Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2013) 21: 1318 – 1327

doi:10.3906/elk-1112-25

Antiwindup design for the speed loop PI controller of a PMSM servo system

Ming YANG,* Li NIU, Dianguo XU
Department of Electrical Engineering, Harbin Institute of Technology, Harbin, China

Received: 07.12.2011 • **Accepted:** 16.03.2012 • **Published Online:** 12.08.2013 • **Printed:** 06.09.2013

Abstract: A novel antiwindup (AW) strategy for a proportional-integral (PI) regulator based on the back-calculation method is presented. The proposed method will calculate the gain of the back-calculation based on the input and output status of the PI regulator without the requirement of motor parameters, such as the inertial and torque-current coefficient, and the back-calculation gain will vary with the speed command in order to obtain the optimal dynamic performance. The proposed method avoids the problem caused by the fixed back-calculation gain in the classical AW strategy, and prevents the system from untimely saturation withdrawing or an unexpected overshoot due to the inappropriate back-calculation gain, respectively. The proposed AW strategy is compared with the classical AW design by simulation and experiment on an application platform of a permanent magnet synchronous motor servo system. A PI controller with the proposed AW design will decrease the overshoot with a small settling time of the system's step-response, which can be observed obviously in wide dynamic variation. It reduces the difficulty of the parameter tuning for the PI controller.

Key words: PMSM, antiwindup, integration saturation, PI controller

1. Introduction

The proportional-integral (PI) control scheme has been widely used in servo motor drive systems. There are several restrictions in the cascade control form; the current command as the inner loop reference must be limited by the maximum load current and the overheating of the servo motor. Therefore, saturation nonlinearity exists in the speed control loop and it leads to an output of the PI controller that is not equal to the input of the plant; hence, the closed-loop performance will significantly deteriorate in terms of the expected linear performance. This phenomenon of deteriorated performance is referred to as the windup phenomenon [1], which causes a large overshoot, slow setting time, and sometimes even instability in the speed response.

Many scholars have done a lot of research on windup and have proposed many schemes for antiwindup (AW) [2–9]. Generally, the AW technique can be classified into 2 categories. One is conditional integration. The control system has a switch that will stop or restrict the integration when the control input is saturated. This method is simple to implement, but its disadvantage is that it is hard to determine the switch moment and its lack of robustness. If the switching moment is chosen improperly, not only can it not improve the integral saturation, but it can even cause system divergence [10–12]. The other AW category is based on the idea of back-calculation, making the difference between the output of the controller and the limited input of the plant as the negative feedback signal to act on the integrator input. The transient performance, such as the overshoot, depends heavily on the feedback gain rather than the PI gains. The advantage of the back-calculation category is that the design of the compensator is easier and the compensation structure has a higher robustness [1,2]. The

*Correspondence: yangming@hit.edu.cn

disadvantage is that it is difficult to synthesize the design of the system. Hence, back-calculation compensation has been widely adopted [13–16]. Both conditional integration and tracking back-calculation were combined in [17–21]. The goal of AW is actually to get the results as close to an unconstrained closed-loop system as possible.

The conventional design of back-calculation is based on the PI controller and the system parameters, including the system's inertia, load torque, etc. (the ideal parameter of the feedback design can be achieved if the system inertia and load torque are attained). However, practical experiments show that even if the system parameters are constant, the fixed ideal feedback coefficient does not always improve the system's performance when different speed step responses are used. Thus, this leads to complicated controller parameter tuning.

