

Chapter 2

Physical Simulation

Without the understanding of the laws behind the physical behaviour of objects by means of interaction, a realistic simulation within a virtual environment cannot be achieved. The fundamental principles of the subject rely on the theories of classical and continuum mechanics. Therefore, in the following a review the physics governing the motion and deformation of bodies is given. The notations of the physical properties are based on [5, 13] and [14] which also give a good introduction, but have broader perspective in the concepts of classical mechanics.

Since the book at hand is not meant to give a complete introduction, it is advised to complement the knowledge by additional literature for a full understanding. The book has to focus on the principles required for the simulation of deformable objects respectively textiles.

For detailed introduction to the concepts of linear respectively for nonlinear elasticity which will be a part of this chapter, the books of Gould [6] and Bonet [3] besides the ones mentioned above can be recommended.

2.1 Elementary Units and Principles

The motion of rigid bodies was one of the first physical phenomena that was scientifically investigated and marks to some extent the beginning of physics. Before this point, all observations and experiments were more dedicated to answer questions on a philosophical level. It started with Galileo Galilei in the late 16th century, who mainly studied the motion of objects in free fall. He experimentally proved that objects of different mass will fall within the same time as long as the resistance of the surrounding medium is negligible. Moreover, he found that a body on a level surface will continue its movement in the same direction at constant speed unless it is disturbed. The latter finding, the principle of inertia, was a great progress in the understanding of motion. Galileo also derived a relationship between the square of the elapsed time to the distance. Although Galileo made many contributions to science, the actual founder of classical mechanics which is the basis of the work is Sir Isaac Newton. He established by his findings and by those of his predecessors

a mathematical basis and presented the groundwork for classical mechanics by his publication of the *Philosophiae Naturalis Principia Mathematica*.

2.1.1 Conventions

In the following the motion of a body in time will be deduced analytically. First of all, to describe the movement the **position** of the body is needed, denoted by \mathbf{r} being a vectorial parameter (indicated by boldface letters) and expressed in a general coordinate system with its time parameter t . For now, it is assumed that the body has a negligible size. By observing the positional change of the body in an infinitesimal time step one arrives at the differential of $\mathbf{r}(t)$ given by

$$\dot{\mathbf{r}} = \frac{d\mathbf{r}}{dt}$$

which leads to the **current velocity** measured in $[\frac{m}{s}]$.

Further, it can be seen that the introduction of the body mass as a quantitative measure of the inertia is an immediate result of Galileo's principle. For example: given an object with a certain velocity, a change in this physical value can only be achieved if a force is applied. This force is required to overcome the inertia and thus leads to the following relationship between force in [N] and mass in [kg] (inertia):

$$\mathbf{F} = \frac{d}{dt}(m\dot{\mathbf{r}}) = \frac{d\mathbf{p}}{dt} \equiv \dot{\mathbf{p}} \quad (2.1)$$

In words, a change of the (vectorial) velocity in time requires a force acting on the body. The latter equation constitutes Newton's second law of dynamics. At the same time, another quantity is introduced, the (linear) **momentum** \mathbf{p} . It is mathematically expressed by the product of mass and velocity. The common notation $\dot{\mathbf{p}}$ denotes the differential change in time of \mathbf{p} which is equivalent to the **impulse** $[\frac{kg \cdot m}{s}]$ being the integral of a force w.r.t. time. The momentum can be seen as mass in motion, e.g. to stop a body with a high momentum, a high force will be needed.

In classical mechanics, constant mass over time is normally assumed, simplifying the previous formula to

$$\mathbf{F} = m \frac{d\dot{\mathbf{r}}}{dt} = m\ddot{\mathbf{r}} \quad (2.2)$$

This results in the rate of change in velocity over time, the **acceleration** $[\frac{m}{s^2}]$. The basic relationship between force and acceleration given by the latter equation is considered to be the fundamental principle of Newtonian dynamics. The principle will later on be used as a basis for the physical simulations. A second-order differential equation $\mathbf{F} = \mathbf{m} \cdot \frac{d^2}{dt^2}\mathbf{r}$ is created which has to be solved numerically for the acceleration. Equation (2.1) additionally provides an important concept namely the **principle of conservation of linear momentum**. The law says that at vanishing net force, i.e. $\mathbf{F} = 0$, the momentum is conserved.

With the linear momentum, a quantity is given describing the motion of an object. Furthermore, an object can also feature a rotational motion. This rotation is described by the **angular momentum** \mathbf{L} relative to an origin \mathbf{O} . It is defined by

$$\mathbf{L} = (\mathbf{r} - \mathbf{O}) \times \mathbf{p} \quad (2.3)$$

The rate of change in time of the angular momentum leads us to

$$\tau = \frac{d\mathbf{L}}{dt} = \underbrace{\frac{d(\mathbf{r} - \mathbf{O})}{dt} \times \mathbf{p}}_{\dot{\mathbf{r}} \times m\dot{\mathbf{r}}=0} + (\mathbf{r} - \mathbf{O}) \times \frac{d\mathbf{p}}{dt} = (\mathbf{r} - \mathbf{O}) \times \mathbf{F} \quad (2.4)$$

which is called **torque**. Similarly to the linear momentum, (2.4) yields the principle of conservation of the angular momentum, which states that the angular momentum is conserved if the torque $\tau = 0$.

2.1.2 Energy and Conservation

With the brief introduction of important vectorial units and their relationships, the current mechanical state of a system can be given in terms of forces and velocities. However, these quantities are only the result of a more basic quantity, namely the energy being inherent in a mechanical system. Energy is an abstract term for a scalar quantity to be associated with the system. Nevertheless energy can be categorised in different forms which will be illustrated in the following. In the field of mechanical and structural engineering, the **potential energy** is the most pronounced nature of energy. Typically in balanced mechanical systems, the biggest quantity of the energy is generally conserved in the potential form, e.g. due to its own weight, as there is no movement. The weight corresponds with the gravitational quantity of the potential energy V . It is defined by the mass m of the body and its height h relative to the ground, respectively to the origin along the line of action of the gravity within the inertial system. Therefore it can be described by

$$V = m \cdot g \cdot h \quad (2.5)$$

It should be noted that the formula is just an approximation, as the introduction of the gravitational acceleration g is only sufficiently accurate at ground level. The previously shown relationship is also a special case of potential energy, i.e. in a more general form (independent of its physical nature) it can be written as a scalar field

$$V : \begin{cases} \mathbb{R}^3 \rightarrow \mathbb{R} \\ \mathbf{r} \mapsto V(\mathbf{r}) \end{cases} \quad (2.6)$$

which maps the object coordinates to an energy level. A fundamental aspect is the **preservation of energy** governing all known natural phenomena. It states that *the*

total energy of a physical (closed) system is always conserved and thus constant through all state transitions of the system.

The term *closed* in this context has the meaning of complete isolation from outside such that energy can neither drain nor flow in. A direct result of this basic principle is the conservation of the momentums already introduced above in (2.1) and (2.4). The conservation of energy is not only a physical requirement but also a tool allowing to define the solution of a physical problem mathematically which will be made use of later.

Another term in this topic is the **work** produced inside a system. It is in contrast to the “common” energy-term used to define the quantities needed or produced in mechanical processes where a force is acting. Since in a system being in an equilibrium such a process cannot start without external influence, a form of mechanical energy has to be provided over a time or a path which is then normally expressed in the terms of work. Work is therefore achieved by applying an external force \mathbf{F} during a displacement or along a path S . Therefore the formal definition of work measured in (N m) W_{ab} along the path is

$$W = \int_S \mathbf{F} \cdot d\mathbf{s} \quad (2.7)$$

The work in only one dimension is observed first, generated in the movement from point a to b . If one looks at an object and inserts the resulting force (2.2) into the above equation, the following formula for the work can be obtained.

$$W_{ab} = \int_a^b m\ddot{\mathbf{r}} \cdot d\mathbf{s} = m \int_a^b \frac{d\dot{\mathbf{r}}}{dt} \cdot d\mathbf{s} = m \int_a^b \frac{d\dot{\mathbf{r}}}{dt} \cdot \dot{\mathbf{r}} dt = \int_a^b \dot{\mathbf{r}} d\dot{\mathbf{r}} \quad (2.8)$$

$$= \frac{m}{2} \int_a^b \frac{d}{dt} (\dot{\mathbf{r}}^2) dt = \frac{m}{2} (\dot{\mathbf{r}}_b^2 - \dot{\mathbf{r}}_a^2) \quad (2.9)$$

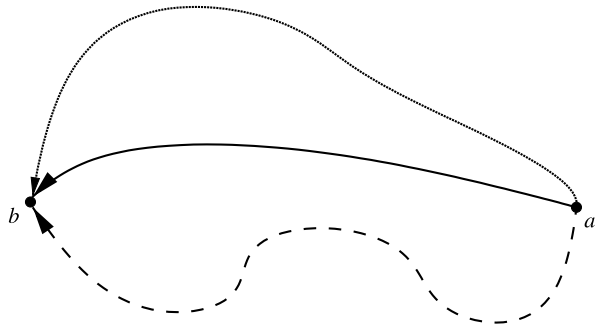
From this result it can be seen that the work solely depends on the scalar quantity $m\dot{\mathbf{r}}^2$ which is computed at the points a and b . This specific quantity being the product of mass and its squared velocity is called the **kinetic energy** of a mass point. It is different from the potential energy by being preserved in the motion of the body. It is therefore also known as the motion energy due to the velocity. With the previous observation expressed in formula (2.9) it is said that the work done can be defined as the difference of kinetic energy at the two points. It simply be written as

$$W_{ab} = T_b - T_a \quad (2.10)$$

Although the path is not explicitly given in the formula, the work done still depends on it. Thus another consideration has to be taken into account: if all arbitrary paths as illustrated in Fig. 2.1 taken from a to b lead to the same amount of work done, the force and the system are said to be **conservative**.

One can arrive at an equivalent conclusion by describing a conservative force by using a closed path. If the work done on an arbitrary path from a to b and back to a vanishes, then the force satisfies the conservation condition. This will always

Fig. 2.1 Examples of physical paths connecting two points



be true for a homogeneous gravitational field. But adding a dissipative component like friction to the same field would break this condition, as the force gets a positive component that does not vanish. Formally, with aforementioned property

$$\oint_a^b \mathbf{F} ds = 0 \quad (2.11)$$

holds. If one uses the conclusions of the “fundamental theorem of calculus” and the “gradient theorem” force can be expressed by a scalar field. An essential and sufficient condition for the existence of such a scalar field is the independence of the path. Then, the scalar function w.r.t. position fulfils the condition of the potential field of (2.6), and one obtains the following relation

$$\mathbf{F} = -\nabla V(\mathbf{r}) \quad (2.12)$$

One should note that the negative sign is physically motivated by V being an energy. To be more precise: In vector calculus the gradient is defined as the steepest increase of a function and to obtain the physical property of energy conservation, the force must always be opposing the direction of increase of energy. Thus, it follows for the work

$$W_{ab} = \int_a^b -\frac{\partial V}{\partial s} ds = V_a - V_b \quad (2.13)$$

As one can see from the equation above, the work is independent of the choice of field origin, as it does not change the value. Combining the results of (2.10) and (2.13), one gets

$$T_b - T_a = V_a - V_b$$

or equivalently

$$T_b + V_b = V_a + T_a \quad (2.14)$$

The result can be seen as a mathematical proof for the energy conservation since the net energy, given by $E = T + V$, is retained in a conservative potential field.

As a consequence discovered by Euler and Lagrange, given a particle with known velocity and position at start and further the end position, then the actual path is

uniquely determined by the energy E and its conservation. This allows to find a path that minimises the time-integral of the energy E along the path. The particular path which minimises the integral called *action* is then the path chosen by nature. This observation is known as *principle of least action*.

But for some problems of mechanics, the work function (2.13) is not only dependent on position but also on time. Therefore the scalar field is not conservative anymore and the law of energy conservation does not hold. Hamilton extended the procedure of Euler and Lagrange such that it is applicable for the latter case. The procedure uses the same start and end positions but determines further the time from start to end of the motion. Hamilton's formulation of least action is given by

$$\delta \int_{t_1}^{t_2} (T - V) dt = 0 \quad (2.15)$$

which assert that the action assumes its minimum for motion taken by the particle. The principle of least action paved the way for a variational approach to the motion of an object which will be further discussed in the continuum mechanics in Sect. 2.2.

2.1.3 Multibody Systems and Constrained Motions

Having formulated a framework for deducing the laws governing the motion of a single body and introduced the important physical quantities, it is now time to generalise the mechanics to multibody systems (MBS). As the name implies, these systems are consisting of many bodies or mass points, that are acted upon by a common force. In the following, it is first looked at simple point mass systems to find the fundamental properties before moving on to the generalised MBS.

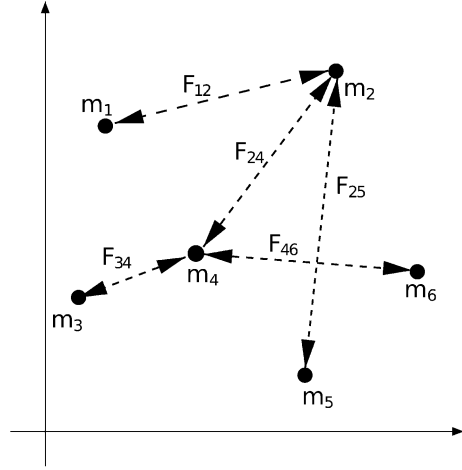
To apply the principles introduced in the previous sections to point masses, it is necessary to distinguish between **external** and **internal** forces. External forces are acting on each mass point individually, whereas the internal forces are created by the relation between different point masses like it is shown in Fig. 2.2. This leads to a motion equation for a mass point different from the one m_i as in (2.1)

$$\underbrace{\sum_j \mathbf{F}_{ji}}_{\text{internal}} + \underbrace{\mathbf{F}_i^e}_{\text{external force}} = \mathbf{\ddot{p}}_i \quad (2.16)$$

Apparently, in case of $i = j$ the inner force has to be $\mathbf{F}_{jj} = 0$ meaning that a particle mass exerts no force on itself. By summing up all moments one gets the total moment in which the Newton's second law (2.2) can be inserted to obtain the equation for all mass points.

$$\frac{d^2}{dt^2} \sum_i m_i \mathbf{r}_i = \sum_i F_i^{ext} + \sum_i \sum_{i \neq j} \mathbf{F}_{ij} \quad (2.17)$$

Fig. 2.2 An example of force-relationships between point-masses



Presuming further the validity of the third law (*actio et reactio*) for the relational force between the particles, then $\mathbf{F}_{ji} = \mathbf{F}_{ij}$ holds and all inner forces vanish. In physics, this assumption is also known as *the principle of weak interaction* and is valid for many materials, especially those which are treated in this work. Similarly, one can reduce the left hand side of the term by condensing the masses to the centre of gravity \mathbf{R} . This is an equivalent definition obtained by creating the weighted average.

$$\mathbf{R} = \frac{\sum_i m_i \mathbf{r}_i}{\sum_i m_i} = \frac{\sum_i m_i \mathbf{r}_i}{M} \quad (2.18)$$

Consequently, a simplified motion equation is obtained for the centre of masses.

$$M \frac{d^2}{dt^2} \mathbf{R} = \sum_i \mathbf{F}_i^e \quad (2.19)$$

Finally, external forces are solely affecting the centre of masses of a MBS for which the weak interaction holds. Analogously, one can extend this statement to moments which yields

$$\mathbf{P} \equiv \sum_i \mathbf{p}_i = M \frac{d\mathbf{R}}{dt} \quad (2.20)$$

From the latter equation it follows that the total moment is preserved if no external forces are acting on the MBS.

