Letters

Fast Current Control Without Computational Delay by Minimizing Update Latency

Mingjin Hu, *Student Member, IEEE*, Wei Hua, *Senior Member, IEEE*, Huafeng Xiao, *Senior Member, IEEE*, Zheng Wang, *Senior Member, IEEE*, Kai Liu, Kun Cai, and Yi Wang

***Abstract*—This letter proposes a fast current control scheme for ac drives by minimizing the latency from current sampling to the updation of a pulsewidth-modulation (PWM) command. By reconstructing the architecture of a linear controller, the minimal necessary steps to obtain the output voltage are deduced as the *primary calculation*, and the other computation work of the current loop is done in the *postcalculation* after updating the PWM command. With the proposed method, the computational delay is reduced with negligible loss in the available modulation index, and a higher stability margin or system bandwidth is realized. Also, the developed controller structure exhibits inherited antiwindup characteristics. The technique has been validated by experimental investigations on a permanent magnet synchronous motor.**

***Index Terms*—Antiwindup, computational delay, current control, stability margin.**

I. Introduction

HIGH-PERFORMANCE ac drives require efficient inner-current loops. Important contributions to the theory and practice have been provided in [1]–[3] on current controller design and parameter tuning. In terms of the dynamic control of current loop, the modulation delay and the computational delay limit the bandwidth and robustness of the control system because of the introduced phase lag [1]. Generally, the typical value of the computational delay is one sampling period T_s due to the nonzero calculation time of microcontrollers and the synchronization of current sampling and updation of pulsewidth modulation (PWM) command to the carrier wave [1]–[5]. The volt–second contribution of the switching process results in an averaged modulation delay of 0.5 T_s. A total digital delay of 1.5 T_s typically exists in electric drives.

Manuscript received March 31, 2021; revised April 22, 2021 and May 12, 2021; accepted May 18, 2021. Date of publication May 20, 2021; date of current version July 30, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 51825701 and in part by the Scientific Research Foundation of Graduate School of Southeast University. *(Corresponding author: Wei Hua.)*

The authors are with the School of Electrical Engineering, Southeast University, Nanjing, 210096, China (e-mail: 1205195957@qq.com; huawei1978@seu.edu.cn; xiaohf@seu.edu.cn; zwang@eee.hku.hk; kliu@seu.edu.cn; seu_caik@163.com; 1074069755@qq.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2021.3082399.

Digital Object Identifier 10.1109/TPEL.2021.3082399

Increasing the sampling frequency can directly reduce the digital delay and, thus, improve the current control performance. In PWM-controlled drives, a basic approach is using the dual-sampling-and-updating scheme, so that twice the switching frequency of the sampling frequency is achieved. The wide bandgap semiconductors allow a higher switching frequency, and thus, the digital delay can be reduced [6]. The sampling frequency can also be higher and independent of the switching frequency if the none-PWM scheme is used, e.g., finite control-set model predictive control [7]. However, the heavy computational burden may restrict its applications.

Alternatively, the digital delay can be decreased by eliminating the computational delay, which requires updating the PWM command immediately after sampling the currents [8], [9]. The timing sequence is shown in Fig. 1, where the current is sampled at the carrier valley and peak. Since the switches should not act during the control loop calculation, the maximum modulation index is limited depending on the update latency. Strong computational power is required to minimize T_{update} and most servo applications utilize the field-programmable gate array (FPGA). Yang *et al.* [10] present a digital signal processor (DSP)-based scheme for *LCL*-type grid-tied inverter with full output voltage range. Improved robustness is realized; however, this method is restricted in the single-phase system. Optionally, the sampling instants can shift forward to allow a full modulation index, but it suffers from aliasing and switching noise distortion [9]. Nevertheless, minimum update latency is desired for both options to expand the output voltage range or reduce the sampled harmonic currents.

Reduced update latency can also be achieved by optimizing the current control schedule. In patent [11], a faster current feedback path in the control algorithm is applied for quick current response; thus, the passivity of the power converter is improved due to the reduced digital delay; other parts of the current controller including integral and resonance control are done in other slower tasks.

