{F}_{\text{ext}} = \boldsymbol{0}. \tag{16.19}$$

Note that in the special case of a linear problem we know that $\boldsymbol{F}_{\text{int}} = \boldsymbol{T}\boldsymbol{U}^h$, so that (16.19) form can be transformed into a symmetric problem by pre-multiplying by $\boldsymbol{A}^{\text{T}}$, i.e.,

$$\boldsymbol{A}^{\text{T}}\boldsymbol{T}\boldsymbol{A}\boldsymbol{U}^h_{\text{red}} - \boldsymbol{A}^{\text{T}}\boldsymbol{F}_{\text{ext}} = \boldsymbol{0} \qquad \Leftrightarrow \qquad \boldsymbol{T}_{\text{red}}\boldsymbol{U}^h_{\text{red}} - \boldsymbol{F}_{\text{ext,red}} = \boldsymbol{0}, \tag{16.20}$$

which is to be solved for the reduced degrees of freedom. While efficient due to the reduced number of degrees of freedom, the pre- and post-multiplication by $\boldsymbol{A}$ can add considerable numerical expenses.

Notice a convenient relation: if matrix $\boldsymbol{A}$ embeds $\boldsymbol{U}^h_{\text{red}}$ according to (16.17), then the total forces acting on the reduced degrees of freedom are given by $\boldsymbol{A}^{\text{T}}\boldsymbol{F}$; e.g., in the above example:

$$\boldsymbol{F}_{\text{red}} = \boldsymbol{A}^{\text{T}}\boldsymbol{F} = \begin{pmatrix} \ldots & & & & & & & & & \\ \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & n_1 & n_2 & n_3 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot \\ & & & & & & & & & \ldots \end{pmatrix} \boldsymbol{F} = \begin{pmatrix} \vdots \\ \boldsymbol{F}^{a-1} \\ \boldsymbol{F}^a \cdot \boldsymbol{n} \\ \boldsymbol{F}^{a+1} \\ \vdots \end{pmatrix}. \tag{16.21}$$

Recall that in the nonlinear case, one may still solve for the reduced the linearized equation to compute the Newton-Raphson updates is

$$\Delta\boldsymbol{U}^h_{\text{red}} = -(\boldsymbol{T}^*)^{-1}(\boldsymbol{U}^h_{\text{red}})\left[\boldsymbol{F}_{\text{int}}(\boldsymbol{A}\boldsymbol{U}^h_{\text{red}}) - \boldsymbol{F}_{\text{ext}}\right] \qquad \text{where} \qquad \boldsymbol{T}^* = \boldsymbol{A}^{\text{T}}\frac{\partial \boldsymbol{F}_{\text{int}}}{\partial \boldsymbol{U}^h}(\boldsymbol{A}\boldsymbol{U}^h_{\text{red}}). \tag{16.22}$$

There are many alternative ways to introduce constraints such as using the penalty method which appends to the potential energy each constraint with a penalty (Lagrange-type) multiplier. For further information, please refer to the literature.

[2] This is an *embedding* operation: node a is constraint to lie on a line manifold and is embedded into 3D space via embedding. The inverse operation is known as *submerging*.

Condensation:

The substitution method introduced above for essential boundary conditions is simple to implement. However, it is quite expensive since the number of equations remains the same when imposing essential boundary conditions. The **condensation method** removes from the linear system to be solved those equations imposing essential boundary conditions.

Let us rewrite the linearized system by moving the third column to the right-hand side:

$$\begin{pmatrix} T_{11} & T_{12} & 0 & T_{14} & \cdot \\ \cdot & \cdot & 0 & \cdot & \cdot \\ \cdot & \cdot & 0 & \cdot & \cdot \\ \cdot & \cdot & 0 & \cdot & \cdot \\ \cdot & \cdot & 0 & \cdot & T_{55} \end{pmatrix} \begin{pmatrix} \Delta u^1 \\ \Delta u^2 \\ \Delta u^3 \\ \Delta u^4 \\ \Delta u^5 \end{pmatrix} = \begin{pmatrix} F^1 \\ F^2 \\ F^3 \\ F^4 \\ F^5 \end{pmatrix} - \begin{pmatrix} \Delta T_{13} \\ \Delta T_{23} \\ \Delta T_{33} \\ \Delta T_{43} \\ \Delta T_{53} \end{pmatrix} \Delta \hat{u}^3 \qquad (16.23)$$

so that we can eliminate the third row and column from the system:

$$\begin{pmatrix} T_{11} & T_{12} & T_{14} & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & T_{55} \end{pmatrix} \begin{pmatrix} \Delta u^1 \\ \Delta u^2 \\ \Delta u^4 \\ \Delta u^5 \end{pmatrix} = \begin{pmatrix} F^1 \\ F^2 \\ F^4 \\ F^5 \end{pmatrix} - \begin{pmatrix} \Delta T_{13} \\ \Delta T_{23} \\ \Delta T_{43} \\ \Delta T_{53} \end{pmatrix} \Delta \hat{u}^3, \qquad (16.24)$$

and solve for the remaining unknowns along with $\Delta u^3 = \Delta \hat{u}^3$.

The clear advantage of this method is the *reduction in size* of the system to be solved. The disadvantage is that it is computationally more involved (can be even more expensive for a small number of essential boundary conditions applied to large systems).

16.4 Rigid body motion

Consider the composite deformation mapping $\boldsymbol{x} = \varphi^*(\varphi(\boldsymbol{X}))$ with $\boldsymbol{x} = \varphi(\boldsymbol{X})$ being an admissible deformation mapping and $\varphi^*(\boldsymbol{x}) = \boldsymbol{Rx}+\boldsymbol{c}$ denoting rigid body motion, i.e., $\boldsymbol{R} \in SO(d)$ and $\boldsymbol{c} \in \mathbb{R}^d$ is a constant vector. Note that the combined deformation gradient is given by $\boldsymbol{F}^* = \boldsymbol{RF}$ where $\boldsymbol{F} = \operatorname{Grad}\varphi$. Recall that the weak form read

$$\mathcal{G}(u,v) = \int_\Omega P_{iJ}(\boldsymbol{F}) v_{i,J}\,\mathrm{d}V - \int_\Omega RB_i v_i\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{T}_i v_i\,\mathrm{d}S = 0. \qquad (16.25)$$

Insertion of the composite mapping yields

$$\mathcal{G}(u^*,v) = \int_\Omega P_{iJ}(\boldsymbol{RF}) v_{i,J}\,\mathrm{d}V - \int_\Omega RB_i v_i\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{T}_i v_i\,\mathrm{d}S = 0. \qquad (16.26)$$

However, note that material frame indifference requires that $\boldsymbol{P} = \boldsymbol{P}(\boldsymbol{C})$ and $\boldsymbol{C}^* = (\boldsymbol{F}^*)^{\mathrm{T}}\boldsymbol{F}^* = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} = \boldsymbol{C}$, so that the rotation has no affect on the weak form (neither does the translation $\boldsymbol{c}$). Therefore, rigid body motion can be superimposed onto any admissible solution and must be suppressed by appropriate essential boundary conditions to ensure uniqueness of solutions.

For the linear elastic case, the tangent matrix $\boldsymbol{T}$ therefore has as many zero eigenvalues as it has **rigid-body modes**, or **zero-energy modes** $\boldsymbol{U}^*$ such that

$$\boldsymbol{U}^* \cdot \boldsymbol{TU}^* = 0. \qquad (16.27)$$

This implies that $\boldsymbol{T}$ has zero eigenvalues and is thus not invertible.

The remedy is to suppress rigid body modes via appropriate essential boundary conditions; specifically in d dimensions we need $d(d-1)/2$ such essential boundary conditions.

17 Error estimates and adaptivity

17.1 Finite element error analysis

Solving systems of PDEs by the finite element method introduces numerous sources of errors that one should be aware of:

(i) The **discretization error** (also known as the **first fundamental error**) arises from discretizing the domain into elements of finite size h. As a result, the body Ω is not represented correctly and the model (e.g., the outer boundary) may not match the true boundary $\partial\Omega$ (e.g., think of approximating a circular domain Ω by CST or Q4 elements). This error can be reduced by mesh **refinement** (and we discussed r-refinement, h-refinement, p-refinement, and hp-refinement).

(ii) The **numerical integration error** results from the application of numerical quadrature:

$$\int_{\Omega_e} f(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi} \cong \sum_{k=1}^{n_{QP}} W_k\, f(\boldsymbol{\xi}_k) \tag{17.1}$$

We discussed that for $f \in C^{k+1}(\Omega)$ (the extension to higher dimensions is analogous)

$$\left| \int_{-1}^{1} f(\xi)\,\mathrm{d}\xi - \sum_{q=1}^{n_{QP}} W_i\, f(\xi_i) \right| \le C\,\|\Omega\|\, h^{k+1} \max_{\substack{\xi\in[-1,1]\\ |\alpha|=k+1}} \|D^\alpha f(\xi)\|\,. \tag{17.2}$$

Hence, the numerical integration error depends on the smoothness of the integrand and calls for a proper choice of the integration order.

(iii) The **solution error** stems from numerically solving linear systems $\boldsymbol{T}\boldsymbol{U}^h = \boldsymbol{F}$. In general, the accuracy of the solution depends on the **condition number** of the matrix,

$$\kappa = \|\boldsymbol{T}\| \cdot \left\|\boldsymbol{T}^{-1}\right\| = \left|\frac{\lambda_{\max}}{\lambda_{\min}}\right| \tag{17.3}$$

with $\lambda_{\max}$ ($\lambda_{\min}$) being the largest (smallest) eigenvalue of $\boldsymbol{T}$. The higher the condition number, the larger the numerical error.

A practical consequence is the guideline to choose wisely the units of model parameters (such as material constants, domain size features, etc.). For example, when performing a linear elastic simulation, it is advisable to normalize elastic constants by 1 GPa instead of assigning, e.g., $E = 210 \cdot 10^9$ (instead, use $E = 2.1$ and know that your results will be in 100 GPa's).

(iv) An **approximation error** is introduced by approximating the functional space $\mathcal{U}$ (in which to find the solution $u(x)$) by a finite-dimensional subspace $\mathcal{U}^h \subset \mathcal{U}$.

As an example, consider an exact solution $u(x)$ in 1D which is approximated by a piecewise linear polynomial function $u^h(x)$. The bar hanging from the ceiling under its own weight was such an example, for which the solution was found to be exact at the nodes with an error $e(x) = u(x) - u^h(x)$ arising within elements. Therefore, we can find a point z within each element such that

$$\frac{\partial e}{\partial x}(z) = 0 \qquad \text{for} \qquad x_i \le x \le x_{i+1}. \tag{17.4}$$

Consequently, we can expand the error to find the solution at a node as

$$e(x_i) = 0 = e(z) + (x_i - z)\frac{\partial e}{\partial x}(z) + \frac{(x_i - z)^2}{2}\frac{\partial^2 e}{\partial x^2}(z) + O\big((x_i - z)^3\big). \tag{17.5}$$

Using (17.4), we find that

$$e(z) = -\frac{(x_i - z)^2}{2}\frac{\partial^2 e}{\partial x^2}(z) + O(h^3). \tag{17.6}$$

Note that

$$(x_i - z)^2 \le \left(\frac{x_{i+1} - x_i}{2}\right)^2 = \frac{h^2}{4}, \tag{17.7}$$

where h denotes the nodal spacing. Altogether, we have thus shown that the maximum error in an element is bounded by

$$|e(x)|_{\max} \le \frac{h^2}{8} \max_{x_i \le x \le x_{i+1}} \left|\frac{\partial^2 u}{\partial x^2}\right| \tag{17.8}$$

As shown in Appendix C, the above error bound can be significantly generalized. for an interpolation of order k and $u \in H^{k+1}(\Omega)$, we have

$$|u_h - u|_{H^1(\Omega)} \le \frac{h^k}{\pi^k}|u|_{H^{k+1}(\Omega)} \qquad \text{and} \qquad \|u_h - u\|_{H^1(\Omega)} \le c\,h^k\,|u|_{H^{k+1}(\Omega)}, \tag{17.9}$$

using Sobolev norms. Thus the error is again determined by the smoothness of the function to be interpolated; and it is expected to decrease with decreasing element size (as $h \to 0$) – the faster the higher the interpolation order.
Note that special caution is required if stress concentrations of any kind are to be represented (e.g., imagine a linear elastic fracture problem and the issues arising from using linear elements to capture the $1/r^n$-type stress concentration near the crack tip).

(v) A **truncation error** is made by every computer when storing and operating numeric values with only a finite number of digits (e.g., floats, doubles, etc.). This is unavoidable and one should be aware of what this error is (especially when choosing, e.g., solver tolerances). Choosing a solver tolerance in itself produces truncation error because we contend with a solution $\boldsymbol{U}^h$ that satisfies $\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}} = \text{tol.}$ (instead of being zero).

(vi) Finally, no simulation is free of **modeling error**, which refers to the large collection of errors made by the selection of the analytical model to be solved (before starting any numerical approximation). For example, we make choices for an appropriate material model, choose material parameters, boundary conditions, and geometric simplifications including reductions to lower dimensions (e.g., plane strain or plane stress instead of a 3D simulation).

The sum of all of the above error sources makes up the numerical error inherent in every simulation.

17.2 Smoothing and adaptivity

Both for postprocessing of approximate solutions and for mesh adaptivity (discussed below) it is convenient to introduce a **smoothing** scheme that takes piecewise-defined solutions (e.g., the stress and strain fields in case of simplicial elements) and computes smoothed nodal values of the discontinuous quantities.

In case of simplicial elements, the quantities of interest are constant within elements. Here one can define nodal quantities, e.g., as

$$(u^a)^* = \frac{\sum_{j=1}^{n_{\text{nb}}} u_{e,j}/V_{e,j}}{\sum_{j=1}^{n_{\text{nb}}} 1/V_{e,j}}, \tag{17.10}$$

where the element quantities u_e of all n_{nb} neighboring elements meeting at a node a are weighted by the respective element volume V_e. This weighting results in smaller elements (whose quadrature points are closer to the node) having a larger weight than large elements (where quadrature points are far from the node).

In case of higher-order elements such as, e.g., the Q4 element, one can extrapolate element quantities that are defined at the quadrature points $\boldsymbol{\xi}_k$ to the nodes by invoking the definition that

$$u(\boldsymbol{\xi}_k) = \sum_{a=1}^{n} (u^a)^* N^a(\boldsymbol{\xi}_k). \tag{17.11}$$

For example for the Q4 element, there are four nodes and four quadrature points, so that the four equations can be solved for the four nodal values u^a. Once the nodal values are known, one again uses a smoothing relation like (17.10) with the element quantities $u_{e,j}$ replaced by the nodal value u_e^a from the respective element, and element volume V_e replaced by the nodal weight (obtained from extrapolating the quadrature point weights $W_k J_k t$ to the nodes).

When performing adaptive mesh refinement, we need an *error norm* where, as an example, we discuss the **ZZ error estimator** named after its inventors, Zienkiewicz and Zhu. If we use a smoothing scheme like the above, we can define a smoothed, continuous displacement gradient $\nabla\boldsymbol{u}^*$ which is contrasted with the approximate solution $\nabla\boldsymbol{u}^h$, so that one may define the error per element by using the energy norm, viz.

$$\|e(\nabla\boldsymbol{u},\nabla\boldsymbol{u}^*)\|_e^2 = \int_{\Omega_e} W(\nabla\boldsymbol{u}^* - \nabla\boldsymbol{u})\,\mathrm{d}V. \tag{17.12}$$

Note that this definition is not unique and one could also use the L_2-norm

$$\|e\|_e^2 = \int_{\Omega_e} \|\nabla\boldsymbol{u}^* - \nabla\boldsymbol{u}\|\;\mathrm{d}V. \tag{17.13}$$

or any other sensible norm.

In order to define an error estimate, it makes sense to introduce the normalization

$$\eta_e = \frac{\|e(\nabla\boldsymbol{u},\nabla\boldsymbol{u}^*)\|_e}{\|e(\nabla\boldsymbol{u},\boldsymbol{0})\|_e}, \tag{17.14}$$

i.e., the error is divided by the energy in the element so as to not over- or underestimate element errors based on element sizes as well as in case of vast differences in element strain energies. The **mesh refinement criterion** states that an element is refined if

$$\eta_e > \eta_{\text{tol.}} \tag{17.15}$$

with some tolerance $\eta_{\text{tol.}}$ determined as a compromise between accuracy and efficiency. Based on the refinement criterion, a set of elements can be identified after each equilibration which is flagged for refinement.

Needed next is a **mesh refinement algorithm**. For exapmle, a frequent choice for triangular elements is **longest edge bisection**, which identifies the longest edge in an element to be refined and inserts a new node at this edge's midpoint, followed by an update of the element connectivity (removing two existing elements and creating four new elements). Note that this involves some book-keeping since adjacent elements flagged for refinement interfere, so that one needs an algorithmic decision about which elements to refine first and how to handle adjacent elements identified for refinement, etc.

18 Element defects: shear locking and hourglassing

Elements make specific approximations about how displacements, stresses and strains vary inside each element, and these approximations may introduce errors. Aside from the above general numerical errors stemming from the interpolation order, the quadrature rule, and geometric errors, etc., particular elements can have defects, some of which will discussed in the following.

The **bilinear Q4 element** displays a defect when used to simulate beam bending. Consider a rectangular element having side lengths a and b in the x and y directions. If bending moments M are applied on the two vertical edges, then the element is expected to undergo bending.

In an actual (slender) beam, we know from elastic beam theory that the stresses in the beam vary linearly across the thickness, so we may write

$$\sigma_{xx} = -\frac{y}{b}\sigma_{\max}, \qquad \sigma_{yy} = \sigma_{xy} = 0, \tag{18.1}$$

where $\sigma_{\max}$ is the maximum tensile stress reached on the surface of the beam. From Hooke's law we obtain

$$\varepsilon_{xx} = -\frac{\sigma_{xx}}{E} = -\frac{y\sigma_{\max}}{bE}, \qquad \varepsilon_{yy} = -\nu\frac{\sigma_{xx}}{E} = \nu\frac{y\sigma_{\max}}{bE}, \qquad \varepsilon_{xy} = 0. \tag{18.2}$$

Let us try to find this solution using a Q4 element. To this end, we apply horizontal forces to all four nodes of the element, obtained from lumping the distributed beam stresses to the nodes:

$$\begin{aligned} F_{\mathrm{ext},1} &= \int_{\partial\Omega_e} \sigma_{xx} N_e^1 \,\mathrm{d}S = -\int_{-1}^{1}\left(-\frac{y}{b}\sigma_{\max}\right) N_e^1(\xi,\eta)\, tb\,\mathrm{d}\eta \\ &= \int_{-1}^{1} \eta\,\sigma_{\max}\frac{(1-\xi)(1-\eta)}{4} tb\,\mathrm{d}\eta = -\frac{\sigma_{\max}tb}{3} = -F, \end{aligned} \tag{18.3}$$

where we introduced the abbreviation $F = \frac{\sigma_{\max}tb}{3}$ for convenience. Analogously, one obtains

$$F_{\mathrm{ext},2} = F, \qquad F_{\mathrm{ext},3} = -F, \qquad F_{\mathrm{ext},4} = F. \tag{18.4}$$

That is, the bottom edge is stretched, while the top edge is compressed (as in beam bending).