Coming now to the work done within a MBS: by naturally treating the total work as the sum of the work done by all mass points. The work of (2.7) is adopted and extended by the force relation. This leads to an expression for the total work

$$W_{ab} = \sum_{i \neq j} \int_a^b \mathbf{F}_{ji} \cdot d\mathbf{s}_i + \sum_i \int_a^b \mathbf{F}_i^{ext} \cdot d\mathbf{s}_i \quad (2.21)$$

2.1.4 Constraints

After deducing the inner forces and their influence on the total motion of a MBS, the computation of the motion of a system is possible by the formulas introduced beforehand. Nevertheless, the class of systems which can be simulated is quite small since we have no ability to restrict the motion to an allowable space. As an example, consider the case of a pearl necklace. The pearls not only have to follow the law of inertia, their motion is also constrained by the chain that might be fastened to some points. Such constraints have to be regarded in the computations. In this example, two problems become visible which have to be considered appropriately. First, it might happen that the coordinates of some or all n particles are not independent anymore, e.g. the distance between particles representing a rigid body have to remain constant. Formally, particles i and j have to obey the condition

$$\mathbf{r}_i^2 - \mathbf{r}_j^2 = l_{ij}^2 \quad (2.22)$$

Thus the coordinates are related to each other by the constraint and the motion of each particle in space cannot be solved individually. With the introduction of so-called **generalised coordinates** q_i one can circumvent the problem. The generalised coordinates are used to derive the $3n$ particle coordinates by employing functions of q_i

$$\mathbf{r}_i = \mathbf{r}_i(q_1, q_2, \dots, q_{3N-k}, t) \quad \forall i = 1, 2, \dots, N \quad (2.23)$$

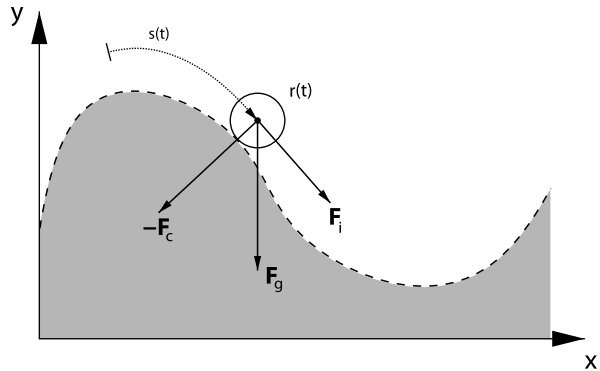
where the generalised coordinates are not necessarily coinciding with spatial dimensions. Depending on the problem and its constraints, a transformation, e.g. expressing particle positions in polar coordinates, could be beneficial in solving the motion of the entire system. In general, there are different types of constraints that can be categorised. The first and easiest to handle is the **holonomic** (*Greek: holistic, integrable*) constraint which can be expressed by functions of the form

$$f(\mathbf{r}_i, t) = 0$$

One example of this type has been given by (2.22) for a rigid body. Its solution can be found by the general coordinates. Constraints that cannot be expressed by such functions and are therefore not directly dependent on positions are consequently called **nonholonomic**. These typically require to solve a differential equation in the first place. An example for such a constraint would be a rolling disk on a plane, where the movement of centre of the disk depends on the velocity at the stationary contact point (Fig. 2.3). Further categorisation is being made by time-dependency, namely **rheonomous** (*Greek, in flow*) for variable in time and **scleronomous** for time-independent constraints.

Besides the dependency of coordinates, another problem becomes visible when constraints are given for a motion. The forces induced by the constraints are not *a priori* known. Therefore, it cannot simply the equation of motion be solved as these forces are only reactions upon the movement. But it is known that the constraint forces are holding the system within the allowable range, respectively within

Fig. 2.3 Rolling disc on a curved path with gravity



the **configuration space** of our system, being the set of all admissible coordinates \mathbf{r}_i aggregated to tuples in \mathbb{R}^{3N} .

It is further a (compatible) **virtual displacement** defined as an infinitesimal change in the configuration namely in the positions (or generalised coordinates) of the system that is consistent with the constraint equations at a fixed time t , i.e. displacements are perpendicular to the constraints. The displacement $\delta \mathbf{r}$ is called *virtual* to distinguish it from a real displacement caused by the change of (*dynamical*) forces and constraints within a time step dt .

If it is supposed having a system which resides within a state of equilibrium such that the net force $\mathbf{F}_i = 0$ vanishes on all particles i , and when further the virtual displacement is added to the system, then the positional change $\delta \mathbf{r}$ does no work as the system is in an equilibrium. Therefore, one can write for the work

$$\sum_i \mathbf{F}_i \cdot \delta \mathbf{r} = 0 \quad (2.24)$$

Equation (2.24) is known as the **principle of virtual work** which implies that virtual displacements $\delta \mathbf{r}$ on systems in equilibrium do no virtual work.

The force \mathbf{F}_i can also be written as the sum of dynamical (external) and constraint force:

$$\mathbf{F}_i = \mathbf{F}_i^{(d)} + \mathbf{F}_i^{(c)} \quad (2.25)$$

Together with a virtual displacement, one gets for the entire system

$$\sum_i \mathbf{F}_i^{(d)} \cdot \delta \mathbf{r}_i + \sum_i \mathbf{F}_i^{(c)} \cdot \delta \mathbf{r}_i = 0 \quad (2.26)$$

As consequence of the virtual work principle it can be seen that the right term has to be zero as the allowed virtual displacements are orthogonal to the constraint forces. Therefore the separation of the net force in the dynamic and constraint component can be omitted. Moreover, since the system is in equilibrium state, it is also known as a consequence of the previous observations on Newton's 2nd law of dynamics that these forces are then equal to the counteracting forces created by $\dot{\mathbf{p}}_i$. Resulting

in

$$\sum_i (\dot{\mathbf{p}}_i - \mathbf{F}_i) \cdot \delta \mathbf{r}_i = 0. \quad (2.27)$$

The vanishing sum of the differences between acting forces and time-derivative momenta is often referred to as the *principle of D'Alembert*. It allows to compute the motion without considering the constraint forces as they are eliminated. Although the solution to both problems of constraints has been found, a valid virtual displacements with respect to the virtual work is still needed. Therefore, a transformation in generalised coordinates is required. According to the given mapping in (2.23), the following transformation rule for the variation of the positions

$$\delta \mathbf{r}_i = \sum_j \frac{\partial \mathbf{r}_i}{\partial q_j} \cdot \delta q_j \quad (2.28)$$

is obtained. Inserting into (2.27) leads to

$$\sum_i \sum_j (\dot{\mathbf{p}}_i - \mathbf{F}_i) \cdot \frac{\partial \mathbf{r}_i}{\partial q_j} \cdot \delta q_j = 0 \quad (2.29)$$

The terms of $\dot{\mathbf{p}}_i$ and \mathbf{F}_i are split up to have a closer look at the individual components. The latter term can be used to define a **generalised force** being

$$Q_j = \sum_i \mathbf{F}_i \cdot \frac{\partial \mathbf{r}_i}{\partial q_j} \quad (2.30)$$

Note that a generalised force is a dimensionless quantity.

2.2 Continuum Formulation

In the previous chapters the essential physical equations that govern the motion of bodies in classical mechanics have been defined. It was always assumed that the bodies are rigid and treated as idealistic point masses. But these equations cannot describe the behaviour of bodies with spatial dimension being furthermore deformable. This is the point where the continuum formulation comes in.

First of all, one has to define what continuum actually means. Looking at an object in at the atomic level, the object typically appears as a conglomerate of atoms enforced to stay in a lattice structure with gaps in it. But from a macroscopic point of view, there are no gaps visible and object's material is perceived to be continuous. When it is discussed about large scale behaviour of an object, the material is treated as a continuum and associate with each material point a spatial coordinate related to a physical quantity. As consequence, there are no gaps between those material points and one can compute the derivatives of these quantities as the limit of the medium at the specified point.

For example, obtaining the density of an object at a point \mathbf{x} , the volume V can be shrunk to an infinitesimal small region and, assuming the existence of the limit of mass occupying the volume element and the volume itself, leading to

$$\lim_{\Delta V \rightarrow 0} \frac{\Delta m}{\Delta V} = \rho(\mathbf{x})$$

As result, the physical laws that were introduced in the case of point masses to the continuum can be applied. Nevertheless, the laws are not completely describing the behaviour of the continuum, since the equations are only accounting for single points. Therefore geometric deformations and material response due to the change in the continuum are not considered by these formulas. For completeness, it is needed to address these properties by adding equations defining the geometric deformation with respect to the undeformed state.

2.2.1 Internal Strains

Given the reference configuration for the body by Ω being a manifold in \mathbb{R}^3 with boundary and define a twice continuously differentiable mapping

$$\mathbf{x}: \begin{cases} \Omega \times \mathbb{R} \rightarrow \mathbb{R}^3 \\ \mathbf{X}, t \mapsto \mathbf{x}(\mathbf{X}, t) \end{cases} \quad (2.31)$$

onto the deformed state \mathbf{x} of the body. Under the assumption of having no gaps and no overlaps (i.e. no self-intersections), it can be concluded that \mathbf{x} is continuous and the mapping has to be bijective as the deformation cannot degenerate. The locations denoted \mathbf{X} are often referred to as **material coordinates**.

With such a mapping the degree of deformation the body has undergone can be estimated. The deformation quantities, the internal strains, create corresponding internal stresses opposing the deformation. The internal strain at point \mathbf{x} with the infinitesimal change of length ds is identified by the mapping (deformation). Observing the squared length, one gets

$$(ds)^2 = (d\mathbf{x}(\mathbf{X}, t))^2 = \sum_{i=1}^3 \sum_{j=1}^3 \left\langle \frac{\partial}{\partial X_i} \mathbf{x}(\mathbf{X}, t), \frac{\partial}{\partial X_j} \mathbf{x}(\mathbf{X}, t) \right\rangle dX_i \cdot dX_j \quad (2.32)$$

In order to reflect the strain with respect to the reference configuration, the reference length denoted by dS has to be consider, thus it follows

$$(ds)^2 - (dS)^2 = \sum_{i=1}^3 \sum_{j=1}^3 \left(\left\langle \frac{\partial}{\partial X_i} \mathbf{x}(\mathbf{X}, t), \frac{\partial}{\partial X_j} \mathbf{x}(\mathbf{X}, t) \right\rangle - \delta_{ij} \right) dX_i \cdot dX_j \quad (2.33)$$

whereas δ_{ij} is the *kronecker-delta*, being one **iff** $i = j$ and zero otherwise. Substituting the term on the right hand side with

$$(ds)^2 - (dS)^2 \equiv 2\varepsilon_{ij}dX_i \cdot dX_j \quad (2.34)$$

To omit the summation over the repeated indices, the *Einstein summation convention* is introduced. The ε_{ij} are the components of **Green's strain tensor** ε^G at a point \mathbf{x} given by

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial x_k}{\partial X_i} \frac{\partial x_k}{\partial X_j} - \delta_{ij} \right) \quad (2.35)$$

Remark: The strain tensor is strongly related to the metric tensor. For each vector pair of the embedding space the metric tensor creates a positive definite, symmetric bi-linear form with which one can measure lengths or angles in Riemannian manifolds. Here, we measure length changes under our mapping $\mathbf{x}(\mathbf{X}, t)$ which can be interpreted as a local deformation of an manifold in the Euclidean space \mathbb{R}^3 . Moreover, the strain tensor is by definition symmetric.

Another notation of the strains is typically created by expressing the deformation in displacements u_i with respect to the reference configuration. Consequently, we can write for the position

$$x_i = u_i + X_i, \quad \frac{\partial x_i}{\partial X_j} = \frac{\partial u_i}{\partial X_j} + \delta_{ij} \quad (2.36)$$

Insertion into (2.35) gives

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j} \right) \quad (2.37)$$

Components ε_{ii} on the diagonal are an estimate for the elongation in the coordinate directions, whereas the others are the shear components. A diagonalisation of the strain-tensor is possible by computing the eigenvectors with its associated eigenvalues. The eigenvectors determine the directions of the principal strains describing the volume or surface change. Therefore, the mean volume change is estimated by $\frac{1}{3}\text{tr}(\varepsilon^G)$.

The latter tensor can be further linearised by omitting the nonlinear terms. As a result we obtain **Cauchy's-strain tensor** \mathbf{C} with its components

$$C_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} \right) \quad (2.38)$$

The benefit of Cauchy's tensor is that it can be computed more efficiently due to its linearity. But at the same time the simplification induces errors for large strains so it should only be used for small strains. Moreover, it does not account for rotations of the frame of reference and is thus variant under rotations. Although the tensor exhibits some drawbacks, its simplicity in computation mostly outweighs the aforementioned problems. With a special treatment of the situation the drawbacks

are kept minimal. As the tensor depends on the orientation of the local frame of reference, the local deformation is separated into two transformations. The first is determined by the rigid rotation of the frame of reference and the other by the deformations given by the Cauchy tensor. This approach of separating the rotations from the strain determination is referred to as **co-rotational formulation** and is often favoured for real-time finite element simulation systems [7, 12].

2.2.2 Mechanical Stresses

With the strain tensor a measure for deformation at an arbitrary point within a continuous medium is obtained. As the strains are a result of acting forces, the forces which act on a continuous body has to be described. These fall into the categories of internal and external forces. The latter is further divided into surface forces and body (or volume) forces. Body forces act on the distribution of mass inside the body, e.g. gravity, while surface forces act on the boundary, e.g. contact forces. The former category of internal force is created by the resistance of the material against deformation. Hereby, the force is termed **stress** or **traction** acting on the surface element of an infinitesimal volume dV . Therefore the stress vector \mathbf{t} depends on the surface orientation and yields a pressure (force density) as unit. It is defined by

$$\mathbf{t}(\mathbf{n}) = \lim_{\Delta S \rightarrow 0} \frac{\Delta \mathbf{F}(\mathbf{n})}{\Delta S} \quad (2.39)$$

where ΔS denotes the surface element, \mathbf{n} its orientation and $\Delta \mathbf{F}$ the force acting on it. The vector $\mathbf{t} \cdot \mathbf{n}$ is the stress in direction of the surface normal \mathbf{n} and is consequently called **normal stress**. The stress component perpendicular to the normal is called **shear stress**. Since the third law of Newton should also hold for the stresses, it follows that $\mathbf{t}(-\mathbf{n}) = -\mathbf{t}(\mathbf{n})$. In general, the stress vector is not restricted to normal direction. By analysing the dependency between normal and stress vector, one can create a mapping from any surface vector to the resulting stress.

