UNIVERSITY OF THE
WITWATERSRAND,
JOHANNESBURG

SCHOOL OF MECHANICAL,
INDUSTRIAL & AERONAUTICAL
ENGINEERING

MECHATRONICS II: MECN4029A – PROJECT ASSIGNEMNT

MAY 2024

MECHATRONIC SYSTEMS DESIGN: ANALYSIS AND CONTROL

PROJECT B: SOLAR TRACKER

GROUP 10

AUTHORS:

1.	Mari Norwie	– 2273452
2.	Kashaka Sithole	– 2309398
3.	Ipeleng Tshabalala	– 2362449
4.	Christy Mkhari	– 2481454

Lecturer: Dr A. Panday

AI DECLARATION

Disclosure – Use of Artificial-Intelligence (AI) Generated Content 2024 V2

Students must acknowledge all use of AI.

Select all applicable statements and complete the sections fully. Delete all statements that are not applicable.

1. Disclosure: No AI use

☒ I acknowledge that no AI tools/technologies (Grammarly, ChatGPT, Bard, Quillbot, OpenAI etc) were used in the completion of this assessment.

☒ ***I declare that the disclosure is complete and truthful.***

Student number(s): 2273452, 2309398, 2362449, 2481454

Course code: MECN4029A

Date: 12/05/2024

Table of Contents

List of Figures

List of Tables

Nomenclature

m - Mass measured in kilograms (kg)

w – Width measured in metres (m)

l – Length measured in metres (m)

d - Depth measured in metres (m)

A – Area measured in square metres (m^2)

β – Elevation angle measured in radians (rad)

K_d – Damping constant measured in N.m/rad.s^{-1}

J – Inertia measured kg. m^2

K_f– Back EMF constant measured V/rad. s^{-1}

K_t – Torque constant measured in N.m/A

L – Inductance measured in Henries [H]

R – Resistance measured in ohms (Ω)

K_g – Gear ratio

h – hour angle measured in degrees

n – day of the year (where n =1 on January 1st)

1. INTRODUCTION

1.1. BACKGROUND

South Africa national electricity constitutes of 85% of the coal generated power as it is the one of the most abundant and affordable sources of fuel. [1] Coal generated electricity dominates the countries system load as shown on Figure 1 below. The use of coal in the generation of the national power brings about challenges that go against the emerging ideal of green energy (which promotes the use of environmentally friendly renewable energy). Solar energy as a renewable energy is a steadily growing power generation field as it conforms to green energy and global climate change standards.

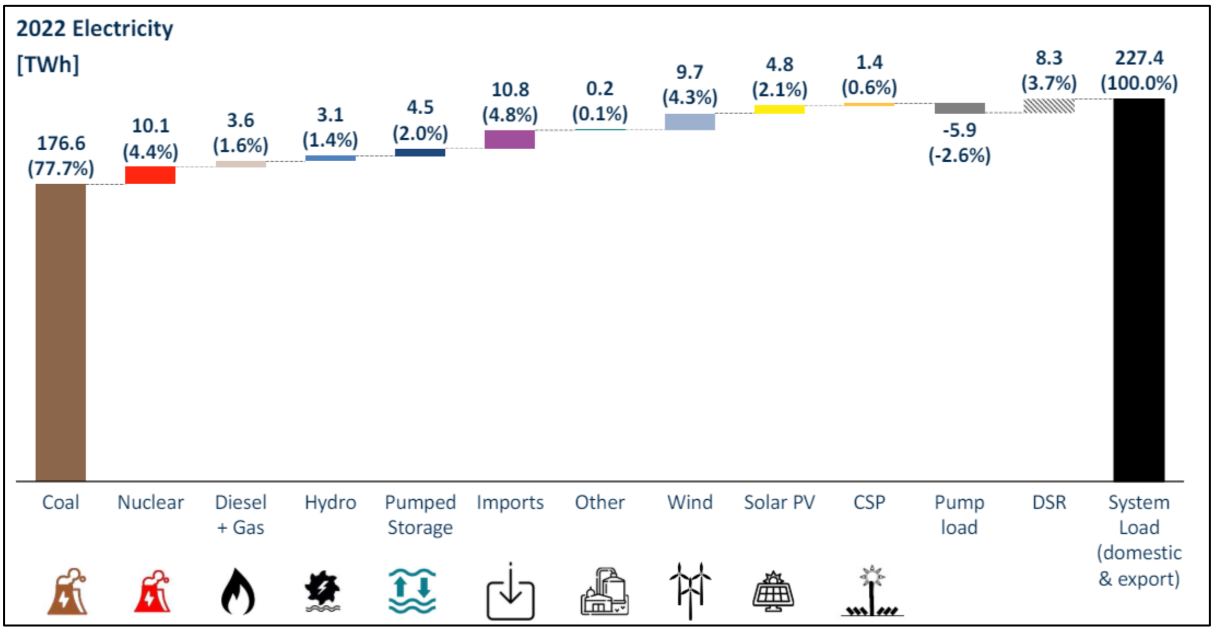


Figure 1: South Africa's 2022 nation grid power generation distribution [2]

The use of solar trackers is one way to improve the efficiency of the solar power generation. Solar trackers are mechanisms that adjust the solar panel orientation and position based on the angles of the sun's rays. The solar trackers are a means of improving the solar energy absorbed by the solar panels for use in powering of other systems. The ability of the solar tracker technology relies on it being efficient in adjusting the solar panels so that they can meet the desired angle of incidence of the sunlight at various times of the day during different seasons [3].

Solar trackers can move based on single or multiple axis. The axis types determine the type and range of motion of the solar system. The single axis tracker rotates the panels in one direction/ motion, whereas multiple axes trackers can displace the panels around 2 or more axes independently or simultaneously. The single axis type of solar trackers is further characterized into horizontal single axis trackers (HSATs) and vertical single axis trackers (VSATs). The HSATs are able to rotate in the lateral directions (left to right) while VSAT rotate the panels up and down [4]. Both types are efficient in their applications yet the application and efficiency of the type of single-axis tracker depends on the location where the solar panel is used.

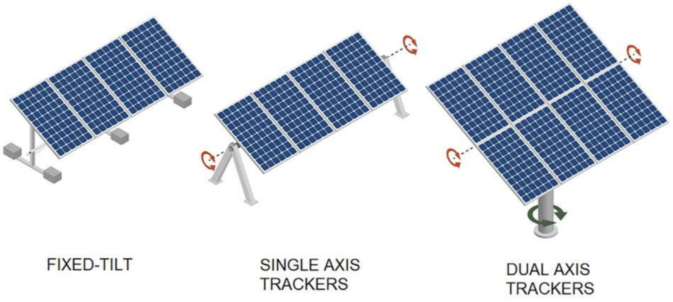


Figure 2: Types of solar panels with solar trackers of different ranges of motion [5]

1.2. PROBLEM SCENARIO

Due to the prevalence of problems in reliable power generation in the Braamfontein area, the University of the Witwatersrand (located in Johannesburg, South Africa) has been experiencing challenges with supplying the entire Braamfontein campus with constant power. The students in the school of Mechanical, Industrial and Aeronautical engineering at this critical time have found the greater need to utilise the school's computer laboratories. This has placed greater financial strain on the school in powering the labs with commercial generators. To reduce this strain, it has been proposed that the school use of actuated solar panels to generate an independent power supply for the MIA computer labs.

It is therefore the aim of this project to design a single axis solar tracker that will determine and adjust to the sun's optimal position to achieve high energy efficiency. The computer lab that the system is intended to power consists of 30 computer and monitor set-ups. Lectures that take place in the computer lab involve the use of simulation software which require a great deal of processing power and therefore the power consumption of each set-up can be as high as 300 W, meaning the total power consumption during a class is approximately 9 kW.

The students were able to purchase twenty-five 400 W solar panels, with the budget allocated to them. The panels therefore have a maximum power production of 10 kW and are expected to consistently produce the demands of the lab and an additional 500 W as safety to prevent overloading the solar system. Therefore, the system is expected to produce 9.5 kW of power between 10AM and 2PM when the lectures usually take place. Assuming the power production of the solar panel can be approximated by applying the cosine rules to the incidence angle using the following formula.

$$P = P_{max} \cos \emptyset$$

Therefore, to meet the power demands of the lab the system must maintain as small an incidence angle as possible and should not exceed an angle of 18.2°.

1.3. CONTROLLER FOR THE VERTICAL SINGLE AXIS SOLAR TRACKER

To solve the power generation problem experienced in the University of the Witwatersrand's engineering students a single axis solar tracker will be designed to better control the solar panels purchased. The solar tracker will be a vertical single axis solar tracker which will thrive in Johannesburg as the location is in the high veld, it is exposed to high levels of sunlight. The single axis tracker must be able to track the sun's movement, with minimal steady state error.

By using the vertical single axis solar tracker , the solar panel will adjust itself to monitor the sun's position from east to west. The east west tracking of the sun's position ensures the maximisation of solar energy absorption. This means that the tracker will not be able to track the sun's seasonal trajectory.

An integral part of this design is the use of the potentiometer to track the sun's position relative to the solar panel. The use of the potentiometer is a simple, energy efficient and reliable design consideration. In the solar tracker, the potentiometer is attached to the rotating axis of the solar panel so that when the solar panel moves to meet the sun's incident rays, the wiper (a component of the potentiometer) changes position, producing an output voltage detected by the controller system. This helps adjust the motor and panel to keep the entire system aligned with the sun.

2. LITERATURE

The literature produced in this section is used to guide the design and the performance of the solar controller as it considers the location of use the solar panel and other important parameters.

2.1. SOLAR INFORMATION FOR JOHANNESBURG, SOUTH AFRICA

As stated in the problem scenario, the location for which the solar tracker is to be developed is Wits University, in Johannesburg. It is therefore important that the solar patterns of the Johannesburg area. The location of interest is at a longitude and latitude of 28.0473051˚, and -26.2041028˚ respectively. [6]

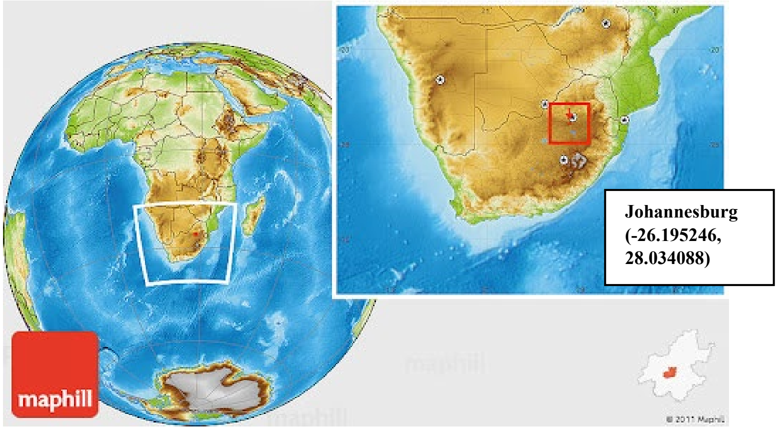


Figure 3: Global with South Africa, Johannesburg co-ordinates [7]

South Africa's winter and summer solstice(s) are commonly on June 21st and December 21st [8]. The summer solstice is significant in that it is the day which produces the greatest sunlight [9], hence the day that produces the most solar energy. The average time of sunlight per day in Johannesburg is 8 hours 49 minutes [10]. The sun rises in the east and sets in the west.

The sun's path in Johannesburg is shown in Figure 4 below. Figure 4 represents the path that the sun is seen traversing in Johannesburg. The trajectory was presenting by plotting the sun's azimuth (motion from east to west) and elevation (motion from north to south) angles for a particular day. In the Figure below, the sun's trajectory is plotted for the days of the winter and summer solstices in green and blue respectively. The sunset time is indicated by the point of intersection of the curves and the 'globe' on the left-hand side, while the right-hand side intersection represents the sunrise time. The azimuth angles are indicated on the outside of the globe diagram.[11]

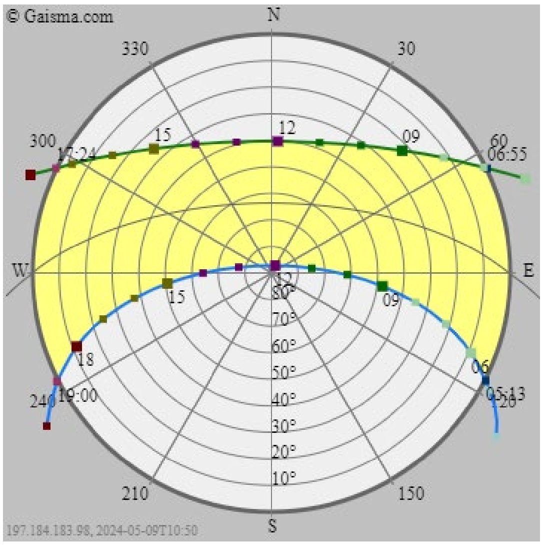


Figure 4: Johannesburg sun path chart [11]

During the winter solstice (shown in green), the sun rises at azimuth and elevation angles of 64˚ and 0˚ respectively. The sunrise and the sunset times are 06:55 and 17:24 respectively. At noon, when the sun is expected to be at its peak, the elevation angle is approximately 40˚, while azimuth angle is 2˚. During the summer solstice (shown in blue), the sun rises at azimuth and elevation angles of 111˚ and 8˚ respectively. The sunrise and the sunset times are 05:13 and 19:00 respectively. At noon, when the sun is expected to be at its peak, the elevation angle is approximately 86˚, while azimuth angle is 27˚.

2.2. Potentiometer

The design of the vertical single axis tracker used to collect and supply the School of Mechanical, Industrial and Aeronautical Engineering's computer labs will make use of a potentiometer. This section will introduce relevant information of the potentiometer device and its use in the solar tracker.

A potentiometer (Figure 5) is a type of three port variable resistor, that operates as means of dividing variable resistance but setting the resistive divider ratio. This device comes in various types of rotary and linear designs. A single-turn potentiometer rotates around 270˚ and it is commonly utilised where reliable precision is provided by a single turn. A multi-turn potentiometer allows for several revolutions (between 5 to 20) to improve precision and accuracy. It is built using a helically shaped a worm-gear mechanism. Dual-gang potentiometers are a combination of two potentiometers on the same housing, allowing for parallel control of two pathways. It is commonly used in radio sound control or applications that require parallel adjustment of two channels with different parameters. A concentric potentiometer contains two potentiometers with individually adjustable concentric shafts, allowing for two controls in one unit and it is used in older car radios [12].

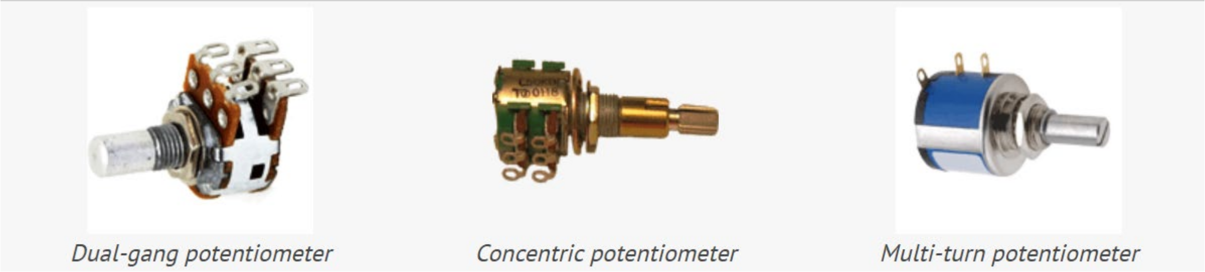


Figure 5: Rotary Potentiometers [12]

For the application on the solar tracker, the single turn potentiometer is sufficient to meet the control required for the single axis solar tracker. The resolution for this potentiometer is 270˚, which is inclusive of the solar tracker's range of motion. The solar tracker will enable the solar panel to track the sun's path for the purpose of achieving optimal angle for solar energy absorption. This potentiometer has a simple design will prove to be desirable in terms of cost, installation, and replacement.

The potentiometer in the solar tracker will act as the device that will detect the angular position of the sun relative to the solar panel. The potentiometer has a component called a wiper which moves along the potentiometer's resistive port converting the resistance to an output voltage. The sun's position will be sent to the rest of the system as a voltage signal which will then be interpreted by the solar tracker controller to make the relevant adjustments.

3. PERFORMANCE SPECIFICATIONS

Performance specifications are the requirements that have be set to provide a baseline as well as the criteria by which judgement of the system's success will be based. For the solar tracker closed system, the designed system is expected to have the following specifications:

- The rise time for the closed loop system is to be less than 1 second.
- The overshoot of the closed loop system should be less than 5% of the final output values.
- The system should have a settling time no more than 10 seconds.
- The system should not achieve a steady state peak higher than 3.3
- The solar tracker is to operate in a longitude and latitude of 28.0473051˚, and -26.2041028˚.
- The tracker should operate within the azimuth range of 0˚ to 140˚
- The tracker should operate within the elevation range of 0˚ to 120˚
- The system is set to have minimal (approximately ±0.4) to no steady state error.
- The solar panel is expected to operate from 6 am to 5 pm daily with sunlight time being 8 hours 49 minutes.
- The solar panel is expected to absorb 10 kW worth of solar energy.
- The alignment error between the solar panel and the sun's position should be less than 5˚.
- The solar tracker should respond to sun position changes in less than 10 seconds.
- The solar tracker's settling time should be in less than 10 seconds.
- The solar panel system must not consume power more than 0.5 kW of power.
- The solar panel system must have a minimum lifespan of 5-10 years.
- The solar tracker controller must log data every 800 seconds.
- The solar panel must operate between the temperature ranges of 10˚ to 40˚.

4. SYSTEM PARAMETERS

4.1. PHYSICAL & MATHEMATICAL MODEL

The figure below illustrates the physical representation of the solar panel system. The J represents the solar panel's inertia, T the torque and the theta angle the representation of the rotation of the solar panel to ensure that the solar is orientated in a position that the optimal sun's position is achieved.

