Interactive Simulation of flexible Parts

Dissertation
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
des Fachbereiches
Physik, Mathematik und Informatik
der
Johannes Gutenberg-Universität Mainz

vorgelegt von
Mireille Grégoire
geboren in Anderlecht, Belgien

Mainz, im Juli 2007

Tag der mündlichen Prüfung: 7. Dezember 2007

D77 - Mainzer Dissertation

Zusammenfassung

Computersimulationen spielen eine immer größere Rolle für die Entwicklung von Produkten der Automobilindustrie. Ein-/Ausbausimulationen, so wie viele andere Prozesse, werden systematisch benutzt schon bevor der erste Prototyp eines Fahrzeugs gebaut wird, um zu verifizieren, ob bestimmte Komponenten einfach eingebaut werden können oder ob ein anderes Bauteil im Weg steht. Üblicherweise ist diese Art von Simulationen nur für starre Körper möglich. Dennoch beinhaltet ein Fahrzeug eine Vielzahl flexibler Bauteile verschiedener Typen: Kabel, Schläuche, Teppiche, Sitzoberflächen, Dämmungen... Da die Mehrheit der Probleme beim Benutzen dieser Simulationen eindimensionale Bauteile betreffen, haben wir uns auf die Entwicklung von numerischen Methoden zur Behandlung eindimensionaler flexibler Teile konzentriert. Die daraus entstandene Software unterstützt die Verlegung von Schläuchen, Kabeln und ganzen Kabelbäumen.

Die Modellierung von Biegung und Torsion folgt dem Cosserat Modell. Für diesen Zweck benutzen wir ein verallgemeinertes Feder-Masse-System und beschreiben seine Konfiguration mit einer sorgfältig gewählten Koordinatenmenge. Das Gewicht und die Kontaktkräfte sowie die Kräfte, welche für die Längenerhaltung verantwortlich sind, werden in kartesischen Koordinaten ausgedrückt. Biegung und Torsion können effizienter behandelt werden, wenn man Quaternionen benutzt, um die Orientierung der Segmente, welche jeweils zwei benachbarten Massenpunkten verbinden, zu beschreiben. Dieses erweiterte System ermöglicht eine einfache Formulierung aller Interaktionen mit dem am besten geeigneten Koordinatentyp und führt zu einer schmalen Hessischen Bandmatrix. Ein Energieminimierungsverfahren ermöglicht eine Lösung, welche frei von den für Feder-Masse-Systemen typischen Schwingungen ist. Die Benutzung von integralen Kräfte, ähnlich zu einem integralen Regler, macht eine genaue Durchsetzung der Zwangsbedingungen möglich. Das ganze System ist numerisch stabil und kann mit interaktiver Frame Rate gelöst werden. Es wurde in der DaimlerChrysler hauseigenen Virtual Reality Software *veo* für Anwendungen wie Ein/Ausbausimulationen und Kabelverlegung integriert und wurde gut von den Benutzern aufgenommen.

Teile dieser Arbeit wurden auf der ACM Solid and Physical Modeling Konferenz 2006 veröffentlicht und wurden zur Publikation in dem Sonderheft des Computer-Aided-Design Journal zu der Konferenz angenommen.

Abstract

Computer simulations play an ever growing role for the development of automotive products. Assembly simulation, as well as many other processes, are used systematically even before the first physical prototype of a vehicle is built in order to check whether particular components can be assembled easily or whether another part is in the way. Usually, this kind of simulation is limited to rigid bodies. However, a vehicle contains a multitude of flexible parts of various types: cables, hoses, carpets, seat surfaces, insulations, weatherstrips... Since most of the problems using these simulations concern one-dimensional components and since an intuitive tool for cable routing is still needed, we have chosen to concentrate on this category, which includes cables, hoses and wiring harnesses.

In this thesis, we present a system for simulating one dimensional flexible parts such as cables or hoses. The modeling of bending and torsion follows the Cosserat model. For this purpose we use a generalized spring-mass system and describe its configuration by a carefully chosen set of coordinates. Gravity and contact forces as well as the forces responsible for length conservation are expressed in Cartesian coordinates. But bending and torsion effects can be dealt with more effectively by using quaternions to represent the orientation of the segments joining two neighboring mass points. This augmented system allows an easy formulation of all interactions with the best appropriate coordinate type and yields a strongly banded Hessian matrix. An energy minimizing process accounts for a solution exempt from the oscillations that are typical of spring-mass systems. The use of integral forces, similar to an integral controller, allows to enforce exactly the constraints. The whole system is numerically stable and can be solved at interactive frame rates. It is integrated in the DaimlerChrysler in-house Virtual Reality Software *veo* for use in applications such as cable routing and assembly simulation and has been well received by users.

Parts of this work have been published at the ACM Solid and Physical Modeling Conference 2006 and have been selected for the special issue of the Computer-Aided-Design Journal to the conference.

Contents

Chapter 1

Introduction

1.1 Introduction

The automotive industry aims to reduce development costs and time while meeting the demand for quality and for an increasing model range. In order to meet this challenge, more and more work is done digitally. Styling reviews, Digital Mock Ups and assembly simulations are used at an early stage in the development process. Thus, potential problems can be detected and solved much earlier, long before the first physical prototype is built. Assembly simulation is one of the applications used in the construction design process: the virtual prototype is tested and optimized for feasibility and ease of assembly. Like in other Virtual Reality applications, the physical behavior of the components needs to be simulated: collisions must be detected and contact forces must be calculated to impede the interpenetration of objects and to make them slide on each other when collisions occur. Our work is based on the Virtual Reality software *veo*, developed by DaimlerChrysler Research and Technologies, which already has a real-time collision detection and interactive contact simulation [Buck and Schömer, 1998] as well as a real-time multibody dynamics [Sauer and Schömer, 1998]. This software had to treat flexible parts as rigid bodies, however, because it did not have a realistic way of dealing with flexible bodies. This implies that deformations that can occur in the physical world cannot be simulated, which limits the possibilities of the tool. A typical example of this would be an assembly simulation in which a cable should be pushed slightly aside to permit the mounting of another part. Studies from the business units show that most of the problems concerning flexible parts are encountered with cables, wire harnesses and hoses. These parts are also involved in another particular application: the routing. On the one hand wire harnesses are becoming more and more complex as the use

of electrical and electronic components in vehicles grows; on the other hand they often need to be modified to accommodate for changes in surrounding parts or for optimization. To meet the interests of the business units, we have chosen to first concentrate on simulating these parts since they cover the most urgent need. For these bodies, one of their dimensions (the length) is much bigger than the other two and their centerline contains most of the information needed to represent them. This one-dimensional nature leads to simplifications in the simulation compared to other objects such as flexible surfaces or bodies.

We will explain the motivation for simulating cables and give an overview of the virtual reality environment, in particular the assembly simulation and the cable routing applications, as well as a theoretical background of rod mechanics, of the simulation of flexible parts, and of the previous work in all the domain of interest.

Since this work has been realized during employment by DaimlerChrysler, the main focus of interest are the automotive applications of the simulation of flexible parts in VR.

1.2 Virtual Reality and its applications

1.2.1 Virtual Reality

Virtual Reality allows the user to represent digital construction data in three dimensions, to view it from all possible perspectives and to interact with it. This can be carried out either with a normal computer monitor or with immersive displays such as powerwalls or CAVES (cubic projection on four to six sides). One of the most important parts of Virtual Reality are the possibilities of interaction with the models, such as the navigation inside of the scene or the selection and manipulation of objects. This is usually supported by a special input device such as a Spacemouse or a tracking system.

In order to make the Virtual Reality more and more realistic and immersive, simulations play an ever growing role. One example is the simulation of the movements of virtual objects and of their interactions, for example during collisions.

1.2.2 Assembly simulation

Assembly simulation for rigid bodies is today state-of-the-art, there are commercially available tools and software which are used in the daily processes

of the automotive industry.

But a typical car is not only made of rigid parts but also of a lot of flexible parts: following [Yazgan, 2005] between 20 and 24% of all parts are flexible, 41% of which are one-dimensional, 33% two-dimensional and 26% three-dimensional. (See below 1.2.3 for a definition) Around 9% of all assembly simulation are not conclusive because of the presence of flexible parts; most of the cases concern one-dimensional parts. The focus will therefore be set on the one-dimensional parts since they are the most problematic ones.

1.2.3 Classification of flexible parts

The different types of flexible parts that can be encountered in a car are classified following their "number of dimensions"[Fleischer, 2005, Specht, 2006] which would be the number of dimensions required for a simplified description of their shape.

One-dimensional parts

Figure 1.1: Some of the one dimensional parts encountered in a car: on the left some cables in a door, on the right a hose

One-dimensional parts are similar to a line. They include (see Figures 1.1 and 1.2)

- *cables*: a coated electrical connection for energy and data transmission. They can be composed of only one conductor or of a combination of several strands, isolated wires or fibers. The cable core is usually made of conductive wires and is kept together with an additional insulating layer, or cable jacket.

- *hoses*: a flexible connection to convey a liquid or a gaseous media. They are used for example for the transport of hot and cold water,

hot air, as brake hoses, as vacuum hoses inside of the engine control system or as air-conditioning pipe. They can also be encountered as a protective sheathing for cables.

- *cable harnesses*: a set of connected cables presenting branchings, thickenings, thinnings...(See Figure 1.3).

Please note that we will use cables for referring to both cables and hoses in the following.

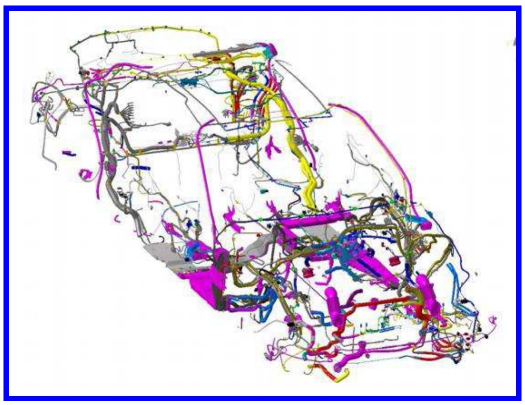

Figure 1.2: One-dimensional parts in a car

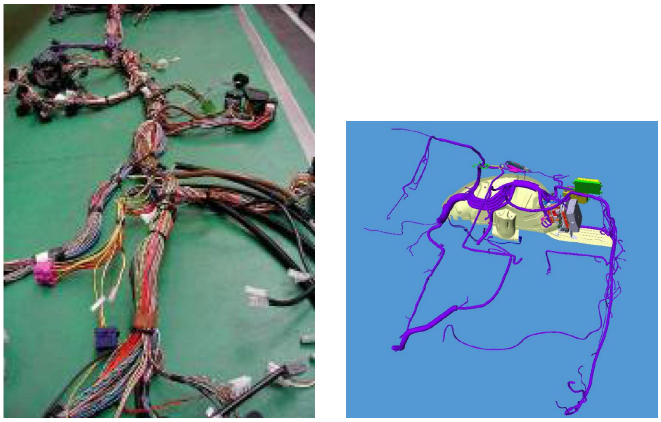

Figure 1.3: Two typical cable harnesses, one real and one in CAD, giving an idea of their complexity

Two-dimensional parts

Two-dimensional parts are similar to a surface or to a surface with a small thickness. They include (see Figures 1.4 and 1.5):

Figure 1.4: Some of the two-dimensional parts encountered in a car: on the left a carpet and on the right a bumper

- insulating parts
- carpets
- antidrum layers
- bumpers
- seat cushions

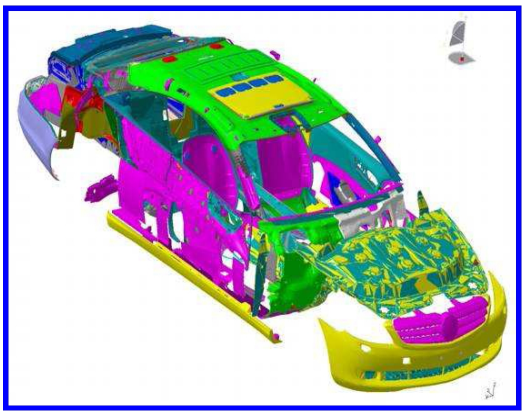

Figure 1.5: Two-dimensional parts in a car

Figure 1.6: Some of the three-dimensional and miscellaneous parts encountered in a car: from left to right a rubber funnel and a clip.

Three-dimensional and miscellaneous parts

Three-dimensional parts need a complete volumetric description to be represented. In this category are also included particular cases that are not easily reducible to the previous ones (see Figures 1.6and 1.7).

- clips
- gaskets
- rubber funnels

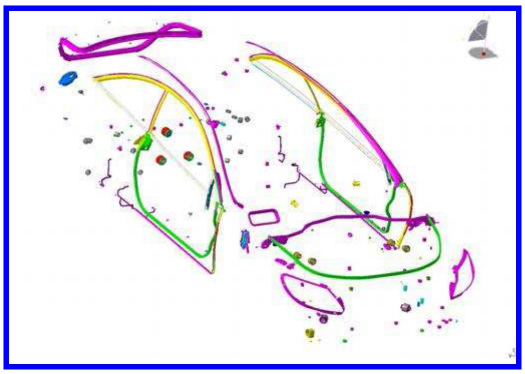

Figure 1.7: Three-dimensional parts in a car

1.2.4 Requirements and typical use cases for the simulation of flexible one-dimensional parts

During a study made by DaimlerChrysler, typical use cases and requirements for a module for simulating one-dimensional flexible parts were de-

fined. These are mainly based on the needs expressed by technical experts from the fields of Virtual Reality, simulation, design, electrics and climate as well as on the use of our prototype. The main goals are to be able to easily and intuitively create - possibly from an existing CAD model -, modify and simulate flexible wires - also taking into account their environment for example through collisions. The main focuses are, on the one hand, as realistic and as physically correct a simulation of wires as possible, and on the other hand an intuitive and easy user interface.

The flexible parts simulation system is to be integrated in the VR systems already in use, as a plug-in or a module. It should be possible to use it with all the possible VR modalities, not only on usual Desktop PC but also on multi-channel, immersive environments for which the tracking and interaction have to be consequently available. Furthermore should the system have a real-time response, which means that the reaction to user input should occur without any noticeable delay (in praxis 10 to 15 frames per second are sufficient).

A mechanism for interchanging data between the CAD and VR systems is necessary. On the one hand, data from the CAD system are tessellated to convert them to VR; the cables should then be converted to the flexible part simulation format (flexibilisation). On the other hand, the results of the modifications or simulations in VR should be transmitted to the CAD; we use for this purpose the VDA-FS (*Verband der Deutsche Automobilindustrie-FlächenSchnittstelle*) format. The returned centerlines are only to be used as an indication; the construction of the cables should still be carried out in CAD and not in VR.

The system should save and take into account all the parameters needed for the simulation. For the cables themselves, these are:

- *global simulation parameter*: the gravity vector (usually $[0, 0, -9.81]$ ms^{-2}) which is important for the simulation of sagging cables.

- *geometrical parameters*: length values (basis length, eventually - after modification of the cable - length difference (absolute or relative) and actual length), length conservation on/off (the cable can be stretched or not), inner and outer radius (the inner radius is 0 for cables) and the minimal bending radius.

- *material parameters*: Young's Modulus (in $[N/m^2]$, it describes the relation between the tension and the extension during the deformation of a rigid body under the assumption linear elastic behavior), Poisson's number (dimensionless, ratio between the relative thickness variation to the relative length variation under the influence of an outer force), material density ($[kg/m^3]$), eventually a formula for calculating the

minimal bending radius as a function of the outer radius and the maximaly allowed transverse and tensile force.

Furthermore, handles (fixed points through which one or more cable(s) have to pass) should be defined. They are characterized by their position and orientation, and whether they are fixed or sliding (if the point of the cable passing through them is fixed or movable).

A number of interaction ways should be possible, like the creation, the selection, the modification (for example by moving the handles or changing the length), the deletion, the prolongation of a cable, as well as its connection to or separation from another cable. Similarly, the user should be able to move or separate bifurcation points.

Rigid parts are often connected to cables or tubes providing them with energy, information or material flows. A mechanism for coupling these rigid and flexible parts is needed: when moving the rigid part, the cables should follow and limit the movement of the rigid part (for example when the cables are in extension).

Another very important point for realism are the collisions: parts should not interpenetrate one another. The collisions are to be detected and the cables have to be deformed in a physically plausible way. The movements of the rigid parts have to be, if necessary, stopped or modified. In the case of the collision of two flexible parts, they both have to be deformed to avoid interpenetration.

Several typical uses cases were defined [Specht, 2006]:

- *Laying proposal for a new cable*: starting and final point are known, the form of the new cable/hose has to be defined, if necessary with the provision for collisions. (See Figures 1.8 and 1.9)

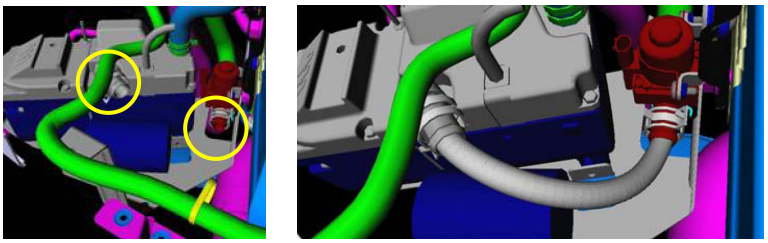

Figure 1.8: Definition of the form of a new cable: on the left the initial situation with the start and end points; on the right the new cable.

- *Small modification of an existing cable*: after the partial flexibilisation of the cable, it has to be modified to adapt to a new environment. The length is to be changed if needed. (See Figure 1.10)

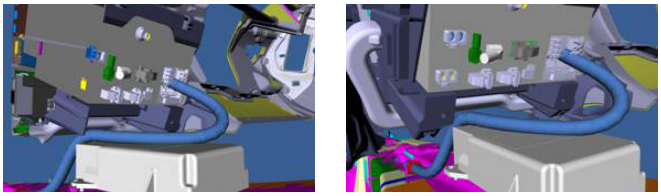

Figure 1.9: Collision free course of a new cable

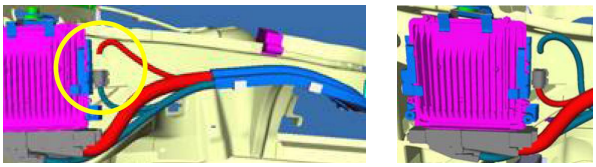

Figure 1.10: Two cables need to be interchanged after a modification of the connectors: before (left) and after(right).

- *Checking a cable in another situation*: the adequacy of a cable harness for another or a newer variant has to be checked.
- *Assembly /disassembly simulation of a control unit without connected cables*: a cable has to be pushed aside for allowing the assembly. Collisions between rigid and rigid on the one hand and rigid and flexible parts on the other hand have to be used. (See Figure 1.11)

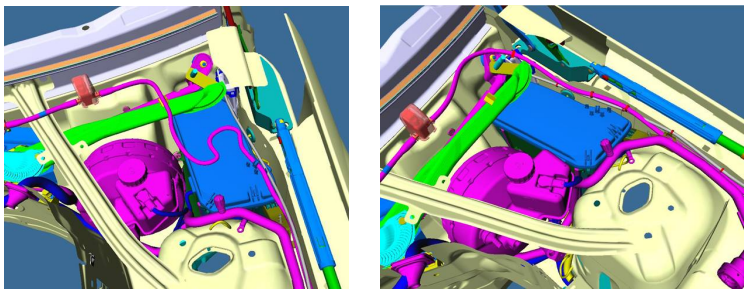

Figure 1.11: The blue control unit should be disassembled. The bothering cable has to be pushed apart before.

- *Assembly /disassembly simulation of a control unit with connected cables*: similar to the previous case, but the behavior of the connected cables should be studied as well, in particular the excess length necessary for connection and assembly. (See Figure 1.12)
- *Simulation of the physical behavior of cables*: simulation of sagging

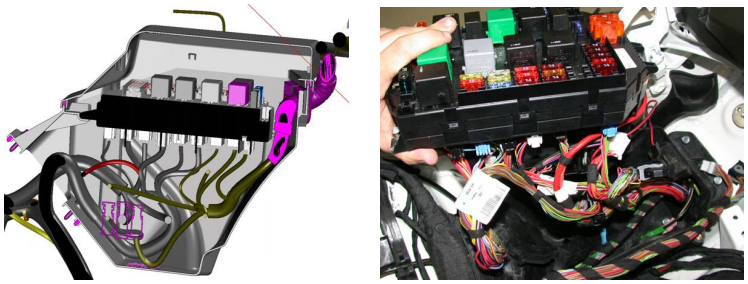

Figure 1.12: CAD - representation of a module in the assembled state and the real situation by assembly.

cables under the influence of gravity. The possible contact and chafing places are to be identified.

- *Threading simulation*: can an existent cable harness, including the connectors, be passed through a given aperture?

1.3 Simulation of flexible parts: previous work

Several approaches have been used for simulating various types of flexible bodies. We focus in particular on one-dimensional bodies, where the spectrum of solutions ranges from purely graphical representation of oscillations without any physics and with very low computational requirements [Barzel, 1997] to a complete finite element simulation, such as has been used to predict the mechanical properties of cloth [Finckh et al., 2004]. The domains which have contributed to the simulation of flexible parts are as diverse as the flexible parts themselves: hair and cloth simulation (see [Ward et al., 2007] and [Magnenat-Thalmann and Hadap., 2000] for a survey on hair simulation and [Thomaszewski and Wacker, 2006] for a survey on cloth simulation) for use in computer graphics, thread simulation for surgical training or laparoscopy, assembly simulation tools... Among the most popular computational models are differential equations [Pai, 2002], chains of rigid bodies [Hergenroether and Daehne, 2000] or spring-mass systems [Loock and Schömer, 2001].

1.3.1 Differential Equations

The behavior of flexible parts is naturally described as partial or as ordinary differential equations. One of those models is the Cosserat model, which will be described more in detail later. In order to simulate suture strands

during laparoscopic surgery, [Pai, 2002] implements the Cosserat model as a boundary value ordinary differential equation (ODE). In the typical use case in surgery simulation, the position and direction of the strand are defined at one end (corresponding to the end fixed in human tissues) and the forces and moments are defined at the other end, corresponding to the needle haptic device. The equation can then be integrated in two passes, the first one for computing stresses and strains, the second one in the opposite direction for computing positions and orientations. Unfortunately, this cannot be easily applied to boundary conditions of the type position/orientation at both ends.

1.3.2 Mass-spring systems

Mass-spring systems (also called particles systems) are a versatile way of simulating deformations and have been used for simulating a variety of flexible parts. [Baraff and Witkin, 1998] use a triangular mesh for modeling cloth, thus freeing themselves from the topological restrictions of rectangular meshes and allowing a versatile formulation of the internal energy forces. Each vertex of the particle system is associated with a constant (u, v) coordinate referring to its position in the undeformed cloth plane. The stretch and shear forces are formulated on a triangle basis, as a function of the derivative of the positions relatively to the plane coordinates (u, v), while the bending force is formulated on an edge basis as a function of the dihedral angle. Cloth/cloth contacts are handled with strong linear springs. [Loock and Schömer, 2001] implement a spring-mass model with stiff linear springs for the length conservation and torsion springs (the energy is proportional to the square of the angle between two segments) for the bending. The numerical solution is done with the implicit algorithm form [Baraff and Witkin, 1998].

For simulating soft tissue in a surgery training simulator, [Brown, 2003] implements a quasi-static algorithm (actually, a kinetic one repeated until the calculation time is elapsed $x_{new} = x_{old} + \alpha f$) for simulating soft tissues. The positions of the nodes are updated following a propagation wave starting from the nodes in contact with the surgery instruments (these nodes have then a predetermined position) and expanding to their neighbors. The new positions of the nodes already calculated are used for calculating the forces on their neighbors. If the maximum displacement of the nodes included in a ring of neighbors is smaller than a predefined value, the following rings are not updated, thus taking advantage of the localization[1] of deformations.

[1]The deformations introduced by the contact with an other object are often limited to a zone surrounding the contact point.

Suturing ropes are simulated with a *Follow the Leader* algorithm: the grasped node is moved to its new position. Its neighbors are then moved along the line joining their old position to the new position of the grasped node so that they are at the correct distance from the grasped node. The other nodes are then moved iteratively. In the case of contradictory moving directions (there are two grasped nodes, or collisions.), the resulting displacement is an average of the displacements induced by each of the constraints, which do not necessarily preserves the constraints, but the constraint violations are not visually noticeable and are averaged out after a few iterations.

1.3.3 Rigid Bodies Chain

For one-dimensional flexible parts, another possibility is to model it as a chain of rigid bodies (normally cylinders) with constraints ensuring that the ends of the cylinders correspond to one another. Springs (or other mechanisms) for flexion and torsion can be added. [Hergenroether and Daehne, 2000] implements an hierarchical, cinematic simulation of cables taking into account the length conservation, the weight and the bending rigidity of the cable. A cable is modeled with 2^n segments. The number of segments depends on the available calculation time and is adapted dynamically: when the cable is moved, a low level of resolution ensures the real-time capacity of the system whereas a static cable will be refined iteratively. For passing from one hierarchy level to the next one, each segment is divided in two. The positions of the segments are optimized by groups of four consecutive segments, which are shifted in a second pass to overlap two groups of the first pass.

[Schotte, 2005] implements an impulse-based simulation for cables. As before, a cable is represented as a chain of rigid bodies. The idea is to replace the constraint forces by correcting impulses. At the time-step t, the system is in a valid state (compliant with all the constraints). The movement of the objects is then predicted along a ballistic trajectory (under the sole influence of the external forces). Impulses for correcting the violation of the constraints are calculated which lead, after the application of these impulses at the beginning of the time-step, to a new trajectory. Iteratively, the correct values of the impulses can then be determined in order to have a valid state at the end of the time-step. The constraints (basically point-to-point articulations, two points of different rigid bodies are constrained to have the same position in world-space, but the equality of the speeds is also possible) are enforced up to a tolerance distance or speed, which determines mostly the run time. Curvature and torsion impulse with damping are added. There is no coupling of the flexion and the torsion. In our opinion, this is due to the choice of the axis for the torsion impulses. For simplifying the calculation,

they are applied along the cylinder axes. Since only the torsion impulses have an influence on that component of the rotation, each cylinder is free to rotate around that axis until the torsion is constant along the cable (the two impulses applied to a cylinder compensate one another). The resulting rotation has no influence on the global form of the cable. Furthermore, this expression of the torsion impulses violates Newton's third law (law of reciprocal actions): the two impulses are equal in magnitude but not in direction (unless the two cylinder are aligned).

1.3.4 Finite Elements Methods (FEM)

The Finite Element Method is often seen to have a good precision as shows their use in mechanical and structural engineering. They are however normally quite slow. [Finckh et al., 2004] developed an FEM model for woven fabrics, down to the individual filaments in the yarns and their interactions. This model is extremely exact and has been used to study the properties (rupture resistance...) of cloth but it is far from being real-time. [Nikitin et al., 2002] uses a precomputed FEM model for simulating cables. The cable is discretized in volume, the solution of the FEM equation $\mathbf{Ku} = \mathbf{f}$ (where $\mathbf{u}$ are the node displacements, $\mathbf{f}$ the forces on the node and $\mathbf{K}$ the stiffness matrix) is precalculated as the inverse $\mathbf{K}^{-1}$, from which only the elements corresponding to the surface nodes (for the graphical representation) or to predetermined modes are stored. An FEM approach is also used in [Julien Lemoir, 2006] for simulating catheter and guidewire.

1.3.5 Other methods

[Barzel, 1997] implements an easy and fast method for faking the dynamics of ropes: a rope is modeled as a basic shape (a catenary for a sagging one, a straight line for an hanging one...) to which gross deformations (bends...) and a wave (a sinus function) are superposed. The intuitive control of deformation is well adapted to the use in animation (for example in the film *Toy Story*), where expressive, exaggerated deformations are better than a physical simulation.

[Moll et al., 2005] uses path-planning techniques applied to minimal energy curves with a constant length. The energy is defined as $\int \kappa^2 + \tau^2 \mathrm{d}s$. The tangent is parameterized by two angles in spherical coordinates, thus automatically fulfilling the length conservation constraint. Each of the two angles is a function of the arclength; this function is parameterized by 5 parameters allowing to modify the shape of the function. After taking into account the end positions and end tangents constrains, 5 independent parameters are

left for the optimization, which is realized either with Lagrangian dynamics, random sampling of the null-space of the Jacobian or with a general purpose constrained optimization technique. For finding a path between two configurations, the parameters of the two configurations are interpolated and the energy for each step is minimized. The resulting path can then be shortened or smoothed. For multiple control points, each segment is optimized for a given length such that the sum of the segments lengths matches the global length. Changing the segment lengths allows to optimize globally the curve. [Moll and Kavraki, 2006] models a curve as a succession of helical segments with constant curvature and torsion. Iteratively, the two neighboring segments whose curvature and torsions differs most are subdivided in two segments each; the parameters of the new segments are then optimized.

[Mikchevitch A. and Gouskov, 2003][Mikchevitch A., 2003] use a decoupled model for an interactive simulation: the *real-time model* is purely geometric, relying on the deformation of patches (analogy between B-splines control points and bar network) through prescribed motion boundary conditions. The *interactive mechanical model* relies on a mechanical modelization of a cable; it is used when the geometrical one is not accurate enough.

[Bertails et al., 2005] use a static version of the Cosserat model for modeling hair. The parameters for each master strand are a sequence of curvatures and torsions. The energy is a function of the curvatures and torsions (taking into account two different bending stiffnesses due to ellipticity of hair) and of gravity (calculating the position of the strand). Collisions are treated with linear springs. Wet hair can be modeled by changing the Young modulus and the radius (wet hair swell)

[Adrien Theetten and Barsky, 2006] use Catmull-Rom splines and non-uniform rational B-splines (NURBS) for modeling rods with 4 variables (3 spatial coordinates and one for the material torsion. The strain energy (stretching, bending and twisting) is defined with an integral over the spline. A Lagragian formulation leads to the dynamic formulation.

1.4 Physical deformations of cables and the Cosserat model

The Cosserat model for rod-like solids (with one dimension - the length - much greater than the other two cross-section dimensions) is a model from continuum mechanics. It was first formulated in 1905 by the Cosserat brothers Eugène and François. The basic idea is to use a simplified representation of the rod: instead of considering the rod as a full three-dimensional body, the

fact that the cross-section is much smaller than the length allows to neglect some variations of position or stresses inside a cross section. The rod is modeled as a one-dimensional body with extra information taking into account the properties of the cross-section. It is then represented by its centerline (a curve in the usual three-dimensional space) associated to frames (whose vectors are the so-called directors) which represent material orientation and deformation. Such a model is well suited for real-time applications since it has a smaller number of variables (compared to finite elements models for example) and can nevertheless take into account a great number of physical properties. Large deformations are neither a problem since all the properties of the system can be defined relatively to the local frames.

1.4.1 Description

In order to simplify the Cosserat model and to restrict it to the sole deformations we are interested in, we will make some assumptions about the rod. Firstly, we consider only unstretchable and unshearable bodies (which represent the vast majority of objects of interest in the context of automotive construction: a cable or a hose cannot normally be stretched or sheared with a realistic amount of forces). We furthermore assume that the cross-section is homogeneous[2] and undeformable (no in-plane shearing neither in-plane stretching), and that the mechanical properties are constant along the length of the cable[3]. The general Cosserat model without the above mentioned restrictions is explained in [Antman, 1995] or in [Rubin, 2000], while the simplified model can be found in [Goss et al., 2005], [van der Heijden et al., 2003] (for an isotropic and weightless rod).

A cable of length L is parameterized by its arc length s. It is described by a function associating to each point of the centerline of a reference configuration (generally a state without tensions like for example a straight line without torsion) a vector $\mathbf{r}(s)$ describing the position of the point of the centerline and a director frame, $(\mathbf{d_1}(s), \mathbf{d_2}(s), \mathbf{d_3}(s))$ representing material directions. Under the above mentioned restrictions, the director frame is a right-handed orthonormal basis and a member of the special orthogonal

[2]The homogeneity of the cross-section only influences the calculation of the bending and torsion constant

[3]If it is not the case, one could consider having simulation properties that are a function of the arc length of the rod, but this would unduly complicate the implementation for an extremely unusual case. The case of properties changing abruptly can be modeled with *superhandles* connecting two cables with different properties.

group SO(3) (the group of rotations of $\mathbb{R}^3$).

$$\begin{array}{rcl} [0, L] & \rightarrow & \mathbb{R}^3 \times \mathrm{SO}(3) \\ s & \mapsto & (\mathbf{r}, (\mathbf{d_1}, \mathbf{d_2}, \mathbf{d_3})) \end{array}$$

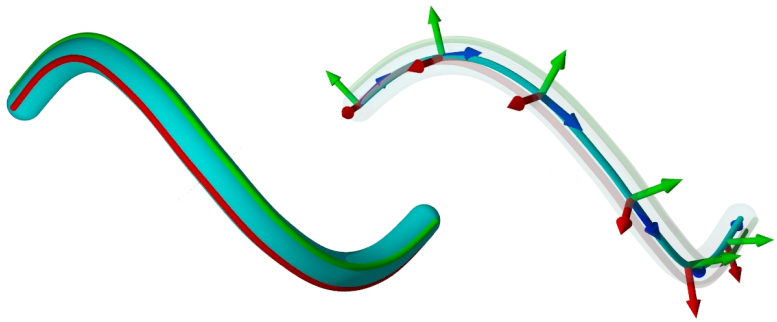

Figure 1.13: A rod and its representation in the Cosserat model: on the left, a rod (cyan) with two material lines (red and green); on the right, its centerline (cyan) with some of the director frames. The blue vector corresponds to the tangent to the centerline, the red and green vectors to the red and green material lines, respectively.

The basis of directors is adapted to the curve: the third director $\mathbf{d}_3$ points in the tangent direction of the curve. The vectors $\mathbf{d}_1$ and $\mathbf{d}_2$ show the position of two material lines in the cross section of the rod. The evolution of the basis $(\mathbf{d}_k)_{1 \leq k \leq 3}$ along the curve is represented by the Darboux vector ω[4]. Similar to the angular velocity vector (replacing the time derivative with a derivative along the arc length), this vector is defining the derivatives of the directors:

$$\frac{\mathrm{d}\,\mathbf{d}_k}{\mathrm{d}s} = \omega \times \mathbf{d}_k \text{ for } k = 1, 2, 3$$

where $\times$ represents the cross-product.

[4]The Darboux vector is usually noted $\mathbf{u}$, but we changed it for avoiding notation collision later.

1.4.2 Forces and torques

At the point of arc length s_0, the rod has a tension $\mathbf{n}(s_0)$ and an inner torque $\mathbf{m}(s_0)$. The rod is also submitted to external distributed forces like the weight or the contact forces, with a linear density f. The static equilibrium leads to

$$\begin{cases} \mathbf{0} &= \dfrac{\mathrm{d}\mathbf{n}}{\mathrm{d}s}(s_0) &+ \mathbf{f}(s_0) \\ \mathbf{0} &= \dfrac{\mathrm{d}\mathbf{m}}{\mathrm{d}s}(s_0) &+ \dfrac{\mathrm{d}\mathbf{r}}{\mathrm{d}s}(s_0) \times \mathbf{n}(s_0) \end{cases}$$

It follows that in the case of an isotropic weightless rod, where no distributed forces are applied ($\mathbf{f} = 0$), the tension $\mathbf{n}$ is constant as well as the dot product of the tension and the inner moment.

$$\begin{aligned} \mathbf{n} &= \text{constant} \\ \mathbf{n}^T\mathbf{m} &= \text{constant} \end{aligned}$$

This corresponds to the conservation of force and to the conservation of torque about the wrench axis.

1.4.3 Material properties

We know from continuum mechanics that the inner torque at a point of a rod submitted to bending and torsion (See Figures 1.14 and 1.15) is

$$\mathbf{m}(s) = B_\tau \tau\ \mathbf{T} + B_\kappa \kappa\ \mathbf{B}$$

with κ the curvature, τ the torsion, $\mathbf{T} = \mathbf{d_3}$ the tangent and $\mathbf{B}$ the binormal.

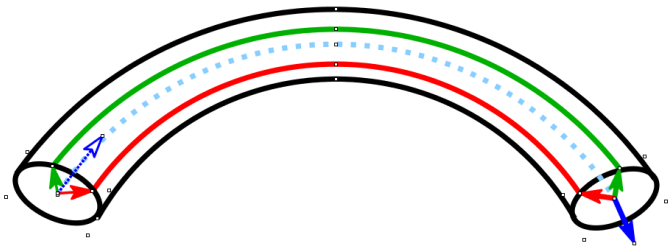

Figure 1.14: Bending of a cable. The red, green and blue lines represent the trajectory of the endpoints of the three directors.

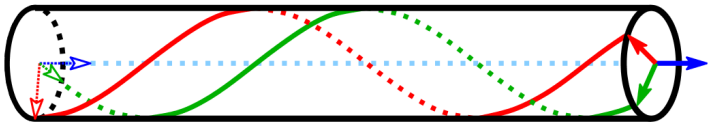

Figure 1.15: Torsion of a cable.

The coefficients B_κ and B_τ are defined in a similar way to the moments of inertia and depend on the material properties (Young's modulus E and shear modulus G) and the geometry of the cross section. For a circular and homogeneous cross section with radius R, we have

- bending stiffness:

$$B_\kappa = EI = \iint_{crosssection} E\, x^2\, dA = E\frac{\pi R^4}{4}$$

- torsional stiffness:

$$B_\tau = GJ = \iint_{crosssection} G\, r^2\, dA = G\frac{\pi R^4}{2}$$

where x and r are the distances to the bending axis (in the cross-section, passing by the centerline) and to the torsion axis (the tangent to the centerline), respectively (see Figure 1.16). In the case of a non-homogeneous cross section, the stiffnesses can be calculated by considering E and G as a function of the positions. In particular, for a hollow body like a hose, the result can be expressed with the help of the inner radius R_{inner}: $B_\kappa = E\frac{\pi(R^4 - R_{inner}^4)}{4}$

1.4.4 Equations

The combination of these equations results in the following differential equations:

$$\begin{cases} \frac{\mathrm{d}\,\mathbf{n}}{\mathrm{d}s} &= -\mathbf{f} \\ \frac{\mathrm{d}\,\mathbf{m}}{\mathrm{d}s} &= -\mathbf{d}_3 \times \mathbf{n} \\ \frac{\mathrm{d}\,\mathbf{d}_k}{\mathrm{d}s} &= \omega \times \mathbf{d}_k \\ \omega &= \kappa_1\,\mathbf{d}_1 + \kappa_2\,\mathbf{d}_2 + \tau\,\mathbf{d}_3 \\ \mathbf{m} &= B_\kappa(\kappa_1\,\mathbf{d}_1 + \kappa_2\,\mathbf{d}_2) + B_\tau\tau\,\mathbf{d}_3 \end{cases}$$

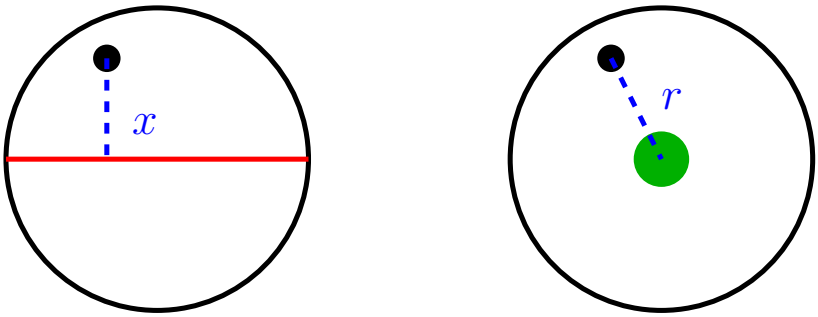

Figure 1.16: Calculation of the bending and torsion constants for the cross-section: the distances x and r in the integrand are defined as the distances from the considered point to the bending axis (a diameter of the cross-section, in red) and to the torsion axis (the centerline, in blue).

These equations have the remarkable property that the torsion is constant over the length of the rod.

$$\tau = \tau_0$$

1.4.5 Relation with the Frenet frame

We will consider the difference between the director frame and a frame that is quite often used for studying three-dimensional curves, the Frenet frame. Let us consider the Frenet frame (tangent $\mathbf{T}$, normal vector $\mathbf{N}$ and binormal vector $\mathbf{B}$) of the centerline. At any given point of the curve, the director and the Frenet frames have at least a common vector, the tangent ($\mathbf{d}_3 = \mathbf{T}$). Therefore there is a rotation around $\mathbf{d}_3$ by an angle θ which transforms $(\mathbf{N}, \mathbf{B}, \mathbf{T})$ to $(\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3)$. This angle θ shows the position of the material lines relatively to the Frenet frames: it is a "pure material torsion". The Cosserat torsion is then the Frenet torsion τ_f (geometrical torsion of the centerline) augmented with this material torsion.

$$\tau = \tau_f + \frac{\mathrm{d}\theta}{\mathrm{d}s}$$

The Darboux vector is given by the following expression:

$$\omega = \kappa_1 \, \mathbf{d}_1 + \kappa_2 \, \mathbf{d}_2 + \tau \, \mathbf{d}_3$$

where κ_1 and κ_2 are the components of the curvature on $\mathbf{d}_1$ and $\mathbf{d}_2$:

$$\kappa \mathbf{B} = \kappa_1 \, \mathbf{d}_1 + \kappa_2 \, \mathbf{d}_2$$

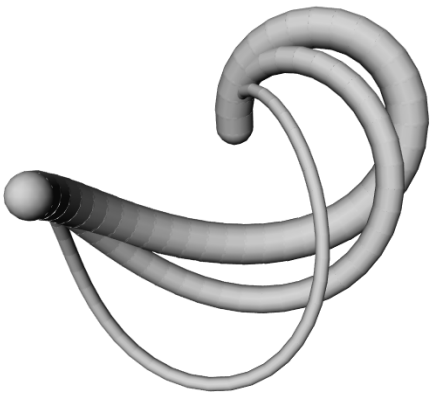

Figure 1.17: Influence of the radius: the different radii lead to different stiffnesses B_κ and B_τ. The influence on the form of the cable can be important. A change in material parameters would also produce similar results.

1.4.6 Behavior of rods

Rods can undergo a pitchfork bifurcation by snap buckling: the buckling is possible in more than one direction. Secondary bifurcations also exist, like "hockling" (formation of a loop) or snarling (highly twisted helical plies). [Goss et al., 2005]

1.5 Dynamic vs. quasi-static simulation

1.5.1 Energy and forces

From the mechanics, it is well known that, if an energy function $E(\mathbf{X})$ can be defined, the force relative to this energy is given by

$$\mathbf{f} = -\frac{\partial E}{\partial \mathbf{X}}$$

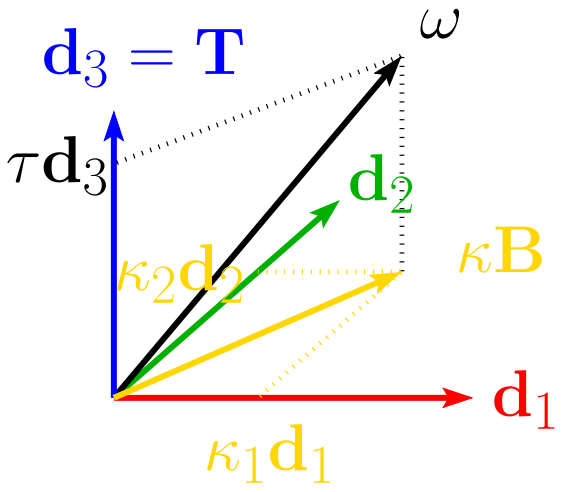

Figure 1.18: Decomposition of the Darboux vector on the Frenet frame and on the director frame: $\omega = \kappa_1\,\mathbf{d}_1 + \kappa_2\,\mathbf{d}_2 + \tau\,\mathbf{d}_3 = \kappa\mathbf{B} + \tau\mathbf{T}$

where $\mathbf{X}$ is the vector of coordinates. The simulation of flexible parts like cloth is usually formulated as a time-varying partial differential equation, with is transformed after discretization into an ordinary differential equation of the form

$$\ddot{\mathbf{X}} = \mathbf{M}^{-1}\left(-\frac{\partial E}{\partial \mathbf{X}} + \mathbf{F}\right)$$

where $\mathbf{X}$ represents the state of the system, $\mathbf{M}$ the mass matrix, $\mathbf{F}$ the external forces and the dot the time derivative. [Baraff and Witkin, 1998]

A dynamic simulation needs typically to solve an equation like the previous one. They are often based on Newton's second law or on a Lagrangian formulation. Effects of the inertia are taken into account and phenomena like oscillations and wave propagation can be represented. On the contrary, a quasi-static simulation is based on the assumption that an equilibrium state is reached after each time-step: the sum of the forces is 0. In the case where all the forces derive from a potential energy (which is the case for a wide range of forces, excluding dissipative effects like friction and damping), this is equivalent to being in a minimum of the total potential energy. Since the cables and hoses do not have a high dynamic range, considering a static solution at each time step is sufficient for most of the applications we are concerned with, like wire routing or assembly simulation.

