

# Controllable, Non-Oscillatory Damping for Deformable Objects

by

Herbert David Young

B.Sc., The University of British Columbia, 2005

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in

The Faculty of Graduate Studies

(Computer Science)

The University of British Columbia

December 2007

© Herbert David Young, 2007

# Abstract

This thesis presents a new method for the controllable damping of deformable objects. The method evolves from physically based techniques; however, it allows for non-physical, but visually plausible motion. This flexibility leads to a simple interface, with intuitive control over the behaviour of the material.

This method is particularly suited for strongly damped materials, which account for the majority of objects of interest to animation, since it produces non-oscillatory behaviour. This is similar to critical damping, except that it affects all modes independently. The new method is based on the minimization of a slightly modified version of total energy. This framework can be used to simulate many other physical phenomena, and therefore lends itself to coupling with other simulations.

Implementation details for a simple example are given. Results are shown for varying parameters and compared to those produced by a traditional method.

# Table of Contents

<b>Abstract</b> . . . . .	ii
<b>Table of Contents</b> . . . . .	iii
<b>List of Tables</b> . . . . .	vi
<b>List of Figures</b> . . . . .	vii
<b>List of Programs</b> . . . . .	viii
<b>Acknowledgements</b> . . . . .	ix
<b>Dedication</b> . . . . .	x
<b>1 Introduction</b> . . . . .	1
1.1 Related Work . . . . .	2
1.1.1 Elasticity . . . . .	2
1.1.2 Energy Minimization . . . . .	5
1.1.3 Perception . . . . .	8
1.2 Thesis Overview . . . . .	9
<b>2 Method</b> . . . . .	10

*Table of Contents*

---

2.1	Total Energy Minimization . . . . .	10
2.1.1	Modified Total Energy . . . . .	13
2.1.2	Explanation of $\tau$ . . . . .	14
2.2	Analysis . . . . .	17
2.2.1	Critical Damping of All Modes . . . . .	17
2.2.2	Equivalent Partial Differential Equation . . . . .	18
2.2.3	New Properties . . . . .	21
<b>3</b>	<b>Implementation . . . . .</b>	<b>23</b>
3.1	Input and Pre-computation . . . . .	23
3.2	Each Time Step . . . . .	24
3.2.1	Minimization . . . . .	25
3.2.2	Calculation of Total Energy . . . . .	26
3.2.3	Update . . . . .	29
3.3	Pseudocode . . . . .	29
<b>4</b>	<b>Results . . . . .</b>	<b>31</b>
4.1	New Method . . . . .	31
4.1.1	Distance from the Rest State . . . . .	32
4.1.2	Examples . . . . .	32
4.2	Traditional Method . . . . .	35
4.2.1	Choosing Damping Parameters . . . . .	36
4.2.2	Material Limits to Critical Damping . . . . .	40
4.3	Comparison of Methods . . . . .	41

*Table of Contents*

---

<b>5</b>	<b>Conclusions</b>	45
5.1	Future Work	45
5.1.1	Oscillation Control	46
5.1.2	Plasticity	47
5.1.3	Extending the Energy Minimization Framework	47
	<b>Bibliography</b>	49

# List of Tables

4.1	Parameters for Examples. . . . .	35
	(a) Example 1 . . . . .	35
	(b) Example 2 . . . . .	35
4.2	Damping Parameters for Traditional Method. . . . .	42
	(a) Example 1 . . . . .	42
	(b) Example 2 . . . . .	42

# List of Figures

2.1	Examples of under-, over-, and critical damping. . . . .	16
2.2	The separate modes of two over-damped systems. . . . .	16
	(a) $c = 3$ . . . . .	16
	(b) $c = 6$ . . . . .	16
4.1	Example 1. . . . .	33
	(a) Rest configuration . . . . .	33
	(b) Initial displacement with fixed nodes marked with red dots . . . . .	33
4.2	Example 2. . . . .	34
	(a) Rest configuration . . . . .	34
	(b) Initial displacement with fixed nodes marked with red dots . . . . .	34
4.3	Method Comparison for Example 1. . . . .	43
	(a) New Method. . . . .	43
	(b) Traditional Method. . . . .	43
4.4	Method Comparison for Example 2. . . . .	44
	(a) New Method. . . . .	44
	(b) Traditional Method. . . . .	44

# List of Programs

3.1 Pseudocode summary of algorithm . . . . .	30
---	----



# Acknowledgements

First and foremost, I thank my supervisor, Robert Bridson. His help, encouragement, and inspiration have made this thesis possible.

I am also thankful to Michiel van de Panne for his helpful comments, and fellow Imager students Christopher, Mike, and Stelian for their suggestions and motivation. As well, financial support from the Natural Sciences and Engineering Research Council of Canada allowed me to pursue this work.

Finally, I would like to thank my family for their constant encouragement and support. For that, I am eternally grateful.

To my wife Alison, for everything.

# Chapter 1

## Introduction

Modelling elasticity is required for many simulations or animations. This is because elastic materials are both common and interesting. Most of the everyday objects we interact with have elastic properties, though some are more obvious than others. Many of the materials of interest to animation are elastic and strongly damped, such as hair, flesh, and clothing.

With the growing popularity of computer-animated feature films and special effects, computer graphics are being pushed ever further. Initially, computer-generated animations were quite rigid, as deformable objects required either a significant effort by the animators or complex computer simulations. Since then, simulation techniques have become more developed and efficient, and with the ever advancing power of hardware, the capabilities of computer-generated content is always increasing. In order to be more realistic, or even more artistic, computer-generated content needs to model elastic objects.

Simulations for film have very different requirements than those for engineering or research. When the target is entertainment, emphasis is placed on visual quality, speed of simulation, and ease of use, while scientific purposes demand proven accuracy.

In this research, a method is presented which allows for intuitive control over the damping of an elastic material. The new method is inspired by physics, but motivated to offer a simple interface to the artist. The results, we believe, are physically plausible, even when the user requests physically impossible motion.

## 1.1 Related Work

Several different topics in computer graphics provide important ground work for this new approach. Some cover previous methods which accomplish similar tasks, others provide motivation, and a few explain the choices made while developing this method.

### 1.1.1 Elasticity

Deformable objects in general have been studied in computer graphics for over three decades, and physically based methods to simulate such objects have been in the field for at least two decades. For an overview of these methods and their history, the complementary survey papers of Gibson and Mirtich [7] and Nealen et al. [13] are excellent resources.

### Mass-Spring Systems

One of the earliest of the physically based models was the mass-spring system. An array of point-masses — often regularly spaced in one, two, or three dimensions — are connected by simple, often linear, springs. Note that multiple connections of springs are often needed to keep the object from

shearing, and that these springs usually include dampers as well. With a sufficient number of point-masses, the system can simulate complex shapes and deformations. These systems have been used in the early work of facial animation [21, 26], and have been used extensively for cloth [2, 3, 5]. The mass-spring system can also be used in other interesting ways: by dynamically adjusting the rest length of different springs, the model reacts to the changes as if being controlled by flexing muscles [12, 25].

Mass-spring models are intuitive, simple to implement, and fast to simulate; however, the exact simulation of an elastic material is impossible [6]. The results depend on the number and distribution of point-masses, and special care must be taken when choosing individual spring parameters in order to even approximate a desired behaviour [6, 13].

### Finite Element Methods

A popular method in several fields, including computer graphics, for modelling elasticity is the finite element method (FEM). In general, it is used to solve partial differential equations. In terms of elastic simulation, the differential equation comes from continuum mechanics,

$$\rho \ddot{x} = \nabla \cdot \sigma + f \tag{1.1}$$

where  $\rho$  is density,  $\ddot{x}$  is the second time-derivative of position  $x$ ,  $\sigma$  is stress, and  $f$  is external force [13]. FEM works by approximating the differential equation with a series of elements, often triangles or quadrilaterals in two dimensions. The approximation of each element is determined by the values

associated with the nearby nodes and the basis functions.

The finite element method has been used to simulate elasticity in combination with more complicated phenomena, such as brittle or ductile fracture [16, 17]. Even Green strain, with non-linear elastic forces, can be computed relatively easily with this method, which allows for greater deformation than linear elasticity.

The accuracy depends on the choice of bases as well as the number of elements. The finite element method, which considers the object as a continuum, is more accurate than the mass-spring approach, which uses a discrete object model [7]. Unlike the mass-spring model, as the resolution is increased, and therefore the size of the elements is decreased, the model converges on a solution.

### **Damping Parameters**

Both of the above methods still require damping parameters which do not easily relate to the time the system takes to come to rest. For the mass-spring method, choosing physically correct damping parameters is very difficult, if not impossible, since the discretization of the object affects the results of the damping parameters. Simply choosing parameters which critically damp each spring does not result in the object being critically damped. Even for other methods which can approximately use physical damping parameters, the correct values can be hard to find for certain materials, and the process of determining the correct values to obtain a certain result is mostly trial-and-error [7].

Modal dynamics have a damping coefficient for each independent mode;

however, this is only efficient with a small number of modes of deformation, and is not extendable to plasticity [20].

### **Quasi-Static Methods**

Quasi-static motion ignores inertia and assumes that the material instantaneously moves into a state of equilibrium after any change to the external forces. The method solves for the internal forces which will exactly match the external forces and keep the material in equilibrium. This method works well for materials where such an assumption about the inertia is approximately true, such as flesh in most circumstances [8, 24]. It allows simulations to run quickly, since the differential equation used by these methods no longer involves time, and a large amount of information can be pre-computed. Moreover, estimating and tuning damping parameters is unnecessary, as damping and oscillation behaviour has been removed from the dynamics.

However, the quasi-static assumption means that these methods are not suitable for ballistic motion, transient waves, creep, or other such phenomena, where inertia does produce visible results. This is because there is no temporal coherence; each step of the algorithm produces a result which is in equilibrium, ignoring the dynamics and the time the system would actually take to reach this equilibrium point. The biggest limitation is that this assumption is not valid for many of the elastic objects commonly animated.

