A ROBUST-ADAPTIVE CONTROL STRATEGY FOR GEAR BEARING DRIVE SYSTEMS

A Thesis Presented

By

Nianyu Jiang

to

The Department of Mechanical and Industrial Engineering

in partial fulfillment of the requirements

for the degree of

Master of Science

in the field of

Mechanical Engineering

Northeastern University
Boston, Massachusetts

April 2018
ABSTRACT

In modern industrial engineering such as space robots, assembly systems and transportation vehicles, a Gear Bearing Drive (GBD) transmission, with the potential to produce up to 5000:1 torque ratio in a compact size, can be of great interest. However, in a practical actuation system consisting of actuator, GBD and workload, ever-present nonlinearities such as disturbances and perturbations can increase the complexity of the modeling process as well as difficulty of the control design. To reduce such complexities, a simplified model of the actuation plant is developed by linearizing all nonlinearities and numerically estimating the disturbances and perturbations inside the plant. For this, a robust-adaptive control platform, especially designed to deal with system uncertainties and unmodeled dynamics, is selected, analyzed and implemented in real-time. In addition, due to the initially unknown parameters, the adaptive part of the controller must adapt to the system parameters variations and changes. Finally, to guarantee asymptotic convergence, a projection operator is developed and combined with the adaptation law. Furthermore, an optimal combination of proportional-integral-derivative (PID) and robust-adaptive sliding mode control (RASMC) is selected and a novel controller is developed and experimentally implemented and verified.

The experimental results of system response are compared to the simulation results to assess control performance and response characteristics. This research suggests that RASMC can ensure its effectiveness in a nonlinear plant with uncertain parameters and unmodeled dynamics and produce a satisfactory experimental performance with minimum settling time and overshoot. Future work is required to explain unpredictable piecewise behavior of the open-loop speed response and other ignored modeling aspects and nonlinearities.
Table of Contents:

Chapter 1. Introduction ............................................................................................................. 1
  1.1 Background ....................................................................................................................... 1
  1.2 Previous Work on GBD .................................................................................................. 2
    1.2.1 Previous Work by Elias Brassitos ............................................................................ 2
    1.2.2 Previous Work by Kwanghyeon Cho ................................................................. 2
  1.3 Thesis Contribution ......................................................................................................... 3
  1.4 Thesis at a Glance ............................................................................................................ 4

Chapter 2. Open-Loop Model ................................................................................................. 6
  2.1 Open-Loop Experiment Setup ...................................................................................... 6
    2.1.1 Experimental Devices Setup ................................................................................. 6
    2.1.2 Experimental Software Setup ................................................................................ 11
  2.2 Open Loop Experimental Description ......................................................................... 14
    2.2.1 Modeling Method ..................................................................................................... 14
  2.3 Experimental Data Analysis .......................................................................................... 15
    2.3.1 Experimental Data Display .................................................................................. 15
    2.3.2 Data Analysis .......................................................................................................... 17
    2.3.3 Conclusion about Experimental Data ................................................................. 19
  2.4 Model Reconstruction .................................................................................................... 19
    2.4.1 Simplified System Dynamic Model ...................................................................... 19
    2.4.2 Regions of System Reconstruction ....................................................................... 22
    2.4.3 System Reconstruction in Region 1 ....................................................................... 22
    2.4.4 System Reconstruction in Region 2 ....................................................................... 25
  2.5 Model Evaluation ............................................................................................................ 26
  2.6 Conclusion ....................................................................................................................... 28

Chapter 3. PID Controller Design ......................................................................................... 29
  3.1 System Analysis .............................................................................................................. 29
  3.2 Traditional Pole-Zero Cancellation for Separate Regions ........................................... 30
List of Figures:

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>GBD Transmission System</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>Block Diagram of Experimental System</td>
<td>7</td>
</tr>
<tr>
<td>2.3</td>
<td>dSPACE R&amp;D DS1104 Controller Board</td>
<td>8</td>
</tr>
<tr>
<td>2.4</td>
<td>Black Box</td>
<td>10</td>
</tr>
<tr>
<td>2.5</td>
<td>DriveWare Interface</td>
<td>11</td>
</tr>
<tr>
<td>2.6</td>
<td>Open Loop Experiment Simulink Model</td>
<td>12</td>
</tr>
<tr>
<td>2.7</td>
<td>Control Desk</td>
<td>13</td>
</tr>
<tr>
<td>2.8</td>
<td>Motor Speed Response of Open Loop System</td>
<td>16</td>
</tr>
<tr>
<td>2.9</td>
<td>Motor Torque Response</td>
<td>17</td>
</tr>
<tr>
<td>2.10</td>
<td>Brake Torque Response</td>
<td>18</td>
</tr>
<tr>
<td>2.11</td>
<td>Total Torque Response</td>
<td>18</td>
</tr>
<tr>
<td>2.12</td>
<td>Schematic of Lumped Parameter System</td>
<td>20</td>
</tr>
<tr>
<td>2.13</td>
<td>Two Parts of Lumped Parameter System</td>
<td>20</td>
</tr>
<tr>
<td>2.14</td>
<td>Schematic of One-Degree-of-Freedom System</td>
<td>22</td>
</tr>
<tr>
<td>2.15</td>
<td>Current-Total Torque Curve</td>
<td>23</td>
</tr>
<tr>
<td>2.16</td>
<td>Speed-Total Torque Curve</td>
<td>24</td>
</tr>
<tr>
<td>2.17</td>
<td>Current-Total Torque Curve</td>
<td>25</td>
</tr>
<tr>
<td>2.18</td>
<td>Speed-Total Torque Curve</td>
<td>26</td>
</tr>
<tr>
<td>2.19</td>
<td>Simulink Model of Open Loop System</td>
<td>27</td>
</tr>
<tr>
<td>2.20</td>
<td>Simulink Result of Speed Response</td>
<td>28</td>
</tr>
<tr>
<td>3.1</td>
<td>Simulation Model for Region 1</td>
<td>33</td>
</tr>
<tr>
<td>3.2</td>
<td>PID Parameter for Region 1</td>
<td>34</td>
</tr>
<tr>
<td>3.3</td>
<td>PID Parameter for Region 2</td>
<td>34</td>
</tr>
<tr>
<td>3.4</td>
<td>PID Parameter for Region 2</td>
<td>34</td>
</tr>
<tr>
<td>3.5</td>
<td>Speed Response for PID Controller on Region 1</td>
<td>35</td>
</tr>
<tr>
<td>3.6</td>
<td>Control Signal for PID Controller on Region 1</td>
<td>36</td>
</tr>
<tr>
<td>3.7</td>
<td>Speed Response for PID Controller on Region 2</td>
<td>36</td>
</tr>
<tr>
<td>3.8</td>
<td>Control Signal for PID Controller on Region 2</td>
<td>37</td>
</tr>
<tr>
<td>3.9</td>
<td>Simulink Model of Close Loop System</td>
<td>38</td>
</tr>
<tr>
<td>3.10</td>
<td>Subsystem inside Piecewise Model Block</td>
<td>38</td>
</tr>
<tr>
<td>3.11</td>
<td>Speed Response for First Set of PID Parameter</td>
<td>39</td>
</tr>
<tr>
<td>3.12</td>
<td>Control Signal for First Set of PID Parameter</td>
<td>40</td>
</tr>
<tr>
<td>3.13</td>
<td>Speed Response for Second Set of PID Parameter</td>
<td>40</td>
</tr>
<tr>
<td>3.14</td>
<td>Control Signal for Second Set of PID Parameter</td>
<td>41</td>
</tr>
<tr>
<td>3.15</td>
<td>Simulation Model for Advanced PID Controller</td>
<td>44</td>
</tr>
<tr>
<td>3.16</td>
<td>Parameter for PID Controller</td>
<td>44</td>
</tr>
<tr>
<td>3.17</td>
<td>Speed Response for Advanced PID Controller</td>
<td>45</td>
</tr>
<tr>
<td>3.18</td>
<td>Control Input for Advanced PID Controller</td>
<td>45</td>
</tr>
<tr>
<td>4.1</td>
<td>System with RASMC Controller</td>
<td>53</td>
</tr>
<tr>
<td>4.2</td>
<td>Simulink Model for Disturbance Estimation</td>
<td>54</td>
</tr>
</tbody>
</table>
Figure 4.3: Simulink Model for Parameter Estimation ........................................... 55
Figure 4.4: Simulink Model for Speed Reference .................................................. 55
Figure 4.5: Simulink Model for Plant ................................................................. 56
Figure 4.6: Simulink Model for RASMC Controller ............................................. 58
Figure 4.7: Overshoot for 80[rad/s] Speed Response .......................................... 60
Figure 4.8: Settling Time for 80[rad/s] Speed Response ....................................... 61
Figure 4.9: Overshoot for 80[rad/s] Speed Response .......................................... 62
Figure 4.10: Settling Time for 80[rad/s] Speed Response ..................................... 62
Figure 4.11: Overshoot for 180[rad/s] Speed Response ....................................... 63
Figure 4.12: Settling Time for 180[rad/s] Speed Response .................................... 64
Figure 4.13: Overshoot for 180[rad/s] Speed Response ....................................... 65
Figure 4.14: Settling Time for 180[rad/s] Speed Response .................................... 65
Figure 4.15: Overshoot for 80[rad/s] Speed Response .......................................... 66
Figure 4.16: Settling Time for 80[rad/s] Speed Response ....................................... 67
Figure 4.17: Overshoot for 80[rad/s] Speed Response .......................................... 68
Figure 4.18: Settling Time for 80[rad/s] Speed Response ....................................... 68
Figure 4.19: Overshoot for 180[rad/s] Speed Response ....................................... 69
Figure 4.20: Settling Time for 180[rad/s] Speed Response ..................................... 70
Figure 4.21: Overshoot for 180[rad/s] Speed Response ....................................... 71
Figure 4.22: Settling Time for 180[rad/s] Speed Response ..................................... 71
Figure 4.23: Overshoot for 80[rad/s] Speed Response .......................................... 72
Figure 4.24: Settling Time for 80[rad/s] Speed Response ....................................... 73
Figure 4.25: Overshoot for 80[rad/s] Speed Response .......................................... 74
Figure 4.26: Settling Time for 80[rad/s] Speed Response ....................................... 74
Figure 4.27: Overshoot for 180[rad/s] Speed Response ....................................... 75
Figure 4.28: Settling Time for 180[rad/s] Speed Response .................................... 76
Figure 4.29: Overshoot for 180[rad/s] Speed Response ....................................... 77
Figure 4.30: Settling Time for 180[rad/s] Speed Response .................................... 77
Figure 4.31: Speed Response Using RASMC ...................................................... 78
Figure 4.32: Control Signal Using RASMC .......................................................... 79
Figure 5.1: Simulink Model of PID Control System ............................................. 82
Figure 5.2: Experimental Speed Response Using PID Controller ......................... 83
Figure 5.3: Experimental Speed Response Error .................................................. 83
Figure 5.4: Experimental Speed Response Using PID Controller ......................... 84
Figure 5.5: Experimental Speed Response Error .................................................. 84
Figure 5.6: Experimental Speed Response Using PID Controller ......................... 85
Figure 5.7: Experimental Speed Response Error .................................................. 85
Figure 5.8: Combined Speed Response Using PID Controller ............................. 86
Figure 5.9: Combined Speed Response Using PID Controller ............................. 87
Figure 5.10: Combined Speed Response Using PID Controller .......................... 87
Figure 5.11: Simulink Model of RASMC system ................................................. 89
Figure 5.12: Experimental Speed Response Using RASMC ............................... 90
List of Tables:

Table 2.1: Experimental Devices ................................................................. 8
Table 6.1: Testing Cases ............................................................................... 96
Table D.1: Routh-Hurwitz Parameter for Region 1 .................................. 108
Table D.2: Routh-Hurwitz Parameter for Region 2 ............................... 109
Chapter 1. Introduction

1.1 Background

In the modern industrial engineering such as space robots and transparent vehicles, many robotic applications require compact Joint Drive Systems (JDS) that can apply high torques at low speeds [36]. Gear Bearing Drive (GBD) transmission, which can produce up to 5000:1 torque ratio [2, pp. 45-46] theoretically, was originally invented by Elias Brassitos for this certain task. However, in the practical actuator system consisting of an actuator, transmission and workload, nonlinearities such as gear tooth surface friction, backlash, stiffness and kinematic error can cause unpredictable behavior in the open-loop operation [1, pp. 87-89]. Therefore, there is a discrepancy between the simplified model and the practical system [3, 4]. To guarantee the stability of the system performance under the circumstance full of unknown nonlinear disturbances, powerful control strategy is required.

Proportional-Integral-Derivative (PID) control is a control loop feedback mechanism widely used in industrial control systems and a verity of other applications requiring continuously modulated control [5] because of its simple design process and sophisticated theory developed in these years. However, due to the inherent deficiency of PID that the controller cannot deal with nonlinearities, sliding mode control (SMC) is taken into consideration.

To solve the effect of unknown perturbations, SMC combined with perturbation estimation (SMCPE), which handles with the structured parameter without upper bound knowledge of the disturbance [6, 7], is developed to enhance the tracking performance. For increasing the robustness and adaptiveness of the system behavior with unmodeled disturbances and uncertain initial parameters, robust-adaptive Sliding Mode Control (RASMC) is introduced from the combination of SMCPE and adaptive control schemes [8].
1.2 Previous Work on GBD

1.2.1 Previous Work by Elias Brassitos

Elias Brassitos was a Ph.D. student working in the Piezoactive System Laboratory until 2016. He invented GBD transmission in the earlier 2016. By using the prototype of Epicyclic Gearing System (EGS) combined with the level principle in the designing process, GBD transmission could produce high torque ratio in the practical operation within a compact size. The efficiency improves with higher torques output and reaches a maximum of 83% [2, pp. 102].

In the practical system, because of the pervasive nonlinearities existing in the system such as gear tooth friction, stiffness, kinematic error and nonlinear damping, precise model of the actuation plant is difficult to generate. In the modeling process made by Brassitos, kinematic error and stiffness have been considered and the simplified model of the GBD transmission was given out.

1.2.2 Previous Work by Kwanghyeon Cho

The open loop test did by Brassitos matched well with the Simulink mode. However, to achieve certain performance task such as set point control and set speed control, a powerful control is required.

Considered about the advantages of brushless motor caused by its structure and material of magnet [9] compared with traditional DC motor, Cho replaced the DC motor with 3-phase brushless direct current (BLDC) motor. For testing the maximum torque that the system can produce practically, a Hysteresis Brake (HB) is added to the plant. In the modeling process, for decreasing the complexity of the practical system, a simplified model by linearizing all the nonlinearities and minimizing all the disturbance and perturbations [10] was developed.

