Dynamic Control of an ROV

Eric Zheng

October 8, 2018

1 Introduction

This document considers the dynamic control of an underwater remotely operated vehicle (ROV). The purpose is to inform Rogue Robotics members in propulsion design decisions for the 2019 MATE ROV competition. This document begins with the basic theoy and then expands into additional dimensions, dynamic considerations, and controller models.

2 Basic Point Model

We begin this discussion with an analysis of the basic two-dimensional dynamics of an ideal ROV system (i.e. one not subject to the larger complexities of underwater control). We model the ROV as a rigid body under the influence of n thrusters, each of which are described by a thrust magnitude f_i, mounting position (r_i, θ_i), and mount angle ψ_i, as shown in Fig. **??**. We are interested in the final output wrench $\mathbf{F} = [F_x, F_y, \tau]^\mathsf{T}$, where F_x is the net force in the x-direction, F_y is the net force in the y-direction, and τ is the net torque. More specifically, we are interested in the vector of motor thrusts $\mathbf{f} = [f_1, f_2, f_3 \ldots f_n]^\mathsf{T}$ that will produce a desired $\mathbf{F}$.

The mapping $\mathbf{f} \mapsto \mathbf{F}$ can be performed with the structure matrix $\mathbf{A}$, which is defined by the mounting configuration of the thrusters. More specifically, we have:

$$\mathbf{F} = \mathbf{A}\mathbf{f} \tag{1}$$

where:

$$\mathbf{A} = \begin{bmatrix} \cos\psi_1 & \cos\psi_2 & \ldots & \cos\psi_n \\ \sin\psi_1 & \sin\psi_2 & \ldots & \sin\psi_n \\ r_1\sin(\psi_1 - \theta_1) & r_2\sin(\psi_2 - \theta_2) & \ldots & r_n\sin(\psi_n - \theta_n) \end{bmatrix} \tag{2}$$

For $n > 3$, this is an under-determined linear system, which, barring singularities, has infinitely many solutions. (Singularities should be avoidable with a good thruster layout.) We want to choose the *optimal* input vector $\mathbf{f}$ to achieve our desired output $\mathbf{F}$; here, we arbitrarily define the optimal solution as the one
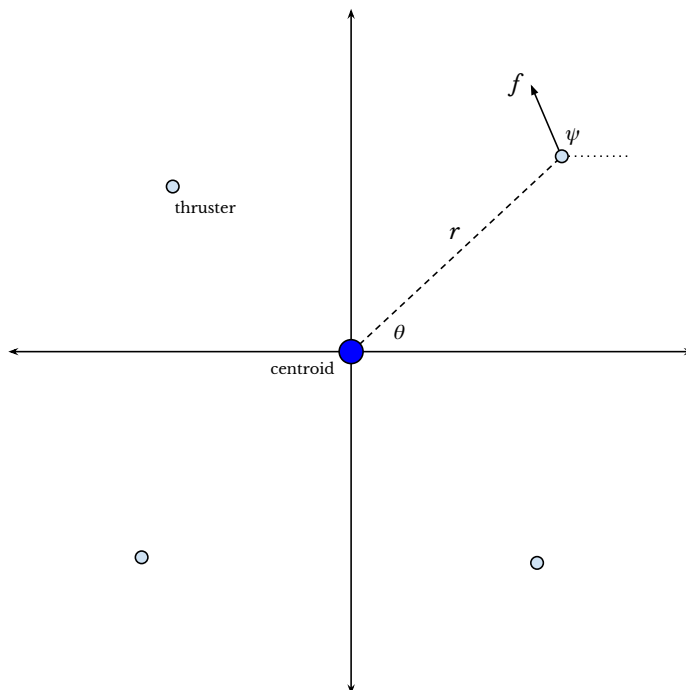
Figure 1: Two-dimensional model of the ROV dynamics. The ROV is considered as a rigid body under the influence of n thrusters, each of which exerts a thrust f as shown.

that minimizes the Euclidean norm $\|\mathbf{f}\|$. (Another objective function of interest is the total current draw, but this requires modeling current and thrust.) This optimization problem is easily solvable with numerical methods, which are available, for example, through `scipy`.