Therefore an infinitesimal tetrahedron as shown in Fig. 2.4 is considered and the stress vectors $\mathbf{t}(\mathbf{e}_i)$ according to the surface elements ΔS_i oriented with respect to the Cartesian basis \mathbf{e}_i are denoted. The motion of tetrahedron has to obey the second law of Newton such that all stresses acting on the surfaces are related to the acceleration \mathbf{a} as follows

$$\mathbf{t}(\mathbf{n})\Delta S + \mathbf{t}_1\Delta S_1 + \mathbf{t}_2\Delta S_2 + \mathbf{t}_3\Delta S_3 + \mathbf{F}^{(e)}\rho\Delta V = \mathbf{a}\rho\Delta V \quad (2.40)$$

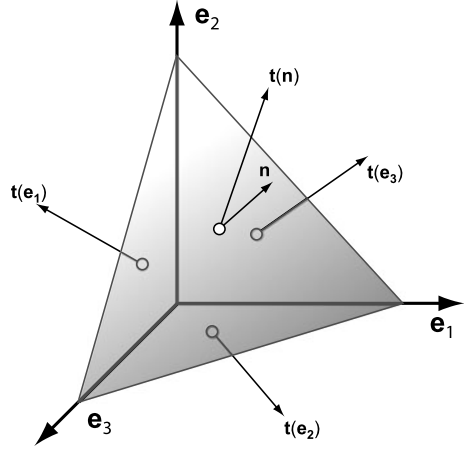
From a special case of the Gauss divergence theorem, it follows that the total vector area of the closed surface is

$$\Delta S\mathbf{n} - \Delta S_1\mathbf{e}_1 - \Delta S_2\mathbf{e}_2 - \Delta S_3\mathbf{e}_3 = 0 \quad (2.41)$$

Thus it can be written for the surface elements

$$\Delta S_i = (\mathbf{n} \cdot \mathbf{e}_i)\Delta S \quad \text{with } i \in \{1, 2, 3\} \quad (2.42)$$

Fig. 2.4 Tetrahedron with surface stresses in Cartesian coordinates



The volume ΔV of the element is computed by the formula

$$\Delta V = \frac{\Delta h}{3} \Delta S \quad (2.43)$$

where Δh denotes the distance between the origin and the slant face. Substituting (2.41) and (2.42) in the initial formula (2.40) of the acting forces and dividing by ΔS leads to

$$\mathbf{t} = \sum_{i=1}^3 (\mathbf{n} \cdot \mathbf{e}_i) \mathbf{t}_i + \rho \frac{\Delta h}{3} (\mathbf{a} - \mathbf{f}) \quad (2.44)$$

Shrinking the tetrahedron by $\Delta h \rightarrow 0$ yields the limit

$$\mathbf{t} = \sum_{i=1}^3 (\mathbf{n} \cdot \mathbf{e}_i) \mathbf{t}_i \quad (2.45)$$

The result can be rewritten as a tensorial relationship whereas the stress-vectors \mathbf{t}_i are identified with the columns of the stress tensor. Consequently, an equivalent tensor notation of the latter equation is obtained if \mathbf{n} is extracted

$$\mathbf{t} = \mathbf{n} \cdot \boldsymbol{\sigma} \quad (2.46)$$

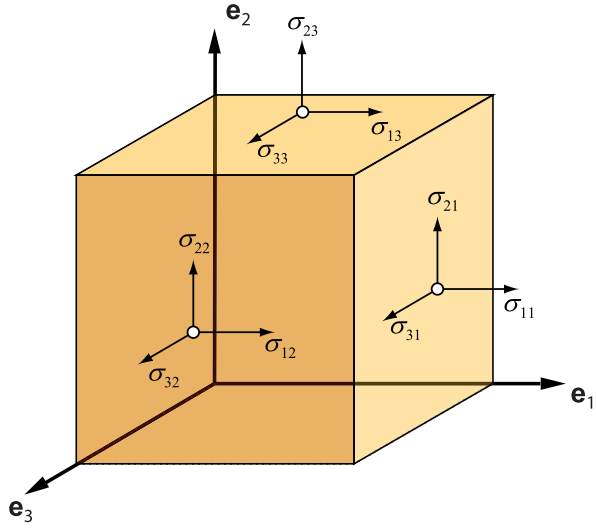
$$\boldsymbol{\sigma} \equiv \mathbf{e}_i \cdot \mathbf{t}_i \quad (2.47)$$

each component of the tensor is given by

$$\sigma_{ij} = \mathbf{t}_i \cdot \mathbf{e}_j \quad (2.48)$$

This notation is possible as the stress tensor defines a property being independent of \mathbf{n} . Another expression of the stress tensor is given by the separation of the or-

Fig. 2.5 Correspondence of stress components to surface elements



thogonal components of \mathbf{t}_i . We can write

$$\mathbf{t}_i = \sigma_{i1}\mathbf{e}_1 + \sigma_{i2}\mathbf{e}_2 + \sigma_{i3}\mathbf{e}_3 \quad (2.49)$$

$$= \sigma_{ij}\mathbf{e}_j \quad (2.50)$$

from which it follows that

$$\sigma = \mathbf{e}_j \mathbf{t}_i = \sigma_{ij} \mathbf{e}_i \mathbf{e}_j \quad (2.51)$$

The entries σ_{ij} represents the force per unit area on a surface element perpendicular to the i th coordinate and the j th coordinate direction (see Fig. 2.5). Equation (2.46) is known as **Cauchy-stress formula** and the tensor σ is termed **Cauchy-stress tensor**. By using the principle of conservation of angular momentum, it can be proven that the tensor needs to be symmetric (cf. [13]).

Similar to the strain tensor, the stress tensor can be diagonalised and the estimation of the eigenvectors yields the principal stresses where the value of diagonal elements $\sigma_{ii} > 0$ indicate tension and $\sigma_{ii} < 0$ indicate pressure. Analogously to the estimation of mean volume change in the strain vector, the mean pressure is given by $\frac{1}{3}\text{tr}(\sigma)$.

The stress tensor σ now allows to describe the internal forces in a body. With \mathbf{f} being the body force per unit mass, it follows for the total body force

$$\int_{\Omega} \rho \mathbf{f} dV \quad (2.52)$$

Moreover, with \mathbf{t} as surface force per unit area, the traction field of the body is given by

$$\int_{\partial\Omega} \mathbf{t} dS \quad (2.53)$$

Cauchy's formula (2.46) yields the relationship to the stress tensor

$$\int_{\partial\Omega} \mathbf{n} \cdot \boldsymbol{\sigma} dS \quad (2.54)$$

By making use of the divergence theorem leads to

$$\int_{\partial\Omega} \mathbf{n} \cdot \boldsymbol{\sigma} dS = \int_{\Omega} \nabla \cdot \boldsymbol{\sigma} dV \quad (2.55)$$

The latter equivalent notation allows to formulate the *principle of conservation of linear momentum* for a continuum

$$\nabla \boldsymbol{\sigma} + \rho \mathbf{f} = \rho \frac{\partial^2 \mathbf{u}}{\partial t^2} \quad (2.56)$$

In order to achieve a static equilibrium of the forces, the displacements have to reach a constant value which lets the derivatives of \mathbf{u} vanish. In Cartesian coordinates, it is obtained

$$\frac{\partial \sigma_{ji}}{\partial x_j} + \rho f_i = 0 \quad (2.57)$$

2.2.3 Constitutive Equations

The degree of deformation is reflected in the strain tensor and the actual forces acting on a volume element is determined. Next, it is needed to relate the deformations to the forces and thus to define a function describing the response of the material in terms of stresses to the corresponding strains. The so-called **response function** \mathcal{C} is a general description of the desired correspondence

$$\boldsymbol{\sigma} = \mathcal{C}(\boldsymbol{\varepsilon}) \quad (2.58)$$

The above equation is termed **constitutive equation** and may consider various influences, e.g. thermal conductivity, affecting the response and thus the stresses. If the response of the material is the same at every material coordinate, the material is **homogeneous** otherwise **heterogeneous**. Moreover, if the stress in the material depends on the strain direction, the material is said to be **anisotropic**. If the behaviour differs only in orthogonal strain directions, the material is **orthotropic** (or exhibits orthogonal anisotropy). A material being indifferent to the direction of strain is called **isotropic**.

The theory of linear elasticity considers only ideally elastic materials. A deformed body, under isothermal conditions, is ideally elastic if it recovers its shape completely when external forces are removed. As a consequence, it restricts the response function to a simple one-to-one mapping of the strains to the stresses and \mathcal{C} is sufficiently described by a fourth-order tensor with 81 coefficients in total.

This restriction is also known as the **generalised Hooke's Law** which also linearly approximates nonlinear stresses at small strains by the assumption of constant stiffness. Hence, it follows

$$\sigma_{ij} = \sum_{k,l=1}^3 C_{ij,kl} \varepsilon_{kl} \quad (2.59)$$

for the stress components.

Since the strain and stress tensors are symmetric by definition, only six components remain free. A reduction of the coefficients is introduced by the **Kelvin-Voigt** notation leading to the single index notation:

$$\sigma_1 = \sigma_{11}, \quad \sigma_2 = \sigma_{22}, \quad \sigma_3 = \sigma_{33}, \quad \sigma_4 = \sigma_{23}, \quad \sigma_5 = \sigma_{13}, \quad \sigma_6 = \sigma_{12} \quad (2.60)$$

$$\varepsilon_1 = \varepsilon_{11}, \quad \varepsilon_2 = \varepsilon_{22}, \quad \varepsilon_3 = \varepsilon_{33}, \quad \varepsilon_4 = 2\varepsilon_{23}, \quad \varepsilon_5 = 2\varepsilon_{13}, \quad \varepsilon_6 = 2\varepsilon_{12} \quad (2.61)$$

The notation now yields the matrix $C \in \mathbb{R}^{6 \times 6}$

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{pmatrix} \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{pmatrix} \quad (2.62)$$

Additional simplification is employed by having an idealised relationship in the work done with respect to the strain. A so-called **hyperelastic** material does not absorb the energy that is induced by the strains and releases the energy completely at recovery.

For such hyperelastic materials, the number of free coefficients of C is reduced to 21. Additional orthotropy or isotropy reduces the number further to 9 or 2 coefficients respectively.

The two remaining free coefficients denoted by λ and μ of isotropic materials are named **Lamé-constants**. These constants can be indirectly obtained by experimentally measuring the engineering parameters of **Young or elastic modulus** E and **Poisson's ratio** ν or transverse contraction. The latter opposes the material compression and ranges from $-1 < \nu < \frac{1}{2}$. The former shows the resistance force according to the tension. The Lamé-constants are then deduced from the previously measured parameters with

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)} \quad (2.63)$$

A special type of linear, isotropic materials (being nonlinear in geometric configuration with Green's tensor) are the **St. Venant-Kirchhoff** materials with the relation

$$\sigma = \lambda \text{tr}(\varepsilon) \mathbf{I} + 2\mu \varepsilon \quad (2.64)$$

Table 2.1 Mechanical properties of elastic materials

Mechanical property	Symbol	Characteristic	Unit
Elastic modulus	E	Linear slope of σ	N/mm ²
Poisson ratio	ν	Ratio of transverse contraction strain	–
Shear modulus	G	Resistance against shearing	N/mm ²
Bulk modulus	K	Resistance against compression	N/mm ²

and their hyperelastic energy function is given by

$$V = \lambda/2 \text{tr}(\varepsilon)^2 + \mu \varepsilon^2 \quad (2.65)$$

By modelling the stress-strain relationship with the St.Venant-Kirchhoff material and the generalised Hooke's law, the constitutive equation becomes very simple. The symmetric matrix is given by

$$C = \begin{pmatrix} 2\mu + \lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2\mu + \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2\mu + \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\mu \end{pmatrix} \quad (2.66)$$

An additional factor affecting the material behaviour is the dependence of the stress response on the strain rates $\dot{\varepsilon}$. Such behaviour is justified by the fact that materials have decreasing stresses under constant strains (**stress relaxation**). The latter behaviour is also known as **creep**. Another behaviour of some materials in situations of periodic load and unload can be experimentally observed. Here, the anomaly is that the stresses lag behind the strains. The presence of one of these effects classifies the material to be **visco-elastic**. Models considering such phase-shift behaviour are described in the impulse response of the material (in stresses) with respect to a strain unit pulse. A history of strains is required to perform the frequency decomposition of the strain signal yielding the temporal stress response. Therefore, proper modelling demands a considerable amount of computation time and storage which makes a precise model not yet suitable for real-time simulation. The interested reader might look into [10] for a comprehensive view. Nevertheless, ignoring the deformation history still allows to simulate some aspects of such materials.

2.2.4 Energy Principles and Variational Approach

As illustrated in Sect. 2.1.2, quantities of the energy and the work done define the motion of a dynamical system which obeys the principle of least action. In the following the estimation of these quantities will be extended to continuous bodies.

For a continuum, the total work done is contributed by the external W_e and internal W_i work. The former contribution is given by a distributed force field $\mathbf{f}(\mathbf{x})$ acting on the continuum (per unit volume) with its displacement field $\mathbf{u}(\mathbf{x})$. Thus work done by this force field equals

$$W_e = - \int_{\Omega} \mathbf{f} \cdot \mathbf{u} dV \quad (2.67)$$

assuming the external forces or moments being independent of the actual displacement. The external work W_e equals the potential energy and its negative sign indicates that the work is performed *on* the body.

Besides the external work internal work is expressed by the integration of stresses as function of the strains. Firstly, the internal work is given

$$U_0 = \int_0^{\varepsilon_{ij}} \sigma_{ij} ds \quad (2.68)$$

at unit element with an actual strain ε_{ij} . The quantity is also often referred to as **strain energy density**. With the existence of such a scalar function U the stresses are said to satisfy the energy equation and are conservative. This implies that the work done depends only on the initial and the final strain. Differentiation of the function leads to

$$\sigma_{ij} = \frac{\partial U_0}{\partial \varepsilon_{ij}} \quad (2.69)$$

The existence of U_0 is often assumed, e.g. for hyperelastic materials, even under large deformations and nonlinear elasticity. Finally, to obtain the total work done in the body integration over the whole domain Ω is needed to get

$$W_i = U = \int_{\Omega} U_0 dV \quad (2.70)$$

Furthermore, it can be concluded for a body being in equilibrium under consideration of D'Alembert's principle as in (2.27) that

$$W_i + W_e = \delta W_i + \delta W_e = 0 \quad (2.71)$$

holds true for the system, where $\delta W = \mathbf{F} \delta \mathbf{u}$ denote the work done by actual forces in moving through virtual displacements $\delta \mathbf{u}$.