#### **1.1.2 Energy Minimization**

Minimizing different energy terms can be a powerful method for physical simulation, and is often closely linked to how physicists understand these

systems. Optimization can be used to find only certain forces, or it can solve for the entire system's new configuration.

Minimizing a sum of energies, ranging from gravitational and elastic potential to an artificial repulsion energy, can be used to enforce inter-particle constraints in cloth simulation [2]. It can also be used when creating patterns for apparel, where stretching and shearing need to be minimized [18].

In Lagrangian dynamics, one can minimize the integral of kinetic energy minus potential energy; however, this applies only to constrained systems, whereas we are interested in unconstrained dynamics.

Quasi-static methods are essentially a minimization of the potential energy, since the minimum of potential must be an equilibrium state. Rigid body contact can be phrased as a kinetic energy minimization with constraints [11]. The pressure projection step used in fluid simulation can also be completed by minimizing kinetic energy, and because of the common framework, it has been used to couple fluids and solids [1].

Optimization of an energy-like term is also possible. For example, the minimization of an energy-like function has led to a new time integration scheme for dynamic systems [10]. Optimal control is another example, where a measure of control energy and the error from a target configuration is optimized to derive a control policy [4].

## **Damping**

Another physical phenomenon which can be treated with the minimization of energy framework is damping. By adding a regularization term to kinetic energy, the optimization method is equivalent to using a Backward Euler



integration scheme.

To see this, start with a typical FEM discretization for force,

$$\mathbf{F} = \mathbf{D}\mathbf{S} \quad (1.2)$$

where  $\mathbf{F}$  is a vector with component  $F_i$  being the net force on modal variable  $i$ ,  $\mathbf{S}$  is the stress tensor at each quadrature point, and  $\mathbf{D}$  is a discrete approximation to the divergence operator, mapping stress at the quadrature points to force on the nodes. If we consider just the damping stress, then

$$\mathbf{S} = -\mathbf{A}\mathbf{D}^T\mathbf{v} \quad (1.3)$$

where  $\mathbf{A}$  is a symmetric positive definite matrix which encodes the damping parameters,  $\mathbf{v}$  is the velocity, and  $\mathbf{D}^T\mathbf{v}$  is an approximation of strain rate.

A Backward Euler solve for damping is

$$\mathbf{M}\mathbf{v}_{new} = \mathbf{M}\tilde{\mathbf{v}} + \Delta t\mathbf{D}\mathbf{S} \quad (1.4)$$

$$\mathbf{S} = -\mathbf{A}\mathbf{D}^T\mathbf{v}_{new} \quad (1.5)$$

with  $\mathbf{M}$  being the mass matrix and  $\tilde{\mathbf{v}}$  is the intermediate velocity with non-damping forces already integrated.

The usual method to solve this is to substitute (1.5) into (1.4),

$$\mathbf{v}_{new} = (\mathbf{M} + \Delta t\mathbf{D}\mathbf{A}\mathbf{D}^T)^{-1}\mathbf{M}\tilde{\mathbf{v}} \quad (1.6)$$

but if we minimize kinetic energy with a regularization term,  $\frac{\Delta t}{2} \mathbf{S}^T \mathbf{A}^{-1} \mathbf{S}$ :

$$\min_{\mathbf{S}} \quad \frac{1}{2} \mathbf{v}_{new}^T \mathbf{M} \mathbf{v}_{new} + \frac{\Delta t}{2} \mathbf{S}^T \mathbf{A}^{-1} \mathbf{S} \quad (1.7)$$

$$\Leftrightarrow \mathbf{0} = \left( \frac{d\mathbf{v}_{new}}{d\mathbf{S}} \right)^T \mathbf{M} \mathbf{v}_{new} + \Delta t \mathbf{A}^{-1} \mathbf{S} \quad (1.8)$$

$$\mathbf{0} = \Delta t \mathbf{D}^T \mathbf{M}^{-1} \mathbf{M} \mathbf{v}_{new} + \Delta t \mathbf{A}^{-1} \mathbf{S} \quad (1.9)$$

$$\mathbf{S} = -\mathbf{A} \mathbf{D}^T \mathbf{v}_{new} \quad (1.10)$$

then we get the same result as a Backward Euler step.

### 1.1.3 Perception

A very important area of discussion in computer graphics is determining how much accuracy is truly necessary. Physically correct motion is important in order to create believable animations. However, if the user is unable to see the difference between a perfectly accurate simulation and a good approximation, then the latter may be more desirable. A method which produces an acceptable approximation may have several benefits, such as being easier to use, faster to implement, or more efficient to run.

Studies have shown people's ability to understand physical systems is limited in various ways. Despite the deterministic behaviour of most physical systems, if more than one parameter or dimension is significant, people lose their intuition [9, 22]. Perceptual studies have been done on determining exactly what motion appears plausible to the viewer, but a robust and accurate metric is still unavailable [19]. A recent study suggests that predicting the behaviour of a physical system and determining plausibility

are independent cognitive processes, and that approximations can be made without producing visually objectionable results [15].

## **1.2 Thesis Overview**

First, the new method is introduced in Chapter 2, and the major components are explained. In Chapter 3, an implementation of the method is described, and details specific to implementing are provided. The results of the implementation are then discussed and demonstrated in Chapter 4. Finally, Chapter 5 concludes with what has been accomplished, and what further work this could lead to.

# Chapter 2

## Method

Optimization has been used to simulate many different physical systems, as discussed in Section 1.1.2. In some circumstances — including damping with a regularization term, rigid-body contact, and incompressible fluid flow — dynamics can be modelled by minimizing kinetic energy, and quasi-static systems by minimizing potential energy. In order to capture both the dynamic behaviour of the former model and the control-like treatment of damping in the latter, we start by combining the concepts in a minimization of total energy. The resulting method is dependant on the time discretization. To overcome this problem, we use a modified version of total energy.

### 2.1 Total Energy Minimization

If we simply minimize the total energy, the sum of kinetic and potential energies, with respect to stress, or force, then an undesirable property results: the method's apparent damping rate is reduced as the step size becomes smaller.

To demonstrate this, we consider an elastic bar in one-spatial dimension. After taking a Backward Euler step, we use Fourier analysis and minimize

the total energy with respect to force. First, define transforms for displacement  $u_n$ , velocity  $v_n$ , and force  $f$ , at time step  $n$ ,

$$u_n = A_n e^{\sqrt{-1}kx} \quad (2.1)$$

$$v_n = B_n e^{\sqrt{-1}kx} \quad (2.2)$$

$$f = F e^{\sqrt{-1}kx} \quad (2.3)$$

where  $x$  is the material space, or rest, position and  $k$  is the wave number.

The time discretization is then

$$B_{n+1} = B_n + \frac{\Delta t}{\rho} F \quad (2.4)$$

$$A_{n+1} = A_n + \Delta t B_{n+1} \quad (2.5)$$

$$= A_n + \Delta t B_n + \frac{\Delta t^2}{\rho} F \quad (2.6)$$

Total energy is the sum of kinetic and potential energy over the domain of the object,  $\Omega$ , which in one dimension is the object length. The energy terms will be defined as

$$KE_n = \int_{\Omega} \frac{1}{2} \rho v_n^2 \quad (2.7)$$

$$PE_n = \int_{\Omega} \frac{1}{2} E \left( \frac{\partial u_n}{\partial x} \right)^2 \quad (2.8)$$

$$TE_n = KE_n + PE_n \quad (2.9)$$

where  $\rho$  is the material density,  $E$  is Young's modulus of elasticity, and  $\frac{\partial u}{\partial x}$  is the strain.

The approximate effect on the displacements follows from minimizing

the total energy at the next time step,

$$\min \int_{\Omega} KE_{n+1} + PE_{n+1} \quad (2.10)$$

$$\Leftrightarrow \min \int_{\Omega} \frac{1}{2} \rho v_{n+1}^2 + \frac{1}{2} E \left( \frac{\partial u_{n+1}}{\partial x} \right)^2 \quad (2.11)$$

$$\begin{aligned} \Leftrightarrow \min_F \int_{\Omega} \frac{\rho}{2} \left( B_n + \frac{\Delta t}{\rho} F \right)^2 \left| e^{\sqrt{-1}kx} \right|^2 \\ + \int_{\Omega} \frac{E}{2} \left( A_n + \Delta t B_n + \frac{\Delta t^2}{\rho} F \right)^2 \left| \sqrt{-1}k e^{\sqrt{-1}kx} \right|^2 \end{aligned} \quad (2.12)$$

$$\Leftrightarrow \min_F \frac{\Omega \rho}{2} \left( B_n + \frac{\Delta t}{\rho} F \right)^2 + |k|^2 \frac{\Omega E}{2} \left( A_n + \Delta t B_n + \frac{\Delta t^2}{\rho} F \right)^2 \quad (2.13)$$

Finding the minimum is equivalent to taking the derivative of this objective function with respect to  $F$ , and finding the zero,

$$\Omega \Delta t \left( B_n + \frac{\Delta t}{\rho} F \right) + |k|^2 \Omega E \frac{\Delta t^2}{\rho} \left( A_n + \Delta t B_n + \frac{\Delta t^2}{\rho} F \right) = 0 \quad (2.14)$$

and therefore  $F$  is

$$\frac{\Delta t}{\rho} \left( 1 + |k|^2 \frac{E \Delta t^2}{\rho} \right) F = -|k|^2 \frac{E \Delta t}{\rho} A_n - \left( 1 + |k|^2 \frac{E \Delta t^2}{\rho} \right) B_n \quad (2.15)$$

$$F = -\frac{|k|^2 E}{1 + |k|^2 \frac{E \Delta t^2}{\rho}} A_n - \frac{\rho}{\Delta t} B_n \quad (2.16)$$

Applying this to (2.4) and (2.6) gives

$$B_{n+1} = -\frac{|k|^2 E \Delta t}{\rho + |k|^2 E \Delta t^2} A_n \quad (2.17)$$

$$A_{n+1} = \frac{1}{1 + |k|^2 \frac{E \Delta t^2}{\rho}} A_n \quad (2.18)$$

and therefore, with the total time  $T = n\Delta t$ ,

$$A_n = \left( \frac{1}{1 + |k|^2 \frac{E\Delta t^2}{\rho}} \right)^n A_0 \quad (2.19)$$

$$A_n \approx e^{-n|k|^2 \frac{E\Delta t^2}{\rho}} A_0 \quad (2.20)$$

$$A(T) \approx e^{-|k|^2 \frac{E}{\rho} T\Delta t} A_0 \quad (2.21)$$

Since the behaviour depends on the step size, the method becomes less efficient as  $\Delta t \rightarrow 0$ . This is an undesirable property for several reasons. Smaller values of  $\Delta t$  mean smaller steps in time, and therefore more steps are needed than with a larger  $\Delta t$  to simulate the same total time. If the method becomes less efficient, then the time to simulate will grow faster than the number of steps needed, making very small values of  $\Delta t$  impracticable.

