arm_pendulum_modeling

October 19, 2021

1 Arm Motion Modeling

1.1 System Description

A double-pendulum system hanging in gravity is shown in the figure above. $q = [\theta_1, \theta_2]$ are the system configuration variables. We assume the z-axis is pointing out from the screen/paper, thus the positive direction of rotation is counter-clockwise.

The solution steps are:

1. Computing the Lagrangian of the system.
2. Computing the Euler-Lagrange equations, and solve them for $\ddot{\theta}_1$ and $\ddot{\theta}_2$.
3. Numerically evaluating the solutions for τ_1 and τ_2, and simulating the system for θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$, $\ddot{\theta}_1$ and $\ddot{\theta}_2$.
4. Animating the simulation.

[1]:
```
from IPython.core.display import HTML
display(HTML("<table><tr><td><img src='./double-pendulum-diagram.png'␣
 ↪width=450' height='300'></table>"))
```

```
<IPython.core.display.HTML object>
```

1.2 Import Libraries and Define System Constants

Import libraries:

[2]:
```
# Imports required for data processing
import os
import csv
import pandas as pd

# Imports required for dynamics calculations
import sympy
from sympy.abc import t
from sympy import symbols, Eq, Function, solve, sin, cos, Matrix, Subs,␣
 ↪substitution, Derivative, simplify, symbols, lambdify
import math
from math import pi
```

```
import numpy as np
import matplotlib.pyplot as plt

# Imports required for animation
from plotly.offline import init_notebook_mode, iplot
from IPython.display import display, HTML
import plotly.graph_objects as go
```

Define the system's constants:

[3]:
```
# Masses, length and center-of-mass positions (calculated using the lab␣
↪measurements)
# Mass calculations (mass unit is kg)
m_body = 90.6                                # Average weights for American adult␣
↪male
                                             # from "Anthropometric Reference Data␣
↪for Children and Adults:
                                             # United States, 2015-2018"
m_body_dict = {'ID': 51, 'JD': 79.5, 'JR': 76, 'KS': 59.3, 'KW': 63.8, 'LC': 61.
↪2,
               'LD': 97.3, 'LS': 82.2, 'MK': 93.5, 'MV': 98.5, 'SM': 68.5, 'TD':
↪ 70,
               'TM': 66.2}

m_upper_arm = 0.028 * m_body                 # Average upper arm weights relative␣
↪to body weight, from "Biomechanics
                                             # and Motor Control of Human Movement"␣
↪by David Winter (2009), 4th edition
m_upper_arm_dict = {'ID': 0.028 * m_body_dict['ID'], 'JD': 0.028 *␣
↪m_body_dict['JD'],
                    'JR': 0.028 * m_body_dict['JR'], 'KS': 0.028 *␣
↪m_body_dict['KS'],
                    'KW': 0.028 * m_body_dict['KW'], 'LC': 0.028 *␣
↪m_body_dict['LC'],
                    'LD': 0.028 * m_body_dict['LD'], 'LS': 0.028 *␣
↪m_body_dict['LS'],
                    'MK': 0.028 * m_body_dict['MK'], 'MV': 0.028 *␣
↪m_body_dict['MV'],
                    'SM': 0.028 * m_body_dict['SM'], 'TD': 0.028 *␣
↪m_body_dict['TD'],
                    'TM': 0.028 * m_body_dict['TM']}

m_lower_arm = 0.7395                         # Average lower prosthetics weights,␣
↪calculated using lab measurements

# Arm length calculations (length unit is m)
```

```
H_body = 1.769                                  # Average height for American adult␣
 ↪male, from "Height and body-mass
                                                # index trajectories of school-aged␣
 ↪children and adolescents from
                                                # 1985 to 2019 in 200 countries and␣
 ↪territories: a pooled analysis
                                                # of 2181 population-based studies␣
 ↪with 65 million participants"
H_body_dict = {'ID': 1.62, 'JD': 1.76, 'JR': 1.77, 'KS': 1.64, 'KW': 1.62, 'LC':
 ↪ 1.58,
               'LD': 1.875, 'LS': 1.635, 'MK': 1.78, 'MV': 1.805, 'SM': 1.79,␣
 ↪'TD': 1.69,
               'TM': 1.735}

L_upper_arm = 0.186 * H_body                    # Average upper arm length relative to␣
 ↪body height
                                                # from "Biomechanics and Motor Control␣
 ↪of Human Movement" by David
                                                # Winter (2009), 4th edition
L_upper_arm_dict = {'ID': 0.186 * H_body_dict['ID'], 'JD': 0.186 *␣
 ↪H_body_dict['JD'],
                    'JR': 0.186 * H_body_dict['JR'], 'KS': 0.186 *␣
 ↪H_body_dict['KS'],
                    'KW': 0.186 * H_body_dict['KW'], 'LC': 0.186 *␣
 ↪H_body_dict['LC'],
                    'LD': 0.186 * H_body_dict['LD'], 'LS': 0.186 *␣
 ↪H_body_dict['LS'],
                    'MK': 0.186 * H_body_dict['MK'], 'MV': 0.186 *␣
 ↪H_body_dict['MV'],
                    'SM': 0.186 * H_body_dict['SM'], 'TD': 0.186 *␣
 ↪H_body_dict['TD'],
                    'TM': 0.186 * H_body_dict['TM']}

L_lower_arm = 0.42                          # Average lower prosthetics length,␣
 ↪calculated using lab measurements

# Arm center of mass length calculations (length unit is m)
L_upper_arm_COM = 0.436 * L_upper_arm       # Average upper arm length from␣
 ↪shoulder to center of mass relative
                                                # to upper arm length, from␣
 ↪"Biomechanics and Motor Control of Human
                                                # Movement" by David Winter (2009),␣
 ↪4th edition
L_upper_arm_COM_dict = {'ID': 0.436 * L_upper_arm_dict['ID'], 'JD': 0.436 *␣
 ↪L_upper_arm_dict['JD'],
```

```
                        'JR': 0.436 * L_upper_arm_dict['JR'], 'KS': 0.436 *␣
 ↪L_upper_arm_dict['KS'],
                        'KW': 0.436 * L_upper_arm_dict['KW'], 'LC': 0.436 *␣
 ↪L_upper_arm_dict['LC'],
                        'LD': 0.436 * L_upper_arm_dict['LD'], 'LS': 0.436 *␣
 ↪L_upper_arm_dict['LS'],
                        'MK': 0.436 * L_upper_arm_dict['MK'], 'MV': 0.436 *␣
 ↪L_upper_arm_dict['MV'],
                        'SM': 0.436 * L_upper_arm_dict['SM'], 'TD': 0.436 *␣
 ↪L_upper_arm_dict['TD'],
                        'TM': 0.436 * L_upper_arm_dict['TM']}
L_lower_arm_COM = 0.2388                # Average lower prosthetics length from␣
 ↪elbow to center of mass,
                                        # calculated using lab measurements
```

1.3 Extracting Data

Extracting angles data and computing angular velocities and angular accelerations from the angles:

[4]:
```
def calculate_Vel(Ang_list, time_list, index):
    return ((Ang_list[index + 1] - Ang_list[index])
          / (time_list[index + 1] - time_list[index]))


def calculate_Acc(Vel_list, time_list, index):
    return ((Vel_list[index + 1] - Vel_list[index])
          / (time_list[index + 1] - time_list[index]))

print("current directory: ", os.getcwd())
data_csv_dir = '../../data/control_data/CSV Converted Files'
frame_frequency = 120

participants_list = []
time_list = []
Elbow_Ang_list, Sholder_Ang_list = [], []
Elbow_Vel_list, Sholder_Vel_list = [], []
Elbow_Acc_list, Sholder_Acc_list = [], []

for file in os.listdir(data_csv_dir):
    file_name = file.split(".")[0]
    participant_name = file.split("_")[0]

    if file.endswith(".csv"):
        frame = 0
        file_time_list = []
        file_R_Elbow_Ang_list, file_R_Sholder_Ang_list = [], []
        file_L_Elbow_Ang_list, file_L_Sholder_Ang_list = [], []
```

```
        file_R_Elbow_Vel_list, file_R_Sholder_Vel_list = [], []
        file_L_Elbow_Vel_list, file_L_Sholder_Vel_list = [], []
        file_R_Elbow_Acc_list, file_R_Sholder_Acc_list = [], []
        file_L_Elbow_Acc_list, file_L_Sholder_Acc_list = [], []

        data_path = os.path.join(data_csv_dir, file)

        # Cutting out weird data behavior on data edges
        if file == 'TD_WN7.csv':
            data_rows = open(data_path).read().strip().split("\n")[40:]
        elif file == 'TD_WN4.csv':
            data_rows = open(data_path).read().strip().split("\n")[24:-12]
        elif file == 'TD_WN11.csv':
            data_rows = open(data_path).read().strip().split("\n")[24:-3]
        else:
            data_rows = open(data_path).read().strip().split("\n")[24:]

        # Extract time [sec], elbow angles [rad], and shoulder angles [rad]␣
↪from data
        for row in data_rows:
            splitted_row = row.strip().split("\t")

            # Check if loop finished all data
            if len(splitted_row) < 80:
                break

            file_time_list.append(frame / frame_frequency)
            file_R_Sholder_Ang_list.append(float(splitted_row[11]) * 2*pi/360)
            file_R_Elbow_Ang_list.append(float(splitted_row[9]) * 2*pi/360)
            file_L_Sholder_Ang_list.append(float(splitted_row[23]) * 2*pi/360)
            file_L_Elbow_Ang_list.append(float(splitted_row[21]) * 2*pi/360)
            frame += 1

        # Extract elbow and shoulder velocities [rad/sec] from angles
        for i in range(len(file_time_list) - 1):
            R_Elbow_Vel = calculate_Vel(file_R_Elbow_Ang_list, file_time_list,␣
↪i)
            R_Sholder_Vel = calculate_Vel(file_R_Sholder_Ang_list,␣
↪file_time_list, i)
            L_Elbow_Vel = calculate_Vel(file_L_Elbow_Ang_list, file_time_list,␣
↪i)
            L_Sholder_Vel = calculate_Vel(file_L_Sholder_Ang_list,␣
↪file_time_list, i)

            file_R_Elbow_Vel_list.append(R_Elbow_Vel)
            file_R_Sholder_Vel_list.append(R_Sholder_Vel)
            file_L_Elbow_Vel_list.append(L_Elbow_Vel)
```

```
            file_L_Sholder_Vel_list.append(L_Sholder_Vel)

        # Extract elbow and shoulder Accelerations [rad/sec^2] from velocities
        for i in range(len(file_time_list) - 2):
            R_Elbow_Acc = calculate_Acc(file_R_Elbow_Vel_list, file_time_list,␣
↪i)
            R_Sholder_Acc = calculate_Acc(file_R_Sholder_Vel_list,␣
↪file_time_list, i)
            L_Elbow_Acc = calculate_Acc(file_L_Elbow_Vel_list, file_time_list,␣
↪i)
            L_Sholder_Acc = calculate_Acc(file_L_Sholder_Vel_list,␣
↪file_time_list, i)

            file_R_Elbow_Acc_list.append(R_Elbow_Acc)
            file_R_Sholder_Acc_list.append(R_Sholder_Acc)
            file_L_Elbow_Acc_list.append(L_Elbow_Acc)
            file_L_Sholder_Acc_list.append(L_Sholder_Acc)

        # Adjust lists length
        file_time_list = file_time_list[:-2]
        file_R_Elbow_Ang_list = file_R_Elbow_Ang_list[:-2]
        file_R_Sholder_Ang_list = file_R_Sholder_Ang_list[:-2]
        file_L_Elbow_Ang_list = file_L_Elbow_Ang_list[:-2]
        file_L_Sholder_Ang_list = file_L_Sholder_Ang_list[:-2]

        file_R_Elbow_Vel_list = file_R_Elbow_Vel_list[:-1]
        file_R_Sholder_Vel_list = file_R_Sholder_Vel_list[:-1]
        file_L_Elbow_Vel_list = file_L_Elbow_Vel_list[:-1]
        file_L_Sholder_Vel_list = file_L_Sholder_Vel_list[:-1]

        participants_list.append(participant_name)
        participants_list.append(participant_name)

        time_list.append(file_time_list)
        time_list.append(file_time_list)

        Elbow_Ang_list.append(file_R_Elbow_Ang_list)
        Sholder_Ang_list.append(file_R_Sholder_Ang_list)
        Elbow_Ang_list.append(file_L_Elbow_Ang_list)
        Sholder_Ang_list.append(file_L_Sholder_Ang_list)
        Elbow_Vel_list.append(file_R_Elbow_Vel_list)
        Sholder_Vel_list.append(file_R_Sholder_Vel_list)
        Elbow_Vel_list.append(file_L_Elbow_Vel_list)
        Sholder_Vel_list.append(file_L_Sholder_Vel_list)
        Elbow_Acc_list.append(file_R_Elbow_Acc_list)
        Sholder_Acc_list.append(file_R_Sholder_Acc_list)
        Elbow_Acc_list.append(file_L_Elbow_Acc_list)
```

```
        Sholder_Acc_list.append(file_L_Sholder_Acc_list)
```

```
current directory:
/home/yael/Documents/MSR_Courses/ME499-Final_Project/Motorized-Prosthetic-
Arm/motor_control/arm_pendulum_modeling
```

1.4 System Modeling

Computing the Lagrangian of the system:

[5]:
```
m1, m2, g, R1, R1_COM, R2, R2_COM = symbols(r'm1, m2, g, R1, R1_COM, R2,␣
 ↪R2_COM')

# The system torque variables as function of t
tau1 = Function(r'tau1')(t)
tau2 = Function(r'tau2')(t)

# The system configuration variables as function of t
theta1 = Function(r'theta1')(t)
theta2 = Function(r'theta2')(t)

# The velocity as derivative of position wrt t
theta1_dot = theta1.diff(t)
theta2_dot = theta2.diff(t)

# The acceleration as derivative of velocity wrt t
theta1_ddot = theta1_dot.diff(t)
theta2_ddot = theta2_dot.diff(t)

# Converting the polar coordinates to cartesian coordinates
x1 = R1_COM * sin(theta1)
x2 = R1 * sin(theta1) + R2_COM * sin(theta1 + theta2)

y1 = -R1_COM * cos(theta1)
y2 = -R1 * cos(theta1) - R2_COM * cos(theta1 + theta2)

# Calculating the kinetic and potential energy of the system
KE = 1/2 * m1 * ((x1.diff(t))**2 + (y1.diff(t))**2) + 1/2 * m2 * ((x2.
 ↪diff(t))**2 + (y2.diff(t))**2)
PE = m1 * g * y1 + m2 * g * y2

# Computing the Lagrangian
L = simplify(KE - PE)
print('L: ')
display(L)
```

```
L:
```

$$0.5R_{1COM}^2 m_1 \left(\frac{d}{dt}\theta_1(t)\right)^2 + R_{1COM} g m_1 \cos\left(\theta_1(t)\right) + g m_2 \left(R_1 \cos\left(\theta_1(t)\right) + R_{2COM} \cos\left(\theta_1(t) + \theta_2(t)\right)\right) +$$

$$0.5 m_2 \left(R_1^2 \left(\frac{d}{dt}\theta_1(t)\right)^2 + 2R_1 R_{2COM} \cos\left(\theta_2(t)\right) \left(\frac{d}{dt}\theta_1(t)\right)^2 + 2R_1 R_{2COM} \cos\left(\theta_2(t)\right) \frac{d}{dt}\theta_1(t) \frac{d}{dt}\theta_2(t) + R_{2COM}^2 \left(\frac{d}{dt}\theta\right.\right.$$

Computing the Euler-Lagrange equations:

[6]:
```
# Define the derivative of L wrt the functions: x, xdot
L_dtheta1 = L.diff(theta1)
L_dtheta2 = L.diff(theta2)

L_dtheta1_dot = L.diff(theta1_dot)
L_dtheta2_dot = L.diff(theta2_dot)

# Define the derivative of L_dxdot wrt to time t
L_dtheta1_dot_dt = L_dtheta1_dot.diff(t)
L_dtheta2_dot_dt = L_dtheta2_dot.diff(t)

# Define the left hand side of the the Euler-Lagrange as a matrix
lhs = Matrix([simplify(L_dtheta1_dot_dt - L_dtheta1),
              simplify(L_dtheta2_dot_dt - L_dtheta2)])

# Define the right hand side of the the Euler-Lagrange as a Matrix
rhs = Matrix([tau1, tau2])

# Compute the Euler-Lagrange equations as a matrix
EL_eqns = Eq(lhs, rhs)

print('Euler-Lagrange matrix for this systems:')
display(EL_eqns)
```