1.5.2 Damping

One of the difficulties of dynamic formulations are the numerical oscillations that are typical for stiff systems: damping forces are needed. The formulation of damping forces for an interaction mode which do not damp any other mode or the rigid body motion is important for a well-behaved simulation. The easiest way to do this is to first define, as in [Baraff and Witkin, 1998], a vector condition $\mathbf{C}(\mathbf{x})$ which should be 0. The energy is then defined as

$$E = \frac{1}{2} k \mathbf{C}(\mathbf{x})^T \mathbf{C}(\mathbf{x})$$

with k a stiffness constant. The force is then

$$f_i = -k \frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_i}^T \mathbf{C}(\mathbf{x})$$

and its derivative

$$\frac{\partial f_i}{\partial x_j} = -k \left(\frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_i}^T \frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_j} + \frac{\partial^2 \mathbf{C}(\mathbf{x})}{\partial x_i \partial x_j}^T \mathbf{C}(\mathbf{x}) \right)$$

forms a symmetric matrix as a second derivative. The damping force is then a function of the time derivative of the vector condition (compare with the expression of f_i above)

$$d_i = -k_d \frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_i}^T \dot{\mathbf{C}}(\mathbf{x})$$

and its derivatives

$$\frac{\partial d_i}{\partial x_j} = -k \left(\frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_i}^T \frac{\partial \dot{\mathbf{C}}(\mathbf{x})}{\partial x_j} + \frac{\partial^2 \mathbf{C}(\mathbf{x})}{\partial x_i \partial x_j}^T \dot{\mathbf{C}}(\mathbf{x}) \right)$$

$$\frac{\partial d_i}{\partial \dot{x}_j} = -k_d \frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_i}^T \frac{\partial \mathbf{C}(\mathbf{x})}{\partial x_j}$$

1.5.3 Integrating Ordinary Differential Equations (ODE)

The system of equations resulting of the modeling are stiff, due to the fact that the stretching forces are much stronger than bending, torsion or shearing forces (most of the object considered are unstretchable while being at the same time easily bendable).[Baraff and Witkin, 1998]

Implicit integration methods are much better at dealing with stiff differential equations than explicit integration schemes.[Baraff and Witkin, 1998] Explicit integration schemes use the values of the derivatives at the current position for solving the generic ODE $\dot{x} = f(x)$ where the dot represent the derivative relatively to the variable t (time):

$$x(t+h) = x(t) + h\dot{f}(x(t))$$

with h the increment (time step). The stiffer the equation is, the smaller is the time step that the explicit scheme can take without becoming instable. Implicit integration schemes use on the contrary the values at the next position, which implies either an approximation or the resolution of an equation for $\dot{f}(x(t+h))$

$$x(t+h) = x(t) + h\dot{f}(x(t+h))$$

[Baraff and Witkin, 1998] have given a very interesting and robust implicit integration scheme. It is also used for example in [Loock and Schömer, 2001] For our equation, after transforming it in a first order one by introducing the velocity $\mathbf{v} = \dot{\mathbf{x}}$

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\mathbf{x}\\ \mathbf{v}\end{pmatrix} = h\begin{pmatrix}\mathbf{v}\\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x},\mathbf{v})\end{pmatrix},$$

the backward Euler method gives

$$\begin{pmatrix}\mathbf{\Delta x}\\ \mathbf{\Delta v}\end{pmatrix} = h\begin{pmatrix}\mathbf{v}_0 + \mathbf{\Delta v}\\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}_0 + \mathbf{\Delta x}, \mathbf{v}_0 + \mathbf{\Delta v})\end{pmatrix}$$

(with Δ the difference of a quantity between the two time steps and $_0$ the current value.) Replacing $\mathbf{f}$ by a Taylor series expansion and regrouping similar terms leads to the final integration scheme which is solved for $\mathbf{\Delta v}$:

$$\left(\mathbf{I} - h\mathbf{M}^{-1}\left(\frac{\partial \mathbf{f}}{\partial \mathbf{v}} + h\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right)\right)\mathbf{\Delta v} = h\mathbf{M}^{-1}\left(\mathbf{f_0} + h\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{v_0}\right)$$

The trivial step $\mathbf{\Delta x} = h(\mathbf{v_0} + \mathbf{\Delta v})$ gives the new position. This equation can be modified for introducing an explicit control of positions and speeds along constraint directions.

Chapter 2

Simple Spring Mass Model

2.1 Introduction

The rod is modeled as a sequence of mass points (lying on the centerline of the cable) which are connected with different kinds of springs: linear springs for the length conservation and torsion springs for the bending. The knowledge of the centerline leaves one degree of freedom unspecified, namely the material rotation around the centerline.

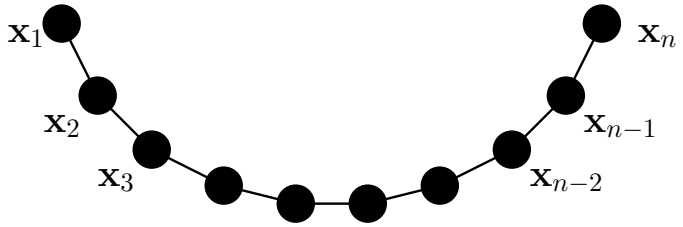

Figure 2.1: The cable is represented as a chain of mass points.

Since the Cosserat theory also considers the material direction, we need a new variable at each point to represent it, for example θ, as the angle between the Frenet and the director frames. (See 1.4.5) The global torsion is the sum of both the Frenet torsion - calculated from the coordinates of the mass points - and the pure material torsion - calculated as the derivative of the fourth coordinate θ. The energy relative to the bending and torsion is then $E = \frac{1}{2}(B_\kappa \kappa^2 + B_\tau \tau^2)$ and the forces are calculated as the negative of the gradient of the energy.

For each considered interaction, we will define an energy from which we will derive the forces and the Hessian, which will be useful for the numerical solution.

2.1.1 Notation

The cable has n points of discretization. The point i for $1 \leq i \leq n$ has the coordinates $\mathbf{x}_i \in \mathbb{R}^3$. It has also a fourth coordinate, θ_i representing the material torsion.

For each segment i between the points i and $i+1$, we define

$$\mathbf{\Delta x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$$

and $L_i = \|\mathbf{\Delta x}_i\|$ is the length of segment i.

$$\mathbf{u}_i = \frac{\mathbf{\Delta x}_i}{\|\mathbf{\Delta x}_i\|}$$

is the unitary vector in the direction of segment i.

2.2 Length Conservation

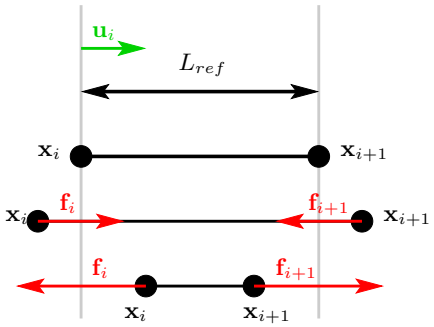

Figure 2.2: The force (in red) due to the length conservation of segment i acts on both points $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ with opposite directions along the segment. Its amplitude is proportional to the difference between the actual length and its set value. From top to bottom: the equilibrium state, the segment is too long, the segment is too short.

Cables are, in praxis, unstretchable: the length of each segment has to be kept constant. We use for this purpose strong linear springs. The energy is defined classically as

$$E_{Length,\,i} = \frac{1}{2} k_{Length} (L_i - L_{ref})^2$$

where k_{Length} is the constant of the spring. The force on the point $\mathbf{x}_i$, just opposed to the force on point $\mathbf{x}_{i+1}$ is (see Figure 2.2)

$$\mathbf{F}_{Length,i,\mathbf{x}_i} = k_{Length} \mathbf{u}_i (L_i - L_{ref}) = -\mathbf{F}_{Length,i,\mathbf{x}_{i+1}}$$

2.3 Weight

We consider the cable as a chain of mass points with equal masses m. The total mass of the cable is nm. The energy is

$$E_{Weight,i} = -m\mathbf{g}^T\mathbf{x_i}$$

where $\mathbf{g} \in \mathbb{R}^3$ is the acceleration of gravity. The force is then

$$F_{Weight,i} = m\mathbf{g}$$

Since the force does not depend on any of the coordinates, the weight has no contribution to the Hessian.

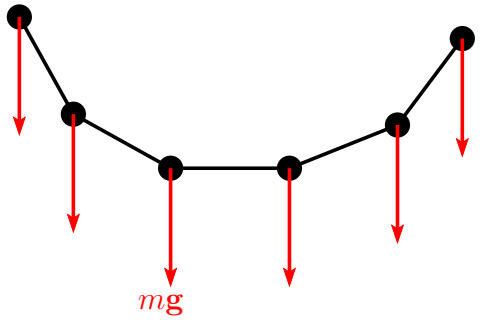

Figure 2.3: Force due to the weight

2.4 Bending

The bending energy is a function of the curvature κ which in turn is the inverse of the curvature radius. For three given consecutive points, the curvature radius at the center point is taken to be the radius of the circumscribed circle to the triangle, i.e. the radius of the circle passing by those 3 points.

In the case where the segment lengths are equals to L, the bending radius can be quite easily calculated. Let A, B, C be three consecutive vertexes of the cable (with the respective coordinates $\mathbf{x}_{i-1}$, $\mathbf{x}_i$, $\mathbf{x}_{i+1}$) and O the center of the circumscribed circle of the related triangle ABC. Let α be the angle between two consecutive segments (the angle between $\overrightarrow{AB}$ and $\overrightarrow{BC}$). If the two segments are aligned and in the same direction, the angle α is 0. Due to symmetry reasons and to the fact that the sum of the angles of a triangle is π, we obtain that the angle at the center is $\widehat{BOC} = \alpha$ and $\sin\frac{\alpha}{2} = \frac{\frac{L}{2}}{R}$ where R is the radius of the circumscribed circle.

We finally have

$$\kappa = \frac{1}{R} = \frac{2}{L}\sin\frac{\alpha}{2}$$

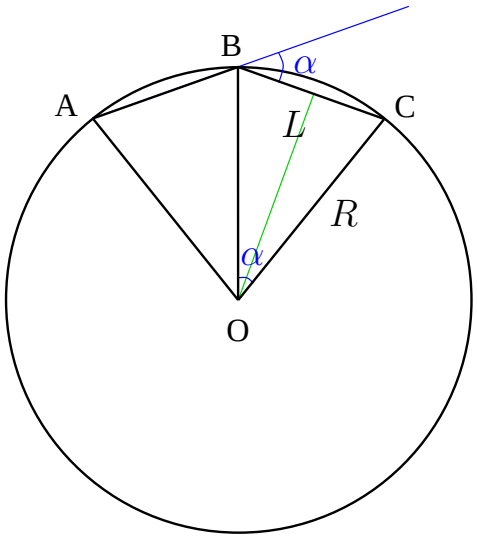

Figure 2.4: Calculation of the curvature radius R as a function of the angle α

The associated energy is then

$$E = \frac{1}{2}k_\kappa\kappa^2$$

We calculate $\cos\alpha$ from the dot product of the two unitary vectors along the neighboring segments. We have

$$\cos\alpha = 1 - 2\sin^2\frac{\alpha}{2} = 1 - \frac{L^2}{2}\kappa^2$$

and the energy is finally

$$E = k_\kappa\frac{1-\cos\alpha}{L^2}$$

$$E = \frac{k_\kappa}{L^2}(1-\cos\alpha) = \frac{k_\kappa}{L^2}(1-\mathbf{u}_i^T\mathbf{u}_{i+1}) = \frac{k_\kappa}{L^2}(1-\mathbf{u}^T\mathbf{v})$$

where $\mathbf{u} = \mathbf{u}_i$ and $\mathbf{v} = \mathbf{u}_{i+1}$.

If we also define $\mathbf{\Delta x} = \mathbf{x}_i - \mathbf{x}_{i-1}$ and $\mathbf{\Delta y} = \mathbf{x}_{i+1} - \mathbf{x}_i$ (see Figure 2.5) for simplifying the notation, the derivatives of the energy are:

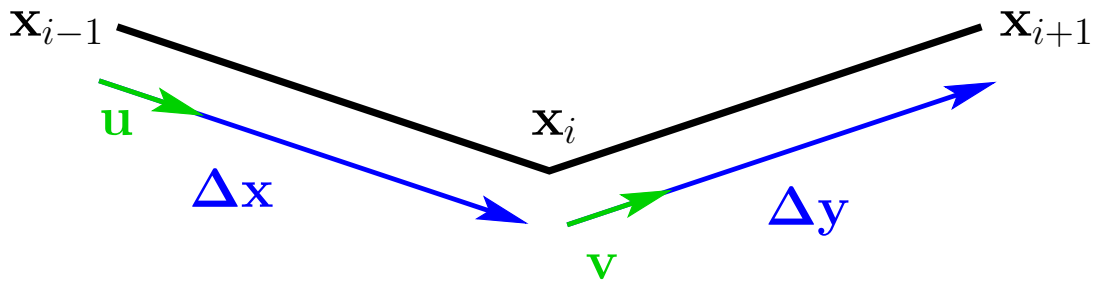

Figure 2.5: Notation for the calculation of the curvature

$$\begin{aligned}\mathbf{F}_{\mathbf{\Delta x}} &= \frac{k_\kappa}{L^3}(\mathbf{v} - \mathbf{u}^T\mathbf{v}\mathbf{u}) \\ \mathbf{F}_{\mathbf{\Delta y}} &= \frac{k_\kappa}{L^3}(\mathbf{u} - \mathbf{u}^T\mathbf{v}\mathbf{v})\end{aligned}$$

These derivatives are perpendicular to the respective segments (See figure 2.6), which means that $\mathbf{F}_{\mathbf{\Delta x}}^T\mathbf{\Delta x} = 0$ and that $\mathbf{F}_{\mathbf{\Delta y}}^T\mathbf{\Delta y} = 0$, and lie in the plane defined by $\mathbf{u}$ and $\mathbf{v}$ (in the case where $\mathbf{u}$ and $\mathbf{v}$ are collinear, the force is 0): this results in a torque on each of the segments. For example the first segment undergoes a torque of

$$M = \|\mathbf{\Delta x}\| F_{\mathbf{\Delta x}} = \frac{k_\kappa}{L^2}\|\mathbf{v} - \mathbf{u}^T\mathbf{v}\mathbf{u}\|$$

2.5 Torsion

The torsion in the Cosserat model is the sum of two components: the Frenet torsion τ_f on the one hand and the material torsion τ_m on the other hand.

$$\tau = \tau_f + \tau_m$$

2.5.1 Frenet torsion

with binormals

We studied two ways of computing the Frenet torsion: one using the binormal vector and one using a function of the derivatives of the coordinates. For the first one, it is well known that

$$\frac{\mathrm{d}\mathbf{B}}{\mathrm{d}s} = -\tau_f\mathbf{N}$$

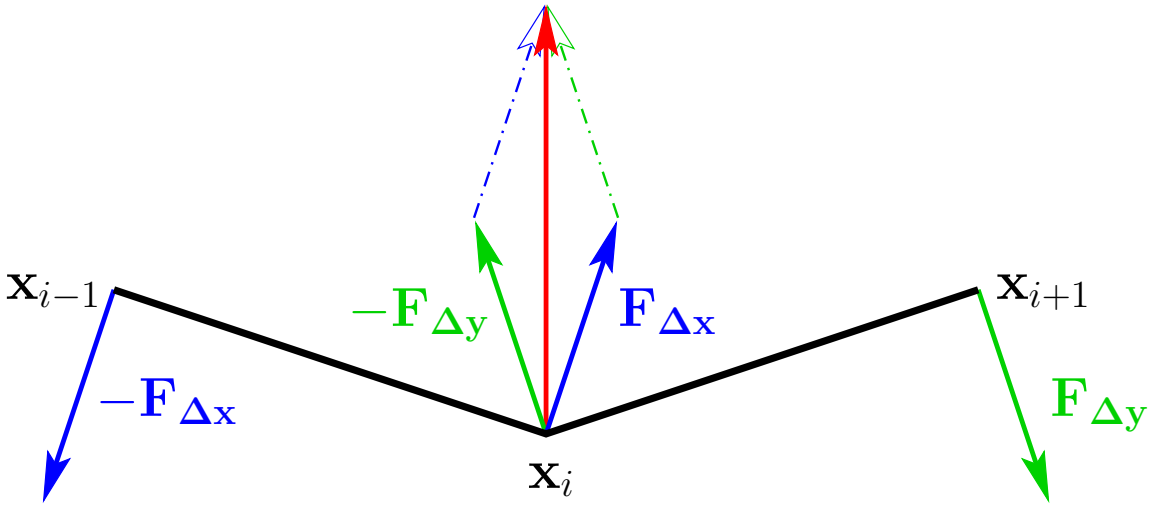

Figure 2.6: curvature forces: in blue the torque on the segment $\mathbf{\Delta x}$, in green the one on the segment $\mathbf{\Delta y}$ and in red the resultant force for the point $\mathbf{x}_i$

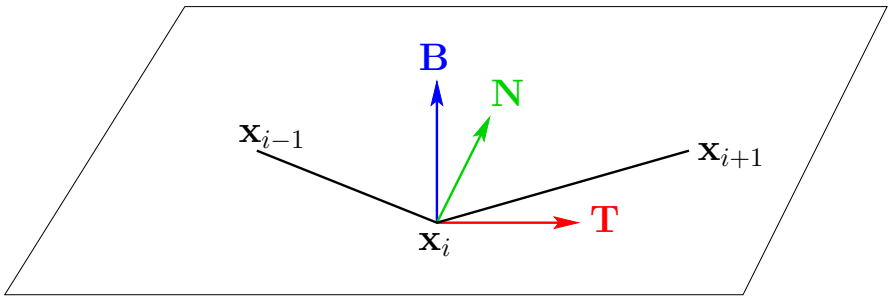

Figure 2.7: binormal

Since $\mathbf{B}$ is a unit vector, the change of $\mathbf{B}$ is a rotation and the torsion is - in the discretized system - proportional to the angle between the two binormal vectors at the points i and $i+1$:

$$\tau_f = \frac{\angle(\mathbf{B}_i, \mathbf{B}_{i+1})}{|s_{i+1} - s_i|}$$

where s_i is the arc length at point i. Since the plane defined by the three points $i-1$, i, $i+1$ contains the tangent $\mathbf{T}$ and the normal $\mathbf{N}$, the binormal at point i is orthogonal to both $\mathbf{u_{i-1}}$ and $\mathbf{u_i}$ and is calculated as:

$$\mathbf{B}_i = \frac{\mathbf{u_{i-1}} \times \mathbf{u_i}}{\|\mathbf{u_{i-1}} \times \mathbf{u_i}\|}$$

with $\mathbf{u_i}$ the unit vector between $\mathbf{r}(s_i)$ and $\mathbf{r}(s_{i+1})$:

$$\mathbf{u_i} = \frac{\mathbf{x}(s_{i+1}) - \mathbf{x}(s_i)}{\|\mathbf{x}(s_{i+1}) - \mathbf{x}(s_i)\|}$$

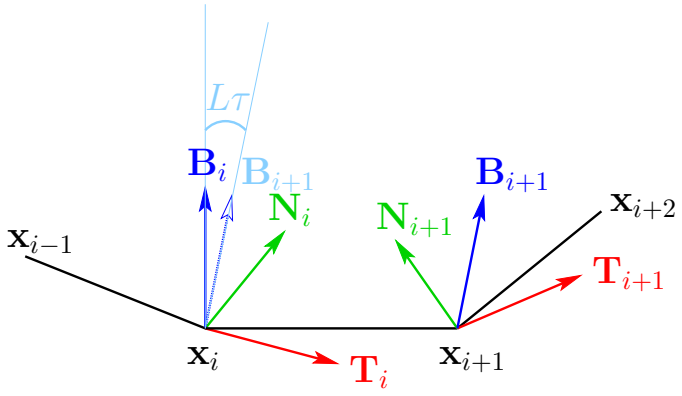

Figure 2.8: The torsion is proportional to the angle between two consecutive binormals.

As we found out, this scheme is - in particular in the case of a small curvature - extremely sensitive to noise in the position of the mass points. When for example $\mathbf{u_{i-1}}$ and $\mathbf{u_i}$ are fixed, and the angle between $\mathbf{u_i}$ and $\mathbf{u_{i+1}}$ is small, a very small change in $\mathbf{u_{i+1}}$ could mean a huge difference for τ_f, just like at the Earth poles a small change in position can mean a large change in longitude. Accordingly, the forces in case of a small curvature are huge, which leads to numerical instability and impedes the convergence.

Inflexion points represent another problem: the binormals are not defined at such points and undergo a discontinuity (opposite directions). In a configuration with (almost) inflexion points, the direction of the binormal can

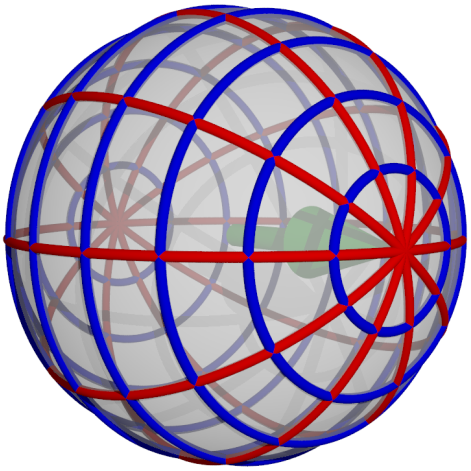

Figure 2.9: Sensitivity of the binormal: the green arrow represent $\mathbf{u}_i$. The sphere represents the possible values for $\mathbf{u}_{i+1}$ (represented with the same origin as $\mathbf{u}_i$). The blue circles are line of constant curvature. The red circles are lines of constant torsion (for a given $\mathbf{u}_{i-1}$). The torsion is not continuous at the point $\mathbf{u}_{i+1}$=$\mathbf{u}_i$, which leads to problems.

change from one iteration to another, which is of course impractical. An easy solution is to precalculate the binormal before each iteration and, when necessary, to multiply it by -1 in order to insure $\mathbf{B}_i^T\mathbf{B}_{i+1} \geq 0$.It is also necessary to keep these places in memory to take into account the possible sign changes for the subsequent calculations.

Derivative

The Frenet torsion can also be calculated as a function of the successive derivatives of the center line $\mathbf{r}(s)$:

$$\tau = \frac{|\dot{\mathbf{r}}\ddot{\mathbf{r}}\dddot{\mathbf{r}}|}{\kappa^2}$$

$$\kappa = \|\dot{\mathbf{x}} \times \ddot{\mathbf{x}}\|$$

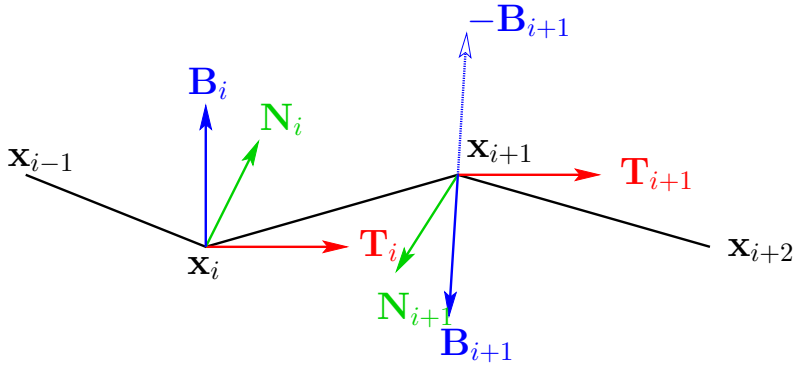

Figure 2.10: Change of sign of the binormal around an inflexion point. By inverting the binormal, the continuity of the binormal can be guaranteed.

After discretization and simplification, the expressions for the curvature and the torsion become

$$\begin{aligned} \kappa &= \tfrac{1}{2L} \left\| \mathbf{u_i} \times (\mathbf{u_{i+1}} - \mathbf{u_{i-1}}) \right\| \\ \tau_f &= \tfrac{1}{L^3 \kappa^2} \, \mathbf{u_i}^T (\mathbf{u_{i+1}} \times \mathbf{u_{i-1}}) \end{aligned}$$

with L the length of a segment. This scheme is more stable and less noise-sensitive as the previous one, but it is still insufficient. Both schemes fail to calculate the torsion in a satisfactory way. Consequently, another way of calculating the torsion is needed.

2.5.2 Material torsion

The torsion will be calculated for a segment. While the Frenet torsion only depends on the position of the points of the centerline, more exactly of the orientation of three consecutive segments, the material torsion depends only on the fourth coordinate θ, the material orientation (see 1.4.5). In a continuous version, we have - with s the arc length-

$$\tau_m = \frac{\mathrm{d}\theta}{\mathrm{d}s}$$

which is discretized as

$$\tau_{m,i} = \frac{\theta_{i+1} - \theta_i}{L}$$

2.6 Appendix: Derivation of the forces and Hessian

2.6.1 Curvature

$$E = \frac{k_\kappa}{L^2}(1-\cos\alpha) = \frac{k_\kappa}{L^2}(1-\mathbf{u}_i^T\mathbf{u}_{i+1}) = K(1-\mathbf{u}_i^T\mathbf{u}_{i+1})$$

with $K = \frac{k_\kappa}{L^2}$. For simplifying the notation, we define $\mathbf{u} = \mathbf{u}_i$ and $\mathbf{v} = \mathbf{u}_{i+1}$ We have

$$\mathbf{u} = \frac{\mathbf{\Delta x}}{\|\mathbf{\Delta x}\|}$$

and

$$\mathbf{v} = \frac{\mathbf{\Delta y}}{\|\mathbf{\Delta y}\|}$$

$$\frac{\partial u_j}{\partial \Delta x_k} = \frac{1}{\|\mathbf{\Delta x}\|}(\delta_k^j - u_j u_k)$$

$$\begin{aligned} F_{\Delta x_k} &= -\tfrac{\partial E}{\partial \Delta x_k} \\ &= K\tfrac{\partial u}{\partial \Delta x_k}^T\mathbf{v} \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|}(e_k - u_k\mathbf{u})^T\mathbf{v} \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|}(v_k - u_k\mathbf{u}^T\mathbf{v}) \\ F_{\mathbf{\Delta x}} &= K\tfrac{1}{\|\mathbf{\Delta x}\|}(\mathbf{v} - \mathbf{u}^T\mathbf{v}\mathbf{u}) \end{aligned}$$

For the second derivative, we have

$$\begin{aligned} \tfrac{\partial^2 E}{\partial \Delta y_j \partial \Delta x_k} &= \tfrac{\partial}{\partial \Delta y_j} K\tfrac{1}{\|\mathbf{\Delta x}\|}(v_k - u_k\mathbf{u}^T\mathbf{v}) \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|}\left(\tfrac{1}{\|\mathbf{\Delta y}\|}(\delta_k^j - v_j v_k) - u_k\tfrac{1}{\|\mathbf{\Delta y}\|}(\mathbf{e}_j - v_j\mathbf{v})^T\mathbf{u}\right) \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|\|\mathbf{\Delta y}\|}\left(\delta_k^j - v_j v_k - u_k(\mathbf{e}_j - v_j\mathbf{v})^T\mathbf{u}\right) \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|\|\mathbf{\Delta y}\|}\left(\delta_k^j - v_j v_k - u_k u_j + u_k v_j\mathbf{v}^T\mathbf{u}\right) \\ \mathbf{H}_{\mathbf{\Delta y},\mathbf{\Delta x}} &= K\tfrac{1}{\|\mathbf{\Delta x}\|\|\mathbf{\Delta y}\|}\left(\mathbf{I}_3 - \mathbf{v}\mathbf{v}^T - \mathbf{u}\mathbf{u}^T + \mathbf{u}\mathbf{v}^T\mathbf{v}^T\mathbf{u}\right) \end{aligned}$$

and

$$\begin{aligned} \tfrac{\partial^2 E}{\partial \Delta x_j \partial \Delta x_k} &= \tfrac{\partial}{\partial \Delta x_j} K\tfrac{1}{\|\mathbf{\Delta x}\|}(v_k - u_k\mathbf{u}^T\mathbf{v}) \\ &= K\left(\tfrac{-u_j}{\|\mathbf{\Delta x}\|^2}(v_k - u_k\mathbf{u}^T\mathbf{v}) - \tfrac{1}{\|\mathbf{\Delta x}\|}(\tfrac{1}{\|\mathbf{\Delta x}\|}(\delta_k^j - u_j u_k)\mathbf{u}^T\mathbf{v} + u_k\tfrac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_j - u_j\mathbf{u})^T\mathbf{v})\right) \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|^2}\left(-u_j(v_k - u_k\mathbf{u}^T\mathbf{v}) - ((\delta_k^j - u_j u_k)\mathbf{u}^T\mathbf{v} + u_k(v_j - u_j\mathbf{u}^T\mathbf{v}))\right) \\ &= K\tfrac{1}{\|\mathbf{\Delta x}\|^2}\left(-u_j v_k - u_k v_j + (u_j u_k - \delta_k^j)\mathbf{u}^T\mathbf{v}\right) \\ \mathbf{H}_{\mathbf{\Delta x},\mathbf{\Delta x}} &= K\tfrac{1}{\|\mathbf{\Delta x}\|^2}\left(-\mathbf{u}\mathbf{v}^T - \mathbf{v}\mathbf{u}^T + (\mathbf{u}\mathbf{u}^T - \mathbf{I}_3)\mathbf{u}^T\mathbf{v})\right) \end{aligned}$$

Chapter 3

Generalized Spring Mass Model

3.1 Introduction

Due to the difficulties encountered in the previous section, we are looking for a coordinate system in which the torsion is easily expressed and not too noise-sensitive. As explained before, the torsion depends on the difference between the orientations along the tangent to the centerline of two neighboring segments. Since the approach with only one coordinate for this orientation failed, one possibility is to describe it with more than one coordinate: a logical choice is a representation of $SO(3)$, the group of rotation of $\mathbb{R}^3$: given a reference orientation (an orthonormal base of $\mathbb{R}^3$, that we will take for facilitating the graphical representation), the orientation of a segment can indeed be described as a rotation from this reference orientation. We then need to add for each segment a condition that ensures that the orientation of the segment (as defined by the rotation) connects the two discretization points which enclose it: this will be the role of the coherence energy and force (3.3.5).

From the several representations of rotations of $SO(3)$, we have chosen unit quaternions. (Quaternions are 4-tuples that can be seen as a generalization of complex numbers. They will be described in more detail in section 3.2.5.) Many properties speak in favor of unit quaternions: they only have 4 coordinates and one constraint (they must have a unit length), which is an advantage compared to rotation matrices (with 9 components and 6 constraints). Furthermore, the rotation of a vector is easily expressed and the composition of two rotations can also be easily calculated as the product of the two corresponding quaternions. This group structure is an advantage for the simulation. Quaternions also lack singular points and gimbal lock, contrarily to Euler or Cardan angles. So we used a system with seven coor-

dinates for each point: the three usual space coordinates and a quaternion which represents the orientation of a segment between two points.

The coordinates vector $\mathbf{X}$ has then in total $7n - 4$ components (the last quaternion is missing). We can classify the different interactions according to the type of coordinates on which they apply and their model internal or external (coming directly from the physical laws) nature (Table 3.1). All these interactions will be explained more in details in section 3.3. For an efficient numerical solution of the system, we will need to calculate also the forces and their derivatives.

	Positions only	Orientations only	Positions and orientations
Model internal	Length conservation	Quaternion norm	Coherence
Model external	Weight	Bending and torsion	

Table 3.1: Summary of the different interactions

3.2 Representation of rotations

We will first review various usual representations of rotation of $SO(3)$. Any rotation $Rot \in SO(3)$ is defined by its axis $\mathbf{v} \in \mathbb{R}^3$ (with $\|\mathbf{v}\| = 1$) and its angle θ. From the Euler rotation theorem, we know that any rotation of $SO(3)$ can be represented with 3 parameters. However, most of the representations use more parameters (adding constraints between these parameters) but allow an easier formulation of the usual operations.

3.2.1 Axis-angle representation and exponential map

It is the simplest representation: the rotation is simply represented by $(\mathbf{v}, \theta) \in \mathbb{R}^4$ (axis-angle representation) or $\theta\mathbf{v} \in \mathbb{R}^3$. It is used for example in medical applications for describing the motion of articulations. Its main advantages are that limitation constraints are easily expressed and that rotations which differ in an 2π angle can be distinguished. The disadvantages are that there are no simple formula for composing two rotations or calculating the image of a vector by the rotation. The Rodrigues Formula gives $Rot(\mathbf{x}) = \mathbf{x} + \sin\theta\mathbf{v} \times \mathbf{x} + \cos\theta\mathbf{v} \times (\mathbf{v} \times \mathbf{x})$

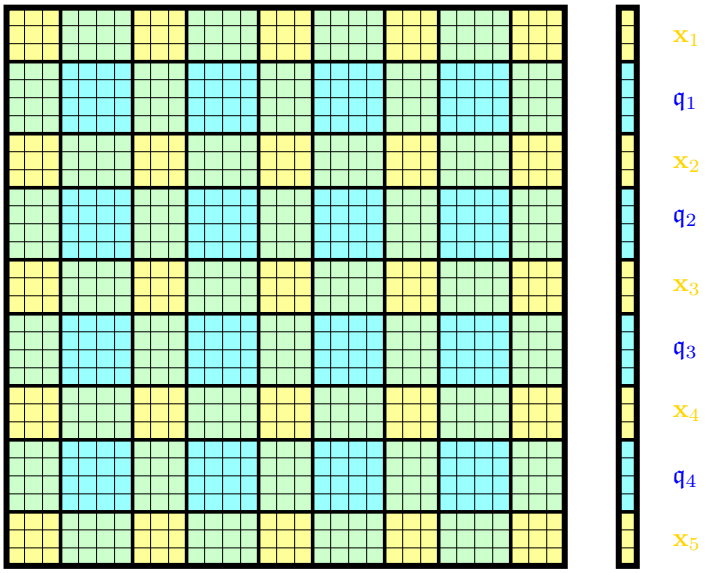

Figure 3.1: Repartition of the different kinds of coordinates, represented here for $n = 5$: the yellow color represents the positions, the blue color the orientations, and in case of the Hessian, the green color represents the mixed terms. The vector on the right show the structure common to the variable $\mathbf{X}$ and the force $\mathbf{F}$: an alternation of positions and orientations. The orientations correspond to the segment joining the two points surrounding it. The structure of the Hessian, on the left, is a checker pattern.

3.2.2 Rotation matrix

A rotation matrix $\mathbf{M} \in \mathbb{R}^{3\times3}$ is an orthogonal matrix $\mathbf{M}^T\mathbf{M} = \mathbf{I}_3$ with a determinant equal to 1. (The orthogonality condition implies that $|det(M)| = 1$. The case of a determinant of -1 corresponds to a rotation combined with a symmetry.) It is however usually not necessary to explicitly enforce the determinant condition due to the continuity of the parameters. The image of a vector $\mathbf{x} \in \mathbb{R}^3$ is simply $Rot(\mathbf{x}) = \mathbf{M}\mathbf{x}$, the composition of two rotations is $Rot_{M_2} \circ Rot_{M_1} = Rot_{M_2M_1}$. Rotation matrices are used for example in computer graphics. With $\mathbf{v}^X \in \mathbb{R}^{3\times3}$ a matrix such that for any vector $\mathbf{y} \in \mathbb{R}^3$, $\mathbf{v} \times \mathbf{y} = \mathbf{v}^X\mathbf{y}$, the rotation matrix can be expressed as $\mathbf{M} = \mathbf{I}_3 + \sin\theta\mathbf{v}^X + \cos\theta\mathbf{v}^X\mathbf{v}^X$

3.2.3 Euler angles

Euler angles use three angles for describing a rotation: pitch, yaw and roll. Each of these 3 angles corresponds to a rotation along a particular axis. Let's start with $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ the canonical Cartesian basis of $\mathbb{R}^3$. A first rotation of angle ψ - the pitch - around $\mathbf{e}_3$ leads us to a new orthonormal basis $(\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3)$ with $\mathbf{f}_3 = \mathbf{e}_3$. A second rotation of angle θ - the yaw - around $\mathbf{f}_1$ gives another orthonormal basis $(\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3)$ with $\mathbf{g}_1 = \mathbf{f}_1$. The last rotation of an angle ϕ - the roll - around $\mathbf{g}_3$ takes us to the final orthonormal basis $(\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3)$ with $\mathbf{h}_3 = \mathbf{g}_3$. (Figure 3.2) The three angles usually have a limited range.

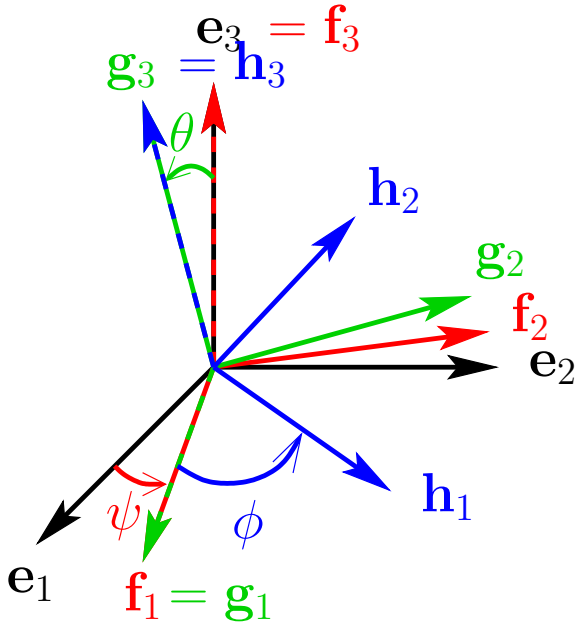

Figure 3.2: Euler angles

The gimbal lock is a problem: when two of the tree rotation axes align, a rotational degree of freedom is lost: the parameters can be varied without changing the rotation (for example in the case $\theta = 0$ the rotation is determined by the sum $\theta + \phi$) and the system cannot react to applied torques.

3.2.4 Cardan Angles

Cardan angles (or Tait-Bryan angles) are similar to Euler angles. The main difference is that the successive rotation axes are axes of the reference Cartesian coordinate frame. They have similar problems to Euler angles.

3.2.5 Unit quaternions

Quaternions are 4-tuples. The four components of a quaternion $\mathfrak{q}$ will be noted q_0, q_1, q_2, q_3 (with $q_i \in \mathbb{R}$). The vectorial part of a quaternion will be noted $\mathbf{q} = (q_1, q_2, q_3)$:

$$\mathfrak{q} = (q_0, q_1, q_2, q_3) = (q_0, \mathbf{q})$$

(Please note that quaternions are sometimes noted in the reverse order $(\mathbf{q}, q_0)$.) Unit quaternions (such that $\|\mathfrak{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1$) form a representation of $SO(3)$: each unit quaternion can indeed be written in the form

$$\mathfrak{q} = (q_0, \mathbf{q}) = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{v}\right)$$

representing the rotation of angle θ around the axis $\mathbf{v}$. $\mathfrak{q}$ and $-\mathfrak{q} = (-q_0, -q_1, -q_2, -q_3)$ represent the same rotation, since a rotation of an angle $2\pi - \theta$ around $-\mathbf{v}$ is equivalent to a rotation of θ around $\mathbf{v}$.

The quaternion product is defined as

$$\mathfrak{p} \cdot \mathfrak{q} = (p_0 q_0 - \mathbf{p}^T\mathbf{q}, p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p} \times \mathbf{q})$$

With this product, the representation of the composition of two rotations represented by two quaternions $\mathfrak{p}$ and $\mathfrak{q}$ is represented by $\mathfrak{p} \cdot \mathfrak{q}$. The image $\mathbf{b} \in \mathbb{R}^3$ of a vector $\mathbf{a} \in \mathbb{R}^3$ by a rotation represented by a unit quaternion $\mathfrak{q}$ can be calculated by:

$$(0, \mathbf{b}) = \mathfrak{q} \cdot (0, \mathbf{a}) \cdot \overline{\mathfrak{q}}$$

where $\overline{\mathfrak{q}} = (q_0, -\mathbf{q})$ is the conjugated of $\mathfrak{q}$. The conjugated $\overline{\mathfrak{q}}$ represents the inverse rotation from $\mathfrak{q}$ (a rotation of angle $-\theta$ around the axis $\mathbf{v}$). Unit quaternions have all the desirable properties for what we need: we will use them as a representation of rotation in $SO(3)$.

3.3 Energy and Forces

For each kind of interaction, we first define an energy function and then derive the forces from this function, following the common practice in mechanics:

$$F_i = -\frac{\partial E}{\partial X_i}$$

We also define the symmetric Hessian matrix $\mathbf{H} \in \mathbb{R}^{(7n-4)\times(7n-4)}$, such that

$$H_{i,j} = \frac{\partial^2 E}{\partial X_i \partial X_j} = -\frac{\partial F_i}{\partial X_j}$$

Representation of rotation	Number of parameters	Number of constraints	Constraints	Rotation of a vector	Inverse of a rotation	Composition of two rotations
Axis-Angle	3 or 4	0 or 1	$\|v\|=1$	Not direct	$(v,-\theta)$	Not direct
Euler Angles	3	0		Not direct	Not direct	Not direct
Cardan Angles	3	0		Not direct	Not direct	Not direct
Rotation Matrix	9	6	$\mathbf{M}^T\mathbf{M}=I_3$	$\mathbf{Mx}$	$\mathbf{M}^T$	$\mathbf{M_2M_1}$
Quaternion	4	1	$\|\mathfrak{q}\|=1$	$\mathfrak{q}\cdot(0,\mathbf{x})\cdot\bar{\mathfrak{q}}$	$\bar{\mathfrak{q}}$	$\mathfrak{q}_2\cdot\mathfrak{q}_1$

Table 3.2: Summary of the different representations of rotation and their properties. Both rotation matrices and quaternions have an easy formulation of the main operations, but the quaternions have really fewer parameters and their only constraint is easier to enforce.

Since each type of interaction can be decomposed as a sum of interactions between either two points, two quaternions or two points and a quaternion, we calculate the energy, the forces and the Hessian as a sum over element groups. In order to enhance the calculation speed, it is important to calculate only once the partial terms that appear several times in the three functions.

For a global position $\mathbf{X}$, we can formulate the energy as the sum of the energies over all points and segments and over all interactions.

$$\begin{aligned} E &= \sum_{Interactions} \sum_{i} E_{Interaction,\,i} \\ &= \sum_{i=1}^{n-1} E_{Length,i} + \sum_{i=1}^{n-1} E_{QuatNorm,i} \\ &\quad + \sum_{i=1}^{n} E_{Weight,i} + \sum_{i=1}^{n-1} E_{Coh,i} \\ &\quad + \sum_{i=1}^{n-2} E_{Bending,i} + \sum_{i=1}^{n-2} E_{Torsion,i} \end{aligned}$$

The forces and the Hessian can be calculated in a similar way. In the following, $k_{Interaction}$ is the constant relative to the interaction $Interaction$ that will be used for defining the relative weights of the different interactions.

3.3.1 Weight

The weight has an important influence on the form of sagging cables since it determines the direction of the deformation of the cable. The longer the cable is and the smaller the ratio of the distance between the two end points and the length of the cable is, the greater is the influence of the weight. Furthermore, in particular undetermined cases (like a cable in which the two end tangents are collinear with the line joining the two end points) where there would be an infinite number of solutions without gravity, it allows to eliminate the ambiguity. We consider the cable as a chain of mass points with equal masses m. The total mass of the cable is n m. The energy is $E_{Weight,i} = -m\mathbf{g}^T\mathbf{x_i}$ where $\mathbf{g} \in \mathbb{R}^3$ is the acceleration of gravity. The force is then $\mathbf{F}_{Weight,i} = m\mathbf{g}$ (Figure 3.3). Since the force does not depend on any of the coordinates, the weight has no contribution to the Hessian.

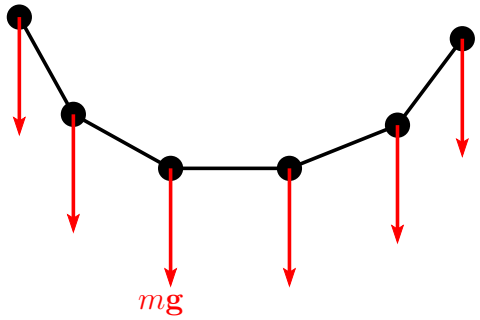

Figure 3.3: Force due to the weight

3.3.2 Length Conservation

Cables are, in praxis, unstretchable: the length of each segment has to be kept constant. We use for this purpose strong linear springs. The energy is defined classically as (see page 30)

$$E_{Length,\,i} = \frac{1}{2}k_{Length}(L_i - L_{ref})^2$$

where k_{Length} is the constant of the spring. The force on the point $\mathbf{x}_i$, just opposed to the force on point $\mathbf{x}_{i+1}$ (Figure 2.2), is

$$\mathbf{F}_{Length,\,i} = k_{Length}\mathbf{u}_i(L_i - L_{ref})$$

and the Hessian has for each segment for blocks of $\mathbb{R}^{3\times3}$: two blocks on the diagonal corresponding to the pairs of points $(\mathbf{x}_i, \mathbf{x}_i)$ and $(\mathbf{x}_{i+1}, \mathbf{x}_{i+1})$, and two

blocks with the opposite sign corresponding to the pairs of points $(\mathbf{x}_i, \mathbf{x}_{i+1})$ and $(\mathbf{x}_{i+1}, \mathbf{x}_i)$:

$$\mathcal{H}_{Length,i} = k_{Length}(\mathbf{I_3} + \frac{L_{ref}}{L_i}(\mathbf{u}_i\mathbf{u}_i^T - \mathbf{I_3}))$$

It is obvious that $\mathbf{u}_i$ is an eigenvector of $\mathcal{H}_{Length,i}$ with a corresponding eigenvalue of 1, and that vectors orthogonal to $\mathbf{u}_i$ are eigenvectors with an eigenvalue of $(1 - \frac{L_{ref}}{L_i})$. In the case where $L_{ref} > L_i$, the block has negative eigenvalues.

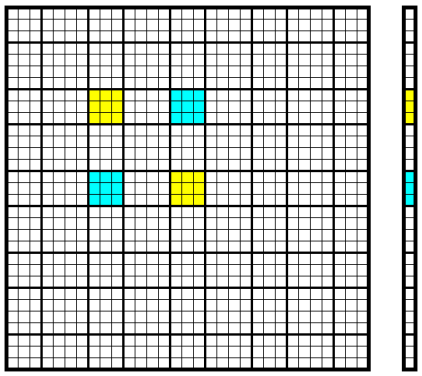

Figure 3.4: Occupation of the Hessian by the terms due to the conservation of the length of segment i: the blue and yellow blocks only differ by a minus sign.

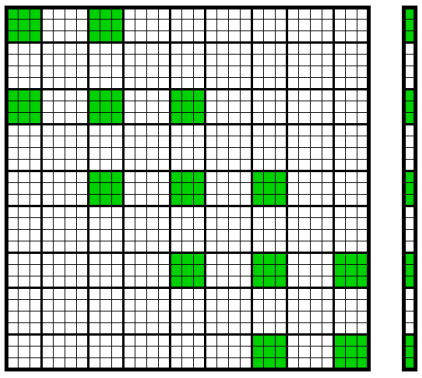

Figure 3.5: Occupation of the Hessian by the terms due to the conservation of the length.

3.3.3 Bending and torsion

The bending and torsion forces are calculated jointly as a function of the relative rotation between two segments. They are determined by the two consecutive quaternions $\mathfrak{q}_i$ and $\mathfrak{q}_{i+1}$. (Observe that the previous formulations needed three segments and that the Frenet torsion is not needed anymore.) The relative rotation from segment i to segment $i+1$ is represented by

$$\mathfrak{q}_{i\to i+1} = \mathfrak{q}_{i+1}\overline{\mathfrak{q}_i}.$$

(The inverse of a quaternion $\mathfrak{q}$ is $\frac{\overline{\mathfrak{q}}}{\|\mathfrak{q}\|}$ and for unitary quaternions simply $\overline{\mathfrak{q}}$.).

