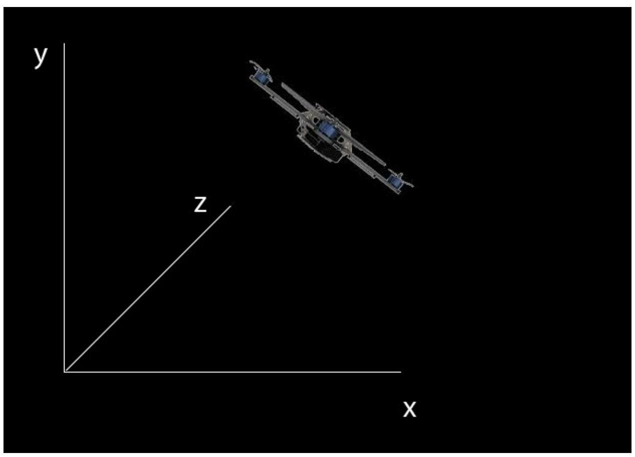
Lecture 2A.1

Transformations

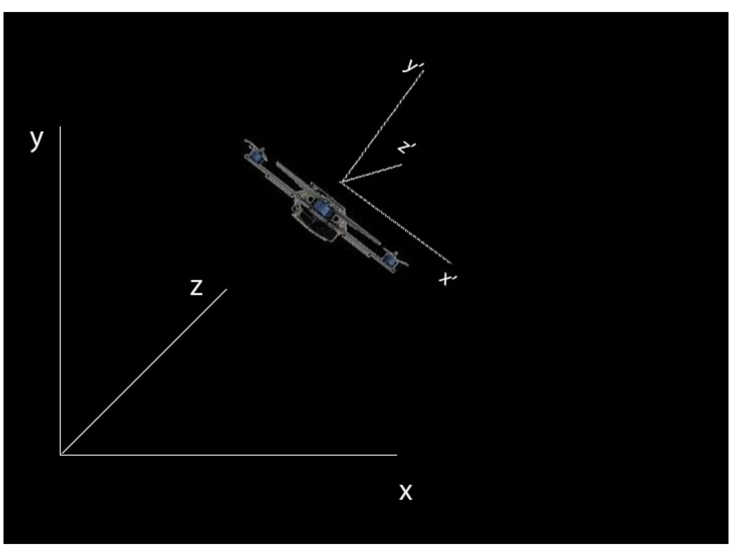
This module will explore the fundamental concepts required to describe three-dimensional motion - both the position and orientation of rigid objects flying through space. We will develop the basic tools to describe three-dimensional displacements through rigid body transformations.

Reference Frames

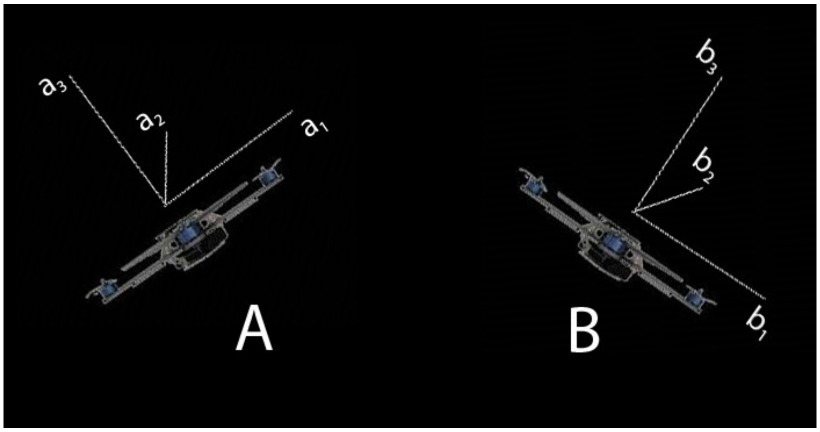
The key concept that underpins these techniques is the **reference frame**, here shown with orthogonal axes x, y, and z. Every position and orientation is described relative to some reference frame:



Each quadrotor can have its own reference frame, distinct from the world reference frame, here shown with the orthogonal axes x', y' and z':



Let's consider two distinct positions and orientations for a quadrotor. We'll call these two positions & orientations A and B, and each will have its own reference frame.



In reference frame {A}, we choose three linearly-independent basis vectors a_1, a_2, and a_3. While these don't have to be mutually orthogonal, it's convenient to choose them to be mutually orthogonal. The key idea though, is that they must at least be linearly independent. Similarly, in frame {B} we have three linearly-independent vectors, b_1, b_2, and b_3.

Now any vector, v, in three-dimensional space can be written as a linear combination of these independent vectors. In frame {A}, we would write it as a linear combination of vectors a_1, a_2, and a_3:

$$\vec{v} = v_1\vec{a}_1 + v_2\vec{a}_2 + v_3\vec{a}_3$$

In Frame {B}, we would write it in terms of b1, b2, and b3.

$$\vec{v} = v_1\vec{b}_1 + v_2\vec{b}_2 + v_3\vec{b}_3$$

Notation

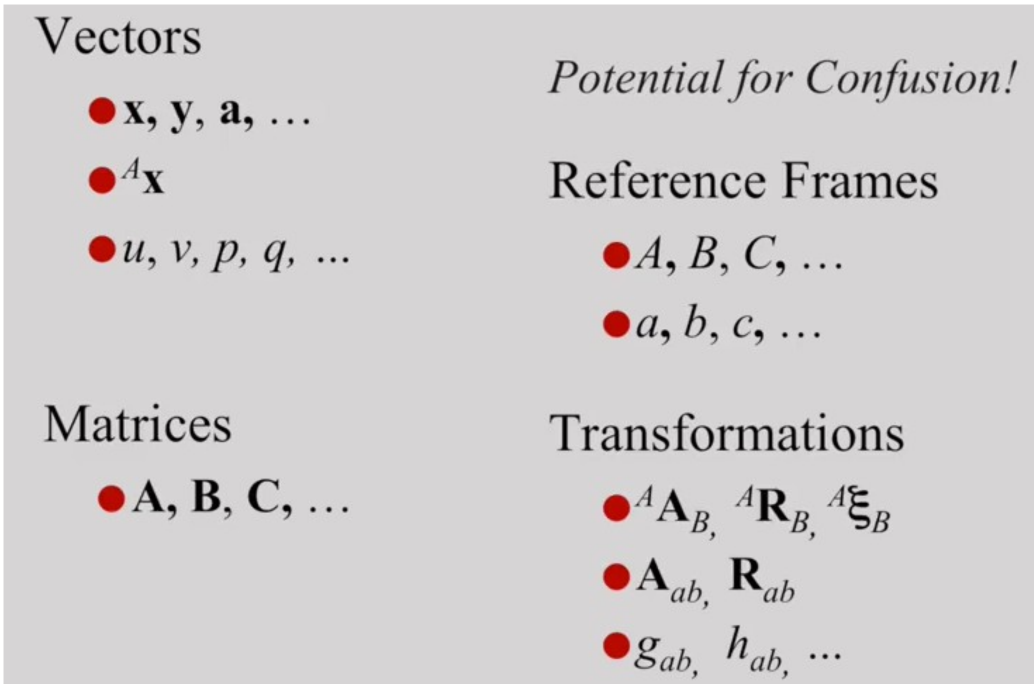
Typically we'll use boldface to denote vectors, and we may use a leading superscript to denote the frame in which we are writing that vector, e.g. ${}^{A}\mathbf{x}$. In some cases, you may also see lower cased italicized letters denoting vectors.

Similarly for reference frames, we use italicized letters but these are usually upper case, e.g. *A*, *B*, *C*… However, depending which texts or papers you read, you might also see italicized lowercase letters denoting reference frames.

There's a lot of potential for confusion. And you will often have to go back and 'recalibrate' yourself with new notation when you read something new.

Matrices are denoted by uppercase, boldface letters.

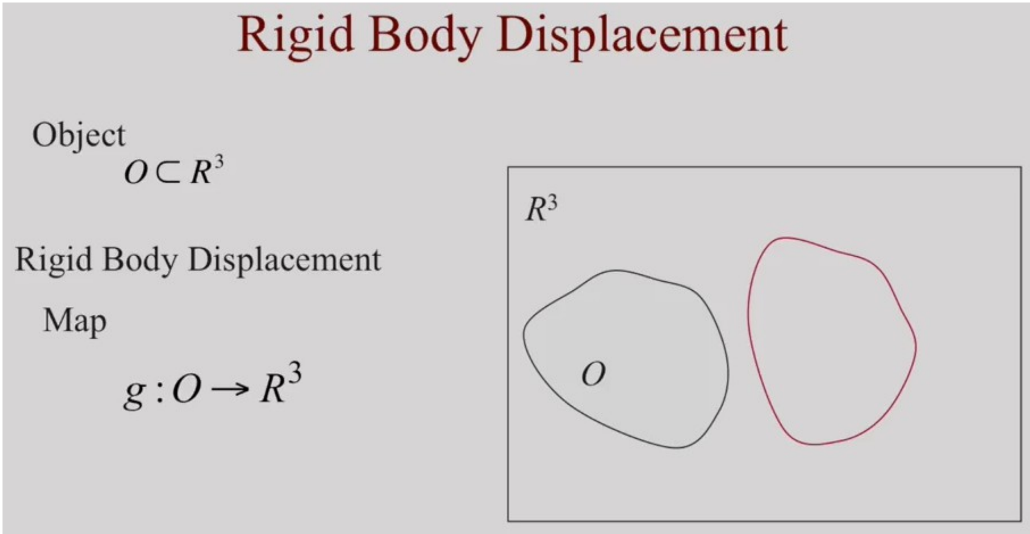
We will also talk about 'transformations'. This is the notation we use to describe how vectors in one frame can be written in another frame. For example, if we have an uppercase letter A with a leading superscript A and a trailing superscript B, ${}^{A}A_{B}$, that usually indicates that we have an object that transforms vectors in frame {B} into vectors in frame {A). A lowercase italicised letter with two trailing subscripts, a & b, g_{ab} denotes a transformation from frame {B} into frame {A}.



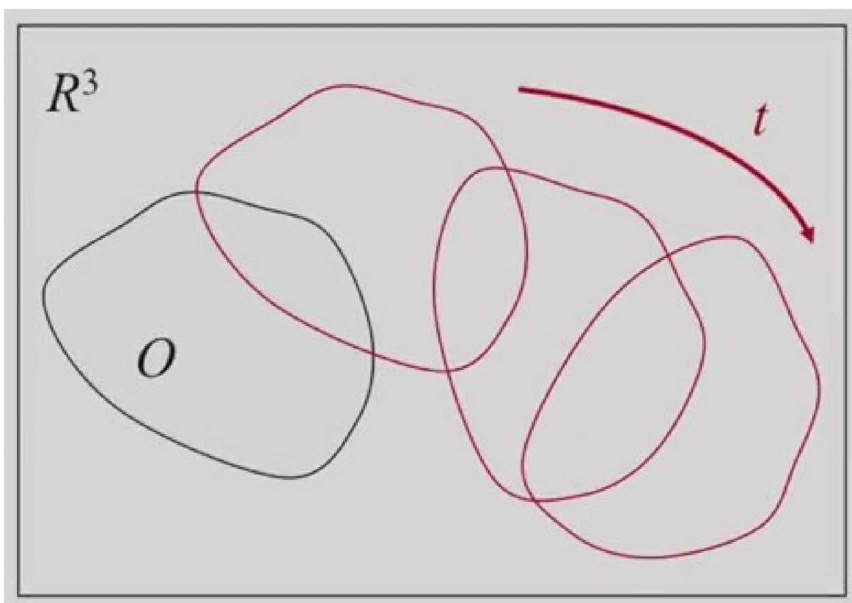
Rigid Body Transformations

A rigid body is an idealised solid body in which deformation is neglected. This means that the distance between any two given points of a rigid body remains constant, regardless of any external forces exerted on it.

