energies

MDPI

Article

Direct Digital Design of PIDF Controllers with ComPlex Zeros for DC-DC Buck Converters

Stefania Cuoghi [1], Lorenzo Ntogramatzidis [1], Fabrizio Padula [1] and Gabriele Grandi [2,*]

[1] School of Electrical Engineering, Computing and Mathematical Sciences, Curtin University, Bentley 6102, Western Australia, Australia; stefania.cuoghi@curtin.edu.au (S.C.); l.ntogramatzidis@curtin.edu.au (L.N.); fabrizio.padula@curtin.edu.au (F.P.)

[2] Department of Electrical, Electronic, and Information Engineering, University of Bologna, 40136 Bologna, Italy

* Correspondance: gabriele.grandi@unibo.it; Tel.: +39-051-2093-571

Received: 22 November 2018; Accepted: 18 December 2018; Published: 23 December 2018

Abstract: This paper presents a new direct digital design method for discrete proportional integral derivative PID + filter (PIDF) controllers employed in DC-DC buck converters. The considered controller structure results in a proper transfer function which has the advantage of being directly implementable by a microcontroller algorithm. Secondly, it can be written as an Infinite Impulse Response (IIR) digital filter. Thirdly, the further degree of freedom introduced by the low pass filter of the transfer function can be used to satisfy additional specifications. A new design procedure is proposed, which consists of the conjunction of the pole-zero cancellation method with an analytical design control methodology based on inversion formulae. These two methods are employed to reduce the negative effects introduced by the complex poles in the transfer function of the buck converter while exactly satisfying steady-state specifications on the tracking error and frequency domain requirements on the phase margin and on the gain crossover frequency. The proposed approach allows the designer to assign a closed-loop bandwidth without constraints imposed by the resonance frequency of the buck converter. The response under step variation of the reference value, and the disturbance rejection capability of the proposed control technique under load variations are also evaluated in real-time implementation by using the Arduino DUE board, and compared with other methods.

Keywords: buck converter; inversion formulae; phase margin; gain crossover frequency

1. Introduction

Many industrial applications need the transformation of a constant DC voltage source to a constant value even under load variation, such as photovoltaic systems, mobile power supply equipment, DC supply systems, etc. The buck converter is one of the most widely utilized DC-DC converters, because of its simplicity, high efficiency, and low cost, see e.g., [1], and therefore each improvement has potentially a major economic and commercial impact. However, the presence of its nonlinear characteristics in the switching behavior and the saturation of the duty cycle render the output voltage control a challenging task. Numerous control strategies have been proposed for the voltage regulation of the buck converter. Each of them has advantages and disadvantages, and the selection of the most appropriate one depends mainly on the design task at hand. A brief review of the main digital control techniques can be found in [2]. Among these, the non-linear sliding mode control, which leads to fast transient response under load variation and high robustness, is worth mentioning [3,4]. However, the control performance is reduced by the introduction of high frequency oscillations around the sliding surface, the so-called chattering. Another practical alternative for the voltage regulation of the buck converter is the fuzzy logic control. This non-linear adaptive technique

provides a robust performance under parameter variations and load disturbances, and can operate with noise and disturbance of different natures. However, these controllers are traditionally designed by trial-and-error, and this, combined with the rich architecture of the controller, constitutes a major drawback in carrying out stability and performance analysis, as well as transfer function and small signal analysis [5]. By contrast, classical linear proportional integral/proportional integral derivtive (PI/PID) control techniques are widely used by industrial practitioners for their simplicity in the design and implementation, and still by far play a major role. In fact, PID control is often taken as a benchmark for comparison with new strategies since it provides a good compromise among various types of performance indices, including voltage tracking and disturbance rejection, while guaranteeing a satisfactory robustness to small variations of the parameters of the buck converter [1]. Many PID-based strategies have also been combined with non-linear techniques to improve the closed-loop performance [2,3,6–10]. However, in the vast majority of the cases, the PID controller is designed in the continuous-time and then, for its practical implementation, it is converted to the discrete-time, see [10–12].

A common approach to the control feedback design involving PID controllers is to consider the ideal (improper) PID transfer function, which is non-causal. By contrast, the discretization of the ideal PID controller results in a casual, thus feasible, discrete transfer function. This explains why, in this context, the discretization appears to be critical. Indeed, frequency domain specifications assigned in the continuous-time domain can be affected by large undesired variations due to the discretization of the controller. Another critical issue in the design of PID controllers for the buck converter is caused by the presence of a resonance peak in the transfer function of the process. In fact, the resonance usually constrains the assignability of the closed-loop bandwidth, which has to be either well below the resonance frequency, or well above. The former solution is usually discarded since, for obvious reasons, it leads to poor performance. However, the latter is typically associated to a very large bandwidth, which is likely to induce severe saturation in the control variable.

This paper presents a new direct design technique for the discrete PID + filter (PIDF) controllers with complex conjugate zeros. Our method hinges on the classical pole-zero cancellation method [10] combined with the so-called discrete "inversion formulae" [13–18]. In the aforementioned design procedure, two parameters of the PIDF controller are used to achieve pole/zero compensation, as in [10], and the remaining two degrees of freedom are used to exactly meet specifications on the phase margin and gain crossover frequency with the use of the inversion formulae. The design approach based on these formulae was first presented for lead, lag and PID controllers in [13–18]. In this paper we introduce a new set of inversion formulae for the design of the time constant of the discrete PIDF controller.

Thanks to the closed-form design of the filter, which guarantees sufficiently large stability margins even in the presence of uncertainty, our method ensures a satisfactory performance in a neighborhood of the operating point.

The approach based on the inversion formulae results in a proper discrete PIDF controller which is directly implementable on a microcontroller, and which exactly satisfies the design requirements in the discrete domain. Thus, unlike the other techniques described above, the specifications are guaranteed to remain exactly satisfied even when considering the discrete implementation of the controller. Note that we avoid indirect tuning procedures and the inherent trial-and-error nature of graphical tuning techniques based on Bode, Nyquist and Nichols plots.

The procedure presented here is analytical in nature, and can be carried out in finite terms via simple equations which are dependent upon the sampling time of the analog-to-digital converter.

The structure of the discrete PIDF controller is obtained from a continuous-time PIDF [19] transfer function through the matched pole-zero mapping discretization method [20]. In this way, the cancelation results in a discrete transfer function, and therefore the controller can be directly designed in the z-domain.

Simulations and comparisons with other methods show the effectiveness of this new control strategy, which can accommodate plant uncertainties and also, importantly, load variations. The proposed method has been first simulated in MATLAB Simulink® and then tested in a real-time digital implementation using the Atmel SAM3X8E microcontroller based on the ARM® Cortex® -M3 processor on an Arduino Due board. The experimental results have been analyzed and compared with classical PID control solutions.

The paper is organized as follows. The digital control schemes and discrete buck converter model are described in Section 2. In Section 3, we propose the discrete PIDF controller with complex conjugate zeros. The control problem and the proposed design solution are presented in Section 4. We describe the simulated and experimental results of the proposed DC-DC buck converter control and the performance comparison with other methods in Section 5. Conclusions and remarks will end the paper.

2. Digital Control Schemes and Discrete Buck Converter Model

The DC-DC buck converter is a step-down switching converter extensively described, e.g., in [21]. The block scheme of the digital voltage mode control and the buck converter circuit considered in this paper are shown in Figure 1. It is assumed that the converter operates in continuous-conduction mode (CCM).

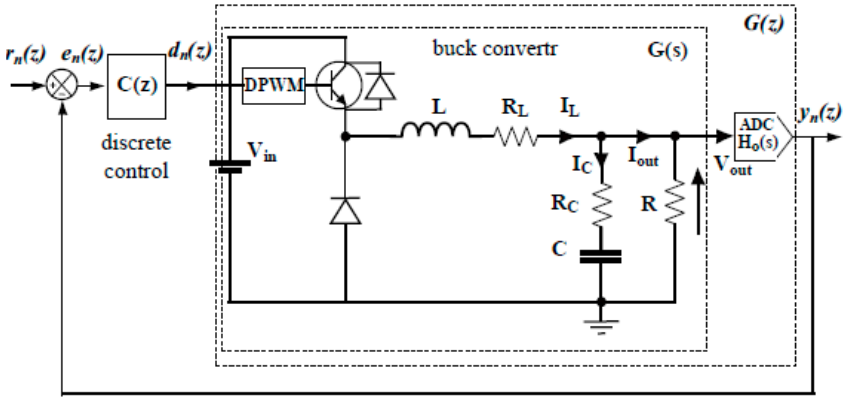

Figure 1. Switching DC-DC converter with digital voltage-mode control.