The Darboux vector is, in global coordinates, for a differentiable function $\mathfrak{q}(s)$ and where the dot represents the derivation with respect to the arclength s: $\dot{\mathfrak{q}} = \frac{\partial \mathfrak{q}}{\partial s}$,

$$(0, \omega_g) = 2\dot{\mathfrak{q}} \cdot \overline{\mathfrak{q}}$$

and in local coordinates

$$(0, \omega_l) = 2\overline{\mathfrak{q}} \cdot \dot{\mathfrak{q}}$$

They can be discretized as (see the Appendix)

$$\omega_g = \frac{2}{L}\overrightarrow{\mathfrak{q}_{i+1} \cdot \overline{\mathfrak{q}}_i}$$

and

$$\omega_l = \frac{2}{L}\overrightarrow{\overline{\mathfrak{q}}_i \cdot \mathfrak{q}_{i+1}}$$

where, for a quaternion $\mathfrak{q} = (q_0, q_1, q_2, q_3)$, $\overrightarrow{\mathfrak{q}} = \mathbf{q} = (q_1, q_2, q_3)$ represents its vectorial part.

On the other hand, the general properties of the Darboux vector give

$$\omega = \kappa\mathbf{B} + \tau\mathbf{T}.$$

The tangent vector $\mathbf{T}$ can be defined either interpolating the positions as

$$\mathbf{T}_i = \frac{\mathbf{u_{i-1}} + \mathbf{u_i}}{\|\mathbf{u_{i-1}} + \mathbf{u_i}\|}.$$

or interpolating the orientations as the image of the reference orientation **ref** by the rotation represented by

$$\mathfrak{q}_T = \frac{\mathfrak{q}_i + \mathfrak{q}_{i+1}}{\|\mathfrak{q}_i + \mathfrak{q}_{i+1}\|}$$

(The two definitions differ very slightly: the first one correspondsto a planar, circular path and the second one to an spiral one. In the cases of a pure torsion or a pure curvature they result in the same expression.).

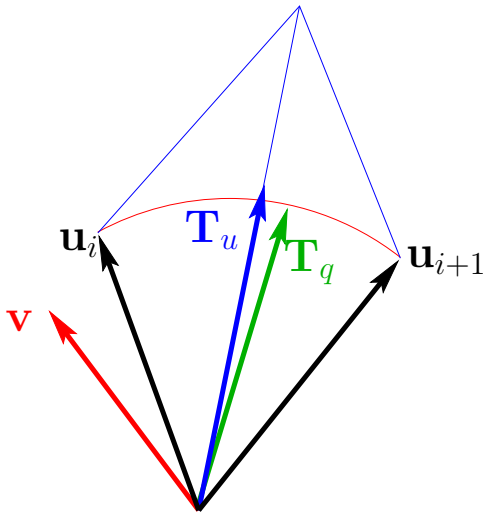

Figure 3.6: Comparison of the two definitions of the tangent $\mathbf{T}$: $\mathbf{T}_u$ is defined with the interpolation of the positions. It lies on the plane defined by $\mathbf{u_i}$ and $\mathbf{u_{i+1}}$. $\mathbf{T}_q$ is defined with the interpolation of the orientations. It lies on the plane defined by the axis of rotation $\mathbf{v}$ and $\mathbf{T}_u$ but is not in general in the plane defined by $\mathbf{u_i}$ and $\mathbf{u_{i+1}}$.

We can now calculate the curvature and the torsion: a decomposition of ω on the basis formed by $\mathbf{T}$ and $\mathbf{B}$ (which is defined as the direction of the residual component $\omega - \tau\mathbf{T} = \kappa\mathbf{B}$) gives us:

$$\tau = \omega_g^T \mathbf{T}$$

and

$$\kappa = \|\omega - \tau\mathbf{T}\| .$$

Considering the expression in the local base leads to simplifications: $\mathbf{T}$ is the image of the rotation of the reference direction **ref**. The local base is also the image of the same rotation of the canonical base of $\mathbb{R}^3$. Hence $\mathbf{T}$ is the direction **ref** in the local base.

$$\tau = \omega^T \mathbf{T} = \omega_l^T \mathbf{ref}$$

In the particular case of $\mathbf{ref} = (0, 1, 0)$,

$$\omega_l = (\kappa_1, \tau, \kappa_2)$$

where $\kappa^2 = \kappa_1^2 + \kappa_2^2$. Coupled with the discretization $\omega_l = \frac{2}{L}\overrightarrow{\bar{\mathfrak{q}}_i \cdot \mathfrak{q}_{i+1}}$, we now have an easy way to calculate the bending and torsion energy:

$$E_{BendingTorsion,i} = E_{Bending,i} + E_{torsion,i} = \frac{1}{2} B_\kappa \kappa^2 + \frac{1}{2} B_\tau \tau^2$$

which can be rewritten as

$$\begin{aligned} E_{BendingTorsion,i} &= \tfrac{1}{2}\left(B_\kappa \omega^T \omega + (B_\tau - B_\kappa)(\omega^T \mathbf{T})^2\right) \\ &= \tfrac{1}{2}\,\omega^T (B_\kappa \mathbf{I}_3 + (B_\tau - B_\kappa)\mathbf{T}\mathbf{T}^T)\omega \end{aligned}$$

The matrix $(B_\kappa \mathbf{I}_3 + (B_\tau - B_\kappa)\mathbf{T}\mathbf{T}^T)$ has the eigenvalues B_τ (for vectors collinear with $\mathbf{T}$) and B_κ (for vectors orthogonal to $\mathbf{T}$) which shows that the bending and torsion energy is nothing else than the square of a particular (local) norm of ω as long as the two constants are positive.

A possibility would be to use only the quaternions as coordinates and calculate iteratively the positions. The length conservation and coherence forces would then disappear. The disadvantage is that the system matrix would be a full matrix (the position of point i depends on all the quaternions between 1 and $i-1$), which makes the calculation much slower. External forces like contact forces would also be difficult to implement.

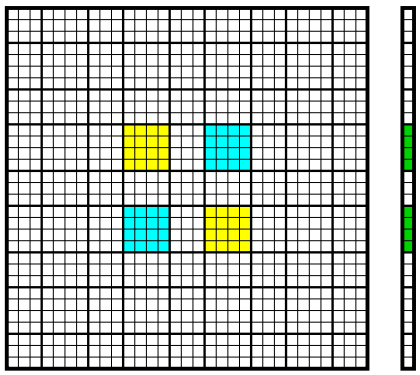

Figure 3.7: Occupation of the Hessian by the terms due to the bending and torsion at vertex i: the blue and yellow blocks only differ by a minus sign.

In the case where the rest state has a non-zero curvature or torsion (which is the case, for example, for molded hoses), the energy becomes

$$E_{BendingTorsion,i} = \frac{1}{2}\,B_\kappa\,(\kappa_1 - \kappa_{1,0})^2 + \frac{1}{2}\,B_\kappa\,(\kappa_2 - \kappa_{2,0})^2 + \frac{1}{2}\,B_\tau\,(\tau - \tau_0)^2$$

where $\kappa_{1,0}$, $\kappa_{2,0}$, τ_0 are the values of the components of the curvature and of the torsion in the rest state. The form of the cable can be greatly influenced by this rest state deformation. Some hoses also show a rest curvature due for example to a storage on a roll. It is a cause for the variability between hoses.

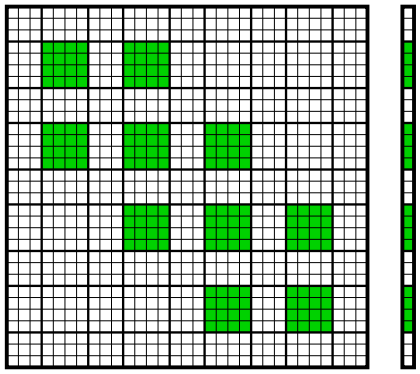

Figure 3.8: Occupation of the Hessian by the terms due to the bending and torsion.

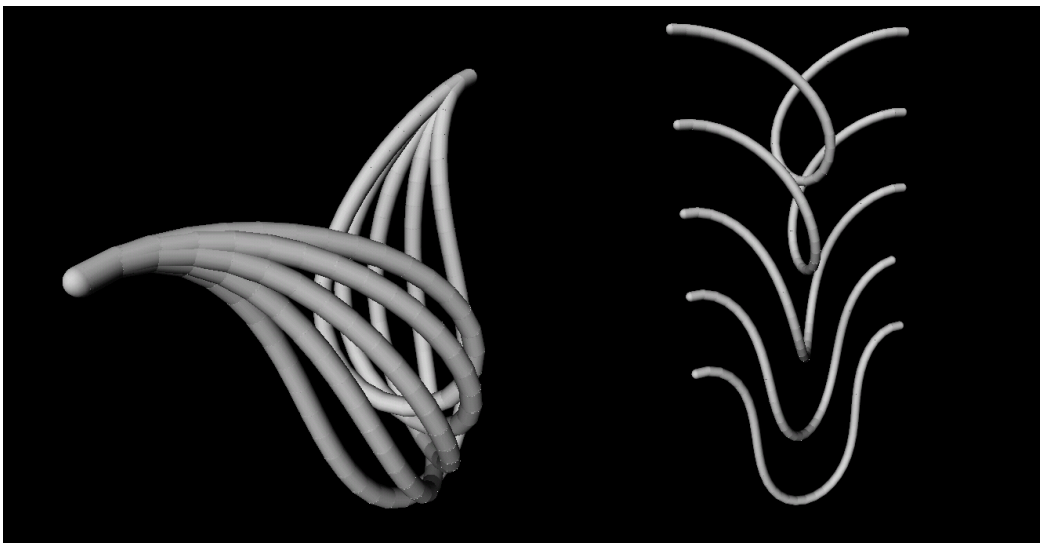

Figure 3.9: Influence of the torsion : the same cable is submitted to different values of the torsion by rotating one of its extremities

For cables which do not have a circular cross-section or whose material repartition is not symmetric, we can simply take the energy to be of the form

$$E_{BendingTorsion,i} = \frac{1}{2} B_{\kappa 1} \kappa_1^2 + \frac{1}{2} B_{\kappa 2} \kappa_2^2 + \frac{1}{2} B_\tau \tau^2$$

where $B_{\kappa 1}$ and $B_{\kappa 2}$ are the bending constants around the two axes corresponding to the two directors $\mathbf{d}_1$ and $\mathbf{d}_2$ (assuming that they are principal axes).

3.3.4 Quaternion norm

Since quaternions represent a pure rotation only when they have a unit norm - otherwise a scaling effect is introduced and a very small curvature and torsion could be obtained - , the norm of the quaternions must be maintained unitary. The energy is simply defined as

$$E_{QuatNorm,i} = \frac{1}{2} k_{QuatNorm} \left(\|\mathfrak{q}_\mathfrak{i}\| - 1 \right)^2$$

for $i \in 1...n-1$ with $\|\mathfrak{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$. This term only depends on the quaternion $\mathfrak{q}_i$. The force on the quaternion $\mathfrak{q}_i$ is

$$F = k_{QuatNorm}\ \mathfrak{q}_\mathfrak{i}\ \left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|} - 1 \right)$$

and the block of the Hessian corresponding to ($\mathfrak{q}_i$,$\mathfrak{q}_i$) is

$$H = k_{QuatNorm} \left(\left(\frac{1}{\|\mathfrak{q}\|} - 1 \right) \mathbf{I} - \frac{\mathfrak{q}\mathfrak{q}^T}{\|\mathfrak{q}\|^3} \right)$$

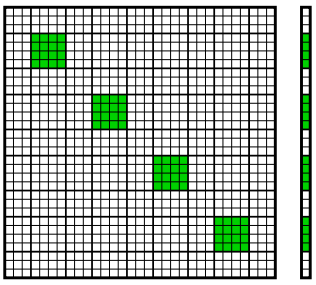

Figure 3.10: Occupation of the Hessian by the terms due to the norm of quaternions

This term is normally easy to enforce and does not pose any difficulty. However, in the case of a cable with very high bending and torsion constants, it can happen that the cable takes a "V" shape with a very sharp bend: the norm of a particular (or two) quaternion(s) collapse to almost 0. The interactions affected by the norm of a quaternion are the bending and torsion (κ and τ are proportional to the norm of ω and consequently to the norm of each of the concerned quaternions, see section 3.3.3) and the coherence (the norm of the "rotation" of the reference direction is proportional to the square of the norm of the quaternion, see section 3.3.5). Whereas the coherence term is limited (the norm of the vector $\mathbf{u}_i - Rot(\mathbf{ref})$ cannot be greater than 2 in general and practically than 1 in this particular case), all the actual bending of the cable is absorbed by this particular point and the global bending and torsion energy tend to 0.

The solution is then to change the formulation of the energy as a function of the inverse of the norm:

$$E_{QuatNorm,i} = \frac{1}{2} k_{QuatNorm} \left(\frac{1}{\|\mathfrak{q}_i\|} - 1 \right)^2$$

If the norm tends to 0, the energy will then tend to the infinite, thus canceling all global energy diminution.(Figure 3.11)

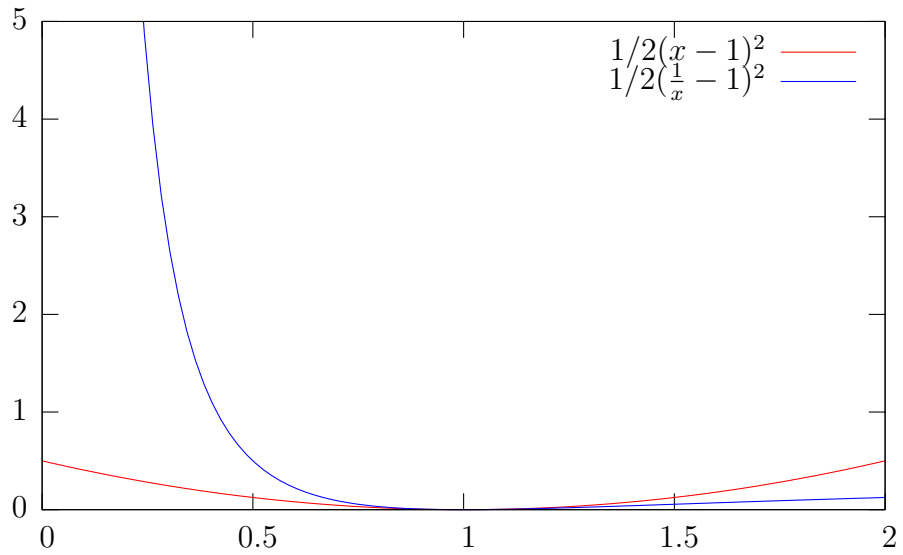

Figure 3.11: Comparison of the values of $\frac{1}{2}(\|\mathfrak{q}\| - 1)^2$ and $\frac{1}{2}(\frac{1}{\|\mathfrak{q}\|} - 1)^2$

If desired, this definition could be taken for the norm smaller than 1 and the old one for the norm greater than one. The resulting function is $\mathcal{C}^2$

(the function and its first two derivatives are continuous in all points and in particular where $\|\mathfrak{q}_\mathbf{i}\| = 1$), which means that there are no problems for the numerical solution and that the behavior of the system should not change significantly near the equilibrium point $\|\mathfrak{q}\| = 1$. However this may not be necessary since a norm greater than 1 augments both the coherence and the bending and torsion energy.

3.3.5 Coherence between quaternions and positions

For the correct functioning of the model, it is important to ensure that the quaternions (and thus the bending and torsion forces and torques) are correctly coupled to the positions. The unit vector $\mathbf{u}_i = \frac{\mathbf{x}_{i+1}-\mathbf{x}_i}{\|\mathbf{x}_{i+1}-\mathbf{x}_i\|} \in \mathbb{R}^3$ in the direction between the two points i and $i+1$ should be equal to the reference direction **ref** rotated by the quaternion $\mathfrak{q}_i$. For a constant **ref** $= (0, 1, 0)$, the image can be directly calculated (see 3.2.5)

$$Rot(\mathbf{ref}) = \big(2(q_1q_2 - q_0q_3), q_0^2 - q_1^2 + q_2^2 - q_3^2, 2(q_0q_1 + q_2q_3)\big)$$

The energy is defined as

$$E_{Coh,i} = \frac{1}{2}k_{Coh}\|(0, \mathbf{u_i}) - \mathfrak{q}_\mathbf{i} \cdot (0, \mathbf{ref}) \cdot \overline{\mathfrak{q}_\mathbf{i}}\|^2$$

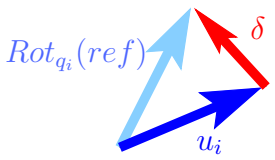

Figure 3.12: The coherence energy depend on the difference δ between the unitary vector $\mathbf{u}_i$ and the rotation of the reference direction represented by $\mathfrak{q}_i$

3.3.6 Handles

Each handle (fixed point) is attached either to a point or to a segment. The two possibilities are offered for an easy interface with the graphical representation: the cable is represented as a sequence of spheres (at the mass points) and cylinders (the segments). The user can attach the handle either to a sphere or a cylinder. The handle can fix either only the position or both

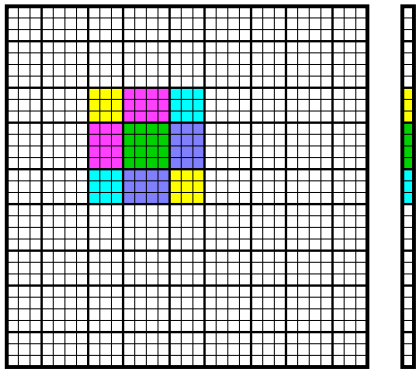

Figure 3.13: Occupation of the Hessian by the terms due to the coherence of positions and orientation for segment i: the cyan and yellow blocks, as well as the magenta and blue ones, only differ by a minus sign.

the position and orientation at a point. In the case of a sphere, the orientation is taken as the (normed) mean value of the two quaternions surrounding it; in the case of a cylinder, the position is the mean value of the positions of the two surrounding points. If a handle is associated with the point $\mathbf{x_i}$ and with the quaternion $\mathfrak{q}_\mathbf{i}$, the energy is

$$E_{HandleSphere,\,i} = k_{Handle}((\mathbf{x_i} - \mathbf{x}_{Handle})^2 + (\frac{\mathfrak{q}_{i-1} + \mathfrak{q}_i}{\|\mathfrak{q}_{i-1} + \mathfrak{q}_i\|} - \mathfrak{q}_{Handle})^2)$$

or

$$E_{HandleCylinder,\,i} = k_{Handle}(\frac{\mathbf{x_i} + \mathbf{x_{i+1}}}{2} - \mathbf{x}_{Handle})^2 + (\mathfrak{q}_\mathbf{i} - \mathfrak{q}_\mathbf{Handle})^2)$$

Handles on spheres tend to have the problem that they "bend", especially with low bending constants: the mean value of the two quaternion at this point is conformed with the desired value for the handles, but the two quaternions have very different values, leading to a sharp bend in the cable. To remedy this problem, we replace the energy by

$$E_{HandleSphere,\,i} = k_{Handle}((\mathbf{x_i} - \mathbf{x}_{Handle})^2 + (\frac{\mathfrak{q}_{i-1} + \mathfrak{q}_i}{2} - \mathfrak{q}_{Handle})^2)$$

Due to the absence of normalization, the norm of the "mean quaternion" is $\|\frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{2}\| = \cos\frac{\theta}{2}$where θ is the angle between the two quaternions in quaternion space. Since $cos\theta = \mathfrak{q}_{i-1}^T\mathfrak{q}_i = q_{i-1,0}q_{i,0} + \mathbf{q}_{i-1}^T\mathbf{q}_i = (\mathfrak{q}_i \cdot \overline{\mathfrak{q}}_{i-1})_0$, θ is also the angle of the relative rotation between them as the previous calculation shows. The norm of the difference $\frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{2} - \mathfrak{q}_{Handle}$ is then at minimum $1 - \cos\frac{\theta}{2}$. In the case of a sharp bend, this term augments the energy and counteracts the effect.

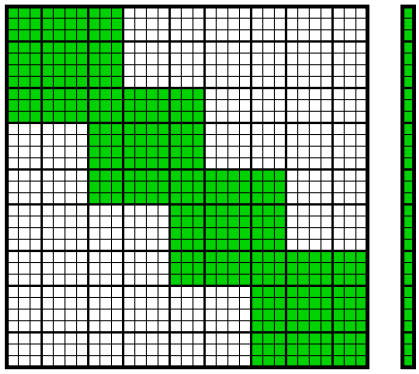

Figure 3.14: Occupation of the Hessian by the terms due to the bending and torsion.

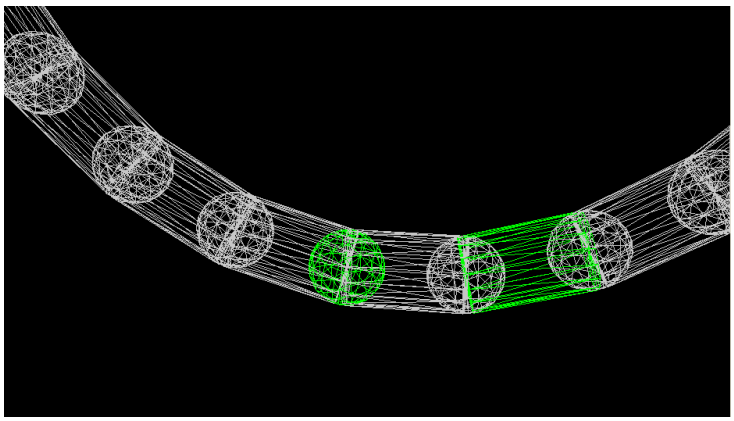

Figure 3.15: The graphical representation of a cable, here in wireframe mode. The spheres correspond to the mass points and the cylinder to the segments.

3.3.7 Special mode for cable routing

Cable routing is the design of the cable form. Normally, the length of a cable is not known beforehand. It is then interesting to have a mechanism to adapt interactively the length of the cable to the desired form. There are two possibilities to do this.

The first one is to diminish the length conservation constant. Since the length constraint is not enforced exactly, the length of each segment varies slightly. If after each iteration of the algorithm the reference length of each segment is updated with its actual value, the cable will adapt progressively to its external constraints.

The second method is to set the reference length for each segment to 0,

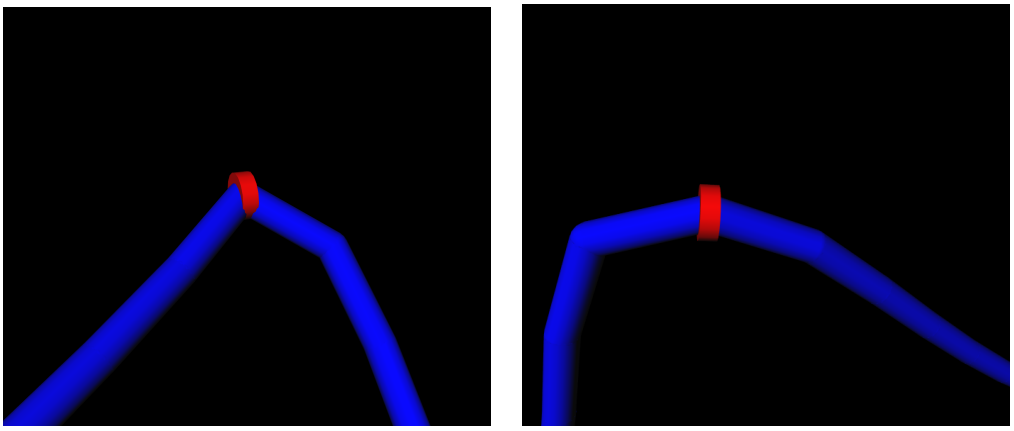

Figure 3.16: Handle: Effect of change in the definition of the energy: on the left, with the normalized mean value; on the right without normalization

although with a smaller constant. The cable will then tend to become smaller. The length conservation constant should not be to small: the weight would stretch the cable. In both cases, the speed of change is controlled by the length constant, the length itself by the bending constant and the weight. It is also important to take into account the different lengths of the segments: for the bending and torsion, the parameter L should be replaced by the mean $\frac{L_i+l_{i+1}}{2}$ of the two segment lengths.

3.3.8 Banded Structure of the Hessian

In the previous sections, we have considered the interactions independently. However, the interplay of all the interactions is decisive for the form of the cable. The sparse pattern of the Hessian matrix is important for the numerical solution of the system. One of the advantages of our formulation is that the Hessian matrix has a banded structure.

3.4 Torsion with angles greater than 4π

The previous scheme works very well, but has a weak point: the quaternions can only represent rotations up to a difference of 4π. Using only quaternions (or any other of the usual rotation representations apart from the axis-angle one), there is no way to distinguish between an orientation and the same orientation composed with a rotation of 2π. In the case of the quaternions, the double cover provides the way of distinguishing the two: a difference

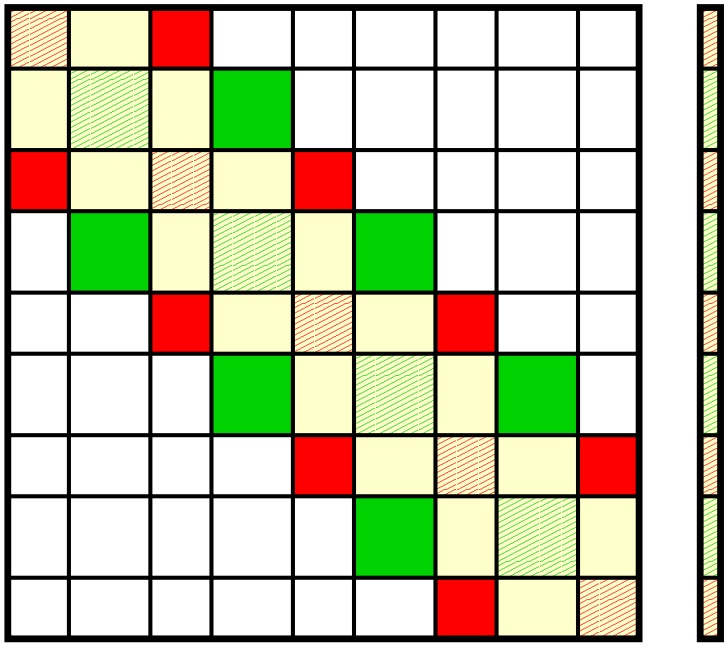

Figure 3.17: Occupation of the Hessian:the yellow color represents the terms due to the coherence, the red the terms for the length conservation and the green the bending, torsion and quaternion norm.

of 2π will give the opposite quaternion.[1] For most of the applications of rotations, this does not matter. If we are however interested in the history of the rotation, it is a problem: a cable whose end orientation will have been rotated by 2π will have a different form - more twisted - than one which have not: it needs to make one more turn. (See Figure 3.19)

It means that, for example, if we rotate one end of the cable around its tangent - the other end being fixed - there will be a moment where the configuration with the same orientation but with a turn less will have a smaller energy than the configuration we actually want, and the system will jump to this configuration instead of getting more and more twisted. (See Figure 3.20)

One way of bypassing the problem would be to predict the value of the torsion and to add an energy of the form $\frac{1}{2}k(\tau - \tau_{expected})^2$. The problem is that the torsion is not easy to calculate without solving the system: it depends not only from the relative rotation of the end segments but also

[1]There is no explicit difference between $\mathfrak{q}$ and $-\mathfrak{q}$ for calculating the energy, but since the quaternions do not normally jump to their opposite, the continuity allows to make the difference.

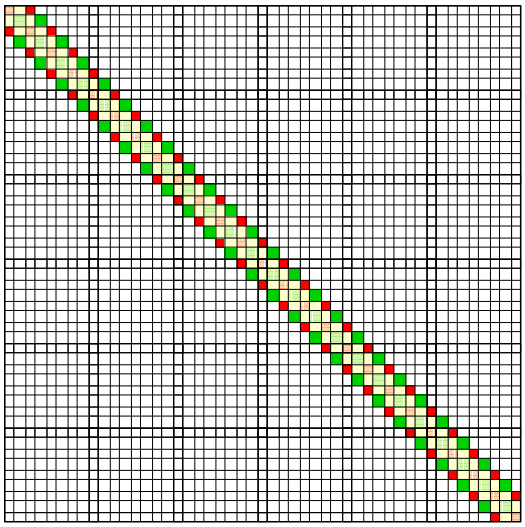

Figure 3.18: Occupation of the Hessian: the banded structure of the Hessian is clearly visible:

from the constants k_κ and k_τ, the relative positions of the endpoints.

We need to keep track of the numbers of turns to avoid this problem. The cable is represented as a sequence of quaternions $\mathfrak{q}_i$ for $i = 1 \dots (n-1)$. The first idea is to associate an angle θ_i to each one of these quaternions such that

$$\cos \frac{\theta_i}{2} = q_{i,0}$$

The value of θ_i has no limits: $\frac{\theta_i}{2\pi}$ is linked to the "number of turns" that were made. A new energy term impedes that the differences $\theta_i - \theta_{i-1}$ become to big.

The first difficulty with this is that the values of $q_{i,0}$ are often restricted to an interval smaller than $[-1:1]$, thus impeding the passage to a different winding number. Let us take as an example a cylinder oriented by the quaternion $(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, 0)$ that is rotated around its axis $(0, 0, 1)$ of an angle ϕ. The resulting quaternion value is $q_{i,0} = \frac{\sqrt{2}}{2} \cos \frac{\phi}{2}$ and $\arccos q_{i,0} \in [\frac{\pi}{4} : \frac{3\pi}{4}]$

To remedy this problem, let us consider instead the angle of the relative rotation from the first quaternion of the chain to the quaternion i. We define

$$\mathfrak{p}_i = \mathfrak{q}_i \overline{\mathfrak{q}_0}$$

and θ_i has the property that

$$\cos \frac{\theta_i}{2} = p_{i,0}$$

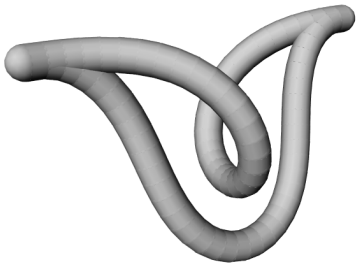

Figure 3.19: Two cables with the same parameters, but with a difference of rotation of 2π

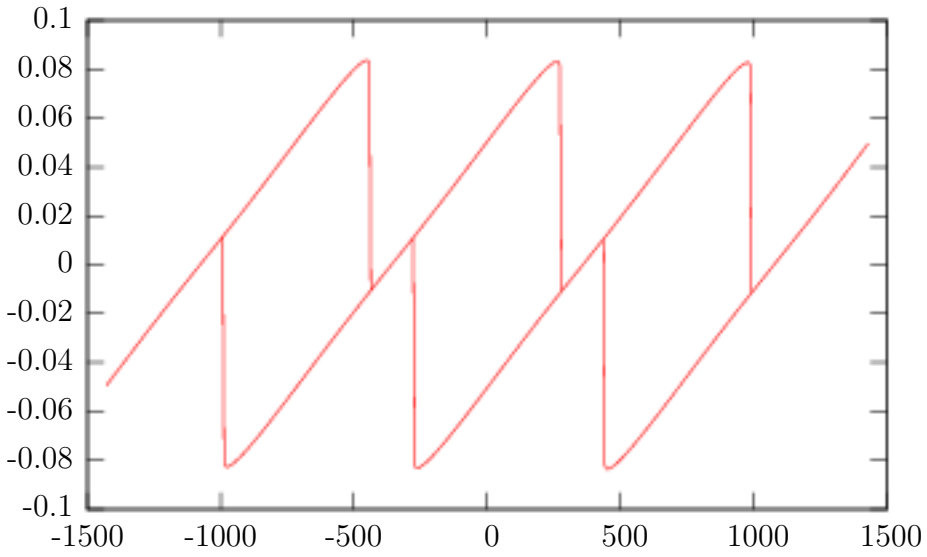

Figure 3.20: A typical hysteresis curve for the torsion. One of the end segments of a cable was rotated around its tangent (all others parameters remain constant). The torsion of the cable as a function of this angle (in degrees) shows a typical hysteresis pattern: note that the jumps correspond to a modification of the angle of 4π (720°).

We have

$$\theta_i = \pm 2 \arccos p_{i,0} + 4k_i\pi$$

for some $k_i \in \mathbb{Z}$.

We now have to determine the value of k_i for each relative rotation. We use for this a continuity assumption: at least for the two end segments, the relative rotation from one time step to the following one cannot have a greater amplitude than π.[2] We will then determine iteratively the values of θ_i, either chronologically (from one time-step to another) or spatially (from one quaternion to its neighbor during the first time-step[3]).

Composition of two rotations In the case of two-dimensional rotations, the angle of the composition of two rotations is easy to determine. Using the complex representation, with θ and ϕ the angles of the two rotations being composed and ψ the angle of the resulting rotation,

$$e^{i\psi} = e^{i\phi}e^{i\theta} = e^{i(\theta+\phi)}$$

which lead to the composition law

$$\psi = \theta + \phi$$

if the two rotations are in the same direction and

$$\psi = \theta - \phi$$

if they are in opposite directions.

In tree dimensions, it is much more complex. The unit quaternions are a very nice representation of SO(3). A rotation of an angle θ around an (unit) axis u is given by

$$\mathfrak{q} = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{u}\right)$$

Let us consider the composition of two rotations of angles θ and ϕ around the axes $\mathbf{u}$ and $\mathbf{v}$ resulting in a rotation of angle ψ around an axis $\mathbf{w}$ with the respective quaternions $\mathfrak{q}$, $\mathfrak{p}$ and $\mathfrak{r}$.[4] We furthermore suppose that $0 < \phi < \pi$.[5] (θ is free since it has to contain information about the winding number.) The composition rule is

$$\mathfrak{r} = \mathfrak{p}\mathfrak{q} = (p_0q_0 - \mathbf{p}^T\mathbf{q}, p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p} \times \mathbf{q})$$

[2]Otherwise it would not be possible to determine the angle of the rotation

[3]$\theta_0 = 0$ by construction

[4]$\mathfrak{p} = \left(\cos\frac{\phi}{2}, \sin\frac{\phi}{2}\mathbf{v}\right)$ and $\mathfrak{r} = \left(\cos\frac{\psi}{2}, \sin\frac{\psi}{2}\mathbf{w}\right)$

[5]The interval $]-\pi : 0[$ can be obtained by reversing the axis $\mathbf{v}$.

We have then

$$\begin{aligned} cos\frac{\psi}{2} &= r_0 \\ &= p_0 q_0 - \mathbf{p}^T \mathbf{q} \\ &= cos\frac{\theta}{2} cos\frac{\phi}{2} - \mathbf{u}^T \mathbf{v} \sin\frac{\theta}{2} \sin\frac{\phi}{2} \\ &= \frac{1}{2}\left(\cos\frac{\theta+\phi}{2} + \cos\frac{\theta-\phi}{2}\right) + \frac{1}{2}\mathbf{u}^T\mathbf{v}\left(\cos\frac{\theta+\phi}{2} - \cos\frac{\theta-\phi}{2}\right) \\ &= \frac{1+\mathbf{u}^T\mathbf{v}}{2}\cos\frac{\theta+\phi}{2} + \frac{1-\mathbf{u}^T\mathbf{v}}{2}\cos\frac{\theta-\phi}{2} \end{aligned}$$

Defining

$$\lambda = \frac{1+\mathbf{u}^T\mathbf{v}}{2} \in [0:1]$$

we have

$$cos\frac{\psi}{2} = \lambda \cos\frac{\theta+\phi}{2} + (1-\lambda)\cos\frac{\theta-\phi}{2}$$

The cosines of the resulting half angle is a linear interpolation between the cosines of the sum and difference of the two half angles. If the two unit axes are parallel and in the same direction ($\mathbf{u}^T\mathbf{v} = 1$), λ is 1 and $cos\frac{\psi}{2} = \cos\frac{\theta+\phi}{2}$. On the contrary, if $\mathbf{u}^T\mathbf{v} = -1$, we have $\lambda = 0$ and $cos\frac{\psi}{2} = \cos\frac{\theta-\phi}{2}$.

In analogy to the two dimensional case, it is reasonable to suppose that

$$\theta - \phi \leq \psi \leq \theta + \phi$$

with the extremal values reached for $\mathbf{u}^T\mathbf{v} = \mp 1$

We now want to determine the value of ψ. Apart from the case where $\theta = 0[2\pi]$, there is only one possible value for ψ in the interval $[\theta - \phi : \theta + \phi]$:

- Case 1: $\nexists k \in \mathbb{Z}/2k\pi \in [\theta - \phi : \theta + \phi]$ The whole interval is included in an interval of the form $[2k\pi : 2(k+1)\pi], k \in \mathbb{Z}$
  - Case 1a: k is even :$\psi = 2\arccos\left(cos\frac{\psi}{2}\right) + 2k\pi$
  - Case 1b: k is odd : $\psi = -2\arccos\left(cos\frac{\psi}{2}\right) + 2(k+1)\pi$
- Case 2: $\exists l \in \mathbb{Z}/2l\pi \in [\theta - \phi : \theta + \phi]$and $\nexists m \in \mathbb{Z}/\theta = 2m\pi$ Let $k \in \mathbb{Z}$ be such that $\theta \in [2k\pi : 2(k+1)\pi]$
  - Case 2a: k is even: $\psi = 2\arccos\left(cos\frac{\psi}{2}\right) + 2k\pi$
  - Case 2b: k is odd: $\psi = -2\arccos\left(cos\frac{\psi}{2}\right) + 2(k+1)\pi$

  In both cases we have $\psi = 2(1 - 2(k[2]))\arccos\left(cos\frac{\psi}{2}\right) + 2(k + (k[2]))\pi$

- Case 3: $\exists k \in \mathbb{Z}/\theta = 2k\pi$ This is the only case where the result is undetermined : two values for $\frac{\psi}{2}$ are possible. Since we always have different values of the quaternions (either from previous iterations for the same segment or from neighboring segments at the same iteration), the easiest is to take one of those (which should be different from $2k\pi$) to define the value of $\frac{\psi}{2}$. To avoid problems with numerical errors, it is certainly wise to apply the same procedure if $|\cos\frac{\theta}{2}| > 1 - \epsilon$.

Calculation of θ_i We can now use the previous results for determining the value of θ_i. Suppose we know the quaternions $\mathfrak{p}_{i-1}$ and $\mathfrak{p}_i$ as well as the value θ_{i-1}. We have then $\cos\frac{\psi}{2} = p_{i,0}$. The value of θ_{i-1} allow us to choose the sign and the value of k.

Implementation

We add an extra variable θ_i for each segment for resolving the problem.

We add two new energies: one for avoiding jumps of 2π and one for coordinating θ and the quaternions.

Energy 1 The first energy is, for p an even integer,

$$E_\theta = \frac{1}{p} k_\theta \left(\frac{\theta_{i+1} - \theta_i}{\pi} \right)^p$$

with derivatives, for $\Delta\theta = \theta_{i+1} - \theta_i$, using a similar notation

$$F_{\Delta\theta} = -k_\theta \frac{1}{\pi^p} \Delta\theta^{p-1}$$

$$H_{\Delta\theta} = (p-1) k_\theta \frac{1}{\pi^p} \Delta\theta^{p-2}$$

The choice of an exponent greater than 2 gives a much "flatter" energy curve, giving a very low weight to small differences and a greater weight to difference bigger than 2π. (See figure 3.22) Since we want to avoid "big" changes between θ_i and θ_{i+1} but we do not know a typical value of the difference, a quadratic energy would be to strong; an energy with an higher exponent would not penalize smaller changes.

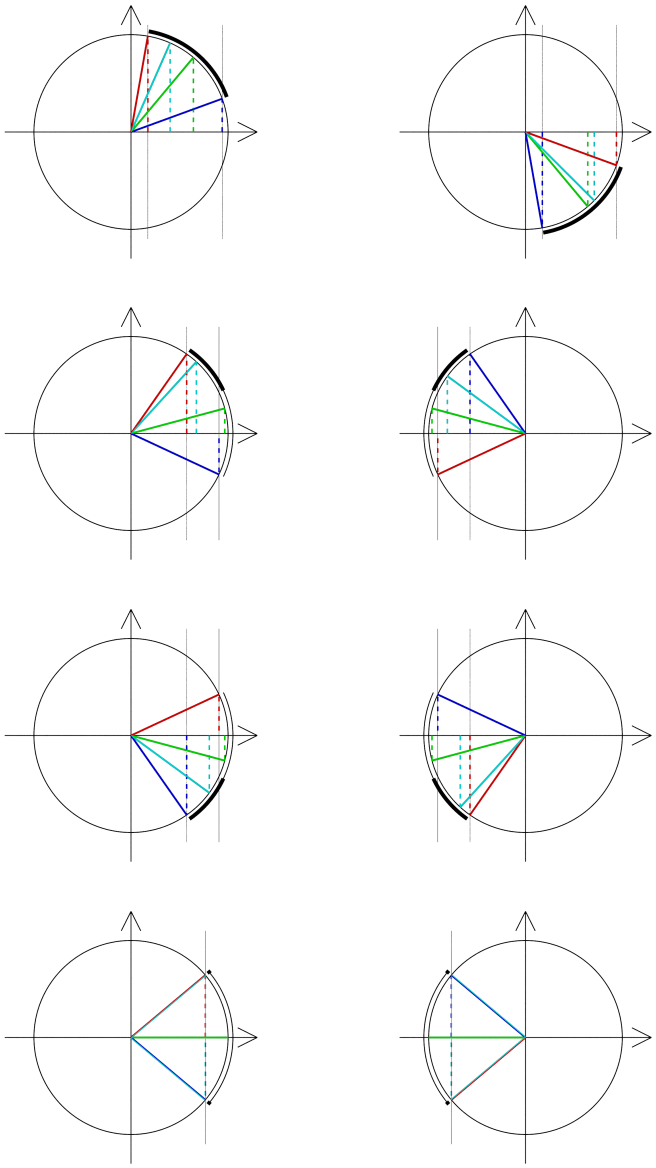

Figure 3.21: Value of ψ: the different possible configurations in the trigonometric circle. $\frac{\theta}{2}$ is represented in green, $\frac{\theta+\phi}{2}$ in red, $\frac{\theta-\phi}{2}$ in blue, and $\frac{\psi}{2}$ in cyan. The vertical lines represent the limits of the possible cosines values, the thin arc circles the angles values in the interval $[\theta - \phi : \theta + \phi]$ and the thick arc circles the possible values for the angle $\frac{\psi}{2}$. From top to bottom and form left to right: the cases 1a, 1b, 2a, 2b and 3, the last 3 cases each with $cos\frac{\theta}{2} > 0$ and $cos\frac{\theta}{2} > 0$.

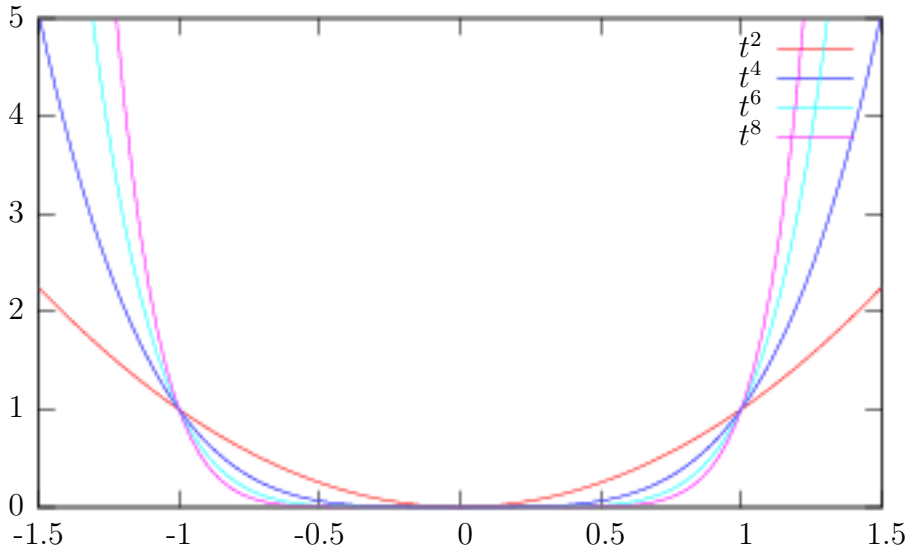

Figure 3.22: Different energies of the form t^p: changing p allows to modulate the difference between θ_{i+1} and θ_i that does not lead to a sensible change in the energy.

