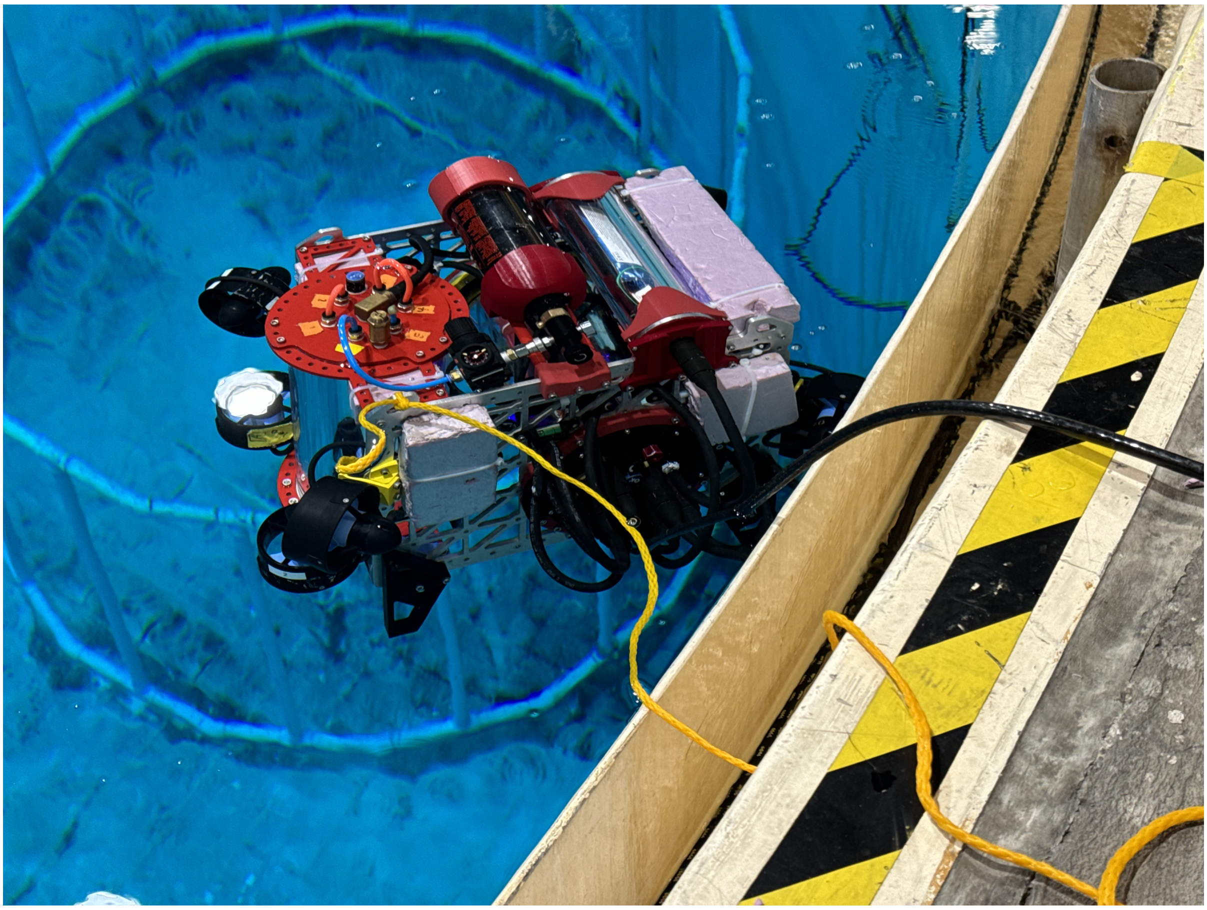
Towards a 6-DoF Dynamics Model for Qubo

Author: Ronen Aniti, Contributor to Robotics at Maryland's Qubo (Underwater Robot) Mechanical Subteam

Picture of Qubo in the Water



Defining Frames of Reference

Defining the World Frame of Reference

I define the world reference frame as ${ \mathcal{W} }$. This reference frame neither translates nor rotates.

