1. Describe in detail how a PID controller works **conceptually** in the context of stabilizing the quadcopter yaw **rate**. What do each of the gains do?

The y component(down) data from the gyroscope must be ran through a second order

Butterworth filter. The PID yaw rate controller will have three components, proportional, integral, and derivative, and use the filtered yaw rate data and a desired yaw rate variable. The error is the difference between the desired yaw rate and actual filtered yaw rate. The error will be used in the controller. To yaw a plus configuration quadcopter motor commands must be sent to motors 1(north), motors 2(south), motors 3(west), and motor 4(east).

Diagram

Description automatically generated

Motor Commands:

Each controller component has a scalar component, and (gains). The proportional

controller is the product of and the error. The integral controller is the product of and the

integral of the error. The derivative controller is the product of and the derivative of the error. The purpose of the proportional controller is to increase/decrease the rise time. Increasing will increase % overshoot and decreasing will decrease % overshoot. The purpose of the integral controller is to reduce steady state error. Increasing will increase % overshoot and

decreasing will decrease % overshoot. The purpose of the derivative controller is to reduce %

overshoot. Increasing will decrease rise time and decreasing will increase rise time.

1. Describe in detail how to experimentally tune the yaw rate stabilization controller using open loop gain tuning (tangent method).

Once the controllers are built a gain tuning method (tangent method) must be implemented to establish starting and values. To perform the Ziegler-Nichols tangent method throttle is set to 50% for all motors, then 20% is subtracted to motors 1 and 2, and 20% is added to motors 3 and 4.

From a graph of the response variables L, τ, and K can be determined. L is the dead time, τ is the time constant, and K is the gain. With these variables and a few sets of equations starting and values can be determined. From here set the desired rate to 60 and increased and/or decreased and according to the and relations stated above until the following target parameters are met.

Parameters:

* Percent Overshoot 20%
* Settling Time 3 second
* Rise Time 2 seconds

1. Show the estimated PID gains from the open loop gain tuning. Make sure to show a plot of the dat used to calculate the gains.

Chart, line chart

Description automatically generated

The graph above is from a step input of 20 = A. Variables L, τ, and K can be determined from the data.

225 deg/s = KA

K = 11.25

The Ziegler-Nichols open loop method table can be determined from variables L, τ, and K:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Kp** | **Ti** | **Td** | **tau** | **L** | **K** |
| **P** |  | - | - | 1.406066 | 0.109983 | 11.25 |
| **PI** | (.9) | (3)L | - |  |  |  |
| **PID** | (1.2) | (2)L | (.5)L |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Kp** | **Ti** | **Td** |  | **L** | **K** |
| **P** | 1.136390977 | - | - | 1.406066 | 0.109983 | 11.25 |
| **PI** | 1.022751879 | 0.329939 | - |  |  |  |
| **PID** | 1.363669172 | 0.219966 | 0.054992 |  |  |  |

From the table and equation above starting are determined for P, PI, PID controllers:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| P controller: | | PI controller: |  |  | PID controller: | |  |
| **K**P |  | **K**P | KI |  | **K**P | KI | **KD** |
| 1.1364 |  | 1.0228 | 3.0998 |  | 1.3637 | 1.5000 | 0.0750 |

1. Conduct a step response where the quadcopter goes from a desired yaw rate of zero to 60   
   degrees/second. Show the results (clearly indicate where the step input is started) and   
   estimate the percent overshoot, settling time, and rise time (time to go from 0 to 60   
   degrees/second).

The Yaw Rate vs. Time graph above is from a PD controller with and values of 2.7 and 1.8 respectively. The highest peak is the second peak which is about 90 deg/s which is about 50% overshoot. Rise time is about 1 seconds. The settling time is approximately 2 seconds. The yaw rate has a large steady state error of almost 15 deg/s.

1. In order to safely fly a quadcopter, your attitude stabilization controller needs to be fast and   
   accurate. Modify your gains until you have met the required performance for a 60   
   degree/second yaw rate step response. **Percent overshoot < 20%, settling time < 3.0s**,   
   and **rise time < 2.0s**. Show the results, with the step input time clearly indicated. Make sure   
   to provide your estimated percent overshoot, settling time, and rise time. Remember, you   
   want your quadcopter to be as stable as possible (you will be performing free-flight testing   
   very soon and the more stable your controller is the easier it will be to fly your quadcopter).

The Yaw Rate vs. Time graph above is from a PD controller with and values of 3.4 and 2.72 respectively. Although this does not meet all the required parameters, it nearly meets them and is the closest I’ve got on any of my pitch/roll/yaw controllers. The highest peak is the third peak which is about 75 deg/s which is about 25% overshoot. Rise time is about .7 seconds. The settling time is approximately 2 seconds. It’s hard to tell if there is any steady state error. It appears to have a steady state error of 2 deg/s.