# <https://en.wikipedia.org/wiki/Line%E2%80%93plane_intersection>

# See wikipedia page for compact matrix formulation.

# [https://upload.wikimedia.org/wikipedia/commons/thumb/e/e3/Line_plane.svg/300px-Line_plane.svg.png](https://en.wikipedia.org/wiki/File:Line_plane.svg)

<https://math.stackexchange.com/questions/83990/line-and-plane-intersection-in-3d>

How would one calculate the intersection of a line and a plane in 3D ?

Given for example are 4 points which form a plane (x1,y1,z1)...(x4,y4,z4) and 2 different points which form a line (x5,y5,z5) and (x6,y6,z6). How is it possible to know where the line intersect with the plain when this info is given. I thought to calculate the equation of the plain and line. Then eq of the line = eq of the plane. to find the point where they intersect. But I don't know how the construct the equation of a line in 3D given the 2 points. And I'm not sure that if I equate the 2 equations that it will give the intersection point. If someone would please be so kind to fill the blanks in of my knowledge or suggest another solution. I would be very grateful.

A common way of representing a plane P is in point-normal form, n⃗ ⋅(X−Y)=0, where n⃗ n→ is the plane normal and both X and Y are points that lie in the plane. This can be rewritten into constant-normal form by distributing the dot product and rearranging terms to obtain: n⃗ ⋅X=d, where d=n⃗ ⋅Y which is equal to the distance from the origin when n⃗ n→ is unit-length. Below is a simple data structure that you might use to represent a plane, and the signature of a constructor that will compute the plane from three points in R3. Implementation is left as an exercise to the reader ;).

struct Plane {

Vector3 n; // normal

float d; // distance from origin

Plane(); // default constructor

Plane(Vector3 a, Vector3 b, Vector3 c); // plane from 3 points

Vector3 intersectLine(Vector3 a, Vector3 b); // we'll get to this later

};

Given two points, A and B, a line can be represented parametrically by adding to one point the vector formed by the two points, scaled by a parameter t. In symbols, L(t)=A+t(B−A). Using your intuition, we insert this equation (whose output is a point), into X in the constant-normal plane representation: n⃗ ⋅[A+t(B−A)]=d

We want to know how many copies of (B−A) we need to add to A to get to a point that lies within the plane, in other words we want to solve for t. Doing some fancy algebra, we obtain: t=(d−n→⋅A)/n→⋅(B−A). We can (finally) stick this expression for t back into the equation for our line to obtain: I=A+(d−(n⃗ ⋅A))/n⃗ ⋅(B−A)\*(B−A).

Armed with this equation, we can now implement a nice function that will tell what we want to know:

Vector3 Plane::intersectLine(Vector3 a, Vector3 b) {

Vector3 ba = b-a;

float nDotA = Vector3::dotProduct(n, a);

float nDotBA = Vector3::dotProduct(n, ba);

return a + (((d - nDotA)/nDotBA) \* ba);

}

If you plan to be doing a lot of this sort of geometric computing it's worthwhile to pick up [Christer Ericson's Real-time Collision Detection](http://realtimecollisiondetection.net/books/rtcd/), which is an excellent reference source for this sort of thing. Alternatively, you could snag some already-constructed classes from something like [OGRE3D](http://www.ogre3d.org/), if you're not particularly interested in creating your own.

Intersection of a line and a plane.

Consider the plane P = 2x + y − 4z = 4.

a) Find all points of intersection of P with the line x = t, y = 2 + 3t, z = t.

b) Find all points of intersection of P with the line x = 1 + t, y = 4 + 2t, z = t.

c) Find all points of intersection of P with the line x = t, y = 4 + 2t, z = t.

Answer: a) To find the intersection we substitute the formulas for x, y and z into the equation for P and solve for t. 2(t) + (2 + 3t) − 4(t) = 4 ⇔ t = 2.