In the buck converter scheme, V_{out} is the output voltage, V_{in} is the input voltage, L is the filter inductance, C is the filter capacitance, R is the load resistance, R_L and R_C denote, respectively, the parasitic series resistances of the inductor and capacitor. Moreover, $r_n(z)$ represents the reference digital signal, while $y_n(z)$ denotes the sampled output of the process. The sampled signal is obtained by the analog-to-digital converter (ADC) with sampling period T_s. The tracking error signal $e_n(z) = r_n(z) - y_n(z)$ is processed by a discrete-time compensator $C(z)$ to generate the control signal $d_n(z)$. The Digital Pulse Width Modulator (DPWM) converts $d_n(z)$ into the corresponding analog duty cycle with values between 0 and 1 according to the desired ratio of V_{out}/V_{in}, and modulates the PWM signal to drive the buck converter switch.

The transfer function of the discrete plant model $G(z)$ is the Z-transform of the product of the continuous-time converter transfer function $G(s)$ and the transfer function of the zero-order hold:

$$H_0(s) = \frac{1 - e^{-sT_s}}{s}$$

with sampling period T_s:

$$G(z) = \mathrm{Z}[H_0(s)G(s)]. \quad (1)$$

Notice that in the hardware device the output voltage of the buck converter is driven into the admissible range of the ADC input voltage by a constant sensor gain H; the resulting output signal of the ADC is then multiplied by the factor $1/H$ to be compared with the reference value r_n. In (1) the

factors $H \cdot (1/H) = 1$ have been simplified and omitted. According to the buck converter averaged model and Equation (2) of [21], the transfer function of the buck converter:

$$G(s) = \frac{V_{out}(s)}{d(s)} = V_{in} \frac{\left(1 + \frac{s}{\omega_o}\right)}{\left(1 + \frac{2\xi}{\omega_n} s + \frac{s^2}{{\omega_n}^2}\right)} \tag{2}$$

is a second-order low-pass filter, with a left-half complex plane zero introduced by the equivalent series resistance of the filter capacitance.

The mathematical averaged model is obtained by the following input/state/output equations, where the diode and transistor conduction losses have been neglected [22].

$$\underbrace{\begin{bmatrix} \frac{di_L}{dt} \\ \frac{dV_C}{dt} \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} \frac{-R_L}{L} - \frac{RR_C}{L(R+R_C)} & \frac{-R}{L(R+R_C)} \\ \frac{R}{C(R+R_C)} & \frac{-1}{C(R+R_C)} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} i_L \\ V_C \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} \frac{V_{in}}{L} \\ 0 \end{bmatrix}}_{B} d(t),$$

$$V_{out}(t) = \underbrace{\begin{bmatrix} \frac{RR_C}{R+R_C} & \frac{R}{R+R_C} \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} i_L \\ V_C \end{bmatrix}}_{x(t)},$$

where:

$$\omega_n = \frac{1}{\sqrt{LC\frac{R+R_C}{R+R_L}}},\ \omega_o = \frac{1}{R_C C},\ \xi = \frac{\omega_n}{2}\left(R_C C + \frac{RR_L C + L}{(R+R_L)}\right). \tag{3}$$

The discrete model of the buck converter is:

$$G(z) = V_{in} \frac{(1 - a - bc)z + e^{-2\xi\omega_n T_s} - a + bc}{z^2 - 2az + e^{-2\xi\omega_n T_s}}, \tag{4}$$

where:

$$a = e^{-\xi\omega_n T_s} \cos\left(\omega_n T_s \sqrt{1 - \xi^2}\right), \tag{5}$$

$$b = e^{-\xi\omega_n T_s} \sin\left(\omega_n T_s \sqrt{1 - \xi^2}\right), \tag{6}$$

$$c = \frac{\xi\omega_o - \omega_n}{\omega_o \sqrt{1 - \xi^2}}, \tag{7}$$

and it can be obtained by applying the definition of the Z-transform to the series plants $H_0(s)G(s)H$. Notice that $G(z)$ is characterized the following two complex conjugate poles:

$$z_{1,2} = e^{(-\xi \pm j\sqrt{1-\xi^2})\omega_n T_s}.$$

Indeed, from (1) it follows that:

$$G(z) = Z\left[\frac{1 - e^{-T_s s}}{s} G(s)\right] = \left(1 - z^{-1}\right) Z\left[\frac{G(s)}{s}\right] = \frac{V_{in}{\omega_n}^2}{\omega_0}\left(1 - z^{-1}\right) Z[R(s)],$$

where:

$$R(s) = \frac{s + \omega_o}{s(s^2 + 2\xi\omega_n s + {\omega_n}^2)}. \tag{8}$$

Expanding $R(s)$ into partial fractions we have:

$$\begin{aligned} R(s) &= \frac{\omega_o}{{\omega_n}^2}\frac{1}{s} - \frac{\omega_o}{{\omega_n}^2}\frac{(s+\xi\omega_n)+\left(\xi\omega_n-\frac{{\omega_n}^2}{\omega_o}\right)}{s^2+2\xi\omega_n s+{\omega_n}^2} \\ &= \frac{\omega_o}{{\omega_n}^2}\frac{1}{s} - \frac{\omega_o}{{\omega_n}^2}\frac{(s+\xi\omega_n)}{(s+\xi\omega_n)^2+{\omega_n}^2\left(1-\xi^2\right)} \\ &= -\frac{\xi\omega_o-\omega_n}{{\omega_n}^2\sqrt{1-\xi^2}}\frac{\omega_n\sqrt{1-\xi^2}}{(s+\xi\omega_n)^2+{\omega_n}^2\left(1-\xi^2\right)}. \end{aligned}$$

Applying the standard manipulation theorems of the Z-transform to $R(s)$ we have:

$$\begin{aligned} R(z) &= \frac{\omega_o}{{\omega_n}^2}\frac{z}{z-1} - \frac{\omega_o}{{\omega_n}^2}\frac{z^2-e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2})z}{z^2-2e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2})z+e^{-2\xi\omega_n T_s}} \\ &-\frac{\xi\omega_o-\omega_n}{{\omega_n}^2\sqrt{1-\xi^2}}\frac{e^{-\xi\omega_n T_s}\sin(\omega_n T_s\sqrt{1-\xi^2})z}{z^2-2e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2})z+e^{-2\xi\omega_n T_s}}. \end{aligned}$$

It follows that (8) can be written as:

$$G(z) = V_{in} - V_{in}\frac{(z-e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2}))(z-1)}{z^2-2e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2})z+e^{-2\xi\omega_n T_s}}$$

$$-V_{in}\frac{\xi\omega_o-\omega_n}{{\omega_o}^2\sqrt{1-\xi^2}}\frac{e^{-\xi\omega_n T_s}\sin(\omega_n T_s\sqrt{1-\xi^2})z}{z^2-2e^{-\xi\omega_n T_s}\cos(\omega_n T_s\sqrt{1-\xi^2})z+e^{-2\xi\omega_n T_s}},$$

which can be rewritten as in (4) using (5)–(7).