Coherence Energy between θ_i and $\mathfrak{q}_i$ For the second one, the relative rotation is for $\mathfrak{Q} = \mathfrak{q}_0$ and $\mathfrak{q} = \mathfrak{q}_i$

$$\mathfrak{p} = \mathfrak{q}\overline{\mathfrak{Q}} = (q_0 Q_0 + \mathbf{q}^T \mathbf{Q}, -q_0 \mathbf{Q} + Q_0 \mathbf{q} + \mathbf{Q} \times \mathbf{q})$$

The second one is divided in two parts

$$E_{\theta q1} = \frac{1}{2} k_{\theta q1} \left(p_{i,0} - \cos \frac{\theta_i}{2} \right)^2 = \frac{1}{2} k_{\theta q1} \left(Q_0 q_0 + Q_1 q_1 + Q_2 q_2 + Q_3 q_3 - \cos \frac{\theta_i}{2} \right)^2$$

with

$$F_{\theta q1, q_{i,j}} = -\frac{\partial E_{\theta q1}}{\partial q_{i,j}} = -k_{\theta q1} Q_j \left(p_{i,0} - \cos \frac{\theta}{2} \right)$$

$$F_{\theta q1, \theta_i} = -\frac{\partial E_{\theta q1}}{\partial \theta_i} = -\frac{1}{2} k_{\theta q1} \sin \frac{\theta}{2} \left(p_{i,0} - \cos \frac{\theta}{2} \right)$$

$$H_{\theta q1, q_{i,j}, q_{i,l}} = \frac{\partial^2 E_{\theta q1}}{\partial q_{i,j} \partial q_{i,l}} = k_{\theta q1} Q_j Q_l$$

$$H_{\theta q1, \theta_i, \theta_i} = \frac{\partial^2 E_{\theta q1}}{\partial \theta_i^2} = \frac{1}{4} k_{\theta q1} \left(\cos \frac{\theta}{2} \left(p_{i,0} - \cos \frac{\theta}{2} \right) + \sin^2 \frac{\theta}{2} \right)$$

$$H_{\theta q1, q_{i,j}, \theta_i} = \frac{\partial^2 E_{\theta q1}}{\partial q_{i,0} \partial \theta_i} = \frac{1}{2} k_{\theta q1} Q_j \sin \frac{\theta}{2}$$

The second part is

$$E_{\theta q2} = \frac{1}{2} k_{\theta q2} \left(\mathbf{p}^T \mathbf{p} - \sin^2 \frac{\theta_i}{2} \right)^2$$

$$F_{\theta q2, q_{i,j}} = -\frac{\partial E_{\theta q2}}{\partial q_{i,j}} = -2 k_{\theta q2} \frac{\partial \mathbf{p}}{\partial q_{i,j}}^T \mathbf{p} \left(\mathbf{p}^T \mathbf{p} - \sin^2 \frac{\theta_i}{2} \right)$$

with

$$\begin{array}{rcl} \frac{\partial \mathbf{p}}{\partial q_{i,0}} & = & (\quad Q_1, \quad Q_2, \quad Q_3 \quad)^T \\ \frac{\partial \mathbf{p}}{\partial q_{i,1}} & = & (\quad Q_0, \quad Q_3, \quad -Q_2 \quad)^T \\ \frac{\partial \mathbf{p}}{\partial q_{i,2}} & = & (\quad -Q_3, \quad Q_0, \quad Q_1 \quad)^T \\ \frac{\partial \mathbf{p}}{\partial q_{i,3}} & = & (\quad Q_2, \quad -Q_1, \quad Q_0 \quad)^T \end{array}$$

$$F_{\theta q2, \theta_i} = -\frac{\partial E_{\theta q2}}{\partial \theta_i} = k_{\theta q2} \cos \frac{\theta_i}{2} \sin \frac{\theta_i}{2} \left(\mathbf{p}^T \mathbf{p} - \sin^2 \frac{\theta_i}{2} \right)$$

$$H_{\theta q2, q_{i,j}, q_{i,k}} = \frac{\partial^2 E_{\theta q2}}{\partial q_{i,j} \partial q_{i,k}} = 2 k_{\theta q2} \left(\frac{\partial \mathbf{p}}{\partial q_{i,j}}^T \frac{\partial \mathbf{p}}{\partial q_{i,k}} \left(\mathbf{p}^T \mathbf{p} - \sin^2 \frac{\theta_i}{2} \right) + 2 \frac{\partial \mathbf{p}}{\partial q_{i,j}}^T \mathbf{p} \frac{\partial \mathbf{p}}{\partial q_{i,k}}^T \mathbf{p} \right)$$

$$H_{\theta q2, q_{i,j}, \theta_i} = \frac{\partial^2 E_{\theta q2}}{\partial q_{i,j} \partial \theta_i} = -2 k_{\theta q2} \cos \frac{\theta_i}{2} \sin \frac{\theta_i}{2} \frac{\partial \mathbf{p}}{\partial q_{i,j}}^T \mathbf{p}$$

$$H_{\theta q2, \theta_i, \theta_i} = \frac{\partial^2 E_{\theta q2}}{\partial \theta_i^2} = -\frac{1}{2} k_{\theta q2} \left(\left(-\sin^2 \frac{\theta_i}{2} + \cos^2 \frac{\theta_i}{2} \right) \left(\mathbf{p}^T \mathbf{p} - \sin^2 \frac{\theta_i}{2} \right) - \cos^2 \frac{\theta_i}{2} \sin^2 \frac{\theta_i}{2} \right)$$

Limitations

This scheme does not take into account the translation movements of the ends of the cable. But these can have an effect on the rotations, as show the 360 and 720 belt tricks: take a cable, extend it along a line, turn one end of 360 (resp. 720) around its tangent, and move one end in a circle around the other one. Now, the cable is twisted in the other direction (360) or is untwisted (720). This behavior can be explained if the closure (a prolongation of the cable to transform it in a circle) is considered: during the belt trick, the closure intersects itself, thus provoking a jump of ± 2 in the link.

Chapter 4

Integral Forces

Note: in this chapter, s represents the variable of the Lagrange Transformation and not the cable arclength.

4.1 Short Introduction to Control Theory

Control theory aims to influence from outside a process changing with time in such a way that it runs in a predefined way (for example, that some values stay as constant as possible.) The classical way to do this is introducing feedback with a closed-loop system: the control signal is a function of the control error (the difference between the actual value of the variable to be controlled and its desired value). (Figure 4.1)

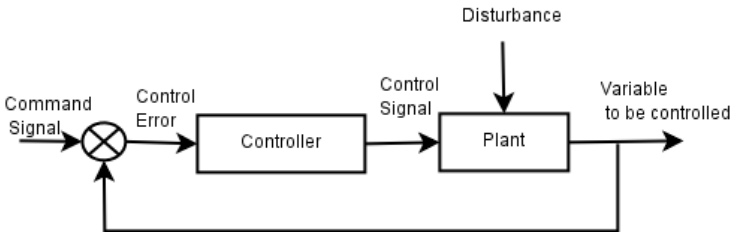

Figure 4.1: The classical control loop

In the case of a segment whose length we want to control, the control variable is the length of the segment, the command signal is the reference length of the segment, the plant is made up of the two points of the segment and the disturbance are all the other interactions (the weight, but also the behavior of the rest of the cable).

A classical form for the controller is a so-called PID-controller: if $e(t)$ is the control error (seen as a function of time), the control signal at time t is

formed by the sum of 3 modes: the proportional one (P) $k_P e(t)$, the integral one (I) $k_I \int_0^t e(u)\mathrm{d}u$ and a derivative one $k_D \frac{\mathrm{de}}{\mathrm{dt}}(t)$ where the constants k_P, k_I, k_D are chosen in function of the properties of the plant and of the desired behavior.

If we look at what we have done in the previous chapter, a spring alone is a proportional controller: the actuating variable (the force) is proportional (with the proportionality constant k_{Length}) to the difference between the desired value (L_{ref}) and the actual one (L_i). The forces for segment i between point i and $i+1$ due to the stretch forces are

$$\mathbf{f}_{P,\,i} = -\mathbf{f}_{P,i+1} = k_{Length}(L_i - L_{ref})\mathbf{u_i}$$

The derivative mode is the viscous damping in a dynamic system.

We want to use the equivalent of a PI-(proportional-integral) controller, which has the property of being able to enforce a constraint exactly so that the steady-state error is null. The PI-controller combines a proportional part (the spring) with an integral one. Its actuating variable is proportional to the integral over the time of the difference between the desired and actual values. When the error is null, the proportional force is also null, but the integral one counteracts the perturbations. The proportional force is nonetheless essential to the stability of the system. As a consequence of the presence of the integral force, the proportional constant can be reduced (in comparison with a spring-only system), which makes the system less stiff as a whole.

4.2 Explanation of the principle: continuous version

Let us take a simple, one-dimensional reference setting for explaining the idea. A (mass) point has the coordinate $x \in \mathbb{R}$. It is submitted to the following forces: a spring force $-kx$ [1], an integral force $-f$ and an external force w independent of x and its derivatives (for example the weight, or any function of the time only). Since we consider a quasi- static system, the force equilibrium is always respected. [2]

$$-kx - f + w = 0$$

[1] A translation gives the traditional form $-k(x - x_0)$

[2] It is equivalent to considering the dynamic of a mass-point of mass $m = 0$: the first Newton principle for a null mass is the equilibrium of forces at all times: $m\ddot{x} = -kx - f + w$ leads to $0 = -kx - f + w$

hence

$$x = \frac{w - f}{k}$$

In the case where $f = 0$, the amplitude is the classical result $\frac{w}{k}$: it is the control deviation. We want now to bring the point in the position $x = 0$ by means of the integral force.

We now set

$$\dot{f} = k_I x$$

where the dot represents the time derivation.

With the Lagrange transformation (the Lagrange transformation is noted in capitals, following the standard use)

$$-kX - \frac{k_I}{s}X + W = 0$$

which leads to the transfer function

$$\frac{X}{W} = \frac{s}{ks + k_I}$$

It is a DT1 system (a differential system with a delay).

The pole is $s = -\frac{k_I}{k}$: it has a negative real part: the system is stable. The limit for an infinite time is

$$\lim_{t\to\infty} x(t) = \lim_{s\to 0} sX(s) = \lim_{s\to 0} s\frac{s}{ks + k_I}W = \lim_{s\to 0} s\frac{s}{ks + k_I}\frac{w_0}{s} = \lim_{s\to 0}\frac{s}{ks + k_I}w_0 = 0$$

in the particular case of a constant (for $t > 0$) $w(t) = w_0$. There is no control deviation.

For the integral force, we have

$$\frac{F}{W} = \frac{k_I}{ks + k_I}$$

It is a PT1-system with a proportional gain of 1.

In time domain, we have

$$x = \frac{w - f}{k}$$

and

$$\dot{f} = k_I x = -\frac{k_I}{k}f + \frac{k_I}{k}w = -\alpha f + \alpha w$$

with $\alpha = \frac{k_I}{k}$. For a constant $w = w_0$, with $x(0) = 0$ and $f(0) = 0$, the solution is then, for $t > 0$

$$f(t) = w_0(1 - e^{-\alpha t}) = w_0(1 - e^{-\frac{k_I}{k}t})$$

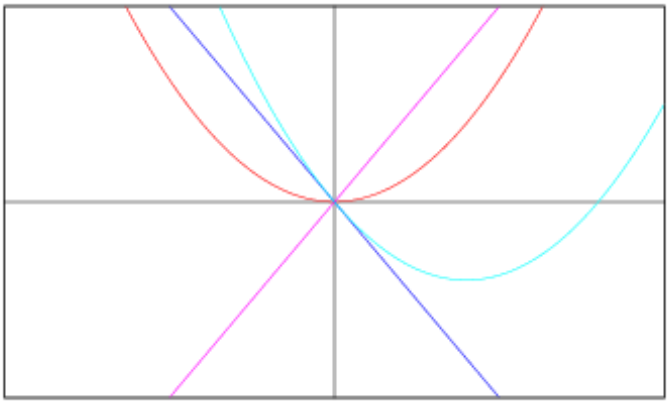

Figure 4.2: The energy from which the force $-kx - f + w$ can be derivated is $E = \frac{1}{2}kx^2 + fx - wx$. Without external influence, the minimum of the energy lies at $x = 0$ (in red). Under the influence of external forces (in blue), the minimum is shifted to some position $x > 0$ (in cyan). The addition of the integral force (in magenta) allows to shift it back to the desired position $x = 0$

and

$$x(t) = \frac{w_0}{k} e^{-\frac{k_I}{k}t}$$

The bigger α is, the faster the system converges.

4.3 Numerical Problems

The continuous system is stable. But, like for all integration schemes of differential equations, this does not mean that the numerical system is stable.

We proceed in the following way: at the beginning, we have $f_0 = 0$ as an initial condition. At each time step i, we solve the energy minimization problem for x, maintaining f constant equal to f_{i-1} :

$$x_{i+1} = \frac{w - f_i}{k}$$

We then update the integral force

$$f_{i+1} = f_i + k_I x_{i+1} = (1 - \alpha) f_i + \alpha w$$

The explicit solution of this sequence is (for w constant)

$$f_i = w_0(1 - (1 - \alpha)^i)$$

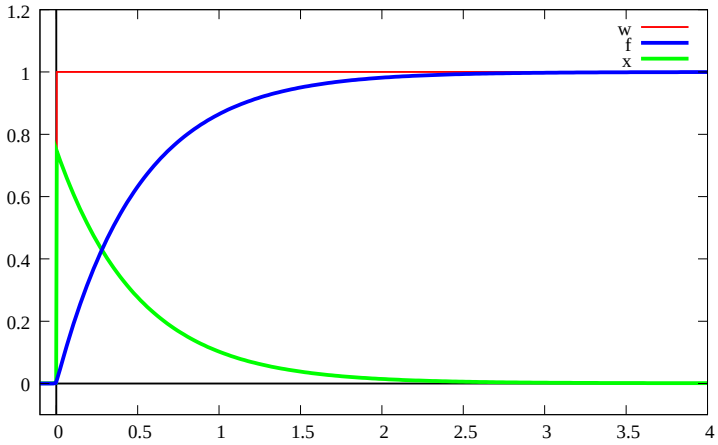

Figure 4.3: Step response of the integral force f and of the position x

and

$$x_i = \frac{w_0}{k}(1-\alpha)^i$$

Both solutions are stable and converge if and only if $|1-\alpha| < 1$, which is equivalent to $0 < \alpha < 2$. There are no oscillations for $0 \leq \alpha \leq 1$. The process is equivalent to an Euler-Forward integration scheme for the integral force.

For having a stable and oscillation-free system, we need

$$0 < \frac{k_I}{k} < 1$$

In particular, the proportional force cannot be suppressed.

The fastest convergence rate possible is then reached for $\alpha = 1$, which means for $k_I = k$

4.4 Length conservation

Let us consider a segment i (between points $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$) and apply the previous principle to the conservation of its length.

With only the proportional mode is used, the energy is, as explained before in the previous chapter,

$$E_{Length,\,i} = \frac{1}{2}k_{Length}(L_i - L_{ref})^2$$

where k_{Length} is the constant of the spring.

The force on the point $\mathbf{x}_i$, just opposed to the force on point $\mathbf{x}_{i+1}$, is

$$\mathbf{F}_{Length,\,i} = k_{Length}\mathbf{u}_i(L_i - L_{ref})$$

If we consider only this spring, the minimum of the energy is clearly at $L_i = L_{ref}$. But the points i and $i+1$ are also submitted to other interactions that result in a disturbing force that stretches (or compresses) the spring. Since the spring can only exert a force when it is not at equilibrium, it alone cannot enforce exactly the constraint of a constant length: we use the integral mode to achieve this.

To implement this, we add constant forces $f_{I,Length,\,i}\,\mathbf{u_i}$ on point i and $-f_{I,Length,\,i}\mathbf{u_i}$ on point $i+1$. This force must have the same direction as the proportional one. The total force on point i due to the stretch of segment i is then

$$\mathbf{f}_i = (-f_{I,\,i} + k_{Length}(L_i - L_{ref}))\mathbf{u_i}$$

This corresponds to an energy of

$$E_{Length,\,i} = \frac{1}{2}k_{Length}(L_i - L_{ref})^2 + f_{I,Length,\,i}(L_i - L_{ref})$$

We have now $n-1$ additional parameters $f_{I,Length,\,i}$. For solving the system, we use an iterative approach: we first solve the system holding the $f_{I,Length,\,i}$ constant, then adjust their values in function of the result and solve anew the system until an equilibrium is reached (The constraints are enforced within an ϵ tolerance that should be chosen slightly bigger than the numerical precision of the system solver).

For updating the integral forces, we adjust each one individually: the new force is the old one augmented with the difference of lengths multiplied by a constant

$$f_{I,Length,\,i,\,new} = f_{I,Length,\,i,\,old} + k_{I,\,Length}(L_i - L_{ref})$$

4.4.1 Results

Several results of the application of integral forces to the length are shown in figures 4.4 to 4.8.

4.5 Coherence between Positions and Quaternions

The same mechanism is used for the coherence between positions and quaternions, but in this case the integral force is three-dimensional.

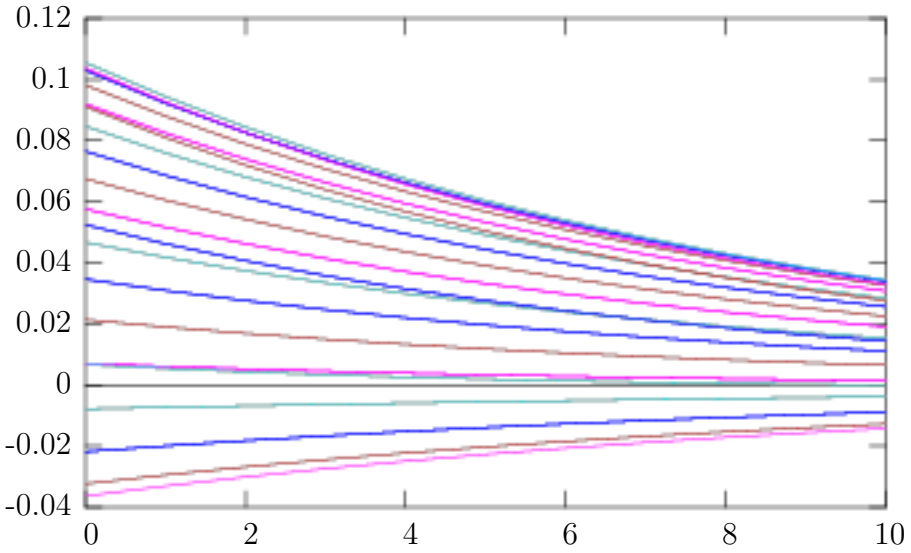

Figure 4.4: Effects of the integral force on the length error: length error for each segment as a function of the number of iterations, here for $\alpha = 0.1$: a slow convergence to 0 can be seen. The system stops before reaching 0 because the small differences in the integral forces are not enough to change the positions.

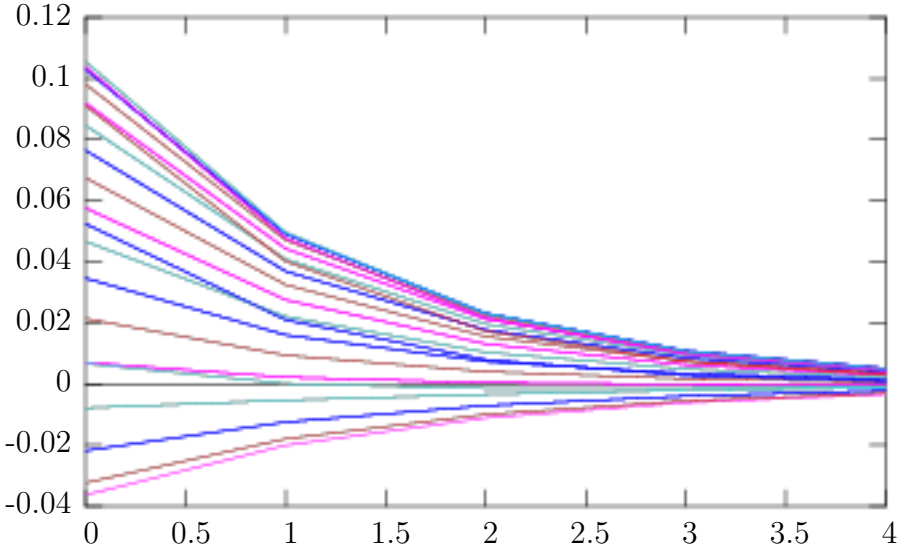

Figure 4.5: Effects of the integral force on the length error: length error for each segment as a function of the number of iterations, here for $\alpha = 0.5$: the convergence to 0 is faster.

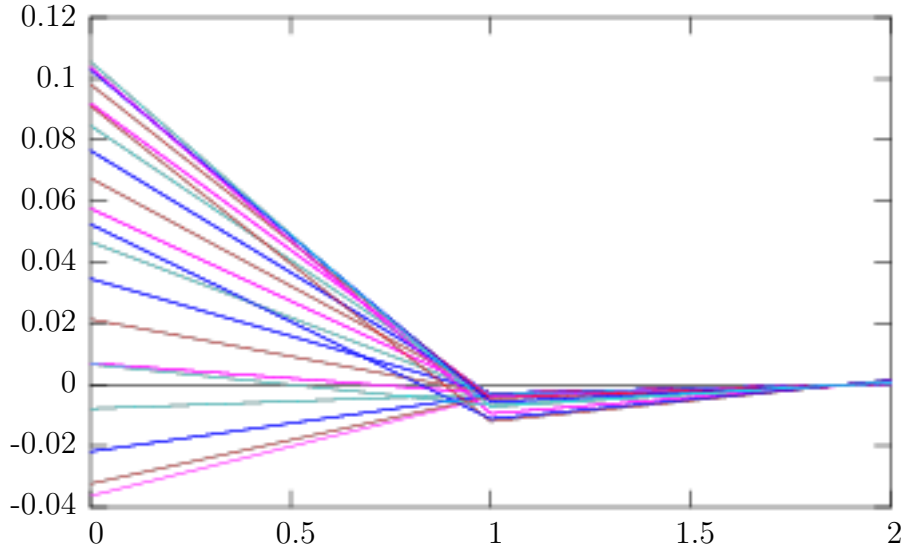

Figure 4.6: Effects of the integral force on the length error: length error for each segment as a function of the number of iterations, here for $\alpha = 1$: the convergence to 0 is faster, but a slight overshoot is visible.

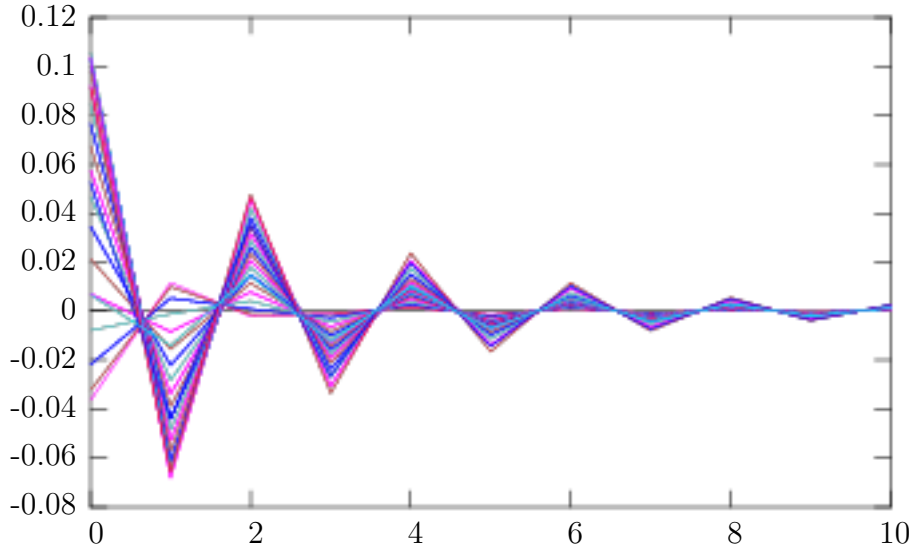

Figure 4.7: Effects of the integral force on the length error: length error for each segment as a function of the number of iterations, here for $\alpha = 1.5$: the system still converges to 0, but after numerous oscillations.

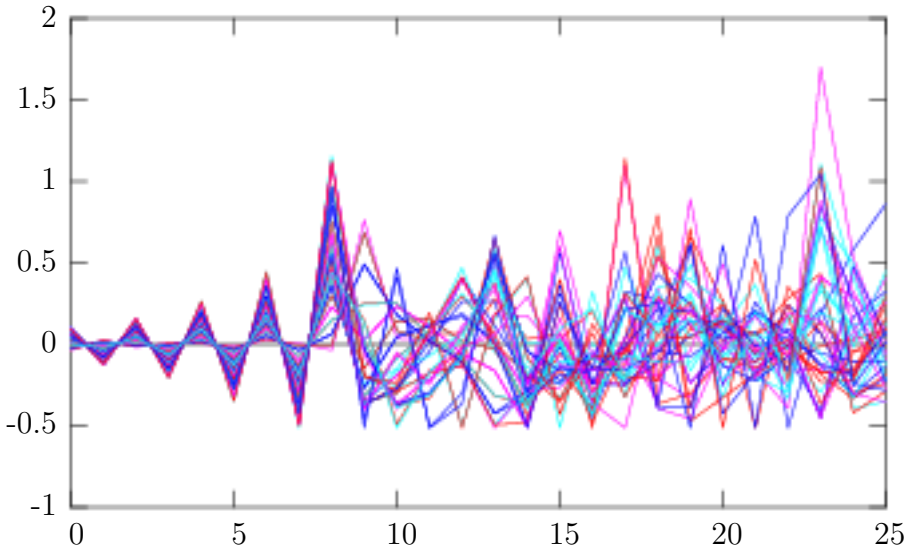

Figure 4.8: Effects of the integral force on the length error: length error for each segment as a function of the number of iterations, here for $\alpha = 2$: the system is unstable.

The energy is defined as

$$E_{Coh,i} = \frac{1}{2} k_{Coh} \mathbf{\Delta}^T \mathbf{\Delta}$$

with

$$\mathbf{\Delta} = \mathbf{u_i} - Rot_{\mathbf{q}_i}(\mathbf{ref})$$

In the same manner as for the length conservation, we introduce an integral force for each of the three components of the coherence between quaternions and positions: we want the vector $\mathbf{\Delta}$ to be $\mathbf{0}$. For each segment, the integral force $\mathbf{f}_{I,i}$ has 3 components and the following update rule is used:

$$\mathbf{f}_{I,i,new} = \mathbf{f}_{I,i,old} + k_{I,\,Coh} \mathbf{\Delta}$$

The modified energy is now

$$E_{Coh,i} = \frac{1}{2} k_{Coh} \mathbf{\Delta}^T \mathbf{\Delta} + \mathbf{\Delta}^T \mathbf{f}_{I,Coh,i}$$

and the forces are[3]

$$F_{Coh,i,x_j} = -k_{Coh} \frac{\partial \mathbf{\Delta}}{\partial x_j}^T \mathbf{\Delta} - \frac{\partial \mathbf{\Delta}}{\partial x_j}^T \mathbf{f}_{I,i}$$

[3]The force is a pure moment on the two point-segment (it has no component along the segment direction $\mathbf{u}_i$) and has, as such, no direct influence on the length of the segment.

The update rule is

$$f_{I,Coh,\,i,\,new} = f_{I,Coh\,i,\,old} + k_{I,\,Coh}\mathbf{\Delta}$$

The results are similar to the one obtained with the length.

The two integral forces cohabit well, except in the case of strong flexion/torsion constants, where oscillations appears. Augmenting $k_{QuatNorm}$ or reducing α_{Length} helps.

4.6 Quaternion Norm

Since the restraining influence of the coherence force is lost, we cannot simply use the quadratic form of the energy $\frac{1}{2}k_{QuatNorm}\left(\|\mathfrak{q}_\mathfrak{i}\|-1\right)^2$. It is much better to use the hyperbolic difference which avoids that the norm gets to small, and we can use it for all the value domain (it is less sensible for a norm bigger than 1 as the other form, but since there is no force pushing the norm to grow, this is not a problem.

The energy is replaced by

$$E_{QuatNorm,i} = \frac{1}{2}k_{QuatNorm}\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)^2 + f_{I,QuatNorm,i}\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)$$

with the update rule

$$f_{I,\,i,\,new} = f_{I,\,i,\,old} + k_{I,\,QuatNorm}\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)$$

But it is not the case for the quaternion, which could lead to interferences with the quaternion norm. A solution would be to modify the coherence energy to make it independent from the quaternion norm. The norm of the image $rot_{\mathfrak{q}_i}(\mathbf{ref})$ is $\|\mathfrak{q}_i\|^2$ Only the definition of $\mathbf{\Delta}$ should be replaced:

$$\mathbf{\Delta} = \mathbf{u_i} - \frac{1}{\|\mathfrak{q}_i\|^2}Rot_{\mathfrak{q}_i}(\mathbf{ref})$$

A drawback of this is that the coherence energy counterbalanced the influence of the bending and torsion energy (which tend to diminish the quaternion norm). Since we loose this advantage, we will also have to add an integral force for the quaternion norm. Another possibility would be to use the normed quaternions for calculating the flexion and torsion energy, but this complicates unnecessarily the derivatives.

Chapter 5

Contact forces

For the realism and the utility of the simulation the interaction with the environment is very important, in particular the detection of collisions and the treatment of contact forces to avoid the interpenetration of the cable and solid objects.

5.1 Basic principle

When a collision is detected (the mass point i has penetrated into another body), we want the mass point to be outside, at a distance equal to the radius, from the surface. To achieve this, we will add a force (and the corresponding energy) for the mass point: we apply a spring and an integral force, in a similar way as before, to keep it on the surface.[1]

Let $\mathbf{y} \in \mathbb{R}^3$ be the orthogonal projection of $\mathbf{x_i}$ on the surface and $\mathbf{n} \in \mathbb{R}^3$ the normal at $\mathbf{y}$, oriented towards the outside and normed. The penetration distance is then $d = (\mathbf{y} - \mathbf{x_i})^T \mathbf{n}$ (d is positive when $\mathbf{x}$ is inside of the body) and the force is

$$\mathbf{f_{Collision,\,i}} = k_{Collision}\; d\mathbf{n} + f_{I,\,Collision,\,i}\mathbf{n}$$

and the energy

$$E_{Collision,\,i} = \frac{1}{2} k_{Collision}\; d^2 + f_{I,\,Collision,\,i}\; d$$

Without the integral force, the minimum is clearly at $d = 0$. As before, the integral force is used to enforce the constraint exactly.

[1] The surface is displaced of a distance equal to the radius along its normal to take into account the radius.

The release mechanism is the following: when the penetration distance of a point is 0 (within the tolerance) and its integral force $f_{I,\,Collision,\,i}$ is negative- the force on the point is directed towards the inside of the body- it means that the point needs to be attracted towards the object in order to be on the surface. In this case, it is not a collision point anymore and the spring and the integral forces are removed. In the case of several collision points, it is important to wait for all points to be on the surface to release any of them, otherwise an oscillation between the states of collision and no-collision could appear.

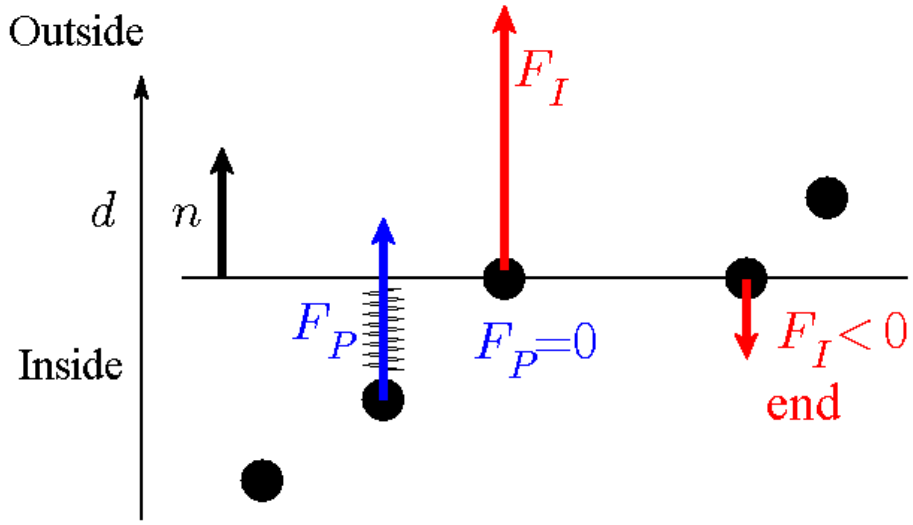

Figure 5.1: Principle of the collision treatment: when the point is on the surface, the proportional force is 0, but the integral force maintains it in position. When the integral force is negative, the point is attracted artificially to the surface: the contact is broken.

The release mechanism is also important in the case where the initial position shows a "big" interpenetration: all points that are inside the object are initially to the surface. Typically, the two exterior points (ie. the first and last points in the order of numeration of the cable to be in contact) will be attracted to the object while the other points will be pushed away by the object. The two exterior points will be released, and this cycle is repeated until a position is reached where no point has to be attracted to the object. In particular for cables with a high bending/torsion stiffness, the correct position after the collision can have only one or two points in contact

with the solid object. In the case where the initial interpenetration is so big that it is not obvious to which side of the object the cable should be pushed, one could imagine to push the points of the cable one at a time, starting by one end of the cable. There can be cases where no satisfying solution can be found (for example, the cable is too short to go round the solid object). A manual intervention would then be necessary.

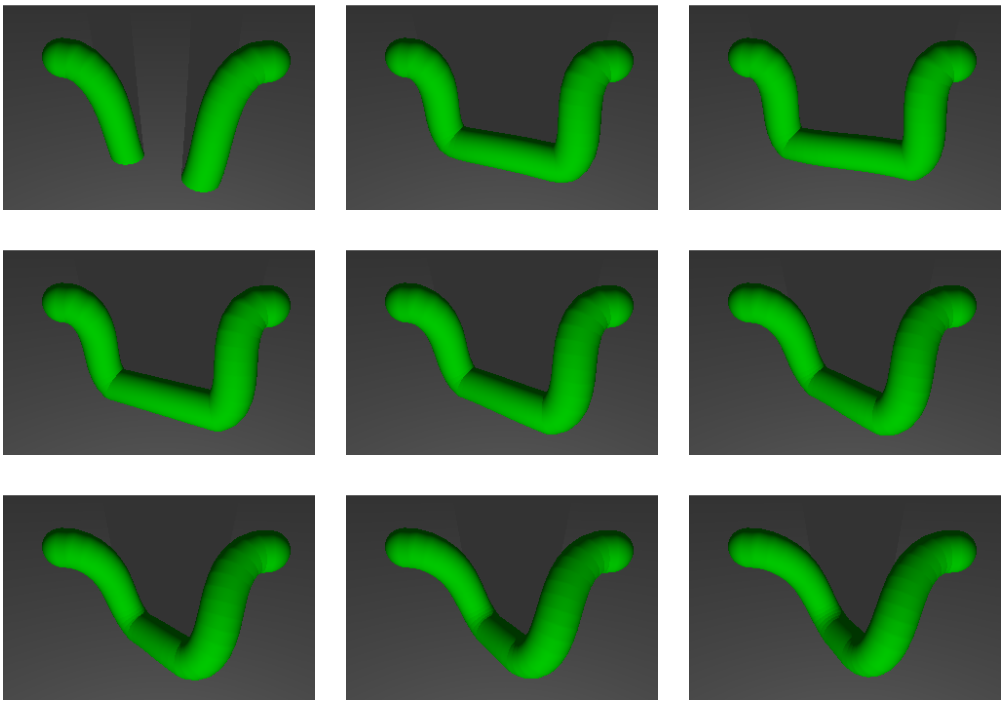

Figure 5.2: Example of resolution with an initial position with interpenetration: from top to bottom and from left to right, the initial position, and successively the different stages of the resolution: between each image, the points with a negative integral force (which were artificially attracted to the surface) were released.

For updating from one time step to another, there are three cases:

- neither the cable nor the solid have moved: there is no need to do anything.
- the solid has moved: the points that were already in collision are updated (the collision point and the normal are possibly modified, the integral force remains the same, even if the point is now outside of the solid, because it is extremely probable that this point will still be a

contact point, but a little less pushed.) The other points are checked for collision and added to the list of collision points if necessary.

- the cable has moved: the new position of the cable is calculated with the collisions of the previous step; we proceed then as in the second case.

5.2 Update of $f_{I,Collision,i}$

$f_{I,Collision,i}$ can be updated with two different methods: an individual or a global one. The first method is similar to the one we used for the length:

$$f_{I,i,new} = f_{I,i,old} + k_{I,Collision} d$$

The second method uses the derivative of the forces as a predictor for the behavior of the system upon a small change of the values of the forces. The Taylor series for the forces (excluding the integral force) is

$$\mathbf{F}(\mathbf{X} + \mathbf{h}) = \mathbf{F}(\mathbf{X}) - \mathbf{H}\mathbf{h}$$

with $\mathbf{h} \in \mathbb{R}^{7n-4}$ the difference between the old and the new position. Since both the old and the new position are equilibriums, we have

$$\mathbf{F}(\mathbf{X}) + f_{I,\,Collision,\,i}\tilde{\mathbf{n}} = \mathbf{0}$$

and

$$\mathbf{F}(\mathbf{X} + \mathbf{h}) + (f_{I,\,Collision,\,i} + \Delta f_I)\tilde{\mathbf{n}} = 0$$

with $\tilde{\mathbf{n}} \in \mathbb{R}^{7n-4}$ the "direction" of the constraint: $\tilde{n}_k = 0$ except for the block corresponding to $\mathbf{x}_i$: $(\tilde{n}_{7i+1},\ \tilde{n}_{7i+2},\ \tilde{n}_{7i+3})$ that takes the values of $\mathbf{n}$. Combining these three equations gives

$$\Delta f_I \tilde{\mathbf{n}} = \mathbf{H}\mathbf{h}$$

In order to fulfill the non-penetration constraint, we need that

$$\mathbf{h}^T \tilde{\mathbf{n}} = d$$

These two equations lead to

$$\Delta f_I = \frac{d}{\tilde{\mathbf{n}}^T \mathbf{H}^{-1} \tilde{\mathbf{n}}}$$

In the case of several contact points, the new force is such that

$$\mathbf{F}(\mathbf{X} + \mathbf{h}) + \sum_{i \in Contactpoints} \Delta f_{I,i} \tilde{\mathbf{n}}_\mathbf{i} = 0$$

and the conditions to fulfill $\tilde{\mathbf{n}}_\mathbf{i}^T\mathbf{h} = d_i$. The combination of these relations results in a matrix equation:

$$\mathbf{M}(\Delta f_{I,1}, ..., \Delta f_{I,c})^T = (d_1, ..., d_c)^T$$

with c the number of contact points and $\mathbf{M} \in \mathbb{R}^{c\times c}$ a matrix such that $\mathbf{M}_{i,j} = \tilde{\mathbf{n}}_\mathbf{i}^T\mathbf{H}^{-1}\tilde{\mathbf{n}}_\mathbf{j}$. Although we have to solve the above system, this method converges experimentally much faster than the other: it only needs one or two iterations to find the correct values for the f_I. It cannot be used as such in the case of the length conservation or in the case of the coherence between positions and quaternions since the direction $\tilde{\mathbf{n}}$ would not be constant.

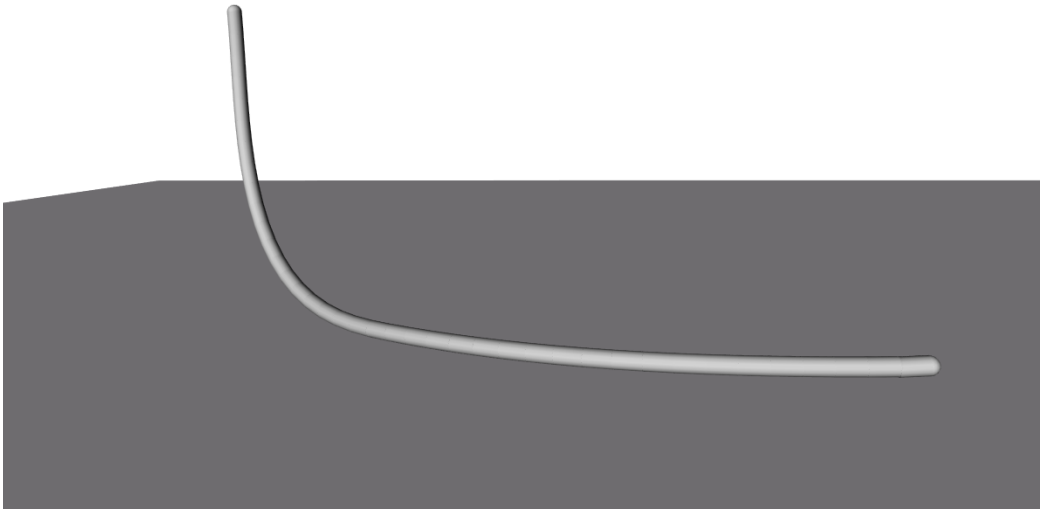

Figure 5.3: Deformation of a cable due to the contact to a plane

5.3 Different kinds of contact forces

5.3.1 Contact of a segment

The contact forces we have considered until now are applied on one point of the cable: the point is constrained to be on the plane defined by a point on the

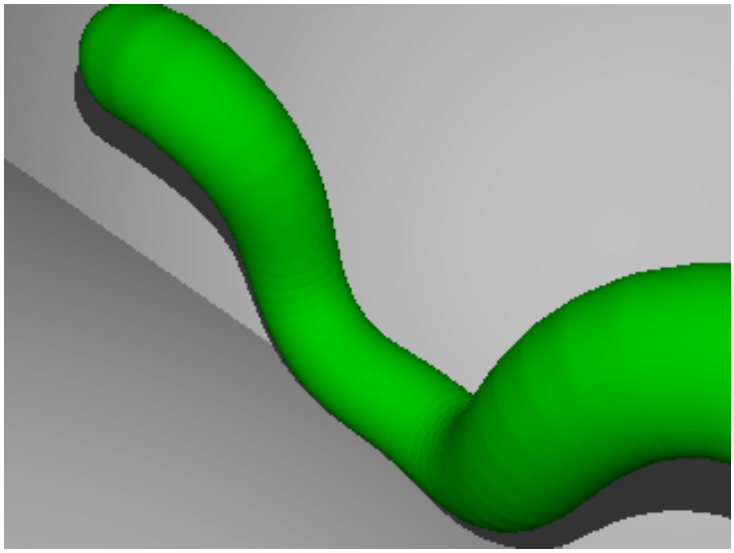

Figure 5.4: Multiple constraints on the same point are no problem: here, a cable colliding with two planes

surface and the normal to the surface. The fact that all the points are outside of a solid does not impede that the segment joining them can interpenetrate the same solid, if the surface is not locally concave. If the curvature radius of the surface is big enough in comparison with the length of a segment, this is not a problem since the penetration distance will be relatively small. (of an amplitude of $R + r - \sqrt{(R+r)^2 - (\frac{l}{2})^2} = (R+r)(1 - \sqrt{1 - (\frac{l}{2(R+r)})^2})$ where R is the (mean) curvature radius of the surface along the direction of the segment, r the radius of the cable and l the length of a segment. If it is a problem, one could constrain the two points of a segment to be in the plane defined by the contact point on the surface and the normal. The segment is then tangent to the surface: the collision of a segment with a surface is then transformed in the collision of the two points of the segment with the plane tangent to the surface.

5.3.2 Contact with a sharp edge

The case of a sharp edge is different: the interpenetration distance can be big. We need a special treatment for them. The idea is to base the energy on the distance between the segment and the point. A sharp edge is defined as the junction of two half-planes, but such that the edge is pointing outwards, like for example the edges of a cube. Let us consider a point $\mathbf{y}$ on the edge, and $\mathbf{n}_1$ and $\mathbf{n}_2$ the normed normals to the two half-planes, and finally $\mathbf{x}_i$ and

$\mathbf{x}_{i+1}$ two points of the cable such that the segment joining them is partly inside of the edge. The previous mechanism ensures that the two points $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ cannot be inside of the sharp edge. Let us define the (normed) vector

$$\mathbf{v} = \frac{\mathbf{n}_2^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_1 - \mathbf{n}_1^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_2}{\|\mathbf{n}_2^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_1 - \mathbf{n}_1^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_2\|} = \frac{(\mathbf{x}_{i+1} - \mathbf{x}_i) \times (\mathbf{n}_1 \times \mathbf{n}_2)}{\|(\mathbf{x}_{i+1} - \mathbf{x}_i) \times (\mathbf{n}_1 \times \mathbf{n}_2)\|}$$

This vector is orthogonal to $\mathbf{x}_{i+1} - \mathbf{x}_i$ (since $\mathbf{v}^T(\mathbf{x}_{i+1} - \mathbf{x}_i) = 0$) and it is in the plane defined by $\mathbf{n}_1$ and $\mathbf{n}_2$ (it is orthogonal to the edge) and is pointing outwards the sharp edge. [2]

The distance between the edge and the segment is given by

$$d = (\mathbf{y} - \mathbf{x}_i)^T\mathbf{v} + r$$

As before, the distance is negative if the segment is outside of the edge. We define the energy as

$$E_{Collision,\,i} = \frac{1}{2}k_{Collision}\; d^2 + f_{I,\,Collision,\,i}\; d$$

This forces the projections of the points $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ on the plane defined by $\mathbf{y}$, $\mathbf{n}_1$ and $\mathbf{n}_2$ to be aligned with $\mathbf{y}$.

The contact is released either when the integral force is negative, or when the edge lies "outside" of the projection of the segment defined by $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$, this is to say that we must have $(\mathbf{x}_{i+1} - \mathbf{x}_i)^T(\mathbf{y} - \mathbf{x}_i) \geq 0$ and $(\mathbf{x}_{i+1} - \mathbf{x}_i)^T(\mathbf{y} - \mathbf{x}_{i+1}) \leq 0$ for the segment to be in contact with the edge. If it not the case, the contact segment is replaced by the next (or previous) segment, following which one of the previous inequalities is not satisfied anymore.

It is also important to notice that this special treatment will not be needed very often since the bending stiffness of the cable will often make the cable not to rest on the edge itself but to go around it. It would only be necessary for a cable under tension.

[2] We have $\mathbf{v}^T(\mathbf{n}_1 + \mathbf{n}_2) = \frac{(\mathbf{n}_2 - \mathbf{n}_1)^T(\mathbf{x}_{i+1} - \mathbf{x}_i)(1 + \mathbf{n}_1^T\mathbf{n}_2)}{\|\mathbf{n}_2^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_1 - \mathbf{n}_1^T(\mathbf{x}_{i+1} - \mathbf{x}_i)\mathbf{n}_2\|}$. Since $\mathbf{n}_1$ and $\mathbf{n}_2$ are unitary, it is a sufficient and necessary condition for the vector $\mathbf{v}$ to point outwards that $(\mathbf{n}_2 - \mathbf{n}_1)^T(\mathbf{x}_{i+1} - \mathbf{x}_i) \geq 0$. If it is not the case, swap $\mathbf{n}_1$ and $\mathbf{n}_2$.

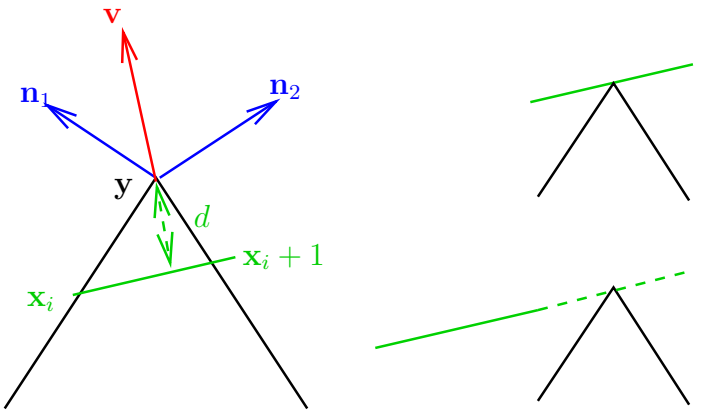

Figure 5.5: Collision with a sharp edge: the left figure shows the notations and the starting position, on the right on top the end position, at the bottom: the contact must be released and replaced by another segment.

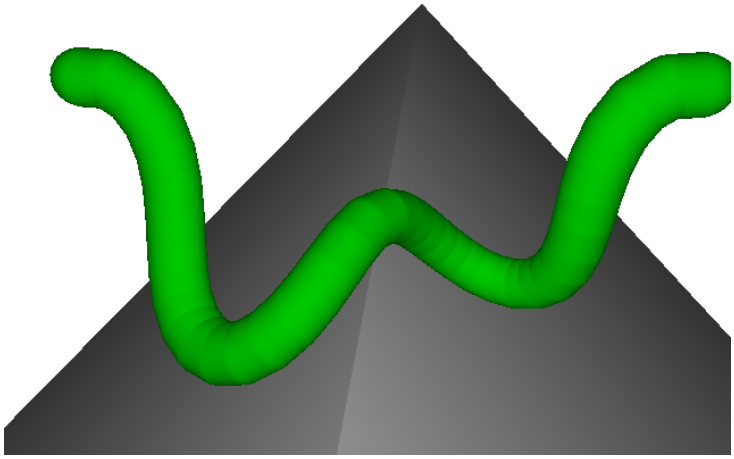

Figure 5.6: A very soft cable in collision with a sharp edge.

