Online material for Fundamentals of Neuromechanics

v1.0 August 2015

Francisco J. Valero-Cuevas

Exercises: Ch. 2 Limb Kinematics

1. Re-write Eqs. 2.6 to 2.9 using 5 homogeneous transformation matrices.
2. Find the homogeneous transformation matrices for a limb based on the one shown in Fig. 2.3, but whose first DOF can also rotate in and out of the plane (i.e., rotate about the $\vec{j}_0$ axis.).
3. For the planar 2-link, 2-DOF limb shown in Fig. 2.3, find the Jacobian using the forward kinematic model in Eq. 2.30.
4. For Ex. 3, find the instantaneous endpoint velocities, $\dot{\vec{x}}$, for the configurations $q_1 = q_2 = 0$ and $q_1 = 0$, $q_2 = 90°$ when $\dot{q_1} = \dot{q_2} = 1\frac{deg}{s}$. Compare and contrast the results. Do they make sense?
5. For the planar 2-link, 2-DOF limb shown in Fig. 2.3, compare and contrast the forward kinematic models in Eqs. 2.30 and 3.13. What assumptions are being made, and what kinds of research questions can/cannot be asked with each of them?
6. For the planar 2-link, 2-DOF limb shown in Fig. 2.3, write all possible forward kinematic models.
7. For the planar 2-link, 2-DOF limb shown in Fig. 2.3, find the Jacobians for all possible forward kinematic models found in Ex. 6.

Exercises: Ch. 3 Limb Mechanics

1. For the planar 3-link, 3-DOF limb shown in Fig. 3.3, find the Jacobian for the forward kinematic model in Eq.3.22.
2. The planar 3-link, 3-DOF limb shown in Fig. 3.3 has the associated Eqs. 3.20 and 3.21. Describe what are the vectors $\vec{\tau}$ and $\vec{w}$.
3. For the planar 3-link, 3-DOF limb shown in Fig. 3.3, find the symbolic expressions for $J(\vec{q})$ and $J(\vec{q})^T$.
4. For Ex. 3, use a a mathematical package to find the symbolic expression for $J(\vec{q})^{-T}$.
5. Given the numerical examples in Sec. 3.4, draw the limb and the different vectors to explain why and how those results make sense.

6. Modify the MATLAB code in file *J2D2DOF.m* to explore the effect of changing posture on the input-output relationships of the system.
7. Modify the MATLAB code in file *J2D2DOF.m* to work for the planar 3-link, 3-DOF limb shown in Fig. 3.3.
8. Use Ex. 7 to explore the effect of changing posture on the input-output relationships of the system.

Exercises: Ch. 4 Tendon-Driven Limbs

1. Based on Table 6.1, create the moment arm matrix for the 5-DOF arm model, as in equation 4.21.
2. Based on Ex. 1, create the moment arm matrix for the 5 DOF arm, as in equation 4.26.
3. Based on Exs. 4 and 2, calculate the tendon excursions for that straight line motion of the hand, relative to the starting point.

Exercises: Ch. 5 Underdetermined Control

1. Based on the information provided, solve the Diet Problem to obtain the solution shown in Eq. 5.17
2. Use the information provided in Sec. 5.3 to find a solution to maximize upward vertical force for a posture and limb parameters of your choice.

Exercises: Ch. 6 Overdetermined Control

1. Based on Eqs. 6.1 to 6.7, find the homogeneous transformation matrices for the 5-DOF arm model shown in Fig. 6.1.
2. Based on Ex. 1, find the forward kinematic model of the 5-DOF arm model shown in Fig. 6.1.
3. Based on Ex. 2, find the Jacobian of the 5-DOF arm model shown in Fig. 6.1.
4. Use the numerical method described in the MATLAB code `simple_inverse_kinematics.m` to solve the inverse kinematic problem for a straight line motion of the hand in 3D space for the the 5-DOF arm model shown in Fig. 6.1.

Numerical code

Inverse kinematics

The `simple_inverse_kinematics.m` script solves a simple inverse kinematic problem using a closed form analytical approach.

5-DOF 17-muscle arm model

The `FiveDOF_model_frisbee_throw.m` script calculates muscle fiber lengths and velocities for a 5DOF, 17-muscle arm model as per Figs 6.3 and 6.6.

Jacobian

The `J2D2DOF.m` script calculates the symbolic and numerical versions of the Jacobian for a planar 2-link 2-DOF limb.

N-cube

The `ncube.m` script returns the vertices of an N-cube for dimensions 3 or higher.

Zonotope

The `zonotope_multi_N_2D.m` script shows how to map N-dimensional n-cubes into 2D and 3D via a random H matrix of dimensions 2 x N and 3 x N, respectively. It then plots the convex hulls of the zonotopes when considering the input to be between 3 and 8 dimensions. That is, a system having 3 to 8 muscles. This function was used to create Fig. 7.9.