JVector Algorithms

Tobias Binna
tbinna@hsr.ch
8. Dezember 2008
Hochschule für Technik Rapperswil

Line3D

`getDistance(Point)`

Given

Line $g = p + \lambda u$

Point q

Wanted

magnitude $|r|$

Solution

$$r \perp u$$
$$r = q - (p + \lambda u) = q - p - \lambda u$$

$$r \cdot u = 0 :$$
$$(q - p - \lambda u) \cdot u = 0$$
$$(q - p) \cdot u - \lambda u \cdot u = 0$$
$$(q - p) \cdot u = \lambda u \cdot u$$
$$\frac{(q - p) \cdot u}{u \cdot u} = \lambda$$

$$|r| = \underline{\underline{\left| q - p - \frac{(q - p) \cdot u}{u \cdot u} \cdot u \right|}}$$

`getIntersection(Line)`

Given

Line $g_1 = p + \lambda u$

Line $g_2 = q + \mu v$

Wanted

Punkt s

Solution

Generate a 3x3 matrix with the parametric equations as follows:

$$g_1 : \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} + \lambda \cdot \begin{pmatrix} r_{xa} \\ r_{ya} \\ r_{za} \end{pmatrix}$$
$$g_2 : \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} + \mu \cdot \begin{pmatrix} r_{xb} \\ r_{yb} \\ r_{zb} \end{pmatrix}$$

$$\Rightarrow \left| \begin{array}{l} a_x + \lambda \cdot r_{xa} = b_x + \mu \cdot r_{xb} \\ a_y + \lambda \cdot r_{ya} = b_y + \mu \cdot r_{yb} \\ a_z + \lambda \cdot r_{za} = b_z + \mu \cdot r_{zb} \end{array} \right| \Rightarrow \left| \begin{array}{l} \lambda \cdot r_{xa} - \mu \cdot r_{xb} = b_x - a_x \\ \lambda \cdot r_{ya} - \mu \cdot r_{yb} = b_y - a_y \\ \lambda \cdot r_{za} - \mu \cdot r_{zb} = b_z - a_z \end{array} \right| \Rightarrow \underline{\begin{bmatrix} r_{xa} & -r_{xb} & b_x - a_x \\ r_{ya} & -r_{yb} & b_y - a_y \\ r_{za} & -r_{zb} & b_z - a_z \end{bmatrix}}$$

If you bring the matrix in a reduced form with Gaussian elimination you should be able to read immediately from the values if the liens are linear independent (all zero in the third line). If so you can read the factors for λ and μ at the position (1, 3) for λ and. (2, 3) for μ. Put either λ or μ into one of the equations and you get the intersection.

No solution

If the lines are skew the following statement becomes true:

$(p - q) \cdot (u \times v) \neq 0$

If the lines are parallel the following statement becomes true:

$u \times v = 0$

If the lines are parallel and have in addition to that have also the same initial vectors (the vector from the origin on to a point on the line) the lines are collinear.

Plane3D

`getDistance(Point)`

Given

Ebene $E = Ax + By + Cz + D = 0$

Punkt p

Wanted

Betrag d

Solution

The distance to the plane is calculated with the help of the “Hessischen Normalform” of the plane which leads us to the following formula:

$$d = \left| \frac{A \cdot p_x + B \cdot p_y + C \cdot p_z + D}{\sqrt{A^2 + B^2 + C^2}} \right|$$

`getIntersection(Line)`

Given

Plane $E = Ax + By + Cz + D = 0$

Line $g = p + \lambda u$

Wanted

Point s

Solution

Write the plane equation a bit diffrent

$$Ax + By + Cz + D = 0$$

$$Ax + By + Cz = -D$$

$$n = \begin{pmatrix} A \\ B \\ C \end{pmatrix}, q = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$q \cdot n = -D$$

Replace point q by the line g

$$(p + \lambda u) \cdot n = -D$$

Solve for λ

$$\lambda = \frac{-D - p \cdot n}{u \cdot n}$$

- if the numerator equals 0, the line lies parallel to the plane
  $n \cdot u = 0$
- if the numerator and denominator are equal to 0, the line lies in the plane
  $n \cdot u = 0 \wedge -D - p \cdot n = 0$

If neither numerator nor denominator are 0 > divide and put λ in to the equation and you get the point of intersection.

$$\underline{\underline{s = p + \frac{-D - p \cdot n}{u \cdot n} \cdot u}}$$

`getIntersection(Plane)`

Given

Plane $E_1 = A_1 x_1 + B_1 y_1 + C_1 z_1 + D_1 = 0$

Plane $E_2 = A_2 x_2 + B_2 y_2 + C_2 z_2 + D_2 = 0$

Wanted

Gerade $g = p + \lambda u$

Solution

Create a matrix with the coefficients in the two plane equations as follows:

$$\begin{bmatrix} A_1 & B_1 & C_1 & D_1 \\ A_2 & B_2 & C_2 & D_2 \end{bmatrix}$$

If the two equations are linear independent the identity shows up in the first two columns after the Gaussian elimination process. From the third column you can read the x and y components of the direction vector and form the fourth column you get the x and y components of a point on the line.

$$\begin{bmatrix} 1 & 0 & -r_x & -p_x \\ 0 & 1 & -r_y & -p_y \end{bmatrix}$$

From this you can construct right away the line of intersection of the two planes as follows:

$$\underline{\underline{\begin{pmatrix} p_x \\ p_y \\ 0 \end{pmatrix} + t \cdot \begin{pmatrix} r_x \\ r_y \\ 1 \end{pmatrix}}}$$

No solution

The system has no solution in case the magnitude of the cross product of the two normal vectors to their plane equals 0.

If in addition to that the constant values D are the same the two planes lie on exactly the same position.

Overview on the methods of the JVector components

Point

Method	Returntype	Description
`getDistance(Point)`	`double`	Returns the distance between this and the given point.
`getDistance(Line)`	`double`	Returns the shortest distance between this point and the given line.
`getDistance(Plane)`	`double`	Returns the shortest distance between this point and the given plane.

Line

Method	Returntype	Description
`getAngle(Line)`	`double`	Returns the angle in radians between this and the given line.
`getAngle(Plane)`	`double`	Returns the angle in radians between this line and the given plane.
`getDistance(Point)`	`double`	Returns the shortest distance between this line and the given point.
`getIntersection(Line)`	`Point`	Returns the point of intersection between this line and the given line.
`getIntersection(Plane)`	`Point`	Returns the point of intersection between this line and the given plane.

Plane

Method	Returntype	Description
`getAngle(Line)`	`double`	Returns the angle in radians between this plane and the given line.
`getAngle(Plane)`	`double`	Returns the angle in radians between this plane and the given plane.
`getDistance(Point)`	`double`	Returns the shortest distance between this plane and the given point.
`getIntersection(Line)`	`Point`	Returns the point of intersection between this plane and the given line.
`getIntersection(Plane)`	`Line`	Returns the line of intersection between this and the given plane.