Double closed-loop PI controller tuning for motor control

Shixuan Huang¹, Feng Tu¹, Qijun Guo², Binyu Wu², Zhifan Huang², Haigen Chen²

¹ State Grid Zhejiang Electric Power Co., Ltd. Jiaxing Power Supply Company, Jiaxing 314000, Zhejiang Province, China
² Yangtze Delta Region Institute Of Tsinghua University, Institute Of Information Technology, Jiaxing 314000, Zhejiang Province, China

*Corresponding Author: wuby@tsinghuait.org

Abstract. This paper demonstrates an algebraic algorithm performed in the control loop of a Brushless DC Motor (BLDCM) speed controller by computing the PI controller parameter that can be accepted. Using methodology of PI controller stability analysis, accurate stability conditions can be derived for the BLDC Motor speed controller. The results of applying the new method ensure the system stability and fast response by tuning the PI controller in the order of the motor and mechanical constant.

1. Introduction

"How do we tune a PI controller?" This is a very common question being asked. To justify the process is somewhat empirical, we typically use Bode plots or some simulation data, and it is very subjective to that specific kind of response that you are looking for [1][2][3]. By following some basic rules that will be explained later in this document, we should be able to tune a PI loop (regardless of whether they are speed loops or current loops) in a much more deterministic way. This analysis of loads has limitations to only real poles. If the load under consideration has prominent complex poles resulting from excessive torsional resonance between the motor and load, then the determinant factor of canceling the resonance effects is to make a more sophisticated controller rather than a simple PI structure. But in the majority of cases, the torsional resonance to the point where using a standard PI control structure can be empirically tamed by replacing a shaft coupler [4]. Also, the assumption of no viscous damping is made for loads and is valid in most design cases. However, if the tuning process presented in this paper does not work for a specific design, it likely has certain number of existing complex poles or viscous damping somewhere in the load which is affecting the results [5][6].

2. Background and Problem Formulation

The background, scope and objective of the problem is set up in this section.

The parallel path topology of a PI controller is shown in Figure 1. The error signal is split into two separate processing paths: one path can amplify and then integrate the error signal amplitude, and the other path will directly amplify the error signal. To drive the steady-state error of the system to zero an integrator is used, since if any non-zero steady-state error be input an integrator, the output of the integrator will be a boundless value. After doing a simple addition, two signal paths are combined at the output [1][5].
We can change the high frequency gain of the control loop by changing the value of $K_p$ term and change the low frequency gain of the control loop by changing $K_i$ term [5][6]. The inflection of the controller frequency plot corresponds the controller zero point.

The integrator is significantly vital in the operation of the PI controller; however, it also brings a set of challenges with its usage. For instance, it is assumed in an ideal condition that has no error in the control loop, which means the target signal of the PI controller needs to control and the commanded signal is equal. Now a small interference offset signal is set up for the controlled signal and observe the following phenomenon. The error signal is no longer zero due to the offset above. Then, the output of integrator will start growing in which it attempts to make the error signal to zero again. Now we remove the interference offset and observe. The controlled signal will eventually return to the command value again, but not immediately. The integrator output is quite large, causing the controlled signal to greatly exceed the command value when the integrator output is cleared. Up to now, the signal that need to be controlled is not fully controlled. It may severely damage your system without constraints. The profile of the "controlled" signal is not yet controlled at all. It may severely damage your system without constraints.

The PI controller form what we will use for outing analysis is a series topology that is commonly used as illustrated in Figure 2.

$$K_{p\text{series}} = K_p$$

$$K_{i\text{series}} = \frac{K_i}{K_p}$$

In this series topology structure, we can change the gain for all frequencies range by choosing different $K_{p\text{series}}$ value, and the zero point of the controller is directly defined via $K_{i\text{series}}$ (the unit is rad/sec). The level of difficulty to analyse both models are pretty much the same in terms of software complexity. However, in series topology, $K_{p\text{series}}$ and $K_{i\text{series}}$ are directly related to physical or electrical parameters of the visible system. Hence, the series form is preferred over the parallel form. The effect of $K_{p\text{series}}$ on the controller's performance is straightforward since it is only a proportional term in the open loop transfer function of the control system [6]. A PI controller contains two parameters that need to be adjusted to appropriate values. For a particular system, there is a
relationship between these two parameters. So there is a way to determine a parameter from the intrinsic parameters of the system, then adjust another parameter to meet the performance requirements of the system.