Solving the problem for the case of linear elasticity reduces to a simple linear system of equations:

$$\boldsymbol{K}_e\boldsymbol{U}_e^h = \boldsymbol{F}_{\mathrm{ext}} = \begin{pmatrix} -F \\ 0 \\ F \\ 0 \\ -F \\ 0 \\ F \\ 0 \end{pmatrix} \qquad \Rightarrow \qquad \boldsymbol{U}_e^h = \begin{pmatrix} -U \\ 0 \\ U \\ 0 \\ -U \\ 0 \\ U \\ 0 \end{pmatrix} \qquad \text{with} \qquad U = \frac{2a\sigma_{\max}}{E}\frac{(1+\nu)(1-2\nu)}{2(1-\nu)+(a\mathrm{n})^2(1-2\nu)}, \tag{18.5}$$

where we evaluated all element forces and stiffness matrices exactly (without quadrature errors).

The resulting strains are evaluated as

$$\varepsilon_{xx} = -\frac{U}{a}\eta, \qquad \varepsilon_{yy} = 0, \qquad \varepsilon_{xy} = -\frac{U}{b}\xi. \tag{18.6}$$

While the axial strain component is reasonable, the presence of a shear stress is problematic. This so-called **parasitic shear** is not physical but a numerical artifact (elastic Euler beams produce no shear stresses). This parasitic shear contributes to the energy of the element, so that – when minimizing the potential energy in a boundary value problem – this shear component introduces artificial stiffness. This becomes apparent when calculating, e.g., the angle of bending θ of the beam in the above problem. Comparing the exact one (obtained form beam theory) to the approximate one (obtained from the above construction), one finds

$$\frac{\theta_{\text{approx}}}{\theta_{\text{exact}}} = \frac{1-\nu^2}{1+\frac{1-\nu}{2}\left(\frac{a}{b}\right)^2}. \tag{18.7}$$

Notice that, as the Q4 element becomes more and more slender ($a/b \to \infty$), the numerically obtained angle approaches 0. That is, the element shows what is known as **locking** or, more specifically, **shear locking**: in case of very slender elements ($a/b \ll 1$) the high shear strain and associated elastic energy prevents the element from deformting, it "locks".

Next, let us use numerical quadrature to evaluate the element quantities of interest. For example, for the energy we have

$$I_e = \int_{\Omega_e} \frac{1}{2} \boldsymbol{\varepsilon}^h \cdot \mathbb{C}\, \boldsymbol{\varepsilon}^h \,\mathrm{d}V \approx \frac{t}{2} \sum_{k=0}^{n_{QP}-1} W_k\, \boldsymbol{\varepsilon}^h(\boldsymbol{\xi}_k) \cdot \mathbb{C}\, \boldsymbol{\varepsilon}^h(\boldsymbol{\xi}_k)\, J(\boldsymbol{\xi}_k). \tag{18.8}$$

We showed before that full integration requires $n_{QP} = 2 \times 2$ quadrature points. If instead we use reduced integration with only a single quadrature point ($n_{QP} = 1$) located at $\boldsymbol{\xi}_0 = \mathbf{0}$, then notice that strain fields of the form (18.6) – which vanish at $\xi = \eta = 0$ – produce no energy. In other words, any strain field of the form (18.6) can appear without causing any energy and therefore also without causing any resistance. This is a **zero-energy mode** of the under-integrated Q4 element and a serious defect. The resulting deformation of elements undergoing alternating strains (18.6) is a numerical artifact and often referred to as **hourglass mode** or **chicken-wire mode** because of its appearance.

Note that note only the underintegrated Q4 element has such a zero-energy mode. For example, the Q8/Q9 elements have a similar zero-energy mode with curved element edges.

Finally, it is possible to use **selective integration**, which applies different quadrature rules for different energy contributions. For example, integrating the Q4 element with a 2 × 2 quadrature rule for the nominal strains ε_{xx} and ε_{yy} while using reduced integration with a single quadrature point at $\boldsymbol{\xi}_k = \mathbf{0}$ removes the spurious shear contribution while maintaining the correct stiffness against axial deformation. Notice that, however, selective integration is harder to implement, not directly applicable beyond linear elasticity, and should be used with caution.

19 Dynamics

19.1 Variational setting

The mechanical problems so far have been limited to quasistatic conditions. Let us consider the extension to dynamic problems where inertial effects matter and the strong form reads (for simplicity stated in linearized kinematics; the finite-deformation setting is analogous)

$$\left[\begin{array}{rl} \sigma_{ij,j} + \rho\, b_i = \rho\, a_i & \text{in } \Omega \\ u_i(\boldsymbol{x},t) = \hat{u}_i(\boldsymbol{x},t) & \text{on } \partial\Omega_D \\ \sigma_{ij} n_j(\boldsymbol{x},t) = \hat{t}(\boldsymbol{x},t) & \text{on } \partial\Omega_N \end{array}\right. \tag{19.1}$$

Now, we have $u : \Omega \times \mathbb{R} \to \mathbb{R}^d$ with sufficient differentiability in both space and time.

The variational approach here makes use of the so-called **action principle** which uses the **action**

$$\mathcal{A}[u] = \int_{t_1}^{t_2} \mathcal{L}[u]\,\mathrm{d}t \qquad \text{with} \qquad \mathcal{L}[u] = T[u] - I[u], \tag{19.2}$$

where I is the potential energy functional from before and T the kinetic energy functional:

$$T[u] = \int_\Omega \frac{\rho}{2} |\dot{\boldsymbol{u}}|^2 \,\mathrm{d}V. \tag{19.3}$$

The **action principle** states that the solution $\boldsymbol{u}(\boldsymbol{x},t)$ renders $\mathcal{A}$ stationary with $\boldsymbol{u}(\boldsymbol{x},t_1) = \boldsymbol{u}_1(\boldsymbol{x})$ and $\boldsymbol{u}(\boldsymbol{x},t_2) = \boldsymbol{u}_2(\boldsymbol{x})$.

For variational material models (with W replaced by W^* for inelasticity), we have

$$\mathcal{A}[u] = \int_{t_1}^{t_2} \left[\int_\Omega \left(\frac{\rho}{2} |\dot{\boldsymbol{u}}|^2 - W(\boldsymbol{\varepsilon}) \right) \mathrm{d}V + \int_\Omega \rho \boldsymbol{b} \cdot \boldsymbol{u} \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{\boldsymbol{t}} \cdot \boldsymbol{u} \,\mathrm{d}S \right] \mathrm{d}t. \tag{19.4}$$

Taking the first variation (with the divergence theorem and the same assumptions from before):

$$\delta\mathcal{A}[u] = 0 = \int_{t_1}^{t_2} \left[\int_\Omega (\rho\, \dot{u}_i\, \delta\dot{u}_i - \sigma_{ij} \delta u_{i,j}) \,\mathrm{d}V + \int_\Omega \rho b_i \delta u_i \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i \delta u_i \,\mathrm{d}S \right] \mathrm{d}t. \tag{19.5}$$

The **weak form** is thus obtained as

$$G(u,v) = \int_{t_1}^{t_2} \left[\int_\Omega (\rho\, \dot{u}_i\, \dot{v}_i - \sigma_{ij} v_{i,j}) \,\mathrm{d}V + \int_\Omega \rho b_i v_i \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i v_i \,\mathrm{d}S \right] \mathrm{d}t = 0 \tag{19.6}$$

with

$$\boldsymbol{v} \in \left\{ \boldsymbol{v} \in H^1(\Omega) \;:\; \boldsymbol{v} = \boldsymbol{0} \text{ on } \partial\Omega_D \text{ and at } t = t_1 \text{ or } t = t_2 \right\}. \tag{19.7}$$

To avoid the time derivative in the variation, let us integrate by parts in time (the “boundary term” vanishes since $\boldsymbol{v} = 0$ at $t = t_1$ and $t = t_2$):

$$G(u,v) = -\int_{t_1}^{t_2} \left[\int_\Omega (\rho\, \ddot{u}_i\, v_i + \sigma_{ij} v_{i,j}) \,\mathrm{d}V - \int_\Omega \rho b_i v_i \,\mathrm{d}V - \int_{\partial\Omega_N} \hat{t}_i v_i \,\mathrm{d}S \right] \mathrm{d}t = 0. \tag{19.8}$$

Note that, without the first term, we recover the elastostatic formulation.

Since in the dynamic problem the displacement field depends on time, we introduce a **semi-discretization**, i.e., we discretize the solution in space but not in time:

$$\boldsymbol{u}^h(\boldsymbol{x},t) = \sum_{a=1}^{n} \boldsymbol{u}^a(t) N^a(\boldsymbol{x}) \qquad \text{and} \qquad \boldsymbol{v}^h(\boldsymbol{x},t) = \sum_{a=1}^{n} \boldsymbol{v}^a(t) N^a(\boldsymbol{x}), \tag{19.9}$$

so that

$$\dot{\boldsymbol{u}}^h(\boldsymbol{x},t) = \sum_{a=1}^{n} \dot{\boldsymbol{u}}^a(t) N^a(\boldsymbol{x}) \qquad \text{and} \qquad \ddot{\boldsymbol{u}}^h(\boldsymbol{x},t) = \sum_{a=1}^{n} \ddot{\boldsymbol{u}}^a(t) N^a(\boldsymbol{x}). \tag{19.10}$$

Insertion into the weak form results in Galerkin's **discrete weak form**:

$$\begin{aligned}\mathcal{G}(\boldsymbol{u}^h,\boldsymbol{v}^h) = -\int_{t_1}^{t_2} \sum_{a=1}^{n}\sum_{b=1}^{n} \Big[& \ddot{u}_i^a v_i^b \int_\Omega \rho\, N^a N^b \,\mathrm{d}V + v_i^b \int_\Omega \sigma_{ij} N^b_{,j} \,\mathrm{d}V \\ & -v_i^b \int_\Omega \rho b_i N^b \,\mathrm{d}V - v_i^b \int_{\partial\Omega_N} \hat{t}_i N^b \,\mathrm{d}S \Big] \,\mathrm{d}t = 0\end{aligned} \tag{19.11}$$

for all $\boldsymbol{v}^b(t)$ histories that vanish at t_1 and t_2.

Analogously to before, we now write

$$\boldsymbol{U}^h(t) = \{\boldsymbol{u}^1(t),\dots,\boldsymbol{u}^n(t)\}, \tag{19.12}$$

so that solving (19.11) for all $\boldsymbol{v}^b(t)$ is equivalent to solving

$$\boldsymbol{M}\,\ddot{\boldsymbol{U}}^h + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}}(t) = \boldsymbol{0} \tag{19.13}$$

with

$$M_{ij}^{ab} = \delta_{ij} \int_\Omega \rho\, N^a N^b \mathrm{d}V, \qquad F^b_{\text{int},i} = \int_\Omega \sigma_{ij} N^b_{,j} \mathrm{d}V, \qquad F^b_{\text{ext},i} = \int_\Omega \rho b_i N^b \,\mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i N^b \mathrm{d}S. \tag{19.14}$$

Matrix $\boldsymbol{M}$ is called the **consistent mass matrix**.

Examples:

The consistent mass matrix for a *two-node bar element* is computed from shape functions

$$N_1(\xi) = \frac{1-\xi}{2}, \qquad N_2(\xi) = \frac{1+\xi}{2}. \tag{19.15}$$

Specifically, we have (with $m = \rho A L_e$)

$$M^{ab} = \int_\Omega \rho N^a N^b \mathrm{d}V = \int_{-1}^{1} \rho N^a N^b A \frac{L_e}{2} \,\mathrm{d}xi \qquad \Rightarrow \qquad \boldsymbol{M}_{1\text{D}} = \frac{m}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}. \tag{19.16}$$

Note that this is the consistent mass matrix for 1D motion. If each node moves has two degrees of freedom (u_1,u_2) in the plane, then each pair of dof is linked by the above mass matrix, so that the total consistent mass matrix becomes

$$\boldsymbol{M}_{2\text{D}} = \frac{m}{6}\begin{pmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \end{pmatrix}. \tag{19.17}$$

Similarly, the consistent mass matrix of the CST is computed by integration of the shape functions, resulting for 1D motion in

$$\boldsymbol{M}_{\text{CST}} = \frac{m}{12}\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \qquad \text{with} \qquad m = \rho A t. \tag{19.18}$$

As before, the corresponding mass matrix for 2D motion is obtained by applying the above matrix for each dof independently.

For other (non-simplicial) elements, the stiffness matrix can be evaluated analogously or computed by *numerical quadrature*:

$$M^{ab} = \int_\Omega \rho N^a N^b \mathrm{d}V \approx \sum_{k=0}^{n_{QP}-1} W_k \rho N^a(\boldsymbol{\xi}_k) N^b(\boldsymbol{\xi}_k) J(\boldsymbol{\xi}_k). \tag{19.19}$$

In summary, the dynamic problem is quite analogous to the quasistatic one. The key difference is the first term in (19.13), which requires a strategy to obtain numerical solutions that are time-dependent. Note that the above formulation in linearized kinematics can easily be adopted for **finite deformations** (the final matrix equations are the same with internal/external force vectors replaced by the finite-deformation counterparts).

We note that various references call this dynamic variational principle the "*principle of least action*" or "*principle of minimum action*", which is in fact not correct since the solution must not necessarily be a minimizer of $\mathcal{A}$ (it is merely guaranteed to be an extremizer).

The consistent mass matrix above is *dense*, which may be inconvenient for numerical solutions. Therefore, one often resorts to the so-called *lumped mass matrix* which is an approximation that is diagonal. For example, by using *particle-mass lumping* for a 2-node bar element, one distributes the mass evenly to the two end points, resulting in

$$\boldsymbol{M}_{\text{1D}} = \frac{m}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad \boldsymbol{M}_{\text{2D}} = \frac{m}{2}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{19.20}$$

Note that a comparison of the kinetic energies (e.g., in 1D) reveals

$$T_{\text{lumped}} = \frac{1}{2}\dot{\boldsymbol{U}}_e^h \cdot \boldsymbol{M}_{\text{lumped}} \dot{\boldsymbol{U}}_e^h = \frac{m}{4}\left[(\dot{u}_e^1)^2 + (\dot{u}_e^2)^2\right] \tag{19.21}$$

and

$$T_{\text{consistent}} = \frac{1}{2}\dot{\boldsymbol{U}}_e^h \cdot \boldsymbol{M}_{\text{consistent}} \dot{\boldsymbol{U}}_e^h = \frac{m}{6}\left[(\dot{u}_e^1)^2 + (\dot{u}_e^2)^2 + \dot{u}_e^1 \dot{u}_e^2\right]. \tag{19.22}$$

Hence, the lumped mass matrix yields to an approximation of the kinetic energy (which is generally not exact).

Particle-mass lumping can be extended to arbitrary elements having n nodes by defining

$$\boldsymbol{M} = \frac{m}{n}\boldsymbol{I}. \tag{19.23}$$

Finally, note that in *structural dynamics*, one often includes velocity-proportional damping through a **damping matrix** $\boldsymbol{C}$ such that (19.13) turns into

$$\boldsymbol{M}\ddot{\boldsymbol{U}}^h + \boldsymbol{C}\dot{\boldsymbol{U}}^h + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) - \boldsymbol{F}_{\text{ext}}(t) = \boldsymbol{0} \tag{19.24}$$

with, oftentimes, mass- and stiffness-proportional damping via

$$\boldsymbol{C} = \alpha \boldsymbol{M} + \beta \boldsymbol{K}, \qquad \alpha, \beta \in \mathbb{R}_+. \tag{19.25}$$

The choice of $\alpha > 0$ controls low-frequency vibration attenuation, while $\beta > 0$ suppresses high-frequency vibrations.

19.2 Free vibrations

Free vibrations of infinitesimal amplitude are a frequent case of interest. Starting with the general, nonlinear equations of motion,

$$\boldsymbol{M}\,\ddot{\boldsymbol{U}}^h + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^h) = \boldsymbol{F}_{\text{ext}}(t), \tag{19.26}$$

we linearize about a stable equilibrium configuration $\boldsymbol{U}_0^h$ with constant forces $\boldsymbol{F}_{\text{ext}} = \boldsymbol{F}_{\text{ext},0}$ such that $\boldsymbol{F}_{\text{int}}(\boldsymbol{U}_0^h) = \boldsymbol{F}_{\text{ext},0}$. Consider now a small time-varying perturbation $\delta\boldsymbol{U}^h(t)$ such that $\boldsymbol{U}^h = \boldsymbol{U}_0^h + \delta\boldsymbol{U}^h$ gives to leading order

$$\boldsymbol{M}\,\delta\ddot{\boldsymbol{U}}^h + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}_0^h) + \frac{\partial \boldsymbol{F}_{\text{int}}}{\partial \boldsymbol{U}^h}(\boldsymbol{U}_0^h)\,\delta\boldsymbol{U}^h + \text{h.o.t.} = \boldsymbol{F}_{\text{ext},0} \tag{19.27}$$

or, invoking equilibrium,

$$\boldsymbol{M}\,\delta\ddot{\boldsymbol{U}}^h + \boldsymbol{T}(\boldsymbol{U}_0^h)\,\delta\boldsymbol{U}^h + \text{h.o.t.} = \boldsymbol{0}. \tag{19.28}$$

Similarly, when considering free vibrations about the undeformed ground state of an elastic body, we would have arrived immediately at

$$\boldsymbol{M}\,\ddot{\boldsymbol{U}}^h + \boldsymbol{K}\,\boldsymbol{U}^h = \boldsymbol{0}. \tag{19.29}$$

Thus, (19.28) is the generalized form of (19.29) for equilibrated systems. In both cases the form of the equation of motion governing free vibrations without external forcing and without damping is

$$\boldsymbol{M}\,\ddot{\boldsymbol{U}}^h + \boldsymbol{T}\boldsymbol{U}^h = \boldsymbol{0}. \tag{19.30}$$

The solution to the above ODE is of the separable form

$$\boldsymbol{U}^h = \hat{\boldsymbol{U}}^h \exp(i\omega t). \tag{19.31}$$

Insertion into (19.30) leads to (exploiting that the equation must hold for all times t)

$$\left(\boldsymbol{T} - \omega^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}^h = \boldsymbol{0}, \tag{19.32}$$

which is an **eigenvalue problem** with **eigenfrequency** ω and **eigenvector** $\hat{\boldsymbol{U}}^h$. The latter is also referred to as the **eigenmode**.

