Share Edit Follow

answered Apr 3 '12 at 16:19

Grumdrig
15.3k 12 54 68

10 Very simple and elegant, but it does not deal well with colinearity, and thus more code needed for that purpose. – charles Jun 3 '13 at 8:44

8 For a solution that also handles collinearity check out geeksforgeeks.org/check-if-two-given-line-segments-intersect – Zsolt Safrany Nov 10 '14 at 10:34

Love this solution. Very simple and short! I made a wxPython program that draws a line and sees if it intersects with another line. I could not place it here so it is somewhere below this comment. – user1766438 Feb 3 '20 at 11:23

76

The equation of a line is:

```
f(x) = A*x + b = y
```

For a segment, it is exactly the same, except that x is included on an interval I.

If you have two segments, defined as follow:

```
Segment1 = {(X1, Y1), (X2, Y2)}
Segment2 = {(X3, Y3), (X4, Y4)}
```

The abcisse Xa of the potential point of intersection (Xa,Ya) must be contained in both interval I1 and I2, defined as follow :

```
I1 = [min(X1,X2), max(X1,X2)]
I2 = [min(X3,X4), max(X3,X4)]
```

And we could say that Xa is included into :

```
Ia = [max( min(X1,X2), min(X3,X4) ),
      min( max(X1,X2), max(X3,X4) )]
```

Now, we need to check that this interval Ia exists :

```
if (max(X1,X2) < min(X3,X4)):
    return False  # There is no mutual abcisses
```

So, we have two line formula, and a mutual interval. Your line formulas are:

```
f1(x) = A1*x + b1 = y
f2(x) = A2*x + b2 = y
```

As we got two points by segment, we are able to determine A1, A2, b1 and b2:

```
A1 = (Y1-Y2)/(X1-X2)  # Pay attention to not dividing by zero
A2 = (Y3-Y4)/(X3-X4)  # Pay attention to not dividing by zero
b1 = Y1-A1*X1 = Y2-A1*X2
b2 = Y3-A2*X3 = Y4-A2*X4
```

If the segments are parallel, then A1 == A2 :

```
if (A1 == A2):
    return False  # Parallel segments
```

A point (Xa,Ya) standing on both line must verify both formulas f1 and f2:

```
Ya = A1 * Xa + b1
Ya = A2 * Xa + b2
A1 * Xa + b1 = A2 * Xa + b2
Xa = (b2 - b1) / (A1 - A2)   # Once again, pay attention to not dividing by zero
```

The last thing to do is check that Xa is included into Ia:

```
if ( (Xa < max( min(X1,X2), min(X3,X4) )) or
     (Xa > min( max(X1,X2), max(X3,X4) )) ):
    return False  # intersection is out of bound
else:
    return True
```

In addition to this, you may check at startup that two of the four provided points are not equals to avoid all that testing.

Share Edit Follow

edited Dec 3 '19 at 9:07

Martijn Pieters ♦
910k 252 3612 3054

answered Oct 1 '10 at 10:29

OMG_peanuts
1,627 1 13 19

1 Segments, they are segments, sorry. Could you update your answer given segments ? – aneuryzm Oct 1 '10 at 10:33

16 This is not so complicated, i wrote a lots of (unessential ?) intermediate steps in a comprehension purpose. The main points to implements are just : Check mutual interval existence, calculate A1, A2, b1, b2, and Xa, and then check that Xa is included in the mutual interval. That's all :) – OMG_peanuts Oct 1 '10 at 13:36

3 A1 - A2 will never be zero because if(A1 == A2) would have returned before this calculation in that case. – inkredibl Oct 22 '12 at 17:49

3 if A1 == A2 and b1 == b2, the segments are on top of each other and have infinitely many intersections – lynxoid May 28 '13 at 21:20

6 Formula A1*x + b1 = y doesn't handle vertical lines hence vertical segments should be handled separately with this method. – dmitri Aug 26 '13 at 7:51