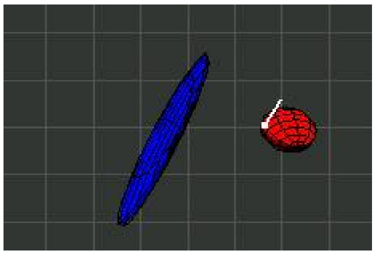
Rigid Body Collision Resolution



This page tackles the problem of how to resolve the collision of two rigid bodies. My first encounter with this question was while designing a computer game in 2004. For over three years now, I did not come across a satisfactory treatment neither in books, nor in the Internet. During my semester in Kiruna, I finally had the opportunity to thoroughly investigate the problem myself.

The input to the problem are:

- Initial states (0. and 1. order) of the bodies at the moment of contact,
- point, and normal of contact.

Without loss of generality, we assume:

- The center of total mass is at position 0, and
- the total linear momentum (impulse) is the null vector.

We are interested in

- resolving the collision within a single instance of time
- the velocity, and the angular velocity after the moment of contact.

A mathematical solution to the problem is derived below. To perform simulations and to visualize the objects in motion, we develop a computer program, which is available for download. The implementation uses double precision, and simple Euler integration to solve the Euler equations of rotation.

Rigid Body Collision Resolution (C++) *	ribocore.zip	230 kB
Download this article	rigid_body_collis...	320 kB

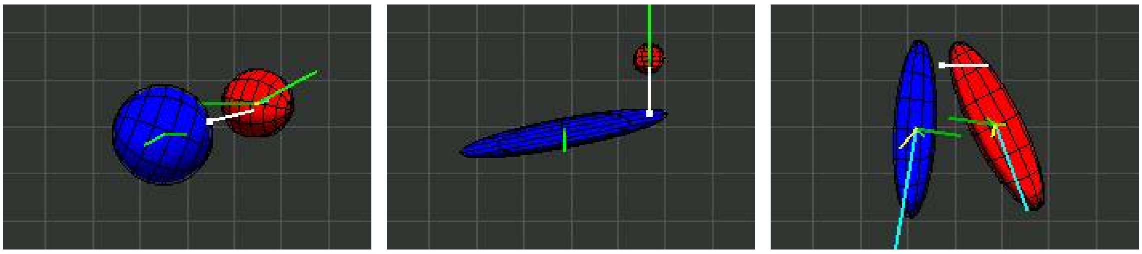
** The program checks the math, and produces the animations on this website.*

It might be the warriors who get the glory,
but it's the engineers who build societies.
B'Elanna Torres

A mathematical solution to the problem

This section suggests forces and torques that - when applied to the bodies during collision - will preserve total linear momentum as well as total angular momentum. In this model, we allow the bodies to intrude each other. However, the forces and torques applied during collision make the bodies separate again.

In the illustrations, the ellipsoids visualize the inertia tensors of the bodies. The ellipsoids are not identical to the exterior of the body.



billiard balls — baseball — stuntmen

Most likely, the following notions are familiar to the reader.

The rigid bodies are enumerated by $i = 1,...,k$. Each body has

mass	$m_i \in \mathbb{R}^+$	
inertia tensor	$I_i \in \mathbb{R}^{3\times 3}$	
position	$p_i \in \mathbb{R}^3$	of the center of mass in world coordinates
orientation	$R_i \in SO(3)$	transforming from object to world coordinates
velocity	$v_i \in \mathbb{R}^3$	in world coordinates
angular velocity	$\omega_i \in \mathbb{R}^3$	in object coordinates

Recall the assumptions stated above. The center of total mass is at position 0, i.e.

$$0 = \sum_i m_i p_i$$

and the total linear momentum (impulse) is the null vector:

$$0=\sum_i m_i v_i$$

We treat the collection of bodies as a closed system. Within this system, the total linear momentum, and the total angular momentum are perserved, i.e. constant vectors, at all times.

The total linear momentum is the sum of the linear momentums of each body

$$M_{\text{lin}}=\sum_i m_i v_i$$

The total angular momentum is the sum of the angular momentums of each body with respect to the center of total mass

$$M_{\text{ang}}=\sum_i p_i\times(m_i v_i)+R_i.I_i.\omega_i$$

Since in our world, the total momentums are constant, the derivative with respect to time is the null vector. I.e. we demand

$$0=d_t M_{\text{lin}}=\sum_i m_i a_i$$

$$\begin{aligned}0=d_t M_{\text{ang}}&=\sum_i v_i\times(m_i v_i)+p_i\times(m_i a_i)+R_i.\Omega_i.I_i.\omega_i+R_i.I_i.\tau_i\\&=\sum_i p_i\times(m_i a_i)+R_i.\Omega_i.I_i.\omega_i+R_i.I_i.\tau_i\end{aligned}$$

In the above derivation, we assume that all quantities depend on time, except mass and the inertia tensors. The introduced variables represent the following:

$$\begin{array}{lll}\text{acceleration} & a_i=d_t v_i & \text{in world coordinates}\\ \text{torque} & \tau_i=d_t\omega_i & \text{in object coordinates}\end{array}$$

while $\Omega_i\in\mathrm{so}(3)$ is the angular velocity in matrix form

$$\Omega_i=\begin{pmatrix}0 & -\omega_i^3 & \omega_i^2\\ \omega_i^3 & 0 & -\omega_i^1\\ -\omega_i^2 & \omega_i^1 & 0\end{pmatrix}$$

We are interested in forces and torques applied to the bodies during the period of contact.

