Nanoquiz

• go to shoutkey.com/primrose
• 3 minutes (timer in the upper right)
• closed book
• closed notes
• no talking
• no email/FB/IM
• be a good person
• make your mother proud
1. Which statements are correct: (choose all that apply)
A. Mechanisms are made of joints and links
B. A closed chain mechanism has no loops
C. A Rigid Body in 2D has 3 Degrees-of-Freedom
D. Kinematics is the study of forces and mass
An Introduction to Continuum Mechanics and Finite Element Methods

David I.W. Levin
EECS and CSAIL
6.S079
Previous Lecture

- Kinematics of Mechanisms
- No forces!
- Rigid Bodies Only!
Today's Lecture

- What if our objects are deformable?
Todays Lecture

- How to model deformable objects using continuum mechanics
- An introduction to the finite element method for simulating soft bodies
Plan for Today

- A Review of Mass Spring Systems
- Mass Spring Systems vs. Continuum Approaches
- Implementing Continuum Mechanics: The Finite Element Method
Why you should be paying attention

Face Example

High Velocity Impact
1/500x Speed
915k Voxels
Why you should be paying attention
One way of modeling deformable objects is as a network of masses and springs.
The Motion of a Mass Spring Systems

- The acceleration of a point mass is given by:
  \[ ma = f \]

- We can use standard methods to integrate this system forward in time in order to compute the velocities and displacements of each point in the mass spring system.
The Motion of a Mass Spring Systems

- Pseudocode:
  - \( a \leftarrow 0 \) //Array of accelerations
  - \( v \leftarrow 0 \) //Array of velocities
  - \( p \leftarrow 0 \) //Array of positions
  - For each particle, \( p \)
    - \( m \leftarrow \text{mass of the particle} \)
    - \( f \leftarrow \text{sum of all spring forces acting on the particle} \)
    - \( a[p] \leftarrow f/m \)
  - End
  - \( v \leftarrow \text{Integrate}(a) \)
  - \( p \leftarrow \text{Integrate}(v) \)
The Motion of a Mass Spring Systems

- **Pseudocode:**
  - \( a \leftarrow 0 \) //Array of accelerations
  - \( v \leftarrow 0 \) //Array of velocities
  - \( P \leftarrow 0 \) //Array of positions
  - For each particle, \( p \)
    - \( m \leftarrow \) mass of the particle
    - \( f \leftarrow \) sum of all spring forces acting on the particle
    - \( a[p] \leftarrow f/m \)
  - End
  - \( v \leftarrow \) Integrate(a)
  - \( p \leftarrow \) Integrate(v)
Mass Spring Systems: Forces

- The inertia on this point mass is given by:

\[
ma = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right)
\]
The acceleration on this point mass is given by:

\[ ma = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right) \]
Mass Spring Systems: Refresher

- The acceleration on this point mass is given by:

\[
ma = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right)
\]

\[
l = |x - y_i|
\]

\[
l_0 = \text{Original length}
\]
Mass Spring Systems: Refresher

- The acceleration on this point mass is given by:

\[
ma = \sum_{i=1}^{springs} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right)
\]

\[
l = |x - y_i|
\]

\[
l_0 = \text{Original length}
\]
Mass Spring Systems: Refresher

- The acceleration on this point mass is given by:

\[ \mathbf{ma} = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{\mathbf{x} - \mathbf{y}_i}{|\mathbf{x} - \mathbf{y}_i|} \right) \]

\[ l = |\mathbf{x} - \mathbf{y}_i| \]

\[ l_0 = \text{Original length} \]
Mass Spring Systems: Refresher

- The acceleration on this point mass is given by:

\[ ma = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right) \]

\[ l = |x - y_i| \]

\[ l_0 = \text{Original length} \]
Requirements to Compute Mass Spring Forces

1. Material Model

2. Measure of Deformation
   - Continuum Methods require the same two pieces of information
   - We’re going to replace our discrete springs with continuous volumes using the *Continuum Assumption*
Requirements to Compute Mass Spring Forces