3. PI Design for Current Controllers

We presented two forms of the PI controller that are widely used today. The frequency responses of PI controller look identical regardless of which topology structure we use as shown in Figure 2. The gain of the PI controller has a significant effect on the stability of the system, as presented in graph. However, the use of PI controller in engineering has proved that the inflection point in the frequency response graph also plays a very important role in the system performance [1][3].

Its "s-domain" transfer function is defined based on the series form of the PI controller from the input error signal to the PI controller output as the following equation:

$$ PI(s) = K_p^{\text{series}} + K_i^{\text{series}} \left( 1 + \frac{s}{K_i^{\text{series}}} \right) $$

(3)

In accordance with the expression presented above, it is clearly showing the pole at \( s = 0 \) and the zero point at \( s = K_i^{\text{series}} \) rad/sec. A PI controller is dropped into a current controller, it monitors the current of the motor and produces an output to adjust this current as shown in Figure 3.

Figure 3. A Motor Current PI Controller

We use a simple series circuit which consists of a back-EMF voltage source, an inductor, and a resistor to make a first-order system; this system module is used to approximate the motor winding. By comparing the changing speed of the back-EMF with that of the motor current, the change of the back-EMF is relatively slow. We can derive a small-signal transfer function from the signal of motor voltage to the signal of motor current by assuming the back-EMF voltage is a constant, as follows:

$$ \frac{I(s)}{V(s)} = \frac{1}{R} / \left(1 + \frac{sL}{R}\right) $$

(4)

For a motor control loop, the system bus voltage and PWM gain parameters also need to be considered. In order to simplify the discussion, we equivalent consider that \( K_p^{\text{series}} \) term contains the above two parameters. We define the transfer function of the open loop control channel \( G_{\text{loop}}(s) \) as the product of the transfer function of motor RL equivalent circuit and the PI controller transfer function.

$$ G_{\text{loop}}(s) = \frac{I(s)}{V(s)} * PI(s) = \frac{K_p^{\text{series}} + K_i^{\text{series}} \left( 1 + \frac{s}{K_i^{\text{series}}} \right)}{s} * \frac{1}{1 + sL/R} $$

(5)

Assume that the gain of the system feedback channel is 1 (H(s) 1), so the closed-loop transfer function of the system is expressed as:

$$ G_{cl}(s) = \frac{G_{\text{loop}}(s)}{1 + G_{\text{loop}}(s)} = \frac{1 + s/K_i^{\text{series}}}{1 + s \left( \frac{1}{K_i^{\text{series}}} + \frac{R}{K_p^{\text{series}} * K_i^{\text{series}}} \right) + \frac{L}{K_p^{\text{series}} * K_i^{\text{series}} * s^2} } $$

(6)
Analyze the above closed-loop transfer function expression, whose denominator determines the number of poles of the system. This denominator is a second order polynomial of \(s\), so the system has two poles. We select \(K_{p}^{\text{series}}\) and \(K_{i}^{\text{series}}\) with caution; otherwise, it is easy to end up with complex poles. If these poles are close to the \(j\omega\) axis, it will easily lead to multiple resonant peaks in our control system. Therefore, the assumption of avoiding having complex poles becomes priority to select \(K_{p}^{\text{series}}\) and \(K_{i}^{\text{series}}\). By factoring the denominator, where \(C\) and \(D\) are real numbers, the following equation is obtained:

\[
1 + s \left( \frac{1}{K_{i}^{\text{series}}} + \frac{R}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} \right) + \frac{L}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} \cdot s^2 = (1 + A(s))(1 + B(s))
\]  \(7\)