Let's talk about rigid body displacements. A rigid body, O, is a collection of points in three dimension. It is a subset of R^3. This object can have multiple positions and orientations. A rigid body displacement is simply a map from this collection of points in the object, to its physical manifestation in R3.



This object may occupy different positions and orientations over time. Accordingly, there will be different rigid body displacements. Each of these is a map of the points in O to their physical manifestations in the real space, R^3.



As the body moves through space, that motion is described by a continuous family of maps, so the displacement g is now parameterised by time:

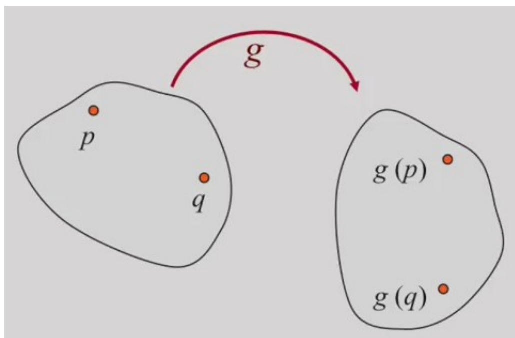
$$g(t): O \rightarrow R^3$$

The collection of points in O moves from one position and orientation to another over time, and this is a continuous set of displacements.

This is what we see in a quadrotor. It starts in a horizontal position, moves to another position, accelerating, changing its orientation, changing the direction of the thrust and then reversing the direction of the thrust by pitching back, and then slowing down to the goal position.

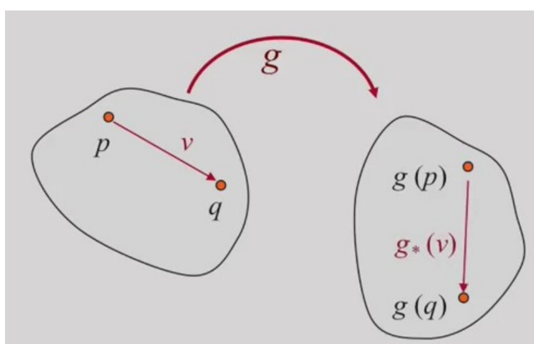
Each of these snapshots is a displacement, and this sequence of displacements represents a continuous family of displacements.

Consider a single point, p, on the rigid body. When the rigid body is displaced, the point, p, gets displaced into a new point, which we will call g(p):



A displacement is essentially a transformation of points. Of course, there are infinite points in a rigid body. If we have a second point, q, the same displacement will take q and move it into a new point, g(q).

Every pair of points defines a vector. The vector **v** emanates from p and terminates at q. Since g moves p to g(p), & q to g(q), it will also move the vector **v** to a new vector, which we call **g*(v)**.



So the displacement g induces a map on vectors. Remember, the displacement acts on points, but g* acts on vectors.

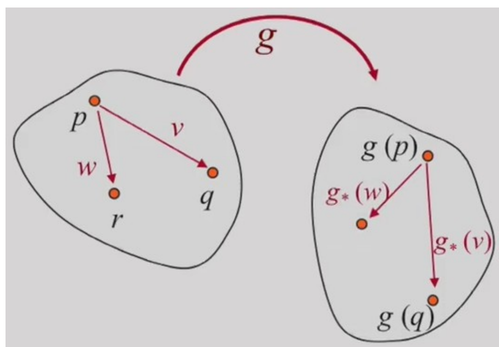
So, what makes the map, g, a *rigid body displacement*?

Two properties must be satisfied. First, the distance between any pair of points remains unchanged in a rigid body displacement. In fact, as explained above, that is the definition of a rigid body. So if we calculate the distance between p and q, and the distance between g(p) and g(q) after the displacement, those two distances must be identical:

$$\|g(p)-g(q)\|=\|p-q\|$$

The second property that rigid body displacements must satisfy has to do with cross-products of vectors that are attached to the rigid body.

So far, we have a single vector emanating from p and terminating in q. Let's choose a third point, r, and generate a second vector going from p to r. What happens to this vector, **w**, when the body gets displaced to its new position and orientation?



g* acts on **w** to generate a new vector, **g*(w)**. Now we can look at the cross-product of **v** and **w** and ask what happens to that cross-product when it goes to **v** cross **w**.

$$g^*(v)\times g^*(w)=g^*(v\times w)$$

If we look at the mapping of **v** into **g*(v)** and the mapping of **w** into **g*(w)**, we can compute the cross-product of **g*(v)** and **g*(w)**. It turns out that the cross product remains the same whether we calculate it before the displacement or after the displacement, provided the displacement is rigid. Thus, cross-products are preserved.

So if we look at the family of maps which preserve distances and cross-products, we essentially get rigid-body displacements.

So, to summarise, the two properties are:

1. Lengths are preserved.
2. Cross-products are preserved.

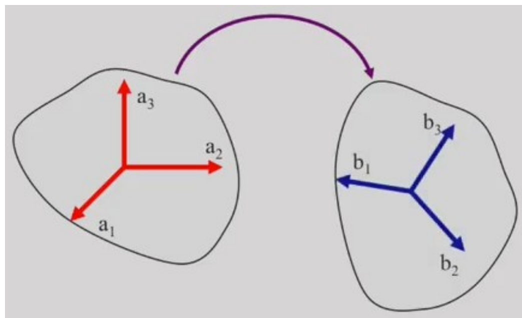
For a rigid body displacement we should be able to prove that, orthogonal vectors map to orthogonal vectors, and further, that g* preserves inner products. That is, if we compute the inner product of two vectors before the displacement, and the inner product of the same two vectors after the displacement, those inner products are the same.

- orthogonal vectors are mapped to orthogonal vectors

- g_* preserves inner products

$$g_\star(v) \cdot g_\star(w) = v \cdot w$$

A set of mutually orthogonal unit vectors attached to the rigid body will remain mutually orthogonal, and will remain unit vectors, after the displacement.



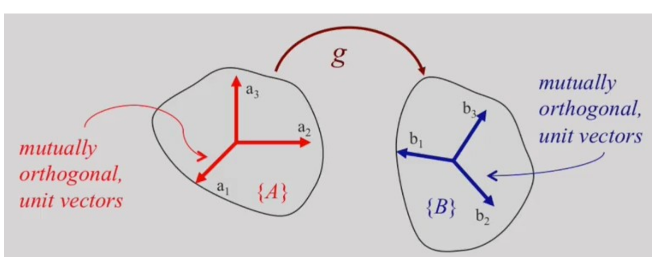
In summary, rigid body transformations, or rigid body displacements, satisfy two important properties. They preserve lengths and they preserve cross-products.

Note: The terms *rigid-body displacement* and *rigid-body transformation* are frequently used interchangeably. There is an important semantic difference between the two.

- *Transformations* generally refer to relationships between reference frames attached to two different rigid bodies,
- *Displacements* describe relationships between positions and orientations of a frame attached to the rigid-body as it moves around.

We have now seen an abstract description of what a rigid-body displacement is. Let's explore how we can actually do calculations to describe rigid body displacements, and then manipulate these displacements.

We start with the assumption that we have mutually orthogonal unit vectors attached to every rigid body.



If it's a transformation, we're referring to two different rigid bodies. If it's a displacement, it is two distinct positions and orientations of the same rigid body.

We can write the mutually orthogonal unit vectors in one frame as a linear combination of the mutually orthogonal unit vectors in the other frame, thus:

$$\mathbf{b}_1 = R_{11}\mathbf{a}_1 + R_{12}\mathbf{a}_2 + R_{13}\mathbf{a}_3$$
$$\mathbf{b}_2 = R_{21}\mathbf{a}_1 + R_{22}\mathbf{a}_2 + R_{23}\mathbf{a}_3$$
$$\mathbf{b}_3 = R_{31}\mathbf{a}_1 + R_{32}\mathbf{a}_2 + R_{33}\mathbf{a}_3$$

Here we see b_1, b_2, b_3 expressed as linear combinations of a_1, a_2, a_3. The coefficients are given by R_{11}, R_{12}, … and so on. This collection of nine coefficients can be gathered into a matrix, and we call this matrix a ***rotation matrix***.

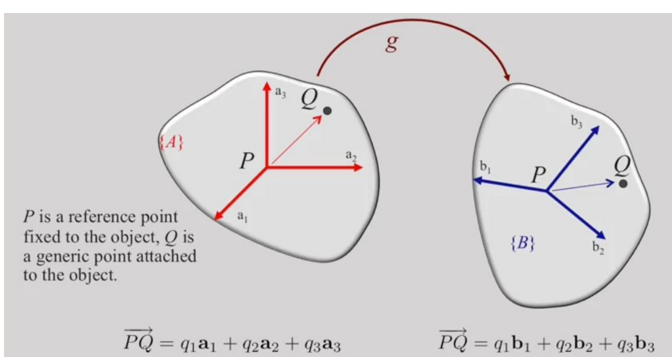
$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

The rotation matrix has several interesting properties.

1. It turns out that this matrix is **orthogonal**, which means if we take this matrix and multiply it by its transpose, we get the identity matrix. If we take the transpose of the matrix and multiply it by the matrix, we also get the identity.
2. The matrix is **special orthogonal**. In other words, the determinant of the matrix is +1.
3. Rotation matrices are **closed under multiplication**. The product of any two rotation matrices is also a rotation matrix.
4. The inverse of a rotation matrix is also a rotation matrix.

These properties will come in useful as we progress.

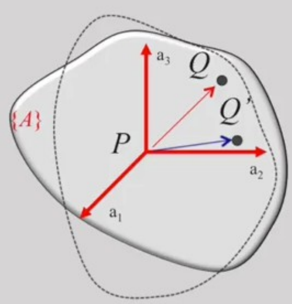
Now let's start looking at the structure of a rotation matrix. We have a rigid body, and the diagram shows two distinct positions and orientations of that body. We also have a point, q, which we will follow as it moves from one position to another:



The position vector **PQ** (where P is the origin) can be written in two different ways, depending on which snapshot we consider. In the first case, we write it as a linear combination of $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$, and in the second case, we've written it as a linear combination of $\mathbf{b}_1$, $\mathbf{b}_2$, and $\mathbf{b}_3$.