A body Ω which is not only subject to a distributed body force $\mathbf{f}(\mathbf{x})$ as before but also a traction field $\mathbf{t}(s)$, yields the total energy of the system regarding virtual displacements

$$\delta W_i + \delta W_e = \int_{\Omega} \sigma_{ij} \delta \varepsilon_{ij} dV - \left(\int_{\Omega} \delta \mathbf{u} dV + \int_{\partial \Omega} \mathbf{t} \cdot \delta \mathbf{u} dS \right) = 0 \quad (2.72)$$

The total energy of the system will be denoted by

$$\Pi(\mathbf{u}) = W_i(\mathbf{u}) + W_e(\mathbf{u}) \quad (2.73)$$

Consequently, a description of the energy residing in a body has been found. With the variation of the displacement field, it is now tied in with the Hamilton principle of least action to determine the motion or configuration respectively according to the formula (2.72).

2.2.4.1 Hamilton's Principle

When one recalls the procedure of Hamilton of varying the tentative path of a particle such that the time-integral of the difference in kinetic and potential energy reaches a minimum, one will see that the procedure is applicable here as well. Hence, it expresses the body forces in terms of kinetic and potential energy in order to find the displacement which leads to a minimum of the aforementioned time integral.

Let the path of each portion of the body be defined as function $\mathbf{u}(\mathbf{x}, t)$ within a time interval between t_1 and t_2 . The variation of the path by a virtual displacement needs to satisfy the following condition

$$\delta \mathbf{u}(\mathbf{x}, t_1) = \delta \mathbf{u}(\mathbf{x}, t_2) = 0 \quad \text{for all } \mathbf{x} \quad (2.74)$$

The condition is crucial to regard the tentative path passing the start- and endpoint in addition to any admissible variation. Furthermore, from the virtual displacement it yields

$$\delta \Pi = \int_{\Omega} \mathbf{f} \cdot \delta \mathbf{u} dV - \int_{\partial\Omega} \mathbf{t} \cdot \delta \mathbf{u} dS - \int_{\Omega} \sigma : \delta \varepsilon dV \quad (2.75)$$

for the work Π done on the body at time t $\delta \mathbf{u}$.

The formula departs from (2.72) in sign change which is reasoned by the observation of the work done on the body and not in the system. The operator: denotes the dyadic product between tensors defined by

$$\sigma : \varepsilon = \sigma_{ij} \varepsilon_{ij} \quad (2.76)$$

Considering further the work done by change of momentum through the variation of \mathbf{u} , one obtains

$$\int_{t_1}^{t_2} \left(\int_{\Omega} \rho \frac{\partial^2 \mathbf{u}}{\partial t^2} \cdot \delta \mathbf{u} dV - \left[\int_{\Omega} (\mathbf{f} \cdot \delta \mathbf{u} - \sigma : \delta \varepsilon) dV + \int_{\partial\Omega} (\mathbf{t} \cdot \delta \mathbf{u}) dS \right] \right) dt = 0. \quad (2.77)$$

With partial integration and using the boundary conditions of $\delta \mathbf{u}$ it leads to

$$- \int_{t_1}^{t_2} \left(\underbrace{\int_{\Omega} \rho \frac{\partial \mathbf{u}}{\partial t} \cdot \frac{\partial \delta \mathbf{u}}{\partial t} dV}_{\text{kinetic energy}:=T} + \underbrace{\int_{\Omega} (\mathbf{f} \cdot \delta \mathbf{u} - \sigma : \delta \varepsilon) dV + \int_{\partial\Omega} (\mathbf{t} \cdot \delta \mathbf{u}) dS}_{\text{potential energy}:=\Pi} \right) dt = 0 \quad (2.78)$$

Thus, the terms yield the Hamilton's Principle of least action mathematically expressed by

$$\delta \int_{t_1}^{t_2} (T - V) dt = 0. \quad (2.79)$$

With the transformation into the latter principle, the solution in terms of displacements is found w.r.t. the forces \mathbf{f} and \mathbf{t} one may apply. These may arise when employing gravitational and interaction (constraint) forces, respectively. The forces on the border induced by a traction field are related to the contact problem which will be discussed in Chap. 3.

2.3 Numerical Simulation

With the equations derived so far it is possible to describe and compute the mechanics of a continuum, but the positional variation in the object in terms of displacements requires to find a finite subspace of functions defined on the domain fulfilling the boundary conditions of the problem. To find a solution to a problem with complex domain would be impractical or even impossible. It is therefore not only convenient but also necessary to have rather an approximate than the exact solution.

2.3.1 Discretisation and Solution

The idea in finding a numerical solution to the stated variational problem is to reduce the solution function space V to finite dimensional subspace V_0 with dimension n . The so-called Ritz method yields an approximation U_n composed by elements φ_i of basis of the subspace V_0 and appropriately chosen coefficients c_i :

$$\mathbf{u}(\mathbf{x}) \approx U_n(x) = \sum_{i=1}^n c_i \varphi_i(\mathbf{x}) \quad (2.80)$$

If $\Pi(\mathbf{u})$ defines the functional and the approximation is of interest then \mathbf{u} has to be substituted with U_n such that the variational problem of $\delta\Pi = 0$ becomes

$$\delta\Pi = \frac{\partial\Pi}{\partial c_1} \delta c_1 + \cdots + \frac{\partial\Pi}{\partial c_n} \delta c_n \quad (2.81)$$

From the definition of basis elements it follows that the derivatives are linearly independent

$$\frac{\partial\Pi}{\partial c_i} \delta c_i = 0, \quad \forall i = 1, \dots, n \quad (2.82)$$

Another method in reducing the problem is the transformation into a so-called **weak form**. Since the solution space was reduced by the subspace elements, the solution of the initial problem might not lie in the approximation space. Here the weak form comes into play by reformulating the problem in an alleviated form. Instead of enforcing the approximation solve the exact differential equation of $\delta\mathcal{I} = 0$, it has just to satisfy the equation in weighted integral

$$\int_0^1 w(\mathbf{x}) \underbrace{\delta\mathcal{I}(\mathbf{U}_n)}_{\equiv R} dx = 0 \quad (2.83)$$

where $w(\mathbf{x})$ is a weight function and $\delta\mathcal{I}(\mathbf{U}_n) \neq 0$ the residual R in the differential equation. The new formulation is equivalent to the original problem if (2.83) is satisfied for all suitable $w(\mathbf{x})$ and sufficiently smooth solution.

To define an appropriate subspace, the original domain Ω is decomposed into a finite sub-domain Ω_e which are usually represented by triangles or tetrahedrons depending on the dimensionality of the manifold. These elements simplify the finding of appropriate basis functions to fulfil the needs of the Ritz method. The decomposition of the domain of problem into elements with the definition of appropriate weight functions is called **Finite Element Method**. Each element of the sub-domain is transformed to the reference element which is used here to solve the equation on a simple domain. The weight functions within each element are represented by interpolation functions forming the subspace basis. Those functions interpolate between nodes the assigned value of the approximated function.

In the following the problem will be formulated on such an element. For simplicity, it will be observed on a tetrahedral element of unit length in Cartesian coordinates. Further, the support is chosen by linear interpolation functions

$$\psi_1 = 1 - \zeta - \nu - \xi \quad (2.84)$$

$$\psi_2 = \zeta \quad (2.85)$$

$$\psi_3 = \nu \quad (2.86)$$

$$\psi_4 = \xi \quad (2.87)$$

to describe a position \mathbf{x} in an element by local coordinates $\zeta, \nu, \xi \in [0, 1]$. The interpolation condition ensures continuity of the function across the borders of elements which requires that

$$\psi_i^e(\mathbf{x}_j^e) = \delta_{ij} \quad (2.88)$$

is satisfied, where \mathbf{x}_j^e is the position of the j -th node in the e -th element.

The application of the Ritz method to a tetrahedral subspace element Ω_e with four nodes leads to

$$\mathbf{u}^e(\mathbf{x}) = \sum_{i=1}^4 c_i^e \varphi_i^e(\mathbf{x}) = \sum_{i=1}^4 \mathbf{u}_i^e \psi_i^e(\mathbf{x}) \quad (2.89)$$

locally approximating the displacement \mathbf{u} . Thus, it can be written for \mathbf{u} and $\delta\mathbf{u}$ in matrix form by the use of interpolation functions

$$\mathbf{u} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \psi^e \Delta^e, \quad \delta\mathbf{u} = \begin{pmatrix} \delta u_x \\ \delta u_y \\ \delta u_z \end{pmatrix} = \psi^e \delta \Delta^e \quad (2.90)$$

where

$$\psi^e = \begin{bmatrix} \psi_1 & 0 & 0 & \psi_2 & 0 & 0 & \dots & \psi_n & 0 & 0 \\ 0 & \psi_1 & 0 & 0 & \psi_2 & 0 & 0 & \dots & \psi_n & 0 \\ 0 & 0 & \psi_1 & 0 & 0 & \psi_2 & 0 & 0 & \dots & \psi_n \end{bmatrix} \quad (2.91)$$

$$\Delta^e = [u_x^1 \ u_y^1 \ u_z^1 \ u_x^2 \ u_y^2 \ u_z^2 \ \dots \ u_x^n \ u_y^n \ u_z^n] \quad (2.92)$$

$$\delta\Delta^e = [\delta u_x^1 \ \delta u_y^1 \ \delta u_z^1 \ \delta u_x^2 \ \delta u_y^2 \ \delta u_z^2 \ \dots \ \delta u_x^n \ \delta u_y^n \ \delta u_z^n] \quad (2.93)$$

To transform the potential Π of (2.78) in vector form on the element, the stress-strain-relationship has to be expressed in Kelvin-Voigt notation and replaced strain by the displacement. Thus, it follows

$$\varepsilon = D\mathbf{u} \quad (2.94)$$

with

$$\varepsilon = (\varepsilon_{xx} \ \varepsilon_{yy} \ \varepsilon_{zz} \ 2\varepsilon_{xz} \ 2\varepsilon_{yz} \ 2\varepsilon_{xy})^T \quad (2.95)$$

$$D^T = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial y} \\ 0 & \frac{\partial}{\partial y} & 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial x} \\ 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 \end{bmatrix}. \quad (2.96)$$

The matrix form is obtained for the motion equation (2.56) of Newton's second law

$$\Delta^T \sigma + \mathbf{f} = \rho \frac{\partial^2 \mathbf{u}}{\partial t^2} \quad (2.97)$$

Thus it results from the principle of the virtual displacement as in (2.72) and the strain-stress-relation (2.62) on the reference element, that

$$\int_{\Omega^e} \underbrace{\left[(D\delta\mathbf{u})^T C (D\mathbf{u}) + \rho \delta\mathbf{u}^T \frac{\partial^2 \mathbf{u}}{\partial t^2} \right]}_{=K^e \mathbf{u}^e + M^e \ddot{\mathbf{u}}^e} dV - \underbrace{\int_{\Omega^e} (\delta\mathbf{u})^T \mathbf{f}}_{=\mathbf{f}^e} dV - \underbrace{\int_{\partial\Omega^e} (\delta\mathbf{u})^T \mathbf{t}}_{=\mathbf{Q}^e} dS = 0 \quad (2.98)$$

where each integrand is evaluated with respect to the subspace basis $\langle \psi_i \rangle$. Since (2.98) must hold under all admissible virtual displacements, the matrices can be

computed and the coefficients assembled for \mathbf{u}^e in one matrix given by

$$K_{ij}^e = \int_{\Omega^e} (D\psi_i)^T C (D\psi_j) dV \quad (\text{stiffness matrix})$$

$$M_{ij}^e = \int_{\Omega^e} \rho \psi_i^T \psi_j dV \quad (\text{mass matrix})$$

$$f_i^e = \int_{\Omega^e} \psi_i^T \mathbf{f} dV, \quad Q_i^e = \int_{\partial\Omega^e} \psi_i^T \mathbf{t} dS, \quad \mathbf{F}^e := \mathbf{f}_i^e + \mathbf{Q}^e \quad (\text{external force vector})$$

Considering further the time-dependency of \mathbf{u} , the evaluation yields a dynamical system

$$M^e \ddot{\mathbf{u}}(t) + \dot{\mathbf{u}}(t) + K^e \mathbf{u} = \mathbf{F}^e \quad (2.99)$$

To find the global solution to the problem, it is needed bring the nodes of the elements into correspondence. With the identification of element nodes u_i^e with a global node, a global linear system is obtained. Here, the local value of the node u_i^e is substituted by a global value U_m at each element. The complete system follows from the initial variational problem (2.81) and the assembly:

$$\delta \Pi = \sum_{I=1}^M \frac{\partial \Pi}{\partial \mathbf{U}_I} \delta \mathbf{U}_I = 0 \quad (2.100)$$

with M being the number of global nodes and Π the sum of the potential energy functions Π^e of the N elements.

$$\Pi = \sum_{e=1}^N \Pi^e \quad (2.101)$$

2.3.2 Textile Simulation Model

While earlier interactive simulators of textiles were mostly modelled by a linear spring particle system, a great improvement was made by employing a non-linear differential equation system allowing to express the physical laws by a particle system. With the approach of Baraff et al. [2] it is now possible to simulate the non-linear material behaviour in large time steps without losing numerical stability. This approach is the basis of the numerical computations used in our VR system. In the following the representation of a textile and its simulation will be explained in analogy to the FE method.

Since the basis of the approach is a particle system, i.e. a discretisation of the continuous material, the two dimensional manifold embedded in \mathbb{R}^3 describing the textile is condensed into mass points. A triangulation as it shown in Fig. 2.6 preserves the topology of the textile and determines the mechanical relations of the

Fig. 2.6 Representation of a textile by a particle system with mass lumping

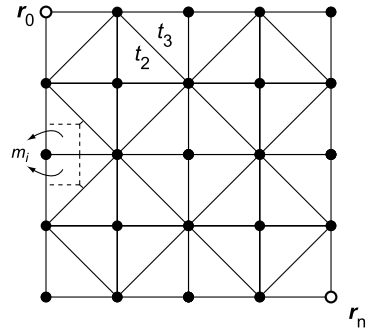
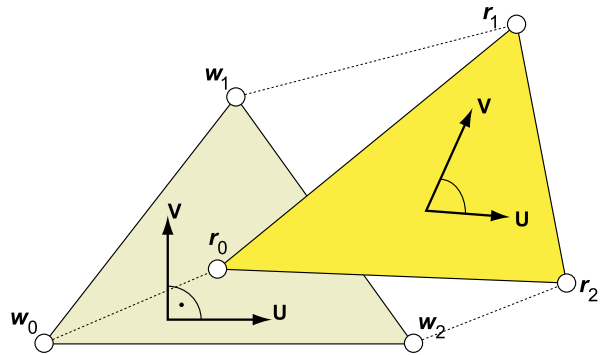


Fig. 2.7 Strains resulting in triangle deformation



particle system by linear elements. Each triangle represents a part of the original continuous surface and the associated vertices store the physical quantities, e.g. energy and inertia. With these quantities, the vertices become particles, i.e. small masses governed by the equation of motion (2.2). All in all, a particle has a mass m and a time-dependent position denoted by $\mathbf{x}(t)$ with its time-derivatives as velocity $\dot{\mathbf{x}}(t)$ and acceleration $\ddot{\mathbf{x}}(t)$, respectively. By combining all positions of the n particles, we get a single vector $\mathbf{r} \in \mathbb{R}^{3n}$.