Euler-Lagrange matrix for this systems:

$$\begin{bmatrix} 1.0R_{1COM}^2 m_1 \frac{d^2}{dt^2}\theta_1(t) + R_{1COM} g m_1 \sin\left(\theta_1(t)\right) + g m_2 \left(R_1 \sin\left(\theta_1(t)\right) + R_{2COM} \sin\left(\theta_1(t) + \theta_2(t)\right)\right) + m_2 \left(R_1^2 \frac{d^2}{dt^2}\theta_1(t \\ R_{2COM} m_2 \left(R_1 \sin\left(\theta_2(t)\right) \left(\frac{d}{dt}\right.\right. \end{bmatrix}$$

$$\begin{bmatrix} \tau_1(t) \\ \tau_2(t) \end{bmatrix}$$

Solve the equations for τ_1 and τ_2:

[7]:
```
# Solve the Euler-Lagrange equations for the shoulder and elbow torques
T = Matrix([tau1, tau2])
soln = solve(EL_eqns, T, dict=True)

# Initialize the solutions
solution = [0, 0]
i = 0
```

```
for sol in soln:
    for v in T:
        solution[i] = simplify(sol[v])
        display(Eq(T[i], solution[i]))
        i =+ 1
```

$$\tau_1(t) = R_1^2 m_2 \frac{d^2}{dt^2}\theta_1(t) - 2.0 R_1 R_{2COM} m_2 \sin(\theta_2(t)) \frac{d}{dt}\theta_1(t) \frac{d}{dt}\theta_2(t) - R_1 R_{2COM} m_2 \sin(\theta_2(t)) \left(\frac{d}{dt}\theta_2(t)\right)^2 + 2.0 R_1 R_{2COM} m_2 \cos(\theta_2(t)) \frac{d^2}{dt^2}\theta_1(t) + R_1 R_{2COM} m_2 \cos(\theta_2(t)) \frac{d^2}{dt^2}\theta_2(t) + R_1 g m_2 \sin(\theta_1(t)) + R_{1COM}^2 m_1 \frac{d^2}{dt^2}\theta_1(t) + R_{1COM} g m_1 \sin(\theta_1(t)) + R_{2COM}^2 m_2 \frac{d^2}{dt^2}\theta_1(t) + R_{2COM}^2 m_2 \frac{d^2}{dt^2}\theta_2(t) + R_{2COM} g m_2 \sin(\theta_1(t) + \theta_2(t))$$

$$\tau_2(t) = R_{2COM} m_2 \left(R_1 \sin(\theta_2(t)) \left(\frac{d}{dt}\theta_1(t)\right)^2 + R_1 \cos(\theta_2(t)) \frac{d^2}{dt^2}\theta_1(t) + R_{2COM} \frac{d^2}{dt^2}\theta_1(t) + R_{2COM} \frac{d^2}{dt^2}\theta_2(t) + g \sin\right.$$

Simulating the system:

[8]:
```
# Substitute the derivative variables with a dummy variables and plug-in the␣
 ↪constants
solution_0_subs = solution[0]
solution_1_subs = solution[1]

theta1_dot_dummy = symbols('thetadot1')
theta2_dot_dummy = symbols('thetadot2')
theta1_ddot_dummy = symbols('thetaddot1')
theta2_ddot_dummy = symbols('thetaddot2')

solution_0_subs = solution_0_subs.subs([(g, 9.81)])
solution_1_subs = solution_1_subs.subs([(g, 9.81)])

solution_0_subs = solution_0_subs.subs([((theta1.diff(t)).diff(t),␣
 ↪theta1_ddot_dummy),
                                        ((theta2.diff(t)).diff(t),␣
 ↪theta2_ddot_dummy)])
solution_1_subs = solution_1_subs.subs([((theta1.diff(t)).diff(t),␣
 ↪theta1_ddot_dummy),
                                        ((theta2.diff(t)).diff(t),␣
 ↪theta2_ddot_dummy)])

solution_0_subs = solution_0_subs.subs([(theta1.diff(t), theta1_dot_dummy),
                                        (theta2.diff(t), theta2_dot_dummy)])
solution_1_subs = solution_1_subs.subs([(theta1.diff(t), theta1_dot_dummy),
                                        (theta2.diff(t), theta2_dot_dummy)])

# Lambdify the thetas and its derivatives
```

```
func1 = lambdify([theta1, theta2, theta1_dot_dummy, theta2_dot_dummy,␣
↪theta1_ddot_dummy,
                  theta2_ddot_dummy, m1, m2, R1, R2, R1_COM, R2_COM],␣
↪solution_0_subs, modules = sympy)
func2 = lambdify([theta1, theta2, theta1_dot_dummy, theta2_dot_dummy,␣
↪theta1_ddot_dummy,
                  theta2_ddot_dummy, m1, m2, R1, R2, R1_COM, R2_COM],␣
↪solution_1_subs, modules = sympy)

# Initialize the torque and power lists
Sholder_tau_list, Elbow_tau_list = [], []
Sholder_current_list, Elbow_current_list = [], []
Sholder_power_list, Elbow_power_list = [], []

motor_kv = 115
torque_const = 8.27 / motor_kv

for i in range(len(time_list)):
    # Initialize the torque and power lists
    tau1_list, tau2_list = [], []
    current1_list, current2_list = [], []
    power1_list, power2_list = [], []

    t_list = time_list[i]
    theta1_list = Sholder_Ang_list[i]
    theta2_list = Elbow_Ang_list[i]
    dtheta1_list = Sholder_Vel_list[i]
    dtheta2_list = Elbow_Vel_list[i]
    ddtheta1_list = Sholder_Acc_list[i]
    ddtheta2_list = Elbow_Acc_list[i]

    # Plug-in the angles, angular velocities and angular accelerations for␣
↪every time step to find the torques
    for j in range(len(t_list)):
        tau1_list.append(func1(theta1_list[j], theta2_list[j], dtheta1_list[j],␣
↪dtheta2_list[j],
                               ddtheta1_list[j], ddtheta2_list[j],␣
↪m_upper_arm_dict[participants_list[i]],
                               m_lower_arm,␣
↪L_upper_arm_dict[participants_list[i]], L_lower_arm,
                               L_upper_arm_COM_dict[participants_list[i]],␣
↪L_lower_arm_COM))

        tau2_list.append(func2(theta1_list[j], theta2_list[j], dtheta1_list[j],␣
↪dtheta2_list[j],
```

```
                                ddtheta1_list[j], ddtheta2_list[j],␣
↪m_upper_arm_dict[participants_list[i]],
                                m_lower_arm,␣
↪L_upper_arm_dict[participants_list[i]], L_lower_arm,
                                L_upper_arm_COM_dict[participants_list[i]],␣
↪L_lower_arm_COM))

        # Calculate the current required to reach the required joints torques␣
↪for every time step
        current1_list.append(torque_const * tau1_list[j])
        current2_list.append(torque_const * tau2_list[j])

        # Calculate the power required to reach the required angular velocities␣
↪and joints torques for every time step
        power1_list.append(dtheta1_list[j] * tau1_list[j])
        power2_list.append(dtheta2_list[j] * tau2_list[j])

    Sholder_tau_list.append(tau1_list)
    Elbow_tau_list.append(tau2_list)

    Sholder_current_list.append(current1_list)
    Elbow_current_list.append(current2_list)

    Sholder_power_list.append(power1_list)
    Elbow_power_list.append(power2_list)

    print(f"Trial {i}/{len(time_list) - 1} finished \t max torque:␣
↪{format(max(tau2_list), '.3f')} [Nm]\t max angular velocity:␣
↪{format(max(dtheta2_list), '.3f')} [rad/sec]\t max power:␣
↪{format(max(power2_list), '.3f')} [W]")
```

```
Trial 0/203 finished     max torque: 1.929 [Nm]  max angular velocity: 1.001
[rad/sec]   max power: 0.968 [W]
Trial 1/203 finished     max torque: 2.216 [Nm]  max angular velocity: 1.510
[rad/sec]   max power: 2.258 [W]
Trial 2/203 finished     max torque: 3.126 [Nm]  max angular velocity: 1.158
[rad/sec]   max power: 3.431 [W]
Trial 3/203 finished     max torque: 3.753 [Nm]  max angular velocity: 1.148
[rad/sec]   max power: 2.448 [W]
Trial 4/203 finished     max torque: 2.113 [Nm]  max angular velocity: 2.199
[rad/sec]   max power: 2.634 [W]
Trial 5/203 finished     max torque: 2.409 [Nm]  max angular velocity: 2.865
[rad/sec]   max power: 4.074 [W]
Trial 6/203 finished     max torque: 1.745 [Nm]  max angular velocity: 2.685
[rad/sec]   max power: 2.063 [W]
Trial 7/203 finished     max torque: 2.379 [Nm]  max angular velocity: 2.222
[rad/sec]   max power: 1.769 [W]
```

```
Trial 8/203 finished     max torque: 2.085 [Nm]  max angular velocity: 2.099
[rad/sec]   max power: 1.952 [W]
Trial 9/203 finished     max torque: 2.202 [Nm]  max angular velocity: 1.322
[rad/sec]   max power: 1.177 [W]
Trial 10/203 finished    max torque: 2.498 [Nm]  max angular velocity: 2.120
[rad/sec]   max power: 2.797 [W]
Trial 11/203 finished    max torque: 3.138 [Nm]  max angular velocity: 2.635
[rad/sec]   max power: 3.980 [W]
Trial 12/203 finished    max torque: 1.827 [Nm]  max angular velocity: 2.388
[rad/sec]   max power: 2.661 [W]
Trial 13/203 finished    max torque: 1.831 [Nm]  max angular velocity: 2.105
[rad/sec]   max power: 1.548 [W]
Trial 14/203 finished    max torque: 2.643 [Nm]  max angular velocity: 2.312
[rad/sec]   max power: 2.506 [W]
Trial 15/203 finished    max torque: 1.949 [Nm]  max angular velocity: 2.050
[rad/sec]   max power: 1.771 [W]
Trial 16/203 finished    max torque: 2.293 [Nm]  max angular velocity: 1.983
[rad/sec]   max power: 2.658 [W]
Trial 17/203 finished    max torque: 2.320 [Nm]  max angular velocity: 2.934
[rad/sec]   max power: 4.121 [W]
Trial 18/203 finished    max torque: 1.781 [Nm]  max angular velocity: 1.858
[rad/sec]   max power: 2.715 [W]
Trial 19/203 finished    max torque: 2.058 [Nm]  max angular velocity: 2.517
[rad/sec]   max power: 3.501 [W]
Trial 20/203 finished    max torque: 2.971 [Nm]  max angular velocity: 1.688
[rad/sec]   max power: 1.741 [W]
Trial 21/203 finished    max torque: 2.432 [Nm]  max angular velocity: 2.243
[rad/sec]   max power: 3.929 [W]
Trial 22/203 finished    max torque: 2.289 [Nm]  max angular velocity: 1.927
[rad/sec]   max power: 2.062 [W]
Trial 23/203 finished    max torque: 1.933 [Nm]  max angular velocity: 1.858
[rad/sec]   max power: 1.531 [W]
Trial 24/203 finished    max torque: 2.088 [Nm]  max angular velocity: 1.005
[rad/sec]   max power: 1.166 [W]
Trial 25/203 finished    max torque: 2.189 [Nm]  max angular velocity: 1.759
[rad/sec]   max power: 2.242 [W]
Trial 26/203 finished    max torque: 1.683 [Nm]  max angular velocity: 2.511
[rad/sec]   max power: 2.473 [W]
Trial 27/203 finished    max torque: 1.957 [Nm]  max angular velocity: 2.184
[rad/sec]   max power: 1.527 [W]
Trial 28/203 finished    max torque: 1.939 [Nm]  max angular velocity: 1.950
[rad/sec]   max power: 1.911 [W]
Trial 29/203 finished    max torque: 1.862 [Nm]  max angular velocity: 1.299
[rad/sec]   max power: 1.182 [W]
Trial 30/203 finished    max torque: 1.635 [Nm]  max angular velocity: 2.446
[rad/sec]   max power: 2.278 [W]
Trial 31/203 finished    max torque: 1.577 [Nm]  max angular velocity: 1.950
[rad/sec]   max power: 1.567 [W]
```

```
Trial 32/203 finished    max torque: 2.348 [Nm]  max angular velocity: 3.104
[rad/sec]   max power: 2.592 [W]
Trial 33/203 finished    max torque: 2.109 [Nm]  max angular velocity: 3.391
[rad/sec]   max power: 4.635 [W]
Trial 34/203 finished    max torque: 1.919 [Nm]  max angular velocity: 2.448
[rad/sec]   max power: 1.520 [W]
Trial 35/203 finished    max torque: 2.203 [Nm]  max angular velocity: 2.917
[rad/sec]   max power: 2.940 [W]
Trial 36/203 finished    max torque: 1.815 [Nm]  max angular velocity: 1.726
[rad/sec]   max power: 1.817 [W]
Trial 37/203 finished    max torque: 1.562 [Nm]  max angular velocity: 1.822
[rad/sec]   max power: 1.380 [W]
Trial 38/203 finished    max torque: 1.874 [Nm]  max angular velocity: 1.462
[rad/sec]   max power: 1.619 [W]
Trial 39/203 finished    max torque: 2.076 [Nm]  max angular velocity: 1.879
[rad/sec]   max power: 2.065 [W]
Trial 40/203 finished    max torque: 2.057 [Nm]  max angular velocity: 1.370
[rad/sec]   max power: 1.557 [W]
Trial 41/203 finished    max torque: 2.135 [Nm]  max angular velocity: 1.722
[rad/sec]   max power: 1.611 [W]
Trial 42/203 finished    max torque: 1.656 [Nm]  max angular velocity: 2.159
[rad/sec]   max power: 2.253 [W]
Trial 43/203 finished    max torque: 1.767 [Nm]  max angular velocity: 2.272
[rad/sec]   max power: 1.735 [W]
Trial 44/203 finished    max torque: 3.006 [Nm]  max angular velocity: 1.301
[rad/sec]   max power: 1.578 [W]
Trial 45/203 finished    max torque: 3.526 [Nm]  max angular velocity: 1.164
[rad/sec]   max power: 2.605 [W]
Trial 46/203 finished    max torque: 2.130 [Nm]  max angular velocity: 0.857
[rad/sec]   max power: 1.060 [W]
Trial 47/203 finished    max torque: 2.309 [Nm]  max angular velocity: 2.113
[rad/sec]   max power: 2.922 [W]
Trial 48/203 finished    max torque: 2.114 [Nm]  max angular velocity: 1.906
[rad/sec]   max power: 2.193 [W]
Trial 49/203 finished    max torque: 1.601 [Nm]  max angular velocity: 1.942
[rad/sec]   max power: 1.672 [W]
Trial 50/203 finished    max torque: 3.968 [Nm]  max angular velocity: 1.296
[rad/sec]   max power: 3.813 [W]
Trial 51/203 finished    max torque: 3.439 [Nm]  max angular velocity: 1.916
[rad/sec]   max power: 4.145 [W]
Trial 52/203 finished    max torque: 2.384 [Nm]  max angular velocity: 2.184
[rad/sec]   max power: 1.641 [W]
Trial 53/203 finished    max torque: 1.807 [Nm]  max angular velocity: 1.839
[rad/sec]   max power: 1.430 [W]
Trial 54/203 finished    max torque: 1.781 [Nm]  max angular velocity: 2.719
[rad/sec]   max power: 2.892 [W]
Trial 55/203 finished    max torque: 1.731 [Nm]  max angular velocity: 2.197
[rad/sec]   max power: 2.061 [W]
```

```
Trial 56/203 finished    max torque: 2.325 [Nm]  max angular velocity: 2.210
[rad/sec]   max power: 2.868 [W]
Trial 57/203 finished    max torque: 2.276 [Nm]  max angular velocity: 3.125
[rad/sec]   max power: 4.228 [W]
Trial 58/203 finished    max torque: 2.123 [Nm]  max angular velocity: 1.619
[rad/sec]   max power: 2.077 [W]
Trial 59/203 finished    max torque: 2.041 [Nm]  max angular velocity: 1.673
[rad/sec]   max power: 1.953 [W]
Trial 60/203 finished    max torque: 2.244 [Nm]  max angular velocity: 1.009
[rad/sec]   max power: 1.126 [W]
Trial 61/203 finished    max torque: 2.346 [Nm]  max angular velocity: 1.791
[rad/sec]   max power: 2.143 [W]
Trial 62/203 finished    max torque: 3.548 [Nm]  max angular velocity: 1.493
[rad/sec]   max power: 2.410 [W]
Trial 63/203 finished    max torque: 3.619 [Nm]  max angular velocity: 1.472
[rad/sec]   max power: 2.826 [W]
Trial 64/203 finished    max torque: 2.534 [Nm]  max angular velocity: 2.166
[rad/sec]   max power: 2.340 [W]
Trial 65/203 finished    max torque: 2.335 [Nm]  max angular velocity: 2.844
[rad/sec]   max power: 4.337 [W]
Trial 66/203 finished    max torque: 1.971 [Nm]  max angular velocity: 1.357
[rad/sec]   max power: 1.505 [W]
Trial 67/203 finished    max torque: 2.061 [Nm]  max angular velocity: 1.996
[rad/sec]   max power: 2.517 [W]
Trial 68/203 finished    max torque: 2.376 [Nm]  max angular velocity: 3.251
[rad/sec]   max power: 3.009 [W]
Trial 69/203 finished    max torque: 2.875 [Nm]  max angular velocity: 4.143
[rad/sec]   max power: 4.121 [W]
Trial 70/203 finished    max torque: 1.904 [Nm]  max angular velocity: 1.039
[rad/sec]   max power: 1.072 [W]
Trial 71/203 finished    max torque: 1.942 [Nm]  max angular velocity: 1.510
[rad/sec]   max power: 1.714 [W]
Trial 72/203 finished    max torque: 2.201 [Nm]  max angular velocity: 1.315
[rad/sec]   max power: 1.443 [W]
Trial 73/203 finished    max torque: 2.129 [Nm]  max angular velocity: 1.755
[rad/sec]   max power: 1.718 [W]
Trial 74/203 finished    max torque: 1.974 [Nm]  max angular velocity: 1.608
[rad/sec]   max power: 1.706 [W]
Trial 75/203 finished    max torque: 1.978 [Nm]  max angular velocity: 2.042
[rad/sec]   max power: 2.523 [W]
Trial 76/203 finished    max torque: 2.469 [Nm]  max angular velocity: 1.173
[rad/sec]   max power: 1.152 [W]
Trial 77/203 finished    max torque: 2.401 [Nm]  max angular velocity: 0.982
[rad/sec]   max power: 1.364 [W]
Trial 78/203 finished    max torque: 1.861 [Nm]  max angular velocity: 1.521
[rad/sec]   max power: 1.752 [W]
Trial 79/203 finished    max torque: 1.875 [Nm]  max angular velocity: 1.458
[rad/sec]   max power: 1.706 [W]
```

```
Trial 80/203 finished    max torque: 2.243 [Nm]  max angular velocity: 1.801
[rad/sec]   max power: 2.217 [W]
Trial 81/203 finished    max torque: 2.522 [Nm]  max angular velocity: 2.781
[rad/sec]   max power: 4.007 [W]
Trial 82/203 finished    max torque: 2.391 [Nm]  max angular velocity: 2.210
[rad/sec]   max power: 2.629 [W]
Trial 83/203 finished    max torque: 2.542 [Nm]  max angular velocity: 3.093
[rad/sec]   max power: 4.609 [W]
Trial 84/203 finished    max torque: 2.741 [Nm]  max angular velocity: 1.219
[rad/sec]   max power: 1.506 [W]
Trial 85/203 finished    max torque: 2.333 [Nm]  max angular velocity: 0.947
[rad/sec]   max power: 1.294 [W]
Trial 86/203 finished    max torque: 2.170 [Nm]  max angular velocity: 2.622
[rad/sec]   max power: 2.756 [W]
Trial 87/203 finished    max torque: 2.589 [Nm]  max angular velocity: 3.198
[rad/sec]   max power: 4.681 [W]
Trial 88/203 finished    max torque: 2.358 [Nm]  max angular velocity: 2.446
[rad/sec]   max power: 2.869 [W]
Trial 89/203 finished    max torque: 1.987 [Nm]  max angular velocity: 1.669
[rad/sec]   max power: 1.439 [W]
Trial 90/203 finished    max torque: 2.278 [Nm]  max angular velocity: 1.613
[rad/sec]   max power: 2.368 [W]
Trial 91/203 finished    max torque: 2.525 [Nm]  max angular velocity: 1.755
[rad/sec]   max power: 2.188 [W]
Trial 92/203 finished    max torque: 2.173 [Nm]  max angular velocity: 1.851
[rad/sec]   max power: 1.750 [W]
Trial 93/203 finished    max torque: 1.912 [Nm]  max angular velocity: 1.039
[rad/sec]   max power: 0.842 [W]
Trial 94/203 finished    max torque: 2.015 [Nm]  max angular velocity: 1.849
[rad/sec]   max power: 2.232 [W]
Trial 95/203 finished    max torque: 2.040 [Nm]  max angular velocity: 2.505
[rad/sec]   max power: 3.430 [W]
Trial 96/203 finished    max torque: 1.774 [Nm]  max angular velocity: 1.642
[rad/sec]   max power: 1.637 [W]
Trial 97/203 finished    max torque: 1.556 [Nm]  max angular velocity: 1.772
[rad/sec]   max power: 1.528 [W]
Trial 98/203 finished    max torque: 2.215 [Nm]  max angular velocity: 2.850
[rad/sec]   max power: 3.812 [W]
Trial 99/203 finished    max torque: 2.002 [Nm]  max angular velocity: 2.277
[rad/sec]   max power: 3.090 [W]
Trial 100/203 finished   max torque: 2.126 [Nm]  max angular velocity: 1.766
[rad/sec]   max power: 2.131 [W]
Trial 101/203 finished   max torque: 2.360 [Nm]  max angular velocity: 2.348
[rad/sec]   max power: 3.469 [W]
Trial 102/203 finished   max torque: 1.947 [Nm]  max angular velocity: 1.414
[rad/sec]   max power: 1.375 [W]
Trial 103/203 finished   max torque: 1.990 [Nm]  max angular velocity: 1.812
[rad/sec]   max power: 2.304 [W]
```

```
Trial 104/203 finished   max torque: 2.212 [Nm]  max angular velocity: 2.002
[rad/sec]   max power: 3.225 [W]
Trial 105/203 finished   max torque: 3.599 [Nm]  max angular velocity: 2.869
[rad/sec]   max power: 4.341 [W]
Trial 106/203 finished   max torque: 2.234 [Nm]  max angular velocity: 1.797
[rad/sec]   max power: 2.158 [W]
Trial 107/203 finished   max torque: 2.002 [Nm]  max angular velocity: 1.902
[rad/sec]   max power: 1.892 [W]
Trial 108/203 finished   max torque: 2.106 [Nm]  max angular velocity: 2.268
[rad/sec]   max power: 2.203 [W]
Trial 109/203 finished   max torque: 1.893 [Nm]  max angular velocity: 1.133
[rad/sec]   max power: 0.919 [W]
Trial 110/203 finished   max torque: 2.191 [Nm]  max angular velocity: 1.696
[rad/sec]   max power: 1.887 [W]
Trial 111/203 finished   max torque: 2.268 [Nm]  max angular velocity: 1.447
[rad/sec]   max power: 1.469 [W]
Trial 112/203 finished   max torque: 1.966 [Nm]  max angular velocity: 1.347
[rad/sec]   max power: 1.381 [W]
Trial 113/203 finished   max torque: 2.072 [Nm]  max angular velocity: 1.900
[rad/sec]   max power: 2.131 [W]
Trial 114/203 finished   max torque: 2.578 [Nm]  max angular velocity: 1.770
[rad/sec]   max power: 1.739 [W]
Trial 115/203 finished   max torque: 1.988 [Nm]  max angular velocity: 1.338
[rad/sec]   max power: 1.013 [W]
Trial 116/203 finished   max torque: 2.339 [Nm]  max angular velocity: 1.912
[rad/sec]   max power: 1.967 [W]
Trial 117/203 finished   max torque: 2.146 [Nm]  max angular velocity: 2.105
[rad/sec]   max power: 2.601 [W]
Trial 118/203 finished   max torque: 2.098 [Nm]  max angular velocity: 1.904
[rad/sec]   max power: 1.906 [W]
Trial 119/203 finished   max torque: 1.820 [Nm]  max angular velocity: 1.657
[rad/sec]   max power: 1.798 [W]
Trial 120/203 finished   max torque: 3.030 [Nm]  max angular velocity: 2.434
[rad/sec]   max power: 2.246 [W]
Trial 121/203 finished   max torque: 2.473 [Nm]  max angular velocity: 1.868
[rad/sec]   max power: 2.025 [W]
Trial 122/203 finished   max torque: 2.410 [Nm]  max angular velocity: 3.167
[rad/sec]   max power: 3.457 [W]
Trial 123/203 finished   max torque: 2.514 [Nm]  max angular velocity: 3.437
[rad/sec]   max power: 5.727 [W]
Trial 124/203 finished   max torque: 2.730 [Nm]  max angular velocity: 2.300
[rad/sec]   max power: 2.621 [W]
Trial 125/203 finished   max torque: 4.763 [Nm]  max angular velocity: 2.048
[rad/sec]   max power: 7.720 [W]
Trial 126/203 finished   max torque: 2.091 [Nm]  max angular velocity: 3.385
[rad/sec]   max power: 2.594 [W]
Trial 127/203 finished   max torque: 2.475 [Nm]  max angular velocity: 3.010
[rad/sec]   max power: 3.097 [W]
```

```
Trial 128/203 finished   max torque: 2.455 [Nm]  max angular velocity: 1.621
[rad/sec]   max power: 1.926 [W]
Trial 129/203 finished   max torque: 1.996 [Nm]  max angular velocity: 1.807
[rad/sec]   max power: 1.769 [W]
Trial 130/203 finished   max torque: 2.532 [Nm]  max angular velocity: 1.535
[rad/sec]   max power: 1.744 [W]
Trial 131/203 finished   max torque: 2.613 [Nm]  max angular velocity: 1.730
[rad/sec]   max power: 2.392 [W]
Trial 132/203 finished   max torque: 2.044 [Nm]  max angular velocity: 2.620
[rad/sec]   max power: 2.620 [W]
Trial 133/203 finished   max torque: 1.554 [Nm]  max angular velocity: 2.086
[rad/sec]   max power: 1.610 [W]
Trial 134/203 finished   max torque: 2.198 [Nm]  max angular velocity: 2.427
[rad/sec]   max power: 2.649 [W]
Trial 135/203 finished   max torque: 1.966 [Nm]  max angular velocity: 2.298
[rad/sec]   max power: 1.637 [W]
Trial 136/203 finished   max torque: 2.294 [Nm]  max angular velocity: 2.187
[rad/sec]   max power: 2.665 [W]
Trial 137/203 finished   max torque: 2.576 [Nm]  max angular velocity: 3.119
[rad/sec]   max power: 4.996 [W]
Trial 138/203 finished   max torque: 2.785 [Nm]  max angular velocity: 1.273
[rad/sec]   max power: 2.246 [W]
Trial 139/203 finished   max torque: 3.341 [Nm]  max angular velocity: 1.257
[rad/sec]   max power: 2.181 [W]
Trial 140/203 finished   max torque: 1.717 [Nm]  max angular velocity: 1.443
[rad/sec]   max power: 1.455 [W]
Trial 141/203 finished   max torque: 1.652 [Nm]  max angular velocity: 1.414
[rad/sec]   max power: 1.183 [W]
Trial 142/203 finished   max torque: 2.295 [Nm]  max angular velocity: 1.958
[rad/sec]   max power: 2.777 [W]
Trial 143/203 finished   max torque: 2.086 [Nm]  max angular velocity: 2.318
[rad/sec]   max power: 3.021 [W]
Trial 144/203 finished   max torque: 2.387 [Nm]  max angular velocity: 0.961
[rad/sec]   max power: 0.947 [W]
Trial 145/203 finished   max torque: 2.380 [Nm]  max angular velocity: 0.978
[rad/sec]   max power: 1.233 [W]
Trial 146/203 finished   max torque: 1.937 [Nm]  max angular velocity: 1.118
[rad/sec]   max power: 1.207 [W]
Trial 147/203 finished   max torque: 3.033 [Nm]  max angular velocity: 2.698
[rad/sec]   max power: 3.924 [W]
Trial 148/203 finished   max torque: 1.970 [Nm]  max angular velocity: 1.627
[rad/sec]   max power: 1.947 [W]
Trial 149/203 finished   max torque: 1.832 [Nm]  max angular velocity: 1.835
[rad/sec]   max power: 1.943 [W]
Trial 150/203 finished   max torque: 1.823 [Nm]  max angular velocity: 1.049
[rad/sec]   max power: 1.170 [W]
Trial 151/203 finished   max torque: 2.183 [Nm]  max angular velocity: 1.280
[rad/sec]   max power: 1.349 [W]
```

```
Trial 152/203 finished   max torque: 2.172 [Nm]  max angular velocity: 2.168
[rad/sec]   max power: 2.224 [W]
Trial 153/203 finished   max torque: 1.603 [Nm]  max angular velocity: 1.795
[rad/sec]   max power: 1.979 [W]
Trial 154/203 finished   max torque: 2.040 [Nm]  max angular velocity: 2.455
[rad/sec]   max power: 2.128 [W]
Trial 155/203 finished   max torque: 1.822 [Nm]  max angular velocity: 2.076
[rad/sec]   max power: 2.508 [W]
Trial 156/203 finished   max torque: 3.221 [Nm]  max angular velocity: 1.307
[rad/sec]   max power: 1.860 [W]
Trial 157/203 finished   max torque: 3.713 [Nm]  max angular velocity: 1.504
[rad/sec]   max power: 3.304 [W]
Trial 158/203 finished   max torque: 2.351 [Nm]  max angular velocity: 0.932
[rad/sec]   max power: 1.031 [W]
Trial 159/203 finished   max torque: 2.428 [Nm]  max angular velocity: 1.211
[rad/sec]   max power: 1.585 [W]
Trial 160/203 finished   max torque: 2.144 [Nm]  max angular velocity: 1.183
[rad/sec]   max power: 1.369 [W]
Trial 161/203 finished   max torque: 2.231 [Nm]  max angular velocity: 1.569
[rad/sec]   max power: 1.931 [W]
Trial 162/203 finished   max torque: 2.292 [Nm]  max angular velocity: 1.828
[rad/sec]   max power: 2.646 [W]
Trial 163/203 finished   max torque: 2.537 [Nm]  max angular velocity: 2.857
[rad/sec]   max power: 3.914 [W]
Trial 164/203 finished   max torque: 2.434 [Nm]  max angular velocity: 1.736
[rad/sec]   max power: 1.870 [W]
Trial 165/203 finished   max torque: 2.354 [Nm]  max angular velocity: 3.012
[rad/sec]   max power: 4.254 [W]
Trial 166/203 finished   max torque: 2.030 [Nm]  max angular velocity: 1.405
[rad/sec]   max power: 1.526 [W]
Trial 167/203 finished   max torque: 2.073 [Nm]  max angular velocity: 1.904
[rad/sec]   max power: 2.896 [W]
Trial 168/203 finished   max torque: 2.131 [Nm]  max angular velocity: 2.737
[rad/sec]   max power: 3.605 [W]
Trial 169/203 finished   max torque: 1.901 [Nm]  max angular velocity: 2.379
[rad/sec]   max power: 2.855 [W]
Trial 170/203 finished   max torque: 2.360 [Nm]  max angular velocity: 2.989
[rad/sec]   max power: 3.497 [W]
Trial 171/203 finished   max torque: 2.789 [Nm]  max angular velocity: 3.068
[rad/sec]   max power: 5.233 [W]
Trial 172/203 finished   max torque: 1.869 [Nm]  max angular velocity: 1.782
[rad/sec]   max power: 1.753 [W]
Trial 173/203 finished   max torque: 1.669 [Nm]  max angular velocity: 1.657
[rad/sec]   max power: 1.630 [W]
Trial 174/203 finished   max torque: 2.013 [Nm]  max angular velocity: 0.970
[rad/sec]   max power: 1.113 [W]
Trial 175/203 finished   max torque: 2.360 [Nm]  max angular velocity: 1.527
[rad/sec]   max power: 1.998 [W]
```

```
Trial 176/203 finished   max torque: 2.137 [Nm]  max angular velocity: 2.103
[rad/sec]   max power: 2.338 [W]
Trial 177/203 finished   max torque: 2.082 [Nm]  max angular velocity: 1.435
[rad/sec]   max power: 1.151 [W]
Trial 178/203 finished   max torque: 2.223 [Nm]  max angular velocity: 1.558
[rad/sec]   max power: 1.230 [W]
Trial 179/203 finished   max torque: 2.251 [Nm]  max angular velocity: 1.236
[rad/sec]   max power: 1.591 [W]
Trial 180/203 finished   max torque: 2.075 [Nm]  max angular velocity: 1.858
[rad/sec]   max power: 2.092 [W]
Trial 181/203 finished   max torque: 1.920 [Nm]  max angular velocity: 2.090
[rad/sec]   max power: 2.020 [W]
Trial 182/203 finished   max torque: 2.279 [Nm]  max angular velocity: 3.045
[rad/sec]   max power: 2.805 [W]
Trial 183/203 finished   max torque: 2.452 [Nm]  max angular velocity: 3.125
[rad/sec]   max power: 2.878 [W]
Trial 184/203 finished   max torque: 2.433 [Nm]  max angular velocity: 1.805
[rad/sec]   max power: 2.116 [W]
Trial 185/203 finished   max torque: 2.447 [Nm]  max angular velocity: 2.272
[rad/sec]   max power: 2.890 [W]
Trial 186/203 finished   max torque: 2.123 [Nm]  max angular velocity: 1.497
[rad/sec]   max power: 1.694 [W]
Trial 187/203 finished   max torque: 1.985 [Nm]  max angular velocity: 1.650
[rad/sec]   max power: 2.341 [W]
Trial 188/203 finished   max torque: 1.871 [Nm]  max angular velocity: 1.904
[rad/sec]   max power: 2.371 [W]
Trial 189/203 finished   max torque: 1.941 [Nm]  max angular velocity: 1.745
[rad/sec]   max power: 2.554 [W]
Trial 190/203 finished   max torque: 2.553 [Nm]  max angular velocity: 1.851
[rad/sec]   max power: 2.004 [W]
Trial 191/203 finished   max torque: 1.659 [Nm]  max angular velocity: 2.002
[rad/sec]   max power: 2.123 [W]
Trial 192/203 finished   max torque: 3.164 [Nm]  max angular velocity: 1.240
[rad/sec]   max power: 2.245 [W]
Trial 193/203 finished   max torque: 3.586 [Nm]  max angular velocity: 0.938
[rad/sec]   max power: 1.153 [W]
Trial 194/203 finished   max torque: 2.758 [Nm]  max angular velocity: 2.006
[rad/sec]   max power: 1.518 [W]
Trial 195/203 finished   max torque: 1.986 [Nm]  max angular velocity: 1.594
[rad/sec]   max power: 1.950 [W]
Trial 196/203 finished   max torque: 1.611 [Nm]  max angular velocity: 2.693
[rad/sec]   max power: 2.922 [W]
Trial 197/203 finished   max torque: 1.658 [Nm]  max angular velocity: 2.000
[rad/sec]   max power: 1.788 [W]
Trial 198/203 finished   max torque: 1.963 [Nm]  max angular velocity: 2.775
[rad/sec]   max power: 3.649 [W]
Trial 199/203 finished   max torque: 1.663 [Nm]  max angular velocity: 2.708
[rad/sec]   max power: 3.701 [W]
```

```
Trial 200/203 finished   max torque: 1.867 [Nm]  max angular velocity: 1.382
[rad/sec]   max power: 1.517 [W]
Trial 201/203 finished   max torque: 2.245 [Nm]  max angular velocity: 1.960
[rad/sec]   max power: 2.314 [W]
Trial 202/203 finished   max torque: 1.546 [Nm]  max angular velocity: 1.780
[rad/sec]   max power: 1.681 [W]
Trial 203/203 finished   max torque: 1.603 [Nm]  max angular velocity: 2.174
[rad/sec]   max power: 1.502 [W]
```

