`quadcopter` package

dinay-kingkiller

I. Introduction

The `quadcopter` package is filled with obtuse and often confusing code derived from various physics equations and mathematical formulas. This paper hopes to bridge the gap between quadcopter theory and the `src` files included. This paper can also be used as a reference for later expansion of this package.

II. The Model Node

While an actual quadcopter would be helpful, a simulation is the next best thing. The model node simulates the physics of a quadcopter. The physics in this section also provide insight on how motor input and sensor output behave.

A. Rotational Kinetics

Quaternions are integral in understanding the implementation of the model node. Firstly, a vector in the inertial frame N can be transformed and expressed in the robot frame R by the quaternion q:

$$\mathbf{v}^R = \mathrm{q}\mathbf{v}^N\mathrm{q}^* \tag{1}$$

That quaternion derivative can be updated via the angular velocity from robot frame to inertial frame $\boldsymbol{\omega}$ via

$$\dot{\mathrm{q}} = \frac{1}{2}\boldsymbol{\omega}\mathrm{q} \tag{2}$$

Integrating over the unit 4-sphere:

$$\mathrm{q}[k+1] = \exp\left(\frac{1}{2}\boldsymbol{\omega}[k]\Delta t\right)\mathrm{q}[k] \tag{3}$$

$$\mathrm{q}_s = \cos\left(\frac{|\boldsymbol{\omega}|\,\Delta t}{2}\right)$$
$$\mathrm{q}_v = \boldsymbol{\omega}\sin\left(\frac{|\boldsymbol{\omega}|\,\Delta t}{2}\right)$$

$$q_0[k+1] = -\exp\left(\frac{1}{2}\omega_0[k]\right) \tag{4}$$

Throughout this paper, I use notation of the sum of scaled unit vectors (versors), but coding is better described as `floats`. The dot product of vectors provides these scalars. The rotation matrices provide a way to move between these `arrays` of `floats`. For example, consider the rotation matrix from basis A to B.

$$\begin{pmatrix}\mathbf{v}\cdot\hat{\mathbf{b}}_x\\ \mathbf{v}\cdot\hat{\mathbf{b}}_y\\ \mathbf{v}\cdot\hat{\mathbf{b}}_z\end{pmatrix} = {}^{\mathrm{B}}R^{\mathrm{A}}\begin{pmatrix}\mathbf{v}\cdot\hat{\mathbf{a}}_x\\ \mathbf{v}\cdot\hat{\mathbf{a}}_y\\ \mathbf{v}\cdot\hat{\mathbf{a}}_z\end{pmatrix} \tag{5}$$

TABLE I
Transforms from the inertial frame N
($s_x = \sin x$, $c_x = \cos x$)

${}^{\mathrm{U}}R^{\mathrm{N}}$	$\hat{\mathbf{n}}_x$	$\hat{\mathbf{n}}_y$	$\hat{\mathbf{n}}_z$
$\hat{\mathbf{a}}_x$	1	0	0
$\hat{\mathbf{a}}_y$	0	c_a	s_a
$\hat{\mathbf{a}}_z$	0	$-s_a$	c_a
$\hat{\mathbf{b}}_x$	c_b	$s_a s_b$	$-s_a c_a$
$\hat{\mathbf{b}}_y$	0	c_a	s_a
$\hat{\mathbf{b}}_z$	s_b	$-s_a c_b$	$c_a c_b$
$\hat{\mathbf{c}}_x$	$c_b c_c$	$-c_a s_c + s_a s_b c_c$	$-s_a s_c - c_a s_b c_c$
$\hat{\mathbf{c}}_y$	$c_b s_c$	$c_a c_c + s_a s_b s_c$	$s_a c_c - c_a s_b s_c$
$\hat{\mathbf{c}}_z$	s_b	$-s_a c_b$	$c_a c_b$

The rotation matrices also collect the dot products to push final calculations down the road.

