Optimization of a Two Link Manipulator in an Industrial Setting

Cameron Moore
ME 555
University of Michigan

April 22, 2014

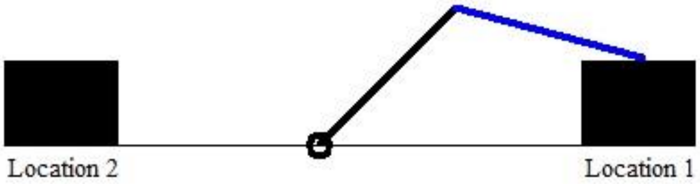
Abstract

The purpose of this study is to minimize the energy used by a robotic arm on an assembly line. The robotic arm is represented as a two link manipulator. The design variables used are the lengths and diameters of the links as well as the control of the X trajectory. Initially the X trajectory was set to be linear and the link lengths and diameters were optimized which gave us results of L_1 = 2.6711 meters, L_2 = 2.5491 meters, D_1 = .04534 meters, and D_2 = .03487 meters which led to the total energy of the system being 58.89 kJ. The X trajectory was then allowed to be controlled by making the distance the object travels in the X direction in a given time step a design variable. The results of that optimization were L_1 = 2.6101 meters, L_2 = 2.6101 meters, D_1 = .0312 meters, and D_2 = .0200 meters. The results of the controlled trajectory will be shown later but it dropped the total energy of the system to around 26.89 kJ which is much less than without optimizing the control. For both situations, the Y trajectory was set as a 4th order system which simulates obstacle avoidance.

1. Introduction

The two link manipulator being studied in this optimization problem is one used in a hypothetical industrial setting. The manipulator consists of two links and an end effector that grabs the desired object which is assumed to be a car hood with a mass of 13.5 Kg. In this hypothetical situation the end effector must follow a specific path to avoid hypothetical obstacles in the factory it will be implemented. The manipulator is assumed to be a 2 link planar robot, that is, it only moves in two dimensions. The overall objective of this study is to optimize the design and control of this manipulator to minimize the energy of the system. The design variables for this problem are the lengths of the links, the diameters of the links, and the control of manipulator. This specific problem will be constrained by a number of constrains. The links should not bend more than a specified amount given the loads each link will be subjected to and the object must not move too quickly for safety reasons. There are also a number of more obvious constraints such as the lengths and diameters having to be greater than zero. The larger the diameter of the less it will bend under the given load but this will also cause the link to be heavier and therefore will require more energy to move. If the links are made shorter it will also bend less but may not reach all the locations it needs to reach. If the object was moved too quickly it would raise the energy of the system but if it was moved too slowly there would be a lot of unneeded potential energy. As shown, there are a number of tradeoffs that will be analyzed. The links are assumed to have a cross section and assumed to be circular. The material for the links are assumed to be steel and the car hood is assumed to be a thin disk to simplify inertia calculations. Projects have been done to similar problems involving manipulators such as Brown et al. [1], Chen et al. [2], and Furry et al. [3].

Figure 1



2. Design Problem

2.1. Mathematical Model

2.1.1. Notations, Design Variables, and Parameters

There are a number of lengths and diameters that could satisfy the constraints given. For this problem many assumptions were made that will be discussed later but it was solved for a specific solution. The solution that was found for the system being optimized at this point in the project was L_1 = 2.6101 meters, L_2 = 2.6101 meters, D_1 = 0.0312 meters, and D_2 = 0.0200 meters. This gave us a total energy consumption of 26.89 kJ for moving the hood from one place to another. See appendix for the variable definition table.

2.1.2. Objective Function

The objective of this project is to minimize the energy required to move a car hood from one location to the other using a two link manipulator. In order to minimize the energy the link parameters and control need optimized. The equation to be optimized is the equation for the total energy of the system. This equation requires angular velocity, linear velocity of the centers of mass of each link and the heights of the center of mass of each link. The following shows how the equations were obtained.

$$P_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; P_1 = \begin{bmatrix} r_1\cos(\theta_1) \\ r_1\sin(\theta_1) \\ 0 \end{bmatrix}; P_A = \begin{bmatrix} L_1\cos(\theta_1) \\ L_1\sin(\theta_1) \\ 0 \end{bmatrix}; P_2 = P_A + \begin{bmatrix} L_2\cos(\theta_1+\theta_2) \\ L_2\sin(\theta_1+\theta_2) \\ 0 \end{bmatrix}$$

The above equations only require L_1, L_2, and the angles of each link and they are used to establish the location of the end effector. The below derivations are used for the objective function.

$$r_1 = \frac{L_1}{2}$$

$$r_2 = \frac{L_2}{2}\left(\frac{m_2 + 2m_o}{m_2 + m_o}\right)$$

$$V_1 = \sqrt{\left(\dot{\theta}_1 r_1 \sin(\theta_1)\right)^2 + \left(\dot{\theta}_1 r_1 \cos(\theta_1)\right)^2}$$

$$V_2 = \sqrt{\left(-\dot{\theta}_1(r_2\sin(\theta_1+\theta_2)+L_1\sin(\theta_1))-\dot{\theta}_2 r_2\sin(\theta_1+\theta_2)\right)^2+\left(\dot{\theta}_1(r_2\cos(\theta_1+\theta_2)+L_1\cos(\theta_1))+\dot{\theta}_2 r_2\cos(\theta_1+\theta_2)\right)^2}$$

$$KE = \frac{1}{2}m_1V_1^2+\frac{1}{2}m_2V_2^2+\frac{1}{2}I_1\dot{\theta}_1+\frac{1}{2}I_2\dot{\theta}_2$$

$$PE = m_1gy_1+m_2gy_2$$

$$E = KE + PE$$

$$\boldsymbol{E = \frac{1}{2}m_1V_1^2+\frac{1}{2}m_2V_2^2+\frac{1}{2}I_1\dot{\theta}_1+\frac{1}{2}I_2\dot{\theta}_2+m_1gy_1+m_2gy_2}$$