Now, according to the equation of motion together with the inner forces written as a potential the following equation for the complete particle system is obtained

$$\ddot{\mathbf{r}} = M^{-1} \left(-\frac{\partial V}{\partial \mathbf{r}} + \mathbf{F} \right) \quad (2.102)$$

where M is the matrix of all particle masses and \mathbf{F} denotes external forces. Within the generic potential V in (2.102), the material behaviour model is hidden, i.e. it models the tensile, shear and bend stresses in dependence of the strains. The tensile and shear stresses inside each triangle corresponding to a fraction of the textile are determined. To calculate the strains, the particles \mathbf{r}_i , $i = 1, \dots, 3$ of each triangle are associated with a reference configuration \mathbf{w}_i in a two dimensional space. Figure 2.7 shows a triangle in (reference and deformed) configuration with the local vectors \mathbf{u} and \mathbf{v} . The axes of this space correspond to the dominant directions inherent to

the manufacturing process of woven fabrics. The mapping of weft direction vector into \mathbb{R}^3 is denoted by \mathbf{U} and the warp direction corresponds to \mathbf{V} respectively. The positions \mathbf{r}_i of the deformed triangle can then be used to calculate the deformed unit warp and weft vectors in three dimensional space for this triangle by solving the linear equation

$$(\mathbf{U} \ \mathbf{V}) \cdot \underbrace{(\mathbf{w}_2 - \mathbf{w}_1 \ \mathbf{w}_3 - \mathbf{w}_1)}_{:=A \in \mathbb{R}^{2 \times 2}} = (\mathbf{r}_2 - \mathbf{r}_1 \ \mathbf{r}_3 - \mathbf{r}_1) \quad (2.103)$$

substituting the reference positions with the constant matrix A and inverting leads to

$$\Leftrightarrow (\mathbf{U} \ \mathbf{V}) = (\mathbf{r}_2 - \mathbf{r}_1 \ \mathbf{r}_3 - \mathbf{r}_1) \cdot A^{-1} \quad (2.104)$$

the strains can directly associated with the deformed axis vectors

$$\Rightarrow \varepsilon_{11} = \|\mathbf{U}\| - 1 \quad \varepsilon_{vv} = \|\mathbf{V}\| - 1 \quad (2.105)$$

For shearing, the strain is given by

$$\varepsilon_{12} = \varepsilon_{21} = \frac{(\mathbf{U}, \mathbf{V})}{\|\mathbf{U}\| \|\mathbf{V}\|} \quad (2.106)$$

With the strain components constant over a triangular element, the stresses occurring in such an element can be derived. With the assumption of having a conservative energy, the strain energy function is formulated by

$$U_m(\varepsilon) = \int_0^{\varepsilon_m} \sigma_m(s) ds \quad \text{with } m \in \{11, 12, 22\} \quad (2.107)$$

In a triangle element, ε depends on the particle positions. It follows for the work U^e done on a triangle from (2.70)

$$U^e = \frac{|A|}{2} \sum_m \int_0^{\varepsilon_m} \sigma_m(s) ds \quad (2.108)$$

with $\frac{|A|}{2}$ being the area of the triangle in the reference configuration.

As the textile is modelled as a two-dimensional manifold an additional work function U^b has to be defined accounting for the stresses caused by out-of-plane deformations, termed bend forces.

The integration of bend energy within the simulation model, require additional constitutive equations necessary to reflect such stresses. The approach to consider these stress is made by defining out-of-plane strains in terms of curvature. The theory of plates or shells provides different possibilities of representing the bend energy within a plate or (curved) shell, but the in-plane stresses will be considered first. Consequently, the potential defined by

$$V^e := U^e$$

The forces \mathbf{F}_i occurring at each particle can be computed by (2.69) and with ε as function of position \mathbf{r}_i , one gets

$$\mathbf{F}_i = -\nabla V^e = -\frac{|A|}{2} \nabla V^e(\varepsilon) \quad (2.109)$$

$$= -\frac{|A|}{2} \cdot \sum_m \sigma_m(\varepsilon_m) \left(\frac{\partial \varepsilon_m}{\partial \mathbf{r}_i} \right)^T \quad (2.110)$$

For the numerical time-integration, which will be discussed in the next section, the evolution of force \mathbf{F}_i w.r.t. to the particle position \mathbf{r}_j is required. Therefore, derivation leads to

$$\frac{\partial \mathbf{F}_i}{\partial \mathbf{r}_j} = -\frac{|A|}{2} \sum_n \sum_m \frac{\partial \sigma_m(\varepsilon_m)}{\partial \varepsilon_m} \frac{\partial \varepsilon_n}{\partial \mathbf{r}_j} \frac{\varepsilon_m}{\partial \mathbf{r}_i} \quad (2.111)$$

$$+ \sum_n \sigma_m(\varepsilon_m) \frac{\partial^2 \varepsilon_n}{\partial \mathbf{P}_i \partial \mathbf{P}_j} \quad \text{with } n \in \{11, 12, 22\} \quad (2.112)$$

2.3.3 Optimised Force Calculations

Since these calculations are quite expensive in terms of floating point operations, Volino et al. proposed in [15] to easier to compute force calculations the following formulas modelling tensile behaviour.

$$\varepsilon_{11} = \|\mathbf{U}\| - 1 \quad \varepsilon_{12} = \frac{\|\mathbf{U} + \mathbf{V}\|}{\sqrt{2}} - \frac{\|\mathbf{U} - \mathbf{V}\|}{\sqrt{2}} \quad (2.113)$$

$$\varepsilon_{22} = \|\mathbf{V}\| - 1 \quad (2.114)$$

for the strain tensor. The initial formulation differs only for shear strain from (2.106) and (2.105) by being dependent on the length of \mathbf{U} and \mathbf{V} . It is justified by having better accuracy at large deformations. For real-time simulation, the strain calculations had to be even further reduced in complexity by

$$\hat{\varepsilon}_{11} = \frac{\mathbf{U}^T \mathbf{U} - 1}{2} \quad \hat{\varepsilon}_{12} = \mathbf{U}^T \mathbf{V} = \|\mathbf{U}\| \|\mathbf{V}\| \cos \angle \mathbf{U}, \mathbf{V} \quad (2.115)$$

$$\hat{\varepsilon}_{22} = \frac{\mathbf{V}^T \mathbf{V} - 1}{2} \quad (2.116)$$

With such a simplification, the costly square roots are avoided and the new functions are differentiable everywhere. However, with these formulas one gets a nonlinearity in ε_{11} and ε_{22} whereas in the ε_{12} component linear relation w.r.t. $\|\mathbf{U}\|$ and $\|\mathbf{V}\|$. Therefore, the latter is only applicable if the strain stress functions given by σ are

reparameterised accordingly.

$$\int_0^{\varepsilon(x)} \sigma_1(s) ds = \int_0^{\hat{\varepsilon}(x)} \hat{\sigma}(s) ds + C \quad (2.117)$$

$$\Leftrightarrow \quad \hat{\sigma}(\hat{\varepsilon}) = \frac{\partial}{\partial \hat{\varepsilon}} \int_0^{\varepsilon(x)} \sigma(s) ds \quad (2.118)$$

$$= \sigma(\varepsilon(x)) \frac{\partial}{\partial \hat{\varepsilon}} \varepsilon(x) \quad (2.119)$$

$$= \sigma(\varepsilon(x)) \frac{\partial}{\partial x} \varepsilon(x) \frac{\partial}{\partial \hat{\varepsilon}} x, \quad \text{with } x = \hat{\varepsilon}^{-1}(\hat{\varepsilon}) \quad (2.120)$$

As an example of for remapping from the accurate ε_{11} strain component given by (2.113) to the simple $\hat{\varepsilon}_{11}$ in (2.115), one has to set x to $|\mathbf{U}|$ and thus it follows for the stress value

$$\hat{\sigma}_{11}(\hat{\varepsilon}_{11}) = \frac{\sigma_{11}(\sqrt{2\hat{\varepsilon}_{11} + 1} - 1)}{\sqrt{2\hat{\varepsilon}_{11} + 1}}$$

The calculation of bending forces follows the approach of Volino et al. described in [17]. The basic idea of the computations is to use the discrete differential operators given by the mesh parametrisation to determine the curvature of the underlying smooth surface discretised by the triangles.

Here, the textile treated as a regular surface being the familiar mapping of $\mathbf{S} : P \subset \mathbb{R}^2 \rightarrow M \subset \mathbb{R}^3$ with the additional property of at least C^2 -continuity. At any point $\mathbf{p} \in P$ of the regular surface one can find a curve γ going through the point \mathbf{p} and sharing the same curvature $\frac{\partial^2}{\partial \gamma^2} S(\gamma(p))$ as the surface at this point. Without loss of generality, it is assumed that the coupled mapping of the curve $S(\gamma(s)) : \mathbb{R} \rightarrow M$ is arc length parameterised.

Assuming further the textile surface does not vary much in local shear strain, the length of the first derivation of the surface curve $\|\frac{\partial}{\partial t} S(\gamma(t))\|$ is equal in length of the derivative in parameter space $\|\frac{\partial}{\partial t} \gamma(t)\|$. This is supported by the fact that tense threads are generally not curved and thus finding γ satisfying $\|\frac{\partial}{\partial t} \gamma(t)\| = 1$ is comparatively easy.

The easily found discrete differential operators by the aforementioned assumptions are used to compute the second derivatives of the surface in the principal and shear direction from three and four particles respectively. Strain function related to the curvature is then defined by

$$\varepsilon(\gamma, t_0) = \left\| \frac{\partial}{\partial t^2} S(\gamma(t)) \right\|_{t=t_0} \quad (2.121)$$

$$= \left\| \sum_{i=1}^n g_i \mathbf{r}_i \right\| \quad (2.122)$$

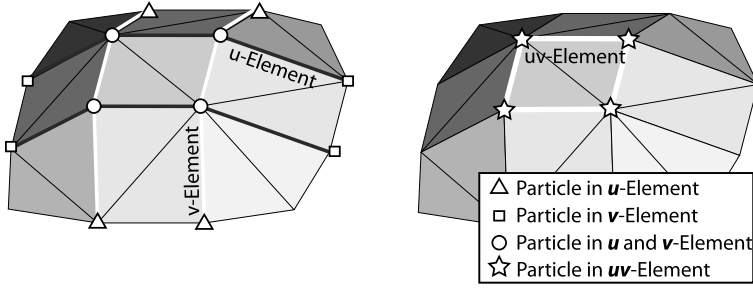


Fig. 2.8 Two types of bending elements (b^3) in left and (b^4) in right figure

Whereas g_i is the weight of the particle position \mathbf{r}_i in the discrete case. By linearising the stress function to $\sigma(\varepsilon) = B\varepsilon$ we get for the bending force \mathbf{F}_j at particle \mathbf{r}_j a weighted sum of particle positions

$$\mathbf{F}_j = -\frac{\partial}{\partial r_j} V^b(\mathbf{r}) = \frac{\partial}{\partial r_j} \int_0^\varepsilon \sigma(s) ds = \frac{\partial}{\partial r_j} \frac{1}{2} B \varepsilon^2 \quad (2.123)$$

$$= B \left\| \sum_{i=1}^n g_i \mathbf{r}_i \right\| \left\| \frac{\partial}{\partial r_j} \varepsilon \right\| \quad (2.124)$$

$$= \frac{1}{2} B \left\| \sum_{i=1}^n g_i \mathbf{r}_i \right\| \left\| \sum_{i=1}^n g_i \mathbf{r}_i \right\|^{-1} 2 \left(\sum_{i=1}^n g_i \mathbf{r}_i \right) g_j \quad (2.125)$$

$$= B g_j \sum_{i=1}^n g_i \mathbf{r}_i \quad (2.126)$$

The exposed interface of the physical simulation library allows to create bending elements associated to the particles with arbitrary weights of g_i and a bending resilience factor B . Figure 2.8 shows the two types of elements which have to be defined on the triangle mesh accounting for bending in \mathbf{u} -, \mathbf{v} - and shear direction.

Finally, internal viscosity of the materials in the different contributions of to the force (tensile, shear and bending) are computed by replacing the positional parameter \mathbf{r}_i with its time derivative $\dot{\mathbf{r}}_i$.

In order to provide a variety of virtual fabrics, the simulation model needs retrieve the material parameters from a database containing the information for the visual and haptic rendering. For the latter, each entry features the set of strain-stress functions accounting for the response in each parameter direction. The exact response values are typically obtained by material tests, e.g. tensile test. For internal representation within the computer, the functions are approximated by spline for in-plane and as a constant factor for out-plane stresses. These functions can be dynamically exchanged in the numerical simulation to represent any fabric material. Furthermore, one entry contains the density and the visual appearance of the fabric in a bitmap. For the force feedback and the tactile rendering, coefficients of static

and dynamic friction in several directions and roughness measurements in weft and warp directions are residing in an entry.