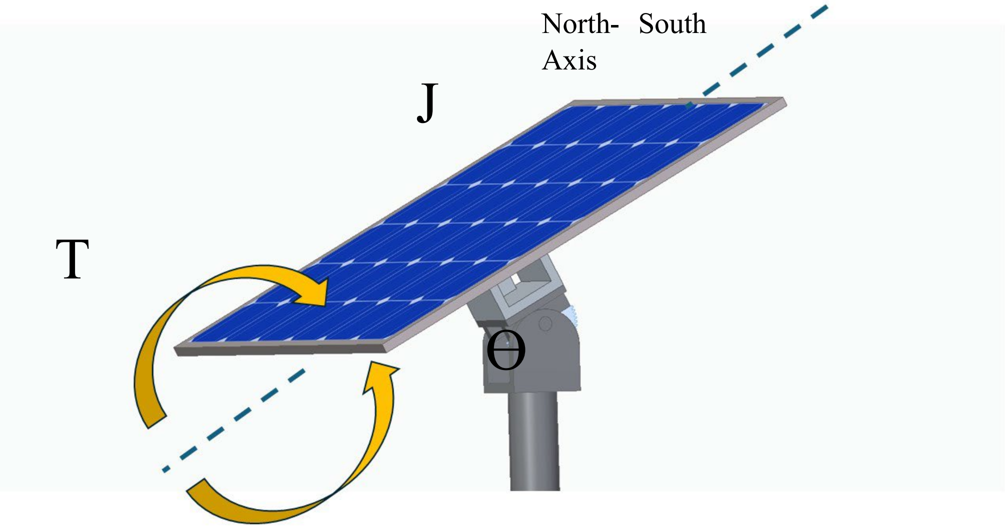


Figure 6: Representation of the solar panel

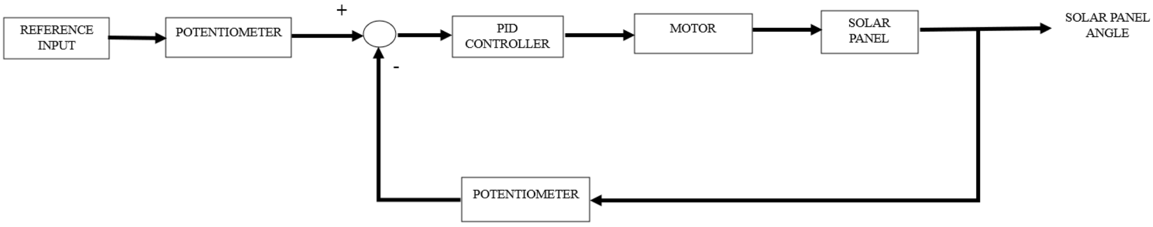


Figure 7: Block diagram of the solar panel system.

4.1.1. Time Domain

The solar tracker system is modelled based on the numerical calculations below. The equations of motion for the solar tracker are sourced from the MathWorks website [13].

The Solar Panel's equations of motion are shown below[13] :

$$J\ddot{\theta} + b\dot{\theta} = T$$

$$T = K_g K_t i$$

$$J\ddot{\theta} + b\dot{\theta} = K_g K_t i$$

$$\ddot{\theta} = \frac{1}{J}(K_gK_ti - b\dot{\theta})$$

Motor equations of motion [13]:

$$L\frac{di}{dt} + Ri = V - e$$

$$e = \ K_gK_f\dot{\theta}$$

$$L\frac{di}{dt} + Ri = V - K_gK_f\dot{\theta}$$

$$\frac{di}{dt} = \frac{1}{L}(V - K_gK_f\dot{\theta} - Ri)$$

4.1.2. S-domain

$$(Js^2 + bs)\theta(s) = \ K_gK_tI(s)$$

$$LsI(s) + RI(s) = \ V(s) - \ K_gK_fs\theta(s)$$

$$I(s)[Ls + R] = V(s) - \ K_gK_fs\theta(s)$$

$$I(s) = \frac{V(s) - K_gK_fs\theta(s)}{Ls + R}$$

4.1.3. Linearisation

Linearisation is a mathematical method used to analyse a system with reference to a specific equilibrium point. It makes it easier to analyse the system, but it is important to analyse the difference between a linear and nonlinear system to deduce any major discrepancies that may affect analysing a linearised model. The equation mentioned in the frequency domain classify the inertial term (J) as the non-linear term and due to the model requiring a time invariant analysis, this term must be appropriately linearised.

$$J\ddot{\theta} + b\dot{\theta} = K_gK_ti$$

$$\therefore f = \ J\ddot{\theta} + b\dot{\theta} - K_gK_ti$$

$$where\, J = \ \frac{m}{12}(l^2cos^2(\beta) + d^2sin^2(\beta) + w^2)$$

$$\therefore f = (\frac{m}{12}(l^2cos^2(\beta) + d^2s\imath n^2(\beta) + w^2))\ddot{\theta} + b\dot{\theta} - K_gK_ti$$

Using the Taylor Series Approach:

$$\frac{df}{d\beta} = \ \ddot{\theta}_0\left[\frac{m}{12}l^2 \cdot -2\cos(\beta)\sin(\beta) + \frac{m}{12}d^2 \cdot 2\cos(\beta)\sin(\beta)\right]|_{\beta_0}\Delta\beta$$

$$\frac{df}{d\beta} = \ \frac{m}{6}\ddot{\theta}_0\cos(\beta_0)\sin\,(\beta_0)(d^2 - l^2)\Delta\beta$$

$$\frac{df}{d\ddot{\theta}} = \frac{m}{12}(l^2cos^2(\beta_0) + d^2sin^2(\beta_0) + w^2)|_{\ddot{\theta}_0}\Delta\ddot{\theta}$$

$$\frac{df}{d\dot{\theta}} = k_d|_{\dot{\theta}_0}\Delta\dot{\theta}$$

$$\frac{df}{di} = -k_gk_t|_{i_0}\Delta i$$

With the initial conditions of $\beta_0 = \frac{pi}{4};\ \ddot{\theta}_0 = 0;\ \dot{\theta}_0 = 0; i = 0$

The zero function is determined to be: $f_0 = 0$

$$\therefore f = f_0 + \frac{m}{6}\ddot{\theta}_0\cos(\beta_0)\sin(\beta_0)\,(d^2 - l^2)\Delta\beta + \frac{m}{12}(l^2cos^2(\beta_0) + d^2sin^2(\beta_0) + w^2)\Delta\ddot{\theta} + k_d\Delta\dot{\theta} - k_gk_t\Delta i = 0$$

$$f_0 + \frac{m}{6}\ddot{\theta}_0\cos(\beta_0)\sin(\beta_0)\,(d^2 - l^2)\Delta\beta + \frac{m}{12}(l^2cos^2(\beta_0) + d^2sin^2(\beta_0) + w^2)\Delta\ddot{\theta} = -k_d\Delta\dot{\theta} + k_gk_t\Delta i$$

By applying the initial conditions, the linearised function will reduce to:

$$(\frac{m}{12}(l^2cos^2(\beta_0) + d^2sin^2(\beta_0) + w^2)\ddot{\theta} = -k_d\dot{\theta} + k_gk_ti$$

There is no change as the dimensions and mass of the solar tracker thus suggesting that the only changing variable is the sun's position. The other changes are zero and the inertial term is reduced to:

$$J_0\ddot{\theta} = -k_d\dot{\theta} + k_gk_ti$$

4.2. ASSUMPTIONS AND APPROXIMATIONS

For the mathematical modelling of the solar tracker system (VSAT) to track the sun's position from east to west the assumptions and approximations were made.

Assumptions and approximations:

The declination angle (δ) is -23.45° in mid-winter (of which June 21st is the winter solstice) while it is 23.45° in mid-summer (of which December 21st is the summer solstice) [8]

The average time of sunlight per day in Johannesburg is 8 hours 49 minutes (equivalent to 8.81 hours) [**10**]

The trajectory of the sun across the sky is constant as depicted in the solar path chart.

Initial Sunlight is caught at 06:30.

Peak sunlight is assumed to occur at noon (12:00).

Ideal weather conditions (no significant cloud coverage, no significant wind) are assumed.

The solar tracker (and its controller), solar panel and the motor operate without any mechanical faults.

The incidence angle should not exceed an angle of 18.2°.

The motor experiences no noise disturbances.

The motor relies on external voltage.

The motor has a constant torque, and it has a constant load.

Electrical and mechanical noises do not affect solar tracking performance.

The solar panel has a constant power source supplied constant power input.

The solar panel controlled by the solar tracker has the following parameters [13]:

Mass : 50 kg

Dimensions (L×W×D) : 1.04 × 1.4 × 0.1 m

Damping constant : 5 Nm./rad.s^{-1}

Latitude : 26.2 °

Primary axis is the zenith axis.

The motor actuating the rotation of the solar panel has the following parameters [13]:

Back EMF constant (Kf): 0.07 V/(rad/s)

Torque constant (Kt) = 0.07 Nm/A

Inductance: 1 $\times 10^{-5}$ H

Resistance: 10 Ohm

Gear ratio: 3000

4.3. MATLAB/ SIMULINK MODEL

The figures below represent the model of the solar tracker modelled on MATLAB's Simulink package. The step response modelled is the sun's position (beta), whereas the Kp is the representation of the potentiometer which converts the input angle of the sun's position beta into a voltage signal which then gets processed into a current in the motor. The output current (Amps) signal goes into the solar panel plant as an input signal which is output as an angle of rotation of the solar panel. This angle adjusts the orientation of the solar panel to achieve optimal angle of incidence with the sunlight, to absorb solar energy. The model below is what is used to investigate the way in which to track sun position and control of the solar panel.

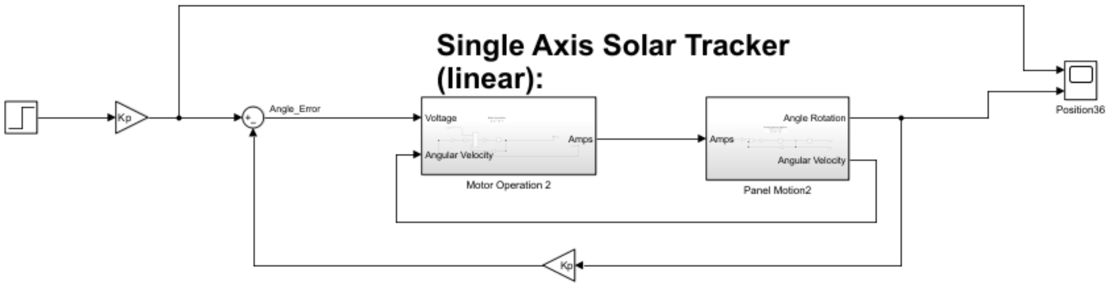


Figure 8: Simulink Model of the linear single axis solar tracker.

The Figures below are the extended operations/processes of the motor and the solar panel respectively.

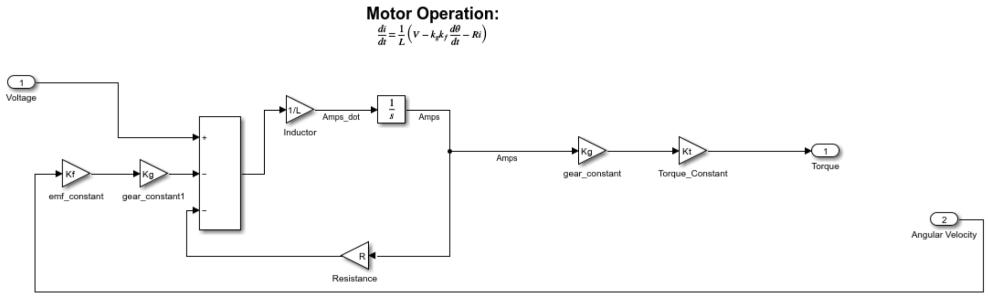


Figure 9: Motor Operation model.

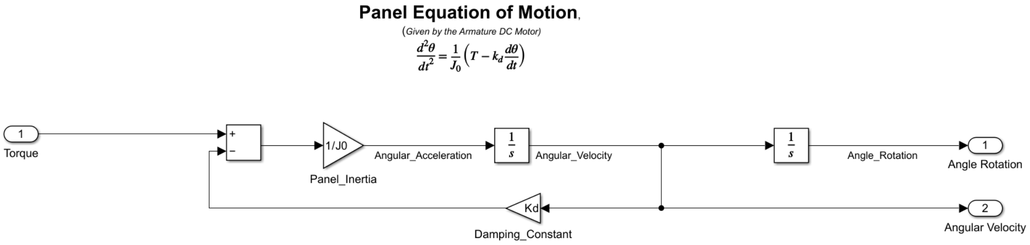


Figure 10: Panel Operation Simulink model

5. SYSTEM ANALYSIS

5.1. ANALYSIS OF BASIC PERFORMANCE

An initial basic analysis is conducted to determine the appropriate control systems required for the system to operate successfully without the use of a controller. The system is a time-invariant linear system, so this section tests the validity of that assumption. The first vital comparison is between the linear and non-linear systems to account for any discrepancies between the original system and its linearised counterpart. The second comparison is conducted between a regular forward open loop and a feedback loop. This assists in the implementation of the appropriate control system and the accuracy of the results obtained.

The Simulink model systems which will be referred to in the analysis are depicted in the figures below.

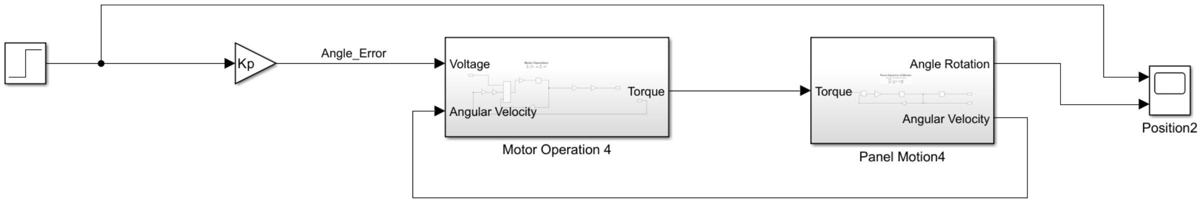


Figure 11: Non-Linear model: Open Loop

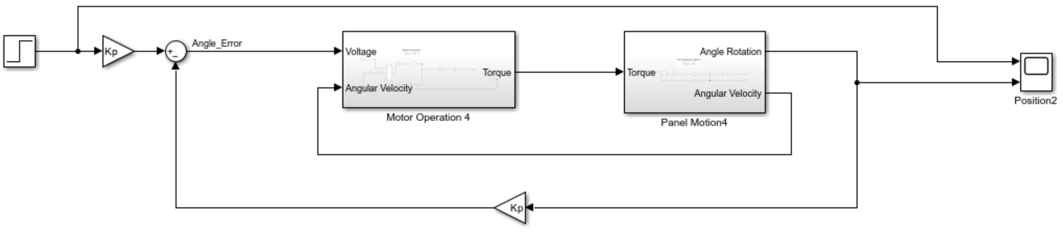


Figure 12: Non-Linear model: Feedback Loop

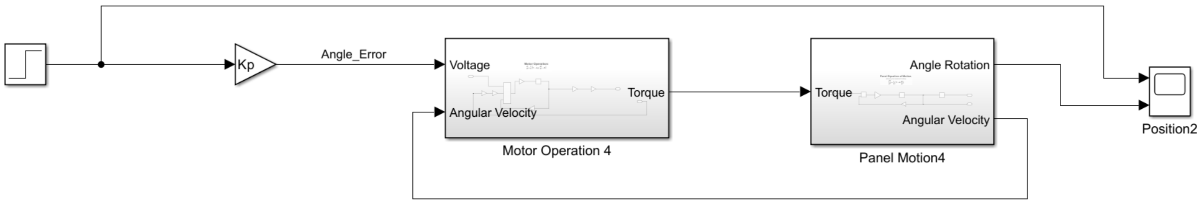


Figure 13: Linear model: Open Loop

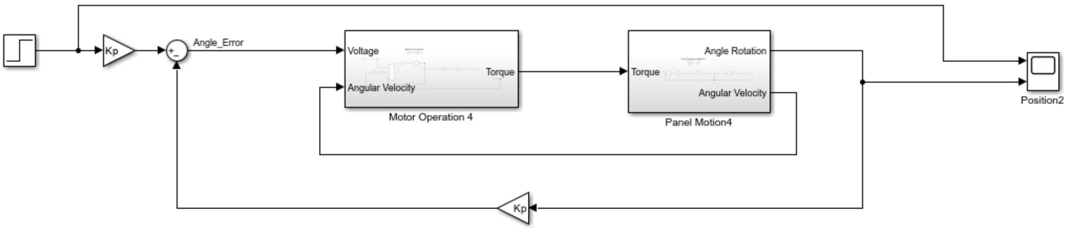


Figure 14: Linear model: Feedback Loop

The linear model and the non-linear systems have the same input behaviour as depicted in the following graphs:

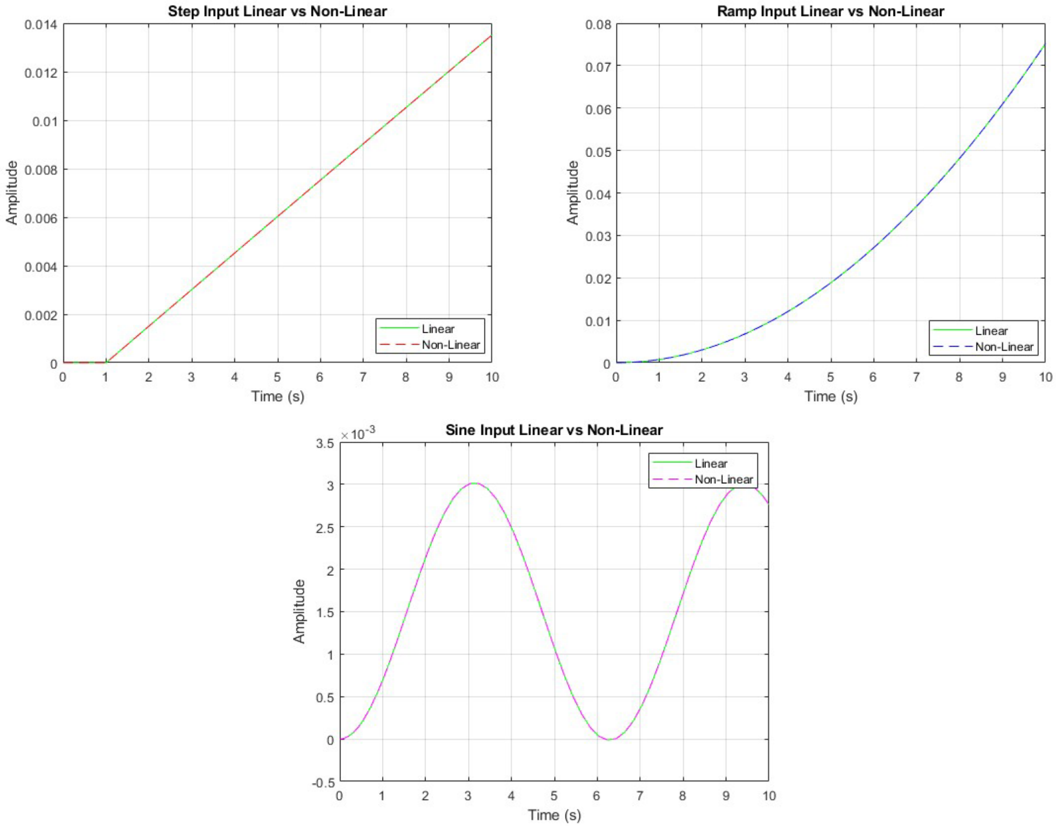


Figure 15: Input profiles for the linear and non-linear models

5.1.1. Time Domain

The figure below shows the performance of the linear and non-linear models in the time domain. The figure compares amplitude on the y-axis and the time on the x-axis. The graphs show that amplitude in the open loop and the feedback loop systems all increase linearly with time. The same is seen in both in linear and non-linear system. The input of the MATLAB model is the sun’s position, while the amplitude on the dependent axis on the graphs below represents the step response of the system.