2.1.3. Constraints

The links must have a length and cross section so therefore they have to be greater than zero. The links must add up to be long enough to reach the objects starting and ending locations which are the farthest away the object will be throughout its cycle. The links must also be appropriate sizes so that the arms do not interfere with the base. It also must be taken into consideration that if the first link were too long the second one would have to be long enough so that it could reach the trajectory. It also is not ideal to have the links deflect too much when the object is being transported so the diameters and lengths have to be picked accordingly. As for the control optimization, the object should not be going extremely fast throughout the air because it could become dangerous. In addition to that, the object should also not just be standing still at any point because that is not efficient so the control aspect is bounded above and below. The goal of this optimization is to minimize energy with a given path so the last constraints guarantee the location of the object. We am assuming the center of mass for each link is exactly in the middle if there is no object in the end effector and we also assuming uniform circular cross sections of each link. The material being used for the links is assumed to be steel and the car hood is assumed to be a thin disk.

$$\boldsymbol{L_1 > 0; L_2 > 0}$$

$$\boldsymbol{D_1 > 0; D_2 > 0}$$

$$\boldsymbol{L_1 + L_2 \geq \sqrt{5^2+1.5^2}}$$

$$\boldsymbol{L_1 - L_2 \leq 1.5}$$

$$\boldsymbol{L_2 - L_1 \leq\ 0}$$

$$\boldsymbol{.1 \leq d_i \leq 3}$$

The above constraints are trivial. The following constraints are a little more complicated. They are used to be sure the links do not deflect too much under the given load of the object and the weight of the links. The constraint equations will be derived below.

$$y_{max} = maximum\ deflection = \frac{PL^3}{3Y_m I_c}$$

Where P is the load being applied, L is the length of the link, Y_m is the modulus of elasticity of the material being used and I_c is the moment of inertia about the centroid axis. At this point in the project we assume uniform circular cross section of each link.

$$I_c = Inertia\ about\ centroid\ of\ circular\ cross\ section = \frac{\pi D^4}{64}$$

$$\boldsymbol{\frac{64(m_2 + m_o)gL_1^3}{3\pi Y_m D_1^4} < 0.05}$$

$$\boldsymbol{\frac{64m_o g L_2^3}{3\pi Y_m D_2^4} < 0.05}$$

The following constraint equations are not used for the minimization part of the problem but instead are used to establish a trajectory and that is used to obtain the angular displacements and angular velocities required which are used to calculate the energy. For the first half of the project the X trajectory was set to be linear but with the optimization of the control it became the more complicated equation seen below.

$$Location\ of\ Link\ 2\ endpoint\ on\ x - axis = L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2)$$

$$Location\ of\ Link\ 2\ endpoint\ on\ y - axis = L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2)$$

$$L_1cos(\theta_1(t)) + L_2(\theta_1(t) + \theta_2(t)) = P_2(1,:) = X = 5 - \sum_{1}^{i} d_i - \frac{d_i}{.5}t + id_i$$

$$L_1sin(\theta_1(t)) + L_2sin(\theta_1(t) + \theta_2(t)) = P_2(2,:) = Y =.007X^4+.0354X^3-.2753X^2-.8836X + 3.9767$$

2.1.4. Summary Model

To summarize the model, we have a two link manipulator in an industrial setting and would like it to be as efficient as possible. To do this we must minimize the objective function below.

$$E = \frac{1}{2}m_1V_1^2 + \frac{1}{2}m_2V_2^2 + \frac{1}{2}I_1\dot{\theta}_1 + \frac{1}{2}I_2\dot{\theta}_2 + m_1gy_1 + m_2gy_2$$

This function will be minimized with respect to the constraints listed in the section above. This will be done by changing various design variables such as the length and diameter of the links as well as the control of the manipulator. At this point, there are fourteen design variables but that could be changed. Four of those variables are the link parameters while the other ten variables are for the control. Ten points were used because it is an appropriate amount to prove a concept while not overworking the computer during the optimization. It is subjected to twenty-nine inequality constraints and one equality constraint. Similar to the design variables, there could be more if the number of control design variables were changed. With these fourteen design variables and the one equality constraint it ends up being an optimization problem with thirteen degrees of freedom. This number will change if the control variables were to change. It will also be constrained to a specific path which could simulate needing to avoid obstacles that may be found in an industrial setting. We are assuming uniform circular cross sections of each link and assuming that the center of gravity for each link is in the exact center when no object is held in the end effector. We are also assuming the car hood to be a thin disk when calculating the center of gravity for the combined mass of the second link and the car hood. This was done using the parallel axis theorem. This problem started out as a simple optimization problem of the link parameters but evolved into an optimization problem of the control as well. It also initially had a constant Y trajectory but in order to make the problem more realistic and more complicated it was changed to a nontrivial path to follow.

2.2. Model Analysis

As stated above, the function being minimized is the function for the total energy consumed by the system. Listed once again below are the constraints as well as the objective function.

$$\boldsymbol{min\,E = \frac{1}{2}m_1V_1^2 + \frac{1}{2}m_2V_2^2 + \frac{1}{2}I_1\dot{\theta}_1 + \frac{1}{2}I_2\dot{\theta}_2 + m_1gy_1 + m_2gy_2}$$

$$\boldsymbol{g_1 = -L_1 < 0 \qquad g_2 = -L_2 < 0}$$

$$\boldsymbol{g_3 = -D_1 < 0 \qquad g_4 = -D_2 < 0}$$

$$\boldsymbol{g_5 = -L_1 - L_2 + \sqrt{2(1.5)^2} \le 0}$$

$$\boldsymbol{g_6 = L_1 - L_2 - 1.5 \le 0}$$

$$\boldsymbol{g_7 = L_2 - L_1 \le\ 0}$$

$$\boldsymbol{g_8 = -d_1 \le\ -.1 \qquad g_9 = -d_2 \le\ -.1}$$

$$\boldsymbol{g_{10} = -d_3 \le\ -.1 \qquad g_{11} = -d_4 \le\ -.1}$$

$$\boldsymbol{g_{12} = -d_5 \le\ -.1 \qquad g_{13} = -d_6 \le\ -.1}$$

$$\boldsymbol{g_{14} = -d_7 \le\ -.1 \qquad g_{15} = -d_8 \le\ -.1}$$

$$\boldsymbol{g_{16} = -d_9 \le\ -.1 \qquad g_{17} = -d_{10} \le\ -.1}$$

$$\boldsymbol{g_{18} = d_1 \le 3 \qquad g_{19} = d_2 \le 3}$$

$$\boldsymbol{g_{20} = d_3 \le 3 \qquad g_{21} = d_4 \le 3}$$

$$\boldsymbol{g_{22} = d_5 \le 3 \qquad g_{23} = d_6 \le 3}$$

$$\boldsymbol{g_{24} = d_7 \le 3 \qquad g_{25} = d_8 \le 3}$$

$$\boldsymbol{g_{26} = d_9 \le 3 \qquad g_{27} = d_{10} \le 3}$$

$$\boldsymbol{g_{28} = \frac{64(m_2 + m_o)gL_1^3}{3\pi Y_m D_1^4} - 0.05 \leq 0}$$