Calculation summary:

[9]:
```
max_Elbow_tau, max_Elbow_power, max_Elbow_Vel = 0, 0, 0
max_Elbow_tau_index, max_Elbow_power_index, max_Elbow_Vel_index = 0, 0, 0

for i in range(len(Elbow_tau_list)):
    if max_Elbow_Vel < max(Elbow_Vel_list[i]):
        max_Elbow_Vel = max(Elbow_Vel_list[i])
        max_Elbow_Vel_index = i

    if max_Elbow_tau < max(Elbow_tau_list[i]):
        max_Elbow_tau = max(Elbow_tau_list[i])
        max_Elbow_tau_index = i

    if max_Elbow_power < max(Elbow_power_list[i]):
        max_Elbow_power = max(Elbow_power_list[i])
        max_Elbow_power_index = i

print(f"maximum elbow angular velocity is {format(max_Elbow_Vel, '.3f')} [rad/
 ↪sec] ({format(max_Elbow_Vel*60/(2*pi), '.3f')} [rpm]), in trial␣
 ↪{max_Elbow_Vel_index}")
print(f"maximum elbow torque is {format(max_Elbow_tau, '.3f')} [Nm], in trial␣
 ↪{max_Elbow_tau_index}")
print(f"maximum elbow power is {format(max_Elbow_power, '.3f')} [W], in trial␣
 ↪{max_Elbow_power_index}")

# The torque equations for the maximum power:
solution_0_subs = solution_0_subs.subs([(m1,␣
 ↪m_upper_arm_dict[participants_list[max_Elbow_tau_index]]), (m2,␣
 ↪m_lower_arm), (R1,␣
 ↪L_upper_arm_dict[participants_list[max_Elbow_tau_index]]), (R2,␣
 ↪L_lower_arm), (R1_COM,␣
 ↪L_upper_arm_COM_dict[participants_list[max_Elbow_tau_index]]), (R2_COM,␣
 ↪L_lower_arm_COM), (g, 9.81)])
```

```
solution_1_subs = solution_1_subs.subs([(m1,␣
↪m_upper_arm_dict[participants_list[max_Elbow_tau_index]]), (m2,␣
↪m_lower_arm), (R1,␣
↪L_upper_arm_dict[participants_list[max_Elbow_tau_index]]), (R2,␣
↪L_lower_arm), (R1_COM,␣
↪L_upper_arm_COM_dict[participants_list[max_Elbow_tau_index]]), (R2_COM,␣
↪L_lower_arm_COM), (g, 9.81)])

print("\nThe torque equations for the maximum torque:")
display(Eq(T[0], solution_0_subs))
display(Eq(T[1], solution_1_subs))

display(Elbow_Ang_list[max_Elbow_tau_index])
display(Elbow_Vel_list[max_Elbow_tau_index])
display(Elbow_Acc_list[max_Elbow_tau_index])
display(Elbow_tau_list[max_Elbow_tau_index])
```

```
maximum elbow angular velocity is 4.143 [rad/sec] (39.560 [rpm]), in trial 69
maximum elbow torque is 4.763 [Nm], in trial 125
maximum elbow power is 7.720 [W], in trial 125

The torque equations for the maximum torque:
```

$\tau_1(t) = 0.111020235768\ddot{\theta}_1 \cos(\theta_2(t)) + 0.152055338724674\ddot{\theta}_1 + 0.055510117884\ddot{\theta}_2 \cos(\theta_2(t)) + 0.042170312888\ddot{\theta}_2 - 0.111020235768\dot{\theta}_1\dot{\theta}_2 \sin(\theta_2(t)) - 0.055510117884\dot{\theta}_2^2 \sin(\theta_2(t)) + 1.732373406 \sin(\theta_1(t) + \theta_2(t)) + 4.915563608124 \sin(\theta_1(t))$

$\tau_2(t) = 0.055510117884\ddot{\theta}_1 \cos(\theta_2(t)) + 0.042170312888\ddot{\theta}_1 + 0.042170312888\ddot{\theta}_2 + 0.055510117884\dot{\theta}_1^2 \sin(\theta_2(t)) + 1.732373406 \sin(\theta_1(t) + \theta_2(t))$

```
[0.4520402762665313,
 0.4661250833301256,
 0.4801400772236401,
 0.49408525794707475,
 0.5079431722079097,
 0.5217487265911849,
 0.5355193743894201,
 0.5492551156026155,
 0.5629908568158108,
 0.5767091447364863,
 0.590340166194562,
 0.6037966547274383,
 0.6170262504575553,
 0.6299765935073532,
 0.642647683876832,
 0.6549697083959121,
 0.6668728538945135,
 0.6783047604950762,
 0.6891607084424809,
```

0.6993534312741279,
0.7088131158199371,
0.7174350423247892,
0.7250621311560044,
0.7315373026809033,
0.7367034772668065,
0.7404559351585942,
0.7427248631861869,
0.7435102613495844,
0.7428993961113863,
0.7410318938117524,
0.738064834083362,
0.7341727498514146,
0.7293905477009502,
0.723648414461889,
0.7168939902566709,
0.7090400086226963,
0.7000166563898856,
0.6898413868507588,
0.6785316532978355,
0.6661398156086757,
0.6527880468309191,
0.6386159733047252,
0.6237981279552933,
0.608491590415303,
0.5928359870249139,
0.5769360375392455,
0.5608790084208978,
0.5447521661324701,
0.5286427771365625,
0.5126730144808144,
0.49694759792034543,
0.4816061537953153,
0.4667534018608436,
0.4525289684570898,
0.43900266675413363,
0.4263141230921349,
0.4145156973486533,
0.4036772026937685,
0.3939033588826002,
0.38524652579270835,
0.3777067034240928,
0.37124898519171384,
0.3658035579254915,
0.36121334199274646,
0.35735616434583894,
0.3540574920595697,
0.35112533891621917,

0.348420078575628,
0.34574972482007665,
0.34299210460192564,
0.3400774047510951,
0.33691835880498533,
0.3334800601785565,
0.3297450555792887,
0.3257482515922217,
0.3215943679724751,
0.317335764597609,
0.3130422546377029,
0.30873129138527694,
0.3044552347178908,
0.30023153792806456,
0.2960602010157981,
0.29195867727361147,
0.28794441999402454,
0.284087242347117,
0.2804395042104489,
0.27710592533913975,
0.27417377219578926,
0.2716779513654374,
0.2697057293106837,
0.26818729286144866,
0.26714009531025207,
0.2664594169019743,
0.2661103510515754,
0.2660230845889757,
0.2660928977590555,
0.26624997739173495,
0.2664594169019743,
0.26672121628977347,
0.26705282884765236,
0.2675589743307307,
0.2683094659090883,
0.2694439299228846,
0.27106708612723934,
0.27328365427727214,
0.2761983541281027,
0.27995081201989047,
0.2845933878301954,
0.2902308013141371,
0.29688050576423547,
0.3045075945954507,
0.3131120678077827,
0.3226590188161917,
0.3330786344505979,
0.34431855483344137,

0.35630896679464236,
0.3690149637491611,
0.3823667325269177,
0.39625955337279256,
0.41057125323914606,
0.4251447524932987,
0.4397880649175312,
0.45436156417168383,
0.4687256239155972,
0.4828976974417911,
0.4969999577979053,
0.5112069379091391,
0.5256059042380923,
0.5402666699548446,
0.5551717817668762,
0.570356146259227,
0.5857324969692971,
0.6012659273120465,
0.6167819043622761,
0.6321058951947863,
0.6470459135918578,
0.661427426628291,
0.6751631678414864,
0.6882182306464041,
0.7005053485804441,
0.7119547084735269,
0.7224964971555726,
0.7320085415789418,
0.7404035752810344,
0.7475768785067312,
0.7534760913784719,
0.7579964941411372,
0.7610682736246475,
0.7626390699514422,
0.7626914298290021,
0.761172993379767,
0.7579964941411372,
0.7531270255280732,
0.7465471342480545,
0.7382917268861214,
0.7284829764899131,
0.7172430561070696,
0.7046417789076707,
0.6907664113543158,
0.6757391264946445,
0.6597693638388964,
0.6430840161898306,
0.6259099763502065,

0.6085264970003429,
0.5911430176504793,
0.5739515245183352,
0.5570916439440701,
0.5405982825127236,
0.524453986931776,
0.5086063973236675,
0.49309042027343797,
0.47795841565864716,
0.4633849164044945,
0.4495793620212193,
0.436751192019061,
0.4249876728606192,
0.4143411644234538,
0.404741853537485,
0.396137380325153,
0.3883881184462981,
0.3813893481458009,
0.37500144308350164,
0.36908477691924085,
0.36355208319041876,
0.35824628226435606,
0.35311501426349273,
0.348123372602789,
0.3432888105747647,
0.3386462347644597,
0.334248005049434,
0.3301464813072474,
0.32637657012293964,
0.32288591161895097,
0.31960469262520164,
0.3164980065566517,
0.31346113365818157,
0.3103544475896317,
0.3069859621332826,
0.30316369107141505,
0.2988178212339491,
0.293930899328365,
0.28859019181726236,
0.28291787174828087,
0.27710592533913975,
0.2714161519776382,
0.26616271092913524,
0.26172957462906965,
0.25837854246524056,
0.25609216114512795,
0.2548006174986522,
0.2542770187230539,

0.2543468318931337,
0.2549402438388117,
0.2560223479750482,
0.25748842454672344,
0.2592163005061978,
0.2610663495133118,
0.26298621169050557,
0.2649584337452592,
0.26712264201773217,
0.26958355626304414,
0.2724982561138747,
0.27600636791038324,
0.2800729850675301,
0.28462829441523524,
0.2895501229058593,
0.2948035639543622,
0.30038861756074403,
0.3063750968950846,
0.3127630019573839,
0.31962214591772153,
0.3269350754835778,
0.33471924394747254,
0.34302701118696555,
0.3518932837870967,
0.36135296833290603,
0.37137115823935346,
0.3818780403363593,
0.3928387080388837,
0.4040611751292072,
0.41547562843725017,
0.42701225479293264,
0.4387408673663346,
0.4507661859125755,
0.46319293018677515,
0.4761956331141328,
0.4899488276198482,
0.5045223268740008,
0.5199161308765908,
0.5359731599949387,
0.5525363345963649,
0.5694485750481899,
0.5865178951326944,
0.6035523086321591,
0.6204296424989443,
0.6369753638078505,
0.6530672995112382,
0.668548369976428,
0.6833313087408199,

0.6972590361717347,
0.7102268325140524,
0.7220776181350941,
0.7326892199872196,
0.7419220117302695,
0.7496887269016443,
0.7559370056237841,
0.7606493946041687,
0.7637909872577585,
0.7653268769995135,
0.7652919704144736,
0.7636513609175989,
0.7604748616789694,
0.7557973792836244,
0.7496189137315645,
0.7419394650227894,
0.7326892199872196,
0.7217983654547749,
0.7092145415478958,
0.6949726548516222,
0.6791599718285534,
0.6618986655263295,
0.6433981754551895,
0.6239552075879728,
0.6038141080199582,
0.5832715827239849,
0.5626068843803721,
0.5421690788395185,
0.5222025121967034,
0.5029689838397259,
0.48460812010874554,
0.4672071874663621,
0.4508185457901353,
0.435529461542665,
0.4213922946015109,
0.4084244982591931,
0.39664352580823137,
0.3860144706635859,
0.3765198795327367,
0.3681073925381241,
0.360742103094708,
0.35428438486232894,
0.3486818779634272,
0.34384731593540285,
0.3396759790231364,
0.3360631474715081,
0.33288664823287845,
0.33000685496708787,

0.32728414133397665,
0.32468360074850516,
0.3221179667480734,
0.31955233274764183,
0.3169168855771304,
0.3141069054814195,
0.3110525792904294,
0.30768409383408035,
0.3040014491123723,
0.29996973854026543,
0.2955540555327198,
0.2907369467972154,
0.2855184123337524,
0.2799682653124104,
0.27413886561074935,
0.26813493298388885,
0.2620611871869486,
0.2561096144376479,
0.25043729436866635,
0.24523621319772326,
0.2406634505574981,
0.23682372620311057,
0.23375194671960056,
0.2314306588144481,
0.22985986248765322,
0.22896974456913607,
0.2287079451813369,
0.22900465115417598,
0.22973768944001358,
0.23080234028373015,
0.2321636971002857,
0.2337344934270806,
0.2355147292641148,
0.23750440461138836,
0.2397384260539411,
0.24233896663941262,
0.24539329283040273,
0.24904103096707086,
0.2534043540970567,
0.2585879819754799,
0.26467918106494004,
0.2716954046579573,
0.2796715593395714,
0.28852037864718255,
0.2982244092882711,
0.3086265716301573,
0.31965705250276144,
0.3311413189808841,

0.3429397447243658,
0.3549476099780868,
0.36716491474204704,
0.3796265656012866,
0.3923674691408452,
0.4054050786532429,
0.4186870342609197,
0.4321958826713558,
0.4459316238845512,
0.4598768046079859,
0.4740488781341799,
0.48846529775565295,
0.5031958766424851,
0.5182755213797161,
0.5336867786748261,
0.549429648527815,
0.5655215842312027,
0.5818753193223896,
0.5984384939238157,
0.6151936547429612,
0.6320709886097465,
0.6490006823540915,
0.6658954695133965,
0.6826331770400221,
0.6990741785938087,
0.7150613945420768,
0.730385385374587,
0.7448716181661399,
0.7583281066990162,
0.7705803180480165,
0.781506079165501,
0.7908959505412304,
0.7985928525425254,
0.8043873456591465,
0.8082096167210142,
0.8100073058505682,
0.8098502262178889,
0.8077732844080155,
0.8038462935910284,
0.798121613644487,
0.7906341511534313,
0.7814013594103812,
0.770405785122817,
0.7576648815832584,
0.7431611954991855,
0.726981993333198,
0.709179634962856,
0.6898937467283185,

```
 0.6693163148473054,
 0.6477440452926555,
 0.625456190744688,
 0.6027669104687616,
 0.5799729104377156,
 0.5573534433318692,
 0.5351004953689414,
 0.5133536928890922]

[1.6901768476313128,
 1.681799267221742,
 1.6734216868121574,
 1.6629497113001994,
 1.6566665259930162,
 1.6524777357882259,
 1.648288945583452,
 1.6482889455834386,
 1.646194550481051,
 1.6357225749690856,
 1.614778623945155,
 1.587551487614045,
 1.5540411659757507,
 1.5205308443374537,
 1.4786429422896055,
 1.4283774598321666,
 1.3718287920675245,
 1.3027137536885693,
 1.2231267397976355,
 1.1351621454971108,
 1.0346311805822463,
 0.9152506597458256,
 0.7770205829878727,
 0.619940950308382,
 0.4502949470145224 4,
 0.2722713633111251 7,
 0.0942477796077012 8,
-0.0733038285837706 2,
-0.224100275956074,
-0.3560471674068478,
-0.4670501078336894,
-0.5738642580557205,
-0.6890559886873483,
-0.810530904626175,
-0.942477796076943,
-1.0828022679372877,
-1.2210323446952172,
-1.3571680263508017,
-1.4870205226991648,
```