The linearity of the first graph on both linear and non-linear systems implies a directly proportional relationship between the amplitude of the step response and the time. The amplitude increases linearly over time without stabilising nor reaching steady state. This behaviour is undesirable as it is expected that the system input will yield a predictable response that may be bounded yet as time progresses the step response amplitude increases, and the system becomes more unstable. The open-loop system has a delayed response to systematic changes as shown by the long period of 16 seconds required for the system to reach amplitude of 0.025. The system has continual divergence as no control algorithms are present in the model. The lack of stability and the continual divergence suggests that the system is and will not be able to maintain accurate sun tracking ability.

The Feedback-loop's graph has an increase in amplitude from 0 to 3 in 6000 seconds. The system then reaches steady state at an amplitude of 3.1426 (in 9000 seconds) which then remains constant through system operation. The system response from 0 to steady state is too long as no realistic system would be expected to 'calibrate' itself for 1 hour and 40 means so that its results can be reliable and accurate. The open loop system can be approximated to reach response amplitude of 1920 seconds. Despite the unstable nature of the open-loop system, this comparison further corroborates that the feedback system's response is too long. The Feedback system being able to stabilise at an amplitude of 3.124 shows that the feedback loop component is successful in regulating the system and ensuring system stability.

The non-linear system's response being linear was unexpected in that generally nonlinear model hold various complexities and as such linear function of the system response is not entirely and accurately representative of the system's realistic behaviour. This response does suggest that the system is doesn't have significant non-linearities, such as hysteresis or dynamic lag. Different inputs to the solar tracker model can result in unexpected responses between linear and non-linear models. With a step input the feedback loop may experience overshooting as it will rapidly attempt to reach stability, while with a sinusoidal input phase lag and errors can be expected.

The linear and non-linear models are mathematically modelled in different ways with the linear model being a simplification of the realistic non-linear model. The linear system does clearly represent the non-linear system if the responses are identical. Due to the graphed relationship being the same the behaviour can imply that linearisation of the model doesn't affect the overall system response of the model. A linearised model may yield similar if not identical behavioural properties as the non-linear model in terms of the system's response improving.

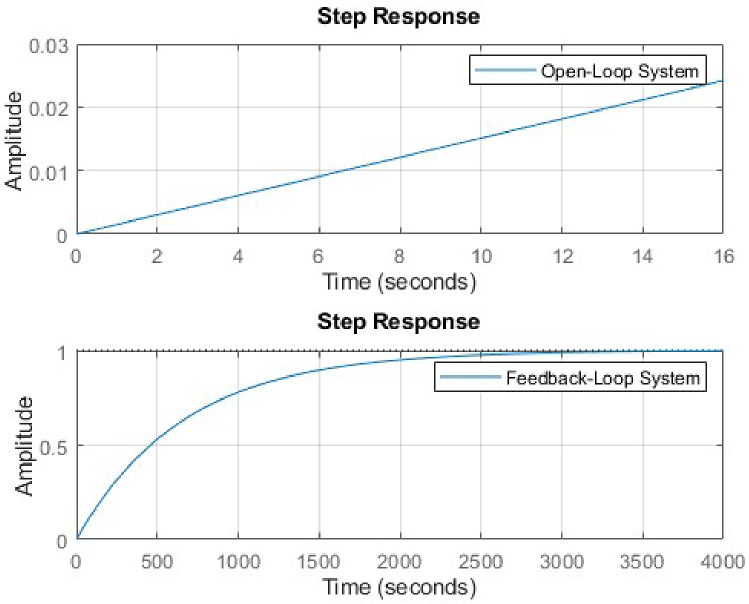


Figure 1616: Time Domain Step Response

The base performance specifications of the system are given as:

Table 1: Time domain base performance specifications

	Linear Open Loop	Linear Feedback Loop
Rise Time	NaN	1452.5 s
Transient Time	NaN	2586.4 s
Settling Time	NaN	2586.4 s
Settling Min	NaN	0.9045
Settling Max	NaN	0.9993
Overshoot	NaN	0
Undershoot	NaN	0
Peak	Inf	0.9993
Peak Time	Inf	4840.9 s

The table above corresponds to the graphical representation of the system response in the time domain. For the open-loop system, the response is seen to be unstable and not reaching steady state which is seen by the peak and peak time equating to infinity and the settling time being non-existent. This response proves that the use of an open loop system for the application of the solar tracker is unsuitable. For the feedback loop system, the rise time (which indicates the time a system takes to reach 90% of its final value) is relatively slow at 4840.9 seconds. The system responds in a delayed manner to system inputs and changes. The overshoot and undershoot being 0 is ideal as it indicates that the system response does not go beyond the required final value as it may be critically or overly damped.

5.1.2. FREQUENCY DOMAIN

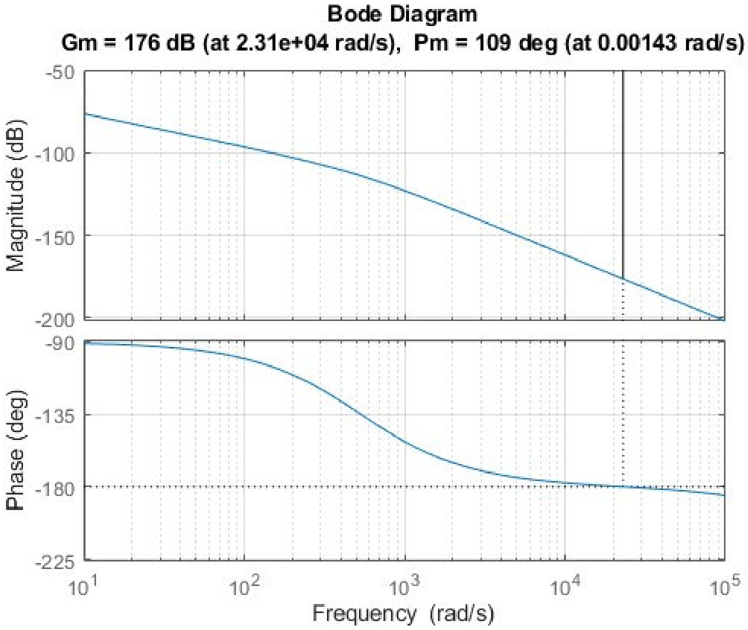


Figure 1717: Frequency domain system response

The Frequency domain performance specifications are as follows:

Table 2: Frequency domain performance specifications

	Feedback Loop
Gm	176
Pm	109
Wcg	23075
Wcp	0.00143

The basic performance of the linear system in the frequency domain is represented by the Bode diagrams shown. The top graph plots the relationship between the system's gain magnitude (measured in decibels) and the frequency in radians/second. The absolute magnitude gains at 2.31×10^4 rad/s is indicated as 176 dB. The graph is seen to be decreasing with increasing frequency; however, the absolute magnitude can be said to be increasing with increasing frequency. The gain value indicated above suggests that the system is unable to withstand increase in gain as the system is unstable and that anymore increase would significantly destabilise the system. The system can be said to be unstable, yet its current unstable behaviour based on the trend on the magnitude and frequency plot is seen to be very predictable.

The bottom phase plot illustrates the relationship between the system's phase margin in degrees and the frequency in radians/second. The absolute phase margin (Pm) is indicated as 109 degrees at 0.00143 rad/s. The graph is seen to be decreasing with increase in frequency, showing in inversely proportional relationship. This negative profile of the graph indicates system instability. Anymore decrease will further the system's instability. This behaviour can be backed up with rationale that any phase margins/ responses below 0 are indicative of a system operating in instability and this leads to the need for corrective actions or additional systems to regain stability. The use of an advanced controller can be such a corrective system, which will help the system regain stability and ensure efficient solar tracking.

In order to corroborate the results output from the Simulink model, analytical analysis of the frequency domain response is conducted as shown below.

Analytical Analysis:

The transfer function of the modelled system is:

$$T(s) = \frac{66.78}{8.292 \times 10^{-5}s^3 + 82.92s^2 + 4.415 \times 10^4 s + 66.78}$$

$$T(i\omega) = \frac{66.78}{8.292 \times 10^{-5}(i\omega)^3 + 82.92(i\omega)^2 + 4.415 \times 10^4(i\omega) + 66.78}$$

$$T(i\omega) = \frac{66.78}{-8.292 \times 10^{-5}(i\omega^3) - 82.92\omega^2 + 4.415 \times 10^4(i\omega) + 66.78}$$

$$T(i\omega) = \frac{66.78}{(66.78 - 82.92\omega^2) + i(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}$$

Multiply with the complex conjugate

$$T(i\omega) = \frac{66.78}{(66.78 - 82.92\omega^2) + i(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)} \times \frac{(66.78 - 82.92\omega^2) - i(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}{(66.78 - 82.92\omega^2) - i(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}$$

$$T(i\omega) = \frac{66.78(66.78 - 82.92\omega^2) - i66.78(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}{(66.78 - 82.92\omega^2)^2 - (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2}$$

$$Re[T(i\omega)] = \frac{66.78(66.78 - 82.92\omega^2)}{(66.78 - 82.92\omega^2)^2 - (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2}$$

$$Im[T(i\omega)] = \frac{-66.78(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}{(66.78 - 82.92\omega^2)^2 - (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2}$$

$$M(\omega) = |T(i\omega)| = \sqrt{Re[T(i\omega)]^2 + Im[T(i\omega)]^2}$$

$$\therefore M(\omega) = 66.78\sqrt{\frac{(66.78 - 82.92\omega^2)^2 + (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2}{((66.78 - 82.92\omega^2)^2 - (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2)^2}}$$

$$\therefore m(\omega) = 20\log(66.78) + 20\log\left(\sqrt{\frac{(66.78 - 82.92\omega^2)^2 + (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2}{((66.78 - 82.92\omega^2)^2 - (4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)^2)^2}}\right)$$

$$\varphi(\omega) = tan^{-1}\left(\frac{Im[T(i\omega)]}{Re[T(i\omega)]}\right)$$

$$\varphi(\omega) = tan^{-1}\left(\frac{(4.415 \times 10^4\omega - 8.292 \times 10^{-5}\omega^3)}{(66.78 - 82.92\omega^2)}\right)$$

The analytical analysis can be found under the following section in the MATLAB Code:

```
%% Frequency-domain Analytical Calculations %%
```

5.2. STABILITY ANALYSIS

The stability of the linear system is explored with use of the Routh-Hurwitz, BIBO and Nyquist criteria.

5.2.1. Absolute Stability

For any performance specifications to hold any relevance or reliability, absolute stability of a system must be achieved and maintained. The absolute stability indicates that for a system initially at equilibrium, after an excitation or input, the system will undergo a change, yet the system will ultimately return to system equilibrium. This ensures that no input or excitation will continue to affect the system indefinitely.

The pulse wave used to test for Absolute Stability:

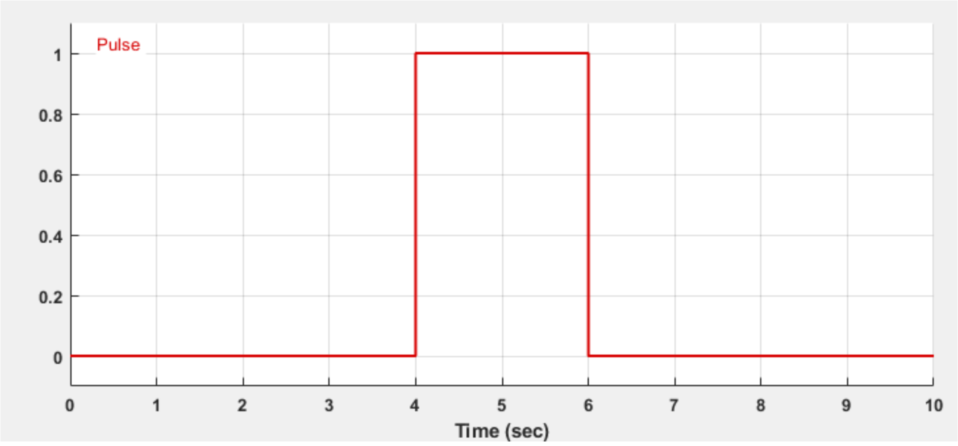


Figure 1818: pulse wave used to test for Absolute Stability

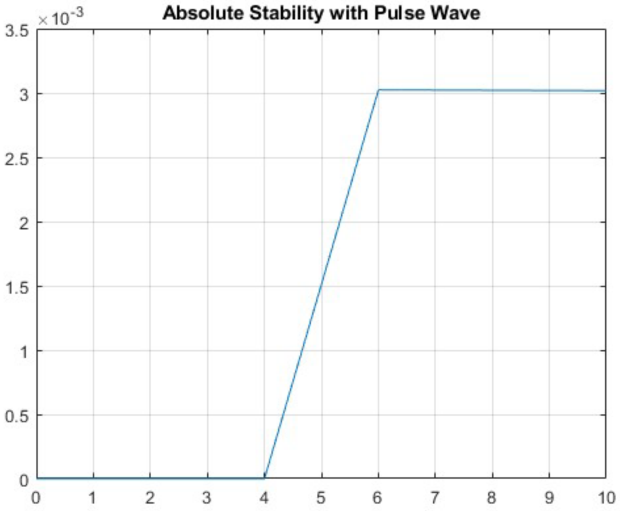


Figure 1919: Absolute stability with pulse wave

The above figure shows the feedback system is unstable. A pulse wave was introduced to the system, but the pulse signal does not filter out over time. This proves that any disturbance that is introduced to the system without any controller will not return to equilibrium.

To determine absolute stability the unity feedback of the solar tracker model must be defined as this will aid in the simulation of stability criteria/ criterion of the solar tracker. The system's steady state parameters (damping ratio, rise or settling time, percent overshoot etc.) can then be defined. The unity negative feedback is calculated as follows:

For the unity Feedback Loop, the system transfer function below is used:

$$\frac{\theta(s)}{V(s)} = \frac{K_g K_t}{(Ls+R)(Js^2+bs)+\ {K_g}^2 K_t K_f s}$$

$$\frac{\theta(s)}{V(s)} = \frac{K_g K_t}{(Ls+R)(Js^2+K_d s)+\ {K_g}^2 K_t K_f s}$$

The computation of the simplification of the non-linear transfer function of the unity feedback is shown in the next step.

$$\frac{\theta(s)}{V(s)} = \frac{\dfrac{K_pK_gK_t}{(Ls+R)(Js^2+K_ds) + \ {K_g}^2K_tK_fs}}{\dfrac{(Ls+R)(Js^2+K_ds) + \ {K_g}^2K_tK_fs + K_pK_tK_g}{(Ls+R)(Js^2+K_ds) + \ {K_g}^2K_tK_fs}}$$

$$T(s) = \frac{\theta(s)}{V(s)} = \frac{K_pK_gK_t}{(Ls+R)(Js^2+K_ds) + \ {K_g}^2K_tK_fs + K_pK_tK_g}$$

The denominator is further simplified to:

$$(Ls+R)(Js^2+K_ds) + \ {K_g}^2K_tK_fs + K_pK_tK_g$$

$$LJs^3 + LK_ds^2 + RJs^2 + \ RK_ds + {K_g}^2K_tK_fs + K_pK_tK_g$$

$$LJs^3 + (LK_d + RJ)s^2 + (RK_d + {K_g}^2K_tK_f)s + K_pK_tK_g$$

Let $A = LJ$; $B = LK_d + RJ;\ C = RK_d + {K_g}^2K_tK_f;\ D = K_tK_gK_p$

The denominator is simplified to the final form shown below:

$$As^3 + Bs^2 + Cs + D$$

Using the Routh- Hurwitz analysis:

Table 33: Symbolic Representation of Routh-Hurwitz Table for Forward Loop

s^3	A	C
s^2	B	0
s^1	$\frac{BC - A(0)}{B}$	0
s^0	ε	0

Table 44: Linear Forward Loop Routh-Hurwitz Table

s^3	8.292e-05	44150
s^2	82.92	0
s^1	44150	0
s^0	ε	0

The stability of the system depends on the sign changes on the graphs above.

Table 55: Symbolic Representation of Routh-Hurwitz Table for Feedback Loop

s^3	A	C
s^2	B	D

s^1	$\frac{BC - AD}{B}$	0
s^0	D	0

Table 66: Linear Feedback Loop Routh-Hurwitz Table

s^3	8.292e-05	44150
s^2	82.92	66.78
s^1	44150	0
s^0	66.78	0

Analysis of the Routh-Hurwitz Table 2 and Table 4 showed that both the forward loop and feedback loop are stable since neither had a zero in their first column, suggesting that neither had poles in the right-hand plane. Neither of the tables had a row of zeroes either and therefore the transfer function of both the forward loop and feedback loop had no poles on the $j\omega$ axis as well. This means that as the system tends towards its steady state response it does not oscillate.

5.2.2. BIBO stability

BIBO (Bounded input bounded output) stability is a criterion which defines stable systems to having a restricted response. A stable system is one that responds to a defined input with a defined magnitude. The figure below shows the poles of the linear models in terms of the BIBO stability criterion.

The BIBO stability graph with the sun's position as sine wave input is shown in the figure below. The graph is seen to be increasing with increasing time. The outputs have small, smooth consistent fluctuations with the increments which mimic the sinusoidal wave input. The graph with the square input as a similar profile, the main difference being that the fluctuations are sharp in the correspondence with a square step input. Despite the shapes of the profiles, it is to be observed that the gradient and the overall output values per time step are the same on both graphs.

Both graphs show no indication of settling into a constant or bounded pattern as they are increasing over time. Since the outputs in both cases are constantly increasing, they are concluded to being unbounded. The BIBO criteria determines systems as being stable if the system input will result in an output that lies within the boundary of other output values.

The lack of bounds on these plots represents that based on the BIBO criteria, the system is unstable. Since the system is unstable it will not be able to effectively track the sun's position over time. The system instability being illustrated may lead to the system's uncontrollable behaviour which will then result in equipment degradation and then ultimately system failure. To ensure stability of the system control mechanisms may be adopted, to avoid such problems.