Chapter 6

Numerical Solution

Now that we have described the energy function, it is time to describe the way to calculate the new positions from the energy. We use an iterative method for the energy minimization based on the continuity of the solution from one time-step to the other.

6.1 Solver

6.1.1 General principles

Since the cables and hoses do not have a high dynamic range, considering a static solution at each time step is sufficient for most of the applications we are concerned with, like wire routing or assembly simulation. Dynamic phenomena like fast oscillations are excluded by the quasi-static system: the system is after each time step at equilibrium. Numerical oscillations, which are often a problem for stiff spring-mass systems, are also excluded by the absence of speeds as variables. We use an energy minimizing algorithm for solving the system. Note that the system could be modified to be transformed in a dynamical one (See below 6.2).

The basic hypothesis is that the new solution (for slightly modified conditions: a point has been moved between the two frames for example) should be near to the old one, and the old solution vector $\mathbf{X}_{old}$ can thus be used as a good starting point for the search of the new solution. We minimize iteratively the energy until we find a minimum, characterized in the following way: If the norm of the forces vector $\mathbf{F}$ is null (in practice small enough $\|\mathbf{F}\| < \epsilon$ in order to account for numerical error), the system is at equilibrium and we have the solution we were looking for. If not, we minimize the energy until we find an equilibrium.

Our algorithm is iterative. In each loop, a sequence of different algorithms is used until a smaller value for the energy is found. If a particular algorithm gives a solution, $\mathbf{X}_{old}$ is replaced by the new solution and a new loop begins until the equilibrium is reached. If it does not find a better solution, the next algorithm is used. When the difference either in position $\|\mathbf{X}_{new} - \mathbf{X}_{old}\|$ or in energy $\|E_{new} - E_{old}\|$ is smaller than a predetermined value, the loop is stopped. The different algorithms that we use are in order: Newton's method applied to the forces, the non-linear conjugate gradient method, the linear conjugate gradient method, the steepest descent method and if all else fails, a line search along the forces. It is important to note that in the vast majority of the cases, only the first or the first two are used; the other algorithms serve as a security for particular stiff cases or for tuning the different constants for the interactions. This structure allows us to have at the same time a fast response in usual cases while retaining the robustness necessary to deal with stiff cases. Thanks to the formulation of the energies and this iterative process, the mean calculation time for an iteration is practically a linear function of the number of discretization points. (See Figure 6.1)

6.1.2 Individual algorithms

Most of the algorithms are based on the approximation of the energy at a position $\mathbf{X} + \mathbf{h}$ as either a second-order or a first order Taylor series at the position $\mathbf{X}$:

$$E_{approx}(\mathbf{X} + \mathbf{h}) = E(\mathbf{X}) - \mathbf{F}(\mathbf{X})^T\mathbf{h} + \frac{1}{2}\mathbf{h}^T\mathbf{H}(\mathbf{X})\mathbf{h}$$

Newton Algorithm

The first algorithm is the Newton Method applied to the forces. It is particularly efficient near the equilibrium. However, it is well-known that it only converges if the starting solution is near enough to the equilibrium. Another possible source of problems is that the algorithm converges to a point at which the forces are null. It is not necessary a minimum: it can be a saddle point, typically with just a few negative eigenvalues, but the energy can be higher in this point. Using the previous approximation, the energy finds its minimum when its gradient is null, i.e. for $-\mathbf{F} + \mathbf{H}\mathbf{h} = \mathbf{0}$ using the linear approximation of the force, which leads to

$$\mathbf{h} = \mathbf{H}^{-1}\mathbf{F}$$

and

$$\mathbf{X_{new}} = \mathbf{X_{old}} + \mathbf{H}^{-1}\mathbf{F}$$

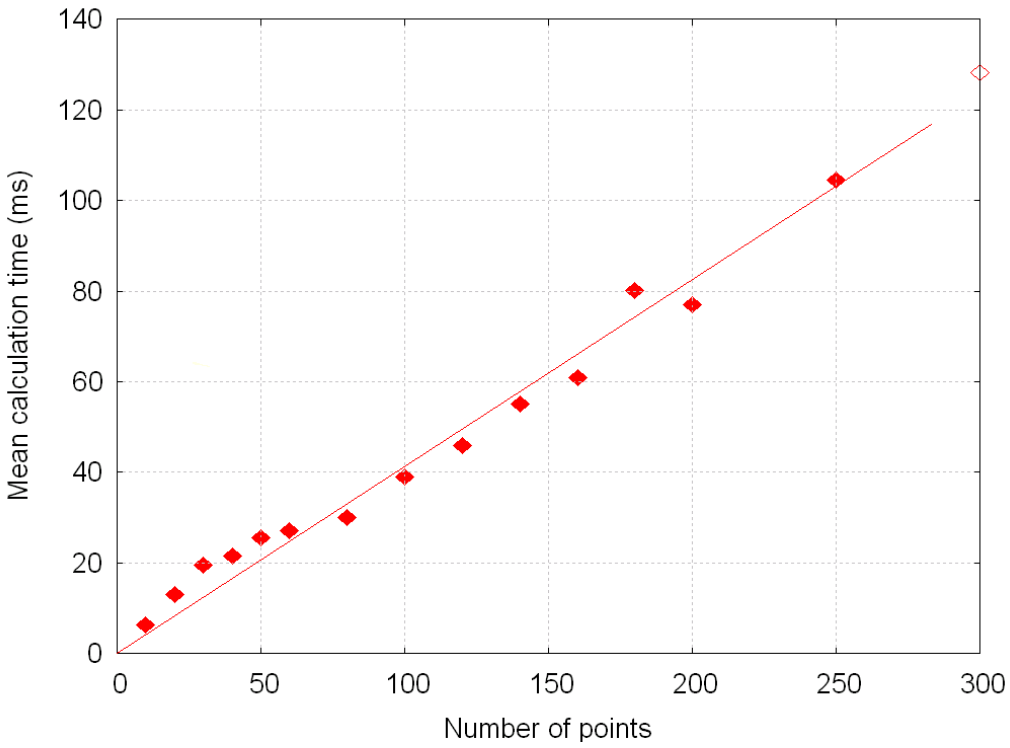

Figure 6.1: Influence of the number of discretization points on the mean calculation time: the calculation time is roughly a linear function of the number of points.

For inverting the Hessian, which is a strongly banded matrix (See Figure 3.18), we use a simple Gauss algorithm slightly modified for taking into account all the zero-elements of the matrix.

Conjugate Gradient Method

The conjugate gradient method uses a family of **H**-conjugated vectors (two vectors **u** and **v** are **H**-conjugated if $\mathbf{u}^T\mathbf{H}\mathbf{v} = 0$ with **H** a symmetric positive definite matrix) to look iteratively for the solution to the system $\mathbf{H}\mathbf{h}+\mathbf{F} = 0$. In the non-linear method, the matrix **H** is recalculated during the process, while the linear method keeps it constant. We use only one iteration of the conjugate gradient method to help getting into the convergence zone of the Newton method.

It is worth noting that the Hessian is not necessarily positive definite. In the cases where it is not, it typically has some very small (in absolute value) negative values. But this is not really a problem, the iteration would not converge to the solution of the equation but they will be of lower energy.

Steepest Descent and Line Search

The steepest descent and the line search use the force as a search direction (The force is the opposite of the gradient, and thus indicate the direction of the steepest descent.) Both methods look for a coefficient α such that $\mathbf{X_{new}} = \mathbf{X_{old}}+\alpha\mathbf{F}$ has a minimum energy. In the case of the steepest descent, the Hessian is used to approximate $E(\alpha) = E(\mathbf{X_{new}}(\alpha))$ by a parabola and finding its minimum, which is found for $\alpha = \frac{\mathbf{F}^T\mathbf{F}}{\mathbf{F}^T\mathbf{H}\mathbf{F}}$. The linear search uses decreasing powers of 2 ($\alpha = \frac{1}{2^k}$) until it finds a solution with a lower energy.

6.2 Dynamical Solution

The system could be easily modified to become a dynamic one, introducing the velocities as supplementary variables and solving the system, for example with the algorithm proposed by Baraff [Baraff and Witkin, 1998]. For ensuring the stability, it would be necessary to add damping forces for each kind of interaction, following the formulation of Baraff: if the constraint is $C(\mathbf{X}) = 0$, the (proportional) energy is $E = \frac{1}{2}kC(\mathbf{X})^TC(\mathbf{X})$, the proportional forces are of the form $-k\frac{\partial C(\mathbf{X})}{\partial \mathbf{X}}C(\mathbf{X})$, the integral forces of the form $-f_I\frac{\partial C(\mathbf{X})}{\partial \mathbf{X}}$ with the update rule $f_{I,new} = f_{I,old} + k_IC(X)$. The damping forces can be expressed as $-k\frac{\partial C(\mathbf{X})}{\partial \mathbf{X}}\dot{C}(\mathbf{X})$. All the interactions except the weight (for which there is no damping force physically speaking) can obviously be rewritten in

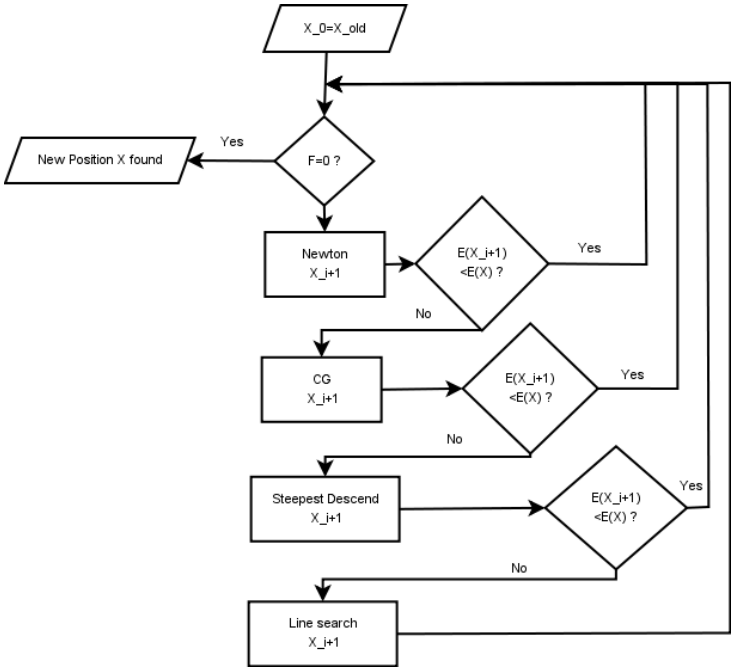

Figure 6.2: Minimization Algorithm

this form. It would be necessary to adapt the values of the constants to the new system.

The Lagrange equations can then be used as evolution equations. With the kinetic energy

$$T = \sum_{i=1}^{n} \frac{1}{2} m \dot{\mathbf{x}}_i^2 + \sum_{i=1}^{n-1} \frac{1}{2} \Omega_i^T \mathbf{J} \Omega_i$$

where $\mathbf{J}$ is the inertial matrix of a cylinder segment in local coordinates,[1] and where $\Omega_i \in \mathbb{R}^3 = 2\overrightarrow{\bar{\mathbf{q}}_i \dot{\mathbf{q}}_i}$ is the vectorial rotational speed in local coordinates and with the Lagrangian

$$L = T - E$$

we obtain, for all coordinates X_i, where the dot represents the time derivative

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{X}_i} - \frac{\partial L}{\partial X_i} = 0$$

In the case of the spatial coordinates, we get the classical result

$$m \ddot{X}_i = F_i$$

[1] $\mathbf{J} = \begin{pmatrix} J_1 & 0 & 0 \\ 0 & J_2 & 0 \\ 0 & 0 & J_1 \end{pmatrix}$ with $J_1 = \frac{1}{12} m(3r^2 + L^2)$ and $J_2 = \frac{mr^2}{2}$

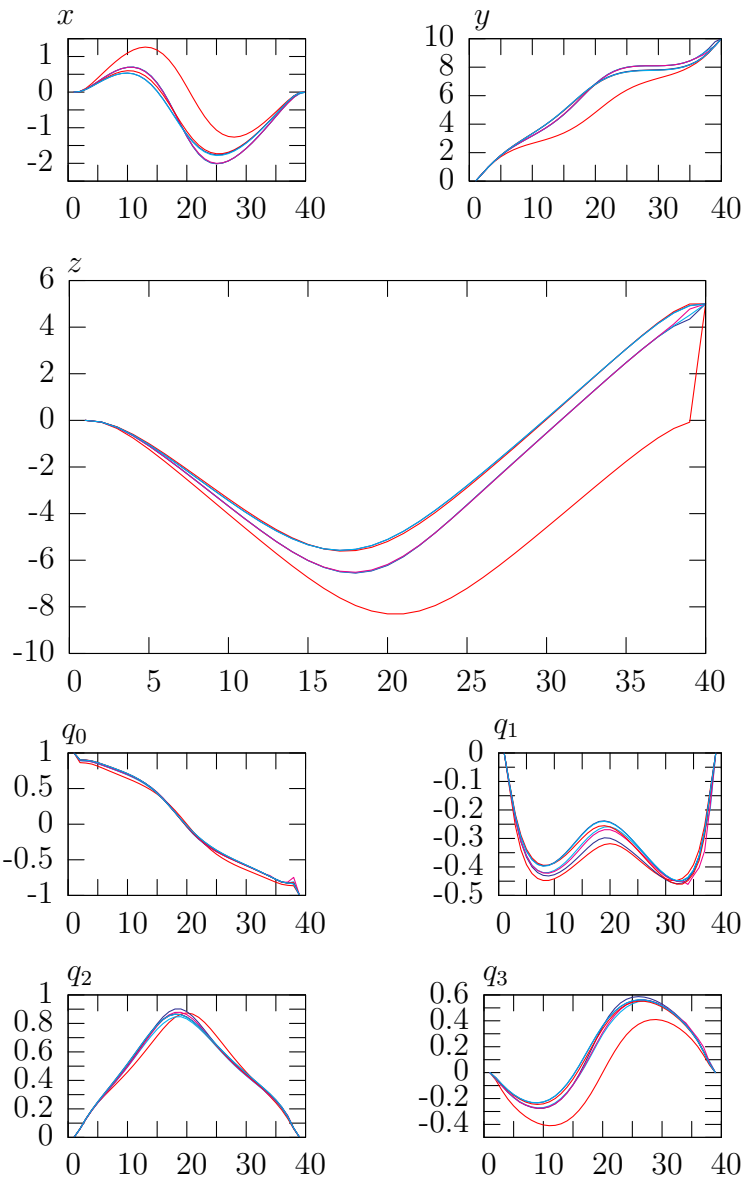

Figure 6.3: Example of the resolution in the case of a big translation along the third coordinate: the different graphics show from top to bottom and from left to right the 7 coordinates along the cable as a function of the indexes of points during the iterative resolution. The starting position (the translation can be seen on the third graphic) as well as the 6 iterations are shown.

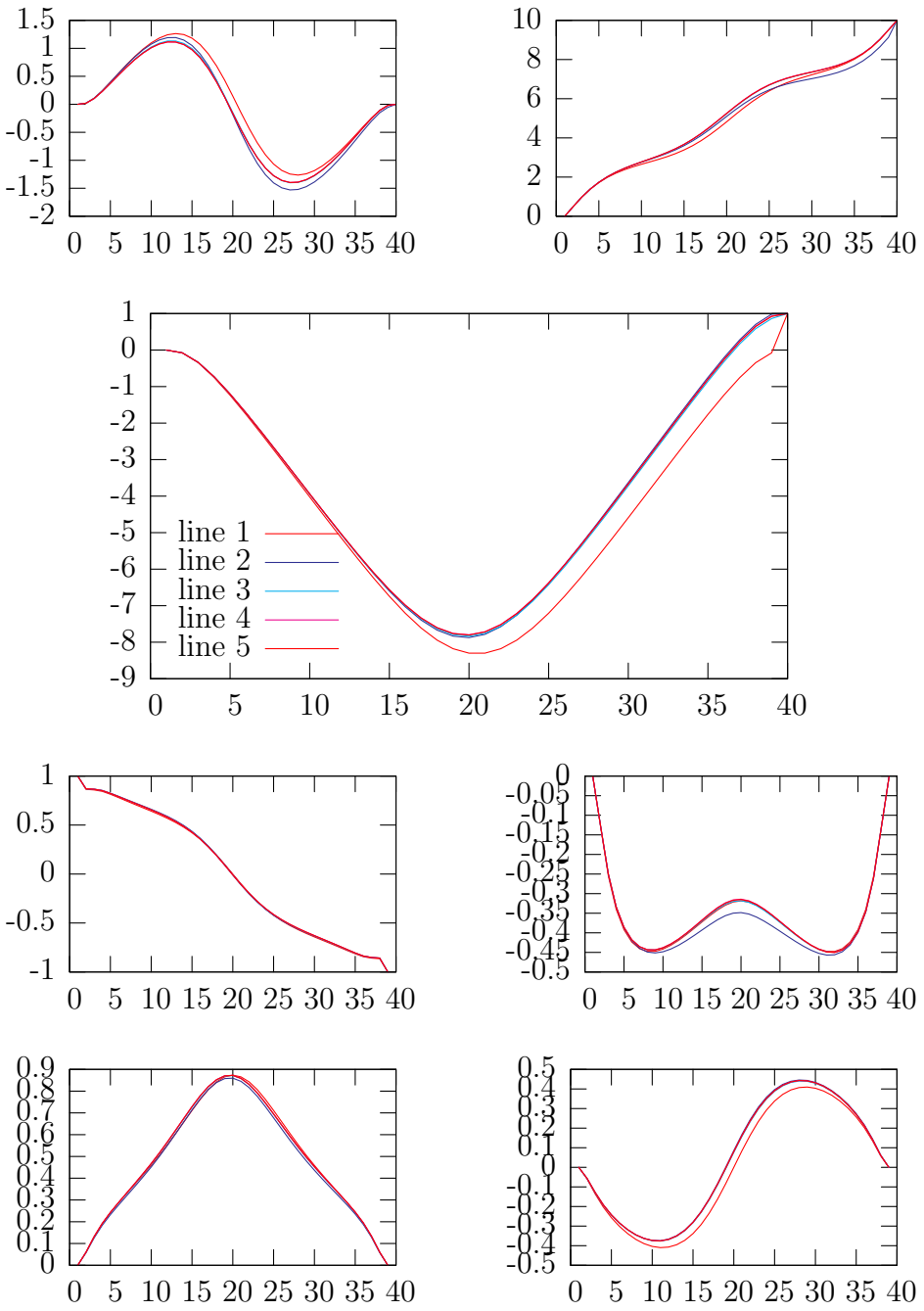

Figure 6.4: Example of the resolution in the case of a smaller translation along the third coordinate: the different graphics show from top to bottom and from left to right the 7 coordinates along the cable as a function of the indexes of points during the iterative resolution. The starting position (the translation can be seen on the third graphic) as well as the 4 iterations are shown.

6.3 Prediction

6.3.1 Linear prediction

In order to accelerate the resolution process, a possibility is to interpolate linearly the modifications and use it as a starting point. The advantage is that the modifications are spread faster along the cable. A linear interpolation is easy to implement and fast to execute, and can speed up the resolution process. Let us consider the case where the position of the cable during the last time step is known; the first end is fixed ($\mathbf{x}_1$ and $\mathfrak{q}_1$ are constant) and the other end is moved. Let us define

$$\mathbf{t} = \mathbf{x}_{n,new} - \mathbf{x}_{n,old}$$

the translation and

$$\mathfrak{r} = \mathfrak{q}_{n-1,new} \cdot \overline{\mathfrak{q}_{n-1,old}}$$

the relative rotation between the old and the new position of the second end. We initialize the search for the solution with the following positions

$$\mathbf{x}_{i,new} = \mathbf{x}_{i,old} + \frac{i-1}{n-1}\mathbf{t}$$

for $i = 1, \ldots, n-1$ and with the following quaternions

$$\mathfrak{q}_{i,new} = \mathfrak{s}^{i-1}\mathbf{q}_{i,old}$$

for $i = 1, \ldots, n$ with, when $\mathfrak{r} = (\cos\frac{\theta}{2}, \sin\frac{\theta}{2})$, $\mathfrak{s} = (\cos\frac{\theta}{2(n-3)}, \sin\frac{\theta}{2(n-3))})$. The solution is then calculated as usual with the energy minimization algorithm. The efficiency of this method depends on the amplitude of the modification. For a small modification, it is a good approximation in the direction of the modification, but the other directions are not modified. For example, a translation taking apart the two end points of a cable results first in a translation of all mass points along the movement direction but also in translations along other directions (the cable would typically "go up" because it has relatively less length for sagging) as well as modification of the orientations. These modifications are not approximated by this scheme.

6.3.2 Approximation by reduction of the number of discretization points

Another possibility is to calculate a solution to the same problem with fewer discretization points and to interpolate then the result. The constants must

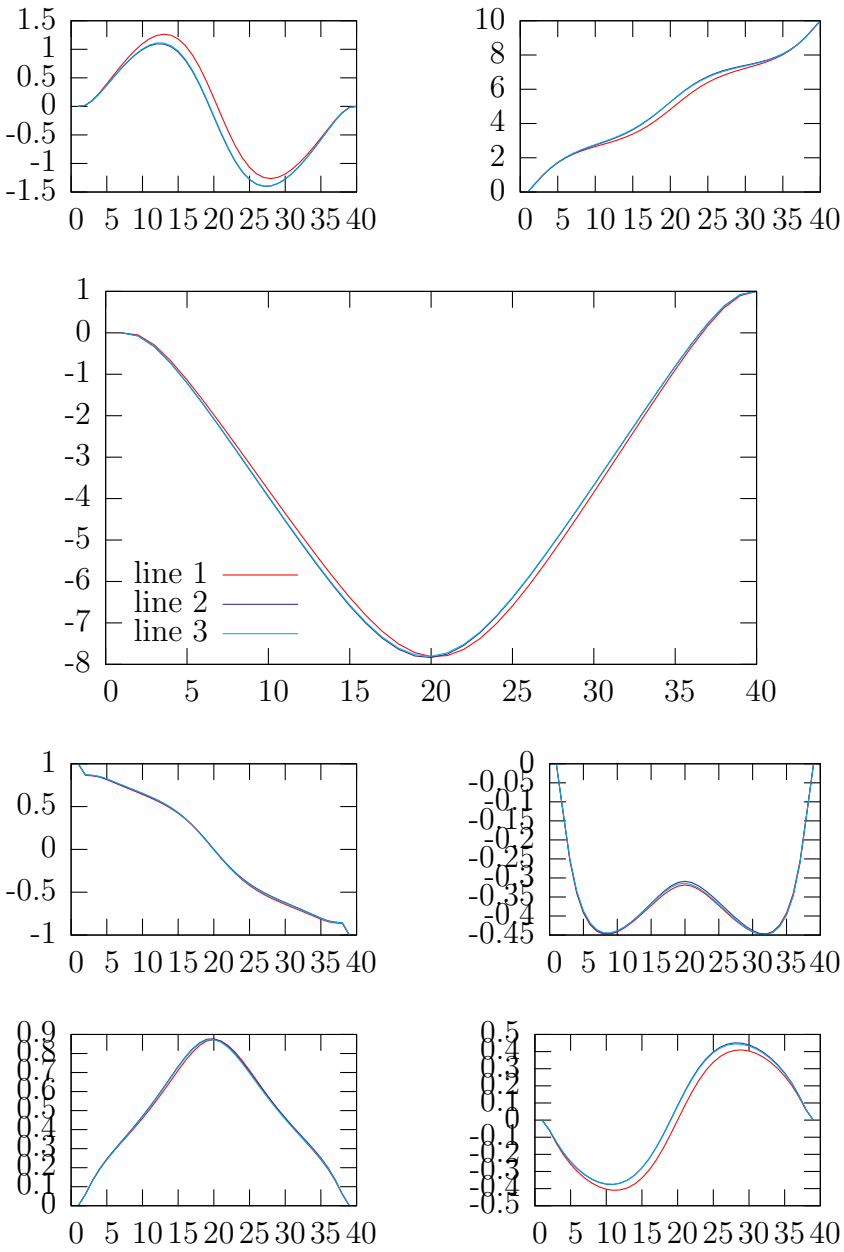

Figure 6.5: Example of the resolution in the case of a translation along the third coordinate with linear interpolation: the different graphics show from top to bottom and from left to right the 7 coordinates along the cable as a function of the indexes of points during the iterative resolution. The starting position (the translation can be seen on the third graphic) as well as the 2 iterations are shown.Compare with the second resolution example (see Figure 6.1.2): fewer iteration are necessary.

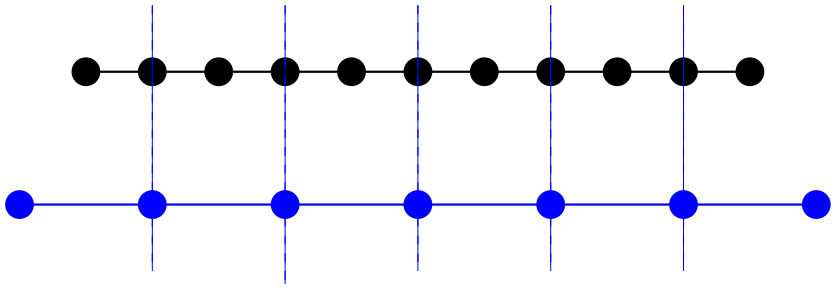

Figure 6.6: The big cable (top, in black) and the small one (bottom, in blue) are shown here for $\alpha = 2$, for $N = 11$ and $n = 7$

then be adapted to ensure that the solution converges to the same form. For avoiding that the first segment displaces the solution, we define the transformation from the "big cable" to the the small one such that the second and the penultimate points of both cables are equal, as well as the first and last quaternions. Let N be the number of points of the big cable, and n of the small one. Let us call α the factor between the number of free segments of the cable:

$$N - 3 = \alpha(n - 3)$$

For example, a factor of $\alpha = 2$ means that we take one point out of two between the two end points.

The length of a segment of the big cable is $\frac{L_{big}}{N-1}$, a segment of the small cable has the length $\alpha\frac{L_{big}}{N-1}$, therefore the length of the small cable is $L - small = \alpha L_{big}\frac{n-1}{N-1}$. The position of the first point of the small cable is $\mathbf{x}_{1,small} = \mathbf{x}_{2,big} - rot(\mathfrak{q}_0)\frac{L_{small}}{n-1}$ and the last point is $\mathbf{x}_{n,small} = \mathbf{x}_{N-1,big} + rot(\mathfrak{q}_{N-1})\frac{L_{small}}{n-1}$.

The point i of the small cable, for $i = 2...N-1$ corresponds to the point $j = \alpha(i-2)+2$ of the big one.. Similarly, a point j of the big cable corresponds to the point $i = \frac{j-2}{\alpha} + 2$ of the small one. A non-integer result indicates that there is no point of the other cable corresponding exactly to the point we are looking for and we need to interpolate.

The advantage of this method is that the movement is interpolated in all the directions. The disadvantage is that it takes more time to calculate.

We cannot interpolate directly the start position of the big cable from the position of the small one, since the lower resolution of the small cable, especially for a low resolution introduces artefacts. The solution is to interpolate the difference between the new and the old position of the cable and to add it to the old position of the big cable and to solve it normally.

However, The gain in term of number of iteration while solving for the big

cable is not obvious when compared to the linear interpolation. A detailled study of the running time in a variety of situations would be necessary.

Chapter 7

Experimental results

7.1 Integration in a Virtual Reality environment

The simulation is integrated in the Virtual Reality software *veo* of Daimler-Chrysler Research and Technology. It provides the whole environment for the simulation, such as graphics, scenes and objects handling. The cable is represented as a sequence of spheres and cylinders. The spheres are set at the discretization points. The cylinders are set along the segments joining two consecutive points. The cylinders and spheres can be moved (in the geometrical limits permitted by the simulation) easily by selecting them and moving them for example with a spacemouse. At each frame, the simulation checks whether the eventually selected objects in the scene correspond to a part of a cable. If yes, its new position and orientation are taken into account, the simulation calculates the new solution and the new positions and orientations of all the spheres and cylinders are passed to the Virtual Reality software for the graphical representation.

A seamless integration into the processes already in place is necessary for the users. The data from the construction - usually in CATIA - are tessellated and converted automatically to the OpenInventor format, used for the Virtual Reality application. We also have an interface that allows to convert all or part of a tessellated wire harness or hose to a flexible one having the same characteristics (like length, radius, position...). A partial conversion is useful since some of the construction parts represent a huge harness that extends through the whole vehicle. The connectivity information of the tessellation triangles is parsed to create a graph representing the interconnections of the vertices. Geometrical considerations allow to find cross-sections on the surface at a given point that we use for cutting the tree at the appropriate

vertices and for calculating the coordinates of points on the centerline, from which we construct the new flexible object. A copy of the tessellation with the remaining points is also created for the graphical representation. The user can also create a complete new cable between two points of the scene.

A graphical user interface groups the commands that are useful for the simulation. The simulation can be turned on or off, handles can be added or removed at selected points or can be set equidistant automatically along the cable, geometrical parameters like the radius or the length and material parameters like the Young's modulus and the Poisson coefficient can be changed interactively, the new centerline of the cables can be exported to CATIA via the VDA-FS (*Verband der Automobileindustrie - Flaechen-Schnittstelle*) format and can then be used as a draft for the construction. Our tool has already been tested by pilot users who have reacted positively to the possibilities offered to them.

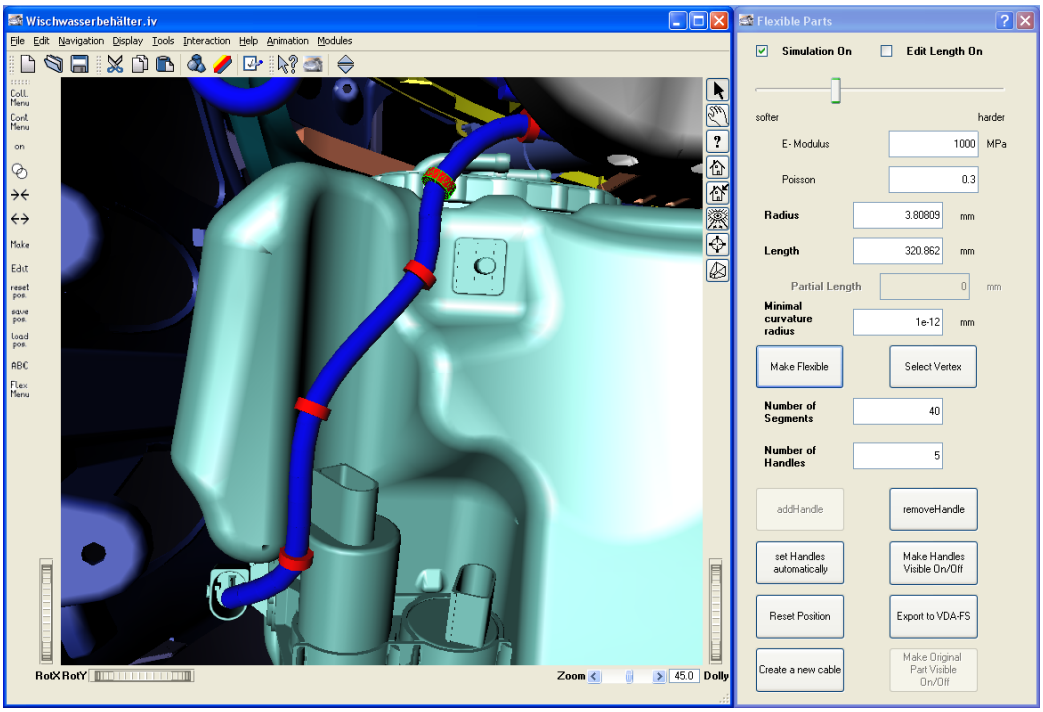

Figure 7.1: The Graphical User Interface

The tool is used in two main contexts: assembly simulations and cable routing. In assembly simulations, flexible parts like cables or hoses are often in the way for mounting another part of the vehicle. The idea is to make the problematic part flexible (which was represented before as a rigid body), push

it out of the way as it could be done with the physical parts and continue with the assembly simulation. For wire routing, a new path for the cable needs either to be defined from scratch and constructed or adapted to changes in the construction of other parts. The length of the cable may need to be adapted to the new form. We have two special modes for this, changing the handling of the stretch forces. In both cases, the integral forces relative to the length are removed. In the first mode, the user chooses two points on the cable between which the segments can be elongated either by pulling them or by typing the new length in a text field in the GUI. The reference length is then replaced either by the actual length of the segment or by the value of the text input. The length can then be changed smoothly from one iteration to another until the user is satisfied with the result. In the second mode, the stretch energy is set to

$$E_{Length,i} = \frac{k_{Length}}{2} L_i^2$$

with a very soft value for k_{Length}. The cable tends to stretch or to become shorter, finding automatically an optimal length for given endpoints and handles, thus allowing the user to route intuitively without worrying about the length. Playing with the ratio of k_{Length} and the bending and torsion stiffnesses make the form of the cable vary from "rather straight" to "rather bend".

In order to simulate cable harnesses, the possibility to join several cables together has been added: the relative position and orientation of the cables at the ramification points are then preserved. These points can either be moved only when they are selected or follow when one of the cables is stretched. There is also the possibility to bind a cable with rigid objects, representing for example connectors or a part to which cables are connected. The connectors can be collision sensitive: the movement of the rigid part is determined by the collisions and by the distance restrictions coming from the cables. This is useful for assembly simulation where the central question is whether the length of the cables hanging from a part is sufficient for allowing it to be built in or out, and if not, how long the cables should be.

7.2 Comparison with the hardware

An important point is how well does the simulation reflect the reality: laser measurements of the deformation of brake hoses in different assembly positions were made. The various geometrical and material parameters were fitted to one of the measured positions. The same material parameters were taken to simulate the other positions. The results can be seen in Figure 7.2.

The good fitting should not hide the fact that real hoses and cables have a very high variability, depending for example on which position of a roll the hoses were stocked during fabrication: the different radii induce a rest curvature which modifies the shape of the hose. Another source of variability are the connectors: a rotation of the connector around the axis of the hose leads to a supplementary torsion. The design engineers count with a variability of one diameter.

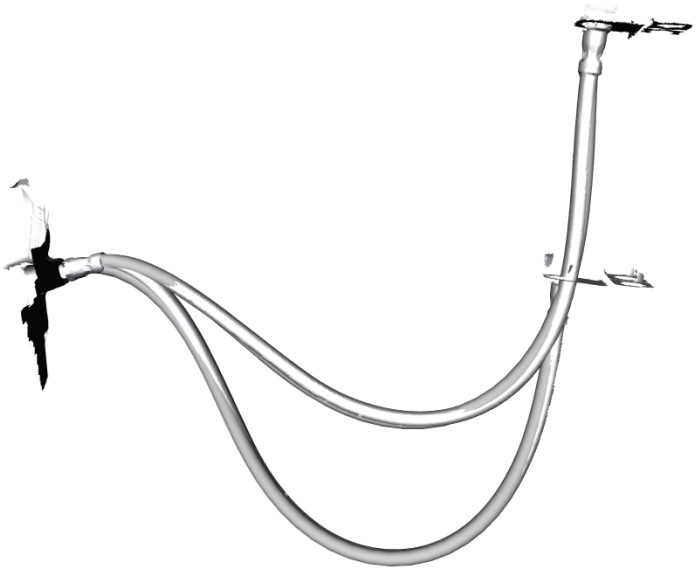

Figure 7.2: Comparison of the simulation with measurements of a brake hose. The simulated hoses are in gray, the measured surface in white. The lower position was used to determine the material parameters.

Chapter 8

Flexibilisation

8.1 Introduction: Objectives, Requirements,...

The components of a car are designed in a CAD system - in our case CATIA - which does not dispose of a cable simulation. The laying of the cable is a long and complicated process, because the cables have to be defined with circles arcs and straight lines. The CAD-files are exported to the virtual reality format (in our case Open Inventor) as tessellated data. For a complete integration in the processes, it is necessary to be able to pass from one format to the other one. In particular, it is interesting to simulate and edit the cables in Virtual Reality that were created from the CAD.

The cables are, as all the other parts exported to VR, modelized as *Indexed Triangle Strip Set*: the external surface is tessellated. The simulation works with another data format. The flexibilisation is then the process of converting the tessellated data to a format that can be used by the simulation. The main step is the extraction of the centerline of the cable. Furthermore, we need to be able to flexibilize only partially a cable: entire cable harnesses covering a whole car are often modelized as a unique part. Since the user will probably be only interested in a small part of it, flexibilizing the whole harness would only lead to an overload of the simulation and to a complicated handling for the user.

8.2 Outline of the algorithm

The tessellated data usually are available as *IndexedTriangleSripSet* in Open-Inventor. It is composed on the one hand of an array of three dimensional coordinates (the vertex with index i (hence the name) has its coordinates at the position i in this array) and on the other hand of a list of triangle

Figure 8.1: A typical cable harness

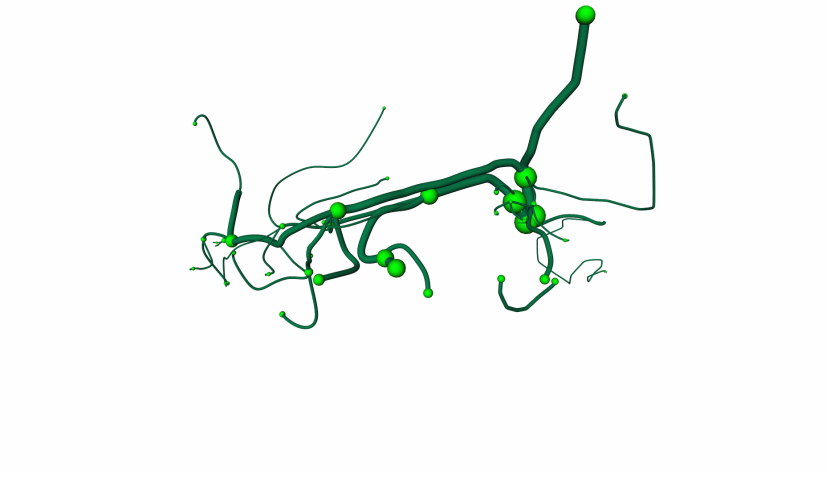

Figure 8.2: After parsing, the green spheres indicate the final and the bifurcation points for an easy selection.

Figure 8.3: The cable harness after partial flexibilisation

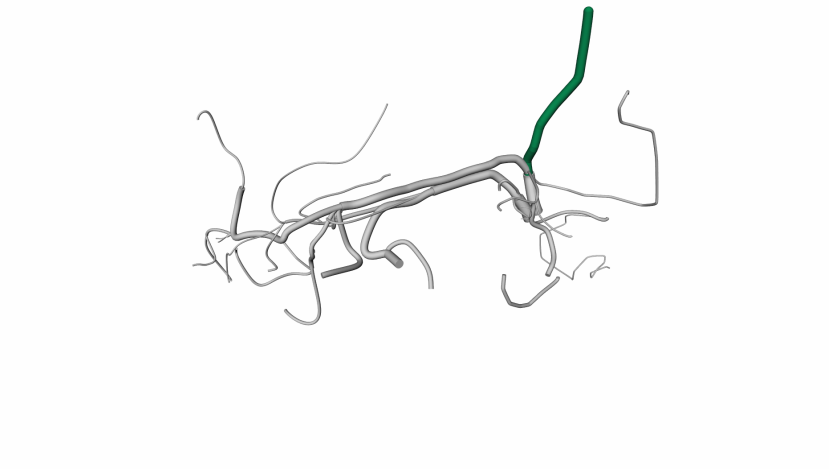

Figure 8.4: The flexibilized part is shown here in green, the non-flexibilized in grey.

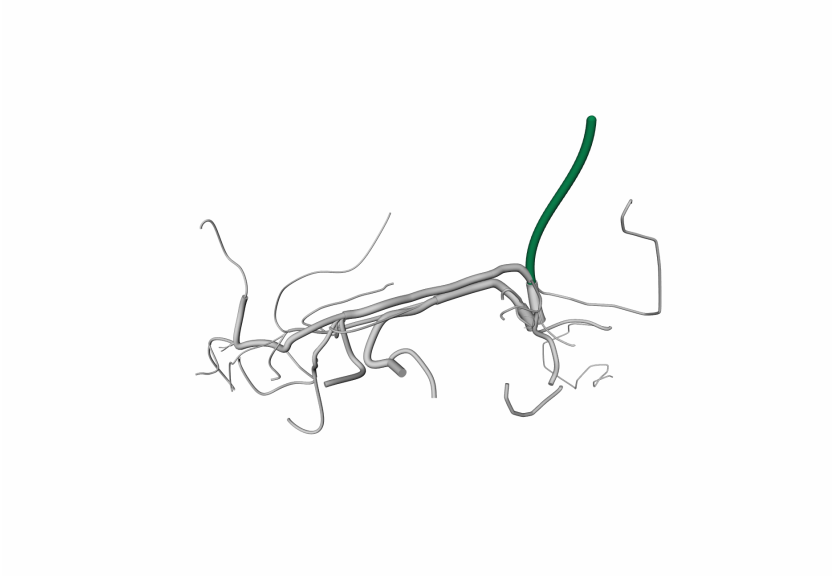

Figure 8.5: After turning the simulation on, the flexibilzed part assumes a more natural form.

strip sets (a slightly compressed way to represent a set of triangles) based on the indexed vertices. The strip set is parsed and transformed into a graph $G = (V, E)$ with V the vertices of the triangle set and E the triangle edges (i.e. two vertices are connected if there exists at least one triangle to which they both belong.) This graph is then the base for the rest of the algorithm. [1]

The vertices are then grouped in so-called circles, which would in an ideal case be the intersection of a plane perpendicular to the centerline (cross-section) and the surface. Since it is not always possible to find a direct connection between vertices lying in the same cross-section, the circles can have a slightly different form ("zig-zag"). Neighborhood information for the circles is saved and used to generate a tree showing the structure of the cable harness. The center line is then calculated by joining the center points of neighboring circles and a new cable is finally created.

[1] It would be also possible to parse an *TriangleStripSet* if we merge the vertices with identical coordinates.

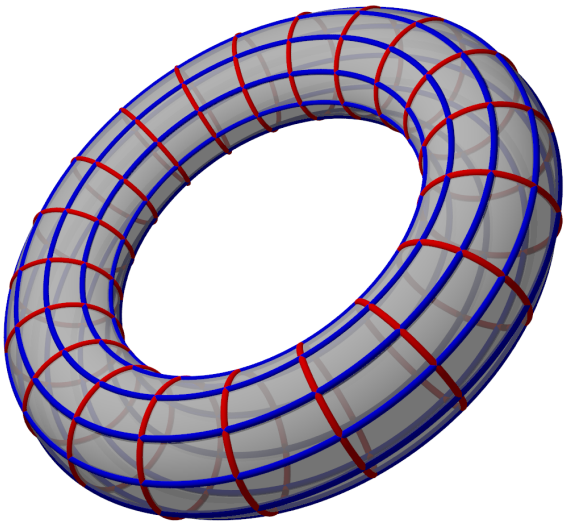

Figure 8.6: Lines of maximal curvature are in red, lines of minimal curvature in blue. The extraction of lines of maximal curvature allows to reconstruct the centerline.

8.3 Algorithm

8.3.1 Prolongation

Let us consider a vertex (the base vertex) B with all the triangles to which it belongs. The other vertices C_i belonging to these triangles form, if the net has no anomalies, a closed chain on the tessellated surface. [2] The base of the algorithm is to look locally on this umbrella for the local lines of minimum and maximum curvature.

We first want to find the 'prolongation' on the umbrella of a vector starting from one of the chain vertices, say C_0 to the base vertex. The prolongation of a vector on a plane is its projection on this plane. If $\mathbf{n}$ is the unitary normal to the plane and $\mathbf{a}$ the vector, its prolongation[3] is

$$p_{plane}(\mathbf{a}) = \mathbf{a} - \mathbf{a}^T\mathbf{nn}$$

[2]The vertices C_i are numbered in such a way that the vertex C_i is connected to the vertices C_{i-1} and C_{i+1}, with the additional condition that $C_0 = C_k$ where k is the number of vertices of the umbrella.

[3]Since we are only interested in the direction of the vector and its prolongation, or in the angle between them, there is no need to norm the vectors.

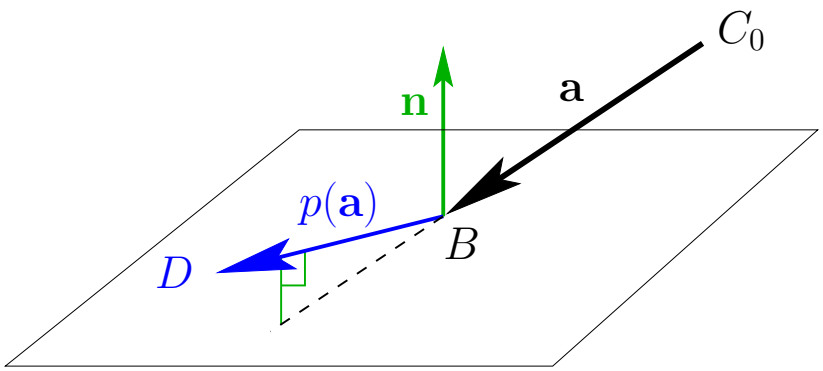

Figure 8.7: The plane prolongation of a vector (in black) is its projection (in blue) on the plane

In our case the plane is defined by two edges $\mathbf{u} = \overrightarrow{BC_i}$ and $\mathbf{v} = \overrightarrow{BC_{i+1}}$. The normal is defined as $\mathbf{n} = \frac{\mathbf{v}\times\mathbf{u}}{\|\mathbf{v}\times\mathbf{u}\|}$. We are only interested in a prolongation that actually lies in the triangle BC_iC_{i+1} defined by the base vertex and the two considered edge vertices. For this, we define the two vectors $\mathbf{b}_u = \mathbf{u} \times \mathbf{n}$ and $\mathbf{b}_v = \mathbf{n} \times \mathbf{v}$. These vectors lie in the plane of the triangle and define two planes passing by the base vertex, which are the limits of a quarter-space for the locus of the point C_0. The condition for the prolongation to lie in the triangle is $(\mathbf{b}_u^T\mathbf{a} \geq 0)\&(\mathbf{b}_v^T\mathbf{a} \geq 0)$.

