LGCIV2041
Numerical Analysis of Civil Engineering Structures

Jean-François Remacle

Contents

CHAPTER 1

Calculus of variations

In the context of calculus, the derivative $f'(x) = \frac{\mathrm{d}f}{\mathrm{d}x}$ of a function $f(x) \in C^2(\mathbb{R})$ is a well defined concept. If $f'(x) = 0$ for $x = x_0$, then f attains either a maximum or a minumum at that point. This means that, for a given neighborhood $|x - x_0| < \delta$, then either $f(x) < f(x_0)$ (f is maximum, $f''(x_0) < 0$) or $f(x) > f(x_0)$ (f is minimum, $f''(x_0) > 0$).

We will see in forthcoming section that finding the static equilibrium of an elastic structure can is a problem that can be written in the following fashion. Consider a deformable structure $\Omega \in \mathbb{R}^3$ and a displacement field

$$\mathbf{u}(\mathbf{x}) = \{u_1(x_1, x_2, x_3), u_2(x_1, x_2, x_3), u_3(x_1, x_2, x_3)\}$$

that is admissible. We will be more clear about what is an admissible displacement in the next chapter but for now, we assume that a displacement field $\mathbf{u}$ is admissible if it belongs to a vector valued function space U that contains all possible admissible displacements.

The static equilibrium of a structure Ω that is loaded by some volume forces $\mathbf{f}(\mathbf{x})$, $\mathbf{x} \in \Omega$ and by surface loads $\mathbf{F}(\mathbf{x})$, $\mathbf{x} \in \Gamma_F$ consist in finding the $\mathbf{u} \in U$ that minimizes the potential energy

$$\pi(\mathbf{u}) = \frac{1}{2}\int_\Omega \sigma_{ij}\epsilon_{ij}\,dv - \int_\Omega f_i u_i\,dv - \int_{\Gamma_F} F_i u_i\,ds \tag{1.1}$$

where

$$\epsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) = \frac{1}{2}\left(\partial_i u_j + \partial_j u_i\right) = u_{(i,j)} \quad , \quad \sigma_{ij} = \lambda\epsilon_{mm}\delta_{ij} + 2\mu\epsilon_{ij},$$

with Lamé coefficients

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad , \quad \mu = \frac{E}{2(1+\nu)},$$

and with E and ν being the Young modulus and the Poisson ratio.

The potential energy π that is defined in (1.1) is not a function in the classical sense. A function $f(x)$ transforms reals into reals: $f: \mathbb{R} \to \mathbb{R}$. The argument $\mathbf{u}$ of $\pi(\mathbf{u})$ is not a real but a function. Such an object that takes a function as argument and returns reals is called a *functional.* How could we possibly find the extremum of a functional? We could try to mimic the definition of the derivative of a function:

$$\frac{\mathrm{d}f}{\mathrm{d}x} = \lim_{dx \to 0} \frac{f(x+dx) - f(x)}{dx}. \tag{1.2}$$

Here is the problem: how could we possibly define $d\mathbf{u}$ when $\mathbf{u}$ is a function?

1.1 The brachistochrone curve

At this point, it is instructive to present a very famous problem that has lead to the development of the calculus of variations. Most of the theory that is developed here is due to the king of mathematicians: Leonhard Euler (1707-1783). In 1696, Johann Bernoulli (1667 - 1747) that was Euler's Ph.D. advisor introduced the following problem[1]. Assume that we have to design the most exciting roller-coaster ever. The ride starts at a point $\mathbf{x}^1(x_1, y_1)$ where the car is launched with a zero speed. The car then goes down following a planar curve $y(x)$ and ends at point $\mathbf{x}^2(x_2, y_2)$. For sake of simplicity, and without loss of generality, we choose $\mathbf{x}^1 = \{0,0\}$ and we assume the acceleration of gravity to be oriented along the positive y's (see Figure 1.1).

We assume now that the most exciting ride will be the one that minimizes the time for going from $\mathbf{x}^1$ to $\mathbf{x}^2$. The curve that minimizes the ride time is called the *brachistochrone curve.*

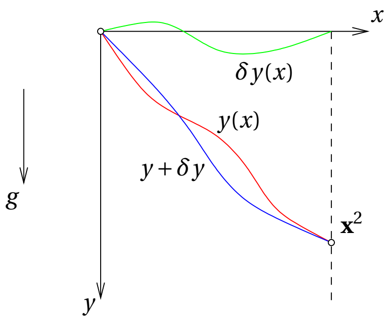

Figure 1.1: Brachistochrone curve.

Classical mechanics allow to write

$$\mathbf{v} = \frac{\partial \mathbf{x}}{\partial t}.$$

[1] The statement has been rephrased

The time T required to go from $\mathbf{x}^1$ to $\mathbf{x}^2$ is computed as follows:

$$T = \int_{\mathbf{x}^1}^{\mathbf{x}^2} \frac{dl}{V} = \int_{\mathbf{x}^1}^{\mathbf{x}^2} \frac{\sqrt{dx^2 + dy^2}}{V} = \int_0^{x_2} \frac{\sqrt{1 + y'^2}}{V} dx$$

where V is the velocity of the car. We have assumed that the car was initially at rest. If m is the mass of the car and if g is the acceleration of gravity, we have

$$mgy = \frac{mV^2}{2} \quad \rightarrow \quad V = \sqrt{2gy}.$$

Time T is finally computed as:

$$\boxed{T(y) = \int_0^{x_2} \sqrt{\frac{1 + y'^2}{2gy}} dx.} \tag{1.3}$$

In (1.3), T is a functional: its argument is the shape of the curve $y(x)$ i.e. a function of y. The argument of functional T is the actual shape of the , i.e. $y(x)$. Now comes the question of admissibility. Ca we choose any $y(x)$? Of course not: $y(x)$ must pass through $\mathbf{x}^1$ and $\mathbf{x}^2$. This is a the admissibility condition and we write $y \in U$ with

$$U = \{y(x) \mid y(x_1) = y_1, \;\; y(x_2) = y_2\}.$$

We now introduce the concept of variation. In order to find the minimum of T among all y's, we have to define the equivalent of a dx in the definition of the derivative (1.2). We have to define a perturbation $\delta y(x)$ that verifies

$$y + \delta y \in U.$$

In other words, the perturbed function $y + \delta y$ should still be admissible in order to be able to evaluate $T(y + \delta y)$. The condition

$$\delta y(x_1) = \delta y(x_2) = 0$$

allows to write

$$y(x_1) + \delta y(x_1) = y_1 \;\; \text{and} y(x_2) + \delta y(x_2) = y_2$$

which means that $y + \delta y$ is admissible and we define $\delta u \in U_0$ with the function space of variations

$$U_0 = \{\delta y(x) \mid \delta y(x_1) = \delta y(x_2) = 0\}$$

There is indeed another admissibility condition. Functional $T(y)$ should be "computable" i.e. $y(x) > 0$. If $y < 0$ for any $x \in [0, x_2]$, then the car would never finish the ride. Here, we do not ask δy to be positive because δy is assumed to be an infinitesimal variation of y, as it is the case for the standard derivative.

1.2 A fundamental result

Proposition 1.2.1 *If a continuous function $f(x)$ in $[a,b]$ and if*

$$\int_a^b f(x)\,w(x)\,\mathrm{d}x = 0$$

for every continuous function $w(x)$ such that $w(a) = w(b) = 0$, then $f(x) = 0$ for all x in $[a,b]$.

Proof Suppose that $f(x)$ is non zero, say positive at some point in $[a,b]$. Then f is positive for some interval $[x_1, x_2]$ contained in $[a,b]$. If we set

$$w(x) = (x - x_1)(x_2 - x)$$

for $x \in [x_1, x_2]$ and $w(x) = 0$ otherwise, then $w(x)$ satisfies the conditions of the hypothesis. However

$$\int_a^b f(x)\,w(x)\,\mathrm{d}x = \int_{x_1}^{x_2} f(x)\,(x - x_1)(x_2 - x)\,\mathrm{d}x > 0$$

since the integrand is positive (except at x_1 and x_2 where it is null). This contradiction proves the result.

Note here that the condition $w(a) = w(b) = 0$ is not necessary: what it is shown here is that proposition 1.2.1 holds even for functions $w(x)$ that vanish at the boundary. Result 1.2.1 is called the *fundemental lemma of the calculus of variations*. This result has many extensions. See [**?**], Chapter 1, §3 for more details.

1.3 Euler-Lagrange equations

Finding the extrema (minima or maxima) of a functional $T(y)$ is similar to finding the extrema of functions $f(x)$. The first variation of a functional T is defined as

$$\delta T(x, y, y') = T(x, y + \delta y, y' + \delta y') - T(x, y, y').$$

Any variation δy can be written as the difference of two admissible functions y_a and y_b. We have then

$$(\delta y)' = (y_a - y_b)' = (y'_a - y'_b) = \delta y'.$$

A development in Taylor series of T around y and y' gives

$$T(x, y + \delta y, y' + \delta y') = T(x, y, y') + \left(\frac{\partial T}{\partial y}\delta y + \frac{\partial T}{\partial y'}\delta y'\right) + \mathcal{O}(\delta^2)$$

where $\mathcal{O}(\delta^2)$ refers to terms containing δy^2, $\delta y'^2$, δy^3... Neglecting those terms, we have

$$\boxed{\delta T(x, y, y') = \frac{\partial T}{\partial y}\delta y + \frac{\partial T}{\partial y'}\delta y'.}$$

T is extremal if and only if its first variation δT is equal to zero for all variations δy. There is a well known version of this result when T has the form

$$T = \int_{x_1}^{x_2} F(x, y, y')dx$$

as it is for the Brachistochrone curve. Using integration by parts, we obtain

$$\begin{aligned}
\delta T &= \int_{x_1}^{x_2} \left(\frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y'\right) dx \\
&= \int_{x_1}^{x_2} \left(\frac{\partial F}{\partial y} - \frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\right)\delta y dx + \frac{\partial F}{\partial y'}\delta y\Big|_{x_1}^{x_2} \\
&= \int_{x_1}^{x_2} \left(\frac{\partial F}{\partial y} - \frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right)\right)\delta y dx = 0 \quad \forall \delta y \in U_0.
\end{aligned}$$

According to the fundamental lemma of calculus of variations 1.2.1, the part of the integrand in parentheses is zero, i.e.

$$\boxed{\frac{\partial F}{\partial y} - \frac{d}{dx}\frac{\partial F}{\partial y'} = 0} \tag{1.4}$$

Equation (1.4) is called the Euler-Lagrange equation. It is an ordinary differential equation, generally non linear, which can be solved to obtain the extremal function $y(x)$.

If F does not depend on x explicitely, equation (1.4) can be simplified. We have

$$\frac{\mathrm{d}F}{\mathrm{d}x} = \frac{\partial F}{\partial y}y' + \frac{\partial F}{\partial y'}y'' + \frac{\partial F}{\partial x},$$

where the last term drops out because F does not depend on x explicitely. Rearranging this yields

$$y'\frac{\partial F}{\partial y} = \frac{\mathrm{d}F}{\mathrm{d}x} - \frac{\partial F}{\partial y'}y''.$$

We then substitue $y'\frac{\partial F}{\partial y}$ into (1.4) to get

$$\frac{\mathrm{d}F}{\mathrm{d}x} - \frac{\partial F}{\partial y'}y'' - y'\frac{\mathrm{d}}{\mathrm{d}x}\frac{\partial F}{\partial y'} = 0.$$

The last term can be expanded as

$$y'\frac{d}{dx}\frac{\partial F}{\partial y'} = \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial y'}y'\right) - \frac{\partial F}{\partial y'}y'',$$

and equation (1.4) can finally be written as

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(F - y'\frac{\partial F}{\partial y'}\right) = 0.$$

that can be integrated to find the so-called Beltrami formula

$$\boxed{F - y' \frac{\partial F}{\partial y'} = C.} \tag{1.5}$$

Let us come back to the Brachistochrone curve. In that case (see Equation (1.3)),

$$F = \sqrt{\frac{1 + y'^2}{2gy}}$$

and Beltrami's formula (1.5) can be used to find the ordinary differential equation of the Brachistochrone curve:

$$\left[1 + (y')^2\right] \, y = \frac{1}{2gC^2} = D \tag{1.6}$$

with $D > 0$. The solution of (1.6) is not obvious. Let us do the following change of variables:

$$y' = \tan t.$$

We have

$$1 + y'^2 = 1 + \tan^2 t = \frac{1}{\cos^2 t}.$$

Then

$$y = D \cos^2 t = \frac{D}{2}(1 + \cos 2t).$$

Finding x is then rather simple. We have $y' = \tan t$. We can also derive y explicitely as

$$y' = \frac{\mathrm{d}y}{\mathrm{d}x} = \frac{\mathrm{d}y}{\mathrm{d}t}\frac{\mathrm{d}t}{\mathrm{d}x} = 2D \sin t \cos t \, \frac{\mathrm{d}t}{\mathrm{d}x}.$$

We have finally

$$dx = 2D \, \cos^2 t \, dt \quad \rightarrow \quad x = D\left(t + \frac{1}{2}\sin 2t\right) + c$$

with an integration constant c. Posing $R = D/2$ and $u = 2t$, we have finally the parametric equations of the Brachistochrone curve

$$\begin{aligned} x &= R(u + \sin u) + c \\ y &= R(1 + \cos u). \end{aligned} \tag{1.7}$$

Those equations correspond to a cycloid. A cycloid is the curve traced by a point on the rim of a circular wheel as the wheel rolls along a straight line without slippage. We still have 2 constants R and c to fix. Point $y = 0$ correspond to $u = \pi$. We need $x(\pi) = 0$ which means that $c = -R\pi$. Then, we want the point $\mathbf{x} = \{x_2, y_2\}$ to belong to the curve, which allows to fix R.

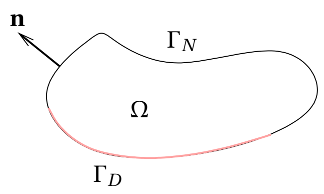

Figure 1.2: A domain Ω and its boundary $\Gamma = \Gamma_D \cup \Gamma_N$.

1.4 The Laplace problem

Consider a domain $\Omega \subset \mathbb{R}^3$ with its boundary $\Gamma = \Gamma_D \cup \Gamma_N$ of normal $\mathbf{n}$ (see Figure 1.2). The Laplace problem consist in finding $u(\mathbf{x}) \in H^1(\Omega)$ solution of

$$\begin{aligned} \nabla \cdot (\kappa \nabla u) &= 0 \text{ on } \Omega \\ u &= \bar{u} \text{ on } \Gamma_D \\ \kappa \nabla u \cdot \mathbf{n} &= \bar{q} \text{ on } \Gamma_N \end{aligned} \tag{1.8}$$

with $\kappa > 0$ function of $\mathbf{x}$. Problem (1.8) correspond to many practical situations: steady state thermal equilibrium, incompressible and irrotational fluids or membrane equilibrium. We now take the problem the other way around. Starting from differential form (1.8), we'd like to build a variational formulation of the Laplace problem.

First use the fundamental lemma of calculus of variations 1.2.1 to claim that

$$\int_\Omega [\nabla \cdot (\kappa \nabla u)] \; w \, dv = 0 \;\; \forall w$$

is equivalent to $\nabla^2 u = 0$ in Ω. Integration by parts in $\mathbb{R}^3$

$$\int_\Omega \nabla \cdot \mathbf{a} b \, dv = -\int_\Omega \mathbf{a} \cdot \nabla b \, dv + \int_\Gamma \mathbf{a} \cdot \mathbf{n} b \, ds.$$

leads to

$$-\int_\Omega \kappa \nabla u \cdot \nabla w \, dv + \int_\Gamma \kappa \nabla u \cdot \mathbf{n} \; w \, ds = 0 \;\; \forall w$$

We then write

$$-\int_\Omega \kappa \nabla u \cdot \nabla w \, dv + \int_{\Gamma_D} \kappa \nabla u \cdot \mathbf{n} \; w \, ds + \int_{\Gamma_N} \kappa \nabla u \cdot \mathbf{n} \; w \, ds = 0 \;\; \forall w.$$

If we choose $w = \delta u$ i.e. if w is a variation, then $w|_{\Gamma_D} = 0$ and we obtain

$$-\int_\Omega \kappa \nabla u \cdot \nabla \delta u \, dv + \int_{\Gamma_N} \kappa \nabla u \cdot \mathbf{n} \, \delta u \, ds = 0 \;\; \forall \delta u.$$

Let us define the space of admissible u's to be

$$U = \{u \in H^1(\Omega) \;\; , u|_{\Gamma_D} = \bar{u}\}$$

and the space of variations

$$U_0 = \{\delta u \in H^1(\Omega) \ \ , u|_{\Gamma_D} = 0\}.$$

The Laplace problem (1.8) is equivalent to find $u \in U$ that verifies

$$\boxed{\int_\Omega \kappa \nabla u \cdot \nabla \delta u \, dv - \int_{\Gamma_N} q \, \delta u \, ds = 0 \ \ \forall \delta u \in U_0.} \tag{1.9}$$

Here, the signs of the terms of the equation has been changed for clarity. Now, equation (1.9) is not written as $\delta T = 0$ with T a functional. It is indeed possible to do so. We define the following functional that is called the potential energy

$$\Pi(u) = \frac{1}{2} \int_\Omega \kappa \, (\nabla u)^2 \ dv - \int_{\Gamma_N} q \ u \ ds. \tag{1.10}$$

Obviously, $\delta\Pi = 0$ is equivalent to (1.9).