Let's focus on the vector, **PQ**.

I have two distinct position vectors, and they are both **PQ**. We can translate the rigid-body so that the two vectors have the same origin. We denote one vector as **PQ** and the second vector as **PQ'**.



$$\overrightarrow{PQ} = q_1\mathbf{a}_1 + q_2\mathbf{a}_2 + q_3\mathbf{a}_3$$

$$\overrightarrow{PQ'} = q_1'\mathbf{a}_1 + q_2'\mathbf{a}_2 + q_3'\mathbf{a}_3$$

Again, we can express the vector as a linear combination of a_1, a_2, and a_3. But now I'm going to get two different sets of coefficients. If I look at **PQ**, it has coefficients q_1, q_2 and q_3. If I look at **PQ'** prime, it has coefficients q_1', q_2', and q_3'.

I can ask how to write q_1', q_2', and q_3' as a function of q_1, q_2, q_3. Is there a matrix that connects them? And I can also ask the reverse question. Is there a matrix that connects q_1, q_2, q_3 to q_1', q_2', and q_3'?

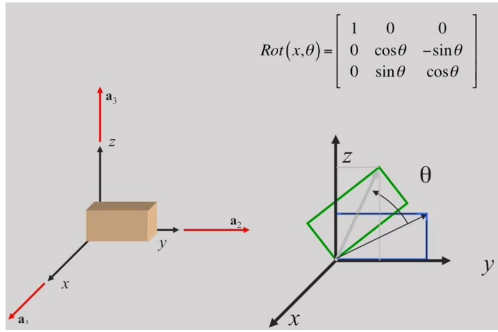
It turns out that there is, and this matrix is exactly the same one that we looked at earlier: the **rotation matrix**:

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \underbrace{\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}}_{\textit{rotation matrix}} \begin{bmatrix} q_1' \\ q_2' \\ q_3' \end{bmatrix}$$

$$\begin{bmatrix} q_1' \\ q_2' \\ q_3' \end{bmatrix} = \underbrace{\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}}_{\textit{rotation matrix}} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}$$

If we know how to write the mutually orthogonal unit vectors in one frame as a function of the vectors in the other frame, and I can do this by calculating the rotation matrix, the same matrix tells me how to transform vectors in one frame to another frame.

As an example, let's first look at something very simple. Consider a rectangular prism, with its axes aligned with the x, y, and z axes (or the a_1, a_2, and a_3 unit vectors).

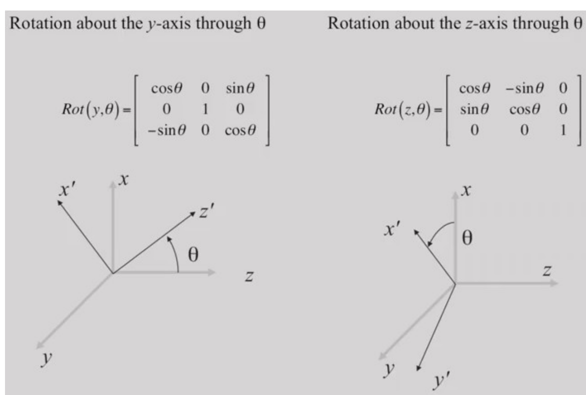


Let's rotate this rigid body about the x axis, through an angle, θ. We only need one parameter to describe this rotation, and that's the angle, θ. We can write this rotation matrix and we find that it has nine numbers:

$$Rot(x,\theta)=\begin{bmatrix}1&0&0\\0&\cos\theta&-\sin\theta\\0&\sin\theta&\cos\theta\end{bmatrix}$$

Four of the numbers are zero, there's an identity at the top-left, and the other numbers are essentially sines and cosines of the angle of rotation, θ. We can verify this through geometry - the calculations are actually quite simple.

Here are two more examples of rotation matrices. The one of the left describes a rotation through an angle, θ, about the y axis. The one on the right describes a rotation through an angle, θ, about the z axis:



Once more, in each case, there are four zeros, a one, and everything else is either the cosine or a sine of the rotation angle, θ.

In general, rotation matrices can look more complicated, but when you have rotations about the x, y, or z axes, they assume these very simple forms.

Lecture 2A.2

Rotations

Let's explore the properties of rotation matrices. To remind you, rotation matrices are 3x3 matrices that have the properties of orthogonality, and the fact that the determinant equals +1.

$$SO(3) = \{R \in \Re^{3x3} \; | R^T R = RR^T = I, \det R = 1\}$$

Matrices of these kinds constitute the **special Orthogonal** group. We use SO(3) to refer to this group. It's the *special Orthogonal group in three dimensions*.

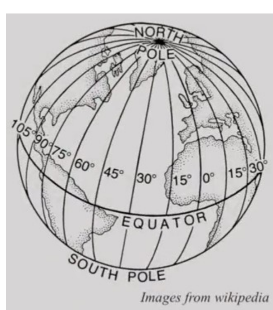
Rotation matrices are a great way of describing rotations, but they're not intuitively very obvious. So sometimes we prefer to use coordinates to describe SO(3).

Coordinates for SO(3):

- Rotation Matrices
- Euler angles
- Axis-angle parameterisation
- Exponential Coordinates
- Quaternions

In addition to **rotation matrices**, we'll discuss a set of angles called **Euler angles** that are often used to describe rotations. We also look at **parameterization** that explicitly describes the **axis of rotation** and the **angle of rotation**. It's also possible to use **exponential coordinates**, although we this isn't discussed in this course, and finally, it's possible to use **quaternions** (again, we don't discuss this in this class).

Let's begin by looking at coordinates on an object that we're familiar with: the sphere. In fact, let's consider the coordinates that we might use to describe a location on the Earth's surface.



We know that coordinates on the Earth's surface are described using latitudes and longitudes, but are these descriptors unique? In other words, given any point on the

Earth's surface, is there a unique combination of latitudes and longitudes that describe that point?

In fact, it turns out that the answer to this question is no. The poles each have unique latitudes, but their longitudes are not well defined. In fact, the North and South Poles can be described by any longitude. It's only the latitude that's well defined at these points on the surface of the Earth.

We want a coordinate chart in which every point on the Earth's surface maps to a pair of coordinates and that these coordinates are unique. Since this is hard to do, what we generally do is we use a collection of coordinates or a collection of charts. So, for example, we can agree to use a different nomenclature to describe the area around the poles as we approach the North or South Pole.

If we have a set of coordinates, there should be a one-to-one map between the Earth's surface and the coordinates.

One question we might want to ask is what's the minimum number of charts we need to cover the Earth's surface? Remember the chart has to lend itself to a coordinate system that's one to one. It turns out that the answer to this question is two.

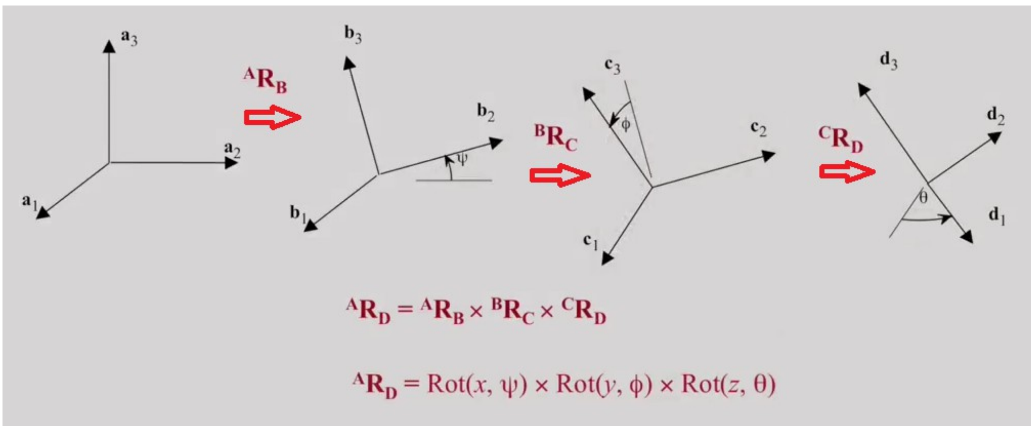
But the more important question is what's the minimum number of charts we need to cover the entire rotation group, SO(3)?

Lecture 2A.3

Euler Angles

The answer to the question that we posed at the end of the previous lecture is three. In fact, Euler showed that it's possible to cover the entire group of rotations using just three coordinates, and these coordinates are called Euler Angles.

To show how Euler angles work, I want to think about three successive rotations. The first rotation, ${}^{A}R_{B}$, going from frame {A} to frame {B}, the second, ${}^{B}R_{C}$, going from {B} to {C}, and the third, ${}^{C}R_{D}$, going from {C} to {D}:



A key idea behind Euler angles is that to describe the rotation, {A} to {D}, we can break it up into three successive rotations, {A} to {B}, {B} to {C}, and then {C} to {D}. We do this by simply multiplying the intermediate rotations.

So in the diagram above, the first rotation about $\mathbf{a}_1$ is through the angle, φ. This is a rotation about the x axis, and it allows us to get to $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$. The second rotation is about the vector $\mathbf{b}_2$, and that gets you to $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$. This rotation is through the angle, ϕ. Finally, the third rotation, is about the vector $\mathbf{c}_3$ and through the angle, θ.

So by simply multiplying rotations about the x-axis through φ, the y-axis through ϕ, and the z-axis through θ, we end up with the net rotation, which is from frame {A} to frame {D}. Again, remember that the nomenclature here refers both to displacements and to transformations.

So a rotation with a superscript A and a subscript D, can serve two purposes. It can transform vectors, written in terms of $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$, into a vector whose components are written in terms of $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$. It can also refer to the fact that a rigid-body has been displaced from its initial orientation at frame {A}, into its new orientation at frame {B}.

The three angles that we have seen here are the **roll**, **pitch**, and **yaw** angles.

$$ {}^{A}\mathbf{R}_{D} = \underbrace{\mathrm{Rot}(x, \psi)}_{roll} \times \underbrace{\mathrm{Rot}(y, \phi)}_{pitch} \times \underbrace{\mathrm{Rot}(z, \theta)}_{yaw} $$

Imagine a vehicle whose axis is oriented along $\mathbf{a}_1$. The first rotation is a roll rotation. The second rotation is a pitching motion about the second axis, which is $\mathbf{b}_2$. The third angle is a yaw rotation, and that is about the third axis which is $\mathbf{c}_3$.

Euler said that any rotation can be described by three successive rotations about linearly independent axes. So if we have three Euler angles, like the angles just described, we can deduce from them a 3x3 rotation matrix.