The input signals used to test BIBO Stability are represented in the figure below

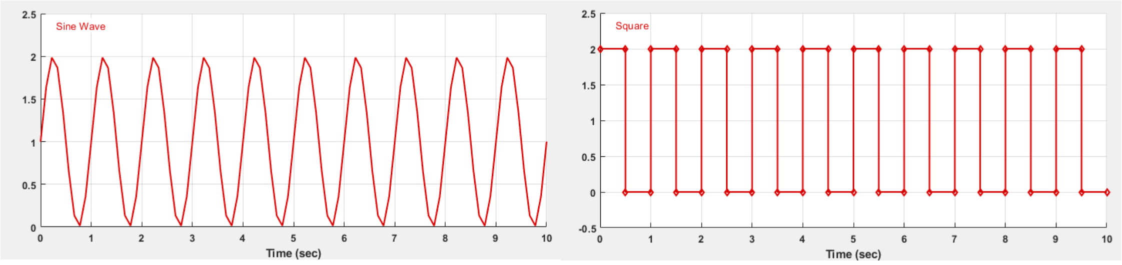


Figure 20: input signals used to test BIBO Stability

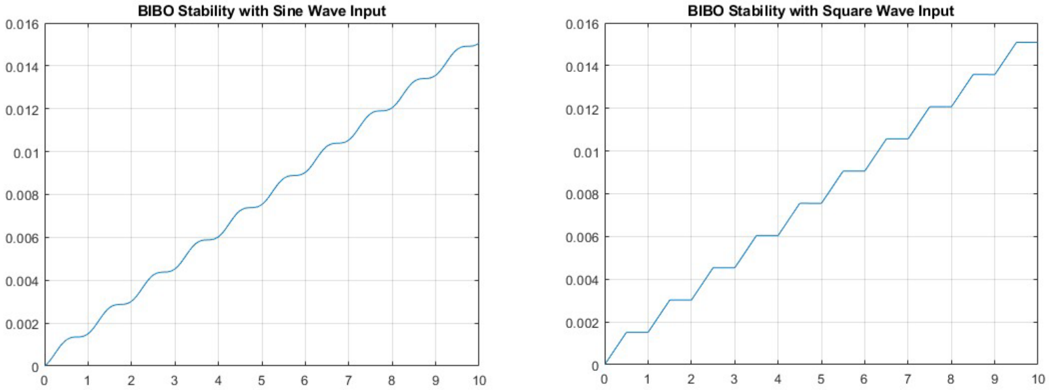


Figure 21:BIBO stability graphs.

5.2.3. Relative stability

The system's relative stability is represented by the Nyquist plot which is a polar plot used to determine the system's roots. It is noteworthy that the Nyquist relative stability criterion is limited to being only to describe the system using a linear differential equation describing the system.

The Nyquist response of the linearised feedback loop system is represented by the figure below. The Nyquist plot exists in the complex plane with an infinite looping which is bounding the frequency range of the system. The plot depicts the response of the system as a circular loop with the critical point at -1+j0. From the provided Nyquist plot, it is seen that the path line of the system response is not encircled in the critical point aforementioned. Since the plot is not looping around the critical point, the plot is depicted to not have any encirclements (loops). The plot does manage to reach close proximity with the critical point. The system can be said to be marginally stable since the poles and number of clockwise encirclements (loops of the plot) are the same. The system may be found to be extremely sensitive to certain types of system variations. Any changes in the system inputs or disturbances might cause the plot to reach and lie in the encirclement which would result in system instability.

The feedback loop has complex roots bounded within the right-hand side of the complex plane. The location of the elliptical curve being on the right side of the s plane (the complex plane) implies that the system has instability. It could be said that the system's instability is controlled via aid of the feedback system since the feedback data helps in correcting any system inputs to yields desirable output. The roots are expected to lie within the boundaries of the elliptical curve. The roots are not always known since the feedback system constantly adapts to various sun positions, and errors between the set point of the solar panel and the actual output of the panel. The randomness and the variations of feedback data affect the stability plotting of the system. The use of PID tuning helps with reducing any instability and improving overall stability.

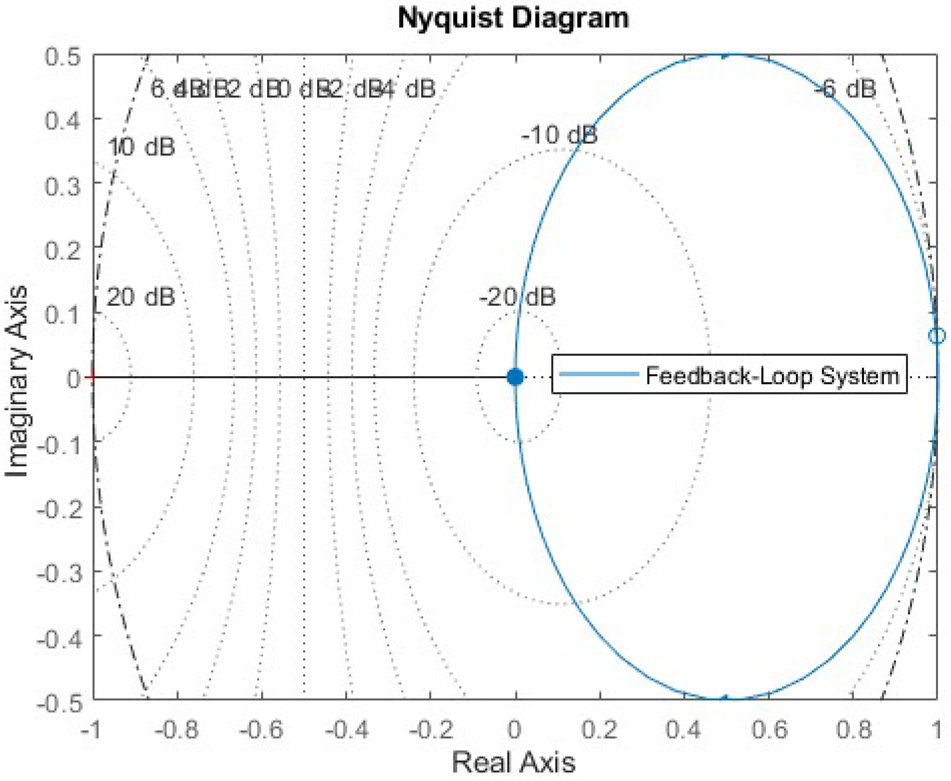


Figure 22: Nyquist linear relative stability.

<u>The Nyquist Analytical Analysis:</u>

$$T(s) = \frac{K_t K_g K_p}{LJ_0 s^3 + (LK_d + RJ_0)s^2 + (LK_d + RJ_0)s + K_t K_g K_p}$$

Let $A = LJ_0$; $B = LK_d + RJ_0$; $C = c$; $D = K_t K_g K_p$

$$\therefore T(s) = \frac{D}{As^3 + Bs^2 + Cs + D}$$

Substitute $s = j\omega$

$$T(j\omega) = \frac{D}{A(j\omega)^3 + B(j\omega)^2 + C(j\omega) + D}$$

$$T(j\omega) = \frac{D}{-Aj\omega^3 - B\omega^2 + Cj\omega + D}$$

$$T(j\omega) = \frac{D}{(D - B\omega^2) + j(C\omega - A\omega^3)}$$

Multiply with the complex conjugate

$$T(j\omega) = \frac{D}{(D - B\omega^2) + j(C\omega - A\omega^3)} \times \frac{(D - B\omega^2) - j(C\omega - A\omega^3)}{(D - B\omega^2) - j(C\omega - A\omega^3)}$$

$$T(j\omega) = \frac{D(D - B\omega^2) - jD(C\omega - A\omega^3)}{(D - B\omega^2)^2 - (C\omega - A\omega^3)^2}$$

$$T(j\omega) = \frac{D(D - B\omega^2)}{(D - B\omega^2)^2 - (C\omega - A\omega^3)^2} - j\frac{D(C\omega - A\omega^3)}{(D - B\omega^2)^2 - (C\omega - A\omega^3)^2}$$

$$\therefore Re[T(j\omega)] = \frac{D(D - B\omega^2)}{(D - B\omega^2)^2 - (C\omega - A\omega^3)^2}$$

$$Im[T(j\omega)] = \frac{-D(C\omega - A\omega^3)}{(D - B\omega^2)^2 - (C\omega - A\omega^3)^2}$$

To obtain the Nyquist plot, the Imaginary function is plotted against the Real function.

The code where the analytic functions were tested can be found under the following section in the MATLAB Code:

```
%% Nyquist Analytical Analysis %%
```

5.3. IMPLICATIONS FOR CONTROL

The considerations for the use of a controller on the uncontrolled solar controller system previously analysed are to be made since the use of a controller has certain implications which are to be considered. The use of a controller can result in the improvement of the plant and the overall system. The improvement in the plant design could be seen in the solar tracker having improved performance, efficiency and motor and solar panel system integration. The use of a controller can also help maintain optimal system performance in the presence of wind disturbances, which could otherwise affect the motion, orientation, and overall function of the system. The feedback system efficiency and reliability are very likely to improve with the combination of controller implementation.

Smooth dynamic motions and adjustability of the solar panel system under influence of optimal solar position could be possible with the controller adaption. The controller can administer continuous monitoring of the sun position against the solar panel's position and other external inputs whilst simultaneously adjusting the solar panel position to ensure optimal absorption of the solar energy. This controller system could dampen or reduce the impacts that the external disturbances could elicit within the system, a characteristic which is not available in the use of standard uncontrolled feedback systems. This is due to the flexibility and robustness of using a controller.

The improved plant design can be achieved by adopting sun position tracking algorithms (with dynamic control capability); using solar, light and wind sensors to accurately monitor environmental data. The system can also be improved by the development of system monitoring algorithms to detect, monitor and communicate faults and initiate troubleshooting protocols. The controller can be tested and then tuned to the local operating conditions.

Although the controller can yield various benefits in the system and plant design, certain limitations need to be considered. The use of controller would require tuning based on operating conditions as the controller cannot universally (without active adjustment by operators at the location of operation) ‘control’ system outputs. The controller, like the feedback system can only consider the inputs that have been considered and modelled directly into the system. The neglected environmental disturbances not input into the system cannot be evaluated by the system nor the controller. The physical mechanical state of the system is likely decline with prolonged use and the controller may operate under the assumption that the system is in peak conditions.

Despite the limitations outlined, it is noteworthy that the use of the controller in systems design and control helps reach and maintain stability and improves system response and performance.

6. CONTROLLER IMPLEMENTATION

6.1. CLOSED-LOOP CONTROLLERS

6.1.1. ROOT-LOCUS TECHNIQUE

The figure below represents the root locus of the feedback system. The poles of the system are shown to be at 0 and -1 on the horizontal axis. The plot shows that at the -9.99×10^5 pole, the gain (K) is equal to zero. The large pole value is indicative of a highly damped and stable system; however, the plot indicates that the system is critically damped. The system has no overshoot (which is ideal) and a high frequency. The pole at -266 has a lower frequency of 266 rad/s, overshoot of 0% and a high gain value. The system maintains its status as critically damped.

The location of the poles and the gains toward the right-hand side (on the real axis) imply that there is slight probability of the system being unstable. The system is currently stable on the far-left side of the real axis. The increase in gain results in the increase of the pole values and the poles move more and more to the right half plane which has instability. It can be concluded that the system has stability on the left half plane while the left half-plane is pivoting on the tip of instability. Any further increase to the gains will push the system on the right half plane to instability.

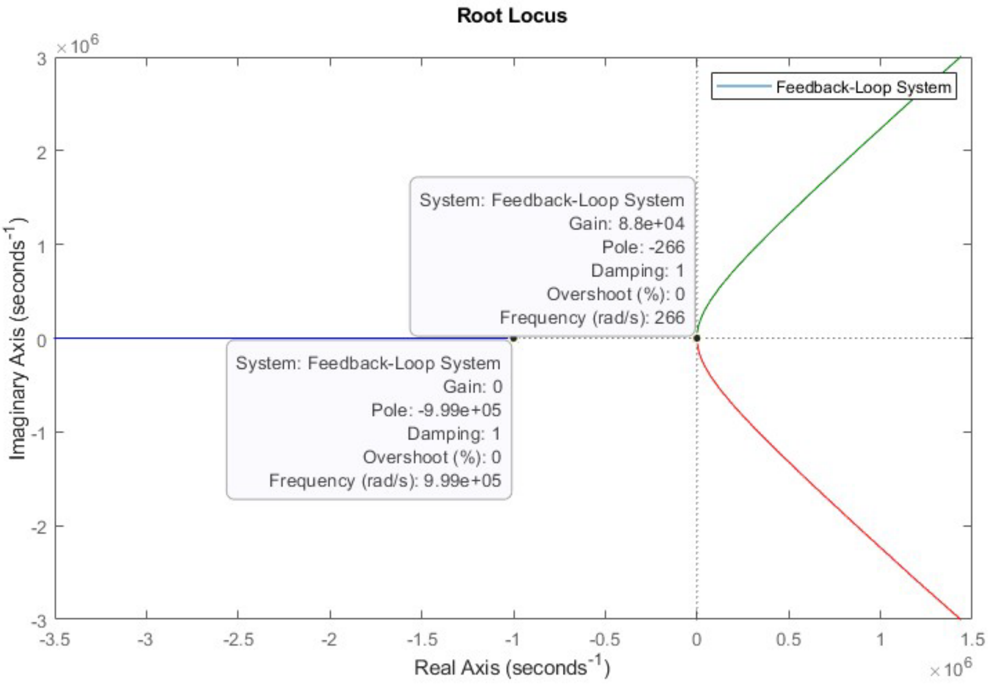


Figure 23: Root-locus plot.

The analytical analysis shown below corresponds to the data shown in the root-locus plot above as shown by the determination of the poles and the gain values. It is therefore permissible to say that the simulated model and its corresponding code is accurate in representing system behaviour.

Analytical Analysis:

The plant Transfer function is:

$$G(s) = \frac{210}{8.292 \times 10^{-5}s^3 + 82.92s^2 + 44150s}$$

There are three poles in this system.

Finite poles: (s = 0; s = -9.9947; s = -0.0053)

Finite Zeros: None

Infinite Poles: Finite Poles>Finite Zeros $\therefore$ 0 Infinite Poles

Infinite Zeros: Finite Poles>Finite Zeros

$$\#Finite\ Poles - \#Finite\ Zeros$$

$$= 3 - 0$$

$$\therefore = 3\ Infinite\ Zeros$$

The Asymptotes:

$$\sigma_a = \frac{\sum Finite\ Poles - \sum Finite\ Zeros}{\#Finite\ Poles - \#Finite\ Zeros}$$

$$= \frac{[0 - 9.9947 - 0.0053] - [0]}{3 - 0}$$

$$= \frac{-10}{3}$$

$$\therefore \sigma_a = -3\frac{1}{3}$$

$$\theta_a = \frac{(2K + 1)\pi}{\#Finite\ Poles - \#Finite\ Zeros}$$

$$\theta_a = \frac{(2K + 1)\pi}{3}\ for\ K = 0, \pm 1, \pm 2, \pm 3$$

$$for\ K = 0 \quad \theta_a = \frac{\pi}{3}$$

$$for\ K = 1 \quad \theta_a = \pi$$

$$for\ K = 2 \quad \theta_a = \frac{5\pi}{3}$$

Breakaway Point:

$$KG(s)H(s) = \frac{66.78K}{8.292 \times 10^{-5}s^3 + 82.92s^2 + 44150s + 66.78}$$

Substitute KG(s)H(s) = -1 and s = σ

$$-1 = \frac{66.78K}{8.292 \times 10^{-5}\sigma^3 + 82.92\sigma^2 + 44150\sigma + 66.78}$$

$$66.78K = -8.292 \times 10^{-5}\sigma^3 - 82.92\sigma^2 - 44150\sigma - 66.78$$

$$K = -1.242 \times 10^{-6}\sigma^3 - 1.242\sigma^2 - 661.1261\sigma - 1$$

$$\frac{dK}{d\sigma} = -3.726 \times 10^{-6}\sigma^2 - 2.484\sigma - 661.1261$$

$$\therefore\ \sigma^2 + 666666.667\sigma + 177435883 = 0$$

Getting the roots:

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

$$\sigma = \frac{-666666.667 \pm 666134.1467}{2}$$

$$\therefore \sigma_1 = -\cancel{666400.4069}\ \ or\ \ \sigma_2 = -266.2602$$

Substituting the σ-value to obtain K

$$K_1 = -3.\cancel{6456 \times 10}^{13}\ \ K_2 = 8.8 \times 10^4$$

There is no jω-crossing, as the turn point of the graph is on the 0 Imaginary Axis.

6.1.2. PID FEEDBACK CONTROL

There are four types of Control: PID, PD, PI, and P.

Each is a combination of one or more of the undermentioned closed-loop responses.

Table 77: Closed Loop Response Behaviours for PID Control [16]

Closed-loop Response	Rise Time	Overshoot	Settling Time	Steady State Error
K_p	Decrease	Increase	Small Change	Decrease
K_i	Decrease	Increase	Increase	Decrease
K_d	Small Change	Decrease	Decrease	No Change

There are 3 forms of PID Tuning:

Balanced, Reference Tracking, and Input disturbance rejection.

The reference tracking tuning is aligned to follow the input presented as the desired input, whereas the input disturbance rejection is aligned to filter out any disturbance introduced into the system. The balanced tuning is a combination of both Reference Tracking and Input Disturbance Rejection. This enables the system to track the reference input efficiently and resist any disturbance introduced.

Thus, the best form of tuning to consider for this plant is balanced Tuning. The system must be able to track the reference input and be able to filter out any disturbances that may be introduced such as any type of gust wind or a bird hitting the panel, modelled as a pulse signal, constant wind, modelled as a constant block.