Defining the Body-Fixed Frame of Reference

I define Qubo's body-fixed reference frame as ${ \mathcal{B} }$. This reference frame moves with Qubo and originates at Qubo's center of mass (CoM).

Describing Rotations Between Frames

I describe rotations between reference frames using rotation matrices, parameterized by Euler angles, and define the overall rotation from body-frame coordinates to world-frame coordinates as

$$
\mathbf{R} = \mathbf{R}_x(\phi) \mathbf{R}_y(\theta) \mathbf{R}_z(\psi).

$$

with

$$
\mathbf{R}_x(\phi) = \begin{bmatrix}
1 & 0 & 0 \
0 & \cos\phi & -\sin\phi \
0 & \sin\phi & \cos\phi
\end{bmatrix},
$$

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

$$
\mathbf{R}_z(\psi) = \begin{bmatrix}
\cos\psi & -\sin\psi & 0 \
\sin\psi & \cos\psi & 0 \
0 & 0 & 1
\end{bmatrix}.
$$

Here, ϕ, θ, and ψ denote rotations about the body-frame x, y, and z axes, respectively.

Defining the Degrees of Freedom

I indicate that Qubo has six degrees of freedom—three translational and three rotational—and write these as a set of independent coordinates:

$$
\mathbf{DoF} = \lbrace x, y, z, \phi, \theta, \psi \rbrace.
$$

Defining the State Vector

I define Qubo's state vector as

$$
\begin{aligned}
\mathbf{s} &=

\begin{bmatrix}
\mathbf{x} \
\mathbf{\Theta} \
\mathbf{v} \
\mathbf{\omega}
\end{bmatrix}

\begin{bmatrix}
x \ y \ z \
\phi \ \theta \ \psi \
\dot{x} \ \dot{y} \ \dot{z} \
p \ q \ r
\end{bmatrix}
\end{aligned}
$$

where $\mathbf{x}$ represents the linear position, $\mathbf{\Theta}$ the Euler angles, $\mathbf{v}$ the linear velocity, and $\mathbf{\omega}$ the body rates.

Defining the Relationship Between Body Rates and Euler Rates

I establish that the transformation, $\mathbf{T}(\mathbf{\Theta})$, between Euler rates and body rates as the following:

$$
\begin{bmatrix}
\dot{\phi} \
\dot{\theta} \
\dot{\psi}
\end{bmatrix}

\begin{bmatrix}
1 & \sin\phi,\tan\theta & \cos\phi,\tan\theta \
0 & \cos\phi & -\sin\phi \
0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta}
\end{bmatrix}
\begin{bmatrix}
p \
q \
r

\end{bmatrix}.
$$

Modeling the Displacement from the Center of Mass (CoM) to the Center of Buoyancy (CoB)

I model the displacement vector from Qubo's center of mass (CoM) to its center of buoyancy (CoB) as

$$
\mathbf{r}*b = \mathbf{r}*{\text{CoB}} - \mathbf{r}_{\text{CoM}},
$$

where
$$\mathbf{r}*{\mathrm{CoM}}$$ and $$\mathbf{r}*{\mathrm{CoB}}$$ denote the coordinates of Qubo's center of mass and center of buoyancy, respectively.

Modeling the External Forces and Moments

Modeling the Thruster Actuation Forces and Actuation Moments

I model the force produced by each of Qubo's thrusters as comprising a thrust magnitude and a unit direction.

$$
\mathbf{u}*i = u_i , \hat{\mathbf{e}}*{u_i}.
$$

It follows from this definition that the combined thrust actuation force can be expressed as

$$
\mathbf{u} = \sum_{i=1}^{8} \mathbf{u}*i = \sum*{i=1}^{8} u_i , \hat{\mathbf{e}}_{u_i}.
$$

Furthermore, the combined thrust actuation moment, about Qubo's center of mass (CoM) can be expressed as

$$
\mathbf{M}*u = \sum*{i=1}^{8} \left(\mathbf{r}*b \times \Bigl(u_i , \hat{\mathbf{e}}*{u_i} \Bigr) \right).
$$

Finally, each thruster, by pushing away water from its blades, induces a reaction moment in the direction opposing the spin direction. The combined reaction moment can be expressed as

$$
\mathbf{M}*{\text{reaction}} = \sum*{i=1}^{8} \mathbf{M}*{\text{reaction},i} = \sum*{i=1}^{8} \left(M_i , \hat{\mathbf{e}}_{M_i} \right),
$$

where $M_i$ is the magnitude of the reaction moment generated by the $i$th thruster, and $\hat{\mathbf{e}}_{M_i}$ is the corresponding unit vector in the direction opposing the thruster's spin.

