Animated Point to Line

Method 1:

This first method is exactly as the one in "Lecture 16 - Normal Line Equation - Animated Point To Line" pdf file, on moodle.

Steps:

a- Check if **Bs** and **Be** are both on the inside half plane (rejection test)
   - **If((n.Bs < n.P0) && (n.Be < n.P0))**

b- Check if **Bs** and **Be** are both on the outside half plane (rejection test)
   - **If((n.Bs > n.P0) && (n.Be > n.P0))**

c- Compute **n.V**
   - If (**n.V** == 0) then no collision

d- Compute **ti**, the time of intersection
   - **ti = (n.P0 – n.Bs) / (n.V)**

e- If (**ti** < 0) or (**ti** > 1) then no collision

f- Check if **Bi** is outside of the area of **P0P1** from **P0** side (rejection test)
   - **if((Bi – P0).(P1 – P0) < 0)**

g- Check if **Bi** is outside of the area of **P0P1** from **P1** side (rejection test)
   - **if((Bi – P1).(P0 – P1) < 0)**

h- If your reach here, it means **Bi** is your intersection point

Method 2: This method is the best because, similar to method 1, we can reuse the 3 dot products used in rejection tests in the computation of ti, and the last two dot products are replaced with only one.

Steps:

a- Check if **Bs** and **Be** are both on the inside half plane (rejection test)
   a. **If((n.Bs < n.P0) && (n.Be < n.P0))**

b- Check if **Bs** and **Be** are both on the outside half plane (rejection test)
   a. **If((n.Bs > n.P0) && (n.Be > n.P0))**

c- Compute **n.V**
   a. If (**n.V** == 0) then no collision

d- Compute **ti**, the time of intersection
   a. **ti = (n.P0 – n.Bs) / (n.V)**

e- If (**ti** < 0) or (**ti** > 1) then reject

f- In this step we replace the steps **f** and **g** from the previous method with only one dot product test to check if **Bi** is within **P0** and **P1** area. We test **if((Bi – P0).(Bi – P1) < 0)** return collision at point **Bi**

Method 3:

a- Get the outward normal of the vector formed by **(Be – Bs)**, where **Be** is the starting moving point position, and **Be** is the end moving point position (both in one frame). The outward normal vector is **M**

b- Now we check if the endless line formed by **(Be, Bs)** is passing in between **P0** and **P1** as follow: **if((BsP0.M)*(BsP1.M) < 0)** then **(Bs,Be)** endless line is intersecting the line segment **(P0,P1)**

c- Compute **ti**, the time of intersection

   a. **ti = (n.P0 – n.Bs) / (n.V)**

d- **If (ti < 0) or (ti > 1)** then no collision

e- Else If you reach here, it means **Bi** is your intersection point.