Using the Wave Equation to Describe Impacts

Elasticity in Collisions Using Complementarity

Ted Wendt

Department of Mathematics
The University of Iowa

GAUSS–Graduate and Undergraduate Student Seminar
November 5, 2007

Outline

Outline

Why Model Collisions?

Accurate simulation of collisions are needed in many areas.

- Video Games
- Physically Based Simulators
- Robotics
- Movie Special Effects

A Simple Motion Problem

The position of a projectile t seconds after being fired from ground level at some initial speed v_0 under the influence of gravity can be described with the function

$$x(t) = -gt^2 + v_0 t.$$

This function is only valid until a collision occurs.

A Generalization of the Impact Problem

Given an arbitrary collection of rigid bodies in motion, we would like to simulate their movement so that

- the bodies obey Newton's Laws of motion;
- the bodies respect each other's boundaries;
- small changes in the input parameters do not significantly change the result.

A Generalization of the Impact Problem

Given an arbitrary collection of rigid bodies in motion, we would like to simulate their movement so that

- the bodies obey Newton's Laws of motion;
- the bodies respect each other's boundaries;
- small changes in the input parameters do not significantly change the result.

A Generalization of the Impact Problem

Given an arbitrary collection of rigid bodies in motion, we would like to simulate their movement so that

- the bodies obey Newton's Laws of motion;
- the bodies respect each other's boundaries;
- small changes in the input parameters do not significantly change the result.

Newton's Laws of Motion

Newton's Laws of Motion

- Objects in motion tend to stay in motion; objects at rest tend to stay at rest (unless acted upon by an exterior force).
- $F = ma$
- For every action, there is an equal and opposite reaction.

The Catch

When two bodies collide, they exert forces upon each other according to Newton's Third Law. This can cause nearly instantaneous changes in the velocities and accelerations of the objects. (i.e. we may no longer have a continuous system!)

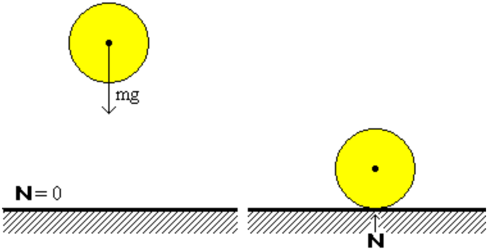
A Modeling Constraint

Computers are limited to only consider a finite number of calculations. In order to simulate a continuous system, we must therefore discretize our system in both time and space.

Outline

Simple Complementarity

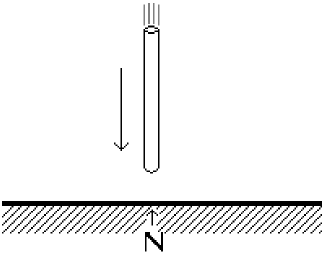
When objects are not in contact, they exert no (impact) forces upon one another. That is, either the distance between them is zero (in contact) or the forces exerted are zero (no contact).



Outline

The One Dimensional System

Consider a vertical rod of length L dropped above a horizontal plane.



The One Dimensional System

Let u represent the displacement of a point x on the rod at time t. The classical (complementarity) formulation for the motion of the rod is

$$\begin{aligned} \rho u_{tt} - \kappa u_{xx} &= f(t,x) \,\mathrm{in}\, (0,L) \\ -\frac{\partial u}{\partial \nu} &= N(t,x) \,\mathrm{at}\, x = 0 \\ \frac{\partial u}{\partial \nu} &= 0 \,\mathrm{at}\, x = L \\ 0 \le u(t,x) - g(x) &\perp N(t,x) \ge 0 \end{aligned}$$

where ρ is the density, κ is the 'stiffness' of the material, g is the 'gap-function', and N is the normal contact force.

Outline

General Formulation

We can generalize the one-dimensional formulation to three-dimension by applying the appropriate boundary forces. The motion of the object can be described by:

$$\begin{aligned} \rho u_{tt} - \kappa \Delta u &= f(t, \mathbf{x}) \,\mathrm{in}\, \Omega \\ -\frac{\partial u}{\partial \nu} &= N(t, \mathbf{x}) \,\mathrm{on}\, \Gamma_C \\ \frac{\partial u}{\partial \nu} &= 0 \,\mathrm{on}\, \Gamma_N \\ 0 \le u(t, \mathbf{x}) - g(\mathbf{x}) &\perp N(t, \mathbf{x}) \ge 0 \end{aligned}$$

The primary difficulty in solving this system is that the normal contact forces *need not be continuous!* That means our displacement *may not be differentiable!*