The authors in [22] compared 4 types of experimental results based on the back-calculation AW, and they concluded that the static and low-order compensation are superior to both the IMC and full order compensation, but other contrasts are not obvious. The authors do not distinguish the differences between linear and nonlinear of severe saturation. Shin [18] deemed that the transient performances of the error, such as the overshoot and settling time, are mainly affected by the steady-state value of the integral, and the steady-state value of the integral is controlled by both the load and speed references. If the speed controller is saturated, the controller will calculate the steady-state value of the integral, and while the controller comes into linear range again, the calculated value is set as the initial value of the integral. This method will decrease the influence of the load and speed reference variations to the system's transient performance efficiently, and will enhance the adaptability of the AW. However, the range of the integral steady-state value should be set offline, and the measurement error will obviously influence the calculated integral value. On the other hand, the prediction value of the integral steady-state is determined by the motor parameters of the inertia, friction coefficient, and torque-current coefficient, but these parameters will vary with the application environment, particularly the system's inertia, which means that the motor parameters should be updated in a different application environment.

This paper presents a new piecewise back-calculate AW based on the traditional back-calculation and tracking (BCAT) AW design, which can automatically configure the feedback gain according to the system states. There is no need for extra calculations and hardware. This new AW design was proven to have more adaptability by its simulation and experiment on a permanent magnet synchronous motor (PMSM) servo system.

2. Traditional AW design

2.1. Principle of the BCAT AW

Every slow time varying link may lead to windup [10]. Windup is usually caused by the integrator of the PI controller. It usually needs to constrain the integral behavior in order to eliminate the windup problem of the controllers.

The traditional structure of BCAT AW design is shown in Figure 1, letting $(u_n - u_s)$ be the negative feedback input of the integrator. The output of the PI controller is described by Eq. (1).

The choice of τ_c is important for the result of the AW. It ranges from τ_i to $8\tau_i$, according to the experiences.

$$u_n = \frac{K_p\tau_c(\tau_i s+1)}{\tau_i(\tau_c s+1)}e + \frac{1}{\tau_c s+1}u_s \tag{1}$$

Assuming that $u_s \in [-0.1, 0.1]$ and $K_p = 1$ for the system shown in Figure 1, the simulation based on the PMSM servo system with the BCAT AW design is done. With the step speed command of $r : 0 \to 1$, the

simulation results of the system with different are shown in Figure 2, where (a) is the speed responses and (b) is q-axis current with different τ_c.

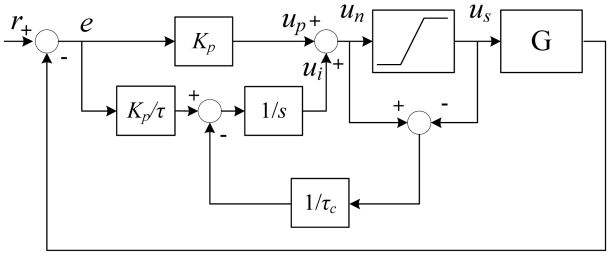

Figure 1. Classical BCAT AW structure.

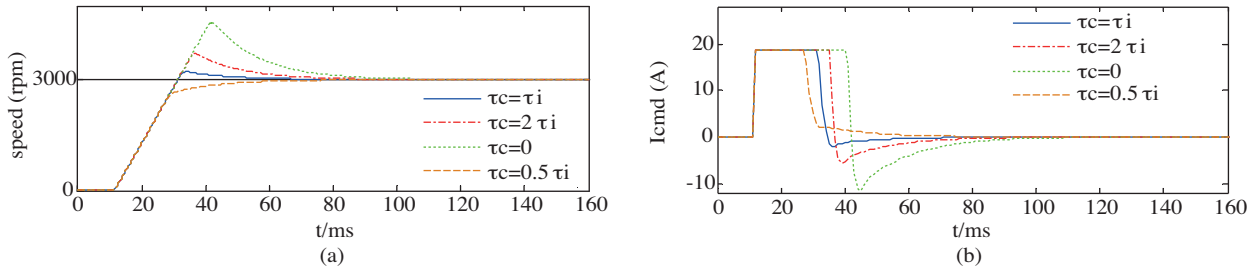

Figure 2. Simulation of the classical BCAT AW, where (a) is the speed responses and (b) is the q-axis current with different τ_c.