3. The Proposed Discrete PIDF Controller with Complex Conjugate Zeros

The controller presented in this paper is a discrete PIDF controller, described by the following transfer function:

$$C(z) = \widetilde{K}_i\frac{z^2-2\delta_d\omega_d z+{\omega_d}^2}{(z-1)\left(z-\frac{\omega_n}{\beta_d}\right)}. \tag{9}$$

when:

$$\omega_d = e^{-\frac{\delta}{\tau}T_s},\ \delta_d = \cos\left(\frac{T_s}{\tau}\sqrt{1-\delta^2}\right),\ \beta_d = e^{(\beta-\delta)\frac{T_s}{\tau}},$$
$$\widetilde{K}_i = 2k_i\tau\beta\frac{1+e^{-\frac{\beta T_s}{\tau}}}{1+2e^{-\frac{\delta}{\tau}T_s}\cos\left(\frac{T_s}{\tau}\sqrt{1-\delta^2}\right)+e^{-\frac{2\delta}{\tau}T_s}},$$

the controller (9) represents the discrete pole-zero mapping transformation with the sampling period T_s of the following continuous-time PIDF controller:

$$C(s) = K_i\frac{1+2\delta\tau s+(\tau s)^2}{s\left(1+\frac{\tau}{\beta}s\right)}. \tag{10}$$

Here K_i is the integral gain, δ is the damping ratio and $1/\tau$ is the natural frequency of the controller zeros, and:

$$\beta = \frac{K_\infty}{\tau K_i}$$

is a parameter that depends on the high frequency controller gain, which is defined as:

$$K_\infty = \lim_{s\to\infty} C(s).$$

The PIDF controller (10) is equivalent to the classical parallel PIDF controller:

$$C(s) = K_p\left(1 + \frac{1}{sT_i} + \frac{sT_d}{1+sT_f}\right). \tag{11}$$

In fact, equivalent parameters for (11) can be obtained from δ, β, K_i, τ by equating (10) and (11): the resulting proportional gain, and the integral, the derivative and the filter time constants are shown in the following Equation (12):

$$K_p = K_i\frac{\tau}{\beta}(2\delta\beta - 1),\ T_i = \frac{\tau}{\beta}(2\delta\beta - 1),\ T_d = \frac{\tau}{\beta}\left(\frac{\beta^2}{2\delta\beta - 1} - 1\right),\ T_f = \frac{\tau}{\beta}. \tag{12}$$

Notice that when $\beta > 1$ and $\delta \geq 1$ the PIDF controller (10) reduces to a series PID controller, when $0 < \beta < 1$ the PIDF controller has complex conjugate zeros, and when $\beta = 1$ and $\delta = 1$ the PIDF controller becomes a PI controller, see [19].

Interestingly, the controller (9) can be written as a digital biquadratic filter:

$$C(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \tag{13}$$

where:

$$b_0 = \widetilde{K}_i,\ b_1 = -2\widetilde{K}_i\delta_d\omega_d,\ b_2 = \widetilde{K}_i{\omega_d}^2,$$
$$a_1 = -\left(\frac{\omega_d}{\beta_d} + 1\right),\ a_2 = \frac{\omega_d}{\beta_d},$$

which has the clear advantage of being directly implementable on a microcontroller by using the difference equation:

$$d[n] = \sum_{i=0}^{2} b_i e[n-i] - \sum_{j=1}^{2} a_j d[n-j]. \tag{14}$$

4. The Design Problem and the Proposed Design Solution

For control design purposes, the control system scheme can be simplified as in Figure 2, where $G(z)$ and $C(z)$ are given by (1) and (9), respectively, while $L(z)$ denotes the loop gain transfer function $L(z) = C(z)G(z)$.

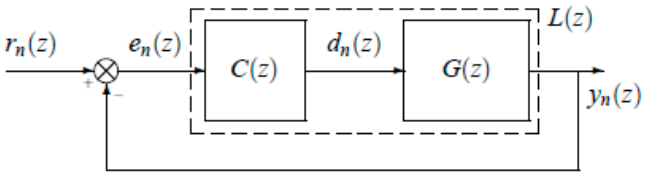

Figure 2. Block scheme of the digital control system.

4.1. Pole/Zero Compensation Method

The PIDF controller (9) introduces a pair of complex conjugate zeros, which can be placed to achieve pole/zero compensation. The PIDF parameters can be selected as follows:

$$\omega_d = e^{-\xi\omega_n T_s},\ \delta_d = \cos\left(\omega_n T_s\sqrt{1-\xi^2}\right), \tag{15}$$

where ξ and ω_n are the parameters of the buck converter described in (3). In this way, the transfer function of the controller can be factorized into two parts. The zeros and the integrator:

$$\frac{z^2 - 2\delta_d\omega_d z + {\omega_d}^2}{z-1} \tag{16}$$

are completely determined by the zero/pole cancellation. The remaining factor is:

$$\widetilde{C}(z) = \frac{\widetilde{K}_i}{z - \frac{\omega_d}{\beta_d}} \tag{17}$$

and it comprises the parameters that are yet to be assigned. Thus, we can consider a new control problem where the controller is $\widetilde{C}(z)$, while the former factor is part of the plant, whose transfer function then becomes:

$$\widetilde{G}(z) = G(z)\frac{z^2 - 2\delta_d\omega_d z + {\omega_d}^2}{z-1}.$$

4.2. Discrete Inversion Formulae Method

The design method based on the so-called "inversion formulae" consists in a set of closed-form expressions that deliver the parameters of the controller to exactly satisfy specifications on the gain crossover frequency ω_g, phase margin Φ_m and/or gain margin G_m. In most cases, these specifications are satisfied if the frequency response associated with the loop gain transfer function:

$$L(e^{\mathrm{j}\omega T_s}) = C(e^{\mathrm{j}\omega Ts})\, G(e^{\mathrm{j}\omega T_s})$$

at frequency ω_g satisfies:

$$\left|L(e^{\mathrm{j}\omega_g T_s})\right| = 1,\ \angle L(e^{\mathrm{j}\omega_g T_s}) = \Phi_m + \pi\,. \tag{18}$$

In other words, the design method based on the inversion formulae is a way of constraining the loop gain polar plot to cross a specific point of the complex plane. In practice, in the vast majority of the situations that are interesting in practice this goal alone is sufficient to guarantee that the specifications on the phase (or gain) margin and crossover frequency are met, see [15] for further details.

The classical feedback design problem is to find a controller $C(z)$ that satisfies the steady-state a zero position error specification, and such that the gain crossover frequency and the phase margin of the loop gain transfer function $L(z)$ are, respectively, ω_g and Φ_m.

The first step of the design method consists in guaranteeing that the steady-state requirement is met. In most situations, the pole at $z = 1$ of the controller is sufficient to automatically satisfy the steady-state requirements. However, in some cases, the number of poles at $z = 1$ of the plant and the single pole at $z = 1$of the controller are not sufficient to meet the desired static requirements, and the factor $\widetilde{K}_i$ in (9) must be chosen accordingly. For example, this is the case of a type-0 plant as the considered buck converter when the steady-state specifications not only require zero position error, but also that the velocity error (i.e., the tracking error in the response of a ramp) be equal to (or smaller than) a given non-zero constant. In the considered case specifications on the steady-state error do not lead to constraints in the value of the integral constant. Let $L(z) = \widetilde{C}(z)\widetilde{G}(z)$ be the loop gain transfer function. We define:

$$M_g \stackrel{def}{=} M(\omega_g) = 1/\left|\widetilde{G}(e^{j\omega_g T_s})\right|, \tag{19}$$

$$\varphi_g \stackrel{def}{=} \varphi(\omega_g) = \Phi_m - \pi - \angle\widetilde{G}(e^{j\omega_g T_s}). \tag{20}$$

The solvability of the feedback design problem amounts to solving the complex equation:

$$L(e^{\mathrm{j}\omega_g T_s}) = e^{\mathrm{j}(\Phi_m - \pi)}$$

in the unknowns $\widetilde{K}_i > 0$ and $\beta_d > 0$. The closed-form solution to this problem is given in the following theorem:

Theorem 1. *The values of $\widetilde{K}_i$ and β_d that solve the control problem are given by the following expressions:*

$$\beta_d = \frac{\omega_d}{\frac{\sin(\omega_g T_s)}{\tan(\varphi_g)} + \cos(\omega_g T_s)}, \tag{21}$$

$$\widetilde{K}_i = -M_g \sin(\varphi_g) \sin(\omega_g T_s) \left(1 + \frac{1}{\tan^2(\varphi_g)}\right). \tag{22}$$

Proof. From (18), the controller (17) has to be designed in such a way that:

$$\widetilde{C}(e^{j\omega_g T_s}) = M_g e^{j\varphi_g} = M_g(\cos\varphi_g + j\sin\varphi_g), \tag{23}$$

holds. The frequency response of (17) for $\omega = \omega_g$ can be written in Cartesian form as:

$$\widetilde{C}(e^{j\omega_g T_s}) = \frac{\widetilde{K}_i}{e^{j\omega_g T_s} - \frac{\omega_d}{\beta_d}}. \tag{24}$$