2.3.4 Numerical Integration

The common method in simulating the dynamics of an arbitrary particle system is the integration of the central force equation (2.2). One should recall that the equation defines the forces acting on the particles w.r.t. their inertia. Hence, with the force known, one can compute the resulting acceleration of each particle. From the mechanical point of view, all important information might be obtained, but for the visualisation and simulation, the resulting change in position is also important. Moreover, as the force on a particle is a function of the positions and velocities of several particles, a system of coupled second-order differential equations has to be solved. In general, the initial state of a particle system at time t_0 is known. The position of the system at the time step t_i is determined by the initial position and its velocity $\dot{\mathbf{r}}(t)$ with $t_0 \leq t \leq t_i$ by

$$\mathbf{r}(t_i) = \mathbf{r}(t_0) + \int_{t_0}^{t_i} \dot{\mathbf{r}}(t) dt \quad \text{with } \mathbf{r}(t_0) = \mathbf{r}_0. \quad (2.127)$$

In the same way, the velocity at time t_i is determined by the acceleration $\ddot{\mathbf{r}}$ and its initial velocity $\dot{\mathbf{r}}_0$

$$\dot{\mathbf{r}}(t_i) = \dot{\mathbf{r}}(t_0) + \int_{t_0}^{t_i} \ddot{\mathbf{r}}(t) dt \quad \text{with } \dot{\mathbf{r}}(t_0) = \dot{\mathbf{r}}_0 \quad (2.128)$$

With the known mass of the particles, the acceleration is computed by (2.102). The latter equations govern the motion of the particles and represent the aforementioned differential equation system in integral form. By combining the positions and velocities into a single vector \mathbf{y} , the system is reduced to first order:

$$\mathbf{y}' = \begin{pmatrix} \frac{d}{dt} \mathbf{r}_0 \\ \vdots \\ \frac{d}{dt} \mathbf{r}_n \\ \frac{d}{dt} \dot{\mathbf{r}}_0 \\ \vdots \\ \frac{d}{dt} \dot{\mathbf{r}}_n \end{pmatrix} = \begin{pmatrix} \dot{\mathbf{r}}_0 \\ \vdots \\ \dot{\mathbf{r}}_n \\ \frac{1}{m_0} \mathbf{F}_0(\mathbf{r}_0, \dots, \mathbf{r}_n, \dot{\mathbf{r}}_0, \dots, \dot{\mathbf{r}}_n) \\ \vdots \\ \frac{1}{m_n} \mathbf{F}_n(\mathbf{r}_0, \dots, \mathbf{r}_n, \dot{\mathbf{r}}_0, \dots, \dot{\mathbf{r}}_n) \end{pmatrix} =: \mathbf{f}(\mathbf{y}) \quad (2.129)$$

As the state of the particle system is known at the start of the application, the latter equation describes an initial value problem (IVP).

2.3.5 Numerical Stability

In general, methods for solving initial value problems are only able to find an approximation to the solution. The quality of such an approximation is called **consistency** and the speed in which the approximation is reached is called **convergency**. These attributes can be associated with the used method. Furthermore, a method is determined to be **stable** if for any pair of initial values in a certain distance, the trajectories stay within a upper bound. The methods more precisely exhibit a region of stability in the complex plane. The plane is being defined by the test equation of Dahlquist or test problem

$$\mathbf{y}' = \lambda \mathbf{y}(t) \quad \text{with } \mathbf{y}(0) = \mathbf{y}_0 \quad (2.130)$$

and λ is a complex number with negative real value. The analytical solution is $\mathbf{y}(t) = \mathbf{y}_0 e^{\lambda t}$. Inserting the used method into the test equation with a fixed timestep Δt , the region of stability is the set of complex numbers $\xi = \lambda \Delta t$ which leads to a monotonic sequence of approximations. If the region contains the halfspace $\{\xi \in \mathbb{C} : \text{Re}(\xi) < 0\}$, the method is called **A-stable**. This means that the method is independent of the problem \mathbf{y}' . Hauth et al. [8] analysed the numerical stability of the methods presented in the next paragraph.

Numerical methods to solve initial value problems can be categorised by their features. According to the number of used supporting points the methods are separated into **single-step** or **multistep** methods. Single-step methods take only the last solution step into account for the computation of the next time step, whereas multistep methods also use solutions for several previous time steps. The benefit of multistep methods is the higher order of consistency with minor additional computational effort compared to single-step methods. Methods are further characterised by being **explicit** or **implicit**. Explicit methods employ only a priori known values for the calculation of the next approximation. Implicit methods use values being additionally computed within the approximation of the next time step. The latter methods require to solve an equation system determining the relation of the additional data to the previously known. While the extended information increases the order of consistency and convergence, implicit methods suffer in speed because of the extra computations. As the VR system uses single-step methods the consideration will be limited to such methods.

The **explicit Euler**-method is the most simple method to compute the numerical solution for a initial value problem. With the choice of a time step width of $\Delta t > 0$ an approximation is found for the subsequent time steps

$$t_i = t_0 + i \Delta t \quad \text{with } i = 1, 2, 3, \dots \quad (2.131)$$

by the iterative equation of the next state vector

$$\mathbf{y}_{i+1} = \mathbf{y}_i + \Delta t \cdot \mathbf{y}'_i = \mathbf{y}_i + \Delta t \cdot \mathbf{f}(\mathbf{y}_i) \quad (2.132)$$

Unfortunately, the simplicity of the method is bought with a drawback in accuracy and stability, i.e. if \mathbf{y}'_i is subject to large variations, the approximation will quickly

diverge from the real solution. Differential equations in which \mathbf{y}' features a high variability with respect to small changes in \mathbf{y} are called **stiff**. Taking into account the stiffness of such systems, the computations of the method need to advance with very small time steps in order to be stable. As textile strain-stress functions are mostly non-linear and of high stiffness (e.g. wild silk reaches 490 N/m at strains of 0.7%), the method would lose its benefit of simplicity by computing too many intermediate steps.

The disadvantage of the explicit methods makes implicit methods more favourable, as their increased stability outweighs the additional computational effort. A variety of implicit methods exist which use sophisticated estimates of the average slope during Δt by calculating intermediate $\mathbf{y}'_{i+\delta}$ for several $t_{i+\delta}$. Some even go as far using a $t_{i+\delta} > t_{i+1}$. A common method among these is the **implicit Euler**-method. Due to its higher order in convergence and consistency it does not suffer in stability from stiff equations. The method differs only from the explicit scheme in its iteration rule

$$\mathbf{y}_{i+1} = \mathbf{y}_i + \Delta t \cdot \mathbf{y}'_{i+1} = \mathbf{y}_i + \Delta t \cdot \mathbf{f}(\mathbf{y}_{i+1}) \quad (2.133)$$

The delicate point here is the use of the a priori unknown state \mathbf{y}_{i+1} for the force computations. Thus an estimation of the evolution of the state is being made by a linearisation at \mathbf{y}_i , approximation. A Taylor expansion at \mathbf{y}_i yields the new iteration rule

$$\mathbf{y}_{i+1} = \mathbf{y}_i + \Delta t \left(\mathbf{y}'_i + \frac{\partial}{\partial \mathbf{y}_i} \mathbf{y}'_i \cdot \underbrace{(\mathbf{y}_{i+1} - \mathbf{y}_i)}_{\Delta \mathbf{y}} \right) \quad (2.134)$$

The substitution of the state differences by $\Delta \mathbf{y}$ leads to

$$\Delta \mathbf{y} - \Delta t \frac{\partial}{\partial \mathbf{y}_i} \mathbf{y}'_i \cdot \Delta \mathbf{y} = \Delta t \mathbf{y}'_i \quad \Leftrightarrow \quad (2.135)$$

$$\underbrace{\left(\mathbf{I} - \Delta t \frac{\partial}{\partial \mathbf{y}_i} \mathbf{f}(\mathbf{y}_i) \right)}_{:=M} \cdot \Delta \mathbf{y} = \Delta t \underbrace{\mathbf{f}(\mathbf{y}_i)}_{:=\mathbf{b}} \quad (2.136)$$

The equation with unknown state vector is now transformed into a linear equation system, where the system matrix M and the vector \mathbf{b} are directly given by evaluation. For the solution $\Delta \mathbf{y}$ it is thus necessary to invert the matrix.

Since the matrix size is of $6n \times 6n$ which can be very big depending on the amount of particles, a reduction by half can be made by removing the first $3n$ degrees of freedom (position) of the state vector and solve them explicitly by using the evaluation formula $\mathbf{r}(t_{i+1}) = \Delta t (\dot{\mathbf{r}}(t_i) + \Delta \dot{\mathbf{r}})$. For consistency, the derivative \mathbf{f} then needs to be removed from the positions as well. Despite the removal of the first $3n$ entries, the new derivative $\hat{\mathbf{f}}$ is still dependent on the positions, which has to be considered by splitting the derivation w.r.t. state vector into separate terms.

Finally, it results in

$$\left(\mathbf{I} - \Delta t \frac{\partial}{\partial \dot{\mathbf{r}}} \hat{\mathbf{f}} - \Delta t^2 \frac{\partial}{\partial \mathbf{r}} \hat{\mathbf{f}} \right) \Delta \dot{\mathbf{r}} = \Delta t \left(\hat{\mathbf{f}}(\mathbf{r}, \dot{\mathbf{r}}) + \Delta t \frac{\partial}{\partial \mathbf{r}} \hat{\mathbf{f}} \cdot \dot{\mathbf{r}} \right) \quad (2.137)$$

Yet the matrix still remains too big for standard linear solvers like inversion schemes, e.g. LU-decomposition, to use in the context of real-time simulation. Fortunately, the matrix M exhibits some nice properties which can accelerate the calculations. As the matrix is symmetric and positive definite—a result of the physical formulation—some iterative solvers exist, which can significantly reduce the effort of finding the solution. Additionally, the sparsity of the matrix due to low interdependency of the particles allows further optimisation within such solvers.

Although improved stability of implicit integration against explicit integration has been shown, it introduces numerical damping which affects as well the physical and thus directly the haptic realism as the visual realism. But it is possible to combine implicit and explicit Euler-method to find a balance between stability and physical realism. As a result, we benefit from this combination named **implicit-midpoint** Euler-method in achieving the improved accuracy of implicit integration at the same as the computational speed of explicit scheme. These advantages make the method more favourable than other popular higher-order integration methods like BDF-2 (cf. [1]). In [16] it is further shown that with minor damping the stability can be improved being competitive to more stable schemes and the method is more robust to perturbations of the solution caused by collisions.

In the scheme, the ordinary differential equation system given by the state vector \mathbf{y}_t and its first derivative \mathbf{y}' as defined previously is used to estimate the next time step as follows. The time step Δt is split apart into two steps by a split factor α , named *implicit factor* in [16]. In the first integration step, the implicit Euler would yield for advancing $\alpha\Delta t$ in time

$$\mathbf{y}_{t+\alpha\Delta t} = \mathbf{y}_t + \alpha\Delta t \cdot \mathbf{y}'_{t+\alpha\Delta t} \quad (2.138)$$

adding for the second half of the total time step an explicit Euler integration leads to

$$\mathbf{y}_{t+\Delta t} = \mathbf{y}_t + \alpha\Delta t \cdot \mathbf{y}'_{t+\alpha\Delta t} + (1 - \alpha)\Delta t \cdot \mathbf{y}'_{t+\alpha\Delta t} \quad \Leftrightarrow \quad (2.139)$$

$$\mathbf{y}_{t+\Delta t} = \mathbf{y}_t + \Delta t \cdot \mathbf{y}'_{t+\alpha\Delta t} \quad (2.140)$$

The main computational effort in the method as always in implicit schemes is the estimation of $\mathbf{y}'_{t+\alpha\Delta t}$. Since the state's derivative at $\alpha\Delta t$ cannot be directly evaluated due to its dependence on the state itself, a Taylor approximation helps to find

$$\mathbf{y}'_{t+\alpha\Delta t} \approx \mathbf{y}'_t + \frac{\partial}{\partial \mathbf{y}} \mathbf{y}'_t (\mathbf{y}_{t+\alpha\Delta t} - \mathbf{y}_t) \quad (2.141)$$

Now \mathbf{y} has to be approximated at $\alpha\Delta t$ by

$$\mathbf{y}_t \approx \mathbf{y}_{t+\alpha\Delta t} - \alpha\Delta t \cdot \mathbf{y}'_{t+\alpha\Delta t} \quad (2.142)$$

Substituting \mathbf{y}_t obtained from (2.142) in (2.141) of at the new time step we have

$$\mathbf{y}'_{t+\alpha\Delta t} \approx \mathbf{y}'_t + \alpha\Delta t \cdot \frac{\partial}{\partial \mathbf{y}} \mathbf{y}'_t \mathbf{y}'_{t+\alpha\Delta t} \quad (2.143)$$

Sorting terms followed by matrix inversion yields

$$\Leftrightarrow \mathbf{y}'_{t+\alpha\Delta t} \approx \left(I - \alpha\Delta t \cdot \frac{\partial}{\partial \mathbf{y}} \mathbf{y}'_t \right)^{-1} \mathbf{y}'_t \quad (2.144)$$

Finally, inserting the result of $\mathbf{y}'_{t+\alpha\Delta t}$ in (2.140)

$$\Rightarrow \mathbf{y}'_{t+\alpha\Delta t} \approx \mathbf{y}_t + \Delta t \cdot \underbrace{\left(I - \alpha\Delta t \frac{\partial}{\partial \mathbf{y}} \mathbf{y}'_t \right)^{-1}}_{:=M} \mathbf{y}'_t \quad (2.145)$$

The integration ends up nicely with a formula needing no longer values of \mathbf{y} and \mathbf{y}' of an intermediate time step.

The hardest part in the simulation is now reduced to the inversion of the matrix M introduced by (2.145). But since the matrix might contain several millions of entries, a straight forward inversion using standard decompositions is still not suitable for the real-time constraints. The best approach chosen is again an iterative solution of the linear equation system of M . Provided M being symmetric and positive definite and sufficiently small Δt is determined by the following inequality equation.

$$\begin{aligned} \alpha\Delta t \mathbf{x}^T \frac{\partial}{\partial \mathbf{y}} \dot{\mathbf{y}}^T \mathbf{x} &< \mathbf{x}^T \cdot \mathbf{x} \\ \Leftrightarrow \Delta t &< \frac{1}{\alpha \lambda_m \frac{\partial}{\partial \mathbf{y}} \mathbf{y}'} \end{aligned} \quad (2.146)$$

Whereas λ_m denotes the largest eigenvalue of M . Despite the requirement of conservative forces yielding a symmetric matrix, the matrix M found in the integration is far from being symmetric:

$$M = \begin{pmatrix} I & -\alpha\Delta t I \\ -\alpha\Delta t \frac{\partial \ddot{\mathbf{r}}}{\partial \mathbf{r}} & -\alpha\Delta t \frac{\partial \ddot{\mathbf{r}}}{\partial \dot{\mathbf{r}}} \end{pmatrix} \quad (2.147)$$

The splitting of M , suggested by [4], into two terms circumvents the problem to solve the complete matrix. This suggestion is supported by the fact that M has physical contributions by \mathbf{r} and $\dot{\mathbf{r}}$ which are also existent in \mathbf{y} . Therefore, a split of \mathbf{y} into these physical contributions yields a symmetric matrix \hat{M} for $\dot{\mathbf{r}}$ with

$$\hat{M} = I - (\alpha\Delta t)^2 \frac{\partial \ddot{\mathbf{r}}}{\partial \mathbf{r}} - \alpha\Delta t \frac{\partial \ddot{\mathbf{r}}}{\partial \dot{\mathbf{r}}} \quad (2.148)$$

and for the upper part of the matrix M

$$\mathbf{r}_{t+\Delta t} = \mathbf{r}_t + \Delta t \cdot (\dot{\mathbf{r}}_t + \alpha \Delta t \ddot{\mathbf{r}}_{t+\alpha \Delta t}) \quad (2.149)$$

$$\mathbf{P}_{t+\Delta t} = \dot{\mathbf{r}}_t + \Delta t \cdot \ddot{\mathbf{r}}_{t+\alpha \Delta t} \quad (2.150)$$

With the splitting of the linear equation given by M into a new symmetric, positive definite \hat{M} an efficient solution has been found for computation of the complete DES, whereas \hat{M} is an implicit Euler step for the acceleration $\ddot{\mathbf{r}}$ and the other part being an explicit Euler step for the position and velocity.