Based on the linearized model, a prototype of RASMC combined with perturbation estimation was designed and tested.
1.3 Thesis Contribution

The main contributions of this research are listed below:

➢ More accurate dynamic model compared with previous work.

In the open-loop experiment, a strange ‘jump behavior’ is detected. The reason for this behavior remains unknown but both the previous models reconstructed by Elias Brassitos and Kwanghyeon Cho neglected the piecewise behavior. In this research, for the sake of modeling accuracy, the piecewise behavior is considered and modeled successfully. The improved model would produce more convincing result when controllers are applied.

➢ New design concept for PID controller.

During the PID based control algorithm design process, pole-zero cancellation method is applied. Compare with previous PID control developed by Kwanghyeon Cho, the new controller gives the designers more freedom to choose the desired poles based on the performance criteria.

➢ Improved robust-adaptive sliding mode controller.

In this research, perturbation estimation and projection operator were added to the original RASMC. The modified RASMC can guarantee the asymptotic convergence of the system. Meanwhile, the method of choosing the suitable control gains is discussed in the thesis which could be used for the future RASMC designing process.

➢ Similarity of the simulation and experimental result.

In the previous work by Kwanghyeon Cho, due to the unsuitable gains selected for RASMC, the result of the simulation is highly different with the real-time experiment. In this research, the system performance not only owns a better performance but presents highly similarity to the Simulink result.
1.4 Thesis at a Glance

In this thesis, prior works on the gear bearing transmission system are introduced to the reader in Chapter 1. Earlier in 2016, Elias Brassitos, a Ph.D. student who was working in Piezoactive system Lab, built up the GBD transmission prototype combined with a DC motor as input actuator and a disk as output inertia. In 2017, Kwanghyeon Cho, a master student who was working in Piezoactive system Lab modeled the system by linearizing all disturbances and developed controllers for the model.

Chapter 2 introduces the method of experimental modeling which used in this research to reproduce the mathematical model of the plant. In the modeling process, nonlinearities such as dynamic frictions and kinematic errors will be linearized to generate a simplified first order system. The parameters of the system will be determined by the data collected in the practical open-loop experiment. The linearized model will be used to design the controllers in the following chapters.

Chapter 3 and Chapter 4 present the designing progress of two robust controllers. In Chapter 3, PID control is developed for the reconstructed piecewise linear model in Chapter 2. During the controller design process, pole-zero cancellation method was applied. In Chapter 4, a traditional RASMC is first designed based on the reconstructed model, perturbation estimation is then added to RASMC to guarantee the asymptotic convergence of the steady-state error. The simulation results of the speed response using both controllers are presented in each chapter.

A detailed discussion about the real-time experiment of the close-loop system using two different controllers is given in chapter 5. By comparing the experimental result with the simulation result, the effectiveness of both controls would be demonstrated. In the end, a conclusion will be presented regarding of the system behavior.

A brief verification and discussion about the unexpected speed jump would be given in chapter 6. Due to the highly nonlinearities and the unmodeled disturbances
combined with the external perturbations, an unpredictable ‘Jump’ happened in both open-loop experiment and close-loop system response around 105 [rad/s] of motor speed during the research. The plausible reason and reasonable assumption for this behavior would be illustrated in Chapter 6 and some experimental tests which aims to verify the assumption would be given.

For students who intend to work on the GBD system in the future, some suggestion and guidance of the potential future work which is valuable to perform have been given in Chapter 7. The guidance concentrates mostly on the practical applications instead of the theory.
Chapter 2. Open-Loop Model

2.1 Open-Loop Experiment Setup

Before implementing the real-time experiment, the experimental devices should be set up correctly. Figure 2.1 depicts the overall system consisting of a BLDC motor, gear bearing drive (GBD) and hysteresis brake (HB).

![GBD Transmission System](image)

**Figure 2.1: GBD Transmission System**

As shown in Figure 2.1, the system is designed for either position or speed control. The brushless motor is connected to input shaft in the front part of the system which would receive input signal from dSPACE controller board. One pair of torque sensor and encoder is set in the shaft between the motor and the GBD transmission to collect data of motor position/speed and the input torque. The hysteresis brake is placed in the last part of the system which is used to provide certain value of resist torque to test the maximum torque which can be developed by GBD. Another pair of torque sensor and encoder is set in the connecting shaft between the GBD and the hysteresis brake to acquire the output position/speed and the output torque.

2.1.1 Experimental Devices Setup

The block diagram which represents the experimental devices scheme is shown in Figure 2.2.
As seen in Figure 2.2, the practical system data is measured as following the flows. After receiving the current command from black box, the BLDC motor starts to rotate and generate the input torque along with motor speed. When the input torque and speed pass through the GBD transmission, they will be transmitted into output torque and speed. Two pairs of encoders and torque sensors are placed in the connection link of the GBD transmission to get the position/speed and the torque information. The measured data will be converted into digital signal through analog to digital converter (ADC). The digital signal is processed by dSPACE using the block built up in Simulink. The digital control signal will be converted to analog signal by digital to analog converter (DAC) inside dSPACE, after that, the control signal will be sent to the black box. In the black box, voltage output from dSPACE controller board will be converted into current signal. The modulation process of the current input will be processed in the special servo driver matched with BLDC motor. Finally, the modified current input is sent to the BLDC motor to generate next input torque and motor speed.

The experimental devices are listed in Table 2.1.
Table 2.1: Experimental Devices

<table>
<thead>
<tr>
<th>ID#</th>
<th>Description</th>
<th>Manufacture</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dSPACE</td>
<td>Germany</td>
<td>DS1104</td>
</tr>
<tr>
<td>2</td>
<td>Torque Sensor 1</td>
<td>FUTEK</td>
<td>TRS605</td>
</tr>
<tr>
<td>3</td>
<td>Torque Sensor 2</td>
<td>FUTEK</td>
<td>TRS605</td>
</tr>
<tr>
<td>4</td>
<td>Encoder 1</td>
<td>TONIC</td>
<td>Ti0200</td>
</tr>
<tr>
<td>5</td>
<td>Encoder 2</td>
<td>TONIC</td>
<td>Ti0200</td>
</tr>
<tr>
<td>6</td>
<td>Servo Drive</td>
<td>Servo2Go</td>
<td>DPRALTE-060B080</td>
</tr>
<tr>
<td>7</td>
<td>BLDC Motor</td>
<td>Hacker</td>
<td>A30-14L</td>
</tr>
<tr>
<td>8</td>
<td>Hysteresis Brake</td>
<td>MAGTROL</td>
<td>HB-3500</td>
</tr>
</tbody>
</table>

dSPCAE Controller Board

The appearance of dSPACE R&D DS1104 Controller Board used in the actual plant is shown in Figure 2.3.
dSPACE DS1104 Controller Board is an open source repository software package typically used for creating open access repositories for scholarly and published digital content and generally refers to a real-time interface used in the control task [11]. The corelative software for experiment set up and control runs on PC, and the simulator hardware is connected to the PC via link board. For the controller board to work properly, the DAQ amplifier (black box with power switch and ribbon cables on the backside) should be turned on before the computer is turned on. The key benefit is that DS1104 controller board is a cost-effective entry-level system with I/O interfaces and a real-time processor on a single board that can be plugged and manipulated directly into a PC. It can upgrade the PC automatically to a development tool for control prototyping and is ideal for education and research purpose.

At the bottom left side of Figure 2.3, there are four columns of cable connectors. The first two columns (inside red frame) are ADCH ports which are used to receive the analog output from the system sensors and convert the analog signal to digital signal inside dSPACE. The second two columns (inside blue frame) are DACH ports which are used to output the signal from dSPACE. Generally speaking, the ADCH ports are used to collect data from practical system and the DACH ports are used to give input signal to practical system. The two connectors in the middle (inside yellow frame) are INC ports which are used to connect encoders in the real-time system to receive the position and velocity information.

**Black Box**

The black box used in the experiment is shown in Figure 2.4.
The ‘Black Box’ shown in Figure 2.2 is consisting of three components as can be seen in Figure 2.4.

On the top side (inside red frame), the small box is called ‘digital servo driver’. Servo driver offers full tuning control of all servo loops and is designed for distributed or centralized servo system [12]. In other words, the servo drive converts the voltage input from the DACH ports to current input for 3-phase brushless DC motor. The especially designed control loop inside the box guarantees the proper working current command for the BLDC motor. The computer software for manipulating servo drive is installed in PC called ‘DriveWare’ and will be introduced later.

On the bottom side (inside blue frame) is the power supply for the digital servo drive. The power supply is powered by computer and the computer must be turned on to guarantee the working condition for BLDC motor.

On the left side (inside yellow frame) is the capacitor set. The capacitors are used
for stabilizing the current command from the servo drive. Within the violent phase change in the high velocity situation, the capacitors help to maintain the stability of current command for the BLDC motor by compensating for the phase change in the rapid violation [13].

2.1.2 Experimental Software Setup

**DriveWare**

The DriveWare software interface is shown in Figure 2.5.

![DriveWare Interface](image)

**Figure 2.5: DriveWare Interface**

On the top side of the picture, the green button (inside red frame) is the switch button for the power supply. Only when the users click ‘Enable’ will the power supply be turned on. Before running the experiment, the initial phase situation should be checked using the ‘Phase Detect’ function of the software. After everything is ready, the red notification ‘DISABLED’ in the bottom side would turn to green and the digital servo driver is connected automatically to the PC for receiving following orders.

On the left side of the picture, under the ‘User Units’ exhibition, all the gains and parameters can be selected by users to modify the convert ratio of voltage to current.
On the right side of the picture, under ‘Scope’ choice, the data of the servo drive system can be shown in the coordinate. The available data contains input current, input voltage, motor speed, etc.

**Matlab**

Matlab is the significant software in the experiment which allows the users to build up controller in Simulink and calculate the input command sent to the Controller Board, the Simulink model for open-loop experimental test is shown in Figure 2.6.

![Simulink Model](image)

**Figure 2.6: Open-Loop Experiment Simulink Model**

As shown in Figure 2.6, all the ports of DS1104 Controller Board can be connected to the PC directly. The ‘DS1104ENC_POS_C1’ block (inside black frame) connects to ‘INC1’ port, the data measured by encoder can be sent to computer through real-time interface. After multiplying the gains correlative with the encoder resolution, the position and velocity data can be shown to researcher. The ‘DS1104DAC_C1’ block (inside blue frame) represents the DACH port 1 of Controller Board. The input signal
from computer is converted to voltage output in the board and sent to digital servo
driver for BLDC motor. Same with DACH, ‘DS1104ADC’ blocks shown in the red
frame connect to ADCH channels which collect the data captured by torque sensors.

In the practical experiment, input command is sent to the controller board with a
pulse block which lasts 20 second and all the measured data is sent to Matlab through
Controller Board which can be depicted in ‘Control Desk’ software.

**Control Desk**

Control Desk is the software matching with Controller Board which mainly used
to exhibit and record data collected by Controller Board. The interface of Control Desk
software is shown in Figure 2.7.

![Control Desk Software Interface](image)

**Figure 2.7: Control Desk**

As shown in Figure 2.7, the software ‘Control Desk’ allows the user to display the
data processed in Controller Board. It also permits the users to collect data in certain
time-period if researchers choose to capture data in the ‘Capture Setting’ button. The
stored information can be reloaded by workspace in Matlab and manipulated in Matlab by researchers.

2.2 Open Loop Experimental Description

2.2.1 Modeling Method

In the practical experimental process, system modeling plays an important role. For the following analysis of the whole system becomes as accurate as possible, a highly precise mathematical model which could represent the system behavior is required. There are two general methods to reconstruct the system model:

1) Analytical modeling method

Analytical modeling is mainly based on the direct modeling of each part in the whole system. The exact dynamic equation of each part of the system should be generated separately. After that, all dynamic equations would be combined to calculate the overall system equation which can relate the input variables with the output variable.

The advantages of this method are its relatively high accuracy and the ability to describe the thorough behavior of the entire system. For this reason, when the accuracy of the system modeling is considered as the priority, this method is the most suitable choice for researchers.

The disadvantage of this method is its high complexity. As the matter of fact, when all the components are under consideration, such as frictions, disturbances and kinematic errors, the finalized dynamic equation which describes the system behavior could become highly nonlinear. Under this situation, the system analysis would become difficult because of the nonlinearities and high orders of the dynamic equation. The controller design and stability prove would also become much harder.

2) Experimental modeling method

Experimental modeling is mainly based on the input and output data collected in the real-time experiment. According to the generalized system performance, the basic
type of the overall system will be determined. By using ‘Curving Fitting’ or other tools, the optimal parameters of the certain system type can be determined. After generating the prototype of the model, it is necessary to evaluate the accuracy of the model. Using certain performance criteria, each parameter can be modified during the evaluation process.

The advantage of the method is its simplicity. Since the details of disturbances which happen inside the system have been neglected and the nonlinearities will be linearized according to the basic system type, the analysis process would become much simpler. Meanwhile, the control design and the stability prove will also become easier.

The disadvantage of the system is the inaccuracy. Because of the actual existing disturbances such as mechanical friction and kinematic error are neglected, the model owns discrepancy compared with actual plant. When unexpected behavior happened in the experiment, few possible reasons and directions can be figured out from the model.

In the GBD transmission system, due to the nonlinearities (friction, operation dead zone, etc.), disturbances (system noise, environmental noise, etc.), unknown parts (black box, BLAC motor, etc.) and even uncertain initial system parameters (inertia, mass and damping factor), analytical modeling would be extremely hard to realize and implement. Therefore, experimental modeling method should be applied to the open-loop system modeling process.

### 2.3 Experimental Data Analysis

#### 2.3.1 Experimental Data Display

Given a certain constant current step input for 20 seconds, the brushless motor speed response (partial) of the open-loop system can be shown in Figure 2.8.
During the experiment, because of the dead zoon effect in the system, the system will only respond to a current input which is greater than 2.9[A]. For protecting the experimental devices, the current input is limited to ±5[A].

Three things are noticed from the speed response:

1) The open loop system performance behaves like a first order system. For each constant current input, the speed response will go to a steady state value within a roughly same time constant.

2) Piecewise behavior happens at the speed of roughly 105[rad/s]. Regardless of the current input, once the speed of the motor passes 105[rad/s], the system will change its behavior to another first order like system, with a larger time constant.

