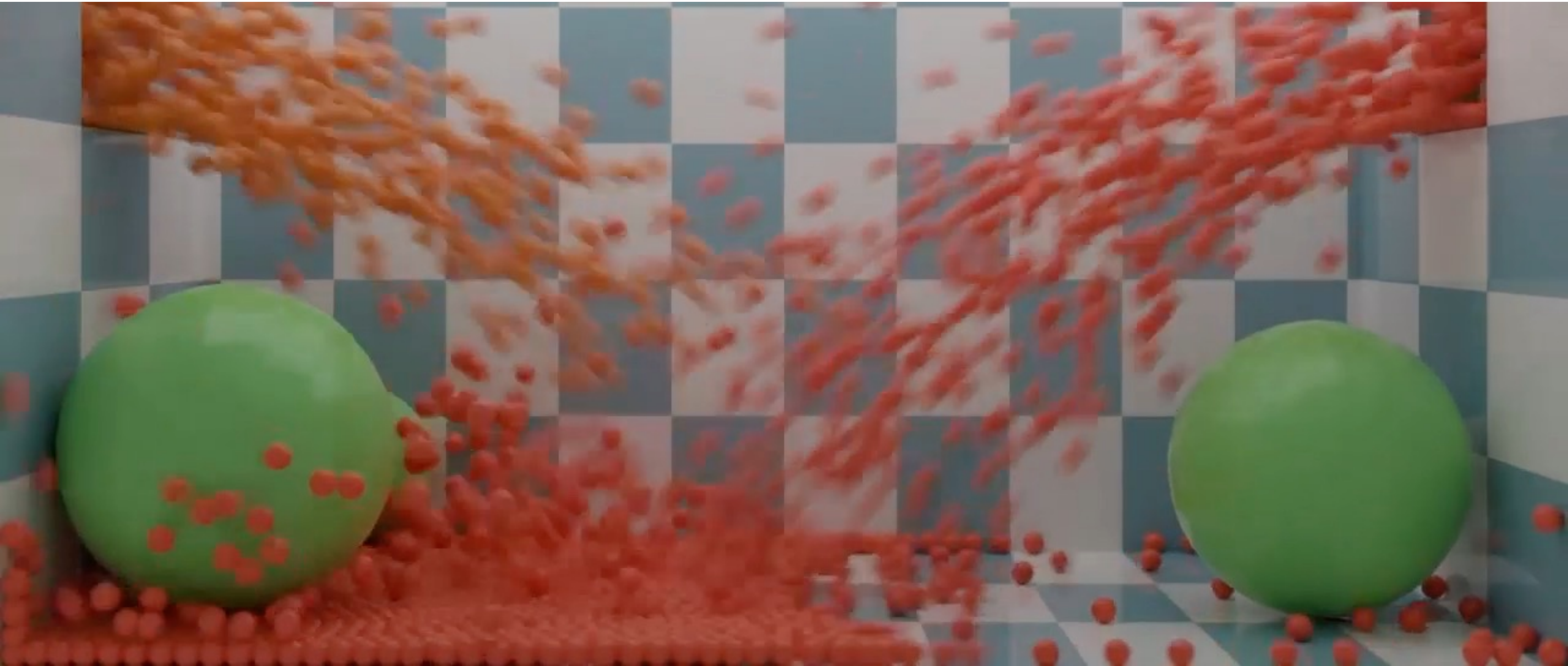
CMM - Assignment 5
Rigid Body Dynamics

Dongho Kang
kangd@ethz.ch

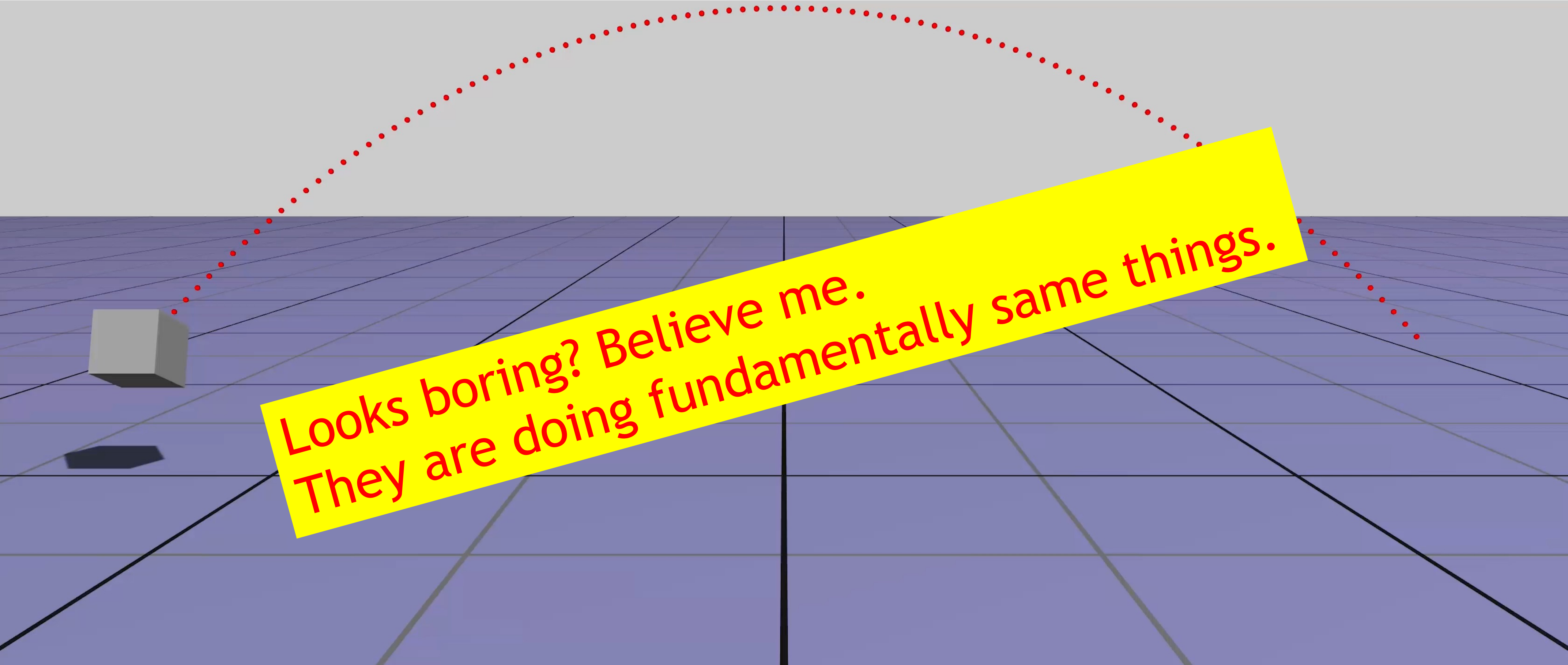
ETH zürich

CRL

We will make a rigid body simulator!



We will make a rigid body simulator!



What does a rigid body simulator do?

Newton-Euler Equation

$$\begin{bmatrix} \mathbf{F} \\ \boldsymbol{\tau} - \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} m\mathbb{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix}$$

Definition of velocity

$$\begin{bmatrix} \mathbf{v} \\ [\boldsymbol{\omega}]_\times \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{R}}\mathbf{R}^T \end{bmatrix}$$

"Skew-symmetric matrix"

$$[\boldsymbol{\omega}]_\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

Given $\mathbf{F}$ and $\boldsymbol{\tau}$, we computes $\mathbf{p}$, $\mathbf{R}$, $\mathbf{v}$, $\boldsymbol{\omega}$ at time t

What does a rigid body simulator do?

$$\begin{bmatrix} \mathbf{F} \\ \boldsymbol{\tau} - \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} m\dot{\mathbf{v}} \\ \mathbf{I}\dot{\boldsymbol{\omega}} \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{v} \\ [\boldsymbol{\omega}]_\times \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{R}}\mathbf{R}^T \end{bmatrix}$$

