What the Hell is the Inertia Tensor?
An introduction for non-physicists, by Dan Morris

Until recently, the depth of my understanding of the inertia tensor was that "it tells you how the mass of an object is distributed." I could say that if someone asked me, but I didn't really know what it meant. For example, I had no idea what it would mean if you made one element zero, or increased the magnitude of a particular element, etc. I'm writing this tutorial so I don't forget what I now understand about the inertia tensor… it's not going to be hugely quantitative, but it gets some important points across.

First let's pose a simple physics question: if I have an object with mass *m*, what force do I need to apply to get an acceleration *a*? The answer, of course, comes from Newton's basic equation of motion :

$$F = ma \qquad \text{(1.1)}$$

For a given object with mass *m*, if I apply a force *F*, I'll get an acceleration *a*. If I apply a force 2*F, I get an acceleration 2*a. If my object gets heavier, I need to apply more force to get the same acceleration. Sweet.

Now I want to ask the same question for *rotation*. First let's assume we have a world with known axes x, y, and z. And let's define the *rotational velocity* ω for an object as a three element-vector $(\omega_x, \omega_y, \omega_z)$ that contains the rate at which the object is spinning around each axis. If all three are zero, the object isn't spinning at all. If ω_x is the only non-zero element, the object is spinning nicely around the x-axis, etc.

So now I want to know what *torque* I need to apply to accelerate the object's rotation by some amount $d\omega$. For example, maybe the object is spinning around the x axis with rotational velocity $(\omega_x, 0, 0)$, and I want to stop it in *t* seconds. So I want to know what torque I need to apply to get a rotational acceleration of $(-\omega_x / t)$.

This is where the inertia tensor comes in. Just like I have F = ma for linear acceleration and force, I have the following equation for rotation :

$$\tau = I * d\omega \qquad \text{(1.2)}$$

…where τ is torque (a 3-element vector, indicating the torque around the x, y, and z axes), $d\omega$ is the rotational acceleration that we discussed before (a 3-element vector), and I – of course – is the magic inertia tensor (a 3-by-3 matrix). Note how it looks just like equation (1.1)!

I'm going to write this equation in matrix form, and we'll play with it a little bit before we see where the inertia tensor comes from.

In matrix form, the above equation looks like :

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \begin{bmatrix} d\omega_x \\ d\omega_y \\ d\omega_z \end{bmatrix} \quad \text{(1.3)}$$

Now I'm going to do my best to throw out an intuitive explanation of what some of the individual elements of the inertia tensor *mean* in the real world. The easy ones are the diagonal elements: I_{xx}, I_{yy}, and I_{zz}. In fact, let's take the simple case, where all the *other* elements in I are zero. This happens to be true for any object that's symmetric around all three axes, like a big sphere, or a big cube that's sitting nicely on the axes. We'll deal with the other elements later.

In this case, for example, I can multiply the first row of equation (1.3) out, to get :

$$\tau_x = I_{xx} * d\omega_x \quad\quad\quad \text{(1.4)}$$

This looks a *lot* like F = ma. Basically in this case, I can think of I_{xx} as the rotational inertia around the *x* axis. So for rotation around the x axis, I_{xx} behaves just like the *m* in *F=ma*. The bigger my I_{xx} is, the more torque I have to apply around the x axis to get it to spin. Or, conversely, for a given torque, a bigger I_{xx} means I'll get *less* rotation around the x axis. Just like m. In case I didn't mention it, it's a lot like m.

So for example, what would it mean for I_{xx} to be zero? It would mean that for a given torque, I would get an infinite rotational acceleration around the x axis. That's an important lesson. For real objects, none of the diagonal elements in *I* can be zero, since we know there's no real object where a tiny touch will set it spinning at an infinite rate.

Okay, now the hard part… what's the deal with the off-diagonal elements, for example I_{xy}. This question is really hard to explain quantitatively without dealing with the inverse of the inertia tensor, so I'm going to blow off the hard parts and deal with it qualitatively. What this element (I_{xy}) tells me is how much my object will be accelerated around the *y* axis when I apply torque around the *x* axis.

What? How's that possible? How could spinning an object around one axis make it rotate around another axis too?

For symmetric objects, it can't. If I take a big uniform cube in space and I torque it around the x axis, intuition (correctly) suggests that it only spins around the x axis.