Equating (24) and (23) directly leads to (21) and (22). □

Remark 1. *It is easy to verify that the parameters β_d and $\widetilde{K}_i$ in (21) and in (22) are positive if and only if:*

$$\tan(\omega_g T_s) > -\tan(\varphi_g) \tag{25}$$

and one of the following conditions holds:

$$\begin{aligned}
&\bullet \omega_g \in \left[0, \tfrac{\pi}{2T_s}\right] && \textit{and} && \varphi_g \in \left[\pi, \tfrac{3}{2}\pi\right],\\
&\bullet \omega_g \in \left[\tfrac{\pi}{2T_s}, \tfrac{\pi}{T_s}\right] && \textit{and} && \varphi_g \in \left[\tfrac{3}{2}\pi, 2\pi\right],\\
&\bullet \omega_g \in \left[\tfrac{\pi}{T_s}, \tfrac{3\pi}{2T_s}\right] && \textit{and} && \varphi_g \in \left[\tfrac{\pi}{2}, \pi\right],\\
&\bullet \omega_g \in \left[\tfrac{3\pi}{2T_s}, \tfrac{2\pi}{T_s}\right] && \textit{and} && \varphi_g \in \left[0, \tfrac{\pi}{2}\right],
\end{aligned}$$

or:

$$\tan(\omega_g T_s) < -\tan(\varphi_g) \tag{26}$$

and one of the following conditions holds:

$$\begin{aligned}
&\bullet \omega_g \in \left[0, \tfrac{\pi}{2T_s}\right] && \textit{and} && \varphi_g \in \left[\tfrac{3}{2}\pi, 2\pi\right],\\
&\bullet \omega_g \in \left[\tfrac{\pi}{2T_s}, \tfrac{\pi}{T_s}\right] && \textit{and} && \varphi_g \in \left[\pi, \tfrac{3}{2}\pi\right],\\
&\bullet \omega_g \in \left[\tfrac{\pi}{T_s}, \tfrac{3\pi}{2T_s}\right] && \textit{and} && \varphi_g \in \left[0, \tfrac{\pi}{2}\right],\\
&\bullet \omega_g \in \left[\tfrac{3\pi}{2T_s}, \tfrac{2\pi}{T_s}\right] && \textit{and} && \varphi_g \in \left[\tfrac{\pi}{2}, \pi\right].
\end{aligned}$$

Note that, if one of the previous conditions fails, the required frequency-domain constraints are infeasible. In other words, the devised inversion formulae provide a solution whenever a feasible solution exists.

It is worth stressing that the proposed approach is based on closed-form expressions that deliver a discrete-time PIDF controller that satisfies exactly the design specification. This is clearly a major advantage since the imposed stability margin is guaranteed, and it is not subject to variations induced by the discretization method.

5. Design of the Buck Converter

5.1. Design Problem

The aim of this section is to apply the proposed designed procedure to the buck converter circuit with the parameters given in Table 1. The steady state requirement is zero position error, while the phase margin and the gain crossover frequency of the open loop frequency response are required to be equal to $\Phi_m = 85^\circ$ and $\omega_g = 1600$ rad/s, respectively.

Table 1. Circuit parameters of the buck converter.

Parameter	Symbol	Value	Units
Input voltage	V_{in}	20	V
Reference voltage	V_{ref}	12	V
Filter Capacitance	C	100	μF
Filter Inductance	L	680	μH
Load resistance	R	20	Ω
ESR of capacitor	R_C	170	$m\Omega$
ESR of inductor	R_L	173	$m\Omega$

5.2. Proposed Solution Using Discrete-Time PIDF Controller

The discrete plant (4) of the buck converter with the parameters given in Table 1 and with sampling period T_s equal to 5×10^{-5} s is:

$$G(z) = \frac{0.603\ z + 0.1122}{z^2 - 1.916z + 0.9513}. \tag{27}$$

The same result can be obtained using the zero-order-hold discretization method on the transfer function of the continuous time averaged model (2):

$$G(s) = \frac{V_{out}(s)}{d(s)} = \frac{5001s + 2.942 \times 10^8}{s^2 + 998.1s + 1.471 \times 10^7}. \tag{28}$$

The steady-state requirements are automatically satisfied by the pole at $z = 1$ of the discrete PIDF controller. Its zeros can be designed to cancel the complex poles of $G(z)$ at $0.96 \pm \mathrm{j}\ 0.18$ by selecting $\delta_d = 0.982$ and $\omega_d = 0.97$ rad/s in (9). It follows that:

$$\widetilde{G}(z) = \frac{0.603\ z + 0.1122}{z - 1}. \tag{29}$$

The complex value $\widetilde{G}(e^{\ \mathrm{j}\omega_g T_s}) = 8.94\ e^{\ \mathrm{j}\ 1.54}$ determines the gain $M_g = 1/8.94 = 0.11$ that the controller has to introduce at frequency ω_g, and the phase $\varphi_g = 85^\circ + 180^\circ + 88.4^\circ = 353.4^\circ$ of the controller at ω_g needed to satisfy the design specification on the phase margin. The parameters of the PIDF controller (9) that solves the problem are $\beta_d = 3.22$, $\widetilde{K}_i = 0.078$, and follow directly from (21–22). The resulting PIDF transfer function is:

$$C(z) = \frac{0.0781z^2 - 0.1496z + 0.0743}{z^2 - 1.303z + 0.3033}, \tag{30}$$

which can also be rewritten as:

$$C(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \tag{31}$$

with $a_1 = -1.303$, $a_2 = 0.3033$, $b_0 = 0.0781$, $b_1 = -0.1496$, $b_2 = 0.0743$. Applying this discrete controller, the design requirements are exactly satisfied, as one can observe by the Nyquist and Bode plots of the open loop frequency response $L(e^{\ \mathrm{j}\omega T_s})$ shown in red in Figures 3 and 4. The step response of the

controlled system is plotted in red in Figure 5 showing the effectiveness of the control in the time domain. Notice that selecting a different value of the sampling time T_s causes the complex conjugate poles of the discrete plant to shift in the complex plane. In this case, new values of δ_d and ω_d can be computed according to (9) as functions of T_s to exactly cancel the shifted poles.

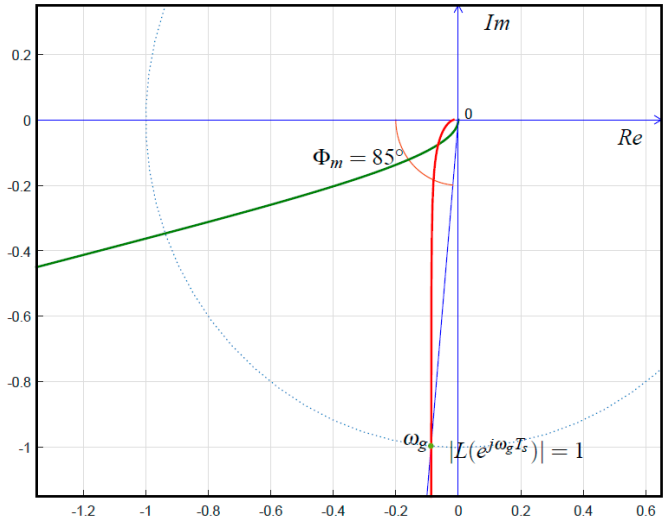

Figure 3. The Nyquist plot of the frequency response of the buck converter (green), and of the open loop frequency response with the discrete-time inversion formulae (red).

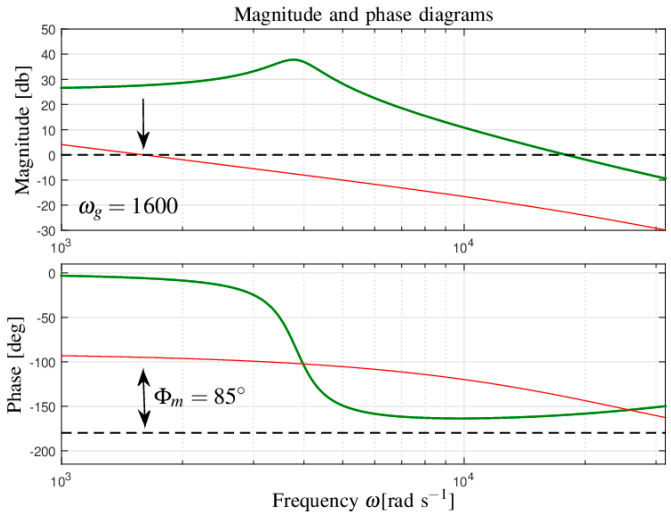

Figure 4. Bode diagrams of the frequency response of the buck converter (green), and of the open loop frequency response with the discrete-time inversion formulae (red).