This letter develops a fast current control scheme to minimize the update latency for three-phase ac drives where the maximum modulation index is not crucial. The proposed scheme comprises a primary calculation that utilizes minimal steps to obtain and update the PWM command, and a postcalculation for other computational burdens of the current loop, which has similarity

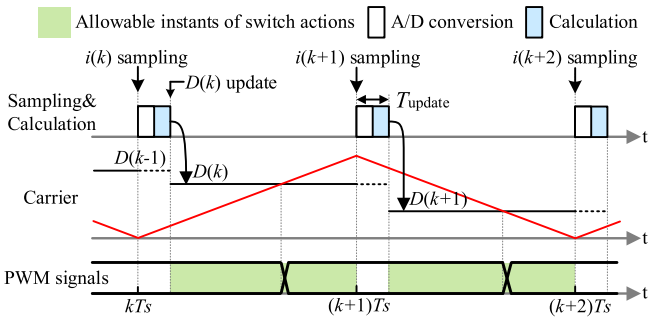

Fig. 1. PWM command $D(k)$ is updated in the present control period with the latency T_{update} to the current sampling, and it remains active until the next update event, i.e., from kT_s+T_{update} to $(k+1)T_s+T_{\text{update}}$. To ensure a definite switching behavior, $D(k)$ should be limited so that it will not intersect with the carrier wave during the analog-to-digital conversion and calculation.

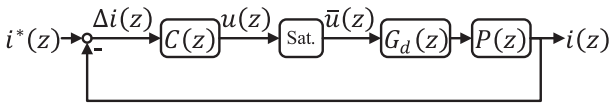

Fig. 2. Block diagram of conventional current control.

with the work in [11]; however, it provides an explicit solution for arbitrary linear controllers to reduce the update latency by modifying the controller architecture. Especially, the update latency is identical for either a simple PI controller or other complex controllers. With the reduced computational delay, higher stability margin and system bandwidth can be further realized.

II. Performance Degradation With the Computational Delay

Fig. 2 demonstrates the conventional current control, where i^* and i are the reference and feedback currents, respectively, u and $\bar{u}$ are the unconstrained and constrained voltage command, respectively, $C(z)$ is the current controller, $G_d(z)$ is the computational delay, and $P(z)$ is the plant model. "Sat." is a saturation block treated as a unit gain in linear analysis. In the synchronous reference frame (*dq*-frame), a typical plant with inductive–resistive behavior in the complex transfer function [3] is given by

$$P\ (z) = \frac{1}{R_s}\ \frac{1-\rho}{e^{j\omega_e T_s} z-\rho} \tag{1}$$

where $\rho\ = e^{-R_s/L_s \cdot T_s}$, with R_s and L_s being the resistance and inductance, respectively, T_s is the sampling period, ω_e is the electrical angular speed and the modulation delay is included in the model by the zero-order hold method.

The one-step computational delay is modeled as $G_d\ (z) = e^{-j\omega_e T_s}\ z^{-1}$ in the *dq*-frame, since the rotational transformation rises extra phase lag. Considering a discrete-time complex PI controller with pole-zero cancellation [3]

$$C\ (z) = \ Kc\frac{\hat{R}_s}{1-\hat{\rho}}\ \frac{e^{j\omega_e T_s} z-\hat{\rho}}{z-1} \tag{2}$$

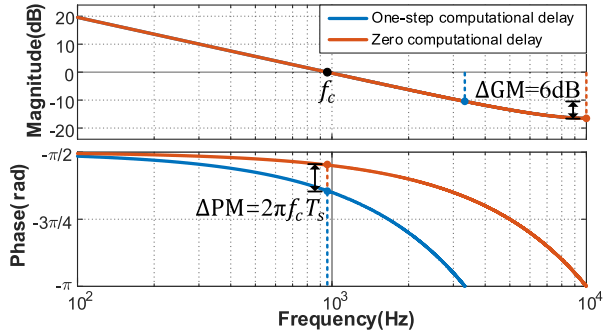

Fig. 3. Frequency responses and margins of the open-loop transfer functions $G_{\text{loop}}(z)$. (Conditions: $K\ =\ 0.3,\ \ T_s\ =\ 50\ \mu$s).