By expanding the expression on the right side of the equation and then comparing it with the second-order expression on the left side of the equation, it can be found that if the system has real poles, the following two expression constraints must be satisfied:

\[
\frac{L}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} = A(s) \cdot B(s)
\]  \(8\)

\[
\frac{R}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} + \frac{1}{K_{i}^{\text{series}}} = A(s) + B(s)
\]  \(9\)

By analyzing expression 9, we can assume that the items on both sides of the equation correspond to each other, as an attempt to solve for terms \(A(s)\) and \(B(s)\), we get the following two expressions:

\[
\frac{R}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} = A(s)
\]  \(10\)

\[
\frac{1}{K_{i}^{\text{series}}} = B(s)
\]  \(11\)

Substitute these two expressions into expression 8, we get the constraint on \(K_{i}^{\text{series}}\) is:

\[
K_{i}^{\text{series}} = \frac{R}{L}
\]  \(12\)

And then we have the following simplified expression:

\[
G_{cl}(s) = \frac{1}{1 + s \cdot K_{i}^{\text{series}}} = \left(1 + \frac{R}{K_{p}^{\text{series}} \cdot K_{i}^{\text{series}}} \cdot s\right) \left(1 + \frac{L}{K_{p}^{\text{series}}} \cdot s\right) = \frac{1}{1 + \frac{L}{K_{p}^{\text{series}}} \cdot s}
\]  \(13\)

The above conversion removes one pole from the closed-loop gain expression at the cost of losing a zero. We are able to generate a closed-loop system response with one real pole and no zeros at all if selecting \(C\) and \(D\) properly. No peaky frequency responses or resonant conditions is shown here. That is just a simple single-pole low-pass response.

Frequency of value of zero occurs in the controller at \(K_{i}^{\text{series}}\). Therefore, to determine the response by using Equation 13, the controller zero frequency is set up to be equal to the pole of the plant.

Rewrite the closed-loop system response \(G(s)\), substitute all expressions discussed above up to now, we obtain:

\[
G_{cl}(s) = \frac{1}{1 + \frac{L}{K_{p}^{\text{series}}} \cdot s} \Rightarrow K_{p}^{\text{series}} = L \cdot \text{Bandwidth}
\]  \(14\)

4. PI Design for Speed Controllers

In fact, it is more complicated to close a speed loop than a current loop. Also, more system parameters beyond the current loop are required to better design the speed loop. Then, a cascaded speed control loop is decided to be put into use. With the term "cascaded," it refers to a control system containing an
outer loop with one or more inner loops. Again, to be clear, we are talking about an ideal situation where the inertia of the system is concentrated and unitary, assuming a rigid connection between the motor and the load and no viscous resistance [1][4].

Before going any further, let's define a list of parameters related to the motor structure. The moment of inertia of the motor and load is "J"; The number of magnetic poles in the motor rotor is "P"; The back electromotive force constant (KE) of the motor coil is "λ"; The inductance of the motor coil is "L"; Laplace frequency variable is "S". The proportional and integral coefficients of the speed loop PI controller are $\text{spdK}_{p,\text{series}}$ and $\text{spdK}_{i,\text{series}}$, respectively. The scale factor of the internal current loop PI controller is $K_{p,\text{series}}$.