$${}^{\mathrm{B}}R^{\mathrm{A}} = \begin{pmatrix}\hat{\mathbf{a}}_x\cdot\hat{\mathbf{b}}_x & \hat{\mathbf{a}}_y\cdot\hat{\mathbf{b}}_x & \hat{\mathbf{a}}_z\cdot\hat{\mathbf{b}}_x\\ \hat{\mathbf{a}}_x\cdot\hat{\mathbf{b}}_y & \hat{\mathbf{a}}_y\cdot\hat{\mathbf{b}}_y & \hat{\mathbf{a}}_z\cdot\hat{\mathbf{b}}_y\\ \hat{\mathbf{a}}_x\cdot\hat{\mathbf{b}}_z & \hat{\mathbf{a}}_y\cdot\hat{\mathbf{b}}_z & \hat{\mathbf{a}}_z\cdot\hat{\mathbf{b}}_z\end{pmatrix} \tag{6}$$

The rotation of the quadcopter can be described by a composition of three rotations: from a Newtonian frame N to A, by an angle a around their common x axis (the *roll*); from intermediate frames A to B, by angle b around their common y axis (the *pitch*); and from B to the quadcopter body frame C, by angle c around their common z axis (the *yaw*). The angles and bases are chosen to make the angular velocities positive. On the quadcopter frame $\hat{\mathbf{c}}_x$ points to the front motor, $\hat{\mathbf{c}}_y$ points to the left motor, and $\hat{\mathbf{c}}_z$ is the cross product of the two.

$$\begin{aligned}
{}_{\mathrm{N}}\boldsymbol{\omega}_{\mathrm{A}} &= \dot{a}\hat{\mathbf{n}}_x &&= \dot{a}\hat{\mathbf{a}}_x && (7)\\
{}_{\mathrm{A}}\boldsymbol{\omega}_{\mathrm{B}} &= \dot{b}\hat{\mathbf{a}}_y &&= \dot{b}\hat{\mathbf{b}}_y && (8)\\
{}_{\mathrm{B}}\boldsymbol{\omega}_{\mathrm{C}} &= \dot{c}\hat{\mathbf{b}}_z &&= \dot{c}\hat{\mathbf{c}}_z && (9)
\end{aligned}$$

Using the notation of the rotation matrices above, the composition of rotation matrices is the matrix product.

$${}^{\mathrm{C}}R^{\mathrm{N}} = \left({}^{\mathrm{C}}R^{\mathrm{B}}\right)\left({}^{\mathrm{B}}R^{\mathrm{A}}\right)\left({}^{\mathrm{A}}R^{\mathrm{N}}\right) \tag{10}$$

The important combinations of dot products can be found in tables **??** and **??**.

B. Linear Equations of Motion

If you choose the right reference frames the linear equations of motion are easy to express. First, the motion of the quadcopter from its origin point is defined simply:

$$\begin{aligned}
\mathbf{r}_{c/o} &= x\hat{\mathbf{n}}_x + y\hat{\mathbf{n}}_y + z\hat{\mathbf{n}}_z && (11)\\
{}_{\mathrm{N}}\mathbf{v}_c &= \dot{x}\hat{\mathbf{n}}_x + \dot{y}\hat{\mathbf{n}}_y + \dot{z}\hat{\mathbf{n}}_z && (12)\\
{}_{\mathrm{N}}\mathbf{a}_c &= \ddot{x}\hat{\mathbf{n}}_x + \ddot{y}\hat{\mathbf{n}}_y + \ddot{z}\hat{\mathbf{n}}_z && (13)
\end{aligned}$$

TABLE II
TRANSFORMS FROM THE ROBOT FRAME C
($s_x = \sin x$, $c_x = \cos x$)

