Modal Analysis for Real-Time Viscoelastic Deformation

Technical Sketch

Chen Shen   Kris K. Hauser   Christine M. Gatchalian   James F. O'Brien
University of California, Berkeley
{*csh,khauser,tine,job*}*@eecs.berkeley.edu*

Introduction

This technical sketch describes how a standard analysis technique known as modal decomposition can be used for real-time modeling of viscoelastic deformation. While most prior work on interactive deformation has relied on geometrically simple models and advantageously selected material parameters to achieve interactive speeds, the approach described here has two qualities that we belive should be required of a real-time deformation method: the simulation cost is decoupled from both the model's geometric complexity and from stiffness of the material's parameters. Additionally, the simulation may be advanced at arbitrarily large time-steps without introducing objectionable errors such as artificial damping.

Background

Modal decomposition is a commonly used tool for analyzing physical systems. A thorough review of the technique may be found in [Maia and Silva, 1998], but we will describe it very briefly here.

A physical system that has been discretized using a finite element, finite differencing, or other similar method can be expressed in the following general form:

$$\boldsymbol{K}\boldsymbol{d} + \boldsymbol{C}\dot{\boldsymbol{d}} + \boldsymbol{M}\ddot{\boldsymbol{d}} = \boldsymbol{f} \tag{1}$$

where $\boldsymbol{d}$ is the vector of node displacements, an over-dot indicates a derivative with respect to time, and $\boldsymbol{K}$, $\boldsymbol{C}$, are $\boldsymbol{M}$ are respectively known as the system's stiffness, damping, and mass matrices. In general the system matrices are not constant, but if the expected deformations are small then the system may be linearized. This linearization is the main limitation of the technique we describe. It means that we cannot use it for materials such as cloth, but we can use it for solid rubbery materials.

Modal decomposition treats (1) as a generalized eigenvalue problem and transforms it into a diagonalized system such as

$$\boldsymbol{\Lambda}\boldsymbol{z} + (\alpha_1\boldsymbol{\Lambda} + \alpha_2\boldsymbol{I})\dot{\boldsymbol{z}} + \ddot{\boldsymbol{z}}) = \boldsymbol{g} \tag{2}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues, $\boldsymbol{z}$ is the displacement vector expressed in the eigen-basis, and α_1 and α_2 ae the Raleigh coefficients.

This idea has been used previously in the graphics community for modeling deformation by [Pentland and Williams, 1989], however they did not actually use a modal decomposition, they only approximated it with global linear and quadratic basis functions. When done properly, (2) is *not* an approximation of (1), it is *exactly* (1), but expressed in a much more efficient basis. Furthermore since (2) is decoupled, analytical solutions may be used for each of the modes so that numerical time integration is no longer necessary. Instead the object's behavior may be computed by convolving the analytic solutions with any externally applied forces.

Real-Time Modeling

To model an object using modal analysis first require having a suitable discrete model of the object. We are using the finite element formulation described by [O'Brien and Hodgins, 1999] to compute the system matrices. The eigen-system is then computed using [Wu and Simon, 1999]. These two steps may require a few hours, but they are pre-computation that only needs to be done once for a given object.

At run-time, our system uses the modal decomposition for computing the response of an object to user interaction. Only the modes that are over-damped, critically-damped, or under-damped with a frequency below half the frame rate are used. Under-damped modes with a frequency greater than half the frame rate would only create temporal aliasing.

The behavior of each retained mode is modeled with a pair of complex oscillators. Interaction forces from the user are projected onto the eigen-basis and used to excite the oscillators. Because the oscillators do not require numerical time integration, we can run the simulation at arbitrarily large time-steps. The computation cost of the simulation is determined by the number of retained modes, not the geometric complexity of the model.

We have implemented this method. A 4000 vertex model can be modeled in real-time on a 500 MHz laptop PC. The bulk of the computation time is consumed by the graphics functions, not the simulation.

References

MAIA, N. M. M., AND SILVA, J. M. M., Eds. 1998. *Theoretical and Experimental Modal Analysis*. Research Studies Press, Hertfordshire, England.

O'BRIEN, J. F., AND HODGINS, J. K. 1999. Graphical modeling and animation of brittle fracture. In *Proceedings of SIGGRAPH 99*, 137–146.

PENTLAND, A., AND WILLIAMS, J. 1989. Good vibrations: Modal dynamics for graphics and animation. In *Proceedings of SIGGRAPH 89*, 215–222.

WU, K., AND SIMON, H. 1999. TRLAN user guide. Tech. Rep. LBNL-42953, Lawrence Berkeley National Laboratory.

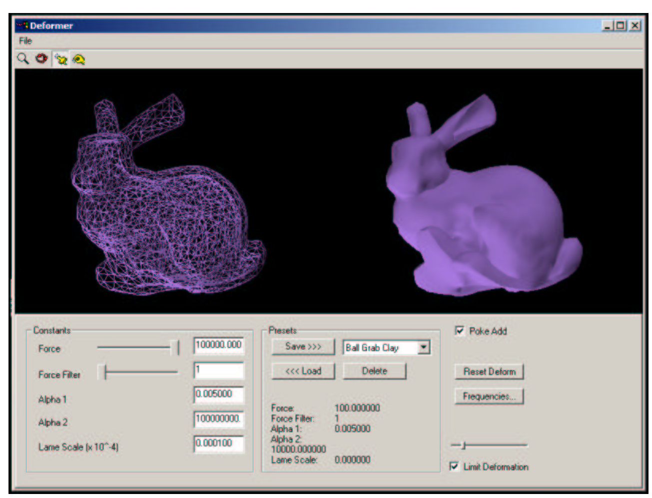

Figure 1: A screen shot from an interactive deformation application.