The simulation results have shown that the system can show an acceptable performance if the gain of the feedback branch is calculated appropriately.

2.2. Application of the BCAT AW

As mentioned earlier, the negative feedback portion is added into the PI controllers with a limitation in order to eliminate the windup phenomenon, which is caused by the integral behavior in dynamic processes. The PI controller of the PMSM servo system usually has large proportional gain and integral coefficients, in order to get a high stiffness of the system and ensure that the motor has a quick response. The output of the controller may saturate immediately when a large step change is given, and the negative feedback portion will work before there is integral saturation in the controller.

For the system in Figure 1, the trend of the error integration u_i of the saturated system depends on $K_c(K_c = 1/\tau_c)$. The system response will have a large overshoot if u_i continues increasing, shown as the dash-dot-line ($\tau_c = 2\tau_i$) in Figure 2a. To avoid the overshoot being too large, there is a method that will increase K_c as well as let $\tau_c < \tau_i$, and the response is shown as the dotted line in Figure 2b.

The traditional BCAT AW design has an obvious disadvantage, where K_c is uncertain and it needs to be preset. There is no guarantee that the system will have a good speed response in all of the operational situations, even though K_c is configured appropriately for a certain step speed change.

Figure 3 shows the simulation results of the PMSM servo system, where (a), (b), and (c) are against the speed commands of 0.2, 0.6, and 1 PU, respectively, and it gives different step commands with constant inertia and control parameters (K_p, K_i, K_c), whose AW compensator adopts the form of the BCAT in PI controller.

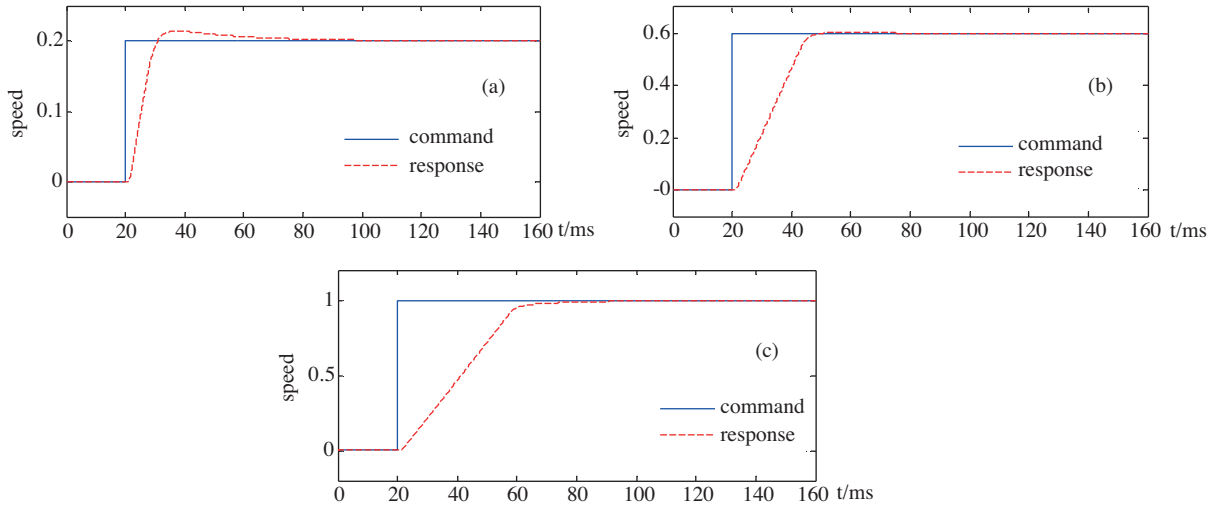

Figure 3. Simulated waveforms of the BCAT AW, where (a), (b), and (c) are the speed command and relative response against the 0.2, 0.6, and 1 PU speed references, respectively.