1. Material Model \( k \)

2. Measure of Deformation \( \left( \frac{l}{l_0} - 1 \right) \)

- Continuum Methods require the same two pieces of information
- We’re going to replace our discrete springs with continuous volumes using the **Continuum Assumption**
An Introduction to Continuum Mechanics
An Example
An Example
Requirements for Continuum Mechanics

1. Material Model

2. Measure of Deformation
Continuum Mechanics: Deformation

- Defining a measure of deformation:

\[
\text{World } \mathbf{x} = \text{World } \phi \left( \mathbf{q}, \text{Body } \mathbf{x} \right)
\]
Rigid Bodies

- Just a reminder: I used the same trick to describe rigid bodies

\[
\text{World } \mathbf{x} = \text{World } \mathbf{q} \phi (\mathbf{q}, \text{Body } \mathbf{x})
\]
Continuum Mechanics

- We’ll leave our mapping undefined (For Now 😊 )

\[
\text{World } \mathbf{x} = \text{World}_{\text{Body}} \phi \left( q, \text{Body } \mathbf{x} \right)
\]
• Consider the effect of $\begin{bmatrix} w \\ \phi \end{bmatrix}$ on single vector in our reference space.
Continuum Mechanics: Deformation

- Consider the effect of $\frac{\omega}{b}$ on single vector in our reference space.

We’re pretending we have springs EVERYWHERE in our object!!!
Continuum Mechanics: Deformation

- New positions are given by passing the reference positions through $\phi$.

\[ x_2 = \phi (X_2) \]
Continuum Mechanics: Deformation

- Just rephrasing so we can see the spring vector

\[ x_1 + dx = \phi (X_1 + dX) \]
An Aside: Taylor Expansion

- Approximate small change in a non-linear function

\[ \text{slope} = \frac{\partial f}{\partial x} \]
An Aside: Taylor Expansion

- Approximate small change in a non-linear function

\[ \Delta f = \frac{\partial f}{\partial x} \Delta x \]
An Aside: Taylor Expansion

- Approximate small change in a non-linear function

\[ f(x + \Delta x) \approx f(x) + \frac{\partial f}{\partial x} \Delta x \]
Taylor Expansion: Multi-dimensional Functions

- Almost exactly the same!

\[ f(x + \Delta x) \approx f(x) + \begin{pmatrix} \frac{\partial f}{\partial x} \end{pmatrix} \Delta x \]

Gradient Matrix
Continuum Mechanics: Deformation

- Apply Taylor Expansion

\[ x_1 + dx = \phi (X_1 + dX) \]
Continuum Mechanics: Deformation

$$\mathbf{x}_1 + d\mathbf{x} \approx \phi(\mathbf{X}_1) + \frac{\partial \phi}{\partial \mathbf{X}} d\mathbf{X}$$
Continuum Mechanics: Deformation

- $\mathbf{x}_1$ and $\phi (\mathbf{X}_1)$ are the same so we’re left with ...

\[
\mathbf{x}_1 + d\mathbf{x} \approx \phi (\mathbf{X}_1) + \frac{\partial \phi}{\partial \mathbf{X}} d\mathbf{X}
\]
Continuum Mechanics: Deformation

\[ dx \approx \frac{\partial \phi}{\partial \mathbf{X}} \, d\mathbf{X} \]
Continuum Mechanics: Deformation

- \( F \) is our deformation measure called the deformation gradient

\[
dx \approx FdX
\]
Continuous Deformation vs. Mass Spring

- Spring Force:

\[-k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right)\]

- Deformation:

\[\left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|}\]

Equivalent to F
Continuous Deformation vs. Mass Spring

- Undeformed Spring

\[ \frac{l}{l_0} = ? \]

- Continuum:
Continuous Deformation vs. Mass Spring

- Undeformed Spring

\[ \frac{l}{l_0} = 1 \]

- Deformation:

\[ \mathbf{F} = ? \]
F is our deformation measure called the deformation gradient

\[ d\mathbf{x} \approx \mathbf{F} d\mathbf{X} \]
A Few Slides Ago...