3) There is a gap between 100[rad/s] to 150[rad/s], which means that no feedforward control can bring the speed response to the velocity region of 100[rad/s] through 150[rad/s].
2.3.2 Data Analysis

To verify the information obtained from the speed response, the steady-state torque response of the system correspond to different current input can be shown in the following plot. Since two torque sensors which detect the torque response from motor side and brake side have been implemented in the experiment, both torque responses would be shown in the following figures.

The current-motor torque plot is shown in Figure 2.9, with a torque bias of -0.0233[N \cdot m].

![Motor Torque Response](image)

**Figure 2.9: Motor Torque Response**

The current-brake torque plot is shown in Figure 2.10, with a torque bias of +0.0519[N \cdot m].
The current-total torque response which is generated by Equation (2.3.1) is shown in Figure 2.11.

\[ T_{total} = (T_{motor} + 0.0233) - \frac{3\cdot T_{brake} - 0.0519}{N} \]  \hspace{1cm} (2.3.1)

From the previous plot shown above, the three notifications mentioned in section 2.3.1 clearly appeared again. For the input current less than 2.9[A], nearly no torque response can be detected by sensor, which verifies the assumption that there is certain
dead zone effect existing in the system. In addition, the torque response can be separated into two linear curves at the current interval of 4.5-4.6[A]. As shown in Figure 2.1, the separation speed 105[rad/s] happens exactly between 4.5-4.6[A]. From Figure 2.3, the brake torque response drops rapidly after the input current passes 4.5[A] and this might give certain explanation to the ‘Jump behavior’ of the GBD system. The reason why the torque response would change after the speed passes certain value remains unknown, a brief discussion and verification about this phenomenon is given in Chapter 6.

2.3.3 Conclusion about Experimental Data

The analysis to the experimental data of open-loop system provides three key directions for reconstructing the system model:

1) The system behaves as a first-order system.
2) Piecewise linear model should be taken into consideration since the system behavior changes at certain speed value.
3) The input signal should be current and the output signal should be motor speed, and they are related by a first-order like system function.

2.4 Model Reconstruction

2.4.1 Simplified System Dynamic Model

For completely reconstructing the whole system, a mathematical analysis of the GBD system will be given based on the simplified dynamic model. The actual GBD transmission system can be simplified into a two-degree-of-freedom lumped parameter system as shown in Figure 2.12.
Based on the lumped parameter system, the dynamic equation of motion for the GBD system can be separated into two parts as seen in Figure 2.13.

The input part (left part) consists of the BLAC motor and the sun gear of GBD transmission and the output part (right part) consists of the plant gear and hysteresis brake. Both parts are assumed to be a rigid body so that there are no stiffness coefficients. Each part is assumed to experience the same damping factor. The kinematic error and backlash of both subsystem have been neglected for the sake of simplicity. $T_f$ contains all the nonlinearities related to the rotation speed of each part. The equation of motion for the two parts can be written as below:
\[ T_m - T_{f1}(\dot{\theta}_m) = (J_m + J_1) \dot{\theta}_m + B_m \dot{\theta}_m \]  \hspace{1cm} (2.4.1) \\

\[ T'_m - T_{f2}(\dot{\theta}'_m) = (J_L + J_2) \dot{\theta}'_m + B_L \dot{\theta}'_m \]  \hspace{1cm} (2.4.2)

Combining the two equations with the gear ratio \( N \), which is assumed to be a constant during the operation process, Equation (2.4.2) can be reformed as:

\[ N = \left| \frac{\tau'_m}{T_m} \right| = \left| \frac{\dot{\theta}_m}{\ddot{\theta}_m} \right| = \left| \frac{\dot{\theta}_m}{\ddot{\theta}_m} \right| \]  \hspace{1cm} (2.4.3)

\[ T_m = \frac{1}{N^2} (J_L + J_2) \ddot{\theta}_m + \frac{1}{N^2} B_L \dot{\theta}_m + T_{f2} \left( \frac{1}{N} \dot{\theta}_m \right) \]  \hspace{1cm} (2.4.4)

Since the output direction of the GBD is opposite with the input speed, the sign of Equation (2.4.4) should be multiplied by -1. Combining Equation (2.4.1) with (2.4.4), the overall dynamic equation calculated for the system can be shown below:

\[ T_m = (J_m + J_1) \dot{\theta}_m + B_m \dot{\theta}_m + T_{f1}(\dot{\theta}_m) \]

\[ = -\frac{1}{N^2} (J_L + J_2) \ddot{\theta}_m - \frac{1}{N^2} B_L \dot{\theta}_m - T_{f2} \left( \frac{1}{N} \dot{\theta}_m \right) \]  \hspace{1cm} (2.4.5)

\[ T_m - T_{f1}(\dot{\theta}_m) - T_{f2} \left( \frac{1}{N} \dot{\theta}_m \right) \]

\[ = \left( J_m + J_1 + \frac{1}{N^2} (J_L + J_2) \right) \ddot{\theta}_m + \left( B_m + \frac{1}{N^2} B_L \right) \dot{\theta}_m \]  \hspace{1cm} (2.4.6)

By representing \( J_t = J_m + J_1 + \frac{1}{N^2} (J_L + J_2) \), \( T_{f1}(\dot{\theta}_m) = T_{f1}(\dot{\theta}_m) + B_m \dot{\theta}_m \) and \( T_{f2}(\dot{\theta}_m) = T_{f2} \left( \frac{1}{N} \dot{\theta}_m \right) + \frac{1}{N^2} B_L \dot{\theta}_m \), Equation (2.4.6) can be further simplified as:

\[ T_m - T_{f1}(\dot{\theta}) - T_{f2}(\dot{\theta}_m) = J_i \ddot{\theta} \]  \hspace{1cm} (2.4.7)

As shown in Equation (2.4.7), the two-degree-of-freedom system can be simplified to only one-degree-of-freedom by using the gear ratio \( N \). \( T_m \) represents the motor torque generate by BLAC motor, \( T_{f1} \) represents all terms related with the motor speed in the first part of the system, \( T_{f2} \) represents all terms related with the motor speed in the second part of the system and \( J_i \) is the generalized inertia of the whole system. Figure 2.14 depicts the schematic of the simplified one-degree-of-freedom system.
Figure 2.14: Schematic of One-Degree-of-Freedom System

2.4.2 Regions of System Reconstruction

Equation (2.4.7) reveals that the system should be a first order system, which is identical to the assumption we made in section 2.3. Thus, the linearized system will be built up using Equation (2.4.7). According to assumption (2), a piecewise linear model should be taken into consideration. Based on the system analysis, two regions corresponding to different speed are separated:

1) Low-speed region, corresponding to the speed response lower than 105[rad/s], with respect to the input current from 2.9-4.5[A].

2) High-speed region, corresponding to the speed response higher than 105[rad/s], with respect to the input current from 4.6-5[A].

2.4.3 System Reconstruction in Region 1

From Equation (2.4.7), \( T_m \) is the motor torque which correspond to the motor torque measured in Figure 2.2, \( T_{f2} \) is the friction torque which combined the brake torque and the friction torque in the right part of the system. Consider that the friction torque would be scaled by \( I/N^2 \), it is safely to neglect the friction torque in the right part of the system and the term \( T_{f2} \) only represent the brake torque measure in Figure 2.10. Thus, \( T_m-T_{f2} \) represents the total torque \( T_{total} \) measure in Figure 2.11.

Using curve fitting app in the Matlab software, the linearized curve of current - total torque is shown in Figure 2.15. The details of fitting process can be shown in
Appendix A.

Figure 2.15: Current-Total Torque Curve

From the optimal parameters the software provided, equation of $T_{total}$ can be made as:

$$T_{total} = T_m - T_{ft2}(\dot{\theta}) = 0.006098 \times i + 0.00261$$  \hspace{1cm} (2.4.8)

In the steady state situation, motor speed becomes steady constant due to the fact the acceleration of the system becomes zero. Using this property, Equation (2.4.7) becomes:

$$T_m - T_{ft2}(\dot{\theta}) = 0$$ \hspace{1cm} (2.4.9)

Combine equation (2.4.8) with Equation (2.4.9):

$$T_{ft1}(\dot{\theta}) = T_m - T_{ft2}(\dot{\theta}) = 0.006098 \times i + 0.00261$$  \hspace{1cm} (2.4.10)

Based on the simplified model analysis, $T_{ft1}$ represents the torque of system which related to the motor speed, the linearized speed-torque curve is also shown in the Figure 2.16.
From the optimal parameters the software provided, equation of $T_{tf1}$ can be made as:

$$T_{tf1} = 0.0001636 \cdot \dot{\theta} + 0.0134 \quad (2.4.11)$$

Combine Equation (2.4.10) with (2.4.11), the equation which could represent the steady state situation of the system can be shown as follow:

$$T_{total} - T_{tf1} = 0.006098i - 0.0001636\dot{\theta} - 0.01079 = 0 \quad (2.4.12)$$

Adding back the inertia term, the numerical equation for the whole speed response can be represented as:

$$0.006098 \cdot i - 0.0001636 \cdot \dot{\theta} - 0.01079 = J_{t1} \ddot{\theta} \quad (2.4.13)$$

Taking Laplace transformation of Equation (2.4.13), the transformation equation for the system can be represented as:

$$\begin{align*}
\hat{\theta} = & \frac{0.006098i}{J_{t1} \cdot s + 0.0001636} - \frac{0.01079}{J_{t1} \cdot s + 0.0001636} \\
\quad (2.4.14)
\end{align*}$$

For determining the $J_{il}$ of the plant, time constant needs to be figured out. Reexamining Figure 2.8, the settling time of the first region can be assumed as 3[sec] since all speed response in the low speed region tend to have the same settling time. The inertia can be roughly set as:
\[ J_{t1} = 0.00012 \] (2.4.15)

The complete speed response can be then written as:

\[ 0.006098 \times i - 0.0001636 \times \dot{\theta} - 0.01079 = 0.00012 \times \ddot{\theta} \] (2.4.16)

2.4.4 System Reconstruction in Region 2

Using the same concept as the reconstruction process in region 1, the current-total torque curve is shown in Figure 2.17 and the speed-total torque is shown in Figure 2.18.
With the parameters provided by curve fitting software, the equations for the two plots can be shown as:

\[ T_{\text{total}} = 0.002849 \times i + 0.01902 \]  \hspace{1cm} (2.4.17)

\[ T_{f1} = 2.306 \times 10^{-5} \times \dot{\theta} + 0.02842 \]  \hspace{1cm} (2.4.18)

Combine Equation (2.4.17) and (2.4.18), we have the following equation which can represent the steady state situation for the system:

\[ 0 = 0.002849 \times i - 2.306 \times 10^{-5} \times \dot{\theta} - 0.0094 \]  \hspace{1cm} (2.4.19)

Adding back the inertia term, the speed response for the system can be developed as:

\[ 0.002849 \times i - 2.306 \times 10^{-5} \times \dot{\theta} - 0.0094 = J_{t2} \ddot{\theta} \]  \hspace{1cm} (2.4.20)

Taking Laplace transformation of equation (2.4.20), transfer function can be shown as:

\[ \dot{\theta} = \frac{0.002849 \times i}{J_{t2} \times s^2 + 2.306 \times 10^{-5} \times s} - \frac{0.0094}{J_{t2} \times s^2 + 2.306 \times 10^{-5}} \]  \hspace{1cm} (2.4.21)

For determining the overall inertia parameter \( J_{t2} \), time constant should be figured out. Reexamining Figure 2.8, the time constant for the speed response in region 2 can be roughly set as 8.5[sec] for all the speed response higher than 105[rad/s] share the same raising time. With the time constant, the inertia of the system in region 2 can be determined as:

\[ J_{t2} = 0.00005 \]  \hspace{1cm} (2.4.22)

The complete speed response for the system can then be written as:

\[ 0.002849 \times i - 2.306 \times 10^{-5} \times \dot{\theta} - 0.0094 = 0.00005 \times \ddot{\theta} \]  \hspace{1cm} (2.4.23)

### 2.5 Model Evaluation

The Simulink model built upon Equation (2.4.16) and (2.4.23) can be shown in Figure 2.19.
Figure 2.19: Simulink Model of Open Loop System
In the Simulink model, ‘Dead Zone’ and ‘Sign’ blocks right after step input are introduced to represent the dead band effect happens in real-time system and the value of dead zone is set to be ±2.9, and the switch threshold is set to be 105. Given the constant input signal (current), the output (speed) signal for the model can be shown in Figure 2.20.

![Speed Response](image)

Figure 2.20: Simulink Result of Speed Response

### 2.6 Conclusion

From Figure 2.20, the speed response using piecewise linear model matches well with the real time experimental data, even the ‘Jump behavior’ can be seen from the plot, so that no further modification is needed for the reconstruction of the system. For the Simulink model is developed based on pure math algorithm, the reason why this unexpected behavior happens remains unknown. However, the piecewise model will be used to help the design process of controllers.
Chapter 3. PID Controller Design

Proportion-Integral-Derivative (PID) control is a powerful classical controller and its theoretical analysis and practical application was developed from the early 1920s onwards [14]. Because of its simply three parameter benchmark controller requirement and complete theory development, PI controller is well suited to the analogous simplified plant control problem. The relative low mathematical requirement also makes researchers easy to understand the whole design process.

3.1 System Analysis

Before moving into the controller design process, complete analysis should be performed on the system to determine the primary direction of following steps.

Recall the equations we developed in Chapter 2 for the system in different regions, put the inertia determined by time constant, the two equations can be written in the following rational form:

\[
\dot{\theta} = \frac{0.006098 + i}{0.00012 + S + 0.0001636} - \frac{0.01079}{0.00012 + S + 0.0001636} \quad (3.1.1)
\]

\[
\dot{\theta} = \frac{0.002849 + i}{0.00005 + S + 2.306 \times 10^{-5}} - \frac{0.0094}{0.00005 + S + 2.306 \times 10^{-5}} \quad (3.1.2)
\]

The structure of the two equations are the same, so the analysis progress will be mainly based on Equation (3.1.1).