where $\hat{\rho} = e^{-\hat{R}_s/\hat{L}_s \cdot T_s}$, $\hat{R}_s$ and $\hat{L}_s$ are the nominal values of R_s and L_s, respectively, and $c\ = e^{j\omega_e T_s}$ compensates for the phase lag introduced from $G_d(z)$. If there is no computational delay ($G_d\ (z) = 1$), $c = 1$ should be used in (2). For the two cases with and without the computational delay, their open-loop transfer functions $G_{\text{loop}}\ (z) = G_d\ (z)C(z)P(z)$ differ in a pure delay. Fig. 3 shows the frequency responses. They have identical magnitude plot with the cross-over frequency denoted by f_c but different phase responses. The one-step computational delay leads to degradations in phase margin (PM) of $\Delta\text{PM} = 2\pi f_c T_s$ and in gain margin (GM) of $\Delta\text{PM}\ =\ 6$ dB (as the GMs are $1/K$ and $2/K$, respectively). Therefore, if the computational delay is eliminated, a higher stability margin can be obtained, or higher control system bandwidth is allowed under a given stability margin.

III. Proposed Method

Eliminating the computational delay from the current loop requires a small update latency. Sampling current at the carrier peak or valley is assumed; thus, the duty-cycle of the PWM command should be constrained by software based on the latency to ensure definite switching actions. This requirement is inherently met in the proposed method as it integrates the signal conditioning antiwindup approach [12].

A. Basic Concept

The causal and biproper controller $C(z)$ that has the same number of poles and zeros can be rewritten as

$$C\ (z) = g_\infty\ + \bar{C}\,(z) \tag{3}$$

where g_∞ is a direct feed-through gain and $\bar{C}(z)$ is a strictly proper transfer function involving the dynamic controller states.

The signal conditioning antiwindup approach proposed in [12] reconstructs the controller with the concept of realizable error, as shown in Fig. 4(a). The realized current error is defined as $\overline{\Delta \text{i}}(z)$ and supposed to generate the constrained voltage reference $\bar{u}(z)$ if fed solely to the controller $C(z)$ instead of $\Delta i\ (z) = i^*\ (z) - i(z)$, which yields

$$\begin{aligned}\bar{u}\ (z) =\ & \text{sat}\left\{g_\infty \Delta i\,(z) + \bar{C}\,(z)\,\Delta i\,(z)\right\} \\ =\ & g_\infty\,\overline{\Delta \text{i}}\,(z) + \bar{C}\,(z)\,\overline{\Delta \text{i}}\,(z)\,.\end{aligned} \tag{4}$$

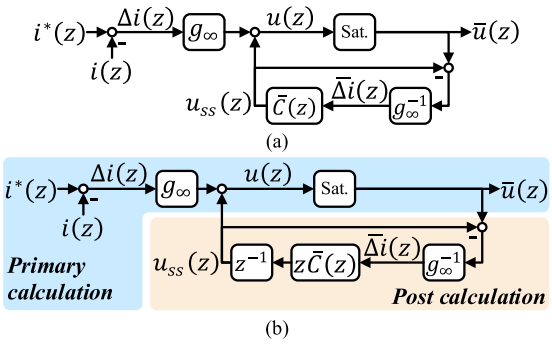

Fig. 4. (a) Conditioned antiwindup current controller architecture [12]. (b) Concept of the proposed scheme to reduce the computation time.

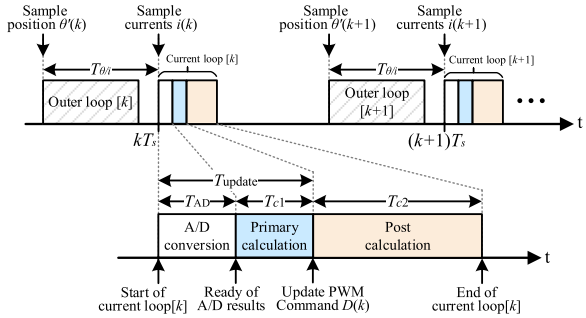

Fig. 5. Schedule for the proposed current control scheme. T_{AD}, T_{c1}, and T_{c2} are the time for A/D conversion, primary calculation, and postcalculation, respectively.