- If there is no deformation $dx = dX$

$$dx \approx \mathbf{F} d\mathbf{X}$$
A Few Slides Ago…

- If there is no deformation $dx = dX$

$$dx \approx FdX$$
Continuous Deformation vs. Mass Spring

- Undeformed Spring
  \[ \frac{l}{l_0} = 1 \]

- Deformation:
  \[ F = 1 \]
Continuous Deformation vs. Mass Spring

- Spring Force:

\[ -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \right) \]

- Deformation:

\[ \left( \frac{l}{l_0} - 1 \right) \frac{x - y_i}{|x - y_i|} \]

This is called Strain
Properties of a Strain

• Spring Strain

\[ \left( \frac{l}{l_0} - 1 \right) \]

• Property: 0 if spring is undeformed

• Can we find a similar measure that would work for an arbitrary volume?

• Any guesses?
Proposed Strain Measure

- Let’s try and use the difference between the lengths of our deformed vector and its undeformed counterpart.
Proposed Strain Measure

- Let’s try and use the difference between the lengths of our deformed vector and its undeformed counterpart.

\[
\begin{align*}
&l_0 \\
&l
\end{align*}
\]
Proposed Strain Measure

- Let’s try and use the difference between the lengths of our deformed vector and its undeformed counterpart.
Proposed Strain Measure: Distance Between Points

- Substitute in formula’s for length squared

\[ X_1 dX^T dX X_2 \]
Proposed Strain Measure: Distance Between Points

- Use $dx \approx FdX$
Proposed Strain Measures: Distance Between Points

- We want to quantify change in shape so we can take the difference of the original and deformed lengths

\[
\frac{1}{2} \left( F^T F - I \right)
\]
Continuous Deformation vs. Mass Spring

- **Undeformed Spring**

\[
\frac{l}{l_0} - 1 = ?
\]

- **Deformation:**

\[
\frac{1}{2} \left( F^T F - I \right) = ?
\]
Recall...

- **Undeformed Spring**

\[
\frac{l}{l_0} - 1 = 0
\]

- **Deformation:**

\[
\frac{1}{2} \left( F^T F - I \right) = ?
\]
Continuous Deformation vs. Mass Spring

- **Undeformed Spring**

  \[ \frac{l}{l_0} - 1 = 0 \]

- **Deformation:**

  \[ \frac{1}{2} (F^T F - I) = 0 \]
1. **Material Model**

2. **Measure of Deformation**
   - Continuum Methods require the same two pieces of information
   - We’re going to replace our discrete springs with continuous volumes using the *Continuum Assumption*
Continuum Mechanics: The Required Stuff

1. Material Model

2. Measure of Deformation

\[
\frac{1}{2} \left( F^T F - I \right)
\]

- We just need the material model now
Material Models in Continuum Mechanics

- Materials models in continuum mechanics convert strain into a force per unit area called a stress

\[ \sigma = \psi (E) \]

Stress, it’s a matrix!
Material Models in Continuum Mechanics

- Materials models in continuum mechanics convert strain into a force per unit area called a stress.

\[ \sigma = \psi (E) \]

- Stress
- Strain
Material Models in Continuum Mechanics

- Materials models in continuum mechanics convert strain into a force per unit area called a stress

\[ \sigma = \psi (E) \]

Stress \rightarrow \psi \rightarrow (E) \rightarrow Strain

Material Model
Material Representation

Stress-Strain relationship 1D

- Strain
  \[ \varepsilon = \frac{u}{L} \]
- Stress
  \[ \sigma = \frac{F}{A} \]
Continuous Deformation vs. Mass Spring

- Undeformed Spring

\[-k\left(\frac{x - y_i}{l_0} - 1\right)\]

- Deformation:

\[\sigma = \psi\left(\mathbf{E}\right)\]

- What’s \(\psi\)?
Continuous Deformation vs. Mass Spring

- Undeformed Spring

\[ k \left( \frac{(x - y_i)}{l_0} - 1 \right) \]

- Deformation:

\[ \sigma = KE \]

4th order tensor
Continuous Deformation vs. Mass Spring

- Undeformed Spring

\[ -k \left( \frac{(x - y_i)}{l_0} - 1 \right) \]

- Deformation:

\[ \sigma = KE \]

More next lecture!