Looks super simple!
Now, let's discretize these equations.

What does a rigid body simulator do?

timestep

$$\mathbf{v}_{i+1} = \mathbf{v}_i + h\frac{\mathbf{F}}{m}$$

Linear velocity (of COM) in world frame

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h\mathbf{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}_i \times \mathbf{I}\boldsymbol{\omega}_i)$$

Angular velocity in world frame

Position in world frame

$$\mathbf{p}_{i+1} = \mathbf{p}_i + h\mathbf{v}_i$$

Orientation (rotational matrix)

$$\mathbf{R}_{i+1} = ?$$

Note. Explicit (forward) Euler integration

Note. $[\boldsymbol{\omega}]_\times = \dot{\mathbf{R}}\mathbf{R}^T$

How to update orientation?

- Option1: $\mathbf{R}_{i+1} = \mathbf{R}_i + h\dot{\mathbf{R}}_i = \mathbf{R}_i + h[\boldsymbol{\omega}_i]_\times \mathbf{R}_i = (\mathbb{I} + h[\boldsymbol{\omega}_i]_\times)\mathbf{R}_i$

Orthonormality of $\mathbf{R}$ is easily broken.

Angle-axis to rotational matrix conversion: Rodrigues' Rotation Formula

- Option2: $\mathbf{R}_{i+1} = Rotmat(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2})\mathbf{R}_i$

Okay, it works. But it's annoying to store 9 numbers for 3DOF…

Note. Explicit (forward) Euler integration

How to update orientation?