From Equation (3.1.1), one thing should be noted that the system is not the traditional single-input-single-output system since the last term of the system is uncontrollable. Thus, in the control design process, this part of the system will be treated like a first order external disturbance. Applying PID function to Equation (3.1.1), the Laplace transfer function of the feedback plant can be represented as:

\[
\hat{\theta}(s) = \frac{61K_dS^2 + 61K_pS + 61K_i}{(1.2 + 61K_d)S^2 + (61K_p + 1.6)S + 61K_i}I(s) - \frac{27.1S}{(152.5K_d + 3)S^2 + (152K_p + 4.1)S + 152.5K_i} \quad (3.1.3)
\]

The feedback transfer function shows that the external disturbance will disappear as time goes to infinity \(s \to 0 \text{ as } t \to \infty\) if a positive \(K_i\) is selected for the controller.
For reducing the complexity of the transfer function, $K_d$ is chosen to be zero. Now, after a PI control is applied to the whole system, the transfer function of the feedback system which ignores the exogenous disturbance can be represented as:

$$G(S) = \frac{61KpS+61Ki}{1.2S^2+(61Kp+1.63)S+61Ki}$$

(3.1.4)

From Equation (3.1.4), the characteristic equation of the plant is:

$$CE(S) = 1.2S^2 + (61 * Kp + 1.63)S + 61Ki$$

(3.1.5)

Applying Routh–Hurwitz stability criterion, system is found out to be stable for any positive value of $Kp$ and $Ki$.

Using same concept to system in speed region 2, the transfer function of the system can be represented as:

$$G(S) = \frac{56.98*(KpS+Ki)}{S^2+(56.98*Kp+0.4612)S+56.98Ki}$$

(3.1.6)

The characteristic equation is:

$$CE(S) = S^2 + (56.98 * Kp + 0.4612)S + 56.98Ki$$

(3.1.7)

Applying Routh–Hurwitz stability criterion, system is found out to be stable for any positive value of $Kp$ and $Ki$. The detail process of Routh–Hurwitz stability criterion is shown in Appendix D.

Based on the analysis of the system, a simple PI controller can theoretically deduce the external disturbance which happens inherently in the system behavior, and the characteristic equation of the feedback system can be reduced to a second order equation with only two parameters left to be determined. Therefore, a PI controller will be introduced to the plant.

### 3.2 Traditional Pole-Zero Cancellation for Separate Regions

Pole-Zero cancellation is a widely used method in PID control, if there is a pole needs to be cancelled, the simplest way is to add a zero which has the same value with the pole in S-domain. In the feedback GBD plant, although no unstable poles exist in the system if positive $Kp$ and $Ki$ are applied, pole-zero cancellation method will still be
taken into consideration. There are two main reasons for this:

1) The place of zero will have certain effect on the system behavior, especially the steady state error [13]. Although the effect is small compared with poles, cancelling the zero is an optional choice so that the close-loop behavior of the system can be precisely predicted using simulation.

2) The characteristic equations for both speed regions are second-order. Since the behavior is required to be theoretically predictable, one pole should be cancelled. Besides, to avoid any possible overshoot in the speed control, the second-order characteristic equation should be reduced to first-order.

Judging by these two reasons, pole-zero cancellation algorithm is going to be implemented during the PID controller design process.

3.2.1 Controller Design for Region 1

Recall Equation (3.1.4) in the following form:

$$G(S) = \frac{50.8333(Kp*S + Ki)}{S^2 + (50.8333*Kp + 1.3583)S + 50.8333Ki}$$ (3.2.1)

Examining the numerator of $G(s)$, the only zero is found out to be:

$$Z = -\frac{Ki}{Kp}$$ (3.2.2)

Since the zero should be cancelled, the denominator of Equation (3.2.1) can be represented in the polynomial form as:

$$\left(S + \frac{Ki}{Kp}\right) * (S + a)$$ (3.2.3)

In Equation (3.2.3), one pole is set to be in the same location as the zero, the other pole is $-a$ in the S-domain which left to be determined. Expand Equation (3.2.3) and match the coefficient with the denominator shown in Equation (3.2.1), the parameters can be determined as:

$$a = 50.8333 * Kp, \quad \frac{Ki}{Kp} = 1.3583$$ (3.2.4)

Combine Equation (3.2.1), (3.2.3) and (3.2.4), the plant transfer function can be represented as:
Based on the new plant transfer function, the time constant can be easily determined as:

\[ \tau = \frac{1}{50.8333 K_p} \]  

(3.5.6)

Assuming the settling time to be 1 [sec], the parameters \( K_p \) and \( K_i \) can be determined as:

\[ K_p = 0.08, \; K_i = 1.3583 \times 0.08 = 0.108664 \]  

(3.2.7)

### 3.2.2 Controller Design for Region 2

Following the same steps as seen in section 3.2.1, recall the transfer function for region 2:

\[ G(S) = \frac{56.98 (K_p S + K_i)}{S^2 + (56.98 K_p + 0.4612) S + 56.98 K_i} \]  

(3.2.8)

Again, the only zero of the system is:

\[ Z = -\frac{K_i}{K_p} \]  

(3.2.9)

Since the zero should be cancelled, the denominator of Equation (3.2.8) can be represented in the polynomial form as:

\[ \left( S + \frac{K_i}{K_p} \right) \times (S + b) \]  

(3.2.10)

In Equation (3.2.10), one pole is set to be the same with the zero shown in Equation (3.2.3), the other pole is represented as \(-b\) in S-domain which remains undetermined. Expand Equation (3.2.10) and match the coefficients with the denominator in Equation (3.2.8), the parameter can be determined as:

\[ a = 56.98 \times K_p, \; K_i = 0.4612 \times K_p \]  

(3.2.11)

Combining Equation (3.2.8), (3.2.10) and (3.2.11), the plant transfer function can be shown as:

\[ G(S) = \frac{56.98 \times K_p}{S + 56.98 K_p} \]  

(3.2.12)

Thus, the time constant can be extracted as:
\[ \tau = \frac{1}{56.98 K_p} \quad (3.2.13) \]

Choosing the settling time to be 2[sec], the parameters of the controller can be calculated as:

\[ K_p = 0.04, \quad K_i = 0.4612 \times 0.04 \quad (3.2.14) \]

### 3.3 Simulation Result for Traditional Method

#### 3.3.1 Simulink Model Construction for Region 1

Using Simulink software in Matlab, the simulation model of the close loop plant for speed region 1 has been built up as shown in Figure 3.1.

![Figure 3.1: Simulation Model for Region 1](image)

As seen in Figure 3.1, the uncontrollable part in the system is treated as the external disturbance with the constant input 1 through the simulation process. The controllable part of the plant has been extracted from the original plant and put inside the feedback loop. The limitation of the saturation block is set to be \( \pm 5.5 \) to compensate for the input limitation from dSPACE controller board in the practical experiment. The step block which starts as \( t=0[\text{sec}] \) stands for the reference speed. The parameters for the PID controller block is set as Figure 3.2.
3.3.2 Simulink Model Construction for Region 2

Similar with the construction process for Region 1, the Simulation model for region 2 can be built up and shown in Figure 3.3.

![Figure 3.3: PID Parameter for Region 2](image)

Again, the uncontrollable part in the system is treated as the external disturbance with the step input through the simulation process. The parameters for PID controller block is shown in Figure 3.4.

![Figure 3.4: PID Parameter for Region 2](image)
3.3.3 Simulation Test on Separate System

In the simulation test, the final value of the step input is adjusted to achieve the different speed regions. For the two models built separately on speed region 1 and region 2, different reference speed which covers the whole region will be tested. The speed response and control signal for each model can be shown in Figure 3.5 through Figure 3.8.

![Speed response of PID Controller on Region 1](image)

Figure 3.5: Speed Response for PID Controller on Region 1
Figure 3.6: Control Signal for PID Controller on Region 1

Figure 3.7: Speed Response for PID Controller on Region 2
From Figure 3.5 and Figure 3.7, the speed response meets the numerically determined settling time in each region. For low speed region in Figure 3.5, the speed response reaches the steady-state value in roughly 2[sec]. In high speed region in Figure 3.5, the speed response cannot rise to 180[rad/s], the reason for this situation can be found by examining Figure 3.6. The control signal reaches the limitation for speed 180[rad/s] so that the speed response cannot arrive the desired value. This situation is easy to understand since the system parameter could change when the speed is over 105[rad/s].

Similar with the result in Figure 3.5, the speed response represented in Figure 3.7 barely meets the expectation. In high speed region, the speed response reaches the desired value at roughly 5[sec]. In low speed region, the speed first goes to negative due to the external disturbance which owns a higher raising rate compared with the controlled response. When the disturbance disappears in a finite time, the speed response goes to the desired value in a relative long time. By checking the control signal
shown in Figure 3.8, all input signal is under the limitation of the saturation block, hence all speed response can meet the desired value in their steady state.

### 3.3.4 Simulation Test on Piecewise Model

Combined with the piecewise linear model developed in Chapter 2 which can precisely represent the system behavior, the close-loop Simulation system which contains PID controller designed in this chapter can be shown in Figure 3.9.

![Figure 3.9: Simulink Model of Close Loop System](image)

The piecewise model inside the subsystem block generated in Chapter 2 can be shown in Figure 3.10.

![Figure 3.10: Subsystem inside Piecewise Model Block](image)
Besides the original model designed in Chapter 2, a quantizer block is added after the position output as digital signal holder. Two slight filters are used to modulate the speed and acceleration output which is same as the output signal processing procedure in the real-time experiment. The starting time in the step block is $t=1[\text{sec}]$. In the simulation test, two sets of PI parameters will be tested. The speed response and control input for each model can be shown in Figure 3.11 through Figure 3.14.

![Speed Response for First Set of PID Parameter](image)

Figure 3.11: Speed Response for First Set of PID Parameter
Figure 3.12: Control Signal for First Set of PID Parameter

Figure 3.13: Speed Response for Second Set of PID Parameter
3.3.5 Conclusion on Simulation Result

From Figure 3.11 and Figure 3.13, the system behavior basically satisfied the requirement when two different sets of PI controller are implemented on the piecewise linear model. As seen in Figure 3.11, the speed response in region 1 gives a good performance. The settling time for the speed is roughly 1[sec] and there is no overshoot. However, in region 2, the output performs a large overshoot in the raising period. By checking the control signal shown in Figure 3.12, the input maintains the maximum value even when the speed passes over the desired value, which means that the first set of PID controller which designed only satisfied speed region 1 cannot guarantee a good system behavior in region 2.

Same phenomenon happens in Figure 3.13, the speed response in region 2 gives a good result with no overshoot and roughly 4[sec] settling time. Nevertheless, the result in region 1 shows the poor effect of the second set of PI control. Although there is no
overshoot, the raising time is too long to meet the performance criteria. By checking
the input signal in Figure 3.14, the control command cannot attain the maximum value
when the speed is lower than the desired value, which means that the second set of PID
controller which designed only satisfied the region 2 cannot guarantee a good system
behavior in region 1.

3.4 Advanced Pole-Zero Cancellation for Both Regions

Section 3.3 shows that the PID controller designed for the separate speed regions
using traditional pole-zero cancellation method cannot meet the performance criteria
for the whole speed region. However, the advanced pole-zero cancellation can meet the
performance requirement using one set parameter. In this section, the advanced
cancellation method will be introduced and a new PI control will be designed based on
this method.

3.4.1 Theory Explanation

In the controller design process of section 3.2, one pole of the system is set to be
exactly same as the only zero to cancel each other. However, in the practical experiment,
the real-time effect of poles and zeros can be cancelled if they are close enough [15],
which means that there is no need to set one pole exactly equal to one zero for
cancellation process. In the following section, this concept will be implemented for
designing the PI controller for both regions

3.4.2 Controller Design for Whole Speed Region

Recall the transfer function of the separate plant for region 1:

\[ G_1(S) = \frac{50.8333(Kp+Ki)}{S^2+(50.8333Kp+1.3583)S+50.8333Ki} \]

(3.4.1)

The only zero is:

\[ Z = \frac{-Ki}{Kp} \]

(3.4.2)
And the two poles can be represented as:

\[ P_{1,2} = \frac{-(50.8333 \times K_p + 1.3583) \pm \sqrt{(50.8333 \times K_p + 1.3583)^2 - 4 \times 50.8333 K_i}}{4} \]  

(3.4.3)

According to the concept of advanced pole-zero cancellation, one pole should be set close to the only zero for cancelling the effect. One thing can be noticed by carefully checking Equation (3.4.2) and (3.4.3), the zero will go close to the origin if \( K_i \) is chosen far smaller than \( K_p \). Same thing happens to the poles, if \( K_i \) is selected smaller enough than \( K_p \), one of the poles, \( P_1 = \frac{-(50.8333 \times K_p + 1.3583) + \sqrt{(50.8333 \times K_p + 1.3583)^2 - 4 \times 50.8333 K_i}}{4} \), will go close to the original point in S-domain.

This information produces the key direction for designing process. Following the guidance, set the parameter of PI controller as:

\[ K_p = 5, \quad K_i = 0.02 \]  

(3.4.4)

By calculation, the zero and poles of the system on region 1 can be shown as:

\[ Z = -0.004 \]  

(3.4.5)

\[ P_1 = -0.003979, P_2 = -255.52 \]  

(3.4.6)

Using the same set of parameters, the zero and poles of the system on region 2 can be shown as:

\[ Z = -0.004 \]  

(3.4.7)

\[ P_1 = -0.003994, P_2 = -285.36 \]  

(3.4.8)

It can be seen from Equation (3.4.5) to (3.4.8), for both systems on each speed region, one pair of zero and pole is close to each other and the other pole is far away from the original point in the left half plane of S-domain. If the pair of pole and zero can cancel each other successfully, the system behavior would depend on the other pole which should meet our design requirement.

### 3.5 Simulation Result for Advanced Method

#### 3.5.1 Simulink Model Construction

Using Simulink software in Matlab, the PID controller designed in section 3.4
combined with piecewise linear model can be shown in Figure 3.15.

Figure 3.15: Simulation Model for Advanced PID Controller

The parameters of PID block is set to be the same as in section 3.4 as shown in Figure 3.16.

Figure 3.16: Parameter for PID Controller

3.5.2 Simulation Test for Advanced Controller

Setting different input signal in the step block, the speed response and the control input can be shown in Figure 3.17 and Figure 3.18.
Figure 3.17: Speed Response for Advanced PID Controller

Figure 3.18: Control Input for Advanced PID Controller
3.5.3 Conclusion on Simulation Result

As it can be seen from Figure 3.17, the performance of all kinds speed response meets the criteria. The settling time is 1[sec] for region 1 and 2[sec] for region 2. There is a tiny steady state error for each speed response since the effect of zero cannot be perfectly cancelled by one of the poles.