There is a great difference among the system's responses with different speed step commands (r = 0.2, 0.6, or 1), which can be seen in Figure 3. The overshoot is quite high when the step command is small, so that the AW compensation is not available, as shown in Figure 3a. The system can show an ideal response due to the balance between the integration and the AW compensation, when the step command is moderate as shown in Figure 3b. As the negative feedback is more effective and has a longer duration when a larger step command is given, the system may be brought out of saturation ahead of time, as shown in Figure 3c, and the transient process may be longer. Figure 4 shows the experimental BCAT AW performance on the speed references of the PMSM servo system, where the experimental conditions are the same as the simulation test in Figure 3.

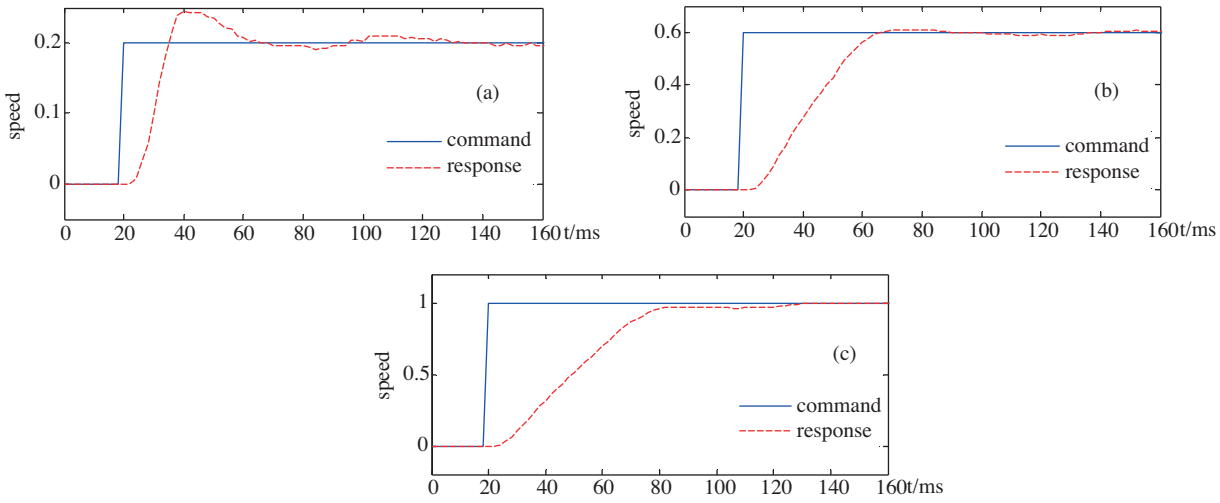

Figure 4. Experimental waveforms of the BCAT AW, where (a), (b), and (c) are the speed command and relative response against the 0.2, 0.6, and 1 PU speed references, respectively.

For the system with the BCAT AW, the simulation and experimental results demonstrate that the speed responses of the motor will vary a lot when different speed commands are preset, even though the inertia and the

control parameters are the same. The depth and duration of the controllers' saturation may change a lot in the dynamic operation, when the commands are different. It leads to the discrepant effect of the AW compensation, as the different performances of the speed responses shown in Figures 3 and 4.

3. Piecewise back-calculate AW

In this paper, we present a better AW approach based on the traditional BCAT AW design, which is called piecewise back-calculate AW. It can be seen from the above analysis that the constant K_c is not suitable for all conditions. The AW compensator proposed in this paper can configure K_c as well as the gain of the feedback portion automatically on the basis of the system's state. This AW design can ensure the rapidity of the system's response, without a large overshoot or the untimely exiting of the saturation state.