Equation (1.9) is a variational or weak formulation of the Laplace problem. It is called weak in opposition to the strong form (1.8) that have solutions for u's that are twice differentiable ($u \in C^2(\Omega)$). Solutions of the weak form (1.9) only require that $\int_\Omega (\nabla u)^2 \, dv < \infty$. The space of functions that have their first derivatives square integrable is called $H^1(\Omega)$. This space is larger than $C^2(\Omega)$ which means that even though every strong solution is a weak solution, there may exist weak solutions that do not correspond to strong solutions: this is the case when κ is discontinuous i.e. when the domain is composed of two distinct "materials". This is one of the reasons why variational forms are prefered to strong forms for solving PDEs on computers.

A last interesting question is to know if u corresponds to a maximum or a minimum of Π. For that, we use a very similar technique as the one we use in calculus: we look at second order derivatives. We have

$$\delta(\delta\Pi(u)) = \frac{1}{2} \int_\Omega \kappa \, (\nabla \delta u)^2 \ dv \geq 0 \tag{1.11}$$

and the solution correspond to the minimum of the potential energy.

1.5 Problems

1.5.1 The Brachistochrone curve with friction

Write the problem of the brachistochrone curve taking into account coulombian friction. First compute the velocity of the car. Then write the functional $T(x, y, y')$. Find the corresponding ODE and solve it.

1.5.2 Geodesics

Another well known minimization problem is the construction of geodesics on a curved surface, meaning the curves of minimal length. Given two points **a** and **v** lying on a surface $S \subset \mathbb{R}^3$, we seek the curve $C \subset S$ that joins them and has the minimal possible length.

Assume the surface to have the simple form

$$z = F(x, y) \quad \to \quad \mathbf{a} = \{a, \alpha, F(a, \alpha)\} \quad \text{and} \quad \mathbf{v} = \{b, \beta, F(b, \beta\}.$$

Assume that C is written in the (x, y) plane as $y = u(x)$. We have of course $z = F(x, u(x))$. Compute the length $L(u)$ of C as a function of u.

CHAPTER 2

Variational principles in mechanics

2.1 Linear Elasticity

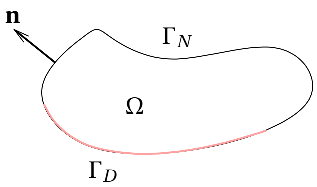

Figure 2.1: A domain Ω and its boundary $\Gamma = \Gamma_D \cup \Gamma_N$.

Consider a domain $\Omega \subset \mathbb{R}^3$ with its boundary $\Gamma = \Gamma_D \cup \Gamma_N$ of normal $\mathbf{n}$ (see Figure 2.1).

The problem of *linear elasticity* consist in finding the deformations ϵ and the internal stresses σ (N/m^2) on every $\mathbf{x}$ of Ω when it is submitted to external loads. We usually distinguish volume loads $\mathbf{f}$ (N/m^3) and surface loads $\mathbf{F}$ (N/m^2). The strong form of the problem consist in finding $\epsilon_{ij}(\mathbf{x})$ and $\sigma_{ij}(\mathbf{x})$ solution of the following equations:

$$\begin{aligned}
\partial_j \sigma_{ij} + f_i &= 0 \text{ on } \Omega && (2.1)\\
\epsilon_{ij} &= \frac{1}{2}\left(\partial_i u_j + \partial_j u_i\right) = u_{(i,j)} && (2.2)\\
\sigma_{ij} &= c_{ijkl}\epsilon_{kl} && (2.3)\\
u_i &= U_i \text{ on } \Gamma_D && (2.4)\\
\sigma_{ij}\, n_j &= F_i \text{ on } \Gamma_N && (2.5)
\end{aligned}$$

The three equations (2.1) (one per direction) are expressing the equilibrium of internal stresses with the external volume loads. Equations (2.5) are expressing surface equilibrium on Γ_N of normal stresses with the external surface loads $\mathbf{F}$. Equations (2.1) and (2.5) both express equilibrium of forces. Those are the equations of static. When a stress field σ verifies (2.1) and (2.5), it is said *statically admissible*. We introduce the space of statically admissible stresses $\mathscr{S}$ as:

$$\mathscr{S} = \left\{\sigma(\mathbf{x}) \mid \partial_j \sigma_{ij} + f_i = 0 \; \forall \mathbf{x} \in \Omega, \; \sigma_{ij}\; n_j = F_i \; \forall \mathbf{x} \in \Gamma_F\right\}.$$

It is now possible to introduce the space variations of σ as

$$\mathscr{S}_0 = \left\{\delta\sigma(\mathbf{x}) \mid \partial_j \delta\sigma_{ij} = 0 \; \forall \mathbf{x} \in \Omega, \; \delta\sigma_{ij}\; n_j = 0 \; \forall \mathbf{x} \in \Gamma_F\right\}$$

Clearly, if $\sigma \in \mathscr{S}$ and $\delta\sigma \in \mathscr{S}_0$, then $\sigma + \delta\sigma \in \mathscr{S}$ and $\delta\sigma$ is a variation.

In continium mecahnics, we imagine Ω to be composed of a set of infinitesimal volumes or material points. Each volume is assumed to be connected to its neighbors without any gaps or overlaps. Certain mathematical conditions have to be satisfied to ensure that gaps/overlaps do not develop when a continuum body is deformed. A body that deforms without developing any gaps/overlaps is called a compatible body. Equations (2.2) and (2.4) are compatibility equations. Equations (2.2) say that, whenever ϵ is expressed as $\epsilon_{ij} = u_{(i,j)}$ where $\mathbf{u}$ is a displacement field, then no gaps/overlaps can appear in the continuum. Then, $\mathbf{u}$ should also be compatible with external kinematical constraints or supports. Equations (2.4) express external compatibility conditions. The space kinematically admissible displacement fields is defiend as

$$\mathscr{U} = \{\mathbf{u}(\mathbf{x}) \mid \mathbf{u} \in U(\Omega), \; u_i = U_i \; \forall \mathbf{x} \in \Gamma_D\}.$$

Here, $U(\Omega)$ a vector-valued function space that will be described in more details later. For now, let's assume that U is a space of functions that are sufficiently smooth such that no gaps/overlaps can appear in the continuum. We then introduce the space of variations:

$$\mathscr{U}_0 = \{\delta\mathbf{u}(\mathbf{x}) \mid \delta\mathbf{u} \in U(\Omega), \; \delta u_i = 0 \; \forall \mathbf{x} \in \Gamma_D\}.$$

Clearly, if $\mathbf{u} \in \mathscr{U}$ and $\delta\mathbf{u} \in \mathscr{U}_0$, then $\mathbf{u} + \delta\mathbf{u} \in \mathscr{U}_0$ and $\delta\mathbf{u}$ is a variation.

Finally, equations (2.3) are the linear and elastic constitutive equations. In the isotropic case, they simplify as

$$\sigma_{ij} = \lambda \epsilon_{mm} \delta_{ij} + 2\mu\epsilon_{ij} \tag{2.6}$$

with Lamé coefficients

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad , \quad \mu = \frac{E}{2(1+\nu)},$$

where E and ν are the Young modulus and the Poisson ratio. Constitutive law (2.17) can be inverted as

$$\epsilon_{ij} = \frac{1+\nu}{E}\sigma_{ij} - \frac{\nu}{E}\sigma_{kk}\delta_{ij}.$$

The elastic energy that is stored in Ω due to its deformation can be written as

$$U(\mathbf{u}) = \frac{1}{2}\int_\Omega \sigma_{ij}\epsilon_{ij}\, dv = \frac{1}{2}\int_\Omega \epsilon_{ij}\epsilon_{kl} c_{ijkl}\, dv.$$

It is a functional, as defined in §1. The energy of a system has to be finite. This is indeed the weakest physical principle that defines how smooth a displacement field should be. Deformations ϵ being derivatives of $\mathbf{u}$, then space U is defined as the space of vector-valued fuctions that have their first derivative square integrable on Ω:

$$U = \left\{ \mathbf{u}\{u_1, u_2, u_3\} \mid \int_\Omega (\partial_j u_i)^2 dv < \infty \right\}$$

2.2 The Principle of Virtual Work

Assume $\mathbf{u} \in \mathscr{U}(\Omega)$ a compatible displacement. A variational form of (2.1) can be written as

$$\int_\Omega \left(\partial_j \sigma_{ij} + f_i\right) \delta u_i \, dv = 0 \quad \forall \delta u_i \in \mathscr{U}_0. \tag{2.7}$$

Equations (2.7) implies (2.1) thanks to the fundamental lemma of the calculus of variations 1.2.1. Integration by parts formula

$$\int_\Omega \nabla \cdot \mathbf{f} g dv = -\int_\Omega \mathbf{f} \cdot \nabla g dv + \int_\Gamma \mathbf{f} \cdot \mathbf{n} g ds,$$

applied to (2.7) gives

$$\int_\Omega \left(-\sigma_{ij} \, \partial_j \delta u_i + f_i \, \delta u_i \right) dv + \int_\Gamma \sigma_{ij} \, n_j \, \delta u_i \, ds = 0 \quad \forall \delta u_i \in \mathscr{U}_0. \tag{2.8}$$

Let us decompose the surface integral in two parts:

$$\int_\Gamma \sigma_{ij} \, n_j \, \delta u_i \, ds = \underbrace{\int_{\Gamma_D} \sigma_{ij} \, n_j \, \delta u_i \, ds}_{\delta u_i|_{\Gamma_D}=0} + \underbrace{\int_{\Gamma_N} \sigma_{ij} \, n_j \, \delta u_i \, ds}_{\sigma_{ij} \, n_j|_{\Gamma_N}=F_i}. \tag{2.9}$$

We now take into account that

$$\sigma_{ij} \, \partial_j \delta u_i = \sigma_{ij} \, \delta u_{(i,j)},$$

to obtain the following variational form: find $\mathbf{u} \in \mathscr{U}$ that verifies

$$\boxed{\int_\Omega \left(-\sigma_{ij} \, \delta\epsilon_{ij} + f_i \, \delta u_i\right) \, dv + \int_{\Gamma_F} F_i \, \delta u_i \, ds = 0 \quad \forall \delta u_i \in \mathscr{U}_0.} \tag{2.10}$$

Form (2.10) is called the *principle of virtual work*. Every solution $\mathbf{u}$ that verifies the string form (2.1)-(2.5) is a solution of (2.10). Yet, some weaker solutions do verify (2.10) while not being sufficiently smooth to verify (2.1)-(2.5). Variational formulation (2.10) is the most general form of the equations of elasticity.

2.3 The Principle of Complementary Virtual Work

There exist another variational principle that is actually the dual of (2.10). Assume $\sigma \in \mathscr{S}$ to be a statically admissible stress field and let's write compatibility condition (2.2) in the following variational form :

$$\int_\Omega \epsilon_{ij} \delta\sigma_{ij} dv = \int_\Omega \frac{1}{2}\left(\partial_i u_j + \partial_j u_i\right)\delta\sigma_{ij} \quad \forall \delta\sigma_{ij} \in \mathscr{S}_0. \tag{2.11}$$

We the use the symmetry of σ to write

$$\partial_i u_j \delta\sigma_{ij} = \partial_i u_j \delta\sigma_{ji} = \partial_j u_i \delta\sigma_{ij}$$

which, combined to (2.11) leads to

$$\int_\Omega \epsilon_{ij} \delta\sigma_{ij} dv = \int_\Omega \partial_j u_i \delta\sigma_{ij} \quad \forall \delta\sigma_{ij} \in \mathscr{S}_0. \tag{2.12}$$

The right hand side of (2.12) can be integrated by parts to give

$$\int_\Omega \epsilon_{ij} \delta\sigma_{ij} dv = \int_\Gamma u_i \delta\sigma_{ij} n_j ds - \underbrace{\int_\Omega u_i \partial_j \delta\sigma_{ij} dv}_{=0 \text{ because } \delta\sigma \in \mathscr{S}_0} \quad \forall \delta\sigma_{ij} \in \mathscr{S}_0. \tag{2.13}$$

Let's decompose the surface integral of (2.13) in two parts:

$$\int_\Gamma u_i \delta\sigma_{ij} n_j ds = \underbrace{\int_{\Gamma_U} \delta\sigma_{ij}\; n_j\; u_i\; ds}_{u_i|_{\Gamma_U} = U_i} + \underbrace{\int_{\Gamma_F} \delta\sigma_{ij}\; n_j\; u_i\; ds}_{\delta\sigma_{ij}\; n_j|_{\Gamma_F} = 0}. \tag{2.14}$$

The following formulation is called the principle of complementary virtual work: find $\sigma \in \mathscr{S}$ solution of

$$\boxed{\int_\Omega \epsilon_{ij} \delta\sigma_{ij} dv = \int_{\Gamma_U} U_i \delta\sigma_{ij} n_j ds \quad \forall \delta\sigma_{ij} \in \mathscr{S}_0.} \tag{2.15}$$

Formulation (2.15) is way less useful in practice than (2.10). It actually requires to start with stresses that are statically admissible *a priori*. This requires to find a function space for σ that is in equilibrim with the external forces and this equilibrium involves a partial differential equation that is to be verified *a priori*! Very few numerical methods use (2.15) in practice.

2.4 The Principle of Total Potental Energy

There are cases when formulation (2.10) is equivalent to the minimization of a functional. Assume a linear elastic material. The principle of virtual work can be written as

$$\int_\Omega \left(\epsilon_{ij} c_{ijkl} \sigma_{kl} + f_i\right)\, \delta u_i\; dv = 0 \quad \forall \delta u_i \in \mathscr{U}_0. \tag{2.16}$$

Let us recall the definition of the elastic energy.

$$U = \frac{1}{2}\int_\Omega c_{ijkl}\epsilon_{kl}\epsilon_{ij}dv$$

We define the functional of total potential energy as the difference between the elastic energy U and the work of external loads **f** and **F**:

$$\pi = U - W_{ext} = \frac{1}{2}\int_\Omega c_{ijkl}\epsilon_{kl}\epsilon_{ij}dv - \int_\Omega f_i u_i \, dv - \int_{\Gamma_N} F_i u_i \, ds,$$

We have:

$$\delta\pi = \int_\Omega c_{ijkl}\epsilon_{kl}\delta\epsilon_{ij} \, dv - \int_\Omega f_i \delta u_i \, dv - \int_{\Gamma_N} F_i \delta u_i \, ds$$

which is indeed formulation (2.10). We have that

$$\delta\pi = 0 \quad , \quad \forall \delta u_i \in \mathscr{U}_0$$

is equivalent to the principle of virtual work (2.10).

It is easy to see that static equilibrium correspond to the minimum of π. For that, we compute

$$\delta^2\pi = \delta(\delta\pi) = \int_\Omega c_{ijkl}\delta\epsilon_{ij}\delta\epsilon_{kl}dv = 2U(\delta\epsilon).$$

Elastic energy being positive, the equilibrium correspond to the minimum of the total potential energy.

2.5 The Principle of Total Complementary Potental Energy

Let's again consider the linear elastic case. The total complementary potental energy is the following functional:

$$\pi^*(\sigma) = U - W^*_{ext} = \frac{1}{2}\int_\Omega d_{ijkl}\sigma_{kl}\sigma_{ij}dv - \int_{\Gamma_D} \sigma_{ij} n_j U_i \, ds.$$

The variation of π^* is

$$\delta^{(1)}\pi^* = \int_\Omega d_{ijkl}\sigma_{kl}\delta\sigma_{ij}dv - \int_{\Gamma_U} \delta\sigma_{ij} n_j U_i \, ds.$$

We have then

$$\delta\pi^* = 0 \quad , \quad \forall \delta_{sigma} \in \mathscr{S}_0$$

is equivalent to the principle of complementary virtual work. Again, this extremum correspond to a minimum.