-1.6022122533308072,
-1.7006488231432595,
-1.7781414419318304,
-1.836784504798835,
-1.8786724068466953,
-1.9079939382802042,
-1.9268434942017205,
-1.9352210746113243,
-1.9331266795089104,
-1.9163715186897814,
-1.8870499872562672,
-1.8409732950036226,
-1.7823302321366008,
-1.7069320084504607,
-1.6231562043547336,
-1.522625239439853,
-1.4158110892177889,
-1.3006193585861794,
-1.1728612573401929,
-1.0388199707870254,
-0.904778684233864,
-0.7749261878854768,
-0.6534512719466725,
-0.5508259119294088,
-0.4628613176289037,
-0.39584067435230935,
-0.35185837720206214,
-0.3246312408709396,
-0.3204424506661641,
-0.3309144261781228,
-0.3497639820996611,
-0.3790855135331741,
-0.4125958351714579,
-0.44820055191214286,
-0.47961647844803257,
-0.49846603436959525,
-0.511032404983935,
-0.5152211951887237,
-0.5173155902911183,
-0.513126800086336,
-0.5068436147791529,
-0.5005604294719697,
-0.4921828490623987,
-0.48171087355043335,
-0.4628613176289037,
-0.43772857640017165,
-0.4000294645570982,
-0.3518583772020602,

-0.2994984996422267,
-0.23666664657043804,
-0.18221237390820663,
-0.12566370614359101,
-0.08168140899333406,
-0.04188790204786811,
-0.0104719755119587,
0.008377580409570958,
0.01884955592153607,
0.025132741228719536,
0.03141592653590275,
0.039793506945467044,
0.06073745796940362,
0.09005898940290612,
0.13613568165555637,
0.19477874452257307,
0.26598817800393265,
0.34976398209966575,
0.45029494701453726,
0.5571090972365845,
0.6764896180730064,
0.7979645340118087,
0.9152506597458286,
1.032536785479835,
1.14563412100908,
1.250353876128747,
1.348790445941224,
1.4388494353441044,
1.524719634542254,
1.6022122533308005,
1.667138501504987,
1.7174039839624031,
1.7488199104983222,
1.7571974909079064,
1.7488199104983222,
1.7236871692696094,
1.7006488231432322,
1.6922712427337065,
1.7048376133480596,
1.7278759594743915,
1.7592918860102875,
1.788613417443796,
1.8221237390821063,
1.8451620852084116,
1.8640116411298915,
1.8619172460275535,
1.8388788999012349,
1.7928022076485848,

```
1.7257815643719905,
1.6482889455834573,
1.5666075365901222,
1.4744541520847692,
1.3739231871699435,
1.2650146418454877,
1.1414453308043044,
1.0074040442511158,
0.8607963870836142,
0.7079055446088959,
0.5424483315198377,
0.36861353802121893,
0.1884955592153632,
0.006283185307189876,
-0.1822123739082133,
-0.3811799086355685,
-0.5843362335676859,
-0.7895869536022443,
-0.9906488834319476,
-1.1770500475450028,
-1.348790445941224,
-1.512153263927881,
-1.665044106402586,
-1.8032741831605634,
-1.9163715186897814,
-2.0022417178879053,
-2.0608847807548405,
-2.0860175219836417,
-2.0860175219836417,
-2.0629791758572966,
-2.0231856689118226,
-1.979203371761587,
-1.937315469713712,
-1.9017107529729764,
-1.8619172460275535,
-1.8158405537749032,
-1.7488199104983289,
-1.6566665259930284,
-1.5393804002590017,
-1.4116222990130163,
-1.2775810124598541,
-1.1519173063162325,
-1.0325367854798422,
-0.9299114254625895,
-0.8398524360596701,
-0.7665486074759126,
-0.7099999397112969,
-0.6639232474586533,
```

-0.6366961111275091,
-0.615752160103602,
-0.5989969992844467,
-0.5801474433629238,
-0.5571090972365986,
-0.5277875658030836,
-0.4921828490623987,
-0.45238934211693166,
-0.41887902047863,
-0.3937462792499216,
-0.3728023282259909,
-0.3644247478164199,
-0.37280232822598425,
-0.40421825476189366,
-0.4586725274241082,
-0.5215043804958999,
-0.5864306286700937,
-0.6408849013323216,
-0.6806784082777819,
-0.6974335690969371,
-0.6827728033801896,
-0.6304129258203561,
-0.5319763560078724,
-0.4021238596594819,
-0.27436575841351385,
-0.1549852375770927,
-0.06283185307179218,
0.008377580409570958,
0.07120943348136313,
0.12985249634837984,
0.1759291886010254,
0.2073451151369262,
0.22200588085368034,
0.2303834612632513,
0.2366666465704345,
0.2597049926967597,
0.2953097094374379,
0.34976398209966575,
0.4209734155810177,
0.4879940588576232,
0.5466371217246199,
0.5906194188748891,
0.6304129258203495,
0.6702064327658231,
0.7183775201208679,
0.7665486074758988,
0.8230972752405215,
0.877551547902756,

0.9341002156673717,
0.9969320687391638,
1.0639527120157448,
1.1351621454971212,
1.2021827887736956,
1.260825851640672,
1.3152801243029335,
1.346696050838823,
1.3697343969651614,
1.3843951626819022,
1.4074335088082408,
1.4430382255489123,
1.4912093129039636,
1.5603243512829257,
1.6503833406858517,
1.7488199104983155,
1.847256480310714,
1.9268434942017467,
1.9875809521711512,
2.0294688542190125,
2.048318410140542,
2.04412961993578,
2.0252800640142237,
1.98548655706875,
1.9310322844065355,
1.857728455822778,
1.7739526517270416,
1.671327291709776,
1.5561355610781302,
1.4220942745250083,
1.2733922222550653,
1.107935009165935,
0.9320058205649839,
0.7497934466567706,
0.5654866776461562,
0.3769911184307797,
0.18430676901060106,
-0.00418879020478881,
-0.19687313962496747,
-0.38117990863554185,
-0.5612978874413941,
-0.741415866247193,
-0.9215338450530185,
-1.1100294042683818,
-1.3069025438933626,
-1.5100588688255066,
-1.709026403552744,
-1.897521962768265,

-2.071356756266874,
-2.2200588085368036,
-2.3331561440660082,
-2.416931948161771,
-2.4651030355167958,
-2.47976380123355,
-2.4525366649024427,
-2.3959879971378135,
-2.308023402837315,
-2.2033036477176484,
-2.0881119170860227,
-1.9666370011472203,
-1.834690109696446,
-1.696460032938405,
-1.5561355610781435,
-1.4137166941154107,
-1.2754866173574597,
-1.13935093570191,
-1.009498439353517,
-0.8838347332099326,
-0.7749261878854902,
-0.672300827868211,
-0.5801474433629238,
-0.5005604294719763,
-0.43353978619539535,
-0.38117990863556184,
-0.3455751918948703,
-0.3267256359733473,
-0.31206487025656315,
-0.307876080051811,
-0.307876080051791,
-0.31625366046137526,
-0.337197611485306,
-0.3665191429188143,
-0.404218254761887,
-0.4419173666049663,
-0.48380526865282775,
-0.529881960905478,
-0.5780530482605293,
-0.6262241356155607,
-0.666017642561041,
-0.6995279641993315,
-0.7204719152232623,
-0.7288494956327943,
-0.7141887299160857,
-0.6806784082777886,
-0.6241297405131729,
-0.548731516827021,

-0.4607669225265059,
-0.3686135380212021,
-0.278554548618296,
-0.18849555921538652,
-0.10681415022205802,
-0.03141592653589942,
0.03560471674068823,
0.08796459430051172,
0.12775810124598874,
0.163362817986667,
0.1884955592153798,
0.21362830044410272,
0.23876104167282894,
0.2680825731063306,
0.31206487025658314,
0.3665191429188143,
0.4377285764001775,
0.5235987755983015,
0.6220353454107852,
0.7309438907352209,
0.8419468311620711,
0.9571385617936968,
1.0618583169133438,
1.1644836769306297,
1.2482594810263459,
1.323657704712434,
1.3781119773747257,
1.415811089217805,
1.4409438304465245,
1.4660765716752373,
1.4953981031087524,
1.5289084247470361,
1.5645131414877278,
1.593834672921223,
1.6210618092523368,
1.6482889455834573,
1.6734216868121703,
1.7006488231432841,
1.7299703545767726,
1.767669466419865,
1.8095573684676303,
1.8493508754132004,
1.889144382358674,
1.9310322844065355,
1.9624482109424315,
1.9875809521711378,
2.0106192982974695,
2.0252800640142503,

```
 2.0315632493214,
 2.0273744591166114,
 2.008524903195082,
 1.972920186454397,
 1.9184659137921825,
 1.8388788999012216,
 1.7383479349863569,
 1.614778623945074,
 1.4702653618800463,
 1.311091334098138,
 1.126784565087537,
 0.9236282401554063,
 0.6953391739945294,
 0.4586725274241282,
 0.21572269554648382,
 -0.018849555921516336,
 -0.24923301718480761,
 -0.47123889803845465,
 -0.6869615935849784,
 -0.8984954989266868,
 -1.1079350091660074,
 -1.3194689145077156,
 -1.5289084247469549,
 -1.740442330088758,
 -1.9415042599185008,
 -2.136283004441054,
 -2.3143065881445053,
 -2.4692918257215846,
 -2.5886723465579924,
 -2.674542545756116,
 -2.7227136331111677,
 -2.735280003725534,
 -2.7143360527015767,
 -2.6703537555513406,
 -2.609616297581923,
 -2.53212367879339]

[-1.0053096491485114,
 -1.0053096491501368,
 -1.2566370614349596,
 -0.7539822368619833,
 -0.502654824574842,
 -0.5026548245728698,
 -1.5987211554602258e-12,
 -0.2513274122865284,
 -1.2566370614358393,
 -2.5132741228716786,
 -3.2672563597331825,
```

-4.021238596595318,
-4.021238596595645,
-5.026548245741785,
-6.031857894892669,
-6.78584013175705,
-8.293804605474621,
-9.550441666912059,
-10.555751316062969,
-12.063715789783739,
-14.325662500370488,
-16.58760921095436,
-18.84955592153882,
-20.357520395263155,
-21.362830044407676,
-21.36283004441087,
-20.106192982976633,
-18.095573684676413,
-15.833626974092855,
-13.320352851220996,
-12.817698026643694,
-13.823007675795385,
-14.576989912659165,
-15.833626974092216,
-16.83893662324131,
-16.58760921095159,
-16.336281798670093,
-15.582299561803634,
-13.823007675797038,
-11.812388377494319,
-9.299114254628485,
-7.037167544040564,
-5.026548245743234,
-3.5185837720210813,
-2.2619467105819453,
-1.0053096491524518,
0.25132741228967337,
2.010619298295471,
3.518583772021721,
5.529203070317329,
7.037167544042642,
9.047786842336787,
10.053096491487281,
12.063715789785643,
12.817698026647736,
13.823007675793095,
15.330972149518438,
16.084954386380048,
16.084954386379422,

```
15.582299561806408,
14.576989912656572,
12.315043202071529,
10.555751316060647,
8.04247719319135,
5.277875658029684,
3.267256359734675,
0.5026548245730584,
-1.2566370614350442,
-2.261946710584605,
-3.5185837720215276,
-4.021238596594073,
-4.272566008882208,
-3.769911184306778,
-2.261946710587499,
-1.507964473720774,
-0.5026548245746438,
-0.2513274122873519,
0.5026548245738711,
0.7539822368619857,
0.7539822368619857,
1.0053096491485083,
1.2566370614358435,
2.261946710583566,
3.0159289474478563,
4.52389342116877,
5.780530482604581,
6.283185307180044,
7.539822368614665,
6.534512719467705,
6.785840131753898,
5.277875658030853,
4.775220833455868,
3.7699111843091426,
2.261946710583567,
1.256637061435818,
0.7539822368620086,
0.7539822368619883,
1.005309649147719,
2.513274122872398,
3.518583772020265,
5.52920307031805,
7.037167544042029,
8.54513201776318,
10.053096491487873,
12.063715789784624,
12.817698026645717,
14.325662500370491,
```

14.576989912656318,
14.074335088082448,
14.074335088080822,
13.571680263509247,
12.566370614360087,
11.812388377497303,
10.80707872834567,
10.304423903777856,
9.299114254625616,
7.791149780902416,
6.031857894889944,
3.7699111843102515,
1.0053096491501137,
-1.0053096491501137,
-3.015928947445545,
-2.7646015351652773,
-1.0053096491430524,
1.5079644737223727,
2.7646015351598416,
3.7699111843075306,
3.518583772021028,
4.021238596597257,
2.764601535156644,
2.2619467105775906,
-0.25132741228055394,
-2.764601535158243,
-5.529203070318031,
-8.042477193191344,
-9.299114254624017,
-9.801769079200247,
-11.058406140642404,
-12.063715789778794,
-13.069025438934744,
-14.828317324942049,
-16.084954386382687,
-17.592918860100266,
-18.34690109696626,
-19.854865570687046,
-20.86017521983433,
-21.614157456702188,
-21.865484868980875,
-22.61946710584846,
-23.87610416728271,
-24.37875899185418,
-24.63008640414709,
-24.12743157956448,
-22.368139693566107,
-20.608847807546635,

```
-19.603538158398894,
-18.346901096964675,
-16.587609210957346,
-13.571680263506204,
-10.304423903774904,
-7.0371675440322505,
-3.0159289474560693,
0.0,
2.7646015351614137,
4.775220833456898,
5.277875658028305,
5.0265482457449995,
4.2725660088882895,
4.775220833450643,
5.529203070318058,
8.042477193188946,
11.05840614063609,
14.07433508808326,
15.330972149518304,
16.08495438637952,
15.079644737234652,
14.325662500366503,
12.315043202070362,
10.807078728350373,
8.796459430050932,
6.785840131753901,
5.5292030703172586,
3.26725635973731,
2.5132741228687956,
2.010619298298642,
2.2619467105827598,
2.764601535159029,
3.518583772021814,
4.272566008882208,
4.775220833456059,
4.021238596596212,
3.0159289474449382,
2.5132741228716937,
1.0053096491485216,
-1.0053096491477223,
-3.7699111843091426,
-6.534512719465767,
-7.539822368615028,
-7.7911497809030745,
-6.534512719467372,
-4.7752208334552595,
-2.0106192982986286,
1.7592918860097013,
```

6.283185307180044,
11.812388377498088,
15.582299561806918,
15.33097214951581,
14.325662500370589,
11.058406140636103,
8.545132017763606,
7.539822368615088,
7.037167544042029,
5.529203070317489,
3.769911184308006,
1.759291886010504,
1.0053096491485183,
0.7539822368619891,
2.764601535159029,
4.272566008881402,
6.534512719467366,
8.545132017762263,
8.042477193192475,
7.037167544039625,
5.277875658032328,
4.7752208334552595,
4.775220833456858,
5.7805304826053865,
5.7805304826037345,
6.7858401317545605,
6.534512719468172,
6.785840131753901,
7.539822368615088,
8.042477193189745,
8.545132017765201,
8.042477193188946,
7.037167544037207,
6.534512719471222,
3.769911184306758,
2.7646015351606144,
1.759291886008902,
2.764601535160641,
4.272566008880589,
5.780530482606186,
8.293804605475476,
10.80707872835116,
11.812388377495704,
11.81238837748787,
9.55044166692344,
7.288494956328559,
5.026548245743374,
2.261946710583559,