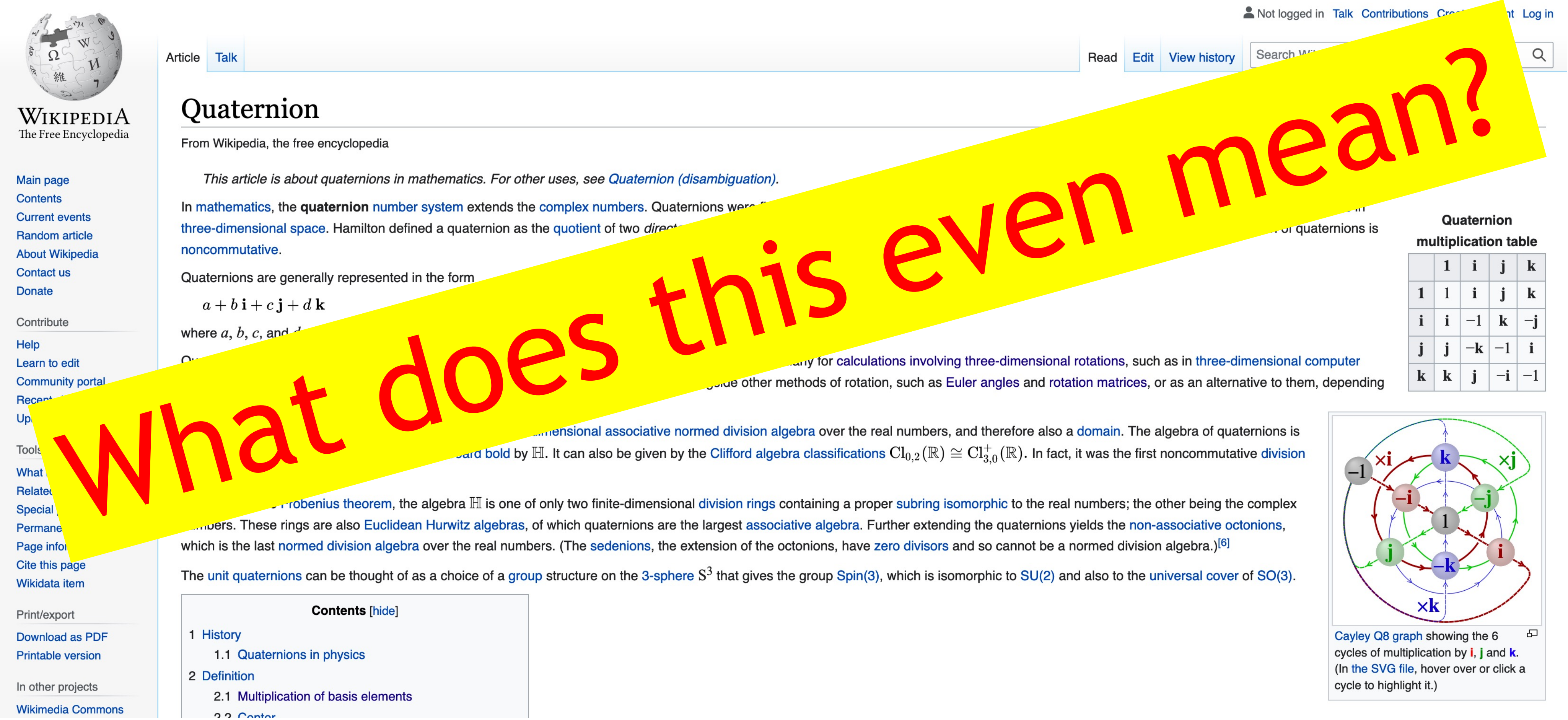
- Option1: $\mathbf{R}_{i+1} = \mathbf{R}_i + h\dot{\mathbf{R}}_i = \mathbf{R}_i + h[\boldsymbol{\omega}_i]_\times \mathbf{R}_i = (\mathbb{I} + h[$

- Opti $\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2})\mathbf{R}_i$

Let's use Quaternion instead!

Note. Explicit (forward) Euler integration

What is Quaternion?


Not logged in Talk Contributions Cre... Log in

Article Talk Read Edit View history Search Wi...

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Current events
Random article
About Wikipedia
Contact us
Donate

Contribute
Help
Learn to edit
Community portal
Recent...
Up...

Tools
What...
Relate...
Special...
Permane...
Page infor...
Cite this page
Wikidata item

Print/export
Download as PDF
Printable version

In other projects
Wikimedia Commons

Quaternion

From Wikipedia, the free encyclopedia

This article is about quaternions in mathematics. For other uses, see Quaternion (disambiguation).

In mathematics, the **quaternion** number system extends the complex numbers. Quaternions were ... three-dimensional space. Hamilton defined a quaternion as the quotient of two *direct*... ... of quaternions is noncommutative.

Quaternions are generally represented in the form

$$a + b\,\mathbf{i} + c\,\mathbf{j} + d\,\mathbf{k}$$

where a, b, c, and ...