Figure 3.18 shows that the control input maintains its maximum value before the speed goes to the reference value. Once the speed goes slightly over, the control input would decrease rapidly so that there is almost no overshoot in the respond period. This gives the credit to the other pole of the plant. As one pole has been cancelled out by the zero, the system behavior would be controlled by the other pole which is far away from the original point in the left half plane of S-domain. With the help of this pole, the system can give out the best performance.
Chapter 4. RASMC Design

4.1 Theorem Introduction

In modern control system, sliding mode control, or SMC, is a nonlinear control method which alters the dynamics of a nonlinear system by application of a discontinuous control signal (or more rigorously, a set-valued control signal) that forces the behavior of output value “slide” along a cross-section of the system’s normal behavior [16]. Such motion of the system as it slides along the boundaries is called a sliding mode [17]. The control law of SMC is constructed based on state feedback control theory, which can switch from one continuous structure to another depends on the state feedback value. The multiple control structure is designed so that the ultimate state value is moving towards to the desired trajectory which may not exist within one control structure [18]. Due to the multiple control structure, motion of state value is sliding along the desired trajectory, thus this kind of motion is regarded as “sliding mode” and the corresponding control method as “sliding mode control”.

SMCPE philosophy is an enhanced version of the conventional SMC [19, 20], which provides robustness of control against perturbations (i.e., the combination of modeling uncertainties is unknown external disturbances) [6].

Robust Adaptive Sliding Mode Control is the combination of both robust and adaptive control theory to SMC. In practical control task, parameters of the system, such as gravitational load [8], overall rotational inertia, damping factor and stiffness may change with respect to operational temperature. To account for this effect, one nonlinear control methods has been popular: adaptive control [21, 22, 23, 24, 25, 26]. By combining the adaptive algorithm into SMC, RASMC becomes a powerful and popular control algorithm in recent years.

In the earlier work, a prototype RASMC has been developed by Cho, however, experience in the adaptive control of linear system [27] suggests that poor initial
parameter estimates may result in poor transient behavior, the control has been improved using projector operator developed by Saeid Bashash and Nader Jalili [28].

4.2 RASMC Controller Design

4.2.1 Mathematical Model Analysis

Recall the general equation derived in Chapter 2, the simplified dynamic model can be written as:

\[ T_m - T_{ft1}(\dot{\theta}) - T_{ft2}(\dot{\theta}) = J_t \ddot{\theta} \] (4.2.1)

Note that both \( T_{ft1} \) and \( T_{ft2} \) are the nonlinear function related with angular velocity and \( T_m \) is the function of current, thus Equation (4.2.1) can be represented using another form as:

\[ J \ddot{\theta} + B \dot{\theta} = u + d \] (4.2.2)

In Equation (4.2.2), \( J \) denotes the overall inertia of the system, \( B \) denotes the general damping constant, \( u \) represents the system input (current command) and \( d \) is assumed to be all disturbances which contribute to the nonlinear term. The following RASMC design is mainly based on Equation (4.2.2).

4.2.2 RASMC Design

Equation (4.2.2) can be reorganized as a function of angular acceleration as:

\[ \ddot{\theta} = \frac{1}{J} \ast (-B \dot{\theta} + u + d) \] (4.2.3)

Assume that the desired arbitrary trajectory of motor position is represented by \( \theta_d \), the error of the position signal can be written as:

\[ e = \theta_d - \theta \] (4.2.4)

Assume that the trajectory is continuous at its first derivation with respect to time, by taking derivative of Equation (4.2.4), the first derivative of the error signal can be represented as:

\[ \dot{e} = \dot{\theta}_d - \dot{\theta} \] (4.2.5)
Based on the error, the sliding surface is defined as:

\[ s = \dot{e} + \lambda \cdot e \tag{4.2.6} \]

Taking derivative of Equation (4.2.6), the slope of sliding surface \( s \) can be represented in the function below.

\[ \dot{s} = \ddot{e} + \lambda \cdot \dot{e} = f(s) \tag{4.2.7} \]

In Equation (4.2.7), \( \lambda \) is selected as a positive scalar, and \( f(s) \) is the reaching law which will determine the convergence routine of sliding surface. Selecting the common reaching law as:

\[ f(s) = -K_1 \cdot S - K_2 \tanh(S) \tag{4.2.8} \]

Combining Equation (4.2.3) to (4.2.8), the input signal can be redefined in the following form.

\[ u = J \left( \ddot{\theta} + \lambda (\dot{\theta}_d - \dot{\theta}) + K_1 S + K_2 \tanh(S) \right) + B \dot{\theta} - d \tag{4.2.9} \]

Now, to cancel the unknown disturbances shown in Equation (4.2.9), one proper replacement of the disturbances should be selected. There are bunch of choices of the disturbance estimation, here one simple method is represented. From Equation (4.2.2), the disturbance can be given as:

\[ d = -J \ddot{\theta} - B \dot{\theta} - u = \psi \tag{4.2.10} \]

In the estimation process, one common method developed by Chia-Shang Liu and Huei Peng \[29\] is to estimate \( \psi(t) \) as \( \psi_{est}(t) \) from the control value at previous step:

\[ \psi_{est} = -J \ddot{\theta} - B \dot{\theta} - u(t - \tau) \tag{4.2.11} \]

In Equation (4.2.11), \( \tau \) is the step size which can be determined by the device resolution. In the experiment, since the resolution of dSPACE is 10[kHz], \( \tau \) is set to be 0.0001. \( \ddot{\theta} \) can also be estimated as:

\[ \ddot{\theta}(t) = [\dot{\theta}(t) - \dot{\theta}(t - \tau)] / \tau \tag{4.2.12} \]

Since the model is not directly analyzed, each parameter in Equation (4.2.11) and (4.2.9) should be replaced by its estimated value:

\[ J = \hat{J}; B = \hat{B} \tag{4.2.13} \]

Substitute Equation (4.2.11) and (4.2.13) into Equation (4.2.9), the control signal
can be given as:

\[ u = \hat{f} \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) + K_1 S + K_2 \tanh(S) \right) - \dot{f} \dot{\theta} + u(t - \tau) \]  \hspace{1cm} (4.2.14)

By combining (4.2.14) and (4.2.10) with Equation (4.2.2), the acceleration is shown as:

\[ \ddot{\theta} = \frac{1}{\hat{J}} \left[ \hat{f} \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) + K_1 S + K_2 \tanh(S) \right) + (\hat{B} - B) \dot{\theta} \right] \]  \hspace{1cm} (4.2.15)

Assuming the estimation error of each parameter can be represented as:

\[ \hat{J} - J; \hat{B} - B = \hat{B} \]  \hspace{1cm} (4.2.16)

Combining Equation (4.2.15) with (4.2.16), the acceleration can be rewritten as:

\[ \ddot{\theta} = \frac{1}{\hat{J}} \left[ \hat{f} \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) + K_1 S + K_2 \tanh(S) \right) - \hat{B} \dot{\theta} \right] \]  \hspace{1cm} (4.2.17)

By subtracting and adding \((\dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}))\) to the right side of Equation (4.2.17), the acceleration can be reformed as:

\[ \ddot{\theta} = -\frac{1}{\hat{J}} \left[ \hat{f} \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \right) + \hat{B} \dot{\theta} \right] \]

\[ + \frac{1}{\hat{J}} \left( K_1 S + K_2 \tanh(S) \right) + \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \]  \hspace{1cm} (4.2.18)

Substituting Equation (4.2.18) into Equation (4.2.7), the derivative of sliding surface can be rewritten in the following form:

\[ \dot{S} = \frac{1}{\hat{J}} \left[ \hat{f} \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \right) + \hat{B} \dot{\theta} \right] - \frac{1}{\hat{J}} \left( K_1 S + K_2 \tanh(S) \right) \]  \hspace{1cm} (4.2.19)

To verify the stability of the function, Lyapunov candidate function is selected as:

\[ V = \frac{1}{2} J s^2 + \frac{1}{2} \vec{\Theta}^T \Gamma^{-1} \vec{\Theta} \]  \hspace{1cm} (4.2.20)

Where,

\[ \vec{\Theta} = \begin{bmatrix} \dot{\theta}_d \\ \dot{\theta}_d \end{bmatrix} \text{ and } \Gamma = \begin{bmatrix} \Gamma_1 & 0 \\ 0 & \Gamma_2 \end{bmatrix} \]  \hspace{1cm} (4.2.21)

Assume that the practical system parameters keep as constant in the experiment, the derivative of the error matrix is:

\[ \dot{\vec{\Theta}} = \begin{bmatrix} \dot{\theta}_d \\ \dot{\theta}_d \end{bmatrix} = \begin{bmatrix} j - \dot{\theta}_d \\ B - \dot{\theta}_d \end{bmatrix} = - \begin{bmatrix} \dot{\theta}_d \\ \dot{\theta}_d \end{bmatrix} \]  \hspace{1cm} (4.2.22)

Based on the equation above, the derivative of Lyapunov function candidate becomes:
\[ \dot{V} = Js\ddot{s} + \bar{T}^T \Gamma^{-1} \hat{T} \]  
(4.2.23)

Substitute Equation (4.2.19) into Equation (4.2.23):

\[
\dot{V} = s \left[ f \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \right) + \bar{B} \dot{\theta} \right] + \bar{T}^T \Gamma^{-1} \hat{T} \\
- \bar{f} \left( K_1 s^2 + K_2 s \cdot \tanh(s) \right) 
\]  
(4.2.24)

Based on the Lyapunov stability theorem, the system is stable under the condition that \( \dot{V} \leq 0 \). Notice that the last term in Equation (4.2.24) is less than zero for positive \( K_1 \) and \( K_2 \), therefore, the stability will be guaranteed if \( s \left[ f \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \right) + \bar{B} \dot{\theta} \right] + \bar{T}^T \Gamma^{-1} \hat{T} \) is set to be zero, using \( \gamma_1 \) and \( \gamma_2 \) to represent each term, Equation (4.2.24) can be simplified as:

\[
\dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) = \gamma_1, \dot{\theta} = \gamma_2 \text{ and } \gamma = \left[ \begin{array}{c} \gamma_1 \\ \gamma_2 \end{array} \right] 
\]  
(4.2.25)

Combining Equation (4.2.25) with (4.2.24), the Lyapunov function can be reformed in the following equation.

\[
\dot{V} = \bar{T} \left( \gamma^T s + \Gamma^{-1} \hat{T} \right) - \bar{f} \left( K_1 s^2 + K_2 s \cdot \tanh(s) \right) 
\]  
(4.2.26)

By setting \( \bar{T} \left( \gamma^T s + \Gamma^{-1} \hat{T} \right) = 0 \) the matrix \( T \) can be represented as:

\[
\hat{T} = -\Gamma \gamma^T s 
\]  
(4.2.27)

The adaption law for the estimation of system parameters can be developed by solving Equation (4.2.27).

\[
\hat{T} = -\left[ \begin{array}{c} \bar{f} \\ \bar{B} \end{array} \right] = -\left[ \begin{array}{cc} \Gamma_1 & 0 \\ 0 & \Gamma_2 \end{array} \right] \left[ \begin{array}{c} \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \\ \dot{\theta} \end{array} \right] [\dot{e} + \lambda e] 
\]  
(4.2.28)

\[
\bar{f} = f_0 + \int_0^t \Gamma_1 \left( \dot{\theta}_d + \lambda (\dot{\theta}_d - \dot{\theta}) \right) (\dot{\theta} + \lambda e) dt 
\]  
(4.2.29)

\[
\bar{B} = B_0 + \int_0^t \Gamma_2 \dot{\theta} (\dot{e} + \lambda e) dt 
\]  
(4.2.30)

In the practical experiment, one thing must be noted that the adaptation law shown in Equation (4.2.29) are no longer applicable with the control signal illustrated in Equation (4.2.14). The reason is that the control law can only guarantee the boundedness of the sliding trajectory, not its asymptotic convergence ability, i.e., s(t)
→ Ω as \( t \to \infty \), where \( \Omega \) is a bounded set [28]. Hence, the integral of inertia represented in Equation (4.2.29) may result in an unbounded value in a finite time. To solve this problem, a projection operator is used in the adaptation law in which the lower and upper bounds of the estimated parameters are required to be known. The projector is introduced as:

\[
\text{Proj}_a[\mathbf{a}]=\begin{cases} 
0, & \text{if } \dot{a}(t) = a_{\max} \text{ and } \mathbf{a} > 0 \\
0, & \text{if } \dot{a}(t) = a_{\min} \text{ and } \mathbf{a} > 0 \\
\mathbf{a}, & \text{otherwise}
\end{cases} \tag{4.2.31}
\]

In Equation (4.2.31), \( a \) represents the estimated parameter (\( \hat{J} \) and \( \hat{B} \) in this problem) with \( a_{\max} \) and \( a_{\min} \) being the upper and lower bounds respectively.

Now, with the projector above, the adaptation law can be modified as:

\[
\dot{J} = J_0 + \int_0^t \text{Proj}_J[\Gamma_1 (\dot{\theta}_d + \lambda(\dot{\theta}_d - \dot{\theta})) (\dot{e} + \lambda e)]dt \tag{4.2.32}
\]

\[
\dot{\hat{B}} = B_0 + \int_0^t \text{Proj}_B[\Gamma_2 (\dot{\theta}(\dot{e} + \lambda e))]dt \tag{4.2.33}
\]

Hence, with the new adaptation law, the estimated parameter is guaranteed to remain bounded by the lower and upper bounds which are initially selected within the bounds, i.e., if the initial estimation value is selected such as \( J_{\min} \leq \hat{J}(0) \leq J_{\max} \), then \( J_{\min} \leq \hat{J}(t) \leq J_{\max} \forall t \in [0, \infty) \). It should be noted that due to the disturbance estimation used in this research, \( \hat{B} \) does not show up in the control law, but it is totally fine for researchers to apply projector method on \( B \).

Furthermore, by combining the new adaptive law (4.2.32) and (4.2.33) to the RASMC design, the bound of the steady-state error [28] can be explicitly derived as:

\[
|e_{ss}(t)| \leq \frac{k_2 \epsilon}{\lambda(K_1 \epsilon + K_2)} \tag{4.2.34}
\]

And the system holds following properties.

\[
\dot{a}(t) \chi(t) \leq \dot{a}(t) \text{Proj}_a[\chi(t)] \tag{4.2.35}
\]

In Equation (4.2.34), \( \epsilon \) is a small positive parameter adjusting the rate of switching operation. Since the RASMC designed in this research is using ‘tanh’ function, \( \epsilon \) can be roughly regarded as 1. The detailed prove for Equation (4.2.34) is discussed in Appendix E.
4.3 Simulation Test for RASMC

4.3.1 Simulation Model Construction

Using Simulink software in Matlab, the simulation model which combines the RASMC and piecewise linear system plant can be represented in Figure (4.1).