Another benefit in using an iterative solver here, is its successive approximation to the solution. This allows us to restrict the time spent for computing one time step by setting limits for the maximum number of iterations and the error in the computation. Additional advantage is given by choosing the conjugate gradient (CG) method. This iterative method needs less memory for each iteration compared to others. Contrary to decomposition methods, it does not require the matrix to reside completely in memory. Moreover, due to the speciality of our physical problem, the matrix consists of blocks with 9×9 entries being different from zero where particles have direct influence on each other. These non-zero entries are related to the elements that have been defined by the force functions in Sect. 2.3.3. But since the elements have a very small stencil on the textile mesh, the matrix is very sparse. At an iteration step of the CG method one can even more reduce the memory consumption by computing the blocks when they are needed to for the multiplication with the search vector. Having such small memory requirements can drastically reduce the time for the computations due to the ability of local bookkeeping of the main computer's CPU.

2.3.6 Linear Solvers

In the special case of solving mechanical systems, at the last step the problem is always reduced to a linear equation system with a system matrix M as seen above. Independent of the used formulation (FEM or particles), the matrix is usually sparse, symmetric, and positive definite. As a consequence, methods that take these properties into account to find the solution are of high importance. The aforementioned conjugate gradient method based on the Krylov subspaces is such a method which will be introduced here. Special attention will also be given to the convergence of the method with respect to preconditioning and the error bounds.

2.3.6.1 Krylov-Subspace Methods

The (orthogonal) Krylov subspace method is a projection method for finding the solution to a regular equation system of the form

$$A\mathbf{x} = \mathbf{b} \quad (2.151)$$

with a regular matrix $A \in \mathbb{R}^{n \times n}$ and $\mathbf{b} \in \mathbb{R}^n$. The Krylov Subspace K_m is defined as

$$K_m = K_m(A, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, A\mathbf{r}_0, \dots, A^{m-1}\mathbf{r}_0\} \quad (2.152)$$

with

$$\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0 \quad (2.153)$$

where \mathbf{x}_0 is the initial guess to the solution. The conjugate gradient method finds the optimal approximation $x_m \in x_0 + K_m$ to the solution $A^{-1}\mathbf{b}$ in terms of the orthogonality condition of the projection method

$$(b - A\mathbf{x}_m) \perp K_m \quad (2.154)$$

with any $\mathbf{x}_m \in \mathbf{x}_0 + K_m$ and $\mathbf{x}_0 \in \mathbb{R}^n$.

The CG-method requires A of (2.151) to be symmetric and positive definite. Let $\mathbf{v}_1, \dots, \mathbf{v}_m \in \mathbb{R}^n$ be the basis vectors of the Krylov Subspace K_m and $V_m = (\mathbf{v}_1, \dots, \mathbf{v}_m) \in \mathbb{R}^{n \times n}$. It follows that $V_m^T A V_m$ is regular and the resulting projection is given by

$$P_m = I - A V_m (V_m^T A V_m)^{-1} V_m^T \quad (2.155)$$

For the estimator of the error vector $\mathbf{e}_m = A^{-1}\mathbf{b} - \mathbf{x}_m$ and the residual vector $\mathbf{r}_m = A\mathbf{e}_m$ we have

$$\|\mathbf{e}_m\| \leq \|A^{-1}P_m\| \min_{x \in P_m^1} \|p(A)\mathbf{r}_0\| \quad (2.156)$$

$$\|\mathbf{r}_m\| \leq \|P_m\| \min_{x \in P_m^1} \|p(A)\mathbf{r}_0\| \quad (2.157)$$

where P_m^1 denotes the set of polynomials of maximal degree m for which $p(0) = I$ holds. In general, with $m = n$ follows the regularity of V_m such that $P_m = 0$ holds and the exact solution is found with (2.156) and (2.157). Thus, (orthogonal) Krylov subspace methods can be seen as direct scheme to solve (2.151). But practically, computers are of limited precision in their computations and introduce rounding errors whereby the solution is not necessarily found in n steps. As a result, these methods are used in an iterative manner, which is also of special interest in this work.

2.3.6.2 Conjugate Gradient-Method

Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite then it follows that for

$$F : \begin{cases} \mathbb{R}^n \rightarrow \mathbb{R} \\ \mathbf{x} \mapsto \frac{1}{2} \langle A\mathbf{x}, \mathbf{x} \rangle - \langle \mathbf{b}, \mathbf{x} \rangle \end{cases} \quad (2.158)$$

the solution to $A\mathbf{x} = \mathbf{b}$ is $\hat{\mathbf{x}} = \min_{\mathbf{x} \in \mathbb{R}^n} F(\mathbf{x})$.

The minimum of F is found by subsequently searching for local minima in designated directions $\mathbf{p} \in \mathbb{R}^n$. These search directions are determined by the condition of being locally optimal w.r.t. to F , i.e. by choosing the negative gradient, and by enforcing orthogonality $A\mathbf{p} \perp K_m$ in each step. The latter ensures that the search directions are globally optimal w.r.t. to the space K_m . The negative gradient of the function F is

$$-\nabla F(\mathbf{x}) = -\frac{1}{2}(A + A^T)\mathbf{x} + \mathbf{b} = \mathbf{b} - A\mathbf{x} = \mathbf{r} \quad (2.159)$$

The residual vectors $\mathbf{r}_0, \dots, \mathbf{r}_m$ lead to the search directions

$$\mathbf{p}_0 = \mathbf{r}_0, \mathbf{p}_m = \mathbf{r}_m + \sum_{j=0}^{m-1} \alpha_j \mathbf{p}_j \quad (2.160)$$

From the orthogonality condition follows

$$0 = \langle A\mathbf{p}_m, \mathbf{p}_i \rangle = \langle A\mathbf{r}_m, \mathbf{p}_i \rangle + \sum_{j=0}^{m-1} \alpha_j \langle A\mathbf{p}_j, \mathbf{p}_i \rangle \quad (2.161)$$

and with $\langle A\mathbf{p}_j, \mathbf{p}_i \rangle = 0$ for $i, j \in \{0, \dots, m-1\}$ and $i \neq j$ one obtains the α_i from

$$\alpha_i = -\frac{\langle A\mathbf{r}_m, \mathbf{p}_i \rangle}{\langle A\mathbf{p}_i, \mathbf{p}_i \rangle} \quad (2.162)$$

For the computed search directions, one has to find the value λ , which minimises F along \mathbf{p} . Formally, the optimum is at

$$\lambda_{opt} := \min_{\lambda \in \mathbb{R}} F(\mathbf{x} + \lambda\mathbf{p}) = \frac{\langle \mathbf{r}, \mathbf{p} \rangle}{\langle A\mathbf{p}, \mathbf{p} \rangle} \quad (2.163)$$

In a higher Krylov space K_{m+1} the solution can be incrementally improved by the formula

$$\mathbf{p}_m = \mathbf{r}_m + \frac{\langle \mathbf{r}_m, \mathbf{r}_m \rangle}{\langle \mathbf{r}_{m-1}, \mathbf{r}_{m-1} \rangle} \mathbf{p}_{m-1} \quad (2.164)$$

$$\lambda_m = \frac{\langle \mathbf{r}_m, \mathbf{r}_m \rangle}{\langle A\mathbf{p}_m, \mathbf{p}_m \rangle} \quad (2.165)$$

Finally, with the latter two equations (2.164) and (2.165) it is possible to start with a minimal Krylov space K_m and iteratively refine the solution by using a higher space until the residual vector is below an error bound. In summary, based on the formulas the algorithm then works as presented in Algorithm 2.1.

In the worst case the algorithm needs $n - 1$ steps to converge to the solution. But such a rough estimation is certainly useless in justifying the use of the CG method. As the projection strongly depends on the matrix A , it is important to find the maximal iteration steps w.r.t. the chosen error tolerance ε_{tol} . With the sequence

Algorithm 2.1 Iterative Conjugate Gradient Method

```

 $\mathbf{r}_0 \leftarrow \mathbf{b} - A\mathbf{x}_0$  {initial guess  $\mathbf{x}_0 \in \mathbb{R}^n$ , e.g.  $\mathbf{x}_0 \leftarrow \mathbf{0}$ }
 $\mathbf{p}_0 \leftarrow \mathbf{r}_0$ 
 $\alpha_0 \leftarrow \langle \mathbf{r}_0, \mathbf{p}_0 \rangle$ 
 $m \leftarrow 0$ 
while  $(m \leq n - 1) \wedge (\alpha_m > \varepsilon_{tol})$  do
   $\mathbf{v}_m \leftarrow A\mathbf{p}_m$ 
   $\lambda_m \leftarrow \frac{\alpha_m}{\langle \mathbf{v}_m, \mathbf{p}_m \rangle}$ 
   $\mathbf{x}_{m+1} \leftarrow \mathbf{x}_m + \lambda_m \mathbf{p}_m$ 
   $\mathbf{r}_{m+1} \leftarrow \mathbf{r}_m - \lambda_m \mathbf{v}_m$ 
   $\alpha_{m+1} \leftarrow \langle \mathbf{r}_{m+1}, \mathbf{r}_{m+1} \rangle$ 
   $\mathbf{p}_{m+1} \leftarrow \mathbf{r}_{m+1} + \frac{\alpha_{m+1}}{\alpha_m} \mathbf{p}_m$ 
   $m \leftarrow m + 1$ 
end while

```

\mathbf{x}_m of approximate solutions and A being symmetric and positive, definite, the error bound of $\mathbf{e}_m = A^{-1}\mathbf{b} - \mathbf{x}_m$ yields

$$\|\mathbf{e}_m\|_A \leq \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^m \|\mathbf{e}_0\|_A \quad (2.166)$$

where $\kappa(A)$ denotes the condition number of matrix A respective to the $\|\cdot\|_2$ norm. We then get from (2.156) the upper bound for the maximal steps m to satisfy the error tolerance $\|\mathbf{e}_m\|_A \leq \varepsilon_{tol} \|\mathbf{e}_0\|_A$

$$m \leq \frac{1}{2} \sqrt{\kappa(A)} \ln \left(\frac{2}{\varepsilon} \right) + 1 \quad (2.167)$$

As one can see from the above equation and (2.156) and (2.157) that the condition number is of high importance for the speed of convergence. It is therefore desirable to have a matrix with a low condition number in order to have an efficient method for finding the solution to the problem. Unfortunately, one cannot influence the problem that formed the matrix to move into the desired direction. But we can find an equivalent matrix \tilde{A} with a reduced condition number. The transformation of a given equation system to an equivalent system with a reduced κ is called **preconditioning** and used in the CG method to decrease the number of iteration steps.

2.3.6.3 Preconditioning of the CG Method

It has been shown that the condition number is the main factor in the upper bound at a certain precision ε for the conjugate gradient steps. Hence, the obvious idea is to transform the problem $A\mathbf{x} = \mathbf{b}$ into an equivalent counterpart

$$P_l A P_r \mathbf{x}_p = P_l \mathbf{b} \quad \text{with } \mathbf{x} = P_r \mathbf{x}_p \quad (2.168)$$

Algorithm 2.2 Preconditioned CG-Method (PCG)

```

 $\mathbf{r}_0 \leftarrow \mathbf{b} - A\mathbf{x}_0$  {initial guess  $\mathbf{x}_0 \in \mathbb{R}^n$ , e.g.  $\mathbf{x}_0 \leftarrow \mathbf{0}$ }
 $\mathbf{z}_0 \leftarrow P\mathbf{r}_0, \quad \mathbf{p}_0 \leftarrow \mathbf{z}_0$ 
 $\alpha_0 \leftarrow \langle \mathbf{r}_0, \mathbf{p}_0 \rangle$ 
 $m \leftarrow 0$ 
while  $(m \leq n - 1) \wedge (\alpha_m > \varepsilon_{tol})$  do
     $\mathbf{v}_m \leftarrow A\mathbf{p}_m$ 
     $\lambda_m \leftarrow \frac{\alpha_m}{\langle \mathbf{v}_m, \mathbf{p}_m \rangle}$ 
     $\mathbf{x}_{m+1} \leftarrow \mathbf{x}_m + \lambda_m \mathbf{p}_m$ 
     $\mathbf{r}_{m+1} \leftarrow \mathbf{r}_m - \lambda_m \mathbf{v}_m$ 
     $\mathbf{z}_{m+1} \leftarrow P\mathbf{r}_{m+1}$ 
     $\alpha_{m+1} \leftarrow \langle \mathbf{r}_{m+1}, \mathbf{z}_{m+1} \rangle$ 
     $\mathbf{p}_{m+1} \leftarrow \mathbf{z}_{m+1} + \frac{\alpha_{m+1}}{\alpha_m} \mathbf{p}_m$ 
     $m \leftarrow m + 1$ 
end while

```

such that $P_l A P_r \approx I$ holds. Evidently, methods for solving linear equation systems are best suited to look for an approximate inverse of the matrix as preconditioner. But to consider the requirements of the CG-method, these method must not destroy neither the symmetry nor the positive, definiteness of the matrix. Otherwise the consistency and convergence of the CG-method would not be guaranteed. Preconditioners leaving the properties of the matrix untouched are therefore scalings, such as the incomplete Cholesky decomposition and symmetric splitting methods, e.g. Jacobi-(Relaxation), Richardson- and the symmetric Gauss-Seidel-(Relaxation-) method. The formal definition of such preconditioners can be found in [11]. The preconditioner can be inserted into Algorithm 2.2 to calculate the solution based on preconditioning.

2.4 Measuring Physical Properties

Finally, with the complete numerical simulation at hand, one has only to insert the correct parameters for any material into the calculations which brings the computed results into correspondence with reality. Therefore constitutive equations determine the amount of experiments needed to find an adequate model of the simulated material. Due to the fact that materials observed at small scale level are a conglomerate of atoms in a complex crystalline structure, materials can vary in their characteristics depending on several influencing factors forcing the structure to change. But fortunately, in most cases the working range of the used materials is very limited such that most properties remain constant. Especially, applications in mechanical engineering computing static configurations of civil structures, have determined important characteristics. Taking these properties into account (cf. Table 2.1), a sufficient description of structures and their reaction upon forces in finding the equilibrium can be given. In civil engineering the quantitative determination of elastic properties is possible with the measurement procedures illustrated in Fig. 2.9.