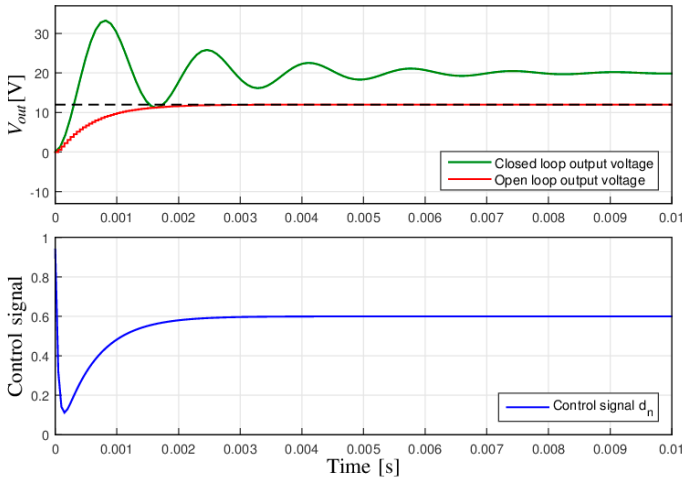

Figure 5. Open-loop step response (green), closed-loop step response (red) and the corresponding control variable (blue).

5.3. Simulation and Experimental Results

The proposed control system for the buck converter regulation has been extensively simulated in MATLAB-Simulink® using the model shown in Figure 6. As a first step, the PIDF controller has been tested introducing the discrete transfer function block contained in the Simulink® library. Then, this block has been substituted with the Infinite Impulse Response (IIR) digital filter shown in Figure 7, which has the advantage to be directly implementable by a microcontroller algorithm.

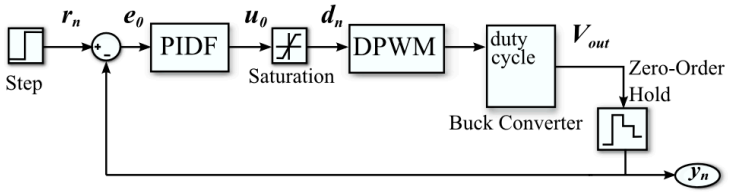

Figure 6. MATLAB-Simulink® model of the buck converter and control system.

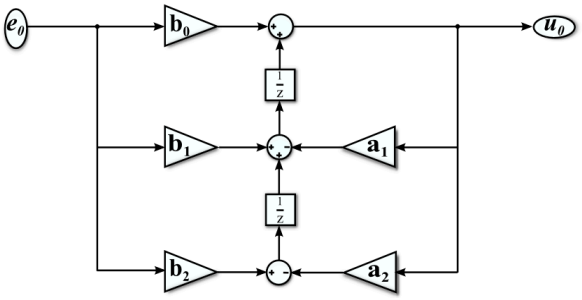

Figure 7. Discrete-time PIDF controller model: biquad cascade IIR filters using a direct form II transposed structure.

The control signal d_n, the inductor current and the output voltage of the converter and inductor under step reference variations from 0 V to 12 V are shown in Figure 8 from which the smoothness and monotonicity of the response achieved with our method can be clearly observed, as well as the notching effect of the complex conjugate zeros, which is well-visible in the first part of the transient response of the control signal. It is also worth noting that the simulated response and the experimental one exhibit a very good matching, demonstrating that our model is effectively descriptive of the real-world buck converter.

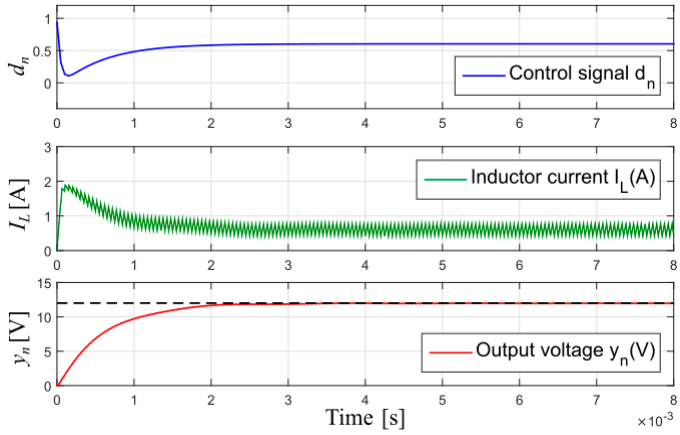

Figure 8. Simulated control signal, inductor current and output voltage under step variation of the reference value from 0 V to 12 V using the proposed control method.

An experimental hardware device has been built to verify the proposed method for the DC-DC buck converter. It is composed by the buck converter and an Arduino Due development board, based on a 32-bit Atmel SAM3X8E ARM® Cortex® M3 CPU, see Figure 9. The main components of the buck converter circuit have been selected as shown in Table 1.

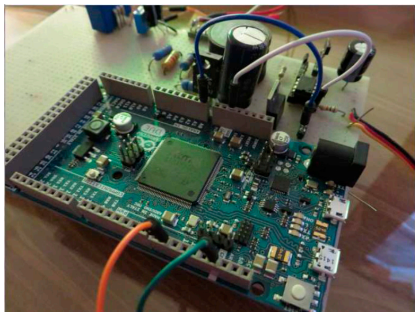

Figure 9. Physical realization of the buck converter.

An interrupt routine is generated every $T_s = 5 \times 10^{-5}$ s. This routine starts the ADC conversion of the output signal of the converter and computes the duty cycle of the PWM control signal. Figure 10 represents the difference Equation (14) of the PIDF controller using the Direct-Form II Transposed structure of a Biquad Cascade IIR Filter shown in Figure 7.

```
//·PIDF·control·algorithm
//·implemented·in·Arduino·board
u0·=·b0·*·e0·+·q1;

q1·=·b1·*·e0·-·a1·*·u0·+·q2;

q2·=·b2·*·e0·-·a2·*·u0;
```

Figure 10. The Biquad Cascade IIR Filter algorithm.

The output voltage of the converter and inductor current I_L under step reference variations from 0 V to 12 V are shown in Figure 11. The measure of I_L has been obtained using the analog transducer LEM 6−*NP* with a 5V supply and a galvanic isolation between the primary and the secondary circuit. The experimental results confirm the behavior already observed in the simulations: the output voltage reaches the desired value in a monotonic fashion, and the inductor current remains always well below

the saturation value. A zoom of the inductor current sensor output in steady-state condition is shown in Figure 12 from which the regularity of the PWM duty cycle when the system has reached the new steady-state can be observed. This is a consequence of the selected bandwidth, which is large enough to obtain a fast set point tracking, but narrow enough to avoid the amplification of high frequency noise and discontinuities due to the PWM behavior. Note that assigning such bandwidth without cancelling the complex conjugate poles would result in large oscillations due to the presence of the resonance peak in the closed-loop system.

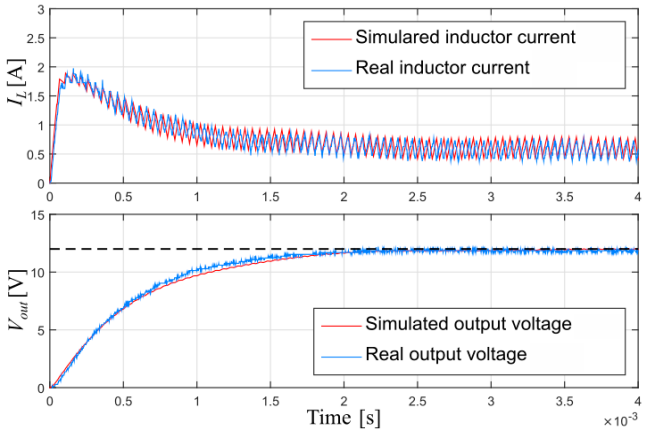

Figure 11. Experimental result using PIDF and the proposed method: inductor current sensor output and converter output signal under reference step variation from 0 V to 12 V.

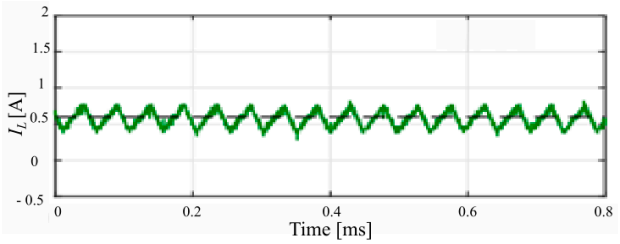

Figure 12. Experimental result using PIDF: zoom of the inductor current sensor output in steady-state condition.

