

# Simulation of the unstable rotation of a cuboid

## Books in space

Ivar Postma   Eamon Nerbonne

Introduction to Computational Science  
School for Computing and Cognition  
University of Groningen

January 19, 2011

# What you should remember from last time...

- ▶ Rotation of a rigid body; a book
- ▶ Clip of an experiment in space
- ▶ Rotation around two of the main axes is stable, rotation around the third is not

# String particle simulation

- ▶ Lecture on Molecular Dynamics: simulation of vibrating string with particles
- ▶ Springs create interacting forces between particles
- ▶ From forces, calculate update velocity of points ( $a = \frac{F}{m}$ )
- ▶ From velocity, calculate update position of points

# Particle simulation

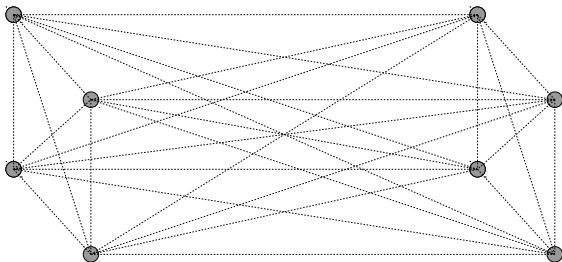
## Model of a book

- ▶ Eight particles, located at the corners
- ▶ Each point has the same mass
- ▶ Interacting force (spring) between each pair of particles
- ▶ Force is proportional to distance between the particles minus initial distance

# Particle simulation

## Model of a book

- ▶ Eight particles, located at the corners
- ▶ Each point has the same mass
- ▶ Interacting force (spring) between each pair of particles
- ▶ Force is proportional to distance between the particles minus initial distance



# Particle simulation

## Computations

- ▶ Calculate force on particle  $i$ :

$$F_i(t) = \sum_{j \neq i} \frac{k}{m} \left( \frac{\|x_i(0) - x_j(0)\|}{\|x_i(t) - x_j(t)\|} - 1 \right) (x_i(t) - x_j(t))$$

- ▶ Calculate velocity of particle  $i$ :

$$v_i(t + \Delta t) = v_i(t) + F_i(t) \Delta t$$

- ▶ Calculate position of particle  $i$ :

$$x_i(t + \Delta t) = x_i(t) + v_i(t) \Delta t$$

# Particle simulation

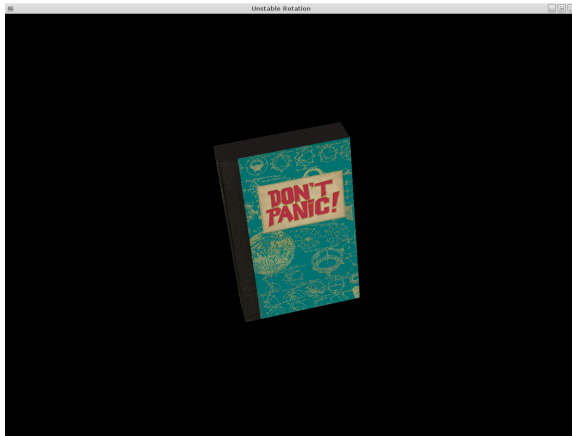
## Computations

Using Euler integration, for  $n$  particles, in each time step we need to compute

- ▶  $\frac{n(n-1)}{2}$  forces
- ▶  $n$  velocities
- ▶  $n$  positions

# Particle simulation

## Demonstration





# Particle simulation

## Observations

- ▶ Book behaves naturally
- ▶ Rotation is stable around two axis and unstable around the third
- ▶ After some time the simulation “explodes”

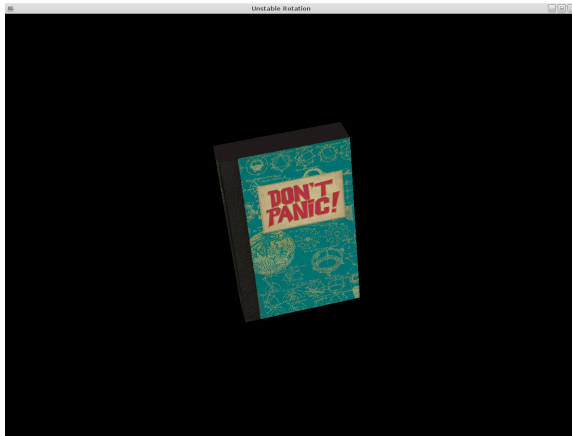
# Particle simulation

## Normalisation

- ▶ Euler integration introduces a small error in each time step
- ▶ Shape of book changes eventually
- ▶ Other techniques can reduce, but not eliminate the error
- ▶ Normalise the model using the conservation of energy

# Particle simulation

Demonstration with normalisation



# Particle simulation

## Normalisation

- ▶ Energy is conserved
- ▶ Angular momentum is transformed into vibration

# A different approach

- ▶ Different approach as introduced last time
- ▶ Calculate angular velocity to update the orientation

