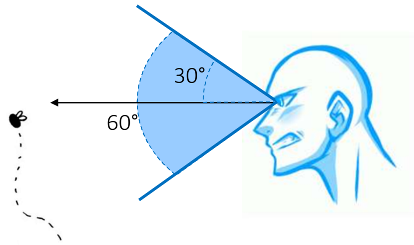
3D videogames

Points, Vectors, Versors: mini tasks and exercises

Marco Tarini



69

Points, Vectors, Versors: mini problems

- The following are examples of spatial problems that you may commonly need to solve when developing a 3D game
  - They are easy to solve using point/vector/versor algebra
  - Many game engines libraries implement functions for many/all of them
- General schema for finding a solution:
  - identify input and output (and their types)
  - maybe draw a schema to help yourself
  - write the equations driven by the drawing, (using your spatial understanding of the operations): e.g. what must be orthogonal to what? Which length is equal to what? Which angle must be acute / obtuse? etc
  - identify the unknowns
  - manipulate the equations according to the rules to extract the unknowns
  - everything is ready to code it! (equations directly become code)

For some of them, the solution will be given in full here. In other, only a trace of the solution is given

72

Points, Vectors, Versors: mini task and exercises

- Try to write pseudo-code that solves the proposed problems, using
  - An existing library (GLM, Unity, Unreal) GLSL..
  - Your own hand-made library for points/vectors/versor

73

Point to point distance (trivial)

"When the player in position p is closer than k to a powerup in pos q, then the powerup is collected"

- Data: p, q points, k scalar
- Test: $\|\mathrm{p}-\mathrm{q}\| < k$
- Optimized as: $\|\mathrm{p}-\mathrm{q}\|^2 < k^2$ ⟵why is this an optimization?
- Pseudo-code example:

```
vec3 p,q;
scalar k;
if ( dot(p-q,p-q) < k*k ) then /*collect*/
```

74

Ray-Plane intersection Ver0

"I shoot a laser from $\mathbf{p}$ in direction $\hat{\mathrm{d}}$ toward a plane which contains points q and has normal $\hat{\mathrm{n}}$.
Which point $\mathbf{q}$ do I hit?"

- Trace:
  - Define $\mathbf{q}$ as a point on the laser (see Ray-Sphere inters.)
  - Define $\mathbf{q}$ as a point on the plane
    (hint: the vector connecting it to any other point on the plane is orthogonal to $\vec{\mathrm{n}}$)
  - Combine the two equations into one
  - Extract the only incognita

75

Projection of a point on a sphere

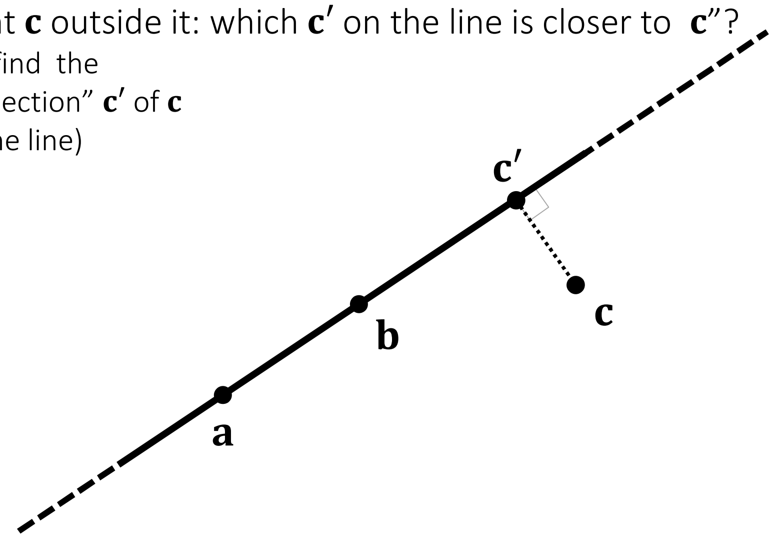
"Given a 3D sphere and a point $\mathbf{Q}$, find the point $\mathbf{Q_2}$ on the sphere closest to $\mathbf{Q}$"

- Input: sphere center $\mathbf{C}$ and radius r
- Output: $\mathbf{Q_2}$

76

Projection of a point on a line

"Consider a line passing through points $\mathbf{a}$ and $\mathbf{b}$ and a point $\mathbf{c}$ outside it: which $\mathbf{c}'$ on the line is closer to $\mathbf{c}$"?
(i.e. find the "projection" $\mathbf{c}'$ of $\mathbf{c}$ on the line)



77

Sphere-sphere intersection (trivial)

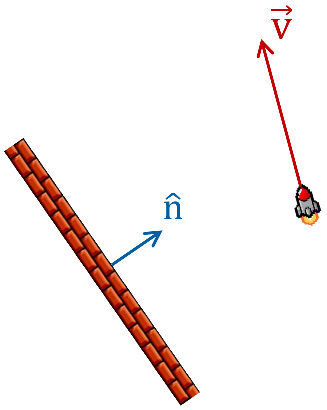
"Given two spheres with center in $\mathbf{c}_0$ and $\mathbf{c}_1$ and radii r_0 and r_1: do they intersect? Do they touch?"