... for calculations involving three-dimensional rotations, such as in three-dimensional computer ... other methods of rotation, such as Euler angles and rotation matrices, or as an alternative to them, depending

...mensional associative normed division algebra over the real numbers, and therefore also a domain. The algebra of quaternions is ...ard bold by $\mathbb{H}$. It can also be given by the Clifford algebra classifications $\mathrm{Cl}_{0,2}(\mathbb{R}) \cong \mathrm{Cl}^{+}_{3,0}(\mathbb{R})$. In fact, it was the first noncommutative division

...Frobenius theorem, the algebra $\mathbb{H}$ is one of only two finite-dimensional division rings containing a proper subring isomorphic to the real numbers; the other being the complex ...mbers. These rings are also Euclidean Hurwitz algebras, of which quaternions are the largest associative algebra. Further extending the quaternions yields the non-associative octonions, which is the last normed division algebra over the real numbers. (The sedenions, the extension of the octonions, have zero divisors and so cannot be a normed division algebra.)[6]

The unit quaternions can be thought of as a choice of a group structure on the 3-sphere S^3 that gives the group Spin(3), which is isomorphic to SU(2) and also to the universal cover of SO(3).

Contents [hide]

Quaternion multiplication table

	1	i	j	k
1	1	i	j	k
i	i	−1	k	−j
j	j	−k	−1	i
k	k	j	−i	−1

Cayley Q8 graph showing the 6 cycles of multiplication by i, j and k. (In the SVG file, hover over or click a cycle to highlight it.)

https://eater.net/quaternions

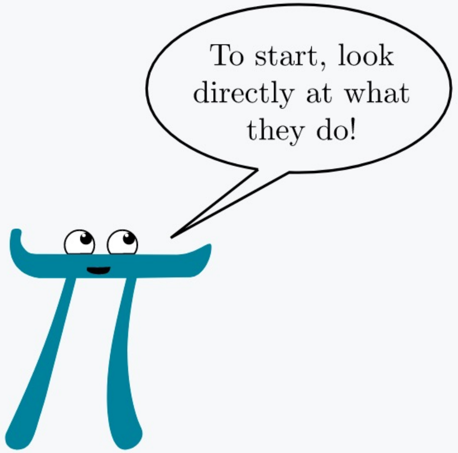
Visualizing quaternions

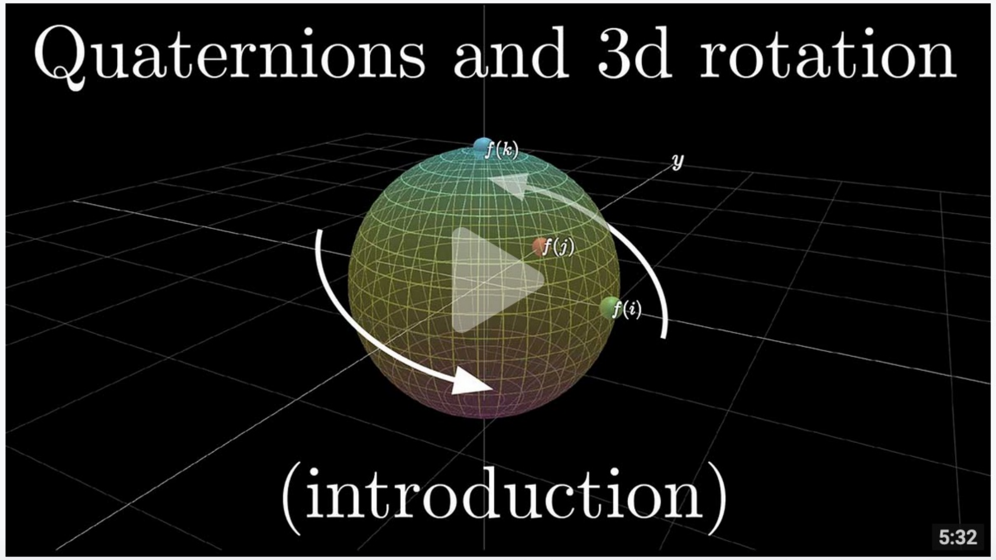
An explorable video series

Lessons by Grant Sanderson
Technology by Ben Eater

Quaternions and 3d rotation

One of the main practical uses of quaternions is in how they describe 3d-rotation. These first two modules will help you build an intuition for which quaternions correspond to which 3d rotations, although how exactly this works will, for the moment, remain a black box. Analogous to opening a car hood for the first time, all of the parts will be exposed to you, especially as you poke at it more, but understanding how it all fits together will come in due time. Here we are just looking at the "what", before the "how" and the "why".





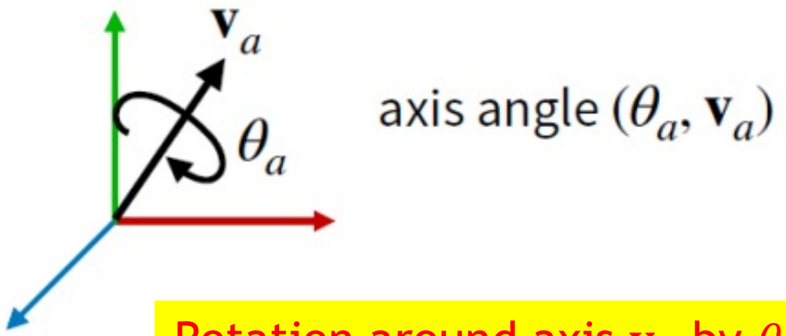
Watch a recording of this explorable video on YouTube.