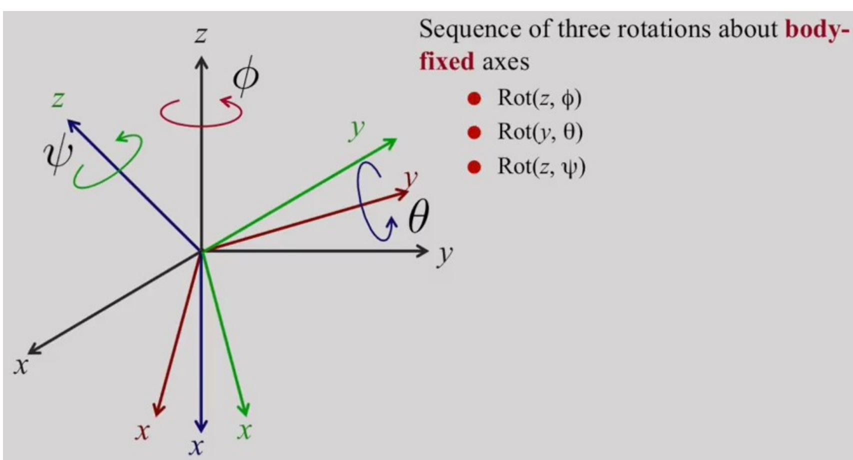
The question we want to ask now is whether the reverse is true. In other words, for every rotation matrix, is there a unique set of Euler angles? Or, to put it another way, is the map between the three Euler angles and the rotation matrix one-to-one?

Sadly, the answer is no. It's almost one to one, but there are points that are analogous to the North Pole and South Pole on the Earth's surface, at which the Euler angles are not well defined.

This set of Euler angles is often called the X-Y-Z Euler angles. That is because of the sequence in which these rotation matrices are applied, i.e. a rotation about the x axis followed by the y axis followed by the z axis. We can also have other types of Euler angles. This particular one is called Z-Y-Z Euler angles:

$$ R = Rot(z,\phi) \times Rot(y,\varphi) \times Rot(x,\theta) $$

Once again, the first rotation is about the z-axis, you rotate about the z-axis through ϕ. And then the second rotation is about the y-axis through θ. And then the third rotation is a rotation about the z-axis through φ.



Notice that every rotation occurs about a body fixed axis. Once the first rotation is complete, the second rotation about the y axis is now about a new y axis, because the first rotation rotated the original y-axis into a new position.

Z-Y-Z Euler angles have two rotations about z-axes. But are they the same z-axis? In other words, are the three axes really linearly independent? Clearly, in this case, the two z axes are not collinear, they are independent.

If this condition is satisfied, then the three angles are Euler angles, and they can parameterize the set of rotations.

Note that if an Euler angle is zero (e.g. $\theta = 0$ is a special (**singular**) case) we might run into problems. In this case, $\theta = 0$ would cause the two z axis to be collinear. As a result, the axes will not be linearly independent. In other words, the three axes about which were performing rotations would no longer be independent. So $\theta = 0$ is analogous to being at the North Pole or the South Pole on the earth's surface.

To explore this further, we'll look inside the computations that are involved in going from a rotation matrix to the three Euler angles. Assume we have a known rotation matrix, containing a set of nine numbers, and we want to recover the three Euler angles: ϕ, θ, and φ.

$$\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

In order to do this, we can write out the result of the multiplication of the three rotation matrices. The rotation about the z axis, followed by the rotation about the y axis, followed by the rotation about the z axis. The first rotation through ϕ, the second through θ, and the third through φ.

$$\begin{bmatrix} \cos\phi\cos\theta\cos\varphi - \sin\phi\sin\varphi & -\cos\phi\cos\theta\sin\varphi - \sin\phi\cos\varphi & \cos\phi\sin\theta \\ \sin\phi\cos\theta\cos\varphi + \cos\phi\sin\varphi & -\sin\phi\cos\theta\sin\varphi + \cos\phi\cos\varphi & \sin\phi\sin\theta \\ -\sin\theta\cos\varphi & \sin\theta\sin\varphi & \cos\theta \end{bmatrix}$$

Equating these two matrices, we see that R_{33} equates to $\cos\theta$, so if we know R_{33} we can calculate θ. Now, if we know θ, we can look at the R_{31} element and calculate φ. In fact, we can get the same information from the R_{32} element, provided we know θ. Finally, we can use the R_{13} and R_{23} elements to calculate ϕ.

$$\begin{aligned} &R_{33} = \cos\theta \\ &R_{32} = \sin\theta\sin\varphi \qquad && R_{31} = -\sin\theta\cos\varphi \\ &R_{13} = \cos\phi\sin\theta \qquad && R_{23} = \sin\phi\sin\theta \end{aligned}$$

If the R_{33} element is non-zero, then we can go ahead and calculate θ. We calculate it by taking the inverse-cosine function. Of course, the inverse-cosine function has some ambiguity, and because of that we don't know the sin of θ. It could either be positive or negative. But modulo that ambiguity we can determine θ.

$$\theta = \sigma \arccos(R_{33}), \qquad \sigma = \pm 1$$

Once we know θ, we can calculate φ and ϕ using the inverse tangent function. Notice that we have two pieces of information for both φ and ϕ, and because of that, we don't use the standard inverse-tangent function. We use something called the **atan2 function**. The atan2 function allows us overcome the ambiguity that exists in inverse-tangent functions. Unlike the traditional inverse-tangent function, which only uses one equation to solve for an angle, here, we use the fact that we have two equations for the same angle. The inverse tangent function atan2 allows us to do this.

$$\varphi = a\tan 2\left(\frac{R_{32}}{\sin\theta}, \frac{-R_{31}}{\sin\theta}\right)$$

$$\phi = a\tan 2\left(\frac{R_{23}}{\sin\theta}, \frac{R_{13}}{\sin\theta}\right)$$

You will see from these equations that we have two sets of Euler angles, and this is true for almost all rotation matrices. These equations were derived, assuming that the magnitude of R_{33} is not equal to one. $|R_{33}| < 1$.

So what happens if the magnitude of R_{33} is equal to one? R_{33} can either be plus one or minus one. In these cases, the angle θ will be equal to either zero or pi. So we have two alternatives, but in both these cases, we can see from the grouping of terms, that the rotation matrices are functions only of the sum of two angles, ϕ and φ.

If $R_{33} = 1$

$$R = \begin{bmatrix} \cos\phi\cos\varphi - \sin\phi\sin\varphi & -\cos\phi\sin\varphi - \sin\phi\cos\varphi & 0 \\ \sin\phi\cos\varphi + \cos\phi\sin\varphi & -\sin\phi\sin\varphi + \cos\phi\cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

If $R_{33} = -1$

$$R = \begin{bmatrix} -\cos\phi\cos\varphi - \sin\phi\sin\varphi & \cos\phi\sin\varphi - \sin\phi\cos\varphi & 0 \\ \cos\phi\sin\varphi - \sin\phi\cos\varphi & \sin\phi\sin\varphi + \cos\phi\cos\varphi & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

In other words, it's impossible to determine, either ϕ or φ uniquely. Given one, we can determine the other. But it's impossible to disambiguate between ϕ and φ, because the groupings only contain terms that combine them. So when $R_{33} = \pm 1$ we have an infinite set of Euler angles.

In most of our work, we use a different set of Euler angles. These are the Z-X-Y Euler angles. Again, they're so called because the three rotation matrices we consider are the rotation about the z axis through φ, followed by the rotation about the x axis through ϕ and then the rotation of the y axis in θ.

Sequence of three rotations about **body-fixed** axes:

Rot(z,φ)
Rot(x,φ)
Rot(y,θ)

Verify by multiplying the rotations that we get a rotation matrix that has the form shown:

$$R = \begin{bmatrix} \cos\varphi\cos\theta - \sin\phi\sin\varphi\sin\theta & -\cos\phi\sin\varphi & \cos\varphi\sin\theta + \cos\theta\sin\phi\sin\varphi \\ \cos\theta\sin\varphi + \cos\varphi\sin\phi\sin\theta & \cos\phi\cos\varphi & \sin\varphi\sin\theta - \cos\theta\sin\phi\cos\varphi \\ -\cos\theta & \sin\phi & \cos\phi\cos\theta \end{bmatrix}$$

We've seen that for every set of Euler angles, we might have at least two solutions for Euler's angles for a given rotation matrix. At some points we can have infinite solutions. What we really want is a second set of Euler angles to take care of the points at which we have infinite solutions.

This suggests that we might have many, many sets of Euler angles that we might want to consider so that we don't have any points at which we have infinite solutions. So a question that is worth asking is what is the minimum number of sets of Euler angles we need to cover all of the rotation group?

Lecture 2A.4

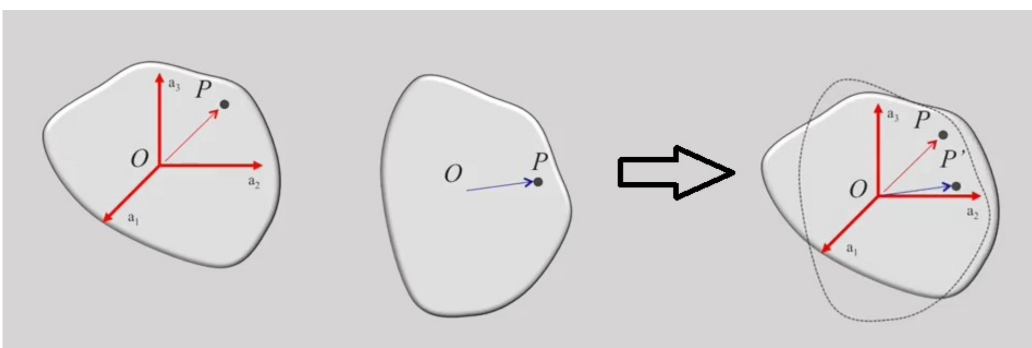
Axis-Angle Representations for Rotations

We've seen two different approaches to parameterising rotations. We've called these **coordinate systems**. For example, using rotation matrices, we can represent rotations with a set of nine numbers or nine coordinates. Those coordinates are redundant, so in reality they are not really coordinates. We've done the same thing with Euler angles where we get a set of three numbers or three coordinates. We're now going to look at a different way of looking at rotations. This involves explicitly representing the axis of rotation and the angle of rotation.

The key idea comes from a theorem by Euler. Euler said, that any displacement of a rigid body, such that a point on the body, O, remains fixed is equivalent to a rotation about a fixed axis through the point O.

What Euler is saying, is that any rotation is equivalent to a rotation about a fixed axis through a point O.

Let's see how to verify that. In order to do that, we're going to consider displacements where the point O is fixed. On the left we have a rigid-body in two positions and orientations. We're going to translate the rigid body from the second position and orientation, so that the two origins are identical as shown on the right:



Now the point P has moved to the new point P'. As we've seen before, the vector **OP** can be written as a linear combination of the basis vectors $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$:

$$\vec{OP} = p_1 a_1 + p_2 a_2 + p_3 a_3$$

Similarly, the point P' can also be written as a linear combination of $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$:

$$\vec{OP}' = q_1 a_1 + q_2 a_2 + q_3 a_3$$

So now we have two 3x1 vectors, the vector p and the vector q, that are related by the rotation matrix:

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

Or, $\mathbf{q}=\mathrm{R}\mathbf{p}$

What we'd like to do, is to ask if there's an axis that remains fixed under this rotation. In other words, is there a point P, such that, after the rotation through R, P stays fixed, i.e. the point P is invariant to the rotation?