$$\boldsymbol{g_{29} = \frac{64m_o gL_2^3}{3\pi Y_m D_2^4} - 0.05 \leq 0}$$

$$\boldsymbol{h_1 = d_1 + d_2 + d_3 + d_4 + d_5 + d_6 + d_7 + d_8 + d_9 + d_{10} = 10}$$

The above equations need certain values to be calculated which can be seen done below. As stated before, the object being transported is a car hood and is assumed to be a thin disk with a mass of 13.5 Kg and a radius of .375 meters.

$$m_1 = Volume * Density = \frac{\pi D_1^2 L_1 \rho}{4}$$

$$m_2 = Volue * Density = \frac{\pi D_2^2 L_2 \rho}{4}$$

$$I_1 = \frac{mL}{12} = \frac{\pi D_1^2 L_1^3 \rho}{48}$$

$$I_2 = I_{rod} + I_{mass} + m_o\left(\frac{L_2}{2}\right)^2 = \frac{m_2 L_2^2}{12} + \frac{m_o r_o^2}{4} + m_o\left(\frac{L_2}{2}\right)^2 = \frac{32\pi\rho D_2^2 L_2^3 + 729 + 5184 L_2^2}{1536}$$

$$r_1 = \frac{L_1}{2}$$

$$r_2 = \frac{m_o L_2 + \frac{m_2 L_2}{2}}{m_o + m_2} = \frac{L_2(2m_o + m_2)}{2(m_o + m_2)}$$

$$y_1 = r_1 \sin(\theta_1)$$

$$y_2 = L_1 \sin(\theta_1) + r_2 \sin(\theta_1 + \theta_2)$$

According to matlab, the active constraints are g_5, g_7, g_{10}, g_{24}, g_{28}, and g_{29}. It is difficult to see whether increasing the d values would increase the objective function or not because of the complicated trajectory so fmincon's ability to alert the user of which inequality constraints were active was used. The system is well bounded because every variable is bounded above and below by either the objective function or by the constraints. See appendix for monotonicity table.

3. Design Optimization

3.1. Optimization Results

Using the attached code the fmincon function was used to minimize the total energy consumed by moving one object to another. As stated earlier, the optimal link parameters found from the code mentioned above came out to **L_1 = 2.6101 meters, L_2 = 2.6101 meters, D_1 = 0.0312 meters, and D_2 = 0.0200 meters**. In addition to optimizing the link parameters we will also be optimizing the control of the manipulator. The optimal d values came out to be **d_1 = 0.6939 meters, d_2 = 0.3174 meters, d_3 = 0.100 meters, d_4 = 0.1429 meters, d_5 = 0.8972 meters, d_6 = 1.6549 meters, d_7 = 3.000 meters, d_8 = 2.3336 meters, d_9 = 0.7585 meters, and d_{10} = 0.1017 meters.** This gave us a total energy consumption of 26.89 kJ. Using the optimal answers found it can be seen that all the constraints are met. The six active constraints can be hand calculated and shown that they are indeed active and the equality constraint can also be confirmed. This point does satisfy KKT conditions because it satisfies all the constraints and the active inequality constraints and equality constraint are linearly independent. Given the constraints, the solution is a global minimum but the slightest change in any of the constraints could cause a drastic change in the optimized energy value which will be seen in the next section. These values led to the mass of the first link to be approximately 16 Kg and the mass of the second link to be approximately 6.6 Kg. The inertia for the first link was found to be 9.09 Kg-m^2 and the inertia for the second link and mass lumped together is calculated to be 30.15 Kg-m^2. The rest of the important values depend on the angular displacements of each joint. These values were found using the fsolve function in matlab and making sure the end point falls on the trajectory established earlier at the given time. Run the attached code to obtain the solutions discussed above. Plots and tables of the optimized solution can be seen in the appendix.

3.2. Parametric Study

Previously in the problem only the link parameters were optimized while leaving the control constant and the optimal link parameters were L_1 = 2.6711 meters, L_2 = 2.5491 meters, D_1 = 0.04534 meters, and D_2 = 0.03487 meters. With those parameters and a linear X trajectory a total energy of 58.89 kJ is obtained which is much larger than the total energy of the optimized control solution. When the above link parameters are paired with the optimized control values a total energy of 46.67 kJ is obtained which is less than the system without control. In addition to the above two situations the control values may be optimized while leaving the link parameters constant to the values above. This outputs the values d_1 = 0.6774 meters, d_2 = 0.6274 meters, d_3 = 1.2514 meters, d_4 = 2.024 meters, d_5 = 3.000 meters, d_6 = 1.9176 meters, d_7 = 0.2023 meters, d_8 = .100 meters, d_9 = .100 meters, and d_{10} = .100 meters. This outputs a total energy of 46.48 kJ which is less than the total energy when using the control values from the solution where the link parameters were optimized in parallel to the control but using the link parameters from the optimization when the control was not optimized. The energy was higher than the situation where all fourteen variables were optimized in parallel so this shows that the small changes in variables outputs a larger energy and shows the optimized solution is indeed the minimum. Plots and tables of the results can be seen in the appendix.