Figure 4.1: System with RASMC Controller
In Figure 4.1, the input saturation block is set to have the limitation of $\pm 5.5$, which is the same with the value used in PI controller plant. The blocks ‘Lam’, ‘K1’, ‘K2’, ‘Gam1’ and ‘Gam2’ are used to upload the value of parameters $\lambda, K_1, K_2, \Gamma_1$ and $\Gamma_2$ into the simulation system from workspace. The step block is used to set the desired motor speed in steady state.

The subsystem in ‘Perturbation’ block in Figure 4.1 can be shown in Figure 4.2.

![Simulink Model for Disturbance Estimation](image)

**Figure 4.2: Simulink Model for Disturbance Estimation**

In Figure 4.2, the perturbation estimation subsystem is built upon Equation (4.2.11). The step size of the ‘Memory’ block is set to be 10kHz which is identical to the resolution of experimental devices.

The subsystem in ‘Estimator’ block in Figure 4.1 can be shown in Figure 4.3.
Figure 4.3: Simulink Model for Parameter Estimation

In Figure 4.3, the simulation model for parameter estimation is reconstructed based on Equation (4.2.32) and (4.2.33). The blocks ‘$J_0$’ and ‘$B_0$’ are built up to insert the initial value of J and B. In the simulation process, they are both set to be zero which indicates that no information can be acquired about the system overall inertia and damping factor. The two integral limitation blocks are set as the projector in the controller design discussed previously.

The subsystem in ‘Speed Reference’ block in Figure 4.1 can be shown in Figure 4.4.

Figure 4.4: Simulink Model for Speed Reference

In Figure 4.4, the input signal from the outside step block would be treated as the reference speed. By taking derivative and integral of this signal, the angular position
and acceleration can be generated.

The subsystem in ‘Plant’ block in Figure 4.1 can be shown in Figure 4.5.

---

**Figure 4.5: Simulink Model for Plant**
In Figure (4.5), the plant which represents the overall system is exactly same as what has been used in PID controller test, except for the output signal would be all position, speed and acceleration.

The subsystem in ‘Controller’ block in Figure 4.1 can be shown in Figure 4.6.
Figure 4.6: Simulink Model for RASMC Controller

In Figure 4.6, the model is reconstructed based on Equation (4.2.9) through (4.2.14).
During the process of choosing the suitable reaching law, both ‘tanh’ and ‘sign’ blocks have been tested. From the test result, ‘tanh’ block gives out a better performance since the curve of function tanh does not have any break point which would cause infinite derivative value. Thus the ‘tanh’ block is chosen to be used in the sliding surface reaching law.

4.3.2 Parameters Selection Process

Before running the simulation model, the parameters for RASMC should be determined. However, there is no sophisticated theory to determine all the five parameters \((\lambda, K_1, K_2, \Gamma_1 \text{ and } \Gamma_2)\) at one time, not to mention find the optimal value for them. For selecting the suitable values of all five parameters, the simulation result is taken into consideration. In the parameter choosing process, the result of simulation response is assumed to be the same as real-time experiment and the values will be determined based on the simulation performances. The general steps can be shown as:

1) Separate all parameters into several groups
2) Pick up only one group of gains which is going to be changed while running the test experiment, fix all the other parameters
3) Vary the values of parameters in a relative big range
4) Collect output data coming from Simulink and shrink the range of variation
5) Repeat step 3 and 4 to reach the optimal value
6) Repeat step 2 until all the parameters are determined
7) Assess the performance based on the parameters determined using previous steps

The process is tedious and no theory can support the effectiveness of such method, however, this is the only way applicable in the research. The method of parameters determination can be assigned as the future task in control theory. From Cho’s previous test [10], the five parameters are initially chosen as:

\[
\lambda = 25; \ K_1 = 100; \ K_2 = 3; \ \Gamma_1 = \Gamma_2 = 5
\]  

(4.3.1)
However, during the real-time experiment as well as the simulation process, the parameters chosen by Cho gives a poor performance. Even after a strong filter, a huge oscillation of steady-state speed response still shows up in speed response. To determine the suitable values for all the gains in the controller, the number should be redefined.

By examining Equation (4.6), $\lambda$ actual determines the root of solution to $s$. The solution of Equation (4.6) would be an exponential term, a big $\lambda$ would cause the solution goes to zero much faster. Before determining the parameter $\lambda$, other parameters should be set at first. In the simulation test for $\lambda$, all the other gains are chosen as the same with Equation (4.3.1) except for $\lambda$ itself.

For no lost in generality, the range of $\lambda$ is chosen from 1 to 1000 in the first simulation test, the overshoot and the settling time for the 80[rad/s] speed responses are shown in Figure 4.7 and Figure 4.8.
As seen in Figure 4.7 and Figure 4.8, the system would give a better performance for both overshoot and settling time if \( \lambda \) is chosen between 0 to 100. Based on this analysis, the performance for \( \lambda \) between 0 to 100 was tested. The overshoot and settling time for 80[rad/s] speed responses are shown in Figure 4.9 and Figure 4.10.
Figure 4.9: Overshoot for 80[rad/s] Speed Response

Figure 4.10: Settling Time for 80[rad/s] Speed Response
From Figure 4.9 and Figure 4.10, the system would give a better performance of overshoot for $\lambda$ between 0 to 10, and a better performance of settling time for $\lambda$ between 10 and 80. However, based on the simulation experience, the overshoot also has a strong relationship with all other four parameters so that no finalized conclusion of $\lambda$ can be demonstrated. Next the overshoot and the settling time Simulink test for the 180[rad/s] speed response performance are shown in Figure 4.11 and Figure 4.12.

Figure 4.11: Overshoot for 180[rad/s] Speed Response
In Figure 4.11 and Figure 4.12, the system would give a better performance of settling time if \( \lambda \) is chosen between 0 to 100 and all values between 0 and 1000 would guarantee the same performance of overshoot around 40 percent. This is mainly because of the poor choice of \( K \) and \( \lambda \). Next, the performance for \( \lambda \) between 0 to 100 will be tested. The overshoot and settling time for 80[rad/s] speed responses are shown in Figure 4.13 and Figure 4.14.
Figure 4.13: Overshoot for 180[rad/s] Speed Response

Figure 4.14: Settling Time for 180[rad/s] Speed Response
From Figure 4.13, the range of the suitable $\lambda$ based on the overshoot is between 0 to 10 which will produce less than 40% overshoot in the speed response process. However, based on Figure 4.14, the range of the suitable $\lambda$ based on the settling time is between 10 to 80 which would make the settling time less than 8[sec].

Combine all the information from the data collected in the Simulation result, the suitable range of the $\lambda$ which would give a better result both in 80[rad/s] and 180[rad/s] can be chosen from 0 to 100. The number of $\lambda$ is chosen to be 80 for the later test on other parameters. Although 80 would not give a good performance for overshoot of 80[rad/s] reference speed, the gain can be modified in future.

Next, the value of $K_1$ and $K_2$ should be determined. Fix all the other gains as:

$$\lambda = 80; \quad \Gamma_1 = \Gamma_2 = 5$$

(4.3.2)

For no lost in generality, the range of $K_1$ and $K_2$ is chosen from 1 to 100 in the first Simulation test, the overshoot and the settling time for the 80[rad/s] speed response is shown in Figure 4.15 and Figure 4.16.

Figure 4.15: Overshoot for 80[rad/s] Speed Response
Figure 4.16: Settling Time for 80[rad/s] Speed Response

From Figure 4.15 and Figure 4.16, the common rule can be found out that $K_1$ is predominant in the Simulink test. The change of value of $K_2$ would not cause too much difference both on settling time and overshoot. This can be briefly verified by Equation (4.2.34). In the sliding process, when the absolute error becomes small enough, the slope $\epsilon$ of ‘tanh’ function grows rapidly which will make $K_1$ becomes dominant in the convergence process. This phenomenon can also be seen in the following experiment verification.

According to the data collected above, a relative low $K_1$ would guarantee a better performance. After several tests for the range of parameters, the performance for $K_1$ and $K_2$ from 0 to 1 are shown in Figure 4.17 and Figure 4.18.
Figure 4.17: Overshoot for 80[rad/s] Speed Response

Figure 4.18: Settling Time for 80[rad/s] Speed Response
Figure 4.17 and Figure 4.18 again verified the assumption that $K_1$ play a magnificent role in the Simulation result. From Figure 4.17, the overshoot grows linearly with respect to $K_1$, and $K_2$ seems do not have the effect on the result. From Figure 4.18, the settling time would grow rapidly after $K_1$ passes 0.05, then decrease slowly after $K_1$ passes 0.2. Combined with the overall respond performance on 80[rad/s], the value of gains $K_1$ and $K_2$ which would be suitable to the controller is between 0 to 0.05.

Besides the 80[rad/s] reference speed, the data response of 180[rad/s] has also been collected to provide information on high speed region. For no lost in generality, the range of $K_1$ and $K_2$ is chosen from 1 to 100 in the first Simulation test, the overshoot and the settling time for the 180[rad/s] speed response is shown in Figure 4.19 and Figure 4.20.

![Figure 4.19: Overshoot for 180[rad/s] Speed Response](image)
Figure 4.20: Settling Time for 180[rad/s] Speed Response

Judging from Figure 4.19 and Figure 4.20, the suitable value for $K_1$ and $K_2$ again comes out from a relative small number. For the high reference speed input, the change of $K_2$ again does not have too much influence on the system behavior both on overshoot and settling time. However, the different value of $K_1$ does make a different result. The overshoot and settling time grows rapidly in the region between 0 to 2 and becomes stable after that. Based on this information, the performance for K1 and K2 from 0 to 1 are shown in Figure 4.21 and Figure 4.22.
Figure 4.21: Overshoot for 180[rad/s] Speed Response

Figure 4.22: Settling Time for 180[rad/s] Speed Response
Similar with 80[rad/s] reference speed response, a small $K_1$ would guarantee the good system behavior based on Figure 4.21 and Figure 4.22. As shown in Figure 4.21, the overshoot grows with respect to $K_1$ from 0 to 1, in Figure 4.22, the settling time grows quickly after $K_1$ passes 0.05 and decreases slowly after $K_1$ goes over 0.2. Combined with the conclusion from the 80[rad/s] reference speed response, the suitable value of gains $K_1$ and $K_2$ is between 0 to 0.05. Combine all the information about $K_1$ and $K_2$ provided above, both of the parameters are set to be 0.002 for the following experiment.

Finally, the only set of gains left to be determined is $\Gamma_1$ and $\Gamma_2$. Fix all other parameters value as:

$$\lambda = 80; \quad K_1 = K_2 = 0.002 \quad (4.3.3)$$

For no lost in generality, the range of $\Gamma_1$ and $\Gamma_2$ are chosen from 0 to 10 in the first Simulation test, the overshoot and the settling time for the 80[rad/s] speed response is shown in Figure 4.23 and Figure 4.24.

![Figure 4.23: Overshoot for 80[rad/s] Speed Response](image-url)
From Figure 4.23 and Figure 4.24, the overshoot and settling time would not change too much for different $\Gamma_1$ and $\Gamma_2$ from 0 to 10, but they do decrease if $\Gamma_1$ and $\Gamma_2$ are chosen to be relatively small. For finding the optimal value, the overshoot and settling time of 80[rad/s] reference speed for $\Gamma_1$ and $\Gamma_2$ between 0 to 1E-6 are given in Figure 4.25 and Figure 4.26.
Figure 4.25: Overshoot for 80[rad/s] Speed Response

Figure 4.26: Settling Time for 80[rad/s] Speed Response
As the data shown in Figure 4.25 and Figure 4.26, the change of $I_2$ does not have an influence on the Simulation result, and the change of $I_1$ would produce a small effect on the Simulation result. Based on the performance of both overshoot and settling time for 80[rad/s] reference speed response, the gains value of $I_1$ and $I_2$ should be chosen in a narrow range of value.

Except the 80[rad/s] reference speed, the data response of 180[rad/s] has also been collected to provide information on high speed region. For no lost in generality, the range of $I_1$ and $I_2$ is chosen from 0 to 10 in the first Simulation test, the overshoot and the settling time for the 180[rad/s] speed response is shown in Figure 4.27 and Figure 4.28.

![Figure 4.27: Overshoot for 180[rad/s] Speed Response](image-url)
Figure 4.28: Settling Time for 180[rad/s] Speed Response

Similar with the data of 80[rad/s] reference speed, the difference of value of $\Gamma_2$ and $\Gamma_1$ would not influence too much on the system behavior both on overshoot and settling time. However, the system behavior does have a small difference for different $\Gamma_1$, the overshoot grows a little for $\Gamma_1$ changes from 0 to 0.5 and the settling time drops a little bit for $\Gamma_1$ changes from 0 to 0.5.

To find the most suitable value, the overshoot and settling time of 80[rad/s] reference speed for $\Gamma_1$ and $\Gamma_2$ between 0 to 1E-5 are given in Figure 4.29 and Figure 4.30.
Figure 4.29: Overshoot for 180[rad/s] Speed Response

Figure 4.30: Settling Time for 180[rad/s] Speed Response
As the data shown in Figure 4.29 and Figure 4.30, the overshoot of the system response increases as the $\Gamma_1$ grows and the settling time decreases as $\Gamma_1$ grows from 0 to 0.05E-5.

Combined with the data analysis for 80[rad/s] reference speed response and 180[rad/s] reference speed response, the suitable $\Gamma_1$ and $\Gamma_2$ which would give the system a good overshoot and settling time behavior should be chosen as a small number.

### 4.3.3 Simulation Result

Based on the system parameters analysis above, the gains for the RASMC in the Simulation test is chosen as:

$$\lambda = 80; \ K_1 = K_2 = 0.002; \ \Gamma_1 = \Gamma_2 = 2E - 7$$

Inserting all the value of parameters in the simulation system, the behavior of the system for different reference input speed (partial) can be shown in Figure 4.31 and Figure 4.32. The step time is set to be 1[sec].

![Speed response of RASMC](image)

**Figure 4.31: Speed Response Using RASMC**
4.4 Conclusion of RASMC

From Equation (4.3.4), the parameters selected based on the Simulink model have a huge difference with the original value chosen by Cho. Judging from the simulation speed response result shown in Figure 4.31, RASMC with the new gains would perform a speed response which meets the performance criteria with almost no overshoot and a short settling time. From the control input data shown in Figure 4.32, the control signal maintains the maximum value if the speed response is less than the reference speed and remains bounded in the steady state with certain oscillation especially for low speed. The reason for this phenomenon is coming from the controller concept. As introduced in section 4.1, SMC is designed based on the combination of multiply control structure and the structure of controller will change so that $s$ will go back and forth around the sliding surface towards to original point, and the input signal will go to the steady state value when $s$ reaches the original point. Fortunately, the control signal is oscillating
within a small range which is acceptable.