For a FE discretization with $n_{\text{dof}} = dn_n$ degrees of freedom (n_n nodes in d dimensions), the eigenvalue problem has n_{dof} eigenfrequencies with n_{dof} associated distinct eigenmodes. Each rigid body mode of the FE problem corresponds to a zero eigenfrequency. Therefore, a 2D (3D) free vibrational problem without essential BCs has 3 (6) *zero eigenfrequencies.*

Example: free vibration of a bar

We can use the exampe of a freely vibrating bar to assess the influence of the different mass matrices. Consider a 2-node bar element with only axial displacements, so that each node has only a single degree of freedom u_i. The mass matrices and the stiffness matrix for this case were derived previously as

$$\boldsymbol{M}_{\text{consistent}} = \frac{m}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \qquad \boldsymbol{M}_{\text{lumped}} = \frac{m}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad \boldsymbol{K} = \frac{EA}{L}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}. \tag{19.33}$$

In both cases of the lumped and consistent mass matrices, we compute the two eigenfrequencies and eigenmodes by solving the eigenvalue problem

$$(\boldsymbol{K} - \omega^2 \boldsymbol{M})\hat{\boldsymbol{U}}^h = \boldsymbol{0} \qquad \Rightarrow \qquad \det\left(\boldsymbol{K} - \omega^2 \boldsymbol{M}\right) = 0. \tag{19.34}$$

Insertion of the stiffness and consistent mass matrix results in the two solutions

$$\omega_0^{\text{consistent}} = 0, \qquad \omega_1^{\text{consistent}} = \sqrt{12\frac{EA}{mL}} \approx 3.464\sqrt{\frac{EA}{mL}}. \tag{19.35}$$

The corresponding eigenvectors follow from insertion of the eigenfrequencies into the eigenvalue problem, giving

$$\hat{\boldsymbol{U}}_0^{\text{consistent}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \hat{\boldsymbol{U}}_1^{\text{consistent}} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{19.36}$$

As expected, we have one zero eigenfrequency associated with rigid body translation. When repeating the above procedure with the lumped mass matrix, we instead obtain

$$\omega_0^{\text{lumped}} = 0, \qquad \omega_1^{\text{lumped}} = 2\sqrt{\frac{EA}{mL}}. \tag{19.37}$$

and

$$\hat{\boldsymbol{U}}_0^{\text{lumped}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \hat{\boldsymbol{U}}_1^{\text{lumped}} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{19.38}$$

Hence, the two cases yield the same eigenmodes but significantly differ in the fundamental frequency. For comparison, let us compute the exact solution by studying the free vibration of a continuous, homogeneous, linear elastic bar. Linear momentum balance, i.e.,

$$E u_{,xx} = \rho \ddot{u}, \tag{19.39}$$

admits the separable solution

$$u(x,t) = \hat{u}(x)\exp(i\omega t) \qquad \Rightarrow \qquad E\hat{u}_{,xx}(x) = -\omega^2 \rho \hat{u}(x) \tag{19.40}$$

Free-end boundary conditions imply that

$$\hat{u}_{,x}(0) = \hat{u}_{,x}(L) \qquad \Rightarrow \qquad \hat{u}(x) = A\cos\left(\frac{n\pi x}{L}\right) \tag{19.41}$$

with an integer n and a constant $A \in \mathbb{R}$. Insertion into (19.39) leads to

$$-E\left(\frac{n\pi}{L}\right)^2 = -\omega^2 \rho \qquad \Rightarrow \qquad \omega_n = n\,\pi\sqrt{\frac{EA}{mL}}. \tag{19.42}$$

Consequently, the eigenfrequencies are obtained as

$$\omega_0 = 0, \qquad \omega_1 = \pi\sqrt{\frac{EA}{mL}}. \tag{19.43}$$

Comparison with the above two cases results in

$$\omega_1^{\text{lumped}} \le \omega_1 \le \omega_1^{\text{consistent}}. \tag{19.44}$$

Note that the eigenfrequency obtained from the consistent mass matrix is an upper bound on the eigenfrequency, while the lumped mass matrix is usually a lower bound (the latter is not rigorous, though, since it depends on the choice of the lumped mass matrix).

19.3 Modal decomposition

Let us introduce a few important features of the eigenmodes discussed above. By considering the eigenvalue problem for two distinct eigenmodes/-frequencies, we may write

$$\begin{aligned} \left(\boldsymbol{T} - \omega_i^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_i^h = \boldsymbol{0} \qquad &\Rightarrow \qquad \hat{\boldsymbol{U}}_j^h \cdot \left(\boldsymbol{T} - \omega_i^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_i^h = \boldsymbol{0},\\ \left(\boldsymbol{T} - \omega_j^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_j^h = \boldsymbol{0} \qquad &\Rightarrow \qquad \hat{\boldsymbol{U}}_i^h \cdot \left(\boldsymbol{T} - \omega_j^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_j^h = \boldsymbol{0}, \end{aligned} \tag{19.45}$$

where we simply pre-multiplied each of the two equations by the respective other eigenvector. Subtraction of the two equations (using that $\boldsymbol{T}$ is by definition symmetric) results in

$$(\omega_i^2 - \omega_j^2)\,\hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_j^h = 0. \tag{19.46}$$

This implies that either $\omega_i = \omega_j$ (considering only positive eigenfrequencies) or $\hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_j^h$. If we eliminate duplicated eigenfrequencies by, e.g., Gram-Schmid orthonormalization, then we may conclude that

$$\hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_j^h = 0 \qquad \text{if} \qquad i \neq j. \tag{19.47}$$

We can normalize the eigenvectors in the following fashion:

$$\hat{\boldsymbol{U}}_i^h \leftarrow \frac{\hat{\boldsymbol{U}}_i^h}{\sqrt{\hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_i^h}} \qquad \Rightarrow \qquad \hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_i^h = 1, \tag{19.48}$$

so that overall

$$\hat{\boldsymbol{U}}_i^h \cdot \boldsymbol{M}\,\hat{\boldsymbol{U}}_j^h = \delta_{ij}. \tag{19.49}$$

If we now consider

$$\left(\boldsymbol{T} - \omega_i^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_i^h = \boldsymbol{0} \qquad \Rightarrow \qquad \hat{\boldsymbol{U}}_j^h \cdot \left(\boldsymbol{T} - \omega_i^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_i^h = 0, \tag{19.50}$$

then we may also conclude that, by invoking (19.49),

$$\hat{\boldsymbol{U}}_j^h \cdot \boldsymbol{T}\,\hat{\boldsymbol{U}}_i^h = 0 \qquad \text{if} \qquad i \neq j. \tag{19.51}$$

Similarly, by pre-multiplying by the respective same eigenvector, i.e.,

$$\hat{\boldsymbol{U}}_i^h \cdot \left(\boldsymbol{T} - \omega_i^2 \boldsymbol{M}\right)\hat{\boldsymbol{U}}_i^h = 0, \tag{19.52}$$

we solve for ω_i^2 and obtain **Rayleigh's quotient**, which here simplifies due to the normalization:

$$\omega_i^2 = \frac{\hat{\boldsymbol{U}}^h_{(i)} \cdot \boldsymbol{T}\hat{\boldsymbol{U}}^h_{(i)}}{\hat{\boldsymbol{U}}_{(i)} \cdot \boldsymbol{M}\hat{\boldsymbol{U}}^h_{(i)}} = \hat{\boldsymbol{U}}^h_{(i)} \cdot \boldsymbol{T}\hat{\boldsymbol{U}}^h_{(i)}. \tag{19.53}$$

This all forms the basis for the method known as **modal decomposition**.

The starting point for modal decomposition of *linearized*, *elastic* system is the Fourier representation

$$\boldsymbol{U}^h(t) = \sum_{i=1}^{n} z_i(t)\,\hat{\boldsymbol{U}}^h_i, \tag{19.54}$$

where $\{\hat{\boldsymbol{U}}^h_1, \ldots, \hat{\boldsymbol{U}}^h_n\}$ are the n eigenmodes of the system. That is, we pre-compute the eigenvectors and seek a solution as a linear superposition of all eigenvectors with some unknown scalar coefficients that are continuous functions of time (maintaining the *semi*-discretization).

We substitute (19.54) into the linearized equations of motion with external forces, $\boldsymbol{M}\ddot{\boldsymbol{U}}^h + \boldsymbol{T}\boldsymbol{U}^h = \boldsymbol{F}_{\text{ext}}$, and pre-multiply the system of equations by $\hat{\boldsymbol{U}}^h_i$, so we arrive at

$$\sum_{i=1}^{n} \left[\ddot{z}_i(t)\,\boldsymbol{M}\hat{\boldsymbol{U}}^h_i + z_i(t)\boldsymbol{T}\hat{\boldsymbol{U}}^h_i\right] = \boldsymbol{F}_{\text{ext}}(t). \tag{19.55}$$

Pre-multiplying by $\boldsymbol{U}^h_j$ and exploiting the above orthogonality gives

$$\sum_{i=1}^{n} \left[\ddot{z}_i(t)\,\hat{\boldsymbol{U}}^h_j \cdot \boldsymbol{M}\hat{\boldsymbol{U}}^h_i + z_i(t)\hat{\boldsymbol{U}}^h_j \cdot \boldsymbol{T}\hat{\boldsymbol{U}}^h_i\right] = \hat{\boldsymbol{U}}^h_j \cdot \boldsymbol{F}_{\text{ext}}(t) \qquad \Leftrightarrow \qquad \ddot{z}_j(t) + \omega_j^2 z_j(t) = \hat{\boldsymbol{U}}^h_j \cdot \boldsymbol{F}_{\text{ext}}(t), \tag{19.56}$$

where no summation over j is implied. This equation is quite remarkable as it is a *scalar*, *linear* ODE for the unknown function $z_j(t)$. Moreover, the system of equations for n unknown equations $z_i(t)$ (with $i = 1, \ldots, n$) has decoupled into n uncoupled scalar ODEs to be solved independently for the $z_i(t)$. The strategy is thus to first pre-compute all eigenfrequencies ω_i and all eigenmodes $\hat{\boldsymbol{U}}^h_j$, so that the coefficients and right-hand sides are known in

$$\ddot{z}_i(t) + \omega_i^2 z_i(t) = \hat{\boldsymbol{U}}^h_i \cdot \boldsymbol{F}_{\text{ext}}(t), \tag{19.57}$$

which can relatively inexpensively be solved for functions $z_i(t)$.

For many practical problems, only a limited number of modes are important (i.e., taking the first $m < n$ modes only). Therefore, numerical efficiency can be gained by truncating the Fourier sum, which is often referred to as **order reduction**:

$$\boldsymbol{U}^h(t) = \sum_{i=1}^{m} z_i(t)\,\hat{\boldsymbol{U}}^h_i, \qquad m < n. \tag{19.58}$$

19.4 Transient time-dependent solutions

The time-dependent solution for the nodal variables $\boldsymbol{U}^h(t)$ is obtained either in a continuous manner (e.g., by modal decomposition or in case of free vibrations as discussed above) or in a time-discretized fashion (e.g., by using finite-difference approximations in time). The turns the *semi*-discretization into a proper discretization in both space and time.

19.5 Explicit time integration

We discretize the solution in time, e.g., we assume constant time increments $\Delta t > 0$ and write

$$\boldsymbol{u}^a(t^\alpha) = \boldsymbol{u}^{a,\alpha}, \qquad \Delta t = t^{\alpha+1} - t^\alpha. \tag{19.59}$$

By using *central-difference approximations*, we obtain

$$\dot{\boldsymbol{U}}^h(t^\alpha) = \frac{\boldsymbol{U}^{h,\alpha+1} - \boldsymbol{U}^{h,\alpha-1}}{2\Delta t}, \qquad \ddot{\boldsymbol{U}}^h(t^\alpha) = \frac{\boldsymbol{U}^{h,\alpha+1} - 2\boldsymbol{U}^{h,\alpha} + \boldsymbol{U}^{h,\alpha-1}}{(\Delta t)^2}. \tag{19.60}$$

Insertion into (19.24) leads to

$$\boldsymbol{M}\frac{\boldsymbol{U}^{h,\alpha+1} - 2\boldsymbol{U}^{h,\alpha} + \boldsymbol{U}^{h,\alpha-1}}{(\Delta t)^2} + \boldsymbol{C}\frac{\boldsymbol{U}^{h,\alpha+1} - \boldsymbol{U}^{h,\alpha-1}}{2\Delta t} + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{h,\alpha}) - \boldsymbol{F}_{\text{ext}}(t^\alpha) = \boldsymbol{0}, \tag{19.61}$$

which can be reorganized into

$$\left[\frac{\boldsymbol{M}}{(\Delta t)^2} + \frac{\boldsymbol{C}}{2\Delta t}\right]\boldsymbol{U}^{\alpha+1} = \frac{2\boldsymbol{M}}{(\Delta t)^2}\boldsymbol{U}^\alpha + \left[\frac{\boldsymbol{C}}{2\Delta t} - \frac{\boldsymbol{M}}{(\Delta t)^2}\right]\boldsymbol{U}^{\alpha-1} - \boldsymbol{F}_{\text{int}}\left(\boldsymbol{U}^{h,\alpha}\right) + \boldsymbol{F}_{\text{ext}}(t^\alpha). \tag{19.62}$$

This is an update rule for $\boldsymbol{U}^{\alpha+1}$, using **explicit time integration**. Note that here and in the following we drop the superscript h for simplicity and to avoid confusion with the time step superscript α.

Note that stability limits the time step of the explicit scheme. Specifically, we must ensure that

$$\Delta t \le \Delta t_{\text{cr}} = \frac{2}{\omega_{\max}}, \tag{19.63}$$

with $\omega_{\max}$ being the highest eigenfrequency. Note that the highest eigenfrequency scales inversely proportional to the smallest element size. For example, recall the two-node bar for which the eigenfrequency was of the form $\omega_1 \propto \frac{1}{L}\sqrt{E/\rho}$ with L being the element size.

19.6 A reinterpretation of finite differences

Instead of introducing finite differences as done above, we could alternatively use interpolation functions in both space and time. For example, consider a discretization in time which evaluates $\boldsymbol{U}^h$ at discrete time intervals Δt and then uses a quadratic interpolation in time, i.e., we define

$$\boldsymbol{U}^h(t) = \boldsymbol{U}^{\alpha+1}N^{\alpha+1}(t) + \boldsymbol{U}^\alpha N^\alpha(t) + \boldsymbol{U}^{\alpha-1}N^{\alpha-1}(t) \qquad \text{for} \quad t \in [t^\alpha - \tfrac{\Delta t}{2}, t^\alpha + \tfrac{\Delta t}{2}]. \tag{19.64}$$

where we dropped the superscript h for simplicity. Shape functions $N(t)$ interpolate in time. The chosen quadratic interpolation and the range of validity ensure that $\boldsymbol{U}^h(t)$ is twice differentiable (as needed for the acceleration) if

$$N^{\alpha+1}(t) = \frac{(t-t^\alpha)(t-t^{\alpha-1})}{2(\Delta t)^2}, \quad N^\alpha(t) = \frac{(t^{\alpha+1}-t)(t-t^{\alpha-1})}{(\Delta t)^2}, \quad N^{\alpha-1}(t) = \frac{(t^{\alpha+1}-t)(t^\alpha-t)}{2(\Delta t)^2}. \tag{19.65}$$

This choice ensures that

$$\frac{\partial}{\partial t}N^{\alpha+1}(t^\alpha - \tfrac{\Delta t}{2}) = 0 \qquad \text{and} \qquad \frac{\partial}{\partial t}N^{\alpha-1}(t^\alpha + \tfrac{\Delta t}{2}) = 0, \tag{19.66}$$

so that $\dot{\boldsymbol{U}}^h(t)$ is indeed continuous. The acceleration evaluates to

$$\begin{aligned}\ddot{\boldsymbol{U}}^h(t) &= \boldsymbol{U}^{\alpha+1}\ddot{N}^{\alpha+1}(t) + \boldsymbol{U}^{\alpha}\ddot{N}^{\alpha}(t) + \boldsymbol{U}^{\alpha-1}\ddot{N}^{\alpha-1}(t)\\ &= \frac{\boldsymbol{U}^{\alpha+1}}{(\Delta t)^2} - 2\frac{\boldsymbol{U}^{\alpha}}{(\Delta t)^2} + \frac{\boldsymbol{U}^{\alpha-1}}{(\Delta t)^2} = \frac{\boldsymbol{U}^{\alpha+1} - 2\boldsymbol{U}^{\alpha} + \boldsymbol{U}^{\alpha-1}}{(\Delta t)^2}.\end{aligned} \tag{19.67}$$

Thus, we recover the second-order central-difference scheme in time.

