

3D video games

Collision Handling

Marco Tarini



1

Course Plan



lec. 1: Introduction ●

lec. 2: Mathematics for 3D Games ●●●●●●

lec. 3: Scene Graph ●

lec. 4: Game 3D Physics ●●●●+●●●

lec. 5: Game Particle Systems ▸

lec. 6: Game 3D Models ●●

lec. 7: Game Textures ▸●

lec. 9: Game Materials ●

lec. 8: Game 3D Animations ▸●●●

lec. 10: 3D Audio for 3D Games ●

lec. 11: Networking for 3D Games ●

lec. 12: Artificial Intelligence for 3D Games ●

lec. 13: Rendering Techniques for 3D Games ●

2

Collision Handling: the other half of the physic system



- Collision detection
 - find out when they occur
- Collision response
 - compute their effects

4

Collision Handling: a preliminary consideration



- Two types of objects in a game:
 - static
 - Never moves (speed = 0)
 - Part of the setting, background
 - Affects other objects, not affected by other objects
 - non-static
 - Can move around (for any reason)
- Two types of collisions:
 - one-way : a non-static object with a static object
 - two-ways : a non-static object with a non-static object

	Static	Movable
Static		One Way
Movable	One Way	Two Ways

6

Collision Handling: a preliminary consideration



By labelling every object as either static or movable, we reduce the needed computation considerably!

E.g., if 50% static, 50% movable then...

- 1/4 of the potential collisions cease to exist (*).
Of the rest:
- 2/3 are **one ways** (easier to handle)
- Only 1/3 are **two-ways**

(*) No collision handling for Static VS static:

That's not just an "optimization", but a feature:

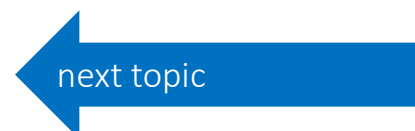
- Wall models can penetrate, to build a house (no collision!)
- Buildings can sink into the terrain (no collision!)
- Etc.

7

Collision Handling: two tasks



- **Collision detection**
 - find out when they occur
- **Collision response**
 - compute their effects



8

Collision response



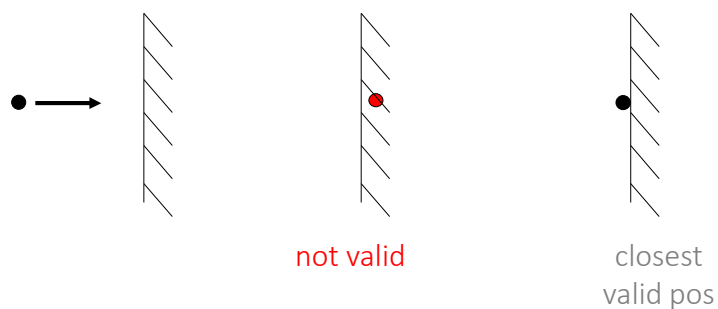
- Enforce **non-penetration**
 - objects must be placed in valid positions
 - (*when to*: **always**)
- **Impacts**
 - with impulses (bounces)
 - (*when to*: collision occurred now, but not in the pref frame)
- **Frictions** between the two objects
 - energy dissipation
 - (*when to*: from 2° consecutive step of collision)
- **Ad-hoc effects**
 - breaking objects, gameplay effects (HP loss?), etc (by scripts)
 - (*when to - if at all*: entirely gameplay dependent)

9

Enforcing non-penetration



- Invalid position?
 - strategy 1: revert to last valid pos (easy to do, not ideal)
 - strategy 2: project to closest valid pos (necessary, in PBD)



10

Enforcing non-penetration



- In PBD:
just another **positional constraint**
 - bonus: velocity updates (similar to inelastic impacts)
 - but we will need to explicitly compute impacts if we want a better control of the behavior (see later)
- **How to enforce** this constraint:
 - *two-ways* :
displace both of them,
minimizing the summed squared displacements $\times$ the mass
 - *one-way* :
only displace the one movable objects by the minimal amount
(equivalent to the above, when fixed object mass $\rightarrow \infty$)

Note: asymmetrical
constraint ($\geq$ not $=$)

A practical problem:
the existence of the
constraint it is **not** known
a-priori.

