

Physics Simulation Interactions

* Rect to Line
  + Calculate the distance each corner is from the line.
  + Determine if the corner lies on the opposite side of the line from the center of the rect
  + Calculate the contact point.
  + Move the rect away from the line based on the smallest penetration value.
  + Calculate the elasticity of both objects.
  + Apply force to the rect at the contact point considering the elasticity.
  + Call any on collision functions.
* Circle to Line
  + Calculate the distance the circle is to the closest point on the line
  + Flip the normal and distance depending on the side of the line the circles lies on
  + Calculate the intersection amount by subtracting the circles radius.
  + Move the circle so its not intersecting with the line based of the normal and intersection amount
  + Calculate the contact point
  + Calculate the elasticity of both objects.
  + Apply force to the circle at the contact point considering the elasticity.
  + Call any on collision functions.
* Circle to Circle
  + Check if the distance from one circle to the other is less then the sum of their radiuses.
  + Subtract the radiuses from the distance to get the penetration value
  + The collision normal is the normalized direction from one circle to the other.
  + Calculate the required amount to move both objects apart based given the penetration value and the collision normal.
  + Calculate the elasticity of both objects.
  + Apply force at the contact point to the objects considering the elasticity.
  + Call any on collision functions.
* Circle to Rect
  + Check if each corner of the rect intersects with the circle
  + Rotate the rect to be axis aligned
  + Check if the circle intersects with a side of the rect.
  + Get the contact normal & penetration from the collision
  + Calculate the required amount to move both objects apart based on the smallest penetration value and the collision normal.
  + Calculate the elasticity of both objects.
  + Apply force at the contact point to the objects considering the elasticity.
  + Call any on collision functions.
* Rect to Rect
  + Rotate one of the rect’s to be axis aligned.
  + Check each corner of the other rect to see if any of its corners intersect.
  + Check which side the other rect lies on.
  + Get the penetration of the intersecting corner and side normal relative to that determined side
  + Repeat for the other rect
  + Calculate the required amount to move both objects apart based on the smallest penetration value and the collision normal.
  + Calculate the elasticity of both objects.
  + Apply force at the contact point to the objects considering the elasticity.
  + Call any on collision functions.

Possible Improvements:

* Non-primitive shape collision with SAT.
  + Polygon shapes for a more diverse representation of physical objects.
* Sweeping
  + Ray Casting
  + Collision detection for fast moving objects
* Contact Friction
  + Accurate interactions between 2 continuously colliding objects

References:

* Stack Over Flow: [Markus Fjellheim](https://stackoverflow.com/users/3876766/markus-fjellheim) - <https://stackoverflow.com/questions/11654990/2d-physics-engine-collision-response-rotation-of-objects>