This is the essence of the proof. So, going back to the basic equation where p gets rotated to q, we want to see if there is a point p that maps onto itself. If there is such a point, then point p would define the axis of rotation

Well if there was such a point p, we could write:

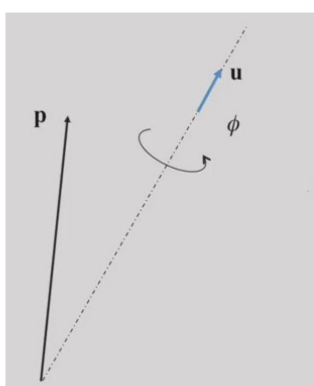
$$\mathbf{p} = \mathrm{R}\mathbf{p}$$

This is nothing but a statement that the eigenvalue of the rotation matrix R is one. We can solve the eigenvalue problem and verify that if $Rp = \lambda p$, then $\lambda = 1$ is a solution to this eigenvalue problem.

This is easy to verify and is a good exercise to try.

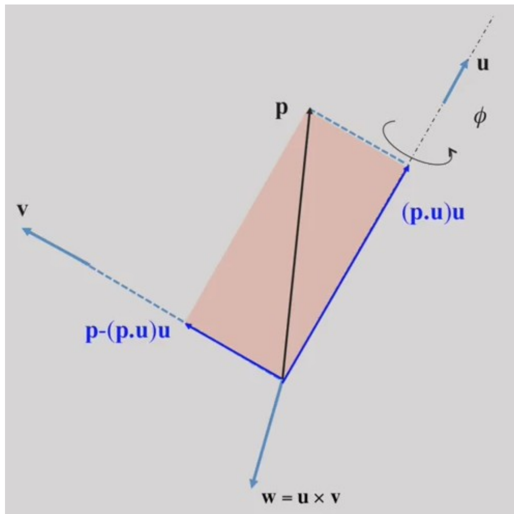
Now, let’s consider the inverse problem. Suppose we have an axis of rotation and an angle of rotation. How do we calculate the rotation matrix armed with these two pieces of information? (And note that we already know how to do this if the axis of rotation is the x axis or the y axis or the z axis - in fact, we discussed formulae for these very simple cases).

So, we have an arbitrarily oriented axis given by the unit vector **u** and an arbitrary angle of rotation given by the angle ϕ. Let's consider a generic vector **p,** and see how the vector p is rotated if the rigid body to which it's attached is rotated about the axis u through an angle ϕ.



First, let’s break this vector up into two components: One parallel to the vector **u**, and the other perpendicular to the vector **u**.

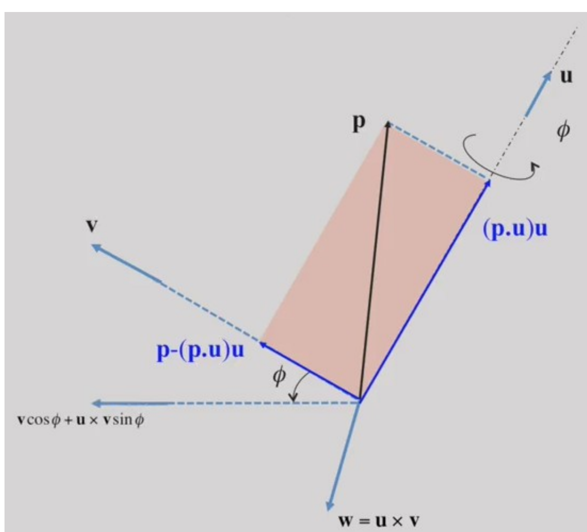
We can obtain the component parallel to the vector **u** by projecting **p** along **u** to get(**p**.**u**)**u**. Then, we can obtain the vector perpendicular to u by subtracting that projection, giving **p** - (**p**.**u**)**u**:



Obviously, the component of vector **p** that lies along the **u** axis is never changed by a rotation about **u**. In other words, the vector (**p**.**u**)**u** remains unchanged by the rotation R. But the perpendicular component, along **v** will be rotated.

We can now use the three vectors, **u**, **v** and the orthogonal vector **w**, where w is given by **u** × **v** as our local coordinate system or local basis.

As we observed above, the vector along **u** does not get rotated, but the vector along **v** does. This is what it looks like:



The resulting vector is given by simple trigonometry as:

$$v\cos\phi + (u \times v)\sin\phi$$

So, after rotation, the vector P consists of two new components. The one parallel to **u** is the same as before, but the one perpendicular to **u** has changed, and that is now given by $v\cos\phi + w\sin\phi$

Adding these components gives the new vector. We can derive an expression for the rotated **p**. That is given by:

$$R\bar{p} = \bar{p}\cos\phi + UU^T(1-\cos\phi)\bar{p} + (\bar{u} \times \bar{p})\sin\phi$$

Now, since **p** was chosen as a generic vector, this expression holds for any vector **p**.

We can to simplify this expression a little. Notice the term $\bar{u} \times \bar{p}$. This is obviously a cross product, but it can also be written as a matrix times **p**. We will call this matrix $\hat{u}$.

$\hat{u}$ is a skew-symmetric matrix, whose components are the same as the components of **u**.

So, we can say that the rotation of a generic vector is given by the right hand side of the formula we just derived:

$$R\bar{p} = \bar{p}\cos\phi + UU^T(1-\cos\phi)\bar{p} + \hat{u}\bar{p}\sin\phi$$

Removing the vector **p** from both sides essentially gives an expression that depends only on the axis of rotation and the angle of rotation. This is the rotation matrix:

$$Rot(u,\phi) = I\cos\phi + UU^T(1-\cos\phi) + \hat{u}\sin\phi$$

It consists of three terms. One is the identity times cosine ϕ. The second is $UU^T(1-\cos\phi)$, and the third is $\hat{u}\sin\phi$, where $\hat{u}$ is this skew-symmetric 3x3 matrix. This formula is called **Rodrigues' formula**.

So, given a generic vector **u**, and a rotation angle, ϕ, we can derive the rotation matrix using this closed-form expression. We can verify that the expression makes sense by testing it on some simple cases we've seen before.

For example, take u to be a unit vector along the x axis (or the y-, or z-axes). We can verify that the result is the same as Rot(x, ϕ): a rotation about the x axis through ϕ.

Let’s return to the question of whether the axis-angle representation is an *onto* representation? i.e. Is it 1-1?

It's *onto* if every rotation matrix can be represented by a combination of a vector **u** and an angle ϕ. It's one-to-one if there's a *unique* representation. Because we know that for every rotation there's an axis of rotation, Euler's theorem tells us that it's *onto*.

What about the 1-1 property?

Well, we can immediately see that this does not hold. It is clear that a rotation about the axis **u** through an angle phi is the same as a rotation about an axis pointing in the opposite direction through the angle $(2\pi - \phi)$ or $-\phi$. To verify it, just imagine rotating a body one way through an angle ϕ, and then the other way through an angle $-\phi$.

We can also substitute these values into Rodrigues' formula, and verify that we get the same rotation matrix.

But what if we restrict ϕ to the interval from zero to π? If we do that, then $-\phi$ will not be in the same interval as ϕ. Likewise $2\pi - \phi$ will not be in the same interval as ϕ. But does this make the map one to one?

One way to explore this question is to look at Rodrigues formula and ask ourselves whether, for a given rotation matrix, we can extract **u** and ϕ from the matrix? To do this, w can verify that if we take the trace of the rotation matrix, τ, which is essentially $R_{11} + R_{22} + R_{33}$, and we subtract one from it and divide by two, we will get $\cos\phi$:

$$\cos\phi = \frac{\tau - 1}{2}$$

This is obtained by directly applying Rodrigues' formula to get the trace. If this is true, we see from Rodrigues's formula that $\hat{u}$ can be obtained thus:

$$\hat{u} = \frac{1}{2\sin\phi}(R - R^T)$$

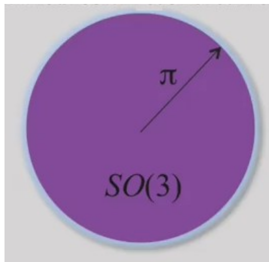
In this case we have obtained **u** without actually solving for the eigenvector. So is the axis angle rotation 1 to 1?

1. (axis-angle) to rotation matrix map is many-to-1.
2. Restricting the angle to the interval $[0 - \pi]$ makes it 1-to-1, except for:
    a. $\tau = 3$
    b. $\tau = -1$

The case where $\tau = 3$ is equivalent to $\cos\phi = 1$, which means that $\phi = 0$. If $\phi = 0$, there is no unique axis of rotation.

If the trace, τ, is equal to -1 then $\phi = \pi$. In this case we could have two axes of rotation: either the axis we obtained using the formula above, **u**, or the opposite, **-u**.

Perhaps the best way to think about the rotation group, SO(3), is by imagining a solid ball of radius π:



Any point on the ball's surface or inside the ball essentially represents a rotation whose angle is given by the radius. The vector that describes the point with the origin of the sphere as the origin of the vector tells you the axis of rotation.

Again, it seems like there's a one-to-one map between points on the surface of the ball or inside the ball and the set of rotations. This is true, again, except for points that are on the surface of the ball.

At any pair of points on the surface of the ball which are diametrically opposite each other essentially correspond to the same rotation.

Angular Velocity

Now that we're reasonably comfortable with the concept of rotations & displacements in general, it's time to start thinking about the rate of change of rotations. That leads us to the concept of an angular velocity vector. What does it mean to differentiate a rotation and get a velocity? We know the analogue for position vectors - differentiate a position vector to get a velocity. Now we want to take a 3x3 rotation matrix R(t) and differentiate that.

The first thing we should recognise is that this matrix is not just a bunch of numbers it's actually orthogonal & there are two relationships which govern the orthogonality:

$$R^T(t)R(t) = I$$
$$R(t)R^T(t) = I$$

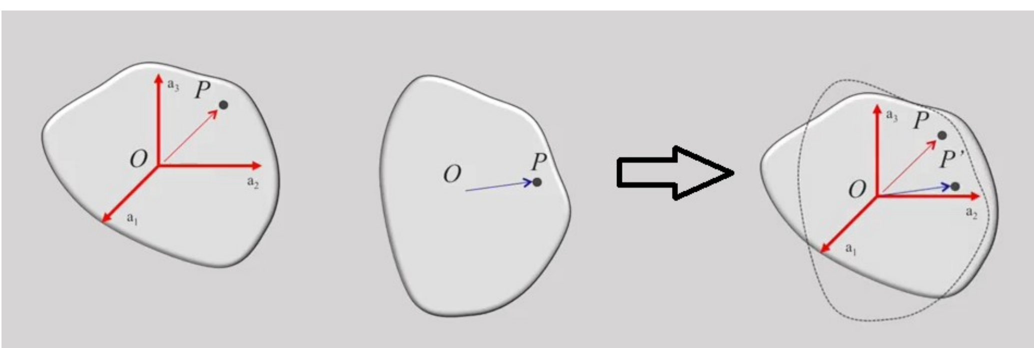
Differentiating both sides of each equation, using the product rule, gives two identities that relate the derivative of the rotation matrix and the transpose of a rotation matrix:

$$\dot{R}^T R + R^T \dot{R} = 0$$
$$R\dot{R}^T + \dot{R}R^T = 0$$

This tells us that $R^T \dot{R}$ (where $\dot{R}$ is a derivative of R) and $\dot{R}R^T$ are both skew-symmetric. In other words, there are only three independent elements in these skew-symmetric matrices.