6.1.2.1. Proportional (P) Controller:

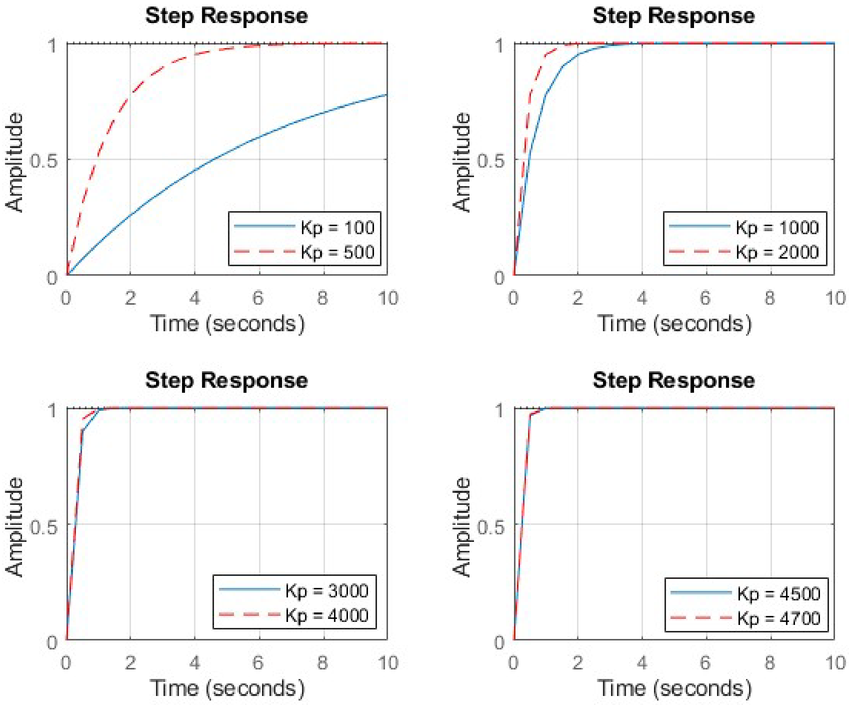


Figure 24: Proportional Control gain plot comparison.

Table 8:Performance Specifications for Performance Gains

Performance Specifications for Performance Gains (Kp)									
	Desired Performance Specifications	Kp = 100	Kp = 500	Kp = 1000	Kp = 2000	Kp = 3000	Kp = 4000	Kp = 4500	Kp = 4700
Rise Time	< 1 second	45.672	9.131	4.564	2.280	1.518	1.138	1.011	0.968
Settling Time	< 10 seconds	81.327	16.261	8.128	4.061	2.706	2.028	1.802	1.725
% Overshoot	< 5%	0	0	0	0	0	0	0	0
Peak	< 3.3	3.143	3.143	3.143	3.143	3.143	3.143	3.143	3.143

The Figure above is a group of plots utilising different proportional gain values. The plot corresponding to the Kp of 100 has a relatively slow response amongst all the plots as takes more than 10 seconds to reach the universal steady state value of 1. The Kp = 500 plot next to the Kp = 100 has a faster rise and settling time. This implies that the lower the Kp gain, the slower the response (settling time) and the rise time toward steady state.

Based on the table above, the only gains that met the desired performance parameters outline are Kp = 4700. The overshoot across all gain values being 0 suggests that the system is stable and that there is no oscillatory behaviour. This shows that based on the specifications the higher Kp value is more responsive and suitable (with the added benefit of continual stability) for the application. This data also shows how tolerant the overall feedback system is to high proportional gains, whilst still maintaining stability.

6.1.2.2. Proportional-Integral (PI) Controller:

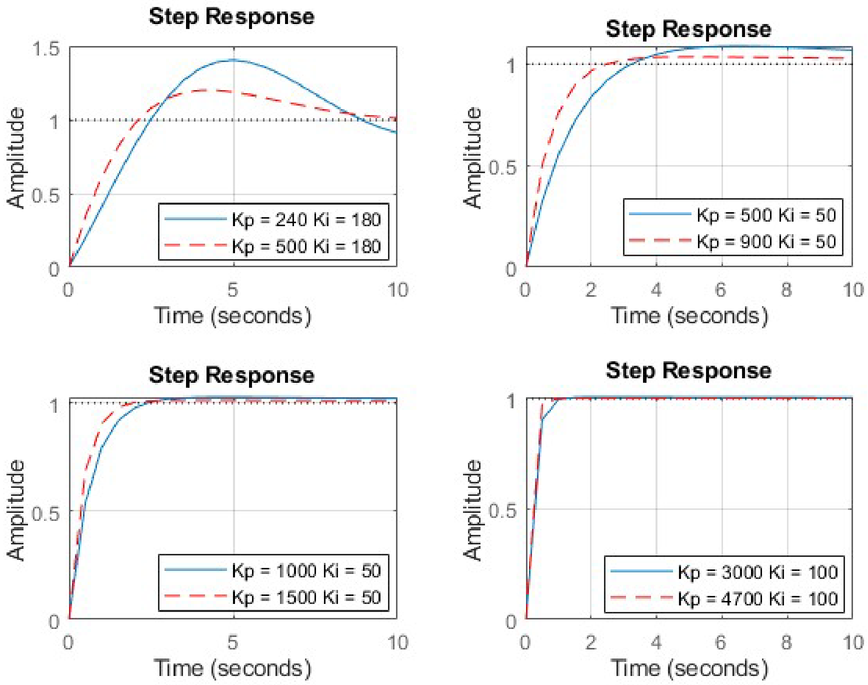


Figure 25: Proportional-Integral Control gain plot comparison.

Table 9: Performance Specifications for Performance Gains (Kp, Ki)

Performance Specifications for Performance Gains (Kp, Ki)									
	Desired Performance Specifications	Kp = 240. Ki = 180	Kp = 500. Ki = 180	Kp = 500. Ki = 50	Kp = 900. Ki = 50	Kp = 1000. Ki = 50	Kp = 1500. Ki = 50	Kp = 3000. Ki = 100	Kp = 4700. Ki = 100
Rise Time	< 1 second	3.627	3.340	5.266	3.904	3.646	2.704	1.425	0.942
Settling Time	< 10 seconds	66.242	33.282	32.301	36.404	36.031	28.233	6.049	1.575
% Overshoot	< 5%	57.764	35.713	18.667	8.434	7.182	3.707	2.015	0.836
Peak	< 3.3	4.9611	4.268	3.732	3.410	3.371	3.261	3.208	3.171

The Proportional-Integral (PI) controller considers the difference between the desired output and current error (like the P-controller) as well as the accumulated past errors to assist in improving stability performance and yield a more refined outcome then the P controller. A direct comparison can be made when looking at the last column where Kp=4700. This system has a lower settling time than the P controller. The downside to this controller is the introduction of overshoot but there are values that lie well beneath the desired limit of 5%. The use of a PI controller – as indicated in the table above – would require the system to have a Kp gain of at least 4700 and Ki gain of about 100 to meet the desired performance specifications criterion. However, this is the only condition that suits the desired specifications thus rendering inefficient. Although this would meet all the set requirements, a PI controller may not be the best solution due to the large gain value, but the data suggests it is possible.

6.1.2.3. Proportional-Derivative (PD) Controller:

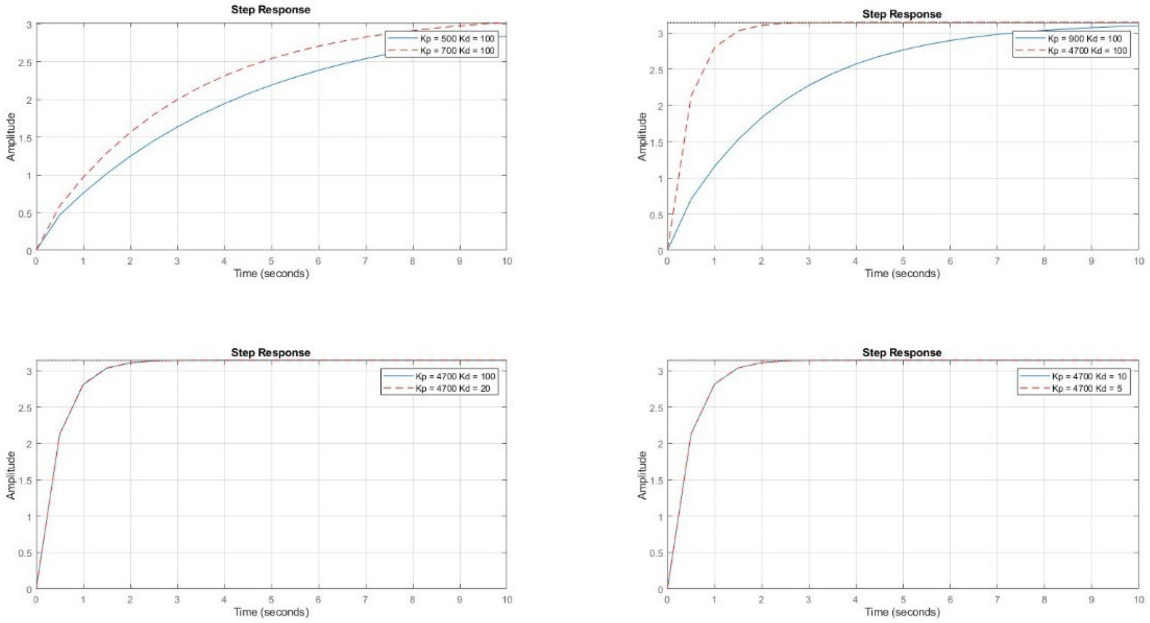


Figure 26: Controller plots Proportional-Derivative (PD)

Table 10: Performance Specifications for Performance Gains (Kp, Kd)

Performance Specifications for Performance Gains (Kp, Kd)									
	Desired Performance Specifications	Kp = 500; Kd = 100	Kp = 700; Kd = 100	Kp = 900; Kd = 100	Kp = 4700; Kd = 100	Kp = 4700; Kd = 50	Kp = 4700; Kd = 20	Kp = 4700; Kd = 10	Kp = 4700; Kd = 5
Rise Time	< 1 second	9.571	6.835	5.315	1.015	0.991	0.977	0.972	0.970
Settling Time	< 10 seconds	16.839	12.026	9.353	1.787	1.7561	1.738	1.731	1.728
% Overshoot	< 5%	0	0	0	0	0	0	0	0
Peak	< 3.3	3.143	3.143	3.143	3.143	3.143	3.143	3.143	3.143

The Figure above compares the proportional-derivative gain combination for the feedback PID controlled system. The plot on the upper left graph shows that the blue curve has a slower response and shallower gradient than the red curve. Although it is slower it has no overshoot implying stability of the system. It is evident that by keeping the Kd constant and increasing the Kp value, the system has faster response. Across the graphs it is seen that the increase in the proportional gain brings the two plots closer together until they coincide. Once they overlap the response of the system to the controller adjustment becomes balanced and faster in reaching 90% of the final steady state value.

The table above shows that the tuning of the gains results in the system meeting the desired performance specifications. Once the Kp gain is constant, the Kd gain has less significant affect on the performance parameters as the change across the various Kd values is small in comparison to the changes when Kd is constant and Kp varies. Despite that the desired performance achieved by the Kp = 4700 and the Kd values of 50,20,10 and 5, with Kd =5 exhibiting the best performance.

6.1.2.4. Proportional-Integral-Derivative (PID) Controller:

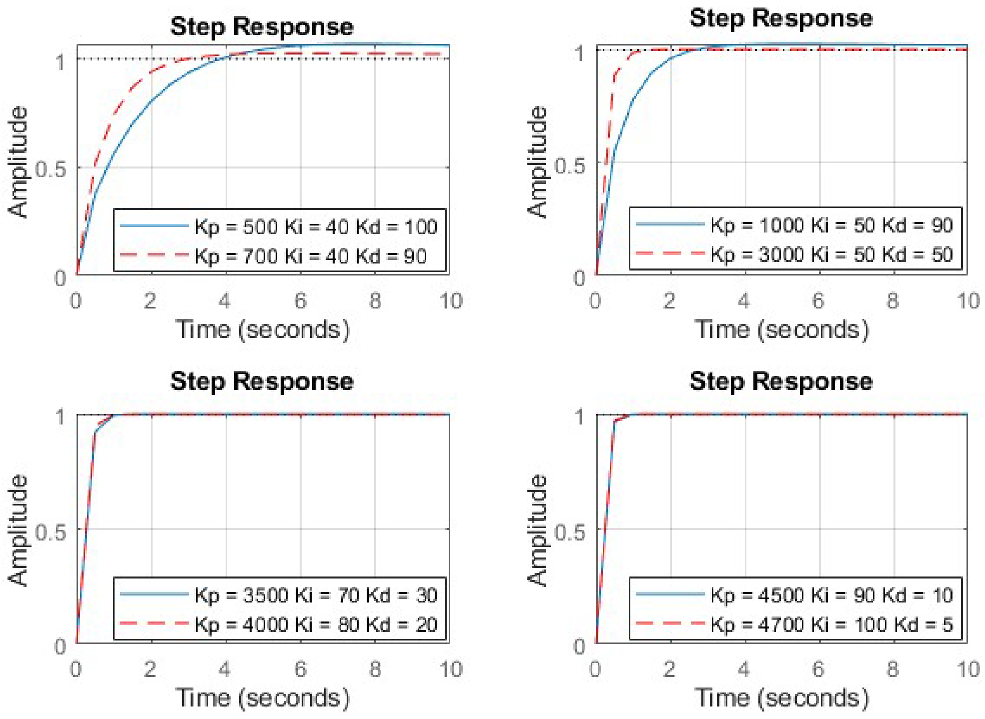


Figure 27: Proportional-Integral-Derivative (PID) Controller plots.

Table 11: Performance Specifications for Performance Gains (Kp, Ki, Kd)

Performance Specifications for Performance Gains (Kp, Ki, Kd)									
	Desired Performance Specifications	Kp = 500. Ki = 40. Kd = 100;	Kp = 900. Ki = 40. Kd = 90;	Kp = 1000. Ki = 50. Kd = 90;	Kp = 3000. Ki = 50. Kd = 50;	Kp = 3500. Ki = 70. Kd = 30;	Kp = 4000. Ki = 80. Kd = 20;	Kp = 4500. Ki = 90. Kd = 10;	Kp = 4700. Ki = 100. Kd = 5;
Rise Time	< 1 second	5.899	4.221	3.790	1.505	1.276	1.115	0.989	0.944
Settling Time	< 10 seconds	37.629	40.407	36.181	2.471	2.098	1.850	1.653	1.578
% Overshoot	< 5%	16.002	7.046	7.113	1.062	1.091	0.926	0.821	0.836
Peak	< 3.3	3.648	3.366	3.368	3.178	3.179	3.174	3.171	3.171

The PID controller incorporates the benefit of all 3 components – rapid settling time, accuracy when the system is in steady state as well as damping any oscillations produced, assisting the system to reach a steady state. The PID is the most efficient as it can achieve all the performance gains – Kp, Ki and Kd – with values of a minimum of 3000, 50, 50 respectively. This suggests that the combination of all three controllers would yield the best result as the system is able to maintain stability with the least number of computational gains. The problem with the PID controller is often overcompensation. The system is simple so it would be cheaper and computationally less strenuous to pick another controller as the differences are not too large.

By comparing all the controllers that initially met the performance requirements of the system for each controller type, the following is obtained:

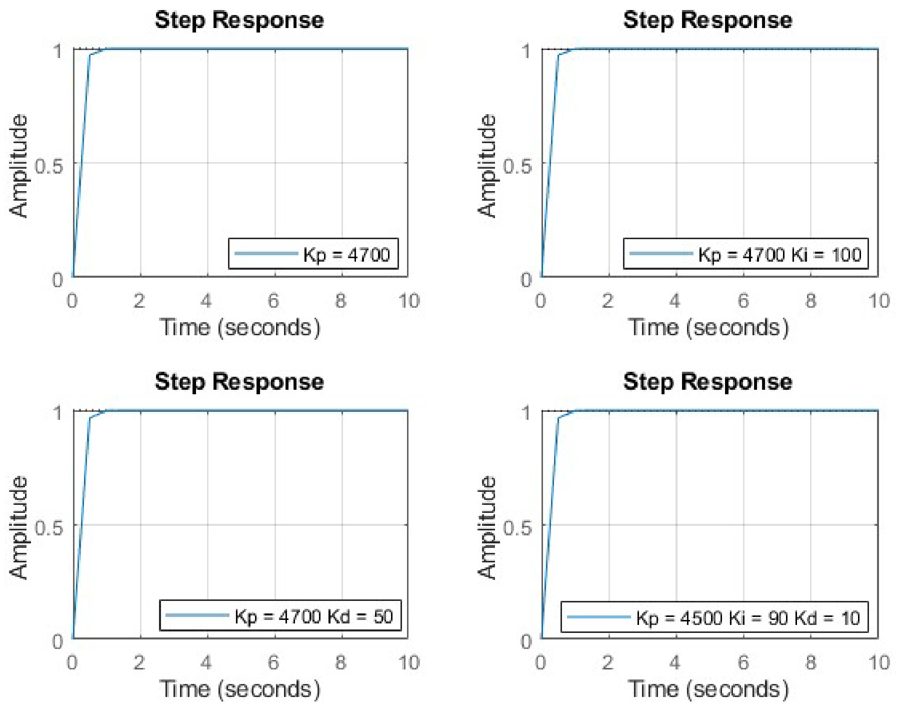


Figure 28: P, PI, PD, PID controller comparison plots.

Table 12: Comparison of controllers that initially met the performance requirements of the system

Controllers that initially met the performance requirements of the system					
	Desired Performance Specifications	Kp = 4700	Kp = 4700. Ki = 100	Kp = 4700. Kd = 50	Kp = 4500. Ki = 90. Kd = 10;
Rise Time	< 1 second	0.968	0.942	0.991	0.989
Settling Time	< 10 seconds	1.725	1.575	1.7561	1.653
% Overshoot	< 5%	0	0.836	0	0.821
Peak	< 3.3	3.143	3.171	3.143	3.171

The controller that meets the performance specifications of the system the best, is one that will have the least % overshoot, has the fastest settling time and the lowest peak. According to the aforementioned criteria, the Proportional controller will be the best fit for the system. The system being modelled is a simple system and is able to perform in spite the use of an integral or derivation component.

The new system transfer function with the implemented P – controller is:

$$T(s) = \frac{3.139 \times 10^5}{8.292 \times 10^{-5}s^3 + 82.92s^2 + 4.415 \times 10^4 s + 3.139 \times 10^5}$$

$$T(s) = \frac{3.139 \times 10^5}{(s + 0.0001)(s + 0.0053)(s + 9.9947)}$$

The Control System is a Type 0 system.

6.2. SYSTEM EVALUATION

6.2.1. Solar tracking performance in winter and summer

During winter the following assumptions were made:

Using the Latitude of 26.2°, on the 172nd day (June 21st) of the year @ 6:30 in the morning. The Declination is +23.45° indicating Wintertime in the Southern Hemisphere.

During summer the following assumptions were made:

Using the Latitude of 26.2°, on the 355th day (December 21st) of the year @ 6:30 in the morning. The Declination is -23.45° indicating Summertime in the Southern Hemisphere.

The linear and the non-linear models produce the same performance results as the linear models are an accurate approximation of the non-linear model. The non-linear model has been accurately linearised, resulting in a linearised model which accurately represents non-linear responses. Therefore, the analysis to be conducted below will involve figures which simultaneously represent linear and non-linear response.

