Collision Events

Eric Borgqvist

1 Collision event in Ball Game

The total energy for a system of N balls before any collision takes place is given by

$$E_b = \sum_{i=1}^{N} \frac{1}{2} m_i (\boldsymbol{v}_i \cdot \boldsymbol{v}_i) \tag{1}$$

Assume two balls are colliding. Introduce the vector $\boldsymbol{N}_i$ defined as the vector from the center of the ball i to the collision point. Further on, we introduce a quantity k defined from

$$\boldsymbol{N_i} = k_i \frac{\boldsymbol{N_i}}{|\boldsymbol{N_i}|} = k_i \bar{\boldsymbol{N}}_i \tag{2}$$

Note that $\bar{\boldsymbol{N}}_1 = -\bar{\boldsymbol{N}}_2$. Assume two balls 1 and 2 are colliding. During collision, it is assumed that body 1 new velocity will get a contribution from $\boldsymbol{N}_2$ and similarly body 2 will get a contribution from $\boldsymbol{N}_1$. Conservation of linear momentum dictates that following relation must hold before and after the collision

$$m_1 \boldsymbol{v}_1 + m_2 \boldsymbol{v}_2 = m_1(\boldsymbol{v}_1 + k_2 \bar{\boldsymbol{N}}_2) + m_2(\boldsymbol{v}_2 + k_1 \bar{\boldsymbol{N}}_1) \tag{3}$$

E.g that

$$m_1 k_2 \bar{\boldsymbol{N}}_2 = -m_2 k_1 \bar{\boldsymbol{N}}_1 \tag{4}$$

And since the vectors are of exactly opposite direction, we have that $m_1 k_2 = m_2 k_1$.

The total energy of the system after the collision is assumed to be given by

$$E_a = \frac{1}{2} m_1 \left((\boldsymbol{v}_1 + k_2 \bar{\boldsymbol{N}}_2) \cdot (\boldsymbol{v}_1 + k_2 \bar{\boldsymbol{N}}_2) \right) + \frac{1}{2} m_2 \left((\boldsymbol{v}_2 + k_1 \bar{\boldsymbol{N}}_1) \cdot (\boldsymbol{v}_2 + k_1 \bar{\boldsymbol{N}}_1) \right) - d(\boldsymbol{v}_2 - \boldsymbol{v}_1) \cdot (\boldsymbol{v}_2 - \boldsymbol{v}_1) \tag{5}$$

where d is a disspative constant chosen by the user. If set equal to zero, all collisions are perfectly elastic and total energy is conserved. Setting the total energies before and after collision to be equal, e.g. $E_b = E_a$, cf. (1) and (5), we obtain a new equation with the unknown k. It is a second degree equation which explicitly becomes

$$\frac{(m_2 k_1^2 + m_1 k_2^2)}{2} + \left(k_2 m_1 (\boldsymbol{v}_1 \cdot \bar{\boldsymbol{N}}_2) + k_1 m_2 (\boldsymbol{v}_2 \cdot \bar{\boldsymbol{N}}_1) \right) - \tilde{d} = 0 \tag{6}$$

where $\tilde{d} = d(\boldsymbol{v}_2 - \boldsymbol{v}_1) \cdot (\boldsymbol{v}_2 - \boldsymbol{v}_1)$ was introduced. Utilizing that $k_2 = \dfrac{m_2}{m_1} k_1$ we can rewrite the above into an equation with only one unknown

$$\frac{m_1^2 + m_2^2}{2m_1} k_1^2 + k_1 m_2 \left((\boldsymbol{v}_1 \cdot \bar{\boldsymbol{N}}_2) + (\boldsymbol{v}_2 \cdot \bar{\boldsymbol{N}}_1) \right) - \tilde{d} = 0 \tag{7}$$

The solution is given by

$$k_1 = -\frac{p}{2} \pm \frac{1}{2} \sqrt{p^2 + \frac{4\tilde{d}}{m_1}} \tag{8}$$

where $p = \dfrac{m_2}{m_1} ((\boldsymbol{v}_1 \cdot \boldsymbol{N}_2) + (\boldsymbol{v}_2 \cdot \boldsymbol{N}_1))$. The k_1 is chosen as $max(k_1^{(1)}, k_2^{(2)})$ if $p < 0$ and as $min(k_1^{(1)}, k_1^{(2)})$ otherwise.

Have to be careful with how to round results since we work on a pixel grid (which is discrete)