Now use t = 2 to find the point of intersection: (x, y, z) = (2, 8, 2).

b) Substituting gives 2(1 + t) + (4 + 2t) − 4(t) = 4 ⇔ 6 = 4. ⇔ no values of t satisfy this equation. There are no points of intersection.

c) Substituting gives 2(t) + (4 + 2t) − 4(t) = 4 ⇔ 4 = 4. ⇔ all values of t satisfy this equation. The line is contained in the plane, i.e., all points of the line are in its intersection with the plane. Here are cartoon sketches of each part of this problem. P (a) line intersects the plane in (b) line is parallel to the plane (c) line is in the plane a point

<https://math.stackexchange.com/questions/1365152/3d-line-in-a-3d-plane-find-the-intersection-of-the-two>

P is the plane containing the three points (−3,4,−2), (1,4,0), and (3,2,−1). ℓ is the line containing the two points (2,4,−3) and (−1,−1,−9). What is the intersection of plane P and line ℓ?

Find an equation of the plane, and one of the line.

* **Plane:**

V1:=(4,0,2) and V2:=(−2,2,1) are two linearly independent vectors in P, so N:=V1×V2=(−4,−8,8) is a normal vector to P. Let n=(1,2,−2) (just for simplicity). Then n is also a normal to P, and P's equation is given by: n⋅(x−1,y−4,z)=0

Therefore: P: x+2y−2z=9

* **Line:**

U:=(3,5,6) is a direction vector of ℓ, and ℓ passes through A:=(2,4,−3).

We know:ℓ: x=xA+xUt, y=yA+yUt, z=zA+zUt

Thus:ℓ: {x=3t+2, y=5t+4, z=6t−3}

Let M=(a,b,c) be the intersection point between P and ℓ. M satisfies the equations of each: P and ℓ, so that for some t: a=3t+2, b=5t+4 and c=6t−3, and a+2b−2c=9. Hence, 3t+2+10t+8−12t+6=9, so t=−7.

Therefore, M=(−19,−31,−45).

<https://math.stackexchange.com/questions/47594/plane-intersecting-line-segment?rq=1>

I have a plane which is represented as a 3d point p⃗ with a normal n^. I also have a line segment specified by two points v⃗1, v⃗2. I want to get the intersection point (if any). Here's what I have:

distv1=n^⋅(v1→−p⃗ )

distv2=n^⋅(v2→−p⃗ )

If distv1⋅distv2≤0 then I know there is an intersection because the distances are signed and the product of numbers with opposite signs is negative. Zero is included to cover the case when an endpoint is exactly on the plane. Then:

x^= (v⃗2 − v⃗1 )/∣∣ v⃗2− v⃗1∣

cosθ=n^⋅x^

Now I test cosθcos⁡θ. If zero then I choose one (of both) the endpoints as the intersection point. If non-zero I proceed to find the intersection point v⃗:

v⃗ = v⃗2 − x(distv2/cosθ)

I'd like to know if my solution can be reworked to be more "elegant" without the last check of cosθ being non-zero to avoid a divide-by-zero?

The formula is correct (your additional comments what to do when cosθ=0cos⁡θ=0 are not) and you can't avoid a possible division by zero because the division by zero is the right result.

If cosθ vanishes, it means that n^ - the normal direction of the plane - is perpendicular to v⃗ 2−v⃗ 1, the direction of the line. In other words, the direction of the line v⃗ 2−v⃗ 1 is parallel to the plane.

If it is parallel, the line either belongs to the plane, in which case there is a whole line of intersections and the division appropriately yields a 0/0 indeterminate form; or the line is outside the plane in which case the division by zero is of the form 1/0 and there is no intersection (or the intersections is infinitely far, if you wish).

You can't get finite unique coordinates of the intersection if cosθ=0 because there aren't any.

<http://paulbourke.net/geometry/pointlineplane/>

# Equation of a plane

Written by [Paul Bourke](http://paulbourke.net/geometry/)  
March 1989

|  |  |
| --- | --- |
| The standard equation of a plane in 3 space is  Ax + By + Cz + D = 0  The normal to the plane is the vector (A,B,C). | http://paulbourke.net/geometry/pointlineplane/planeeq1.gif |

Given three points in space (x1,y1,z1), (x2,y2,z2), (x3,y3,z3) the equation of the plane through these points is given by the following determinants.

http://paulbourke.net/geometry/pointlineplane/planeeq2.gif

Expanding the above gives  
A = y1 (z2 - z3) + y2 (z3 - z1) + y3 (z1 - z2)   
B = z1 (x2 - x3) + z2 (x3 - x1) + z3 (x1 - x2)   
C = x1 (y2 - y3) + x2 (y3 - y1) + x3 (y1 - y2)   
- D = x1 (y2 z3 - y3 z2) + x2 (y3 z1 - y1 z3) + x3 (y1 z2 - y2 z1)

Note that if the points are collinear then the normal (A,B,C) as calculated above will be (0,0,0).