-0.5026548245714597,
-2.2619467105867566,
-4.775220833456845,
-6.5345127194657735,
-8.796459430050932,
-10.0530964914884,
-12.315043202071934,
-13.82300767579753,
-16.084954386374694,
-17.844246272393214,
-19.854865570695708,
-21.11150263211308,
-21.865484868985675,
-22.116812281273805,
-22.619467105845263,
-23.12212193042152,
-22.619467105846866,
-23.12212193042152,
-22.116812281269006,
-21.614157456702344,
-21.61415745669595,
-21.614157456699132,
-22.61946710584368,
-23.624776754997775,
-24.378758991857364,
-23.876104167268583,
-22.61946710586138,
-20.86017521983319,
-17.84424627239159,
-13.571680263504605,
-10.053096491491544,
-5.780530482603015,
-1.7592918860105007,
3.2672563597328734,
6.785840131755527,
10.55575131605986,
12.566370614360034,
13.823007675795132,
14.576989912656344,
15.833626974092962,
16.58760921096499,
16.838936623230534,
17.090264035528005,
16.587609210954174,
16.33628179866602,
15.58229956180723,
15.079644737230176,
13.069025438933133,

```
12.315043202073545,
11.058406140634505,
9.55044166691373,
8.042477193189745,
6.283185307180044,
4.272566008883,
2.2619467105827664,
1.7592918860141045,
0.5026548245702339,
2.3980817331903466e-12,
-1.0053096491501137,
-2.5132741228717004,
-3.518583772021008,
-4.523893421168737,
-4.52389342116953,
-5.0265482457433945,
-5.529203070318052,
-5.7805304826061725,
-5.780530482603788,
-4.775220833457658,
-4.0212385965948725,
-2.5132741228717004,
-1.005309649143852,
1.7592918860049447,
4.021238596595672,
6.785840131753901,
9.04778684233826,
10.555751316061851,
11.058406140636496,
10.807078728348774,
10.807078728349174,
9.801769079199454,
9.047786842339065,
8.042477193190546,
6.283185307178841,
4.77522083345726,
4.272566008881405,
3.0159289474455484,
3.0159289474466,
3.015928947447157,
3.518583772020212,
5.277875658030323,
6.534512719467766,
8.54513201776361,
10.304423903774916,
11.812388377498088,
13.069025438932334,
13.320352851222074,
```

13.823007675795132,
12.566370614357677,
12.315043202074358,
10.053096491485976,
9.047786842330613,
6.534512719474671,
4.523893421169543,
3.015928947446344,
3.015928947445545,
3.5185837720218274,
4.02123859659406,
4.272566008883014,
3.5185837720194293,
3.267256359733673,
3.267256359734472,
3.0159289474455715,
3.267256359733673,
3.51858377201863,
4.523893421171115,
5.026548245731837,
4.775220833468182,
4.775220833456845,
5.026548245743401,
3.7699111843075306,
3.015928947444772,
2.764601535159815,
1.7592918860136981,
0.7539822368579889,
-0.5026548245746572,
-2.261946710583559,
-4.272566008882188,
-6.5345127194657735,
-9.550441666915342,
-12.063715789783805,
-14.828317324953986,
-17.34159144780247,
-19.10088333382906,
-22.116812281272207,
-24.378758991855765,
-27.39468793930533,
-28.399997588448247,
-29.153979825317425,
-28.148670176160117,
-27.646015351595054,
-26.64070570243774,
-25.886723465582946,
-25.384068641005094,
-25.13274122871856,

```
 -25.38406864100508,
 -25.1327412287088,
 -25.384068641015105,
 -24.127431579569237,
 -23.37344934270645,
 -21.362830044414242,
 -18.59822850924958,
 -14.32566250036899,
 -10.304423903774877,
 -5.7805304826062125,
 -1.507964473723971 5,
 2.5132741228748845,
 5.277875658028358,
 7.288494956330131,
 9.299114254624017,
 10.304423903777275]

[2.37458493223537,
 2.53541101912472,
 2.71535162581618,
 2.79712712480384,
 2.89753012980830,
 2.88033858629321,
 2.68583182360936,
 2.43815069603204,
 2.24715930653493,
 2.08851568149113,
 1.79325891224888,
 1.42505522402717,
 0.991032435352766,
 0.545332315715792,
 0.173334316294113,
 -0.092169202052051 0,
 -0.356910119386626,
 -0.556229603336114,
 -0.688975930569938,
 -0.828206038295802,
 -0.835397119784771,
 -0.781750442214531,
 -0.687441173315559,
 -0.708922081786844,
 -0.795921606658805,
 -1.07442897670189,
 -1.49510915981511,
 -2.20626982987132,
 -2.96034947568970,
 -3.48177435558243,
 -3.58499282214374,
```

-3.28070835966980,
-2.80585816932437,
-2.25792935162583,
-1.77749832239307,
-1.41077217511196,
-1.12867900018221,
-1.08253711325750,
-1.25301799295598,
-1.54378903345873,
-1.92268190898073,
-2.24654769912022,
-2.36051680528867,
-2.34040995004839,
-2.10944678691686,
-1.70335464920963,
-1.12477803821310,
-0.445718968651639,
0.254523974009796,
0.961267715399454,
1.47241821070387,
1.92500823913052,
2.14231021889541,
2.30519479941741,
2.29716188141883,
2.18409345531697,
2.00170604267668,
1.83884777037637,
1.74221310907935,
1.69803553221544,
1.82354052958535,
1.99054295327268,
2.15441539257439,
2.23922652070092,
2.33824689986800,
2.37553314719150,
2.43116487623577,
2.38953401573107,
2.33674922594060,
2.27846321634097,
2.11467998491661,
1.96778270346646,
1.81103358913916,
1.65489437042977,
1.54398385163993,
1.42466459383764,
1.39739624590167,
1.27591036189222,
1.23288083556482,

1.15896138194169,
1.14700343457193,
1.06650843576467,
1.06896289926102,
0.991928912202756,
0.877665545486183,
0.731466811825421,
0.507826198895476,
0.318038281521116,
0.0586009643994303,
-0.122291360236784,
-0.376208386354174,
-0.418376430243631,
-0.456868413003972,
-0.395493788206801,
-0.167758652059083,
0.130279633982383,
0.474923218929945,
0.807988707282930,
1.19654113391877,
1.47088108570767,
1.76738430941024,
1.95090156999584,
2.06708005124034,
2.01997722774019,
1.87850256425890,
1.63997089338220,
1.43676090659792,
1.23166358260621,
0.997880582550426,
0.788092842472417,
0.750751163132323,
0.719070606455161,
0.867757137822967,
1.15213933452118,
1.55710229542858,
2.04099380103566,
2.55780302025742,
2.97842830918301,
3.45801182577891,
3.81154007574963,
4.23352299807736,
4.39243625332261,
4.33713481039933,
3.87117503894947,
3.06921548857122,
2.22500050219847,
1.35294394102616,

0.565172384847906,
-0.121351529445584,
-0.789952785910573,
-1.22102419271835,
-1.52754976465975,
-1.55473933274663,
-1.50787791122609,
-1.43066664899792,
-1.43284557668049,
-1.42823113730022,
-1.47994869915601,
-1.43806859788781,
-1.50380004881975,
-1.58170300977571,
-1.74449823202215,
-1.87663224170525,
-1.95577201306974,
-2.04489394832649,
-2.13199851516969,
-2.18584733869068,
-2.21623083854497,
-2.14122994181443,
-1.87838434777639,
-1.52082392002619,
-1.21558627875285,
-1.04638268106299,
-0.805146619215699,
-0.524378388661950,
-0.320144229369448,
-0.171536077031253,
-0.217033234070275,
-0.383298551653490,
-0.735231827070570,
-1.11124975360824,
-1.45717410052883,
-1.78032361166821,
-2.06924230002622,
-2.03314953166972,
-1.72614537543678,
-1.05151819036077,
-0.0938648337055176,
0.830491212979932,
1.47369440998143,
1.87993463942697,
1.97759185013814,
1.81934778705237,
1.56615298634576,
1.41128029754344,

1.30077120250425,
1.38855354655352,
1.65575864159937,
2.03327535909121,
2.26663431367170,
2.54453468620915,
2.89915978881068,
2.99733303239569,
3.01997135388723,
2.86900824539646,
2.56785106212009,
2.33372988266067,
2.11793561529295,
1.86172102022318,
1.76822203621511,
1.57920925326627,
1.40859057158386,
1.26375001379820,
1.00916740009926,
0.880746426486188,
0.747365006387870,
0.831536472501864,
0.991478048649568,
1.28084357761487,
1.58833318619039,
1.89654879152089,
2.08382393351076,
2.10281774041561,
1.84672327798251,
1.56540934055042,
1.23742042720609,
0.911243157239476,
0.700547879001012,
0.372967276884770,
0.127651417697091,
-0.0522426994629726,
-0.257707659607243,
-0.432159061627324,
-0.423478150559904,
-0.183534023557145,
0.0865377731563774,
0.379246848531574,
0.600719087537192,
0.851169582296447,
1.11199071516310,
1.31722578323350,
1.54020577573228,
1.77046392385542,

```
2.07008036197878,
2.33294664450427,
2.43201884028784,
2.34066245863830,
2.04502274722664,
1.81985267544246,
1.52123724588481,
1.41326075003249,
1.32942863463229,
1.35641612299481,
1.52903843098497,
1.73473525350266,
2.04279550690847,
2.40246347412489,
2.69805244690665,
3.12380983974689,
3.34918642541620,
3.29464992574070,
2.84601844676672,
2.04520622956515,
0.973459770929467,
-0.0495195638273093,
-0.720330126681764,
-1.19584690354048,
-1.47997711180648,
-1.63293124966863,
-1.63105520578438,
-1.56226062294776,
-1.27127363889373,
-0.982833394728821,
-0.705666284870486,
-0.508412003861125,
-0.465961096036631,
-0.405898065088628,
-0.499110274425991,
-0.542213594693419,
-0.683942838312524,
-0.680160442568611,
-0.794836304251133,
-0.963980664357190,
-1.15588801614652,
-1.34923300002753,
-1.57523919179938,
-1.67767344824183,
-1.77913668729539,
-1.81937384739947,
-1.83669106379736,
-1.72974115687304,
```

-1.58134417009204,
-1.28776935588232,
-0.996226788385274,
-0.706979011183793,
-0.536052258747343,
-0.441890221301216,
-0.446544447959346,
-0.625512344077446,
-0.798348234277397,
-1.03046380796191,
-1.32037977332058,
-1.51647651450990,
-1.70430398839660,
-1.74099368458762,
-1.70907235002173,
-1.48847526117933,
-1.02810840154425,
-0.492525300427788,
0.127266497210867,
0.641682189536678,
1.12511884462002,
1.48532854864884,
1.67553469038672,
1.69021575849138,
1.62122577093692,
1.67363325540104,
1.85430577061458,
2.04083159729013,
2.20773981614017,
2.35341961029397,
2.51289109493072,
2.60214642067591,
2.68335862862224,
2.63815337233702,
2.43132503597525,
2.12823739570138,
1.72986756285683,
1.51405728035090,
1.33022545766517,
1.24905037632719,
1.24010102230214,
1.16244566579778,
1.17844603645418,
1.12861331085284,
1.08235758875940,
1.09810515521836,
1.05596466796138,
1.10783744127622,

1.05668126634454,
1.06511528153338,
0.981017320701407,
0.977517659160762,
0.835705107367067,
0.742257180069464,
0.606215345589318,
0.561523880332962,
0.525091833656777,
0.459193376207291,
0.406157662137246,
0.240348476725220,
0.208071256211169,
0.142894981162377,
0.120291548347883,
0.0616870570492459,
0.192239265512194,
0.389429953333776,
0.641653425060474,
0.944597853652721,
1.29586962091933,
1.58302381801034,
1.80487399192662,
1.84601265552949,
1.80878964451341,
1.64493352245057,
1.45049618582695,
1.26566978943382,
1.11934362706156,
1.07082670269049,
1.11712108317344,
1.22705772723290,
1.51516292607050,
1.81778376597393,
2.22081903258372,
2.44010965116183,
2.68457179443071,
2.86499699562554,
3.17294239174878,
3.64642002273844,
4.16238928709426,
4.57659145972332,
4.76261017316802,
4.62772684793620,
4.24499829228963,
3.56279806467952,
2.64903634499104,
1.64642008007233,

```
0.743459622832136,
-0.0862734426315485,
-0.729885934360766,
-1.32553211978379,
-1.82975375514394,
-2.26187294336344,
-2.60521878353510,
-2.69166558764497,
-2.74585512735467,
-2.58131491821398,
-2.31546319218056,
-2.07750294253581,
-1.74080977796506,
-1.50775999409640,
-1.41094655782713,
-1.47078094566135,
-1.58214821307130,
-1.70525004313246,
-1.73463000463735,
-1.75465769879555,
-1.75076144279425,
-1.81720052094792,
-1.72793119906239,
-1.57866438712533,
-1.40824582078194,
-1.22854998679261,
-1.03995015643837,
-0.700525014954498,
-0.486612059389739,
-0.315459723637888,
-0.168169693708658,
-0.0638339494714057,
-0.0464419062922593,
-0.0845930726697237,
-0.433916136879505,
-1.02393514279949,
-1.69529587697098,
-2.24187773474466,
-2.53888368976720,
-2.62626061103947,
-2.38820483991652,
-1.82531969503752,
-1.11142149836049,
-0.405735186932426]
```

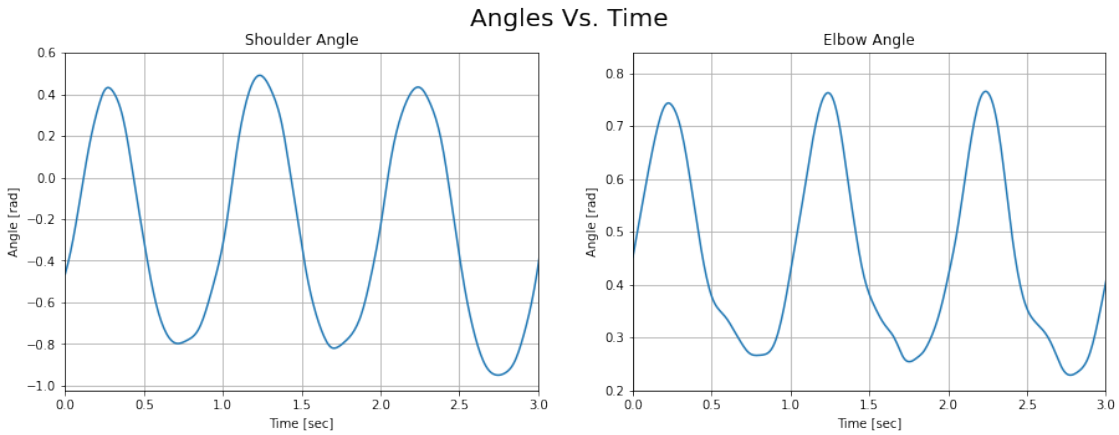
Example for the trial with the largest elbow torque & power:

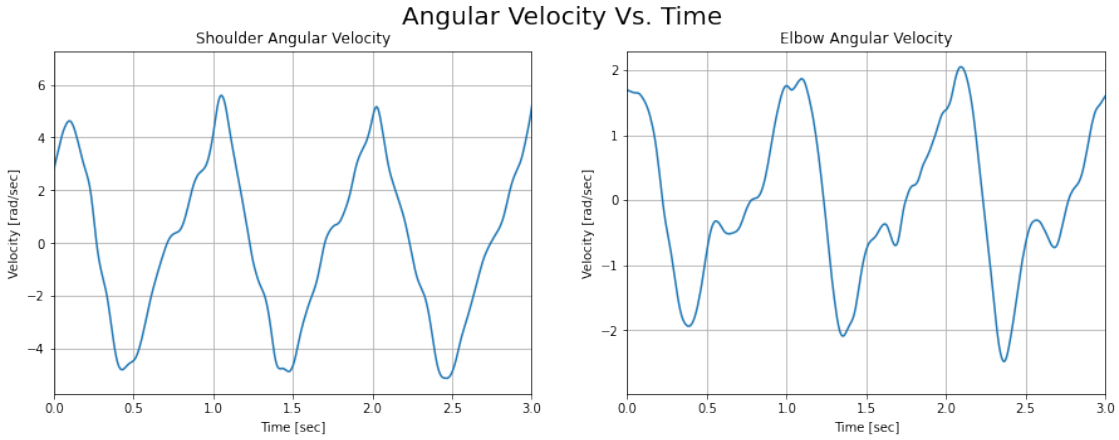
[10]:
```
index = 125

t_list = time_list[index]
theta1_list = Sholder_Ang_list[index]
theta2_list = Elbow_Ang_list[index]
dtheta1_list = Sholder_Vel_list[index]
dtheta2_list = Elbow_Vel_list[index]
ddtheta1_list = Sholder_Acc_list[index]
ddtheta2_list = Elbow_Acc_list[index]
tau1_list = Sholder_tau_list[index]
tau2_list = Elbow_tau_list[index]
current1_list = Sholder_current_list[index]
current2_list = Elbow_current_list[index]
power1_list = Sholder_power_list[index]
power2_list = Elbow_power_list[index]

# Compute the trajectory of the arm's motion
N = int((max(t_list) - min(t_list))/(1/frame_frequency))
tvec = np.linspace(min(t_list), max(t_list), N)
traj = np.zeros((6, N))
for i in range(N):
    traj[0, i] = theta1_list[i]
    traj[1, i] = theta2_list[i]
    traj[2, i] = dtheta1_list[i]
    traj[3, i] = dtheta2_list[i]
    traj[4, i] = ddtheta1_list[i]
    traj[5, i] = ddtheta2_list[i]

# Calculate the length difference between the time list and the trajectory lists
diff = (len(t_list) - len(traj[0]))

# Plot the trajectory lists (angles, velocities, accelerations, torques, and␣
 ↪power)
plt.figure(figsize=(15,5))
plt.suptitle('Angles Vs. Time', fontsize=20)
plt.subplot(121)
plt.plot(t_list[:-diff], traj[0])
plt.ylabel('Angle [rad]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Shoulder Angle')

plt.subplot(122)
plt.plot(t_list[:-diff], traj[1])
plt.ylabel('Angle [rad]')
plt.xlabel('Time [sec]')
```