We connect a speed PI controller in series before the current loop PI controller of the motor. The Error signal of current loop PI controller is the open loop output of the speed PI controller. Then the open loop transfer function of the two cascade controllers is:

$$GH(s) = \frac{3P\lambda \cdot \text{spdK}_{p,\text{series}} \cdot \text{spdK}_{i,\text{series}} (1 + \frac{s}{\text{spdK}_{i,\text{series}}})}{S^2 \left(1 + \frac{L}{K_{p,\text{series}} S}\right) \cdot 4J} \quad (15)$$

Once the parameters of inner loop current controller is defined, the trade off between speed loop stability and bandwidth is quantified and visualized with proper damping factor ($\delta$). Using the same mathematical transformation as the simplified current loop PI controller, we can get the parameters of the speed loop PI control as follows:

$$\text{spdK}_{i,\text{series}} = \frac{K_{p,\text{series}}}{\delta^2 \cdot L} \quad (16)$$

$$\text{spdK}_{p,\text{series}} = \frac{4\delta \cdot \text{spdK}_{i,\text{series}}}{3P\lambda} \quad (17)$$

We choose An Anaheim Automation 24V permanent magnet synchronous motor to verify the designed controller. The characteristics of this motor are as follows:

- The motor's equivalent series resistance "$R_s$" and equivalent series inductance "$L_s$" are 0.5 ohms and 0.7 mH, respectively.
- The Back-EMF of motor is 0.0054 v-sec/radians.
- The inertia of motor is 1.9E-3 Kg * m², the number of poles in the rotor is 8.
- The bandwidth of desired speed = 795 rad/sec, and we would like a mechanical damping factor ($\delta$) of 5.

The above system parameters are used to implement the cascade PI controller designed by us, and a step input signal (corresponding to the desired target velocity) is applied to the control system. The simulation response of the system to the step input is shown in Figure 4. For easy viewing, the timeline has been scaled appropriately in this design.
5. Conclusions

In summary, there are couple of simple techniques you may take advantage of to design the current loop of your PI controller.

The zero of the PI controllers is set up to be $K_i^{\text{series}}$. When the transfer function of a motor control system needs to be reduced from second order to first order by mathematical transformation, $K_i^{\text{series}}$ can be set as the value of one of the poles. After $K_i^{\text{series}}$ has been set to the value of the real pole, it cancels out pole/zeros, and generates a closed-loop response with only one single real pole. In other words, there will not be any resonant peaking in the closed-loop system which implies the system is reasonably stable.

The bandwidth of the closed-loop system response is set up to be $K_p^{\text{series}}$. As visualized by Equation 12, the higher the value of $K_p^{\text{series}}$ is, the higher the current loop bandwidth it will be. This relationship can be observed when $K_p^{\text{series}}$ is equal to the inductive impedance for any bandwidth frequency.

The benefit of this approach is straightforward that only requires two meaningful system parameters: the bandwidth of the current controller and the damping coefficient of the speed loop. Then, it is no longer necessary to tune four PI coefficients empirically, instead, since it has seemingly tiny correlation to the system performance. Once it satisfies this condition, the four PI coefficients can be calculated automatically. The bandwidth of current controller is certainly an important system parameter; however, in speed-controlled systems, it is more important for the bandwidth of the speed controller to be specified first, and then followed with the bandwidth of current controller based on that.

Acknowledgments

State Grid Zhejiang Electric Power Company Science and Technology Project: Research on key technologies of spacer installation and maintenance robots (5211JX1900CU)

References

[1] Ogata, K., Yang, Y. (1970) Modern Control Engineering. Publishing House of Electronics Industry(PHEI), Beijing.
[2] Babuska, R., Stramigioli, S. (1999) Matlab and Simulink for Modeling and Control. Delft University of Technology, Delft, the Netherlands.
[3] Johnson, M.E. (2002) Model Checking Safety Properties of Servo-loop Control Systems. In: Dependable Systems and Networks. New York. : 45–50.
[4] Nise, N.S. (2007) Control Systems Engineering. John Wiley & Sons(JW.A), Hoboken, New Jersey, USA.

[5] Ghosh, S. (2010) Control Systems, vol. 1000. Pearson Education, London.

[6] Rashid, A. (2017) Formal Analysis of Linear Control Systems using Theorem Proving. http://save.seecs.nust.edu.pk/projects/falcstp.