Weak Formulations

In order to solve this system of partial differential equations, we need to redefine what we mean by 'solution.' For now, let's assume that u is at least twice differentiable and satisfies

$$\begin{aligned}
\rho u_{tt} - \kappa \Delta u &= f(t, \mathbf{x}) \,\mathrm{in}\, \Omega \\
-\frac{\partial u}{\partial \nu} &= N(t, \mathbf{x}) \,\mathrm{on}\, \Gamma_C \\
\frac{\partial u}{\partial \nu} &= 0 \,\mathrm{on}\, \Gamma_N \\
0 \le u(t, \mathbf{x}) - g(\mathbf{x}) &\perp N(t, \mathbf{x}) \ge 0
\end{aligned}$$

Weak Formulations

Multiply by a continuously differentiable function $v(x)$ and integrate by parts:

$$\begin{aligned}
\int_\Omega \rho u_{tt} \cdot v \, d\mathbf{x} &= \int_\Omega \kappa \Delta u \cdot v \, d\mathbf{x} + \int_\Omega f \cdot v \, d\mathbf{x} \\
\frac{d^2}{dt^2} \int_\Omega \rho u \cdot v \, d\mathbf{x} &= -\kappa \int_{\Gamma_C} N \cdot v \, d\Gamma \\
&\quad - \kappa \int_\Omega \nabla u \cdot \nabla v \, d\mathbf{x} \\
&\quad + \int_\Omega f \cdot v \, d\mathbf{x}
\end{aligned}$$

This formulation only requires u to be once differentiable in space.

Outline

Finite Element Method

Instead of trying to directly solve the system of differential equations on the entire object, we instead choose to *approximate* the solution on very small subsets of the body. This turns the problem from infinite dimensional (i.e. find the solution at all points in the body) to finite dimensional (find the solution at finitely many prechosen points).

Finite Element Basis

Approximate Ω by N tetrahedral elements Ω_h, and label the nodes of this triangulation $\{x_i\}_{i=1}^N$. Choose piecewise linear basis functions $\{\phi_i\}_{i=1}^N$ so that $\phi_i(x_j) = \delta_{ij}$. We will write our approximate solution

$$u(t, \mathbf{x}) \approx u_h(t, \mathbf{x}) = \sum_{i=1}^{N} u_{h,i}(t)\phi_i(\mathbf{x}).$$

Finite Element Basis

Then our problem becomes: Find u_h such that

$$\begin{aligned}\sum_{i=1}^{N} \frac{d^2 u_{h,i}}{dt^2} \int_{\Omega_h} \rho \phi_i \cdot \phi_j \, d\mathbf{x} &= -\kappa \int_{\Gamma_C} \phi_j \cdot N \, d\Gamma \\ &\quad -\kappa \sum_{i=1}^{N} u_{h,i} \int_{\Omega_h} \nabla \phi_i \cdot \nabla \phi_j \, d\mathbf{x} \\ &\quad + \int_{\Omega_h} f \phi_j \, d\mathbf{x}.\end{aligned}$$

Simplification

Putting

$$M_{ij}^h = \int_{\Omega_h} \phi_i(\mathbf{x})\phi_j(\mathbf{x})\, d\mathbf{x},$$

$$K_{ij}^h = \int_{\Omega_h} \nabla\phi_i(\mathbf{x})\nabla\phi_j(\mathbf{x})\, d\mathbf{x}$$

$$f_j^h = \int_{\Omega_h} f \cdot \phi_j(\mathbf{x})\, d\mathbf{x}$$

and

$$\int_{\Gamma_C} \phi_j(\mathbf{x}) \cdot N(t,\mathbf{x})\, d\Gamma(\mathbf{x}) = \mathbf{n}(t) \int_{\Gamma_C} \phi_j(\mathbf{x})\, d\Gamma(\mathbf{x}) := \mathbf{N}$$

gives

$$\rho \mathbf{M}^h \frac{d^2\mathbf{u}_h}{dt^2}(t) = -\kappa\mathbf{N}(t) + \kappa\mathbf{K}^h\mathbf{u}_h(t) + \mathbf{f}^h(t).$$

This is convenient because $\mathbf{M}^h$, $\mathbf{K}^h$, and $\mathbf{f}^h$ can be precomputed.

Using The Reference Element

Instead of computing each integral

$$\int_{\Omega_H} \phi_i(\mathbf{x})\phi_j(\mathbf{x})\, d\mathbf{x}$$

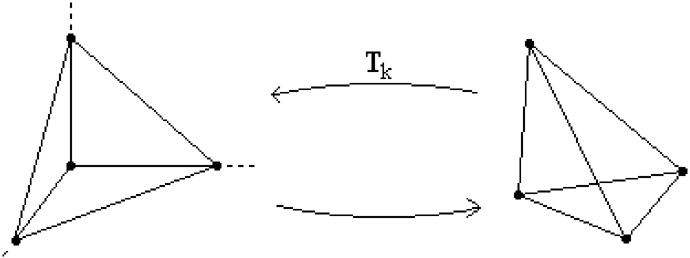
for each tetrahedron in the approximate mesh, we instead calculate a representative integral over a 'nice' tetrahedron, then use a basic transformation rule to find the value we need for our simulation.