5.4. Comparison with Other Methods

There are various design techniques for determining the parameters of a PID controller when the mathematical model of the plant is explicitly available. For a comparison with the proposed design method, three classical PID tuning techniques have been considered, see Table 2.

Table 2. PID parameters.

Control	K_p	K_i	K_d
IMC-Chien	0.033	958.7	6.519×10^{-5}
Pole placement	0.55	247.1,	7.353×10^{-5}
Pole-zero cancellation	0.02	294.7	2.004×10^{-5}

The first controller has been obtained using the Internal Model Control IMC-Chien method described in [23], and by neglecting the capacitor and inductor resistances in (2). The second has been obtained by placing one zero of the PID controller an octave below the cut-off frequency, approximately at 480 rad/s, while the other zero has been placed at 7×10^3 rad/s, see [12]. The third controller has been obtained by selecting the PID zeros to approximately cancel the complex conjugate poles of the converter at the cut-off frequency and a phase margin equal to 95°, see [10]. The considered

continuous-time PID controllers have been simulated via the Simulink® PID(*s*) block which implements a PID controller in the form:

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d \frac{N}{1 + N/s}. \quad (32)$$

The MATLAB-Simulink® PID(*s*) model uses a lowpass filter in the derivative term to obtain a proper transfer function. The default value of the coefficient N in the filter is set at 100. Using this value in (32), all the considered PID controls generate large oscillations during the step response transient. These oscillations are considerably reduced by setting the coefficient N of the filter to the value N = 200,000. It is clear that the time constant of the filter is a critical component in the design of a PID controller and that a systematic design method should be taken into account. Accordingly, in our method, the time constant of the filter is selected to achieve the desired closed-loop system performance, and it is not designed by using trial-and-error, empiric or rule-of-thumb methods.

For the practical implementation of the controller on the Arduino board, the continuous-time PID controllers are converted to the discrete-time by using the backward Euler's integration method, as suggested in [10]. The discrete control algorithm will therefore implement the causal difference equation:

$$d[n] = K_p e(n) + K_i T_s \sum_{i=0}^{n} e(n) + \frac{K_d}{T_s}[e(n) - e(n-1)].$$

Moreover, an anti-windup filter based on the conditional integration method (see [24] for details) has been implemented in order to minimize the detrimental effect of the large saturation resulting from the techniques listed in Table 3.

Table 3. Phase margin variations from continuous to discrete-time control using backward Euler's discretization method.

PID Control	Phase Margin		
	Continuous-Time	Discrete-Time	
		N = 100,000	N = 200,000
IMC-Chien	90°	47.5°	50.5°
Pole placement	98.6°	26.3°	29.5°
Pole-zero cancellation	95.7°	65.2°	67.6°

The simulated step responses of these three methods in the continuous-time are shown in Figure 13. The simulated step response using the IMC-Chien method is very fast, with a settling time of 0.2 ms. However, the peak of the resulting control signal is approximately 80.

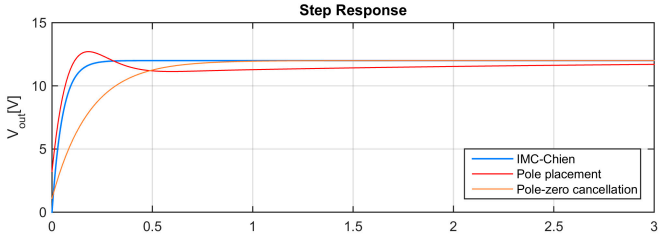

Figure 13. *Cont.*

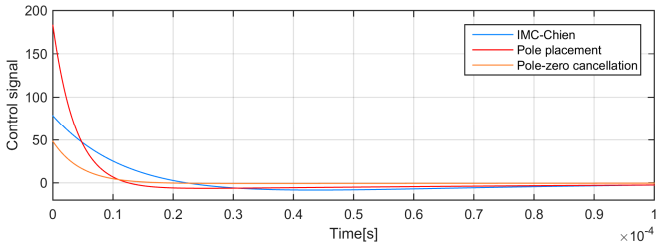

Figure 13. Simulated step responses using IMC-Chien method, the pole placement method and pole-zero control method in the continuous-time case: output voltages and control signals.

On the other hand, the saturation of the duty cycle range [0,1] leads to an oscillatory behavior in the experimental output voltage, see Figure 14b. In fact, the resulting settling time is 20 times greater than the one obtained in the simulated test. Moreover, a steady-state ripple in the output voltage and in the inductor current is present because of the excessively aggressive tuning.

The simulated continuous-time step response using the pole placement method exhibits a rise time of 8.1×10^{-5} s, a settling time of 4.2 ms, and an overshoot equal to 6%. As in the previous case, the control signal reaches a very high value, in this case with a peak of nearly 180. The converter signals obtained using this control method in the experimental hardware device are shown in Figure 14c. The main drawback of this type of control is the large steady-state ripple in the output voltage, see [10]. This is due to an excessively large closed-loop bandwidth that results in an aggressive control action which tries to compensate the high frequency noise. The saturation of the duty cycle in the range [0,1] leads to an ON-OFF behavior in the hardware device and high power dissipation both during the transient response and in the maintenance of the steady-state. Note also that the voltage ripple is unsuitable for most sensitive electronic equipment and the resulting current may cause heating and damage of capacitors over time, see [25].

The simulated continuous-time step response using the pole-zero cancellation method has a rise time of 4.1×10^{-4} s, a settling time of 0.7 ms. The peak of the control signal is 48, which is considerably lower than the ones obtained with the previously described techniques, but still orders of magnitude above the saturation level. The corresponding experimental results are shown in Figure 14d. Notice that the steady-state output ripple is not present using this type of control because of the less aggressive tuning of the parameters, which also results in a lower peak of the control variable. However, the non-linear saturation of the control signal is not considered in (2). As a consequence, the zeros of the controller only partially compensate the oscillatory effects of the buck converter poles in the transient period. It follows that the settling time rises to 4 ms in practice, and the experimental output voltage exhibits an oscillatory behavior with an overshoot of 20–30%.