6.2.1.1. Linear system PID Behaviour with no Disturbance:

The summer and winter graphs for the sun position and the panel tracking in the Figure below coincide perfectly. This indicates that the PID controlled solar tracker system is effective in tracking the sun's position and maintaining the optimal angle of the solar panel to ensure optimal solar energy absorption. The solar tracker's feedback loop for the linear system using the proportional gain of the PID controller is accurate in tracking and following the sun's position. The proportional gain does ensure stability of the system even when the sun's position is constantly changing, and in the event of any sudden changes, the proportional gain responds quickly and accurately to changes. Although this matching behaviour is ideal for the solar tracker, the assumption of no disturbance doesn't replicate the realistic operating conditions as realistic conditions have disturbances.

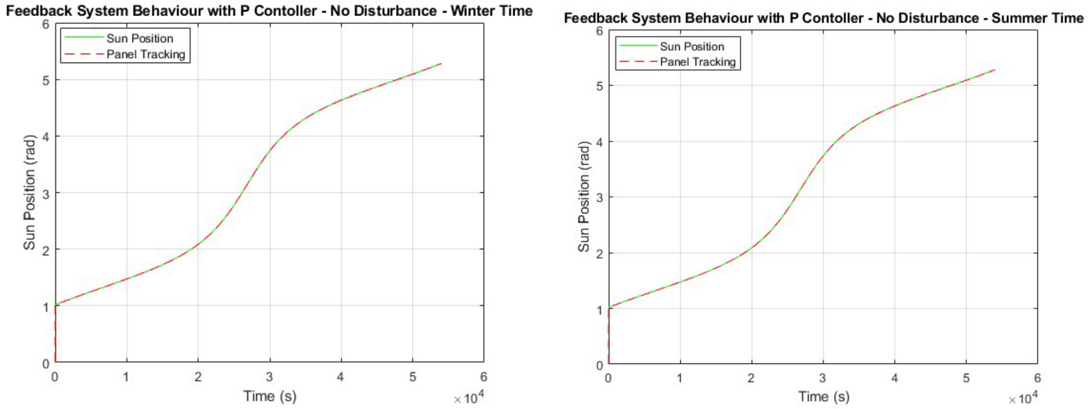


Figure 29: Feedback system behaviour with Proportion controller in winter and summer.

6.2.1.2. Non-Linear system PID Behaviour with no disturbance

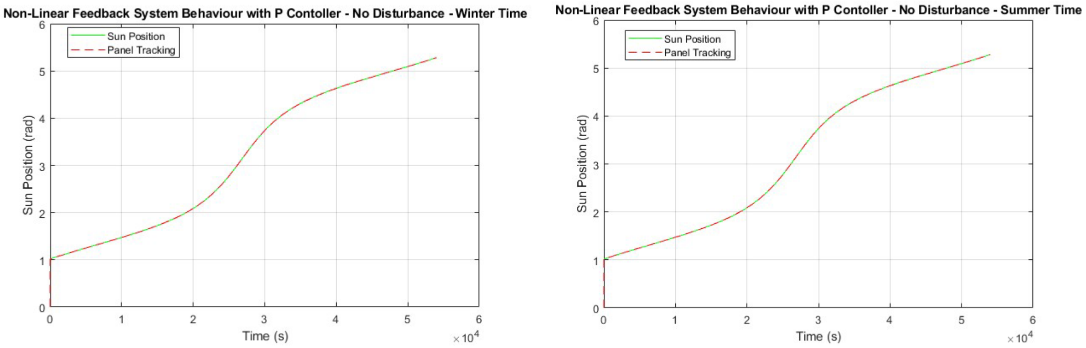


Figure 30: Feedback system behaviour with Proportion controller in winter and summer with no disturbance

The summer and winter graphs for the sun position and the panel tracking in the Figure above are identical. The alignment of the sun position and the panel tracking indicates that the PID controlled solar tracker system is effective in tracking the sun's position and maintaining the optimal angle of the solar panel to ensure optimal solar energy absorption. The solar tracker's feedback loop for the linear system using the proportional gain of the PID controller is accurate in tracking and following the sun's position. The proportional gain does ensure stability of the system even when the sun's position is constantly changing, and in the event of any sudden changes, the proportional gain responds quickly and accurately to changes. The linear and non-linear behaviour of the solar panel's sun tracking are the same. This compliments the aforementioned assumption that the linear model accurately reflects the non-linear behaviour.

6.2.1.3. Linear System PID Behaviour with a Constant Disturbance: (Constant Wind Disturbance)

The graphs for the sun position and the panel tracking in the Figure below coincide perfectly. This indicates that the solar tracker system is effective in tracking the sun's position and maintaining the optimal angle of the solar panel to ensure optimal solar energy absorption. The existence of a disturbance does not have any effect on the system's performance in tracking the sun's position accurately. The solar tracker's feedback loop for the linear system using the proportional gain of the PID controller is accurate in tracking and following the sun's position. The proportional gain does ensure stability of the system even when the sun's position is constantly changing, and in the event of any sudden changes, the proportional gain responds quickly and accurately to changes. This matching behaviour is ideal for the solar tracker, the assumption of the existence of a disturbance does replicate some the realistic operating conditions. However, disturbances that exist in realistic operating conditions such as wind disturbances are non-constant and may have patterns of fluctuation. Therefore, it is prudent to test the system behaviour with the Proportional controller for non-constant disturbances.

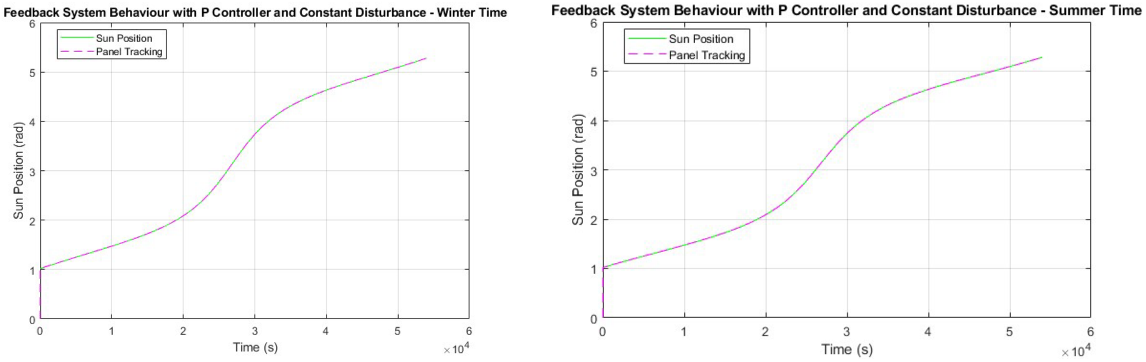


Figure 31: Feedback system behaviour with Proportion controller in winter and summer with constant disturbance

6.2.1.4. Linear System PID Behaviour with a Pulse Disturbance: (Wind Gust)

The graphs for the sun position and the panel tracking in the Figure below coincide perfectly. This indicates that the solar tracker system is effective in tracking the sun's position and maintaining the optimal angle of the solar panel to ensure optimal solar energy absorption. The existence of a disturbance does not have any effect on the system's performance in tracking the sun's position accurately. The solar tracker's feedback loop for the linear system using the proportional gain of the PID controller is accurate in tracking and following the sun's position. The proportional gain does ensure stability of the system even when the sun's position is constantly changing, and in the event of any sudden changes due to the pulse disturbances, the proportional gain responds quickly and accurately to changes. This matching behaviour is ideal for the solar tracker, the assumption of the existence of fluctuating disturbances does replicate some the realistic operating conditions. The use of the proportional gain for the PID controller manages to dampen and minimize the influence/ significance of the pulsating disturbance, resulting in optimal performance.

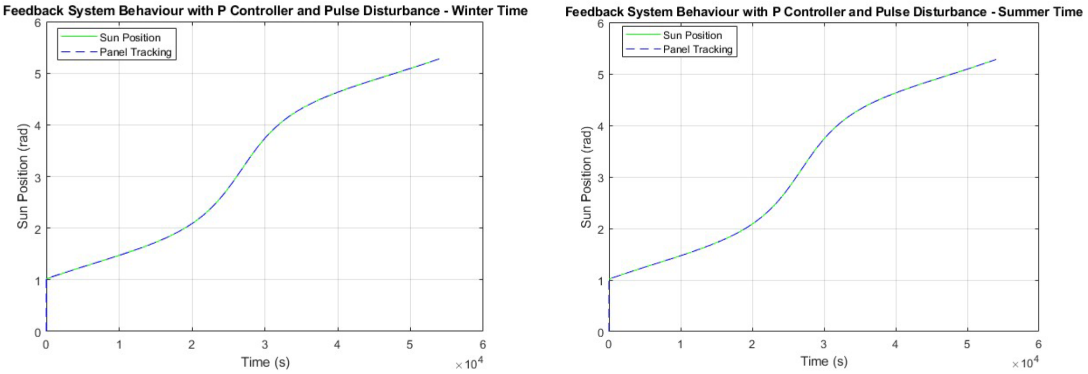


Figure 32: Feedback system behaviour with Proportion controller in winter and summer with pulsing disturbance

6.2.1.5. Linear System PID Behaviour with a Sine Wave Disturbance: (bent shaft)

The deep meshing of the gears, that resulted from a bent shaft, can cause a sine wave disturbance. This disturbance the affects the solar tracker's performance. The system is now unable to make the correct corrections and adjustments to align with the sun's position. The system with the proportional control is unable to compute for sinusoidal disturbances. The actual system seems to be diverging further and further away from the ideal orientations. The system is now unstable as the proportional control alone is insufficient to compute sinusoidal disturbances as the P controller cannot compensate for periodic disturbances. Due to divergence of the system, the system's efficiency in orienting itself to capture solar rays of incidence for power generation is significantly reduced. Since the P controller alone cannot compensate for such disturbances, the PI or PID controllers will need to be used.

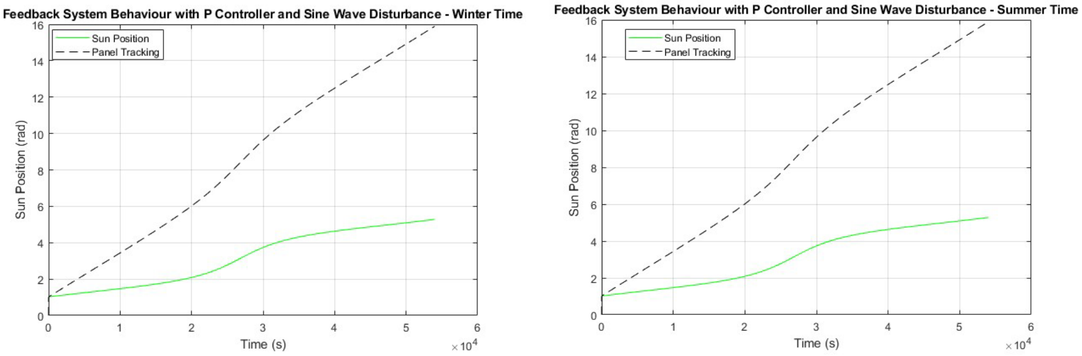
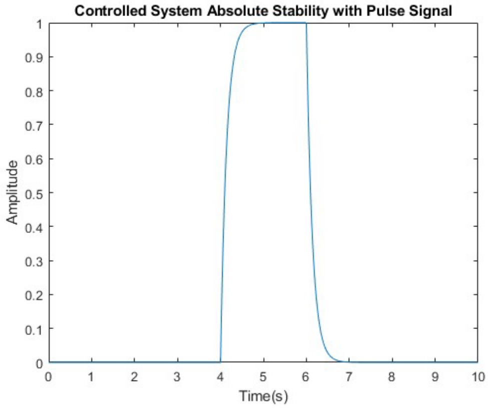


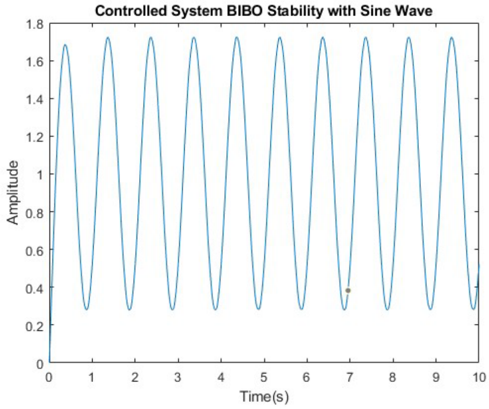
Figure 33: Feedback system behaviour with Proportion controller in winter and summer with sine wave disturbance

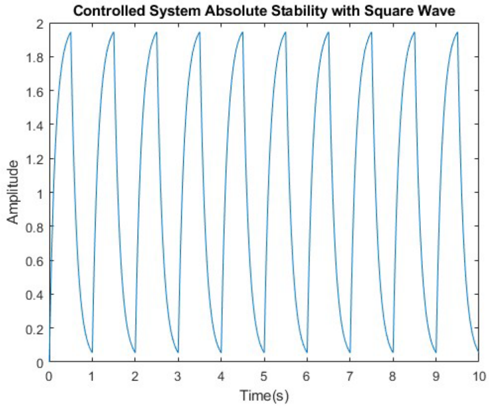
Summer VS Winter

The season for the designed tracker doesn't influence the performance of the solar tracker. The system is designed to be robust and reliable no matter what the season is, the solar tracker will be able to track the sun's position. The difference in the systems, however, lie in the amount of solar energy collected in winter being less than in summer due to cloud coverage and rain. It was evident in the analysis that the solar tracking performance in winter and summer is the same. This is due to the solar tracker being a vertical single axis tracker capable to tracking the sun's trajectory from east to west. This tracker design is unable to track the sun's position due to seasonal changes.

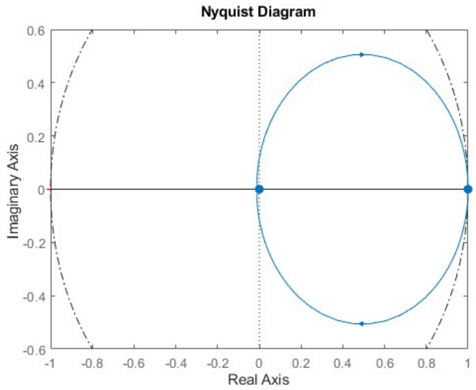
6.2.2. Control System Stability







The Control system is said to have absolute stability and BIBO stability because in the case of absolute stability, the system returns to the steady state after the pulse has been introduced to the system. The system also behaves bounded when a bounded function is introduced.



The Nyquist plot is inside the minimum stability circle, indicating that the system have reached stability as it does not encircle the point $(1, i0)$

Nyquist and Bode Analytical Analysis:

$$T(s) = \frac{3.139 \times 10^5}{8.292 \times 10^{-5}s^3 + 82.92s^2 + 4.415 \times 10^4 s + 3.139 \times 10^5}$$

$$T(i\omega) = \frac{3.139\times10^{5}}{8.292\times10^{-5}(i\omega)^{3}+82.92(i\omega)^{2}+4.415\times10^{4}(i\omega)+3.139\times10^{5}}$$

$$T(i\omega) = \frac{3.139\times10^{5}}{-i8.292\times10^{-5}\omega^{3}-82.92\omega^{2}+i4.415\times10^{4}\omega+3.139\times10^{5}}$$

$$T(i\omega) = \frac{3.139\times10^{5}}{(3.139\times10^{5}-82.92\omega^{2})+i(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}$$

Multiply with the Complex Conjugate

$$T(i\omega) = \frac{3.139\times10^{5}}{(3.139\times10^{5}-82.92\omega^{2})+i(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})} \times\frac{(3.139\times10^{5}-82.92\omega^{2})-i(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}{(3.139\times10^{5}-82.92\omega^{2})-i(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}$$

$$T(i\omega) = \frac{3.139\times10^{5}(3.139\times10^{5}-82.92\omega^{2})-i3.139\times10^{5}(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}{(3.139\times10^{5}-82.92\omega^{2})^{2}-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2}}$$

$$\therefore Re[T(i\omega)] = \frac{3.139\times10^{5}(3.139\times10^{5}-82.92\omega^{2})}{(3.139\times10^{5}-82.92\omega^{2})^{2}-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2}}$$

$$\therefore Im[T(i\omega)] = \frac{-3.139\times10^{5}(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}{(3.139\times10^{5}-82.92\omega^{2})^{2}-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2}}$$

$$\therefore M(\omega) = |T(i\omega)| = \sqrt{Re[T(i\omega)]^{2}+Im[T(i\omega)]^{2}}$$

$$\therefore M(\omega) = |T(i\omega)| = D\sqrt{\frac{(3.139\times10^{5}-82.92\omega^{2})^{2}+(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2}}{((3.139\times10^{5}-82.92\omega^{2})^{2}-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2})^{2}}}$$

$$\therefore m(\omega) = 20\log(D) + 20\log(\sqrt{\frac{(3.139\times10^{5}-82.92\omega^{2})^{2}+(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2}}{((3.139\times10^{5}-82.92\omega^{2})^{2}-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})^{2})^{2}}})$$

$$\therefore \varphi(\omega) = tan^{-1}\left(\frac{Im[T(i\omega)]}{Re[T(i\omega)]}\right)$$

$$\therefore \varphi(\omega) = tan^{-1}(\frac{-(4.415\times10^{4}\omega-8.292\times10^{-5}\omega^{3})}{(3.139\times10^{5}-82.92\omega^{2})})$$

To obtain the Nyquist plot, the Imaginary function must be plotted against the Real function.

The Bode plot is obtained by plotting the m(ω) against Omega and by plotting the phase angle against Omega.

The analytical analysis can be found under the following section In the Matlab code:

```
%% System Evaluation %%
```

6.3. CONTROLLER PERFORMANCE

The controller chosen was the singular Proportional controller. The gain that provided the best response time with the best % overshoot was 4700. The gain introduced enough robustness to the system that 2/3 disturbances introduced, were rejected.

The controller enabled the system to meet the specified requirements. The controller reduced the settling time of the system from 4840s to less than 1.6s. This yielded a big improvement in the performance of the system, while not introducing an overshoot in the step response.

The controller also increased the stability of the system. Before the controller, the stability of the system never converged, making it necessary for a controller to be introduced. After the controller was introduced, the system converged back to its steady state, after a certain signal was introduced.

The behaviour of the controller was tested against three different types of disturbances, and it was found that the controller aided the system in successfully tracking the sun's position during the simulated weather disturbances however, the mechanical disturbance was not that easily overcome.