2.6 Matrix notations

Strains ϵ, stresses σ or Hooke's law (2.3) can be written in index notations. Matrix notations can also be used, especially if we restrict our interest to standard euclidian coordinates. In this case, we can write Hooke's law in the following form:

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{xy} \\ 2\epsilon_{xz} \\ 2\epsilon_{yz} \end{pmatrix} \tag{2.17}$$

or in compact form

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\epsilon}$$

with **C** that is called Hooke's rigidity matrix. The inverse relation is written

$$\begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{xy} \\ 2\epsilon_{xz} \\ 2\epsilon_{yz} \end{pmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1+\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & 1+\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & 1+\nu \end{bmatrix} \begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{pmatrix} \tag{2.18}$$

or in compact form

$$\boldsymbol{\epsilon} = \mathbf{C}^{-1}\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\sigma}$$

where **D** is called Hooke's matrix of flexibility or complicance. The principle of virtual work can be written as: find $\mathbf{u} \in \mathscr{U}$ solution of

$$\boxed{\int_\Omega \boldsymbol{\sigma}^T(\mathbf{u})\ \boldsymbol{\epsilon}(\delta\mathbf{u})\ dv = \int_\Omega \mathbf{f}\,\delta\mathbf{u}\ dv + \int_{\Gamma_N} \mathbf{F}\,\delta\mathbf{u}\ ds\ \ \forall\delta\mathbf{u} \in \mathscr{U}_0.} \tag{2.19}$$

The principle of complementary virtual work can be written as: find $\boldsymbol{\sigma} \in \mathscr{S}$ solution of

$$\boxed{\int_\Omega \boldsymbol{\epsilon}^T\ \delta\boldsymbol{\sigma}\ dv = \int_{\Gamma_D} (\boldsymbol{\sigma}\cdot\mathbf{n})\cdot\mathbf{u}\ ds\ \ \forall\delta\boldsymbol{\sigma} \in \mathscr{S}_0.} \tag{2.20}$$

2.7 The hypercircle of Prager and Synge

Assume $\boldsymbol{\epsilon}_{ex}, \boldsymbol{\sigma}_{ex}$ to be the exact solution of a given problem of elasticity. Let $\boldsymbol{\sigma} \in \mathscr{S}$ be a statically admissible stress and $\mathbf{u} \in \mathscr{U}$ be a compatible displacement. We write now the following functional

$$E^2(\boldsymbol{\epsilon}(\mathbf{u}), \boldsymbol{\sigma}) = \frac{1}{2}\int_\Omega (\boldsymbol{\sigma} - \mathbf{C}\boldsymbol{\epsilon})^T\,\mathbf{D}\,(\boldsymbol{\sigma} - \mathbf{C}\boldsymbol{\epsilon})\,dv. \tag{2.21}$$

which is called the error in constitutive relation. Functional E of (2.21) is non negative and is equal to zero only if $\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\epsilon}$ i.e. if the constitutive law is fullfilled. We then add $\boldsymbol{\sigma}_{ex} - \mathbf{C}\boldsymbol{\epsilon}_{ex}$ in (2.21) to obtain

$$\begin{aligned} E^2(\boldsymbol{\epsilon}(\mathbf{u}),\boldsymbol{\sigma}) &= \frac{1}{2}\int_\Omega (\boldsymbol{\sigma} - \mathbf{C}\boldsymbol{\epsilon} + \boldsymbol{\sigma}_{ex} - \mathbf{C}\boldsymbol{\epsilon}_{ex})^T \mathbf{D}\,(\boldsymbol{\sigma} - \mathbf{C}\boldsymbol{\epsilon} + \boldsymbol{\sigma}_{ex} - \mathbf{C}\boldsymbol{\epsilon}_{ex})\,dv \\ &= E^2(\boldsymbol{\epsilon}_{ex},\boldsymbol{\sigma}) + E^2(\boldsymbol{\epsilon},\boldsymbol{\sigma}_{ex}) + \int_\Omega (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})^T(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{ex})dv \qquad (2.22) \end{aligned}$$

Let us expand the last term of (2.22) as

$$\begin{aligned} &\int_\Omega (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})^T(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{ex})dv \overset{\mathbf{u},\mathbf{u}_{ex}\in\mathscr{U}}{=} \int_\Omega (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})^T 1/2(\nabla + \nabla^T)(\mathbf{u} - \mathbf{u}_{ex})dv = \\ &-\int_\Omega \underbrace{\nabla\,(\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})^T}_{=0 \text{ car } \boldsymbol{\sigma},\boldsymbol{\sigma}_{ex}\in\mathscr{S}}\;(\mathbf{u} - \mathbf{u}_{ex})dv + \int_\Gamma (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})\,\mathbf{n}\,(\mathbf{u} - \mathbf{u}_{ex})ds = \\ &\int_{\Gamma_F} \underbrace{(\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})\,\mathbf{n}}_{=0 \text{ car } \boldsymbol{\sigma},\boldsymbol{\sigma}_{ex}\in\mathscr{S}}\;(\mathbf{u} - \mathbf{u}_{ex})ds + \int_{\Gamma_U} (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex})\,\mathbf{n}\;\underbrace{(\mathbf{u} - \mathbf{u}_{ex})}_{=0 \text{ car } \mathbf{u},\mathbf{u}_{ex}\in\mathscr{U}}\;ds \\ &= 0. \qquad (2.23) \end{aligned}$$

We should regognize now that $\boldsymbol{\sigma} - \boldsymbol{\sigma}_{ex} \in \mathscr{S}_0$ and that $\mathbf{u} - \mathbf{u}_{ex} \in \mathscr{S}_0$: relation (2.23) expresses then the orthogonality of $\delta\mathbf{u}$'s and $\delta\boldsymbol{\sigma}$'s. The hypercircle theorem of Prager and Synge writes then

$$E^2(\boldsymbol{\epsilon}(\mathbf{u}),\boldsymbol{\sigma}) = E^2(\boldsymbol{\epsilon}_{ex},\boldsymbol{\sigma}) + E^2(\boldsymbol{\epsilon},\boldsymbol{\sigma}_{ex}). \qquad (2.24)$$

It says that the square of the distance (measured in term of E) between two admissible fields $\boldsymbol{\epsilon}(\mathbf{u})$ and $\boldsymbol{\sigma}$ is equal to the sum of the square of the distances between $\boldsymbol{\epsilon}_{ex}$ and $\boldsymbol{\sigma}$ and between $\boldsymbol{\epsilon}$ and $\boldsymbol{\sigma}_{ex}$). This has a graphical interpretation (see Figure 2.2).

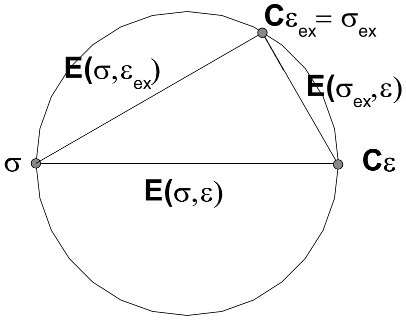

Figure 2.2: Hypercircle of Prager and Synge

CHAPTER 3

Trusses

3.1 Kinematic model of a rod.

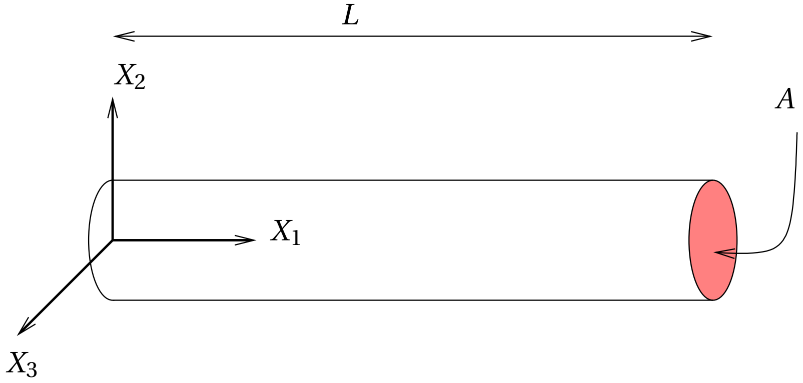

Figure 3.1: A rod with its local frame.

We are going to develop structural elements starting from the simplest one. The kinematic behavior of each element will be defined first using its local or natural frame $\mathbf{X} = \{X_1, X_2, X_3\}$ as $\mathbf{U} = \{U_1, U_2, U_3\}$ with

$$\begin{aligned} U_1 &= U_1(X_1, Y_1, Z_1), \\ U_2 &= U_2(X_1, Y_1, Z_1), \\ U_3 &= U_3(X_1, Y_1, Z_1). \end{aligned}$$

The rod is the simplest structural element (see Figure 3.1): we use X_1 for its principal axis. Rods only accomodate axis loads (traction or compression). A rod can only have displacements along its principal axis:

$$
\begin{aligned}
U_1 &= U_1(X_1) \\
U_2 &= 0 \\
U_3 &= 0
\end{aligned}
$$

Deformations can be written as

$$
\epsilon = \left\{ \begin{array}{c} \frac{\mathrm{d}U_1}{\mathrm{d}X_1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right)
$$

The energy of deformation of the bar is written as

$$
U = \frac{1}{2}\int_{\Omega} E \left(\frac{\mathrm{d}U_1}{\mathrm{d}X_1}\right)^2 d\mathbf{X}. \tag{3.1}
$$

Assuming that the bar has a constant section $A = \int dX_2 dX_3$ and that Young modulus does not depend on $\mathbf{X}$, formula (3.1) simplifies to

$$
U = \frac{EA}{2}\int_0^L \left(\frac{\mathrm{d}U_1}{\mathrm{d}X_1}\right)^2 dX_1 = \frac{EA}{2}\int_0^L \left(U_1'\right)^2 dX_1. \tag{3.2}
$$

Let us now use results of §2 to establish solutions for simple configurations. Assume first that our rod is fixed at its origin $X_1 = 0$ and that a force $F[N]$ is applied to it at its end $X_1 = L$. We have

$$
\pi = \frac{EA}{2}\int_0^L \left(U_1'\right)^2 dX_1 - FU_1(L).
$$

We have

$$
\delta\pi = EA\int_0^L U_1'\delta U_1' \, dX_1 - F\delta U_1(L) = 0 \ \ \forall \delta U_1.
$$

Integration by parts gives

$$
-EA\int_0^L U_1''\delta U_1 \, dX_1 + EAU_1'\delta U_1\big|_0^L - F\delta U_1(L) = 0 \ \ \forall \delta U_1.
$$

Regrouping terms and taking into account that $\delta U_1(0) = 0$, we have

$$
EA\int_0^L U_1''\delta U_1 \, dX_1 + (F - EAU_1')\delta U_1(L) = 0 \ \ \forall \delta U_1.
$$

First, we choose a family of δU_1 that are zero at both ends. This means that

$$
\int_0^L U_1''\delta U_1 \, dX_1 = 0 \ \ \forall \delta U_1.
$$

The fundamental lemma of the calculus of variation ensure us that

$$U_1'' = 0 \quad \rightarrow \quad U_1 = \alpha + \beta X_1$$

with α and β two constants. We have first to choose $\alpha = 0$ because U_1 is kinematically admissible so that $U_1(0) = 0$.

Now, let us consider the other variations that are non zero on $X_1 = L$. We have $EAU_1' = F$ for $X_1 = L$. This means that $EA\beta = F$. The solution is then

$$U_1 = \frac{F}{EA} X_1.$$

Let's now assume that the rod is submitted to its own weight. Assume that the gravity acts on the positive X_1 direction and that the rod is made of a material of density ρ. Gravity is a volume force of amplitude ρg. We have

$$\pi(U_1) = \frac{EA}{2} \int_0^L \left(U_1'\right)^2 dX_1 - A \int_0^L \rho g U_1 dX_1.$$

We then find

$$\delta\pi = A\left(\int_0^L \left[EU_1'\delta U_1' - \rho g \delta U_1\right] dX_1\right) = 0 \;\; \forall \delta U_1.$$

Integration by parts give

$$\int_0^L (-EU_1'' - \rho g)\delta U_1 dX_1 + EU_1'\delta U_1(L) = 0 \;\; \forall \delta U_1.$$

We finally get

$$U_1(X_1) = -\frac{\rho g}{2E} X_1^2 + \alpha X_1 + \beta.$$

Condition $U_1(0) = 0$ leads to $\beta = 0$. Condition $U_1'(L) = 0$ leads to

$$-\frac{\rho g L}{E} + \alpha = 0.$$

Finally, we get

$$\boxed{U_1(X_1) = \frac{\rho g}{2E}\left(2LX_1 - X_1^2\right).} \tag{3.3}$$

It is indeed possible to compute the normal effort in every section X_1 of the rod as

$$N = A\sigma_{11} = EA\epsilon_{11} = EA\frac{\mathrm{d}U_1}{\mathrm{d}X_1} = \rho g A(L - X_1).$$

3.2 The Ritz method

The Ritz method is a direct method to find an approximate solution for boundary value problems. Let's come back to the problem of a rod submitted to its own weight.

Exact solution of the problem is given in (3.3). Assume that we look for an approximate solution of the following form

$$U_1^r(X_1) = \alpha + \beta X_1. \tag{3.4}$$

We'd like to use the principle of total potential energy. For that, we need U_1^r to be compatible: constant α has then to be equal to zero for having $U_1^r(0) = 0$. This last form $U_1^r = \beta X_1$ is of course an approximation of the exact solution. Taking into account the approximation, the total potential energy of the system can be written as

$$\pi(U_1^r) = \frac{EA}{2}\int_0^L \beta^2 \, dX_1 - A\int_0^L \rho g \beta X_1 \, dX_1 = \frac{EAL\beta^2}{2} + \rho g A \beta \frac{L^2}{2}.$$

In this case, π is a function of one parameter β: it is not a functional anymore. Its minimum can be computed by zeroing the derivative of π with respect to β. After simple calculations, we find

$$\frac{\mathrm{d}\pi}{\mathrm{d}\beta} = 0 \quad \to \quad \beta = \frac{\rho g L}{2E}$$

and

$$U_1^r(X_1) = \frac{\rho g L}{2E} X_1. \tag{3.5}$$

Now let's look at the difference between the exact solution (3.3) and its linear approximation (3.5). It is interresting to see that (3.5) is exact at both end of the rod. This fact is not a stroke of luck.

Let us come back to the problem of elasticity. Assume $\mathbf{u}^r$ to be an approximate solution that has been found by the method of Ritz among a family of parametrized solutions $\mathbf{u}^* \in \mathscr{U}^* \subset \mathscr{U}$. We have

$$\pi(\mathbf{u}^r) \leq \pi(\mathbf{u}^*) \quad , \quad \forall \mathbf{u}^*$$

or, if we assume no volume forces,

$$\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^r) : \mathbf{C} : \epsilon(\mathbf{u}^r) \, dv - \int_{\Gamma_N} \mathbf{F} \cdot \mathbf{u}^r \, ds - \frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^*) : \mathbf{C} : \epsilon(\mathbf{u}^*) \, dv + \int_{\Gamma_N} \mathbf{F} \cdot \mathbf{u}^* \, ds \leq 0.$$

This can be re-written as

$$\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^r) : \mathbf{C} : \epsilon(\mathbf{u}^r) \, dv - \frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^*) : \mathbf{C} : \epsilon(\mathbf{u}^*) \, dv - \int_{\Gamma_N} \mathbf{F} \cdot (\mathbf{u}^r - \mathbf{u}^*) \, ds \leq 0. \tag{3.6}$$

Assume that $\mathbf{u} \in \mathscr{U}$ is the exact solution: it verifies the principle of virtual work

$$\int_\Omega \epsilon(\mathbf{u}) : \mathbf{C} : \epsilon(\delta\mathbf{u}) \, dv - \int_{\Gamma_N} \mathbf{F} \cdot (\delta\mathbf{u}) ds = 0 \quad \forall \delta\mathbf{u} \in \mathscr{U}_0.$$

It should be clear at that point that $\delta\mathbf{u} = \mathbf{u}^r - \mathbf{u}^* \in \mathscr{U}_0$ in (3.6) is a variation. Then,

$$\int_{\Gamma_N} \mathbf{F} \cdot (\mathbf{u}^r - \mathbf{u}^*) \, ds = \int_\Omega \epsilon(\mathbf{u}) : \mathbf{C} : \epsilon(\mathbf{u}^r - \mathbf{u}^*) \, dv.$$

We have then

$$\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^r):\mathbf{C}:\epsilon(\mathbf{u}^r)\;dv-\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^*):\mathbf{C}:\epsilon(\mathbf{u}^*)\;dv-\int_\Omega \epsilon(\mathbf{u}):\mathbf{C}:\epsilon(\mathbf{u}^r-\mathbf{u}^*)\;dv\leq 0. \tag{3.7}$$

Strains ϵ are linear operators. Equation (3.7) can be re-written as

$$\boxed{\underbrace{\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^r-\mathbf{u}):\mathbf{C}:\epsilon(\mathbf{u}^r-\mathbf{u})\;dv}_{U(\mathbf{u}^r-\mathbf{u})}-\underbrace{\frac{1}{2}\int_\Omega \epsilon(\mathbf{u}^*-\mathbf{u}):\mathbf{C}:\epsilon(\mathbf{u}^*-\mathbf{u})\;dv}_{U(\mathbf{u}^*-\mathbf{u})}\leq 0.} \tag{3.8}$$

Equation (3.7) has a very interresting and useful interpretation. It says

$$U(\mathbf{u}^r-\mathbf{u})\leq U(\mathbf{u}^*-\mathbf{u})$$

which means that the difference in term of energy between the exact solution $\mathbf{u}$ and its Ritz approximation $\mathbf{u}^r$ is smaller than the difference between $\mathbf{u}$ and any other approximation $\mathbf{u}^*$. This means that the solution obtained by Ritz method is *the best in terms of energy*. It is an orthogonal projection of the exact solution onto the space of approximation where distances are measured in terms of energy (see Figure 3.2). It actually also means that, whenever the space of approximation contains the exact solution, Ritz method will find it.