Using The Reference Element

Let $\mathbf{T}_k$ be a linear transformation from the mesh tetrahedron K to the reference element $\hat{K}$. Then

$$\int_K \phi_i(\mathbf{x})\phi_j(\mathbf{x})\,d\mathbf{x} = \frac{1}{6}\int_{\hat{K}} \hat{\phi}_i(\mathbf{x})\hat{\phi}_j(\mathbf{x})\,d\mathbf{x} \cdot |\det \mathbf{T}_k|$$

$$\int_K \nabla\phi_i(\mathbf{x})\nabla\phi_j(\mathbf{x})\,d\mathbf{x} = \frac{1}{6}\int_{\hat{K}} \nabla\hat{\phi}_i(\mathbf{x})\nabla\hat{\phi}_j(\mathbf{x})\,d\mathbf{x} \cdot |\det \mathbf{T}_k|$$



Outline

ODE Formulation and Discretization

Put $\mathbf{v} = \frac{d\mathbf{u}}{dt}$. Then we have a system of ODEs

$$\mathbf{v} = \frac{d\mathbf{u}}{dt}$$
$$\rho \mathbf{M}\frac{d\mathbf{v}}{dt} = \kappa \mathbf{K}\mathbf{u}(t) + \mathbf{f}(t) + \kappa \mathbf{N}(t).$$

We use an implicit midpoint scheme to discretize **u**, **v**, and **f**, and a fully implicit scheme to discretize **N**.

$$\left(\frac{\mathbf{v}^{l+1} + \mathbf{v}^{l}}{2}\right) = \left(\frac{\mathbf{u}^{l+1} - \mathbf{u}^{l}}{\Delta t}\right)$$
$$\rho \mathbf{M}\left(\frac{\mathbf{v}^{l+1} - \mathbf{v}^{l}}{\Delta t}\right) = \kappa \mathbf{K}\left(\frac{\mathbf{u}^{l+1} + \mathbf{u}^{l}}{2}\right) + \left(\frac{\mathbf{f}^{l+1} + \mathbf{f}^{l}}{2}\right) + \kappa \mathbf{N}^{l+1}.$$

Time Stepping

If we solve for $\mathbf{u}^{l+1}$ and $\mathbf{v}^{l+1}$, we find

$$\begin{aligned}
\mathbf{v}^{l+1} &= \left(\mathsf{M} - \frac{\kappa \Delta t^2}{4\rho}\mathsf{K}\right)^{-1} \\
&\quad \times \left[\left(\mathsf{M} + \frac{\kappa \Delta t^2}{4\rho}\right)\mathbf{v}^l + \frac{\kappa \Delta t}{\rho}\mathbf{u}^l + \frac{\Delta t}{2\rho}\left(\mathbf{f}^{l+1} + \mathbf{f}^l\right) + \frac{\kappa \Delta t}{\rho}\mathsf{N}^{l+1}\right] \\
\mathbf{u}^{l+1} &= \frac{\Delta t}{2}\left(\mathbf{v}^{l+1} + \mathbf{v}^l\right) + \mathbf{u}^l.
\end{aligned}$$

The only unknown value is N^{l+1}, which we can find by applying the complementarity condition.

Outline

Complementarity Matrix Formulation

Put

$$\begin{aligned} \mathsf{T} &= \frac{\kappa \Delta t^2}{2\rho}\left(\mathsf{M} - \frac{\kappa \Delta t^2}{4\rho}\mathsf{K}\right)^{-1} \\ \mathbf{q} &= \left\{\frac{\Delta t}{2}\mathbf{v}^l + \mathbf{u}^l + \frac{\Delta t}{2}\mathbf{v}^{l+1} - \mathbf{g}\right\}. \end{aligned}$$

Then the complementarity condition becomes

$$\mathbf{0} \leq \mathsf{T}\mathsf{N}^{l+1} + \mathbf{q} \perp \mathsf{N}^{l+1} \geq \mathbf{0}.$$

This is a linear complementarity problem, which can be solved exactly using Lemke's algorithm.

Simulation Examples

Let's look at a few different configurations of the model.

Possible Extensions

- Allow for rotations (simple generalization)
- Allow for more complex shapes (difficulty depends on shape)
- Add friction (hard)
- Allow for dissipation of energy due to viscosity

Brogliato, B.
Nonsmooth Mechanics.
Springer, 1999.

Duvaut, G. and Lions, J.
Inequalities in Mechanics and Physics.
Springer, 1972.

Stronge, W.
Impact Mechanics.
Cambridge University Press, 2000.

Erleben, K.
Stable, Robust, and Versatile Multibody Dynamics Animation.
Doctoral Thesis, 2004.

Resources

Presentation references and materials can be found at the following website:

http://www.math.uiowa.edu/~twendt/