Solving (4) gives the realized current error based on the constrained voltage reference

$$\overline{\Delta \mathrm{i}}\ (z) = g_{\infty}^{-1}\ \left[\bar{u}\,(z) - \bar{C}\,(z)\,\overline{\Delta \mathrm{i}}\,(z)\right]. \tag{5}$$

When the saturation is triggered, using $g_{\infty}\overline{\Delta \mathrm{i}}(z)$ or $g_{\infty}\Delta i(z)$ has an equal contribution to the constrained voltage $\bar{u}(z)$, therefore

$$\begin{aligned}\bar{u}\ (z) &= \mathrm{sat}\left\{g_{\infty}\overline{\Delta \mathrm{i}}\,(z) + \bar{C}\,(z)\,\overline{\Delta \mathrm{i}}\,(z)\right\}\\ &= \mathrm{sat}\left\{g_{\infty}\Delta i\,(z) + \bar{C}\,(z)\,\overline{\Delta \mathrm{i}}\,(z)\right\}.\end{aligned} \tag{6}$$

Combining (5) and (6) gives the conditioned controller architecture in Fig. 4(a) [12]. In addition to the characteristics of antiwindup and no requirement for algebraic reformulation, another feature of this architecture that has not been explored previously will be used here to reduce the updating latency. The variable $u_{ss}(z)$, which can be considered as the steady-state output (when $\Delta i = 0$), is obtained by

$$u_{ss}\ (z) = \bar{C}\ (z)\,\overline{\Delta \mathrm{i}}\ (z) = \frac{g_{\infty}^{-1}\bar{C}\,(z)}{1 + g_{\infty}^{-1}\bar{C}\,(z)}\,\bar{u}\,(z). \tag{7}$$

As $\bar{C}(z)$ is strictly proper, $u_{ss}(z)$ is only involved with the past constrained voltage references and independent of the error information at the present control period. This feature allows calculating the voltage reference with the only $\Delta i(z)$ and $u_{ss}(z)$, and updating other controller states later; thus, the update latency can be reduced. In other words, part of the current loop calculation will be done after refreshing the voltage reference.

Since calculating u_{ss} usually leads to extra multiplication and addition operations with the states in $\bar{C}(z)$ (e.g., the realization of $\bar{C}(z)$ using the direct form II), an explicit delay is separated from $\bar{C}(z)$, as shown in Fig. 4(b) so that u_{ss} can be directly obtained from the state z^{-1} and no other operations are required for u_{ss} in the primary calculation. Meanwhile, $z\bar{C}(z)$ is still proper and realizable because $\bar{C}(z)$ is strictly proper. Then, the current control can be implemented in following two steps.

Step 1. Primary Calculation: Multiplying the current error Δi with g_{∞} and accumulating u_{ss}, and then, the obtained voltage reference is saturated and updated at the end of this step.

Step 2. Post Calculation: Update the states in $z\bar{G}_c(z)$ based on the signals of this period.

Fig. 5 shows the schedule of the controller tasks. The timing to sample the rotor position for the field-oriented control (FOC) of motor drives or phase angle for the voltage-oriented control (VOC) will be discussed in Section III-B. It should be noted that the update latency $T_{\mathrm{update}} = T_{c1} + T_{\mathrm{AD}}$ will be identical in a given hardware platform for either a complicated controller or a simple one, as the computational burden differs in the postcalculation. Thus, those complex controllers, such as several resonant controllers in parallel to eliminate multiple harmonics [5], can benefit more from this architecture in reducing the update latency.

B. Eliminating Frame Transformation in Primary Calculation

The variable θ representing the rotor position in FOC or voltage phase angle in VOC is required in the outer loop and the coordinate transformation. The time to obtain θ may be much longer than the current loop, e.g., the communication with an absolute encoder typically spends dozens of microseconds. To consider a general case and avoid the adverse effect on current control, the instant to sample θ is shifted before current sampling with time $T_{\theta/i}$, as shown in Fig. 5, where $\theta'(k)$ is the sampled value, and θ at the instant of current sampling is calculated by

$$\hat{\theta}\ (k) = \theta'\ (k) + \omega_e T_{\theta/i}. \tag{8}$$

The trigonometric functions required in FOC or VOC can also be calculated in advance within the outer loop.