For the system in Figure 1, supposing that $u_s \in$ [–0.1,0.1] and $K_p = 1$, the controller will saturate immediately, as well as $u_n > u_s > 0$, when a step command $r : 0 \rightarrow 1$ is given. Letting $e_u = u_n - u_s$, the output of the integrator (u_i) will remain invariable if K_c is in the type in Eq. (2).

$$K_c = \frac{eK_p}{u_n - u_s} K_i \tag{2}$$

If $u_i(0) = 0$, from Eq. (2) it can be seen that $u_i(t) = 0$. In this case, we have:

$$u_n = u_p = eK_p, \tag{3}$$

and we can then deduce:

$$K_c = \bar{\Theta} K_i, \tag{4}$$

where

$$\bar{\Theta} = \frac{u_n}{u_n - u_s} = 1 + \frac{0.1}{u_n - 0.1}. \tag{5}$$

Function $\bar{\Theta}$ is described as the dotted line in Figure 5, where $u_n > 0.1$. The system is in linear control when $u_n \leq 0.1$.

If K_c/K_i is calculated according to $\bar{\Theta}$, the integrator may stop when the controller is saturated. This is equivalent to the integral constraint AW.

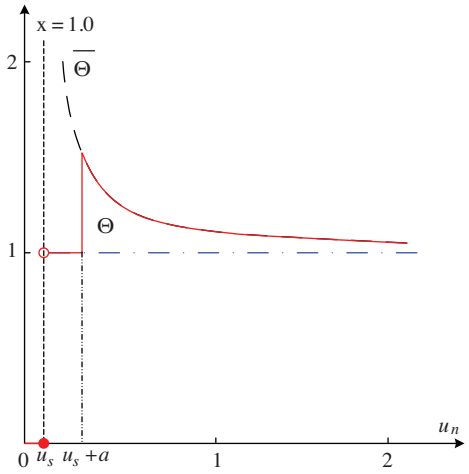

Figure 5. Gain function of the feedback branch.

Excessive inhibition of the integration will deteriorate the performance of the speed response, especially when the load is high, as shown in Figure 6, where (a) is the overall speed response and (b) is the final stage of the speed transient. The integration must be held up to a proper level so that the controller will get out of saturation at the right time and the motor will get peak acceleration in the dynamic process.

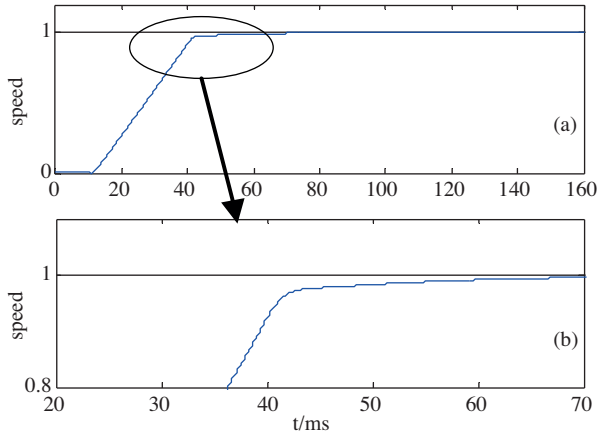

Figure 6. System response with the integral limited, where (a) is the overall speed response and (b) is the final stage of the speed transient.

Now, set a as the threshold value, indicating the level of saturation in the controller. The controller will calculate K_c as in Eq. (2) to suppress the integration, because it is highly saturated when $e_u = u_n - u_s \geq a$. Letting $K_c = K_i$, when $0 < e_u = u_n - u_s < a$, and the AW process is the same as the BCAT at this time. The tuning principle of K_c is shown in Eq. (6):

$$K_c = \Theta K_i, \tag{6}$$

where

$$\Theta = \begin{cases} 1 & u_s < u_n < u_s + a \\ \frac{eK_p}{u_n - u_s} & u_n \geq u_s + a \end{cases}. \tag{7}$$

The error e and the output of the controller u_n are the variables of Θ, as shown in Eq. (7). The structure of the piecewise back-calculate AW design is shown in Figure 7.

Defining a as the threshold constant, will ensure that the overshoot of the system's response without any load is about 5%. The overshoot can be absorbed when the system has load, so that a good response can be obtained [11]. K_c can be tuned automatically online on the basis of the system's state without any management.