4\textsuperscript{th} order tensor
1. Material Model

\[ \sigma = KE \]

2. Measure of Deformation

\[ \frac{1}{2} \left( F^T F - I \right) \]

- We have all the pieces now
1. Material Model
\[ \sigma = KE \]

2. Measure of Deformation
\[ \frac{1}{2} (F^T F - I) \]

• One more thing...
Mass Spring Systems

- We need the equivalent formula for computing force from stress!

\[ m \mathbf{a} = \sum_{i=1}^{\text{springs}} -k \left( \left( \frac{l}{l_0} - 1 \right) \frac{\mathbf{x} - \mathbf{y}_i}{|\mathbf{x} - \mathbf{y}_i|} \right) \]
Continuous Deformation vs. Mass Spring

- **Undeformed Spring**
  \[-k \left( \frac{(x - y_i)}{l_0} - 1 \right)\]

- **Continuum Mechanics:**
  \[\int_{\Omega} \nabla \cdot \sigma \, d\Omega\]

- **Without derivation (just trust me)**
Implementing Continuum Mechanics

- We have formulas for deformation, strain, stress and force
- How do we simulate deformable objects using these formulas?
- The Finite Element Method
One way of modeling deformable objects is as a network of masses and springs.
One way of modeling deformable objects is as a mesh of elements.
The Motion of a Finite Element System

- The acceleration of a vertex is given by:
  \[ ma = f \]

- We can use standard methods to integrate this system forward in time in order to compute the velocities and displacements of each point in the mass spring system.
The Motion of a Mass Spring Systems

- Pseudocode:
  - Let a be an n dimensional array of accelerations
  - \(a \leftarrow 0\) //Array of accelerations
  - \(v \leftarrow 0\) //Array of velocities
  - \(p \leftarrow 0\) //Array of positions
  - For each particle, \(p\)
    - \(m \leftarrow \text{mass of the particle}\)
    - \(f \leftarrow \text{sum of all spring forces acting on the particle}\)
    - \(a[p] \leftarrow f/m\)
  - End
  - \(v \leftarrow \text{Integrate}(a)\)
  - \(p \leftarrow \text{Integrate}(v)\)
The Motion of a Mass Spring Systems

• Pseudocode:
  - Zero all arrays
  - For each triangle, t
    • \( m \leftarrow \text{density of triangle} \times \text{area} \)
    • For each edge, e, in t
      - \( f_e \leftarrow \text{force for this edge} \)
      - \( f[\text{end point 0}] += 0.5 \times f \)
      - \( f[\text{end point 1}] += 0.5 \times f \)
    • End
  - For each vertex, v, in t
    • \( m[v] += m/3 \)
  • End
  - End
  - For each vertex, p
    • \( a[p] \leftarrow f[p] / m[p] \)
  - End
  - \( v \leftarrow \text{Integrate}(a) \)
  - \( p \leftarrow \text{Integrate}(v) \)
Pseudocode:

- Zero all arrays
- For each triangle, $t$
  - $m = \text{density of triangle} \times \text{area}$
  - For each edge, $e$, in $t$
    - $f = \text{force for this edge}$
    - $f[\text{end point 0}] += 0.5 \times f$
    - $f[\text{end point 1}] += 0.5 \times f$
  - End
  - For each vertex, $v$, in $t$
    - $m[v] += \frac{m}{3}$
  - End
- End
- For each vertex, $p$
  - $a[p] = \frac{f[p]}{m[p]}$
- End
- $v = \text{Integrate}(a)$
- $p = \text{Integrate}(v)$
- Iterate over all triangles
The Motion of a Mass Spring Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, \( t \)
    - \( m \) \( \leftarrow \) density of triangle * area
    - For each edge, \( e \) in \( t \)
      - \( f_e \) \( \leftarrow \) force for this edge
      - \( f_{[\text{end point 0}]} \) += 0.5*\( f \)
      - \( f_{[\text{end point 1}]} \) += 0.5*\( f \)
    - End
    - For each vertex, \( v \) in \( t \)
      - \( m[v] \) += \( m/3 \)
    - End
  - End
  - For each vertex,
    - \( a[p] \) \( \leftarrow \) \( f[p]/m[p] \)
  - End
  - \( v \) \( \leftarrow \) Integrate(\( a \))
  - \( p \) \( \leftarrow \) Integrate(\( v \))