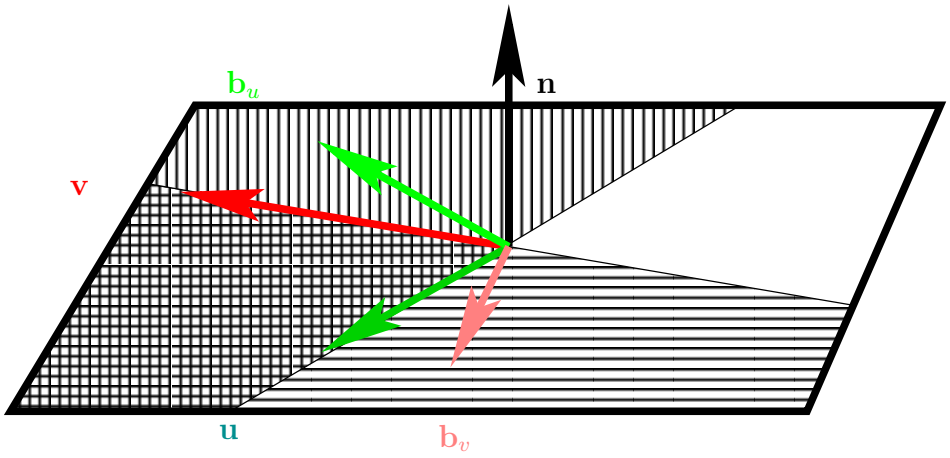

Figure 8.8: The plane prolongation is only valid if the prolongation lies in the face delimited by $\mathbf{u}$ and $\mathbf{v}$. The conditions $(\mathbf{b}_u^T\mathbf{a} \geq 0)$ and $(\mathbf{b}_v^T\mathbf{a} \geq 0)$ define two half spaces whose intersection is the possible locus for C_0

A list with all the possible prolongations for each edge vertex is then

constructed. The prolongations along the edges (i.e. a normalized vector along the edges themselves) are also added. The prolongation whose dot product with the base vector **a** is the bigger (the angle is the flatter) is then chosen as the prolongation of the base vector on the umbrella.

$$p_{umbrella}(\mathbf{a})^T\mathbf{a} = \max_{f \in faces \cup edges} p_f(\mathbf{a})^T\mathbf{a}$$

8.3.2 Circle part

The circle part at one vertex is the line on the net on which a circle would run, normally the line of maximal curvature. It is represented by a pair of edge vertices: the two vertices of the circle framing our base vertex. For finding the circle part at one base vertex, we consider the angles C_iBP_i where C_i is an edge vertex, B the base vertex and P_i lies on the prolongation of C_i. The pair (C_i, P_i) corresponding to the sharpest angle is taken.[4] If the selected pair is not made of two points of the tessellation, the second point is replaced by one of the two points defining the triangle for with the prolongation was defined. We can either take the point C_k such that either the angle P_iBC_k or the angle C_iBC_k is the sharpest possible. It is a min-max problem. We now have for one particular vertex B the line of maximum curvature on the tessellated surface. We now have to follow this line.

8.3.3 Circle

Given two neighboring vertices defining the beginning of a circle, we need a function constructing a circle, which is first represented as a list. As long as the circle does not close on itself (the last vertex only appears once in the list) and the list is not longer than a predetermined value [5] for catching problematic cases, the circle part at the last vertex is calculated. In an ideal case, one of two vertices of the circle part is the penultimate vertex of the circle; the other vertex of the circle part is then added to the circle. If this is not possible, several other mechanisms are used. If the cosine of the part is bigger than 0.99, we have a quasi planar umbrella: the curvature information cannot be reliably extracted. This happens with some tessellations when the faces are subdivided. The algorithm continues then "straight ahead ", adding the vertex corresponding to the prolongation of the vector $V_{i-1}V_i$. If the cosine is greater than 0.98, the direction of the cross section is planar anf

[4] If the cosine of the angle is inferior to 0.3, we have an (almost) right angle. The flattest angle is then taken instead. It is an edge vertex lying at the end of a cable or cable part. The vertex is characterized as such for further processing.

[5] fixed to 40, when the standard tessellation is with 8 to 12 vertices on a circle

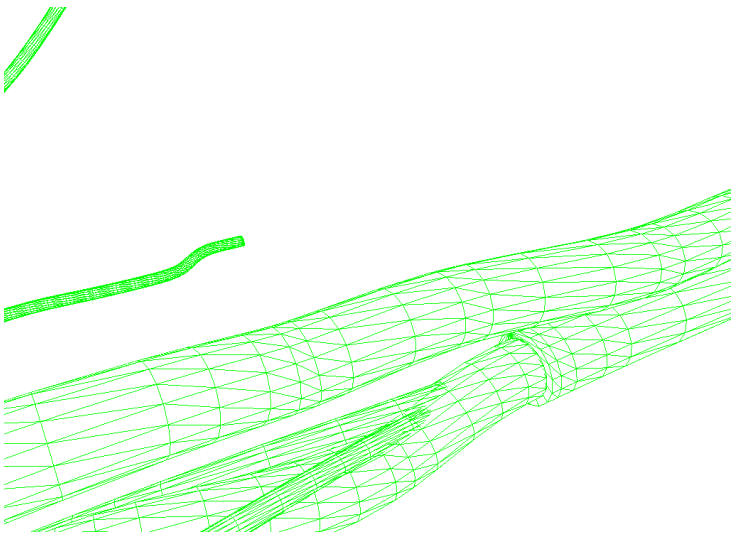

Figure 8.9: A typical triangle net structure, showing several cables and a bifurcation.

the direction of the centerline is slightly bend. We then take as circle part the prolongation with the flattest angle and proceed as before. If all this fails, we take the member of the circle part that is nearest to the prolongation of $V_{i-1}V_i$ (defined with the cosine of the angle $V_{i-1}V_iP_j$)

For creating a circle, we first calculate the circle part at a vertex that does not belong yet to a circle. We create a circle beginning with the chosen base vertex followed by the first element of the circle part. If the first and last elements of the circle are different, we try with the other element. If it is not ideally closed neither, we take the shortest (in number of elements) of the two circles.

If the cosine of the circle part at V_0 is bigger than 0.98, it is a planar point, and the curvature information is not sufficient or reliable for finding a starting direction. In this case, we construct circles for all the edge vertices (circles beginning with V_0R_i). If at least one of the circles is ideal (no supplementary vertices), we take the shortest one. If not, we take the circle with the fewest supplementary vertices. [6]

8.3.4 Cable

We now know how to calculate the circle passing by a particular vertex. It is time to consider the whole cable. As long as there are still vertices

[6] In case where the minimum number of supplementary vertices is reached by two or more circles, we take one of them.

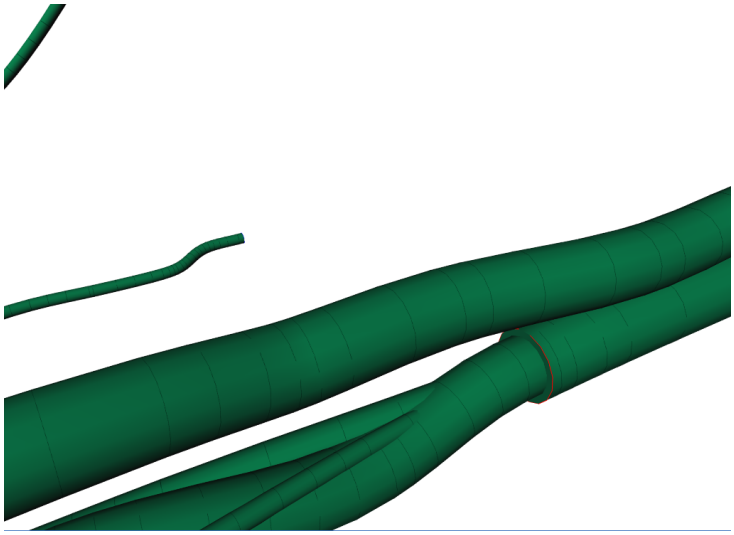

Figure 8.10: A view of the circles on the cable. (They are normaly not represented)

without circles, we take one of those vertices and construct the circle passing by this vertex. We now look for the vertices neighboring this circle (defined as vertices that are neighbors of the main vertices or of the supplementary vertices of the circle) that do not belong yet to a circle. We look for the circle starting in one of those vertices. For every main vertex of the new circle, we look if it already belongs to a circle. If it is a full vertex of the other circle, we fusionate the two circles (simply merging the main and supplementary vertices list of the two circles). If it is a supplementary vertex, we remove it from the supplementary vertices of the second circle. If a supplementary vertex of the new circle is a member of another circle, we remove it from the supplementary vertices. We continue until there is no more neighboring vertex without a circle. The neighborhood relationship between the new circles and the base circle is registered. We now look in the same manner for the neighbors of the new circles. Normally, a circle has two neighbors. The circles at the end of a cable part have only one. The algorithm will usually gives only a new neighbor for each circle, apart from the first one and the one at the end of the cable. The circles begin with the first selected vertex and propagate in two opposite direction until the end of the cable.

We now have covered the whole cable with circles. And we have a graph showing the neighborhood relationship of the circles. But the different parts of a cable harness are often not connected at the tessellation level: each cable part is closed. It is now necessary to use the geometrical proximities of the circles to add connexions between circles that are still not connected. We first

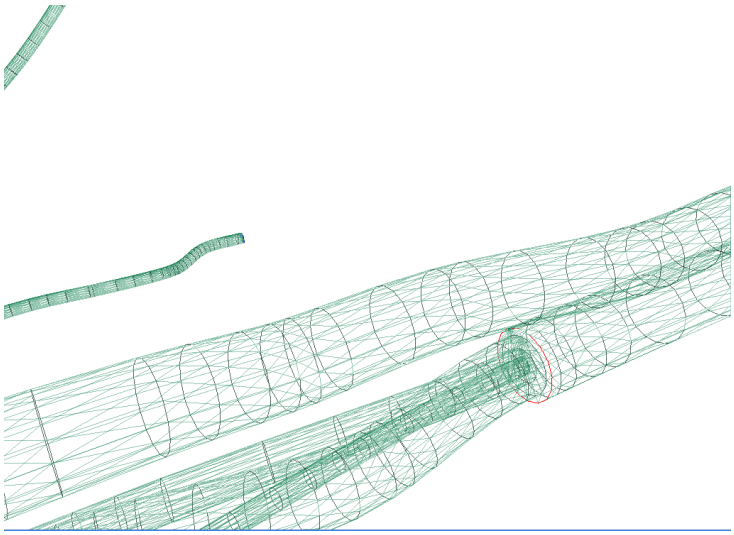

Figure 8.11: A view of the circles on the cable

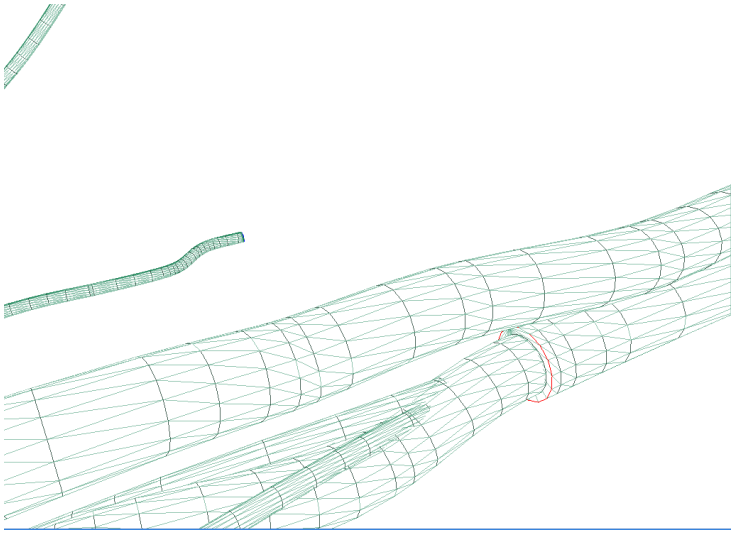

Figure 8.12: A view of the circles on the triangle net. (They are normaly not represented)

look for the end circles (defined as the circles that have only one neighbor). We connect as neighbors end circles sharing their center. To account for numerical errors in the values of the centers, we replace the condition with the one that the two centers must be at a distance less than 1 % of the smallest of the radii. If more than two end circles share their center, we pick the one with the biggest radius as a base; all the other are considered connected to it but not between them (The largest circle is the center of a star.) for avoiding loops in the connection graph. For each of the end circles that can not be connected in this way, we look for the minimal distance between each of them and all the other circles with neighbors. If the end circle is less than 1% of the radius away form the center of the nearest circle, we connect both. After connecting all the circles in this way, the result is a tree (or possibly a forest) showing the structure of the cable harness. For the ease of use, we show spheres at the end circles and at the bifurcation circles; the user can select them for selecting the bifurcation point.

There are two possibilities: either the user may choose to flexibilize the whole cable harness or to flexibilize it only between two points (given by two vertices of the tessellated surface). In the first case, we create cables that go from one end or bifurcation point to another one. In the second case we only take the path between the two circles corresponding to the selected vertices. For finding a path between two circles, we simply go through the graph until, starting with the first circle until we find the second one. We have to be able to detect the case where the two circles are not connected (this could be the case in a complex harness).

For creating the flexibilized cable itself, we linearly interpolate the centerline of the cable (the line passing through the centers of the circles in the path). The quaternion values are calculated to match the segment between two points of the cable: let $\mathbf{u}_{i-1}$ and $\mathbf{u}_i$ be the two unit vectors of the segment $i-1$ and i respectively. Assume we know the quaternion $\mathfrak{q}_{i-1}$ corresponding to the segment $i-1$. Let's define $\mathbf{v} = \mathbf{u}_{i-1} + \mathbf{u}_i$. The norm of this vector is $\|\mathbf{v}\| = 2\cos\frac{\alpha}{2}$ where α is the angle between the two vectors. A rotation of an angle α around the axis $\mathbf{v} \times \mathbf{u}_i$ transforms $\mathbf{u}_{i-1}$ in $\mathbf{u}_i$. The quaternion $(\frac{\|\mathbf{v}\|}{2}, \frac{\mathbf{v}}{\|\mathbf{v}\|} \times \mathbf{u}_i)$ gives us directly the relative rotation corresponding to a minimal torsion. We can easily calculate the successive quaternions, using $(1,0,0,0)$ as $\mathfrak{q}_{-1}$. The radius of the new cable is the mean of the radius of all the circles on the path.

In the case of a partial flexibilization, the triangle net is parsed and the circles are created. The bifurcation points and the ends of the cable are then indicated to the user by means of green spheres to facilitate their selection. The user can then choose freely two points on the surface of the cable. These points are either vertices of the triangle net or one of the spheres. Only the

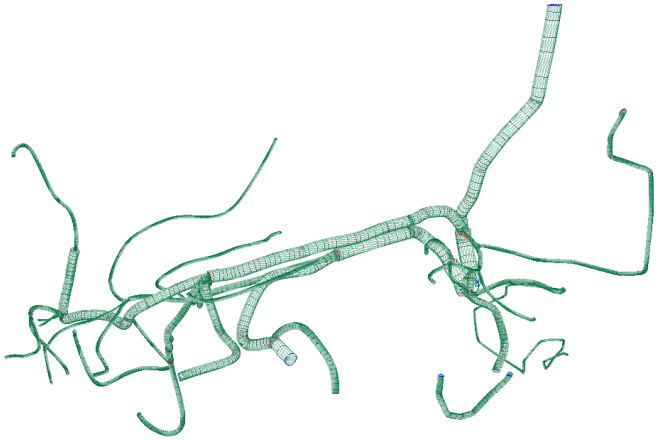

Figure 8.13: A view of the circles on the cable harness. the bifurcation circles are in red, the end circles are in blue.

part of the cable harness between the corresponding circles are then taken as starting and ending points for the flexibilization. The non-flexibilized part of the cableis represented as a triangle net. A new IndexedTriangleStripSet is created; the strip sets which contain vertices that should not be represented are either suppressed, truncated or cut in two. The circles that are in the path of the flexibilized cable have to be cut out. In the case where the end circles of the cable are not edges or if they are edges and they have more than two neighbors, they should be represented and they are taken out of the path.

Chapter 9

Cosserat Surface

9.1 Introduction

A Cosserat surface is a way of modeling thin objects (with one dimension, the thickness, much smaller than the other two). It is modeled by a surface (lying in the middle of the thickness of the object) with a so called "director vector" at each point which represents the direction and stretching of a material fiber along the thickness of the objects. The physical quantities like stretch, shearing, curvature are expressed as a function of the two fundamental forms of the surface: the metric tensor and the curvature tensor.

Let us consider the center surface. It can be parametrized by (u, v). The position of a point is $\mathbf{x}(u, v) \in \mathbb{R}^3$. $\mathbf{a}_1 \in \mathbb{R}^3$ and $\mathbf{a}_2 \in \mathbb{R}^3$ are the two tangents to the surface: $\mathbf{a}_1 = \frac{\partial x}{\partial u}$ and $\mathbf{a}_2 = \frac{\partial x}{\partial v}$. $\mathbf{a}_3 \in \mathbb{R}^3$ is the unitary normal to the surface.

The Cosserat Surface will be modeled by points having 3 own coordinates $\mathbf{x}_i \in \mathbb{R}^3$ plus 3 coordinates for the normal $\mathbf{d}_i \in \mathbb{R}^3$ at that point(or for the director, which we constrain to be along the normal).

We will use a square net with $numberOfVertexes = numberOfLines^2$ vertexes, disposed in lines and columns like a matrix. The initial coordinates of the vertex (i, j) with i and j integers, $1 \leq i \leq numberOfLines$, $1 \leq j \leq numberOfLines$ is then $(iL, jL, 0)$ with $L = 1$ the reference length of a segment. The director is initially along the normal of the surface ($\mathbf{d}_i = (0, 0, 1)$). The vertexes are linked by edges : the vertex (i, j) belongs to 4 edges (apart from the vertexes on the border) with the vertexes $(i-1, j)$,$(i+1, j)$,$(i, j-1)$and $(i, j+1)$ if existent.

We will add handles (fixed points) for avoiding that the surface falls under the influence of the weight and for influencing the form of the surface.

The Cosserat Surface needs the following interactions for being modeled:

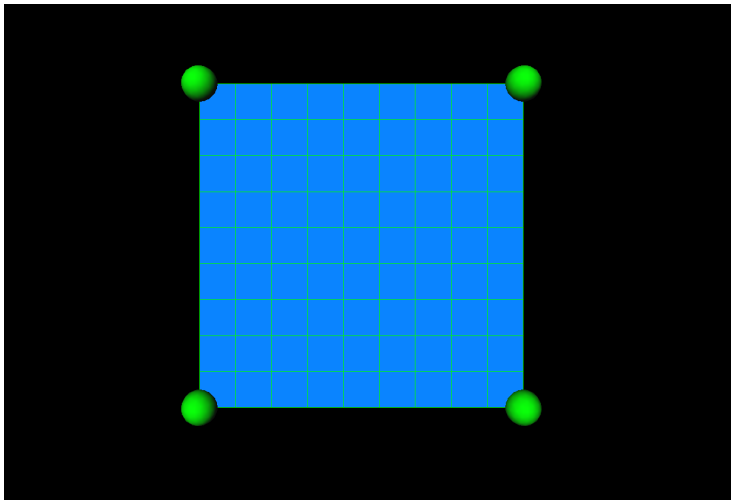

Figure 9.1: Initial net with $numberOfLines$=10. the green spheres are handles

- length conservation for each segment
- norm of normal at each point
- orthogonality of the normal to the 2 tangents at each point
- curvature at each point

We will also consider the weight and the handles.

9.2 Metric Tensor : In-Plane Stretching and Shearing

The metric tensor is given by $a_{\alpha\beta} = \mathbf{a}_\alpha^T \mathbf{a}_\beta$. The diagonal term are a measure for the stretch, the extra-diagonal ones for the shear. The reference value for the tensor is $a_{\alpha\beta} = \delta_\alpha^\beta$.

We have to calculate the tangents from the positions of the vertexes. For symmetry reasons, we will calculate it with the two vertexes surrounding the vertex we are calculating the tangent at: the two unsymmetrical forms $\mathbf{a}_{1+,(i,j)} = \frac{\mathbf{x}_{i+1,j} - \mathbf{x}_{i,j}}{L}$ and $\mathbf{a}_{1-,i,j} = \frac{\mathbf{x}_{i,j} - \mathbf{x}_{i-1,j}}{L}$ are clearly unsatisfying. We choose

$$\mathbf{a}_{1,(i,j)} = \frac{\mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j}}{\|\mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j}\|}$$

and

$$\mathbf{a}_{2,(i,j)} = \frac{\mathbf{x}_{i,j+1} - \mathbf{x}_{i,j-1}}{\|\mathbf{x}_{i,j+1} - \mathbf{x}_{i,j-1}\|}$$

In the case of points on the border, the non-existent vertex is replaced by (i, j) itself.

9.2.1 Length conservation

Since the tangents as defined above have a unitary norm, they cannot be used for the stretch. Instead, we enforce the length conservation on an edge basis. For an edge k relying the vertexes (i_1, j_1 and (i_2, j_2)), the energy is

$$E_{Length,k} = \frac{1}{2} k_{Length} \left(\|\mathbf{x}_{i_2,j_2} - \mathbf{x}_{i_1,j_1}\| - L \right)^2$$

The forces are, with $\mathbf{\Delta x} = \mathbf{x}_{i_2,j_2} - \mathbf{x}_{i_1,j_1} \in \mathbb{R}^3$

$$\begin{aligned} F_{Length,k,\Delta x_p} &= -\frac{\partial E_{Length,k}}{\partial \Delta x_p} \\ &= -\frac{\partial}{\partial \Delta x_p} \left(\frac{1}{2} k_{Length} \left(\|\mathbf{\Delta x}\| - L \right)^2 \right) \\ &= -k_{Length} \frac{\Delta x_p}{\|\mathbf{\Delta x}\|} \left(\|\mathbf{\Delta x}\| - L \right) \\ &= -k_{Length} \Delta x_p \left(1 - \frac{L}{\|\mathbf{\Delta x}\|} \right) \\ &= -k_{Length} \left(\|\mathbf{\Delta x}\| - L \right) u_p \end{aligned}$$

The Hessian terms are

$$\begin{aligned} H_{Length,k,\Delta x_p,\Delta x_q} &= \frac{\partial^2 E_{Length,k}}{\partial \Delta x_p \partial \Delta x_q} \\ &= \frac{\partial}{\partial \Delta x_q} \left(k_{Length} \Delta x_p \left(1 - \frac{L}{\|\mathbf{\Delta x}\|} \right) \right) \\ &= k_{Length} \left(\delta_p^q \left(1 - \frac{L}{\|\mathbf{\Delta x}\|} \right) + \Delta x_p L \left(\frac{\Delta x_q}{\|\mathbf{\Delta x}\|^3} \right) \right) \end{aligned}$$

$$H_{Length,k,\mathbf{\Delta x},\mathbf{\Delta x}} = k_{Length} \left(\left(1 - \frac{L}{\|\mathbf{\Delta x}\|} \right) \mathbf{I}_3 + \left(\frac{L}{\|\mathbf{\Delta x}\|^3} \right) \mathbf{\Delta x} \mathbf{\Delta x}^T \right)$$

$$H_{Length,k,\mathbf{\Delta x},\mathbf{\Delta x}} = k_{Length} \left(\left(1 - \frac{L}{\|\mathbf{\Delta x}\|} \right) \mathbf{I}_3 + \left(\frac{L}{\|\mathbf{\Delta x}\|} \right) \mathbf{u} \mathbf{u}^T \right)$$

9.2.2 Shear

In order to keep the two tangents orthogonal, we define for each vertex (i, j)

$$E_{Shear,k} = \frac{1}{2} k_{Shear} (\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)})^2$$

The forces are then, if $\mathbf{a}_{1,(i,j)}$ has been calculated from $\mathbf{\Delta x} = \mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j} \in \mathbb{R}^3$

$$
\begin{aligned}
F_{Shear,(i,j),\Delta x_p} &= -\frac{\partial E_{Shear,(i,j)}}{\partial \Delta x_p} \\
&= -k_{Shear} \frac{\partial}{\partial \Delta x_p} (\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)})(\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \\
&= -k_{Shear} \mathbf{a}_{2,(i,j)}^T \frac{\partial \mathbf{a}_{1,(i,j)}}{\partial \Delta x_p} (\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \\
&= -k_{Shear} \mathbf{a}_{2,(i,j)}^T \frac{\partial \frac{\mathbf{\Delta x}}{\|\mathbf{\Delta x}\|}}{\partial \Delta x_p} (\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \\
&= -k_{Shear} \frac{1}{\|\mathbf{\Delta x}\|} \mathbf{a}_{2,(i,j)}^T (\mathbf{e}_p - \frac{\Delta x_p \mathbf{\Delta x}}{\|\mathbf{\Delta x}\|^2})(\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \\
&= -k_{Shear} \frac{1}{\|\mathbf{\Delta x}\|} \mathbf{a}_{2,(i,j)}^T (\mathbf{e}_p - a_{1,(i,j),p} \mathbf{a}_{1,(i,j)})(\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \\
&= -k_{Shear} \frac{1}{\|\mathbf{\Delta x}\|} (a_{2,(i,j),p} - a_{1,(i,j),p} \mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)})(\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)})
\end{aligned}
$$

The forces due to the variation of $\mathbf{a}_{2,(i,j)}$ are similar.

The Hessian terms are

$$
\begin{aligned}
H_{Shear,(i,j),\Delta x_p,\Delta x_q} &= \frac{\partial^2 E_{Shear,(i,j)}}{\partial \Delta x_p \partial \Delta x_q} \\
&= \frac{\partial}{\partial \Delta x_q} \left(k_{Shear} \frac{1}{\|\mathbf{\Delta x}\|} (a_{2,(i,j),p} - a_{1,(i,j),p} \mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)})(\mathbf{a}_{1,(i,j)}^T \mathbf{a}_{2,(i,j)}) \right)
\end{aligned}
$$

We have

$\frac{\partial}{\partial \Delta x_q}\left(\frac{1}{\|\mathbf{\Delta x}\|}\right) = -\frac{\Delta x_q}{\|\mathbf{\Delta x}\|^3} = -\frac{a_{1,(i,j),q}}{\|\mathbf{\Delta x}\|^2}$
$\frac{\partial}{\partial \Delta x_q}\mathbf{a}_{1,(i,j)} = \frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_q - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)})$
$\frac{\partial}{\partial \Delta x_q}\left(a_{2,(i,j),p} - a_{1,(i,j),p}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}\right)$
$= -\frac{1}{\|\mathbf{\Delta x}\|}(\delta_p^q - a_{1,(i,j),q}a_{1,(i,j),p})\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)} - a_{1,(i,j),p}\frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_q - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)})^T\mathbf{a}_{2,(i,j)}$
$= -\frac{1}{\|\mathbf{\Delta x}\|}((\delta_p^q - a_{1,(i,j),q}a_{1,(i,j),p})\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)} + a_{1,(i,j),p}(a_{2,(i,j),q} - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}))$
$= -\frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}(\delta_p^q - 2a_{1,(i,j),q}a_{1,(i,j),p}) + a_{1,(i,j),p}a_{2,(i,j),q})$
$\frac{\partial}{\partial \Delta x_q}\left(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}\right) = \frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_q - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)})^T\mathbf{a}_{2,(i,j)}$
$= \frac{1}{\|\mathbf{\Delta x}\|}(a_{2,(i,j),q} - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})$
$\frac{\partial}{\partial \Delta x_q}\left((a_{2,(i,j),p} - a_{1,(i,j),p}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})\right)$
$= -\frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}(\delta_p^q - 2a_{1,(i,j),q}a_{1,(i,j),p}) + a_{1,(i,j),p}a_{2,(i,j),q})(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})$
$+(a_{2,(i,j),p} + a_{1,(i,j),p}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})\frac{1}{\|\mathbf{\Delta x}\|}(a_{2,(i,j),q} - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})$
$= \frac{1}{\|\mathbf{\Delta x}\|}(-(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}(\delta_p^q - 2a_{1,(i,j),q}a_{1,(i,j),p}) + a_{1,(i,j),p}a_{2,(i,j),q})(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})$
$+(a_{2,(i,j),p} + a_{1,(i,j),p}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})(a_{2,(i,j),q} - a_{1,(i,j),q}\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})$
$= \frac{1}{\|\mathbf{\Delta x}\|}(a_{2,(i,j),p}a_{2,(i,j),q}$
$+\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}(-a_{1,(i,j),p}a_{2,(i,j),q} + a_{1,(i,j),p}a_{2,(i,j),q} - a_{2,(i,j),p}a_{1,(i,j),q})$
$-(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})^2(\delta_p^q - 2a_{1,(i,j),q}a_{1,(i,j),p} + a_{1,(i,j),p}a_{1,(i,j),q})$
$= \frac{1}{\|\mathbf{\Delta x}\|}(a_{2,(i,j),p}a_{2,(i,j),q}$
$+\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)}(-a_{2,(i,j),p}a_{1,(i,j),q})$
$-(\mathbf{a}_{1,(i,j)}^T\mathbf{a}_{2,(i,j)})^2(\delta_p^q - a_{1,(i,j),q}a_{1,(i,j),p}))$

9.2.3 simplified Shear

In order to keep the two tangents orthogonal, we define for each vertex (i, j)

$$E_{Shear,k} = \frac{1}{2}k_{Shear}(\mathbf{\Delta x}^T\mathbf{\Delta y})^2$$

where

$$\mathbf{a}_{1,(i,j)} = \frac{\mathbf{\Delta x}}{\|\mathbf{\Delta x}\|}$$

and

$$\mathbf{a}_{2,(i,j)} = \frac{\mathbf{\Delta y}}{\|\mathbf{\Delta y}\|}$$

The forces are then, if $\mathbf{a}_{1,(i,j)}$ has been calculated from $\mathbf{\Delta x} = \mathbf{x}_{i+1,j} -$

$\mathbf{x}_{i-1,j} \in \mathbb{R}^3$

$$\begin{aligned}
F_{Shear,(i,j),\Delta x_p} &= -\frac{\partial E_{Shear,(i,j)}}{\partial \Delta x_p} \\
&= -k_{Shear}\frac{\partial}{\partial \Delta x_p}(\mathbf{\Delta x}^T\mathbf{\Delta y})^2 \\
&= -k_{Shear}(\mathbf{e}_p^T\mathbf{\Delta y})(\mathbf{\Delta x}^T\mathbf{\Delta y}) \\
&= -k_{Shear}\Delta y_p(\mathbf{\Delta x}^T\mathbf{\Delta y}) \\
F_{Shear,(i,j),\Delta y_p} &= -k_{Shear}\Delta y_p(\mathbf{\Delta x}^T\mathbf{\Delta y})
\end{aligned}$$

The terms of the Hessian are

$$\begin{aligned}
H_{Shear,(i,j),\Delta x_p,\Delta x_q} &= \frac{\partial^2 E_{Shear,(i,j)}}{\partial \Delta x_p \partial \Delta x_q} \\
&= \frac{\partial}{\partial \Delta x_q}\left(k_{Shear}\Delta y_p(\mathbf{\Delta x}^T\mathbf{\Delta y})\right) \\
&= k_{Shear}\left(\Delta y_p(\mathbf{e}_q^T\mathbf{\Delta y})\right) \\
&= k_{Shear}\Delta y_p \Delta y_q \\
\mathbf{H}_{Shear,(i,j),\mathbf{\Delta x},\mathbf{\Delta x}} &= k_{Shear}\Delta y \Delta y^T
\end{aligned}$$

$$\begin{aligned}
H_{Shear,(i,j),\Delta y_p,\Delta y_q} &= k_{Shear}\Delta x_p \Delta x_q \\
\mathbf{H}_{Shear,(i,j),\mathbf{\Delta y},\mathbf{\Delta y}} &= k_{Shear}\Delta x \Delta x^T
\end{aligned}$$

$$\begin{aligned}
H_{Shear,(i,j),\Delta x_p,\Delta y_q} &= \frac{\partial^2 E_{Shear,(i,j)}}{\partial \Delta x_p \partial \Delta y_q} \\
&= \frac{\partial}{\partial \Delta y_q}\left(k_{Shear}\Delta y_p(\mathbf{\Delta x}^T\mathbf{\Delta y})\right) \\
&= k_{Shear}\left(\delta_p^q\mathbf{\Delta x}^T\mathbf{\Delta y} + \Delta y_p\mathbf{\Delta x}^T\mathbf{e}_q)\right) \\
&= k_{Shear}\left(\delta_p^q\mathbf{\Delta x}^T\mathbf{\Delta y} + \Delta y_p\Delta x_q\right) \\
\mathbf{H}_{Shear,(i,j),\mathbf{\Delta x},\mathbf{\Delta y}} &= k_{Shear}\left(\mathbf{\Delta x}^T\mathbf{\Delta y}\mathbf{I}_3 + \mathbf{\Delta y}\mathbf{\Delta x}^T\right)
\end{aligned}$$

9.2.4 Shear with additional edges

For modeling the shear in an easier way, we add diagonal edges. The are then submitted to length constraints (with another constant). If α is the angle between two edges, the length of the corresponding diagonal is $2L\sin\frac{\alpha}{2}$. The reference length is then $2L\sin\frac{\pi}{2}$. The energy for an edge is $\frac{1}{2}k4L^2(|\sin\frac{\alpha}{2}| - \sin\frac{\pi}{4})^2$. Considering the different edges around a point, the energy is a function of $(|\sin\frac{\alpha}{2}| - \sin\frac{\pi}{4})^2 + (|\cos\frac{\alpha}{2}| - \cos\frac{\pi}{4})^2 = 4\sin(\frac{\alpha}{4} - \frac{\pi}{8})^2$ (if the sines and cosines are positive)

9.3 Director: Thickness and Out-of-Plane Shear

An absence of out-of-plane shear means that the director is perpendicular to the two tangents. A constant thickness means that the norm of the director is equal to 1.

9.3.1 Norm of the director

The energy is

$$E_{Norm,i} = \frac{1}{2} k_{Norm} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1)^2$$

and the force is

$$\begin{aligned}
F_{Norm,(i,j),d_{(i,j),p}} &= -\frac{\partial E_{Norm,(i,j)}}{\partial d_{(i,j),p}} \\
&= -\frac{\partial}{\partial d_{(i,j),p}} \left(\tfrac{1}{2} k_{Norm} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1)^2 \right) \\
&= -k_{Norm} \frac{\partial \mathbf{d}_{(i,j)}}{\partial d_{(i,j),p}}^T \mathbf{d}_{(i,j)} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) \\
&= -k_{Norm} \mathbf{e}_p^T \mathbf{d}_{(i,j)} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) \\
&= -k_{Norm} d_{(i,j),p} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1)
\end{aligned}$$

The Hessian terms are

$$\begin{aligned}
H_{Norm,(i,j),d_{(i,j),p},d_{(i,j),q}} &= \frac{\partial^2 E_{Norm,(i,j)}}{\partial d_{(i,j),p} \partial d_{(i,j),q}} \\
&= \frac{\partial}{\partial d_{(i,j),q}} \left(k_{Norm} d_{(i,j),p} (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) \right) \\
&= k_{Norm} \left(\delta_p^q (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) + d_{(i,j),p} (2\mathbf{e}_q^T \mathbf{d}_{(i,j)}) \right) \\
&= k_{Norm} \left(\delta_p^q (\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) + 2 d_{(i,j),p} d_{(i,j),q} \right)
\end{aligned}$$

$$H_{Norm,(i,j),\mathbf{d}_{(i,j)},\mathbf{d}_{(i,j)}} = k_{Norm} \left((\mathbf{d}_{(i,j)}^T \mathbf{d}_{(i,j)} - 1) \mathbf{I}_3 + 2 \mathbf{d}_{(i,j)} \mathbf{d}_{(i,j)}^T \right)$$

9.3.2 Orthogonality of the director

The energy is

$$E_{Orth,(i,j)} = \frac{1}{2} k_{Orth} ((\mathbf{d}_i^T \mathbf{a}_1)^2 + (\mathbf{d}_i^T \mathbf{a}_2)^2)$$

and the forces with if $\mathbf{a}_{1,(i,j)}$ has been calculated from $\mathbf{\Delta x} = \mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j} \in \mathbb{R}^3$

$$\begin{aligned}
F_{Orth,(i,j),\Delta x_p} &= -\frac{\partial E_{Orth,(i,j)}}{\partial \Delta x_p} \\
&= -\tfrac{1}{2} k_{Orth} \frac{\partial}{\partial \Delta x_p} ((\mathbf{d}_i^T \mathbf{a}_1)^2 + (\mathbf{d}_i^T \mathbf{a}_2)^2) \\
&= -k_{Orth} \mathbf{d}_{(i,j)}^T \frac{\partial \mathbf{a}_{1,(i,j)}}{\partial \Delta x_p} (\mathbf{a}_{1,(i,j)}^T \mathbf{d}_{(i,j)}) \\
&= -k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} \mathbf{d}_{(i,j)}^T (\mathbf{e}_p - a_{1,(i,j),p} \mathbf{a}_{1,(i,j)}) (\mathbf{a}_{1,(i,j)}^T \mathbf{d}_{(i,j)}) \\
&= -k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} (d_{(i,j),p} - a_{1,(i,j),p} \mathbf{a}_{1,(i,j)}^T \mathbf{d}_{(i,j)}) (\mathbf{a}_{1,(i,j)}^T \mathbf{d}_{(i,j)})
\end{aligned}$$

The derivative for $\mathbf{a}_{2,(i,j)}$ is similar.

$$
\begin{aligned}
F_{Orth,(i,j),d_{(i,j),p}} &= -\frac{\partial E_{Orth,(i,j)}}{\partial d_{(i,j),p}} \\
&= -\tfrac{1}{2} k_{Orth} \frac{\partial}{\partial d_{(i,j),p}} ((\mathbf{d}_i^T \mathbf{a}_1)^2 + (\mathbf{d}_i^T \mathbf{a}_2)^2) \\
&= -k_{Orth} \frac{\partial \mathbf{d}}{\partial d_{(i,j),p}}^T (\mathbf{a}_1 (\mathbf{d}_i^T \mathbf{a}_1) + \mathbf{a}_2 (\mathbf{d}_i^T \mathbf{a}_2)) \\
&= -k_{Orth} \mathbf{e}_p^T (\mathbf{a}_1 (\mathbf{d}_i^T \mathbf{a}_1) + \mathbf{a}_2 (\mathbf{d}_i^T \mathbf{a}_2)) \\
&= -k_{Orth} (a_{1,(i,j),p} (\mathbf{d}_i^T \mathbf{a}_1) + a_{2,(i,j),p} (\mathbf{d}_i^T \mathbf{a}_2))
\end{aligned}
$$

Let's calculate the Hessian terms for the first term $E_{Orth,(i,j)} = \frac{1}{2} k_{Orth} (\mathbf{d}_i^T \mathbf{a}_1)^2$

$$
\begin{aligned}
H_{Orth,(i,j),d_{(i,j),p},d_{(i,j),q}} &= \frac{\partial^2 E_{Orth,(i,j)}}{\partial d_{(i,j),p} \partial d_{(i,j),q}} \\
&= \frac{\partial}{\partial d_{(i,j),q}} \left(k_{Orth} a_{1,(i,j),p} (\mathbf{d}_i^T \mathbf{a}_1) \right) \\
&= k_{Orth} a_{1,(i,j),p} a_{1,(i,j),q} \\
\mathbf{H}_{Orth,(i,j),d_{(i,j)},d_{(i,j)}} &= k_{Orth} \mathbf{a}_1 \mathbf{a}_1^T
\end{aligned}
$$

$$
\begin{aligned}
H_{Orth,(i,j),\Delta x_{1,(i,j),p},d_{(i,j),q}} &= \frac{\partial^2 E_{Orth,(i,j)}}{\partial \Delta x_{1,(i,j),p} \partial d_{(i,j),q}} \\
&= \frac{\partial}{\partial d_q} \left(k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} (d_p - a_{1,p} \mathbf{a}_1^T \mathbf{d}) (\mathbf{a}_1^T \mathbf{d}) \right) \\
&= k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} \left((\delta_p^q - a_{1,p} \mathbf{a}_1^T \mathbf{e}_q) (\mathbf{a}_1^T \mathbf{d}) + (d_p - a_{1,p} \mathbf{a}_1^T \mathbf{d}) (\mathbf{a}_1^T \mathbf{e}_q) \right) \\
&= k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} \left((\delta_p^q - a_{1,p} a_{1,q}) (\mathbf{a}_1^T \mathbf{d}) + (d_p - a_{1,p} \mathbf{a}_1^T \mathbf{d}) (a_{1,q}) \right) \\
&= k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} \left((\delta_p^q - 2 a_{1,p} a_{1,q}) (\mathbf{a}_1^T \mathbf{d}) + d_p a_{1,q} \right) \\
\mathbf{H}_{Orth,(i,j),d_{(i,j)},\Delta x_{1,(i,j)}} &= k_{Orth} \frac{1}{\|\mathbf{\Delta x}\|} \left((\mathbf{I}_3 - 2 \mathbf{a}_1 \mathbf{a}_1^T) (\mathbf{a}_1^T \mathbf{d}) + \mathbf{d} \mathbf{a}_1^T \right)
\end{aligned}
$$

$$
\begin{aligned}
H_{Orth,(i,j),\Delta x_{(i,j),p},\Delta x_{(i,j),q}} &= \frac{\partial^2 E_{Orth,(i,j)}}{\partial \Delta x_{(i,j),p} \partial \Delta x_{(i,j),q}} \\
&= \frac{\partial}{\partial \Delta x_q}\left(k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|}(d_{(i,j),p} - a_{1,(i,j),p}\mathbf{a}_{1,(i,j)}^T\mathbf{d}_{(i,j)})(\mathbf{a}_{1,(i,j)}^T\mathbf{d}_{(i,j)})\right) \\
&= \frac{\partial}{\partial \Delta x_q}\left(k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(\mathbf{a}_1^T\mathbf{d})\right)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial \Delta x_q}\mathbf{a}_1 &= \frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_q - a_{1,q}\mathbf{a}_1) \\
\frac{\partial}{\partial \Delta x_q}a_{1,p} &= \frac{1}{\|\mathbf{\Delta x}\|}(\delta_p^q - a_{1,q}a_{1,p}) \\
\frac{\partial}{\partial \Delta x_q}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d}) &= -\frac{1}{\|\mathbf{\Delta x}\|}(\delta_p^q - a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} - a_{1,p}\frac{1}{\|\mathbf{\Delta x}\|}(\mathbf{e}_q - a_{1,q}\mathbf{a}_1)^T\mathbf{d} \\
&= -\frac{1}{\|\mathbf{\Delta x}\|}\left((\delta_p^q - a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} + a_{1,p}(d_q - a_{1,q}\mathbf{a}_1^T\mathbf{d})\right) \\
&= -\frac{1}{\|\mathbf{\Delta x}\|}\left((\delta_p^q - 2a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} + a_{1,p}d_q\right) \\
\frac{\partial}{\partial \Delta x_q}\mathbf{a}_1^T\mathbf{d} &= \frac{1}{\|\mathbf{\Delta x}\|}(d_q - a_{1,q}\mathbf{a}_1^T\mathbf{d}) \\
\frac{\partial}{\partial \Delta x_q}\|\mathbf{\Delta x}\| &= \frac{\Delta x_q}{\|\mathbf{\Delta x}\|} \\
\frac{\partial}{\partial \Delta x_q}\frac{1}{\|\mathbf{\Delta x}\|} &= -\frac{\Delta x_q}{\|\mathbf{\Delta x}\|^3}
\end{aligned}
$$

$$
\begin{aligned}
H_{Orth,(i,j),\Delta x_{(i,j),p},\Delta x_{(i,j),q}} &= -k_{Orth}\frac{\Delta x_q}{\|\mathbf{\Delta x}\|^3}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(\mathbf{a}_1^T\mathbf{d}) \\
&\quad -k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|}\frac{1}{\|\mathbf{\Delta x}\|}\left((\delta_p^q - 2a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} + a_{1,p}d_q\right)(\mathbf{a}_1^T\mathbf{d}) \\
&\quad +k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})\frac{1}{\|\mathbf{\Delta x}\|}(d_q - a_{1,q}\mathbf{a}_1^T\mathbf{d}) \\
&= -k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|^2}\big(a_{1,q}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(\mathbf{a}_1^T\mathbf{d}) \\
&\quad + \left((\delta_p^q - 2a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} + a_{1,p}d_q\right)(\mathbf{a}_1^T\mathbf{d}) \\
&\quad -(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(d_q - a_{1,q}\mathbf{a}_1^T\mathbf{d})\big) \\
&= -k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|^2}\big(a_{1,q}(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(\mathbf{a}_1^T\mathbf{d}) \\
&\quad + \left((\delta_p^q - 2a_{1,q}a_{1,p})\mathbf{a}_1^T\mathbf{d} + a_{1,p}d_q\right)(\mathbf{a}_1^T\mathbf{d}) \\
&\quad -(d_p - a_{1,p}\mathbf{a}_1^T\mathbf{d})(d_q - a_{1,q}\mathbf{a}_1^T\mathbf{d})\big) \\
&= -\frac{k_{Orth}}{\|\mathbf{\Delta x}\|^2}\big((\delta_p^q - 4a_{1,p}a_{1,q})(\mathbf{a}_1^T\mathbf{d})^2 + 2(a_{1,q}d_p + a_{1,p}d_q)(\mathbf{a}_1^T\mathbf{d}) - d_p d_q\big)
\end{aligned}
$$

$$
\mathbf{H}_{Orth,(i,j),\Delta x_{1,(i,j)},\Delta x_{1,(i,j)}} = -k_{Orth}\frac{1}{\|\mathbf{\Delta x}\|^2}\big((\mathbf{I}_3 - 4\mathbf{a}_1\mathbf{a}_1^T)(\mathbf{a}_1^T\mathbf{d})^2 + 2(\mathbf{d}\mathbf{a}_1^T + \mathbf{a}_1\mathbf{d}^T)(\mathbf{a}_1^T\mathbf{d}) - \mathbf{d}\mathbf{d}^T\big)
$$

9.4 Curvature Tensor

The curvature tensor is

$$
b_{\alpha\beta} = \frac{\partial^2 \mathbf{x}}{\partial u_\alpha \partial u_\beta} \cdot \mathbf{d}
$$

The discretization is then

$$\frac{\partial^2 \mathbf{x}}{\partial^2 u_1} = \frac{1}{L^2}(\mathbf{x}_{i+1,j} - 2\mathbf{x}_{i,j} + \mathbf{x}_{i-1,j})$$

$$\frac{\partial^2 \mathbf{x}}{\partial^2 u_2} = \frac{1}{L^2}(\mathbf{x}_{i,j+1} - 2\mathbf{x}_{i,j} + \mathbf{x}_{i,j-1})$$

$$\frac{\partial^2 \mathbf{x}}{\partial u_1 \partial u_2} = \frac{1}{4L^2}(\mathbf{x}_{i+1,j+1} - \mathbf{x}_{i+1,j-1} - \mathbf{x}_{i-1,j+1} + \mathbf{x}_{i-1,j-1})$$

$$E_{curv,i} = \frac{1}{2}k_{curv}(b_{\alpha\beta})^2 = \frac{1}{2}k_{curv}(b_{1,1}^2 + b_{2,2}^2 + 2b_{1,2}^2)$$

The force is then

$$\begin{aligned} F_{curv,i,X_k} &= -k_{curv}\frac{\partial}{\partial X_k}((\frac{\partial^2 r}{\partial^2 u_1}\cdot d)^2 + (\frac{\partial^2 r}{\partial^2 u_2}\cdot d)^2 + 2(\frac{\partial^2 r}{\partial u_1 \partial u_2}\cdot d)^2) \\ &= -k_{curv}((\frac{\partial}{\partial X_k}\frac{\partial^2 r}{\partial^2 u_1}\cdot d + \frac{\partial^2 r}{\partial^2 u_1}\cdot\frac{\partial d}{\partial X_k})(\frac{\partial^2 r}{\partial^2 u_1}\cdot d) \\ &\quad +(\frac{\partial}{\partial X_k}\frac{\partial^2 r}{\partial^2 u_2}\cdot d + \frac{\partial^2 r}{\partial^2 u_2}\cdot\frac{\partial d}{\partial X_k})(\frac{\partial^2 r}{\partial^2 u_2}\cdot d) \\ &\quad +(\frac{\partial}{\partial X_k}\frac{\partial^2 r}{\partial u_1 \partial u_2}\cdot d + \frac{\partial^2 r}{\partial u_1 \partial u_2}\cdot\frac{\partial d}{\partial X_k})(\frac{\partial^2 r}{\partial u_1 \partial u_2}\cdot d)) \end{aligned}$$

A lot of these terms are equal to 0. For example,

$$F_{curv,(i,j),\mathbf{x}_{(i+1,j),p}} = -k_{curv}\frac{1}{L^2}(\mathbf{e}_p \cdot d)(\frac{\partial^2 r}{\partial^2 u_1}\cdot d)$$

There is also the possibility of defining a non-zero reference curvature. The energy is then, for B the reference curvature tensor

$$\begin{aligned} E_{curv,i} &= \frac{1}{2}k_{curv}\sum(b_{\alpha\beta} - B_{\alpha\beta})^2 \\ &= \frac{1}{2}k_{curv}((b_{1,1} - B_(1,1)^2 + (b_{2,2} - B_(2,2)^2 + 2(b_{1,2} - B_(1,2))^2) \end{aligned}$$

9.5 First results

Here are a few images with different values of the constants.