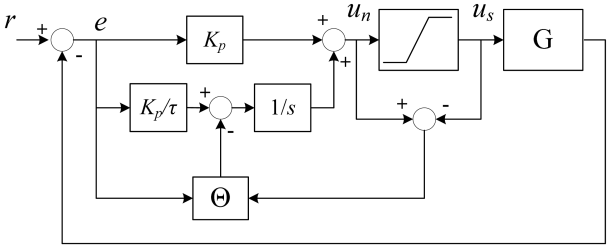

Figure 7. Piecewise back-calculate AW structure.

4. Simulation and experiment

The specifications of the PMSM servo system used in both the simulation and experiment are listed in Table, and the experimental platform is based on the TMS320F2812 digital signal processor (DSP). For the sake of the loading test, another 750 W servo pack by Yaskawa is involved in the coupling with the prototype motor, and the current control frequency is 10k Hz in both the simulation and experimental conditions. The data from the experiment waveforms are collected by the DSP and the waveforms are plotted using MATLAB in this paper. The photograph of the experimental platform is shown as Figure 8.

Table. Specification of the PMSM Servo motor.

Parameter	Value
Rated power (kW)	0.75
Rated torque (N·m)	2.4
Rated speed (r/min)	3000
Rotor inertia (N·m^2)	8.53e^{-5}
Stator resistance (Ω)	0.45
Stator inductance (mH)	3.9
Maximum current (A)	18.6

The simulation waveform of the control systems with the above 2 different AW designs is shown in Figure 9, where the performance of the 2 systems with no load are compared in response to speed references: Figure 9a, $r : 0 \rightarrow 0.2$ and Figure 9b, $r : 0 \rightarrow 1$.

The 2 AW methods are equivalent because the output of the controller is less than the threshold value a in the dynamic process when the step change is small. This leads to the same responses as shown in Figure 9a. Controllers with the proposed piecewise back-calculate AW design can resolve the problem of the overshoot in the speed response effectively when a large step speed reference is given, and can avoid untimely saturation withdrawing, such as that in the BCAT AW design caused by the improper feedback gain, which can be seen from the simulation results in Figure 9b.

Figure 10 shows the simulated comparison of the responses for the load and no load conditions for the control system with piecewise back-calculate AW. The slight overshoot decreases obviously if the control system has a rated load and there is no performance degradation due to the withdrawing saturation being earlier than expected.

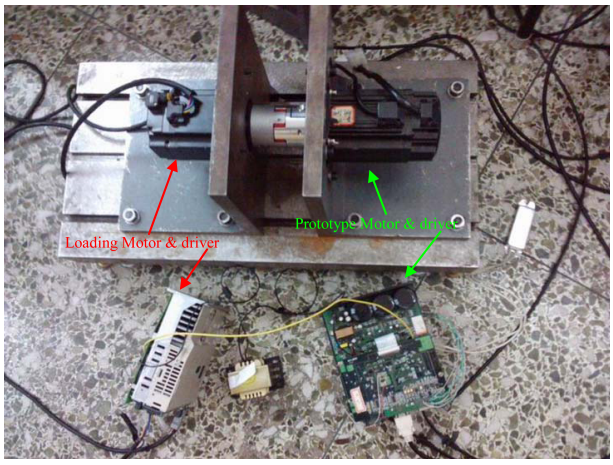

Figure 8. Photograph of the experimental platform.