The full, discretized displacement field is now

$$\boldsymbol{u}^h(\boldsymbol{x},t) = \sum_{a=1}^{n}\sum_{\gamma=\alpha-1}^{\alpha+1} \boldsymbol{u}^{a,\gamma} N^{\gamma}(t) N^a(\boldsymbol{x}) \qquad \text{for} \quad t \in [t^\alpha - \tfrac{\Delta t}{2}, t^\alpha + \tfrac{\Delta t}{2}], \tag{19.68}$$

where $\boldsymbol{u}^{a,\gamma}$ is the displacement at node a at time t^γ. The acceleration field follows as the piecewise-constant approximation

$$\ddot{\boldsymbol{u}}^h(\boldsymbol{x},t) = \sum_{a=1}^{n} \frac{\boldsymbol{u}^{a,\alpha+1} - 2\boldsymbol{u}^{a,\alpha} + \boldsymbol{u}^{a,\alpha-1}}{(\Delta t)^2} N^a(\boldsymbol{x}) \qquad \text{for} \quad t \in [t^\alpha - \tfrac{\Delta t}{2}, t^\alpha + \tfrac{\Delta t}{2}]. \tag{19.69}$$

Next, one may wish to evaluate information at the n_t discrete time steps, motivating the choice of the trial function as

$$\boldsymbol{v}^h(\boldsymbol{x},t) = \sum_{b=1}^{n} \boldsymbol{v}^b(t) N^b(\boldsymbol{x}) = \sum_{b=1}^{n}\sum_{\alpha=1}^{n_t} \boldsymbol{v}^{b,\alpha} N^b(\boldsymbol{x}) \delta(t - t^\alpha) \tag{19.70}$$

When this choice of $\boldsymbol{v}^h$ along with $\boldsymbol{u}^h$ into the weak form, which derived earlier as

$$\mathcal{G}(u,v) = \int_{t_1}^{t_2} \left[\int_\Omega (-\rho\, \ddot{u}_i v_i - \sigma_{ij} v_{i,j})\, \mathrm{d}V + \int_\Omega \rho b_i v_i \, \mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i v_i \, \mathrm{d}S \right] \mathrm{d}t = 0, \tag{19.71}$$

then we may integrate explicitly with respect to time, resulting in

$$\begin{aligned}\sum_{b=1}^{n_t} v_i^b \Bigg[\int_\Omega \left(-\rho \frac{u_i^{a,\alpha+1} - 2u_i^{a,\alpha} + u_i^{a,\alpha-1}}{(\Delta t)^2} N^a N^b - \sigma_{ij}(\nabla \boldsymbol{u}^\alpha) N^b_{,j} \right) \mathrm{d}V \\ + \int_\Omega \rho b_i(t^\alpha) N^b \, \mathrm{d}V + \int_{\partial\Omega_N} \hat{t}_i(t^\alpha) N^b \, \mathrm{d}S \Bigg] = 0\end{aligned} \tag{19.72}$$

for each time step t^α and for all admissible choices of v_i^b. The latter implies that we must have

$$\begin{aligned}\int_\Omega \rho N^a N^b \, \mathrm{d}V \frac{u_i^{a,\alpha+1} - 2u_i^{a,\alpha} + u_i^{a,\alpha-1}}{(\Delta t)^2} + \int_\Omega \sigma_{ij}(\nabla \boldsymbol{u}^\alpha) N^b_{,j} \, \mathrm{d}V \\ - \int_\Omega \rho b_i(t^\alpha) N^b \, \mathrm{d}V - \int_{\partial\Omega_N} \hat{t}_i(t^\alpha) N^b \, \mathrm{d}S = 0.\end{aligned} \tag{19.73}$$

This, however, is equivalent to

$$\boldsymbol{M} \frac{\boldsymbol{U}^{h,\alpha+1} - 2\boldsymbol{U}^{h,\alpha} + \boldsymbol{U}^{h,\alpha-1}}{(\Delta t)^2} + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{h,\alpha}) - \boldsymbol{F}_{\text{ext}}(t^\alpha) = \boldsymbol{0}, \tag{19.74}$$

which was derived above for a second-order finite-difference time discretization, see (19.61).

19.7 Implicit time integration

Next, **implicit time integration** uses the same discretization in time but requires solving a (non)linear system of equations for $\boldsymbol{U}^{\alpha+1}$.

The most popular scheme for mechanical problems, is the so-called **Newmark-β** method which is a combination of the linear acceleration and average acceleration schemes. Specifically, one assumes

$$\boldsymbol{U}^{\alpha+1} = \boldsymbol{U}^{\alpha} + \Delta t\,\dot{\boldsymbol{U}}^{\alpha} + \frac{(\Delta t)^2}{2}\left[2\beta\,\ddot{\boldsymbol{U}}^{\alpha+1} + (1-2\beta)\ddot{\boldsymbol{U}}^{\alpha}\right] \tag{19.75a}$$

$$\dot{\boldsymbol{U}}^{\alpha+1} = \dot{\boldsymbol{U}}^{\alpha} + \Delta t\left[\gamma\ddot{\boldsymbol{U}}^{\alpha+1} + (1-\gamma)\ddot{\boldsymbol{U}}^{\alpha}\right] \tag{19.75b}$$

with parameters $0 \le \beta \le \gamma/2$ and $0 \le \gamma \le 1$, often chosen as

- $\beta = \frac{1}{4}$, $\gamma = \frac{1}{2}$ in the **average acceleration** scheme, which is implicit and unconditionally stable (the underlying approximation is that the acceleration within the time interval Δt remains constant).
- $\beta = \frac{1}{6}$, $\gamma = \frac{1}{2}$ in the **linear acceleration** scheme,
- $\beta = \gamma = 0$ returns to the explicit central-difference scheme discussed above.

Most popular is the average accelerations scheme which is unconditionally stable for arbitrary $\Delta t > 0$.

For linear structural dynamics problems, the method is *unconditionally stable* if $2\beta \ge \gamma \ge 1/2$. It is *conditionally stable* if $\gamma < 1/2$. For $\gamma = 1/2$ the scheme is at least second-order accurate, while being first-order accurate otherwise.

For implementation purposes, let us solve (19.75) for the acceleration at the new time $t^{\alpha+1}$:

$$\ddot{\boldsymbol{U}}^{\alpha+1} = \frac{1}{\beta(\Delta t)^2}\left(\boldsymbol{U}^{\alpha+1} - \boldsymbol{U}^{\alpha} - \Delta t\,\dot{\boldsymbol{U}}^{\alpha}\right) - \frac{1-2\beta}{2\beta}\ddot{\boldsymbol{U}}^{\alpha}. \tag{19.76}$$

Insertion into (19.75b) yields

$$\begin{aligned}\dot{\boldsymbol{U}}^{\alpha+1} &= \dot{\boldsymbol{U}}^{\alpha} + \Delta t(1-\gamma)\ddot{\boldsymbol{U}}^{\alpha} + \frac{\gamma}{\beta\,\Delta t}\left(\boldsymbol{U}^{\alpha+1} - \boldsymbol{U}^{\alpha} + \Delta t\,\dot{\boldsymbol{U}}^{\alpha}\right) - \gamma\,\Delta t\frac{1-2\beta}{2\beta}\ddot{\boldsymbol{U}}^{\alpha}\\ &= \left(1 - \frac{\gamma}{\beta}\right)\dot{\boldsymbol{U}}^{\alpha} + \frac{\gamma}{\beta\,\Delta t}\left(\boldsymbol{U}^{\alpha+1} - \boldsymbol{U}^{\alpha}\right) - \Delta t\left(\frac{\gamma}{2\beta} - 1\right)\ddot{\boldsymbol{U}}^{\alpha}\end{aligned} \tag{19.77}$$

Next, inserting both velocity (19.76) and acceleration (19.77) into the equation of motion at the new time $t^{\alpha+1}$,

$$\boldsymbol{M}\ddot{\boldsymbol{U}}^{\alpha+1} + \boldsymbol{C}\dot{\boldsymbol{U}}^{\alpha+1} + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1}) - \boldsymbol{F}_{\text{ext}}(t^{\alpha+1}) = \boldsymbol{0}, \tag{19.78}$$

leads after some rearrangement to

$$\begin{aligned}&\left(\frac{1}{\beta(\Delta t)^2}\boldsymbol{M} + \frac{\gamma}{\beta\,\Delta t}\boldsymbol{C}\right)\boldsymbol{U}^{\alpha+1} + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1}) - \boldsymbol{F}_{\text{ext}}(t^{\alpha+1})\\ &\qquad = \boldsymbol{M}\left[\frac{1}{\beta(\Delta t)^2}\boldsymbol{U}^{\alpha} + \frac{1}{\beta\,\Delta t}\dot{\boldsymbol{U}}^{\alpha} + \left(\frac{1}{2\beta} - 1\right)\ddot{\boldsymbol{U}}^{\alpha}\right]\\ &\qquad\quad + \boldsymbol{C}\left[\frac{\gamma}{\beta\,\Delta t}\boldsymbol{U}^{\alpha} + \left(\frac{\gamma}{\beta} - 1\right)\dot{\boldsymbol{U}}^{\alpha} + \Delta t\left(\frac{\gamma}{2\beta} - 1\right)\ddot{\boldsymbol{U}}^{\alpha}\right].\end{aligned} \tag{19.79}$$

The right-hand side is fully known as it only involves $\boldsymbol{U}^\alpha$, $\dot{\boldsymbol{U}}^\alpha$, and $\ddot{\boldsymbol{U}}^\alpha$ from the previous time step. The left-hand side is generally nonlinear and requires an iterative solver.

Note that the implementation is quite similar to that of the quasistatic Newton-Raphson solver discussed before. The right-hand side is significantly longer (but contains only known information). The left-hand side is extended by the first term in (19.79), which is linear in $\boldsymbol{U}^{\alpha+1}$ and the tangent matrix used for iterations is simply

$$\boldsymbol{T}^* = \frac{1}{\beta(\Delta t)^2}\boldsymbol{M} + \frac{\gamma}{\beta\,\Delta t}\boldsymbol{C} + \boldsymbol{T}. \tag{19.80}$$

Boundary conditions can be implemented in the same fashion as for the quasistatic solvers discussed above.

As an **alternative derivation** of the implementation, we may introduce the changes in displacements, velocities and accelerations during a time step as

$$\Delta\boldsymbol{U}^\alpha = \boldsymbol{U}^{\alpha+1} - \boldsymbol{U}^\alpha, \qquad \Delta\dot{\boldsymbol{U}}^\alpha = \dot{\boldsymbol{U}}^{\alpha+1} - \dot{\boldsymbol{U}}^\alpha, \qquad \Delta\ddot{\boldsymbol{U}}^\alpha = \ddot{\boldsymbol{U}}^{\alpha+1} - \ddot{\boldsymbol{U}}^\alpha. \tag{19.81}$$

We can rearrange the first equation of (19.75) into

$$\boldsymbol{U}^{\alpha+1} - \boldsymbol{U}^\alpha = \Delta t\,\dot{\boldsymbol{U}}^\alpha + \frac{(\Delta t)^2}{2}\left[2\beta(\ddot{\boldsymbol{U}}^{\alpha+1} - \ddot{\boldsymbol{U}}^\alpha) + \ddot{\boldsymbol{U}}^\alpha\right] \tag{19.82}$$

or

$$\Delta\boldsymbol{U}^\alpha = \Delta t\,\dot{\boldsymbol{U}}^\alpha + \frac{(\Delta t)^2}{2}\left[2\beta\,\Delta\ddot{\boldsymbol{U}}^\alpha + \ddot{\boldsymbol{U}}^\alpha\right] \quad\Leftrightarrow\quad \Delta\ddot{\boldsymbol{U}}^\alpha = \frac{1}{\beta(\Delta t)^2}\Delta\boldsymbol{U}^\alpha - \frac{1}{\beta\,\Delta t}\dot{\boldsymbol{U}}^\alpha - \frac{1}{2\beta}\ddot{\boldsymbol{U}}^\alpha. \tag{19.83}$$

Similarly, the second equation of (19.75) gives

$$\dot{\boldsymbol{U}}^{\alpha+1} - \dot{\boldsymbol{U}}^\alpha = \Delta t\left[\gamma(\ddot{\boldsymbol{U}}^{\alpha+1} - \ddot{\boldsymbol{U}}^\alpha) + \ddot{\boldsymbol{U}}^\alpha\right] \quad\Leftrightarrow\quad \Delta\dot{\boldsymbol{U}}^\alpha = \Delta t\,\gamma\,\Delta\ddot{\boldsymbol{U}}^\alpha + \Delta t\,\ddot{\boldsymbol{U}}^\alpha, \tag{19.84}$$

so that insertion of the final result in (19.83) results in

$$\begin{aligned}\Delta\dot{\boldsymbol{U}}^\alpha &= \Delta t\,\gamma\left[\frac{1}{\beta(\Delta t)^2}\Delta\boldsymbol{U}^\alpha - \frac{1}{\beta\,\Delta t}\dot{\boldsymbol{U}}^\alpha - \frac{1}{2\beta}\ddot{\boldsymbol{U}}^\alpha\right] + \Delta t\,\ddot{\boldsymbol{U}}^\alpha \\ &= \frac{\gamma}{\beta\,\Delta t}\Delta\boldsymbol{U}^\alpha - \frac{\gamma}{\beta}\dot{\boldsymbol{U}}^\alpha + \Delta t\left(1 - \frac{\gamma}{2\beta}\right)\ddot{\boldsymbol{U}}^\alpha.\end{aligned} \tag{19.85}$$

The equations of motion (19.24) in their general time-discretized form are

$$\boldsymbol{M}\,\ddot{\boldsymbol{U}}^{\alpha+1} + \boldsymbol{C}\,\dot{\boldsymbol{U}}^{\alpha+1} + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1}) - \boldsymbol{F}_{\text{ext}}(t^{\alpha+1}) = \boldsymbol{0}. \tag{19.86}$$

Subtracting the above equation for two consecutive time steps results in the incremental form

$$\boldsymbol{M}\,\Delta\ddot{\boldsymbol{U}}^\alpha + \boldsymbol{C}\,\Delta\dot{\boldsymbol{U}}^\alpha + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha) - \left[\boldsymbol{F}_{\text{ext}}(t^{\alpha+1}) - \boldsymbol{F}_{\text{ext}}(t^\alpha)\right] = \boldsymbol{0}. \tag{19.87}$$

Now inserting (19.83) and (19.85) leads to

$$\begin{aligned}&\boldsymbol{M}\left[\frac{1}{\beta(\Delta t)^2}\Delta\boldsymbol{U}^\alpha - \frac{1}{\beta\,\Delta t}\dot{\boldsymbol{U}}^\alpha - \frac{1}{2\beta}\ddot{\boldsymbol{U}}^\alpha\right] + \boldsymbol{C}\left[\frac{\gamma}{\beta\,\Delta t}\Delta\boldsymbol{U}^\alpha - \frac{\gamma}{\beta}\dot{\boldsymbol{U}}^\alpha + \Delta t\left(1 - \frac{\gamma}{2\beta}\right)\ddot{\boldsymbol{U}}^\alpha\right] \\ &\quad + \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha) - \left[\boldsymbol{F}_{\text{ext}}(t^{\alpha+1}) - \boldsymbol{F}_{\text{ext}}(t^\alpha)\right] = \boldsymbol{0},\end{aligned} \tag{19.88}$$

which can be regrouped into

$$\begin{aligned}\left[\frac{1}{\beta(\Delta t)^2}\boldsymbol{M}+\frac{\gamma}{\beta\,\Delta t}\boldsymbol{C}\right]\Delta\boldsymbol{U}^\alpha+\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^{\alpha+1})&=\boldsymbol{F}_{\text{ext}}(t^{\alpha+1})-\boldsymbol{F}_{\text{ext}}(t^\alpha)+\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha)\\&-\left[-\frac{1}{2\beta}\boldsymbol{M}+\Delta t\left(1-\frac{\gamma}{2\beta}\right)\boldsymbol{C}\right]\ddot{\boldsymbol{U}}^\alpha-\left[-\frac{1}{\beta\,\Delta t}\boldsymbol{M}-\frac{\gamma}{\beta}\boldsymbol{C}\right]\dot{\boldsymbol{U}}^\alpha,\end{aligned}\tag{19.89}$$

whose right-hand side is entirely known from the previous time step (and the external forces at the current time). Noting that $\Delta\boldsymbol{U}^\alpha=\boldsymbol{U}^{\alpha+1}-\boldsymbol{U}^\alpha$, this is a generally nonlinear system of equations to be solved for $\Delta\boldsymbol{U}^\alpha$ or $\boldsymbol{U}^{\alpha+1}$. For example, we can use a *Newton-Raphson iterative approach* that starts with an initial guess $\Delta\boldsymbol{U}_0^\alpha$ and incrementally finds

$$\Delta\boldsymbol{U}_{i+1}^\alpha=\Delta\boldsymbol{U}_i^\alpha+\delta\boldsymbol{U}_i^\alpha,\tag{19.90}$$

so that we need to solve

$$\left[\frac{1}{\beta(\Delta t)^2}\boldsymbol{M}+\frac{\gamma}{\beta\,\Delta t}\boldsymbol{C}\right](\Delta\boldsymbol{U}_i^\alpha+\delta\boldsymbol{U}_i^\alpha)+\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha+\Delta\boldsymbol{U}_i^\alpha+\delta\boldsymbol{U}_i^\alpha)=\text{RHS}^\alpha\tag{19.91}$$

with

$$RHS^\alpha=\boldsymbol{F}_{\text{ext}}(t^{\alpha+1})-\boldsymbol{F}_{\text{ext}}(t^\alpha)+\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha)+\left[\frac{1}{2\beta}\boldsymbol{M}-\Delta t\left(1-\frac{\gamma}{2\beta}\right)\boldsymbol{C}\right]\ddot{\boldsymbol{U}}^\alpha+\left[\frac{1}{\beta\,\Delta t}\boldsymbol{M}+\frac{\gamma}{\beta}\boldsymbol{C}\right]\dot{\boldsymbol{U}}^\alpha.\tag{19.92}$$

Linearization about $\delta\boldsymbol{U}_i^\alpha=\boldsymbol{0}$ finally gives

$$\begin{aligned}&\left[\frac{1}{\beta(\Delta t)^2}\boldsymbol{M}+\frac{\gamma}{\beta\,\Delta t}\boldsymbol{C}+\boldsymbol{T}(\boldsymbol{U}^\alpha+\Delta\boldsymbol{U}_i^\alpha)\right]\delta\boldsymbol{U}_i^\alpha\\&\qquad\approx\text{RHS}^\alpha-\left[\frac{1}{\beta(\Delta t)^2}\boldsymbol{M}+\frac{\gamma}{\beta\,\Delta t}\boldsymbol{C}\right]\Delta\boldsymbol{U}_i^\alpha-\boldsymbol{F}_{\text{int}}(\boldsymbol{U}^\alpha+\Delta\boldsymbol{U}_i^\alpha).\end{aligned}\tag{19.93}$$

After a time step α, we know $\boldsymbol{U}^\alpha$, $\dot{\boldsymbol{U}}^\alpha$ and $\ddot{\boldsymbol{U}}^\alpha$ as well as $\boldsymbol{F}_{\text{ext}}(t^{\alpha+1})$ and $\boldsymbol{F}_{\text{ext}}(t^\alpha)$. To solve for the new solution at $t^{\alpha+1}$, we choose an initial guess $\Delta\boldsymbol{U}_0^\alpha$ (e.g., $\Delta\boldsymbol{U}_0^\alpha=\boldsymbol{0}$), so that the entire right-hand side in (19.93) as well as the martix in front of the left-hand side in (19.93) are known. Next, the linear system in (19.93) is solved for $\delta\boldsymbol{U}_i^\alpha$, followed by the update $\Delta\boldsymbol{U}_{i+1}^\alpha=\Delta\boldsymbol{U}_i^\alpha+\delta\boldsymbol{U}_i^\alpha$, until convergence is achieved when $\|\delta\boldsymbol{U}_i^\alpha\|\to 0$. Once $\Delta\boldsymbol{U}^\alpha=\lim_{i\to\infty}\Delta\boldsymbol{U}_i^\alpha$ has been found, we can use (19.83) to compute $\Delta\ddot{\boldsymbol{U}}^\alpha$ and $\ddot{\boldsymbol{U}}^{\alpha+1}=\ddot{\boldsymbol{U}}^\alpha+\Delta\ddot{\boldsymbol{U}}^\alpha$ as well as (19.85) to compute $\Delta\dot{\boldsymbol{U}}^\alpha$ and $\dot{\boldsymbol{U}}^{\alpha+1}=\dot{\boldsymbol{U}}^\alpha+\Delta\dot{\boldsymbol{U}}^\alpha$. Then, the procedure restarts for the next time step.