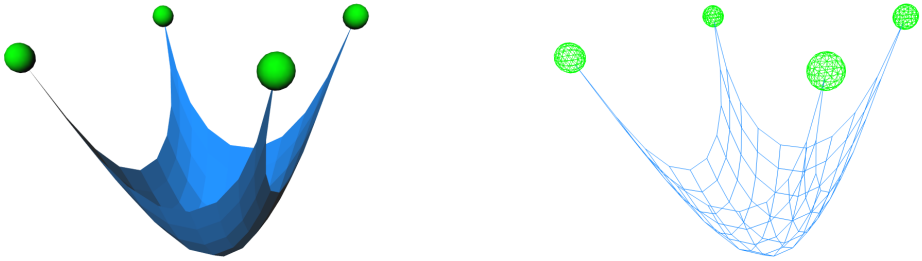

Table 9.1: with $k_{length} = 10$ and $m = 0.1$: the net is extremely stretched

Figure 9.2: with $k_{length} = 10$ and $m = 0.01$

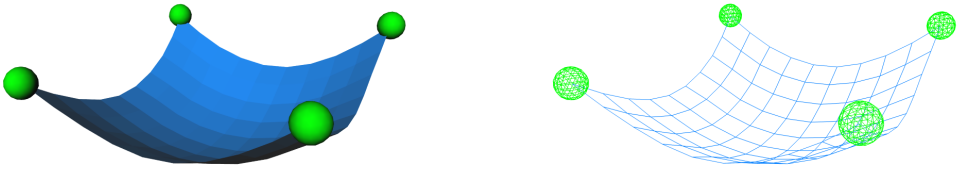

Figure 9.3: with $k_{length} = 10$ and $m = 0.01$ and $k_{Shear} = 1$

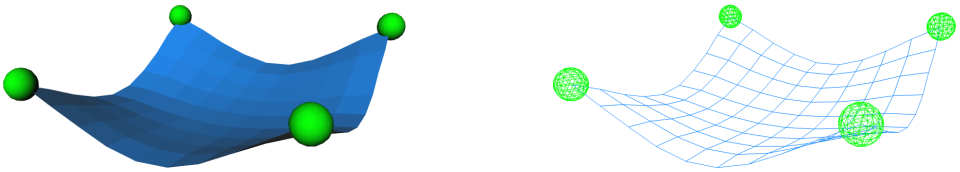

Figure 9.4: with $k_{length} = 10$ and $m = 0.01$ and $k_{Shear} = 10$

Figure 9.5: with $k_{Curvature} = 1$: the surface is smoother

Figure 9.6: with $k_{Curvature} = 10$

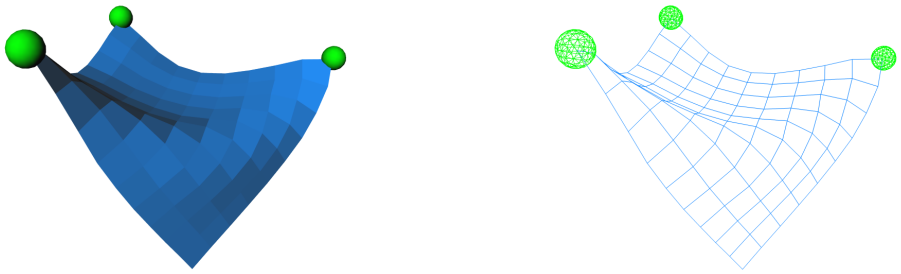

Figure 9.7: with $k_{Curvature} = 0$

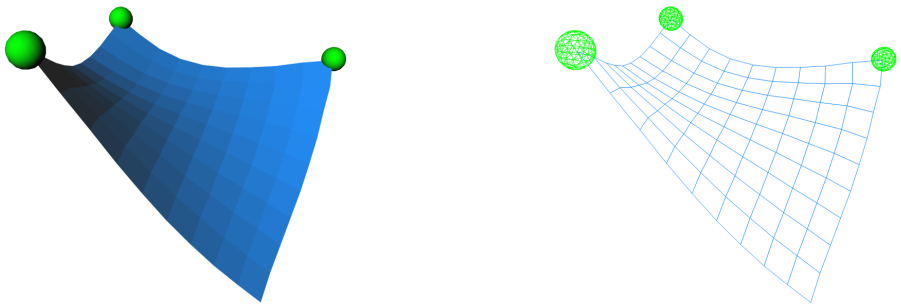

Figure 9.8: with $k_{Curvature} = 1$

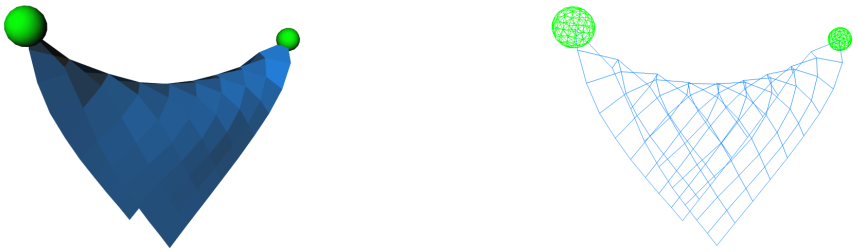

Figure 9.9: with $k_{Curvature} = 0$

Figure 9.10: with $k_{Curvature} = 0$ and $k_{Shear} = 1$

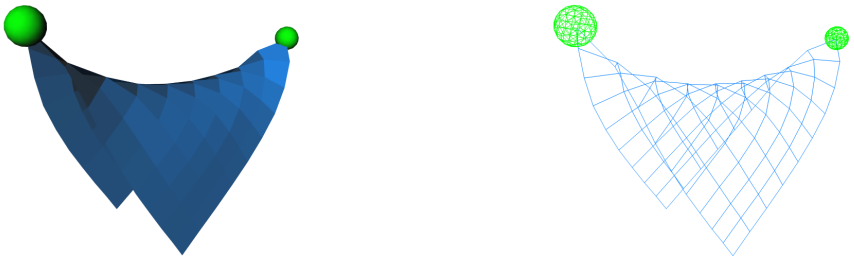

Figure 9.11: with $k_{Curvature} = 0.1$

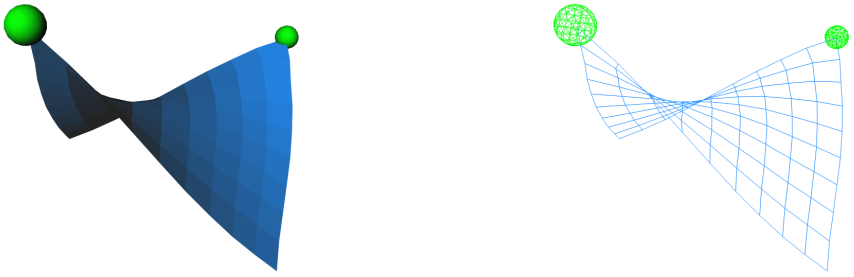

Figure 9.12: with $k_{Curvature} = 1$

Figure 9.13: with $k_{Curvature} = 10$

Figure 9.14: with $k_{Curvature} = 0$

Figure 9.15: with $k_{Curvature} = 0.1$

Figure 9.16: with $m = 0$ and $B_{1,1} = 0.5$: the surface takes a cylindrical form.

Figure 9.17: with $m = 0$ and $B_{2,2} = -0.5$

Figure 9.18: with $m = 0$ and $B_{1,2} = 0.4$

9.6 limitations

The forces on the positions due to the curvature terms are all in the direction of the director. If the curvature is high, these forces are almost compensated by the forces for the length conservation: the system cannot "get out" of this situation.

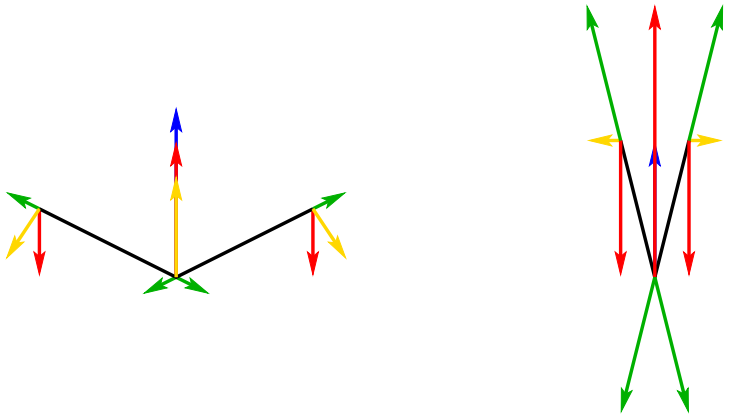

Figure 9.19: Problem with the curvature: the curvature forces (in red), along the director (in blue) tend to be compensated by the length conservation forces (in green): the resultant (in yellow) is small

Figure 9.20: In the case of the first model of a cable, the curvature forces where orthogonal to the segments: there is no interference between the curvature and length forces.

Appendix A

Appendix - Derivation of the forces and Hessian

A.1 Preliminary calculations

A.1.1 Definition of $\mathbf{\Delta x}_i$

Let us define $\mathbf{\Delta x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i \in \mathbb{R}^3$ for $i = 1...n-1$ and $\Delta x_{i,j} \in \mathbb{R}$ its coordinates, $j = 1, 2, 3$. Remark that for any function $f(\mathbf{\Delta x}_i, ...)$, we have, thanks to the chain derivation rule,

$$\frac{\partial f}{\partial \Delta x_{i,\,j}} = \frac{\partial f}{\partial x_{i+1,\,j}} = -\frac{\partial f}{\partial x_{i,\,j}}$$

and

$$\begin{aligned}\frac{\partial^2 f}{\partial \Delta x_{i,\,j}\,\partial \Delta x_{i,\,k}} &= \frac{\partial^2 f}{\partial x_{i+1,\,j}\,\partial x_{i+1,\,k}} = \frac{\partial^2 f}{\partial x_{i,\,j}\,\partial x_{i,\,k}} \\ &= -\frac{\partial^2 f}{\partial x_{i+1,\,j}\,\partial x_{i,\,k}} = -\frac{\partial^2 f}{\partial x_{i,\,j}\,\partial x_{i+1,\,k}}\end{aligned}$$

(All the energy functions depending on the position except the contact interaction can be put under this form, which is a sign that the formulation is translation invariant).

A.1.2 Notation

In order to represent in a more compact form the derivatives, parts of the forces and parts of the Hessian, we introduce the following notation: for

$\mathbf{v} \in \mathbb{R}^p$ a part of $\mathbf{X}$ (there exists a subset A of $1...(7n-4)$ such that for each $j = 1...p$ $v_j = X_{A_j}$ with $A_j < A_{j+1}$), $\mathbf{F_v}$ is the corresponding part of $\mathbf{F}$:

$$F_{\mathbf{v},j} = F_{A_j}.$$

Furthermore, we introduce

$$\mathbf{F}_{Interaction,i,\mathbf{\Delta x}_i} = \mathbf{F}_{Interaction,i,\mathbf{x}_{i+1}} = -\mathbf{F}_{Interaction,i,\mathbf{x}_i}$$

for the interactions depending on $\mathbf{\Delta x}_i$.

In a similar way, we introduce for $\mathbf{v} \in \mathbb{R}^p$ and $\mathbf{w} \in \mathbb{R}^q$, parts of $\mathbf{X}$ with respective subsets A and B, the block $\mathcal{H}_{Interaction,i,\mathbf{v},\mathbf{w}} \in \mathbb{R}^{p \times q}$ such that for $j = 1...p$ and $k = 1...q$,

$$H_{Interaction,i,\mathbf{v},\mathbf{w},j,k} = H_{Interaction,i,A_j,B_k}.$$

We also have

$$\mathbf{H}_{Interaction,i,\mathbf{\Delta x}_i,\mathbf{w}} = \mathbf{H}_{Interaction,i,\mathbf{x}_{i+1},\mathbf{w}} = -\mathbf{H}_{Interaction,i,\mathbf{x}_i,\mathbf{w}}$$

and

$$\mathbf{H}_{Interaction,i,\mathbf{v},\mathbf{\Delta x}_i} = \mathbf{H}_{Interaction,i,\mathbf{v},\mathbf{x}_{i+1}} = -\mathbf{H}_{Interaction,i,\mathbf{v},\mathbf{x}_i}$$

The identity matrix of $\mathbb{R}^{j \times j}$ for j integer will be noted $\mathbf{I}_j$.

A.1.3 Derivative of the length of a segment

The length of a segment is then

$$L_i = \|\mathbf{\Delta x_i}\| = \sqrt{\sum_{m=1}^{3} \Delta x_{i,\,m}^2}$$

The derivative of L_i is then

$$\frac{\partial L_i}{\partial \Delta x_{i,\,j}} = \frac{1}{2} \frac{2\Delta x_{i,\,j}}{\sqrt{\sum_{m=1}^{3} \Delta x_{i,\,m}^2}} = \frac{\Delta x_{i,\,j}}{L_i} = u_{i,\,j}$$

A.1.4 Derivative of the unit vector $\mathbf{u}_i$

$\mathbf{u}_i$ only depends on $\mathbf{\Delta x}_i$. We have, with $\mathbf{e}_j = (\delta_1^j,\, \delta_2^j,\, \delta_3^j)$ the j^{th} vector of the cartesian coordinate system of $\mathbb{R}^3$ where δ_i^j is the Kronecker symbol.

$$\begin{aligned}
\frac{\partial \mathbf{u}_i}{\partial \Delta x_{i,\,j}} &= \frac{\partial}{\partial \Delta x_{i,\,j}} \frac{\mathbf{\Delta x}_i}{L_i} \\
&= \frac{\mathbf{e}_j L_i - \mathbf{\Delta x}_i \; u_{i,\,j}}{L_i^2} \\
&= \frac{1}{L_i} \left(\mathbf{e}_j - u_{i,\,j} \mathbf{u}_i \right)
\end{aligned}$$

A.1.5 Derivative of the quaternion norm

$$\|\mathfrak{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = \sqrt{\sum_{i=1}^{4} q_i^2}$$

$$\frac{\partial \|\mathfrak{q}\|}{\partial q_j} = \frac{1}{2} \frac{2 q_j}{\sqrt{\sum_{k=1}^{4} q_k^2}} = \frac{q_j}{\|\mathfrak{q}\|}$$

A.1.6 Image of the reference Orientation by a rotation

Let us define the vector $\mathbf{r} \in \mathbb{R}^3$ as the image of the reference orientation **ref** by a rotation defined by a quaternion $\mathfrak{q}$. From the properties of the quaternions, we know that

$$(0, \mathbf{r}) = \mathfrak{q} \cdot (0, \mathbf{ref}) \cdot \overline{\mathfrak{q}}$$

where the quaternion product is used

$$\mathfrak{p} \cdot \mathfrak{q} = (p_0 q_0 - \mathbf{p}^T \mathbf{q},\; p_0 \mathbf{q} + q_0 \mathbf{p} + \mathbf{p} \times \mathbf{q})$$

In our case, we set the reference direction to be $\mathbf{ref} = (0, 1, 0)$ which leads to a simplification of the previous expression:

$$\begin{aligned}
\mathfrak{q} \cdot (0, \mathbf{ref}) &= (0 - \mathbf{q}^T\mathbf{ref}, q_0\mathbf{ref} + \mathbf{0} + \mathbf{q} \times \mathbf{ref}) \\
&= (-q_2,\ q_0\mathbf{ref} + (-q_3, 0, q_1)) \\
&= (-q_2,\ (-q_3, q_0, q_1)) \\
(\mathfrak{q} \cdot (0, \mathbf{ref})) \cdot \overline{\mathfrak{q}} &= (-q_2q_0 + (-q_3, q_0, q_1)^T\mathbf{q}, \\
&\quad q_2\mathbf{q} + q_0(-q_3, q_0, q_1) - (-q_3, q_0, q_1) \times (q_1, q_2, q_3)) \\
&= (-q_2q_0 - q_3q_1 + q_0q_2 + q_1q_3, \\
&\quad q_2\mathbf{q} + q_0(-q_3, q_0, q_1) - (-q_3, q_0, q_1) \times (q_1, q_2, q_3)) \\
&= (0,\ (q_2q_1 - q_0q_3, q_2^2 + q_0^2, q_2q_3 + q_0q_1) \\
&\quad -(q_0q_3 - q_1q_2, q_1^2 + q_3^2, -q_3q_2 - q_0q_1)) \\
&= (0,\ (2q_1q_2 - 2q_0q_3, q_2^2 + q_0^2 - q_1^2 - q_3^2, 2q_3q_2 + 2q_0q_1))
\end{aligned}$$

We have then

$$\mathbf{r} = \begin{pmatrix} 2\,(q_1q_2 - q_0q_3) \\ q_0^2 - q_1^2 + q_2^2 - q_3^2 \\ 2\,(q_0q_1 + q_2q_3) \end{pmatrix}$$

A.1.7 Derivative of a normalized vector

Let us consider the vectors $\mathbf{u}$ and $\mathbf{v}$ such that $\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|}$ and a variable x. We have then

$$\begin{aligned}
\frac{\partial \mathbf{u}}{\partial x} &= \frac{\partial}{\partial x}\frac{\mathbf{v}}{\|\mathbf{v}\|} \\
&= \frac{\frac{\partial \mathbf{v}}{\partial x}\|\mathbf{v}\| - \mathbf{v}\frac{\partial \|\mathbf{v}\|}{\partial x}}{\|\mathbf{v}\|^2} \\
&= \frac{1}{\|\mathbf{v}\|}\frac{\partial \mathbf{v}}{\partial x} - \frac{\mathbf{u}}{\|\mathbf{v}\|}\frac{\partial \|\mathbf{v}\|}{\partial x} \\
&= \frac{1}{\|\mathbf{v}\|}\frac{\partial \mathbf{v}}{\partial x} - \frac{\mathbf{u}}{\|\mathbf{v}\|}\frac{\mathbf{v}^T\frac{\partial \mathbf{v}}{\partial x}}{\|\mathbf{v}\|} \\
&= \frac{1}{\|\mathbf{v}\|}(\mathbf{I} - \mathbf{u}\mathbf{u}^T)\frac{\partial \mathbf{v}}{\partial x}
\end{aligned}$$

A.2 Length conservation

A.2.1 Energy

The energy, which only depends on $\mathbf{\Delta x}_i$, is

$$E_{Length,\,i} = \frac{1}{2}k_{Length}\,(L_i - L_{ref})^2$$

A.2.2 Force

$$\begin{aligned} F_{Length,\,i,\,\Delta x_{i,j}} &= -\frac{\partial E_{Length,\,i}}{\partial \Delta x_{i,\,j}} \\ &= -k_{Length}\frac{\partial L_i}{\partial \Delta x_{i,\,j}}\ (L_i - L_{ref}) \\ &= -k_{Length}u_{i,\,j}\ (L_i - L_{ref}) \\ \mathbf{F}_{Length,\,i,\,\mathbf{\Delta x}_i} &= -k_{Length}\mathbf{u}_i\,(L_i - L_{ref}) \end{aligned}$$

The force is proportional to the control deviation of the length and is in the direction of the segment.

A.2.3 Hessian

We have for $j = 1...3$ and $k = 1...3$, the elements of the block $\mathcal{H}_{Length,\,i,\,\mathbf{\Delta x}_i,\,\mathbf{\Delta x}_i} \in \mathbb{R}^{3\times 3}$ corresponding to ($\mathbf{\Delta x}_i$,$\mathbf{\Delta x}_i$) of the Hessian is

$$\begin{aligned} H_{Length,\,i,\,\Delta x_{i,j},\,\Delta x_{i,k}} &= -\frac{\partial F_{Length,\,i,\,\Delta x_{i,j}}}{\partial \Delta x_{i,\,k}} \\ &= k_{Length}\left(\frac{\partial u_{i,\,j}}{\partial \Delta x_{i,\,k}}\,(L_i - L_{ref}) + u_{i,\,j}\frac{\partial L_i}{\partial \Delta x_{i,\,k}}\right) \\ &= k_{Length}\left(\frac{1}{L_i}\left(\delta^k_j - u_{i,\,k}u_{i,\,j}\right)(L_i - L_{ref}) + u_{i,\,j}u_{i,\,k}\right) \\ &= k_{Length}\left(\left(1 - \frac{L_{ref}}{L_i}\right)\left(\delta^k_j - u_{i,\,k}u_{i,\,j}\right) + u_{i,\,j}u_{i,\,k}\right) \\ &= k_{Length}\left(\left(1 - \frac{L_{ref}}{L_i}\right)\delta^k_j + \frac{L_{ref}}{L_i}u_{i,\,j}u_{i,\,k}\right) \end{aligned}$$

The block is then

$$\mathbf{H}_{Length,i,\,\mathbf{\Delta x}_i,\,\mathbf{\Delta x}_i} = k_{Length}\left(\left(1 - \frac{L_{ref}}{L_i}\right)\mathbf{I_3} + \frac{L_{ref}}{L_i}\mathbf{u}_i\mathbf{u}_i^T\right)$$

A.3 Quaternion Norm: first formulation

A.3.1 Energy

The energy is defined as

$$E_{QuatNorm,i} = \frac{1}{2}k_{QuatNorm}\left(\|\mathfrak{q}_i\| - 1\right)^2$$

for $i \in 1...n-1$ with $\|\mathfrak{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$.

A.3.2 Force

The energy only depends on the components q_j, $j = 1...4$ of $\mathfrak{q}_i$.

$$\begin{aligned} F_{QuatNorm,\,i,\,q_{i,j}} &= -\frac{\partial E_{QuatNorm,i}}{\partial q_j} \\ &= -k_{QuatNorm}\frac{\partial \|\mathfrak{q}_\mathfrak{i}\|}{\partial q_j}\left(\|\mathfrak{q}_\mathfrak{i}\| - 1\right) \\ &= -k_{QuatNorm}\frac{q_j}{\|\mathfrak{q}_\mathfrak{i}\|}\left(\|\mathfrak{q}_\mathfrak{i}\| - 1\right) \\ &= k_{QuatNorm}\; q_j\; \left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|} - 1\right) \end{aligned}$$

$$F_{QuatNorm,\,i,\,\mathfrak{q}_i} = k_{QuatNorm}\; \mathfrak{q}_\mathfrak{i}\; \left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|} - 1\right)$$

A.3.3 Hessian

For $j = 1...4$ and $k = 1...4$, the elements of the block $\mathbf{H}_{QuatNorm,i,\mathfrak{q}_i,\mathfrak{q}_i} \in \mathbb{R}^{4\times 4}$ corresponding to ($\mathfrak{q}_i$,$\mathfrak{q}_\mathfrak{i}$) of the Hessian are

$$\begin{aligned} \mathbf{H}_{QuatNorm,\,i,\,q_{i,j},\,q_{i,k}} &= -\frac{\partial F_j}{\partial q_k} \\ &= -\frac{\partial k_{QuatNorm}\; q_j\; \left(\frac{1}{\|\mathfrak{q}\|} - 1\right)}{\partial q_k} \\ &= -k_{QuatNorm}\left(\delta_j^k\left(\frac{1}{\|\mathfrak{q}\|} - 1\right) - q_j\frac{q_k}{\|\mathfrak{q}\|}\frac{1}{\|\mathfrak{q}\|^2}\right) \\ &= -k_{QuatNorm}\left(\delta_j^k\left(\frac{1}{\|\mathfrak{q}\|} - 1\right) - \frac{q_j q_k}{\|\mathfrak{q}\|^3}\right) \end{aligned}$$

The corresponding block of the Hessian is

$$\mathbf{H}_{QuatNorm,i,\mathfrak{q}_i,\mathfrak{q}_i} = k_{QuatNorm}\left(\left(1 - \frac{1}{\|\mathfrak{q}\|}\right)\mathbf{I} + \frac{\mathfrak{q}\mathfrak{q}^T}{\|\mathfrak{q}\|^3}\right)$$

where $\mathbf{I}$ is the identity matrix.

If we define $\mathfrak{p} = \frac{\mathfrak{q}}{\|\mathfrak{q}\|}$, we have

$$\mathbf{H}_{QuatNorm,i,\mathfrak{q}_i,\mathfrak{q}_i} = k_{QuatNorm}\left(\left(1 - \frac{1}{\|\mathfrak{q}\|}\right)\mathbf{I} + \frac{\mathfrak{p}\mathfrak{p}^T}{\|\mathfrak{q}\|}\right)$$

A.4 Quaternion Norm: second formulation

A.4.1 Energy

The energy is defined as

$$E_{QuatNorm,i} = \frac{1}{2}k_{QuatNorm}\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|} - 1\right)^2$$

for $i \in 1...n-1$.

A.4.2 Force

The energy only depends on the components q_j, $j = 1...4$ of $\mathfrak{q}_i$.

$$\begin{aligned} F_{QuatNorm,\,i,\,q_{i,j}} &= -\frac{\partial E_{QuatNorm,i}}{\partial q_j} \\ &= -k_{QuatNorm}\frac{\partial \|\mathfrak{q}_\mathfrak{i}\|}{\partial q_j}\left(-\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|^2}\right)\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right) \\ &= -k_{QuatNorm}\frac{q_j}{\|\mathfrak{q}_\mathfrak{i}\|}\left(-\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|^2}\right)\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right) \\ &= k_{QuatNorm}\frac{q_j}{\|\mathfrak{q}_\mathfrak{i}\|^3}\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right) \end{aligned}$$

$$F_{QuatNorm,\,i,\,\mathfrak{q}_i} = k_{QuatNorm}\ \frac{\mathfrak{q}_\mathfrak{i}}{\|\mathfrak{q}_\mathfrak{i}\|^3}\ \left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)$$

A.4.3 Hessian

For $j = 1...4$ and $k = 1...4$, the elements of the block $\mathbf{H}_{QuatNorm,i,\mathfrak{q}_i,\mathfrak{q}_i} \in \mathbb{R}^{4\times4}$ corresponding to ($\mathfrak{q}_i$,$\mathfrak{q}_\mathfrak{i}$) of the Hessian are

$$\begin{aligned} \mathbf{H}_{QuatNorm,\,i,\,q_{i,j},\,q_{i,k}} &= -\frac{\partial F_j}{\partial q_k} \\ &= -\frac{\partial}{\partial q_k}\left(k_{QuatNorm}\ \frac{q_j}{\|\mathfrak{q}_\mathfrak{i}\|^3}\ \left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)\right) \\ &= -k_{QuatNorm}\left(\delta_j^k\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|^3}+q_j\frac{q_k}{\|\mathfrak{q}_\mathfrak{i}\|}\left(-\frac{4}{\|\mathfrak{q}\|^5}+\frac{3}{\|\mathfrak{q}\|^4}\right)\right) \\ &= -k_{QuatNorm}\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|^3}\left(\delta_j^k\left(\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}-1\right)+\frac{q_j q_k}{\|\mathfrak{q}_\mathfrak{i}\|^2}\left(-\frac{4}{\|\mathfrak{q}\|}+3\right)\right) \end{aligned}$$

The corresponding block of the Hessian is

$$\mathbf{H}_{QuatNorm,i,\mathfrak{q}_i,\mathfrak{q}_i} = k_{QuatNorm}\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|^3}\left(\left(1-\frac{1}{\|\mathfrak{q}_\mathfrak{i}\|}\right)\mathbf{I}+\frac{\mathfrak{q}\mathfrak{q}^T}{\|\mathfrak{q}_\mathfrak{i}\|^2}\left(\frac{4}{\|\mathfrak{q}\|}-3\right)\right)$$

A.5 Coherence of Positions and Quaternions

A.5.1 Energy

Let us define the vector $\mathbf{\Delta}_i \in \mathbb{R}^3$ equal to the difference between the unitary vector joining the two points and the image $\mathbf{r_i} \in \mathbb{R}^3$ of the reference orientation **ref** by the rotation defined by the quaternion $\mathfrak{q}_i$:

$$(0, \mathbf{r}_i) = \mathfrak{q}_i \cdot (0, \mathbf{ref}) \cdot \overline{\mathfrak{q}_i}$$

$$\mathbf{\Delta}_i = \mathbf{u}_i - \mathbf{r}_i$$

The energy is defined as

$$E_{Coh,\,i} = \frac{1}{2}\, k_{Coh}\, \|\mathbf{\Delta}_i\|^2$$

A.5.2 Forces

We have

$$-\frac{\partial E_{Coh,\,i}}{\partial X_j} = -k_{Coh}\mathbf{\Delta}_i^T \frac{\partial \mathbf{\Delta}_i}{\partial X_j}$$

$$\begin{aligned}
F_{Coh,i,\Delta x_{i,j}} = -\tfrac{\partial E_{Coh,\,i}}{\partial \Delta x_{i,j}} &= -k_{Coh}\ \mathbf{\Delta}_i^T\ \tfrac{\partial \mathbf{\Delta}_i}{\partial \Delta x_{i,j}} \\
&= -k_{Coh}\ \mathbf{\Delta}_i^T\ \tfrac{\partial \mathbf{u}_i}{\partial \Delta x_{i,j}} \\
&= -k_{Coh}\ (\mathbf{u}_i - \mathbf{r}_i)^T\ \tfrac{\partial \mathbf{u}_i}{\partial \Delta x_{i,j}} \\
&= k_{Coh}\ \mathbf{r}_i^T\ \tfrac{\partial \mathbf{u}_i}{\partial \Delta x_{i,j}} \quad \text{since } \mathbf{u}_i^T \tfrac{\partial \mathbf{u}_i}{\partial \Delta x_{i,j}} = 0 \text{ (because } \|\mathbf{u}_i\| = 1) \\
&= k_{Coh}\ \tfrac{1}{L_i}\ \mathbf{r}_i^T\ (\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i) \\
\mathbf{F}_{Coh,i,\Delta x_i} &= k_{Coh}\ \tfrac{1}{L_i}\ (\mathbf{I}_3 - \mathbf{u}_i\mathbf{u}_i^T)\mathbf{r}_i
\end{aligned}$$

With $\mathbf{r}_i = (2\ (q_{i,1}q_{i,2} - q_{i,0}q_{i,3}),\ q_{i,0}^2 - q_{i,1}^2 + q_{i,2}^2 - q_{i,3}^2,\ 2\ (q_{i,0}q_{i,1} + q_{i,2}q_{i,3}))$

$$\begin{aligned}
F_{Coh,\,i,\,q_{i,j}} &= -\tfrac{\partial E_{Coh,\,i}}{\partial q_{i,j}} \\
&= -k_{Coh}\ \mathbf{\Delta}_i^T\ \tfrac{\partial \mathbf{\Delta}_i}{\partial q_{i,j}} \\
&= k_{Coh}\ \mathbf{\Delta}_i^T\ \tfrac{\partial \mathbf{r}_i}{\partial q_{i,j}} \\
&= k_{Coh}\ \mathbf{u}_i^T\ \tfrac{\partial \mathbf{r}_i}{\partial q_{i,j}}
\end{aligned}$$

$$\begin{aligned}
\tfrac{\partial \mathbf{r}_i}{\partial q_{i,0}} &= (\ -2q_{i,3}\ ,\ \ 2q_{i,0}\ ,\ 2\ q_{i,1}\ ) \\
\tfrac{\partial \mathbf{r}_i}{\partial q_{i,1}} &= (\ \ 2q_{i,2}\ ,\ -2q_{i,1}\ ,\ 2\ q_{i,0}\ ) \\
\tfrac{\partial \mathbf{r}_i}{\partial q_{i,2}} &= (\ \ 2q_{i,1}\ ,\ \ 2q_{i,2}\ ,\ 2\ q_{i,3}\ ) \\
\tfrac{\partial \mathbf{r}_i}{\partial q_{i,3}} &= (\ -2q_{i,0}\ ,\ -2q_{i,3}\ ,\ 2\ q_{i,2}\ )
\end{aligned}$$

A.5.3 Hessian

We have to consider 3 different kinds of blocks: $\frac{\partial^2 E_{Coh,\,i}}{\partial \Delta x_{i,j} \partial \Delta x_{i,k}}$, $\frac{\partial^2 E_{Coh,\,i}}{\partial q_{i,j} \partial q_{i,k}}$ and $\frac{\partial^2 E_{Coh,\,i}}{\partial \Delta x_{i,j} \partial q_{i,k}}$.

$\mathbf{H}_{Coh,\,i,\,\mathbf{\Delta x}_i,\,\mathbf{\Delta x}_i}$

$$
\begin{aligned}
H_{Coh,\,i,\,\Delta x_{i,j}\,\Delta x_{i,k}} &= \frac{\partial^2 E_{Coh,\,i}}{\partial \Delta x_{i,j} \partial \Delta x_{i,k}} \\
&= -\frac{\partial}{\partial \Delta x_{i,k}} \left(k_{Coh}\ \frac{1}{L_i}\ \mathbf{r}_i^T\ \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right)\right) \\
&= -k_{Coh}\ \frac{\partial}{\partial \Delta x_{i,k}} \left(\frac{1}{L_i}\ \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right)\right)^T \mathbf{r}_i \\
&= -k_{Coh} \left(\frac{\frac{\partial \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right)}{\partial \Delta x_{i,k}} L_i - \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right) \frac{\partial L_i}{\partial \Delta x_{i,k}}}{L_i^2} \right)^T \mathbf{r}_i \\
&= -\frac{k_{Coh}}{L_i^2} \left(\frac{\partial (\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i)}{\partial \Delta x_{i,k}} L_i\ -\ \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right) \frac{\partial L_i}{\partial \Delta x_{i,k}} \right)^T \mathbf{r}_i \\
&= -\frac{k_{Coh}}{L_i^2} \left(-\left(\frac{1}{L_i} \left(\delta_j^k - u_{i,\,k} u_{i,\,j}\right) \mathbf{u}_i + u_{i,\,j} \frac{1}{L_i} \left(\mathbf{e}_k - u_{i,\,k}\mathbf{u}_i\right)\right) L_i \right. \\
&\quad \left. -\ \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right) u_{i,\,k} \right)^T \mathbf{r}_i \\
&= \frac{k_{Coh}}{L_i^2} \left(\left(\delta_j^k - u_{i,\,k} u_{i,\,j}\right) \mathbf{u}_i + u_{i,\,j} \left(\mathbf{e}_k - u_{i,\,k}\mathbf{u}_i\right)\ + \right. \\
&\quad \left. \left(\mathbf{e}_j - u_{i,\,j}\mathbf{u}_i\right) u_{i,\,k} \right)^T \mathbf{r}_i \\
&= \frac{k_{Coh}}{L_i^2} \left(\left(\delta_j^k - 3 u_{i,\,k} u_{i,\,j}\right) \mathbf{u}_i + u_{i,\,j} \mathbf{e}_k\ +\ u_{i,\,k} \mathbf{e}_j \right)^T \mathbf{r}_i \\
\mathbf{H}_{Coh,\,i,\,\mathbf{\Delta x}_i,\,\mathbf{\Delta x}_i} &= \frac{k_{Coh}}{L_i^2} \left(\left(\mathbf{I} - 3\mathbf{u}_i\mathbf{u}_i^T\right) \mathbf{u}_i^T\ \mathbf{r}_i + \mathbf{u}_i \mathbf{r}_i^T + \mathbf{r}_i \mathbf{u}_i^T \right)
\end{aligned}
$$

$\mathbf{H}_{Coh,\,i,\,\mathbf{q}_i,\,\mathbf{q}_i}$

$$
\begin{aligned}
\frac{\partial^2 E_{Coh,\,i}}{\partial q_{i,j} \partial q_{i,k}} &= -\frac{\partial}{\partial q_{i,k}} (k_{Coh}\ \mathbf{u}_i^T\ \frac{\partial \mathbf{r}_i}{\partial q_{i,j}}) \\
&= -k_{Coh} \mathbf{u}_i^T \frac{\partial}{\partial q_{i,k}} (\ \frac{\partial \mathbf{r}_i}{\partial q_{i,j}})
\end{aligned}
$$

$$
\frac{\partial^2 r_{i,1}}{\partial q_{i,j} \partial q_{i,k}} = \begin{pmatrix} 0 & 0 & 0 & -2 \\ 0 & 0 & 2 & 0 \\ 0 & 2 & 0 & 0 \\ -2 & 0 & 0 & 0 \end{pmatrix} \quad \frac{\partial^2 r_{i,2}}{\partial q_{i,j} \partial q_{i,k}} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}
$$

$$
\frac{\partial^2 r_{i,3}}{\partial q_{i,j} \partial q_{i,k}} = \begin{pmatrix} 0 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}
$$

$$
\begin{aligned}
\mathbf{H}_{Coh,\,i,\,\mathbf{q}_i,\,\mathbf{q}_i} &= -2k_{Coh} \begin{pmatrix} u_{i,2} & u_{i,3} & 0 & -u_{i,1} \\ u_{i,3} & -u_{i,2} & u_{i,1} & 0 \\ 0 & u_{i,1} & u_{i,2} & u_{i,3} \\ -u_{i,1} & 0 & u_{i,3} & -u_{i,2} \end{pmatrix} \\
&= 2k_{Coh} \begin{pmatrix} -u_{i,2} & -u_{i,3} & 0 & u_{i,1} \\ -u_{i,3} & u_{i,2} & -u_{i,1} & 0 \\ 0 & -u_{i,1} & -u_{i,2} & -u_{i,3} \\ u_{i,1} & 0 & -u_{i,3} & u_{i,2} \end{pmatrix}
\end{aligned}
$$

$\mathbf{H}_{Coh,\,i,\,\boldsymbol{\Delta}\mathbf{x}_i,\,\mathfrak{q}_i}$

$$\frac{\partial^2 E_{Coh,\,i}}{\partial \Delta x_{i,j} \partial q_{i,k}} = -k_{Coh} \frac{\partial \mathbf{u}_i}{\partial \Delta x_{i,j}}^T \frac{\partial \mathbf{r}_i}{\partial q_{i,k}}$$

A.6 Handles

A.6.1 Energy

For a spherical handle, the energy is

$$E_{HandleSphere,\,i} = \frac{1}{2} k_{Handle} \left(\left(\mathbf{x}_i - \mathbf{x}_{Handle} \right)^2 + \left(\frac{\mathfrak{q}_{i-1} + \mathfrak{q}_i}{\left\| \mathfrak{q}_{i-1} + \mathfrak{q}_i \right\|} - \mathfrak{q}_{Handle} \right)^2 \right)$$

or

$$E_{HandleSphere,\,i} = \frac{1}{2} k_{Handle} \left(\left(\mathbf{x}_i - \mathbf{x}_{Handle} \right)^2 + \left(\frac{\mathfrak{q}_{i-1} + \mathfrak{q}_i}{2} - \mathfrak{q}_{Handle} \right)^2 \right)$$

whereas for a cylindrical one, it is

$$E_{HandleCylinder,\,i} = \frac{1}{2} k_{Handle} \left(\left(\frac{\mathbf{x}_i + \mathbf{x}_{i+1}}{2} - \mathbf{x}_{Handle} \right)^2 + \left(\mathfrak{q}_i - \mathfrak{q}_{Handle} \right)^2 \right)$$

A.6.2 Forces

Spherical Handle

$$\begin{aligned} F_{HandleSphere,\,x_{i,j}} &= -\frac{\partial E_{HandleSphere,\,i}}{\partial x_{i,j}} = -k_{Handle} \mathbf{e}_j^T \left(\mathbf{x}_i - \mathbf{x}_{Handle} \right) \\ &= k_{Handle} \left(x_{Handle,j} - x_{i,j} \right) \end{aligned}$$

$$\mathbf{F}_{HandleSphere,\,\mathbf{x}_i} = -k_{Handle} \left(\mathbf{x}_i - \mathbf{x}_{Handle} \right)$$

$$\begin{aligned} F_{HandleSphere,\,q_{i,j}} &= -\frac{\partial E_{HandleSphere,\,i}}{\partial q_{i,j}} \\ &= -k_{Handle} \left(\frac{\partial \frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{\left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\|}}{\partial \mathfrak{q}_{i,j}} \right)^T \left(\frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{\left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\|} - \mathfrak{q}_{Handle} \right) \\ &= k_{Handle} \left(\frac{\mathfrak{e}_j \left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\| - \left(\mathfrak{q}_{i-1}+\mathfrak{q}_i \right) \frac{1}{2} \frac{2\left(q_{i,j}+q_{i+1,j} \right)}{\left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\|}}{\left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\|^2} \right)^T \mathfrak{q}_{Handle} \\ &\quad \text{because } \frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{\left\| \mathfrak{q}_{i-1}+\mathfrak{q}_i \right\|} \text{ is unitary} \\ &= k_{Handle} \left(\frac{\mathfrak{e}_j \left\| \mathfrak{p} \right\| - \mathfrak{p} \frac{p_j}{\left\| \mathfrak{p} \right\|}}{\left\| \mathfrak{p} \right\|^2} \right)^T \mathfrak{q}_{Handle} \quad \text{with } \mathfrak{p} = \mathfrak{q}_{i-1} + \mathfrak{q}_i \\ &= k_{Handle} \frac{1}{\left\| \mathfrak{p} \right\|} \left(\mathfrak{e}_j - \frac{\mathfrak{p} p_j}{\left\| \mathfrak{p} \right\|^2} \right)^T \mathfrak{q}_{Handle} \end{aligned}$$

or

$$\begin{aligned} F_{HandleSphere,\,q_{i,j}} &= -\frac{\partial E_{HandleSphere,\,i}}{\partial q_{i,j}} \\ &= -k_{Handle}\left(\frac{\partial \frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{2}}{\partial \mathfrak{q}_{i,j}}\right)^T \left(\frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{2} - \mathfrak{q}_{Handle}\right) \\ &= -k_{Handle}\left(\frac{\mathfrak{e}_j}{2}\right)^T \left(\frac{\mathfrak{q}_{i-1}+\mathfrak{q}_i}{2} - \mathfrak{q}_{Handle}\right) \\ &= -\frac{k_{Handle}}{2}\left(\frac{q_{i-1,j}+q_{i,j}}{2} - q_{Handle,j}\right) \\ &= -\frac{\partial E_{HandleSphere,\,i}}{\partial q_{i-1,j}} \end{aligned}$$

Cylindrical Handle

$$-\frac{\partial E_{HandleCylinder,\,i}}{\partial x_{i,j}} = -\frac{k_{Handle}}{2}\left(\frac{x_{i,j}+x_{i+1,j}}{2} - x_{Handle,j}\right) = -\frac{\partial E_{HandleCylinder,\,i}}{\partial x_{i+1,j}}$$

$$-\frac{\partial E_{HandleCylinder,\,i}}{\partial \mathfrak{q}_{i,j}} = k_{Handle}\left(q_{Handle,j} - q_{i,j}\right)$$

A.6.3 Hessian

Since the energy can be decomposed in two terms, one depending on the position(s), the other on the quaternion(s), there are no mixed-type terms (like $\frac{\partial^2 E_{Handle,i}}{\partial x_{i,j}\partial q_{i,k}}$) in the Hessian.