# Angular velocity based simulation

## Computations

- ▶ Angular velocity calculated from angular momentum and moment of inertia

$$\begin{aligned}I(t) &= R(t)^T I_{body} R(t) \\ I(t)^{-1} &= R(t) I_{body}^{-1} R(t)^T \\ \omega(t) &= I(t)^{-1} L\end{aligned}$$

- ▶ Change in orientation calculated from angular velocity

$$R(t + \Delta t) = R(t) + \omega^*(t) R(t) \Delta t$$

# Angular velocity based simulation

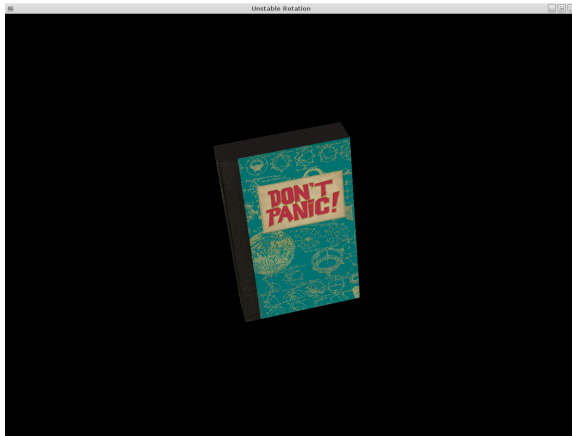
## Computations

Using Euler integration, in each time step we need to do

- ▶ 5 matrix multiplications
- ▶ 1 vector to tensor calculation
- ▶ 1 matrix inverse

# Angular velocity based simulation

## Demonstration





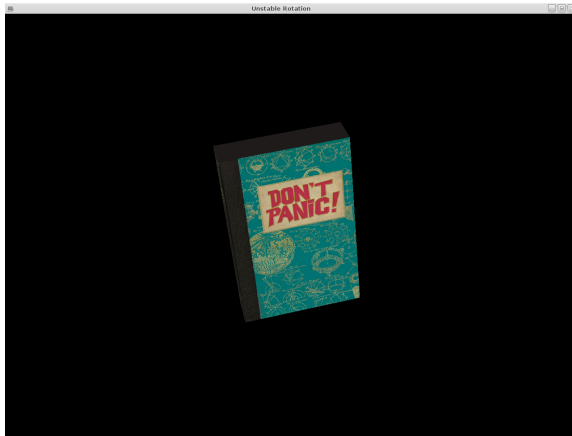
# Calculating the orientation

## Normalisation

- ▶ Euler integration introduces a small error in each time step
- ▶ Book changes shape
- ▶ Orientation matrix normalised by singular value decomposition (SVD)

# Angular velocity based simulation

Demonstration with normalisation

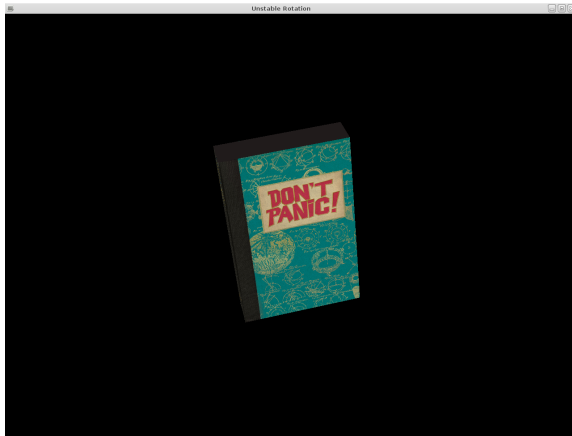


# Quaternions

- ▶ Orientation has 3 degrees of freedom, but a  $3 \times 3$  matrix has 9
- ▶ Quaternions have 4 degrees of freedom
- ▶ Again, normalisation to prevent shape from changing

# Angular velocity based simulation

Demonstration with quaternions



# Conclusion

- ▶ (Un)stable rotation of a book can be simulated with particles
- ▶ Simulation eventually produces “unnatural” results
- ▶ Other approach preserves angular momentum
- ▶ Simulation is more robust
- ▶ Only works when moment of inertia is known

# Questions?

?

# Questions

## about particle simulation

Q: What happens if you change the spring constant in calculating the forces?

A: For a high spring constant the shape and size of the book is contained better, but the book quickly goes to the vibrating state. For a low spring constant it takes longer to reach the vibrating state, but the shape and size is distorted more.

Q: Why don't you normalise the distance between particles?

A: If we do that, how will we calculate the forces?

# Questions

about particle simulation

Q: Can you improve this model?

A: There are several extensions one can think of. For example, damping may produce better results.



# Questions

## about quaternions

Q: What are quaternions?

A: They're an extension to complex numbers of the form  $q = a + b \cdot i + c \cdot j + d \cdot k$  that can be used to describe a rotation in 3d space. For a point at position  $p(0, x, y, z)$  the position after rotation is  $p'(0, x', y', z') = q \cdot p \cdot \bar{q}$ .

# Questions

about quaternions vs. matrix method

Q: Which method is faster?

A: In theory, quaternions produce better results and can be normalised less frequently. However, in this case the simulation is fast enough to do a normalisation in every time step.