4. Discussion

As seen from the optimal solution, all the constraints are met. The problem model implies that the larger the links and diameters, the higher the energy will be. When optimizing the control the links ended up being the same length but when only optimizing the link parameters the links ended up being different lengths. This is because the driving force behind minimizing the energy when the control remains constant is to vary the lengths and diameters of the links. When there are other variables available to change the link lengths are not necessarily the main variables to change to minimize energy. The above and below bounded constraints for the control variables limit the potential to the problem. This is because we do not want the object moving at high speeds because that could become dangerous in an industrial setting. These values only were active in two situations but without those constraints the control values would have all changed. A simple design rule for this situation is to make the links as light as possible. This could be done by making sure the diameters are as small as possible or the lengths are as small as possible. In this specific case, the limits for those variables were bounded by being able to reach specific points and making sure the links do not bend. The control variables do not necessarily have a design law because it strongly depends on the expected trajectory. From here, this problem can be taken to three dimensions where it would be far more realistic. The geometries of the links could also vary because in the real world a perfectly round cylindrical link would not be found. It would also be helpful to have many more control variables. Ten was the number chosen to make the code more manageable for the computer being used.

5. Acknowledgments

The idea for this project came from experience gained in a previous class at The Ohio State University which was titled "Mechanics and Controls or Robots". This class was taught by Dr. Manoj Srinivasan. The equations of motion and equations for conservation of energy were derived during the class.

6. References

[1]. Brown, R; Nguyen, H; Skinner, N, "Optimal Design of a Throwing Robot", 2009.

[2]. Chen, H; Ng, G, "Optimal Balancing of Robotic Manipulator Using Counter Weights", 1993.

[3]. Furry, S; Piland, K; "Optimal Design of a Robotic Manipulator for Maximum Manipulability", 1988.

[4]. Collins, J; Busby, H; Staab, G, *Mechanical Design of Machine Elements and Machines*, John Wiley and Sons, 2010.

7. Appendix

7.1. Variable Definitions

Table 1

Variable	Description	Type	Unit	Sample Value
$\boldsymbol{\theta_1}$	Angle between ground and link 1	Dynamic Variable	Radian	(Depends on t)
$\boldsymbol{\theta_2}$	Angle between link 1 and link 2	Dynamic Variable	Radian	(Depends on t)
$\boldsymbol{\dot{\theta}_1}$	Angular velocity between ground and link 1	Dynamic Variable	Rad/sec	(Depends on t)
$\boldsymbol{\dot{\theta}_2}$	Angular Velocity between link 1 and link 2	Dynamic Variable	Rad/sec	(Depends on t)
$\boldsymbol{L_1}$	Length of link 1	Design Variable	Meter	2.6101 Meter
$\boldsymbol{L_2}$	Length of link 2	Design Variable	Meter	2.6101 Meter
$\boldsymbol{r_1}$	Center of gravity of link 1	Parameter	Meter	1.3050 Meter
$\boldsymbol{r_2}$	Center of gravity of link 2 coupled with object	Parameter	Meter	1.7040 Meter
$\boldsymbol{m_1}$	Mass of link 1	Dependent on D.V.	Kg	35 Kg
$\boldsymbol{m_2}$	Mass of link 2	Dependent on D.V.	Kg	20 Kg
$\boldsymbol{m_o}$	Mass of object	Parameter	Kg	13.5 Kg
$\boldsymbol{I_1}$	Inertia of link 1	Dependent on D.V.	m^2*Kg	20 m^2*Kg
$\boldsymbol{I_2}$	Inertia of link 2 and mass	Dependent on D.V.	m^2*Kg	36 m^2*Kg
$\boldsymbol{D_1}$	Cross sectional diameter of link 1	Design Variable	Meter	.045 Meters
$\boldsymbol{D_2}$	Cross sectional diameter of link 2	Design Variable	Meter	.035 Meters
$\boldsymbol{g}$	Gravity	Parameter	m/s^2	9.81 m/s^2
$\boldsymbol{\rho}$	Density of steel	Parameter	Kg/m^3	8050 Kg/m^3
$\boldsymbol{d_i}$	Distance Traveled in i^{th} time step	Design Variable	Meter	2.8763 Meters
$\boldsymbol{V_1}$	Linear velocity of center of mass of link 1	Dynamic Variable	m/s	(Depends on t)
$\boldsymbol{V_2}$	Linear velocity of center of mass of link 2 + object	Dynamic Variable	m/s	(Depends on t)
$\boldsymbol{y_1}$	Height of center of mass of link 1	Dynamic Variable	Meter	(Depends on t)
$\boldsymbol{y_2}$	Height of center of mass of link 2 + object	Dynamic Variable	Meter	(Depends on t)
$\boldsymbol{Y_m}$	Modulus of elasticity of steel	Parameter	Pascal	200 x 10^9 Pa
$\boldsymbol{KE}$	Kinetic energy of system	Dynamic Variable	Joule	(Depends on t)
$\boldsymbol{PE}$	Potential energy of system	Dynamic Variable	Joule	(Depends on t)
$\boldsymbol{E}$	Total energy of system	Objective	Joule	23,300 Joules
$\boldsymbol{X(t)}$	Desired path of object on x-axis	Parameter	Equation	$X = 5 - \sum_{1}^{i} d_i - \frac{d_i}{.5} t + i d_i$
$\boldsymbol{Y(t)}$	Desired path of object on y-axis	Parameter	Equation	$4th\ Order\ Equation$

7.2. Monotonicity Table

Table 2: Monotonicity Table

	L_1	L_2	D_1	D_2	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}
E	+	+	+	+	?	?	?	?	?	?	?	?	?	?
g_1	-													
g_2		-												
g_3			-											
g_4				-										
g_5	-	-												
g_6	+	-												
g_7	-	+												
g_8					-									
g_9						-								
g_{10}							-							
g_{11}								-						
g_{12}									-					
g_{13}										-				
g_{14}											-			
g_{15}												-		
g_{16}													-	
g_{17}														-
g_{18}					+									
g_{19}						+								
g_{20}							+							
g_{21}								+						
g_{22}									+					
g_{21}										+				
g_{24}											+			
g_{25}												+		
g_{26}													+	
g_{27}														+
g_{28}	+		-											
g_{29}		+		-										