6.4. INSTRUMENTATION USED TO IMPLEMENT CONTROLLER

To fully implement the PID controller in the vertical single axis solar tracker that is coupled with a potentiometer, instrumentation will be needed. The types of instrumentation are:

1. Solar panels which will be used to absorb the solar energy which will be converted into solar power.
2. A linear actuator to convert the linear motion of the panel to rotational motion.
3. DC motor for control of movement such as rotation of the solar panel.
4. Control panel box to house the controller's electronics.
5. Potentiometer for sensing the solar panel's angular position relative to the standard axis.
6. Temperature sensors to monitor and record the temperature of the solar panel and/or the ambient conditions.
7. Anemometers to measure the wind speed that is a disturbance input.
8. Mechanical switch for manual control in the event of installation or emergency.
9. Display system with simple user interface so that the operators can manually control, set-up and calibrate the solar panel installation.
10. Power inverter to convert the DC electrical power generated by the solar panels (through solar absorption) to AC electricity.
11. Energy monitoring system to keep track out the energy input and out from the solar system.
12. Tracker mount to keep the panel in an inclined position after being oriented to face the sun [17].
13. Driver for control of the motor shaft's rotation [18].

If one of the states is immeasurable via standard instruments, estimation algorithms, iterative learning control algorithms state estimation techniques can be used. The iterative learning control algorithms can track system immeasurable parameters, or performance by integrating the unknown state and then using iterative control measures and algorithms to determine unknown state values. The estimation algorithms can be used via adopting sampling data points just like the aforementioned algorithm. The sampling data will then be run through separately developed algorithms which will use numerical and statistical analysis to approximate the unknown states. The Simulink model can use the control system toolbox to design linear steady-state and time-varying Kalman filters. These techniques can estimate the state values by using the Kalman filters, and/or particle filters The Kalman filters estimate a set of consecutive measurements in a certain period and that information is used to incorporate noises, disturbances and any other inputs that could cause inaccuracy and then produce an estimation of the unknown variable [19].

7. FULL SYSTEM SPECIFICATIONS

Type of tracker: Single Vertical Axis

Controller: Raspberry pi or Arduino

Operation Specifications

Primary axis: Zenith axis (from east to west)

Range of motion: 120°

Temperature Operational Range: 25°C to 35°C (For best performance) [13]

Mechanical Specifications

Dimensions (L×W×D): 1.04 x 1.4 x 0.1 m

Weight: 50 kg

Maximum Surface area of modules: 2.44 m^2

Motor type: DC motor

Operating Voltage: between 12 and 24 V [14]

Number of panels supported: 36

Back EMF constant: 0.07 V/(rad/s)

Torque constant: 0.07

Potentiometer gain: 0.318

Inductance: 1e-5 H

Resistance: 10 Ω

Gear Ratio: 3000

Solar tracker Specifications

Rise Time: 0.968 seconds

Settling Time: 1.725 seconds

Overshoot: 0

7.1. CAPABILITIES AND LIMITATIONS

7.1.1. Capabilities

- Enhanced solar tracking efficiency.
- Higher energy absorption compared a fixed solar panel.
- Increased energy output.
- Less expensive than dual axis solar trackers.
- Simple mechanical design.
- Easy to maintain and install.
- Improved performance through use of potentiometer and PID controller.

7.1.2. Limitations

- Limited range of motion: rotation around azimuth axis
- Limited solar energy captured during winter.
- Mechanical parts are vulnerable to wear and tear.
- Extreme weather conditions can impact solar tracker performance.
- System has its own power consumption.
- The potentiometer has limited accuracy and resolution which can affect overall performance.

REFERENCES

[1] International Trade Administration (2024). *South Africa - Energy*. [online] www.trade.gov. Available at: https://www.trade.gov/country-commercial-guides/south-africa-energy#:~:text=Current%20Status%3A [Accessed 7 May 2024].

[2] Statistics of utility-scale power generation in South Africa .www.csir.co.za. (2023). CSIR releases statistics on power generation in South Africa for 2022 | CSIR. [online] Available at: https://www.csir.co.za/csir-releases-statistics-on-power-generation-south-africa-2022.

[3] mathworks (2021). MATLAB-Simulink-Challenge-Project-Hub/projects/Solar Tracker Control Simulation at main · mathworks/MATLAB-Simulink-Challenge-Project-Hub. [online] GitHub. Available at: https://github.com/mathworks/MATLAB-Simulink-Challenge-Project-Hub/tree/main/projects/Solar%20Tracker%20Control%20Simulation [Accessed 7 May 2024].

[4] sinovoltaics.com. (n.d.). Single Axis Trackers. [online] Available at: https://sinovoltaics.com/learning-center/csp/single-axis-trackers/[Accessed 7 May 2024].

[5] India, R. (2021). Solar Trackers. [online] renewsysworld. Available at: https://www.renewsysworld.com/post/solar-trackers [Accessed 9 May 2024]

[6] sunrise.maplogs.com. (n.d.). Johannesburg, South Africa Sunrise Sunset Times. [online] Available at: https://sunrise.maplogs.com/johannesburg_south_africa.63577.html [Accessed 9 May 2024].

[7] www.maphill.com. (n.d.). Physical Location Map of JOHANNESBURG, within the entire country. [online] Available at: http://www.maphill.com/south-africa/guateng/johannesburg/location-maps/physical-map/entire-country/[Accessed 9 May 2024].

[8] Abood, A.A., 2015. A comprehensive solar angles simulation and calculation using matlab. International Journal of Energy and Environment, 6(4), p.367.

[9] dray (2018). What Is a Solstice and What Is an Equinox (and Why Should I Care)? | The Franklin Institute. [online] fi.edu. Available at: https://fi.edu/en/blog/what-solstice-and-what-equinox-and-why-should-i-care#:~:text=Solstices [Accessed 9 May 2024].

[10] www.climate.top. (n.d.). Sunshine & Daylight Hours in Johannesburg, South Africa Sunlight, Cloud & Day length. [online] Available at: https://www.climate.top/south-africa/johannesburg/sunlight/#:~:text=The%20longest%20day%20of%20the [Accessed 9 May 2024].

[11] Gaisma. (n.d.). Johannesburg, South Africa - Sunrise, sunset, dawn and dusk times for the whole year. [online] Available at: https://www.gaisma.com/en/location/johannesburg.html. [Accessed 9 May 2024]

[12] eepower (n.d.). Potentiometer | Resistor types | Resistor Guide. [online] eepower.com. Available at: https://eepower.com/resistor-guide/resistor-types/potentiometer/#.

[13] www.mathworks.com. (n.d.). Getting Started with Simulink – Solar Tracker. [online] Available at: https://www.mathworks.com/videos/getting-started-with-simulink-69027.html [Accessed 10 May 2024].

[14] (2023). The Impact of Temperature on Solar Panel Efficiency: How Heat Affects your Solar Energy System . EcoFlow [Online]. Available at: https://blog.ecoflow.com/us/effects-of-temperature-on-solar-panel-efficiency/#:~:text=The%20optimal%20temperature%20for%20solar%20panels%20is%20around%2025%C2%B0,%25%2C%20affecting%20overall%20energy%20production. [Accessed 9 May 2024]

[15] Tingey, L. (2020). Understanding Solar Panel Voltage for Better Output. Sunsoaked Solar, [Online] Available at: https://sunsoakedsolar.com/solar-panel-voltage/#:~:text=Solar%20panels%20produce%20DC%20voltage%20that%20ranges%20from,on%20the%20number%20of%20cells%20in%20the%20panel. [Accessed 9 May 2024].

[16] ctms.engin.umich.edu. (n.d.). Control Tutorials for MATLAB and Simulink - Introduction: PID Controller Design. [online] Available at: https://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID#7.

[17] Republic Of Solar. (2022). Solar Panel Tracking Systems- Things You Should Know. [online] Available at: https://arka360.com/ros/solar-panel-tracking-systems/#:~:text=Components%20Of%20A%20Solar%20Tracker [Accessed 17 May 2024].

[18] Republic Of Solar. (2022). Solar Panel Tracking Systems- Things You Should Know. [online] Available at: https://arka360.com/ros/solar-panel-tracking-systems/#:~:text=Components%20Of%20A%20Solar%20Tracker [Accessed 17 May 2024].

[19] www.mathworks.com. (n.d.). State Estimation - MATLAB & Simulink. [online] Available at: https://www.mathworks.com/help/control/state-estimation.html.]

APPENDICES

Appendix A: MATLAB &/OR SIMULINK CODE

A.1. MATLAB CODE

```
%% Group 5
% Mari Norwie
% Kashaka Sithole
% Christy Mkhari
% Ipeleng Tshabalala

% Code 100% Completed on 13/05/2024

syms s k;
stf = tf('s');

% SunPosition Data Sourced from "Getting Started with Simulink Video"
load sunPositionData.mat;
%% System Initialization
%%Shout-out to Matlab starting with Simulink that we can use their system.
% Motor parameters
Kf = 0.07;             % Back EMF cosntant, [V/(rad/s)]
Kt = 0.07;             % Torque constant, [N*m/A]
Kp = 0.318;              % Potentiometer gain,
L = 1e-5;              % Inductance, [H]
R = 10;                % Resistance, [Ohm]
Kg = 3000;                % Gear ratio, []
Td = 44.4;

% Solar panel parameters
m = 50;               % Mass, [kg]
w = 1.04;             % Width, [m]
l = 1.4;              % Length, [m]
d = 0.1;              % Depth, [m]
A = w*l;              % Area, [m^2]
Kd = 5;               % Damping constant, [N*m/(rad/s)]
Ap = 7;
sec = 10;
Lat = 26.2;
n = 172;
D = 23.45*sin((360/365)*(n+284));
local = 6.5;
h = (local-12)*15;
beta =90-(asin(sin(Lat)*sin(D)+cos(Lat)*cos(D)*cos(h))); % Elevation angle, [rad]
J = (m/12)*(l^2*cos(beta)^2+d^2*sin(beta)^2+w^2);
J0 = (m/12)*(l^2*cos(beta)^2+d^2*sin(beta)^2+w^2);

%% Numerators and Denominators Intialization

plant_num = Kg*Kt;
plant_den = (L*stf+R)*(J0*stf^2 + Kd*stf)+Kg^2*Kt*Kf*stf;

Motor_Num = 1;
Motor_Den = [L R];

Panel_Num = 1;
Panel_Den_lin = [J0 Kd];
```

```
%% Denominator Coefficients

A = 8.292e-05;
B = 82.92;
C = 4.415e04;
D = 66.78;
den_Open = [A B C 0];
den_Feedback = [A B C D];

%% Transfer Functions

plant_TF = plant_num/plant_den;
Open_System_TF = Kp*plant_TF;
Cl_system_TF = feedback(Kp*plant_TF,1);


%% Time-domain

figure(1); subplot(2,1,1); step(Open_System_TF); grid on; legend('Open-Loop
System');
subplot(2,1,2); step(Cl_system_TF); grid on; legend('Feedback-Loop System');
stepinfo(Open_System_TF);
stepinfo(Cl_system_TF);

%% Frequency-domain

figure(2); bode(Open_System_TF); grid on; legend('Open-Loop System');
figure(3); bode(Cl_system_TF); grid on; legend('Feedback-Loop System');
figure(4); margin(Cl_system_TF);grid on;
[Gm,Pm,Wcg,Wcp] = margin(Cl_system_TF);

%% Frequency-domain Analytical Calculations
% Code Inspiration from CH6_Stability_examples_Nyquist_GM_PM.m

w = -500:0.1:500;
Re_Freq = (66.78*(66.78-82.92*w.^2))./((66.78-82.92*w.^2).^2-(44150*w-8.292e-
05*w.^3).^2);
Im_Freq = (-66.78*(44150*w-8.292e-05*w.^3))./((66.78-82.92*w.^2).^2-(44150*w-
8.292e-05*w.^3).^2);
Mag_Freq = sqrt(Re_Freq.^2+Im_Freq.^2);
Phi_Freq = atan(Im_Freq./Re_Freq);
figure; plot(w,Mag_Freq); xlabel('Frequency');ylabel('Magnitude');
figure; plot(w,Phi_Freq); xlabel('Frequency');ylabel('Phase Angle');

%% Routh-Hurwits

Routh_Open = Routh_Hurwitz(den_Open);
Routh_Feedback = Routh_Hurwitz(den_Feedback);

%% Nyquist Criterion

figure(4); nyquist(Open_System_TF); grid on; legend('Open-Loop System');
figure(5); nyquist(Cl_system_TF); grid on; legend('Feedback-Loop System');

%% Nyquist Analytical Analysis
% Code Inspiration from CH6_Stability_examples_Nyquist_GM_PM.m

w = -500:0.01:500;
```

```
Re_Nyq = (D*(D-B*w.^2))./((D-B*w.^2).^2-(C*w-A*w.^3).^2);
Im_Nyq = (-D*(C*w-A*w.^3))./((D-B*w.^2).^2-(C*w-A*w.^3).^2);
figure;     plot(Re_Nyq,Im_Nyq);     xlabel('Real-Axis');ylabel('Imaginary-Axis');
title('Analytical Nyquist Plot');

%% BIBO Stability

pole_Open = pole(Open_System_TF);
pole_Feedback = pole(Cl_system_TF);

%% State-Space

Linear_system = ss(A,B,C,D);
pole_sys = pole(Linear_system);

%% PID Selection
%% Proportional Control
t = 0:0.5:10;
KP = 100;
C_P = pid(KP);
sys_feedback_P1 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P1);
stepinfo(sys_feedback_P1);

KP = 500;
C_P = pid(KP);
sys_feedback_P2 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P2);
stepinfo(sys_feedback_P2);

KP = 1000;
C_P = pid(KP);
sys_feedback_P3 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P3);
stepinfo(sys_feedback_P3);

KP = 2000;
C_P = pid(KP);
sys_feedback_P4 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P4);
stepinfo(sys_feedback_P4);

KP = 3000;
C_P = pid(KP);
sys_feedback_P5 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P5);
stepinfo(sys_feedback_P5);

KP = 4000;
C_P = pid(KP);
sys_feedback_P6 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P6);
stepinfo(sys_feedback_P6);

KP = 4500;
C_P = pid(KP);
sys_feedback_P7 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P7);
stepinfo(sys_feedback_P7);
```

```
KP = 4700;
C_P = pid(KP);
sys_feedback_P8 = feedback(Kp*C_P*plant_TF,1);
step(sys_feedback_P8);
stepinfo(sys_feedback_P8);

figure(6);        subplot(2,2,1);        step(sys_feedback_P1,t);hold        on;
stepplot(sys_feedback_P2,t,'r--'); grid on; legend('Kp = 100', 'Kp = 500');
subplot (2,2,2);step(sys_feedback_P3,t); hold on; stepplot(sys_feedback_P4,t,'r--
'); grid on; legend('Kp = 1000','Kp = 2000');
subplot (2,2,3);step(sys_feedback_P5,t); hold on; stepplot(sys_feedback_P6,t,'r--
'); grid on; legend('Kp = 3000','Kp = 4000');
subplot (2,2,4);step(sys_feedback_P7,t); hold on; stepplot(sys_feedback_P8,t,'r--
'); grid on; legend('Kp = 4500','Kp = 4700');

%% Proportional & Integral Control
KP = 240;
KI = 180;
C_PI = pid(KP,KI);
sys_feedback_PI1 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI1);
stepinfo(sys_feedback_PI1);

KP = 500;
KI = 180;
C_PI = pid(KP,KI);
sys_feedback_PI2 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI2);
stepinfo(sys_feedback_PI2);

KP = 500;
KI = 50;
C_PI = pid(KP,KI);
sys_feedback_PI3 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI3);
stepinfo(sys_feedback_PI3);

KP = 900;
KI = 50;
C_PI = pid(KP,KI);
sys_feedback_PI4 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI4);
stepinfo(sys_feedback_PI4);

KP = 1000;
KI = 50;
C_PI = pid(KP,KI);
sys_feedback_PI5 = feedback(Kp*C_PI*plant_TF,1);
step(sys_feedback_PI5);
stepinfo(sys_feedback_PI5);

KP = 1500;
KI = 50;
C_PI = pid(KP,KI);
sys_feedback_PI6 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI6);
stepinfo(sys_feedback_PI6);
```

```
KP = 3000;
KI = 100;
C_PI = pid(KP,KI);
sys_feedback_PI7 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI7);
stepinfo(sys_feedback_PI7);

KP = 4700;
KI = 100;
C_PI = pid(KP,KI);
sys_feedback_PI8 = feedback(Kp*C_PI*plant_TF, 1);
step(sys_feedback_PI8);
stepinfo(sys_feedback_PI8);


figure(7);         subplot(2,2,1);         step(sys_feedback_PI1,t);hold         on;
stepplot(sys_feedback_PI2,t,'r--'); grid on; legend('Kp = 240 Ki = 180', 'Kp = 500
Ki = 180');
subplot (2,2,2); step(sys_feedback_PI3,t); hold on; stepplot(sys_feedback_PI4,t,'r-
-'); grid on; legend('Kp = 500 Ki = 50','Kp = 900 Ki = 50');
subplot (2,2,3); step(sys_feedback_PI5,t); hold on; stepplot(sys_feedback_PI6,t,'r-
-'); grid on; legend('Kp = 1000 Ki = 50','Kp = 1500 Ki = 50');
subplot (2,2,4); step(sys_feedback_PI7,t); hold on; stepplot(sys_feedback_PI8,t,'r-
-'); grid on; legend('Kp = 3000 Ki = 100','Kp = 4700 Ki = 100');

%% Proportional & Derivative Control

KP = 500;
KI = 0;
KD = 100;
C_PD = pid(KP,KI,KD);
sys_feedback_PD1 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD1);
stepinfo(sys_feedback_PD1);

KP = 700;
KI = 0;
KD = 100;
C_PD = pid(KP,KI,KD);
sys_feedback_PD2 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD2);
stepinfo(sys_feedback_PD2);

KP = 900;
KI = 0;
KD = 100;
C_PD = pid(KP,KI,KD);
sys_feedback_PD3 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD3);
stepinfo(sys_feedback_PD3);

KP = 4700;
KI = 0;
KD = 100;
C_PD = pid(KP,KI,KD);
sys_feedback_PD4 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD4);
stepinfo(sys_feedback_PD4);
```

```
KP = 4700;
KI = 0;
KD = 50;
C_PD = pid(KP,KI,KD);
sys_feedback_PD5 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD5);
stepinfo(sys_feedback_PD5);

KP = 4700;
KI = 0;
KD = 20;
C_PD = pid(KP,KI,KD);
sys_feedback_PD6 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD6);
stepinfo(sys_feedback_PD6);

KP = 4700;
KI = 0;
KD = 10;
C_PD = pid(KP,KI,KD);
sys_feedback_PD7 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD7);
stepinfo(sys_feedback_PD7);

KP = 4700;
KI = 0;
KD = 5;
C_PD = pid(KP,KI,KD);
sys_feedback_PD8 = feedback(Kp*C_PD*plant_TF,1);
step(sys_feedback_PD8);
stepinfo(sys_feedback_PD8);

figure(8); subplot(2,2,1); step(sys_feedback_PD1,t);hold on; stepplot(sys_feedback_PD2,t,'r--'); grid on; legend('Kp = 500 Kd = 100', 'Kp = 700 Kd = 100');
subplot (2,2,2); step(sys_feedback_PD3,t); hold on; stepplot(sys_feedback_PD4,t,'r--'); grid on; legend('Kp = 900 Kd = 100','Kp = 4700 Kd = 100');
subplot (2,2,3); step(sys_feedback_PD5,t); hold on; stepplot(sys_feedback_PD6,t,'r--'); grid on; legend('Kp = 4700 Kd = 100','Kp = 4700 Kd = 20');
subplot (2,2,4); step(sys_feedback_PD7,t); hold on; stepplot(sys_feedback_PD8,t,'r--'); grid on; legend('Kp = 4700 Kd = 10','Kp = 4700 Kd = 5');

%% Proportional, Integral and Derivative Control

KP = 500;
KI = 40;
KD = 100;
C_PID = pid(KP,KI,KD);
sys_feedback_PID1 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID1);
stepinfo(sys_feedback_PID1);

KP = 900;
KI = 40;
KD = 90;
C_PID = pid(KP,KI,KD);
sys_feedback_PID2 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID2);
stepinfo(sys_feedback_PID2);
```

```
KP = 1000;
KI = 50;
KD = 90;
C_PID = pid(KP,KI,KD);
sys_feedback_PID3 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID3);
stepinfo(sys_feedback_PID3);

KP = 3000;
KI = 50;
KD = 50;
C_PID = pid(KP,KI,KD);
sys_feedback_PID4 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID4);
stepinfo(sys_feedback_PID4);

KP = 3500;
KI = 70;
KD = 30;
C_PID = pid(KP,KI,KD);
sys_feedback_PID5 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID5);
stepinfo(sys_feedback_PID5);

KP = 4000;
KI = 80;
KD = 20;
C_PID = pid(KP,KI,KD);
sys_feedback_PID6 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID6);
stepinfo(sys_feedback_PID6);

KP = 4500;
KI = 90;
KD = 10;
C_PID = pid(KP,KI,KD);
sys_feedback_PID7 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID7);
stepinfo(sys_feedback_PID7);

KP = 4700;
KI = 100;
KD = 5;
C_PID = pid(KP,KI,KD);
sys_feedback_PID8 = feedback(Kp*C_PID*plant_TF,1);
step(sys_feedback_PID8);
stepinfo(sys_feedback_PID8);

figure(9);      subplot(2,2,1);       step(sys_feedback_PID1,t);hold       on;
stepplot(sys_feedback_PID2,t,'r--'); grid on; legend('Kp = 500 Ki = 40 Kd = 100',
'Kp = 700 Ki = 40 Kd = 90');
subplot          (2,2,2);step(sys_feedback_PID3,t);          hold          on;
stepplot(sys_feedback_PID4,t,'r--'); grid on; legend('Kp = 1000 Ki = 50 Kd = 90','Kp
= 3000 Ki = 50 Kd = 50');
subplot          (2,2,3);step(sys_feedback_PID5,t);          hold          on;
stepplot(sys_feedback_PID6,t,'r--'); grid on; legend('Kp = 3500 Ki = 70 Kd = 30','Kp
= 4000 Ki = 80 Kd = 20');
```

```
subplot (2,2,4);step(sys_feedback_PID7,t); hold on;
stepplot(sys_feedback_PID8,t,'r--'); grid on; legend('Kp = 4500 Ki = 90 Kd = 10','Kp
= 4700 Ki = 100 Kd = 5');

figure(10); subplot(2,2,1); step(sys_feedback_P8,t);grid on; legend('Kp = 4700');
subplot (2,2,2); step(sys_feedback_PI8,t); grid on; legend('Kp = 4700 Ki = 100');
subplot (2,2,3); step(sys_feedback_PD5,t); grid on; legend('Kp = 4700 Kd = 50');
subplot (2,2,4); step(sys_feedback_PID7,t); grid on; legend('Kp = 4500 Ki = 90 Kd
= 10');

%% Root Locus

figure(11); rlocus(Cl_system_TF); legend('Feedback-Loop System');


%% Obtaining Simulink data

% Time Variable
time = out.tout;

% Linear Vs Non-linear with Various inputs
StepLin = out.Linear_Step_Out;
StepNLin = out.Non_Linear_Step_Out;
RampLin = out.Linear_Ramp_Out;
RampNLin = out.Non_Linear_Ramp_Out;
SineLin = out.Linear_Sine_Out;
SineNLin = out.Non_Linear_Sine_Out;
figure(12); plot(time,StepLin,'g'); hold on; plot(time,StepNLin,'r--'); grid on;
title('Step Input Linear vs Non-Linear'); legend('Linear','Non-Linear');
xlabel('Time (s)'); ylabel('Amplitude');
figure(13); plot(time,RampLin,'g'); hold on; plot(time,RampNLin,'b--'); grid on;
title('Ramp Input Linear vs Non-Linear'); legend('Linear','Non-Linear');
xlabel('Time (s)'); ylabel('Amplitude');
figure(14); plot(time,SineLin,'g'); hold on; plot(time,SineNLin,'m--'); grid on;
title('Sine Input Linear vs Non-Linear'); legend('Linear','Non-Linear');
xlabel('Time (s)'); ylabel('Amplitude');

% Absolute Stability
Pulse = out.Abs_Stability_feedback_pulse;
figure(15); plot(time, Pulse); title('Absolute Stability with Pulse Wave'); grid
on;

% BIBO Stability
SineW = out.BIBO_Stability_feedback_Sin;
Square = out.BIBO_Stability_feedback_Square;
figure(16); plot(time,SineW); grid on; title('BIBO Stability with Sine Wave Input')
figure(17); plot(time,Square); grid on; title('BIBO Stability with Square Wave
Input');

% System Behaviour with P Controller - No Disturbance
AngleIn = out.Linear_PID_In_B;
AngleOut = out.Linear_PID_OUT_B;
figure(18); plot(time, AngleIn,'g'); hold on; plot(time, AngleOut,'r--'); grid on;
title('Feedback System Behaviour with P Contoller - No Disturbance - Winter Time');
legend('Sun Position','Panel Tracking'); xlabel('Time (s)'); ylabel('Sun Position
(rad)');

% System Behaviour with P Controller - No Disturbance - Non-Linear
AngleIn = out.Non_Linear_PID_In_B;
```