### 2.1.1 Modified Total Energy

In order to avoid this problem, we minimize a slightly different value. We multiply the potential energy by a dimensionless fraction, which means we maintain the units of energy. This new coefficient has the units of time over time which results in not only eliminating the above problem, but also giving the user control over the length of time the simulation takes to come to rest. Our new minimization has the form

$$\min \int KE + \frac{\tau}{\Delta t} PE \quad (2.22)$$

Now by adjusting  $\tau$ , we have control over the time the system takes to come to rest, which we will call  $t_{rest}$ . The effect of  $\tau$  is inversely related to

$t_{rest}$ , which will become more clear with a similar derivation to (2.21), but with the new coefficient:

$$\min \int_{\Omega} \frac{1}{2} \rho (v_{n+1})^2 + \frac{\tau}{2\Delta t} E \left( \frac{\partial u_{n+1}}{\partial x} \right)^2 \quad (2.23)$$

$$F = -\frac{|k|^2 E \frac{\tau}{\Delta t}}{1 + \tau |k|^2 \frac{E \Delta t}{\rho}} A_n - \frac{\rho}{\Delta t} B_n \quad (2.24)$$

$$A_n = \left( \frac{1}{1 + \tau |k|^2 \frac{E \Delta t}{\rho}} \right)^n A_0 \quad (2.25)$$

$$A_n \approx e^{-n\tau |k|^2 \frac{E \Delta t}{\rho}} A_0 \quad (2.26)$$

$$A(T) \approx e^{-\tau |k|^2 \frac{E}{\rho} T} A_0 \quad (2.27)$$

Therefore, if the system comes to rest by time  $T = t_1^{rest}$  using  $\tau = \tau_1$ , then by using  $\tau = \alpha \tau_1$ , the same rest state is achieved by time  $\frac{1}{\alpha} t_1^{rest}$ . This gives the user a much simpler interface to control the behaviour of elastic material than manually adjusting damping parameters.

It is important to note that the minimization of the modified total energy does not have the same inefficiency problem as the unmodified version. In (2.27), unlike (2.21), there is no  $\Delta t$ , and therefore the size of the time step does not affect the rate at which the system comes to rest.

### 2.1.2 Explanation of $\tau$

The behaviour of  $\tau$  seems unintuitive at first, especially considering it has the units of time and not the inverse. Initially, the coefficient of potential energy,  $\frac{\tau}{\Delta t}$ , looks like the number of steps of size  $\Delta t$  to reach time  $\tau$ . However, as we increase this fraction, the time to reach rest, or any particular state,



decreases. So using the same  $\Delta t$  and increasing  $\tau$ , decreases the time to come to rest. To understand  $\tau$  better, we will take a look at the damping of a simple mass-spring system.

Taking the basic equation of motion for displacement  $x(t)$ , a function of time, with mass  $m$ , damping coefficient  $c$ , and stiffness  $k$ , we get

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{2.28}$$

where  $\dot{x} = \frac{dx}{dt}$  and  $\ddot{x} = \frac{d^2x}{dt^2}$ . Assuming that the solution to this ordinary differential equation is of the form  $x(t) = e^{rt}$ , we can find  $r$  by solving

$$mr^2 + cr + k = 0 \tag{2.29}$$

using the quadratic formula,

$$r = \frac{-c \pm \sqrt{c^2 - 4mk}}{2m} \tag{2.30}$$

In this form, (2.30), we can easily identify how the system is damped by the value in the square root,  $\alpha = c^2 - 4mk$ . If  $\alpha$  is zero, so  $c^2 = 4mk$ , then the system is critically damped, and there exists one, real solution:  $r_1 = r_2 \in \mathbb{R}$ . In the over-damped case,  $\alpha$  is greater than zero and there are two unique, real solutions:  $r_1 \neq r_2 \in \mathbb{R}$ ; in the under-damped case,  $\alpha$  is less than zero and there are two unique, complex solutions:  $r_1 \neq r_2 \in \mathbb{C}$ .

By graphing an example for each case, Figure 2.1, the effects of different values of  $\alpha$  are very clear. To solve (2.28), we use  $x(t) = Ae^{r_1t} + Be^{r_2t}$ . These graphs use  $m = 1, k = 1, A = 0.5, B = 0.5$  with  $c = 1$  for under-damping,

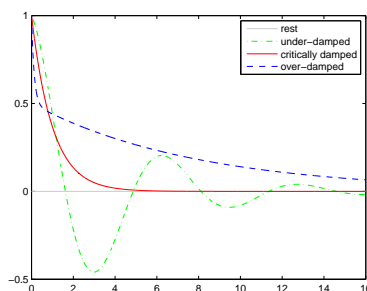
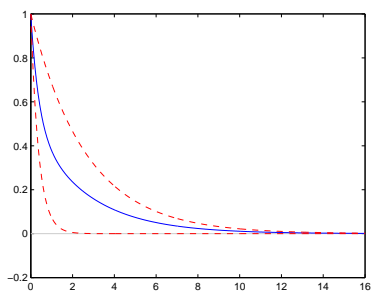
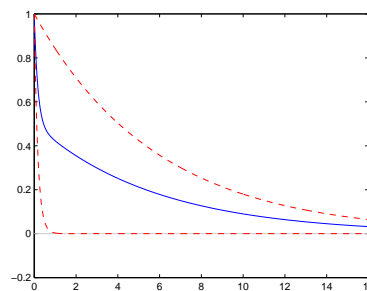


Figure 2.1: Examples of under-, over-, and critical damping.

$c = 2$  for critical damping, and  $c = 4$  for over-damping. The values of  $r$  in each case are  $-0.5 \pm \frac{\sqrt{3}}{2}i$  for under-damped,  $-1$  for critically damped, and  $-2 \pm \sqrt{3}$  for over-damped.



(a)  $c = 3$



(b)  $c = 6$

Figure 2.2: The separate modes of two over-damped systems.

Since the differential equation (2.28) is second-order, it involves finding the solution of a quadratic equation. The two roots mean that there are two modes in the solution. In the critically damped case, the two modes are equal, but in the other cases, one mode comes to rest faster; let that

time be denoted as  $t_{fast}$  and the other slower time as  $t_{slow}$ . If the damping parameter is increased past critically damping, then  $t_{slow}$  increases and so does the time the whole system takes to come to rest; however, as  $t_{slow}$  increases,  $t_{fast}$  decreases. This can be observed in Figure 2.2 where the modes of an over-damped system are plotted separately.

As shown in Section 2.2.1, our new method of simulating elastic material critically damps all modes, and because of this, the duality of the above method does not occur. The minimization works on each mode independently, and since the minimum will be related to  $t_{fast}$ , the other mode,  $t_{slow}$  is not seen. However, if  $t_{slow}$  is theoretically increased, the system comes to rest sooner, and it is this reason, that  $\tau$  can be treated as the theoretical value of  $t_{slow}$ , with units of time and affecting the material inversely. The behaviour of  $\tau$  can also be further understood through the analysis given in Section 2.2.2.

## 2.2 Analysis

### 2.2.1 Critical Damping of All Modes

A major benefit of this method is the damping control. As mentioned above, the user can adjust the time the system takes to come to rest with a single, easily-understood parameter. Another effect of the minimization framework is that all modes are critically damped. Such behaviour is not attainable with traditional methods, regardless of the choice of damping parameters.

This unique result of the new method is due to the behaviour of mini-

mization. Parseval's theorem states that

$$\int_{\Omega} |A|^2 = c(\Omega) \sum_{n=-\infty}^{\infty} |a_n|^2 \quad (2.31)$$

where  $a_n$  are the coefficients in the Fourier series of  $A$  and  $c$  is some constant depending on  $\Omega$ . If we apply this to our modified total energy definition in (2.23), we get

$$\min_{\sigma} \quad \frac{\rho}{2} \int_{\Omega} (v_{n+1})^2 + \frac{\tau E}{2\Delta t} \int_{\Omega} \left( \frac{\partial u_{n+1}}{\partial x} \right)^2 \quad (2.32)$$

$$\min_{\hat{\sigma}} \quad \frac{\rho}{2} c(\Omega) \sum_{k=-\infty}^{\infty} |\hat{v}_k^{n+1}|^2 + \frac{\tau E}{2\Delta t} c(\Omega) \sum_{k=-\infty}^{\infty} |\hat{u}_k^{n+1} \sqrt{-1}k|^2 \quad (2.33)$$

where  $\hat{v}^{n+1}$ ,  $\hat{u}^{n+1}$ , and  $\hat{\sigma}$  are the Fourier series coefficients for  $v_{n+1}$ ,  $u_{n+1}$ , and  $\sigma$ . Since minimizing with respect to the vector  $\hat{\sigma}$  is equivalent to minimizing with respect to each component  $\hat{\sigma}_k$  independently, each mode of energy depends on only one mode of the stress, and therefore each mode is critically damped.