We specialize the formulas on two rigid bodies, i.e. we assume $k=2$. We introduce a few more symbols that characterize the contact between the two bodies:

$$\begin{array}{lll}\text{degree of penetration} & \mu\in\mathbb{R}_0^+ & \\ \text{normal of contact} & n\in\mathbb{R}^3 & \text{in world coordinates away from body 1}\\ \text{point of contact} & r_i\in\mathbb{R}^3 & \text{in world coordinates with respect to } p_i\end{array}$$

If bodies are intruding each other, then let μ be proportional to the distance of intrusion. If the bodies are not in contact, we set $\mu=0$. We suggest to apply the following forces and torques

$$\begin{aligned}m_1 a_1&=-\mu n\\ m_2 a_2&=+\mu n\\ R_1.I_1.\tau_1&=-R_1.\Omega_1.I_1.\omega_1-r_1\times(\mu n)\\ R_2.I_2.\tau_2&=-R_2.\Omega_2.I_2.\omega_2+r_2\times(\mu n)\end{aligned}$$

We show that under these conditions both, the total linear momentum and the total angular momentum are preserved. Conservation of total linear momentum follows because

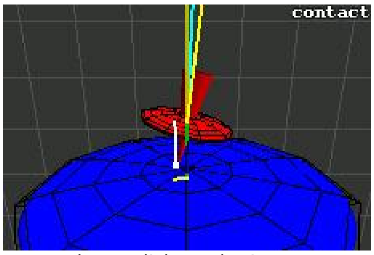
$$m_1 a_1+m_2 a_2=-\mu n+\mu n=0$$

The total angular momentum does not change because

$$\begin{aligned}&p_1\times(m_1 a_1)+R_1.\Omega_1.I_1.\omega_1+R_1.I_1.\tau_1+p_2\times(m_2 a_2)+R_2.\Omega_2.I_2.\omega_2+R_2.I_2.\tau_2\\ &=-p_1\times(\mu n)-r_1\times(\mu n)+p_2\times(\mu n)+r_2\times(\mu n)\\ &=(p_2+r_2-r_1-p_1)\times(\mu n)=0\times(\mu n)=0\end{aligned}$$

These formulas produce the animations shown earlier. As a bonus application, we implement these collision mechanics to animate a gyroscope: We impose friction to slow down linear velocity, and angular velocity when bodies are in contact. This ultimately causes the gyroscope to tip over.

Animation of a gyroscope



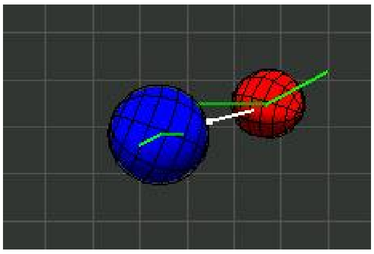
please click on the image

Man muss wissen, bis wohin man zu weit gehen kann.
Jean Cocteau

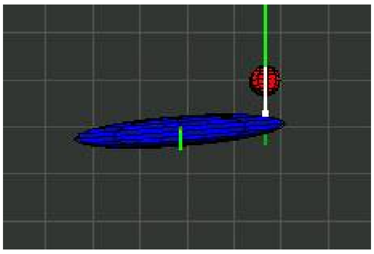
There is no time for procrastination.

In the introduction, we have stated the ultimate goal of this project: Given the linear and angular velocities just prior to the contact, we would like to have a formula that computes the linear and angular velocities just after the contact. This section suggests a solution to this problem. The solution is consistent with the definitions and derivations carried out earlier.

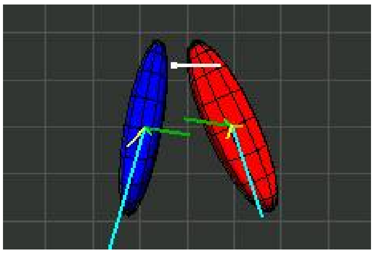
In the illustrations, the ellipsoids visualize the inertia tensors of the bodies. The ellipsoids are not identical to the exterior of the body.



billiard balls



baseball



stuntmen

I hope you enjoy the upcoming formulas as much as I do.