Note that, if essential boundary conditions of the type $\boldsymbol{U}^\alpha=\hat{\boldsymbol{U}}^\alpha$ are to be imposed, this implies that $\Delta\boldsymbol{U}^\alpha=\hat{\boldsymbol{U}}^{\alpha+1}-\hat{\boldsymbol{U}}^\alpha$ is known for all times.

20 Internal variables and inelasticity

20.1 Inelastic material models

Inelastic material models describe a variety of phenomena, e.g.,

- **viscoelasticity**, i.e., time- and rate-dependent reversible behavior (the stress–strain relation depends on the loading rate; stresses and strains evolve over time); internal variables, e.g., for viscoelasticity with n Maxwell elements are the inelastic strain contributions $\boldsymbol{z} = \{\boldsymbol{e}_\mathrm{p}^1, \dots, \boldsymbol{e}_\mathrm{p}^n\}$.
- **plasticity**, i.e., history-dependent irreversible behavior (the stress–strain relation depends on the loading history); internal variables are usually the plastic strains, accumulated plastic strains, and possibly further history variables: $\boldsymbol{z} = \{\boldsymbol{e}_\mathrm{p}, \epsilon_\mathrm{p}, \dots\}$
- **viscoplasticity**, i.e., history- and time-dependent irreversible behavior; internal variables are similar to the case of plasticity above.
- **damage**, i.e., irreversible degradation of the elastic stiffness with loading; internal variable is a damage parameter, e.g., a scalar measure $z = d$.
- **ferroelectricity**, i.e., irreversible electro-mechanical-coupling; internal variable can be, e.g., the polarization field $\boldsymbol{z} = \boldsymbol{p}$.

All these phenomena can be described by the same underlying principles.

The general description of an inelastic (variational) material model starts with a strain energy density

$$W = W(\boldsymbol{\varepsilon}, \boldsymbol{z}), \tag{20.1}$$

where $\boldsymbol{z}$ denotes a collection of internal variables. While the stress tensor and linear momentum balance remain untouched, i.e., (in linearized kinematics)

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\varepsilon}} \qquad \text{and} \qquad \operatorname{div} \boldsymbol{\sigma} + \rho \boldsymbol{b} = \boldsymbol{0}, \tag{20.2}$$

the internal variables evolve according to a kinetic law

$$\frac{\partial W}{\partial \boldsymbol{z}} + \frac{\partial \phi^*}{\partial \dot{\boldsymbol{z}}} \ni 0, \tag{20.3}$$

where ϕ^* denotes the **dual (dissipation) potential** and we assume that such a potential exists. The differential inclusion is replaced by an equality in case of rate-dependent models. (20.3) can alternatively be cast into an effective variational problem:

$$\dot{\boldsymbol{z}} = \arg\inf \left\{ \dot{W} + \phi^* \right\}. \tag{20.4}$$

Note that $P = \dot{W} + \phi^*$ is often referred to as the *total power* since it denotes the rate of change of energy in the system induced by external forces (considering changes to the stored energy and the dissipated energy).

20.2 Example: viscoelasticity

As an example, consider the so-called standard linear solid which demonstrates viscoelastic material behavior. In 1D, strain ε is decomposed into elastic and inelastic contributions according to $\varepsilon = \varepsilon_e + \varepsilon_p$. The stored energy is given by

$$W(\varepsilon, \varepsilon_p) = \frac{E_2}{2}(\varepsilon - \varepsilon_p)^2 + \frac{E_1}{2}\varepsilon_p^2 \tag{20.5}$$

so that

$$\sigma = \frac{\partial W}{\partial \varepsilon} = E_2(\varepsilon - \varepsilon_p) \tag{20.6}$$

and

$$y = -\frac{\partial W}{\partial \varepsilon_p} = E_2(\varepsilon - \varepsilon_p) - E_1\varepsilon_p = \sigma - E_1\varepsilon_p. \tag{20.7}$$

The kinetic rule is given as

$$y = \eta\,\dot{\varepsilon}_p, \tag{20.8}$$

which is equivalent to defining

$$\phi^*(\dot{\varepsilon}_p) = \frac{\eta}{2}(\varepsilon_p)^2 \tag{20.9}$$

such that

$$\frac{\partial W}{\partial \varepsilon_p} + \frac{\partial \phi^*}{\partial \dot{\varepsilon}_p} = 0. \tag{20.10}$$

20.3 Variational constitutive updates

For the numerical treatment, let us introduce a discretization in time: $t^\alpha = \alpha\,\Delta t$, where we assume constant time steps $\Delta t = t^{\alpha+1} - t^\alpha$ and, for conciseness, we write $\Delta(\cdot) = (\cdot)^{\alpha+1} - (\cdot)^\alpha$, where $(\cdot)^\alpha$ denotes a quantity at time t^α. Using simple backward-Euler rules then gives $\dot{W} = (W^{\alpha+1} - W^\alpha)/\Delta t$ and $\dot{\boldsymbol{z}} = (\boldsymbol{z}^{\alpha+1} - \boldsymbol{z}^\alpha)/\Delta t = \Delta\boldsymbol{z}/\Delta t$.

Thus,

$$\frac{\Delta \boldsymbol{z}}{\Delta t} = \arg\inf\left\{\frac{W^{\alpha+1} - W^\alpha}{\Delta t} + \phi^*\left(\frac{\Delta \boldsymbol{z}}{\Delta t}\right)\right\}. \tag{20.11}$$

Multiplication by Δt and omitting W^α (since it does not depend on $\boldsymbol{z}^{\alpha+1}$) leads to

$$\boldsymbol{z}^{\alpha+1} = \arg\inf\left\{W^{\alpha+1}(\boldsymbol{\varepsilon}^{\alpha+1}, \boldsymbol{z}^{\alpha+1}) + \Delta t\,\phi^*\left(\frac{\boldsymbol{z}^{\alpha+1} - \boldsymbol{z}^\alpha}{\Delta t}\right)\right\}, \tag{20.12}$$

where the right-hand side defines an **effective incremental potential**:

$$\boxed{\mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1}, \boldsymbol{z}^{\alpha+1}) = W^{\alpha+1}(\boldsymbol{\varepsilon}^{\alpha+1}, \boldsymbol{z}^{\alpha+1}) + \Delta t\,\phi^*\left(\frac{\boldsymbol{z}^{\alpha+1} - \boldsymbol{z}^\alpha}{\Delta t}\right)} \tag{20.13}$$

Notice that $\boldsymbol{\sigma}^{\alpha+1} = \partial W/\partial \boldsymbol{\varepsilon}^{\alpha+1}$ so that the effective potential can be used to replace the classical strain energy density in the total potential energy:

$$I_{\boldsymbol{z}^\alpha}[\boldsymbol{u}^{\alpha+1}, \boldsymbol{z}^{\alpha+1}] = \int_\Omega \mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1}, \boldsymbol{z}^{\alpha+1})\,\mathrm{d}V - \int_\Omega \rho\,\boldsymbol{b}\cdot\boldsymbol{u}^{\alpha+1}\,\mathrm{d}V - \int_{\partial\Omega_N} \hat{\boldsymbol{t}}\cdot\boldsymbol{u}^{\alpha+1}\,\mathrm{d}S. \tag{20.14}$$

By the subscripts $_{\boldsymbol{z}^\alpha}$ we denote that those potentials do depend on $\boldsymbol{z}^\alpha$ (i.e., the internal variables at the previous time step) but that those fields are known when evaluating the respective quantities.

The solution can now be found from

$$\{\boldsymbol{u}^{\alpha+1},\boldsymbol{z}^{\alpha+1}\} = \arg\inf I_{\boldsymbol{z}^\alpha}[\boldsymbol{u}^{\alpha+1},\boldsymbol{z}^{\alpha+1}]. \tag{20.15}$$

We can exploit that only the internal energy term depends on the internal variables and further assume that the energy density is *local* in the internal variables (this does not apply, e.g., for phase field models whose energy involves gradients of the internal variables). Then,

$$\begin{aligned}\inf_{\boldsymbol{u}^{\alpha+1}}\inf_{\boldsymbol{z}^{\alpha+1}}\int_\Omega \mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}^{\alpha+1})\,\mathrm{d}V - \ldots &= \inf_{\boldsymbol{u}^{\alpha+1}}\int_\Omega \inf_{\boldsymbol{z}^{\alpha+1}}\mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}^{\alpha+1})\,\mathrm{d}V - \ldots\\ &= \inf_{\boldsymbol{u}^{\alpha+1}}\int_\Omega W^*_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1})\,\mathrm{d}V - \ldots,\end{aligned} \tag{20.16}$$

where

$$W^*_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1}) = \inf_{\boldsymbol{z}^{\alpha+1}}\mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}^{\alpha+1}) = \mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}_*^{\alpha+1}), \qquad \boldsymbol{z}_*^{\alpha+1} = \arg\inf \mathcal{F}_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1},\cdot) \tag{20.17}$$

is often referred to as the **condensed energy density** (the internal variables have been "condensed out"). Notice that the omitted terms $(\ldots)$ do not depend on the internal variables.

This finally leads to the **incremental variational problem**

$$\boxed{I_{\boldsymbol{z}^\alpha}[\boldsymbol{u}^{\alpha+1}] = \int_\Omega W^*_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1})\,\mathrm{d}V - \int_\Omega \rho\,\boldsymbol{b}\cdot\boldsymbol{u}^{\alpha+1}\,\mathrm{d}V - \int_{\partial\Omega_N}\hat{\boldsymbol{t}}\cdot\boldsymbol{u}^{\alpha+1}\,\mathrm{d}S,} \tag{20.18}$$

which has the same structure as before. This concept of introducing internal variables into the variational framework is also known as **variational constitutive updates** and goes back to Ortiz and Stainier (2000).

Note that – for numerical implementation purposes – evaluation of $W^*_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1})$ always requires us to compute the updated internal variables $\boldsymbol{z}_*^{\alpha+1}$ based on $\boldsymbol{\varepsilon}^{\alpha+1}$, before the energy, stresses, or incremental stiffness matrix can be evaluated.

The stresses are now

$$\begin{aligned}\boldsymbol{\sigma}^{\alpha+1} &= \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\varepsilon}^{\alpha+1}}W^*_{\boldsymbol{z}^\alpha}(\boldsymbol{\varepsilon}^{\alpha+1})\\ &= \frac{\partial\mathcal{F}}{\partial\boldsymbol{\varepsilon}^{\alpha+1}}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}_*^{\alpha+1}) + \frac{\partial\mathcal{F}}{\partial\boldsymbol{z}^{\alpha+1}}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}_*^{\alpha+1})\cdot\frac{\partial\boldsymbol{z}_*^{\alpha+1}}{\partial\boldsymbol{\varepsilon}^{\alpha+1}}\\ &= \frac{\partial W^*}{\partial\boldsymbol{\varepsilon}^{\alpha+1}}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{z}_*^{\alpha+1}),\end{aligned} \tag{20.19}$$

where the second term vanished because $\boldsymbol{z}_*^{\alpha+1}$ renders $\mathcal{F}$ stationary by definition.

The same trick unfortunately does not apply when computing the incremental stiffness matrix,

$$\mathbb{C}^{\alpha+1}_{ijkl} = \frac{\mathrm{d}\sigma^{\alpha+1}_{ij}}{\mathrm{d}\varepsilon^{\alpha+1}_{kl}} = \frac{\partial\sigma^{\alpha+1}_{ij}}{\partial\varepsilon^{\alpha+1}_{ij}} + \frac{\partial\sigma^{\alpha+1}_{ij}}{\partial\boldsymbol{z}^{\alpha+1}}\cdot\frac{\partial\boldsymbol{z}_*^{\alpha+1}}{\partial\varepsilon^{\alpha+1}_{ij}}, \tag{20.20}$$

where the second term does *not* vanish in general. It requires calculating the sensitivity of the internal variables with respect to the strain tensor components.

20.4 Example: viscoelasticity, (visco)plasticity

Viscoelasticity and (visco)plasticity all start with the same fundamental structure (here presented in linearized kinematics). The total strain tensor *decomposes additively* into elastic and inelastic (or plastic) contributions:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{\mathrm{e}} + \boldsymbol{\varepsilon}_{\mathrm{p}}, \tag{20.21}$$

where $\boldsymbol{\varepsilon}_{\mathrm{p}}$ belongs to the set of internal variables. (Note that in finite deformations, the decomposition is multiplicative: $\boldsymbol{F} = \boldsymbol{F}_{\mathrm{e}}\boldsymbol{F}_{\mathrm{p}}$.)

In case of history dependence, one introduces additional internal history variables. For example for **von Mises plasticity**, the *accumulated plastic strain* ϵ_{p} captures the history of plastic strains $\boldsymbol{\varepsilon}_{\mathrm{p}}$ through the coupling

$$\dot{\epsilon}_{\mathrm{p}} = \|\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}}\|_{\mathrm{vM}} = \sqrt{\frac{2}{3}\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}} \cdot \dot{\boldsymbol{\varepsilon}}_{\mathrm{p}}}. \tag{20.22}$$

With internal variables $\boldsymbol{z} = \{\boldsymbol{\varepsilon}_{\mathrm{p}}, \epsilon_{\mathrm{p}}\}$, the Helmholtz energy density decomposes into elastic and plastic energy:

$$W(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}_{\mathrm{p}}, \epsilon_{\mathrm{p}}) = W_{\mathrm{el}}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{\mathrm{p}}) + W_{\mathrm{pl}}(\epsilon_{\mathrm{p}}). \tag{20.23}$$

In case of reversible behavior (viscoelasticity), we have no plastic energy storage, i.e., $W_{\mathrm{pl}} = 0$.

The dual dissipation potential can be defined, e.g., by the general power-law structure

$$\phi^*(\epsilon_{\mathrm{p}}) = \sigma_0 \,|\dot{\epsilon}_{\mathrm{p}}| + \frac{\tau_0 \dot{\epsilon}_0}{m+1}\left(\frac{\dot{\epsilon}_{\mathrm{p}}}{\dot{\epsilon}_0}\right)^{m+1} \tag{20.24}$$

with positive constants σ_0 (initial yield stress), τ_0 (hardening rate), $\dot{\epsilon}_0$ (reference rate), and m (rate sensitivity). In case of viscoelasticity we choose $\sigma_0 = 0$. By contrast, for rate-independent plasticity one chooses $\tau_0 = 0$.

The differential inclusion in (20.3) is required because of the first term, $\sigma_0\,|\dot{\epsilon}_{\mathrm{p}}|$, whose derivative is not defined at the origin (for $\dot{\epsilon}_{\mathrm{p}} = 0$). Here, a **subdifferential** is required. Assume, e.g., $\tau_0 = 0$ (which is known as **rate-independent plasticity**) such that

$$\frac{\partial \phi^*}{\partial \dot{\epsilon}_{\mathrm{p}}} = \frac{\partial}{\partial \dot{\epsilon}_{\mathrm{p}}}\sigma_0\,|\dot{\epsilon}_{\mathrm{p}}| = \begin{cases} \sigma_0, & \text{if } \dot{\epsilon}_{\mathrm{p}} > 0, \\ -\sigma_0, & \text{if } \dot{\epsilon}_{\mathrm{p}} < 0, \\ (-\sigma_0, \sigma_0) & \text{if } \dot{\epsilon}_{\mathrm{p}} = 0. \end{cases} \quad \Rightarrow \quad \begin{cases} y = \sigma_0, & \text{if } \dot{\epsilon}_{\mathrm{p}} > 0, \\ y = -\sigma_0, & \text{if } \dot{\epsilon}_{\mathrm{p}} < 0, \\ y \in (-\sigma_0, \sigma_0) & \text{if } \dot{\epsilon}_{\mathrm{p}} = 0. \end{cases} \tag{20.25}$$

In the first two cases, the kinetic law becomes an equality: *plastic flow* occurs with $\dot{\epsilon}_{\mathrm{p}} > 0$ (or $\dot{\epsilon}_{\mathrm{p}} < 0$) while the driving force y equals the critical stress σ_0 (or $-\sigma_0$). In the third case, *elastic* loading implies that $|y| < \sigma_0$ (here, the differential inclusion is required).

In case $\tau_0 \neq 0$, the material behavior becomes viscoplastic and the driving force is no longer constant by rate-dependent in the plastic regime. However, $\sigma_0 \neq 0$ still introduces a threshold to plastic flow.

The specific forms of W_{el}, W_{pl}, and ϕ^* depend on the particular material model.