How do these fit with the existing 3blue1brown YouTube videos?

In addition to this sequence of explorable videos, there are two videos on YouTube on the subject. Some of the material here is duplicated, but you may find a different take on it helpful:

- What are quaternions, and how do you visualize them? A story of four dimensions. Describes a way to visualize a hypersphere using stereographic projection and understand quaternion multiplication in

Unit Quaternion



Rotation around axis $\mathbf{v}_a$ by θ_a

- Unit Quaternion

$$\mathbf{q} = [w \quad x \quad y \quad z] = \left[\cos(\frac{\theta_a}{2}) \quad \sin(\frac{\theta_a}{2})\mathbf{v}_a\right]$$

Where $||\mathbf{q}|| = 1 \ldots$

How to update orientation?

Angle-axis to rotational matrix conversion: Rodrigues' Rotation Formula

- Rotation matrix: $\mathbf{R}_{i+1} = Rotmat\left(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right)\mathbf{R}_i$

- Quaternion: $\mathbf{q}_{i+1} = Rotquat\left(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right) \otimes \mathbf{q}_i$

$$Rotquat\left(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right) = [w \quad x \quad y \quad z] = \left[\cos\left(\frac{h\|\boldsymbol{\omega}_i\|_2}{2}\right) \quad \sin\left(\frac{h\|\boldsymbol{\omega}_i\|_2}{2}\right)\frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right]$$

Back to integration…

timestep

$$\mathbf{v}_{i+1} = \mathbf{v}_i + h\frac{\mathbf{F}}{m}$$

Linear velocity (of COM) in world frame

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h\mathbf{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}_i \times \mathbf{I}\boldsymbol{\omega}_i)$$

Angular velocity in world frame

Position in world frame

$$\mathbf{p}_{i+1} = \mathbf{p}_i + h\mathbf{v}_i$$

Orientation (rotational matrix)

$$\mathbf{R}_{i+1} = ?$$

Note. Explicit (forward) Euler integration

Back to integration…

timestep

$$\mathbf{v}_{i+1} = \mathbf{v}_i + h\frac{\mathbf{F}}{m}$$

Linear velocity (of COM) in world frame

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h\mathbf{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}_i \times \mathbf{I}\boldsymbol{\omega}_i)$$

Angular velocity in world frame

Position in world frame

$$\mathbf{p}_{i+1} = \mathbf{p}_i + h\mathbf{v}_i$$

Orientation (Quaternion)

$$\mathbf{q}_{i+1} = Rotquat\left(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right) \otimes \mathbf{q}_i$$

Note. Explicit (forward) Euler integration

That’s it. Time to code.

Exercises

- Baseline
  - Ex.1 Numerical Integration (40%)
  - Ex.2-1 Fixed Springs (10%)
  - Ex.2-2 Spring Attaching Rigid Bodies (10%)
  - Ex.3 Stable Simulation (20%)

- Advanced
  - Ex.4 Imp [illegible] ollision (20%)

We don’t discuss details today…

Ex.1 Numerical Integration

- We are going to use explicit (forward) Euler!

$$\mathbf{v}_{i+1} = \mathbf{v}_i + h\frac{\mathbf{F}}{m}$$

Linear velocity (of COM) in world frame

$$\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h\mathbf{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}_i \times \mathbf{I}\boldsymbol{\omega}_i)$$

Angular velocity in world frame

Position in world frame

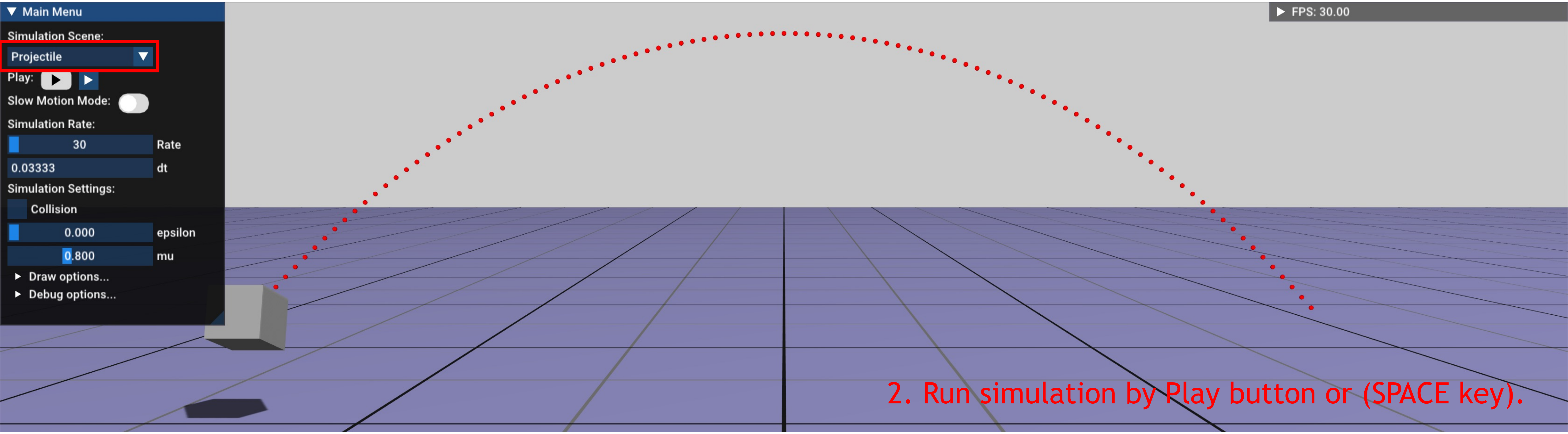
$$\mathbf{p}_{i+1} = \mathbf{p}_i + h\mathbf{v}_i$$