First, we introduce a few more symbols

velocity after collision	$\tilde{v}_i \in \mathbb{R}^3$ in world coordinates
angular velocity after collision	$\tilde{\omega}_i \in \mathbb{R}^3$ in object coordinates
constant	$\lambda \in \mathbb{R}_0^+$

We suggest: The linear and angular velocities just prior to the contact relate to the linear and angular velocities just after the contact in the following way

$$\begin{aligned} m_1 \tilde{v}_1 &= m_1 v_1 - \lambda n \\ m_2 \tilde{v}_2 &= m_2 v_2 + \lambda n \\ R_1.I_1.\tilde{\omega}_1 &= R_1.I_1.\omega_1 - r_1 \times (\lambda n) \\ R_2.I_2.\tilde{\omega}_2 &= R_2.I_2.\omega_2 + r_2 \times (\lambda n) \end{aligned}$$

We show that under these conditions both, the total linear momentum and the total angular momentum are preserved. Conservation of total linear momentum follows because

$$(m_1 \tilde{v}_1 + m_2 \tilde{v}_2) - (m_1 v_1 + m_2 v_2) = -\lambda n + \lambda n = 0$$

The total angular momentum after collision is

$$p_1 \times (m_1 v_1 - \lambda n) + R_1.I_1.\omega_1 - r_1 \times (\lambda n) + p_2 \times (m_2 v_2 + \lambda n) + R_2.I_2.\omega_2 + r_2 \times (\lambda n)$$

From that, we subtract the total angular momentum before collision

$$p_1 \times (m_1 v_1) + R_1.I_1.\omega_1 + p_2 \times (m_2 v_2) + R_2.I_2.\omega_2$$

and yield a difference of

$$\begin{aligned} & -p_1 \times (\lambda n) - r_1 \times (\lambda n) + p_2 \times (\lambda n) + r_2 \times (\lambda n) \\ &= (p_2 + r_2 - r_1 - p_1) \times (\lambda n) \\ &= 0 \times (\lambda n) = 0 \end{aligned}$$

At this point, λ is the only remaining unknown. As we will show next, λ is uniquely determined under the assumption that the two bodies collide elastically.

In an elastic collision, the total energy before the collision equals the total energy after the collision, i.e.

$$\frac{1}{2}\sum_i m_i v_i . v_i + \omega_i . I_i . \omega_i = \frac{1}{2}\sum_i m_i \tilde{v}_i . \tilde{v}_i + \tilde{\omega}_i . I_i . \tilde{\omega}_i$$

Previously, we have suggested the relations

$$\tilde{v}_1 = v_1 - \frac{\lambda}{m_1} n$$
$$\tilde{v}_2 = v_2 + \frac{\lambda}{m_2} n$$
$$\tilde{\omega}_1 = \omega_1 - \lambda q_1$$
$$\tilde{\omega}_2 = \omega_2 + \lambda q_2$$

where

$$q_i := I_i^{-1} . R_i^{-1} . (r_i \times n)$$

Now, determining λ so that the total energy is preserved reduces to solve a quadratic polynomial. Evidently, $\lambda = 0$ is one solution to conserve total energy, which corresponds to "no collision". However, we are interested in

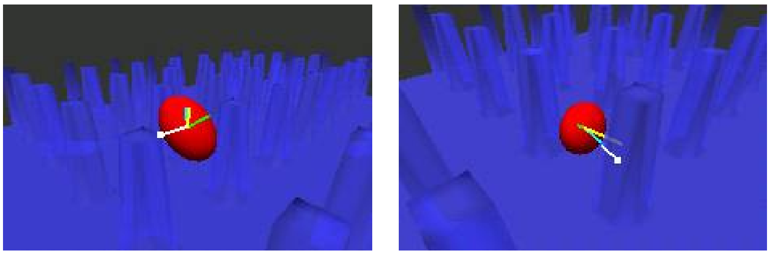
$$\lambda = 2\frac{v_1 . n - v_2 . n + q_1 . I_1 . \omega_1 - q_2 . I_2 . \omega_2}{(\frac{1}{m_1} + \frac{1}{m_2}) n . n + q_1 . I_1 . q_1 + q_2 . I_2 . q_2}$$

Basically, we are done at this point. The latter value of λ generates the three animations you see above. We conclude with some remarks:

- The forces and torques stated in the first section produce the same outcome if the collision process is modelled stiffly enough.
- Taking a fraction (for instance 95%) of λ will damp the collision, which results in more realistic animations. Different fractions may correspond to different (non rigid) materials. The total linear momentum as well as the total angular momentum are preserved nevertheless.
- More general, both momentums are conserved for any vector n, which is a convenient gateway for adapting the formula to less-elastic collisions.

To produce a final demonstration, we use the RAPID C-library for collision detection.

Ω Ultimate consumer



Please click on the images. The animation to the left imitates elastic collisions, whereas the right one uses 95% of λ.

Wenn du Zeit gewinnen willst,
musst du Zeit verlieren.
Jean-Jacques Rousseau

Further reading

The Open Dynamics Engine (ODE) originally by Russell Smith seems to be a popular C/C++ library for developers in gaming, but also in robotics. The library is suitable for *simulating articulated rigid body structures*. However, *ODE emphasizes speed and stability over physical accuracy.*

Recently, computer graphics researchers from Stanford University came up with a successful strategy to animate and collide innumerable rigid bodies. The paper Nonconvex Rigid Bodies with Stacking describes how to achieve breathtaking results. An open source implementation of their method is written by Danny Chapman. Again, the approach lacks physical precision.