But now let's say I attach a big-ass weight somewhere on the sphere. For example, let's say I'm spinning the Earth around the axis that runs from the north pole to the south pole (and we're

pretending of course that the Earth is a uniform sphere), and I attach a big weight to Canada. Now suddenly I've accelerated that big weight, and it wants to "pull" that part of the world along with it. This is going to make the Earth want to "tilt", so the north pole moves along with big heavy Canada. This is a rotational acceleration along an axis *other* than the one I torqued!

Okay, so we're blown away by that, and that's about as far as I'm going to go with the off-diagonal elements. You can see that they're always zero for a perfectly symmetric object, so the object spins only around the axes you torque it around.

Another footnote, while we're on the topic… for those of you familiar with rotational bases, it's interesting to note that for *any* object, there's some basis along which the inertia tensor is diagonal (all the off-diagonal elements are zero). That means that there's some set of perpendicular axes (x,y,z) around which I can torque the object and get rotational acceleration only along the axis I torqued… another way of saying this is that there's some set of "principal axes" around which an object will "spin stably", meaning it can spin around those axes without "wobbling" off-axis. Finding those axes is straightforward, but I'm not going to talk about it here.

The last thing I want to cover is how the inertia tensor is defined… so far we've treated it like a magic quantity that emerges from the sky. Let's look at the equations for on of the diagonal elements, which is representative of how we'd compute all of them :

$$I_{xx} = M \int_V (y^2 + z^2) dV \quad \text{(1.5)}$$

Here M represents the mass of the object, and V is the volume of the object. y and z are the positions of a particular point along the y and z axes, where zero is at the center of mass of the object (that's important).

So what does the I_{xx} equation mean? We're taking an integral over the whole volume, and we're summing the squared distance of every point from the x axis (y^2+z^2). So for a given amount of mass, I'm going to have a bigger I_{xx} value if most of my object is far away from the x axis. Remember, that means that to get a given acceleration around the x axis, I need to apply more torque around the x axis. How can we understand that intuitively?

The classic example for understanding this comes from figure skating… have you ever watched an ice skater get ready to spin super fast on the ice? What does he/she do? If he wants to spin fast, he puts his arms way above his head.

What does this have to do with the inertia tensor? Imagine the axis of the skater's body is the x axis. He's going to spin himself around this axis with some torque (whatever he can generate with his legs). What we've learned is that the lower his I_{xx} value is, the more rotational acceleration he'll get for that torque. And what we learned just now is that he can make his I_{xx} value smaller by putting his mass closer to the x axis… and he does that by putting his arms – which carry some of his weight – along the axis of his body. Amazing.

Another way to think of this… if the ice skater holds his hands way out (which makes I_{xx} bigger), and he wants to rotate his whole body at a certain rate, his hands are going to have to move much faster than his feet (because they cover more ground each time he spins around, just like the horses at the outside of a merry-go-round have to move faster than the inner ones). So if he sticks his hands out, and he has to move some of his mass hella fast, it's intuitive that he should have to apply more torque to get the same rotational velocity. Voila, we have I_{xx}.

And now, of course, we hit the off-diagonal elements. The equation for I_{xy} looks like this :

$$I_{xy} = -M \int_V xy \, dV$$

$\qquad$ (1.6)

Unfortunately, we're again stuck with the fact that this is really hard to explain, since (a) I don't understand it that well and (b) it's hard to reason about the counter-intuitive effect that applying torque along one axis can induce rotation around another axis. But we can see how we compute this component, and we can get a couple interesting points out of it.

Possibly the most important is that if the object is symmetric, every point on one side of each axis is going to "cancel out" the corresponding point on the other side of the axis. Hence we get off-diagonal elements that are zero, just like we discussed above.

Another thing we notice is that I_{xy} will be exactly the same as I_{yx}, since the inside of the above integral is just xy, which is the same as yx. This tells us that all inertia tensors are symmetric, which makes them numerically friendly for many applications.

Okay, I'm going to stop there… this was just my way of re-iterating my basic intuition for the inertia tensor. I'm not about to solve hard problems in mechanics, but this intuition is typically more than enough to implement and reason about algorithms in physical simulation. Another good resource is the popular SIGGRAPH mini-course on physical simulation :

http://www-2.cs.cmu.edu/~baraff/sigcourse/

Good luck, and happy tensor-ing.

-Dan