Orientation (Quaternion)

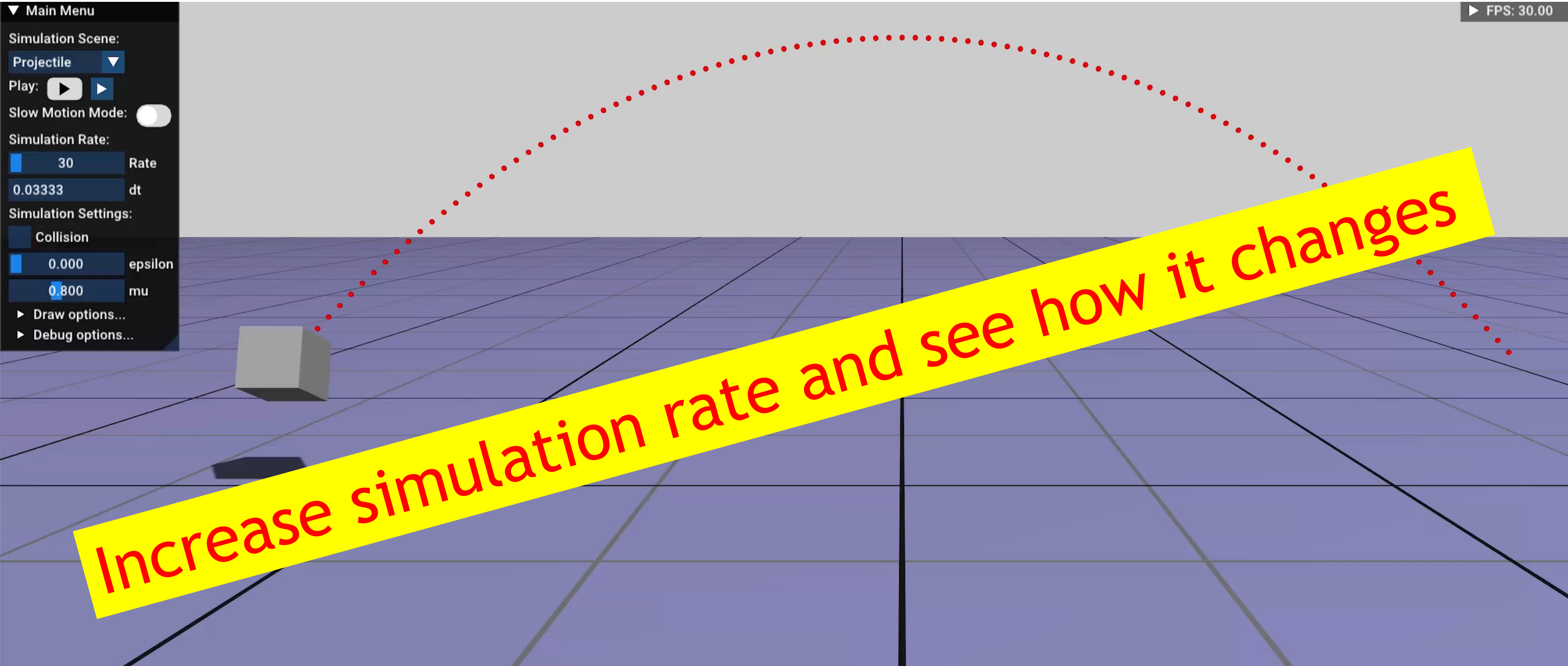
$$\mathbf{q}_{i+1} = Rotquat\left(h\|\boldsymbol{\omega}_i\|_2, \frac{\boldsymbol{\omega}_i}{\|\boldsymbol{\omega}_i\|_2}\right) \otimes \mathbf{q}_i$$