Based on the input signal and the speed reference, RASMC designed in this chapter can meet the requirement of the desired performance criteria.
Chapter 5. Close-Loop Experiment

5.1 Experimental Setup

5.1.1 Experimental Devices Setup

Before collecting data from the close-loop experiment, the experimental devices should be set up in the correct condition.

The overall experimental devices are built up to be the same with the open-loop experimental test except for pulling off the torque sensor from both sides and the encoder from the output side. Since the GBD transmission is treated as a rigid body, the transmission ratio is taken as 40:1 without considering kinematic error. The only data which are collected and sent back to computer are motor speed and current input.

5.1.2 Computer Software Setup

The computer software used in the close loop experiment test is identical to the open loop experiment test. Simulink software was still used to build up the simulation model of the feedback control loop, Control Desk was mainly responsible for data collecting and real-time information exhibition, Driveware software functioned as the power switch for digital motor servo.

5.2 PID Controller Experiment

5.2.1 Simulink Model Construction

The Simulink model using PID controller is shown in Figure 5.1.
All the data capture settings are the same with open loop test except for the command input. In Figure 5.1, the input signal is generated after PID control block (inside black frame), after calculating the control signal by PID algorithm, the signal will be sent to Controller Board through DS1104DAC_C1 Simulink block. By changing the values of $K_p$ and $K_i$, different PID controllers can be implemented to the experiment.

5.2.2 Experimental Result

The experimental data of speed response and the error of speed response for 80[rad/s] and 180[rad/s] using the PID controller designed based on region 1, region 2 and the advanced PID controller can be shown in Figure 5.2 through Figure 5.7.
Figure 5.2: Experimental Speed Response Using PID Controller

Figure 5.3: Experimental Speed Response Error
Figure 5.4: Experimental Speed Response Using PID Controller

Figure 5.5: Experimental Speed Response Error
Figure 5.6: Experimental Speed Response Using PID Controller

Figure 5.7: Experimental Speed Response Error
5.2.3 Experiment Conclusion

For convenience of comparison, the experimental speed response and the simulation speed response using all three sets of PID controller have been combined in one plot shown in Figure 5.8 to Figure 5.10

![Combined Speed Response Using PID Controller](image)

**Figure 5.8: Combined Speed Response Using PID Controller**
Figure 5.9: Combined Speed Response Using PID Controller

Figure 5.10: Combined Speed Response Using PID Controller
From Figure 5.8 to Figure 5.10, the experimental speed response is almost the same as the simulation result for all the PID controls designed in Chapter 3.

As seen in Figure 5.8, the PID controller using pole-zero cancellation based on the speed region 1 would guarantee a good performance for the reference speed under 105[rad/s]. For the reference speed higher than 105[rad/s], the speed response would first experience a large overshoot and then decreases to the reference speed after finite time. This behavior is easy to understand since the gains of PID control are chosen based on the speed region 1 and cannot fit to another speed region. After the speed passes the switch value, the PID controller would not be suitable for the new system function anymore.

Same thing happens in Figure 5.9. The PID controller using pole-zero cancellation based on speed region 2 would permit a good performance for the reference speed above 105[rad/s]. However, for the speed reference under the switch value, the system would experience a relative long settling time. Since the PID parameters are chosen based on the second system function, they would not be suitable for the first system function as the long settling time would not meet the performance criteria.

As for the advanced technique developed for pole-zero cancellation method, the experimental speed response can guarantee the good behavior of the system for whole speed region as shown in Figure 5.10. The experimental result of the speed response using advanced PID controller is almost the same as the simulation result which would acquire a short settling time and no overshoot speed response for all reference speed.

All the PID controller designed in Chapter 3 would give a same experimental speed response as simulation result, however, only the PID controller using advanced pole-zero cancellation would meet the performance criteria.
5.3 RASMC Experiment

5.3.1 Simulink Model Construction

The Simulink model of RASMC feedback system is shown in Figure 5.11.

![Simulink Model of RASMC System](image)

Figure 5.11: Simulink Model of RASMC system

Same as PI control test, the data measured by encoder is sent back to the RASMC control and the input signal is calculated by the RASMC control algorithm. After generating the command input, the signal will be sent to Controller Board using DS1104DAC_C1 block.

5.3.2 Experimental Result

The experimental data of speed response and current command using RASMC designed in Chapter 4 is shown in Figure 5.12 and Figure 5.13.
Figure 5.12: Experimental Speed Response Using RASMC

Figure 5.13: Experimental Control Input Using RASMC
5.3.3 Experiment Conclusion

The combined speed response from Simulink result and experimental data can be shown in Figure 5.14.

![Combined Speed Response](image)

**Figure 5.14: Combined Speed Response**

Judging from Figure 5.14, the RASMC designed in Chapter 4 can provide a good speed response experimental result. The simulation result almost match with the experimental response.

However, one thing should be noticed that there is an obvious steady-state error for 100[rad/s] reference response in the experimental result. The plausible reason for this is the speed resonance happens in the system. For a certain transmission system, such resonance behavior is popular and unpredictable. Unfortunately, the consequence of the resonance experienced by the whole system is the dramatic dissipation of energy which occurs in certain velocity region. As can be seen in Figure 2.8, the speed response of the open loop system for 4.5[A] input experienced a relative more speed resonance.
compared with other speed response. Since the resonance region is around 100[rad/s] shown in Figure 2.8 and the energy would be consumed, it is reasonable that there is a steady-state error for 100[rad/s] reference input.

From Figure 5.13, the control signal also meets the requirement. Due to the unmodeled nonlinearities such as disturbances and perturbations exist in the system, the current command oscillates through the operation process and never settles down. Such oscillation acts more obviously for low speed control task since the ratio of disturbances to system dynamic is relatively high. However, since the control input would change in a certain bounded range, the result is acceptable for experimental application.
Chapter 6. Piecewise Behavior Assessment

6.1 Piecewise Behavior Description

In the experiment process, the ‘Jump behavior’ which happens both in open-loop and close-loop test becomes attractive. As shown in Figure 2.8, in the open loop test result, under a constant current input, the speed response of the system ‘jump’ to another speed level when it passes a certain speed limit.

During the process of studying this behavior, another similar experimental result which also contains the ‘Jump behavior’ received a lot interest. This behavior is mentioned in Tuttle’s research result as shown in Figure 7.1 [30].

Figure 6.1: Speed Response of Harmonic Drive
Figure 7.1 depicts the speed response of a harmonic drive system tested in Tuttle’s research. From the plot, two ‘Jump behavior’ can be noticed, one is around 1.5[krpm] and another one is around 3.8[krpm]. In Tuttle’s conclusion, he credits this behavior to the resonance of speed. This is a plausible answer since the speed response around the switch limitation is oscillating violently. After the speed goes through the switch limitation, the speed response becomes much more stable with a less resonance behavior. The identical behavior can be observed in our research from the open-loop test. As seen in Figure 2.8, current commands ranging from 4.6 to 5.0 amps all produces the ‘Jump behavior’. At instant the velocity reaches the switch limitation, which is roughly 105[rad/s], it would oscillate violently around the limitation. As the average speed continues to grow up, it jumps out the resonance region and behaves like another kind of system. Based on this conclusion, the first assumption of the ‘Jump behavior’ can be attribute to the system resonance.

The components attribute to the resonance speed region can be in variety. The main component agreed in recent research is the rapid change of gear tooth friction happens in tooth contact surface. In recently years, tooth friction of spur gear transmission system has received large attention by researchers and different kinds of analysis based on the influence of tooth friction to the practical plant has been published [31, 32, 33, 34, 35]. As the constant step command of current brings the speed response to the region of resonance, the rotation frequency would reach the same value as the frequency of kinematic error, the torque fluctuates increasingly along with the resonance. With the increase of the torque amplitude, the energy dissipated by the Coulomb-like friction mechanism would increase. Due to the growth of the torque fluctuation, the average of the speed would increase but keep roughly the same since the torque generated by the system are mostly used to cover the energy dissipates by tooth surface friction. Because of the self-reinforce model of the frictional mechanism, such transmission system would maintain its speed response at the resonance region for certain time.
When the average speed exits the resonance region, unpredictable jump behavior would happen as the partnership of the resonance behavior. The speed response of the current command at 4.6[A] jumps to 160[rad/s] after it escaped from the resonance region. Due to the velocity vibration behavior discussed above, the speed is restrained in the certain resonance region combined with an increasing energy dissipation. When the speed overcomes the resonance, the energy consumed by the tooth friction drops immediately due to the decline of both the frequency and the amplitude of the torque oscillation. With all the energy loses minimized after the velocity escapes away from the resonance region, there is no surprise that the speed grows drastically.

Besides the significant impact of tooth friction, certain amounts of other plausible reasons have been studied and published which contributes to the unpredictable jump behavior in the gear transmission system. Among them contains the study focused on dynamic error [37, 38, 39] and hysteresis behavior on torsion stiffness [40, 41].

Although this theory sounds highly plausible, experimental test is still needed to verify this assumption. According to the analysis above, the main reason which is responsible for the unexpected jump behavior should be the kinematic error inside GBD transmission. Thus, the first step to verify the source of the jump behavior is to identify the component which is responsible for this behavior.

In the verification test, this plant is separated into three different parts: 3-phase brushless DC motor, GBD transmission and the hysteresis brake. In the practical experimental verification process, the 3-phase brushless DC motor will be replaced by a traditional DC motor, and the hysteresis brake will be replaced by an inertia item. By changing the combination of the system, the reasonable part which causes the system to perform the jump behavior will be figured out.
6.2 Experimental Verification

Table 6.1: Testing Cases

<table>
<thead>
<tr>
<th>System No</th>
<th>BLDC</th>
<th>GBD</th>
<th>Brake</th>
<th>Jump</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>2</td>
<td>○</td>
<td>○</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>3</td>
<td>×</td>
<td>○</td>
<td>○</td>
<td>×</td>
</tr>
<tr>
<td>4</td>
<td>×</td>
<td>○</td>
<td>×</td>
<td>×</td>
</tr>
</tbody>
</table>

It should be noted that based on the theory discussed in previous sections, the change of the actuator and load should not influence the resonance region since it only related with the inherent characteristic of the GBD transmission. In other word, when the BLDC motor and the hysteresis brake are replaced by other actuators and inertia load, the resonance region and the unexpectable jump behavior will be the same if it comes from the GBD transmission part.

However, based on the experimental result, Table 6.1 tells the different story. As seen in Table 6.1, system No.1, which is exactly the original system with BLDC and hysteresis brake, performs the ‘Jump behavior’ as expected. However, System No.2 to system No.4, which also contains the GBD transmission part in the system, did not duplicate the same speed response behavior as the original system.

By scrutinizing the combination of the system, one thing can be found out that only when all BLDC, GBD and HB exist in the system does the speed response performs the ‘Jump behavior’. From this aspect, the assumption that ‘Jump behavior’ only has the relationship with the GBD transmission part is not reasonable. Judging from the performance, the ‘Jump behavior’ is the collaborate product by BLDC motor, GBD transmission and Hysteresis brake.

If more precise conclusion wants to be made, the GBD transmission should be replace by another joint drive system which has the same torque ratio. Unfortunately,
Piezoactive Lab cannot provide such replacement in a short time, so the right explanation for this phenomenon remains unknown.

6.3 Conclusion

‘Jump behavior’, which is widely discovered in the transmission joint devices, is mostly caused by the speed resonance along with energy dissipation of friction between tooth surface. When this behavior appears in the practical application, it means that the efficiency of the transmission is constrained by the energy dissipation and the speed respond is influenced by the speed resonance. If the economic effectiveness of the joint drive system wants to be improved to a higher level, this unexpected jump is worth studying.

However, for the whole system combined with BLDC, GBD and HB, the regular explanation cannot match with the experiment verification. Through different combination of actuator, transmission and load, the unexpected jump in speed response is certainly not only related with the pure GBD transmission. The real reason for this behavior remains unknown and it will be singed as a potential future work.
Chapter 8. Future Work

Although the close-loop system can perform exceptional using both PID and RASMC control algorithm, it lacks the ability to give out the optimal behavior in the close loop behavior. High level control algorithm such as optimal control is needed for the system in a verity of commercial application. In other words, the control algorithm for using minimum control effort to finish the performance tasks is required to be obtained. Besides, the physical reason which causes the ‘Jump behavior’ in the system behavior remains unknown. Although a brief verification test and a short discussion has been given in chapter 6, the details of the unexpectable jump behavior remains to be explored.

Some of the immediate steps which can be implemented to the GBD transmission system can be:

➢ Research and develop an optimal controller to replace RASMC and PID control. The optimal control algorithm should be able to guarantee the stability of the system, in the meantime, the control law could satisfy certain cost functional to meet the different requirement of the practical application.

➢ Complete the stability prove of the PID control combined with the piecewise linear model developed in this thesis.

➢ Explore experimental prove and theory which can explain the details of ‘Jump behavior’ happens at roughly 105[rad/s].

➢ Study the gear tooth surface friction and develop dynamic model of each tooth contact surface. Since the tooth friction contribute largely to the resonance behavior, a right model of the tooth friction would make the system model more accurate and plausible than which has been developed in this thesis.
Appendix

Appendix A: Curve Fitting

Open Matlab, click on the ‘Curve Fitting’ bottom in ‘App’ choice.

**Figure A.1: Curve Fitting App**

The interface of the software can be shown in Figure A.2.