11

Contact frictions



- Apply it on prolonged contact
 - collision with an object that was colliding last frame too
- Affects component of velocity parallel to **contact plane**
- Can be implemented with:
(1) forces, or (2) velocity damping
- Forces:
 - Opposite to current velocity,
projected on contact plane (note: I need its normal)
 - Magnitude: proportional to the speed

12

Resolving the impacts



- so, it's the effect of an **impulse**
- And, for rigid body dynamics: also new angular velocity
- **Sudden** velocity change
 - resolve the impact = determine the new velocities $\vec{v}_{new}$
 - **equivalently**, determine the impulses $\vec{i} = m (\vec{v}_{new} - \vec{v}_{old})$
 - we'll write formulas for whichever is easier to write
- All impacts preserve total **momentum** $m \vec{v}$
 - Always, no matter what
- To resolve the impact, we need further assumptions, different for each type of the impact:
 - **elastic**
 - **inelastic**

a vector
(italian: «quantità di moto»)

13

Different type of impacts



(completely)
elastic
impact

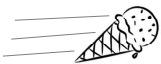
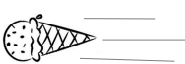


(completely)
inelastic
impact

14

“Bounciness” (or impact elasticity)



		“Bounciness” = 1.0
...		
		“Bounciness” = 0.5
...		
		“Bounciness” = 0.0

15

“Bounciness” (or impact elasticity)

[notes]



- Elastic impact: no energy lost
- How is energy lost, in reality? (examples)
 - objects get damaged, heat is produced, sounds are emitted
- “Bounciness”:
 - a (made-up) property of physical objects in games
 - It models the behavior of the object under impacts, as a mix between the two “pure” behaviors above
 - Associated by designers to all virtual objects in the game
- Note: that’s not how real stuff works!
 - not even for the two extremes
 - it’s an approximation (especially for mixed bounciness)
 - Remember: we are just aiming at *plausibility*

16

What about this impact?



“Bounciness” = ???

- Practical solution:
adopt some formula between the bounciness values associated to the two objects
 - For example: **avg, min, max**
 - It’s a choice of the game engine
 - (can be hard-wired in the physics engine, or exposed to the users)

17

The assumptions for the two “pure” types of impact

	Assumptions		
Elastic	After the impact, the total energy is the same as before	The impulse is in the direction of the impact normal	...and the total momentum is the same as before
Inelastic	After the impact, the two bodies share the same velocity		...and the total momentum is the same as before

Remember that the impulse (force x time) is the (instantaneous) change of momentum!
So, this is a way to say that the total impulse is zero

$i_A + i_B = 0$
that is,
 $i_A = -i_B$
aka the 3rd law of dynamics.

18

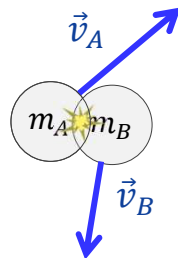
The assumptions for the two “pure” types of impact

- (completely) **elastic** impacts:
 - preservation of total **kinetic energy** $\frac{1}{2} m \|\vec{v}\|^2$ a scalar
 - impulse direction = the **normal of impact point**
- (completely) **inelastic** impacts:
 - after the impact, the two bodies have the same velocity
 - (as if the impact momentarily glued them together)
(they will still move apart in subsequent frames)
- mixed cases:
 - solve for both cases, interpolate resulting velocities
 - the weight of the interpolation = the “**bounciness**”

19

(completely) inelastic impact

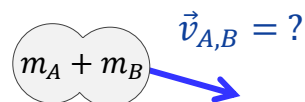
BEFORE:



Momentum:

$$m_A \vec{v}_A + m_B \vec{v}_B$$

AFTER:



the only unknown, so ...

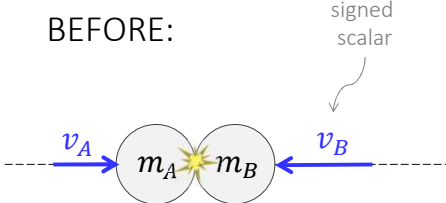
Momentum:

$$(m_A + m_B) \vec{v}_{A,B}$$

20

(completely) elastic impact: 1D case 

BEFORE:



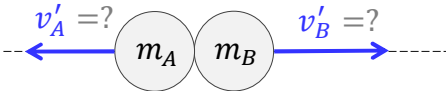
momentum:

$$m_A v_A + m_B v_B$$

energy:

$$\frac{1}{2} m_A v_A^2 + \frac{1}{2} m_B v_B^2$$

AFTER:



momentum:

$$m_A v'_A + m_B v'_B$$

energy:

$$\frac{1}{2} m_A v'^2_A + \frac{1}{2} m_B v'^2_B$$

21

(completely) elastic impact: 1D case 

new velocities are defined by the impulses:

$$v'_A = v_A + \frac{i_A}{m_A} \quad v'_B = v_B + \frac{i_B}{m_B}$$

signed scalars

momentum conservation: $i_B = -i_A$ (a.k.a. the 3rd law of dynamics)

energy conservation:

$$\begin{aligned} \cancel{\frac{1}{2} m_A v_A^2} + \cancel{\frac{1}{2} m_B v_B^2} &= \cancel{\frac{1}{2} m_A v'^2_A} + \cancel{\frac{1}{2} m_B v'^2_B} \\ \Rightarrow m_A v_A^2 + m_B v_B^2 &= m_A \left(v_A + \frac{i_A}{m_A} \right)^2 + m_B \left(v_B + \frac{i_B}{m_B} \right)^2 \\ \Rightarrow \cancel{m_A v_A^2} + \cancel{m_B v_B^2} &= \cancel{m_A v_A^2} + \frac{i_A^2}{m_A} + 2 v_A i_A + \cancel{m_B v_B^2} + \frac{i_B^2}{m_B} + 2 v_B i_B \\ \Rightarrow 0 &= \frac{i_A^2}{m_A} + 2 v_A i_A + \frac{i_B^2}{m_B} + 2 v_B i_B \end{aligned}$$

25

(completely) elastic impact: 1D case



substituting: $\frac{i_A^2}{m_A} + 2 v_A i_A + \frac{i_A^2}{m_B} - 2 v_B i_A = 0$

$$i_A^2 \frac{m_A + m_B}{m_A m_B} + i_A 2(v_A - v_B) = 0$$

$$i_A \left(i_A \frac{m_A + m_B}{m_A m_B} + 2(v_A - v_B) \right) = 0$$

solution 1

$$i_A = i_B = 0$$

before the impact

solution 2

$$i_A = -i_B = \frac{2 m_A m_B}{m_A + m_B} (v_B - v_A)$$

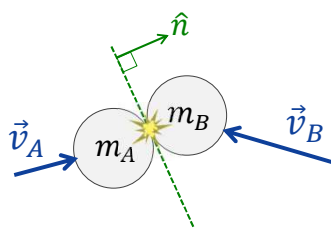
after the impact

26

(completely) elastic impact: 3D case



BEFORE:



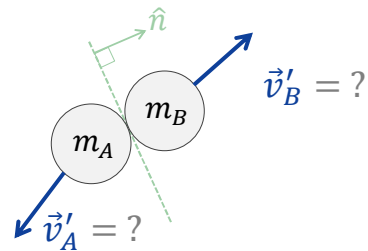
momentum:

$$m_A \vec{v}_A + m_B \vec{v}_B$$

energy:

$$\frac{1}{2} m_A \|\vec{v}_A\|^2 + \frac{1}{2} m_B \|\vec{v}_B\|^2$$

AFTER:



momentum:

$$m_A \vec{v}'_A + m_B \vec{v}'_B$$

energy:

$$\frac{1}{2} m_A \|\vec{v}'_A\|^2 + \frac{1}{2} m_B \|\vec{v}'_B\|^2$$

27

(completely) elastic impact: 3D case



- Additional assumption:

- $\exists$ **impact plane**, with normal $\hat{n}$
 - o, in 2D: impact line
 - impulses must be orthogonal to this plane $\vec{i}_{A,B} = i_{A,B} \hat{n}$

so, we need
this data!

vector
impulses

scalar
impulses,
pos. or neg.
(the unknowns)

- To solve the impact

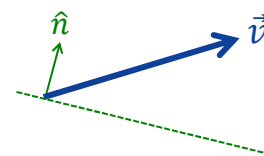
- find **scalar velocities** $v_{A,B}$ as the component of **vector velocities** $\vec{v}_{A,B}$ along $\hat{n}$: $v_{A,B} = \vec{v}_{A,B} \cdot \hat{n}$
 - find **scalar impulses** $i_{A,B}$ (use the 1D case)
 - find **vector impulses** $\vec{i}_{A,B} = i_{A,B} \hat{n}$
 - apply them to **vector velocities**

28

Remember this geometric subproblem?