Modeling The Body Forces and Body Moments

Modeling the Hydrodynamic Drag Forces and Moments

I model the hydrodynamic drag as damping forces and moments acting on Qubo's translational and rotational degrees of freedom, respectively:

$$
\mathbf{F}_d = -\mathbf{C}_x,\dot{\mathbf{x}},
$$

$$
\mathbf{M}d = -\mathbf{C}\omega,\mathbf{\omega},
$$

where

$$
\mathbf{C}x = \begin{bmatrix}
c_x & 0 & 0 \
0 & c_y & 0 \
0 & 0 & c_z
\end{bmatrix}
\quad \text{and} \quad
\mathbf{C}\omega = \begin{bmatrix}
c_\phi & 0 & 0 \
0 & c_\theta & 0 \
0 & 0 & c_\psi
\end{bmatrix}.
$$

Modeling the Gravity Force

I model the gravity force as

$$
\mathbf{F}_g = F_g \begin{bmatrix} 0 \ 0 \ -1 \end{bmatrix},
$$

where F_g is the magnitude of the gravitational force. The gravity force acts in the negative z world-frame direction and is applied at Qubo's center of mass (CoM).

Modeling the Buoyancy Force and Buoyancy Moment

I model the buoyancy force and moment as

$$
\mathbf{F}_b = F_b \begin{bmatrix} 0 \ 0 \ 1 \end{bmatrix},
$$

and

$$
\mathbf{M}_b = \mathbf{r}_b \times \mathbf{F}_b = \mathbf{r}_b \times \left(F_b \begin{bmatrix} 0 \ 0 \ 1 \end{bmatrix} \right),
$$

where F_b is the magnitude of the buoyancy force. The buoyancy force acts in the positive z world-frame direction and is applied at Qubo's center of buoyancy (CoB); the resulting moment is computed about Qubo's center of mass (CoM).

Formulating the Rigid-Body Dynamics

Formulating the Translational Rigid-Body Dynamics

I formulate the translational dynamics by employing Newton's second law:

$$
\mathbf{u} + \mathbf{F}_g + \mathbf{F}_b + \mathbf{F}_d = m,\ddot{\mathbf{x}}.
$$

It follows that the translational accelerations are

$$
\ddot{\mathbf{x}} = \frac{1}{m}\left(\mathbf{u} + \mathbf{F}_g + \mathbf{F}_b + \mathbf{F}_d \right).
$$

Formulating the Rotational Rigid-Body Dynamics

I formulate the rotational dynamics by employing Euler's equation for rigid-body motion. Since Euler's equation is formulated in the body frame, the buoyancy moment must be rotated into the body frame using the transpose of the rotation matrix:

$$
\mathbf{R}^T \mathbf{M}_b + \mathbf{M}_u + \mathbf{M}_d = \mathbf{I},\dot{\mathbf{\omega}} + \mathbf{\omega} \times \left(\mathbf{I},\mathbf{\omega} \right),
$$

with $\mathbf{I}$ being the inertia matrix and $\dot{\mathbf{\omega}}$ representing the body-frame angular acceleration.