The sign of s = Ax + By + Cz + D determines which side the point (x,y,z) lies with respect to the plane. If s > 0 then the point lies on the same side as the normal (A,B,C). If s < 0 then it lies on the opposite side, if s = 0 then the point (x,y,z) lies on the plane.

**Alternatively**

If vector **N** is the normal to the plane then all points **p** on the plane satisfy the following

**N** . **p** = k

where . is the dot product between the two vectors.

ie: **a** . **b** = (ax,ay,az) . (bx,by,bz) = ax bx + ay by + az bz

Given any point **a** on the plane

**N** . (**p** - **a**) = 0

# Minimum Distance between a Point and a Plane

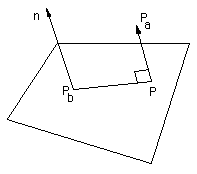
Written by [Paul Bourke](http://paulbourke.net/geometry/)  
March 1996

Let **Pa** = (xa, ya, za) be the point in question.

A plane can be defined by its normal **n** = (A, B, C) and any point on the plane **Pb** = (xb, yb, zb)

Any point **P** = (x,y,z) lies on the plane if it satisfies the following

A x + B y + C z + D = 0



The minimum distance between Pa and the plane is given by the absolute value of

|  |  |
| --- | --- |
| (A xa + B ya + C za + D) / sqrt(A2 + B2 + C2) | . . . 1 |

To derive this result consider the projection of the line (**Pa** - **Pb**) onto the normal of the plane **n**, that is just ||**Pa** - **Pb**|| cos(theta), where theta is the angle between (**Pa** - **Pb**) and the normal **n**. This projection is the minimum distance of **Pa** to the plane.

This can be written in terms of the dot product as

minimum distance = (**Pa** - **Pb**) dot **n** / ||**n**||

That is

|  |  |
| --- | --- |
| minimum distance = (A (xa - xb) + B (ya - yb) + C (za - zb)) / sqrt(A2 + B2 + C2) | . . . 2 |

Since point (xb, yb, zb) is a point on the plane

|  |  |
| --- | --- |
| A xb + B yb + C zb + D = 0 | . . . 3 |

Substituting equation 3 into equation 2 gives the result shown in equation 1.

# The shortest line between two lines in 3D

Written by [Paul Bourke](http://paulbourke.net/geometry/)  
April 1998

Two lines in 3 dimensions generally don't intersect at a point, they may be parallel (no intersections) or they may be coincident (infinite intersections) but most often only their projection onto a plane intersect.. When they don't exactly intersect at a point they can be connected by a line segment, the shortest line segment is unique and is often considered to be their intersection in 3D.

|  |  |
| --- | --- |
| The following will show how to compute this shortest line segment that joins two lines in 3D, it will as a bi-product identify parallel lines. In what follows a line will be defined by two points lying on it, a point on line "a" defined by points P1 and P2 has an equation.  Pa = P1 + mua (P2 - P1)  similarly a point on a second line "b" defined by points P4and P4 will be written as  Pb = P3 + mub (P4 - P3)  The values of mua and mub range from negative to positive infinity. The line segments between P1 P2 and P3 P4 have their corresponding mu between 0 and 1. | http://paulbourke.net/geometry/pointlineplane/lineline3d1.gif |

There are two approaches to finding the shortest line segment between lines "a" and "b". The first is to write down the length of the line segment joining the two lines and then find the minimum. That is, minimise the following

|| Pb - Pa ||2

Substituting the equations of the lines gives

|| P1 - P3 + mua (P2 - P1) - mub (P4 - P3) ||2

The above can then be expanded out in the (x,y,z) components. There are conditions to be met at the minimum, the derivative with respect to mua and mub must be zero. Note: it is easy to convince oneself that the above function only has one minima and no other minima or maxima. These two equations can then be solved for mua and mub, the actual intersection points found by substituting the values of mu into the original equations of the line.

