3/15 Recitation

- quiz results
- review rec 7
- quat math
- quat vs DCM (rotation matrix)

quiz stuff:

$$\min_x \; f(x)$$

$$\text{s.t.} \quad c(x) = 0$$
$$g(x) \leq 0$$

any of these are non convex
our problem is nonconvex

expectations	Convex	nonconvex
guarantee of global solution	✓	✗
guarantee of local solution	✓	✗
find a feasible solution if one exists	✓	✗
we get a solution in a reliable amount of time	✓	✗
initialization doesn't matter (for being able to solve)	✓	✗ ✗ ✗ ✗ ✗

Due to the results on the last quiz, I went through some of the guarantees we can expect with convex/nonconvex optimization. With nonconvex optimization, there are basically no guarantees, but it can work well in practice if we are careful.

quat math as matrix math

earlier:

- $q_1 \odot q_2 = L(q_1)\, q_2 = R(q_2)\, q_1$
- $L(q)^T = L(q^+)$
- $R(q)^T = R(q^+)$
- $\underset{\uparrow \atop q}{\begin{bmatrix} s \\ v \end{bmatrix}}^+ = \begin{bmatrix} s \\ -v \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & & & \\ & -1 & & \\ & & -1 & \\ & & & -1 \end{bmatrix}}_{T} \begin{bmatrix} s \\ v \end{bmatrix}$
- $\hat{a} = \begin{bmatrix} 0 \\ a \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{H} a$
- $a = H^T \hat{a}$

Here are all the quaternion operations done in normal matrix-vector matrix-matrix products. This is nice when it comes to manipulating these equations, taking derivatives etc.

Rotate a vector w/ quaternion

(a) $x_N = {}^NQ^B x_B$

(b) $\hat{x}_N = {}^NQ^B \hat{x}_B ({}^NQ^B)^T$

2 options for this ((a) is way faster)

$$\hat{x}_N = {}^Nq^B \odot \hat{x}_B \odot ({}^Nq^B)^+$$

$$H x_N = (L({}^Nq^B) H x_B) \odot ({}^Nq^B)^+$$

$$H x_N = R({}^Nq^B)^T L({}^Nq^B) H x_B$$

$$x_N = \underbrace{H^T R({}^Nq^B)^T L({}^Nq^B) H}_{3\times 3 \text{ rotation matrix!}} x_B$$

Here is how to get a DCM from a quaternion. There are at least 5 different formulas out there that are all equivalent, just be careful it's not the transpose of what you think it should be.

$$\boxed{{}^NQ^B = H^T R({}^Nq^B)^T L({}^Nq^B) H}$$

DCM_from_quat

Kinematics

translation: acceleration $\in \mathbb{R}^3$ $\xrightarrow{\int}$ velocity $\in \mathbb{R}^3$ $\xrightarrow{\int}$ position $\in \mathbb{R}^3$

attitude: angular accel $\alpha \in \mathbb{R}^3$ $\xrightarrow{\int}$ angular velocity $\omega \in \mathbb{R}^3$ $\xrightarrow{\int}$ nonsense $\otimes$

Here is the source of all our attitude heartache. When we have accelerations, we can directly integrate them to velocities, and integrate those velocities into positions. They are all vectors in R3 and we just simply integrate up the chain.

With attitude, we can integrate angular acceleration to angular velocity, but if we integrate angular velocity we get nothing useful. There are many crazy geometrical/math justifications for this, I have found the simplest conceptual explanation is that attitude "wraps" on itself. So when I spin 360 degrees in the same direction, I return to my original attitude. There is nothing in the translation world that mimics this behavior.

Kinematics: config = f(config, velocity)

DCM: $${}^N\dot{Q}^B = {}^NQ^B[{}^N\hat{\omega}^B_B]$$

No assumptions on N, B

quat: $${}^N\dot{q}^B = \frac{1}{2}\,{}^Nq^B \odot [{}^N\hat{\omega}^B_B]$$

$${}^N\dot{q}^B = \frac{1}{2}L({}^Nq^B)H\,{}^N\omega^B_B$$

$\frac{1}{2}$ comes from $\begin{bmatrix} \cos\frac{\theta}{2} \\ r\sin\frac{\theta}{2} \end{bmatrix}$

Here are the kinematics for quaternions and DCM's. Remember that kinematics is concerned with the relationship between velocities and configurations, so these functions are ODE's of the configurations as a function of the configurations and velocities.

DCM	quat	quat (Matrix Math)
$x_N = {}^NQ^B x_B$		
$\hat{x}_N = {}^NQ^B \hat{x}_B ({}^NQ^B)^T$	$\hat{x}_N = {}^Nq^B \odot \hat{x}_B \odot ({}^Nq^B)^\dagger$	$H x_N = R({}^Nq^B)^T L({}^Nq^B) H x_B$
${}^AQ^C = {}^AQ^B\,{}^BQ^C$	${}^Aq^C = {}^Aq^B \odot {}^Bq^C$	${}^Aq^C = L({}^Aq^B)\,{}^Bq^C$
${}^AQ^B = ({}^BQ^A)^T$	${}^Aq^B = ({}^Bq^A)^\dagger$	${}^Aq^B = T\,{}^Bq^A$
${}^N\dot{Q}^B = {}^NQ^B \widehat{({}^N\omega^B_B)}$	${}^N\dot{q}^B = \frac{1}{2}\,{}^Nq^B \odot [\widehat{{}^N\omega^B_B}]$	${}^N\dot{q}^B = \frac{1}{2} L({}^Nq^B) H\,{}^N\omega^B_B$

other 3-parameter attitudes

$$q = \begin{bmatrix} \cos\frac{\theta}{2} \\ r\sin\frac{\theta}{2} \end{bmatrix} = \begin{bmatrix} s \\ v \end{bmatrix}$$

axis angle: $\phi = r\theta$

Rodrigues parameter: $\frac{v}{s}$ from quat, singular at 180° ($s = \cos(\frac{\pi}{2}) = 0$)
"Gibbs vector"

Cayley Map: $\text{q\_from\_rp}(g) = \text{normalize}\left(\begin{bmatrix} 1 \\ g \end{bmatrix}\right)$

Modified RP: $\rho = r \cdot \tan\left(\frac{\theta}{4}\right)$

$\rho = \dfrac{v}{1+s}$, singular at 360° ($s = \cos(\frac{2\pi}{2}) = -1$)

	params	singular	kinematics
DCM	9	no	$\dot{Q} = f(Q, \omega)$ - nice, no trig
quat	4	no	$\dot{q} = f(q, \omega)$ - nice, no trig
axis-angle	3	180°	$\dot{\phi} = f(\phi, \omega)$ - awful, don't use
RP	3	180°	$\dot{g} = f(g, \omega)$ - nice
MRP	3	360°	$\dot{p} = f(p, \omega)$ - nice