Solving for the body-frame angular accelerations, I obtain

$$
\dot{\mathbf{\omega}} = \mathbf{I}^{-1}\left(\mathbf{R}^T \mathbf{M}_b + \mathbf{M}_u + \mathbf{M}_d - \mathbf{\omega} \times \left(\mathbf{I},\mathbf{\omega} \right) \right).
$$

Assembling the State-Space Model

$$
\dot{\mathbf{s}} = \begin{bmatrix}
\dot{\mathbf{x}} \
\dot{\mathbf{\Theta}} \
\ddot{\mathbf{x}} \
\dot{\mathbf{\omega}}
\end{bmatrix}

\begin{bmatrix}
\mathbf{v} \
\mathbf{T}(\mathbf{\Theta}),\mathbf{\omega} \
\frac{1}{m}\left(\mathbf{u} + \mathbf{F}_g + \mathbf{F}_b + \mathbf{F}_d \right) \
\mathbf{I}^{-1}\left(\mathbf{R}^T \mathbf{M}_b + \mathbf{M}*u* + *mathbf{M}*{\text{reaction}}
+\mathbf{M}_d - \mathbf{\omega} \times \left(\mathbf{I},\mathbf{\omega} \right) \right)
\end{bmatrix}.
$$

MATLAB Simulation

This week, the week of 2/24/25, I am working to implement this math model in a MATLAB code file. The following is the plan for my implementation.

Defining the Desired Simulation Output

The simulation must advance the state of Qubo's dynamics forward by one time step.

Defining the Simulation Input Parameters

The simulation must accept as input numeric values for each of Qubo's state variables and must also accept as input numeric values for Qubo's control inputs, namely thrust values. The simulation must allow the user to input values for Qubo's inertia elements, Qubo's center of mass (CoM) position and Qubo's center of buoyancy (CoB) position.

Implementing the State Equations as a Function

I will implement the state equations as a single MATLAB function, i.e. `advanceState`, which will accept as input Qubo's mass, inertia matrix, center of mass (CoM), center of Buoyancy (CoB), as well as its full state and control vectors. The function will, from these inputs, advance Qubo's state forward in time by some approximately infinitesimal amount of time, i.e. `dt` seconds, meaning it will output numeric values for the updated state vector. I will employ Euler integration to achieve this advancing of the state forward in time.

Verifying The Functionality of the Simulation

I will verify the functionality of the simulation by testing various control input and parameter combinations to ensure Qubo's time response is as expected. For example, I will configure Qubo's CoB to be above Qubo's CoM and ensure Qubo is able to level itself when I remove the hydrodynamic drag effects.

Progress So Far

This week is the week of 2/24/25. I am just starting to implement this model in MATLAB. So far, I have contructed a script to symbolically derive the state equations. I am making this script available for viewing on the GitHub repository associated with this document. In the coming week, I will implement the `advanceState` function.

- View Symbolic Derivation MATLAB Script

Insights from the Process

Design Points: On the Mechanical Design Considerations for the Qubo Underwater Robot

Design Point 1: On the Relative Positioning of Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB)

1. **Claim:** Qubo's Center of Mass (CoM) should be positioned below Qubo's Center of Buoyancy (CoB).
2. **Support:** If Qubo's CoM is positioned below Qubo's CoB, then, when Qubo is rotationally displaced in any of its three rotational degrees of freedom (DoF), the buoyancy force will function as a restoring force, acting to drive Qubo back to a level position.
3. **Support:** If Qubo's CoM and CoB are coincident, no such restoring force would be generated. In this case, Qubo would be dependent on its thrusters to return to a level orientation or on other external forces to return to a level orientation.
4. **Support:** If Qubo's CoM is positioned above Qubo's CoB, then if Qubo is rotationally displaced in any of its three rotational degrees of freedom (DoF), then the moment generated about the CoM would act to increase the displacement. If this resulting moment is not sufficiently offset by Qubo's thrusters or by other external forces from Qubo's surrounding environment, Qubo could turn upside-down.
5. **Support:** We should design Qubo to achieve passive hydrostatic stability. If Qubo is passive hydrostatic stable, then Qubo's actuators would not need to work as hard to maintain Qubo's orientation.
6. **Counterclaim:** If Qubo's mission involves mainly agile maneuvering, then it may be logical to diminish the degree of Qubo's passive hydrostatic stability, which would be possible by adjusting the relative positioning of Qubo's CoM and CoB.
7. **Question:** To what extent does Qubo's mission involve agile maneuvering?

