The Math

T.J. Simmons

August 2019

This "robot fish" is modeled as an N-joint manipulator tasked with tracking a time-dependent plane curve known as the Lighthill curve:

$$Y(x,t) = (c_1 x + c_2 x^2) sin(kx - \omega t) + c_1 x sin(\omega t)$$

The Lighthill curve is supposed to model a fish's swimming waveform (or gait) in the XY-plane. Each fish's gait has its own unique amplitude profile, wavenumber, and angular frequency. These parameters are also a function of swimming speed, acceleration, etc.

The control input to the manipulator joints is a torque vector τ which tries to track a vector of reference angles q_d. The reference angles are the angles subtended by the chain of manipulator links of length L whose joints fall on the Lighthill curve. The reference joint positions at time t can be found by using the bisection method (or any root-finding algorithm) to solve the equation:

$$(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 = L^2$$

for x_i with

$$y_i = Y(x_i, t)$$

$$x_0 = 0$$

for each joint. Then the reference angles are derived from the trigonometry of the reference positions:

$$q_i = tan^{-1}(y_i/x_i) - \sum_{n=0}^{i-1} q_n$$

with

$$q_0 = 0$$

If the small angle approximation is used then we can eliminate the need for root finding since the manipulator is always straight meaning that:

$$x_{i+1} = x_i + L$$

The evolution of the manipulator joint angles is determined by a manipulator equation:

$$M(q)\ddot{q} + B\dot{q} + Kq = \tau$$

Here M(q) is a diagonal matrix which holds the instantaneous moments of inertia of the joints along its main diagonal. Note M(q) depends on q because the angles affect the distance between non-adjacent joints. With a point mass m_i for each joint (and $m_0 = 0$) the full expression for elements M_{ii} is:

$$M_{ii} = \sum_{k=1}^{k<i} \left[m_k L^2 \left(\left(1 + \sum_{j=k+1}^{i-1} cos(\sum_{n=j}^{i-1} q_n) \right)^2 + \left(\sum_{j=k+1}^{i-1} sin(\sum_{n=j}^{i-1} q_n) \right)^2 \right) \right] +$$

$$\sum_{k>i}^{N} \left[m_k L^2 \left(\left(1 + \sum_{j=i+1}^{k-1} cos(\sum_{n=i+1}^{j} q_n) \right)^2 + \left(\sum_{j=i+1}^{k-1} sin(\sum_{n=i+1}^{j} q_n) \right)^2 \right) \right]$$

Once again the small approximation can be used so that M(q) no longer depends on q and becomes a constant with:

$$M_{ii} = \sum_{k \neq i} m_k L^2 (i-k)^2$$

The other two matrices K and B are joint spring and damper matrices generated from a special matrix of the form:

$$D_7 = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

which can be generalized to D_N following the constant diagonal pattern. The sign change in the diagonal and off-diagonal elements represents the oppositely directed spring and damper torques produced by adjacent joints. When the spring and damper coefficients k and b are constant we get:

$$K = kD$$

$$B = bD$$

Now that the manipulator matrices are calculated a torque can be found that tracks the Lighthill curve. Using the so-called Lyapunov approach on the manipulator equation the simplest result would be:

$$\tau = -M(q)(K_d \dot{e} + K_p e) + M(q)\ddot{q_d} + B\dot{q} + Kq$$

where the q_d is the reference angle vector and the error signal $e = q - q_d$. This is a control input that fully cancels the system dynamics but it can be effectively reduced to a PD controller when constant gain matrices K_d and K_p dominate:

$$\tau = -M(q)(K_d \dot{e} + K_p e)$$