### 2.2.2 Equivalent Partial Differential Equation

To better understand the new method, we derive the partial differential equation that it is essentially solving. To simplify the process, we show only the one-dimensional problem with spatially uniform material properties — with displacement  $u$ , velocity  $v$ , stress  $\sigma$ , and strain  $\varepsilon = \frac{\partial u}{\partial x}$  — and we do not expand strain, even though it is not a primary variable. The problem is

discretized in time using

$$v^{n+1} = v^n + \frac{\Delta t}{\rho} \frac{\partial \sigma}{\partial x} \quad (2.34)$$

$$u^{n+1} = u^n + \Delta t v^{n+1} \quad (2.35)$$

and therefore

$$\varepsilon^{n+1} = \varepsilon^n + \Delta t \frac{\partial v^{n+1}}{\partial x} \quad (2.36)$$

We now use variational calculus and assume that we have a stress,  $\sigma$ , which minimizes the modified total energy, given in (2.22). If we perturb that solution by  $\delta s$ , for some arbitrary stress  $s$ , such that the stress of the new system is  $\sigma + \delta s$ , then the new energy can be written as a function of  $\delta$ :

$$g(\delta) = \int \frac{\rho}{2} \left| v^{n+1} + \frac{\Delta t}{\rho} \delta \frac{\partial s}{\partial x} \right|^2 + \frac{\tau}{\Delta t} \int \frac{E}{2} \left| \varepsilon^{n+1} + \frac{\Delta t^2}{\rho} \delta \frac{\partial^2 s}{\partial x^2} \right|^2 \quad (2.37)$$

Since we assume that  $\sigma$  minimizes the energy,  $\delta = 0$  must be the minimum for the new function,  $g(\delta)$ . At this minimum, the derivative of the function must be zero:

$$g'(\delta) = \int \frac{\rho}{2} \left( 2 \frac{\Delta t}{\rho} v^{n+1} \frac{\partial s}{\partial x} + 2\delta \left( \frac{\Delta t}{\rho} \frac{\partial s}{\partial x} \right)^2 \right) + \frac{\tau}{\Delta t} \int \frac{E}{2} \left( 2 \frac{\Delta t^2}{\rho} \varepsilon^{n+1} \frac{\partial^2 s}{\partial x^2} + 2\delta \left( \frac{\Delta t^2}{\rho} \frac{\partial^2 s}{\partial x^2} \right)^2 \right) \quad (2.38)$$

$$g'(0) = \int \Delta t v^{n+1} \frac{\partial s}{\partial x} + \tau \int E \frac{\Delta t}{\rho} \varepsilon^{n+1} \frac{\partial^2 s}{\partial x^2} = 0 \quad (2.39)$$

Considering the domain between  $a$  and  $b$ , this last equation can be sim-

plified by dividing by  $\Delta t$  and using integration by parts,

$$0 = \int_a^b v^{n+1} \frac{\partial s}{\partial x} + \frac{\tau E}{\rho} \int_a^b \varepsilon^{n+1} \frac{\partial^2 s}{\partial x^2} \quad (2.40)$$

$$\begin{aligned} &= - \int_a^b \frac{\partial v^{n+1}}{\partial x} s + [v^{n+1} s]_a^b + \\ &\quad \frac{\tau E}{\rho} \left( \int_a^b \frac{\partial^2 \varepsilon^{n+1}}{\partial x^2} s + \left[ \varepsilon^{n+1} \frac{\partial s}{\partial x} \right]_a^b - \left[ \frac{\partial \varepsilon^{n+1}}{\partial x} s \right]_a^b \right) \end{aligned} \quad (2.41)$$

Ignoring boundary conditions to further simplify the equation results in

$$\int \left( -\frac{\partial v^{n+1}}{\partial x} + \frac{\tau E}{\rho} \frac{\partial^2 \varepsilon^{n+1}}{\partial x^2} \right) s = 0 \quad (2.42)$$

Since  $s$  is an arbitrary stress, the only way the above equation holds true is when the first factor of the integrand is exactly zero,

$$-\frac{\partial v^{n+1}}{\partial x} + \frac{\tau E}{\rho} \frac{\partial^2 \varepsilon^{n+1}}{\partial x^2} = 0 \quad (2.43)$$

Using (2.36), this is equivalent to

$$\frac{\varepsilon^{n+1} - \varepsilon^n}{\Delta t} = \frac{\tau E}{\rho} \frac{\partial^2 \varepsilon^{n+1}}{\partial x^2} \quad (2.44)$$

which is Backward Euler applied to a strain diffusion equation,

$$\frac{\partial \varepsilon}{\partial t} = \frac{\tau E}{\rho} \frac{\partial^2 \varepsilon}{\partial x^2} \quad (2.45)$$

Using this new equation in Fourier space shows the same results as the previous section: each mode is independently and critically damped. With

$\widehat{\varepsilon}_i$  being the weight for mode  $i$  of strain,

$$\frac{d\widehat{\varepsilon}_i}{dt} = -\frac{\tau E}{\rho} i^2 \widehat{\varepsilon}_i \quad (2.46)$$

and therefore the decay of each mode is exponential,

$$\widehat{\varepsilon}_i(t) = e^{-\frac{\tau E}{\rho} i^2 t} \widehat{\varepsilon}_i(0) \quad (2.47)$$

Since strain is the spatial derivative of displacement, taking the antiderivative with respect to  $i$  leads to similar equations for displacement.

A similar analysis in two dimensions leads to two diffusion equations for the displacement,

$$\frac{1}{\tau} \frac{\partial u_1}{\partial t} = \left( \frac{\lambda + 2\mu}{\rho} \right) \frac{\partial^2 u_1}{\partial x^2} + \left( \frac{\lambda + 2\mu}{\rho} \right) \frac{\partial^2 u_2}{\partial x \partial y} + \left( \frac{2\mu}{\rho} \right) \frac{\partial^2 u_1}{\partial y^2} \quad (2.48)$$

$$\frac{1}{\tau} \frac{\partial u_2}{\partial t} = \left( \frac{2\mu}{\rho} \right) \frac{\partial^2 u_2}{\partial x^2} + \left( \frac{\lambda + 2\mu}{\rho} \right) \frac{\partial^2 u_1}{\partial x \partial y} + \left( \frac{\lambda + 2\mu}{\rho} \right) \frac{\partial^2 u_2}{\partial y^2} \quad (2.49)$$

both analogous to Equation (2.45). Note that  $\lambda$  and  $\mu$ , the Lamé parameters, are both functions of  $E$ , Young's modulus of elasticity, and  $\nu$ , Poisson's ratio. They

### 2.2.3 New Properties

Choosing the stress by optimization leads to a new constitutive law. The relationship between stress and strain is no longer defined physically, which gives the method many of its characteristics.

Regular elasticity leads to a wave equation with a damping term,  $D$ ,

$$\frac{\partial^2 \varepsilon}{\partial t^2} = \frac{E}{\rho} \frac{\partial^2 \varepsilon}{\partial x^2} + D \quad (2.50)$$

but the new method solves a diffusion equation, (2.45), which can be written as

$$\frac{1}{\tau} \frac{\partial \varepsilon}{\partial t} = \frac{E}{\rho} \frac{\partial^2 \varepsilon}{\partial x^2} \quad (2.51)$$

Therefore the strain energy diffuses over the whole object without propagating in waves, which is why the new method shows critical damping behaviour.

The new dynamics are non-oscillatory and the rate of strain diffusion can be controlled by the new parameter  $\tau$ . The higher the value of  $\tau$ , the faster strain diffuses, and therefore the faster the system comes to rest.

By taking the previous energy-derived techniques and finding a middle ground, this new method has several interesting properties. The kinetic energy term captures inertial effects, and since it still solves equation (1.1), momentum is conserved and ballistic motion is possible. The augmented potential energy term takes the instant damping of quasi-static motion and makes it a controllable behaviour. The system can be more directly controlled than previously possible, and the results are plausible even if the requested action is physically impossible.



# Chapter 3

## Implementation

This section provides the details of a two-dimensional, triangle-based implementation. This simple implementation assumes the simulation involves a single, homogenous, isotropic material. No collision detection is involved, including that for self-intersection.

### 3.1 Input and Pre-computation

The inputs are a triangle mesh, along with a list of fixed vertices and a list of initial vertex displacements, and the global material parameters of the object. These include the mass associated with each vertex, Lamé parameters  $\lambda$  and  $\mu$ , as well as the value of the new parameter  $\tau$ .

The deformable object, defined in its original rest state using material coordinates  $\mathbf{p}$ , has world space coordinates of  $\mathbf{x}$ . The strain on each element, a triangle in this case, is calculated with Green strain,

$$\varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \right) \quad (3.1)$$

where  $u_i$  is the displacement at vertex  $i$ . This strain tensor can also be

written as

$$\mathbf{G} = \frac{1}{2}(\mathbf{A}^T \mathbf{A} - \mathbf{I}) \quad (3.2)$$

where  $\mathbf{A} = \frac{\partial \mathbf{x}}{\partial \mathbf{p}}$  is the deformation gradient and  $\mathbf{I}$  is the identity matrix.

As shown by Teran et al. [23], if we define the columns of two,  $2 \times 2$  matrices,  $\mathbf{D}_m$  and  $\mathbf{D}_s$ , using the edge vectors of each triangle in both material and world space coordinates,

$$\mathbf{D}_m = (\mathbf{p}_1 - \mathbf{p}_0 \mid \mathbf{p}_2 - \mathbf{p}_0) \quad (3.3)$$

$$\mathbf{D}_s = (\mathbf{x}_1 - \mathbf{x}_0 \mid \mathbf{x}_2 - \mathbf{x}_0) \quad (3.4)$$

then

$$\mathbf{D}_s = \mathbf{A} \mathbf{D}_m \quad (3.5)$$

$$\mathbf{A} = \mathbf{D}_s \mathbf{D}_m^{-1} \quad (3.6)$$

The advantage with this definition is that the  $\mathbf{D}_m^{-1}$  matrix is constant and can be pre-computed, and since  $\mathbf{D}_s$  is simple to construct, the computation of the strain, using equation (3.2) with (3.6), is fast and easy.