${}^{\mathrm{U}}R^{\mathrm{C}}$	$\hat{\mathbf{c}}_x$	$\hat{\mathbf{c}}_y$	$\hat{\mathbf{c}}_z$
$\hat{\mathbf{b}}_x$	c_c	s_c	0
$\hat{\mathbf{b}}_y$	$-s_c$	c_c	0
$\hat{\mathbf{b}}_z$	0	0	1
$\hat{\mathbf{a}}_x$	$c_b c_c$	$c_b s_c$	s_b
$\hat{\mathbf{a}}_y$	$-s_c$	c_c	0
$\hat{\mathbf{a}}_z$	$s_b c_c$	$-s_b s_c$	c_b
$\hat{\mathbf{n}}_x$	$c_b c_c$	$c_b s_c$	s_b
$\hat{\mathbf{n}}_y$	$-c_a s_c + s_a s_b s_c$	$c_a c_c + s_a s_b s_c$	$-s_a c_b$
$\hat{\mathbf{n}}_z$	$-s_a s_c - c_a s_b c_c$	$s_a c_c - c_a s_b s_c$	$c_a c_b$

For our model we assume each motor puts out a force proportional to its square. If we let m be the total mass of the robot and k be the proportionality constant:

$$m_{\mathrm{N}}\mathbf{a}_c = k\left(\omega_f^2 + \omega_l^2 + \omega_b^2 + \omega_r^2\right)\hat{\mathbf{c}}_y - mg\hat{\mathbf{n}}_y \tag{14}$$

Where f, l, b, and r stand for the front, right, back and left motors respectively. Expressed in the inertial frame N the equations of motion are:

$$\begin{aligned}
\ddot{x} &= \frac{k}{m}\Omega\cos b\sin c && (15)\\
\ddot{y} &= \frac{k}{m}\Omega\cos a\cos c + \frac{k}{m}\Omega\sin a\sin b\sin c - g && (16)\\
\ddot{z} &= \frac{k}{m}\Omega\sin a\cos c - \frac{k}{m}\Omega\cos a\sin b\sin c && (17)
\end{aligned}$$

where Ω is the sum of the squares of the motor inputs.

III. ROTATIONAL MOMENTUM

Supposing that most of the mass of the quadcopter is in the motors, the inertia dyadics for the four motors is:

$$\begin{aligned}
\mathbf{I}^{f/c} &= \frac{mL^2}{4}\hat{\mathbf{c}}_y\hat{\mathbf{c}}_y + \frac{mL^2}{4}\hat{\mathbf{c}}_z\hat{\mathbf{c}}_z && (18)\\
\mathbf{I}^{l/c} &= \frac{mL^2}{4}\hat{\mathbf{c}}_x\hat{\mathbf{c}}_x + \frac{mL^2}{4}\hat{\mathbf{c}}_z\hat{\mathbf{c}}_z && (19)\\
\mathbf{I}^{b/c} &= \frac{mL^2}{4}\hat{\mathbf{c}}_y\hat{\mathbf{c}}_y + \frac{mL^2}{4}\hat{\mathbf{c}}_z\hat{\mathbf{c}}_z && (20)\\
\mathbf{I}^{r/c} &= \frac{mL^2}{4}\hat{\mathbf{c}}_x\hat{\mathbf{c}}_x + \frac{mL^2}{4}\hat{\mathbf{c}}_z\hat{\mathbf{c}}_z && (21)
\end{aligned}$$

The total inertial dyadic of the robot is then

$$\mathbf{I}^{R/c} = \frac{mL^2}{2}\left(\hat{\mathbf{c}}_x\hat{\mathbf{c}}_x + \hat{\mathbf{c}}_y\hat{\mathbf{c}}_y + 2\hat{\mathbf{c}}_z\hat{\mathbf{c}}_z\right) \tag{22}$$

The angular velocity is the compositions of simple velocities given in equations **??**-**??**.

$${}_{\mathrm{N}}\boldsymbol{\omega}_C = \dot{a}\hat{\mathbf{a}}_x + \dot{b}\hat{\mathbf{b}}_y + \dot{c}\hat{\mathbf{c}}_z \tag{23}$$