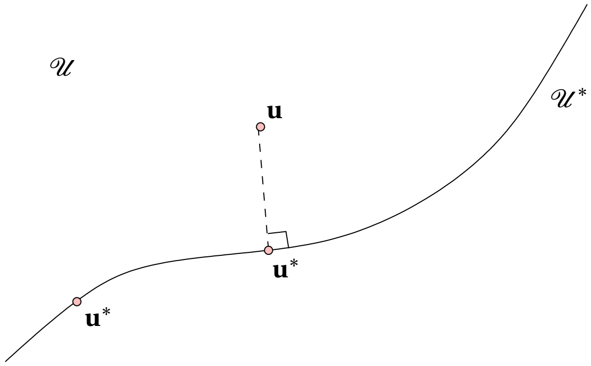

Figure 3.2: Illustration of the orthogonality property (3.8).

3.3 Elastic energy of a truss structure

Equation (3.2) allows to compute the elastic energy of one single rod. In (3.4), we have then assumed that the axial displacement $U_1(X_1)$ of the rod was linear. This is indeed a very common assumption: if the rod is loaded on its two ends, this is indeed the exact solution. When volume forces are involved such as gravity, then this is an approximate solution. The form (3.4) is not the most convenient: coefficients

α and β do not have a clear significations. One more interresting way of expressing the displacement along the rod is to use

$$\begin{aligned} U_1(X_1) &= U_1^1\left(1-\frac{X_1}{L}\right)+U_1^2\frac{X_1}{L} \\ &= U_1^1 M_1(X_1)+U_1^2 M_2(X_1). \end{aligned} \tag{3.9}$$

Here, coefficients U_1^1 and U_1^2 have a clear meaning. For $X_1 = 0$, we get $U_1(0) = U_1^1$ which means that U_1^1 is the displacement of the left end of the rod. Similarly, for $X_1 = L$, we get $U_1(L) = U_1^2$ which means that U_1^2 is the displacement of the right end of the rod. In the finite element method, the unknowns are usually the displacements of the nodes of the structure. This allows in fact to compute the elastic energy of a whole truss structure.

Consider (3.9). The elastic energy (3.2) of this rod can be expressed as

$$U = \frac{EA}{2}\int_0^L (U_1')^2\, dX_1 = \frac{EA}{2L}(U_1^2-U_1^1)^2 = \frac{1}{2}\left(\begin{array}{c} U_1^1 \\ U_1^2 \end{array}\right)^T \left[\begin{array}{cc} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{array}\right]\left(\begin{array}{c} U_1^1 \\ U_1^2 \end{array}\right) \tag{3.10}$$

Writing the elastic energy in matrix form will be very useful in what follows.

A truss consists of straight rods connected at joints or nodes. In a truss, loads are applied to nodes only. A planar truss is a truss for which all members are in a given plane and where all the loads are applied in the same plane. We consider planar trusses first.

The kinematic behavior of a planar truss can be described by its nodal displacements. Assume a truss with N nodes and B bars (or rods). Node i has coordinates $\mathbf{x}^i = \{x_1^i, x_2^i\}$ and its displacement is noted $\mathbf{u}^i = \{u_1^i, u_2^i\}$. Note here that positions and displacements of the nodes are given using the same system of coordinates (we use lower case letters for coordinates in the common euclidian frame). Bar j connects node b_1^j and b_2^j. It has a length L^j, a section A^j and it is made of a material of Young modulus E^j We want now to compute the elastic energy U^j of bar j as a function of global displacements $\mathbf{u}^{b_1^j}$ and $\mathbf{u}^{b_2^j}$. Assume a bar that has its local axis X_1 inclined with an angle θ with the global x_1 axis (see Figure 3.3). Any vector has two sets of coordinates, one expressed in the global frame $\mathbf{v} = \{v_1, v_2\}$ and one in the local frame $\mathbf{V} = \{V_1, V_2\}$. We have

$$\mathbf{V} = \left(\begin{array}{c} V_1 \\ V_2 \end{array}\right) = \left[\begin{array}{cc} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{array}\right]\left(\begin{array}{c} v_1 \\ v_2 \end{array}\right) = [\mathbf{T}]\mathbf{v}.$$

and

$$\mathbf{v} = \left(\begin{array}{c} v_1 \\ v_2 \end{array}\right) = \left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array}\right]\left(\begin{array}{c} V_1 \\ V_2 \end{array}\right) = [\mathbf{T}]^T\mathbf{V}.$$

The elastic energy of rod j can be expressed as

$$U^j = \frac{1}{2}\left(\begin{array}{c} U_1^{b_1^j} \\ U_2^{b_1^j} \\ U_1^{b_2^j} \\ U_2^{b_2^j} \end{array}\right)^T \left[\begin{array}{cccc} \frac{E^jA^j}{L^j} & 0 & -\frac{E^jA^j}{L^j} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{E^jA^j}{L^j} & 0 & \frac{E^jA^j}{L^j} & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]\left(\begin{array}{c} U_1^{b_1^j} \\ U_2^{b_1^j} \\ U_1^{b_2^j} \\ U_2^{b_2^j} \end{array}\right) = \frac{1}{2}(\mathbf{U}^j)^T[\mathbf{K}^j](\mathbf{U}^j)$$

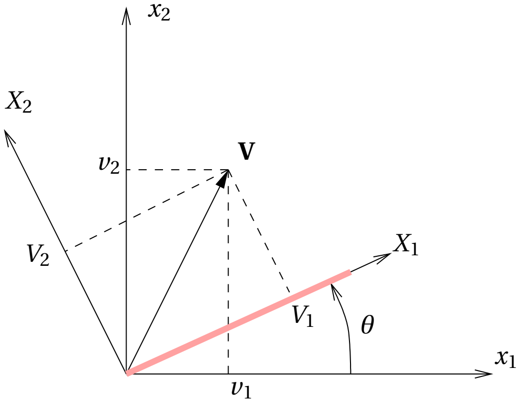

Figure 3.3: Local and global frames

where $[\mathbf{K}^j]$ is the local stiffness matrix of bar j and $(\mathbf{U}^j)$ its local displacement vector. The zeros in the local stifness matrix indicate that no energy is stored for any displacement along the local X_2 axis. We use then the transformation formula $(\mathbf{U}^j) = [\mathbf{T}^j](\mathbf{u}^j)$ to express the energy as a function of global displacements $(\mathbf{u}^j)$

$$U^j = \frac{1}{2}(\mathbf{U}^j)^T[\mathbf{K}^j](\mathbf{U}^j) = \frac{1}{2}(\mathbf{u}^j)^T[\mathbf{T}^j]^T[\mathbf{K}^j][\mathbf{T}^j](\mathbf{u}^j) = \frac{1}{2}(\mathbf{u}^j)^T[\mathbf{k}^j](\mathbf{u}^j).$$

Here,

$$[\mathbf{k}^j] = [\mathbf{T}^j]^T[\mathbf{K}^j][\mathbf{T}^j] \tag{3.11}$$

is the global stifness matrix of bar j. Energy is an extensive quantity and the energy of a given truss is computed as

$$U = \sum_{j=1}^{B} U^j.$$

3.4 Global equilibrium of a truss

3.4.1 Stiffness Matrix

Let's now define the global displacement vector

$$(\mathbf{u}) = \begin{pmatrix} u_1^1 \\ u_2^1 \\ u_1^2 \\ u_2^2 \\ \vdots \\ u_1^N \\ u_2^N \end{pmatrix}$$

as a vector of size $2N$ that contains all nodal displacements of the truss. Node i of the truss correspond to entries $2i-1$ and $2i$, entry $2i-1$ corresponding to the displacement along x_1 and entry $2i$ corresponding to the displacement along x_2.

Our aim now is to express the elastic energy of the truss as the quadratic form

$$U = \frac{1}{2}(\mathbf{u})^T[\mathbf{k}](\mathbf{u})$$

where $[\mathbf{k}]$ is the global stifness matrix of the truss of size $2N \times 2N$. The energy of bar j can also be written as the quadratic form

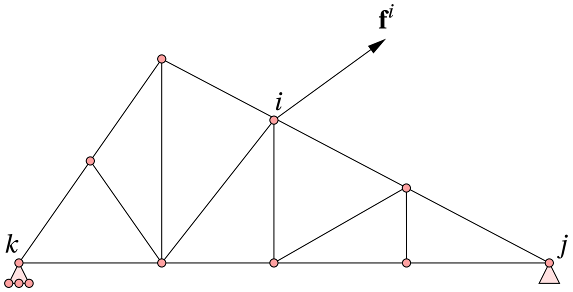

Figure 3.4: A truss with loads and fixations.

$$U^j = \frac{1}{2}(\mathbf{u})^T[\mathbf{k}_e^j](\mathbf{u})$$

with $[\mathbf{k}_e^j]$ the extended stiffness matrix i.e. a $2N \times 2N$ matrix that is essentially filled up with zeros and where the only 16 non zero entries are the ones of $[\mathbf{k}^j]$. The indices of the 4 degrees of freedom of bar j are

$$\iota = \{\iota_1, \iota_2, \iota_3, \iota_4\} = \{2b_1^j - 1, 2b_1^j, 2b_2^j - 1, 2b_2^j\}$$

and the 16 non zero entries are

$$[\mathbf{k}_e^j]_{\iota_i,\iota_j} = [\mathbf{k}^j]_{i,j} \quad , \quad i,j = 1,\ldots,4.$$

The elastic energy of a truss is therefore

$$U = \sum_{j=1}^{B} U^j = \frac{1}{2}\sum_{j=1}^{B}(\mathbf{u})^T[\mathbf{k}_e^j](\mathbf{u}) = \frac{1}{2}(\mathbf{u})^T\left[\sum_{j=1}^{B}[\mathbf{k}_e^j]\right](\mathbf{u}) = \frac{1}{2}(\mathbf{u})^T[\mathbf{k}](\mathbf{u})$$

with $[\mathbf{k}] = \sum_{j=1}^{B}[\mathbf{k}_e^j]$ i.e. the global stiffness matrix $[\mathbf{k}]$ is simply “assembled” as the sum of extended stiffness matrices $[\mathbf{k}_e^j]$.

3.4.2 Force Vector

A truss is loaded at its vertices. If $\mathbf{f}^i = \{f_1^i, f_2^i\}$ (see Figure 3.4) is the external load at vertex i, we define the global force vector as

$$(\mathbf{f}) = \begin{pmatrix} f_1^1 \\ f_2^1 \\ f_1^2 \\ f_2^2 \\ \vdots \\ f_1^N \\ f_2^N \end{pmatrix}.$$

The work of external loads can the be expressed as

$$W_{ext} = (\mathbf{u})^T (\mathbf{f})$$

and the total potential energy of the truss is

$$\boxed{\pi = \frac{1}{2}(\mathbf{u})^T [\mathbf{k}] (\mathbf{u}) - (\mathbf{u})^T (\mathbf{f}).} \tag{3.12}$$

3.4.3 Supports

The natural thing to do now could be to minimize π with respect to $(\mathbf{u})$. Yet, the princple of the minimum of total potential energy only applies to displacements that are admissible i.e. that verify *a priori* the essential boundary conditions i.e. the displacements that are compatible with the supports. On the truss of Figure 3.4, displacement of node j along x_2 should be equal to zero: $(\mathbf{u})_{2j} = 0$. Similarly, the displacement vector of node k should be equal to zero: $(\mathbf{u})_{2k-1} = (\mathbf{u})_{2k} = 0$.

Lagrange multipliers

The most elegant way of dealing with supports is to use constrained optimization. Assume that the supports of the truss consist in M linear constraints

$$[\mathbf{c}](\mathbf{u}) = (\bar{\mathbf{u}}) \tag{3.13}$$

where $[\mathbf{c}]$ is a $M \times 2N$ matrix of full rank and where $(\bar{\mathbf{u}})$ is the right hand side i.e. a vector of size M.

In mathematical optimization, *the method of Lagrange multipliers* is a strategy for finding the local maxima and minima of a function subject to equality constraints.

For instance, consider the optimization problem maximize $f(x, y)$ subject to $g(x, y) = 0$. We introduce a new variable λ called a *Lagrange multiplier* and study the Lagrangian defined by

$$\mathscr{L}(x, y, \lambda) = f(x, y) + \lambda g(x, y).$$

We then solve

$$\nabla_{x,y,\lambda}\mathscr{L}(x,y,\lambda)=0.$$

This is the method of Lagrange multipliers. Note that $\nabla_\lambda\mathscr{L}(x,y,\lambda)=0$ of course implies $g(x,y)=0$.

We apply the method of Lagrange multipliers to our truss problem. Let us define $(\boldsymbol{\lambda})$ to be a M-vector that contains the Lagrange multipliers. We define the Lagrangian

$$\mathscr{L}((\mathbf{u}),(\boldsymbol{\lambda}))=\frac{1}{2}(\mathbf{u})^T[\mathbf{k}](\mathbf{u})-(\mathbf{u})^T(\mathbf{f})+[[\mathbf{c}](\mathbf{u})-(\bar{\mathbf{u}})]^T(\boldsymbol{\lambda}). \tag{3.14}$$

We then look for a stationary point of $\mathscr{L}$:

$$\frac{\partial\mathscr{L}}{\partial(\mathbf{u})}=(\mathbf{0})\quad\rightarrow\quad[\mathbf{k}](\mathbf{u})-(\mathbf{f})+[\mathbf{c}]^T(\boldsymbol{\lambda})=(\mathbf{0}),$$

$$\frac{\partial\mathscr{L}}{\partial(\boldsymbol{\lambda})}=(\mathbf{0})\quad\rightarrow\quad[\mathbf{c}](\mathbf{u})-(\bar{\mathbf{u}})=(\mathbf{0}).$$

The following system of $2N+M$ equations allows to find $(\mathbf{u})$ and $(\boldsymbol{\lambda})$ in one linear system solve:

$$\boxed{\begin{bmatrix}[\mathbf{k}] & [\mathbf{c}]^T\\ [\mathbf{c}] & [\mathbf{0}]\end{bmatrix}\begin{pmatrix}(\mathbf{u})\\(\boldsymbol{\lambda})\end{pmatrix}=\begin{pmatrix}(\mathbf{f})\\(\bar{\mathbf{u}})\end{pmatrix}.} \tag{3.15}$$

Lagrange multipliers usually have an interresting interpretation. Here $[\mathbf{c}]^T(\boldsymbol{\lambda})$ are *reaction forces* that are associated to the constraints. In the simplest case of the simply supported node j of Figure 3.4, the Lagrange multiplier relative to constraint $u_{2j}=0$ is the vertical reaction force at node j. The formulation (3.15) has the great advantage to provide both displacements and reaction forces!

3.5 Example

Finite elements are not meant to be solved on paper: they should be programmed and solved on a computer. Nevertheless, let's solve one simple truss unsing finite elements and Lagrange multipliers in order to illustrate the method.