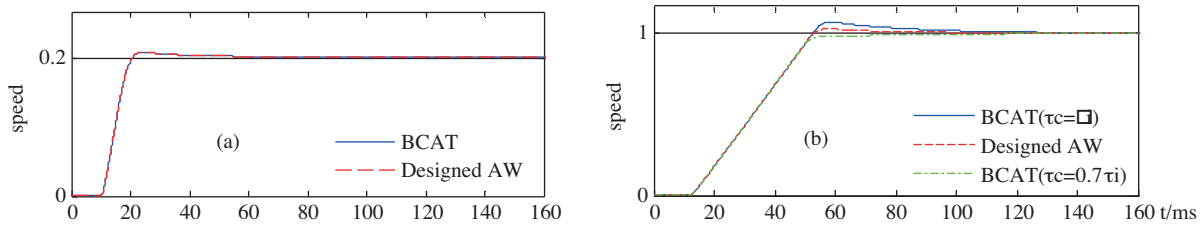

Figure 9. Simulated performance of the different AW systems, where (a) is with a 0.2 PU speed command and (b) is with a 1 PU speed command.

The experiment is carried out in the same way as what has been done in the simulation, and the experimental results coincide with simulation results as shown in Figures 11 and 12.

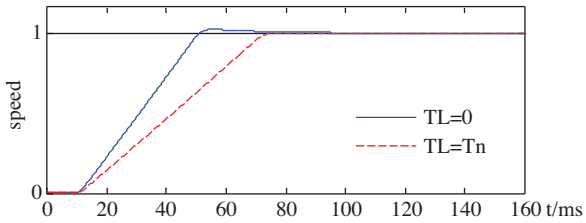

Figure 10. Simulation of the piecewise back-calculate AW structure under different load conditions.

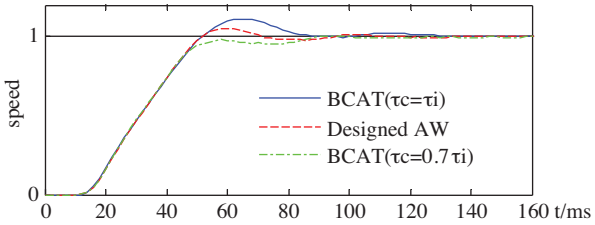

Figure 11. Experimental performance of the different AW systems.

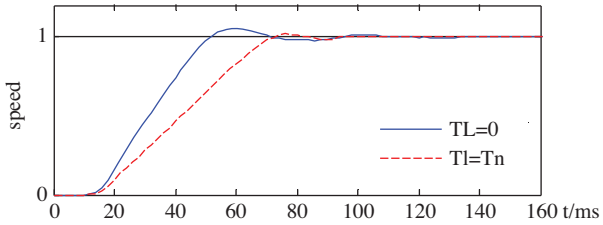

Figure 12. Experiment of the piecewise back-calculate AW structure under different load conditions.

5. Conclusions

This paper presents the piecewise back-calculate AW strategy based on the traditional back-calculation AW method. The proposed method for the PMSM servo system can ensure the rapidity of the motor's speed response, and avoid a large overshoot or untimely saturation withdrawing. The proposed method does not need the motor parameters of the inertia, friction coefficient, and torque-current coefficient, because it allows the application of the proposed method without the consideration of the motor parameters. Based on the simulation and experimental results, the proposed method resolves the problem where the parameter of the PI controller in the traditional AW strategy is difficult to configure, and the controllers with the proposed AW strategy will increase the dynamic performance of the control system.

Acknowledgment

This work has been supported by the National Natural Science Foundation of China (51007012).

References

[1] K. Astrom, T. Hagglund, PID Controllers: Theory, Design, and Tuning. Research, Triangle Park, North Carolina, The Instrumentation, Systems, and Automation Society; 2 Sub edition, 1995.

[2] P. Youbin, D. Vrancic, R. Hanus, "Anti-windup, bumpless, and conditioned transfer techniques for PID controllers", IEEE Control Systems Magazine, Vol. 16, pp. 48–57, 1996.

[3] D. Zhang, H. Li, E.G. Collins, "Digital anti-windup PI controller for variable-speed motor drives using FPGA and stochastic theory", IEEE Transactions on Power Electronics, Vol. 21, pp. 1496–1501, 2006.

[4] G. Herrmann, M.C. Turner, I. Postlethwaite, G. Guoxiao, "Practical implementation of a novel anti-windup scheme in a HDD-dual-stage servo system", IEEE/ASM Transactions on Mechatronics, Vol. 9, pp. 580–592, 2004.