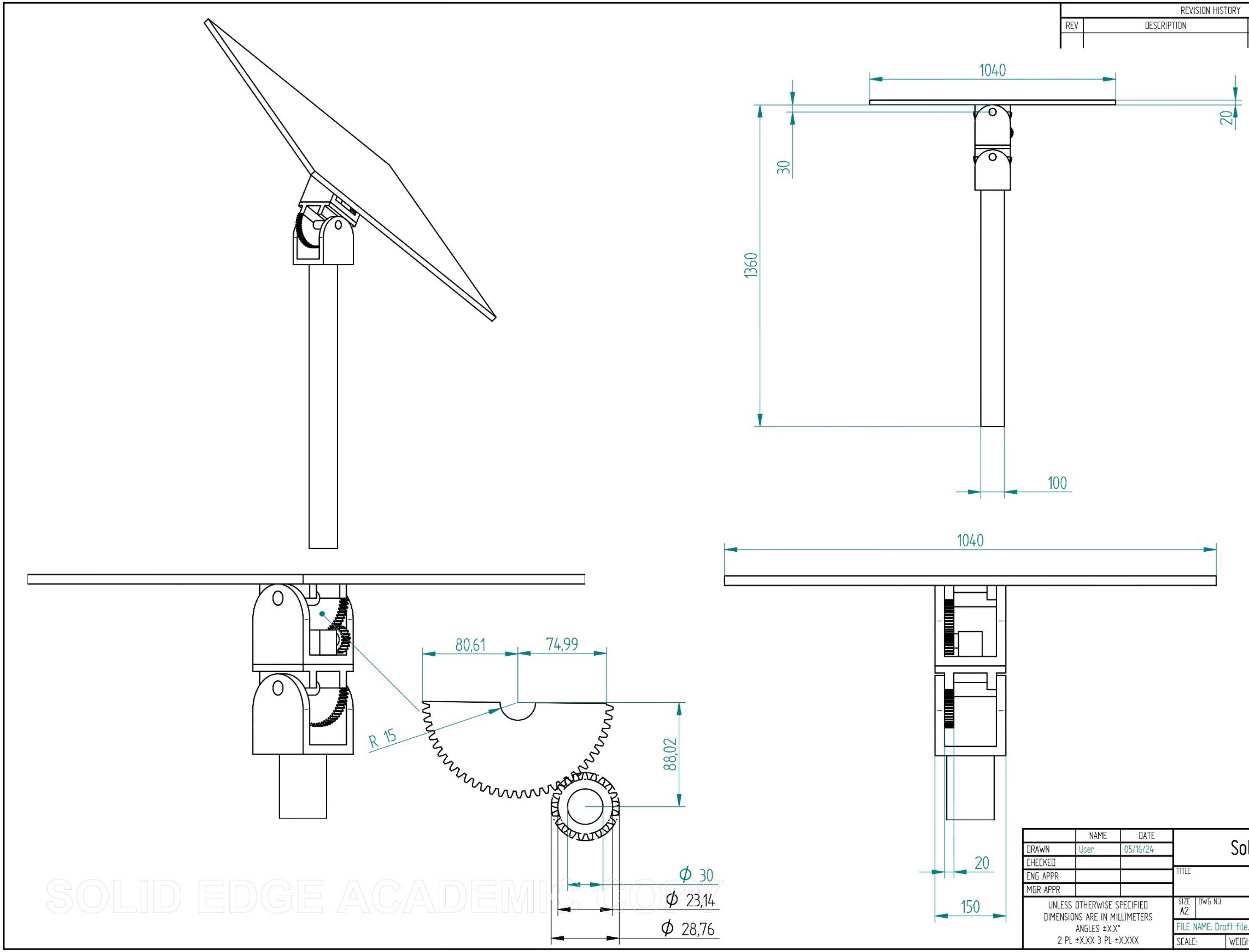
```
AngleOut = out.Non_Linear_PID_OUT_B;
figure(19); plot(time, AngleIn,'g'); hold on; plot(time, AngleOut,'r--'); grid on;
title('Non-Linear Feedback System Behaviour with P Contoller - No Disturbance -
Winter Time');
legend('Sun Position','Panel Tracking'); xlabel('Time (s)'); ylabel('Sun Position
(rad)');

% System Behaviour with Constant Disturbance
AngleIn = out.Linear_PID_In_ConstD;
AngleOut = out.Linear_PID_Out_ConstD;
figure(20); plot(time,AngleIn,'g'); hold on; plot(time, AngleOut,'m--'); grid on;
title('Feedback System Behaviour with P Controller and Constant Disturbance - Winter
Time');
legend('Sun Position','Panel Tracking'); xlabel('Time (s)'); ylabel('Sun Position
(rad)');

% System Behaviour with Pulse Disturbance
AngleIn = out.Linear_PID_In_PulseD;
AngleOut = out.Linear_PID_Out_PulseD;
figure(21); plot(time,AngleIn,'g'); hold on; plot(time, AngleOut,'b--'); grid on;
title('Feedback System Behaviour with P Controller and Pulse Disturbance - Winter
Time');
legend('Sun Position','Panel Tracking'); xlabel('Time (s)'); ylabel('Sun Position
(rad)');

% System Behaviour with Sine Wave Disturbance
AngleIn = out.Linear_PID_In_SineD;
AngleOut = out.Linear_PID_Out_SineD;
figure(22); plot(time,AngleIn,'g'); hold on; plot(time, AngleOut,'k--'); grid on;
title('Feedback System Behaviour with P Controller and Sine Wave Disturbance -
Winter Time');
legend('Sun Position','Panel Tracking'); xlabel('Time (s)'); ylabel('Sun Position
(rad)');

%% System Evaluation

SYSTEM = sys_feedback_P8;
figure; nyquist(SYSTEM);
figure; margin(SYSTEM);

TIME = out.tout;
AbsStab = out.Abs_Stability_SYSTEM_pulse;
BIBOSin = out.BIBO_Stability_SYSTEM_Sin;
BIBOSqu = out.BIBO_Stability_SYSTEM_Square;
figure; plot(TIME, AbsStab); title('Controlled System Absolute Stability with Pulse
Signal'); xlabel('Time(s)'); ylabel('Amplitude');
figure; plot(TIME, BIBOSin); title('Controlled System BIBO Stability with Sine
Wave'); xlabel('Time(s)'); ylabel('Amplitude');
figure; plot(TIME, BIBOSqu); title('Controlled System Absolute Stability with Square
Wave'); xlabel('Time(s)'); ylabel('Amplitude');

W = -60:0.5:60;
RE_Nyq = (3.139e05*(3.139e05-82.92*W.^2))./((3.139e05-82.92*W.^2).^2-(4.415e04*W-
8.292e-05*W.^3).^2);
IM_Nyq = (-3.139e05*(4.415e04*W-8.292e-05*W.^3))./((3.139e05-82.92*W.^2).^2-
(4.415e04*W-8.292e-05*W.^3).^2);
figure; plot(RE_Nyq,IM_Nyq); xlabel('Real-Axis');ylabel('Imaginary-Axis');
title('Analytical Nyquist Plot');
MAG = sqrt(RE_Nyq.^2+IM_Nyq.^2);
```

```
PHI = atan(IM_Nyq./RE_Nyq);
figure; plot(W, m);xlabel('Frequency'); ylabel('Magnitude');
figure; plot(W, PHI);xlabel('Frequency'); ylabel('Phase Angle');


%% Routh-Hurwitz Table Function
%%Shoutout to Joe Schenkel for this Routh-Hurwitz Table code that works
%%perfectly%%
function [routh_table] = Routh_Hurwitz(Den)

syms e;              %create a symbol for epsilon (in case zero's show up in Routh
Table)
syms k1;            %create symbol for K (in case user enters that value)
syms k2;
eps_flag = 0;       %flag for if a first column zero appears (e replaces it)
row_of_zeros = 0;   %variable for the power of s in which a row of zeros occured
constant_flag = 0;  %flag for if user is using symbolic constants

den_poly = Den;
fprintf('\r\n');

%%%%%%%%%%%%%%%%%%% Create the first 2 rows from input information %%%%%%%%%%%%%%
column_size = length(den_poly);
routh_table_1 = den_poly(1);
for x = 3 : 2 : column_size
    routh_table_1 = [routh_table_1, den_poly(x)];
end

routh_table_2 = den_poly(2);


for x = 4 : 2 : column_size
    routh_table_2 = [routh_table_2, den_poly(x)];
end

if(rem(column_size, 2) == 1)  % even polynomial
    routh_table_2 = [routh_table_2, 0];
    new_column_size = ((column_size + 1) / 2) - 1;   % 3rd column size for even
polynomials
else
    new_column_size = (column_size / 2) - 1;  % 3rd column size for odd polynomials
end

routh_table = [routh_table_1; routh_table_2];


%%%%%%%%%%%%%%%%%%% First 2 rows created, now finish the table %%%%%%%%%%%%%%

for z = 1 : 1 : column_size - 2  % each loop is a new row
    determinant   =   [routh_table(z,1)   routh_table(z,2);   routh_table(z+1,   1)
routh_table(z+1,2)];
    routh_table_temp = - (det(determinant) / routh_table(z+1,1));

    if routh_table_temp == 0    %% Test to see if zero will be in table
        routh_table_temp = e;   %% Replace the zero with epsilon
        eps_flag = 1;
    end
    routh_table_new_row = routh_table_temp;
    for x = 2 : 1 : new_column_size     % Each loop is for a column in the table
        determinant = [routh_table(z,1) routh_table(z,x+1); routh_table(z+1,1)
routh_table(z+1,x+1)];
        routh_table_temp = -(det(determinant) / routh_table(z+1,1));
        routh_table_new_row = [routh_table_new_row, routh_table_temp];
    end

    for y = length(routh_table_new_row) : 1 : new_column_size   % Fill the last
columns with zeros
        routh_table_new_row = [routh_table_new_row, 0];
    end

    if row_of_zeros == 0   %% These if statements are for determining if a row of
zeros has appeared
        if eps_flag == 1
            if routh_table_new_row(1, 2:new_column_size) ==
zeros(1,(new_column_size-1))
                row_of_zeros = column_size - 2 - z;
            end
        end
    end
    routh_table = [routh_table; routh_table_new_row];   % Add the new row to the
table
end

%%%%%%%%%%%% Now it's time to display the Routh Table (and warnings) %%%%%%%%%%%
fprintf('\r\n');
fprintf(' Routh Table:\r\n\r\n');
if row_of_zeros > 0  % a row of zeros occurred, alert user of where and what to do
    disp(' This means you might have poles on the jw-axis');
    fprintf('\r\n');
    disp(' To complete the table, please do the following (if necessary):');
    disp(sprintf('\t1.  Please differentiate the s^%d row', (row_of_zeros+1)));
    disp(sprintf('\t2.  Substitute answer in s^%d row', row_of_zeros));
    disp(sprintf('\t3.  Re-run program with s^%d as the highest order of the
polynomial', (row_of_zeros+1)));
    disp(sprintf('\t    and the differentiated answer as the s^%d row.  \r\n\r\n',
(row_of_zeros)));
end

sym(routh_table);
disp(sprintf(' Highest order of Polynomial Entered = s^%d',(column_size-1)));

if constant_flag == 1        %Use pretty print if constants used
    pretty(routh_table);
elseif eps_flag == 1         %Use pretty print if symbol 'e' used
    pretty(routh_table);
else                         %Use regular print otherwise
    disp(routh_table);
end
fprintf('\r\n\r\n\r\n');

end
```

Appendix B: CAD

REVISION HISTORY

REV	DESCRIPTION	DATE	APPROVED



	NAME	DATE
DRAWN	User	05/16/24
CHECKED		
ENG APPR		
MGR APPR		

UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN MILLIMETERS
ANGLES ±X.X°
2 PL ±X.XX 3 PL ±X.XXX

Solid Edge

TITLE

SIZE	DWG NO	REV
A2		

FILE NAME: Draft file.dft

SCALE:	WEIGHT:	SHEET 1 OF 1