Ex.1 Numerical Integration

1. Select Simulation Scene: Projectile (you can reset scene by select "Projectile" from drop-down again)

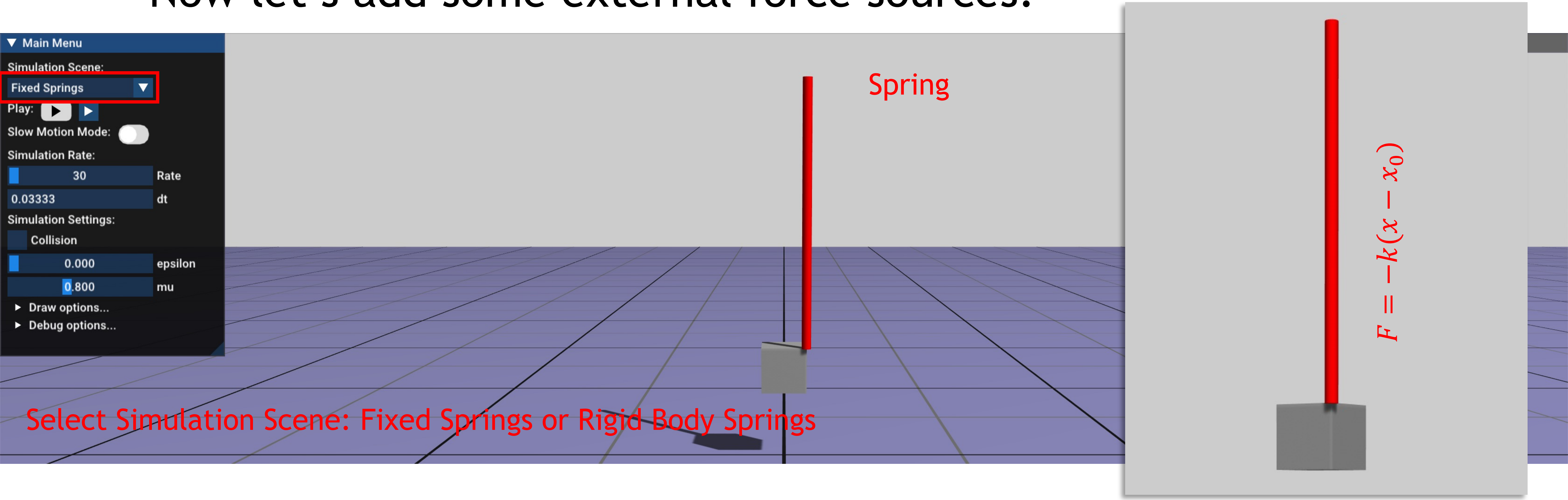


Ex.1 Successful Implementation

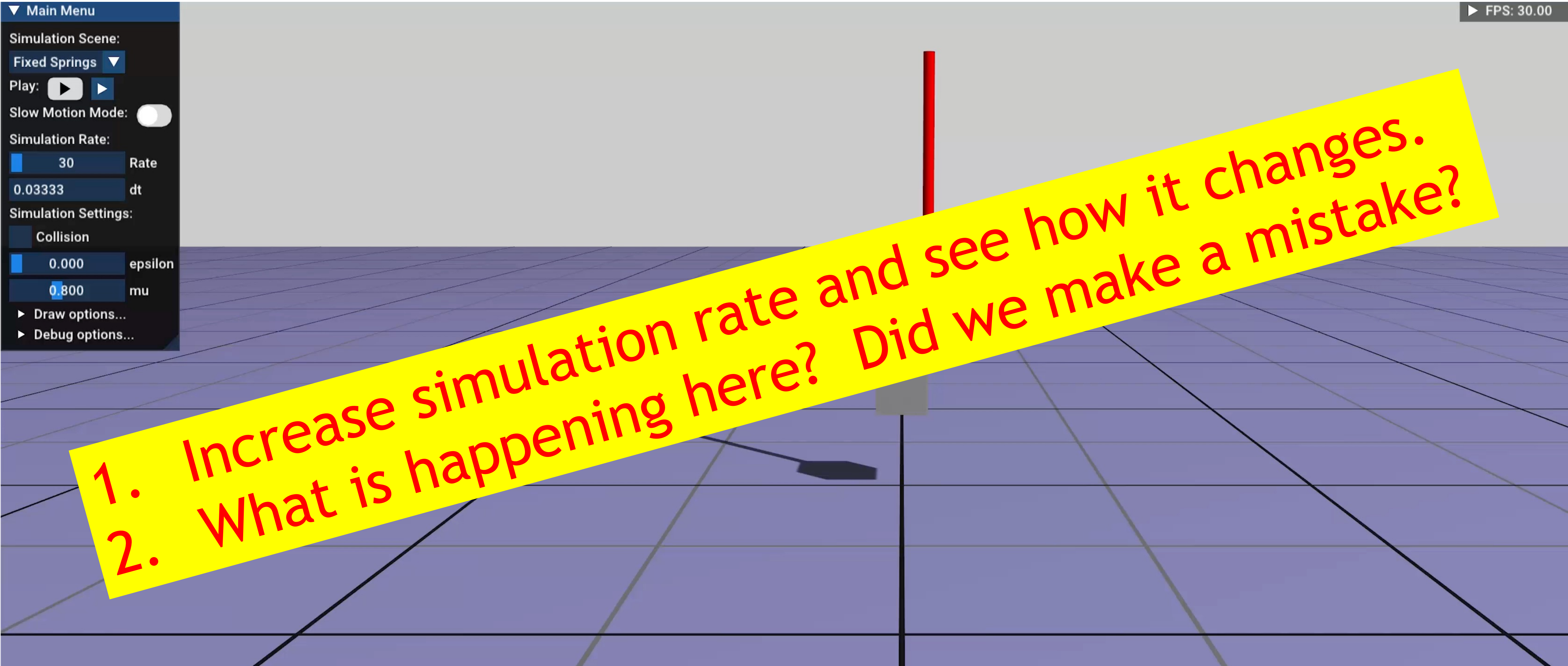


Ex.2 Springs

- Now let’s add some external force sources.



Ex.2 Successful(?) Implementation



1. Increase simulation rate and see how it changes.
2. What is happening here? Did we make a mistake?

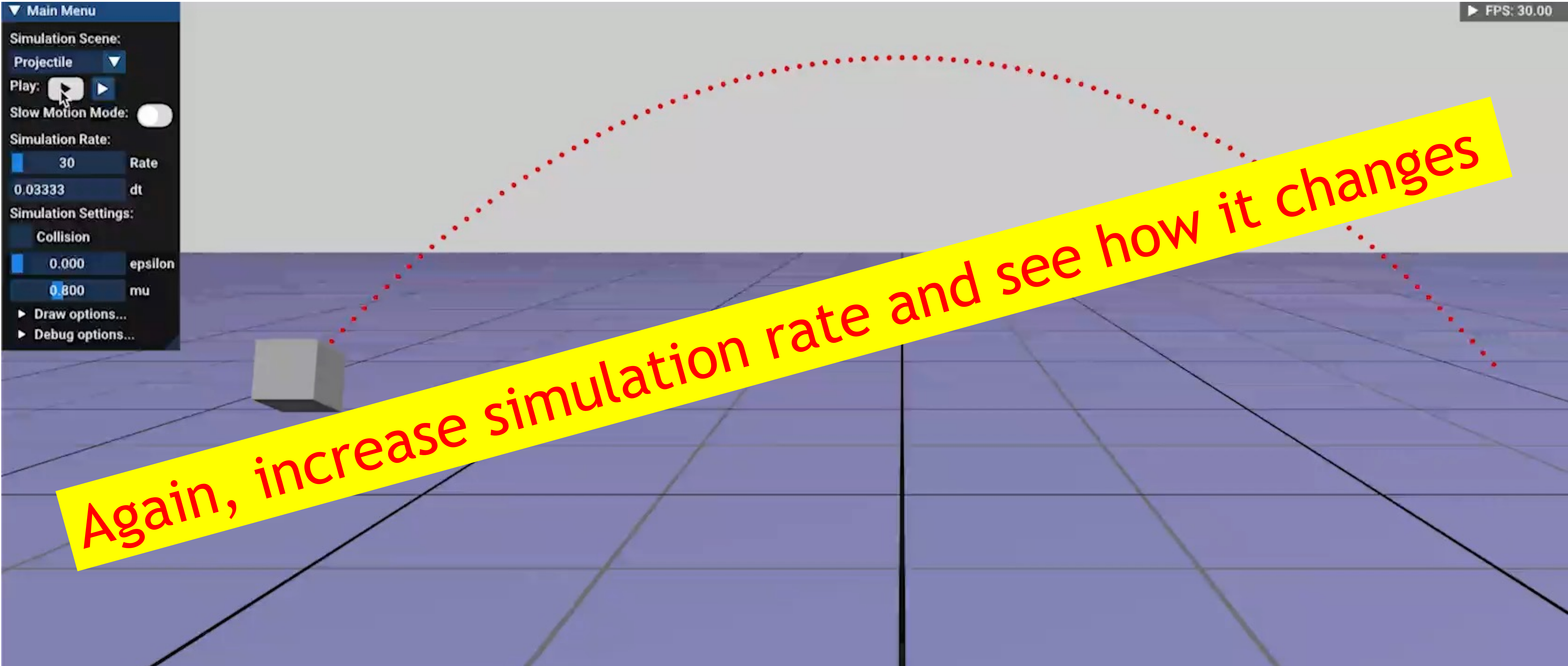
Ex.3 Stable Simulation

- Question: why our simulation is blown up?

If you have followed the lectures well,
you may already know what the problem is.

- Your task: make the simulation stable! It should be stable enough to run simulation with $dt = 1/30$ (rate = 30 Hz).

Ex.3 Successful Implementation



Note for Ex.4

- We will discuss more details in the next tutorial session.
- If you want to work on it beforehand, see README and comments.
- Also read ImpulseBasedCollisions on the course website.

Questions?

- Please actively use GitHub issue.
  - https://github.com/cmm-21/a5/issues

- Contact me if you have other questions.
  - kangd@ethz.ch