## 3.2 Each Time Step

For each time step, we find the stress tensors which minimize the modified total energy. Then we calculate the force produced by that stress, and use these to update the position of each vertex.

### 3.2.1 Minimization

Virtually any optimization algorithm can be used for this method. For this example we used steepest descent, because of the ease of implementation and to keep the main focus of this new method clear.

For each iteration, steepest descent, or gradient descent, takes a step proportional to the negative gradient. For this method, the update at iteration  $i$  is

$$\sigma_{i+1} = \sigma_i - \alpha_i \nabla TE(\sigma_i) \quad (3.7)$$

and

$$TE(\sigma_{i+1}) \leq TE(\sigma_i) \quad (3.8)$$

where  $TE(\sigma)$  is the total energy at stress  $\sigma$ .

#### Step Length

The choice of  $\alpha_i$  is arbitrary, as long as the inequality (3.8) is satisfied for all  $i \geq 0$ . Several line search strategies exist, each with a balance between being fast to compute and returning better results; see Nocedal and Wright for some examples [14].

For this implementation, we start with a default value of  $\alpha$ , and if condition (3.8) fails, then the step length is halved. This is repeated until a value of  $\alpha$  which satisfies the non-increasing condition is found, or until  $\alpha$  is sufficiently small. This procedure does work; however, at some points, many iterations are needed because of unnecessarily slow progress. This is the result of the default step length being too small.

To reduce this problem, without turning to a more complex algorithm, the initial value of  $\alpha$  is doubled whenever an iteration does not need to reduce the initial step size. If the previous iteration does reduce  $\alpha_{i-1}$  from the initial guess, then the initial value is set to the final value of the previous iteration,  $\alpha_i^{initial} = \alpha_{i-1}$ . This adaptive method quickly moves over large areas with small gradients, but readily adapts to less smooth areas. It is similar to the technique used by Okabe et al. [18].

### 3.2.2 Calculation of Total Energy

The objective function used in the optimization is the modified definition of total energy,

$$TE(\sigma) = KE(\sigma) + \frac{\tau}{\Delta t} PE(\sigma) \quad (3.9)$$

and is calculated exactly as shown, with the kinetic and potential energy functions as defined below.

#### Kinetic Energy

The kinetic energy function first computes the force on each vertex resulting from the given stress. This force is used to update the velocity of each vertex, and then the kinetic energy of the whole system is given by

$$KE(\sigma) = \frac{1}{2} \mathbf{v}(\sigma)^T \mathbf{M} \mathbf{v}(\sigma) \quad (3.10)$$

where  $\mathbf{M}$  is the mass matrix, and velocity has been updated using

$$\mathbf{v}_{new}(\sigma) = \mathbf{v}_{old} + \Delta t \mathbf{M}^{-1} \mathbf{F}(\sigma) \quad (3.11)$$

The old velocity is from the last time step, and each component of force,  $F_i$ , is the sum of forces on vertex  $i$ , from each adjacent triangle,

$$F_i(\sigma) = \sum_{t \in \text{adj}(i)} \sigma_t \frac{(x_{t_k} - x_{t_j})^\perp}{2} \quad (3.12)$$

where  $\text{adj}(i)$  are all the triangles with vertex  $i$ ,  $\sigma_t$  is the stress on triangle  $t$ , and  $x_{t_k}$  and  $x_{t_j}$  are the two vertices of triangle  $t$  opposite vertex  $i$ .

### Potential Energy

The potential energy of the whole system is the sum of the potential energy of each triangle. The potential energy density on each triangle is constant, since the stress has been approximated as constant across each element.

$$PE(\sigma) = \sum_t PE_t(\sigma_t) \quad (3.13)$$

For a single triangle, the potential energy is

$$PE_t(\sigma_t) = \iint_{\Omega_t} \frac{1}{2} \sigma_t : G(\sigma_t) \quad (3.14)$$

where  $A : B = \sum_{i,j} A_{ij} B_{ij}$  is the double contraction operator. Combining our standard definition of stress,

$$\sigma_{ij} = \lambda \sum_{i=1}^2 \varepsilon_{ii} \delta_{ij} + 2\mu \varepsilon_{ij} \quad (3.15)$$

where  $\delta_{ij}$  is the Kronecker delta function defined as

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \quad (3.16)$$

with  $\varepsilon = G$  and (3.14), gives

$$PE_t(\sigma_t) = \frac{a_t}{2} \left( \lambda \left( \sum_{i=1}^2 G_{ii} \right)^2 + 2\mu \sum_{i=1}^2 \sum_{j=1}^2 G_{ij}^2 \right) \quad (3.17)$$

The last line, (3.17), replaces the integral with a multiplication by the area of triangle  $t$ ,  $a_t$ , since  $PE_t(\sigma_t)$  is constant over each element.

### Gradient of Total Energy

Now that we can compute the total energy at any given stress, we use finite differences to estimate the total energy gradient. Since stress is a symmetric  $2 \times 2$  matrix, only three variables need to be considered. As well, a change in the stress of one triangle only affects that triangle and its neighbours, making this a local operation and therefore feasible for a large simulation.

Centred differences are used, resulting in the following approximation to the total energy gradient,

$$\nabla TE(\sigma) = \frac{1}{2\delta} \begin{pmatrix} TE(\sigma+\Delta^{11})-TE(\sigma-\Delta^{11}) & TE(\sigma+\Delta^{12})-TE(\sigma-\Delta^{12}) \\ TE(\sigma+\Delta^{21})-TE(\sigma-\Delta^{21}) & TE(\sigma+\Delta^{22})-TE(\sigma-\Delta^{22}) \end{pmatrix} \quad (3.18)$$

where  $\delta$  is some small constant and  $\Delta^{kl}$  is a  $2 \times 2$  matrix defined as

$$\Delta_{ij}^{kl} = \begin{cases} \delta & \text{if } ij = kl \text{ or } ji = kl \\ 0 & \text{otherwise} \end{cases} \quad (3.19)$$

### 3.2.3 Update

Once the stress that minimizes the objective function is found, it is used to update all, non-fixed nodes in the mesh. The the new velocities are calculated with (3.11), and the new positions by

$$\mathbf{x}_{new}(\sigma) = \mathbf{x}_{old} + \Delta t \mathbf{v}_{new}(\sigma) \quad (3.20)$$

## 3.3 Pseudocode

The new method, as described above and as we implemented, is summarized in Program 3.1.

---

**Program 3.1** Pseudocode summary of algorithm

---

```
// initialize
read in mesh
compute the inverse rest edge matrix for each triangle
read in and apply initial displacement

// steepest descent
calculate gradTE with finite differences
stress_trial = stress - alpha * gradTE
calculate TE_new(stress_trial)
while TE_new > TE_old
    alpha = alpha / 2
    recalculate stress_trial and TE_new
stress = stress_trial

// update mesh
for each non-fixed vertex
    calculate force(stress)
    v_new = v_old + dt/m * force
    x_new = x_old + dt * v_new
```

---



# Chapter 4

## Results

### 4.1 New Method

The implementation described in Chapter 3 has been run with varying inputs in order to observe the new method. In practice, the new approach to damping control is easy to use, and produces visually-plausible elastic motion.

Ideally, the exact value of  $\tau$  for a given time to come to rest would be calculated; however, realistically,  $\tau$  is estimated,  $t_{rest}$  is observed, and then the desired value of  $\tau$  is computed exactly. Although this leads to two simulations being needed, it is still a considerable improvement from the many iterations of trial-and-error commonly taken with traditional damping parameters. Theoretically, it would be possible to calculate  $\tau$  beforehand, but because of the complex relationship between  $\tau$  and  $t_{rest}$ , it is faster to run through the simulation an extra time. This action is further supported by the ability of the method to take large time steps, and to quickly deliver meaningful practice runs.

### 4.1.1 Distance from the Rest State

In order to quantitatively measure when each simulation comes to rest, we have graphed two measures of the distance from the rest state. The first metric,  $d_1$ , is the  $p$ -2, or Euclidean, norm of a matrix containing all the differences of each vertex's position from the rest state,

$$d_1 = \|\mathbf{x} - \mathbf{p}\|_2 \quad (4.1)$$

and the second,  $d_2$  is the maximum distance any vertex is from its rest position,

$$d_2 = \max_{i \in \Omega} \|x_i - p_i\|_2 \quad (4.2)$$

The first measure,  $d_1$ , is global and considers how far the object as a whole is from rest, while  $d_2$  is local and measures the most extreme vertex.

### 4.1.2 Examples

#### Example 1

The first example is a simple mesh with twelve triangles arranged in a trapezoid, shown in its rest state in Figure 4.1a. The initial displacement, Figure 4.1b, is a single vertex at the top moved inward, and the vertices along the bottom are all fixed. The material parameters are given in Table 4.1a. Note that here,  $\rho$  is similar to the material density, but is just the mass at each vertex, and it is used to construct the mass matrix,  $\mathbf{M}$ .

In order to demonstrate the behaviour of the simulation, the distance from the rest state using the two metrics described in Section 4.1.1 are plot-

ted against time. Figure 4.3a shows this example with two different values of  $\tau$ . The graph demonstrates the effects of  $\tau$  quite well: the simulation using  $\tau = 1$  comes to rest at approximately  $T = 3s$ , while  $\tau = 4$  arrives at the same state by  $T = 0.75s$ . Note that the behaviours of both simulations are the same qualitatively, and that they show a critically-damped-like approach to the rest state. This is because adjusting the value of  $\tau$  is essentially a time-scaling operation. Also note that the two different metrics are virtually identical in this example because of the small number of vertices.