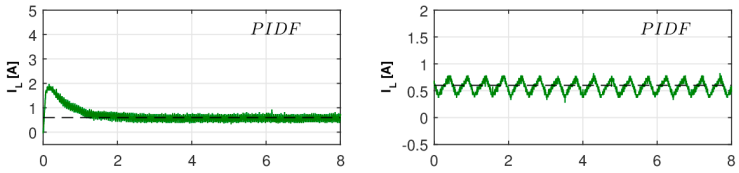

Figure 14. *Cont.*

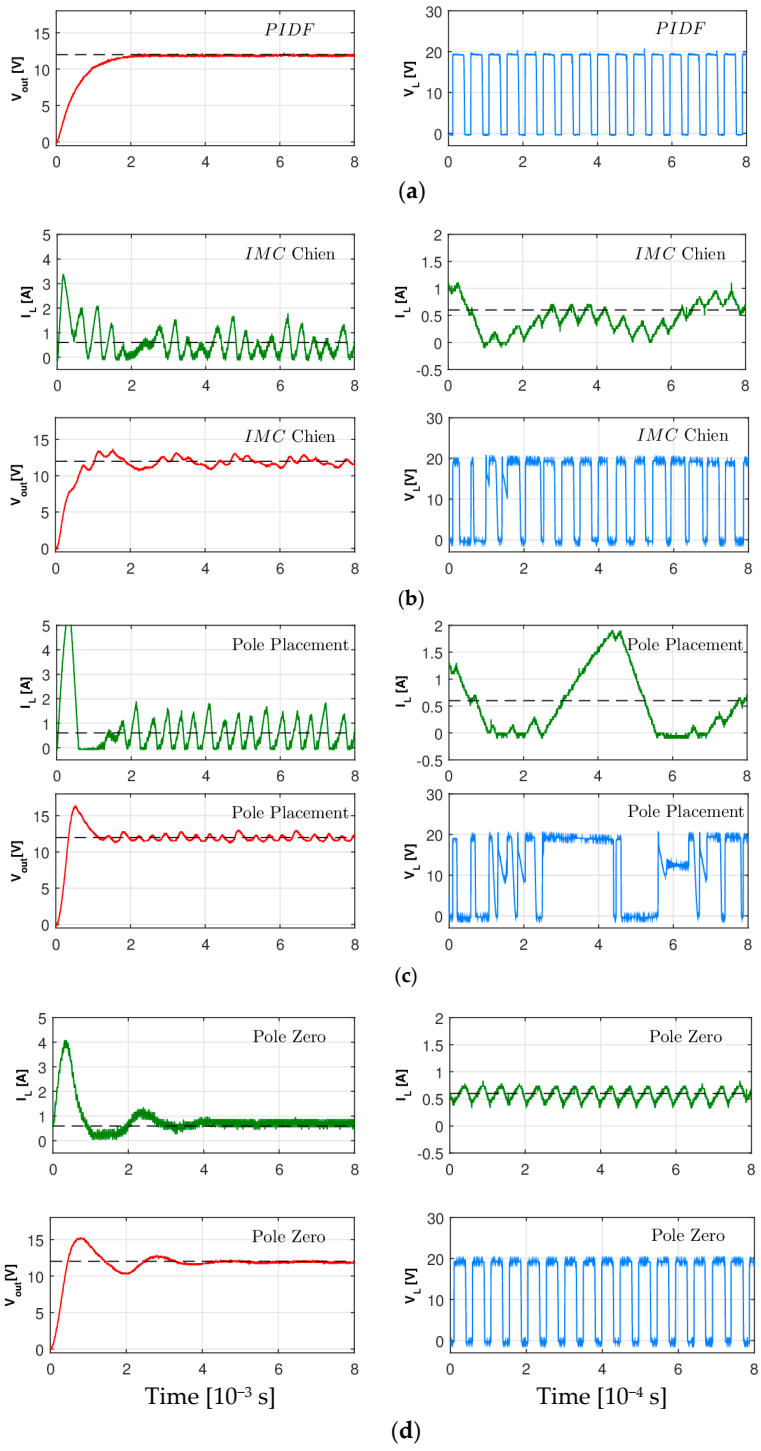

Figure 14. Inductor current and output voltage under step variation of reference value from 0 V to 12 V and zoom of inductor current and inductor current at steady-state using: (**a**) Inversion Formulae, (**b**) IMC-Chien, (**c**) Pole Placement, and (**d**) Pole Zero methods.

Compared to all the considered methods, the proposed control procedure leads to a good matching between simulated and experimental results, due to the design which is carried out directly in discrete-time via closed-form formulae. As such, the phase margin that we obtain with the discrete PIDF controller is exactly the design one. On the contrary, other approaches are based on the design of the controller in the discrete domain, and eventually, on the discretization of the obtained continuous controller. However, this results in a discrete controller that often delivers a phase margin considerably different from the one that

would have ideally been obtained in the continuous time, see Table 2, where the phase margins obtained from the considered methods and a discrete PID of the following form are presented:

$$PID = K_p + K_i T_s \frac{z}{z-1} + K_d \frac{N}{1 + N T_s \frac{z}{z-1}}.$$

5.5. The Proposed Control under Output Load and Converter Parameters Variations

For the widespread diffusion of a control technique in practical applications, robustness to parameter variation and model uncertainty is clearly a key feature. For this reason, we study the behavior of the output voltage under different load resistance variations. Experimental results of the step load testing under different output loads (10 Ω, 20 Ω and 30 Ω) are shown in Figure 15a. Notice that the output voltage presents an almost overlapping behavior in the three considered cases, confirming that the control is not affected by load variation in the range ±50% of the nominal value, see Table 1. Other experimental results on load variations from 20 Ω to 10 Ω and from 20 Ω to 30 Ω in steady-state condition are shown in Figure 15b. Notice that the proposed control system promptly stabilizes the voltage output with negligible undershoot and overshoot, thus providing a good performance in the case of load variations. Moreover, the set-point step response remains virtually the same irrespectively of the load resistance.

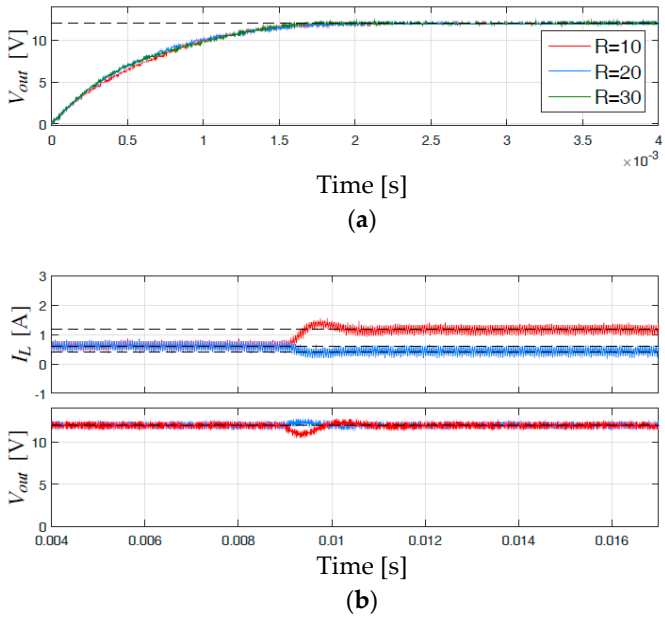

Figure 15. (**a**) Step responses under different output loads (10 Ω, 20 Ω and 30 Ω), (**b**) inductor current and output voltage under load variation from 20 Ω to 30 Ω (blue), and from 20 Ω to 10 Ω (red).

While load variations are due to normal operations of the buck converter, other parameters of the circuit of the converter, such as the inductance and capacitance, may vary as well as a result of the uncertainties affecting the production of the electrical components. In particular, the resonance frequency is directly related to the inductance and capacitance. In fact, since $R>>R_C$ and $R>>R_L$, in practice we have:

$$\omega_n \cong \frac{1}{\sqrt{LC}}.$$

Therefore, the inductor current and the output voltage under variations of the capacitor and inductor in the buck converter are also studied, and the results are shown in Figures 16 and 17. The proposed system delivers a good robust performance under parameter variations, and a monotonic response is obtained with all the considered combinations.

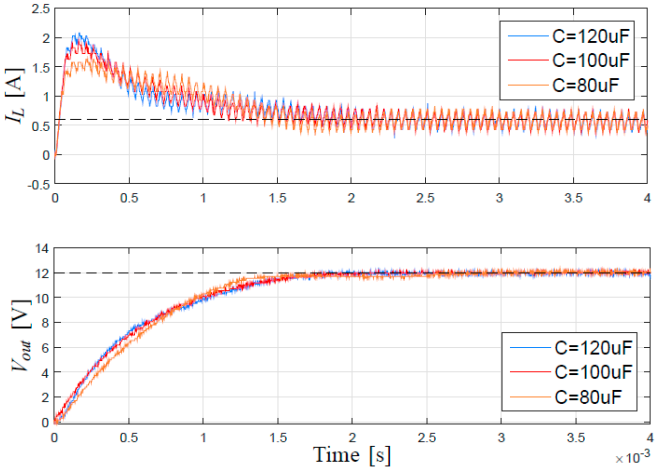

Figure 16. Step responses with the proposed design procedure when the model value of the capacitor is 100 μF−20%, 100 μF+20%.

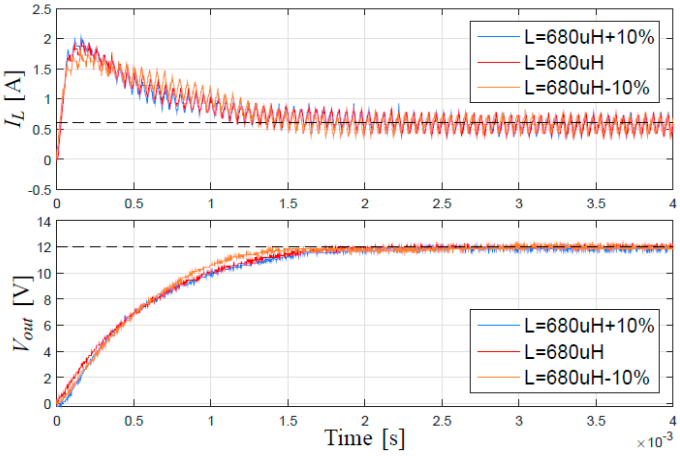

Figure 17. Step responses with the proposed design procedure when the model value of the inductor is set to 680 μH−10% and 680 μH+10%.

6. Conclusions

A new design framework for the control of buck converters has been presented in this paper. The proposed methodology is based on the discrete PIDF controller, and hinges on a direct design procedure that can be easily implemented in any non-specific platform. Indeed, the proposed methodology delivers a closed-form solution to meet suitable phase margin and gain crossover frequency values without a simulation environment. Moreover, the proposed design procedure and the discrete control algorithm are simple, they require small tuning times and they can be implemented by inexpensive microcontrollers.