Most of the current controllers are designed in $\alpha\beta$-frame or *dq*-frame for the advantages including clearer physical insights and less coupling among the controlled variables. To address the frame transformations, the controller in transfer matrix form shown in Fig. 6(a) is considered now, where $\boldsymbol{T}$ and $\boldsymbol{T}_{\mathrm{inv}}$ are the matrixes of transformation and inverse transformation, respectively; the variables of current and voltage become real-value vectors and are denoted in bold type, the gain $\boldsymbol{g}_{\infty}$ is a square matrix, and the subscript "T" indicates a variable in the transformed frame. The method of converting a complex transfer function to a transfer matrix can be found in [13]. Surely, the computation latency is increased with the frame transformations.

To further reduce T_{update}, removing the frame transformation from the primary calculation is expected. It may be a choice to

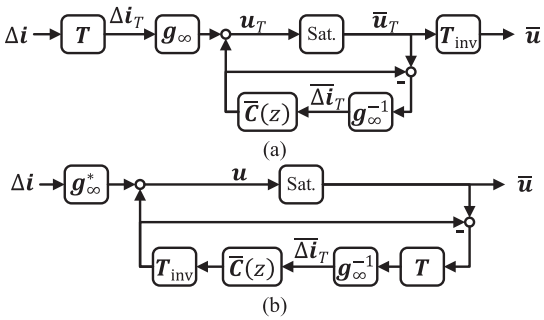

Fig. 6. Controller requiring frame transformation.

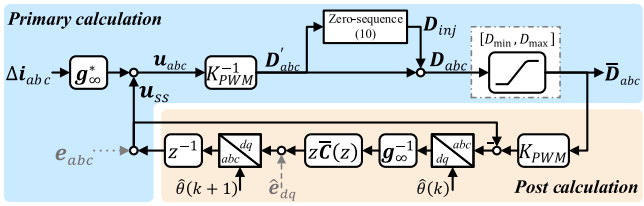

Fig. 7. Block diagram for the proposed fast current control scheme. The feedforward control with $\hat{\boldsymbol{e}}_{dq}$ or $\boldsymbol{e}_{abc}$ is optional and mutually exclusive.

transform the controller in $\alpha\beta$-frame or dq-frame to the three-phase coordinate, but the controller complexity will increase significantly, especially for anisotropic or high-order controllers.

The separation of controller gain and dynamic states in the utilized controller architecture allows a different solution. Only the controller gain is converted, whereas the other part of the controller remains in the original frames, as shown in Fig. 6(b). The new controller gain is calculated by

$$\boldsymbol{g}_{\infty}^{*} = \boldsymbol{T}_{\text{inv}}\, \boldsymbol{g}_{\infty} \boldsymbol{T}. \tag{9}$$

It should be mentioned that the architectures in Fig. 6(a) and (b) may not be identical if different saturation strategies are implemented. In this letter, the saturation is explicitly applied to the duty cycle of each phase PWM command, so that the limitation can be set directly based on the update latency T_{update} and the utilization of dc-bus voltage can be maximized. The detailed block diagram of the whole current control is shown in Fig. 7, where K_{PWM} scales the voltage command $\boldsymbol{u}_{abc}$ to per-unit value $\boldsymbol{D}'_{abc}$ and can be omitted by being included in $\boldsymbol{g}_{\infty}$. Noting that $\hat{\theta}(k+1)$ is used for the dq-to-abc transformation instead of $\hat{\theta}(k)$ to compensate the effect of frame rotation as the delay block here is behind the transformation. The space vector PWM uses the well-known zero-sequence injection method [14] with the injected signals being

$$D_{inj} = \frac{1}{2}\left(1 - \max \boldsymbol{D}'_{abc} - \min \boldsymbol{D}'_{abc}\right). \tag{10}$$