Iterate over all triangles

Forces computed on each edge and distributed to the end points
The Motion of a Mass Spring Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, t
    - \( m \leftarrow \text{density of triangle} \times \text{area} \)
    - For each edge, e, in t
      - \( f_e \leftarrow \text{force for this edge} \)
      - \( f[\text{end point 0}] += 0.5 \times f \)
      - \( f[\text{end point 1}] += 0.5 \times f \)
    - End
    - For each vertex, v, in t
      - \( m[v] += m/3 \)
    - End
  - End
  - For each vertex, p
    - \( a[p] \leftarrow f[p]/m[p] \)
  - End
  - \( v \leftarrow \text{Integrate}(a) \)
  - \( p \leftarrow \text{Integrate}(v) \)

Iterate over all triangles

Forces computed on each edge and distributed to the end points

Mass computed per-triangle and distributed to vertices
The Motion of a Finite Element Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, t
    - $m \leftarrow \text{density of triangle} \times \text{area}
    - Compute $F$
    - Compute $\epsilon = \frac{1}{2} (F^T F - I)$
    - Compute $\sigma = \psi (\epsilon)$
    - For each edge, e, in t
      - $fe \leftarrow \sigma n l$
      - $f[\text{end point 0}] += 0.5*fe$
      - $f[\text{end point 1}] += 0.5*fe$
    - End
    - For each vertex, v, in t
      - $m[v] += m/3$
    - End
  - End
  - ...
Overview of Finite Elements

1. Pick an object to simulate
2. Divide object into “elements”, small simple volumetric shapes
Overview of Finite Elements

3. Compute forces acting on each element and integrate

For each tet:

\[ \int_{\Omega} \nabla \cdot \sigma \, d\Omega \]
The Motion of a Finite Element Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, t
    - \( m \leftarrow \text{density of triangle} \times \text{area} \)
    - Compute \( F \)
    - Compute \( \varepsilon = \frac{1}{2} (F^T F - I) \)
    - Compute \( \sigma = \psi (\varepsilon) \)
    - For each edge, e, in t
      - \( \text{fe} \leftarrow \sigma n l \)
      - \( f[\text{end point 0}] += 0.5 \times \text{fe} \)
      - \( f[\text{end point 1}] += 0.5 \times \text{fe} \)
    - End
  - For each vertex, v, in t
    - \( m[v] += m/3 \)
  - End
- End
- ...
Finite Elements, One Element at a Time

- More details
- A single element becomes a triangle
Continuum Mechanics

- Let’s define our mapping (Finally!)

\[
\mathbf{x}_{\text{World}} = \mathbf{x}_{\text{Body}} \phi (q_{\text{Body}}, \mathbf{x})
\]
Let’s define our mapping (Finally!)

\[
\text{World } x = \text{World } \phi (q, \text{Body } x)
\]
Let’s just interpolate positions from the vertices to the interior of the mesh.
Finite Elements: Positions

- Let’s just interpolate positions from the vertices to the interior of the mesh

\[ w \phi = b_1 \lambda_1 + b_2 \lambda_2 + b_3 \lambda_3 \]

\[ \lambda_1, \lambda_2, \lambda_3 \] - Barycentric Coordinates
Barycentric Coordinates
Barycentric Coordinates
The Motion of a Finite Element Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, t
    - \( m \leftarrow \text{density of triangle} \times \text{area} \)
    - Compute \( F \)
    - Compute \( \epsilon = \frac{1}{2} (F^T F - I) \)
    - Compute \( \sigma = \psi (\epsilon) \)
    - For each edge, e, in t
      - \( \text{fe} \leftarrow \sigma n l \)
      - \( f[\text{end point 0}] += 0.5 \times \text{fe} \)
      - \( f[\text{end point 1}] += 0.5 \times \text{fe} \)
    - End
  - For each vertex, v, in t
    - \( m[v] \leftarrow m/3 \)
  - End
  - End
  - ...
Finite Elements: Deformation