It is indeed easy to compute analytically the stiffness matrix (6.3) of bar j with its characteristics A^j, E^j and L^j:

$$[\mathbf{k}^j]=[\mathbf{T}^j]^T[\mathbf{K}^j][\mathbf{T}^j]=$$
$$\frac{E^jA^j}{L^j}\begin{bmatrix}\cos^2(\theta) & \sin(\theta)\cos(\theta) & -\cos^2(\theta) & -\sin(\theta)\cos(\theta)\\ \sin(\theta)\cos(\theta) & \sin^2(\theta) & -\sin(\theta)\cos(\theta) & -\sin^2(\theta)\\ -\cos^2(\theta) & -\sin(\theta)\cos(\theta) & \cos^2(\theta) & \sin(\theta)\cos(\theta)\\ -\sin(\theta)\cos(\theta) & -\sin^2(\theta) & \sin(\theta)\cos(\theta) & \sin^2(\theta)\end{bmatrix}.$$

Consider the truss depicted on Figure 3.5. Assume A and E constant for both bars. The truss has 2 bars and 3 nodes. The stiffness matrix of bar 1 ($b_1^1=2$, $b_2^1=3$, $\theta=$

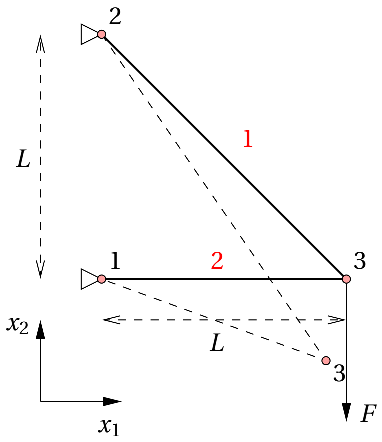

Figure 3.5: A simple truss.

$-\pi/4$) is

$$[\mathbf{k}^1] = \frac{EA}{2\sqrt{2}L}\begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

The stiffness matrix of bar 2 ($b_1^2 = 1$, $b_2^2 = 3$, $\theta = 0$) is

$$[\mathbf{k}^2] = \frac{EA}{L}\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The stiffness matrix of the truss is assembled as

$$[\mathbf{k}] = \underbrace{\frac{EA}{L}\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{[\mathbf{k}_e^1]} + \underbrace{\frac{EA}{2\sqrt{2}L}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 & -1 & 1 \end{bmatrix}}_{[\mathbf{k}_e^2]}.$$

We see here that it is mandatory to impose the value of $(\mathbf{u})_2$ because this degree of freedom is not associated to any stiffness: all elements of row 2 of $[\mathbf{k}]$ are identically equal to 0. In fact, at least 3 constraints have to be imposed to a truss in order to avoid global translation (2 kinematic modes) and rotation (1 kinematic mode). In the case of the truss of Figure 3.5, we find

$$[\mathbf{c}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

and $(\bar{\mathbf{u}}) = [\mathbf{0}]$. The force vector is then computed as

$$(\mathbf{f})^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -F \end{bmatrix}.$$

At that point, solving the problem on paper is cumbersome: we have here a system of 10 equations with 10 unknowns!

In the case of simple constraints such as zeroing a displacement, one way to simplify the resolution on paper is to remove lines and columns of the stiffness matrix that correspond to the displacements that are null. We have then:

$$[\mathbf{k}] = \frac{EA}{2\sqrt{2}L}\begin{bmatrix} 1+2\sqrt{2} & -1 \\ -1 & 1 \end{bmatrix}.$$

Here, the constraints have been implicitely taken into account and finding the static equilibrium of the truss consist in solving

$$(\mathbf{u}) = [\mathbf{k}]^{-1}(\mathbf{f}).$$

We have

$$[\mathbf{k}]^{-1} = \frac{L}{EA}\begin{bmatrix} 1 & 1 \\ 1 & 1+2\sqrt{2} \end{bmatrix}.$$

and the solution is

$$u_1^3 = -\frac{FL}{EA} \quad , \quad u_2^3 = -\frac{FL}{EA}(1+2\sqrt{2}).$$

CHAPTER 4

Programming a truss finite element solver in C

In this section, some C functions are developed that allow to compute displacements and internal efforts of a two-dimensional truss structure.

4.1 Linear system solver

As it was explained in Chapter 3, computing the equilibrium of a truss structure requires to solve a linear system of equations (see (3.15)). Here, we asssume the existence of a linear solver Application Program Interface (API). In such an API, the following functions should be availabe:

- `matrix_t * create_matrix (int n, int m);` This function allows to create a $n \times m$ matrix. Sparse matrices are preferred in finite elements because of the low bandwidth of finite element stifness matrices. Matrix that is returned is initialized with zeros.

- `void delete_matrix (matrix_t *K);` This one removes the memory allocated by `create_matrix`.

- `void add_to_matrix (matrix_t *K, int n, int m, double val);` This one allows to add value `val` to row n and column m of matrix `K`.

- `void solve_linear_system (matrix_t *K, double *f, double *x);` This one solves the linear system $[\mathbf{K}](\mathbf{x}) = (\mathbf{F})$.

4.2 Input and output

A finite element solver of a truss takes as input the following datas:

- An array `double *xy` of size $2N$ that contains the coordinates of the N nodes (or joints) of the truss. We assume that coordinates x_1^j and x_2^j of node j are `xy[2*j]` and `xy[2*j+1]`, $j = 0, \dots, N-1$.

- An array `int *ind` of size $2B$ that contains the topology of the truss i.e. the index of starting and ending nodes of each of the B rods (or bars) of the truss. We assume that bar j's starting node b_1^j is `ind[2*j]` and that bar j's ending node b_2^j is `ind[2*j+1]`, $j = 0, \dots, B-1$.
- An array `double *f` of size $2N$ that contains the nodal forces applied to each of the nodes of the truss. The two components f_1^j and f_2^j of the force that acts on node j are respectively `f[2*j]` and `f[2*j+1]`.
- The way fixations (or supports) are defined is a little more tricky. Here, we propose not to be 100% general. We could indeed ask the user to provide matrix $[\mathbf{c}]$ and vector $(\bar{\mathbf{u}})$ of (3.13) but this is not very intuitive. The truss solver should build $[\mathbf{c}]$ and $(\bar{\mathbf{u}})$ by itself using straightforward informations. We propose to use 2 arrays. Assume that M constraints have to be applied to the truss and that each of the constraints is applied to one node of the truss (this is the non general part). The first array `int *nc` of size M contains all nodes that are associated with constraints. The second array `double *vc` of size $3M$ assigns one vector per constrained node. Vector with components `vc[3*j]` and `vc[3*j+1]` is the direction of the prescribed displacement at that node `nc[j]` and `vc[3*j+2]` is the value of that displacement.
- An array `double *ea` of size B that contains products $E^j A^j$ for each bar.

The output (the results) of the solver are the displacements at all nodes as well as reaction forces corresponding to the constraints:

- An array `double *x` of size $2N$ that contains the nodal displacements of each node of the truss. Displacement u_1^j and u_2^j are respectively `x[2*j]` and `x[2*j+1]`.
- An array `double *r` of size M that contains normal reaction forces.

4.3 Stiffness matrix

The function that computes the stiffness matrix of a bar is given in 4.1. This code is a simple translation of formula (3.16).

4.4 The solver

The function computes the equilibrium of a truss is given in Listings 4.2. The stiffness matrix of the truss is assembled between lines 8 and 16. The force vector is set on line 17. The constraints are set between lines 18 and 27. Note that we consider that the third component `v[2]` is the absolute value of the displacement that is prescribed along direction `v[0], v[1]`. We then put a unit vector in the matrix (lines 22 to 25). If the system cannot be inverted (zero determinant), it actually means that the truss is unstable (hypostatic). This can be due to external hypostaticity (not enough constraints) or internal hypostaticity (frame modes).

```
void stiffnessMatrix (double x1, double y1,
                      double x2, double y2, double ea , double k[4][4])
{
  double t = atan2(y2-y1,x2-x1);
  double c = cos(t);
  double s = sin(t);
  double l = sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));
  double f = ea/l;
  k[0][0] = k[2][2] = c*c*f;
  k[1][1] = k[3][3] = s*s*f;
  k[0][2] = k[2][0] = -c*c*f;
  k[3][1] = k[1][3] = -s*s*f;
  k[0][1] = k[1][0] = k[3][2] = k[2][3] = s*c*f;
  k[3][0] = k[0][3] = k[2][1] = k[1][2] = -s*c*f;
}
```

Listing 4.1: Computation of the stiffness matrix of a bar

Let us now try our truss solver on the same simple truss as the one of §3.5. The code given in 4.3 shows how to define input data for this simple truss. Note that this way of defining data is clearly not the one that should be use to solve large trusses. Computer Aided Design systems like AutoCAD, ArchiCAD, ProEngineer, CATIA or SolidWorks allow to define complex structures in a WYSIWYG fashion. We should in principle be able to connect such a system and get relevant informations that can be automatically translated into our input format.

The computed displacement at node 2 is $u_1^2 = -4.7619e-05$ which is in accordance to what we found in §3.5 i.e.

$$-\frac{FL}{EA} = \frac{1000 \times 1}{210 \times 10^{-5}} = -4.7619 \times 10^{-5}.$$

Similarly, $u_2^2 = -0.000182306$ which is also correct. The value of the 4 lagrangle multipliers is

$$(\boldsymbol{\lambda}) = \{-1000, 0, 1000, -1000\}$$

which is obviously the right value of reactions forces at nodes 0 and 1.

4.5 Post processing

It is indeed possible to compute normal efforts n^j, $j = 0, \ldots, N-1$ in each bar of the truss in a post processing stage. For that, let us give the following interresting interpretation of the local stiffness matrix $[\mathbf{k}^j]$ of bar j. Let us isolate bar j and let's apply to bar j the displacements

$$(\mathbf{u}^j) = \{u_1^{b_1^i}, u_2^{b_1^i}, u_1^{b_2^i}, u_2^{b_2^i}\}.$$

```
int trussSolver (int N, int B, int M, double *xy, int *ind,
                  double *f, int *nc, double *vc, double *ea,
                  double *x, double *r) {
  double stiff [4][4];
  matrix_t *K = create_matrix (2*N+M, 2*N+M);
  double *F = (double*) malloc ((2*N+M)*sizeof(double));
  double *X = (double*) malloc ((2*N+M)*sizeof(double));
  for (int i=0;i<B;i++){
    int n1 = ind[2*i];
    int n2 = ind[2*i+1];
    int indx[4] = {2*n1,2*n1+1,2*n2,2*n2+1};
    stiffnessMatrix ( xy[indx[0]], xy[indx[1]],
                      xy[indx[2]], xy[indx[3]], ea[i] , stiff );
    for (int j=0;j<4;j++) for (int k=0;k<4;k++)
      add_to_matrix(K,indx[j],indx[k],stiff[j][k]);
  }
  for (int i=0;i<2*N;i++) F[i] = f[i];
  for (int i=0;i<M;i++) {
    int indx[2] = {2*nc[i], 2*nc[i] +1};
    double v[3] = {vc[3*i], vc[3*i+1], vc[3*i+2]};
    double norm = sqrt(v[0]*v[0] + v[1]*v[1]);
    add_to_matrix(K,2*N+i,indx[0], v[0]/norm);
    add_to_matrix(K,2*N+i,indx[1], v[1]/norm);
    add_to_matrix(K,indx[0],2*N+i, v[0]/norm);
    add_to_matrix(K,indx[1],2*N+i, v[1]/norm);
    F[2*N+i] = v[2];
  }
  bool result = solve_linear_system (K, F, X);
  if (result == false)printf("ERROR : the truss is not stable\n");
  for (int i=0;i<2*N;i++) x[i] = X[i];
  for (int i=0;i<M;i++) r[i] = X[2*N+i];
  delete_matrix(K);
  free(F);
  free(X);
  return result;
}
```

Listing 4.2: A truss solver

```
int main(void) {
  const double L = 1.0;
  const double F = 1000;
  const double A = 1e-4;
  const double E = 210e9;
  const int N = 3;
  double xy[2*N] = {0,0,0,L,L,0};
  const int B = 2;
  int ind[2*B] = {1,2,0,2};
  double ea[B] = {E*A,E*A};
  double f [2*N] = {0,0,0,0,0,-F};
  const int M = 4;
  int nc[M] = {0,0,1,1};
  double vc[3*M] = {1,0,0,0,1,0,1,0,0,0,1,0};
  double x[2*N];
  double r[M];
  int result = trussSolver (N,B,M, xy,ind,f,nc,vc,ea,x,r);
  return result;
}
```

Listing 4.3: Example of use of the truss solver

that were computed by the truss solver. The product

$$[\mathbf{f}^j] = [\mathbf{k}^j](\mathbf{u}^j)$$

allows to compute two force vectors

$$(\mathbf{f}^j) = \{\underbrace{f_1^{b_1^i}, f_2^{b_1^i}}_{\mathbf{f}^{b_1^i}}, \underbrace{f_1^{b_2^i}, f_2^{b_2^i}}_{\mathbf{f}^{b_2^i}}\}$$

at both ends b_1^i and b_2^i of the bar. Matrix $[\mathbf{k}^j]$ of (3.16) is of rank 2: its rows 1 and 3 are the opposite of each other. This is also true for rows 2 and 4. This means actually that $f_1^{b_1^i} = -f_1^{b_2^i}$ and $f_2^{b_1^i} = -f_2^{b_2^i}$. Then, we have $\mathbf{f}^{b_1^i} = -\mathbf{f}^{b_2^i}$. This makes sense: the bar is in equilibrium. Then it is easy to see that $\mathbf{f}^{b_1^i}$ is aligned with the axis of the bar. Consider a vector $\mathbf{n} = \{\sin\theta, -\cos\theta\}$ that is orthogonal to bar j. It is easy to see that $\mathbf{f}^{b_1^i} \cdot \mathbf{n} = 0$. This result again makes a lot of sense: the bar can only handle axial forces along X_1. The normal effort at both ends can then be computed as

$$n^j = \mathbf{f}^{b_1^i} \cdot \mathbf{t} = 0$$

with $\mathbf{t} = \{\cos\theta, \sin\theta\}$. We choose to use $\mathbf{f}^{b_1^i}$ in order that a positive n^j correspond to a bar in traction. The code given in 4.4 allows to compute normal efforts in every bar of the truss. In our simple example, we find

$$(\mathbf{n}) = \{1414.21, -1000\}$$

which is obviously the right answer.

```
void postPro (int B, double *xy, int *ind,
              double *ea, double *x, double *n) {
  double stiff [4][4];
  for (int i=0;i<B;i++){
    int n1 = ind[2*i];
    int n2 = ind[2*i+1];
    int indx[4] = {2*n1,2*n1+1,2*n2,2*n2+1};
    stiffnessMatrix ( xy[indx[0]], xy[indx[1]],
                      xy[indx[2]], xy[indx[3]], ea[i] , stiff );
    double f [4] = {0,0,0,0};
    for (int j=0;j<4;j++) {
      for (int k=0;k<4;k++) {
          f[j] += stiff[j][k] * x[indx[k]];
      }
    }
    double t = atan2( xy[indx[1]]- xy[indx[3]], xy[indx[0]]-
        xy[indx[2]]);
    n[i] = f[0] * cos(t) + f[1] * sin(t);
  }
}
```

Listing 4.4: Computing normal efforts in all bars

4.6 Example

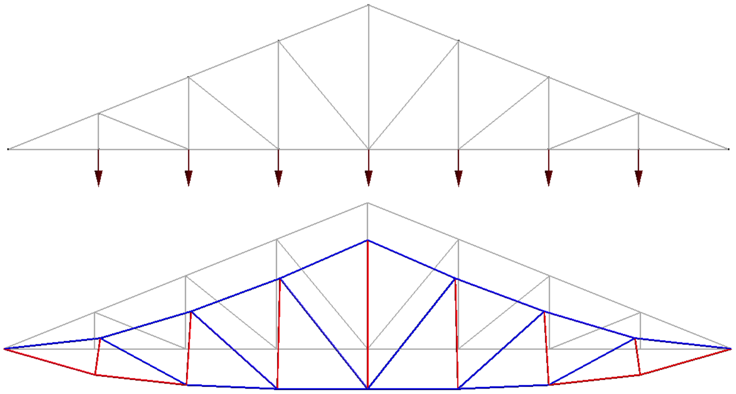

Figure 4.1: A Howe truss. The deformation of the truss is enhanced 100 times. The color correspond to traction (red) and compression (blue)

We consider here a truss with a well known topology that is called a Howe truss (see Figure 4.1. The Howe Truss was designed by William Howe in 1840. It used mostly wood in construction and was suitable for longer spans than the Pratt truss.