Numerical and experimental verifications confirm that the proposed method goes well beyond the well-known zero/pole cancellation strategy and other control methods available in the literature. Indeed, the proposed approach enables the designer to assign an arbitrary bandwidth, which is therefore no longer constrained by the resonant peak. This aspect leads to a double benefit. On the one hand, this method avoids an excessively large bandwidth, which would result in noise/ripple amplification and ultimately in an increase in power consumption and a decrease in the component life. On the other hand, this method avoids the discretization problem that derives from discretizing a controller which assigns a bandwidth that is too large with respect to the sampling period. This, in particular, avoids detrimental effects on the stability margin due to the discretization. Moreover, experimental results confirm that the selection of large phase margin with the direct proposed method delivers a good system performance under load variations and plant uncertainties.

Author Contributions: S.C. provided for theoretical developments, simulations and experimental tests, L.N. and F.P. contributed to theoretical developments and manuscript organization, G.G. contributed to experimental results and manuscript finalization.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations and symbols are used in this manuscript:

ADC	Analog-to-digital converter
CCM	Continuous-conduction mode
DC	Direct current
DC-DC	Direct current to direct current
DPWM	Digital pulse width modulator
ESR	Equivalent series resistance
IIR	Infinite impulse response (digital filter)
IMC	Internal model control
PI	Proportional-integral (controller)
PID	Proportional-integral-derivative (controller)
PIDF	PID + filter
PWM	Pulse width modulation

List of Symbols

a_i, b_i	PIDF coefficients
C	Buck converter capacitance
$C(s)$, $C(z)$	Continuous and discrete-time controller transfer function
D	Control signal
E	Tracking error
$G(s)$, $G(z)$	Continuous and discrete-time plant model
G_m	Gain margin
H	Constant sensor gain
$H_0(s)$	Zero-order hold transfer function
i_L	Buck converter inductor current
K_d	Controller derivative gain
K_i	Controller integral gain
K_∞	High frequency controller gain
K_p	Controller proportional gain
L	Buck converter inductance
$L(s)$, $L(z)$	Continuous and discrete-time loop gain transfer function
N	Filter coefficient
R	Output load resistance
R_C	ESR of buck converter capacitor
R_L	ESR of buck converter inductor
r_n	Reference digital signal
T_d	Derivative time constant of the controller
T_f	Filter time constant of the controller
T_i	Integral time constant of the controller
T_s	Sampling period
u_0	Output signal of IIR filter
V_C	Buck converter capacitor voltage
V_{in}	Buck converter input voltage
V_{out}	Buck converter output voltage
V_{ref}	Reference voltage
y_n	Sampled output of the process

δ	Damping ratio of the controller zeros
Φ_m	Phase margin
ξ	Buck converter damping ratio
ω_g	Gain crossover frequency
ω_n	Buck converter natural frequency
$1/\tau$	Natural frequency of the controller zeros

References

1. Yang, J.; Wu, B.; Li, S.; Yu, X. Design and qualitative robustness analysis of an DOBC approach for DC-DC buck converters with unmatched circuit parameter perturbations. *IEEE Trans. Circuits Syst.* **2016**, *63*, 551–560. [CrossRef]
2. Liu, Y.F.; Meyer, E.; Liu, X. Recent developments in digital control strategies for DC/DC switching power converters. *IEEE Trans. Power Electron.* **2009**, *24*, 2567–2577.
3. Al-Hosani, K.; Malinin, A.; Utkin, V.I. Sliding mode PID control of buck converters. In Proceedings of the 2009 European Control Conference (ECC), Budapest, Hungary, 23–26 August 2009; pp. 2740–2744.
4. Roshan, Y.M.; Moallem, M. Control of nonminimum phase load current in a boost converter using output redefinition. *IEEE Trans. Power Electron.* **2014**, *29*, 5054–5062. [CrossRef]
5. Perry, A.G.; Feng, G.; Liu, Y.F.; Sen, P.C. A design method for PI-like fuzzy logic controllers for DC–DC converter. *IEEE Trans. Ind. Electron.* **2007**, *54*, 2688–2696. [CrossRef]
6. Mantz, R.J.; Tacconi, E.J. Complementary rules to Ziegler and Nichols' rules for a regulating and tracking controller. *Int. J. Control* **1989**, *49*, 1465–1471. [CrossRef]
7. Son, Y.I.; Kim, I.H. Complementary PID controller to passivity-based nonlinear control of boost converters with inductor resistance. *IEEE Trans. Control Syst. Technol.* **2012**, *20*, 826–834. [CrossRef]
8. Kapat, S.; Krein, P.T. Formulation of PID control for DC–DC converters based on capacitor current: A geometric context. *IEEE Trans. Power Electron.* **2012**, *27*, 1424–1432. [CrossRef]
9. Sira-Ramirez, H. Nonlinear PI controller design for switchmode DC-to-DC power converters. *IEEE Trans. Circuits Syst.* **1991**, *38*, 410–417. [CrossRef]
10. Guo, L.; Hung, J.Y.; Nelms, R.M. PID controller modifications to improve steady-state performance of digital controllers for buck and boost converters. In Proceedings of the Seventeenth Annual IEEE Applied Power Electronics Conference and Exposition, Dallas, TX, USA, 10–14 March 2002; Volume 1, pp. 381–388.
11. Guo, L.; Hung, J.Y.; Nelms, R.M. Digital controller design for buck and boost converters using root locus techniques. In Proceedings of the IECON 2003 29th Annual Conference of the IEEE Industrial Electronics Society, Roanoke, VA, USA, 2–6 November 2003; Volume 2, pp. 1864–1869.
12. Guo, L.; Hung, J.Y.; Nelms, R.M. Evaluation of DSP-based PID and fuzzy controllers for DC–DC converters. *IEEE Trans. Ind. Electron.* **2009**, *56*, 2237–2248.
13. Marro, G.; Zanasi, R. New formulae and graphics for compensator design. In Proceedings of the 1998 IEEE International Conference on Control Applications, Trieste, Italy, 4 September 1998; Volume 1, pp. 129–133.
14. Zanasi, R.; Cuoghi, S.; Ntogramatzidis, L. Analytical and graphical design of lead–lag compensators. *Int. J. Control* **2011**, *84*, 1830–1846. [CrossRef]
15. Ntogramatzidis, L.; Ferrante, A. Exact tuning of PID controllers in control feedback design. *IET Control Theory Appl.* **2011**, *5*, 565–578. [CrossRef]
16. Zanasi, R.; Cuoghi, S. Direct methods for the synthesis of PID compensators: Analytical and graphical design. In Proceedings of the IECON 2011 37th Annual Conference of the IEEE Industrial Electronics Society, Melbourne, Australia, 7–10 November 2011; pp. 552–557.
17. Cuoghi, S.; Ntogramatzidis, L. Direct and exact methods for the synthesis of discrete-time proportional–integral–derivative controllers. *IET Control Theory Appl.* **2013**, *7*, 2164–2171. [CrossRef]
18. Cuoghi, S.; Ntogramatzidis, L. New inversion formulae for PIDF controllers with complex zeros for DC-DC buck converter. In Proceedings of the IECON 2011 42th Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 235–240.
19. Kristiansson, B.; Lennartson, B. Robust tuning of PI and PID controllers: Using derivative action despite sensor noise. *IEEE Control Syst. Mag.* **2006**, *26*, 55–69.
20. Dorf, R.C.; Bishop, R.H. *Modern Control Systems*; Pearson: Upper Saddle River, NJ, USA, 2011.

21. Severns, R.P.; Bloom, G. *Modern DC-to-DC Switchmode Power Converter Circuits*; Springer: New York, NY, USA, 1985.
22. Abbas, G.; Abouchi, N.; Pillonnet, G. Optimal state-space controller for power switching converter. In Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems, Kuala Lumpur, Malaysia, 6–9 December 2010; pp. 867–870.
23. Panda, R.C.; Yu, C.C.; Huang, H. PID tuning rules for SOPDT systems: Review and some new results. *ISA Trans.* **2004**, *43*, 283–295. [CrossRef]
24. Visioli, A. *Practical PID Control*; Springer: London, UK, 2006.
25. Mohan, N.; Undeland, T.M. *Power Electronics: Converters, Applications, and Design*; John Wiley and Sons: Hoboken, NJ, USA, 2007.