[5] S. Tarbouriech, M. Turner, "Anti-windup design: an overview of some recent advances and open problems", IET Control Theory and Applications, Vol. 3, pp. 1–19, 2009.

[6] B. Bahrani, S. Kenzelmann, A. Rufer, "Multivariable-PI-based *dq* current control of voltage source converters with superior axes decoupling capability", IEEE Transactions on Industrial Electronics, Vol. 58, pp. 3016–3026, 2011.

[7] R.J. Wai, J.D. Lee, K.L. Chuang, "Real-time PID control strategy for Maglev transportation system via particle swarm optimization", IEEE Transactions on Industrial Electronics, Vol. 58, pp. 629–646, 2011.

[8] A. Draou, A. Miloudi, E.A. Al-Radadi, "A variable gain PI controller used for speed control of a direct torque neuro fuzzy controlled induction machine drive", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 15, pp. 37–49, 2007.

[9] A.F. Bakan, S. Partal, İ. Şenol, K.N. Bekiroğlu, "Online speed control of a brushless AC servomotor based on artificial neural networks", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 19, pp. 373–383, 2011.

[10] J.W. Choi, S.C. Lee, "Antiwindup strategy for PI-type speed controller", IEEE Transactions on Industrial Electronics, Vol. 56, pp. 2039–2046, 2009.

[11] J.K. Seok, K.T. Kim, D.C. Lee, "Automatic mode switching of P/PI speed control for industry servo drives using online spectrum analysis of torque command", IEEE Transactions on Industrial Electronics, Vol. 54, pp. 2642–2647, 2007.

[12] J.K. Seok, "Frequency-spectrum-based antiwindup compensator for PI controlled systems", IEEE Transactions on Industrial Electronics, Vol. 53, pp. 1781–1790, 2006.

[13] A. Syaichu-Rohman, R.H. Middleton, "Anti-windup schemes for discrete time systems: an LMI-based design", Proceedings of the 5th Asian Control Conference, pp. 554–561, 2004.

[14] S. Lambeck, O. Sawodnt, "Design of anti-windup-extensions for digital control loops", Proceedings of the American Control Conference, pp. 5309–5314, 2004.

[15] F. Cupertino, E. Mininno, D. Naso, B. Turchiano, L. Salvatore, "On-line genetic design of anti-windup unstructured controllers for electric drives with variable load", IEEE Transactions on Evolutionary Computation, Vol. 8, pp. 347–364, 2004.

[16] P. March, M.C. Turner, "Anti-windup compensator designs for nonsalient permanent-magnet synchronous motor speed regulators", IEEE Transactions on Industry Applications, Vol. 45, pp. 1598–1608, 2009.

[17] A.S. Hodel, C.E. Hall, "Variable-structure PID control to prevent integrator windup", IEEE Transactions on Industrial Electronics, Vol. 48, pp. 442–451, 2001.

[18] H.B. Shin, "New antiwindup PI controller for variable-speed motor drives", IEEE Transactions on Industrial Electronics, Vol. 45, pp. 445–450, 1998.

[19] J.G. Park, J.H. Chung, H.B. Shin, "Anti-windup integral-proportional controller for variable-speed motor drives", Journal of Power Electronics, Vol. 2, pp. 130–138, 2002.

[20] M. Yang, D. Xu, X. Gui, "Study of AC PMSM speed servo system anti-windup design", Proceedings of the CSEE, Vol. 27, pp. 28–32, 2007.

[21] A.T.A. Sevinç, "A full adaptive observer for DC servo motors", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 11, pp. 117–130, 2003.

[22] H.B. Shin, J.G. Park, "Anti-windup PID controller with integral state predictor for variable-speed motor drives", IEEE Transactions on Industrial Electronics, Vol. 59, pp. 1509–1516, 2012.