Therefore, it became very popular and was considered one of the best designs for railroad bridges back in the day.

CHAPTER 5

Frames

5.1 The Bernoulli-Euler model for beams

The kinematic behavior of a beam involves in-plane flexure. In its local coordinates, the Bernoulli-Euler for a beam writes:

$$\begin{aligned} U_1 &= U_1(X_1) - X_2 \frac{dU_2}{dX_1} \\ U_2 &= U_2(X_1) \\ U_3 &= 0. \end{aligned} \tag{5.1}$$

Here, U_2 represents the displacement of the neutral axis of the beam along X_2 (see Figure 5.1).

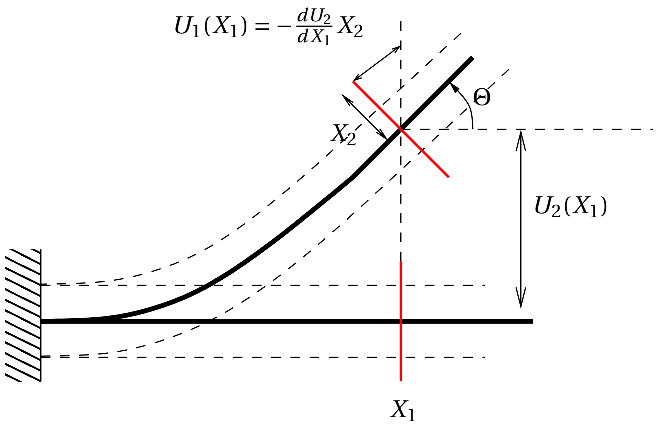

Figure 5.1: The Bernoulli-Euler model for a beam (pure bending).

This model can be seen as the sum of two model: (i) a bar model (3.1) that allows energy to be stored in traction/compression and (ii) a pure flexure model that allows to store flexural energy only. Assume the bar to be of length L, with a bending stiffness EI and a axial stiffness EA. The energy of deformation of the bar is written as

$$U = \frac{1}{2}\int_0^L EA\left(\frac{dU_1}{dX_1}\right)^2 \, dX_1 + \frac{1}{2}\int_0^L EI\left(\frac{d^2U_2}{dX_1^2}\right)^2 \, dX_1. \tag{5.2}$$

The energy of deformation has to be finite. This means that U_1 has to have its first derivative square-integrable. It is possible to prove that this condition is verified for every $U_1 \in C^0$, the space of continuous functions. The flexure term involves the square of the second derivative of U_2. Functions that have their second derivative square-integrable should not only be continuous. They should also have their first derivative continuous. We have to choose $U_2 \in C^1$, the space of continuously derivable functions.

5.2 Finite Elements for Beams

The extra difficulty that arises here comes from that fact that U_2 should be C^1. The most obvious fashion of ensuring that a discretization is C^0 is to use degrees of freedom that are displacements at nodes. For getting a C^1 discretization, one has simply to add degrees of freedom that are the first derivatives of U_2 at nodes. A beam element has therefore 6 degrees of freedom:

- U_1^1 and U_1^2, the two nodal displacements along X_1.
- U_2^1 and U_2^2, the two nodal displacements along X_2.
- Θ_3^1 and Θ_3^2, the two nodal derivatives of U_2 with respect to X_1. Those are noted Θ_3 because theu correspond to rotations around X_3.

Displacement $U_2(X_1)$ depends on 4 parameters U_2^1, U_2^2, Θ_3^1 and Θ_3^2. Cubic polynomials $N_j(X_1)$, $j = 1,\ldots,4$ are used to discretize U_2 as:

$$\begin{aligned} U_2(X_1) &= N_1(X_1)U_2^1 + N_2(X_1)\Theta_3^1 + N_3(X_1)U_2^2 + N_4(X_1)\Theta_3^2 \\ &= \sum_{i=1}^{4} \tilde{U}_i N_i(X_1) \end{aligned} \tag{5.3}$$

with $\tilde{\mathbf{U}} = \{U_2^1, \Theta_3^1, U_2^2, \Theta_3^2\}$. It is easy to computes functions N_j. The sixteen constants that are required to describe the 4 cubic polynomials are selected in order to satisfy the following 16 relation

$$\begin{aligned} (U_2)(0) = U_2^1 &\rightarrow N_1(0) = 1,\; N_2(0) = 0,\; N_3(0) = 0,\; N_4(0) = 0, \\ (U_2')(0) = \Theta_3^1 &\rightarrow N_1'(0) = 0,\; N_2'(0) = 1,\; N_3'(L) = 0,\; N_4'(L) = 0, \\ (U_2)(L) = U_2^2 &\rightarrow N_1(L) = 0,\; N_2(L) = 0,\; N_3(L) = 1,\; N_4(L) = 0, \\ (U_2')(L) = \Theta_3^2 &\rightarrow N_1'(L) = 0,\; N_2'(L) = 0,\; N_3'(L) = 0,\; N_4'(L) = 1. \end{aligned}$$

For example, we can assume

$$N_1(X_1) = A + BX_1 + CX_1^2 + DX_1^3.$$

We have $N1_(0) = 1 \rightarrow A = 1$. Then, $N_1'(0) = 0 \rightarrow B = 0$, $N_1(L) = 0 \rightarrow 1 + CL^2 + DL^3 = 0$ and $N_1'(L) = 0 \rightarrow 2CL + 3DL^2 = 0$. It is easy to determine B and C using

$$\begin{bmatrix} L^2 & L^3 \\ 2L & 3L^2 \end{bmatrix} \begin{pmatrix} C \\ D \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}.$$

We find $C = -3/L^2$ and $D = 2/L^3$. Then, posing $t = \left(\frac{X_1}{L}\right)$, we find

$$N_1(t) = 1 - 3t^2 + 2t^3.$$

After computations, we find

$$\begin{aligned} N_1(t) &= 1 - 3t^2 + 2t^3 \\ N_2(t) &= Lt(t-1)^2 \\ N_3(t) &= t^2(-2t+3) \\ N_4(t) &= -L(1-t)t^2. \end{aligned} \tag{5.4}$$

Polynomials N_i of (5.4) are called *Hermite polynomials*. They are represented in Figure 5.2 Let us now write the U (see (5.2)) using approximation (5.3) for the transversal displacement U_2 and using (3.9) for U_1:

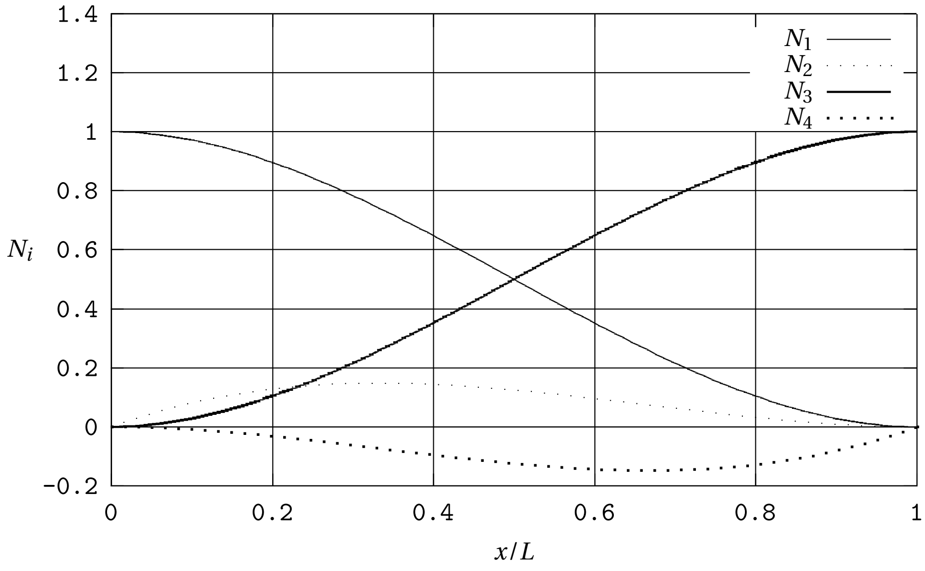

Figure 5.2: Cubic Hermite polynomials.

$$\begin{aligned} U &= \frac{1}{2}\sum_{i=1}^{4}\sum_{j=1}^{4}\tilde{U}_i\tilde{U}_j\int_0^L EI\frac{d^2N_i}{dX_1^2}\frac{d^2N_j}{dX_1^2}\,dX_1 + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2}U_1^iU_1^j\int_0^L EA\frac{dM_i}{dX_1}\frac{dM_j}{dX_1}\,dX_1 \\ &= \frac{1}{2}\sum_{i=1}^{4}\sum_{j=1}^{4}\tilde{U}_i\tilde{U}_jK_{ij}^f + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2}U_1^iU_1^jK_{ij}^a. \end{aligned} \tag{5.5}$$

Assuming EA and EI to be constant along the bar, we have the expressions

$$[\mathbf{K}^f] = \frac{EI}{L^3}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \tag{5.6}$$

and

$$[\mathbf{K}^a] = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{5.7}$$

Grouping the two terms, we get

$$\begin{aligned} U &= \frac{1}{2}\begin{pmatrix} U_1^1 \\ U_2^1 \\ \Theta_3^1 \\ U_1^2 \\ U_2^2 \\ \Theta_3^2 \end{pmatrix}^T \begin{bmatrix} K_{11}^a & 0 & 0 & K_{12}^a & 0 & 0 \\ 0 & K_{11}^f & K_{12}^f & 0 & K_{13}^f & K_{14}^f \\ 0 & K_{21}^f & K_{22}^f & 0 & K_{23}^f & K_{24}^f \\ K_{21}^a & 0 & 0 & K_{22}^a & 0 & 0 \\ 0 & K_{31}^f & K_{32}^f & 0 & K_{33}^f & K_{34}^f \\ 0 & K_{41}^f & K_{42}^f & 0 & K_{43}^f & K_{44}^f \end{bmatrix} \begin{pmatrix} U_1^1 \\ U_2^1 \\ \Theta_3^1 \\ U_1^2 \\ U_2^2 \\ \Theta_3^2 \end{pmatrix} \\ &= \frac{1}{2}(\mathbf{U})^T[\mathbf{K}](\mathbf{U}). \end{aligned} \tag{5.8}$$

We have now to give an expression of the energy in global coordinates i.e. using the 6 global degrees of freedom of the beam. Assume that the bar is inclined of an angle α with respect to the X_1 axis, we can write

$$(\mathbf{u}) = [\mathbf{T}](\mathbf{U})$$

with

$$[\mathbf{T}] = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & 0 & 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.9}$$

We have then

$$U = \frac{1}{2}(\mathbf{U})^T[\mathbf{K}](\mathbf{U}) = \frac{1}{2}(\mathbf{u})^T[\mathbf{T}]^T[\mathbf{K}][\mathbf{T}](\mathbf{u}) = \frac{1}{2}(\mathbf{u})^T[\mathbf{k}](\mathbf{u}).$$

The analytical computation of $[\mathbf{k}] = [\mathbf{T}]^T[\mathbf{K}][\mathbf{T}]$ is not given explicitely. This stiffness matrix is usually computed numerically. The assembly procedure is depicted in Figure 5.3.

5.3 Articulated nodes

Up to now, only rigid nodes have been taken into account. A rigid node can tranfer bending moments between bars. In the case of trusses, all nodes were articulated.

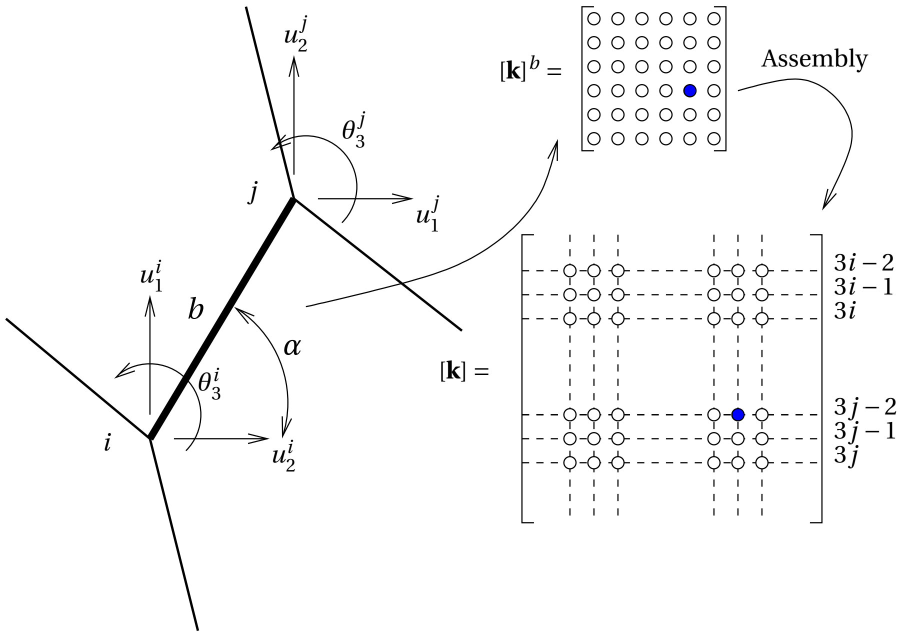

Figure 5.3: The finite element assembly procedure for a beam b.

It is indeed possible to model articulated nodes in a frame i.e. to fix the bending moment at a given end of a bar. Assume that A articulated nodes are present in the frame. A table `int *an` of size $A \times 2$ contains the information about articulated nodes. Here, `an[i][0]`, $i = 1, \ldots, A$, is the number j of the bar that has an articulated node and `an[i][0]` is 1 or 2, depending if the articulated node is b_1^j or b_2^j.

If a bar b has articulated nodes, its assembly process is slightly modified. In a bar with rigid nodes i and j, the degrees of freedom associated with b are

$$\mathbf{u}^b = \{u_1^i, u_2^i, \theta_3^i, u_1^j, u_2^j, \theta_3^j\}$$

with their global numbers $\{3i-2, 3i-1, 3i, 3j-2, 3j-1, 3j\}$ (see Figure 5.3). Assume that node i of bar b is articulated. Then, a new degree of freedom $\theta_3^{b,1}$ has to be defined so that $\mathbf{u}^b$ becomes

$$\mathbf{u}^b = \{u_1^i, u_2^i, \theta_3^{b,1}, u_1^j, u_2^j, \theta_3^j\}.$$

Assume that $\theta_3^{b,1}$ correspond to th kth entry of table `an`, its global number is $3n + k$. Figure 5.4 presents the assembly procedure for a beam with one articulated node. Here is an important note: assume a node i that has M adjacent bars. If all adjacent bars consider this node as articulated, then degree of freedom θ_3^i has to be fixed to zero. This is the case when this node is totally articulated. In this case, no bar has a contribution to degree of freedom θ_3^i.

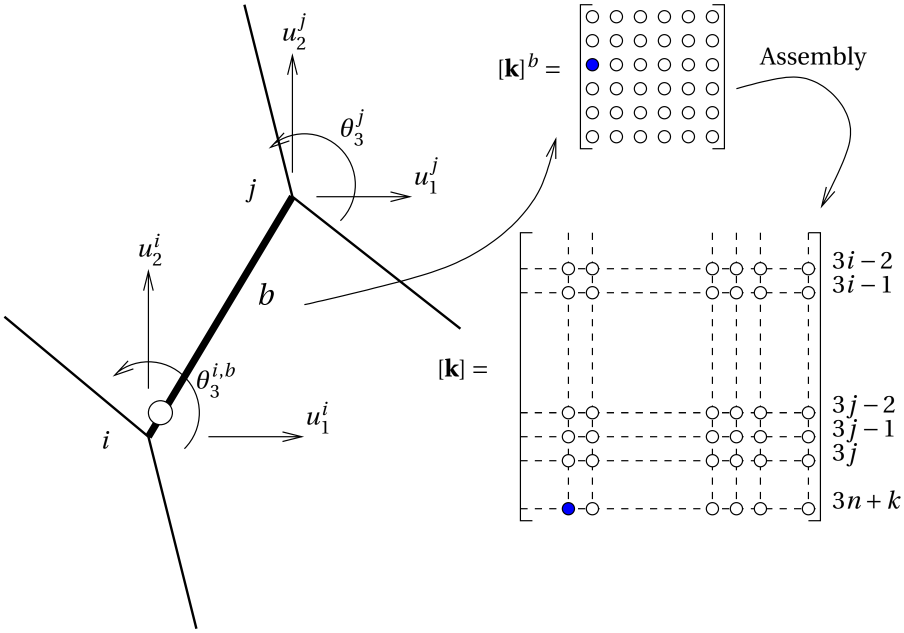

Figure 5.4: The finite element assembly procedure for a beam b with one articulated node.