7.3. Codes

7.3.1. Initial Code

```
clc
clear
close all
options=optimset('Display','off');
rho=8050;%kg/m^3
mo=13.5;%kg
g=9.81;%m/s^2
td=.1;%sec
X0=[2.8;2.8;.05;.05];
Aeq=[];
Beq=[];
A=[-1 -1 0 0;
    1 -1 0 0;
    -1 1 0 0];
B=[-sqrt(5^2+1.5^2);1.5;0];
UB=[inf;inf;inf;inf];
LB=[0;0;0;0];
[x,fval]=fmincon(@opt,X0,A,B,Aeq,Beq,LB,UB,@(x) MyCon(x))
Plots(x,td,options)

function [Em] = opt(x)
L1=x(1);%m
L2=x(2);%m
D1=x(3);%m
D2=x(4);%m
rho=8050;%kg/m^3
mo=13.5;%kg
g=9.81;%m/s^2
td=.1;%sec
m1=pi*D1^2*L1*rho/4;%kg
m2=pi*D2^2*L2*rho/4;%kg
I1=pi*D1^2*L1^3*rho/48;%kg*m^2
I2=(32*pi*rho*D2^2*L2^3+729+5184*L2^2)/1536;%kg*m^2
param.L1=L1;
param.L2=L2;
t=0:td:5;
options=optimset('Display','off');
for i=1:length(t)
    param.t=t(i);
    if i==1
        guess=[pi/3,pi];
    else
        guess=[thetas(i-1,1),thetas(i-1,2)];
    end
    thetas(i,:)=fsolve(@(theta) residual(theta,param),guess,options);
    if i==1
        theta1dot=0;
        theta2dot=0;
    else
        theta1dot=(thetas(i,1)-thetas(i-1,1))/td;
        theta2dot=(thetas(i,2)-thetas(i-1,2))/td;
    end
    Y1=L1*sin(thetas(i,1))/2;
    Y2=L1*sin(thetas(i,1))+L2*(m2+2*mo)*sin(thetas(i,1)+thetas(i,2))/...
        (2*(mo+m2));
    V1s=theta1dot^2*L1^2/4;
    V2s=(theta1dot*(L1*sin(thetas(i,1))+L2*sin(thetas(i,1)+thetas(i,2))*...
        (m2+2*mo)/(2*(mo+m2)))+(L2*theta2dot*sin(thetas(i,1)+...
        thetas(i,2))*(m2+2*mo)/(2*(mo+m2))))^2+(theta1dot*...
        (L1*cos(thetas(i,1))+L2*cos(thetas(i,1)+thetas(i,2))*(m2+2*mo)/...
        (2*(mo+m2)))+(L2*theta2dot*cos(thetas(i,1)+thetas(i,2))*...
        (m2+2*mo)/(2*(mo+m2))))^2;
    E(i)=.5*m1*V1s+.5*(m2+mo)*V2s+.5*I1*theta1dot+.5*I2*theta2dot+...
        m1*g*Y1+(m2+mo)*g*Y2;
end
Em=sum(E);
end


function [c,ceq] = MyCon(x)
L2=x(2);%m
D2=x(4);%m
rho=8050;%kg/m^3
YM=200E9;%Pa
mo=13.5;%kg
g=9.81;%m/s^2
m2=pi*D2^2*L2*rho/4;%kg
c(1)=64*(m2+mo)*g*x(1)^3/(3*pi*YM*x(3)^4)-.05;
c(2)=64*mo*g*x(2)^3/(3*pi*YM*x(4)^4)-.05;
ceq=[];
end



function f = residual(theta,param)
L1=param.L1;
L2=param.L2;
t=param.t;
xx=5-2*t;
yy=.007*xx^4+.0354*xx^3-.2753*xx^2-.8836*xx+3.9767;
f=[L1*cos(theta(1))+L2*cos(theta(1)+theta(2))-xx;
    L1*sin(theta(1))+L2*sin(theta(1)+theta(2))-yy];
end


function [] = Plots(x,td,options)
L1=x(1);
L2=x(2);
param.L1=L1;
param.L2=L2;
t=0:td:5;
for i=1:length(t)
    param.t=t(i);
    if i==1
        guess=[pi/3,pi];
    else
        guess=[thetas(i-1,1),thetas(i-1,2)];
    end
    thetas(i,:)=fsolve(@(theta) residual(theta,param),guess,options);
    figure(1);
    plot(0,0,'ro')
    hold on
    axis([-6 6 -6 6])
    P1=[L1*cos(thetas(i,1));L1*sin(thetas(i,1))];
    P2=[L1*cos(thetas(i,1))+L2*cos(thetas(i,1)+thetas(i,2));
        L1*sin(thetas(i,1))+L2*sin(thetas(i,1)+thetas(i,2))];
    plot([0 P1(1)],[0 P1(2)],'linewidth',2)
    plot([P1(1) P2(1)],[P1(2) P2(2)],'linewidth',2)
    pause(.1)
    hold off
end
end
```