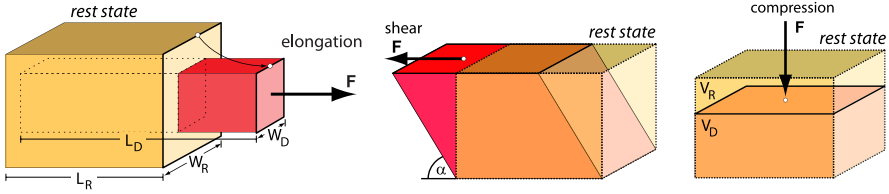


Fig. 2.9 Measurement of elastic, shear and bulk modulus

2.4.1 Textile Parameters

In addition to the previously illustrated material properties in the section before, textile materials have a more complex nonlinear behaviour. In other words, the working range in which the properties remain constant is very small. However, for our purpose of touching and stretching textiles it is sufficient to consider only nonlinear elasticity and omit other properties like creep or hysteresis. Contrary to materials like iron, concrete and wood is the regular structure of a fabric, it is composed by yarns which are themselves made of fibers. Fabrics are moreover distinguished in being woven or non-woven. Woven fabrics exhibit preferential directions due to their manufacturing process. More precisely, yarns are fixed in machine direction, the **warp** direction. A successive lifting and lowering of some of these yarns followed by a insertion of another yarn (“shooting”), **weft** direction, creates a weave pattern. Apparently, the weave structure has a strong influence on the mechanical characteristics of the fabric. Additionally, the yarns itself have different structures from the assembly of fibers, e.g. they can be twisted to a single yarn or two single yarns twisted together. Due to the complex manufacturing of fabrics, modelling textiles with linear elasticity would not reflect their real physical behaviour. Among the various parameters affecting the material behaviour those influencing the comfort of wearing are more important factors to the consumer. With the comfort being a very subjective assessment of the wearer, the textile industry has coined the term “subjective hand” or “fabric hand” as a general assessment of a set of subjective properties. A specially trained person has to assess the scale of each property in several manipulative tasks in order to characterise the fabric. As the assessment needs a skilled person to be present and is more importantly not objective, researchers correlate these subjective characteristics with the mechanical properties.

Tests for an objective assessment have been created resembling the subjective evaluation. With the Kawabata Evaluation System for Fabrics (KES-F) [9] named after its inventor Sueo Kawabata, such objective evaluation was made possible and has become the de facto standard in fabric assessment.

He constructed machines dedicated to run predefined test procedures yielding quantities related to the fabric properties. Thus the subjective properties are transformed into objective quantifiable parameters with a mechanical correspondence. As the subjective factors are influenced by the mechanical behaviour of fabric, the measurements taken by the machines cover all important mechanical properties,

Table 2.2 Characteristic values obtained by the KES-F system [9]

Property	Symbol	Characteristic	Unit
Tensility	<i>LT</i>	Linearity	gf ^a cm/cm ²
	<i>WT</i>	Tensile energy	
	<i>RT</i>	Resilience	%
Bending	<i>B</i>	Bending rigidity	gf cm ² /cm
	<i>2HB</i>	Hysteresis	gf cm ² /cm
Shearing	<i>G</i>	Shear stiffness	gf/cm degree
	<i>2HG</i>	Hysteresis at 0.5°	gf/cm
	<i>2HG5</i>	Hysteresis at 5°	
Compression	<i>LC</i>	Linearity	gf cm/cm ²
	<i>RC</i>	Resilience	%
Surface	<i>MIU</i>	Coefficient of friction	
	<i>MMD</i>	Mean deviation of MIU	
	<i>SMD</i>	Geometrical roughness	μm
Weight	<i>W</i>	Weight per unit area	mg/cm ²
Thickness	<i>T</i>	Thickness at 0.5 gf/cm ²	mm

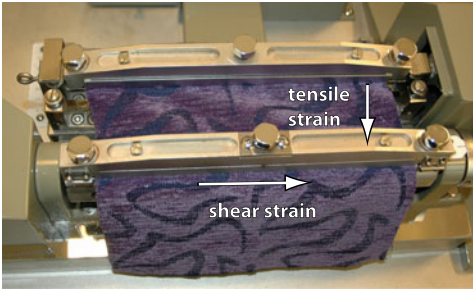
^agf: gram force = 0.00980665 N

e.g. bending, tensility, shear. Table 2.2 shows the various parameters evaluated by the machines of the KES-F.

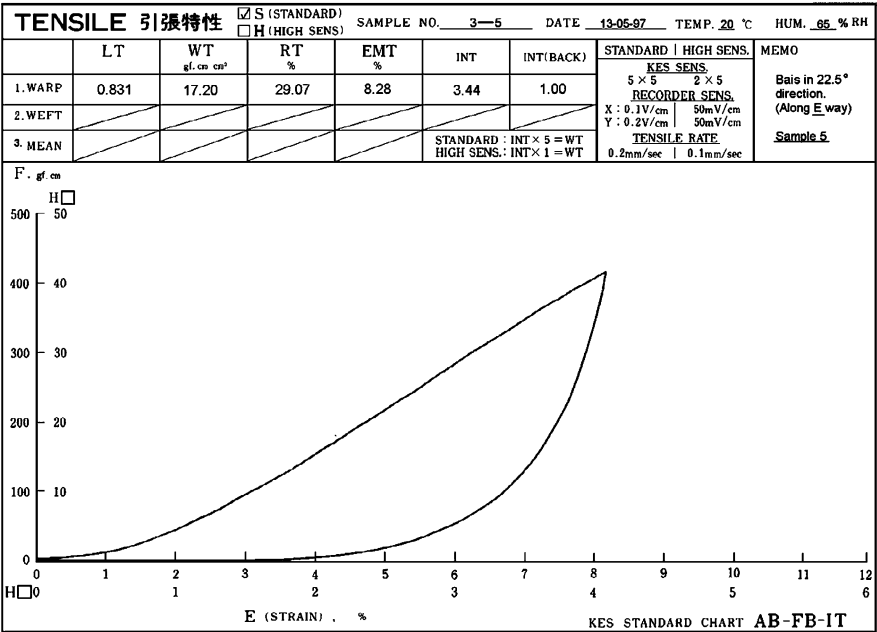
The complete extraction of the fabric's properties is done in four units. The first unit as shown in Fig. 2.10(a) measures the elastic behaviour of the fabric. A specimen 20 cm wide and 5 cm high is placed horizontally into an attachment which fixes the fabric at top and bottom of the machine. The attachment separates top and bottom ending with a constant speed yielding a one-directional strain at constant rate. The resulting force is plotted with respect to the strain. The strain increases until 500 gf/cm² in resistance force has been reached. At this limit point the movement is reversed and the recovery of the specimen is measured.

The identical setup is used in the measurement of shearing. Here, the specimen is sheared by a parallel movement of one attachment relative to the other. During the movement the resistance force is plotted until a shearing of 8° is reached. Analogous to the tensility measurement, by reversing the movement the recovery of the fabric is recorded as well. Reaching of the initial position, the machine continues to move in the opposite direction and measures the shearing behaviour in negative angle.

With the second unit as depicted in Fig. 2.11(a), the fabric's bend behaviour is evaluated. A specimen of 20 cm height and 1 cm width is hanging vertically in the machine's attachment, fixing left and right ends. In this test, the machine creates a curvature over the 1 cm width by moving one attachment on a circular path around the other starting from 0 to 90 degrees. The torques produced by bending are recorded by a sensor inside the rigid attachment.



(a)

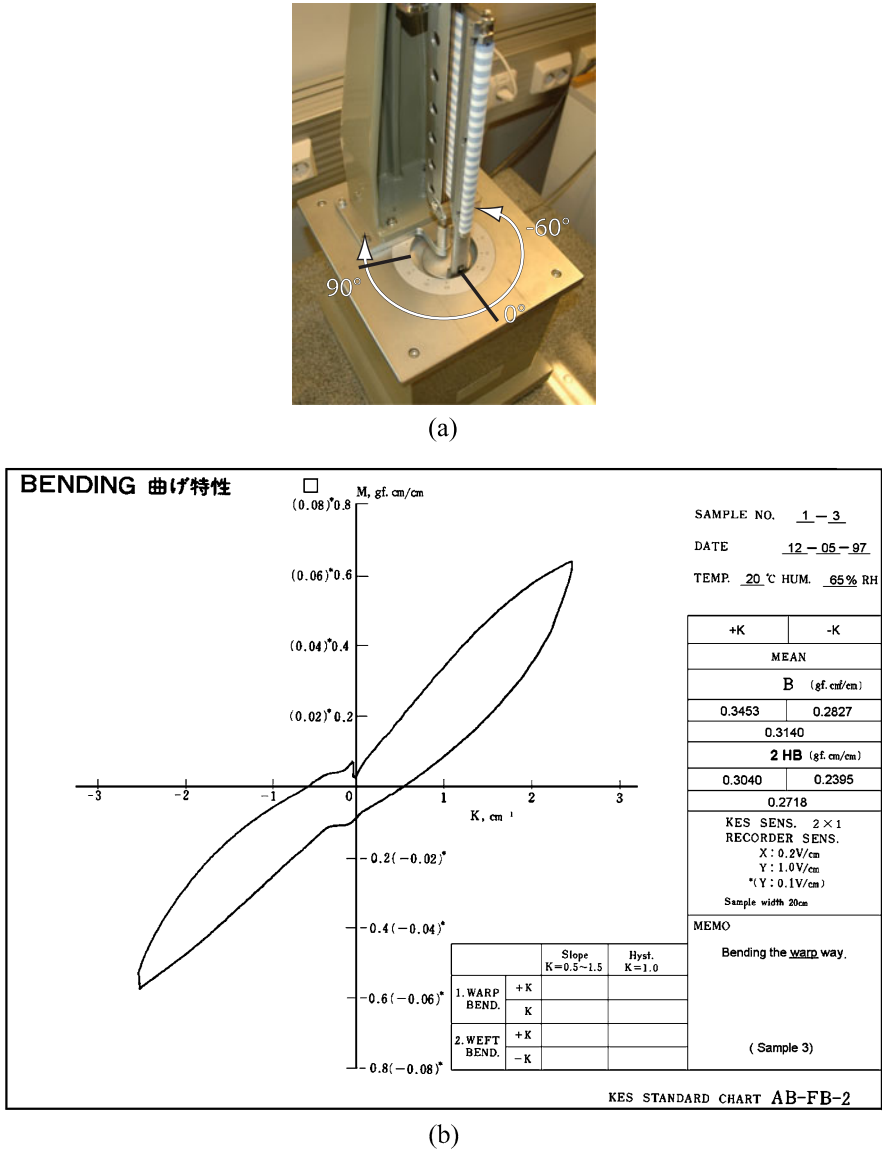


(b)

Fig. 2.10 (a) Specimen in shear and tensile strain. (b) Plot of tensile forces

Compression and thickness of a specimen is measured in a third unit. Two circular steel plates are pressed together to find the compressional resiliency of the material. In the evaluation process the position is plotted with respect to the pressure. The process reverses when 50 gf/cm² have been reached. Thickness of the material is simultaneously measured when the pressure attains 0.5 gf/cm².

The last block is used to determine the surface features of the fabric, the geometrical roughness and the friction. As the geometrical roughness is defined as the variation of the height of the surface, the machine consequently records these variations by a probe moving over the specimen with a constant speed of 1 mm/s. The probe itself consists of a U-shaped steel wire with a diameter of 0.5 mm which is pressed



by the movement and the normal force of 50 gf. An additional filtering is used to compute the mean values in both variations (cf. [9] for details).

With the measurements made by the KES-F, it is possible to extract the strain-stress relationship of the material. But still, simplifications have to be introduced to consider the measurements to be complete. Thus, we assume an orthotropic material behaviour of woven textiles, restricting ourselves to fabrics with symmetric weave patterns, independence of the deformation modes (tensile and shearing), where the nonlinear response is determined by the stress-strain curves obtained in each single weft/warp and shear direction (as seen in Figs. 2.10(b) and 2.11(b)).

References

1. Ascher, U.M., Petzold, L.R.: Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations. Society for Industrial Mathematics, Philadelphia (1998)
2. Baraff, D., Witkin, A.: Large steps in cloth simulation. In: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, pp. 43–54 ACM, New York (1998)
3. Bonet, J., Wood, R.D.: Nonlinear Continuum Mechanics for Finite Element Analysis. Cambridge University Press, Cambridge (1997)
4. Eberhardt, B., Eitzmuß, O., Hauth, M.: Implicit-explicit schemes for fast animation with particle systems. In: Eurographics Computer Animation and Simulation Workshop, vol. 2000, Springer, Berlin (2000)
5. Goldstein, H., Poole, C., Safko, J., Addison, S.R.: Classical Mechanics. Addison-Wesley, Reading (2002)
6. Gould, P.L.: Introduction to Linear Elasticity. Springer, Berlin (1994)
7. Hauth, M.: Visual simulation of deformable models. PhD thesis, Eberhard-Karls-Universität Tübingen, Germany, Dissertation (2004)
8. Hauth, M., Eitzmuß, O., Straßer, W.: Analysis of numerical methods for the simulation of deformable models. *Vis. Comput.* **19**(7), 581–600 (2003)
9. Kawabata, S.: The standardization and analysis of hand evaluation. Technical report, The Hand Evaluation and Standardization Committee, The Textile Machinery Society of Japan. Osaka Science and Technology Center Bld., 8-4, Utsubo-1-chome, Nishi-ku, Osaka 550 Japan (1980)
10. Lakes, R.S.: Viscoelastic Materials. Cambridge University Press, Cambridge (2009)
11. Meister, A.: Numerik Linearer Gleichungssysteme: Eine Einführung in Moderne Verfahren. Vieweg, Wiesbaden (2005)
12. Mezger, J.: Visual simulation of deformable models. PhD thesis, Eberhard-Karls-Universität Tübingen, Germany, Dissertation (2008)
13. Reddy, J.N.: Energy Principles and Variational Methods in Applied Mechanics. Wiley, New York (2002)
14. Reddy, J.: An Introduction to the Finite Element Method. McGraw-Hill, New York (2006)
15. Volino, P., Magnenat-Thalmann, N.: Accurate garment prototyping and simulation. *Comput. Aided Des. Appl.* **2**(1–4) (2005)
16. Volino, P., Magnenat-Thalmann, N.: Implicit midpoint integration and adaptive damping for efficient cloth simulation. *Comput. Animat. Virtual Worlds* **16** (2005)
17. Volino, P., Magnenat-Thalmann, N.: Accurate anisotropic bending stiffness on particle grids. In: International Conference on Cyberworlds, CW'07, pp. 300–307 (2007)

Haptic Interaction with Deformable Objects

Modelling VR Systems for Textiles

Böttcher, G.

2011, XII, 140 p., Hardcover

ISBN: 978-0-85729-934-5