Design Point 2: On the Relative Positioning of Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB)

- **Claim:** Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB) should be vertically aligned or close to being vertically aligned.
- **Support:** Only if Qubo's CoM and CoB are vertically aligned about an axis perpendicular to Qubo's bottom face will the buoyant force not produce any moment about Qubo's CoM when Qubo is in a

level orientation.

- **Underlying Assumption:** We should design Qubo so that Qubo is hydrostatically passively stable.
- **Counterclaim:** For the buoyant force to produce a moment when Qubo is in a level orientation is not necessarily undesirable.
- **Counterclaim:** For any such moment to produce a rotation requires the overcoming of resistive forces from the surrounding water. Thus, even if such a moment exists when Qubo is in a level orientation, this does not imply that Qubo will rotate even a small amount.
- **Question:** What's more important to us, ability to perform agile maneuvers or ability to self-correct when perturbed from a level orientation?

Design Point 3: On the Relative Positioning of Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB)

- **Claim:** Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB) should be relatively close together but should not be coincident.
- **Support:** The moment generated by the buoyant force about Qubo's CoM depends entirely on the cross product of the displacement vector from Qubo's CoM to Qubo's CoB and on the magnitude of the buoyant force, which is a function of Qubo's geometry. For Qubo to be hydrostatically passively stable, Qubo's CoB must be positioned above Qubo's CoM, but not so far away as to render Qubo overly stiff with respect to its rotational degrees of freedom. However, if the CoM and CoB are coincident, then the buoyant force will not generate a restoring torque, meaning Qubo would need to rely on its thrusters or other means to restore itself to level orientation.
- **Counterclaim:** For Qubo to not be hydrostatic passive stable may not necessarily be undesirable.

Design Point 4

- **Claim:** Qubo's Center of Mass (CoM) and Center of Buoyancy (CoB) should both be centrally located with respect to Qubo's entire body.

Determining Qubo's Center of Mass

- **Problem:** Density of parts components is unknown.
- **Solution:** Form a rough estimate based on visual inspection of "where the heavy parts are". Later, revise this estimate with more sophisticated and precise methods.

Determining Qubo's Center of Buoyancy

- **Problem:** Although part geometry is given in Solidworks assembly, it's subject to change.
- **Solution:** If Qubo is able to float and return to equilibrium after being rotationally perturbed in any of its three rotational degrees of freedom, Qubo likely has met **Design Point 1**.

Insight from Qubo Mechanical Subteam Meeting on 2/7/25

- After consulting with members of Qubo Mechanical, I learned that the team is trying to design Qubo to be capable of performing \(360^\circ\) rotations about each of its three body axes.

Thoughts Towards Integrating Mechanical Design with Control Objectives

Thoughts

- **Question:** What should be the relative positioning of Qubo's CoM and CoB?
- **Question:** How should actuators be placed on Qubo to achieve agile maneuverability?
- **Question:** What would it take to build a model to understand the interplay between CoM, CoB, actuator placement, stability, and control?

Rationale for Developing a Dynamics Model

- **Claim:** It is worthwhile for Ronen Aniti to expend time and energy focused on developing a well-documented, simplified 6-DoF dynamics model for Qubo.
- **Support:** A well-documented, simplified 6-DoF dynamics model for Qubo would aid in understanding the interplay between the rigid body dynamics aspect of Qubo and the control aspect of Qubo.

Reflection on Discussion on 2/9/25

- **Reflection:** By discussing the topic with team members today, I was able to confirm the importance of designing Qubo for agile maneuverability in the water. They seem to agree that it is important to meet Design Points 1, 2, 3, and 4 regarding the relative placement of Qubo's CoM and Qubo's CoB.
- **Realization:** Leaving today's meeting, I realize it's not obvious to me what the justification is for Design Point 4, which involves having Qubo's CoM and CoB centrally located. Perhaps once I begin constructing the math model, I will be able to justify for or against this design point.
- **Thought:** I am realizing that, although I didn't bring it up in today's discussion, the robot's rotational inertia, a property of mass distribution that is distinct from mass center, plays a foundational role in the dynamics. This is something worth exploring more.