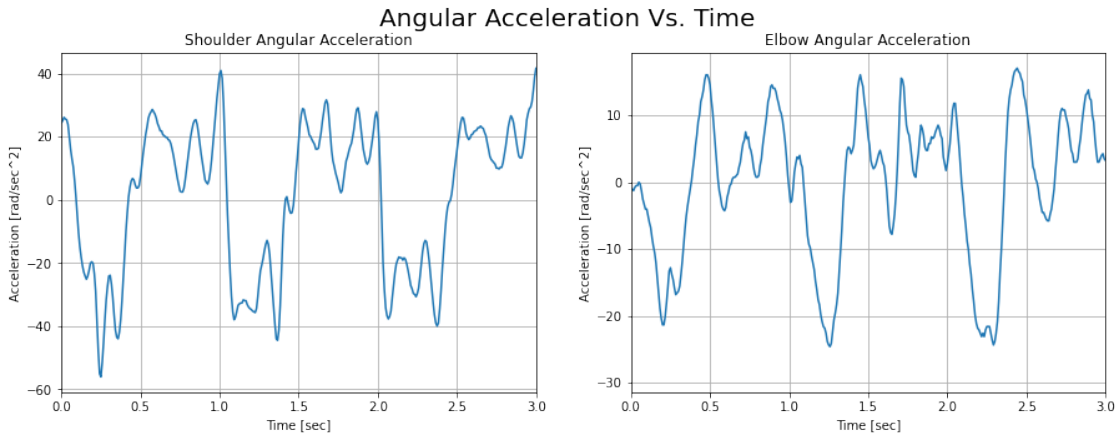
```
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Elbow Angle')
plt.show()

plt.figure(figsize=(15,5))
plt.suptitle('Angular Velocity Vs. Time', fontsize=20)
plt.subplot(121)
plt.plot(t_list[:-diff], traj[2])
plt.ylabel('Velocity [rad/sec]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Shoulder Angular Velocity')

plt.subplot(122)
plt.plot(t_list[:-diff], traj[3])
plt.ylabel('Velocity [rad/sec]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Elbow Angular Velocity')
plt.show()

plt.figure(figsize=(15,5))
plt.suptitle('Angular Acceleration Vs. Time', fontsize=20)
plt.subplot(121)
plt.plot(t_list[:-diff], traj[4])
plt.ylabel('Acceleration [rad/sec^2]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Shoulder Angular Acceleration')

plt.subplot(122)
plt.plot(t_list[:-diff], traj[5])
plt.ylabel('Acceleration [rad/sec^2]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Elbow Angular Acceleration')
plt.show()

plt.figure(figsize=(15,5))
plt.suptitle('Torque Vs. Time', fontsize=20)
plt.subplot(121)
plt.plot(t_list, tau1_list)
```

```
plt.ylabel('Torque [Nm]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Shoulder Torque')

plt.subplot(122)
plt.plot(t_list, tau2_list)
plt.ylabel('Torque [Nm]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Elbow Torque')
plt.show()

plt.figure(figsize=(15,5))
plt.suptitle('Power Vs. Time', fontsize=20)
plt.subplot(121)
plt.plot(t_list, power1_list)
plt.ylabel('Power [W]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Shoulder Power')

plt.subplot(122)
plt.plot(t_list, power2_list)
plt.ylabel('Power [W]')
plt.xlabel('Time [sec]')
plt.xlim([0, int(max(tvec))])
plt.grid()
plt.title('Elbow Power')
plt.show()

plt.figure(figsize=(15,5))
plt.suptitle('Speed Vs. Torque', fontsize=20)
plt.subplot(121)
plt.plot(tau1_list[:-diff], traj[2])
plt.ylabel('Velocity [rad/sec]')
plt.xlabel('Torque [Nm]')
plt.grid()
plt.title('Shoulder Speed-Torque')

plt.subplot(122)
plt.plot(tau2_list[:-diff], traj[3])
plt.ylabel('Velocity [rad/sec]')
plt.xlabel('Torque [Nm]')
```

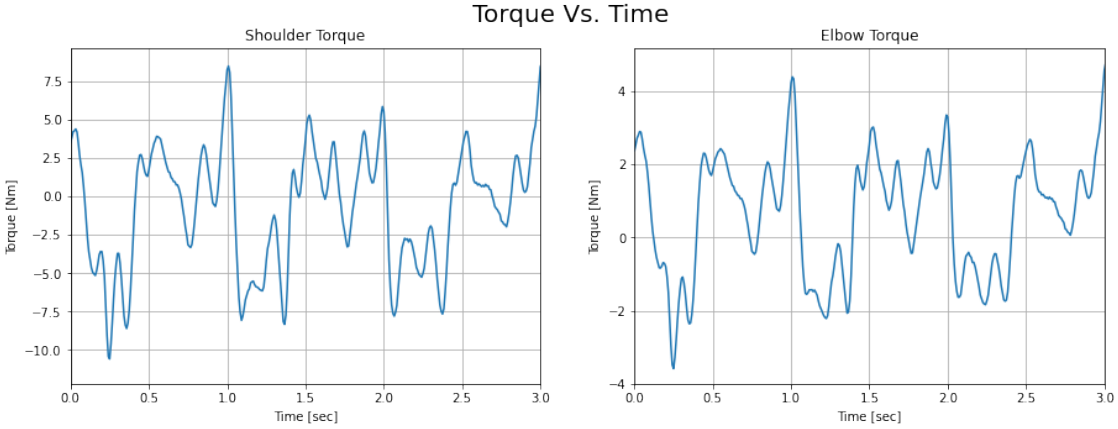
```
plt.grid()
plt.title('Elbow Speed-Torque')
plt.show()
```



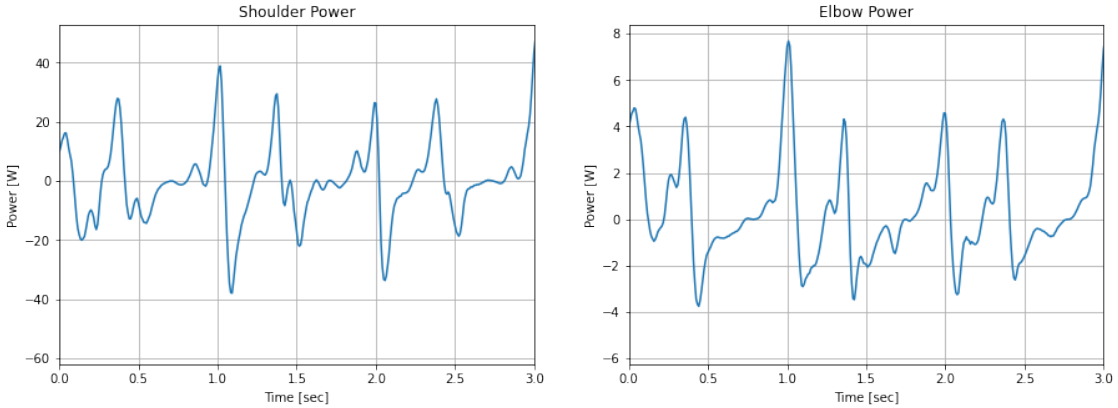




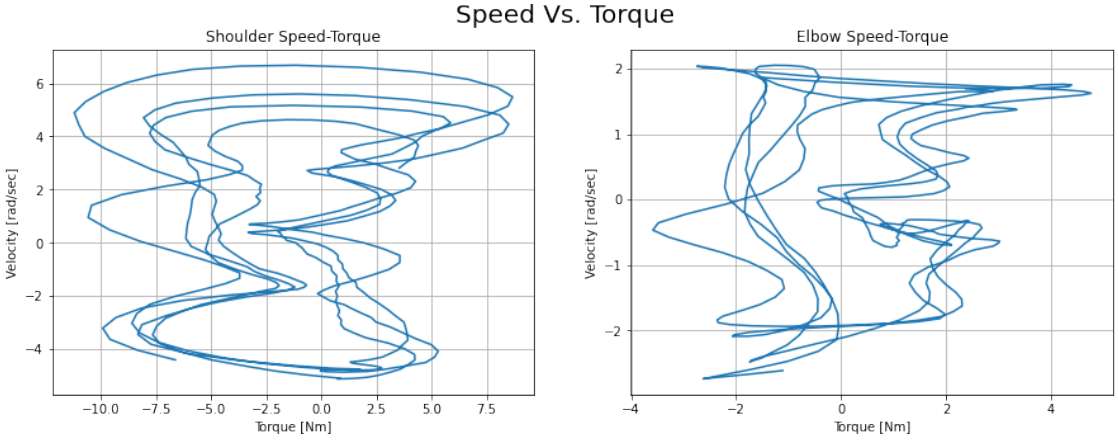
Torque Vs. Time



Power Vs. Time



Speed Vs. Torque



Animating the simulation:

[11]:
```
def animate_double_pend(traj, L1, L2, L1_COM, L2_COM, T):
    """
    Function to generate web-based animation of double-pendulum system

    Parameters:
        traj:         trajectory of theta1 and theta2
        L1:           length of the upper arm
        L2:           length of the lower arm
        L1_COM:       length of the center of mass of the upper arm from the␣
 ↪shoulder
        L2_COM:       length of the center of mass of the lower arm from the␣
 ↪elbow
        T:            length/seconds of animation duration

    Returns: None
    """

    # Browser configuration
    def configure_plotly_browser_state():
        import IPython
        display(IPython.core.display.HTML('''
            <script src="/static/components/requirejs/require.js"></script>
            <script>
              requirejs.config({
                paths: {
                  base: '/static/base',
                  plotly: 'https://cdn.plot.ly/plotly-1.5.1.min.js?noext',
                },
              });
            </script>
            '''))
    configure_plotly_browser_state()
    init_notebook_mode(connected=False)

    # Getting data from pendulum angle trajectories
    xx1 = L1 * np.sin(traj[0])
    yy1 = -L1 * np.cos(traj[0])
    xx1_COM = L1_COM * np.sin(traj[0])
    yy1_COM = -L1_COM * np.cos(traj[0])
    xx2 = xx1 + L2 * np.sin(traj[0] + traj[1])
    yy2 = yy1 - L2 * np.cos(traj[0] + traj[1])
    xx2_COM = xx1 + L2_COM * np.sin(traj[0] + traj[1])
    yy2_COM = yy1 - L2_COM * np.cos(traj[0] + traj[1])
    N = len(traj[0])
```

```
    # Using these to specify axis limits
    xm = np.min(xx1)
    xM = np.max(xx1)
    ym = np.min(yy1) - 0.6
    yM = np.max(yy1) + 0.6

    # Defining data dictionary
    data = [dict(x=xx1, y=yy1,
                 mode='lines', name='Arm',
                 line=dict(width=5, color='blue')
                ),
            dict(x=xx1_COM, y=yy1_COM,
                 mode='lines', name='Upper Arm Center of Mass',
                 line=dict(width=2, color='green')
                ),
            dict(x=xx2_COM, y=yy2_COM,
                 mode='lines', name='Lower Arm Center of Mass',
                 line=dict(width=2, color='orange')
                ),
            dict(x=xx1, y=yy1,
                 mode='markers', name='Elbow Trajectory',
                 marker=dict(color="green", size=2)
                ),
            dict(x=xx2, y=yy2,
                 mode='markers', name='Hand Trajectory',
                 marker=dict(color="orange", size=2)
                )
           ]

    # Preparing simulation layout
    layout = dict(xaxis=dict(range=[xm, xM], autorange=False,␣
↪zeroline=False,dtick=1),
                  yaxis=dict(range=[ym, yM], autorange=False,␣
↪zeroline=False,scaleanchor = "x",dtick=1),
                  title='Simulation of Arm Modeled as a Double Pendulum',
                  hovermode='closest',
                  updatemenus= [{'type': 'buttons',
                                 'buttons': [{'label': 'Play', 'method':␣
↪'animate',
                                              'args': [None, {'frame':␣
↪{'duration': T, 'redraw': False}}]},
                                             {'args': [[None], {'frame':␣
↪{'duration': T, 'redraw': False}, 'mode': 'immediate',
                                              'transition': {'duration':␣
↪0}}],'label': 'Pause', 'method': 'animate'}
                                            ]
```

```
                                }]
                )

    # Defining the frames of the simulation
    frames = [dict(data=[dict(x=[0,xx1[k],xx2[k]],
                              y=[0,yy1[k],yy2[k]],
                              mode='lines',
                              line=dict(color='red', width=4)),
                         go.Scatter(
                              x=[xx1_COM[k]],
                              y=[yy1_COM[k]],
                              mode="markers",
                              marker=dict(color="blue", size=12)),
                         go.Scatter(
                              x=[xx2_COM[k]],
                              y=[yy2_COM[k]],
                              mode="markers",
                              marker=dict(color="purple", size=12)),
                        ]) for k in range(N)]

    # Putting it all together and plotting
    figure = dict(data=data, layout=layout, frames=frames)
    iplot(figure)

# Animate the system
L1 = L_upper_arm_dict[participants_list[index]]
L2 = L_lower_arm
L1_COM = L_upper_arm_COM_dict[participants_list[index]]
L2_COM = L_lower_arm_COM
T = 5

animate_double_pend(traj, L1, L2, L1_COM, L2_COM, T)
```

```
<IPython.core.display.HTML object>
```

1.5 Motor Selection

Plotting the torque-speed curve of all trials and the torque-speed curve of some motors:

[12]:
```
# Compute the torque and speed vectors of the arm's motion for all trials
tot_dtheta1_list = []
tot_dtheta2_list = []
tot_tau1_list = []
tot_tau2_list = []

for lst in range(len(time_list)):
    for i in range(len(time_list[lst])):
        tot_dtheta1_list.append(Sholder_Vel_list[lst][i])
```

```
        tot_dtheta2_list.append(Elbow_Vel_list[lst][i])
        tot_tau1_list.append(Sholder_tau_list[lst][i])
        tot_tau2_list.append(Elbow_tau_list[lst][i])

# Compute the torque and speed vectors of some motors
######################################## T-Motor␣
↪########################################
# T-Motor, GL80 (KV30):
tau_stall_GL80_30 = 1.75
no_load_speed_GL80_30 = 720*2*pi/60
motor_speed_GL80_30 = [2*no_load_speed_GL80_30, no_load_speed_GL80_30, 0, -.
↪5*no_load_speed_GL80_30]
motor_torque_GL80_30 = [-tau_stall_GL80_30, 0, tau_stall_GL80_30, 1.
↪5*tau_stall_GL80_30]

# T-Motor, GL80 (KV60):
tau_stall_GL80_60 = 2.9
no_load_speed_GL80_60 = 1440*2*pi/60
motor_speed_GL80_60 = [2*no_load_speed_GL80_60, no_load_speed_GL80_60, 0, -.
↪5*no_load_speed_GL80_60]
motor_torque_GL80_60 = [-tau_stall_GL80_60, 0, tau_stall_GL80_60, 1.
↪5*tau_stall_GL80_60]

# T-Motor, G80 (KV30):
tau_stall_G80_30 = 2.9
no_load_speed_G80_30 = 700*2*pi/60
motor_speed_G80_30 = [2*no_load_speed_G80_30, no_load_speed_G80_30, 0, -.
↪5*no_load_speed_G80_30]
motor_torque_G80_30 = [-tau_stall_G80_30, 0, tau_stall_G80_30, 1.
↪5*tau_stall_G80_30]

# T-Motor, G80 (KV60):
tau_stall_G80_60 = 2.9
no_load_speed_G80_60 = 1400*2*pi/60
motor_speed_G80_60 = [2*no_load_speed_G80_60, no_load_speed_G80_60, 0, -.
↪5*no_load_speed_G80_60]
motor_torque_G80_60 = [-tau_stall_G80_60, 0, tau_stall_G80_60, 1.
↪5*tau_stall_G80_60]

# T-Motor, GL60 (KV25):
tau_stall_GL60_25 = 1.75
no_load_speed_GL60_25 = 600*2*pi/60
motor_speed_GL60_25 = [2*no_load_speed_GL60_25, no_load_speed_GL60_25, 0, -.
↪5*no_load_speed_GL60_25]
motor_torque_GL60_25 = [-tau_stall_GL60_25, 0, tau_stall_GL60_25, 1.
↪5*tau_stall_GL60_25]
```

```
# T-Motor, GL60 (KV55):
tau_stall_GL60_55 = 1.75
no_load_speed_GL60_55 = 1200*2*pi/60
motor_speed_GL60_55 = [2*no_load_speed_GL60_55, no_load_speed_GL60_55, 0, -.
 ↪5*no_load_speed_GL60_55]
motor_torque_GL60_55 = [-tau_stall_GL60_55, 0, tau_stall_GL60_55, 1.
 ↪5*tau_stall_GL60_55]

# T-Motor, GL100 (KV10):
tau_stall_GL100 = 7.7
no_load_speed_GL100 = 250*2*pi/60
motor_speed_GL100 = [2*no_load_speed_GL100, no_load_speed_GL100, 0, -.
 ↪5*no_load_speed_GL100]
motor_torque_GL100 = [-tau_stall_GL100, 0, tau_stall_GL100, 1.5*tau_stall_GL100]

# T-Motor, G100 (KV10):
tau_stall_G100 = 7.7
no_load_speed_G100 = 250*2*pi/60
motor_speed_G100 = [2*no_load_speed_G100, no_load_speed_G100, 0, -.
 ↪5*no_load_speed_G100]
motor_torque_G100 = [-tau_stall_G100, 0, tau_stall_G100, 1.5*tau_stall_G100]

# T-Motor, R60 (KV115):
tau_stall_R60 = 16.96
no_load_speed_R60 = 5520*2*pi/60
motor_speed_R60 = [2*no_load_speed_R60, no_load_speed_R60, 0, -.
 ↪5*no_load_speed_R60]
motor_torque_R60 = [-tau_stall_R60, 0, tau_stall_R60, 1.5*tau_stall_R60]

# T-Motor, R80 (KV110):
tau_stall_R80 = 17.73
no_load_speed_R80 = 5280*2*pi/60
motor_speed_R80 = [2*no_load_speed_R80, no_load_speed_R80, 0, -.
 ↪5*no_load_speed_R80]
motor_torque_R80 = [-tau_stall_R80, 0, tau_stall_R80, 1.5*tau_stall_R80]

# T-Motor, AK60-6:
tau_stall_AK60_6 = 37.49
no_load_speed_AK60_6 = 560*2*pi/60
motor_speed_AK60_6 = [2*no_load_speed_AK60_6, no_load_speed_AK60_6, 0, -.
 ↪5*no_load_speed_AK60_6]
motor_torque_AK60_6 = [-tau_stall_AK60_6, 0, tau_stall_AK60_6, 1.
 ↪5*tau_stall_AK60_6]

# T-Motor, AK80-6:
```