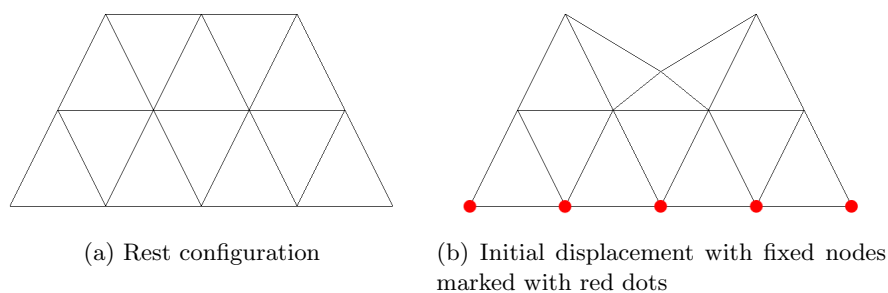


Figure 4.1: Example 1.

### Example 2

The second example is a mesh with 414 triangles laid out roughly in an annulus. The rest state of the mesh and the initial deformation is shown in Figure 4.2. In this example, the bottom vertices all fixed, and initially, the annulus is being pulled to the left and released.

The material and time parameters for this animation are given in Table 4.1b. The values of  $\Delta t$  are different for the two examples in order to simulate

the larger mesh in a similar time, not because of any restrictions on size of the time steps.

The behaviour over time, shown in Figure 4.4a, provides more interesting detail than the first example. The relationship between the rest times of  $\tau = 1$  and  $\tau = 0.33$  is as expected; however,  $\tau = 2$  does not quite come to rest in half the time of  $\tau = 1$ . This is likely due to numerical error, as the rest times are becoming relatively small. Also of interest is the oscillatory behaviour of  $d_2$ . The object as a whole still behaves as if critically damped, seen with  $d_1$ , but individual vertices may still overshoot the rest state. This gives the actual animation an interesting motion, while still effectively damping the object as desired.

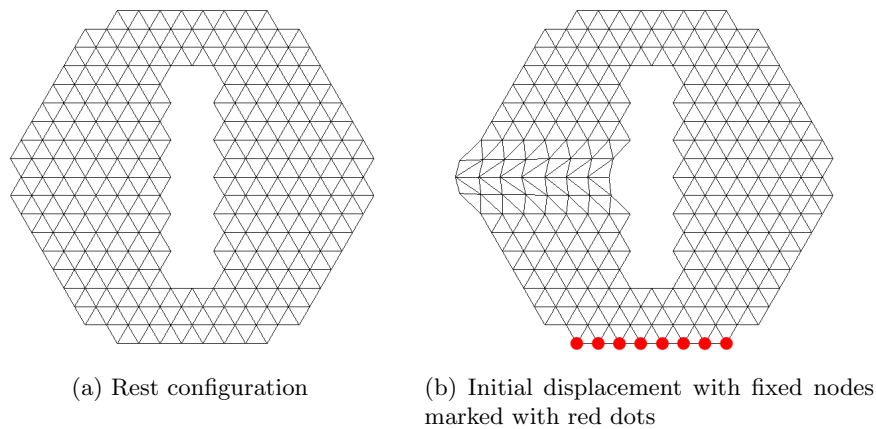


Figure 4.2: Example 2.

Parameter	Value
$\lambda$	3
$\mu$	1
$\rho$	1
$\Delta t$	0.01s
$T_{final}$	5s

(a) Example 1

Parameter	Value
$\lambda$	2
$\mu$	0.5
$\rho$	0.1
$\Delta t$	0.05s
$T_{final}$	5s

(b) Example 2

Table 4.1: Parameters for Examples.

## 4.2 Traditional Method

We have also implemented a two-dimensional, triangle-based, traditional technique for comparison. As much as possible, the same framework was used as described in Chapter 3.

We start by computing the strain tensor with the same definition, (3.2), and same technique as we used with the new method. Then elastic stress,  $\sigma^e$ , is calculated by (3.15) and is used to find the elastic force,  $\mathbf{F}_e$ , with (3.12). Intermediate velocities and positions are now given by

$$\tilde{\mathbf{v}} = \mathbf{v}_{old} + \Delta t \mathbf{M}^{-1} \mathbf{F}_e \quad (4.3)$$

$$\tilde{\mathbf{x}} = \mathbf{x}_{old} + \Delta t \tilde{\mathbf{v}} \quad (4.4)$$

These intermediate values are used to calculate the strain rate,

$$\dot{\mathbf{G}} = \frac{1}{2} (\dot{\mathbf{A}}^T \mathbf{A} + \mathbf{A}^T \dot{\mathbf{A}}) \quad (4.5)$$

and  $\dot{\mathbf{A}}$  was calculated by  $\dot{\mathbf{A}} = \dot{\mathbf{D}}_s \mathbf{D}_m^{-1}$ , with  $\mathbf{D}_m$  from (3.3) and

$$\dot{\mathbf{D}}_s = ( \tilde{\mathbf{v}}_1 - \tilde{\mathbf{v}}_0 \mid \tilde{\mathbf{v}}_2 - \tilde{\mathbf{v}}_0 ) \quad (4.6)$$

We can now find the damping stress,  $\sigma^d$ , similarly to (3.15),

$$\sigma_{ij}^d = \phi \sum_{i=1}^2 \dot{G}_{ii} \delta_{ij} + 2\psi \dot{G}_{ij} \quad (4.7)$$

and the damping force,  $\mathbf{F}_d$ , is given by (3.12) using  $\sigma^d$ .

The total force is the sum of the elastic and damping forces,  $\mathbf{F} = \mathbf{F}_e + \mathbf{F}_d$ , and the final velocities and positions are given by

$$\mathbf{v}_{new} = \mathbf{v}_{old} + \Delta t \mathbf{M}^{-1} \mathbf{F} \quad (4.8)$$

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \Delta t \mathbf{v}_{new} \quad (4.9)$$

### 4.2.1 Choosing Damping Parameters

In order to reasonably compare the new method to existing techniques, we must be careful when selecting the damping parameters for the traditional methods. Since these methods can only critically damp a single mode, we find parameters which critically damp the mode which is most visually obvious, corresponding to either the size of the elements or the size of the object.

The differential equation (1.1) without external force can be written as

$$\rho \ddot{u}_i = \frac{\partial \sigma_{ij}}{\partial x_j} \quad (4.10)$$

$$= \frac{\partial}{\partial x_j} (\lambda \varepsilon_{kk} \delta_{ij} + 2\mu \varepsilon_{ij} - \phi \dot{\varepsilon}_{kk} \delta_{ij} - 2\psi \dot{\varepsilon}_{ij}) \quad (4.11)$$

Define strain with

$$\varepsilon_{ij} = \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (4.12)$$

and the Fourier transform for displacement using

$$u = U(t) e^{\sqrt{-1} k x} \quad (4.13)$$

This is similar to (2.2), but continuous in time. Substituting (4.13) into (4.12) gives

$$\varepsilon_{ij} = U_i \sqrt{-1} k_j e^{\sqrt{-1} k x} + U_j \sqrt{-1} k_i e^{\sqrt{-1} k x} \quad (4.14)$$

and therefore

$$\frac{\partial \varepsilon_{ij}}{\partial x_j} = U_i (\sqrt{-1} k_j)^2 e^{\sqrt{-1} k x} + U_j (\sqrt{-1} k_i) (\sqrt{-1} k_j) e^{\sqrt{-1} k x} \quad (4.15)$$

$$= -(U_i k_j^2 + U_j k_i k_j) e^{\sqrt{-1} k x} \quad (4.16)$$

Expanding this in three dimensions gives

$$\begin{aligned}
 -(U_i k_j^2 + U_j k_i k_j) &= \begin{pmatrix} U_1 |k|^2 + U_1 k_1^2 + U_2 k_1 k_2 + U_3 k_1 k_3 \\ U_2 |k|^2 + U_1 k_1 k_2 + U_2 k_2^2 + U_3 k_2 k_3 \\ U_3 |k|^2 + U_1 k_1 k_3 + U_2 k_2 k_3 + U_3 k_3^2 \end{pmatrix} \quad (4.17) \\
 &= \begin{pmatrix} |k|^2 + k_1^2 & k_1 k_2 & k_1 k_3 \\ k_1 k_2 & |k|^2 + k_2^2 & k_2 k_3 \\ k_1 k_3 & k_2 k_3 & |k|^2 + k_3^2 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} \\
 &= (|k|^2 I + k \otimes k) U \quad (4.18)
 \end{aligned}$$

Now the equation (4.11) can be written

$$\rho \ddot{U} + (\mu |k|^2 I + (\mu + \lambda)(k \otimes k)) U + (\psi |k|^2 I + (\psi + \phi)(k \otimes k)) \dot{U} = 0 \quad (4.19)$$

The matrix  $|k|^2 I + k \otimes k$  has one eigenvector parallel to  $k$  and two more perpendicular to  $k$ . Let  $S$  be the component of  $U$  which is perpendicular to  $k$ , and  $T$  the component parallel to  $k$ . Now we can rewrite (4.19) with two independent equations,

$$\rho \ddot{S} + \mu |k|^2 S + \psi |k|^2 \dot{S} = 0 \quad (4.20)$$

$$\rho \ddot{T} + (2\mu + \lambda) |k|^2 T + (2\psi + \phi) |k|^2 \dot{T} = 0 \quad (4.21)$$

These two simple differential equations, (4.20 & 4.21), can now be easily solved. Assuming that the solution to (4.20) looks like  $S(t) = e^{rt}$ , we



substitute this in and solve for  $r$ ,

$$0 = \rho r^2 + \psi |k|^2 r + \mu |k|^2 \quad (4.22)$$

$$r = \frac{-\psi |k|^2 \pm \sqrt{\psi^2 |k|^4 - 4\rho\mu |k|^2}}{2\rho} \quad (4.23)$$

Since we want to find parameters which critically damp the material, we know that the discriminant of (4.23) must be zero,

$$0 = \psi^2 |k|^4 - 4\rho\mu |k|^2 \quad (4.24)$$

$$\psi = \sqrt{\frac{4\rho\mu}{|k|^2}} \quad (4.25)$$

By doing the same with (4.21), we get

$$\rho r^2 + (2\psi + \phi) |k|^2 r + (2\mu + \lambda) |k|^2 = 0 \quad (4.26)$$

and

$$0 = (2\psi + \phi)^2 |k|^4 - 4\rho(2\mu + \lambda) |k|^2 \quad (4.27)$$

$$\phi = \sqrt{\frac{4\rho(2\mu + \lambda)}{|k|^2}} - 2\psi \quad (4.28)$$

Now, to critically damp any particular wavelength  $\Lambda$ , we choose the mode  $k = \frac{2\pi}{\Lambda}$ , and put this — along with material parameters  $\lambda$ ,  $\mu$ , and  $\rho$  — into equations (4.25) and (4.28).

### 4.2.2 Material Limits to Critical Damping

It is interesting to note here what equations (4.25) and (4.28) imply. Thermodynamics requires that the damping parameters be non-negative, so combining these equations and  $\phi \geq 0$  means

$$\sqrt{\frac{4\rho(2\mu + \lambda)}{|k|^2}} - 2\sqrt{\frac{4\rho\mu}{|k|^2}} \geq 0 \quad (4.29)$$

$$\frac{2\sqrt{\rho}}{|k|} \left( \sqrt{2\mu + \lambda} - 2\sqrt{\mu} \right) \geq 0 \quad (4.30)$$

$$\sqrt{2\mu + \lambda} \geq 2\sqrt{\mu} \quad (4.31)$$

$$2\mu + \lambda \geq 4\mu \quad (4.32)$$

$$\lambda \geq 2\mu \quad (4.33)$$

Both steps (4.31) and (4.32) are because  $\rho, |k|, \mu, \lambda \geq 0$ .

This means that critical damping at a particular wavelength for both shear and compression waves is only possible in materials where the inequality (4.33) holds. Substituting the definitions of  $\lambda$  and  $\mu$ ,

$$\mu = \frac{E}{2(1 + \nu)} \quad (4.34)$$

$$\lambda = \frac{E\nu}{(1 + \nu)(1 - 2\nu)} \quad (4.35)$$

into (4.33), and assuming that  $E > 0$  and  $-1 < \nu < \frac{1}{2}$ , this can be simplified

further,

$$\frac{E\nu}{(1+\nu)(1-2\nu)} \geq 2 \left( \frac{E}{2(1+\nu)} \right) \quad (4.36)$$

$$\frac{\nu}{(1-2\nu)} \geq 1 \quad (4.37)$$

$$\nu \geq \frac{1}{3} \quad (4.38)$$

Therefore, with assumptions which are true for all practical materials, critical damping is only possible when Poisson's ratio,  $\nu$ , satisfies (4.38). This is an interesting limit, as many metals have a ratio near one third, and several common materials have ratios less than a third, such as concrete, glass, and steel.

### 4.3 Comparison of Methods

Now that we can choose damping parameters based on critically damping a desired wavelength, we will compare the new method to three different pairs of parameters. One set will critically damp the wavelength which corresponds to the size of the elements, the length of a triangle edge in this case, one will correspond to the radius, and the other to the diameter of the object. These parameters are given for both examples in Table 4.2.

In order to compare the behaviours, the distance from rest is plotted for simulations using the new method with varying values of  $\tau$ , and the traditional method for the above mentioned wavelengths. Figure 4.3 shows Example 1, where the traditional method shows much more oscillation than the new method, which is capable of critically damping both faster and

slower than the other method.

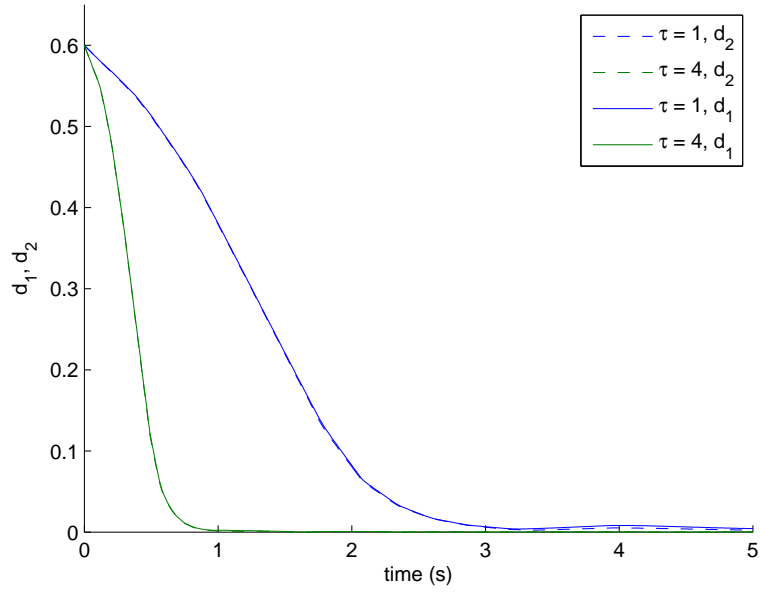
In the second example, Figure 4.4, the different wavelengths cause either too much oscillation overall or are over-damped. The new method shows the desired result for a heavily damped material, and although there are some oscillations shown in the  $d_2$  norm, both metrics show the new method coming to rest as fast or faster than the other method.

Parameter	Value	Parameter	Value
$\Lambda$	1	$\Lambda$	1
$k$	$2\pi$	$k$	$2\pi$
$\phi$	0.053134	$\phi$	0.022622
$\psi$	0.22508	$\psi$	0.050329
$\Lambda$	2	$\Lambda$	8
$k$	$\pi$	$k$	$\frac{1}{4}\pi$
$\phi$	0.10627	$\phi$	0.18098
$\psi$	0.45016	$\psi$	0.40263
$\Lambda$	4	$\Lambda$	16
$k$	$\frac{1}{2}\pi$	$k$	$\frac{1}{8}\pi$
$\phi$	0.21254	$\phi$	0.36196
$\psi$	0.90032	$\psi$	0.80527

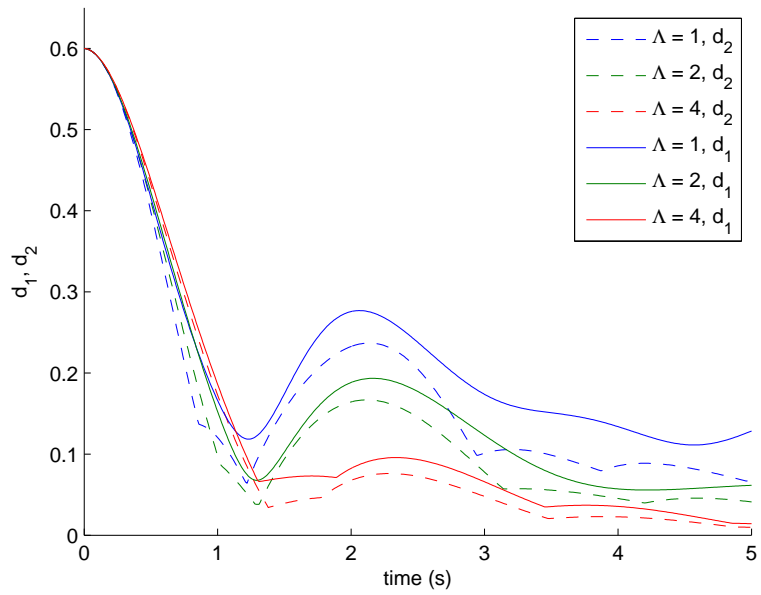
(a) Example 1

(b) Example 2

Table 4.2: Damping Parameters for Traditional Method.

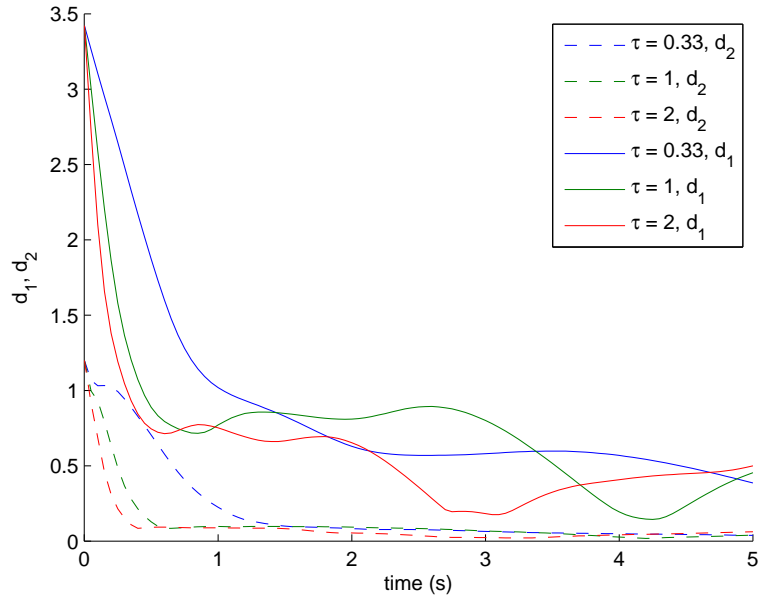


(a) New Method.

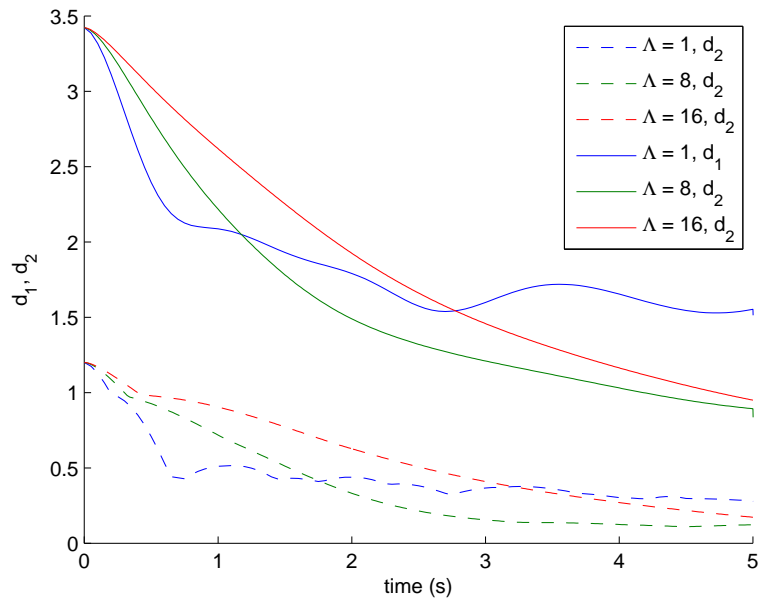


(b) Traditional Method.

Figure 4.3: Method Comparison for Example 1.



(a) New Method.



(b) Traditional Method.

Figure 4.4: Method Comparison for Example 2.

# Chapter 5

## Conclusions

We have presented a new method for animating deformable objects. This new method is inspired by physically-based methods, but is adapted to critically damp all modes and to provide an intuitive interface for animators. A new and much simpler form of control over the damping properties of these deformable materials is achieved. The method is capable of producing plausible motion, even when the requested action is not physically possible.

The new framework removes the need to choose unintuitive damping parameters, while still allowing for inertial effects and ballistic motion, providing an interesting middle-ground between quasi-static simulation and full physical simulation. The method maintains temporal coherence, and uses a framework compatible with techniques for simulating other materials.

### 5.1 Future Work

There are several advances this method could take. As well, further analysis could lead to a better understanding of traditional damping, and perhaps more intuitive control over traditional damping parameters for a physically accurate simulation.

### 5.1.1 Oscillation Control

In some situations, possibly for artistic effect, the oscillating behaviour of an under-damped system is desired. In order to incorporate this into the new method, we phrase a Backward Euler step of elastic forces in the minimization framework. Since elastic force is the negative gradient of the elastic potential energy, this step can be written as

$$\mathbf{v}^{new} = \mathbf{v} + \Delta t \mathbf{M}^{-1}(-\nabla PE(\mathbf{x}^{new})) \quad (5.1)$$

which solves

$$\min_{\mathbf{v}^{new}} \frac{1}{2}(\Delta \mathbf{v})^T \mathbf{M}(\Delta \mathbf{v}) + PE(\mathbf{x}^{new}) \quad (5.2)$$

where  $\Delta \mathbf{v} = \mathbf{v}^{new} - \mathbf{v}$  and  $\mathbf{x}^{new} = \mathbf{x} + \Delta t \mathbf{v}^{new}$ . The new velocity is a function of stress, and therefore so is  $\Delta \mathbf{v}$ . We can then define the new kinetic energy-like term as

$$KE_{osc}(\sigma) = \frac{1}{2}(\Delta \mathbf{v}(\sigma))^T \mathbf{M}(\Delta \mathbf{v}(\sigma)) \quad (5.3)$$

In order to combine the methods, we can blend between the fully elastic term and the kinetic energy in (3.10),

$$\min \int (\kappa) KE_{osc} + (1 - \kappa) KE + \frac{\tau}{\Delta t} PE \quad (5.4)$$

With  $\kappa = 0$ , we have the new method, (2.22), and with  $\kappa = 1$ , the system is no longer damped. For  $0 < \kappa < 1$ , we have a method with simple control over oscillations.



### 5.1.2 Plasticity

An important extension for this work is incorporating plastic flow. An object undergoes plastic deformation when the stress,  $\sigma$ , reaches the yield limit,  $\sigma_y$ . With the current framework, the modified total energy is minimized with respect to unconstrained stress. By adding constraints on stress, the calculated deformation would be purely elastic, and when the yield surface is reached, additional plastic deformation could be calculated.

$$\min_{\substack{\sigma \\ |\sigma| \leq \sigma_y}} \int_{\Omega} KE(\sigma) + \frac{\tau}{\Delta t} PE(\sigma) \quad (5.5)$$

The complications arise when the constraint becomes active and the object permanently changes, affecting even the material coordinates. The minimization may need to be completely restarted when this happens, and combined with the fact that permanent deformations remove some opportunities for pre-computations, the method may be quite slow without creative optimization.

### 5.1.3 Extending the Energy Minimization Framework

Since the underlying energy optimization structure can be applied to many other physical phenomena, from frictional contact to compressible flow, this new method lends itself to coupling with other simulations. For example, frictional contact between controlled-damping cloth and flesh could be calculated as a minimization of total kinetic energy and scaled potential energies with respect to internal stresses and contact forces. The contact forces would be subject to inequality constraints, requiring the normal component to be

positive and the tangential components to be within the friction cone. This could be further immersed in, and coupled to, an incompressible fluid simulation, by adding the kinetic energy of the fluid to the objective function, and adding fluid pressure as an independent variable.

# Bibliography

- [1] Christopher Batty, Florence Bertails, and Robert Bridson. A fast variational framework for accurate solid-fluid coupling. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 100, New York, NY, USA, 2007. ACM Press.
- [2] David E. Breen, Donald H. House, and Michael J. Wozny. Predicting the drape of woven cloth using interacting particles. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 365–372, New York, NY, USA, 1994. ACM Press.
- [3] R. Bridson, S. Marino, and R. Fedkiw. Simulation of clothing with folds and wrinkles. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*, page 3, New York, NY, USA, 2005. ACM Press.
- [4] P. Dorato, C. Abdallah, and V. Cerone. *Linear-Quadratic Control: An Introduction*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.
- [5] O. Eitzmuss, J. Gross, and W. Strasser. Deriving a particle system from continuum mechanics for the animation of deformable objects.

- Transactions on Visualization and Computer Graphics*, 9(4):538–550, 2003.
- [6] Allen Van Gelder. Approximate simulation of elastic membranes by triangulated spring meshes. *J. Graph. Tools*, 3(2):21–42, 1998.
- [7] S. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical Report TR-97-19, Mitsubishi Electric Research Lab., 1997.
- [8] Gentaro Hirota, Susan Fisher, Andrei State, Chris Lee, and Henry Fuchs. An implicit finite element method for elastic solids in contact. In *Proc. of Computer Animation*, pages 136–146, 2001.
- [9] Mary K. Kaiser, Dennis R. Proffitt, Susan M. Whelan, and Heiko Hecht. Influence of animation on dynamical judgments. *Journal of Experimental Psychology: Human Perception and Performance*, 18(3):669–690, 1992.
- [10] L. Kharevych, Weiwei Yang, Y. Tong, E. Kanso, J. E. Marsden, P. Schröder, and M. Desbrun. Geometric, variational integrators for computer animation. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 43–51, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.
- [11] Victor J. Milenkovic and Harald Schmidl. Optimization-based animation. In *SIGGRAPH '01: Proceedings of the 28th annual conference on*

## Bibliography

---

- Computer graphics and interactive techniques*, pages 37–46, New York, NY, USA, 2001. ACM Press.
- [12] Gavin S. P. Miller. The motion dynamics of snakes and worms. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 169–173, New York, NY, USA, 1988. ACM Press.
- [13] Andrew Nealen, Matthias Mller, Richard Keiser, Eddy Boxerman, and Mark Carlson. Physically based deformable models in computer graphics. *Eurographics State of the Art Report*, 2005.
- [14] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 1999.
- [15] Manfred Nusseck, Julien Lagarde, Benoit Bardy, Roland Fleming, and Heinrich H. Bülthoff. Perception and prediction of simple object interactions. In *APGV '07: Proceedings of the 4th symposium on Applied perception in graphics and visualization*, pages 27–34, New York, NY, USA, 2007. ACM Press.
- [16] James F. O'Brien, Adam W. Bargteil, and Jessica K. Hodgins. Graphical modeling and animation of ductile fracture. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 291–294, New York, NY, USA, 2002. ACM Press.
- [17] James F. O'Brien and Jessica K. Hodgins. Graphical modeling and animation of brittle fracture. In *SIGGRAPH '99: Proceedings of the 26th*

## Bibliography

---

- annual conference on Computer graphics and interactive techniques*, pages 137–146, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [18] Hidehiko Okabe, Haruki Imaoka, Takako Tomiha, and Haruo Niwaya. Three dimensional apparel cad system. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 105–110, New York, NY, USA, 1992. ACM Press.
- [19] Carol O’Sullivan, John Dingliana, Thanh Giang, and Mary K. Kaiser. Evaluating the visual fidelity of physically based animations. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 527–536, New York, NY, USA, 2003. ACM Press.
- [20] A. Pentland and J. Williams. Good vibrations: model dynamics for graphics and animation. In *SIGGRAPH '89: Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 215–222, New York, NY, USA, 1989. ACM.
- [21] Stephen M. Platt and Norman I. Badler. Animating facial expressions. In *SIGGRAPH '81: Proceedings of the 8th annual conference on Computer graphics and interactive techniques*, pages 245–252, New York, NY, USA, 1981. ACM Press.
- [22] Dennis R. Proffitt and David L. Gilden. Understanding natural dynamics. *Journal of Experimental Psychology: Human Perception and Performance*, 15(2):384–393, 1989.

## Bibliography

---

- [23] J. Teran, S. Blemker, V. Ng Thow Hing, and R. Fedkiw. Finite volume methods for the simulation of skeletal muscle. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 68–74, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [24] Joseph Teran, Eftychios Sifakis, Geoffrey Irving, and Ronald Fedkiw. Robust quasistatic finite elements and flesh simulation. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 181–190, New York, NY, USA, 2005. ACM Press.
- [25] Xiaoyuan Tu and Demetri Terzopoulos. Artificial fishes: physics, locomotion, perception, behavior. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 43–50, New York, NY, USA, 1994. ACM Press.
- [26] Keith Waters. A muscle model for animation three-dimensional facial expression. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 17–24, New York, NY, USA, 1987. ACM Press.