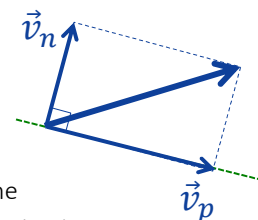


- Given: velocity vector $\vec{v}$
impact plane normal $\hat{n}$,
split $\vec{v}$ in the vector sum
 $\vec{v} = \vec{v}_n + \vec{v}_p$ with
 - $\vec{v}_n$ orthogonal to the plane (= parallel to $\hat{n}$)
 - $\vec{v}_p$ parallel to the plane (= orthogonal to $\hat{n}$)



- Solution in 3 steps:

- $s_n \leftarrow \vec{v} \cdot \hat{n}$ with s_n a scalar (the signed "speed")
- $\vec{v}_n \leftarrow s_n \hat{n}$
- $\vec{v}_p \leftarrow \vec{v} - \vec{v}_n$ (or: $\vec{v}_p \leftarrow \vec{v} \times \vec{v}_n \times \vec{v}_n$)



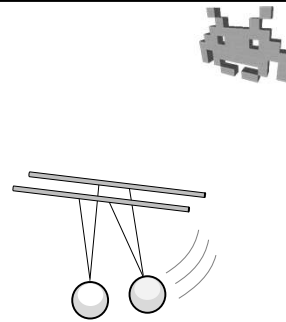
- Needed because:

- only $\vec{v}_n$ is affected by **elastic impacts** with the plane
- only $\vec{v}_p$ is affected by **frictions** with the plane (e.g.: drag)
- s_n is used to *solve* elastic impacts (use 1D case)

29

Special case: (exercise: verify!) Equal masses

- Completely **elastic** case (1D):
 - the two velocities just *swap*
- Completely **elastic** case (3D):
 - The two velocity components orthogonal to the impact plane *swap*
- Completely **inelastic** case (3D):
 - the new velocity of both particles is the (vector) average of their pre-impact velocities

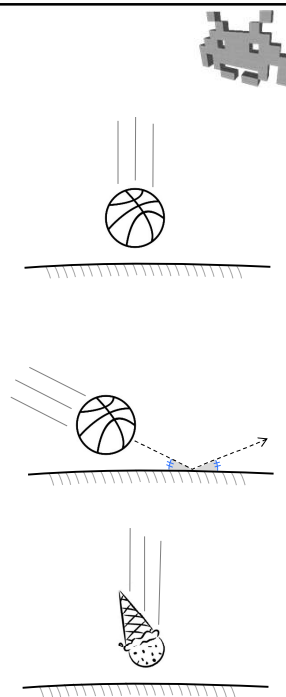


31

Special case: (exercise: verify!) one-way collision (A is static)

$$m_A \rightarrow \infty \\ \& \\ \vec{v}_A = 0$$

- Completely **elastic** case (1D):
 - v_b just *flips*
- Completely **elastic** case (3D):
 - The component of v_B orthogonal to impact plane just *flips*
- Completely **inelastic** case (3D):
 - B stops dead ($\vec{v}'_B = 0$)



32

Notes on

impacts between *rigid bodies*

that is,
considering
angular velocities too



- We only have seen impacts between *particles*
 - i.e., we disregarded angular velocities
 - when **rigid bodies** are implicitly implemented as particles + distance constraints, this is all we need to do!
 - Effect of elastic / inelastic impacts on angular velocities will be an (approximated) **emerging behavior** 👍
- Impacts between *explicit rigid bodies* require to *explicitly* compute the two post-impact angular velocities too
- Different math, stemming from the same principles:
 - **Angular** momentum: it is *always* preserved, no matter what
 - *Anelastic impact*: post-impact **angular velocities** must also match
 - *Elastic impact*: kinetic **rotational energy** must also be preserved
 - *Bounciness* $\in [0,1]$: interpolate **angular velocities** of the above

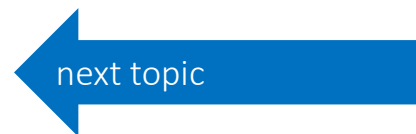
33

Collision Handling: two tasks



- **Collision detection**
 - find out when they occur
 - if so, produce **collision data** for the response
- **Collision response**
 - compute their effects

next topic



34