- Hint:
  - remember that working with *squared norms* is more efficient (and more accurate) than working with vector norms

78

The missile and the wall (trivial)

"A missile is moving at constant velocity $\vec{\mathbf{v}}$ (meter per sec), in the general proximity of a large (infinite) wall with normal $\hat{\mathbf{n}}$.
After some time t (sec), how much closer to (or farther from) the wall is it?"



79

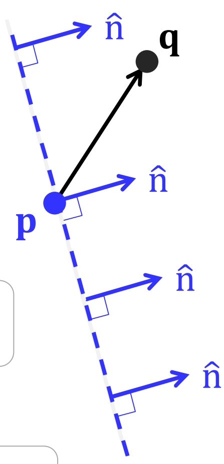
Plane VS Point test

- Input: a point $\mathbf{q}$ and a plane given by:
  - its normal: $\hat{\mathrm{n}}$
  - a point on it at random: $\mathbf{p}$
- *Q*: on which side of the plane is $\mathbf{q}$?
- *A*: it's the sign of

$$\hat{\mathrm{n}} \cdot (\mathbf{q} - \mathbf{p}) =$$
$$\hat{\mathrm{n}} \cdot \mathbf{q} - \vec{n} \cdot \mathbf{p} =$$
$$\hat{\mathrm{n}} \cdot \mathbf{q} + k =$$
$$(n_x, n_y, n_z, k) \cdot (q_x, q_y, q_z, 1)$$

$k = -\vec{n} \cdot \mathbf{p}$
(minus distance of plane from origin)

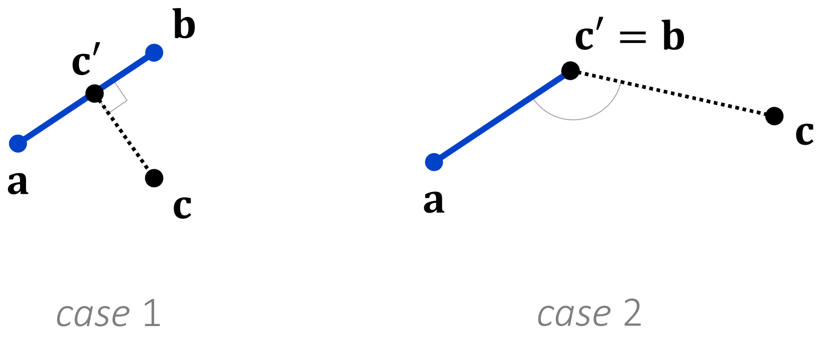
a 4D vector representing the plane:
a more convenient representation for a plane



80

Projection of a point on a segment

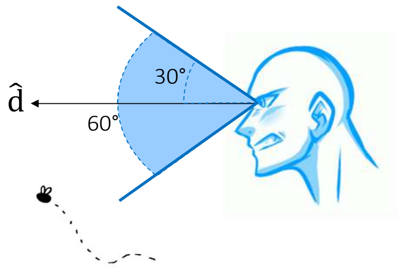
"Which $\mathbf{c}'$ point on a segment connecting point $\mathbf{a}$ and $\mathbf{b}$ is closer to a third point $\mathbf{c}$"?



81

Vision cones

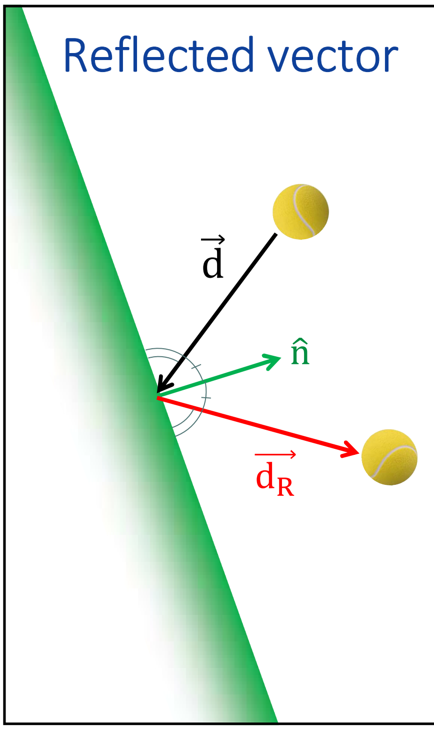
"A guard has eyes in position $\mathbf{q}$ and looks in direction $\hat{\mathbf{d}}$.
Do they spot a fly in position $\mathbf{p}$, if their cone of vision is 60° wide?"
(that is, the FOV = 60°)