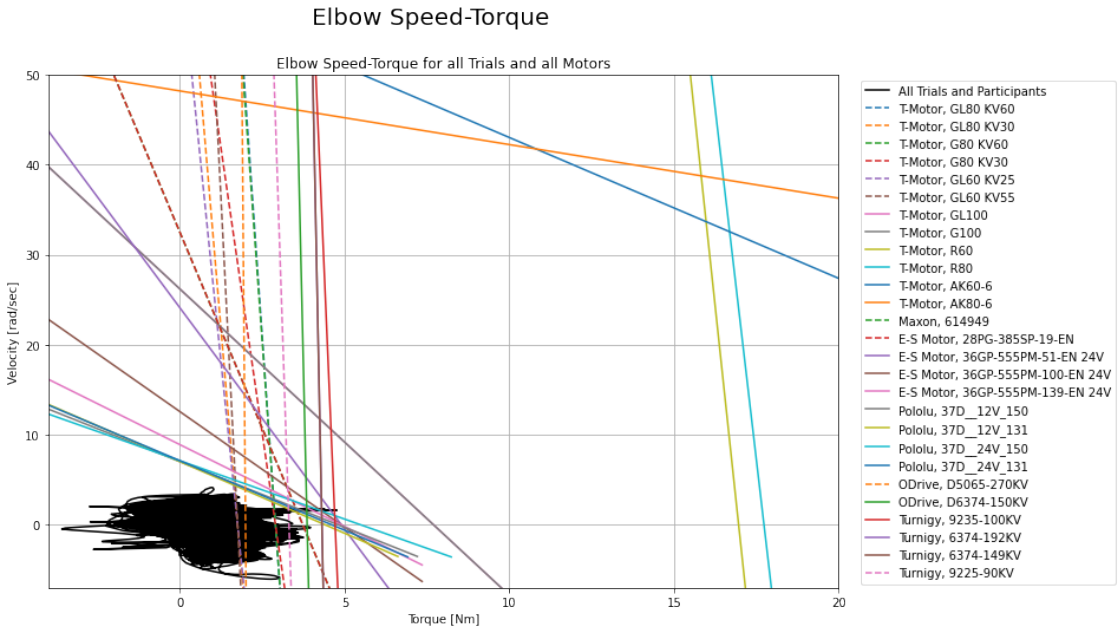
```
tau_stall_AK80_6 = 80.888
no_load_speed_AK80_6 = 460*2*pi/60
motor_speed_AK80_6 = [2*no_load_speed_AK80_6, no_load_speed_AK80_6, 0, -.
 ↪5*no_load_speed_AK80_6]
motor_torque_AK80_6 = [-tau_stall_AK80_6, 0, tau_stall_AK80_6, 1.
 ↪5*tau_stall_AK80_6]

######################################### Maxon␣
 ↪#########################################
# Maxon, 614949:
tau_stall_614949 = 4.3
no_load_speed_614949 = 4300*2*pi/60
motor_speed_614949 = [2*no_load_speed_614949, no_load_speed_614949, 0, -.
 ↪5*no_load_speed_614949]
motor_torque_614949 = [-tau_stall_614949, 0, tau_stall_614949, 1.
 ↪5*tau_stall_614949]

######################################### E-S Motor␣
 ↪#########################################
# E-S Motor, 28PG-385SP-19-EN:
tau_stall_28PG = 3.73
no_load_speed_28PG = 310*2*pi/60
motor_speed_28PG = [2*no_load_speed_28PG, no_load_speed_28PG, 0, -.
 ↪5*no_load_speed_28PG]
motor_torque_28PG = [-tau_stall_28PG, 0, tau_stall_28PG, 1.5*tau_stall_28PG]

# E-S Motor, 36GP-555PM-51-EN 24V:
tau_stall_36GP_51 = 4.90
no_load_speed_36GP_51 = 230*2*pi/60
motor_speed_36GP_51 = [2*no_load_speed_36GP_51, no_load_speed_36GP_51, 0, -.
 ↪5*no_load_speed_36GP_51]
motor_torque_36GP_51 = [-tau_stall_36GP_51, 0, tau_stall_36GP_51, 1.
 ↪5*tau_stall_36GP_51]

# E-S Motor, 36GP-555PM-100-EN 24V:
tau_stall_36GP_100 = 4.90
no_load_speed_36GP_100 = 120*2*pi/60
motor_speed_36GP_100 = [2*no_load_speed_36GP_100, no_load_speed_36GP_100, 0, -.
 ↪5*no_load_speed_36GP_100]
motor_torque_36GP_100 = [-tau_stall_36GP_100, 0, tau_stall_36GP_100, 1.
 ↪5*tau_stall_36GP_100]

# E-S Motor, 36GP-555PM-139-EN 24V:
tau_stall_36GP_139 = 4.90
no_load_speed_36GP_139 = 85*2*pi/60
```

```
motor_speed_36GP_139 = [2*no_load_speed_36GP_139, no_load_speed_36GP_139, 0, -.
↪5*no_load_speed_36GP_139]
motor_torque_36GP_139 = [-tau_stall_36GP_139, 0, tau_stall_36GP_139, 1.
↪5*tau_stall_36GP_139]

######################################## Pololu␣
↪########################################
# Pololu, 150:1 Metal Gearmotor 37Dx73L mm 12V with 64 CPR Encoder (Helical␣
↪Pinion):
tau_stall_37D__12V_150 = 4.805
no_load_speed_37D__12V_150 = 67*2*pi/60
motor_speed_37D__12V_150 = [2*no_load_speed_37D__12V_150,␣
↪no_load_speed_37D__12V_150, 0, -.5*no_load_speed_37D__12V_150]
motor_torque_37D__12V_150 = [-tau_stall_37D__12V_150, 0,␣
↪tau_stall_37D__12V_150, 1.5*tau_stall_37D__12V_150]

# Pololu, 131:1 Metal Gearmotor 37Dx73L mm 12V with 64 CPR Encoder (Helical␣
↪Pinion):
tau_stall_37D__12V_131 = 4.41
no_load_speed_37D__12V_131 = 76*2*pi/60
motor_speed_37D__12V_131 = [2*no_load_speed_37D__12V_131,␣
↪no_load_speed_37D__12V_131, 0, -.5*no_load_speed_37D__12V_131]
motor_torque_37D__12V_131 = [-tau_stall_37D__12V_131, 0,␣
↪tau_stall_37D__12V_131, 1.5*tau_stall_37D__12V_131]

# Pololu, 150:1 Metal Gearmotor 37Dx73L mm 24V with 64 CPR Encoder (Helical␣
↪Pinion):
tau_stall_37D__24V_150 = 5.49
no_load_speed_37D__24V_150 = 68*2*pi/60
motor_speed_37D__24V_150 = [2*no_load_speed_37D__24V_150,␣
↪no_load_speed_37D__24V_150, 0, -.5*no_load_speed_37D__24V_150]
motor_torque_37D__24V_150 = [-tau_stall_37D__24V_150, 0,␣
↪tau_stall_37D__24V_150, 1.5*tau_stall_37D__24V_150]

# Pololu, 131:1 Metal Gearmotor 37Dx73L mm 24V with 64 CPR Encoder (Helical␣
↪Pinion):
tau_stall_37D__24V_131 = 4.61
no_load_speed_37D__24V_131 = 79*2*pi/60
motor_speed_37D__24V_131 = [2*no_load_speed_37D__24V_131,␣
↪no_load_speed_37D__24V_131, 0, -.5*no_load_speed_37D__24V_131]
motor_torque_37D__24V_131 = [-tau_stall_37D__24V_131, 0,␣
↪tau_stall_37D__24V_131, 1.5*tau_stall_37D__24V_131]

# ODrive, DUAL SHAFT MOTOR - D5065 270KV:
tau_stall_D5065 = 1.99
no_load_speed_D5065 = 8640*2*pi/60
```

```
motor_speed_D5065 = [2*no_load_speed_D5065, no_load_speed_D5065, 0, -.
↪5*no_load_speed_D5065]
motor_torque_D5065 = [-tau_stall_D5065, 0, tau_stall_D5065, 1.5*tau_stall_D5065]

# ODrive, DUAL SHAFT MOTOR - D6374 150KV:
tau_stall_D6374 = 3.86
no_load_speed_D6374 = 5760*2*pi/60
motor_speed_D6374 = [2*no_load_speed_D6374, no_load_speed_D6374, 0, -.
↪5*no_load_speed_D6374]
motor_torque_D6374 = [-tau_stall_D6374, 0, tau_stall_D6374, 1.5*tau_stall_D6374]

# Turnigy, 9235-100KV Brushless Multi-Rotor Motor:
tau_stall_9235 = 4.71
no_load_speed_9235 = 3840*2*pi/60
motor_speed_9235 = [2*no_load_speed_9235, no_load_speed_9235, 0, -.
↪5*no_load_speed_9235]
motor_torque_9235 = [-tau_stall_9235, 0, tau_stall_9235, 1.5*tau_stall_9235]

# Turnigy, SK8 6374-192KV Sensored Brushless Motor (14P):
tau_stall_6374_192KV = 4.31
no_load_speed_6374_192KV = 7373*2*pi/60
motor_speed_6374_192KV = [2*no_load_speed_6374_192KV, no_load_speed_6374_192KV,␣
↪0, -.5*no_load_speed_6374_192KV]
motor_torque_6374_192KV = [-tau_stall_6374_192KV, 0, tau_stall_6374_192KV, 1.
↪5*tau_stall_6374_192KV]

# Turnigy, SK8 6374-149KV Sensored Brushless Motor (14P):
tau_stall_6374_149KV = 4.31
no_load_speed_6374_149KV = 7373*2*pi/60
motor_speed_6374_149KV = [2*no_load_speed_6374_149KV, no_load_speed_6374_149KV,␣
↪0, -.5*no_load_speed_6374_149KV]
motor_torque_6374_149KV = [-tau_stall_6374_149KV, 0, tau_stall_6374_149KV, 1.
↪5*tau_stall_6374_149KV]

# Turnigy, 9225-90KV Turnigy Multistar Brushless Multi-Rotor Motor:
tau_stall_9225 = 3.31
no_load_speed_9225 = 3456*2*pi/60
motor_speed_9225 = [2*no_load_speed_9225, no_load_speed_9225, 0, -.
↪5*no_load_speed_9225]
motor_torque_9225 = [-tau_stall_9225, 0, tau_stall_9225, 1.5*tau_stall_9225]

# Plotting the torque-speed curves of the arm's motion and the motors
plt.figure(figsize=(12,8))
plt.plot(tot_tau2_list, tot_dtheta2_list, color='black', label='All Trials and␣
↪Participants')
```

```
plt.plot(motor_torque_GL80_60, motor_speed_GL80_60, '--', label='T-Motor, GL80␣
↪KV60')
plt.plot(motor_torque_GL80_30, motor_speed_GL80_30, '--', label='T-Motor, GL80␣
↪KV30')
plt.plot(motor_torque_G80_60, motor_speed_G80_60, '--', label='T-Motor, G80␣
↪KV60')
plt.plot(motor_torque_G80_30, motor_speed_G80_30, '--', label='T-Motor, G80␣
↪KV30')
plt.plot(motor_torque_GL60_25, motor_speed_GL60_25, '--', label='T-Motor, GL60␣
↪KV25')
plt.plot(motor_torque_GL60_55, motor_speed_GL60_55, '--', label='T-Motor, GL60␣
↪KV55')
plt.plot(motor_torque_GL100, motor_speed_GL100, label='T-Motor, GL100')
plt.plot(motor_torque_G100, motor_speed_G100, label='T-Motor, G100')
plt.plot(motor_torque_R60, motor_speed_R60, label='T-Motor, R60')
plt.plot(motor_torque_R80, motor_speed_R80, label='T-Motor, R80')
plt.plot(motor_torque_AK60_6, motor_speed_AK60_6, label='T-Motor, AK60-6')
plt.plot(motor_torque_AK80_6, motor_speed_AK80_6, label='T-Motor, AK80-6')
plt.plot(motor_torque_28PG, motor_speed_28PG, '--', label='Maxon, 614949')
plt.plot(motor_torque_28PG, motor_speed_28PG, '--', label='E-S Motor,␣
↪28PG-385SP-19-EN')
plt.plot(motor_torque_36GP_51, motor_speed_36GP_51, label='E-S Motor,␣
↪36GP-555PM-51-EN 24V')
plt.plot(motor_torque_36GP_100, motor_speed_36GP_100, label='E-S Motor,␣
↪36GP-555PM-100-EN 24V')
plt.plot(motor_torque_36GP_139, motor_speed_36GP_139, label='E-S Motor,␣
↪36GP-555PM-139-EN 24V')
plt.plot(motor_torque_37D__12V_150, motor_speed_37D__12V_150, label='Pololu,␣
↪37D__12V_150')
plt.plot(motor_torque_37D__12V_131, motor_speed_37D__12V_150, label='Pololu,␣
↪37D__12V_131')
plt.plot(motor_torque_37D__24V_150, motor_speed_37D__24V_150, label='Pololu,␣
↪37D__24V_150')
plt.plot(motor_torque_37D__24V_131, motor_speed_37D__24V_150, label='Pololu,␣
↪37D__24V_131')
plt.plot(motor_torque_D5065, motor_speed_D5065, '--', label='ODrive,␣
↪D5065-270KV')
plt.plot(motor_torque_D6374, motor_speed_D6374, label='ODrive, D6374-150KV')
plt.plot(motor_torque_9235, motor_speed_9235, label='Turnigy, 9235-100KV')
plt.plot(motor_torque_6374_192KV, motor_speed_6374_192KV, label='Turnigy,␣
↪6374-192KV')
plt.plot(motor_torque_6374_149KV, motor_speed_6374_149KV, label='Turnigy,␣
↪6374-149KV')
plt.plot(motor_torque_9225, motor_speed_9225, linestyle='--', label='Turnigy,␣
↪9225-90KV')
```

```
plt.ylabel('Velocity [rad/sec]')
plt.xlabel('Torque [Nm]')
plt.xlim([-4, 20])
plt.ylim([-7, 50])
plt.grid()
plt.legend(loc='upper left', bbox_to_anchor=(1.02,1))
plt.suptitle('Elbow Speed-Torque', fontsize=20)
plt.title('Elbow Speed-Torque for all Trials and all Motors')
plt.show()
```



[ ]: