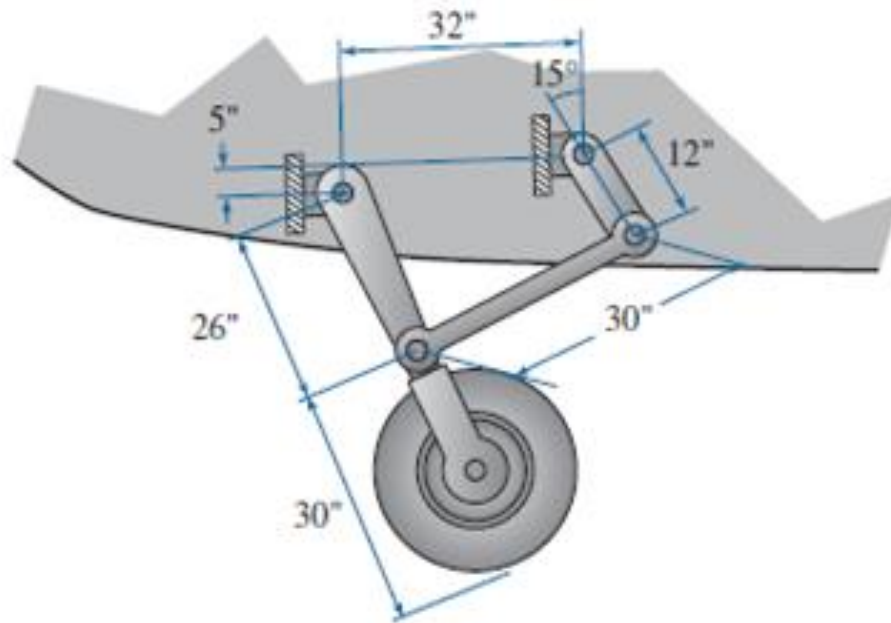


Airplane Landing Gear
ME 3011 Kinematics & Dynamics of Machines
Capstone Term Project



Jonathan Bowman
Russ College of Engineering
Ohio University

October 15th, 2019
Revised: July 31st, 2020

ME 3011 - Kinematics & Dynamics of Machines
Dr. Robert Williams

Abstract

Kinematic and Dynamic Analysis of an airplane's landing gear is detailed. With 90,000 planes in the air at any given time perhaps the most crucial element for a safe return is the airplane landing equipment. This analysis is modeled on a one degree-of-freedom with one motor input. Complete position, velocity and acceleration analysis can be detailed for the gears entire range of motion, including a snapshot of a point of interest. The dynamic analysis details the input torque and the shaking force and moment acting on the ground link.

Table of Contents

Abstract.....	2
1 Introduction	5
2 Background	5
3 Kinematic Diagram	6
3.1 Mobility and Degrees of Freedom	7
3.2 Grashof Analysis.....	7
3.3 Limits of Landing Gear	8
3.4 Position, Velocity and Acceleration Inputs	8
4 Kinematic Analysis	9
4.1 Position Analysis	9
4.1.1 Graphic Solution.....	11
4.1.2 Using MATLAB to Show a Point of Interest.....	11
4.2 Velocity Analysis.....	12
4.3 Acceleration Analysis	13
5 MATLAB Results and Discussion	14
5.1 Position Graphs	15
5.2 Velocity Graphs	16
5.3 Acceleration Graphs.....	17
5.4 Methodology Comparisons.....	18
6 Inverse Dynamics Problem Statement.....	19
6.1 Center of Gravity Estimation.....	19
6.2 Mass Estimation	19
6.3 Mass Moment of Inertia	20
6.4 Calculations and Tabulation.....	21
7 Inverse Dynamics Analysis	23
7.1 Link 2 Details	23
7.1.1 Link 2 Snapshot Analysis	23

7.2 Link 3 Details	24
7.2.1 Link 3 Snapshot Analysis	24
7.3 Link 4 Details	24
7.3.1 Link 4 Snapshot Analysis	25
7.4 Matrix Solutions	25
7.5 Solution Discussion	26
8 MATLAB Inverse Dynamics Results and Discussion	27
8.1 Input Torque	27
8.2 Shaking Moment Force and Moment	27
9 Further Development.....	29
10 References	30
Appendix A: Position Analysis Mathematics.....	31
Appendix B: Velocity Analysis Mathematics	33
Appendix C: Acceleration Analysis Mathematics.....	35
Appendix D: MATLAB Code Kinematic Analysis.....	36
Appendix E: MATLAB Code for Point of Interest	38
Appendix F: Link Mass Calculations	41
Appendix G: Rotational Moments of Inertia Calculations	41
Appendix H: Link Details	43
Appendix I: Inverse Dynamics Mathematics.....	46
Appendix J: MATLAB Code Inverse Dynamics.....	49
Appendix K: MATLAB Matrix Solver	55

1 Introduction

A full kinematic and dynamic study of airplane landing gear will be detailed. This includes position, velocity and acceleration analysis of the full range of motion of the landing gear system.

2 Background

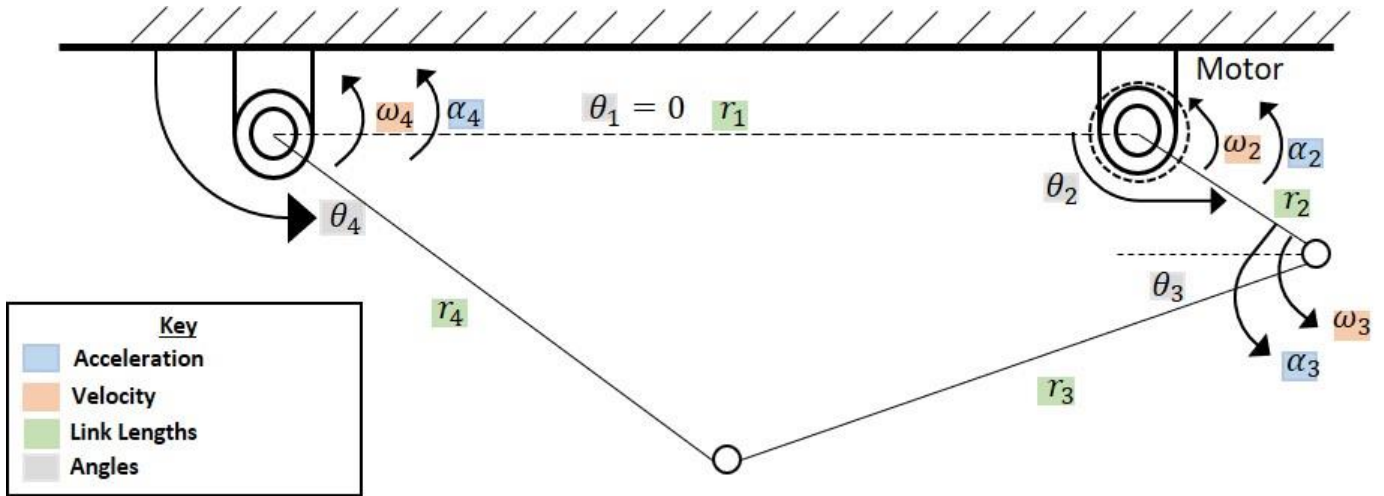
Airplane landing gear, much like the airplanes that they find themselves on are not entirely alike and can be quite complex. This model will be representing a very simple four bar model to simulate the motion of the ascension and retraction of the landing gear system. A real-life version of an example of this kind of airplane landing gear is shown in Figure 1.



Figure 1 Real World Landing Gear Mechanism

3 Kinematic Diagram

A kinematic diagram is essential to describing how this mechanism will move. In Figure 2 the linkage lengths are denoted by r_i and the angle between the x axis and the lengths by θ_i . The angular velocities are denoted by an ω_i and the angular acceleration is described by α_i .



To begin our analysis, we will need to quantify the input values for position, velocity and acceleration. These values can be found listed in Table 1.

Table 1: Table of Input Values

Input Information	Symbol	Value
Link Number 1	r_1	32 in.
Link Number 2	r_2	12 in.
Link Number 3	r_3	30 in.
Link Number 4	r_4	26 in.
Input Angle (Snapshot)	θ_2	315 degrees
Input Velocity (Snapshot)	ω_2	0.824 rad/s
Input Acceleration (Snapshot)	α_2	-0.2303 rad/s ²

3.1 Mobility and Degrees of Freedom

The degrees-of-freedom (dof) of a device describes the number of ways that it can move. From our understanding of the landing gear mechanism, a dof of 1 is desired. This is implied by the name, as a mechanism is a device that has one and only one degree-of-freedom. We can ensure that this is in fact the case by using Kutzbach's Mobility Equation (EQ 1) for Planar Jointed Devices to confirm the dof.

$$M = 3(N - 1) - 2 * J_1 - J_2 \text{ (EQ. 1)}$$

In this Mobility Equation we have several variables to define. N will be the number of links and will include the ground link. The number of one dof joints will be J_1 (ex. revolute or prismatic joints) and the number of two dof joints (ex. Cam or gear joints) will be J_2 .

The landing gear mechanism that is being analyzed will have a total number of links of 4. Additionally, the mechanism will be entirely comprised of 4 revolute joints or J_1 . It can be proven that the mobility will be 1, just as we were expecting.

$$M = 3(4 - 1) - 2 * 4 - 0 = 1$$

3.2 Grashof Analysis

Some four bar mechanisms are somewhat unique in that we can predict the nature of the rotation of the input and output links using the ground link location and the length of the links. When analyzed two outcomes can be determined: a link can be considered a **crank** which has full rotatability or a **rocker** that will have a limited range of rotatability.

To determine whether our landing gear can be considered a Grashof Mechanism, we will need understanding of Grashof's Law. Begin by considering the longest link as L, the shortest link as S, and the two-intermediate links as P and Q. To qualify as a Grashof Mechanism, $L + S < P + Q$.

$$\textit{We have: } 32 + 12 < 30 + 26$$

Now that we have qualified as a Grashof Mechanism, we can begin to investigate the kinematic inversion that fits our mechanism. We know that our shortest link S, is going to be adjacent to the ground link for in mechanism. According to Grashof, we can expect to have an input of a crank and an output of a rocker.

3.3 Limits of Landing Gear

While it is possible to have an output of a crank-rocker, we cannot expect this in our real-life mechanism. When the landing gear is lowered, the wheels will need to make a 90-degree contact with the ground to safely support the weight of the aircraft. Inversely, when the gear is raised it will need to be tucked into the airplane to give better performance and fuel economy. Therefore, we can consider the limits of the input angle as shown in our kinematic diagram in Section 3 as $360 > \theta > 236.09$.

3.4 Position, Velocity and Acceleration Inputs

When we look at a snapshot of this mechanism and analyze the kinematics, we want to be sure that the inputs we choose are meaningful. The position input represented here by θ_2 , is limited due to factors described in Section 3.3. It can be assumed that the kinematics at the halfway point between the fully extended and the fully retracted will be a point experiencing a unique velocity and acceleration. For this position we will consider $\theta_2 = 315^\circ$.

The input velocity will also need to be estimated. We estimate that a typical landing gear system will take about 5 seconds to retract and will travel 26 inches in our mechanism as shown in the kinematic diagram in Figure 1. Given a cycloidal input for the rotational acceleration, the input now is $\omega_2 = 0.824 \frac{rad}{s}$.

Finally, we can expect this mechanism to have an acceleration as it ascends and retracts. Following the same cycloidal function, we can expect $\alpha_2 = -0.2303 \frac{rad}{s^2}$.

4 Kinematic Analysis

4.1 Position Analysis

When we begin our position analysis of this mechanism, we will need to begin by defining our givens and our desired values. These values are summarized in Table 2.

Table 2: Position Given and Desired Values

Given	$r_1, r_2, r_3, r_4, \theta_1, \theta_2$
Find	θ_3, θ_4

We can then derive our Vector Loop Diagram shown in Figure 3 and the Vector Loop Eq (VLE) in Equation 2.

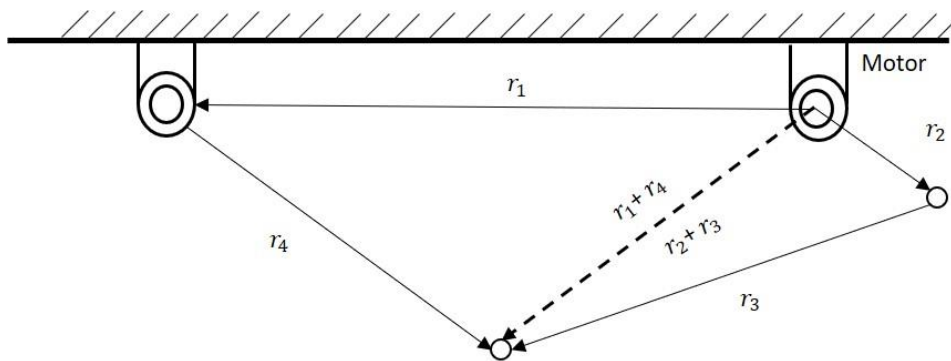


Figure 3: Vector Loop Diagram

$$\text{Vector Loop Equation: } r_2 + r_3 = r_1 + r_4 \text{ (EQ 2)}$$

The derivation of θ_4 can be found in appendix A and the resulting Equation 4 is shown reproduced below. We can plug our given values into G, F, and E from Table 2 and solve for t from equation 3.

$$E = 2r_4(-r_1c_1 - r_2c_2) = 2 * 26(-32 * \cos 0^\circ - 12 * \cos 315^\circ) = -2105.23$$

$$F = 2r_4(r_1s_1 - r_2s_2) = 2 * 26(32 * \sin 0^\circ - 12 * \sin 315^\circ) = 441.23$$

$$\begin{aligned} G &= r_1^2 + r_4^2 + r_2^2 - r_3^2 + 2r_1r_2 \cos(\theta_1 + \theta_2) \\ &= 32^2 + 26^2 + 12^2 - 30^2 + (2 * 32 * 12 * \cos(0^\circ + 315^\circ)) = 1487.06 \end{aligned}$$

We will only look at the positive open branch, therefore only 1 t.

$$t_1 = \frac{-F \pm \sqrt{F^2 - G^2 + E^2}}{(G - E)} = \frac{-441.23 - \sqrt{441.23^2 - 1487.06^2 + (-2105.23)^2}}{(1487.06 - (-2105.23))} = -0.555$$

Now we can plug in for θ_4 :

$$\theta_4 = 2 \tan^{-1}(t) = 2 \tan^{-1}(-0.555) = -58.1^\circ = 301.9^\circ$$

We can now use equation # to solve for θ_3 :

$$\theta_3 = \tan^{-1} \frac{-r_1 s_1 - r_4 s_4 + r_2 s_2}{-r_1 c_1 + r_4 c_4 - r_2 s_2} = \tan^{-1} \frac{-32 * \sin(0^\circ) - 26 * \sin(-58.1^\circ) + 12 * \sin(315^\circ)}{-32 * \cos(0^\circ) - 26 * \cos(-58.1^\circ) - 12 * \cos(315^\circ)}$$

$$\theta_3 = -26.92^\circ$$

4.1.1 Graphic Solution

A graphic solution of the derived solutions from section 4.1 can be shown. The solution detailing this landing gear mechanism can be found in Figure 4.

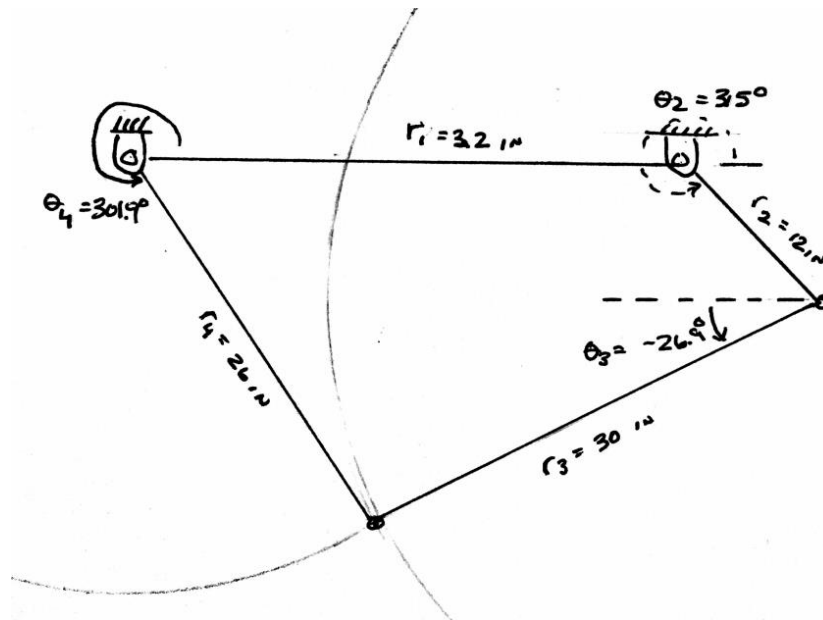


Figure 4: Graphical Solution Representation of the Landing Gear Mechanism

4.1.2 Using MATLAB to Show a Point of Interest

It can be considered that the most important point on this mechanism is the connection point between the traditional four bar mechanism and the wheel. This point can be found at the intersection of the r_3 and r_4 . If this point is defined in the MATLAB code and just how this point will move over time is shown. Figure 6 shows this result using the MATLAB Code in Appendix E.

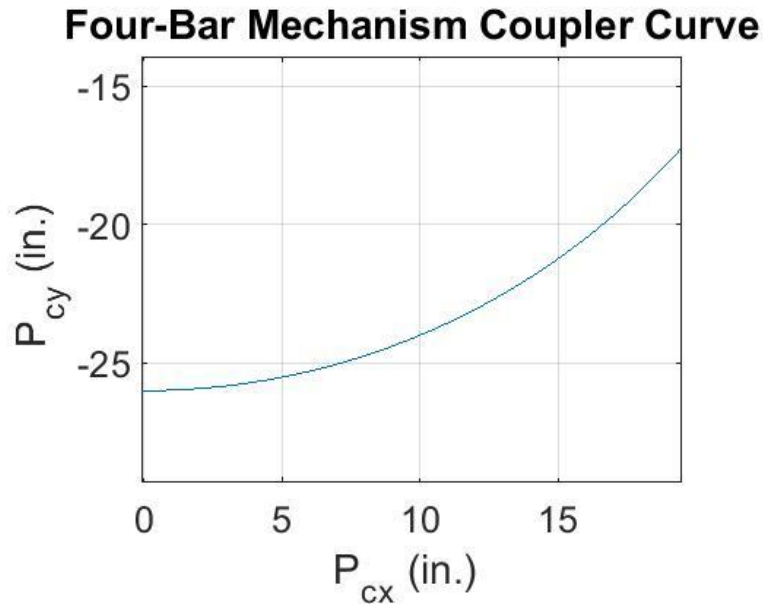


Figure 6: Point of interest displacement in X & Y.

4.2 Velocity Analysis

Much like the position analysis, we will start again by defining our givens and our desired values. These values for velocity are summarized in Table 3.

Table 3: Velocity Analysis Given and Desired Values

Given	$r_1, r_2, r_3, r_4, \theta_1, \theta_2, \theta_3, \theta_4; \omega_2$
Find	ω_3, ω_4

Using the derivation from Appendix B, we can label our constants and solve for ω_4 from equation 5.

$$a = r_3 s_3 = 30 \sin(-26.9^\circ) = -13.6 \quad b = r_4 s_4 = 26 \sin(301.9^\circ) = -22.07$$

$$d = -r_3 c_3 = 30 \cos(-26.9^\circ) = -26.75 \quad e = r_4 c_4 = 26 \cos(301.9^\circ) = 13.74$$

$$c = r_2 \omega_2 s_2 = 12 * 0.824 * \sin(315^\circ) = -6.992 \quad f = r_2 \omega_2 c_2 = 12 * 0.824 * \cos(315^\circ) = 6.992$$

$$\omega_4 = \frac{af - dc}{ae - db} = \frac{(-13.6)(6.992) - (-26.75)(-6.992)}{(-13.6)(13.74) - (-26.75)(-22.07)} = 0.363$$

Now we can solve for ω_3 using equation 6.

$$\omega_3 = \frac{ce - bf}{ae - db} = \frac{(-6.992)(13.74) - (-22.07)(6.992)}{(-13.6)(13.74) - (-26.75)(-22.07)} = -0.0749$$

4.3 Acceleration Analysis

As with both the Position and Velocity Equations, the givens and desired values can be found in Table 4.

Table 4: Acceleration Analysis Given and Desired Values

Given:	$r_1, r_2, r_3, r_4, \theta_1, \theta_2, \theta_3, \theta_4, \omega_2, \omega_3, \omega_4, \alpha_2$
Find:	α_3, α_4

Now using the derivation from Appendix C, we can label our constants and solve for α_4 from equation 6.

$$\begin{aligned}
 a &= r_3 s_3 = 30 * \sin(-26.3^\circ) = -13.6 & b &= r_4 s_4 = 26 * \sin(301.9^\circ) = -22.07 \\
 d &= -r_3 c_3 = -30 * \cos(-26.3^\circ) = -26.75 & e &= r_4 c_4 = 26 \cos(301.9^\circ) = 13.74 \\
 c &= r_2 \alpha_2 s_2 + r_2 \omega_2^2 c_2 - r_3 \omega_3^2 c_3 - r_4 \omega_4^2 c_4 \\
 &= 12 * (-0.2303) * \sin(-26.3^\circ) + 12 * (0.82404^2) * \cos(315^\circ) + 30 \\
 &\quad * (-0.075^2) \cos(-26.3^\circ) + 26 * (-0.363^2) \cos(301.9^\circ) = 5.7543 \\
 f &= r_2 \alpha_2 c_2 - r_2 \omega_2^2 s_2 - r_3 \omega_3^2 s_3 + r_4 \omega_4^2 s_4 \\
 &= 12 * (-0.2303) \cos(315^\circ) - 12 * (0.82404^2) \sin(315^\circ) + 30 \\
 &\quad * (-0.075^2) \sin(-26.3^\circ) - 26 * (-0.363^2) \sin(301.9^\circ) = 0.9739
 \end{aligned}$$

$$\alpha_4 = \frac{af - dc}{ae - bd} = \frac{(-22.07 * 0.9739) - (-26.75 * 5.7543)}{(-13.6 * 13.74) - (-22.07 * -26.75)} = -0.181$$

And now we can solve for α_3 with equation 7:

$$\alpha_3 = \frac{ce - bf}{ae - db} = \frac{(5.7543 * 13.74) - (-22.07 * 0.9739)}{(-13.6 * 13.74) - (-22.07 * -26.75)} = -0.1294$$

5 MATLAB Results and Discussion

An analysis was also completed in MATLAB and can be compared to the results of the subsequent sections. The complete MATLAB code can be found in Appendix D. A full range of motion of the mechanism was animated to the screen and can be seen in Figure 5.

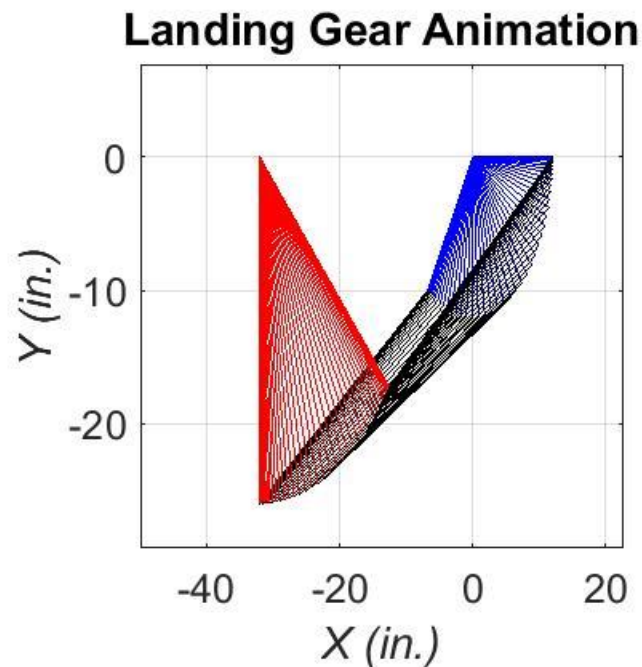


Figure 5 Landing Gear Mechanism Animation

This full range of motion diagram also provides support for the input θ_2 angles determined in section 3.4. It should also be noted that to make the gear retract to a perfectly straight alignment the ground link r_1 will need to be lengthened.

A plot comparing the input angle, velocity and acceleration can be seen in Figure 6. The input lengths will need to be modeled as a sinusoidal wave to avoid the intense loads at the start and end of the mechanisms motion. This is especially crucial in that of landing gear, as meaningful data will be needed to ensure that there's appropriate factors of safety.

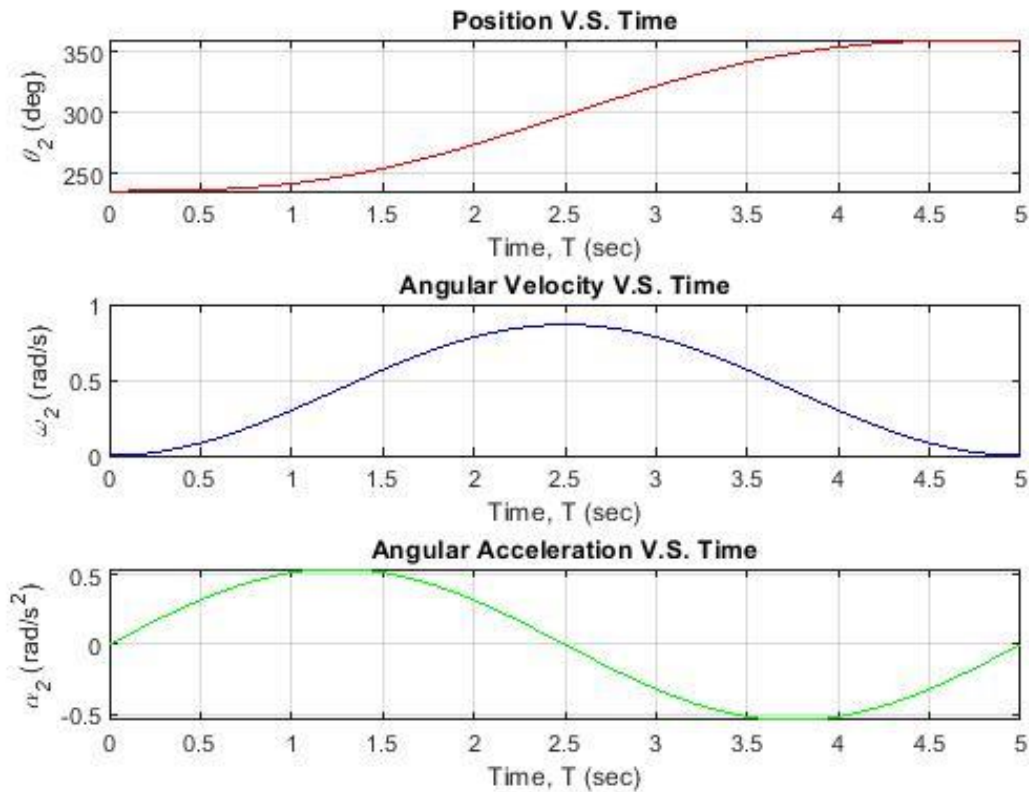


Figure 6: Sinusoidal Inputs for θ_2 , ω_2 , α_2 V.S. Time, t

5.1 Position Graphs

When modeled in MATLAB the θ values can be plotted in comparison to each other. These graphs represent much of how we would expect these angles to move Figure 7. The θ_3 will start by swinging open briefly and then will gradually decrease to the final retracted position. Additionally, θ_4 will increase as this is the arm directly in line with the landing gear wheel. These angles are limited by the dimensions of the links.

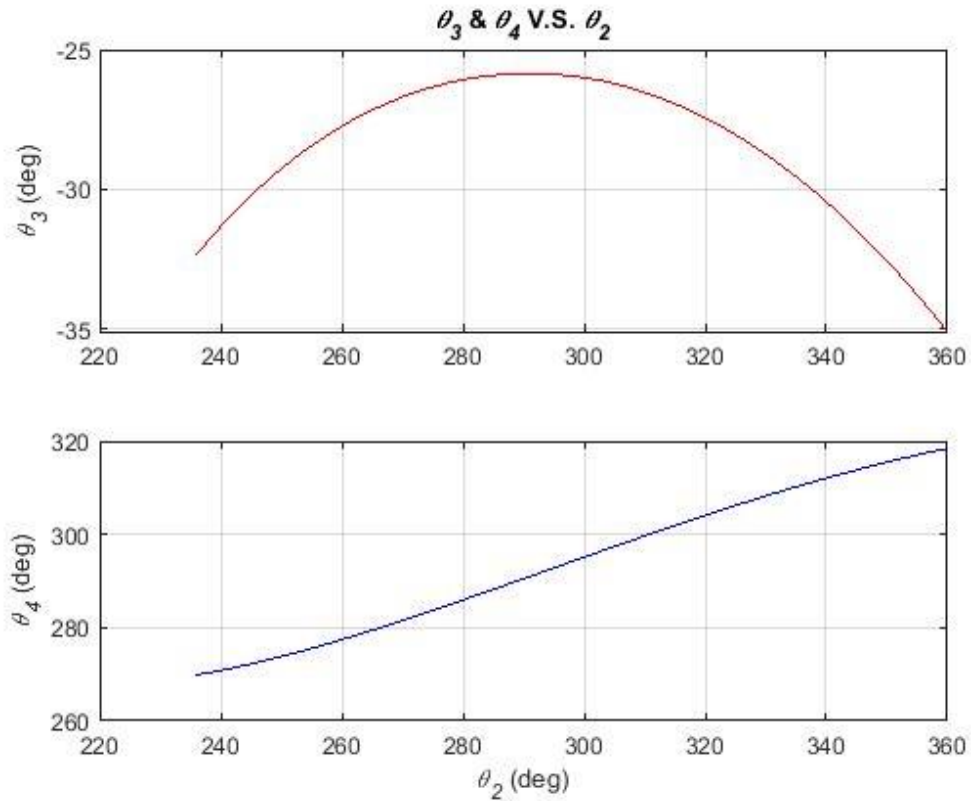


Figure 7: Mechanism ϑ Values

5.2 Velocity Graphs

The angular velocity values for Link 3 and 4 are shown in comparison to input angle θ_2 in Figure 8. A few trends in these graphs represent real life phenomenon. Both ω_3 and ω_4 start and end at 0, showing that this model will both begin at rest and will end at rest as well. The representation of ω_3 shows a sign change which corresponds to the links rotation, during the beginning of the cycle the link is in the negative x values and at the end is in the positive x position. This graph also reaffirms that ω_4 is moving in the path on the positive x axis. Both exhibit a gradual increase and decrease in speed, which is desirable because of the adverse effects of the momentum of the mechanism on the rest of the aircraft.

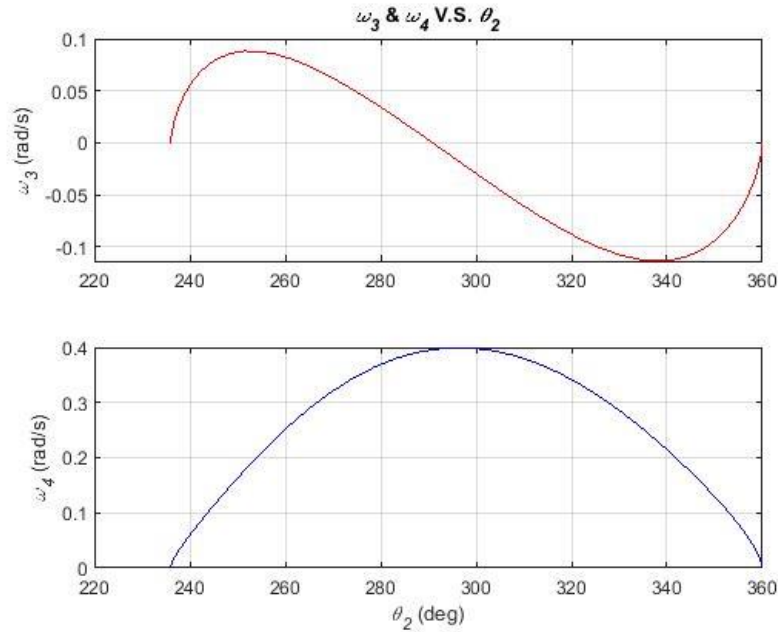


Figure 8 Angular Velocity and Position

5.3 Acceleration Graphs

Angular acceleration, α_3 and α_4 , are modeled throughout the cycle in Figure 9. Similarly to the velocity and analysis, both links begin and end at rest as desired. The accelerations follow suit as having a gradual increase and decrease which help to offset any momentum effects on the aircraft.

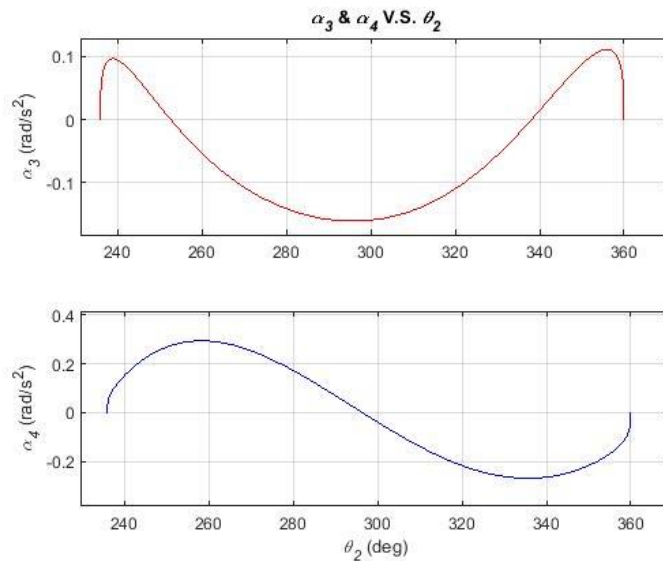


Figure 9 Angular Acceleration and Position Comparison

5.4 Methodology Comparisons

With the various methods of analysis now modeled in previous sections, Graphical Solution (Section 4.1.1), Snapshot Calculations (Sections 4.1, 4.2, 4.3) and the MATLAB Results (Section 5), we can compare them. The largest percent difference between the graphical solution and snapshot calculations for θ_3 and θ_4 is 4.6%. Similarly, the largest percent difference between the snapshot calculations and the MATLAB solution is 2.2%. Overall, this shows a very precise set of data and adds to the credibility of the results.

Table 5 Snapshot Solution Comparisons

Solution Type	θ_3	θ_4	ω_3	ω_4	α_3	α_4
Graphical Solutions	-27.5	305.0	N/A	N/A	N/A	N/A
Snapshot Calculations	-26.3	301.9	-0.075	0.363	-0.129	-0.181
MATLAB Solutions	-26.9	301.9	-0.075	0.363	-0.129	-0.181

6 Inverse Dynamics Problem Statement

If given the mechanism, the external forces and moments acting on the mechanism and the mechanism motion we can find the required driving force and the internal joint forces. A tabulated chart for the given and desired values can be found in Table 6.

Table 6: Inverse Dynamic Requirements and Unknowns

Given	$r_1, r_2, r_3, r_4, \theta_1, \theta_2, \theta_3, \theta_4, \omega_2, \omega_3, \omega_4, \alpha_2, \alpha_3, \alpha_4, A_{G2}, A_{G3}, A_{G4},$ $CG_2, CG_3, CG_4, I_{GZ2}, I_{GZ3}, I_{GZ4}, m_2, m_3, m_4, F_{E3}, F_{E4}, M_{E3}, M_{E4}$
Find	$\tau_2, F_{21}, F_{32}, F_{43}, F_{14}$

6.1 Center of Gravity Estimation

Center of gravity of the links is essential for the inverse dynamics portion of the analysis and will decide where the weight of each link will be acting. This will be assumed as being at the exact midpoint of the links, as shown in Table 7. This means that the angle between the link and the moment arms of the forces will be precisely 0°. This assumption will provide a benefit to the project as the absence of more complicated geometry of the links will make computations easier.

It can be estimated that the moment differences caused by the weight and external forces will be negligible for this analysis. It is important however, to acknowledge that the information gained can be useful if this project we're to be continued in the future, especially if designing landing gear for commercial use.

Table 7: Center of Mass Locations

	Link 2	Link 3	Link 4
Center of Mass, R_g (in.)	6	15	13

6.2 Mass Estimation

Finding direct schematics for a possible retractable landing gear mechanism will require more research time that will is not available with this report. In absence of this, an approximate

representation is given for the link diameters and material used in the manufacturing. An example of the landing gear this attempts to model is found in Figure 10.



Figure 10 The Approximate size of the landing gear modeled.

The masses will be determined using the volume that the links fill and the density of the material chosen. This will provide the best representation outside of having an actual physical mechanism to measure and weigh.

6.3 Mass Moment of Inertia

The links are assumed to have a slender rectangular cross section in the 2D plane that our analysis is occurring. As provided in the Mechanism Kinematics & Dynamics NotesBook, Figure 11 shows the assumed profile of each link. Around the centroid of a rectangular cross section the mass moment of

inertia is defined in the equation: $I_{GZ} = \frac{m \cdot L^2}{12}$

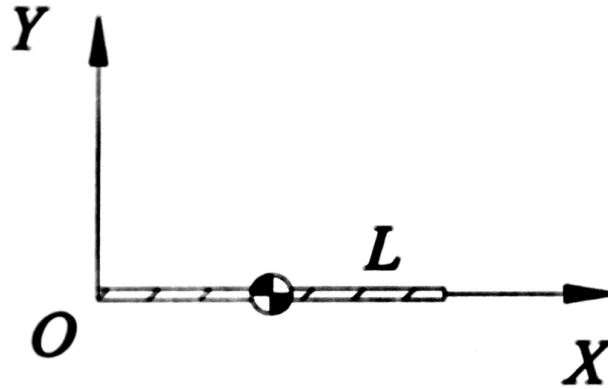


Figure 11: Source: Notebook - Planar Link Slender Rod [1]

6.4 Calculations and Tabulation

A common material for landing gear is high strength steel. This is for a few reasons, the first of which is that steel will have a very high strength which is required to support the weight of the aircraft. Steel is also resistant to fatigue or the cyclical loading of stress, such as turbulence in the air or from the jostling of an uneven landing strip. AISI 4340 Steel is assumed for the landing gear components and the material information from Matweb [5] is provided in Table 8.

Table 8 AISI 4340 Steel Material Properties

AISI 4340 Steel Normalized, 4 in Round	Yield Strength (KSI)	Density (lb _m /in ³)	Brinell Harness
	161	0.284	321

Each link will be considered to have a circular cross section with a diameter of 4 inches and will provide a cylindrical volume. This can then be multiplied by the density to give the mass of the link which can be factored into the inverse dynamics' analysis. Since link 1 is going to be considered the ground link we do not need the mass and the mass moment of inertia, and it will not be included. Table 9 summarizes the results shown in Appendix G.

Table 9 Mass and Mass Moments of Inertia

Quantity	Link 2	Link 3	Link 4
Mass (lb _m)	42.8	107.1	92.8
Mass (slug)	1.3	3.3	2.9
Mass Moment of Inertia (lb _m *in ²)	513.6	8032.5	5227.7
Mass Moment of Inertia (slug*in ²)	16.0	249.7	162.5

7 Inverse Dynamics Analysis

Building upon the Kinematic Analysis, this Inverse Dynamics section will show the forces and torques acting on the mechanism. For this analysis, it is considered that the mechanism to be tested in a laboratory setting and no external forces either from the plane weight or the wind resistance acting of the links. That is to say, F_e and $M_e = 0$.

A snapshot calculation has also been completed to confirm the values determined by the MATLAB Code. This snapshot calculation will build upon previous calculations determined in subsequent sections, a table of these values is reproduced in Table 10.

Table 10: Snapshot Calculation Values

Link #	Length (in)	Angle (deg)	Angular Velocity (rad/s)	Angular Acc. (rad/s ²)	Mass (slug)	Mass Moment of Inertia (slug*in ²)
1	32	0.0	N/A	N/A	N/A	N/A
2	12	315.0	0.824	-0.230	1.330	15.963
3	30	-26.3	-0.075	-0.129	3.329	249.660
4	26	301.9	0.363	-0.181	2.884	162.480

7.1 Link 2 Details

Using the equations from Appendix H, the details for link 2 can be shown as:

$$r_{12} = \begin{pmatrix} r_{12x} \\ r_{12y} \end{pmatrix} = \begin{pmatrix} -6 * \cos(\theta_2) \\ -6 * \sin(\theta_2) \end{pmatrix}$$

$$r_{32} = r_{12} + r_2 = \begin{pmatrix} -6 * \cos(\theta_2) + 12 * \cos(\theta_2) \\ -6 * \sin(\theta_2) + 12 * \sin(\theta_2) \end{pmatrix}$$

$$A_{g2} = \begin{pmatrix} A_{g2x} \\ A_{g2y} \end{pmatrix} = \begin{pmatrix} -6 * \alpha_2 * \sin(\theta_2) - 6 * \omega_2^2 * \cos(\theta_2) \\ 6 * \alpha_2 * \cos(\theta_2) - 6 * \omega_2^2 * \sin(\theta_2) \end{pmatrix}$$

7.1.1 Link 2 Snapshot Analysis

$$r_{12} = \begin{pmatrix} r_{12x} \\ r_{12y} \end{pmatrix} = \begin{pmatrix} -6 * \cos(315^\circ) \\ -6 * \sin(315^\circ) \end{pmatrix} = \begin{pmatrix} -4.24 \\ 4.24 \end{pmatrix}$$

$$r_{32} = r_{12} + r_2 = \begin{pmatrix} -6 * \cos(315^\circ) + 12 * \cos(315^\circ) \\ -6 * \sin(315^\circ) + 12 * \sin(315^\circ) \end{pmatrix} = \begin{pmatrix} 4.245 \\ -4.245 \end{pmatrix}$$

$$A_{g2} = \begin{pmatrix} A_{g2x} \\ A_{g2y} \end{pmatrix} = \begin{pmatrix} -6 * \alpha_2 * \sin(315^\circ) - 6 * \omega_2^2 * \cos(315^\circ) \\ 6 * \alpha_2 * \cos(315^\circ) - 6 * \omega_2^2 * \sin(315^\circ) \end{pmatrix} = \begin{pmatrix} -3.858 \\ 1.9039 \end{pmatrix}$$

7.2 Link 3 Details

Using the equations shown in Appendix H, the details equation can be shown as:

$$r_{23} = \begin{pmatrix} r_{23x} \\ r_{23y} \end{pmatrix} = \begin{pmatrix} -15 * \cos(\theta_3) \\ -15 * \sin(\theta_3) \end{pmatrix}$$

$$r_{43} = \begin{pmatrix} r_{43x} \\ r_{43y} \end{pmatrix} = \begin{pmatrix} -15 * \cos(\theta_3) + 30 * \cos(\theta_3) \\ -15 * \sin(\theta_3) + 30 * \sin(\theta_3) \end{pmatrix}$$

$$A_{g3} = \begin{pmatrix} A_{g3x} \\ A_{g3y} \end{pmatrix} \\ = \begin{pmatrix} -12 * \alpha_2 * \sin(\theta_2) - 12 * \omega_2^2 * \cos(\theta_2) - 15 * \alpha_3 * \sin(\theta_3) - 15 * \omega_3^2 * \cos(\theta_2) \\ 12 * \alpha_2 * \cos(\theta_2) - 12 * \omega_2^2 * \sin(\theta_2) + 15 * \alpha_3 * \cos(\theta_3) - 15 * \omega_3^2 * \sin(\theta_3) \end{pmatrix}$$

7.2.1 Link 3 Snapshot Analysis

$$r_{23} = \begin{pmatrix} r_{23x} \\ r_{23y} \end{pmatrix} = \begin{pmatrix} -15 * \cos(-26.3) \\ -15 * \sin(-26.3) \end{pmatrix} = \begin{pmatrix} 13.446 \\ 6.648 \end{pmatrix}$$

$$r_{43} = \begin{pmatrix} r_{43x} \\ r_{43y} \end{pmatrix} = \begin{pmatrix} -15 * \cos(-26.3) + 30 * \cos(-26.3) \\ -15 * \sin(-26.3) + 30 * \sin(-26.3) \end{pmatrix} = \begin{pmatrix} -13.448 \\ -6.441 \end{pmatrix}$$

$$A_{g3} = \begin{pmatrix} A_{g3x} \\ A_{g3y} \end{pmatrix} = \begin{pmatrix} -12 * -0.2303 * \sin(315) - 12 * 0.82404^2 * \cos(315) \\ 12 * -0.2303 * \cos(315) - 12 * 0.82404^2 * \sin(315) \\ -15 * -0.129 * \sin(-26.3) - 15 * -0.075^2 * \cos(-26.3) \\ + 15 * -0.129 * \cos(-26.3) - 15 * -0.075^2 * \sin(-26.3) \end{pmatrix} = \begin{pmatrix} -8.536 \\ 1.997 \end{pmatrix}$$

7.3 Link 4 Details

Using the equations derived in Appendix H, the details equation can be shown as:

$$r_{14} = \begin{pmatrix} r_{14x} \\ r_{14y} \end{pmatrix} = \begin{pmatrix} -13 * \cos(\theta_4) \\ -13 * \sin(\theta_4) \end{pmatrix}$$

$$r_{34} = r_{14} + r_4 = \begin{pmatrix} -13 * \cos(\theta_4) + 26 * \cos(\theta_4) \\ -13 * \sin(\theta_4) + 26 * \sin(\theta_4) \end{pmatrix}$$

$$A_{g4} = \begin{pmatrix} A_{g4x} \\ A_{g4y} \end{pmatrix} = \begin{pmatrix} -13 * \alpha_4 * \sin(\theta_4) - 13 * \omega_4^2 * \cos(\theta_4) \\ 13 * \alpha_4 * \cos(\theta_4) - 13 * \omega_4^2 * \sin(\theta_4) \end{pmatrix}$$

7.3.1 Link 4 Snapshot Analysis

$$r_{14} = \begin{pmatrix} r_{14x} \\ r_{14y} \end{pmatrix} = \begin{pmatrix} -13 * \cos(301.9) \\ -13 * \sin(301.9) \end{pmatrix} = \begin{pmatrix} -6.8697 \\ 11.037 \end{pmatrix}$$

$$r_{34} = r_{14} + r_4 = \begin{pmatrix} -13 * \cos(301.9) + 26 * \cos(301.9) \\ -13 * \sin(301.9) + 26 * \sin(301.9) \end{pmatrix} = \begin{pmatrix} 6.8696 \\ -11.0366 \end{pmatrix}$$

$$A_{g^4} = \begin{pmatrix} A_{g^4x} \\ A_{g^4y} \end{pmatrix} = \begin{pmatrix} -13 * -0.181 * \sin(301.9) - 13 * 0.363^2 * \cos(301.9) \\ 13 * -0.181 * \cos(301.9) - 13 * 0.363^2 * \sin(301.9) \end{pmatrix} = \begin{pmatrix} -2.903 \\ 0.211 \end{pmatrix}$$

7.4 Matrix Solutions

Sections 7.1 – 7.3 details 9 equations and for our inverse dynamics problem we have a total of 9 unknowns (Table #). This means that a real solution can be determined and a way to handle these solutions efficiently is to utilize a matrix solution.

As detailed in Appendix I, the [A] matrix can be shown as:

$$[A] = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -4.24 & 4.24 & 4.245 & -4.245 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6.648 & -13.446 & 6.441 & -13.448 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -11.03 & -6.87 & -11.04 & -6.87 & 0 \end{bmatrix}$$

Additionally, the known matrix {b} can be described as:

$$\{b\} = \begin{bmatrix} -0.42769 \\ 43.05 \\ -0.02552 \\ -2.36802 \\ 107.74 \\ -0.22365 \\ -0.69769 \\ 92.92 \\ -0.204228 \end{bmatrix}$$

After performing the Inverse Matrix Solution in a separate MATLAB program and given in Appendix K, our unknowns at a snapshot are determined to be:

$$F_{12x} = -45.1402$$

$$F_{12y} = -119.6602$$

$$F_{32x} = -45.5679$$

$$F_{32y} = -76.6102$$

$$F_{43x} = -47.9359$$

$$F_{43y} = 31.1298$$

$$F_{14x} = -48.6336$$

$$F_{14y} = 124.0498$$

$$T_2 = 834.5850$$

7.5 Solution Discussion

It is important to acknowledge that these values do not align with the Full Range of Motion Code (FROM) discussed in Section 8. After troubleshooting, the {b} matrix is confirmed to be the exact same as the FROM code but the [A] matrix is unable to be confirmed with the code setup. For this reason, it is believed that the error is originating from here. It is believed that the MATLAB Code for FROM is the more reliable of the two sources because of the exact values that is used in calculations.

8 MATLAB Inverse Dynamics Results and Discussion

8.1 Input Torque

After solving using Matrix Inversion, the input torque can be determined. This input torque shown in Figure 12, shows the amount of torque needed to drive the mechanism to its retracted position. This is crucial when determining what type of motor is needed and how much power it will require.

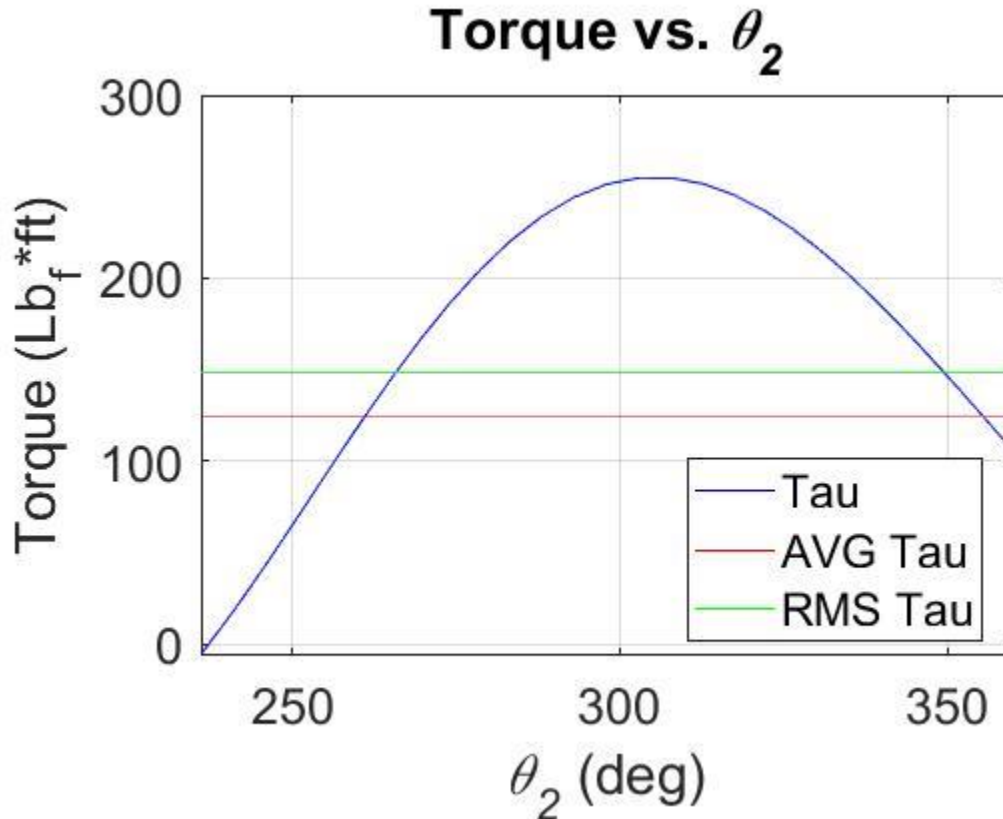


Figure 12: Driving Torque, Average Torque and Root-Mean-Squared Torque VS Input Angle

8.2 Shaking Moment Force and Moment

The shaking force and moment relate to the reaction force of the ground link r_1 . This is especially important in this case, as the ground link is designed to be attached to an aircraft. If the aircraft cannot withstand the reaction forces, the mechanism can become misaligned putting those on board in danger. The MATLAB results of these forces and moments, is shown in Figure 13.

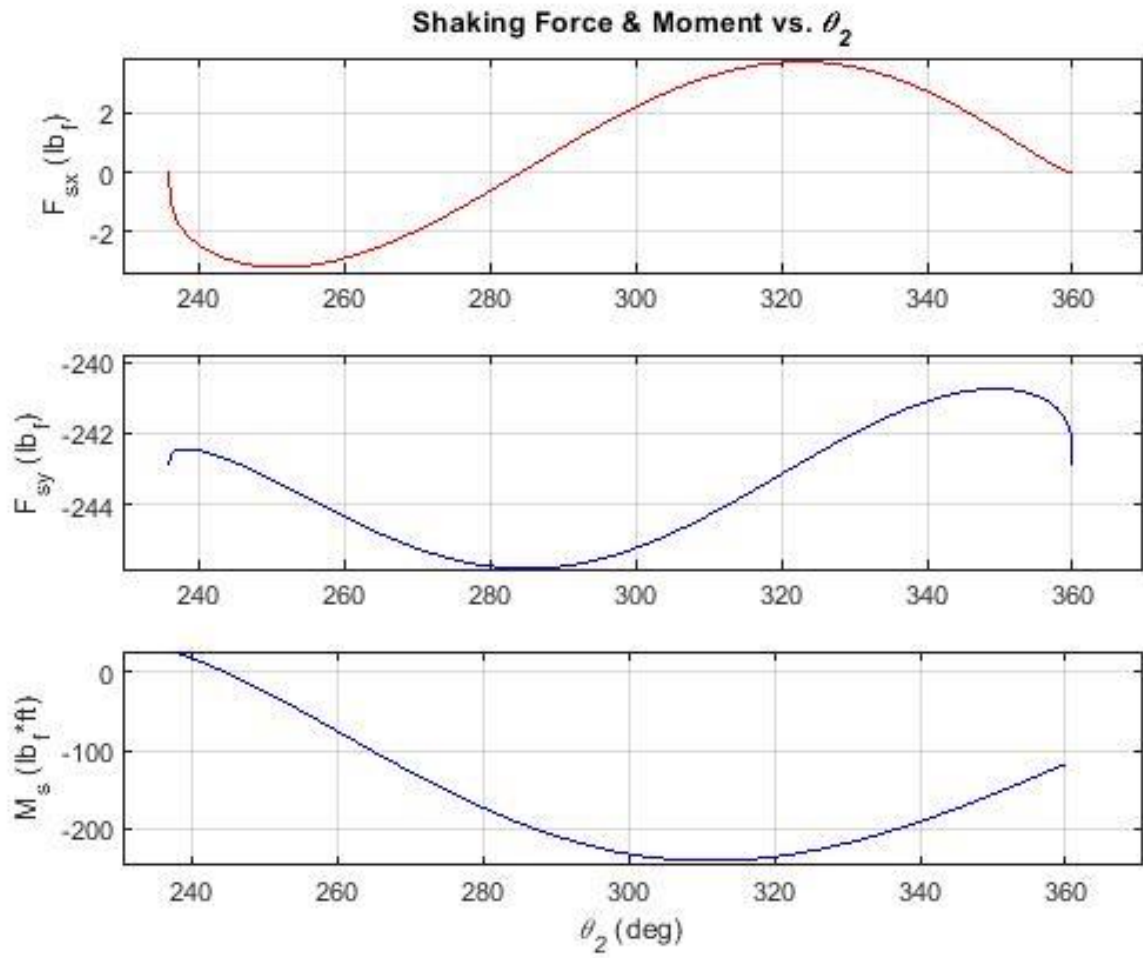


Figure 13: Shaking Force and Moment compared to Input Angle

9 Further Development

While the work presented in this report is to be best of the author's ability, if given the chance for further development a few things could be pursued. Firstly, a redesign of the link lengths will provide a more aerodynamic design. Secondly, some of the MATLAB results and snapshot results do not align exactly, and more time could work out those bugs. Lastly, some of the external forces that could present themselves during use could be included and would inevitably make the problem both more difficult and more useful.

10 References

- [1] Williams RL. Mechanism Kinematics & Dynamics. 2019th ed. LuLu; 2019.
- [2] "Beechcraft Bonanza," *Wikipedia*, 02-Oct-2019. [Online]. Available: https://en.wikipedia.org/wiki/Beechcraft_Bonanza. [Accessed: 03-Nov-2019].
- [3] "Chegg.com," 2–7. *Figure P2.7 Shows A Mechanism That Operates T...* | Chegg.com. [Online]. Available: <https://www.chegg.com/homework-help/questions-and-answers/2-7-figure-p27-shows-mechanism-operates-landing-gear-small-airplane-use-working-model-soft-q26836831>. [Accessed: 03-Nov-2019].
- [4] "AISI 4340 Steel, normalized, 100 mm (4 in.) round," AISI 4340 Steel, normalized, 100 mm (4 in.) round. [Online]. Available: <http://www.matweb.com/search/DataSheet.aspx?MatGUID=4a3cfc1e1cfd451091e67d3f3b66bb>[5] 80.
- [5] [Accessed: 18-Nov-2019].A. SpampinatoJun, "The Materials Used in the Design of Aircraft Wings," AZoM.com, 14-May-2019. [Online]. Available: <https://www.azom.com/article.aspx?ArticleID=12117>. [Accessed: 18-Nov-2019].

Appendix A: Position Analysis Mathematics

$$\text{Given: } r_1 = 32, r_2 = 12, r_3 = 30, r_4 = 26, \theta_1, \theta_2$$

$$\text{Find } = \theta_3, \theta_4$$

X & Y Components

$$x) r_2 \cos \theta_2 - r_3 \cos \theta_3 = -r_1 \cos \theta_1 + r_4 \cos \theta_4$$

$$y) -r_2 \sin \theta_2 - r_3 \sin \theta_3 = -r_1 \sin \theta_1 - r_4 \sin \theta_4$$

Square

$$x) (r_3 c_3)^2 = (r_1 c_1)^2 + (r_4 c_4)^2 + (r_2 c_2)^2 - 2 * r_1 c_1 r_4 c_4 - 2 * r_4 c_4 r_2 c_2 + 2 * r_1 c_1 r_2 c_2$$

$$y) (r_3 s_3)^2 = (r_1 s_1)^2 + (r_4 s_4)^2 + (r_2 s_2)^2 + 2 * r_1 s_1 r_4 s_4 - 2 * r_4 s_4 r_2 s_2 - 2 * r_2 s_2 r_1 s_1$$

Add X & Y, using trig identity.

$$r_3^2 = r_1^2 + r_4^2 + r_2^2 + 2r_4 c_4 (-r_1 c_1 - r_2 c_2) + 2r_4 s_4 (-r_1 s_1 - r_2 s_2) + 2r_1 r_2 (c_1 c_2 - s_1 s_2)$$

Simplify, Let:

$$E = 2r_4(-r_1 c_1 - r_2 c_2) \quad (\text{EQ \#4})$$

$$F = 2r_4(r_1 s_1 - r_2 s_2)$$

$$G = r_1^2 + r_4^2 + r_2^2 - r_3^2 + 2r_1 r_2 \cos(\theta_1 + \theta_2)$$

Now we have:

$$E \cos \theta_4 + F \sin \theta_4 + G = 0$$

Use:

$$t = \tan \frac{\theta_4}{2} \text{ so, } \cos \theta_4 = \frac{1 - t^2}{1 + t^2} \text{ and } \sin \theta_4 = \frac{2t}{1 + t^2}$$

Therefore:

$$E \frac{1 - t^2}{1 + t^2} + F \frac{2t}{1 + t^2} + G = 0$$

$$E(1 - t^2) + F(2t) + G(1 + t^2) = 0$$

$$E - Et^2 + F2t + G + Gt^2 = 0$$

$$(G - E)t^2 + 2F(t) + (G + E) = 0$$

Use Quadratic Equation

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$t_{1,2} = \frac{-2F \pm \sqrt{(2F)^2 - 4(G - E)(G + E)}}{2(G - E)}$$

$$t_{1,2} = \frac{-F \pm \sqrt{F^2 - G^2 + E^2}}{(G - E)} \quad (EQ \#3)$$

Solving For θ_4

$$\theta_4 = 2 \tan^{-1}(t_{1,2})$$

Solving For θ_3 , use ratio of Y to X

$$\frac{y}{x} = \frac{-r_3 s_3}{-r_3 c_3} = \frac{-r_1 s_1 - r_4 s_4 + r_2 s_2}{-r_1 c_1 + r_4 c_4 - r_2 s_2}$$

Therefore:

$$\theta_{3,2} = \tan^{-1} \frac{-r_1 s_1 - r_4 s_4 + r_2 s_2}{-r_1 c_1 + r_4 c_4 - r_2 s_2}$$

Appendix B: Velocity Analysis Mathematics

$$\text{Given: } r_1 = 32, r_2 = 12, r_3 = 30, r_4 = 26, \theta_1, \theta_2$$

$$\text{Find } = \theta_3, \theta_4$$

Previously,

$$\text{Vector Loop Equation: } r_2 + r_3 = r_1 + r_4$$

$$x) r_2 \cos \theta_2 - r_3 \cos \theta_3 = -r_1 \cos \theta_1 + r_4 \cos \theta_4$$

$$y) -r_2 \sin \theta_2 - r_3 \sin \theta_3 = -r_1 \sin \theta_1 - r_4 \sin \theta_4$$

Take First Time Derivative

$$x) -r_2 \omega_2 s_2 + r_3 \omega_3 s_3 = -r_4 \omega_4 s_4$$

$$y) -r_2 \omega_2 c_2 - r_3 \omega_3 c_3 = -r_4 \omega_4 c_4$$

Simplify and gather unknowns on Left Hand Side (LHS)

$$x) r_3 \omega_3 s_3 + r_4 \omega_4 s_4 = r_2 \omega_2 s_2$$

$$y) -r_3 \omega_3 c_3 + r_4 \omega_4 c_4 = r_2 \omega_2 c_2$$

Simplify, and let:

$$a = r_3 s_3 \quad b = r_4 s_4$$

$$c = r_2 \omega_2 s_2 \quad d = -r_3 c_3$$

$$e = r_4 c_4 \quad f = r_2 \omega_2 c_2$$

Now we have:

$$x) a \omega_3 + b \omega_4 = c$$

$$y) d \omega_3 + e \omega_4 = f$$

Solve for unknowns:

$$\omega_3 = \frac{c - b \omega_4}{a}$$

$$d \left(\frac{c - b \omega_4}{a} \right) + e \omega_4 = f$$

$$\omega_4 (ae - db) = af - dc$$

Therefore:

$$\omega_4 = \frac{af - dc}{ae - db} \quad (\text{EQ \#5})$$

Now, solve for ω_3

$$\omega_3 = \frac{ce - bf}{ae - db} \text{ (EQ \#6)}$$

Appendix C: Acceleration Analysis Mathematics

$$\text{Given: } r_1 = 32, r_2 = 12, r_3 = 30, r_4 = 26, \theta_1, \theta_2$$

$$\text{Find } = \theta_3, \theta_4$$

Previous Velocity Equations

$$x) -r_2\omega_2s_2 + r_3\omega_3s_3 = -r_4\omega_4s_4$$

$$y) -r_2\omega_2c_2 - r_3\omega_3c_3 = -r_4\omega_4c_4$$

Take First Time Derivative

$$x) -r_2\alpha_2s_2 - r_2\omega_2^2c_2 + r_3\alpha_3s_3 + r_3\omega_3^2c_3 = -r_4\alpha_4s_4 - r_4\omega_4^2c_4$$

$$y) -r_2\alpha_2c_2 + r_2\omega_2^2s_2 - r_3\alpha_3c_3 + r_3\omega_3^2s_3 = -r_4\alpha_4c_4 + r_4\omega_4^2s_4$$

Group unknowns on LHS

$$x) -r_3\alpha_3s_3 - r_4\alpha_4s_4 = r_4\omega_4^2c_4 - r_2\alpha_2s_2 - r_2\omega_2^2c_2 + r_3\omega_3^2c_3$$

$$y) r_3\alpha_3c_3 - r_4\alpha_4c_4 = -r_4\omega_4^2s_4 - r_2\alpha_2c_2 + r_2\omega_2^2s_2 + r_3\omega_3^2s_3$$

Let:

$$a = -r_3s_3 \quad b = -r_4s_4 \quad c = -r_2\alpha_2s_2 - r_2\omega_2^2c_2 + r_3\omega_3^2c_3 + r_4\omega_4^2c_4$$

$$d = r_3c_3 \quad e = -r_4c_4 \quad f = -r_2\alpha_2c_2 - r_2\omega_2^2s_2 + r_3\omega_3^2s_3 + r_4\omega_4^2s_4$$

This Gives:

$$x) a\alpha_3 + b\alpha_4 = c$$

$$y) d\alpha_3 + e\alpha_4 = f$$

Solve for unknowns:

$$\alpha_3 = \frac{c - b\alpha_4}{a}$$

$$dc - db\alpha_4 + ae\alpha_4 = af$$

$$\alpha_4(ae - db) = af - dc$$

$$\alpha_4 = \frac{af - dc}{ae - bd} \quad (\text{EQ \#7})$$

Now we can find that:

$$\alpha_3 = \frac{ce - bf}{ae - bd} \quad (\text{EQ \#8})$$

Appendix D: MATLAB Code Kinematic Analysis

```
10/18/19 1:00 AM Y:\ProjectFourBarFullRangeOfMotionAddOn.m

%-----
% Jonathan Bowman Air Plane Landing Gear Four Bar Kinematic Range of Motion
% ME 3011 Intern Report
%-----

clc; clear; % clear screen and defined variables

% Inputs
DR = pi/180;

%Position Inputs
r1 = 32; r2 = 12; r3 = 30; r4 = 20; rca = 0;

th1 = 0*DR; th2 = 315*DR; % conversion of degree to rad
r1x = r1*cos(th1); r1y = r1*sin(th1);

th20 = 270*DR; th2f = 360*DR; th2 = [th20:th2f]*DR; % th2 array

t0 = 0; dt = 0.1; tf = 5; t = [t0:dt:tf]

N = (tf-t0)/dt + 1; % number of times to repeat loop for F.R.O.M. using time!
figure;

for i=1:N, % F.R.O.M. loop over all input th2

    %Full Cycloidal FNC From Page 82

    th2(i) = th20+(th2f-th20)*((t(i)/tf)-((1/(2*pi))*sin(2*pi*t(i)/tf)));
    w2(i) = ((th2f - th20)/tf)*(1 - cos((2*pi*t(i))/tf));
    alpha2(i) = ((2*pi*(th2f - th20))/(tf^2))*sin((2*pi*t(i))/tf);
    beta2(i) = ((4*(pi^2)*(th2f - th20))/(tf^3))*cos((2*pi*t(i))/tf);

    % Position analysis: Solving for theta4
    E = 2*r4*(-r1*cos(th1) - r2*cos(th2(i)));
    F = 2*r4*(r1*sin(th1) - r2*sin(th2(i)));
    G = r1^2 + r2^2 - r3^2 + r4^2 + 2*r1*r2*cos(th1+th2(i));

    topen = (-F - sqrt(E^2 + F^2 - G^2)) / (G-E); % open branch
    th4(i) = 2*atan(topen); %The 2 Multiplier is important!

    % Solve for th3, coupler point, transmission angle; open branch only
    ax = -r2*cos(th2(i)); % Point A
    ay = r2*sin(th2(i));
    bx = r4*cos(th4(i)); % Point B
    by = -r4*sin(th4(i));
    cx = -r1*cos(th1);
    cy = -r1*sin(th1);
    th3(i) = atan2(by+ay+cy,bx+ax+cx); % theta3
    mu(i) = abs(th4(i)-th3(i)); % transmission angle
    pcx(i) = r2*cos(th2(i));
    pcy(i) = r2*sin(th2(i));

    %Velocity Analysis Open Branch
    a = r3*sin(th3(i)); %Define Variables
    b = r4*sin(th4(i));
    c = r2*w2(i)*sin(th2(i));
```

```

d = -r3*cos(th3(i));
e = r4*cos(th4(i));
f = r2*w2(i)*cos(th2(i));

%Solutions

w4(i) = ((a*f)-(d*c))/((a*e)-(d*b)); %Velocity Outputs
w3(i) = ((c*e)-(b*f))/((a*e)-(d*b));

%Acceleration Analysis

C = -r2*alpha2(i)*sin(th2(i))-r2*((w2(i))^2)*cos(th2(i)) + r3*((w3(i))^2)*cos(th3(i)) + r4*((w4
(i))^2)*cos(th4(i));
F = -r2*alpha2(i)*cos(th2(i))+r2*((w2(i))^2)*sin(th2(i)) + r3*((w3(i))^2)*sin(th3(i)) - r4*((w4
(i))^2)*sin(th4(i));

%Solving for Acceleration Alphas

alpha4(i) = ((-a*F)-(-d*C))/((-a*e)-(-b*d));
alpha3(i) = ((C*e)-(-b*F))/((-a*e)-(-b*d));

% Draw four-bar Mechanism to the screen, open branch only
x2 = [0 r2*cos(th2(i))]; % Coordinates for Link #2
y2 = [0 r2*sin(th2(i))];
x3 = [r2*cos(th2(i)) r1x+r4*cos(th4(i)) pcx(i)]; % Coordinates for link #3
y3 = [r2*sin(th2(i)) r1y+r4*sin(th4(i)) pcy(i)];
x4 = [r1x r1x+r4*cos(th4(i))]; % Coordinates for link #4
y4 = [r1y r1y+r4*sin(th4(i))];
plot(x2,y2,'b',x4,y4,'r');
patch(x3,y3,'g');
set(gca,'FontSize',19); xlabel('\itX (\itm)'); ylabel('\itY (\itm)');
axis('square'); axis([-r2 r1+r4 -(r1+r2+r4)/2 (r1+r2+r4)/2]); grid;
pause(1/3); %Pause to make animation visible
if i==1
    pause; % Clicking Enter Will Proceed Animation
end
end

% Plots outside loop

figure;
plot(t,th2/DR); xlabel('time (sec)'); ylabel('theta2 (deg)'); grid; title('Theta2 V.S. Time');
axis([0 5 269 361]);

figure;
plot(t,w2); xlabel('time (sec)'); ylabel('angular velocity (rad/s)'); grid; title('Omega2 V.S.
Time');
axis([0 5 0 0.6283]);

figure;
plot(t,alpha2); xlabel('time (sec)'); ylabel('angular acceleration (rad/s^2)'); grid; title('Alpha V.
S. Time');
axis([0 5 -4 4]);

```

Appendix E: MATLAB Code for Point of Interest

```
%-----  
% Jonathan Bowman Four Bar Landing Gear Mechanism  
% ME 3011 Interm Report Point of Interest  
%-----  
clc; clear; % clear screen and defined variables  
  
% Inputs  
DR = pi/180;  
  
%Position Inputs  
r1 = 32; r2 = 12; r3 = 30; r4 = 20; rca = 0;  
  
th1 = 0*DR; th2 = 315*DR; % conversion of degree to rad  
rlx = r1*cos(th1); rly = r1*sin(th1);  
  
th20 = 270*DR; th2f = 360*DR; th2 = [th20:th2f]*DR; % th2 array  
  
t0 = 0; dt = 0.1; tf = 5; t = [t0:dt:tf]  
  
N = (tf-t0)/dt + 1; % number of times to repeat loop for F.R.O.M. using time!  
figure;  
  
for i=1:N, % F.R.O.M. loop over all input th2  
  
    %Full Cycloidal FNC From Page 82  
  
    th2(i) = th20+(th2f-th20)*((t(i)/tf)-((1/(2*pi))*sin(2*pi*t(i)/tf)));  
    w2(i) = ((th2f - th20)/tf)*(1 - cos((2*pi*t(i))/tf));  
    alpha2(i) = ((2*pi*(th2f - th20))/(tf^2))*sin((2*pi*t(i))/tf);  
    beta2(i) = ((4*(pi^2)*(th2f - th20))/(tf^3))*cos((2*pi*t(i))/tf);  
  
    % Position analysis: Solving for theta4  
    E = 2*r4*(-r1*cos(th1) - r2*cos(th2(i)));  
    F = 2*r4*(r1*sin(th1) - r2*sin(th2(i)));  
    G = r1^2 + r2^2 - r3^2 + r4^2 + 2*r1*r2*cos(th1+th2(i));  
  
    topen = (-F - sqrt(E^2 + F^2 - G^2)) / (G-E); % open branch  
    th4(i) = 2*atan(topen); %The 2 Multiplier is important!  
  
    % Solve for th3, coupler point, transmission angle; open branch only  
    ax = -r2*cos(th2(i)); % Point A  
    ay = r2*sin(th2(i));  
    bx = r4*cos(th4(i)); % Point B  
    by = -r4*sin(th4(i));  
    cx = -r1*cos(th1);  
    cy = -r1*sin(th1);  
    th3(i) = atan2(by+ay+cy,bx+ax+cx); % theta3  
    mu(i) = abs(th4(i)-th3(i)); % transmission angle  
    pcx(i) = r4*cos(th4(i));  
  
end
```

```

pcy(i) = r4*sin(th4(i));

%Velocity Analysis Open Branch
a = r3*sin(th3(i)); %Define Variables
b = r4*sin(th4(i));
c = r2*w2(i)*sin(th2(i));

d = -r3*cos(th3(i));
e = r4*cos(th4(i));
f = r2*w2(i)*cos(th2(i));

%Solutions

w4(i) = ((a*f)-(d*c))/((a*e)-(d*b)); %Velocity Outputs
w3(i) = ((c*e)-(b*f))/((a*e)-(d*b));

%Accleration Analysis

C = -r2*alpha2(i)*sin(th2(i))-r2*((w2(i))^2)*cos(th2(i)) + r3*((w3(i))^2)*cos(th3(i)) ✓
+ r4*((w4(i))^2)*cos(th4(i));
F = -r2*alpha2(i)*cos(th2(i))+r2*((w2(i))^2)*sin(th2(i)) + r3*((w3(i))^2)*sin(th3(i)) ✓
- r4*((w4(i))^2)*sin(th4(i));

%Solving for Accleration Alphas

alpha4(i) = ((-a*F)-(-d*C))/((-a*-e)-(-b*-d));
alpha3(i) = ((C*-e)-(-b*F))/((-a*-e)-(-b*-d));

% Draw four-bar Mechanism to the screen, open branch only
x2 = [0 r2*cos(th2(i))]; % Coordinates for Link #2
y2 = [0 r2*sin(th2(i))];
x3 = [r2*cos(th2(i)) r1x+r4*cos(th4(i)) pcx(i)]; % Coordinates for link #3
y3 = [r2*sin(th2(i)) r1y+r4*sin(th4(i)) pcy(i)];
x4 = [r1x r1x+r4*cos(th4(i))]; % Coordinates for link #4
y4 = [r1y r1y+r4*sin(th4(i))];
plot(x2,y2,'b',x4,y4,'r');
patch(x3,y3,'g');
set(gca,'FontSize',19); xlabel('\itX (\itm)'); ylabel('\itY (\itm)');
axis('square'); axis([-r2 r1+r4 -(r1+r2+r4)/2 (r1+r2+r4)/2]); grid;
pause(1/3); %Pause to make animation visible
if i==1
    pause; % Clicking Enter Will Procede Animation
end
end

% Plots outside loop

figure;
plot(t,th2/DR); xlabel('time (sec)'); ylabel('theta2 (deg)'); grid; title('Theta2 V.S. ✓
Time');
axis([0 5 269 361]);

```

```
figure;
plot(t,w2); xlabel('time (sec)'); ylabel('angular velocity (rad/s)'); grid; title('Omega2
V.S. Time');
axis([0 5 0 0.6283]);

figure;
plot(t,alpha2); xlabel('time (sec)'); ylabel('angular acceleration (rad/s^2)'); grid;
title('Alpha V.S. Time');
axis([0 5 -4 4]);

figure;
plot(pcx,pcy); grid; axis('equal');
set(gca,'FontSize',19); title('Four-Bar Mechanism Coupler Curve');
xlabel('{\itX} ({}itm)'); ylabel('{\itY} ({}itm)');
```


Appendix F: Link Mass Calculations

Given: D = link diameter, L = link length, ρ = Density

$$\text{Volume of a cylinder} = V = \frac{\pi}{4} D^2 * L$$

$$\text{Mass} = M = \rho * V$$

Link 2:

$$V = \frac{\pi}{4} D^2 * L = \frac{\pi}{4} * 4^2 * 12 = 150.8 \text{ in}^3$$

$$M = 0.284 * 150.8 = 42.8 \text{ lb}_m$$

Link 3:

$$V = \frac{\pi}{4} D^2 * L = \frac{\pi}{4} * 4^2 * 30 = 377.0 \text{ in}^3$$

$$M = 0.284 * 377.0 = 107.1 \text{ lb}_m$$

Link 4:

$$V = \frac{\pi}{4} D^2 * L = \frac{\pi}{4} * 4^2 * 26 = 326.7 \text{ in}^3$$

$$M = 0.284 * 326.7 = 92.8 \text{ lb}_m$$

Appendix G: Rotational Moments of Inertia Calculations

Given: Mass = m , L = Link length

$$\text{Mass Moment of Inertia} = I_{GZ} = \frac{m * L^2}{12}$$

Link 2:

$$I_{GZ} = \frac{m * L^2}{12} = \frac{42.8 * 12^2}{12} = 513.6 \text{ lb}_m * \text{in}^2$$

Link 3:

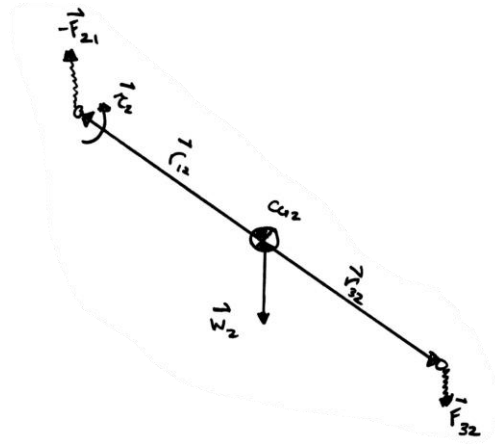
$$I_{GZ} = \frac{m * L^2}{12} = \frac{107.1 * 30^2}{12} = 8032.5 \text{ lb}_m * \text{in}^2$$

Link 4:

$$I_{GZ} = \frac{m * L^2}{12} = \frac{92.8 * 26^2}{12} = 5227.7 \text{ lb}_m * \text{in}^2$$

Appendix H: Link Details

Link 2



Moment Arm Length

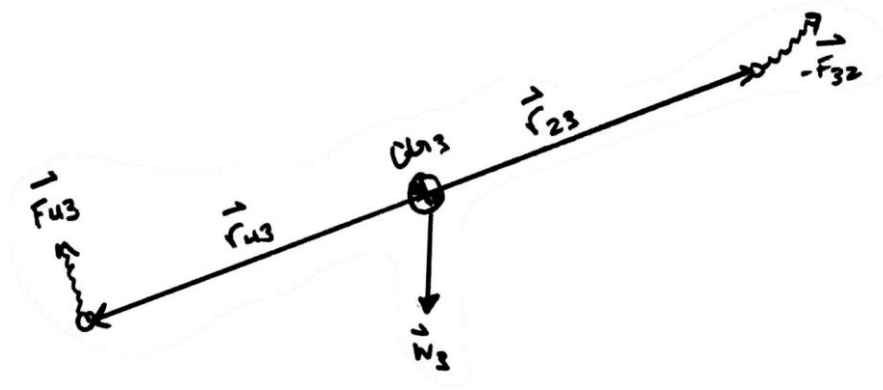
$$r_{12} = \begin{pmatrix} r_{12x} \\ r_{12y} \end{pmatrix} = \begin{pmatrix} -R_{g2} * \cos(\theta_2) \\ -R_{g2} * \sin(\theta_2) \end{pmatrix}$$

$$r_{32} = r_{12} + r_2 = \begin{pmatrix} r_{12x} + r_2 * \cos(\theta_2) \\ r_{12y} + r_2 * \sin(\theta_2) \end{pmatrix}$$

Acceleration of Center Point

$$A_{g2} = \begin{pmatrix} A_{g2x} \\ A_{g2y} \end{pmatrix} = \begin{pmatrix} -R_{g2} * \alpha_2 * \sin(\theta_2) - R_{g2} * \omega_2^2 * \cos(\theta_2) \\ R_{g2} * \alpha_2 * \cos(\theta_2) - R_{g2} * \omega_2^2 * \sin(\theta_2) \end{pmatrix}$$

Link 3



Moment Arm Length

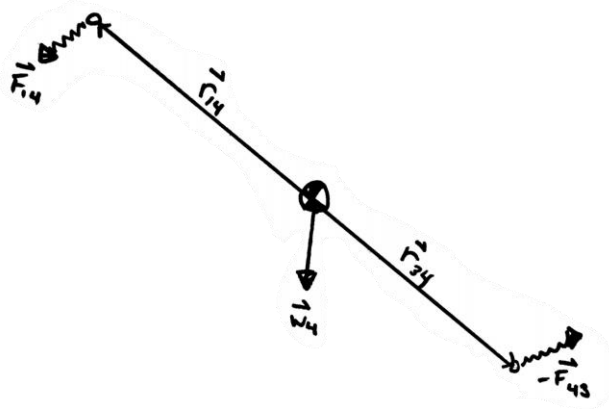
$$r_{23} = \begin{pmatrix} r_{23x} \\ r_{23y} \end{pmatrix} = \begin{pmatrix} -R_{g3} * \cos(\theta_3) \\ -R_{g3} * \sin(\theta_3) \end{pmatrix}$$

$$r_{43} = \begin{pmatrix} r_{43x} \\ r_{43y} \end{pmatrix} = \begin{pmatrix} r_{23x} + r_3 * \cos(\theta_3) \\ r_{23y} + r_3 * \sin(\theta_3) \end{pmatrix}$$

Acceleration of Center Point

$$A_{g3} = \begin{pmatrix} A_{g3x} \\ A_{g3y} \end{pmatrix}$$
$$= \begin{pmatrix} -r_2 * \alpha_2 * \sin(\theta_2) - r_2 * \omega_2^2 * \cos(\theta_2) - R_{g3} * \alpha_3 * \sin(\theta_3) - R_{g3} * \omega_3^2 * \cos(\theta_3) \\ r_2 * \alpha_2 * \cos(\theta_2) - r_2 * \omega_2^2 * \sin(\theta_2) + R_{g3} * \alpha_3 * \cos(\theta_3) - R_{g3} * \omega_3^2 * \sin(\theta_3) \end{pmatrix}$$

Link 4



Moment Arm Length

$$r_{14} = \begin{pmatrix} r_{14x} \\ r_{14y} \end{pmatrix} = \begin{pmatrix} -R_{g4} * \cos(\theta_4) \\ -R_{g4} * \sin(\theta_4) \end{pmatrix}$$

$$r_{34} = r_{14} + r_4 = \begin{pmatrix} r_{14x} + r_4 * \cos(\theta_4) \\ r_{14y} + r_4 * \sin(\theta_4) \end{pmatrix}$$

Acceleration of Center Point

$$A_{g4} = \begin{pmatrix} A_{g4x} \\ A_{g4y} \end{pmatrix} = \begin{pmatrix} -R_{g4} * \alpha_4 * \sin(\theta_4) - R_{g4} * \omega_4^2 * \cos(\theta_4) \\ R_{g4} * \alpha_4 * \cos(\theta_4) - R_{g4} * \omega_4^2 * \sin(\theta_4) \end{pmatrix}$$

Appendix I: Inverse Dynamics Mathematics

Link 2 Information

Newton's Second Law

$$\sum F_2 = F_{32} - F_{21} + W_2 = m_2 A_{G2}$$

Euler's Rotational Dynamics Equation

$$\sum M_{G2} = \tau_2 + r_{32} \times F_{32} - r_{12} \times F_{21} = I_{G2z} \alpha_2$$

XYZ Scalar Equations

$$\begin{aligned} F_{32x} - F_{21x} &= m_2 A_{G2x} \\ F_{32y} - F_{21y} &= m_2 (A_{G2y} + g) \\ \tau_2 + (r_{32x} F_{32y} - r_{32y} F_{32x}) - (r_{12x} F_{21y} - r_{12y} F_{21x}) &= I_{G2z} \alpha_2 \end{aligned}$$

Link 3 Information

Newton's Second Law

$$\sum F_3 = F_{43} - F_{32} + W_2 = m_3 A_{G3}$$

Euler's Rotational Dynamics Equation

$$\sum M_{G3} = r_{43} \times F_{43} - r_{23} \times F_{32} = I_{G3z} \alpha_3$$

XYZ Scalar Equations

$$\begin{aligned} F_{43x} - F_{32x} &= m_3 A_{G3x} \\ F_{43y} - F_{32y} - m_3 g &= m_3 A_{G3y} \\ (r_{43x} F_{43y} - r_{43y} F_{43x}) - (r_{23x} F_{32y} - r_{23y} F_{32x}) &= I_{G3z} \alpha_3 \end{aligned}$$

Link 4 Information

Newton's Second Law

$$\sum F_4 = F_{14} - F_{43} + W_4 = m_4 A_{G4}$$

Euler's Rotational Dynamics Equation

$$\sum M_{G4} = r_{14} \times F_{14} - r_{34} \times F_{43} = I_{G4z} \alpha_4$$

XYZ Scalar Equations

$$\begin{aligned}
F_{14x} - F_{43x} &= m_4 A_{G4x} \\
F_{14y} - F_{43y} &= m_4 (A_{G4y} + g) \\
(r_{14x} F_{14y} - r_{14y} F_{14x}) - (r_{34x} F_{43y} - r_{34y} F_{43x}) &= I_{G4z} \alpha_4
\end{aligned}$$

Matrix Solution

Known Matrices: [A] and {b}

Unknown Matrices: {v}

Table 11: Source: [1] Williams RL. Mechanism Kinematics & Dynamics. 2019th ed. LuLu; 2019

$$\begin{bmatrix}
-1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
r_{12Y} & -r_{12X} & -r_{32Y} & r_{32X} & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & r_{23Y} & -r_{23X} & -r_{43Y} & r_{43X} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & r_{34Y} & -r_{34X} & -r_{14Y} & r_{14X} & 0
\end{bmatrix}
\begin{Bmatrix}
F_{21X} \\
F_{21Y} \\
F_{32X} \\
F_{32Y} \\
F_{43X} \\
F_{43Y} \\
F_{14X} \\
F_{14Y} \\
\tau_2
\end{Bmatrix}
=
\begin{Bmatrix}
m_2 A_{G2X} \\
m_2 (A_{G2Y} + g) \\
I_{G2Z} \alpha_2 \\
m_3 A_{G3X} - F_{E3X} \\
m_3 (A_{G3Y} + g) - F_{E3Y} \\
I_{G3Z} \alpha_3 - r_{E3X} F_{E3Y} + r_{E3Y} F_{E3X} - M_{E3} \\
m_4 A_{G4X} - F_{E4X} \\
m_4 (A_{G4Y} + g) - F_{E4Y} \\
I_{G4Z} \alpha_4 - r_{E4X} F_{E4Y} + r_{E4Y} F_{E4X} - M_{E4}
\end{Bmatrix}$$

$$[A]\{v\} = \{b\}$$

Using Matrix Inversion, the unknown matrix can be solved.