So rather than think in terms of the derivative of a rotation matrix, we will think in terms of the derivative, pre-multiplied by R^T, or post-multiplied by R^T.

Let's return to our canonical example of a rigid-body with a generic position vector, p, and its rotated position, which we call **p'**. Once again, we'll ensure that the origins are the same, and we'll consider the displacement from P to P':

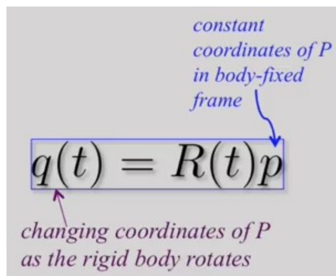


Recall that we use the position vector **p** to denote the coordinates of the point P, with respect to A_1, A_2, A_3, and the vector **q** to denote the coordinates of the point P' with respect to A_1, A_2, A_3. Remember that the vector **p** is a 3x1 vector that consists of coordinates of P in a body fixed frame.

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

$$\mathbf{Q}(t)=R(t)\mathbf{p}$$

Now if the rigid-body stays rigid, then vector **p** doesn't change. The only thing that changes as the body rotates is the vector **q**. As the rigid-body rotates, the rotation matrix changes as a function of time, and the vector **q** changes as a function of time, but the vector **p** stays constant.



If we differentiate both sides of this equation (and take advantage of the fact that **p** remains constant) we obtain:

$$\dot{q} = \dot{R}p$$

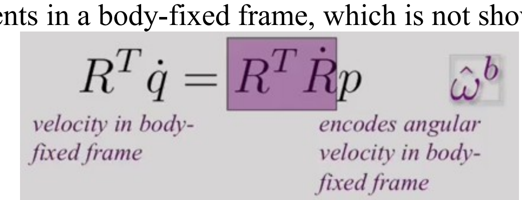
The only derivatives that appear are the derivatives of **q** and R.

As we've seen before, **p** is the position in a body fixed frame. $\dot{q}$, on the other hand, is the velocity in an *inertial* frame.

If we pre-multiply both sides by R^T we get a familiar form on the right hand side:

$$R^T\dot{q} = R^T\dot{R}p$$

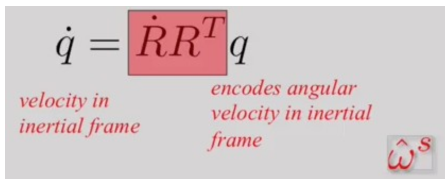
On the left-hand side, we have the velocity in the body-fixed frame, $\dot{q}$ - the velocity in the inertia frame has been transformed back to the body-fixed frame. And on the right-hand side, the familiar quantity $R^T\dot{R}$ encodes the angular velocity in a body-fixed frame. Let's call that $\hat{\omega}^b$ where the superscript b denotes that it's a body-fixed angular velocity. Again, it is the angular velocity of the rigid body, but we've chosen to write the components in a body-fixed frame, which is not shown here.



Another way of writing the equation relating the velocity in inertial frame and the position in the body-fixed frame is by rewriting **p** on the right-hand side in terms of R^T and **q**:

$$\dot{q} = \dot{R}R^T q$$

Let's take a look at the quantities on both sides of this equation. $\dot{q}$ is the velocity in the inertial frame. We've not transformed it back to the body-fixed frame. But on the right-hand side we see another familiar quantity, $\dot{R}R^T$, which is a skew-symmetric matrix. This encodes the angular velocity in the inertial frame. We call that $\hat{\omega}^s$ with the superscript s denoting the fact that it is not in the body-fixed frame but instead is a spatial angular velocity.



We've seen before that skew-symmetric matrices encode cross products. What we see in these two equations is essentially our ability to generate velocities by taking the cross product of an angular velocity vector with a position vector.

In the first equation, it's the angular velocity in the body-fixed frame which yields the velocity in the body-fixed frame. In the second equation, it's the angular velocity in the inertial frame which then yields the velocity in the inertial frame.

So we have two different representations of the same angular velocity vector. The first is written in terms of basis vectors on the body-fixed frame, and the second is written in terms of basis vectors in inertial frame.

Let's consider a simple rotation, a rotation about the z axis:

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This is easy to visualize, and it's also easy to write down. The transpose of R is given by:

$$R^T = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Next, we'll differentiate R with the respect to Time:

$$\dot{R} = \begin{bmatrix} -\sin(\theta) & -\cos(\theta) & 0 \\ \cos(\theta) & -\sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \dot{\theta}$$

So $\dot{R}$ only depends on the derivative of θ, so as this angle changes R changes and we can write $\dot{R}$ as a function of $\dot{\theta}$ by simply pre-multiplying it by a matrix that has mostly zeroes except for cos(θ) and sin(θ).

If we do the same computations for this very simple matrix and derive the expressions for $R^T \dot{R}$, and $\dot{R}R^T$ we find these two matrices are the same. And this happens in this very special case where axis of rotation is constant.

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R^T \dot{R} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\sin(\theta) & -\cos(\theta) & 0 \\ \cos(\theta) & -\sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \dot{\theta}$$

$$= \dot{R}R^T = \dot{\theta} \begin{bmatrix} -\sin(\theta) & -\cos(\theta) & 0 \\ \cos(\theta) & -\sin(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \dot{\theta} \qquad = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{\theta}$$

In this particular case, this skew-symmetric matrix corresponds to the [0, 0, 1]' vector which is the z-axis. This is something we should have expected. If we rotate a rigid-body about the z-axis and the axis of rotation is constant, then clearly, the angular velocity vector will also be along the z axis.

What happens if our rotation is obtained by composing two rotations? For example, $R=R_z(\theta)R_x(\phi)$.

In this case the rotations are about the z-axis by θ, and about the x-axis by ϕ. Now if we do the computations, we differentiate R and pre-multiply by R^T to get the body-fixed angular velocity:

$$\begin{aligned} \hat{\omega}^b = R^T \dot{R} &= (R_z R_x)^T (\dot{R}_z R_x + R_z \dot{R}_x) \\ &= R_x^T R_z^T \dot{R}_z R_x + R_x^T \dot{R}_x \end{aligned}$$

We get two terms. One depends only on the rate of change of R_z and the second that depends only on the rate of change of R_x. The same thing is true if I use the spatial angular velocity which is $\dot{R}R^T$. I get two terms, one that depends on the rate of change of R_z, and the second that depends on the rate of change of R_x.

$$\begin{aligned}\hat{\omega}^s = \dot{R}R^T &= (\dot{R}_z R_x + R_z \dot{R}_x)(R_z R_x)^T \\ &= \dot{R}_z R_z^T + R_z \dot{R}_x R_x^T R_z^T\end{aligned}$$

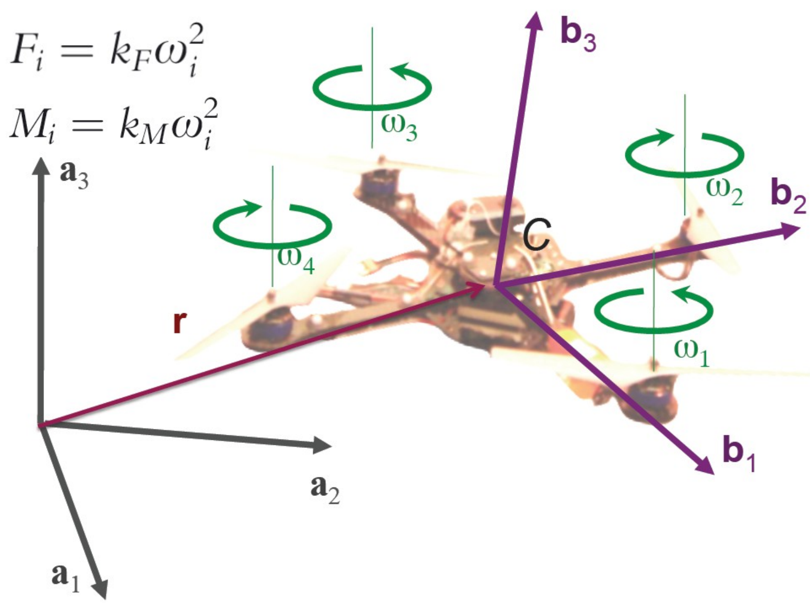
We can see that these two expressions are different. In both case they consist of two terms. One depending on $\dot{\theta}$, and the second depending on $\dot{\phi}$, but because the axis of rotation is not fixed the two expressions are different.

The body-fixed angular velocity and the spatial angular velocity have different expressions.

Formulation

Now that we've looked at positions and velocities, it's time to study the dynamics of a quadrotor. Again, we have two coordinate systems: one attached to the moving robot, and the other is the inertial coordinate system.

The body fixed coordinate system is described by the set of unit vectors b_1, b_2, and b_3. Similarly, the unit vectors a_1, a_2, and a_3 describe a coordinate system that's fixed to the inertial frame.



The robot has four rotors, each of which is independently actuated. **r** is the position vector of the centre of mass, and we know expressions for the thrust that the motors produce on the airframe:

$$F_i = k_F \omega_i^2$$

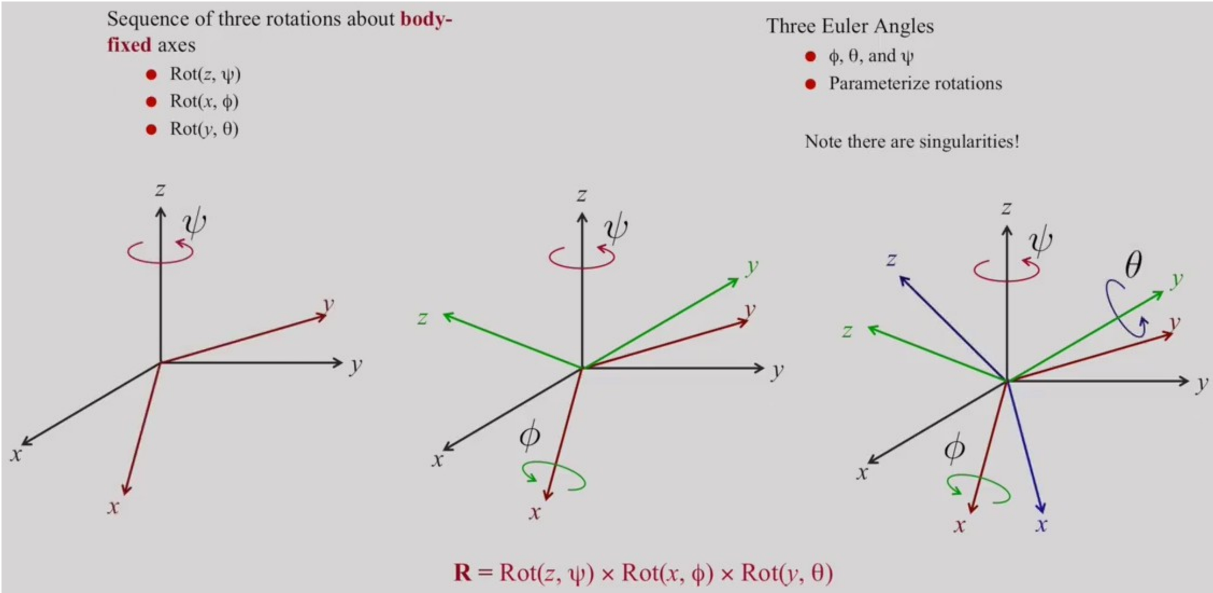
We also know the expressions for the reaction moments:

$$M_i = k_M \omega_i^2$$

Both of these are proportional to the square of the angular speeds of the rotors.