An alternative approach but one that gives the exact same equations is to realise that the shortest line segment between the two lines will be perpendicular to the two lines. This allows us to write two equations for the dot product as

(Pa - Pb) dot (P2 - P1) = 0

(Pa - Pb) dot (P4 - P3) = 0

Expanding these given the equation of the lines

( P1 - P3 + mua (P2 - P1) - mub (P4 - P3) ) dot (P2 - P1) = 0

( P1 - P3 + mua (P2 - P1) - mub (P4 - P3) ) dot (P4 - P3) = 0

Expanding these in terms of the coordinates (x,y,z) is a nightmare but the result is as follows

d1321 + mua d2121 - mub d4321 = 0

d1343 + mua d4321 - mub d4343 = 0

where

dmnop = (xm - xn)(xo - xp) + (ym - yn)(yo - yp) + (zm - zn)(zo - zp)

Note that dmnop = dopmn

Finally, solving for mua gives

mua = ( d1343 d4321 - d1321 d4343 ) / ( d2121 d4343 - d4321 d4321 )

and back-substituting gives mub

mub = ( d1343 + mua d4321 ) / d4343

**Source Code**

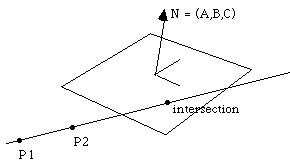
Original C source code from the author: [lineline.c](http://paulbourke.net/geometry/pointlineplane/lineline.c)  
Contribution by Dan Wills in MEL (Maya Embedded Language): [source.mel](http://paulbourke.net/geometry/pointlineplane/source.mel).  
A Matlab version by Cristian Dima: [linelineintersect.m](http://paulbourke.net/geometry/pointlineplane/linelineintersect.m).  
A Maxscript function by Chris Johnson: [LineLineIntersect.ms](http://paulbourke.net/geometry/pointlineplane/LineLineIntersect.ms)  
LISP version for AutoCAD (and Intellicad) by Andrew Bennett: [int1.lsp](http://paulbourke.net/geometry/pointlineplane/int1.lsp) and [int2.lsp](http://paulbourke.net/geometry/pointlineplane/int2.lsp)  
A contribution by Bruce Vaughan in the form of a Python script for the SDS/2 design software: [L3D.py](http://paulbourke.net/geometry/pointlineplane/L3D.py)  
C# version by Ronald Holthuizen: [calclineline.cs](http://paulbourke.net/geometry/pointlineplane/calclineline.cs)  
VBA VB6 version by Thomas Ludewig: [vbavb6.txt](http://paulbourke.net/geometry/pointlineplane/vbavb6.txt)

# Intersection of a plane and a line

Written by [Paul Bourke](http://paulbourke.net/geometry/)  
August 1991

Contribution by Bryan Hanson: [Implementation in R](http://paulbourke.net/geometry/pointlineplane/intersectionLinePlane.R)

This note will illustrate the algorithm for finding the intersection of a line and a plane using two possible formulations for a plane.



### Solution 1

The equation of a plane (points **P** are on the plane with normal **N** and point **P3** on the plane) can be written as

**N** dot (**P** - **P3**) = 0

The equation of the line (points **P** on the line passing through points **P1** and **P2**) can be written as

**P** = **P1** + u (**P2** - **P1**)

The intersection of these two occurs when

**N** dot (**P1** + u (**P2** - **P1**)) = **N** dot **P3**

Solving for u gives

http://paulbourke.net/geometry/pointlineplane/planeline2.gif

**Note**

* If the denominator is 0 then the normal to the plane is perpendicular to the line. Thus the line is either parallel to the plane and there are no solutions or the line is on the plane in which case there are an infinite number of solutions
* If it is necessary to determine the intersection of the line segment between **P1** and **P2** then just check that u is between 0 and 1.

### Solution 2

A plane can also be represented by the equation

A x + B y + C z + D = 0

where all points (x,y,z) lie on the plane.

Substituting in the equation of the line through points **P1** (x1,y1,z1) and **P2** (x2,y2,z2)

P = P1 + u (P2 - P1)

gives

A (x1 + u (x2 - x1)) + B (y1 + u (y2 - y1)) + C (z1 + u (z2 - z1)) + D = 0

Solving for u

http://paulbourke.net/geometry/pointlineplane/planeline3.gif

**Note**

* the denominator is 0 then the normal to the plane is perpendicular to the line. Thus the line is either parallel to the plane and there are no solutions or the line is on the plane in which case are infinite solutions
* if it is necessary to determine the intersection of the line segment between P1 and P2 then just check that u is between 0 and 1.