$$\{V\} = [A]^{-1} * \{b\}$$

Appendix J: MATLAB Code Inverse Dynamics

11/17/19 6:30 PM D:\ME 3011 Fin...\LandingGearFinalVer16.m 1 of 6

```
%-----  
% Jonathan Bowman      Four Bar Landing Gear Mechanism  
% ME 3011 Final Report  Version 16  
%-----  
clc; clear; close all;% clear screen and defined variables  
  
%Universal Knowns  
DR = pi/180;  
  
%Kinematic Knowns  
r1 = 32; r2 = 12; r3 = 30; r4 = 26; rca = 0;  
  
th1 = 0*DR;  
r1x = -r1*cos(th1);  
r1y = r1*sin(th1);  
  
%Inverse Dynamics Knowns  
%Mass  
mass2 = 42.8/32.174; %Units Slugs  
mass3 = 107.1/32.174;  
mass4 = 92.8/32.174;  
  
%Mass Moment of Inertia  
Ig2z = 513.6/32.174; % Units: slug*in^2  
Ig3z = 8032.5/32.174;  
Ig4z = 5227.7/32.174;  
  
%Given that the Weight Acts at the Center of the Links  
Rg2 = r2/2;  
Rg3 = r3/2;  
Rg4 = r4/2;  
  
%Loop Determining Factors  
th20 = 236.09*DR; th2f = 360*DR; th2 = [th20:th2f]*DR; % th2 array  
  
t0 = 0; dt = 0.1; tf = 5; t = [t0:dt:tf];  
  
N = (tf-t0)/dt + 1; % number of times to repeat loop for F.R.O.M. using time t  
figure;  
  
for i=1:N, % F.R.O.M. Loop over all input for Time  
  
    %Full Cycloidal FNC From Page 82 in Notebook  
  
    th2(i) = th20+(th2f-th20)*((t(i)/tf)-((1/(2*pi))*sin(2*pi*t(i)/tf)));  
    w2(i) = ((th2f - th20)/tf)*(1 - cos((2*pi*t(i))/tf));  
    alpha2(i) = ((2*pi*(th2f - th20))/(tf^2))*sin((2*pi*t(i))/tf);  
    beta2(i) = ((4*(pi^2)*(th2f - th20))/(tf^3))*cos((2*pi*t(i))/tf);  
  
    % Position analysis: Solving for theta4
```

```

E = -2*r4*(r1*cos(th1) + r2*cos(th2(1)));
F = -2*r4*(r1*sin(th1) + r2*sin(th2(1)));
G = r1^2 + r2^2 - r3^2 + r4^2 + 2*r1*r2*cos(th1+th2(1));

topen = (-F - sqrt(E^2 + F^2 - G^2)) / (G-E); % Solution for T (Open Branch)
th4(1) = 2*atan(topen); %Keep 2 Mult. because tan is double valued

% Solution for th3, coupler point, and transmission angle (Open Branch)
ax = r2*cos(th2(1)); % Point A
ay = -r2*sin(th2(1));
bx = r1*cos(th1); % Point B
by = -r1*sin(th1);
cx = -r4*cos(th4(1)); %Point C
cy = r4*sin(th4(1));
th3(1) = atan2(by+ay+cy,bx+ax+cx); % theta3

%Coupler Point Position
mu(1) = abs(th4(1)-th3(1)); % transmission angle
pcx(1) = r4*cos(th4(1));
pcy(1) = r4*sin(th4(1));

%Velocity Analysis (Open Branch)
a = r3*sin(th3(1)); %Define Variables
b = r4*sin(th4(1));
c = r2*w2(1)*sin(th2(1));

d = -r3*cos(th3(1));
e = r4*cos(th4(1));
f = r2*w2(1)*cos(th2(1));

%Solutions
w4(1) = ((a*f)-(d*c))/((a*e)-(d*b)); %Angular Velocity Output for 4
w3(1) = ((c*e)-(b*f))/((a*e)-(d*b)); %Angular Velocity Output for 3

%Acceleration Analysis

C = -r2*alpha2(1)*sin(th2(1))-r2*((w2(1))^2)*cos(th2(1)) + r3*((w3(1))^2)*cos(th3(1)) %
+ r4*((w4(1))^2)*cos(th4(1));
F = -r2*alpha2(1)*cos(th2(1))+r2*((w2(1))^2)*sin(th2(1)) + r3*((w3(1))^2)*sin(th3(1)) %
- r4*((w4(1))^2)*sin(th4(1));

%Solving for Acceleration Alphas

alpha4(1) = ((-a*F)-(-d*C))/((-a*e)-(-b*d)); %Angular Acceleration Output for 4
alpha3(1) = ((C*e)-(-b*F))/((-a*e)-(-b*d)); %Angular Acceleration Output for 3

%Inverse Dynamics
%Link 2 Details
r12x(1) = -lg2*cos(th2(1));
r12y(1) = -lg2*sin(th2(1));

```

```

r32x(1) = r12x(1) + r2*cos(th2(1));
r32y(1) = r12y(1) + r2*sin(th2(1));

Ag2x(1) = -Rg2*alpha2(1)*sin(th2(1))-Rg2*(w2(1)^2)*cos(th2(1));
Ag2y(1) = Rg2*alpha2(1)*cos(th2(1))-Rg2*(w2(1)^2)*sin(th2(1));

%Link 3 Details
r23x(1) = Rg3*cos(th3(1));
r23y(1) = -Rg3*sin(th3(1));

r43x(1) = r23x(1) - r3*cos(th3(1));
r43y(1) = r23y(1) + r3*sin(th3(1));

Ag3x(1) = -r2*alpha2(1)*sin(th2(1)) - r2*(w2(1)^2)*cos(th2(1)) - Rg3*alpha3(1)*sin(th3(1)) - Rg3*(w3(1)^2)*cos(th3(1));
Ag3y(1) = r2*alpha2(1)*cos(th2(1)) - r2*(w2(1)^2)*sin(th2(1)) + Rg3*alpha3(1)*cos(th3(1)) - Rg3*(w3(1)^2)*sin(th3(1));

%Link 4 Details
r14x(1) = -Rg4*cos(th4(1));
r14y(1) = -Rg4*sin(th4(1));

r34x(1) = r14x(1) + r4*cos(th4(1));
r34y(1) = r14y(1) + r4*sin(th4(1));

Ag4x(1) = -Rg4*alpha4(1)*sin(th4(1)) - Rg4*(w4(1)^2)*cos(th4(1));
Ag4y(1) = Rg4*alpha4(1)*cos(th4(1)) - Rg4*(w4(1)^2)*sin(th4(1));

%Matrix Variables
A1 = (r12y(1)/12);
A2 = (-r12x(1)/12);
A3 = (-r32y(1)/12);
A4 = (-r32y(1)/12);
A5 = (r23y(1)/12);
A6 = (-r23x(1)/12);
A7 = (-r43y(1)/12);
A8 = (r43x(1)/12);
A9 = (r34y(1)/12);
A10 = (-r34x(1)/12);
A11 = (-r14y(1)/12);
A12 = r14x(1);

AMatrix = [ -1 0 1 0 0 0 0 0 0;
            0 -1 0 1 0 0 0 0 0;
            A1 A2 A3 A4 0 0 0 0 1;
            0 0 -1 0 1 0 0 0 0;
            0 0 0 -1 0 1 0 0 0;
            0 0 A5 A6 A7 A8 0 0 0;

```

```

0 0 0 0 -1 0 1 0 0;
0 0 0 0 0 -1 0 1 0;
0 0 0 0 A9 A10 A11 A12 0;];

%Define B Matrix Variables

B1(i) = (mass2*(Ag2x(i)+0.08333333)); %Change from in/s^2 to ft/s^2
B2(i) = (mass2*((Ag2y(i)+0.08333333)+32.2)); %Change g to 32.2 ft/s
B3(i) = ((Ig2x/144)*alpha2(i));
B4(i) = (mass3*(Ag3x(i)+0.08333333));
B5(i) = (mass3*((Ag3y(i)+0.08333333)+32.2));
B6(i) = ((Ig3x/144)*alpha3(i));
B7(i) = (mass4*(Ag4x(i)+0.08333333));
B8(i) = (mass4*((Ag4y(i)+0.08333333)+32.2));
B9(i) = ((Ig4x/144)*alpha4(i));

BMatrix = [B1;B2;B3;B4;B5;B6;B7;B8;B9];

%Solve the Matrix

XMatrix = linsolve(AMatrix,BMatrix);

%Take Values From Matrix Solution

F21x(i) = XMatrix(1,i);
F21y(i) = XMatrix(2,i);
F32x(i) = XMatrix(3,i);
F32y(i) = XMatrix(4,i);
F43x(i) = XMatrix(5,i);
F43y(i) = XMatrix(6,i);
F14x(i) = XMatrix(7,i);
F14y(i) = XMatrix(8,i);
Tau(i) = XMatrix(9,i);

%Shaking Force

Fsx(i) = F21x(i) - F14x(i);
Fsy(i) = F21y(i) - F14y(i);

% Draw the Landing Gear Mechanism to the screen

x2 = [0 r2*cos(th2(i))]; % Coordinates for Link #2
y2 = [0 r2*sin(th2(i))];
x3 = [r2*cos(th2(i)) rlx+r4*cos(th4(i)) ]; % Coordinates for link #3
y3 = [r2*sin(th2(i)) rly+r4*sin(th4(i))];
x4 = [rlx rlx+r4*cos(th4(i))]; % Coordinates for link #4
y4 = [rly rly+r4*sin(th4(i))];
plot(x2,y2,'b',x4,y4,'r');
title('Landing Gear Animation');
hold on; % Shows time change on plot
patch(x3,y3,'g');
```

```

set(gca,'FontSize',19); xlabel('\itX (in.)'); ylabel('\itY (in.)');
axis('square'); grid;
pause(1/20); %Pause to make animation visible (Remember to find good Timing!)
if i==1
    pause; % Click Enter To Proceede the Landing Gear Animation
end
end

% Plots for Cycle Functions

figure;
subplot(3,1,1);
plot(t,th2/DR,'r');
xlabel('Time, T (sec)');
ylabel('\it\theta_2 (deg)'); grid;
title('Input Angle V.S. Time');

subplot(3,1,2);
plot(t,w2,'b');
xlabel('Time, T (sec)');
ylabel('\it\omega_2 (rad/s)'); grid;
title('Input Angular Velocity V.S. Time');

subplot(3,1,3);
plot(t,alpha2,'g');
xlabel('Time, T (sec)');
ylabel('\it\alpha_2 (rad/s^2)'); grid;
title('Input Angular Acceleration V.S. Time');

figure;
plot(pcx,pcy); grid; axis('equal');
set(gca,'FontSize',19);
title('Four-Bar Mechanism Coupler Curve');
xlabel('P_c_x (in.) '); ylabel('P_c_y (in.) ');

figure;
subplot(2,1,1);
plot(th2/DR,th3/DR,'r');
ylabel('\it\theta_3 (deg)'); grid;
title('\it\theta_3 s \it\theta_4 V.S. \it\theta_2');

subplot(2,1,2);
plot(th2/DR,th4/DR+360,'b');
xlabel('\it\theta_2 (deg)');
ylabel('\it\theta_4 (deg)'); grid;

figure;
subplot(2,1,1);
plot(th2/DR,w3,'r');
ylabel('\it\omega_3 (rad/s)'); grid;
title('\it\omega_3 s \it\omega_4 V.S. \it\theta_2');

```

```

subplot(2,1,2);
plot(th2/DR,w4,'b');
xlabel('\it\theta_2 (deg)');
ylabel('\it\omega_4 (rad/s)'); grid;

figure;
subplot(2,1,1);
plot(th2/DR,alpha3,'r');
ylabel('\it\alpha_3 (rad/s^2)'); grid;
title('\it\alpha_3 & \it\alpha_4 V.S. \it\theta_2');

subplot(2,1,2);
plot(th2/DR,alpha4,'b');
xlabel('\it\theta_2 (deg)');
ylabel('\it\alpha_4 (rad/s^2)'); grid;

%Torque Plot
TauAVG = mean(Tau);
TauRMS = norm(Tau)/sqrt(i+1);
wuns = ones(1, i);

figure;
plot(th2/DR, Tau, 'b', th2/DR, TauAVG*wuns, 'r', th2/DR, TauRMS*wuns, 'g'); grid;
title('Torque vs. \it\theta_2');
set(gca,'FontSize',19); legend('Tau', 'AVG Tau', 'RMS Tau');
xlabel('\it\theta_2 (deg)'); ylabel('Torque (lb_f*ft)');

%Shaking Force

figure;
subplot(2,1,1);
plot(th2/DR,Fax,'r');
ylabel('Shaking Force, Fax (lb_f)'); grid;
title('Shaking Force vs. \it\theta_2');

subplot(2,1,2);
plot(th2/DR,Fay,'b');
xlabel('\it\theta_2 (deg)');
ylabel('Shaking Force, Fay (lb_f)'); grid;

```

Appendix K: MATLAB Matrix Solver

11/17/19 11:55 PM \\home.ohio.edu\home\j...\MatrixSolver.m 1 of 1

```
%-----  
% Jonathan Bowman Matrix Solver  
% ME 3011 - Landing Gear Project - Matrix Snapshot  
%-----  
  
clc; clear;  
  
%LHS Matrix (Knowns)  
A = [ -1 0 1 0 0 0 0 0 0;  
      0 -1 0 1 0 0 0 0 0;  
      -4.24 4.24 4.245 4.245 0 0 0 0 1;  
      0 0 -1 0 1 0 0 0 0;  
      0 0 0 -1 0 1 0 0 0;  
      0 0 6.648 -13.446 6.441 -13.448 0 0 0;  
      0 0 0 0 -1 0 1 0 0;  
      0 0 0 0 0 -1 0 1 0;  
      0 0 0 0 -11.0366 -6.8696 -11.037 -6.8696 0;]  
  
%RHS (Knowns)  
B = [-0.42769; 43.05; -0.0255297146; -2.36802; 107.74; -0.22365; -0.69769; 92.92; ✓  
-0.20422833];  
  
%Solve for Unknowns - Gaussian Elimination  
X = linsolve(A,B)  
  
%Solve for Unknowns - Matrix Inversion  
  
v = inv(A)*B;  
v %Shown Matrix
```

A =

Columns 1 through 8

-1.0000	0	1.0000	0	0	0	0	0
0	-1.0000	0	1.0000	0	0	0	0
-4.2400	4.2400	4.2450	4.2450	0	0	0	0
0	0	-1.0000	0	1.0000	0	0	0
0	0	0	-1.0000	0	1.0000	0	0
0	0	6.6480	-13.4460	6.4410	-13.4480	0	0
0	0	0	0	-1.0000	0	1.0000	0
0	0	0	0	0	-1.0000	0	1.0000
0	0	0	0	-11.0366	-6.8696	-11.0370	-6.8696

Column 9

0
0
1.0000
0
0
0
0
0
0

X =

-45.1402
-119.6602
-45.5679
-76.6102
-47.9359
31.1298
-48.6336
124.0498
834.5850

v =

-45.1402
-119.6602
-45.5679
-76.6102
-47.9359
31.1298
-48.6336
124.0498

834.5850

>>