In real-time control, the maximum value of T_{update} is bounded, which can be known in advance or measured in runtime. Then, the constraint for the PWM command is given by

$$\underbrace{T_{\text{update}}/T_s}_{D_{\min}} \le \bar{D}_a, \bar{D}_b, \bar{D}_c \le 1 - \underbrace{T_{\text{update}}/T_s}_{D_{\max}}. \tag{11}$$

C. Feedforward Control

The voltage feedforward control can be combined into the proposed scheme. A feedforward signal denoted by $\boldsymbol{e}_{abc}$ is added to the output voltage $\boldsymbol{u}_{abc}$; since the antiwindup will feedback the output voltage to the controller, the effect of $\boldsymbol{e}_{abc}$ should be counteracted, resulting in $\boldsymbol{e}_{abc}$ injected to the control loop as the gray dotted arrow shown in Fig. 7. The feedforward control can benefit from the reduced computational delay, providing that $\boldsymbol{e}_{abc}$ is timely. In grid-connected voltage source converters, the sampling instants of the grid voltage are not constrained like the converter current; thus, it is recommended to sample the grid voltage and finish calculating the feedforward signal prior to the current feedback control, e.g., in the outer loop [cf., Fig. 5], so that a timely $\boldsymbol{e}_{abc}$ would be available in the primary calculation. In motor drives, an alternative is feedforwarding the estimated back electromotive force (EMF) $\hat{\boldsymbol{e}}_{dq}$ in the synchronous reference frame as the gray dashed arrow shown in Fig. 7. The delay block would have a quite limited effect on feedforwarding $\hat{\boldsymbol{e}}_{dq}$ as it is usually quasi-constant (proportional to the rotor speed whose rate of change is much slower).

IV. Experimental Results

The experiments are conducted on a three-phase PMSM connected to a voltage-source inverter with the parameters being pole-pairs $P_r = 2$, phase resistance $R_s = 0.29\ \Omega$, inductance $L_s = 0.5$ mH, dc-bus voltage $u_{\text{dc}} = 30$ V, and switching frequency $f_{\text{sw}} = 10$ kHz. The dual-sampling-and-updating scheme is applied with a sampling period $T_s = 50\ \mu$s. Another PMSM is mechanically coupled with the tested motor as a load. A control board with a 200-MHz microcontroller TMS320F28379D is used, where two 12-b A/D converters sample the currents of phases A and B simultaneously. No other coprocessor is used. The tested motor is operated in current control mode in the experiments.

A. Stability Margin

The PI controller in (2) is used as the current controller to compare the robustness of current control with and without the computational delay. Fig. 8 shows the current responses with the controller parameters $\hat{L}_s = \lambda L_s$, $\hat{R}_s = \lambda R_s$ slowly increasing from $\lambda = 1$ to $\lambda = 9$. When the one-step computational delay exists, observable oscillation appears at $\lambda \approx 3.3$ and soon the drive protection is triggered, as shown in Fig. 8(a). This is coincident with the theoretical GM ($1/K = 3.33$). The contrastive results with the proposed method to minimize the computational delay are given in Fig. 8(b). The oscillation appears at $\lambda \approx 5.4$, which may be resulted from the measurement noise who has become more influential. At $\lambda \ge 2/K = 6.66$ the system does not go unstable, which could be the limited dc-bus voltage and antiwindup mechanism. Nevertheless, a more robust performance against parameter variations with the reduced computational delay is validated.

To evaluate the small-signal performance, a 0.3-A-sinusoidal perturbation is summed to the constant reference current, and the calculated open-loop frequency responses are shown in Fig. 9.