**Figure A.2: Interface of Curve Fitting**

The data choice in the left side of the software can be directly read from workspace. After the total torque of the system been calculated by Matlab and stored in workspace, they can be immediately used in the App. The curve fitting result and its corresponding math equation are shown in Figure A.3 through Figure A.6.
$T_{total} = 0.006098 \cdot i + 0.00261$ \hspace{1cm} (A.1)

$T_{tf1} = 0.0001636 \cdot \dot{\theta} + 0.0134$ \hspace{1cm} (A.2)
Figure A.5: Current-Total Torque in Speed Region 2

\[ T_{total} = 0.002849 \times i + 0.01902 \] (A.3)

Figure A.6 Speed-Total Torque in Speed Region 2

\[ T_{ft} = 2.306 \times 10^{-5} \times \dot{\theta} + 0.02842 \] (A.4)
Since the weighting function for each curve has been selected by software automatically and the error is constraint to no more than 1%, no additional adjustment is needed. Equation (A.1) through Equation (A.4) can be directly used in system reconstruction of Chapter 2
Appendix B: PID Controller Design

Recall the system transfer function in speed region 1 as G1
\[ G_1 \text{ Controllable } = \frac{0.0061}{0.00012 S + 0.000163} ; \quad G_1 \text{ Uncontrollable } = \frac{0.01084}{0.00012 S + 0.000163} \]

Recall the system transfer function in speed region 2 as G2
\[ G_2 \text{ Controllable } = \frac{0.002849}{0.00005 S + 2.306E-5} ; \quad G_2 \text{ Uncontrollable } = \frac{0.0094}{0.00005 S + 2.306E-5} \]

Design the PID controller using the following form
\[ P(D) = K_p + \frac{K_i}{S} + K_d S \]

The feedback system in speed region 1 using PID controller can be shown as
\[ X_1 = \text{simplify} \left( \frac{\text{PID-G1 Controllable}}{1 + \text{PID-G1 Controllable}} \right) ; \quad Y_1 = \text{simplify} \left( \frac{G_1 \text{ Uncontrollable}}{1 + \text{PID-G1 Controllable}} \right) ; \]
\[ X_1 = \frac{61 \cdot K_d S^2 + 61 \cdot K_d + 61 \cdot K_p S + 61 \cdot K_i}{(61 \cdot 0.0000000 K_d + 1.2) S^2 + (61 \cdot 0.0000000 K_p + 1.63) S + 61 \cdot 0.0000000 K_i} \]
\[ Y_1 = \frac{(152.5 \cdot K_d + 3 \cdot 0.0000000) S^2 + (152.5 \cdot K_p + 4.075) S + 152.5 \cdot K_i}{271.1 S} \]

The feedback system in speed region 2 using PID controller can be shown as
\[ X_2 = \text{simplify} \left( \frac{\text{PID-G2 Controllable}}{1 + \text{PID-G2 Controllable}} \right) ; \quad Y_2 = \text{simplify} \left( \frac{G_2 \text{ Uncontrollable}}{1 + \text{PID-G2 Controllable}} \right) ; \]
\[ X_2 = \frac{2849 \cdot K_d S^2 + 2849 \cdot K_d + 2849 \cdot K_p S + 2849 \cdot K_i}{(2849 \cdot 0.0000000 K_d + 50.) S^2 + (2849 \cdot 0.0000000 K_p + 23.06) S + 2849 \cdot 0.0000000 K_i} \]
\[ Y_2 := \frac{188. S}{56.98 \cdot Kd \cdot S^2 + 56.98 \cdot Kp \cdot S + 56.98 \cdot Ki + S^3 + 0.4612 \cdot S} \]

For reducing the order of the feedback transfer function, \( Kd \) is chosen to be zero, the new PID controller can be defined as

\[ \text{PID\_New} := Kp + \frac{Ki}{S} \]

\[ \text{PID\_New} := Kp + \frac{Ki}{S} \]

The new feedback system in speed region 1 using PID controller can be shown as

\[ X_{1\_t} := \text{simplify}\left(\frac{\text{PID\_New} \cdot G_1 \text{Controllable}}{1 + \text{PID\_New} \cdot G_1 \text{Controllable}}\right) \]

\[ Y_{1\_t} := \text{simplify}\left(\frac{G_1 \text{Uncontrollable}}{1 + \text{PID\_New} \cdot G_1 \text{Controllable}}\right) \]

\[ X_{1\_t} := \frac{61 \cdot Kp \cdot S + 61 \cdot Ki}{1.2 \cdot S^2 + (61.00000000 \cdot Kp + 1.63) \cdot S + 61.00000000 \cdot Ki} \]

\[ Y_{1\_t} := \frac{271 \cdot S}{3.00000000 \cdot S^2 + (152.5 \cdot Kp + 4.075) \cdot S + 152.5 \cdot Ki} \]

The new feedback system in speed region 2 using PID controller can be shown as

\[ X_{2\_t} := \text{simplify}\left(\frac{\text{PID\_New} \cdot G_2 \text{Controllable}}{1 + \text{PID\_New} \cdot G_2 \text{Controllable}}\right) \]

\[ Y_{2\_t} := \text{simplify}\left(\frac{G_2 \text{Uncontrollable}}{1 + \text{PID\_New} \cdot G_2 \text{Controllable}}\right) \]

\[ X_{2\_t} := \frac{2849 \cdot Kp \cdot S + 2849 \cdot Ki}{50 \cdot S^2 + (2849.000000 \cdot Kp + 23.66) \cdot S + 2849.000000 \cdot Ki} \]

\[ Y_{2\_t} := \frac{188. S}{56.98 \cdot Kp + 0.4612 \cdot S + 56.98 \cdot Ki} \]
Appendix C: Matlab Script for RASMC Parameter Determination

1) Parameter λ determination code:

clear;
% Gain Part 
Lmin=0.1;Lmax=100;
R=180;
K1=100;
K2=3;
Gam1=5;
Gam2=5;
J0=0.00012;
B0=0.0001636;
N=200;

for i = 1:N

    P(i) = Lmin+(Lmax-Lmin)*i/N;
    Lam=P(i);
    sim('RobustAdaptiveSlidingModeControl');

    temp = get(speed);
    speed = temp.Data;
    Time = temp.Time;

    overshoot(i)=(max(speed)-R)/R;

    settlingtime(i) = 0;
    for n = 1:size(Time)
        if abs(speed(n)-R)>0.02*R
            settlingtime(i) = Time(n);
        end
    end
end

title('settlingtime');xlabel('Lam');
figure()
plot(P,settlingtime);
figure()
plot(P,overshoot);
figure()
title('overshoot');xlabel('Lam');
2) **Parameter $K_1$ $K_2$ determination code:**

clear;

% Gain Part %
K1min=1E-10;K1max=1;  
K2min=1E-10;K2max=1;  
Lam=80;  
Gam1=5;  
Gam2=5;  
J0=0.00012;  
B0=0.0001636;  
N=20;M=20;  
R=180;  
for i = 1:N  
    for j=1:M  
        P(i,j) = K1min+(K1max-K1min)*(i-1)/N;  
        D(i,j)=K2min+(K2max-K2min)*(j-1)/M;  
        K1=P(i,j);  
        K2=D(i,j);  
        sim('RobustAdaptiveSlidingModeControl');  
        temp = get(speed);  
        speed = temp.Data;  
        Time = temp.Time;  
        overshoot(i,j)=(max(speed)-R)/R;  
        settlingtime(i,j) = 0;  
        for n = 1:size(Time)  
            if abs(speed(n)-R)>(R*0.02)  
                settlingtime(i,j) = Time(n);  
            end  
        end  
    end  
end  
figure()  
mesh(P,D,settlingtime);  
title('settlingtime');xlabel('K1');ylabel('K2');  
figure()  
mesh(P,D,overshoot);  
title('overshoot');xlabel('K1');ylabel('K2');
3) Parameter $\Gamma_1$, $\Gamma_2$ determination code:

clear;
% Gain Part %
R=80;
Lam=80;
K1=0.002;
K2=0.002;
Gam1min=1E-9; Gam1max=10;
Gam2min=1E-9; Gam2max=10;
N=20; M=20;
J0=0.00012;
B0=0.0001636;
% Parameters %
for i = 1:N
    for j=1:M
        P(i,j) = Gam1min+(Gam1max-Gam1min)*(i-1)/N;
        D(i,j)=Gam2min+(Gam2max-Gam2min)*(j-1)/M;
        Gam1=P(i,j);
        Gam2=D(i,j);

        sim('RobustAdaptiveSlidingModeControl');

        temp = get(speed);
        speed = temp.Data;
        Time = temp.Time;

        overshoot(i,j) = (max(speed)-R)/R;
        settlingtime(i,j) = 0;
        for n = 1:size(Time)
            if abs(speed(n)-R)>(R*0.02)
                settlingtime(i,j) = Time(n);
            end
        end
    end
end

figure()
mesh(P,D,settlingtime);
title('settlingtime');xlabel('Gam1');ylabel('Gam2');
figure()
mesh(P,D,overshoot);
title('overshoot');xlabel('Gam1');ylabel('Gam2');
Appendix D: Stability Prove Using Routh-Hurwitz Criteria

In control system theory, the Routh-Hurwitz stability criterion is a mathematical test that is a necessary and sufficient condition for the stability of a linear time invariant control system [42]. This theory is an efficient recursive algorithm that developed and proposed by Edward John Routh [43] in 1876. By checking the sign of all the parameters listed in the Routh-Hurwitz table, the location of all the poles can be determined. The criteria act so efficient and effective that researchers nowadays still choose this method to check the stability of the system.

1) Stability prove for speed region 1:

Recall the feedback transfer function of speed region 1:

\[ G(S) = \frac{50.83333(Kp + S + Ki)}{S^2 + (50.83333Kp + 1.3583)S + 50.83333Ki} \]  \hspace{1cm} (D1)

The characteristic equation can be extracted from Equation (D.1) as:

\[ CE(S) = S^2 + (50.8333 * Kp + 1.3583)S + 50.83333Ki \]  \hspace{1cm} (D2)

Routh-Hurwitz criteria algorithm is shown as Table D1 in below.

<table>
<thead>
<tr>
<th>Row No.</th>
<th>Coefficient</th>
<th>Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( a_2=1 )</td>
<td>( a_0=50.8333 Ki )</td>
</tr>
<tr>
<td>2</td>
<td>( a_1=50.8333Kp+1.3583 )</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>( b_1 )</td>
<td>0</td>
</tr>
</tbody>
</table>

Using the equation:

\[ b_1 = \frac{a_1 \cdot a_0 - a_2 \cdot 0}{a_1} = 50.8333 Ki \] \hspace{1cm} (D3)

Based on the Routh-Hurwitz criteria [21], \( a_2, a_1 \) and \( b_1 \) should have same sign for stabilizing the system behavior. Since \( a_2 \) has the sign of ‘+’, \( Ki \) is required to be positive. For simplicity, \( Kp \) is chosen to be positive to meet the requirement.
2) Stability Prove for Speed Region 2

Reexamine the transfer function for speed region 2.

\[ G(S) = \frac{56.98 \cdot (Kp \cdot S + Ki)}{S^2 + (56.98 \cdot Kp + 0.4612)S + 56.98Ki} \] \hspace{1cm} (D4)

The characteristic equation of this plant can be shown as:

\[ CE(S) = S^2 + (56.98 \cdot Kp + 0.4612)S + 56.98Ki \] \hspace{1cm} (D5)

Routh-Hurwitz method is applied as shown in Table D2 below.

**Table D.2: Routh-Hurwitz Parameter for Region 2**

<table>
<thead>
<tr>
<th>Row No.</th>
<th>Coefficient</th>
<th>Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(a_2=1)</td>
<td>(a_0=56.98Ki)</td>
</tr>
<tr>
<td>2</td>
<td>(a_1=56.98Kp+0.4612)</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>(b_1)</td>
<td>0</td>
</tr>
</tbody>
</table>

The parameter \(b_1\) can be calculated using the following equation:

\[ b_1 = \frac{a_1 \cdot a_0 - a_2 \cdot 0}{a_1} = 56.98Ki \] \hspace{1cm} (D6)

Regarding Routh-Hurwitz criteria, \(a_2\), \(a_1\) and \(b_1\) should have same sign for stabilizing the system behavior. Since \(a_2=1\), \(Ki\) should be greater than zero to maintain the same sign as \(a_2\). For sake of simplicity, \(Kp\) is determined to be greater than zero to fulfill the stability requirement.
Appendix E: Bounded Error Prove

Rewrite the time derivative of the Lyapunov function established in Equation (4.2.20), after applying the projector designed in Equation (4.2.32) and (4.2.33), control law (4.2.14) and projector property shown in Equation (4.2.35) becomes:

$$\dot{V} \leq -(K_1 s(t)^2 + K_2 s(t) \cdot \tanh(s)) + \tilde{p}(t) \cdot s(t)$$  \hspace{1cm} (E.1)

In Equation (E.1), \(\tilde{p}(t)\) represents the overall system parameter and disturbance error regarding all the adaptation and estimation equations used in the control design process. Assume that the sliding variable \(s(t)\) starts from the outside boundary layer \(\epsilon\) at \(t = 0\), based on the property of tanh function, \(s(t)\) will be forced toward to zero. After certain amount of time, \(s(t)\) enters the boundary layer, \(|s(t)| \leq \epsilon\). At this moment the controller changed because of the tanh function and the derivative of the Lyapunov function can be developed as:

$$\dot{V} \leq -(K_1 s(t)^2 + K_2 s(t)^2) + \tilde{p}(t) \cdot s(t) = s(t)(\tilde{p}(t) - (K_1 + K_2)s(t))$$  \hspace{1cm} (E.2)

If \(s(t)\) stays inside a special region which satisfies \(|\tilde{p}(t)|/(K_1 + K_2) \leq s(t)\), then it follows that \(\dot{V} \leq 0\) and \(s(t)\) will be continue forced to move inside. If \(s(t)\) travels into a region where \(|\tilde{p}(t)|/(K_1 + K_2) \geq s(t)\), the derivative of Lyapunov function becomes \(\dot{V} \geq 0\) and \(s(t)\) will be forced to move outside of the region. Eventually, \(s(t)\) will be entrapped in the region side where \(|s(t)| \leq |\tilde{p}(t)|/(K_1 + K_2)\) after a finite time.

Recall the sliding function:

$$s = \dot{e} + \lambda \cdot e$$  \hspace{1cm} (E.3)

Combine the discussion above, the sliding function can be rewritten as:

$$s = \dot{e} + \lambda \cdot e \leq |\tilde{p}(t)|/(K_1 + K_2) \leq K_2/(K_1 + K_2)$$  \hspace{1cm} (E.4)

Due to the unmodeled disturbances and unknown perturbations, \(|\tilde{p}(t)|\) keeps changing with time. Under the assumption that the disturbances and perturbations is bounded in the actual system, the conclusion can be made that after finite time,
$|\hat{p}(t)|/(K_1 + K_2)$ can be treated as a constant defined as $K_2/(K_1 + K_2)$. By solving Equation E.4, the steady state error can be calculated as:

$$|e_{ss}(t)| \leq \frac{K_2}{\lambda(K_1 + K_2)} = \delta$$

(E.5)

The entail converse process can be shown in Figure E1.

Figure E.1: Steady-State Error
Reference