And the angular momentum is the dot product of the two

$$\begin{aligned}
{}^{\mathrm{N}}\mathbf{H}^{R/c} &= \mathbf{I}^{R/c}\cdot{}_{\mathrm{N}}\boldsymbol{\omega}_{\mathrm{C}}\\
&= \frac{mL^2}{4}\left(\dot{a}\cos b\cos c - \dot{b}\sin c\right)\hat{\mathbf{c}}_x\\
&\quad + \frac{mL^2}{4}\left(\dot{a}\cos b\sin c + \dot{b}\cos c\right)\hat{\mathbf{c}}_y\\
&\quad + \frac{mL^2}{4}\left(\dot{a}\sin b + 2\dot{c}\right)\hat{\mathbf{c}}_z && (24)
\end{aligned}$$

The angular acceleration (in the inertial frame) can be found with some manipulation of derivatives in reference frames.

$$\begin{aligned}
{}_{\mathrm{N}}\boldsymbol{\alpha}_{\mathrm{C}} &= \frac{{}^{\mathrm{N}}\mathrm{d}}{\mathrm{d}t}{}_{\mathrm{N}}\boldsymbol{\omega}_{\mathrm{C}}\\
&= \frac{{}^{\mathrm{N}}\mathrm{d}}{\mathrm{d}t}\dot{a}\hat{\mathbf{n}}_x + \frac{{}^{\mathrm{A}}\mathrm{d}}{\mathrm{d}t}\dot{b}\mathbf{a}_y + {}_{\mathrm{N}}\boldsymbol{\omega}_{\mathrm{A}}\times\dot{b}\hat{\mathbf{a}}_y\\
&\quad + \frac{{}^{\mathrm{B}}\mathrm{d}}{\mathrm{d}t}\dot{c}\hat{\mathbf{b}}_z + {}_{\mathrm{N}}\boldsymbol{\omega}_{\mathrm{B}}\times\dot{c}\hat{\mathbf{b}}_z\\
&= \ddot{a}\hat{\mathbf{n}}_x + \ddot{b}\hat{\mathbf{a}}_y + \ddot{c}\hat{\mathbf{c}}_z\\
&\quad + \dot{a}\dot{b}\hat{\mathbf{a}}_z + \dot{b}\dot{c}\hat{\mathbf{b}}_x + \dot{a}\dot{c}\left(\hat{\mathbf{n}}_x\times\hat{\mathbf{c}}_z\right) && (25)
\end{aligned}$$

The moment from the reference frame $\mathbf{M}^* = \mathbf{I}\cdot\boldsymbol{\alpha} + \boldsymbol{\omega}\times\mathbf{H}$ was calculated with a script

$$\begin{aligned}
\mathbf{M}^*\cdot\hat{\mathbf{c}}_x &= \ddot{a}\cos b\cos c + \ddot{b}\sin c - \dot{a}\dot{b}\sin b\cos c\\
&\quad - \dot{a}\dot{c}\sin c\cos b + \dot{b}\dot{c}\cos c && (26)\\
\mathbf{M}^*\cdot\hat{\mathbf{c}}_y &= -\ddot{a}\sin c\cos b + \ddot{b}\cos c + \dot{a}\dot{b}\sin b\sin c\\
&\quad - \dot{a}\dot{c}\cos b\cos c - \dot{b}\dot{c}\sin c && (27)\\
\mathbf{M}^*\cdot\hat{\mathbf{c}}_z &= \ddot{a}\sin b + \ddot{c} + \dot{a}\dot{b}\cos b && (28)
\end{aligned}$$

APPENDIX
QUATERNION KINEMATICS

Quadcopters hover, and rotate, and strafe, and sometimes flip. Its important to track any of these movements smoothly, and hopefully without adding code to handle edge cases. Quaternions are one such valuable representation.