5.4 What about loads?

The resolution of a planar frame uses the same kind of process as the one we described in Chapters 3 and 4. Here, three degrees of freedom $\mathbf{u}^j = \{u_1^j, u_2^j, \theta_3^j\}$ have to be defined for each node j of the frame. Each bar is associated to a 6×6 stiffness matrix. Forces and moments can be applied to joints of the frame in the same fashion as it was done for trusses. Yet, in the case of frames, continuous distributions of forces $\tau(X_1)[N/m]$ can be applied, for taking into account proper weights of elements or for taking into accound wind forces.

Let us try to solve the problem of a Cantilever beam as it is depicted on Figure 5.5.

Its exact solution can easily be found:

$$U_2(X_1) = -\frac{\tau X_1^2(6L^2 - 4LX_1 + X_1^2)}{24EI}, \quad M(X_1) = EIU_2'' = -\frac{\tau(L^2 - 2LX_1 + X_1^2)}{2}, \quad T(X_1) = M'(X_1) = \tau(X_1 - L).$$

Now let's try to solve this problem using one single finite element. The frame has two nodes as depicted in Figure 5.6.

The work of load $\tau(X_1)$ is computed as

$$W_{ext} = \int_0^L \tau(X_1)U_2(X_1)dX_1 = \sum_{i=1}^4 \tilde{U}_i \int_0^L \tau(X_1)N_i(X_1)dX_1 = \sum_{i=1}^4 \tilde{U}_i\tilde{F}_i$$

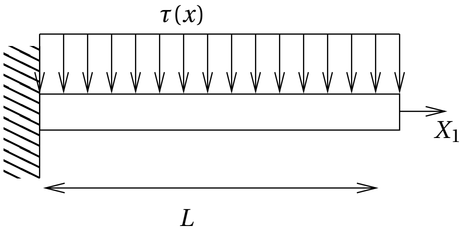

Figure 5.5: A Cantilever Beam.

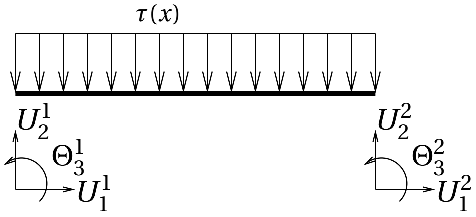

Figure 5.6: A Cantilever Beam with 6 degrees of freedom.

with $(\tilde{\mathbf{U}}) = \{U_2^1, \Theta_3^1, U_2^2, \Theta_3^2\}$ and $(\tilde{\mathbf{F}}) = \{T_2^1, M_3^1, T_2^2, M_3^2\}$ a nodal force vector that is a projection of τ onto the shape functions N_i of the beam. This process can be seen as replacing a continuous distribution of forces $\tau(X_1)$ by nodal forces: two vertical loads T_2^1 and T_2^2 associated to nodes 1 and 2 and two moments of force M_3^1 and M_3^2 associated to nodes 1 and 2 as well. Solving the Cantilever problem with one finite element consist in solving a simplified problem where the continuous load have been replaced by "equivalent" nodal forces and moments. In this case, the finite element method will find the exact solution of the simplified problem, but not to the initial problem with continuous loads. Assume τ to be constant. We have

$$(\tilde{\mathbf{F}}) = \left\{-\tau L/2, \tau L^2/12, -\tau L/2, -\tau L^2/12\right\}.$$

The solution to the simplified problem is solved using finite elements. We have $U_2^1 = \Theta_3^1 = 0$ and

$$\frac{EI}{L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{pmatrix} U_2^2 \\ \Theta_3^2 \end{pmatrix} = \begin{pmatrix} -\tau L/2 \\ -\tau L^2/12 \end{pmatrix}$$

or

$$\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{pmatrix} U_2^2 \\ \Theta_3^2 \end{pmatrix} = -\frac{\tau L^4}{12EI}\begin{pmatrix} 6 \\ L \end{pmatrix}.$$

We find

$$U_2^2 = -\frac{5\tau L^4}{24EI} \quad \text{and} \quad \Theta_3^2 = -\frac{\tau L^3}{3EI}.$$

The finite element solution finally writes

$$U_2(t) = -\frac{5\tau L^4}{24EI} t^2 (-2t+3) + \frac{\tau L^4}{3EI} (1-t)\, t^2.$$

For $X_2 = L$ we of course get $U_2(1) = U_2^2 = -\frac{5\tau L^4}{24EI}$ which is different from the exact value $-\frac{3\tau L^4}{24EI}$.

The best way of improving the results is to split the beam in multiple finite elements. Figure 5.7 shows the simplified finite element problem arising from a 2 element discretization. Here, we see that moments at interior nodes fall down to zero because both left and right beams provide the node opposite contributions. Here, the problem has 6 degrees of freedom (degress of freedom for node 1 are all zero) and it is not possible to solve it by hand.

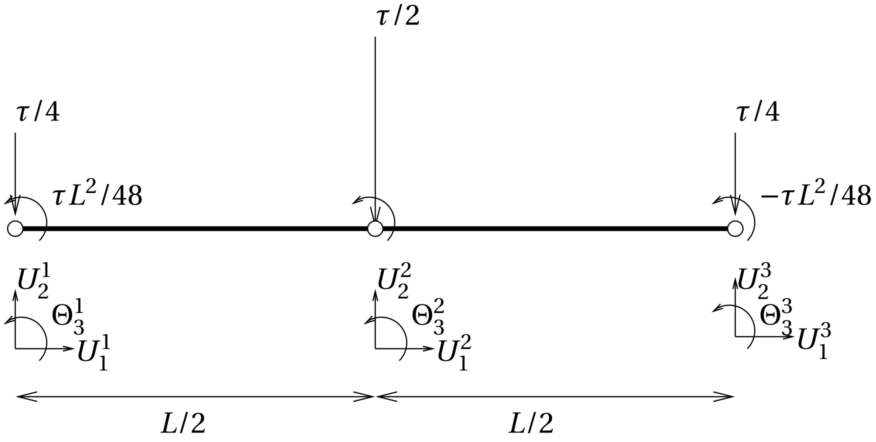

Figure 5.7: Discretization using 2 finite elements

5.5 The Timoshenko model for beams

In §5.1, the Bernoulli-Euler model for beams has been presented. This model does not involve shear. It is interresting at that point to validate that model. Stepan Prokopovych Tymoshenko (December 22, 1878 - May 29, 1972), was a Russian and US engineer. Timoshenko wrote seminal works in the areas of engineering mechanics, elasticity and strength of materials, many of which are still widely used today. The Timoshenko beam theory was developed by Timoshenko early in the 20th century.

Timoshenko has extended Bernoulli's model in the following fashion. Two modes of deformations are taken into account, one that is a pure bending mode like in Bernoulli's model and one which is a pure shear mode (see Figure 5.8) The two

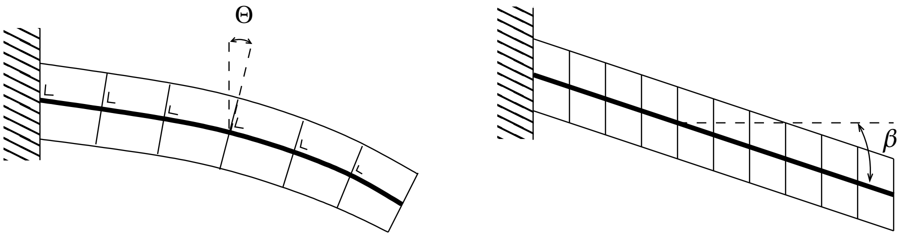

Figure 5.8: Pure bending mechanism (left) and pure shear mechanism (right).

mechanisms can be combines as

$$\frac{dU_2}{dX_1} = \Theta(X_1) + \beta(X_1).$$

ighaThe kinematic model of a Timoshenko beam can therefore be written as:

$$\begin{aligned} & U_1 = -X_2\Theta_3 = -X_2\left[U_2' - \beta\right] \\ & U_2 = U_2(X_1) \\ r \qquad & U_3 = 0. \end{aligned} \tag{5.10}$$

This model allows two non-zero components to the small deformation tensor:

$$\epsilon_{11} = -X_2\Theta'(X_1) \quad , \quad \epsilon_{12} = \frac{1}{2}\beta(X_1).$$

Assuming that shear angle β only depends on X_1 clearly violates the surface equilibrium. No shear forces are acting on upper and lower sides of the beams. There should not be any $\sigma_{12} = G\epsilon_{12}$ on those boundaries. Here,

$$G = \frac{E}{2(1+\nu)}$$

is the shear modulus. Then, $\epsilon_{12} = 1/2\beta$ should be zero as well, which implies that β should be zero on $X_2 = \pm h/2$ where h is the vertical dimension of the beam. So, β should depend on X_2 as well. This hypothesis will be corrected by moderating σ_{11} as

$$\sigma_{11} = KG\beta(x).$$

with $K \leq 1$. There are many ways to choose K. For beams with rectangular sections, the value $K = 5/6$ is usually chosen.

5.6 Finite Elements for Timoshenko beams

The energy of deformation of a timoshenko beam is written as

$$U(U_2, \Theta) = \frac{1}{2}\int_0^L EI\left(\Theta'\right)^2 dX_1 + \frac{1}{2}\int_0^L KGA\left(U_2' - \Theta\right)^2 dX_1 = U^f + U^s.$$

The Timoshenko model involves two distintct fields Θ and U_2 that should be discretized separately. There is a priori no reason to use the same finite element shape functions to interpolate Θ and U_2. We will then assume that U_2 requires M shape functions and that Θ requires N shape functions:

$$U_2(X_1) = \sum_{i=1}^{M} U_2^i M_i(X_1)$$

and

$$\Theta(X_1) = \sum_{i=1}^{N} \Theta^i N_i(x)$$

The energy of deformation is a function of $N+M$ variables:

$$\begin{aligned}
U(U_2^1,\ldots,U_2^M,\Theta_1,\ldots,\Theta_N) &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\Theta^i\Theta^j\int_0^L EI\frac{\mathrm{d}N_i}{\mathrm{d}X_1}\frac{\mathrm{d}N_j}{\mathrm{d}X_1}dX_1 \\
&+ \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M}U_2^iU_2^j\int_0^L KGA\frac{\mathrm{d}M_i}{\mathrm{d}X_1}\frac{\mathrm{d}M_j}{\mathrm{d}X_1}dX_1 \\
&+ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\Theta^i\Theta^j\int_0^L KGA\,N_i\,N_j dX_1 \\
&- \sum_{i=1}^{M}\sum_{j=1}^{N}U_2^i\Theta^j\int_0^L KGA\,\frac{\mathrm{d}M_i}{\mathrm{d}X_1}\,N_j dX_1
\end{aligned}$$

If we assume that E, I, K and A do not vary over the beam, we get, assuming $t = X_1/L$:

$$[\mathbf{K}] = \frac{EI}{L}\begin{bmatrix} 0 & 0 \\ 0 & \left[\int_0^1 \frac{dN_i}{dt}\frac{dN_j}{dt}dt\right]\end{bmatrix} + \frac{GKA}{L}\begin{bmatrix} \left[\int_0^1 \frac{dM_i}{dt}\frac{dM_j}{dt}dt\right] & -L\left[\int_0^1 N_i\frac{dM_j}{dt}dt\right] \\ -L\left[\int_0^1 \frac{dM_j}{dt}N_jdt\right] & L^2\left[\int_0^1 N_iN_jdt\right]\end{bmatrix} = [\mathbf{K}^b]+[\mathbf{K}^s]$$

where $[\mathbf{K}^b]$ is the bending stiffness and where $[\mathbf{K}^s]$ is the shear stiffness.

5.7 Shear locking

We have recognize here that Timoshenko's model leads to energy terms that only involves the first derivative of the unknown fields Θ and U_2. Then, it is theoretically possible to use standard C^0 finite elements for discretizing both unknowns as:

$$\Theta(X_2) = \Theta^1\underbrace{(1-X_1/L)}_{N_1}+\Theta^2\underbrace{X_1/L}_{N_2} = \Theta^1(1-t)+\Theta^2 t$$

and

$$U_2(X_2) = U_2^1\underbrace{(1-X_1/L)}_{M_1}+U_2^2\underbrace{X_1/L}_{M_2} = U_2^1(1-t)+U_2^2 t.$$

It is indeed easy to compute analytically the different terms of the U as

$$\frac{GKA}{L}\left[\int_0^1 \frac{dM_i}{dt}\frac{dM_j}{dt}dt\right] = \frac{GKA}{L}\left[\begin{array}{cc} 1 & -1 \\ -1 & 1 \end{array}\right],$$

$$-L\frac{GKA}{L}\left[\int_0^1 N_i\frac{dM_j}{dt}dt\right] = -\frac{GKA}{2}\left[\begin{array}{cc} -1 & -1 \\ 1 & 1 \end{array}\right],$$

$$L^2\frac{GKA}{L}\left[\int_0^1 N_i N_j dt\right] = \frac{GKAL}{6}\left[\begin{array}{cc} 2 & 1 \\ 1 & 2 \end{array}\right].$$

$$\frac{EI}{L}\left[\int_0^1 \frac{dN_i}{dt}\frac{dN_j}{dt}dt\right] = \frac{EI}{L}\left[\begin{array}{cc} 1 & -1 \\ -1 & 1 \end{array}\right].$$

Let us define α as the ratio between shea stiffness GKA/L and bending stiffness EI/L^3, i.e.,

$$\alpha = \frac{GKAL^2}{EI}.$$

The stiffness matrix can be re-written as

$$[\mathbf{K}] = \frac{GKA}{6L}\left[\begin{array}{cccc} 6 & -6 & 3L & 3L \\ -6 & 6 & -3L & -3L \\ 3L & -3L & 2L^2(1+3/\alpha) & L^2(1-6/\alpha) \\ 3L & -3L & L^2(1-6/\alpha) & 2L^2(1+3/\alpha) \end{array}\right] \tag{5.11}$$

Assume one beam that is clamped on $X_1 = 0$ ($U_2^1 = \Theta^1 = 0$) and that is uniformly loaded (τ [N/m]). The force vector is written as

$$(\mathbf{F}) = \left(\begin{array}{c} \tau L/2 \\ \tau L/2 \\ 0 \\ 0 \end{array}\right)$$

Due to bounary conditions, rows and columns1 and 3 of the stiffness matrix can be removed and the final system to solve is

$$\frac{GKA}{6L}\left[\begin{array}{cc} 6 & 3L \\ 3L & 2L^2(1+3/\alpha) \end{array}\right]\left(\begin{array}{c} U_2^2 \\ \Theta^2 \end{array}\right) = \left(\begin{array}{c} \tau L/2 \\ 0 \end{array}\right). \tag{5.12}$$

with solution

$$U_2^2 = \tau L\frac{2L(1+3/\alpha)}{GKA(1+6/\alpha)} \text{ and } \Theta^2 = \tau L\frac{3}{GKA(1+6/\alpha)}.$$

This solution is clearly problematic. Assume a beam of length L and of typical transversal dimension h. We have $I = \mathcal{O}(h^4)$ and $A = \mathcal{O}(h^2)$ so that $\alpha = \mathcal{O}(L^2/h^2)$. Factor α is typically big which means that shear stiffness is way higher than bending stiffness. At the limit, i.e. when $\alpha \to \infty$, the solution of the problem essentially depends on the shear stiffness and not on the bending stiffness. The computed displacement is way too small: the numerical solution “locks”.

Let us write (5.11) as a function of the 4 degrees of freedom:

$$U^f = \frac{1}{2}\frac{EI}{L}(\Theta_2 - \Theta_1)^2 \simeq \frac{1}{2}EIL\Theta'^2$$

which makes sense.

$$U^s = \frac{1}{2}\frac{GKA}{L}(w_2 - w_1)^2 - \frac{1}{2}GKA(w_2 - w_1)(\Theta_2 + \Theta_1) + \frac{1}{2}\frac{GKAL}{6}((\Theta_2 + \Theta_1)^2 + \Theta_2^2 + \Theta_1^2)$$

or, regrouping the terms

$$U^s = \frac{1}{2}GKAL\left(\frac{w_2 - w_1}{L} - \frac{\Theta_1 + \Theta_2}{2}\right)^2 + \frac{1}{2}\frac{GKAL^3}{12}\left(\frac{\Theta_2 - \Theta_1}{L}\right)^2.$$

Shear energy should be equal to zero when there is no shear, i.e. when $\beta = \Theta - U_2' = 0$. The first term of U^s

$$\frac{1}{2}GKAL\left(\frac{w_2 - w_1}{L} - \frac{\Theta_1 + \Theta_2}{2}\right)^2 \simeq \frac{1}{2}GKAL\beta^2$$

is a finite difference approximation of the shear energy. The second term

$$\frac{1}{2}\frac{GKAL^3}{12}\left(\frac{\Theta_2 - \Theta_1}{L}\right)^2 \simeq \frac{1}{2}\frac{GKAL^3}{12}\Theta'^2$$

is problematic because it adds to the shear energy a term that is a bending energy with a shear stiffness. This means that, for $\beta = 0$, some extra bending energy $\frac{1}{2}\frac{GKAL^3}{12}\Theta'^2$ is added to $\frac{1}{2}EIL\Theta'^2$, this extra bending energy being $\mathcal{O}(L^2/h^2)$ larger than the physical one.