7.3.2. Final Code

```
clc
clear
close all
options=optimset('Display','off');
n=10;%Number of Time Steps
tt=5;%Total Time To Cycle
td=tt/50;%Sample Time
X0=[1;1;1;1;1;1;1;1;1;1;2.8;2.8;.05;.05];%Initial Guess
Aeq=[1 1 1 1 1 1 1 1 1 1 0 0 0 0];%Equality Constraint
Beq=[10];%Equality Answer
A=[0 0 0 0 0 0 0 0 0 0 -1 -1 0 0;
    0 0 0 0 0 0 0 0 0 0 1 -1 0 0;
    0 0 0 0 0 0 0 0 0 0 -1 1 0 0];%Inequality Constraint
B=[-sqrt(5^2+1.5^2);1.5;0];%Inequality Answers
UB=[3;3;3;3;3;3;3;3;3;3;inf;inf;inf;inf];%Upper Bound
LB=[.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;0;0;0;0];%Lower Bound
[x,fval]=fmincon(@opt,X0,A,B,Aeq,Beq,LB,UB,@(x) MyCon(x))
Plots(x,tt,td,n,options)


function [Em] = opt(x)
n=10;
d1=x(1);
d2=x(2);
d3=x(3);
d4=x(4);
d5=x(5);
d6=x(6);
d7=x(7);
d8=x(8);
d9=x(9);
d10=x(10);
L1=x(11);
L2=x(12);
D1=x(13);
D2=x(14);
```

```
tt=5;
rho=8050;%kg/m^3
mo=13.5;%kg
ro=.375;%m
g=9.81;%m/s^2
td=tt/50;%sec
m1=pi*D1^2*L1*rho/4;%kg
m2=pi*D2^2*L2*rho/4;%kg
I1=pi*D1^2*L1^3*rho/48;%kg*m^2
I2=(m2*L2^2)/12+(mo+ro^2)/4+(mo*L2^2)/4;%kg*m^2
param.d1=d1;
param.d2=d2;
param.d3=d3;
param.d4=d4;
param.d5=d5;
param.d6=d6;
param.d7=d7;
param.d8=d8;
param.d9=d9;
param.d10=d10;
param.L1=L1;
param.L2=L2;
param.D1=D1;
param.D2=D2;
param.tt=tt;
t=0:td:tt;
options=optimset('Display','off');
for i=1:length(t)
    tn=t(i);
    if i<length(t)/n
        j=1;
    elseif i<length(t)/n*2
        j=2;
    elseif i<length(t)/n*3
        j=3;
    elseif i<length(t)/n*4
        j=4;
    elseif i<length(t)/n*5
        j=5;
    elseif i<length(t)/n*6
        j=6;
    elseif i<length(t)/n*7
        j=7;
    elseif i<length(t)/n*8
        j=8;
    elseif i<length(t)/n*9
        j=9;
    else
        j=10;
    end
    param.j=j;
    param.tn=tn;
    if i==1
        guess=[pi/3,pi];
    else
        guess=[thetas(i-1,1),thetas(i-1,2)];
    end
    thetas(i,:)=fsolve(@(theta) residual(theta,param),guess,options);
    if i==1
        theta1dot=0;
        theta2dot=0;
    else
        theta1dot=(thetas(i,1)-thetas(i-1,1))/td;
        theta2dot=(thetas(i,2)-thetas(i-1,2))/td;
    end
    Y1=L1*sin(thetas(i,1))/2;
    Y2=L1*sin(thetas(i,1))+L2*(m2+2*mo)*sin(thetas(i,1)+...
        thetas(i,2))/(2*(mo+m2));
    V1s=theta1dot^2*L1^2/4;
    V2s=(theta1dot*(L1*sin(thetas(i,1))+L2*sin(thetas(i,1)...
        +thetas(i,2))*(m2+2*mo)/(2*(mo+m2)))+(L2*theta2dot*...
        sin(thetas(i,1)+thetas(i,2))*(m2+2*mo)/(2*(mo+m2))))^2+...
        (theta1dot*(L1*cos(thetas(i,1))+L2*cos(thetas(i,1)+...
        thetas(i,2))*(m2+2*mo)/(2*(mo+m2)))+(L2*theta2dot*...
        cos(thetas(i,1)+thetas(i,2))*(m2+2*mo)/(2*(mo+m2))))^2;
    E(i)=.5*m1*V1s+.5*(m2+mo)*V2s+.5*I1*theta1dot+.5*I2*theta2dot+...
        m1*g*Y1+(m2+mo)*g*Y2;
end
Em=sum(E);
end


function [c,ceq] = MyCon(x)
L2=x(12);%m
D2=x(14);%m
L1=x(11);%m
D1=x(13);%m
rho=8050;%kg/m^3
YM=200E9;%Pa
mo=1.35;%kg
g=9.81;%m/s^2
m2=pi*D2^2*L2*rho/4;%kg
c(1)=64*(m2+mo)*g*L1^3/(3*pi*YM*D1^4)-.05;
c(2)=64*mo*g*L2^3/(3*pi*YM*D2^4)-.05;
ceq=[];
end


function f = residual(theta,param)
L1=param.L1;
L2=param.L2;
d1=param.d1;
d2=param.d2;
d3=param.d3;
d4=param.d4;
d5=param.d5;
d6=param.d6;
d7=param.d7;
d8=param.d8;
d9=param.d9;
d10=param.d10;
d=[d1 d2 d3 d4 d5 d6 d7 d8 d9 d10];
t=param.tn;
```

```
j=param.j;
xx=5-sum(d(1:j))-(d(j)/.5)*t+j*d(j);
yy=.007*xx^4+.0354*xx^3-.2753*xx^2-.8836*xx+3.9767;
f=[L1*cos(theta(1))+L2*cos(theta(1)+theta(2))-xx;
    L1*sin(theta(1))+L2*sin(theta(1)+theta(2))-yy];
end


function [] = Plots(x,tt,td,n,options)
d1=x(1);
d2=x(2);
d3=x(3);
d4=x(4);
d5=x(5);
d6=x(6);
d7=x(7);
d8=x(8);
d9=x(9);
d10=x(10);
L1=x(11);
L2=x(12);
param.d1=d1;
param.d2=d2;
param.d3=d3;
param.d4=d4;
param.d5=d5;
param.d6=d6;
param.d7=d7;
param.d8=d8;
param.d9=d9;
param.d10=d10;
param.L1=L1;
param.L2=L2;
param.tt=tt;
t=0:td:tt;
for i=1:length(t)
    tn=t(i);
    if i<length(t)/n
        j=1;
    elseif i<length(t)/n*2
        j=2;
    elseif i<length(t)/n*3
        j=3;
    elseif i<length(t)/n*4
        j=4;
    elseif i<length(t)/n*5
        j=5;
    elseif i<length(t)/n*6
        j=6;
    elseif i<length(t)/n*7
        j=7;
    elseif i<length(t)/n*8
        j=8;
    elseif i<length(t)/n*9
        j=9;
    else
```

```
        j=10;
    end
    param.j=j;
    param.tn=tn;
    if i==1
        guess=[pi/3,pi];
    else
        guess=[thetas(i-1,1),thetas(i-1,2)];
    end
    thetas(i,:)=fsolve(@(theta) residual(theta,param),guess,options);
    if i==1
        theta1dot(i)=0;
        theta2dot(i)=0;
    else
        theta1dot(i)=(thetas(i,1)-thetas(i-1,1))/td;
        theta2dot(i)=(thetas(i,2)-thetas(i-1,2))/td;
    end
    figure(1);
    plot(0,0,'ro')
    hold on
    axis([-8 8 -8 8])
    P1=[L1*cos(thetas(i,1));L1*sin(thetas(i,1))];
    P2=[L1*cos(thetas(i,1))+L2*cos(thetas(i,1)+thetas(i,2));
        L1*sin(thetas(i,1))+L2*sin(thetas(i,1)+thetas(i,2))];
    plot([0 P1(1)],[0 P1(2)],'linewidth',2)
    plot([P1(1) P2(1)],[P1(2) P2(2)],'linewidth',2)
    pause(.05)
    hold off
end
figure;
plot(t,theta1dot,'b')
hold on
plot(t,theta2dot,'r')
xlabel('Time [sec]')
ylabel('Angular Velocity [Rad/sec]')
legend('dtheta_1','dtheta_2',2)
s1=d1;
s2=s1+d2;
s3=s2+d3;
s4=s3+d4;
s5=s4+d5;
s6=s5+d6;
s7=s6+d7;
s8=s7+d8;
s9=s8+d9;
s10=s9+d10;
figure;
plot([.5,.5],[0,s1],'linewidth',2)
hold on
plot([1,1],[0,s2],'linewidth',2)
plot([1.5,1.5],[0,s3],'linewidth',2)
plot([2,2],[0,s4],'linewidth',2)
plot([2.5,2.5],[0,s5],'linewidth',2)
plot([3,3],[0,s6],'linewidth',2)
plot([3.5,3.5],[0,s7],'linewidth',2)
plot([4,4],[0,s8],'linewidth',2)
plot([4.5,4.5],[0,s9],'linewidth',2)
plot([5,5],[0,s10],'linewidth',2)
plot(.5,s1,'bo','linewidth',2,'MarkerSize',6)
plot(1,s2,'bo','linewidth',2,'MarkerSize',6)
plot(1.5,s3,'bo','linewidth',2,'MarkerSize',6)
plot(2,s4,'bo','linewidth',2,'MarkerSize',6)
plot(2.5,s5,'bo','linewidth',2,'MarkerSize',6)
plot(3,s6,'bo','linewidth',2,'MarkerSize',6)
plot(3.5,s7,'bo','linewidth',2,'MarkerSize',6)
plot(4,s8,'bo','linewidth',2,'MarkerSize',6)
plot(4.5,s9,'bo','linewidth',2,'MarkerSize',6)
plot(5,s10,'bo','linewidth',2,'MarkerSize',6)
axis([0 5 0 10])
xlabel('Time [sec]')
ylabel('X Distance Traveled [m]')
end
```