20.5 Example: viscoplasticity

Since plastic/viscous deformation is observed to be isochoric, one commonly assumes

$$\operatorname{tr}\boldsymbol{\varepsilon}_\mathrm{p} = 0 \qquad \text{so that} \qquad \boldsymbol{\varepsilon}_\mathrm{p} = \boldsymbol{e}_\mathrm{p}. \tag{20.26}$$

Similarly, only the deviatoric stress tensor $\boldsymbol{s}$ should cause plastic deformation. Here and in the following, we denote the **deviatoric** tensors by

$$\operatorname{dev}(\cdot) = (\cdot) - \frac{1}{3}\operatorname{tr}(\cdot)\boldsymbol{I}, \qquad \text{so that} \qquad \boldsymbol{e} = \operatorname{dev}\boldsymbol{\varepsilon}, \quad \boldsymbol{s} = \operatorname{dev}\boldsymbol{\sigma}. \tag{20.27}$$

Using the above definitions of energy density and dual potential, we obtain

$$\mathcal{F}_{\{\boldsymbol{e}_\mathrm{p}^\alpha,\epsilon_\mathrm{p}^\alpha\}}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{e}_\mathrm{p}^{\alpha+1},\epsilon_\mathrm{p}^{\alpha+1}) = W^{\alpha+1}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{e}_\mathrm{p}^{\alpha+1},\epsilon_\mathrm{p}^{\alpha+1}) + \Delta t\,\phi^*\left(\frac{\epsilon_\mathrm{p}^{\alpha+1}-\epsilon_\mathrm{p}^\alpha}{\Delta t}\right) \tag{20.28}$$

Minimization with respect to $\boldsymbol{e}_\mathrm{p}^{\alpha+1}$ gives

$$\begin{aligned}\frac{\partial\mathcal{F}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}} &= \frac{\partial W_\mathrm{el}^{\alpha+1}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}} + \frac{\partial W_\mathrm{pl}^{\alpha+1}}{\partial\epsilon_\mathrm{p}^{\alpha+1}}\frac{\partial\epsilon_\mathrm{p}^{\alpha+1}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}} + \Delta t\frac{\partial\phi^*}{\partial\epsilon_\mathrm{p}^{\alpha+1}}\frac{1}{\Delta t}\frac{\partial\epsilon_\mathrm{p}^{\alpha+1}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}}\\ &= -\operatorname{dev}\boldsymbol{\sigma}^{\alpha+1} + \left[\tau(\epsilon_\mathrm{p}^{\alpha+1}) + \tau^*(\epsilon_\mathrm{p}^{\alpha+1})\right]\frac{\partial\epsilon_\mathrm{p}^{\alpha+1}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}} \ni 0\end{aligned} \tag{20.29}$$

with back-stresses

$$\tau(\epsilon_\mathrm{p}^{\alpha+1}) = \frac{\partial W_\mathrm{pl}^{\alpha+1}}{\partial\epsilon_\mathrm{p}^{\alpha+1}}, \qquad \tau^*(\epsilon_\mathrm{p}^{\alpha+1}) = \frac{\partial\phi^*}{\partial\epsilon_\mathrm{p}^{\alpha+1}}. \tag{20.30}$$

Further, note that

$$\frac{\partial\epsilon_\mathrm{p}^{\alpha+1}}{\partial\boldsymbol{e}_\mathrm{p}^{\alpha+1}} = \frac{\partial\Delta\epsilon_\mathrm{p}^{\alpha+1}}{\partial\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}} = \frac{\partial}{\partial\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}}\sqrt{\frac{2}{3}\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}\cdot\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}} = \frac{2\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}}{3\Delta\epsilon_\mathrm{p}^{\alpha+1}}. \tag{20.31}$$

Altogether, this results in

$$-\boldsymbol{s}^{\alpha+1} + \left[\tau(\epsilon_\mathrm{p}^{\alpha+1}) + \tau^*(\epsilon_\mathrm{p}^{\alpha+1})\right]\frac{2\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}}{3\Delta\epsilon_\mathrm{p}^{\alpha+1}} \ni 0 \tag{20.32}$$

or

$$\boldsymbol{s}^{\alpha+1} \in \left[\tau(\epsilon_\mathrm{p}^{\alpha+1}) + \tau^*(\epsilon_\mathrm{p}^{\alpha+1})\right]\frac{2\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}}{3\Delta\epsilon_\mathrm{p}^{\alpha+1}}. \tag{20.33}$$

Let us consider the *elastically isotropic* case (i.e., $\boldsymbol{\sigma} = \kappa(\operatorname{tr}\boldsymbol{\varepsilon})\boldsymbol{I} + 2\mu\,\boldsymbol{e}_\mathrm{e}$). Assume that $\Delta\epsilon_\mathrm{p}^{\alpha+1} > 0$, so we have

$$2\mu\left(\boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_\mathrm{p}^{\alpha+1}\right) = \left[\tau(\epsilon_\mathrm{p}^{\alpha+1}) + \sigma_0 + \tau_0\left(\frac{\Delta\epsilon_\mathrm{p}^{\alpha+1}}{\dot\epsilon_0\,\Delta t}\right)^m\right]\frac{2\Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}}{3\Delta\epsilon_\mathrm{p}^{\alpha+1}}. \tag{20.34}$$

and we introduce an **elastic predictor** (i.e., strain if the plastic strain remained unaltered)

$$\boldsymbol{e}_\mathrm{pre}^{\alpha+1} = \boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_\mathrm{p}^\alpha \qquad \text{such that} \qquad \boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_\mathrm{p}^{\alpha+1} = \boldsymbol{e}_\mathrm{pre}^{\alpha+1} - \Delta\boldsymbol{e}_\mathrm{p}^{\alpha+1}. \tag{20.35}$$

That gives

$$2\mu\, \boldsymbol{e}_{\text{pre}}^{\alpha+1} = 2\mu\, \Delta\boldsymbol{e}_{\text{p}}^{\alpha+1} + \frac{2}{3\Delta\epsilon_{\text{p}}^{\alpha+1}}\left[3\mu\, \Delta\epsilon_{\text{p}}^{\alpha+1} + \tau(\epsilon_{\text{p}}^{\alpha+1}) + \sigma_0 + \tau_0\left(\frac{\Delta\epsilon_{\text{p}}^{\alpha+1}}{\dot\epsilon_0\,\Delta t}\right)^m\right]\Delta\boldsymbol{e}_{\text{p}}^{\alpha+1}. \tag{20.36}$$

or

$$\frac{2\mu\, \boldsymbol{e}_{\text{pre}}^{\alpha+1}}{2\mu + \frac{2}{3\Delta\epsilon_{\text{p}}^{\alpha+1}}\left(\tau(\epsilon_{\text{p}}^{\alpha+1}) + \sigma_0 + \tau_0\left(\frac{\Delta\epsilon_{\text{p}}^{\alpha+1}}{\dot\epsilon_0\,\Delta t}\right)^m\right)} = \Delta\boldsymbol{e}_{\text{p}}^{\alpha+1}. \tag{20.37}$$

Now, let us use that

$$\Delta\epsilon_{\text{p}}^{\alpha+1} = \sqrt{\frac{2}{3}\Delta\boldsymbol{e}_{\text{p}}^{\alpha+1}\cdot\Delta\boldsymbol{e}_{\text{p}}^{\alpha+1}} = \frac{2\mu\left\|\boldsymbol{e}_{\text{pre}}^{\alpha+1}\right\|_{\text{vM}}}{2\mu + \frac{2}{3\Delta\epsilon_{\text{p}}^{\alpha+1}}\left(\tau(\epsilon_{\text{p}}^{\alpha+1}) + \sigma_0 + \tau_0\left(\frac{\Delta\epsilon_{\text{p}}^{\alpha+1}}{\dot\epsilon_0\,\Delta t}\right)^m\right)}. \tag{20.38}$$

This is a scalar equation to be solved for the increment $\Delta\epsilon_{\text{p}}^{\alpha+1}$, which is then inserted into (20.37) in order to determine $\Delta\boldsymbol{e}_{\text{p}}^{\alpha+1}$. This completes the calculation of the updated internal variables.

Notice that the above equation is equivalent to saying (introducing the **von Mises stress** σ_{vM})

$$\sigma_{\text{vM}} = \sqrt{\frac{3}{2}\boldsymbol{s}^{\alpha+1}\cdot\boldsymbol{s}^{\alpha+1}} = \tau(\epsilon_{\text{p}}^{\alpha+1}) + \tau^*(\epsilon_{\text{p}}^{\alpha+1}). \tag{20.39}$$

Analogous relations can be obtained for $\Delta\epsilon_{\text{p}}^{\alpha+1} < 0$.

Note that rate-independent plasticity (including the special case of **von Mises plasticity**) assumes that $\tau_0 = 0$; i.e., the dissipation potential only provides the initial yield threshold σ_0.

20.6 Example: linear viscoelasticity

In the simplest viscoelastic case, we take $W_{\text{pl}} = 0$ and thus $\tau(\epsilon_{\text{p}}^{\alpha+1}) = 0$, i.e., the material has no "memory". Also, the yield threshold is removed by choosing $\sigma_0 = 0$, and the accumulated plastic stress ϵ_{p} is of no interest nor needed. The dissipation is reformulated as

$$\phi^*(\dot{\boldsymbol{e}}_p) = \frac{\eta}{2}\left\|\dot{\boldsymbol{e}}_p\right\|^2, \tag{20.40}$$

which agrees with the plastic case above with velocity-proportional damping ($m = 1$), viscosity $\eta = \tau_0/\dot\epsilon_0$, and the von Mises norm replaced by the classical vector norm.

In this case, (20.34) reduces to

$$2\mu\left(\boldsymbol{e}_{\text{pre}}^{\alpha+1} - \Delta\boldsymbol{e}_{\text{p}}\right) = \tau_0\frac{\Delta\boldsymbol{e}_{\text{p}}}{\dot\epsilon_0\,\Delta t} = \eta\frac{\Delta\boldsymbol{e}_{\text{p}}}{\Delta t}, \tag{20.41}$$

which can be rearranged to

$$2\mu\, \boldsymbol{e}_{\text{pre}}^{\alpha+1} = \eta\frac{\Delta\boldsymbol{e}_{\text{p}}}{\Delta t} + 2\mu\,\Delta\boldsymbol{e}_{\text{p}} \tag{20.42}$$

giving

$$2\boldsymbol{e}_{\text{pre}}^{\alpha+1} = \left(\frac{\eta/\mu}{\Delta t} + 2\right)\Delta\boldsymbol{e}_{\text{p}}. \tag{20.43}$$

If we define the relaxation time $\tau = \eta/\mu$, then

$$\Delta \boldsymbol{e}_{\mathrm{p}} = \frac{2}{2+\tau/\Delta t}\boldsymbol{e}_{\mathrm{pre}}^{\alpha+1} \tag{20.44}$$

or

$$\boldsymbol{e}_{\mathrm{p}}^{\alpha+1} = \boldsymbol{e}_{\mathrm{p}}^{\alpha} + \frac{2}{2+\tau/\Delta t}\left(\boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_{\mathrm{p}}^{a}\right). \tag{20.45}$$

This can be extended to the **generalized Maxwell model**. Let us assume isotropic elasticity with shear and bulk modulus μ_∞ and κ_∞, respectively, while the n viscoelastic branches are characterized by shear stiffnesses μ_i and viscosities η_i (for $i = 1,\dots,n$). The effective incremental energy density now becomes

$$\begin{aligned}\mathcal{F}_{\{\boldsymbol{e}_p^{i,\alpha}\}}(\boldsymbol{\varepsilon}^{\alpha+1},\boldsymbol{e}_p^{1,\alpha+1},\dots,\boldsymbol{e}_p^{n,\alpha+1}) = \frac{\kappa_\infty}{2}(\operatorname{tr}\boldsymbol{\varepsilon}^{\alpha+1})^2 + \mu_\infty\,\boldsymbol{e}^{\alpha+1}\cdot\boldsymbol{e}^{\alpha+1} + \sum_{i=1}^n \mu_i\left\|\boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_p^{i,\alpha+1}\right\|^2 \\ + \sum_{i=1}^n \frac{\eta_i}{2\Delta t}\left\|\boldsymbol{e}_p^{i,\alpha+1} - \boldsymbol{e}_p^{i,\alpha}\right\|^2,\end{aligned} \tag{20.46}$$

whose minimization with respect to the new internal variables yields

$$\boldsymbol{e}_p^{i,\alpha+1} = \boldsymbol{e}_p^{i,\alpha} + \frac{2}{2+\tau_i/\Delta t}\left(\boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_p^{i,\alpha}\right) \tag{20.47}$$

with relaxation times $\tau_i = \eta_{(i)}/\mu_{(i)}$. Note that this agrees with (20.45) for a single Maxwell element.

Insertion and differentiation leads to the stress tensor

$$\boldsymbol{\sigma}^{\alpha+1}(\varepsilon^{\alpha+1},\boldsymbol{e}_p^{1,\alpha},\dots,\boldsymbol{e}_p^{n,\alpha}) = \kappa^\infty(\operatorname{tr}\boldsymbol{\varepsilon}^{\alpha+1})\boldsymbol{I} + 2\mu_\infty\boldsymbol{e}^{\alpha+1} + \sum_{i=1}^n 2\mu_i\frac{\tau_i/\Delta t}{2+\tau_i/\Delta t}\left(\boldsymbol{e}^{\alpha+1} - \boldsymbol{e}_p^{i,\alpha}\right). \tag{20.48}$$

Similarly, the consistent incremental tangent matrix can be computed by differentiating $\boldsymbol{\sigma}^{\alpha+1}$:

$$\begin{aligned}\mathbb{C}_{ijkl}^{\alpha+1} = \frac{\partial\boldsymbol{\sigma}^{\alpha+1}}{\partial\boldsymbol{\varepsilon}^{\alpha+1}} = \left[\kappa_\infty - \frac{2}{3}\left(\mu_\infty + \sum_{i=1}^n \mu_i\frac{\tau_i/\Delta t}{2+\tau_i/\Delta t}\right)\right]\delta_{ij}\delta_{kl} \\ + \left(\mu_\infty + \sum_{i=1}^n \mu_i\frac{\tau_i/\Delta t}{2+\tau_i/\Delta t}\right)(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}).\end{aligned} \tag{20.49}$$

Finally, the condensed incremental energy density W^* can be computed analytically be inserting (20.47) into (20.46).

Index

A Introduction, Vector Spaces

We define a **set** Ω as a collection of points $\boldsymbol{X} \in \Omega$.

We further say Ω is a **(proper) subset** of all space if $\Omega \subseteq \mathbb{R}^d$ in d dimensions (*proper* if $\Omega \subset \mathbb{R}^d$).

We usually take Ω to be an **open set**, i.e., $\Omega \cap \partial\Omega = \varnothing$ with **boundary** $\partial\Omega$.

Deformation and motion are described by a **mapping**

$$\varphi: \ \boldsymbol{X} \in \Omega \to \boldsymbol{\varphi}(\boldsymbol{X}) \in \mathbb{R}^d \qquad \text{or} \qquad \varphi : \Omega \to \mathbb{R}^d, \tag{A.1}$$

where Ω is the **domain** and $\mathbb{R}^d$ the **range** of φ. The mapped (current) configuration of Ω is $\varphi(\Omega)$.

Every **function** $f(x) : \mathbb{R} \to \mathbb{R}$ is a mapping from $\mathbb{R}$ to $\mathbb{R}$.

We call a mapping **injective** (or **one-to-one**) if for each $\boldsymbol{x} \in \varphi(\Omega)$ there is one unique $\boldsymbol{X} \in \Omega$ such that $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X})$. In other words, no two points $\boldsymbol{X} \in \Omega$ are mapped onto the same position $\boldsymbol{x}$. A mapping is **surjective** (or **onto**) if the entire set Ω is mapped onto the entire set $\varphi(\Omega)$; i.e., for every $\boldsymbol{X} \in \Omega$ there exists at least one $\boldsymbol{x} \in \varphi(\Omega)$ such that $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X})$. If a mapping is both injective and surjective (or one-to-one and onto) we say it is **bijective**. A bijective mapping is also called an **isomorphism**.

For example, $\varphi : \Omega \to \mathbb{R}^d$ is injective, whereas $\varphi : \Omega \to \varphi(\Omega)$ would be bijective.

For time-dependent problems, we have $\varphi: \ \Omega \times \mathbb{R} \to \mathbb{R}^d$ and $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t)$. This describes a family of configurations $\varphi(\Omega, t)$, from which we arbitrarily define a reference configuration Ω for which $\varphi = \mathrm{id}$ (the **identity mapping**).

A **linear/vector space** $\{\Omega, +; \mathbb{R}, \cdot\}$ is defined by the following identities. For any $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in \Omega$ and $\alpha, \beta \in \mathbb{R}$ it holds that

(i) *closure*: $\alpha \cdot \boldsymbol{u} + \beta \cdot \boldsymbol{v} \in \Omega$
(ii) *associativity w.r.t.* +: $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$
(iii) *null element*: $\exists\ \boldsymbol{0} \in \Omega$ such that $\boldsymbol{u} + \boldsymbol{0} = \boldsymbol{u}$
(iv) *negative element*: for all $\boldsymbol{u} \in \Omega \ \ \exists\ -\boldsymbol{u} \in \Omega$ such that $\boldsymbol{u} + (-\boldsymbol{u}) = \boldsymbol{0}$
(v) *commutativity*: $\boldsymbol{u} + \boldsymbol{v} = \boldsymbol{v} + \boldsymbol{u}$
(vi) *associativity w.r.t.* $\cdot$: $(\alpha\beta) \cdot \boldsymbol{u} = \alpha(\beta \cdot \boldsymbol{u})$
(vii) *distributivity w.r.t.* $\mathbb{R}$: $(\alpha + \beta) \cdot \boldsymbol{u} = \alpha \cdot \boldsymbol{u} + \beta \cdot \boldsymbol{u}$
(viii) *distributivity w.r.t.* Ω: $\alpha \cdot (\boldsymbol{u} + \boldsymbol{v}) = \alpha \cdot \boldsymbol{u} + \alpha \cdot \boldsymbol{v}$
(ix) *identity*: $1 \cdot \boldsymbol{u} = \boldsymbol{u}$

examples:

- $\mathbb{R}^d$ is a vector space. By contrast, $\Omega \subset \mathbb{R}^d$ is *not* a vector space, since – in general – it violates, e.g., (i) closure and (iv) null element.
- $\mathbb{P}_2 = \{ax^2 + bx + c \ : \ a, b, c \in \mathbb{R}\}$ is the **space of all second-order polynomial functions**, or an **ordered triad** $(a, b, c) \in \mathbb{R}^3$. More generally, $\mathbb{P}_k(\Omega)$ is the space of all kth-order polynomial functions defined on Ω. $\mathbb{P}_k(\Omega)$ is a *linear space*.

We call $\mathbb{P}_2$ a **linear subspace** of $\mathbb{P}_k$ with $k \geq 2$, and we write $\mathbb{P}_2 \subseteq \mathbb{P}_k$.

B Function Spaces

Consider a function $u(x) : \Omega \to \mathbb{R}$ and $\Omega \subset \mathbb{R}$.

u is **continuous at a point** x if, given any scalar $\epsilon > 0$, there is a $r(\epsilon) \in \mathbb{R}$ such that

$$|u(y) - u(x)| < \epsilon \qquad \text{provided that} \qquad |y - x| < r. \tag{B.1}$$

A function u is **continuous over** Ω if it is continuous at all points $x \in \Omega$.

u is of **class** $C^k(\Omega)$ with an integer $k \geq 0$ if it is k *times continuously differentiable* over Ω (i.e., u possesses derivatives up to the kth order and these derivatives are continuous functions).

Examples:

- Functions $u(x) \in \mathbb{P}_k$ with $k \geq 0$ are generally $C^\infty(\mathbb{R})$.
- Consider a continuous, piecewise-linear function $u : \Omega = (0, 2) \to \mathbb{R}$. Function u is $C^0(\Omega)$ but not $C^1(\Omega)$.
- The Heavyside function $H(x)$ is said to be $C^{-1}(\mathbb{R})$ since its "**zeroth derivative**" (i.e., the function itself) is not continuous.

If there are no discontinuities such as cracks, shocks, etc. (or discontinuities in the BCs/ICs) we usually assume the solution fields are $C^\infty(\Omega)$, so we may take derivatives; otherwise, derivatives exist almost everywhere (**a.e.**)

To evaluate the global errors of functions, we need *norms*.

Consider a linear space $\{\mathcal{U}, +; \mathbb{R}, \cdot\}$. A mapping $\langle\cdot,\cdot\rangle : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ is called **inner product** on $\mathcal{U} \times \mathcal{U}$ if for all $u, v, w \in \mathcal{U}$ and $\alpha \in \mathbb{R}$:

(i) $\langle u + v, w\rangle = \langle u, w\rangle + \langle v, w\rangle$
(ii) $\langle u, v\rangle = \langle v, u\rangle$
(iii) $\langle \alpha \cdot u, v\rangle = \alpha \langle u, v\rangle$
(iv) $\langle u, u\rangle \geq 0 \quad$ and $\quad \langle u, u\rangle = 0 \Leftrightarrow u = 0$

A linear space $\mathcal{U}$ endowed with an inner product is called an **inner product space**.

Examples:

- $\langle \boldsymbol{u}, \boldsymbol{v}\rangle = u_i v_i = \boldsymbol{u} \cdot \boldsymbol{v}$ defines an inner product for $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$.
- The **L$_2$-inner product** for functions $u, v \in \mathcal{U}$ with domain Ω:
$$\langle u, v\rangle_{L_2(\Omega)} = \int_\Omega u(x)\, v(x)\, \mathrm{d}x \qquad \text{and often just} \qquad \langle u, v\rangle = \langle u, v\rangle_{L_2(\Omega)}\,. \tag{B.2}$$

Note that if $\langle u, v\rangle = 0$ we say u and v are **orthogonal**.

Examples:

- *Legendre polynomials*:
$$p_n(x) = \frac{1}{2^n n!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}(x^2 - 1)^n \qquad \text{so that} \qquad p_0 = 1, \quad p_1 = x, \quad p_2 = \frac{1}{2}(3x^2 - 1), \ldots \tag{B.3}$$
orthogonality on $\Omega = (-1, 1)$:
$$\int_{-1}^{1} p_n(x)\, p_m(x)\, \mathrm{d}x = \frac{2}{2n + 1}\delta_{mn} \tag{B.4}$$

- *trigonometric functions*:

  $$p_n(x) = \cos\left(\frac{\pi n x}{L}\right) \tag{B.5}$$

  orthogonality on $\Omega = (-L, L)$

  $$\int_{-L}^{L} p_n(x)\, p_m(x)\,\mathrm{d}x = \begin{cases} 2L, & \text{if } m = n = 0 \\ L, & \text{if } m = n \neq 0 \\ 0, & \text{else} \end{cases} \tag{B.6}$$

Now we are in place to define the **distance** between x_1 and x_2:

$$d(x_1, x_2) = \sqrt{\langle x_1 - x_2, x_1 - x_2\rangle} \tag{B.7}$$

We need this concept not only for points in space but also to define the closeness or proximity of functions.

Consider a linear space $\{\mathcal{U}, +; \mathbb{R}, \cdot\}$. A mapping $\|\cdot\| : \mathcal{U} \to \mathbb{R}_+$ is called a **norm** on $\mathcal{U}$ if for all $u, v \in \mathcal{U}$ and $\alpha \in \mathbb{R}$:

(i) $\|u + v\| \leq \|u\| + \|v\|$ (**triangle inequality**)

(ii) $\|\alpha \cdot u\| = |\alpha|\, \|u\|$

(iii) $\|u\| \geq 0$ and $\|u\| = 0 \;\Leftrightarrow\; u = 0.$

A linear space Ω endowed with a norm is called a **normed linear space** (NLS).

Examples of norms:

- Consider the d-dimensional Euclidean space, so $\boldsymbol{x} = \{x_1, \dots, x_d\}^{\mathrm{T}}$. Then we define
  - the 1-norm: $\|\boldsymbol{x}\|_1 = \sum_{i=1}^{d} |x_i|$
  - the 2-norm: $\|\boldsymbol{x}\|_2 = \left(\sum_{i=1}^{d} |x_i|^2\right)^{1/2}$ (**Euclidean norm**)
  - the n-norm: $\|\boldsymbol{x}\|_n = \left(\sum_{i=1}^{d} |x_i|^n\right)^{1/n}$
  - the ∞-norm: $\|\boldsymbol{x}\|_\infty = \max_{1\leq i\leq n} |x_i|$
- Now turning to functions, the **$\mathbf{L}_p$-norm** of a function $u : \Omega \to \mathbb{R}$:

  $$\|u\|_{L_p(\Omega)} = \left(\int_\Omega u^p \,\mathrm{d}x\right)^{1/p} \tag{B.8}$$

The most common norm is the **L_2-norm**:

$$\|u\|_{L_2(\Omega)} = \langle u, u\rangle_{L_2(\Omega)}^{1/2} = \left(\int_\Omega u^2(x)\,\mathrm{d}x\right)^{1/2}. \tag{B.9}$$

Furthermore, notice that

$$\|u\|_{L_\infty(\Omega)} = \operatorname*{ess\,sup}_{x\in\Omega} |u(x)|, \tag{B.10}$$

where we introduced the **essential supremum**

$$\operatorname*{ess\,sup}_{x\in\Omega} |u(x)| = M \qquad \text{with the smallest } M \text{ that satisfies } |u(x)| \leq M \text{ for a.e. } x \in \Omega. \tag{B.11}$$

Now, that we have norms, we can generalize our definition of the distance. If $u_n,\, u \in \mathcal{U}$ equipped with a norm $\|\cdot\| : \mathcal{U} \to \mathbb{R}$, then we define the **distance** as

$$d(u_n, u) = \|u_n - u\|\,. \tag{B.12}$$

Now, we are in place to define the **convergence** of a sequence of functions u_n to u in $\mathcal{U}$: we say $u_n \to u \in \mathcal{U}$ if for all $\epsilon > 0$ there exists $N(\epsilon)$ such that $d(u_n, u) < \epsilon$ for all $n > N$.

Examples:

- Consider $u_n \in \mathcal{U} = \mathbb{P}_2(\Omega)$ with L_2-norm and $\Omega \subset \mathbb{R}$

$$u_n(x) = \left(1 + \frac{1}{n}\right) x^2 \quad \to \quad u(x) = x^2 \qquad \text{since} \qquad d(u_n - u) = \frac{1}{n} \int x^2 \,\mathrm{d}x \tag{B.13}$$

  with $u \in \mathcal{U} = \mathbb{P}_2(\Omega)$. For example, for $d(u_n - u) < \epsilon$ we need $n > N = \int_\Omega x^2 \,\mathrm{d}x / \epsilon$.

- *Fourier series*:

$$u(x) = \sum_{i=0}^{\infty} c_i\, x^i \qquad \Rightarrow \qquad u_n(x) = \sum_{i=0}^{n} c_i\, x^i \qquad \text{such that} \qquad u_n \to u \text{ as } n \to \infty. \tag{B.14}$$

Given a point u in a normed linear space $\mathcal{U}$, a **neighborhood** $\mathcal{N}_r(u)$ of radius $r > 0$ is defined as the set of points $v \in \mathcal{U}$ for which $d(u, v) < r$. Now, we can define sets properly:

A subset $\mathcal{V} \subset \mathcal{U}$ is called **open** if, for each point $u \in \mathcal{V}$, there exists a neighborhood $\mathcal{N}_r(u)$ which is fully contained in $\mathcal{V}$. The complement $\widetilde{\mathcal{V}}$ of an open set $\mathcal{V}$ is, by definition a closed set. The **closure** $\overline{\mathcal{V}}$ of an open set $\mathcal{V}$ is the smallest closed set that contains $\mathcal{V}$. In simple terms, a closed set is defined as a set which contains all its limit points. Therefore, note that

$$\sup_{x \in \Omega} |u(x)| = \max_{x \in \overline{\Omega}} |u(x)|\,. \tag{B.15}$$

For example, $(0, 1)$ is an open set in $\mathbb{R}$. $[0, 1]$ is a closed set, and $[0, 1]$ is the closure of $(0, 1)$.

A linear space $\mathcal{U}$ is a **complete space** if every sequence u_n in $\mathcal{U}$ converges to $u \in \mathcal{U}$. In simple terms, the space must contain all *limit points*.

A *complete normed linear space* is called a **Banach space**; i.e., $\{\mathcal{U}, +; \mathbb{R}, \cdot\}$ with a norm $\|\cdot\|$ and $u_n \to u \in \mathcal{U}$. A *complete inner product space* is called a **Hilbert space**.

Note that $\|\cdot\| = \langle \cdot, \cdot \rangle^{1/2}$ defines a norm. Hence, every Hilbert space is also a Banach space (but not the other way around).

As an example, consider $\mathcal{U} = \mathbb{P}_n$ (the space of all polynomial functions of order $n \in \mathbb{N}$). This is a *linear space* which we equip with a norm, e.g., the L_2-norm. It is *complete* since $(a_n, b_n, c_n, \dots) \to (a, b, c, \dots)$ for $a, b, c, \dots \in \mathbb{R}$. And an *inner product* is defined via $\langle u, v \rangle = \int_\Omega uv \,\mathrm{d}x$. With all these definitions, $\mathcal{U}$ is a Hilbert space.

We can use these norms to define function spaces, e.g., the **L$_2$-space of functions**:

$$L_2(\Omega) = \left\{ u : \Omega \to \mathbb{R} \;:\; \int_\Omega u^2 \,\mathrm{d}x < \infty \right\} \tag{B.16}$$

We say $L_2(\Omega)$ contains all functions that are **square-integrable** on Ω.

Examples:

- $u : \Omega \to \mathbb{R}$ with $u \in \mathbb{P}_k(\Omega)$ and $\operatorname{ess\,sup}_{x\in\Omega} |u(x)| < \infty$. Then, $u \in \text{Ł}_2(\Omega)$.
- $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = x^{-2}$ is not in $L_2(\Omega)$ if $0 \in \Omega$.
- Piecewise constant functions u (with $\operatorname{ess\,sup}_{x\in\Omega} |u(x)| < \infty$) are square-integrable and thus in L_2.

Note that we can write alternatively

$$L_2(\Omega) = \left\{ u : \Omega \to \mathbb{R} \;:\; \|u\|_{L_2(\Omega)} < \infty \right\}. \tag{B.17}$$

C Approximation Theory

Motivation: in computational mechanics, we seek approximate solutions $u_h(x) = \sum_{a=1}^{N} u_a N_a(x)$, e.g., a linear combination of basis functions $N_a(x)$ with amplitudes $u_a \in \mathbb{R}$.

Questions: How does $u_h(x)$ converge to $u(x)$, if at all? Can we find an error estimate $\|u_h - u\|$? What is the rate of convergence (how fast does it converge, cf. the truncation error arguments for grid-based direct methods)?

Fundamental tools for estimating errors are the **Poincaré inequalities**:

(i) **Dirichlet-Poincaré inequality**:

$$\int_0^h |v(x)|^2 \,\mathrm{d}x \le c_h \int_0^h \left[v'(x)\right]^2 \,\mathrm{d}x \qquad \text{if} \qquad v(0) = v(h) = 0. \tag{C.1}$$

with a constant $c_h > 0$ that depends on the interval size h.

(ii) **Neumann-Poincaré** (or Poincaré-Wirtinger) inequality:

$$\int_0^h |v(x) - \overline{v}|^2 \,\mathrm{d}x \le c_h \int_0^h \left[v'(x)\right]^2 \,\mathrm{d}x \qquad \text{with} \qquad \overline{v} = \frac{1}{h}\int_0^h u(x)\,\mathrm{d}x \tag{C.2}$$

In 1D an optimal constant can be found: $c_h = h^2/\pi^2$.

(iii) extension:

$$\int_0^h |v(x)|^2 \,\mathrm{d}x \le \frac{h^2}{\pi^2}\left[\int_0^h \left[v'(x)\right]^2 \,\mathrm{d}x + |v(x_0)|^2\right] \qquad \text{with} \qquad x_0 \in [0,h]. \tag{C.3}$$

Now, let us use those inequalities to find error bounds. Suppose a general function $u(x)$ is approximated by a *piecewise linear* approximation $u_h(x)$. Let's first find a *local* error estimate.

Consider $v(x) = u_h'(x) - u'(x)$ and note that by *Rolle's theorem*

$$u_h'(x_0) - u'(x_0) = 0 \qquad \text{for some} \qquad x_0 \in (0,h). \tag{C.4}$$

Next, use inequality (iii):

$$\int_0^h \left|u_h'(x) - u'(x)\right|^2 \,\mathrm{d}x \le \frac{h^2}{\pi^2}\int_0^h \left|u_h''(x) - u''(x)\right|^2 \,\mathrm{d}x, \tag{C.5}$$

but since $u_h(x)$ is piecewise linear, we have $u_h''(x) = 0$, so that we arrive at the **local error estimate**

$$\int_0^h \left|u_h'(x) - u'(x)\right|^2 \,\mathrm{d}x \le \frac{h^2}{\pi^2}\int_0^h \left|u''(x)\right|^2 \,\mathrm{d}x. \tag{C.6}$$

Now, let's seek a **global error estimate** by using

$$\int_a^b (\cdot)\,\mathrm{d}x = \sum_{i=0}^{n}\int_{x_i}^{x_{i+1}} (\cdot)\,\mathrm{d}x \qquad \text{with} \qquad x_0 = a,\ x_{n+1} = b,\ x_{i+1} = x_i + h \tag{C.7}$$

so that

$$\boxed{\int_a^b \left|u_h'(x) - u'(x)\right|^2 \,\mathrm{d}x \le \frac{h^2}{\pi^2}\int_a^b \left|u''(x)\right|^2 \,\mathrm{d}x} \tag{C.8}$$

Taking square roots, we see that for $\Omega = (a, b)$

$$\left\| u_h' - u' \right\|_{L_2(\Omega)} \le \frac{h}{\pi} \left\| u'' \right\|_{L_2(\Omega)} \tag{C.9}$$

and hence that $\left\| u_h' - u' \right\|_{L_2(\Omega)} \to 0$ as $h \to 0$ linearly in h.

We want to write this a bit more concise. Let us define the **Sobolev semi-norm**:

$$\boxed{|u|_{H^k(\Omega)} = \left[\int_\Omega \left| D^k u \right|^2 \, \mathrm{d}x \right]^{1/2}} \qquad \text{or for short} \qquad |u|_k = |u|_{H^k(\Omega)} \tag{C.10}$$

where in 1D $D^k u = u^{(k)}$. A **semi-norm** in general must satisfy the following conditions:

(i) $\|u + v\| \le \|u\| + \|v\|$ (like for a *norm*)
(ii) $\|\alpha \cdot u\| = |\alpha| \, \|u\|$ (like for a *norm*)
(iii) $\|u\| \ge 0$ (a *norm* also requires $\|u\| = 0$ iff $u = 0$, not so for a semi-norm).

Examples in 1D:

- from before:

$$|u|_{H^1(a,b)} = \left[\int_a^b \left| u'(x) \right|^2 \, \mathrm{d}x \right]^{1/2} \tag{C.11}$$

- analogously:

$$|u|_{H^2(a,b)} = \left[\int_a^b \left| u''(x) \right|^2 \, \mathrm{d}x \right]^{1/2} \tag{C.12}$$

so that we can write (C.8) as

$$|u_h - u|^2_{H^1(a,b)} \le \frac{h^2}{\pi^2} |u|^2_{H^2(a,b)} \qquad \Rightarrow \qquad \boxed{|u_h - u|_{H^1(a,b)} \le \frac{h}{\pi} |u|_{H^2(a,b)}} \tag{C.13}$$

Hence, the topology of convergence is bounded by the regularity of u. Convergence with h-refinement is linear.

- note the special case

$$|u|^2_{H^0} = \int_\Omega u(x)^2 \, \mathrm{d}x = \|u\|^2_{L^2} \qquad (L_2\text{-norm}) \tag{C.14}$$

We can extend this to **higher-order interpolation**. For example, use a piecewise quadratic interpolation u_h. From Poincaré:

$$\int_0^h \left|u_h' - u'\right|^2 \,\mathrm{d}x \le \frac{h^2}{\pi^2}\int_0^h \left|u_h'' - u''\right|^2 \,\mathrm{d}x \le \frac{h^4}{\pi^4}\int_0^h \left|u_h''' - u'''\right|^2 \,\mathrm{d}x = \frac{h^4}{\pi^4}\int_0^h \left|u'''\right|^2 \,\mathrm{d}x \quad \text{(C.15)}$$

Extension into a **global error estimate** with quadratic h-convergence:

$$|u_h - u|_{H^1(a,b)} \le \frac{h^2}{\pi^2}|u|_{H^3(a,b)}\,. \tag{C.16}$$

For a **general interpolation of order** k:

$$\boxed{|u_h - u|_{H^1(a,b)} \le \frac{h^k}{\pi^k}|u|_{H^{k+1}(a,b)}} \tag{C.17}$$

Why is the Sobolev semi-norm not a norm? Simply consider the example $u(x) = c > 0$. All higher derivatives vanish on $\mathbb{R}$, so that $|u|_{H^k(\Omega)} = 0$ for $\Omega \subset \mathbb{R}$ and $k \ge 1$. However, that does not imply that $u = 0$ (in fact, it is not).

Let us introduce the **Sobolev norm** (notice the double norm bars)

$$\|u\|_{H^k(\Omega)} = \left(\sum_{m=0}^{k}|u|^2_{H^m(\Omega)}\right)^{1/2} \qquad \text{or for short} \qquad \|u\|_k = \|u\|_{H^k(\Omega)} \tag{C.18}$$

For example, in one dimension

$$\|u\|^2_{H^1(\Omega)} = |u|_0^2 + |u|_1^2 = \int_\Omega u(x)^2\,\mathrm{d}x + \int_\Omega \left[u'(x)\right]^2\,\mathrm{d}x = \|u\|^2_{L_2(\Omega)} + |u|^2_{H^1(\Omega)} \tag{C.19}$$

Note that this also shows that, more generally,

$$\|u\|^2_{H^k(\Omega)} \ge |u|^2_{H^k(\Omega)}\,. \tag{C.20}$$

Let us derive a final global error estimate, one that involves proper norms – here for the example of a piecewise linear u_h. Start with Poincaré inequality (i):

$$\|u_h - u\|^2_{L_2(a,b)} = \int_a^b |u_h - u|^2\,\mathrm{d}x \le c_h \int_a^b \left|u_h' - u'\right|^2\,\mathrm{d}x = c_h\,|u_h - u|^2_{H^1(a,b)} \tag{C.21}$$

and from (C.19):

$$\begin{aligned}\|u_h - u\|^2_{H^1(\Omega)} &= \|u_h - u\|^2_{L_2(\Omega)} + |u_h - u|^2_{H^1(\Omega)}\\ &\le (1 + c_h)\,|u_h - u|^2_{H^1(a,b)} \le c^* h\,|u|^2_{H^2(a,b)}\end{aligned} \tag{C.22}$$

which along with (C.13) gives

$$\boxed{\|u_h - u\|_{H^1(a,b)} \le ch\,|u|_{H^2(a,b)} \le ch\,\|u\|_{H^2(a,b)}} \tag{C.23}$$

Summary and Extension of Norms:

$$\begin{aligned}
L_p\text{-norm:}\quad & \|u\|_{L_p(\Omega)} = \left(\int_\Omega u^p \,\mathrm{d}x\right)^{1/p}\\
\text{Sobolev semi-norm}:\quad & |u|_{H^k(\Omega)} = \left[\int_\Omega \left|D^k u\right|^2 \,\mathrm{d}x\right]^{1/2} = |u|_k\\
\text{Sobolev norm}:\quad & \|u\|_{H^k(\Omega)} = \left(\sum_{m=0}^k |u|^2_{H^m(\Omega)}\right)^{1/2} = \|u\|_k\\
\text{generalization}:\quad & |u|_{W^{k,p}(\Omega)} = \left[\int_\Omega \left|D^k u\right|^p \,\mathrm{d}x\right]^{1/p} = |u|_{k,p}\\
& \|u\|_{W^{k,p}(\Omega)} = \left(\sum_{m=0}^k |u|^p_{W^{m,p}(\Omega)}\right)^{1/p} = \|u\|_{k,p}
\end{aligned}$$

C.1 Sobolev spaces

The Sobolev norm is used to define a **Sobolev space**:

$$H^k(\Omega) = \left\{u : \Omega \to \mathbb{R} \quad \text{such that} \quad \|u\|_k < \infty\right\}, \tag{C.24}$$

which includes all functions whose kth-order derivatives are **square-integrable**.