This second part of the shear energy should indeed be equal to zero. It is possible to see that the following approximation of the mass matrix:

$$L^2\frac{GKA}{L}\left[\int_0^1 N_i N_j dt\right] \simeq \frac{GKAL}{4}\left[\begin{array}{cc} 1 & 1 \\ 1 & 1 \end{array}\right].$$

allows to solve the problem i.e. transforms the shear energy in

$$U^s = \frac{1}{2}GKAL\left(\frac{w_2 - w_1}{L} - \frac{\Theta_1 + \Theta_2}{2}\right)^2.$$

This strange value for the mass matrix can be computed by under-integrating the part of the shear energy that depends on the bending angle Θ. The stable Timoshenko beam stiffness matrix can then be written as The stiffness matrix can be re-written as

$$[\mathbf{K}] = \frac{GKA}{6L}\left[\begin{array}{cccc} 6 & -6 & 3L & 3L \\ -6 & 6 & -3L & -3L \\ 3L & -3L & \frac{3}{2}L^2(1+3/\alpha) & \frac{3}{2}L^2(1-6/\alpha) \\ 3L & -3L & \frac{3}{2}L^2(1-6/\alpha) & \frac{3}{2}L^2(1+3/\alpha) \end{array}\right] \tag{5.13}$$

CHAPTER 6

Gometric non linear effects in trusses

6.1 Finite Strain Analysis

Consider truss of Figure 6.1. In the standard theory presented above, this truss is essentially unstable. Yet, we intuitively know that this is not true. This Figure illustrate the “geometrical’ difficulties” that arise in the simple linear theory. If deformations are infinitesimal, analyzing the equilibrium *in the undeformed configuration* provides accurate results. If deformations are *finite*, the analysis should take into account the deformed configuration. Finite deformation analysis should be used as strains are large (> 0.05%). Moreover, finite deformation analysis can be used to analyze the *buckling of structures*.

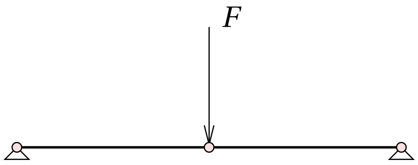

Figure 6.1: Linearly unstable truss, yet non-lineary stable.

Let us now reconsider hypothesis (3.1) by taking into account displacements along X_2. The setup is presented on Figure 6.2. We consider a bar k with nodes i and j. We assume that the displacement of the bar is linear i.e.

$$\begin{aligned}
&U_1(X_1) = U_1^i(1 - X_1/L) + U_1^j X_1/L \\
&U_2(X_1) = U_2^i(1 - X_1/L) + U_2^j X_1/L \\
&U_3 = 0
\end{aligned}$$

The length L_d of the deformed bar is computed in local coordinates X_1, X_2 as

$$\begin{aligned} L_d &= \sqrt{(X_1^2 + U_1^2 - X_1^1 - U_1^1)^2 + (U_2^2 - U_2^1)^2} \\ &= \sqrt{(L + dU_1)^2 + dU_2^2} \end{aligned} \tag{6.1}$$

with $dU_s = U_s^j - U_s^i$, $s = 1, 2$. We then assume that deformations are moderate: $|dU_i|/L < 1$ is not negligible but $|dU_s|^2 \ll |dU_s|$. Taylor expansion of

$$\sqrt{1+\epsilon} = 1 + \frac{\epsilon}{2} + \mathcal{O}(\epsilon^2)$$

allows allows to write

$$L_d \simeq L + dU_1 + \frac{1}{2L}(dU_2)^2$$

and the stretch $\Delta = L_d - L$ in the bar is expressed as

$$\Delta \simeq (U_1^2 - U_1^1) + \frac{1}{2L}(U_2^2 - U_2^1)^2.$$

Note here that $|dU_1|$ and $|dU_2|^2$ are independant so one cannot be compared with the other!

Assume that the absolute value of the tension in a bar k is N (N will be computed later). We consider geometrical non linear effects i.e. force N is not applied along X_1 but along an axis that is rotated of an angle ζ with respect to X_1 (see Figure 6.2). We have

$$\cos\zeta = \frac{L + U_1^j - U_1^i}{L_d} \simeq 1.$$

This is the strongest hypothesis that we make. Then,

$$\sin\zeta = \frac{U_2^j - U_2^i}{L_d} \simeq \frac{U_2^j - U_2^i}{L}.$$

We have then (see Figure 6.2)

$$F_1^j = -F_1^i = N \ \text{ and } \ F_2^j = -F_2^i = N\sin\zeta$$

or in matrix form

$$\begin{pmatrix} F_1^i \\ F_2^i \\ F_1^j \\ F_2^j \end{pmatrix} = \underbrace{\frac{EA}{L}\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{[\mathbf{K}]^k} \begin{pmatrix} U_1^i \\ U_2^i \\ U_1^j \\ U_2^j \end{pmatrix} + \underbrace{\frac{N}{L}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}}_{[\mathbf{K}]_g^k} \begin{pmatrix} U_1^1 \\ U_2^1 \\ U_1^2 \\ U_2^2 \end{pmatrix} \tag{6.2}$$

or in matrix form

$$\{[\mathbf{K}]_e^k + [\mathbf{K}]_g^k\}(\mathbf{u})^k = (\mathbf{F})^k.$$

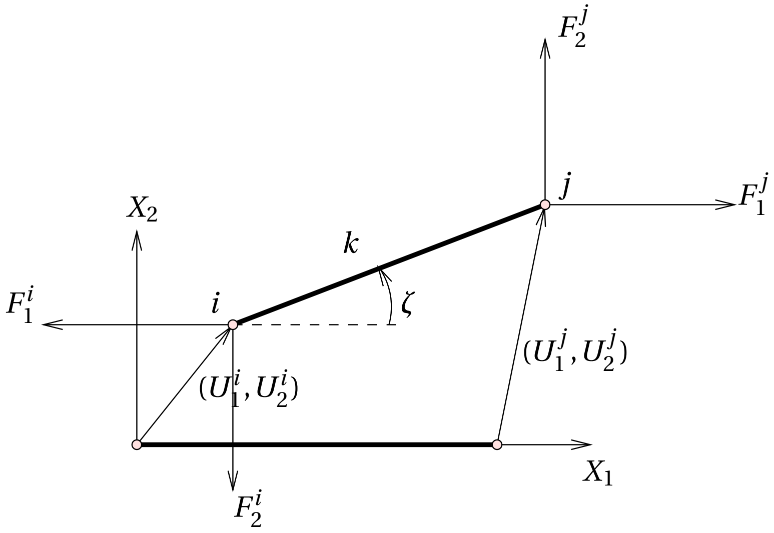

Figure 6.2: Finite Deformations in a bar.

The stiffness matrix of bar k is composed of two terms: the usual elastic stiffness $[\mathbf{K}]_e^k$ that has been used in the linear case and $[\mathbf{K}]_g^k$ that accounts for non linear effects.

The problem is non linear because N depends on the displacements. Let us now compute N. The engineering strain is defined by the elongation of the bar in the current state divided by the length of the bar in the reference state:

$$e = \frac{\Delta}{L} \simeq \frac{U_1^j - U_1^i}{L} + \frac{1}{2L^2}(U_2^j - U_2^i)^2.$$

There exist other definitions of the strain. All these definitions contain the same information: the ratio $\frac{\Delta}{L}$. N is then computed as

$$N = \frac{EA}{L}(U_1^j - U_1^i) + \frac{EA}{2L^2}(U_2^j - U_2^i)^2.$$

It is possible to compute the stiffness matrix in global axes like we did in (6.3):

$$[\mathbf{k}]^k = [\mathbf{k}]_e^k + [\mathbf{k}]_g^k = [\mathbf{T}]_k^T\{[\mathbf{K}]_e^k + [\mathbf{K}]_g^k\}[\mathbf{T}]_k \tag{6.3}$$

with

$$[\mathbf{T}]_k = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & \cos\alpha & \sin\alpha \\ 0 & 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \tag{6.4}$$

Here, we should heve chosen $\zeta + \alpha$ but we make the approximation that rotation matrix is independant of N. This is the second strong approximation of the method.

Here is a summary of the approximations invoked in this finite strain analysis:

- $\cos\zeta \simeq 1$ so that $[\mathbf{K}]_e^k$ is independent of N and $\mathbf{u}$ (error $\approx 0.1\%$)
- $(N/L_d) \approx (N/L)$ so that $[\mathbf{K}]_g^k$ depends on $\mathbf{u}$ only through N (error $\approx 0.001\%$)
- $\zeta \ll 1$ so that $[\mathbf{T}]$ is independent of $\mathbf{u}$ (error $\approx 0.1\%$)
- $N = \frac{EA}{L}(U_1^j - U_1^i) + \frac{EA}{2L^2}(U_2^j - U_2^i)^2$ (error $\approx 0.001\%$)

Only the first approximation is required for the stiffness matrix assembly process. The other approximations merely simplify the calculations.

6.2 Solving the non linear system

The assembly process is similar to the one that has been described in §3. Assuming a truss with N nodes and M kinematic constraints. The following system of $2N+M$ equations allows to find $(\mathbf{u})$ and $(\boldsymbol{\lambda})$ in one linear system solve:

$$\boxed{\begin{bmatrix} [\mathbf{k}]_e + [\mathbf{k}]_g & [\mathbf{c}]^T \\ [\mathbf{c}] & [\mathbf{0}] \end{bmatrix} \begin{pmatrix} (\mathbf{u}) \\ (\boldsymbol{\lambda}) \end{pmatrix} = \begin{pmatrix} (\mathbf{f}) \\ (\bar{\mathbf{u}}) \end{pmatrix}.} \tag{6.5}$$

Matrix $[\mathbf{k}]_g$ depends on $\mathbf{u}$ so that the problem is non linear. A strategy has to be chosen in order to solve it efficiently.

Recall the truncated Taylor series expansion of a nonlinear function $\mathbf{F}(\mathbf{x})$ around $\mathbf{x} = \mathbf{x}^0$:

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}(\mathbf{x}^0) + \left[\frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{x}}\right]_{\mathbf{x}^0} (\mathbf{x} - \mathbf{x}^0) + \text{ h.o.t.}$$

Newton's method proceed in an incremental fashion:

$$\mathbf{x}^{i+1} = \mathbf{x}^i - \left[\frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{x}}\right]_{\mathbf{x}^i}^{-1} \mathbf{F}^i, \quad i = 0, 1, 2\ldots$$

The process stops when convergence occurs i.e. when $\|\mathbf{x}^{i+1} - \mathbf{x}^i\|$ is sufficiently small. Matrix $[\mathbf{J}]^i = \left[\frac{\mathrm{d}\mathbf{F}}{\mathrm{d}\mathbf{x}}\right]_{\mathbf{x}^i}$ is called the jacobian (or tangent) matrix. Even though it is often possible to compute it exactly, it is usually desirable to use simplified schemes.

6.2.1 Picard iteration

In the problem of finite deformation analysis of structures, the equilibrium condition is represented by a system of non-linear algebraic equations of the form

$$[\mathbf{k}]_{(\mathbf{u})}(\mathbf{u}) - (\mathbf{f}) = 0$$

where $[\mathbf{k}]_{(\mathbf{u})}$ depends on$(\mathbf{u})$. The jacobian matrix of the system can be computed as

$$[\mathbf{J}]^i = [\mathbf{k}]_{(\mathbf{u})^i} + \left[\frac{\mathrm{d}[\mathbf{k}]}{\mathrm{d}(\mathbf{u})}\right]_{(\mathbf{u})^i} (\mathbf{u})^i$$

In this first approach, we use the structural stiffness matrix, as if it were the jacobian matrix:

$$[\mathbf{J}]^i \simeq [\mathbf{k}]_{(\mathbf{u})^i}.$$

Let us define the residual

$$(\mathbf{r})^i = [\mathbf{k}]_{(\mathbf{u})^i}(\mathbf{u})^i - (\mathbf{f}).$$

Newton scheme becomes a Picard scheme:

$$\begin{aligned}(\mathbf{u})^{i+1} &= (\mathbf{u})^i - [\mathbf{k}]^{-1}_{(\mathbf{u})^i}(\mathbf{r})^i \\ &= (\mathbf{u})^i - (\mathbf{u})^i + [\mathbf{k}]^{-1}_{(\mathbf{u})^i}(\mathbf{f}) \\ &= [\mathbf{k}]^{-1}_{(\mathbf{u})^i}(\mathbf{f}) \ , \quad i = 0,1,2\ldots\end{aligned}$$

A picard iteration consist in a series of structural solves where the stiffness matrix $[\mathbf{k}]_{(\mathbf{u})^i}$ is simply updated to take into account displacements of the previous iteration. In principle, this approach will converge to the correct solution. However, this approach can be quite slow to converge, and may not converge at all if the problem has an inflection point.

6.2.2 Broyden iteration (quasi Newton)

The Broyden iteration is the generalization of the well known secant method that is used to solve one non linear equation. Le us consider the non linear equation

$$f(x) = 0.$$

Newton's iteration consist in solving

$$x^{i+1} = x^i - \frac{f(x^i)}{f'(x^i)}, \quad i = 0,1,2\ldots$$

Secant method consist in approximating $f'(x^i)$ by finite differences

$$f'(x^i) \simeq \frac{f(x^i) - f(x^{i-1})}{x^i - x^{i-1}}.$$

Secant's scheme is therefore written as

$$x^{i+1} = x^i - f(x^i)\frac{(x^i - x^{i-1})}{f(x^i) - f(x^{i-1})}, \quad i = 0,1,2\ldots$$

Broyden's method is a generalization of the secant method to nonlinear systems:

$$D_k(\mathbf{x}^{k+1} - \mathbf{x}^k) = \mathbf{F}(\mathbf{x}^{k+1}) - \mathbf{F}(\mathbf{x}^k). \tag{6.6}$$

Instead of using the Jacobian matrix, Broyden uses a matrix satisfying the same formula. Note the system provides only N equations to determine the $N \times N$ matrix. The above equation for finding D_k is therefore underdetermined. Broyden's method consist in finding D_k that is solution of (6.6) and that minimizes the Frobenius norm of

$$\|D_k - D_{k-1}\|_F$$

with

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\operatorname{trace}(A^* A)} = \sqrt{\sum_{i=1}^{\min\{m,n\}}\sigma_i^2}$$

where $A*$ denotes the conjugate transpose of A, σ_i are the singular values of A, and the trace function is used. It is possible to prove that

$$D_k = D_{k-1} + \frac{(\mathbf{F}(\mathbf{x}^{k+1}) - \mathbf{F}(\mathbf{x}^k)) - D_{k-1}(\mathbf{x}^{k+1} - \mathbf{x}^k)}{\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2} \otimes (\mathbf{x}^{k+1} - \mathbf{x}^k)$$

where $\mathbf{u}\otimes\mathbf{v}$ is the outer products of the two $N-$ vectors. Note that the jacobian matrix as computed by Broyden's method is not symmetric.

The second term of D_k is of the form $\mathbf{u}\otimes\mathbf{v}$ which is a rank-one correction to D_{k-1}:

$$D_k = D_{k-1} + \mathbf{u}\otimes\mathbf{v}$$

Matrix $\mathbf{u}\otimes\mathbf{v}$ is of rank 1 and the modification of B is called a rank-one update. It is then possible to obtain D_k^{-1} knowing D_{k-1}^{-1} (it is called the Sherman-Morrison Formula):

$$D_k^{-1} = D_{k-1}^{-1} - \frac{D_{k-1}^{-1}(\mathbf{u}\otimes\mathbf{v})D_{k-1}^{-1}}{1 - \mathbf{v}D_{k-1}^{-1}\mathbf{u}}.$$