7.4. Results

7.4.1. Initial Results – Uniform X Trajectory

Table 3: Initial Link Results

L_1	L_2	D_1	D_2	**Energy**
2.6711 m	2.5491 m	0.04534 m	0.03487 m	58.89 kJ

7.4.2. Final Results – Optimized X Trajectory

Table 4: Final Control and Link Results

d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}
.6939 m	.3174 m	.100 m	.1429 m	.8972 m	1.6549 m	3.000 m	2.3336 m	.7585 m	.1017 m

L_1	L_2	D_1	D_2	**Energy**
2.6101 m	2.6101 m	.0312 m	.0200 m	26.89 kJ

Figure 2: Final Control Optimization

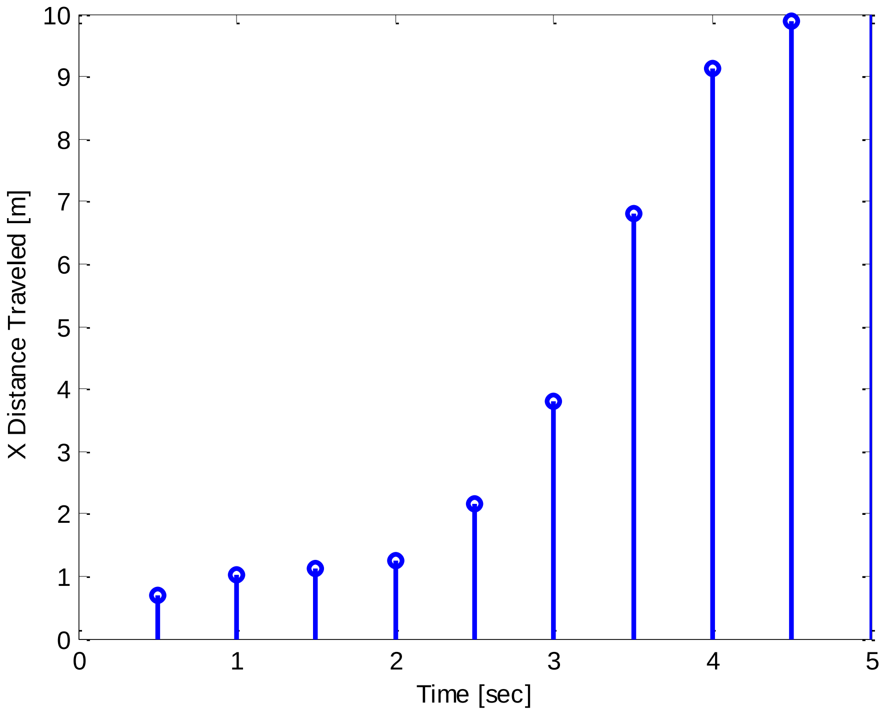
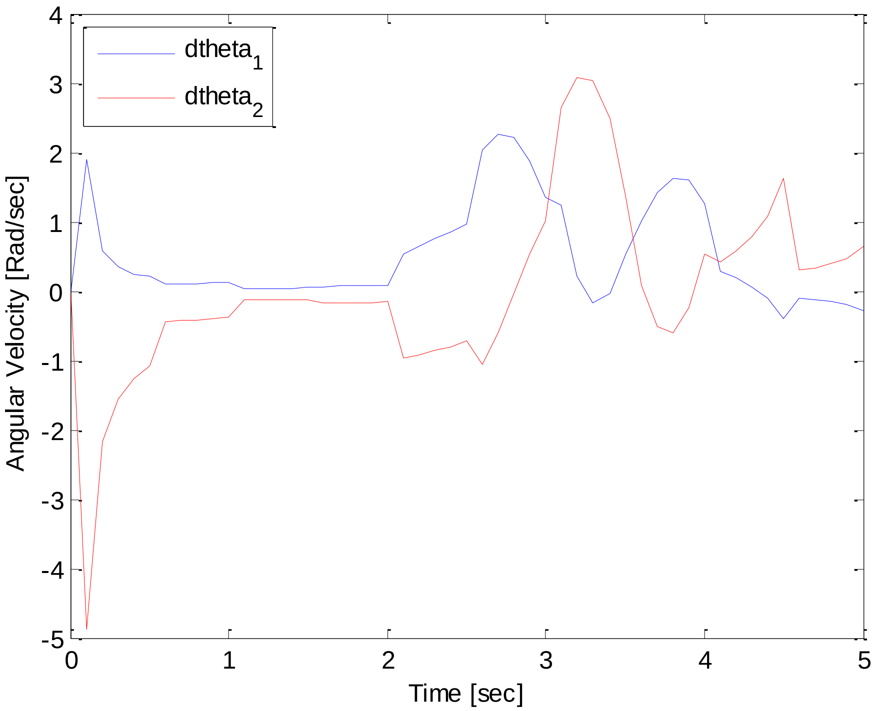