- Hypotheses: no occlusions
- Trace:
  - For angles α, β in 0..180°: $\alpha < \beta \leftrightarrow \cos(\alpha) > \cos(\beta)$
  - Find cosine of angle between view direction and the vector connecting $\mathbf{q}$ to $\mathbf{p}$
  - Determine if this cosine is > $\cos(60°/2)$

82

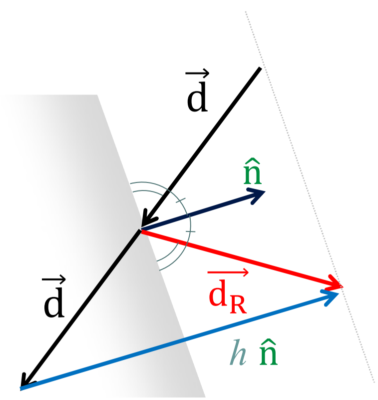
Reflected vector

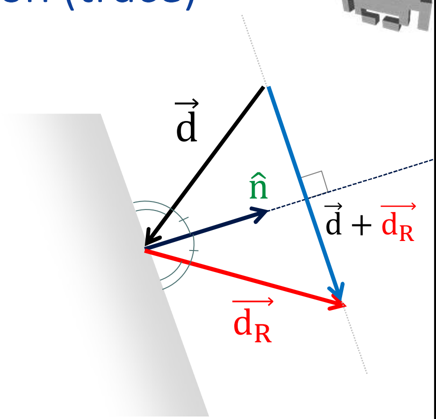


- A tennis ball (or a photon, or a laser beam, etc) traveling with velocity $\vec{d}$ hits a surface with normal $\hat{n}$ and bounces back.
- Determine «reflected» velocity $\overrightarrow{d_R}$
- Physics tells us:
  - $\overrightarrow{d_R}$ is coplanar to $\vec{d}$ and $\hat{n}$
  - angles with plane are equal

83

Reflected vector: a solution (trace)





Observe

(1) Vector $\overrightarrow{d_R}$ can be found as the sum of $\vec{d}$ and some vector oriented in direction $\hat{n}$ (that is, and $h\ \hat{n}$ for a given unknown h)

(2) Vector $\vec{d} + \overrightarrow{d_R}$ is orthogonal to $\hat{n}$

84

Ray-sphere intersection

"I shoot a laser from p to direction $\hat{\mathrm{d}}$. Do I hit a sphere in position q of radius r ? Where?"

- Data: p, q points, r scalar, $\hat{\mathrm{d}}$ versor
- Trace:
  - Hit-point is s on laser ray:
    $\mathrm{s} = \mathrm{p} + k\,\hat{\mathrm{d}}$, for some unknown scalar $k \geq 0$
  - Hit-point is s on sphere:
    $\|\mathrm{q} - \mathrm{s}\| = r \quad \leftrightarrow \quad (\mathrm{q} - \mathrm{s}) \cdot (\mathrm{q} - \mathrm{s}) = r^2$
  - Combine the two equations (substitute s in second), solve for k (it's a 2nd degree equation), test that k exists and that it is >0)

85

Sphere-line intersection

"A basketball (a sphere with radius r) falling out of the window is passing close to an (inifinite) clothesline passing from position p and having direction $\hat{\mathrm{d}}$. The basketball is centered in position q. Is the basketball hitting the wire?"

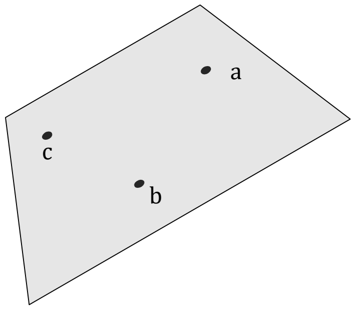
- Trace:
  - Find the position s on the line closest to q :
    $\mathrm{s} = \mathrm{p} + k\,\hat{\mathrm{d}}$, for and unknown scalar k (any sign)
  - Hint: the vector from s to q must be orthogonal to $\hat{\mathrm{d}}$!
  - Check whether s is inside the sphere
  - (note: this problem is different, and simpler, that the ray-to-sphere intersection)

86

Problem: surface normal (trivial)

"I have three points on $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ on a plane: find the normal $\hat{\mathbf{n}}$ of this plane (a versor)"

- Trace:
  find any two
  different vectors
  on the plane
  …
- Question: what determines the direction of $\hat{\mathbf{n}}$?