Examples:

- Consider a piecewise linear function $u \in C^0$ defined on $\Omega = (0,2)$. Then $u \in H^1(\Omega)$ since the first derivative is piecewise constant and therefore square-integrable.
- Consider the Heavyside step function $H(x) \in C^{-1}$ defined on $\mathbb{R}$. Then, e.g., $h \in H^0(\Omega)$ with $\Omega = (-1,1)$ since the first derivative (the Dirac delta function) is not square-integrable over $(-1,1)$.

Overall, note that the above examples imply that

$$\boxed{H^m(\Omega) \subset C^k(\Omega) \qquad \text{with} \qquad m > k.} \tag{C.25}$$

For example, if a function has a kth continuous derivative, then the $(k+1)$th derivative is defined piecewise and therefore square-integrable.

C.2 Higher dimensions

To extend the above concepts to higher dimensions, we need multi-indices. A **multi-index** is an array of non-negative integers:

$$\alpha = (\alpha_1, \ldots, \alpha_n) \quad \in (\mathbb{Z}_0^+)^n \tag{C.26}$$

The **degree** of a multi-index is defined as

$$|\alpha| = \alpha_1 + \ldots + \alpha_n. \tag{C.27}$$

This can be used to define a **monomial** for $\boldsymbol{x} \in \mathbb{R}^n$:

$$\boldsymbol{x}^\alpha = x_1^{\alpha_1} \cdot \ldots \cdot x_n^{\alpha_n} \tag{C.28}$$

For example, we can now extend our definition of polynomials to higher dimensions:

$$p(\boldsymbol{x}) \in \mathbb{P}_k(\mathbb{R}^2) \qquad \Rightarrow \qquad p(\boldsymbol{x}) = \sum_{\beta=0}^{k} \sum_{|\alpha|=\beta} a_\alpha \boldsymbol{x}^\alpha \tag{C.29}$$

Specifically, the monomials above for $\boldsymbol{x} \in \mathbb{R}^2$ are

$$\begin{aligned} &\text{for } |\alpha| = 0: && \{x^0 y^0\} = \{1\} \\ &\text{for } |\alpha| = 1: && \{x^1 y^0, x^1 y^0\} = \{x, y\} \\ &\text{for } |\alpha| = 2: && \{x^2 y^0, x^1, y^1, x^0 y^2\} = \{x^2, xy, y^2\} \quad \ldots \end{aligned} \tag{C.30}$$

so that

$$p(\boldsymbol{x}) = a_{(0,0)} + a_{(1,0)} x_1 + a_{(0,1)} x_2 + a_{(2,0)} x_1^2 + a_{(1,1)} x_1 x_2 + a_{(0,2)} x_2^2 + \ldots \tag{C.31}$$

Note that this defines a **complete polynomial** of degree k.

Now we can use multi-indices to define partial derivatives via

$$\boxed{D^\alpha u = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \cdot \ldots \cdot \partial x_n^{\alpha_n}} \qquad \text{and} \qquad D^0 u = u} \tag{C.32}$$

A common notation is

$$\sum_{|\alpha|=\beta} D^\alpha u = \sum_{\substack{\alpha_1,\ldots,\alpha_n \\ \text{s.t. } |\alpha|=\beta}} \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \cdot \ldots \cdot \partial x_n^{\alpha_n}} \tag{C.33}$$

With the above derivatives, we may redefine the *inner product*

$$\langle u, v \rangle_{H^m(\Omega)} = \int_\Omega \sum_{\beta=0}^{m} \sum_{|\alpha|=\beta} D^\beta u \, D^\beta u \, \mathrm{d}x \tag{C.34}$$

and the *Sobolev norm*

$$\|u\|_{H^m(\Omega)} = \langle u, u \rangle_{H^m(\Omega)}^{1/2} = \left[\sum_{\beta=0}^{m} \sum_{|\alpha|=\beta} \int_\Omega (D^\alpha u)^2 \, \mathrm{d}x \right]^{1/2} = \left[\sum_{\beta=0}^{m} \sum_{|\alpha|=\beta} \|D^\alpha u\|_{L_2(\Omega)}^2 \right]^{1/2} \tag{C.35}$$

Let's look at some examples; e.g., consider $\Omega = \mathbb{R}^2$ and $m = 1$. Then we have

$$D^0 u = u \qquad \text{and} \qquad D^1 u = \left\{ \frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2} \right\} \tag{C.36}$$

so that

$$\langle u, v \rangle_{H^1(\mathbb{R}^2)} = \int_{\mathbb{R}^2} \left(uv + \frac{\partial u}{\partial x_1} \frac{\partial v}{\partial x_1} + \frac{\partial u}{\partial x_2} \frac{\partial v}{\partial x_2} \right) \mathrm{d}x_1 \, \mathrm{d}x_2 \tag{C.37}$$

and

$$\|u\|_{H^1(\mathbb{R}^2)}^2 = \int_{\mathbb{R}^2} \left[u^2 + \left(\frac{\partial u}{\partial x_1} \right)^2 + \left(\frac{\partial u}{\partial x_2} \right)^2 \right] \mathrm{d}x_1 \, \mathrm{d}x_2. \tag{C.38}$$

Altogether we can now properly define a **Sobolev space** in arbitrary dimensions:

$$\boxed{H^m(\Omega) = \left\{ u : \Omega \to \mathbb{R} \ : \ D^\alpha u \in L_2(\Omega) \ \forall \ \alpha \leq m \right\}} \tag{C.39}$$

This is the set of all functions whose derivatives up to mth order all exist and are square-integrable.

As an example, $u \in H^1(\Omega)$ implies that u and all its first partial derivatives must be square-integrable over Ω because (C.38) must be finite.

Let us look at the example $u(x) = |x|$ and take $\Omega = (-1,1)$. Then, we have $u'(x) = H(x)$ (the Heaviside jump function) and $u''(x) = \delta(x)$ (the Dirac delta function). Therefore,

$$\begin{aligned}
\int_a^b u^2(x)\,\mathrm{d}x < \infty \qquad &\Rightarrow \qquad u \in L_2(\Omega) = H^0(\Omega)\\
\int_a^b \left(\frac{\partial u}{\partial x}\right)^2 \mathrm{d}x = \int_a^b H^2(x)\,\mathrm{d}x < \infty \qquad &\Rightarrow \qquad \frac{\partial u}{\partial x} \in L_2(\Omega) \qquad \text{and} \qquad u \in H^1(\Omega)\\
\int_a^b \left(\frac{\partial^2 u}{\partial x^2}\right)^2 \mathrm{d}x = \int_a^b \delta^2(x)\,\mathrm{d}x = \infty \qquad &\Rightarrow \qquad \frac{\partial^2 u}{\partial x^2} \notin L_2(\Omega) \qquad \text{and} \qquad u \notin H^2(\Omega)
\end{aligned} \tag{C.40}$$

Note that one usually indicates the *highest order* k that applies (since this is what matters for practical purposes), so here we thus conclude that $u \in H^1(\Omega)$.

From the above, we also see that

$$H^\infty \subset \ldots \subset H^2 \subset H^1 \subset H^0 = L_2. \tag{C.41}$$

Notice that even though polynomials $u \in \mathbb{P}_k(\Omega)$ are generally in $H^\infty(\Omega)$ for any bounded $\Omega \subset \mathbb{R}^d$, they are not square-integrable over $\Omega = \mathbb{R}^d$. Luckily, in practical problems we usually consider only finite bodies Ω. To more properly address this issue, let us introduce the **support** of a continuous function u defined on the open set $\Omega \in \mathbb{R}^d$ as the closure in Ω of the set of all points where $u(x) \neq 0$, i.e.,

$$\operatorname{supp} u = \overline{\left\{x \in \Omega \ : \ u(x) \neq 0\right\}} \tag{C.42}$$

This means that $u(x) = 0$ for $x \in \Omega \setminus \operatorname{supp} u$. We may state, e.g., that functions $u : \Omega \to \mathbb{R}$ with a *finite support* $\Omega \subset \mathbb{R}^d$ and ess $\sup_\Omega < \infty$ are square-integrable over Ω.

Finally, let us define by $C_0^k(\Omega)$ the set of all functions contained in $C^k(\Omega)$ whose support is a bounded subset of Ω. Also, notice that

$$C_0^k(\Omega) \subset H_0^k(\Omega) \tag{C.43}$$

and

$$C_0^\infty(\Omega) = \bigcap_{k \geq 0} C_0^k(\Omega). \tag{C.44}$$

D Operators

Energies are often defined via operators. Generally, we call $\mathcal{A}$ an **operator** if

$$\mathcal{A} : u \in \mathcal{U} \to \mathcal{A}(u) \in \mathcal{V}, \tag{D.1}$$

where both $\mathcal{U}$ and $\mathcal{V}$ are function spaces.

A simple example is

$$\mathcal{A}(u) = c\,\frac{\mathrm{d}u}{\mathrm{d}x}, \tag{D.2}$$

which is a (linear differential) *operator* requiring $u \in C^1$.

An operator $\mathcal{A} : \mathcal{U} \to \mathcal{V}$ is **linear** if for all $u_1, u_2 \in \mathcal{U}$ and $\alpha, \beta \in \mathbb{R}$

$$\mathcal{A}(\alpha \cdot u_1 + \beta \cdot u_2) = \alpha \cdot \mathcal{A}(u_1) + \beta \cdot \mathcal{A}(u_2). \tag{D.3}$$

For example, $\mathcal{L}$ is a linear operator in

$$a\,u_{,xx} + b\,u_{,x} = c \qquad \Leftrightarrow \qquad \mathcal{L}(u) = c \qquad \text{with} \qquad \mathcal{L}(\cdot) = a(\cdot)_{,xx} + b(\cdot)_{,x}. \tag{D.4}$$

Operators (such as the inner product operator) can also act on more than one function. Consider, e.g., an operator $\mathcal{B} : \mathcal{U} \times \mathcal{V} \to \mathbb{R}$ where $\mathcal{U}, \mathcal{V}$ are Hilbert spaces. $\mathcal{B}$ is called a **bilinear operator** if for all $u, u_1, u_2 \in \mathcal{U}$ and $v, v_1, v_2 \in \mathcal{V}$ and $\alpha, \beta \in \mathbb{R}$

(i) $\mathcal{B}(\alpha \cdot u_1 + \beta \cdot u_2, v) = \alpha \cdot \mathcal{B}(u_1, v) + \beta \cdot \mathcal{B}(u_2, v)$
(ii) $\mathcal{B}(u, \alpha \cdot v_1 + \beta \cdot v_2) = \alpha \cdot \mathcal{B}(u, v_1) + \beta \cdot \mathcal{B}(u, v_2)$

An example of a bilinear operator is the inner product $\langle \cdot, \cdot \rangle : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ for some Hilbert space $\mathcal{U}$.

An operator $\mathcal{A} : \mathcal{U} \to \mathcal{V}$ is called **symmetric** if

$$\langle \mathcal{A}(u), v \rangle = \langle u, \mathcal{A}(v) \rangle \qquad \text{for all} \qquad u, v, \in \mathcal{U}. \tag{D.5}$$

Furthermore, the operator is **positive** if

$$\langle \mathcal{A}(u), u \rangle \geq 0 \qquad \text{for all} \qquad u \in \mathcal{U}. \tag{D.6}$$

An example of a symmetric operator is $\mathcal{A}(\boldsymbol{u}) = \boldsymbol{M}\boldsymbol{u}$ with $\boldsymbol{u} \in \mathbb{R}^d$ and $\boldsymbol{M} \in \mathbb{R}^{d \times d}$, which is positive if $\boldsymbol{M}$ is positive-semidefinite.

E Uniqueness

One of the beauties of the above variational problem (3.37) is that a unique minimizer exists by the **Lax-Milgram theorem**. This is grounded in (assuming $|\Omega| < \infty$ and $u, v \in \mathcal{U}$ with some Hilbert space $\mathcal{U}$):

- *boundedness* of the bilinear form:

  $$|\mathcal{B}(u,v)| \leq C \|u\| \, \|v\| \qquad \text{for some} \qquad C > 0. \tag{E.1}$$

  For a bilinear form $\mathcal{B}(u,v) = \langle \operatorname{Grad} u, \operatorname{Grad} v \rangle$, this is satisfied by the *Cauchy-Schwarz inequality* (using L_2-norms):

  $$|\mathcal{B}(u,v)| \leq C \|\operatorname{Grad} u\|_{L_2(\Omega)} \|\operatorname{Grad} v\|_{L_2(\Omega)} \leq C \|\operatorname{Grad} u\|_{H^1(\Omega)} \|\operatorname{Grad} v\|_{H^1(\Omega)} \tag{E.2}$$

- *coercivity* of the bilinear form (*ellipticity*):

  $$\mathcal{B}(u,u) \geq c \|u\|^2 \qquad \text{for some} \qquad c > 0. \tag{E.3}$$

  Again, for a bilinear form $\mathcal{B}(u,v) = \langle \operatorname{Grad} u, \operatorname{Grad} v \rangle$ this is satisfied by *Poincaré's inequality*:

  $$\mathcal{B}(u,u) = \|\operatorname{Grad} u\|^2_{L_2(\Omega)} \geq c \|u\|^2_{L_2(\Omega)} \tag{E.4}$$

These two requirements imply the well-posedness of the variational problem and thus imply the existence of a unique solution (or, that the potential has a unique global minimizer). In simple terms, the two conditions that the functional has sufficient *growth properties* (i.e., the bilinear form has upper and lower bounds).

F Vainberg's theorem

The question arises whether or not a general form like (3.35) always exists for any set of PDEs/ODEs as governing equations. Vainberg's theorem helps us answer this question. Consider a weak form

$$\mathcal{G}[u,v]=0 \qquad \forall \quad v\in\mathcal{U}_0(\Omega). \tag{F.1}$$

Let us see if $\mathcal{G}$ derives from a potential I via its first variation. That would imply that

$$\mathcal{G}(u,\delta u)=D_{\delta u}I[u]=\lim_{\epsilon\to 0}\frac{\mathrm{d}}{\mathrm{d}\epsilon}I[u+\delta u]. \tag{F.2}$$

Now recall from calculus that for any continuously differentiable function $f(x,y)$ we must have by *Schwartz' theorem*

$$\frac{\partial}{\partial y}\frac{\partial f}{\partial x}=\frac{\partial}{\partial x}\frac{\partial f}{\partial y}. \tag{F.3}$$

We can use the same strategy to formulate whether or not a weak form derives from a potential. Specifically, we can take one more variation and state that

$$\boxed{D_{\delta u_2}\mathcal{G}(u,\delta u_1)=D_{\delta u_1}\mathcal{G}(u,\delta u_2) \qquad \text{if and only if } I[u] \text{ exists}} \tag{F.4}$$

This is known as **Vainberg's theorem**.

We can easily verify this for the general form given in (3.35):

$$\begin{aligned}\mathcal{G}(u,\delta u_1)=D_{\delta u_1}I[u]&=\int_\Omega[\lambda\,\mathrm{Grad}\,u\ \mathrm{Grad}\,\delta u_1-S\,\delta u_1]\,\mathrm{d}x-\int_{\partial\Omega_N}\hat{Q}\,\delta u_1\,\mathrm{d}x\\ \Rightarrow\quad D_{\delta u_1}\mathcal{G}(u,\delta_2)&=\int_\Omega\lambda\,\mathrm{Grad}\,\delta u_2\ \mathrm{Grad}\,\delta u_1\,\mathrm{d}x=D_{\delta u_2}\mathcal{G}(u,\delta u_1)\end{aligned} \tag{F.5}$$

In simple terms (and not most generally), Vainberg's theorem holds if the potential obeys symmetry. This in turn implies that the governing PDE contains derivatives of *even* order (such as linear momentum balance which is second-order in both spatial and temporal derivatives, or the equilibrium equations of beams which are fourth-order in space and second-order in time). If the PDEs are of *odd* order (such as, e.g., the time-dependent diffusion or heat equation), then no direct potential I exists – there are work-arounds using so-called *effective potentials* that will be discussed later in the context of internal variables.

Of course, knowing that a variational structure exists is beneficial but it does not reveal anything about the actual solution u which will be obtained by solving the above system of equations.

G Energy norm

For many purposes it will be convenient to introduce the so-called **energy norm**

$$|\cdot|_E = \sqrt{\mathcal{B}(\cdot,\cdot)}. \tag{G.1}$$

For example, if in subsequent sections we find an approximate solution T^h for the temperature field, then the error can be quantified by

$$\left|T - T^h\right|_E = \sqrt{\mathcal{B}(T - T^h, T - T^h)} = \sqrt{\int_\Omega \lambda \left\|\text{Grad}(T - T^h)\right\|^2 \mathrm{d}V}. \tag{G.2}$$

Notice that in this case $\left\|T - T^h\right\|$ does not necessarily imply $T - T^h = 0$ so that, strictly speaking, the above energy norm is only a *semi-norm*. To turn it into a proper norm, we need to exclude solutions corresponding to *rigid-body motion* (i.e., solutions that imply uniform translations or rotations of the T-field but without producing gradients in T). If we assume that essential boundary conditions are chosen appropriately to suppress rigid body motion by seeking solutions

$$T \in \mathcal{U} = \left\{T \in H^1 \;:\; T = \hat{T} \text{ on } \partial\Omega_\mathrm{D}\right\}, \tag{G.3}$$

then, for this space, $\|\cdot\|_E$ is indeed a proper norm.