- How do we compute $\mathbf{F}$ from $^w_b \phi$?

$$^w_b \phi = ^b \mathbf{x}_1 \lambda_1 + ^b \mathbf{x}_2 \lambda_2 + ^b \mathbf{x}_3 \lambda_3$$

$\lambda_1, \lambda_2, \lambda_3$ - Barycentric Coordinates
Continuum Mechanics: Deformation

- $\mathbf{F}$ is our deformation measure called \textit{the deformation gradient}

\[
\text{Reference Space} \quad \text{World Space}
\]

\[
\mathbf{X}_1 \rightarrow \mathbf{X}_2
\]

\[
d\mathbf{x} \approx \mathbf{F}d\mathbf{X}
\]
Finite Elements: Deformation

- How do we compute $F$ from $\begin{bmatrix} w \\ \phi \end{bmatrix}$?

$$w = b x_1 \lambda_1 + b x_2 \lambda_2 + b x_3 \lambda_3$$

$\lambda_1, \lambda_2, \lambda_3$ - Barycentric Coordinates
Finite Elements: Deformation

- How do we compute $F$ from $\mathbf{w}^T \phi$?

$$\frac{\partial \phi}{\partial \mathbf{b}} = \begin{pmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{pmatrix} \begin{pmatrix} X_1 - X_4 & X_2 - X_4 & X_3 - X_4 \\ Y_1 - Y_4 & Y_2 - Y_4 & Y_3 - Y_4 \\ Z_1 - Z_4 & Z_2 - Z_4 & Z_3 - Z_4 \end{pmatrix}^{-1}$$
Finite Elements: Strain

- Compute $F$
- Compute strain: $\frac{1}{2} (F^T F - I)$
The Motion of a Finite Element Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, $t$
    - $m \leftarrow$ density of triangle * area
    - Compute $F$
    - Compute $\varepsilon = \frac{1}{2} (F^T F - I)$
    - Compute $\sigma = \psi(\varepsilon)$
    - For each edge, $e$, in $t$
      - $fe \leftarrow \text{snl}$
      - $f[\text{end point 0}] += 0.5 \times fe$
      - $f[\text{end point 1}] += 0.5 \times fe$
    - End
  - For each vertex, $v$, in $t$
    - $m[v] += m/3$
  - End
- End
- ...

- ...
Finite Elements: Computing Forces

- How do we evaluate the forces:
  \[ \int_{\Omega} \nabla \cdot \sigma d\Omega \]

- For each edge in our triangle:
  - Compute \( f = \sigma n l \)
  - Distribute \( f \) evenly to vertices on the edge

(\( l \) is the length of the edge)
The Motion of a Finite Element Systems

- **Pseudocode:**
  - Zero all arrays
  - For each triangle, t
    - \( m \leftarrow \text{density of triangle} \times \text{area} \)
    - Compute \( F \)
    - Compute \( \epsilon = \frac{1}{2} (F^TF - I) \)
    - Compute \( \sigma = \psi(\epsilon) \)
    - For each edge, e, in t
      - \( \text{fe} \leftarrow \sigma n l \)
      - \( f[\text{end point 0}] += 0.5 \times \text{fe} \)
      - \( f[\text{end point 1}] += 0.5 \times \text{fe} \)
    - End
  - For each vertex, v, in t
    - \( m[v] += m/3 \)
  - End
  - End
  - ...

...
So now you to can do this:
Next Week

- Integrating real-data into Finite Element Models
- Material Models and (a little) Measurement

- Further Reading (Very Good !!)
  
  http://www.matthiasmueller.info/realtimephysics/