87

Problem: sphere normal (trivial)

I have a point $\mathbf{a}$ and
a sphere S centered in point $\mathbf{c}$ with radius r :
find…

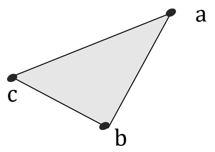
- whether $\mathbf{a}$ is inside, outside, or on the S
- either way,
  point $\mathbf{a}'$ that is the point on S closer to $\mathbf{a}$
  (that is, the projection of $\mathbf{a}$ on S)
- the normal of S in $\mathbf{a}'$

88

Problem: triangle area (trivial)

“I have three points on $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ in space.
Find the area of the triangle connecting them”

- Hint:
  it’s half the area
  of a parallelogram



89

Orthonormal base completion (trivial)

“I have only two axes $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ of an orthonormal bases, how do I find the third vector $\hat{\mathbf{z}}$?”

- Data: $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$ versors
- Hypotheses: $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are already orthogonal

Less trivial variant:

- Now $\hat{\mathbf{y}}$ is not exactly orthogonal to $\hat{\mathbf{x}}$, but I want to change it the least to make it orthogonal
  ($\hat{\mathbf{x}}$ is to be kept constant)
  (see next problem)

90

Vector orthogonalization

"Find a versor $\hat{\mathrm{u}}'$ that is ortogonal to a given $\hat{\mathrm{n}}$ such that it is as similar as possible to a given versor $\hat{\mathrm{u}}$"

Solution: $\hat{\mathrm{u}}' = \hat{\mathrm{n}} \times \hat{\mathrm{u}} \times \hat{\mathrm{n}}$, then renormalize it.

in this case, order of operations is irrelevant:
(in the general case, cross is not associative)

Coding examples, in different languages:

GLSL
```
vec3 n,u;
u = cross( cross( n , u ) , n );
u = normalize( u );
```

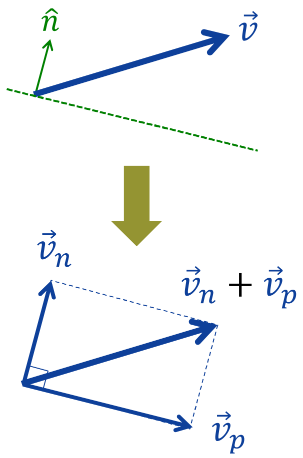
C++, with UE
```
FVector n,u;
u = FVector::CrossProduct( FVector::CrossProduct(n,u),n );
u.Normalize();
```

C#, with Unity
```
Vector3 n,u;
u = Vector3.Cross( Vector3.Cross( n , u ) , n );
u = u.normalized;
```

91

Decompose a vector into components

- Given a vector $\vec{v}$ and a 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 $\hat{n}$)



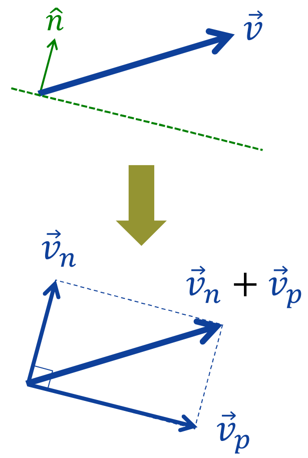
- A solution in 3 steps:
  (1) $k \leftarrow \vec{v} \cdot \hat{n}$
  (2) $\vec{v}_n \leftarrow k \ \hat{n}$
  (3) $\vec{v}_p \leftarrow \vec{v} - \vec{v}_n$

k is a (signed) scalar: the extension of $\vec{v}$ along dir $\hat{n}$

92

Decompose a vector into components

- Given a vector $\vec{v}$ and a 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 $\hat{n}$)



dot product

(lil notation abuse) vector scaling. Or, viceversa!

A cuter solution:

$\vec{v}_n = \hat{n} \cdot \vec{v} \cdot \hat{n}$

$\vec{v}_p = \hat{n} \times \vec{v} \times \hat{n}$

Like in prev exercise (see why the length is just right?)

93

Line-Line "intersection"

"Given two 3D lines A and B, find the two points on both lines that are as close as possible to each other"

(they are the same point, if the lines intersect!)

- Input: a point on line "A" $\mathrm{p_A}$ and its direction $\hat{\mathrm{d}}_\mathrm{A}$
  a point on line "B" $\mathrm{p_B}$ and its direction $\hat{\mathrm{d}}_\mathrm{B}$
- Output: two points $\mathrm{q_A}$ and $\mathrm{q_B}$
- Trace:
  the segment connecting $\mathrm{q_A}$ to $\mathrm{q_B}$
  must be orthogonal to *both* $\hat{\mathrm{d}}_\mathrm{A}$ and $\hat{\mathrm{d}}_\mathrm{B}$!
  Impose these conditions as two dot product.
  Solve for the two incognitas…

94

Ray-Plane intersection

"I shoot a laser from $\mathbf{p}$ in direction $\hat{\mathrm{d}}$ toward a plane which contains points $\mathbf{a}$ $\mathbf{b}$ $\mathbf{c}$. Which point $\mathbf{q}$ do I hit?"

- Hypotheses: $\mathbf{a}$ $\mathbf{b}$ $\mathbf{c}$ are not colinear (not on a line)
- Trace:
  - Find vector $\vec{\mathrm{n}}$ orthogonal to plane, use cross product (question for later: are magnitude and verse important?)
  - Define $\mathbf{q}$ as a point on the laser (see Ray-Sphere inters.)
  - Define $\mathbf{q}$ as a point on the plane (hint: the vector connecting it to any other point on the plane is orthogonal to $\vec{\mathrm{n}}$)
  - Combine the two equations into one
  - Extract the only incognita

95

Plane-plane intersection

"Given two 3D planes, find the 3D line they share"

- Input: a point on plane "A" $\mathbf{p}_\mathrm{A}$ and its normal $\hat{\mathrm{n}}_\mathrm{A}$
  a point on plane "B" $\mathbf{p}_\mathrm{B}$ and its normal $\hat{\mathrm{n}}_\mathrm{B}$
- Output:
  a point on the line $\mathbf{q}$ and the line direction $\hat{\mathrm{d}}$
- Trace:
  1. find $\hat{\mathrm{d}}$ first
  2. find a direction $\hat{\mathrm{e}}$ *on* plane "A" orthogonal to $\hat{\mathrm{d}}$
  3. shoot a ray from $\mathbf{p}_\mathrm{A}$ in direction $\hat{\mathrm{e}}$
  4. $\mathbf{q}$ is where that ray hits plane "B"

96

Point projection on a plane

"Find the point $\mathbf{q}'$ on a plane (passing through point $\mathbf{p}$ with normal $\hat{\mathrm{n}}$) closest to a given point $\mathbf{q}$"

- Input: point $\mathbf{q}$ and
  point on plane $\mathbf{p}$ and its normal $\hat{\mathrm{n}}$
- Output: point $\mathbf{q}'$

97

Plane-sphere intersection

"Given a ball and a wall (a plane), find if they intersect, and, if they do, the point of the ball deeper inside the wall"

- Input: center $\mathbf{c}$ and radius r
  a point on plane $\mathbf{p}$ and its normal $\hat{\mathrm{n}}$
- Output:
  a Boolean b, and a point $\mathbf{q}$ if b = *true*

98

Shooting a walking target (with a finite speed bullet) 1/2

"I shoot a bullet from p with velocity $\vec{\mathrm{v}}$. At which time the bullet will be the closest to a target currently in position q and moving with velocity $\vec{\mathrm{w}}$? Where will bullet and target be, at that time?"

(useful, e.g., for a sniper AI "*leading*" a target)

- Data: p, q points, $\vec{\mathrm{v}}$ and $\vec{\mathrm{w}}$ vectors
- Hypothesis: nothing accelerates (everything keeps moving at a constant speed)

99

Shooting a walking target (with a finite speed bullet) 2/2

Trace

- Position of bullet at time t : $\mathrm{p} + t\,\vec{\mathrm{v}}$
- Position of target at time t : $\mathrm{q} + t\,\vec{\mathrm{w}}$
- Squared distance between the two at time t :

$$\| (\mathrm{p} + t\,\vec{\mathrm{v}}) - (\mathrm{q} + t\,\vec{\mathrm{w}}) \|^2$$
$$=$$
$$\| (\mathrm{p} - \mathrm{q}) + t\,(\vec{\mathrm{v}} - \vec{\mathrm{w}}) \|^2$$

- Work on formulas (remember that $\|\vec{\mathrm{v}}\|^2 = \vec{\mathrm{v}} \cdot \vec{\mathrm{v}}$)
  find derivative on *t*,
  equate derivative to 0, extract *t*

100