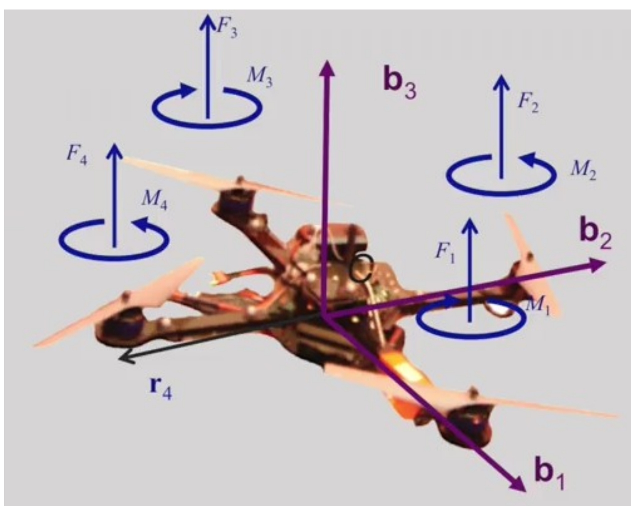
We've seen how Euler Angles are used to represent rotations. We will use the Z-X-Y convention, the first rotation being about the z-axis to φ, the second one about the x-axis, and the third about the y-axis. The first axis is yaw, the second is roll, and the third is pitch.

Again just to remind ourselves, this is the Z-X-Y convention:



The first rotation is about the z-axis through φ, the second is about the x-axis through ϕ, this is the roll angle, and finally, the pitch about the y-axis through θ. Of course, as we have seen, there are singularities. Singularities occur when the roll angle, $\phi = 0$. Also, even when the angle $\phi \neq 0$, we can have two sets of Euler angles for every rotation.

Let's look at the external forces and moments that act on the airframe. We have four thrusts: F_1, F_2, F_3, and F_4, and then four moments, M_1, M_2, M_3 and M_4.



The sum of the forces is obtained by adding up the thrust vectors and the gravity vector:

$$F = F_1 + F_2 + F_3 + F_4 + mg\bar{a}_3$$

The sum of the moments is obtained by adding up the reaction moments, as well as the moments of the thrust forces:

$$M = (r_1 \times F_1) + (r_2 \times F_2) + (r_3 \times F_3) + (r_4 \times F_4) + M_1 + M_2 + M_3 + M_4$$

Remember, these are vector computations. You have to add 3x1 vectors in each of these equations.

To predict the net acceleration, we have to write down the equations of motion. These come from Newton and Euler and they're called the Newton-Euler Equations.

Newton-Euler Equations

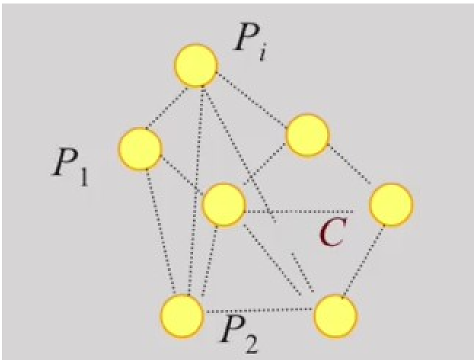
Most of us are familiar with Newton's equations of motion for a single particle of mass *m*:

$$\mathbf{F} = m\mathbf{a}$$

If we know the resultant force acting on a single particle of mass, *m*, then the acceleration is simply obtained from Newton's second law.

What does this equation look like when we consider a system of particles or a rigid -body?

Let's start with a system of particles. Before we can define Newton's Second Law for a system of particles, we must first define the centre of mass. The image below shows a set of particles. P_1, P_2, and so on.



Let's assume that we can write position vectors $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_i$, for each of these particles. Let's denote the mass of the i_{th} particle by M_i.

We can now compute the average position vector, by weighting each position vector with the appropriate mass.

$$r_c = \frac{1}{m}\sum_{i=1,N} m_i p_i$$

In this equation you see here, $m_i p_i$, is essentially the weighted sum of all the position vectors. We divide that by the total mass, *m*, to obtain a new position vector. This position vector defines the centre of mass.

It turns out that the centre of mass for a system of particles, S, behaves in exactly the same way as a single particle, with mass m (equal to the total mass of the system) would have behaved, if it had been located at the centre of mass.

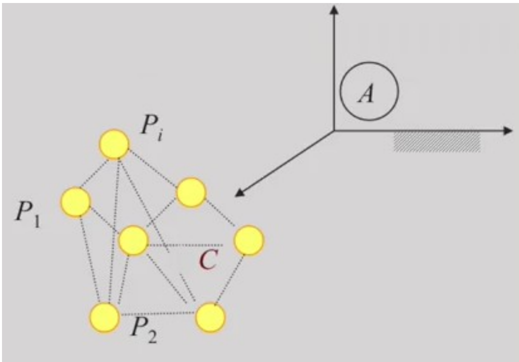
So instead of writing F = ma for a system of particles, we write F = m × the acceleration of the centre of mass:

$$F = \sum_{i=1}^{N} F_i = m \frac{{}^{A}d\,{}^{A}v^{C}}{dt}$$

We sum the individual forces to get F, and equate that to the total mass, m, times the derivative of the velocity of the centre of mass.

The notation probably requires some explanation.

1. The superscript A refers to the fact that we're computing these quantities in an inertial frame A.



2. The superscript C, refers to the fact that we're computing the velocity of the centre of mass.

Remember that the right-hand side is essentially the rate-of-change of linear momentum. If L the Linear Momentum is L, then the total force, F, is given by:

$$F = \frac{{}^{A}dL}{dt}$$

This is Newton's second law for a system of particles. Again the derivative of each of these quantities, must be performed in an inertial frame in order for this equation to be valid. This is also true for a rigid body.

One way to think of a rigid-body is to consider it as simply an infinite set of particles, which are all glued together rigidly. So if this is valid for a set of particles, it must also be valid for an infinite collection of particles.

So that was Newton's second law. What are the rotational laws of motion for a rigid-body?

Let's derive the rotational equations of motion for a rigid body. For linear motion we considered the linear momentum of the rigid body and it's derivative. For rotational motion, the analogous quantity to consider is the angular momentum.

The rate of change of angular momentum of a rigid-body B, relative to the centre-of-mass, C, in an inertial frame {A} is equal to the resultant moment of all external forces acting on the body relative to C:

$$\frac{{}^{A}d\,{}^{A}H_{C}^{B}}{dt} = M_{C}^{B}$$

Where H is the angular momentum of a rigid body with origin C (the centre-of-mass), in an inertial frame {A}. We want to calculate the rate-of-change of angular momentum of the rigid-body B. That equals the net moment applied to the rigid-body. Once again, the differentiation must be done in an inertial frame.

In order to actually perform the calculation on the left-hand side, we have to replace H with something that we can easily measure. It turns out that the angular momentum is nothing but the body's inertia times its angular velocity.

$${}^{A}H_{C}^{B} = I_{C}\,{}^{A}\omega^{B}$$

These computations have to be done in three dimensions. The angular momentum and the angular velocity are 3-dimensional vectors. The quantity in between is the inertia tensor, whose components can be written as a 3x3 matrix. Here, the subscript C denotes the fact that we measured the components with the centre-of-mass, C, as the origin. The angular velocity is also obtained in the inertial frame, and the leading superscript A captures this. The trailing superscript B, tells us that this is the angular velocity of the rigid body B that we are measuring.

Finally, M is the net moment. Take all the external forces, compute their moments, and add these moments to all external couples or torques. Once again, the trailing subscript C reminds us that we're computing moments with the centre-of-mass, C, as the origin.

Lecture 2C.3

Principal Axes and Principal Moments of Inertia

Before we can write down the rotational equations of motion we have to define principle axes and moments of inertia.

A principal axis of inertia is defined by a direction. Let's say U is that direction.

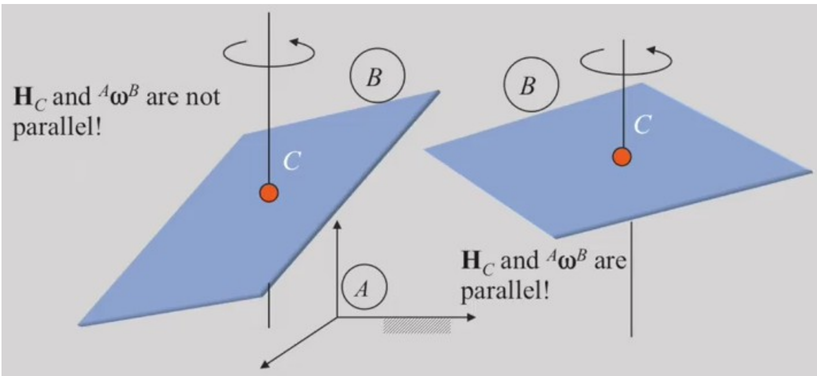
- **u** is a unit vector along a principal axis if **I**.**u** is parallel to **u**.

There's a theorem that says that says we can always find three such independent principle axes. In other words there are three independent axes such that **I** times a unit vector along that axis will give a vector that's parallel to that axis.

The moment of inertia, **I**, is essentially a scaling term. If we take I .u and it differs from u by a scalar factor, that scalar factor is the moment of inertia. These moments of inertia are called **principal moments of inertia**.

- The moment-of-inertia with respect to a principal axis, **u**.**I**.**u**, is called a **principal moment of inertia**.

Here's a simple example that illustrates what the moments of inertia tell us:



We have a rigid frame, {A}, and a parallel plate that's spinning about a vertical axis. The plate is shown in two different configurations.