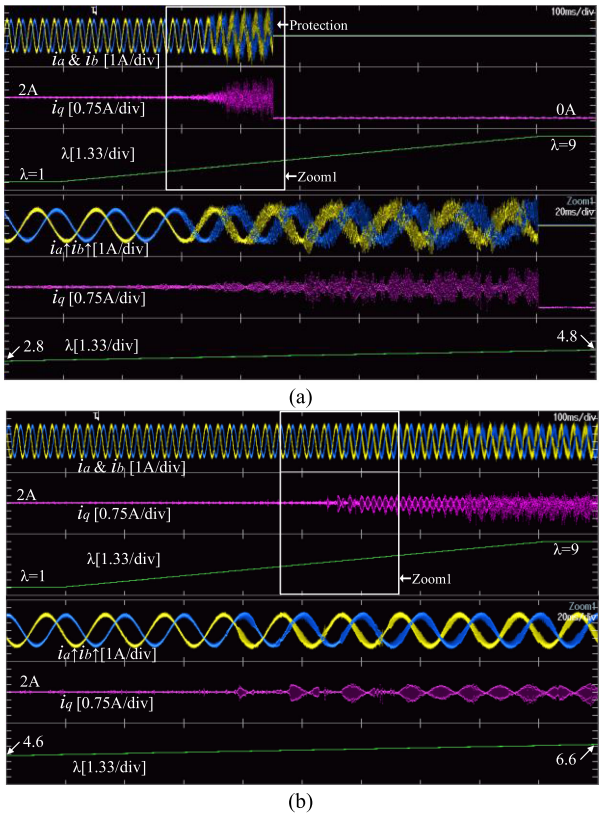

Fig. 8. Current responses (a) with and (b) without the one-step computational delay when $\lambda = \hat{L}_s/L_s = \hat{R}_s/R_s$ slowly increases from 1 to 9 @1500 r/min. (Condition: $K = 0.3$).

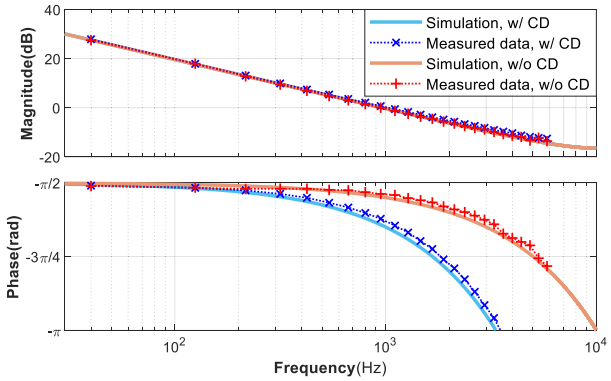

Fig. 9. Measured open-loop frequency responses when $K = 0.3$. The abbreviation "CD" denotes the one-step computational delay.

At each high-frequency test point, 100 sequential sine waves are recorded to increase the signal-to-noise ratio. The measured responses appear to be accordant with the simulation results; the tiny difference may come from the mismatch of the controller parameter or the current scaling error. The magnitude plots of the responses with and without the one-step computational delay are almost identical, whereas the phase plots differ. Higher PM of the response with the proposed method is observed. Therefore, the improvement achieved in reducing the computational delay is proved.

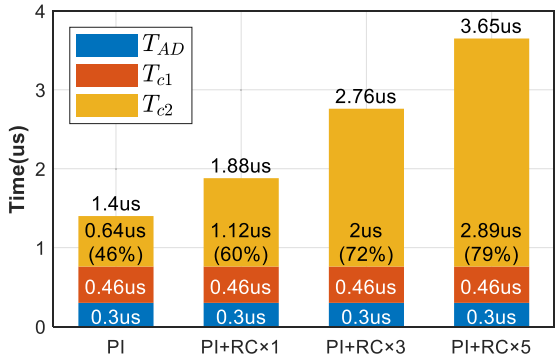

Fig. 10. Time consumption with the proposed scheme.

B. Evaluation of Time Consumption

The time consumption of the proposed scheme is measured by recoding the counter of the PWM timer at the corresponding instants. In addition to the single PI controller, more complicated controllers are evaluated by adding multiple resonant controllers (RCs) in parallel. The results are shown in Fig. 10. The algorithm is programmed in C code and the compiler optimization level is "-o2." The update latency $T_{\text{update}} = T_{c1} + T_{\text{AD}} = 0.76\ \mu s$ is identical for the tested controllers as discussed in Section III-A, and the complex controllers can benefit more by migrating the computational burden to the postcalculation. Accordingly, the constraints for the PWM duty cycle can be set as $D_{\min} = 1.6\%$ and $D_{\max} = 98.4\%$ with a little margin, yielding an acceptable loss of 3.2% in the output-voltage range.