Spherical Handle

$$\frac{\partial^2 E_{HandleSphere,i}}{\partial x_{i,j}\partial x_{i,k}} = k_{Handle}\delta_j^k$$

$$\begin{aligned} \frac{\partial^2 E_{HandleSphere,i}}{\partial q_{i,j}\partial q_{i,k}} &= \frac{\partial^2 E_{HandleSphere,i}}{\partial q_{i-1,j}\partial q_{i,k}} = \frac{\partial^2 E_{HandleSphere,i}}{\partial q_{i,j}\partial q_{i-1,k}} = \frac{\partial^2 E_{HandleSphere,i}}{\partial q_{i-1,j}\partial q_{i-1,k}} \\ &= \frac{k_{Handle}}{4}\delta_j^k \end{aligned}$$

Cylindrical Handle

$$\begin{aligned} \frac{\partial^2 E_{HandleCylinder,i}}{\partial x_{i,j}\partial x_{i,k}} &= \frac{\partial^2 E_{HandleCylinder,i}}{\partial x_{i+1,j}\partial x_{i,k}} = \frac{\partial^2 E_{HandleCylinder,i}}{\partial x_{i,j}\partial x_{i+1,k}} = \frac{\partial^2 E_{HandleCylinder,i}}{\partial x_{i+1,j}\partial x_{i+1,k}} \\ &= \frac{k_{Handle}}{4}\delta_j^k \end{aligned}$$

$$\frac{\partial^2 E_{HandleCylinder,i}}{\partial q_{i,j}\partial q_{i,k}} = k_{Handle}\delta_j^k$$

A.7 Bending and Torsion

A.7.1 Energy

The calculation of the bending and torsion is complexer than for other interactions. The energy is defined as

$$E_{BendingTorsion,i} = E_{Bending,i} + E_{torsion,i} = \frac{1}{2} B_\kappa \kappa^2 + \frac{1}{2} B_\tau \tau^2$$

The Darboux vector in local coordinates is then

$$\boldsymbol{\omega_l} = \frac{2}{L} \overrightarrow{\overline{\mathfrak{q}}_i \cdot \mathfrak{q}_{i+1}} = (\kappa_1, \tau, \kappa_2)$$

where $\kappa^2 = \kappa_1^2 + \kappa_2^2$.

The energy can be then rewritten as

$$\begin{aligned}
E_{BendingTorsion,i} &= \tfrac{1}{2} B_\kappa \kappa^2 + \tfrac{1}{2} B_\tau \tau^2 \\
&= \tfrac{1}{2} B_\kappa \left(\|\omega\|^2 - \tau^2\right) + \tfrac{1}{2} B_\tau \tau^2 \\
&= \tfrac{1}{2} B_\kappa \|\omega\|^2 + \tfrac{1}{2} \left(B_\tau - B_\kappa\right) \tau^2 \\
&= \tfrac{1}{2} B_\kappa \|\omega\|^2 + \tfrac{1}{2} \left(B_\tau - B_\kappa\right) (\omega^T \mathbf{ref})^2 \\
&= \tfrac{1}{2} \left(B_\kappa \omega^T \omega + \left(B_\tau - B_\kappa\right) \omega^T \mathbf{ref} \mathbf{ref}^T \omega\right) \\
&= \tfrac{1}{2} \omega^T (B_\kappa \mathbf{I}_3 + \left(B_\tau - B_\kappa\right) \mathbf{ref} \mathbf{ref}^T) \omega \\
&= \tfrac{1}{2} \omega^T \mathbf{A} \omega
\end{aligned}$$

with

$$\mathbf{A} = (B_\kappa \mathbf{I}_3 + \left(B_\tau - B_\kappa\right) \mathbf{ref} \mathbf{ref}^T) = \begin{pmatrix} B_\kappa & 0 & 0 \\ 0 & B_\tau & 0 \\ 0 & 0 & B_\kappa \end{pmatrix}$$

A.7.2 Forces

Let us proceed step by step. The relative rotation is

$$\overline{\mathfrak{q}_i} \cdot \mathfrak{q}_{i+1} = \begin{pmatrix} q_{i,0}\, q_{i+1,0} & + & q_{i,1}\, q_{i+1,1} & + & q_{i,2}\, q_{i+1,2} & + & q_{i,3}\, q_{i+1,3}, \\ q_{i,0}\, q_{i+1,1} & - & q_{i,1}\, q_{i+1,0} & - & q_{i,2}\, q_{i+1,3} & + & q_{i,3}\, q_{i+1,2}, \\ q_{i,0}\, q_{i+1,2} & - & q_{i,2}\, q_{i+1,0} & - & q_{i,3}\, q_{i+1,1} & + & q_{i,1}\, q_{i+1,3}, \\ q_{i,0}\, q_{i+1,3} & - & q_{i,3}\, q_{i+1,0} & - & q_{i,1}\, q_{i+1,2} & + & q_{i,2}\, q_{i+1,1} \end{pmatrix}$$

We have

$$\mathbf{B} = \left(\frac{\partial (\overline{\mathfrak{q}_i} \cdot \mathfrak{q}_{i+1})_j}{\partial q_{i,\,k}} \right)_{\substack{1 \le j \le 4 \\ 1 \le k \le 4}} = \begin{pmatrix} q_{i+1,0} & q_{i+1,1} & q_{i+1,2} & q_{i+1,3} \\ q_{i+1,1} & -q_{i+1,0} & -q_{i+1,3} & q_{i+1,2} \\ q_{i+1,2} & q_{i+1,3} & -q_{i+1,0} & -q_{i+1,1} \\ q_{i+1,3} & -q_{i+1,2} & q_{i+1,1} & -q_{i+1,0} \end{pmatrix}$$

$$\mathbf{C} = \left(\frac{\partial q_{i \to i+1,\, j}}{\partial q_{i+1,\, k}} \right)_{\substack{1 \le j \le 4 \\ 1 \le k \le 4}} = \begin{pmatrix} q_{i,0} & q_{i,1} & q_{i,2} & q_{i,3} \\ -q_{i,1} & q_{i,0} & q_{i,3} & -q_{i,2} \\ -q_{i,2} & -q_{i,3} & q_{i,0} & q_{i,1} \\ -q_{i,3} & q_{i,2} & -q_{i,1} & q_{i,0} \end{pmatrix}$$

and

$$\frac{\partial \omega_j}{\partial q_{i,\, k}} = \frac{2}{L} \frac{\partial (\overline{\mathfrak{q}_i} \cdot \mathfrak{q}_{i+1})_j}{\partial q_{i,\, k}}$$

$$\frac{\partial \omega_j}{\partial q_{i+1,\, k}} = \frac{2}{L} \frac{\partial (\overline{\mathfrak{q}_i} \cdot \mathfrak{q}_{i+1})_j}{\partial q_{i+1,\, k}}$$

Since the energy is $E_{BendingTorsion,i} = \frac{1}{2}\, \omega^T \mathbf{A} \omega$ the corresponding force is then

$$F_{BendingTorsion,i,q_{i,j}} = -\frac{\partial \omega}{\partial q_{i,j}}^T \mathbf{A} \omega$$

since $\mathbf{A}$ is symmetric, and similarly

$$F_{BendingTorsion,i,q_{i+1,j}} = -\frac{\partial \omega}{\partial q_{i+1,j}}^T \mathbf{A} \omega$$

A.7.3 Hessian

We have three kind of terms to consider: $\frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i,j} \partial q_{i,k}}$, $\frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i+1,j} \partial q_{i+1,k}}$ and $\frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i,j} \partial q_{i+1,k}}$.

$$\begin{aligned} \frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i,j} \partial q_{i,k}} &= \frac{\partial}{\partial q_{i,j}} \frac{\partial E_{BendingTorsion,i}}{\partial q_{i,k}} \\ &= \frac{\partial}{\partial q_{i,j}} \left(\frac{\partial \omega}{\partial q_{i,k}}^T \mathbf{A} \omega \right) \\ &= \frac{\partial^2 \omega}{\partial q_{i,j} \partial q_{i,k}}^T \mathbf{A} \omega + \frac{\partial \omega}{\partial q_{i,j}}^T \mathbf{A} \frac{\partial \omega}{\partial q_{i,k}} \\ &= \frac{\partial \omega}{\partial q_{i,j}}^T \mathbf{A} \frac{\partial \omega}{\partial q_{i,k}} \end{aligned}$$

since $\frac{\partial^2 \omega}{\partial q_{i,j} \partial q_{i,k}} = 0$.

$$\begin{aligned} \frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i+1,j} \partial q_{i+1,k}} &= \frac{\partial}{\partial q_{i+1,j}} \frac{\partial E_{BendingTorsion,i}}{\partial q_{i+1,k}} \\ &= \frac{\partial}{\partial q_{i+1,j}} \left(\frac{\partial \omega}{\partial q_{i+1,k}}^T \mathbf{A} \omega \right) \\ &= \frac{\partial^2 \omega}{\partial q_{i+1,j} \partial q_{i+1,k}}^T \mathbf{A} \omega + \frac{\partial \omega}{\partial q_{i+1,j}}^T \mathbf{A} \frac{\partial \omega}{\partial q_{i+1,k}} \\ &= \frac{\partial \omega}{\partial q_{i+1,j}}^T \mathbf{A} \frac{\partial \omega}{\partial q_{i+1,k}} \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 E_{BendingTorsion,i}}{\partial q_{i,j} \partial q_{i+1,k}} &= \frac{\partial}{\partial q_{i,j}} \frac{\partial E_{BendingTorsion,i}}{\partial q_{i+1,k}} \\ &= \frac{\partial}{\partial q_{i,j}} \left(\frac{\partial \omega}{\partial q_{i+1,k}}^T \mathbf{A} \omega \right) \\ &= \frac{\partial^2 \omega}{\partial q_{i,j} \partial q_{i+1,k}}^T \mathbf{A} \omega + \frac{\partial \omega}{\partial q_{i,j}}^T \mathbf{A} \frac{\partial \omega}{\partial q_{i+1,k}} \end{aligned}$$

with

We have

$$\mathcal{H}_{BendingTorsion,\mathfrak{q}_i,\mathfrak{q}_i} = \frac{4}{L^2} B^T_{\substack{2\leq j\leq 4\\1\leq k\leq 4}} A B_{\substack{2\leq j\leq 4\\1\leq k\leq 4}}$$

$$\mathcal{H}_{BendingTorsion,\mathfrak{q}_{i+1},\mathfrak{q}_{i+1}} = \frac{4}{L^2} C^T_{\substack{2\leq j\leq 4\\1\leq k\leq 4}} A C_{\substack{2\leq j\leq 4\\1\leq k\leq 4}}$$

A.8 Rotation speed - Darboux Vector

Is is well known that for $\boldsymbol{\omega}$ the Darboux vector, the speed $\mathbf{v}$ of a point $\mathbf{x}$ is

$$\mathbf{v} = \boldsymbol{\omega} \times \mathbf{x}$$

At the same time, if $\mathbf{x}$ is the image by the rotation represented by $\mathfrak{q}$ of $\mathbf{y}$ we have

$$\begin{aligned}
(0,\mathbf{v}) &= (0,\tfrac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}) \\
&= \tfrac{\mathrm{d}}{\mathrm{d}t}(\mathfrak{q}\cdot(0,\mathbf{y})\cdot\overline{\mathfrak{q}}) \\
&= \dot{\mathfrak{q}}\cdot(0,\mathbf{y})\cdot\overline{\mathfrak{q}} + \mathfrak{q}\cdot(0,\mathbf{y})\cdot\overline{\dot{\mathfrak{q}}} \\
&= \dot{\mathfrak{q}}\cdot(\overline{\mathfrak{q}}\cdot(0,\mathbf{x})\cdot\mathfrak{q})\cdot\overline{\mathfrak{q}} + \mathfrak{q}\cdot(\overline{\mathfrak{q}}\cdot(0,\mathbf{x})\cdot\mathfrak{q})\cdot\overline{\dot{\mathfrak{q}}} \\
&= \dot{\mathfrak{q}}\cdot\overline{\mathfrak{q}}\cdot(0,\mathbf{x}) + (0,\mathbf{x})\cdot\mathfrak{q}\cdot\overline{\dot{\mathfrak{q}}}
\end{aligned}$$

For $\mathfrak{a}$ and $\mathfrak{b}$ two quaternions, we have

$$\begin{aligned}
\mathfrak{a}\cdot\mathfrak{b} &= (a_0b_0 - \mathbf{a}^T\mathbf{b}, a_0\mathbf{b} + b_0\mathbf{a} + \mathbf{a}\times\mathbf{b}) \\
\mathfrak{b}\cdot\overline{\mathfrak{a}} &= (a_0b_0 + \mathbf{a}^T\mathbf{b}, a_0\mathbf{b} - b_0\mathbf{a} + \mathbf{a}\times\mathbf{b}) \\
\mathfrak{a}\cdot\mathfrak{b} + \mathfrak{b}\cdot\overline{\mathfrak{a}} &= 2(a_0b_0, a_0\mathbf{b} + \mathbf{a}\times\mathbf{b})
\end{aligned}$$

With $\mathfrak{a} = \dot{\mathfrak{q}}\cdot\overline{\mathfrak{q}}$ and $\mathfrak{b} = (0,\mathbf{x})$, we obtain, since $a_0 = \sum_{i=0}^{3}\dot{q}_iq_i = 0$ (the quaternion has a constant norm),

$$(0,\mathbf{v}) = (0, 2\mathbf{a}\times\mathbf{x})$$

and

$$(0,\boldsymbol{\omega_g}) = 2\dot{\mathfrak{q}}\cdot\overline{\mathfrak{q}}$$

This is expressed in world coordinates. In the local coordinates, we have

$$\begin{aligned}
(0,\boldsymbol{\omega_l}) &= \overline{\mathfrak{q}}\cdot(0,\boldsymbol{\omega_g})\cdot\mathfrak{q} \\
&= 2\overline{\mathfrak{q}}\cdot\dot{\mathfrak{q}}\cdot\overline{\mathfrak{q}}\cdot\mathfrak{q} \\
&= 2\overline{\mathfrak{q}}\cdot\dot{\mathfrak{q}}
\end{aligned}$$

A.9 Discretization of the rotation speed

A.9.1 slerp

The slerp defines a rotation at constant speed from a quaternion $\mathfrak{q}_0$ to a quaternion $\mathfrak{q}_1$:

$$\mathfrak{q}(t) = \frac{\sin(\alpha(1-t))\mathfrak{q}_0 + \sin(\alpha t)\mathfrak{q}_1}{\sin\alpha}$$

where $\cos\alpha = \mathfrak{q}_0^T\mathfrak{q}_1$ and $t \in [0;1]$. It is interesting to note that

$$(\mathfrak{q}_1 \cdot \overline{\mathfrak{q}_0})_0 = q_{1,0}q_{0,0} + \mathbf{q}_1^T\mathbf{q}_0 = \cos\alpha$$

If the relative rotation $\mathfrak{q}_1 \cdot \overline{\mathfrak{q}_0}$ is a rotation of an angle θ around an axis $\mathbf{v}, \|\mathbf{v}\| = 1$, we get $\cos\alpha = \cos\frac{\theta}{2}$ and $\alpha = \frac{\theta}{2}$.

In particular, for $t = \frac{1}{2}$, we have

$$\mathfrak{q}_{\frac{1}{2}} = \mathfrak{q}\left(\frac{1}{2}\right) = \frac{\sin(\frac{1}{2}\alpha)}{\sin\alpha}(\mathfrak{q}_0 + \mathfrak{q}_1) = \frac{1}{2\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_0 + \mathfrak{q}_1)$$

The rotation from $\mathfrak{q}_0$ to $\mathfrak{q}_{\frac{1}{2}}$ is equal to the rotation from $\mathfrak{q}_{\frac{1}{2}}$ to $\mathfrak{q}_1$

$$\mathfrak{q}_{\frac{1}{2}} \cdot \overline{\mathfrak{q}_0} = \mathfrak{q}_1 \cdot \overline{\mathfrak{q}_{\frac{1}{2}}} = \frac{1}{2\cos(\frac{1}{2}\alpha)}((1,0,0,0) + \mathfrak{q}_1 \cdot \overline{\mathfrak{q}_0})$$

In particular, for $\mathfrak{q}_0 = (1,0,0,0)$, $\cos\alpha = q_{1,0}$, $\mathfrak{q}_{\frac{1}{2}} \cdot \mathfrak{q}_{\frac{1}{2}} = \mathfrak{q}_1$: it is a kind of "square root"

A.9.2 Discretization of the speed of rotation in world coordinates

Let's consider the quaternions $\mathfrak{q}_i$ and $\mathfrak{q}_{i+1}$. We want to define an approximation for ω We can write, with L the length of a segment,

$$\dot{\mathfrak{q}} = \frac{\mathfrak{q}_{i+1} - \mathfrak{q}_i}{L}$$

and

$$\mathfrak{q} = \frac{1}{2\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_i + \mathfrak{q}_{i+1})$$

interpolating the quaternion value.

In global coordinates, we get

$$\begin{aligned}(0,\boldsymbol{\omega_g}) &= 2\dot{\mathfrak{q}}\cdot\overline{\mathfrak{q}}\\ &= 2\tfrac{\mathfrak{q}_{i+1}-\mathfrak{q}_i}{L}\cdot\overline{\tfrac{1}{2\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_i+\mathfrak{q}_{i+1})}\\ &= \tfrac{1}{L\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_{i+1}-\mathfrak{q}_i)\cdot(\overline{\mathfrak{q}}_i+\overline{\mathfrak{q}}_{i+1})\\ &= \tfrac{1}{L\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_{i+1}\cdot\overline{\mathfrak{q}}_i+(1,0,0,0)-(1,0,0,0)-\mathfrak{q}_i\cdot\overline{\mathfrak{q}}_{i+1})\\ &= \tfrac{1}{L\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_{i+1}\cdot\overline{\mathfrak{q}}_i-\mathfrak{q}_i\cdot\overline{\mathfrak{q}}_{i+1})\end{aligned}$$

But for two quaternions $\mathfrak{a}$ and $\mathfrak{b}$, we have

$$\begin{aligned}\mathfrak{a}\cdot\overline{\mathfrak{b}} &= (a_0b_0+\mathbf{a}^T\mathbf{b},-a_0\mathbf{b}+b_0\mathbf{a}-\mathbf{a}\times\mathbf{b})\\ \mathfrak{b}\cdot\overline{\mathfrak{a}} &= (a_0b_0+\mathbf{a}^T\mathbf{b},a_0\mathbf{b}-b_0\mathbf{a}+\mathbf{a}\times\mathbf{b})\\ \mathfrak{a}\cdot\overline{\mathfrak{b}}-\mathfrak{b}\cdot\overline{\mathfrak{a}} &= 2(0,-a_0\mathbf{b}+b_0\mathbf{a}-\mathbf{a}\times\mathbf{b})=2(0,\overrightarrow{\mathfrak{a}\cdot\overline{\mathfrak{b}}})\end{aligned}$$

Since the scalar part of the quaternion is 0 in both cases, we have

$$\boldsymbol{\omega_g}=\frac{2}{L\cos(\frac{1}{2}\alpha)}\overrightarrow{\mathfrak{q}_{i+1}\cdot\overline{\mathfrak{q}}_i}$$

If $\mathfrak{q}_{i+1}\cdot\overline{\mathfrak{q}}_i=(\cos\frac{\theta}{2},\sin\frac{\theta}{2}\mathbf{v})$, $\alpha=\frac{\theta}{2}$ (see A.9.1) and

$$\boldsymbol{\omega_g}=\frac{2}{L\cos(\frac{\theta}{4})}\sin\frac{\theta}{2}\mathbf{v}$$

Since θ can be expected to be small, we finally have

$$\boldsymbol{\omega_g}=\frac{2}{L}\sin\frac{\theta}{2}\mathbf{v}$$

$$\boldsymbol{\omega_g}=\frac{2}{L}\overrightarrow{\mathfrak{q}_{i+1}\cdot\overline{\mathfrak{q}}_i}$$

A.9.3 Discretization of the speed of rotation in local coordinates

In a similar way to the previous section, we have in local coordinates

$$\begin{aligned}(0,\boldsymbol{\omega_l}) &= 2\overline{\mathfrak{q}}\cdot\dot{\mathfrak{q}}\\ &= 2\overline{\tfrac{1}{2\cos(\frac{1}{2}\alpha)}(\mathfrak{q}_i+\mathfrak{q}_{i+1})}\cdot\tfrac{\mathfrak{q}_{i+1}-\mathfrak{q}_i}{L}\\ &= \tfrac{1}{L\cos(\frac{1}{2}\alpha)}(\overline{\mathfrak{q}}_i\cdot\mathfrak{q}_{i+1}-\overline{\mathfrak{q}}_{i+1}\cdot\mathfrak{q}_i)\end{aligned}$$

and

$$\boldsymbol{\omega_l}=\frac{2}{L\cos(\frac{1}{2}\alpha)}\overrightarrow{\overline{\mathfrak{q}}_i\cdot\mathfrak{q}_{i+1}}$$

With α small

$$\boldsymbol{\omega_l}=\frac{2}{L}\overrightarrow{\overline{\mathfrak{q}}_i\cdot\mathfrak{q}_{i+1}}$$

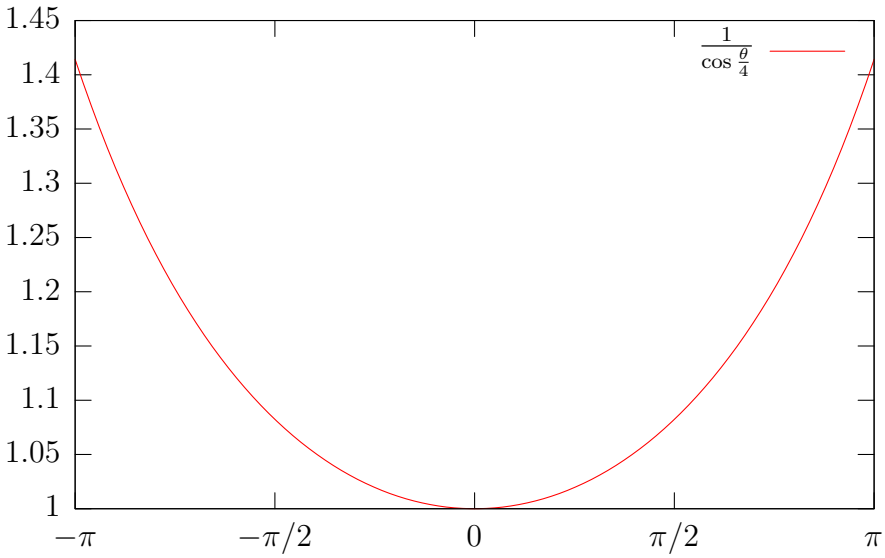

Figure A.1: Values of $\frac{1}{\cos\frac{\theta}{4}}$: for small values of θ is 1 a good approximation.

Appendix B

Appendix - Constants values

In this appendix we want to show the influence on the values of the energy constants of several factors: the number of discretization points and the choice of the length unit (ie. if the scene is in meters or centimeters) as well as the scale (the influence of the length of the cable, which is useful if we want to modify the length of a cable without modifying its other properties). We want our simulated cable to show the same behavior independently of these factors. For achieving it, we need to adapt the values of the constants. The easiest way of doing this is to keep the energy constant. The key idea is that the energy per length unit has to be constant, which corresponds to the additivity principle. We will consider a continuous version of the energy for the different interactions and compare it with the discrete version to find the relations we need. K is the constant per unit length for the continuous version, and k for the discrete version (depending on the length L and the number of discretization points n).

B.1 Length conservation

Defining γ the extension rate such that $L = (1+\gamma)L_{ref}$, we have a continuous definition of the energy

$$\begin{aligned} E &= \tfrac{1}{2}K\int_0^L \gamma^2 \mathrm{d}s \\ &= \tfrac{1}{2}KL\gamma_{eff}^2 \end{aligned}$$

and a discrete definition

$$\begin{aligned} E &= \tfrac{1}{2}\sum_{i=1}^{n-1} k_L(L_i - L_{i,0})^2 \\ &= \tfrac{1}{2}\sum_{i=1}^{n-1} k_L L_{i,0}^2\gamma_i^2 \\ &= \tfrac{1}{2}(n-1)k_L\tfrac{L^2}{(n-1)^2}\gamma_{eff}^2 \\ &= \tfrac{1}{2}k_L\tfrac{L^2}{(n-1)}\gamma_{eff}^2 \end{aligned}$$

Consequently, we have

$$k_L = K\frac{n-1}{L}$$

For a change of unit, we have $x_{new} = \alpha x_{old}$ (for example, if x_{new} is in centimeters and x_{old} in millimeters, $\alpha = 0.1$.) The new energy is

$$E_{new} = \frac{1}{2}\sum_{i=1}^{n-1} k_{L,new}(L_{i,new} - L_{i,0,new})^2 = \frac{1}{2}\sum_{i=1}^{n-1} k_{L,new}(L_i - L_{i,0})^2\alpha^2 = \alpha^2 E_{old}\frac{k_{L,new}}{k_{L,old}}$$

so that

$$k_{L,new} = \frac{k_{L,old}}{\alpha^2}$$

B.2 Weight

With μ the linear mass density and $h(s)$ the height at arclength s, the continuous energy is

$$E = \int_0^L \mu g h \mathrm{d}s = \mu g \bar{h} L$$

and is approximated by

$$E = \sum_{i=1}^{n} k_m h_i = n k_m \bar{h}$$

It follows that

$$k_m = \mu g \frac{L}{n}$$

For a change of unit length, we have

$$E_{new} = n k_{m,new} \alpha \bar{h} = n k_{m,old} \bar{h}$$

and

$$k_{m,new} = \frac{k_{m,old}}{\alpha}$$

B.3 Quaternion Norm

Similarly, we have

$$E = \frac{1}{2}K_n \int_0^L \|\mathfrak{q}\|^2 \mathrm{d}s = \frac{1}{2}K_n L \|\mathfrak{q}_{eff}\|^2$$

and

$$E = \frac{1}{2}\sum_{i=1}^{n-1} k_n \|\mathfrak{q}_i\|^2 = \frac{1}{2}(n-1)k_n\|\mathfrak{q}_{eff}\|^2$$

Therefore,

$$k_n = K_n \frac{L}{n-1}$$

There is no change relative to the unit length.

$$k_{n,old} = k_{n,new}$$

B.4 Coherence

We have

$$E = \frac{1}{2}K_{xq}\int_0^L (\|\mathbf{T} - rot(\mathfrak{q})\|)^2 \mathrm{d}s = \frac{1}{2}K_{xq}\int_0^L \|\mathbf{\Delta}\|^2 \mathrm{d}s = \frac{1}{2}K_{xq}L\|\mathbf{\Delta}\|^2_{eff}$$

and

$$E = \frac{1}{2}\sum_{i=1}^{n-1} k_{xq}(\|\mathbf{u}_i - rot(\mathfrak{q}_i)\|)^2 = \frac{1}{2}\sum_{i=1}^{n-1} k_{xq}\|\mathbf{\Delta_i}\|^2 = \frac{1}{2}(n-1)k_{xq}\|\mathbf{\Delta}\|^2_{eff}$$

It follows

$$k_{xq} = K_{xq}\frac{L}{n-1}$$

There is no change relative to the unit length.

$$k_{xq,old} = k_{xq,new}$$

B.5 Curvature and torsion

We have, for the curvature,

$$E = \frac{1}{2}K\int_0^L \kappa^2 \mathrm{d}s = \frac{1}{2}KL\kappa^2_{eff}$$

and

$$E = \frac{1}{2}\sum_{i=1}^{n-2} k_\kappa \kappa_i^2 = \frac{1}{2}(n-2)k_\kappa\kappa^2_{eff}$$

It follows

$$k_\kappa = K_\kappa \frac{L}{n-2}$$

For a change of unit length, we have

$$E_{new} = \frac{1}{2}\sum_{i=1}^{n-2} k_\kappa \frac{1}{\alpha^2}\kappa_i^2$$

and

$$k_{\kappa,new} = \alpha^2 k_{\kappa,old}$$

B.6 Summary

If we combine the information before, if n and L are the new number of points and length, and $\mathcal{N}$ and $\mathcal{L}$ the old values, the new constants will be:

$$k_{l,new} = \frac{1}{\alpha^2}\frac{n-1}{\mathcal{N}-1}\frac{\mathcal{L}}{L}k_{l,old}$$

$$k_{m,new} = \frac{1}{\alpha}\frac{\mathcal{N}}{n}\frac{L}{\mathcal{L}}k_{m,old}$$

$$k_{n,new} = \frac{\mathcal{N}-1}{n-1}\frac{L}{\mathcal{L}}k_{n,old}$$

$$k_{xq,new} = \frac{\mathcal{N}-1}{n-1}\frac{L}{\mathcal{L}}k_{xq,old}$$

$$k_{\kappa,new} = \alpha^2\frac{\mathcal{N}-2}{n-2}\frac{L}{\mathcal{L}}k_{\kappa,old}$$

The length constant is the only one augmenting with the number of points. (See Figure B.3: the curves for different number of points are similar, exept for $n = 20$ where the first segment is too large.)

B.7 Influence of the parameters on the condition number of the Hessian

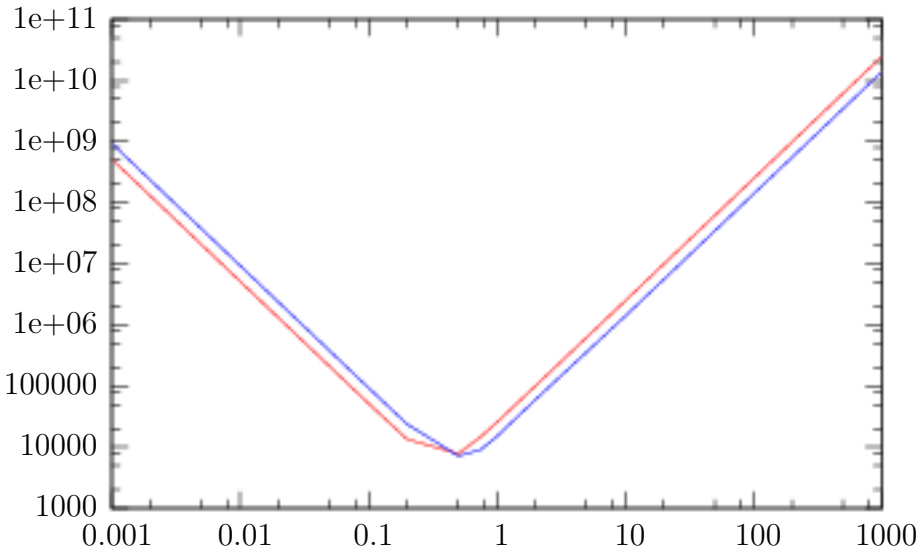

Figure B.1: Influence of the factor α on the condition number of the Hessian for two diferent values of k_κ and k_τ : the minimum is around $\alpha = 0.1...1$, where the positions and quaternions have the same order of magnitude..

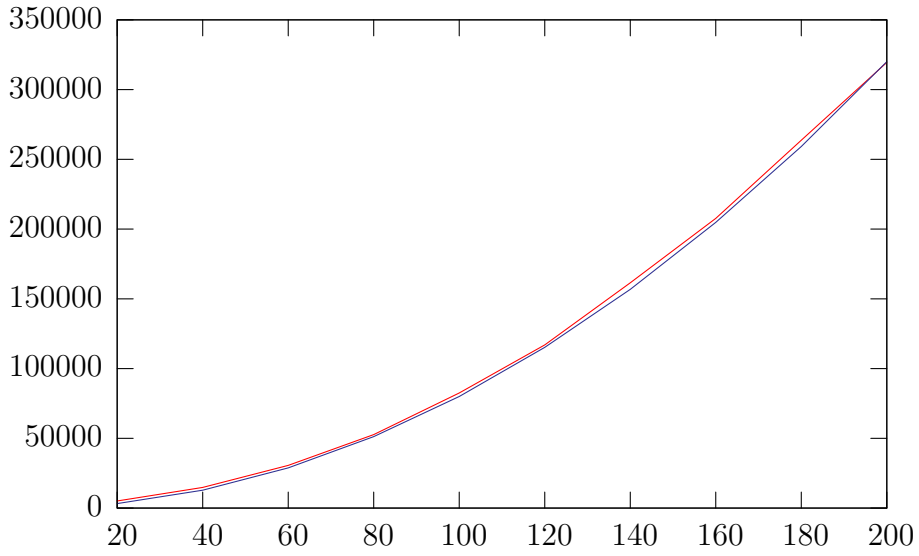

Figure B.2: Influence of the number of discretization points on the condition number of the Hessian: the condition number (in red) grows like the square of the number of points (in green)

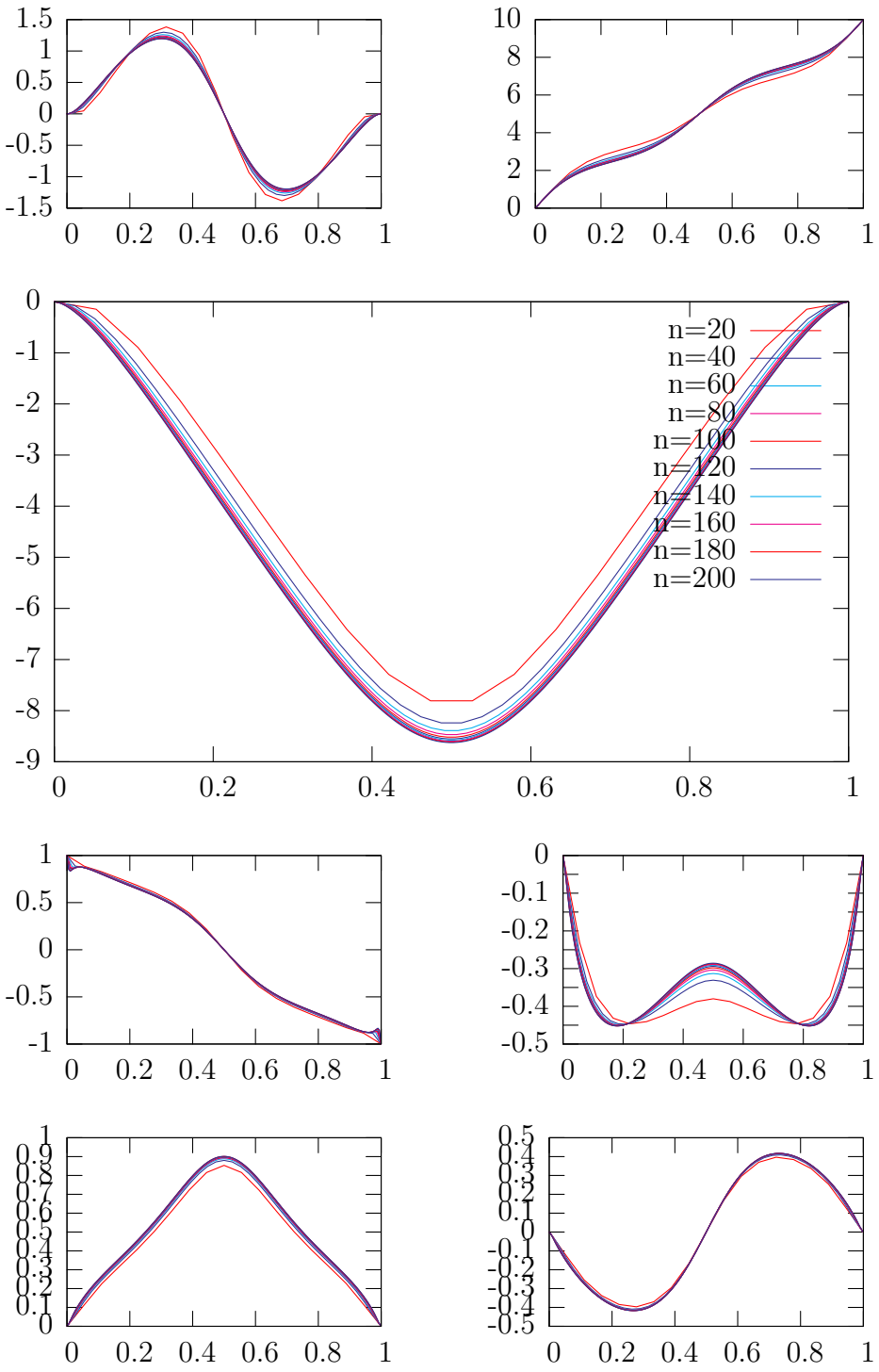

Figure B.3: The different coordinates for various number of discretization points: apart of $n = 20$, all the curves are similar: the transformations as shown above conserve the properties of the cable.

Bibliography

[Adrien Theetten and Barsky, 2006] Adrien Theetten, Laurent Grison, i. C. A. and Barsky, B. (2006). Geometrycally exact dynamic splines. Technical report, INRIA.

[Antman, 1995] Antman, S. S. (1995). *Nonlinear Problems of Elasticity*, volume 107 of *Applied Mathematical Sciences*. Springer Verlag.

[Baraff, 1996] Baraff, D. (1996). Linear-time dynamics using Lagrange multipliers. *Computer Graphics*, 30(Annual Conference Series):137–146.

[Baraff and Witkin, 1998] Baraff, D. and Witkin, A. (1998). Large steps in cloth simulation. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 43–54, New York, NY, USA. ACM Press.

[Barzel, 1997] Barzel, R. (1997). Faking dynamics of ropes and springs. *IEEE Computer Graphics and Application*, 17(3)(3):31–39.

[Bertails et al., 2006] Bertails, F., Audoly, B., Cani, M.-P., Querleux, B., Leroy, F., and Lévêque, J.-L. (2006). Super-helices for predicting the dynamics of natural hair. pages 1180–1187.

[Bertails et al., 2005] Bertails, F., Audoly, B., Querleux, B., Leroy, F., Lévêque, J.-L., and Cani, M.-P. (2005). Predicting natural hair shapes by solving the statics of flexible rods. In Dingliana, J. and Ganovelli, F., editors, *Eurographics (short papers)*. Eurographics.

[Bertails et al., 2003] Bertails, F., Kim, T.-Y., Cani, M.-P., and Neumann, U. (2003). Adaptive wisp tree: a multiresolution control structure for simulating dynamic clustering in hair motion. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 207–213, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

[Brown, 2003] Brown, J. (2003). *Real-time soft Tissue and suture simulation*. PhD thesis, Computer Science, Stanford University.

[Buck and Schömer, 1998] Buck, M. and Schömer, E. (1998). Interactive rigid body manipulation with obstacle contacts. *Journal of Visualization and Computer Animation*, 9:243–257.

[Finckh et al., 2004] Finckh, H., Stegmaier, T., and Planck, P. H. (2004). Numerische Simulation der mechanischen Eigenschaften textiler Flächengebilde - Gewebeherstellung. In *3. LS-DYNA Anwenderforum 2004, 14.-15. Oktober 2004, Bamberg*.

[Fleischer, 2005] Fleischer, S. (2005). Statusbericht flexible Bauteile.

[Goss et al., 2005] Goss, V., van der Heijdenand J.M.T. Thompson, G., and Neukirch, S. (2005). Experiments on snap buckling, hysteresis and loop formation in twisted rods. *Experimental Mechanics*, 45:101–111.

[Hergenroether and Daehne, 2000] Hergenroether, E. and Daehne, P. (2000). Real-time virtual cables based on kinematic simulation. In *Proceedings of the WSCG 2000*.

[Julien Lemoir, 2006] Julien Lemoir, Stephane Cotin, C. D. P. N. (2006). Interactive physically-based simulation of cathether and guidewire. *COmputers and Graphics*, 30:417–423.

[Loock and Schömer, 2001] Loock, A. and Schömer, E. (2001). A virtual environment for interactive assembly simulation: From rigid bodies to deformable cables. 3:325–332.

[Magnenat-Thalmann and Hadap., 2000] Magnenat-Thalmann, N. and Hadap., S. (2000). State of the art in hair simulation. In *International Workshop on Human Modeling and Animation*, pages 3–9. Korea Computer Graphics Society.

[Mikchevitch A., 2003] Mikchevitch A., J.-C. Ln, A. G. (2003). Realistic force simulation in path planning for virtual assembly of flexible beam parts. In *Virtual Concept 2003 Biarritz - France November, 5-7*.

[Mikchevitch A. and Gouskov, 2003] Mikchevitch A., J.-C. L. and Gouskov, A. (2003). Numerical modeling of flexible components for assembly path planning using a virtual reality environment. In *Proceedings of DETC03 ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference Chicago, Illinois USA, September 2-6, 2003*.

[Moll and Kavraki, 2006] Moll, M. and Kavraki, L. E. (2006). Path planning for deformable linear objects. *IEEE Transactions on Robotics*, 22(4):625–636.

[Moll et al., 2005] Moll, M., Kavraki, L. E. I. P. o. T. I. I. C. o. R., and (ICRA), A. (2005). Path planning for variable resolution minimal-energy curves of constant length. In *Proceedings of The IEEE International Conference on Robotics and Automation (ICRA), Barcelona, Spain*, page 21432147, April. IEEE Press.

[Nikitin et al., 2002] Nikitin, I., Nikitina, L., Frolov, P., Goebbels, G., Göbel, M., Klimenko, S., and Nielson, G. M. (2002). Real-time simulation of elastic objects in virtual environments using finite element method and precomputed green's functions. pages 47–52.

[Pai, 2002] Pai, D. K. (2002). Strands: Interactive simulation of thin solids using cosserat models. *Computer Graphic Forum (Proceedings of Eurographics 2002)*, 21(3):347–352.

[Rubin, 2000] Rubin, M. B. (2000). *Cosserat Theories: Shells, Rods and Points*. Kluwer Academic Publ., Dordrecht,.

[Sauer and Schömer, 1998] Sauer, J. and Schömer, E. (1998). A constraint-based approach to rigid body dynamics for virtual reality applications. In *Proc. ACM Symposium on Virtual Reality Software and Technology*, pages 153–161.

[Schotte, 2005] Schotte, W. (2005). Simulation des dynamischen verhaltens von kabeln fr einbau-montage-simulation in vr. Master's thesis, Technische Universit Darmstadt.

[Specht, 2006] Specht, R. (2006). Spezification eines erweiterungsmodul zur simulation von leitungsszen in virtual reality anwendungen.

[Thomaszewski and Wacker, 2006] Thomaszewski, B. and Wacker, M. (2006). Bending Models for Thin Flexible Objects.

[van der Heijden et al., 2003] van der Heijden, G., Neukirch, S., Goss, V., and Thompson, J. (2003). Instability and self contact phenomena in the writhing of clamped rods. *Int. J. Mech. Sci.*, 45(1):161–196.

[Ward et al., 2007] Ward, K., Bertails, F., Kim, T.-Y., Marschner, S. R., Cani, M.-P., and Lin, M. (2007). A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 13(2):213–234.

[Ward et al., 2006] Ward, K., Galoppo, N., and Lin, M. C. (2006). A simulation-based vr system for interactive hairstyling. In *VR '06: Proceedings of the IEEE Virtual Reality Conference (VR 2006)*, page 37, Washington, DC, USA. IEEE Computer Society.

[Ward and M. C. Lin.. In, 2003] Ward, K. and M. C. Lin.. In, p. . (2003). Adaptive grouping and subdivision for simulating hair dynamics. In *Pacific Graphics Conference on Computer Graphics and Applications*, page 234243.

[Ward K. and Macri, 2003] Ward K., M. C. Lin, J. L. S. F. and Macri, D. (2003). Modeling hair using level-of-detail representations. In *International Conference on Computer Animation and Social Agents*, pages 41–47.

[Ward K. and Lin, 2004] Ward K., N. G. and Lin, M. C. (2004). Modeling hair influenced by water and styling products. pages 207–214.

[Yazgan, 2005] Yazgan, G. (2005). Effizienz von dmu. Master's thesis.

Lebenslauf

Name	**Mireille GRÉGOIRE**
Adresse	Reinsburgstr. 128 70197 Stuttgart
Telephon	0711 674 05 71 0151 59 11 71 38 (Handy)
E-mail	mireille.gregoire@gmx.de
Geburtstag, Geburtsort	19. September 1978 in Anderlecht (Brüssel), Belgien
Staatsangehörigkeit	französisch
Familienstand	verheiratet

Schulbildung und Studium

Dezember 2001 - Mai 2002	Diplomarbeit bei Robert Bosch GmbH Nachbildung der menschlichen Hand-Arm-Dynamik zur Verwendung in Prüfständen für Elektrowerkzeuge.
Oktober 2000 - Januar 2003	Studium der ***Technischen Kybernetik*** an der ***Universität Stuttgart*** im Rahmen des Doppeldiplomprogramms TIME.
September 1998 - Juni 2000	Studium der ***Allgemeinen Ingenieurwissenschaften*** an der Technischen Universität ***Ecole Centrale Paris***, Frankreich.
September 1996 - Juni 1998	Vorbereitungsstudium (*classes préparatoires*) für die Aufnahmeprüfungen der Technischen Universitäten Frankreichs
Juni 1996	Mathematisch- naturwissenschaftliches Abitur (Frankreich) (Note: sehr gut)

Berufliche Erfahrung

März 2004 - März 2007	***Doktorarbeit bei DaimlerChrysler: Interaktive Simulation flexibler Bauteile*** Entwicklung eines echtzeitfähigen und interaktiven Simulationsverfahren für flexible Bauteile (Schwerpunkt: Kabel und Schläuche). Integration in einer *Virtual Reality* Anwendung. Die mündliche Prüfung ist am 7. Dezember 2007.

Oktober 2000 - September 2001	Werkstudentin bei der Robert Bosch GmbH (10 Stunden pro Woche): Programmierung eines Bildverarbeitungssystems für Hochgeschwindigkeitskamerabilder, Durchführung und Auswertung von Messungen
Juli - September 2000	Praktikum bei der Robert Bosch GmbH in der Abteilung Angewandte Physik, Akustik : Programmierung eines Auswertungssystems für Frequenzmessungen eines Hand-Arm-Systems und einer Ansteuerung und Regelung eines Shakers.
Juli - August 1999	Neunwöchiges Praktikum bei der SIEMENS AG in Erlangen : Programmierung und Tests von elektronischen Karten.

Sprachkenntnisse

Französisch	Muttersprache
Deutsch	Fließend
Englisch	Fließend (*Cambridge Certificate of Proficiency in English*)
Spanisch	Fließend
Japanisch	Grundkenntnisse (Stufe 4 des *Japanese Language Proficiency Test*)

Veröffentlichungen

M. Grégoire und E. Schömer: *Interactive simulation of one-dimensional flexible parts* in ACM Symposium on Solid and Physical Modeling, SPM'06, S. 95-103

Eine erweiterte Version dieses Papers wurde für die Sonderausgabe zu SPM'06 der Computer-Aided Design Zeitschrift selektiert. (Volume 39, Issue 8, August 2007, Pages 694-707)

EDV-Kenntnisse

Programmiersprachen: C++, Matlab, Simulink, Labview, C, Caml, Java, Fortran, Skriptsprachen, qt
FEM - Simulation: I-deas, Abaqus
Betriebssysteme: Windows, Unix, Linux
Word, Excel, PowerPoint, Access, Outlook, OpenOffice, LaTeX, Lotus Notes, CVS

Weitere Aktivitäten und Interessen

Mitorganisation und Kassenführung einer Studentenstaffel an der E.C.P. in 1999 und 2000
Klavier spielen, japanische Filme, fantastische Literatur, Karate, Tennis.

Stuttgart, den 12. Dezember 2007