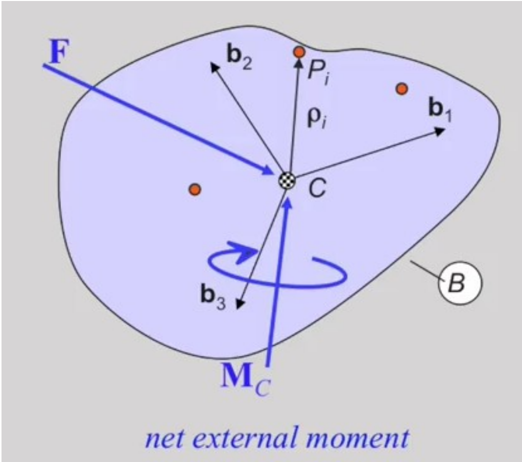
On the right side, the axis is perpendicular to the plate. In fact this configuration is symmetric. If we compute the angular momentum, we find that the angular-momentum vector and the angular-velocity vector are parallel.

On the left side, the configuration is symmetric, but the axis is not perpendicular to the plate. In this configuration, the angular-momentum vector and the angular-velocity vector are not parallel. This is because the axis of rotation does not coincide with any of the principal axes.

Now we can deal with Euler's Equations which will tell us the rotational equations of motion. Once again, it comes down to the basic observation that the rate-of-change of angular-momentum is equal to the net moment applied to the rigid body.

We take C, the centre-of-mass, as the origin for all our calculations.

In the figure below, b_1, b_2 and b_3 are a set of body-fixed unit vectors that define a body-fixed frame. We'll now specify that these vectors point along principal axes and we'll write our angular velocity as linear combinations of b_1, b_2, and b_3, and the components are ω_1, ω_2, and ω_3:



$$^{A}\omega^{B} = \omega_1 b_1 + \omega_2 b_2 + \omega_3 b_3$$

There are two key aspects to what we are doing:

1. We’re taking C, the centre-of-mass, as the origin.
2. We're requiring that the body-fixed frame in which we'll do our calculations will be along the three principal axes.

We can take the basic equation we derived in the last lecture:

$$\frac{^{A}d\,{}^{A}H_C^B}{dt} = M_C^B$$

And break the left-hand side two terms:

$$\frac{^{B}d\ H_C}{dt} + {}^{A}\omega^{B} \times H_C = M_C$$

The first term involves the derivative in a body fixed frame. The second term involves a “correction factor”, which is a vector that takes into account the fact that the differentiation is done in the body fixed frame. This correction factor is simply the angular velocity of the moving body-fixed frame crossed with the angular momentum.

This "correction factor", as we've referred to it, is actually a well-known fact in mechanics. Anytime you differentiate a vector in a moving frame, its derivative is different from the derivative in a fixed frame. That difference is obtained by simply factoring in the cross product of the angular velocity with that vector.

The first term on the left-hand side can be written in terms of inertia matrix times the angular velocity vector. Because we have chosen principle axes, it turns out that the off-diagonal elements in the inertia tensor are zero. Therefore, the first term which involves I × ω (the inertia tensor times the angular-velocity vector) consists of three terms:

The diagonal terms I_{11}, I_{22}, I_{33}, which multiply with ω_1, ω_2, and ω_3, yield the three terms that we see on the right hand side:

$$\frac{{}^{B}d\,H_C}{dt} = I_{11}\dot{\omega}_1 b_1 + I_{22}\dot{\omega}_2 b_2 + I_{33}\dot{\omega}_3 b_3$$

The ω × H term can also be written in component form, and that gives us the following matrix equation:

$$\begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix}\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} + \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}\begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} M_{C,1} \\ M_{C,2} \\ M_{C,3} \end{bmatrix}$$

These are Euler's Equations of Motion. They are quite compact.

The first term is essentially the derivative of the angular momentum in a body-fixed frame, and the second term is the correction. The term on the right-hand side, is the net moment.

In what follows we will use **p**, **q** and **r** to denote the components of the angular velocity vector along $\mathbf{b}_1$, $\mathbf{b}_2$, and $\mathbf{b}_3$.

Quadrotor Equations of Motion

Now let's return to the Quadrotor dynamics. These equations tell us the net force and the net moment:

$$F = F_1 + F_2 + F_3 + F_4 - mg\vec{a}_3$$

$$M = (r_1 \times F_1) + (r_2 \times F_2) + (r_3 \times F_3) + (r_4 \times F_4) + M_1 + M_2 + M_3 + M_4$$

If we combine the net force and net moment with the Newton-Euler Equations we get these two sets of equations:

$$m\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix}$$

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

On the right side of the first equation, we have the total thrust which is $\mathbf{u_1}$ (this thrust vector is known in the body-fixed frame). The matrix R is rotating this thrust vector to an inertial frame.

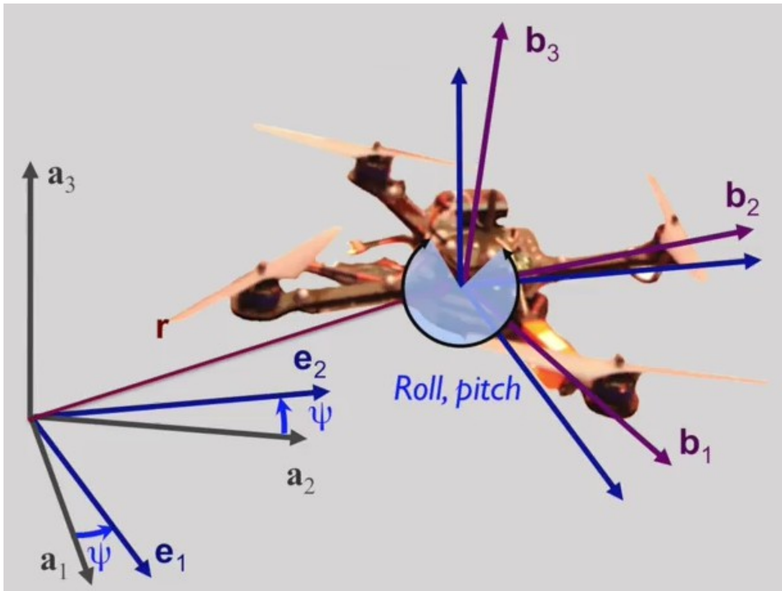
At the bottom you see the net moment, also known in the body fixed frame.

The equations as they've been written have components in the inertial frame on the top, and in the body-fixed frame at the bottom. These are the Newton-Euler equations, and these are the equations we'll use to develop controllers and planners for our vehicles.

A reasonable question to ask is: how do we actually calculate these parameters. Or in an online setting, how do we estimate these parameters.

In fact, the parameters we really need to know are those corresponding to the geometry, such as the length, L, for example, and those corresponding to the physical properties, like the mass, m and inertia, I. These parameters appear linearly in these equations, so, if we have a system that allows us to measured positions, velocities, and accelerations, it's actually not that hard to estimate the lengths, masses and inertias.

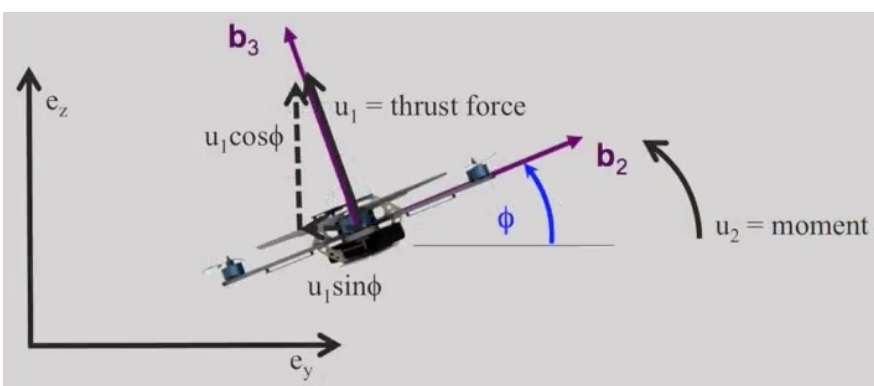
It is worth verifying that it's quite easy to calculate the angular velocity in the body-fixed frame. If you know the pitch, roll, and yaw angles, and also the rate-of-change of the pitch, roll, and yaw angles (on the right of the equation below), a simple transformation yields the angular velocity components p, q, and r along b_1, b_2, and b_3.



$$\begin{array}{ll} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\cos\phi\sin\theta \\ 0 & 1 & \sin\phi \\ \sin\theta & 0 & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} & \begin{matrix} \text{Roll} \\ \text{Pitch} \\ \text{Yaw} \end{matrix} \end{array}$$

This model is quite complicated. It involves three components of position, velocity, and acceleration, and three components of rotations, angular-velocities, and angular -accelerations. To get a feel for the control problem, let's look fist at the Planar version of the model.

We start by looking at the equations of motion in the Y-Z plane. We will assume that the robot cannot move out of this plane or, in other words, that there is no motion in the x-direction. We will also assume that there are no yaw or pitch motions:



$$\begin{bmatrix} \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{m}\sin\phi & 0 \\ \frac{1}{m}\cos\phi & 0 \\ 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

In this configuration, we come up with the three equations that you see here:

$$\begin{bmatrix} \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{m}\sin\phi & 0 \\ \frac{1}{m}\cos\phi & 0 \\ 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

These equations describe the rates of change of velocity in the y and z directions, and the rate of change of the angular-velocity in the roll direction.

To describe these kinds of systems it is useful to define a state vector. In the three-dimensional case, we have six variables that describe the configuration of a robot, and a state vector that includes: the configuration and its derivative:

$$q = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \varphi \end{bmatrix}, x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$$

If Q is the six-dimensional configuration vector, q and $\dot{q}$ constitute a 12 dimensional description off the state. The space of all such state vectors is called a state space.

If we look at the equilibrium configuration, and that configuration is defined by the position x_0, y_0, z_0, the configuration by definition also has a zero roll angle and a zero pitch angle. It could, of course, have any yaw angle. The equilibrium configuration must also correspond to zero velocity. If we write it down in terms of a state space vector, we have the equilibrium configuration q_0 and a zero-velocity vector:

$$q_0 = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 0 \\ 0 \\ \varphi \end{bmatrix}, x_e = \begin{bmatrix} q_e \\ 0 \end{bmatrix}$$

This gives a 12-dimensional vector, where the bottom six elements are all 0.

Similarly, for planar quadrotors, we have a three dimensional configuration space, a six-dimensional state-space, and equilibrium state that can be similarly defined:

Planar Quadrotor

State Vector

- q describes the configuration (position) of the system
- x describes the state of the system

$$q = \begin{bmatrix} y \\ z \\ \varphi \end{bmatrix}, \; x = \begin{bmatrix} q \\ \hdashline \dot{q} \end{bmatrix}$$

Equilibrium at Hover

- q_e describes the equilibrium configuration of the system
- x_e describes the equilibrium state of the system

$$q_e = \begin{bmatrix} y_0 \\ z_0 \\ 0 \end{bmatrix}, \; x_e = \begin{bmatrix} q_e \\ \hdashline 0 \end{bmatrix}$$