V. Conclusion

This letter develops a method to reduce the update latency from current sampling to refreshing PWM command with a reconstructed controller architecture, which allows us to calculate the PWM command with minimal steps in the primary calculation and avoid frame transformation. Especially, the update latency is identical for either simple or complex linear controllers. Antiwindup is also inherited from the controller structure. With the proposed method, small update latency is realizable in conventional microcontrollers; thus, the computational delay is minimized with negligible loss in output-voltage range, and the stability margin or the current control bandwidth can be expanded.

V. References

[1] D. G. Holmes, T. A. Lipo, B. P. McGrath, and W. Y. Kong, "Optimized design of stationary frame three phase AC current regulators," *IEEE Trans. Power Electron.*, vol. 24, no. 11, pp. 2417–2426, Nov. 2009.

[2] A. G. Yepes, A. Vidal, J. Malvar, O. López, and J. Doval-Gandoy, "Tuning method aimed at optimized settling time and overshoot for synchronous proportional-integral current control in electric machines," *IEEE Trans. Power Electron.*, vol. 29, no. 6, pp. 3041–3054, Jun. 2014.

[3] H. Kim, M. W. Degner, J. M. Guerrero, F. Briz, and R. D. Lorenz, "Discrete-Time current regulator design for AC machine drives," *IEEE Trans. Ind. Appl.*, vol. 46, no. 4, pp. 1425–1435, Jul. 2010.

[4] B.-H. Bae and S.-K. Sul, "A compensation method for time delay of full-digital synchronous frame current regulator of PWM AC drives," *IEEE Trans. Ind. Appl.*, vol. 39, no. 3, pp. 802–810, May 2003.

[5] F. Hans, W. Schumacher, S.-F. Chou, and X. Wang, "Design of multifrequency proportional–resonant current controllers for voltage-source converters," *IEEE Trans. Power Electron.*, vol. 35, no. 12, pp. 13573–13589, Dec. 2020.

[6] X. Yuan, I. Laird, and S. Walder, "Opportunities, challenges, and potential solutions in the application of fast-switching SiC power devices and converters," *IEEE Trans. Power Electron.*, vol. 36, no. 4, pp. 3925–3945, Apr. 2021.

[7] P. Karamanakos and T. Geyer, "Guidelines for the design of finite control set model predictive controllers," *IEEE Trans. Power Electron.*, vol. 35, no. 7, pp. 7434–7450, Jul. 2020.

[8] S. Buso and T. Caldognetto, "A nonlinear wide-bandwidth digital current controller for DC–DC and DC–AC converters," *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7687–7695, Dec. 2015.

[9] M.-W. Naouar, E. Monmasson, A. A. Naassani, I. Slama-Belkhodja, and N. Patin, "FPGA-based current controllers for AC machine drives—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1907–1925, Aug. 2007.

[10] D. Yang, X. Ruan, and H. Wu, "A real-time computation method with dual sampling mode to improve the current control performance of the LCL-type grid-connected inverter," *IEEE Trans. Ind. Electron.*, vol. 62, no. 7, pp. 4563–4572, Jul. 2015.

[11] X. Ke and B. Buchman, "Current control for passivity of a power converter," Int. Patent Appl. WO2020048579A1, Dec. 3, 2020.

[12] B. P. McGrath, D. G. Holmes, and L. McNabb, "A signal conditioning antiwindup approach for digital stationary frame current regulators," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 6036–6046, Nov. 2019.

[13] L. Harnefors, "Modeling of three-phase dynamic systems using complex transfer functions and transfer matrices," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 2239–2248, Aug. 2007.

[14] A. M. Hava, R. J. Kerkman, and T. A. Lipo, "A high-performance generalized discontinuous PWM algorithm," *IEEE Trans. Ind. Appl.*, vol. 34, no. 5, pp. 1059–1071, Sep. 1998.