Now, let's extend this model to the three-dimensional case. Here, each motor still exerts some thrust f_i, but its position is described by the spherical coordinates (r_i, θ_i, ϕ_i), and its thrust orientation is described by a polar angle ψ_i and an azimuthal angle α_i. This can be more simply thought of as each thruster being situated at coordinates (r_i, θ_i, ϕ_i) in an ROV-fixed reference frame with the origin at the ROV's centroid. A force described by (f_i, ψ_i, α_i) is then applied in a translated reference frame whose origin is fixed at the thruster, as shown in Fig. **??**. We can then similarly model the net forces and torques $\mathbf{F} = [F_x, F_y, F_z, \tau_x, \tau_y, \tau_z]^\intercal$ on the ROV in terms of the structure matrix $\mathbf{B}$ and actuation vector $\mathbf{f}$:

$$\mathbf{F} = \mathbf{B}\mathbf{f} \tag{3}$$

where, for i between 1 and n:

$$\mathbf{B} = \begin{bmatrix} \sin\psi_i \cos\alpha_i \\ \sin\psi_i \sin\alpha_i \\ \cos\psi_i \\ r_i \cos\psi_i \cos\theta_i - r_i \sin\psi_i \cos\alpha_i \sin\theta_i \sin\phi_i \\ r_i \sin\psi_i \cos\alpha_i \sin\theta_i \cos\phi_i - r_i \cos\psi_i \cos\theta_i \\ -r_i \sin\psi_i \sin\theta_i \cos(\alpha_i + \phi_i) \end{bmatrix} \tag{4}$$

To recap, the control problem for the ROV can be stated as this: given a desired net force/torque $\mathbf{F}$ on the ROV (typically from the joystick input), calculate the motor thrusts $\mathbf{f}$ needed to produce this desired resultant. In the case of the Rogue Robotics ROV, we have $n = 6$, meaning that this is a well-determined system (the matrix $\mathbf{B}$ is square). Then, in most cases, there is a unique solution vector $\mathbf{f} = \mathbf{B}^{-1}\mathbf{F}$. Since $\mathbf{B}$ is fixed by the ROV design, We can pre-compute its inverse, making the calculation very fast.
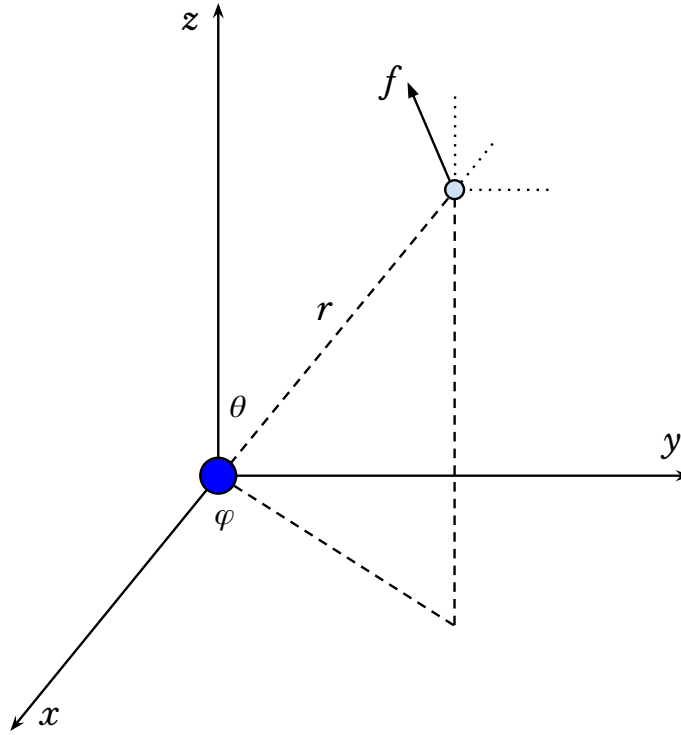

Figure 2: Three-dimensional model of ROV dynamics.

3 Nonlinear Dynamic Considerations

This simple dynamic model of the ROV is sufficient for basic joystick control, but more advanced software control needs to take into account the difficulties of navigating underwater. In the submarine environment, the ROV is subject to numerous nonlinear effects, such as added mass and drag forces. These are very difficult to computationally solve, but they can be reasonably estimated from physical data.

I'll get around to proper citations later, but in general, the equations of motion for an ROV can be written as:

$$\mathbf{M}\dot{\mathbf{v}} + C(\mathbf{v})\mathbf{v} + D(\mathbf{v})\mathbf{v} + g(\eta) = \mathbf{F} \tag{5}$$

where η describes the ROV's position and orientation in a world-fixed reference frame, $\mathbf{v}$ describes the ROV's linear and angular velocity in a body-fixed reference frame, $\mathbf{M}$ is the inertia and added mass matrix, $C(\mathbf{v})$ is the matrix of Coriolis and centripetal effects, $D(\mathbf{v})$ is the drag force matrix, $g(\eta)$ is the restoring buoyant and gravitational force matrix, and $\mathbf{F}$ is the thruster force matrix, described in the preceding section. Note that besides the usual rigid-body inertial terms in the matrix $\mathbf{M}$, there is also an effective added mass from movement in water.