Figure 3: Control of Links



7.4.3. Parametric Study Results – Using Link Parameters from Initial Results

Table 5: Results from Parametric Study

d_1	d_2	d_3	d_4	d_5
.6774 m	.6274 m	1.2514 m	2.024 m	3.000 m

d_6	d_7	d_8	d_9	d_{10}	Energy
1.9176 m	.2023 m	.100 m	.100 m	.100 m	46.48 kJ

Figure 4: Parametric Study Optimization

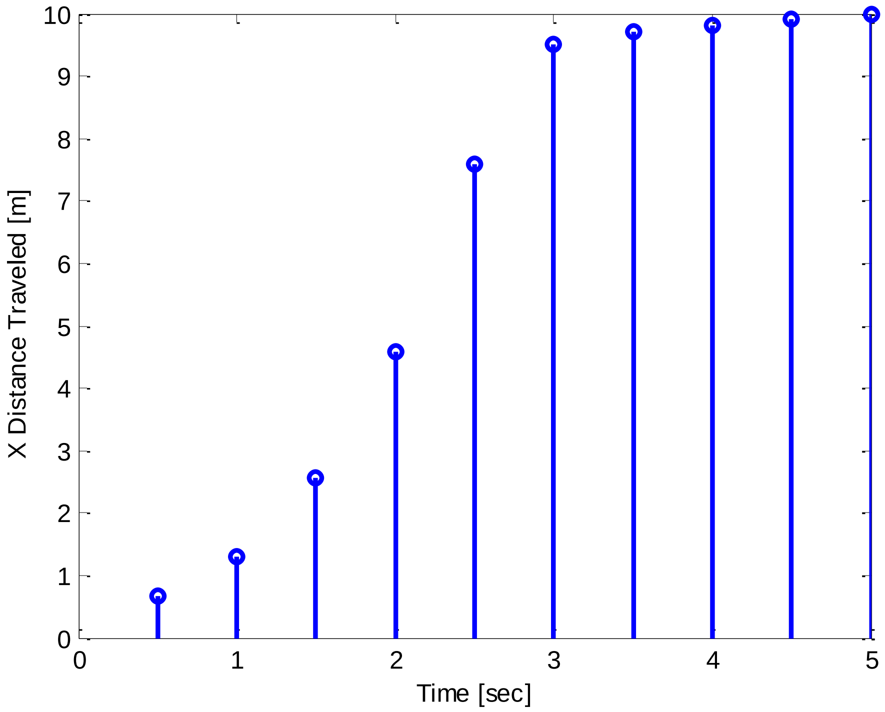
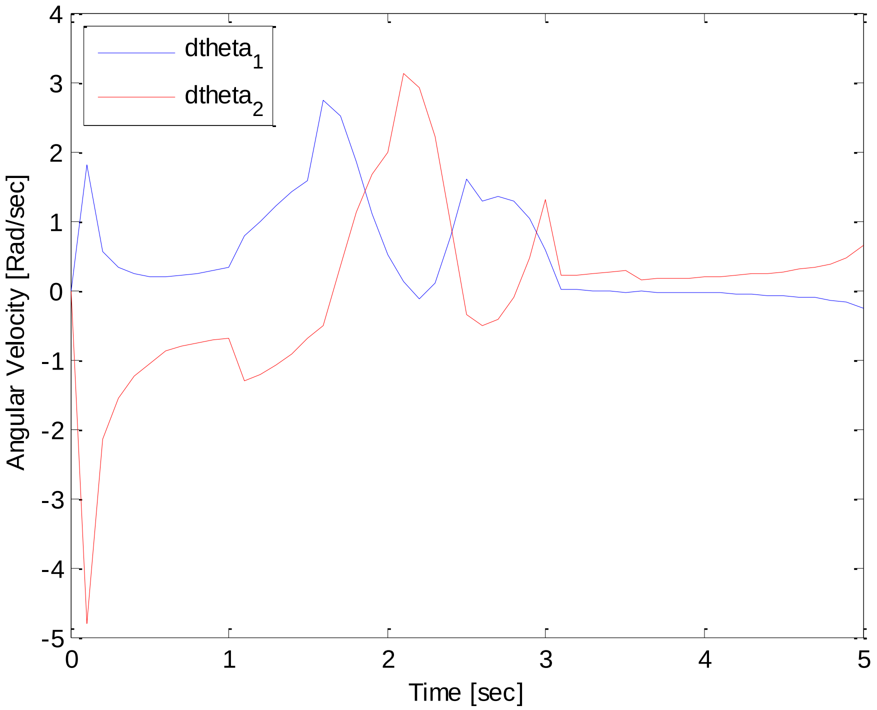


Figure 5: Parametric Study Link Control


dtheta$_1$
dtheta$_2$
Angular Velocity [Rad/sec]
Time [sec]