

Article

DSP-HIL Comparison between IM Drive Control Strategies

Luis E. Ortega-García ^{1,†}, Daniela Rodriguez-Sotelo ^{1,†}, Jose C. Nuñez-Perez ², Yuma Sandoval-Ibarra ³
and Francisco J. Perez-Pinal ^{1,*,†} 

¹ Instituto Tecnológico de Celaya, Tecnológico Nacional de México, Celaya 38010, Guanajuato, Mexico; m1803049@itcelaya.edu.mx (L.E.O.-G.); D1903002@itcelaya.edu.mx (D.R.-S.)

² Centro de Investigación y Desarrollo de Tecnología Digital, Instituto Politécnico Nacional, Tijuana 22435, Baja California, Mexico; nunez@citedi.mx

³ Departamento de Posgrado, Universidad Politécnica de Lázaro Cárdenas (UPLC), Lázaro Cárdenas 60950, Michoacán, Mexico; yumasandoval@uplc.edu.mx

* Correspondence: francisco.perez@itcelaya.edu.mx

† These authors contributed equally to this work.

Abstract: Due to their high robustness and simple maintenance, induction motors (IM) are commonly applied in household appliances and industry. Recently, advanced control techniques are being applied to traditional controllers such as field-oriented control (FOC) and torque control (DTC). Dynamic performance improvement, hardware simplification and software resource reduction are some of the characteristics reported by these advanced techniques, where a comparison of the new proposal with a traditional structure is generally reported for its validation. However, an assessment between advanced techniques is usually missing. Therefore, we evaluated the traditional FOC and DTC with two additional advanced control modifications, fuzzy and predictive. The resulting six structures were numerically evaluated using MATLAB SIMULINK in a 5 HP four-pole three-phase IM and practically validated using hardware-in-the-loop (Typhoon HIL 402 and DSP TMS320F28035). Speed, torque, phase current and flux response are reported for the six controllers and practical insights are summarized.

Keywords: automatic control; induction motors; variable speed drives



Citation: Ortega-García, L.E.; Rodriguez-Sotelo, D.; Nuñez-Perez, J.C.; Sandoval-Ibarra, Y.; Perez-Pinal, F.J. DSP-HIL Comparison between IM Drive Control Strategies. *Electronics* **2021**, *10*, 921. <https://doi.org/10.3390/electronics10080921>

Academic Editor: Ki-Bum Park

Received: 6 February 2021

Accepted: 8 April 2021

Published: 13 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Nowadays, at least 90% of industrial systems use electric machines [1]. They are commonly used in pumps, ventilators, mills, elevators, electric vehicles, etc. In particular, induction motors (IM) are robust, trustworthy and can be employed in hazardous environments [2]. Different control techniques exist for IM, the most representative being voltage/frequency (V/f), field-oriented control (FOC) and direct torque control (DTC).

The V/f method provides an easy and functional way to control the IM's velocity, which allows a satisfactory response in steady state; however, its behavior in transient state is not ideal [3]. FOC is one of the most popular algorithms used, and there are two types: direct FOC (DFOC) and indirect FOC (IFOC). In DFOC, the magnetic flux value is measured inside the motor using at least two flux sensors. In IFOC, the rotor flux angle is observed/estimated through stator current measurements. The disadvantages of these methods are sensor fragility and sensitiveness to parameter variation, respectively [4,5]. On the other hand, DTC has gained popularity due to simple structure and performance in high torque applications, but its main limitation is the excessive torque ripple due to current's hysteresis controllers [6].

With new technological trends, recent techniques that improve the behavior of IM have been integrated into FOC and DTC. One of them is fuzzy logic (FL); this is a method that allows approximately replicating the process of human reasoning based on their previous knowledge and experience [7]. In this technique, the overall membership functions relate human thought to the FL algorithm in a mathematical way [8]. The general FL algorithm

is based on the following procedures: fuzzification, fuzzy rules (or decision making), interference methods and defuzzification.

Another technique that has grown in importance is the model of predictive control (MPC), principally in engineering applications as a consequent of high-speed processors [9]. This technique was developed in the 1960s and has been applied to industrial processes such as chemical plants and oil refineries [10]. MPC features the following: the use of dynamic model system with high accuracy [11], fast-dynamic response, multivariable control and the inclusion of non-linear constraints [9]. MPC has a prediction horizon, i.e., the steps of future sampling based on the current one, which involves a high load of computational calculations [12]. An optimization function is used in MPC to evaluate different inputs, the present sampling, and the input that minimize the error to be used in the next sampling [13]. Likewise, the proposed function has a weighting factor that is tuned by a heuristic process [14].

FL and MPC are usually compared to a traditional FOC or DTC structure for performance validation. However, in the literature, an assessment between advanced techniques is usually lacking; and many of those reported evaluations are validated only by using numerical software or real-time hardware, which, in the authors' opinion, limits the reported conclusions. After a deep study of the literature, it was noticed that work is focused on two different paths: (1) comparing IM controllers, which mainly use real-time controllers to implement the different control techniques; and (2) developing HIL test beds to validate power electronic converter applications.

For instance, a comparison between indirect FOC and DTC for IM was reported by Hemavathy et al. [15]. Here, speed tracking, dynamic and load disturbance response are reported. Unfortunately, only software implementation is given. In [16], a comparison of FOC, DTC, predictive torque control (PTC) and predictive current control (PCC) was experimentally performed in 2.2 kW squirrel-cage IM. The IM was driven by a modified SERVOSTAR620 14 kVA inverter (Radford, VT, USA) and controlled by a 1.4 GHz real-time controller, which provided full control of the IGBT gates. The authors concluded that overall the strategies had good functioning in the entire speed range with or without load. However, the implementation of different control strategies in a Field-Programmable Gate Array (FPGA), digital signal processor (DSP) or advanced microcontroller was lacking, which limits the overall conclusions. Different control and optimization techniques for IM were reviewed by Hannan et al. [17]. They reviewed the literature on conventional scalar and vector controls, focusing on biology- and physics-based optimization algorithms. However, only a qualitative description and comparison is reported. Indeed, it is reported that only genetic, lighting and quay lighting optimization algorithm are practically implemented in a DSP, and the others have only been simulated or executed in real-time controllers.

A comparison between DTC and PTC was reported by Karlovsky and Lettl [18]. They presented numerical and experimental results. This paper reports that a smaller ripple is observed in torque waveforms using PTC compared with traditional DTC. Unfortunately, the implementation of both algorithms was carried out using a real-time controller (dSPACE ds1103 system), which limited the overall conclusions. An interesting work was reported by Nael and Todd [19]. They proposed an integral-proportional (IP) controller instead of a classical proportional-integral (PI) one. The proposed structure is placed in the current and speed controller of a traditional FOC squirrel cage IM. The system consists of a Typhoon HIL, a Typhoon docking station and a TI microcontroller (TMS320F28335). The PI and IP control structures were both discretized and coded. The authors reported that IP improved speed tracking without affecting the disturbance rejection. Unfortunately, the implementation of IP in other controllers' topologies was not performed, and details about the DSP implementation are absent. A qualitatively comparison of adjustable-speed drives (ADS) for shaft powers of 20 MW or more was reported by Rauber and den Bakker [20]. The comparison was done between voltage source inverter (VSI) and load-commutated inverters (LSI). The authors concluded that LSI topology has the highest reliability, while

VSI-based drives have flexibility from the system integration point of view. However, no comparison of control techniques applied in the ADS is reported.

The second research effort is focused on PEC's HIL applications, development and reliability studies.

For instance, stability of different power hardware-in-the-loop (PHIL) configurations was reported by Lauss and Strunz [21]. They presented transfer functions for the entire PHIL and quantified the overall involved time delays, which enhanced simulation stability and precision. A multi-rate co-simulation architecture that integrates PHIL, a real-time quasi-static time-series, and a reduced-equivalent electromagnetic transient of the distribution feeder in a large distribution network was reported by Prabakar et al. [22]. This co-simulation was tested in an interconnected 500-kVA advanced photovoltaic (PV) inverter in PHIL architecture. A custom real-time HIL emulator to test transients and dynamic behavior of power electronics converters was reported by Iranian et al. [23]. A MATLAB/Simulink hardware description language (HDL) and LabVIEW environment was used to model, control and test a 2.5 MW doubly-feed induction generator.

Validation in a HIL 402 real-time emulator of a novel cubic converter was reported by Ahmad et al. [24]. The proposed converter was compared with conventional boost and quadratic boost converter in a maximum power point tracking (MPPT) for a Solar PV array. Operation modes, component sizing, power losses, efficiency and thermal modeling was reported. Unfortunately, the control stage was missed. A real-time silicon carbide IGBT model based on the Wiener–Hammerstein configuration was reported by Liang et al. [25]. A Beijing–Shanghai AC traction application was used as a testbed and implemented on the hybrid multiprocessor system-on-chip (MPSoC) and field-programmable gate array (FPGA) platform. Comparisons with PSCAD/EMTDC and Saber RD was done to verify system and device level performance. A scalable modeling approach, which relies on multi-objective targets between model functionalities, accuracy and execution time was reported by Chakraborty et al. [26]. A dSPACE SCALEXIO HIL test bench was used for real-time (RT) model implementation for the EV's HV DC/DC converter. Additionally, passive component losses were verified using Finite Element Method (FEM) software.

An example of a high bandwidth motor emulator was reported by Luo et al. [27]. The system comprises a 2 kV, 50 A silicon carbide MOSFET six-pack full bridge module from Wolfspeed (CCS050M12CM2), inductive coupling, a reconfigurable real-time model for an interior permanent magnet machine implemented in FPGA and a hybrid predictive control method. The motor model and PWM pattern were implemented in an Avnet PicoZed, which contains a Xilinx ZYNQ-7000 (XC7Z015-1SBG485) system on chip (SoC). A systematic methodology for HIL power electronic converters by using LabVIEW software was reported by Estrada et al. [28]. Five simple steps are proposed for HIL converters: design, modeling, solving the model using a numerical method, programming an off-line simulation of the model using fixed-point representation and implementing the solution of the model in FPGA. A buck converter and a three-phase voltage source inverter are implemented and compared with the simulation of commercial software (PSIM® v9.0) and real power converters. Finally, a new model-free controller to ameliorate the load frequency control performance of a shipboard micro grid was reported by Khooban et al. [29], who used an OPAL-RT HIL test bed to validate the proposed controller.

We investigated the following IM controllers: FOC, Fuzzy FOC, Predictive Current Control, DTC, Fuzzy DTC and Predictive Torque Control (Figure 1). The goals of this investigation were to experimentally compare for the first time, to the best of the authors' knowledge, traditional and advance controllers on HIL and DSP qualitatively and quantitatively. Thus, the following contributions to the state of the art are hereby emphasized:

- HIL Typhoon 402 was used to integrate the fidelity of the physical simulation and the flexibility of numerical simulations. It emulated the induction machine, inverter and sensors. Performance of the overall *.dll strategies were compared with the MATLAB SIMULINK and a good agreement among them was achieved.

- The six controllers were fully implemented into a DSP TMS320F28035 considering practical implementation issues. It was found that sampling time, controller gain discretization, variable type selection and memory allocation are the parameters that must be solved to achieve a high-performance variable speed drive. The DSP implementations on the HIL board was in consonance with the MATLAB simulation.
- It was concluded that predictive current control is computationally simple, it has no practical complexity and it achieves a higher performance compared with the other approaches. Indeed, in the author's experience, the use of real time controller limits the conclusion previously reported, because real life control challenges are completely removed.

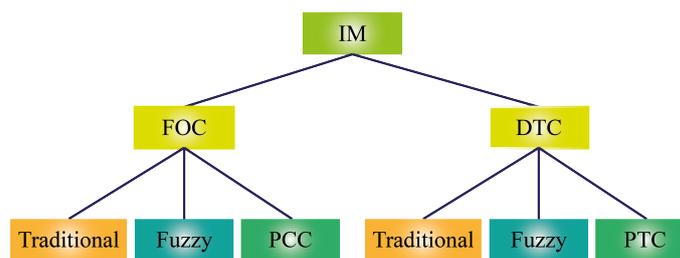


Figure 1. IM control techniques.

The paper is organized as follows. Section 2 presents the different variants and their control and tuning. The experimental results of different configurations, such as their qualitative and quantitative comparison, are described in Section 3. Finally, the most important conclusions of this work are detailed in Section 4. The work in this paper corroborates the elasticity of digital controllers and confirms the control execution can alter the assortment of drive sections.

2. Control Techniques

2.1. Field Oriented Control (FOC)

This technique is used in IM, and its purpose is to emulate the behavior of DC motors. In DC motors, the variables that produce the electric torque, flux and armature current (stator) are uncoupled, which does not occur in IM. For this reason, the equations that determine the behavior of IM are modified to a new model with a coordinate transformation. New coordinates are in a plane that is mounted to the rotor flux, that is, it is aligned with the axis d , of a complex plane dq , and turns at the same speed as the rotor flux.

This transformation brings some benefits. IM uses three-phase currents of alternating current (AC); once they are transformed, only two signals of DC are created, i_{sd} and i_{sq} , which can be under control without any complication. Usually, the rotor flux is aligned to one axis; analytically, instead of two components, only one axis is created (Equations (1) and (2)). In general, there are two kind of implementations, direct and indirect FOC, the latter was used here (Figure 2) (please refer to [1] and the references therein for more information about FOC).

$$\tilde{\Psi}_{rdq} = \Psi_{rd} + j0 = |\tilde{\Psi}_r| = L_m i_{sd} \quad (1)$$

$$V_{rdq} = i_{rqd} R_r + \frac{d}{dt} \Psi_{rdq} - j\omega_r \Psi_{rdq} \quad (2)$$

where i_{sd} and i_{sq} are the real and imaginary parts of the stator current component, respectively; Ψ_{rd} is the real rotor flux component; Ψ_{rdq} is the rotor flux; the rotor current is i_{rdq} ; and the rotor voltage is denoted by V_{rdq} . All these variables are based on the excitation framework, while the rotor speed is represented as ω_r .

measured and transformed, are used to estimate the flux magnitude and its position on a fictional plane placed on the stator (Equations (4) and (5)). Indeed, the torque calculus is performed in (6). The estimation results are fed back for the flux and the torque control application. As in FOC, the actual/real value of torque is generated by a PI controller, where the error of rotor speed is the input.

$$\vec{V}_s = R_s \vec{i}_s + \frac{d}{dt}(\vec{\Psi}_s) + j(\omega_g - \omega_s) \vec{\Psi}_s \tag{4}$$

$$\theta = \arctan\left(\frac{\Psi_{s\beta}}{\Psi_{s\alpha}}\right) \tag{5}$$

$$\tau_e = \frac{2}{3} P_p \text{Im} \{ \vec{\Psi}_s \times \vec{i}_s \} \tag{6}$$

In these equations, τ_e is the electromagnetic torque, P_p is the pair of poles and $\Psi_{s\alpha}$ and $\Psi_{s\beta}$ are the real and imaginary stator fluxes on a complex plane, respectively.

In this control scheme, the flux and torque error are inputs of the hysteresis band that limited the signals' values; in other words, errors are inside of an acceptable range. Thus, DTC has more ripple in the flux and torque in comparison to FOC. After these signals lead to a voltage vector selection table, in which the inputs are the estimated flux position and the outputs are hysteresis bands according to its value, a voltage vector is applied (Figure 3).

2.5. Fuzzy DTC (FTC)

It is the same idea as DTC with a combination of FL. The implementation is easier compared with Fuzzy FOC because the hysteresis band and the voltage vector selection table are discrete and bounded. Once again, with more membership functions, the controller response will be more accurate. Some differences between DTC and DTC fuzzy are the latter substitutes the hysteresis bands, and the voltage vector selection table uses a fuzzy analysis. In other words, the fuzzy block inputs are the error of torque, flux error and flux position. The output of the same block is six functions *singleton* that represents the voltage vector index (0–7) in the case of the two-level inverter (Figure 4).

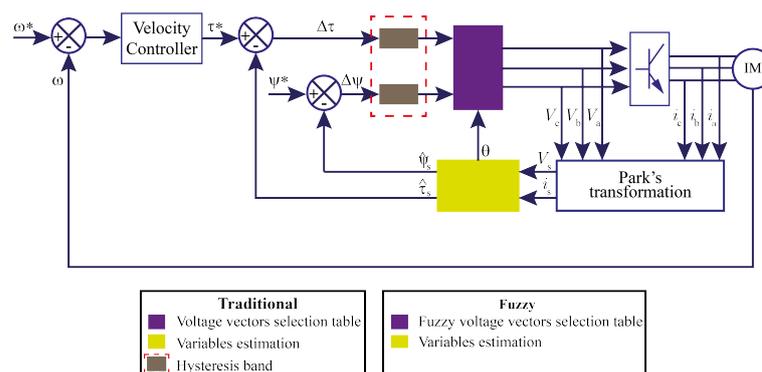


Figure 4. Block diagram of traditional and fuzzy DTC.

2.6. Predictive Torque Control (PTC)

It is a combination of DTC and predictive philosophy. PTC uses the rotor flux estimations and torque, adding only the rotor flux that is important for the flux and torque predictions. Because these are the variables to control, the torque is a consequence of the flux and current of the stator, as shown (7)–(10). It can be observed that $\Psi_s(k + 1)$ and $\tau_e(k + 1)$ are the system feedback. It is interesting to note that, to predict a variable, it is necessary to evaluate the possible system states. In this case, there are eight possible voltage vectors for a two-level inverter, and, consequently, there are eight possible values of $\Psi_s(k + 1)$ as well as for $\tau_e(k + 1)$. These are compared with their corresponding refer-

ences, which will originate eight errors for each variable, and they will be evaluated by an optimization algorithm. The hysteresis band and the selection table are replaced by this algorithm, which is only an objective function (11). Therefore, this objective function will derive eight possible values, and the result with the minimum value (minimization action) represents the voltage vector that will be applied. The block diagram of this control can be observed in Figure 5.

$$\Psi_s(k + 1) = \Psi_s(k) + T_s V_s(n) - T_s R_s I_s(k) \tag{7}$$

$$\Psi_r(k) = \frac{L_r}{L_m} \Psi_s(k) + I_s(k) \left(L_m - \frac{L_r L_s}{L_m} \right) \tag{8}$$

$$I_s(k + 1) = \left(1 + \frac{T_s}{\tau_\sigma} I_s(k) \right) + \frac{T_s}{\tau_\sigma T_s} \left\{ \frac{1}{R_\sigma} \left[\left(\frac{k_r}{\tau_r} - k_r j\omega \right) \Psi_r(k) + V_s(n(k + 1)) \right] \right\} \tag{9}$$

$$r_\sigma = R_s + R_r k_r^2; \tau_r = \frac{L_r}{R_r}; k_s = \frac{L_m}{L_r}; \sigma = 1 - k_r k_s; \tau_\sigma = \frac{\sigma L_s}{r_\sigma}$$

$$\tau_e = \frac{3}{2} P_p (\Psi_{s\alpha} i_{s\beta} - \Psi_{s\beta} i_{s\alpha}) \tag{10}$$

$$g = |\tau_e^*(k) - \tau_e(k + 1)| + \lambda |\Psi_s^*(k) - \Psi_s(k + 1)| \tag{11}$$

$i_{s\alpha}^*$ and $i_{s\beta}^*$ are the real and imaginary reference stator current components on a complex plane, respectively; T_s is the sampling time; τ_σ is the time constant of the linkage factor; R_σ is the resistance of linkage factor; k_r is the relation of mutual-inductance with rotor inductance; τ_r is the rotor constant time; λ is the weighting factor; τ_e^* is the electromagnetic torque reference; Ψ_s^* is the flux stator reference; σ is the linkage factor; and $i_{s\alpha}$ and $i_{s\beta}$ are the real and imaginary stator current components on a complex plane, respectively.

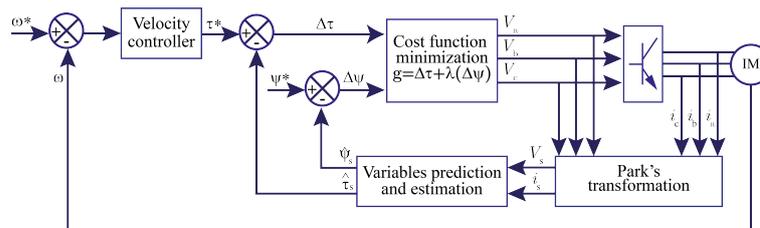


Figure 5. Block diagram of PTC.

3. Experimental Results

This section reports the step-by-step process used to implement the several variable speed IM strategies described above. Initially, a detailed software tuning process is given, then features of the DSP-HIL test bed are reported and, finally, DSP-HIL and SIMULINK comparative results are shown. The real-time HIL system simulated in real-time the power and motor hardware, and the converter and motor under test were controlled using physical hardware, in this case, a Texas Instrument DSP TMS320F28035.

3.1. Software Tuning Process

The six variable speed strategies described above were simulated in MATLAB SIMULINK and Virtual HIL Device (a software-based emulator of the Typhoon HIL 402 real-time device). Numerical implementations were carried out in three stages. Initially, each controller was implemented by using standard SIMULINK Simscape Electrical libraries and standard block functions.

Figure 6 shows the steps to compare the overall *.dll strategies with the MATLAB SIMULINK and MATLAB SIMULINK C functions, where a good agreement between them was achieved. A control panel is also accessible to define the simulation parameters and display the system's performance by using a virtual oscilloscope and monitoring the system's evolution.

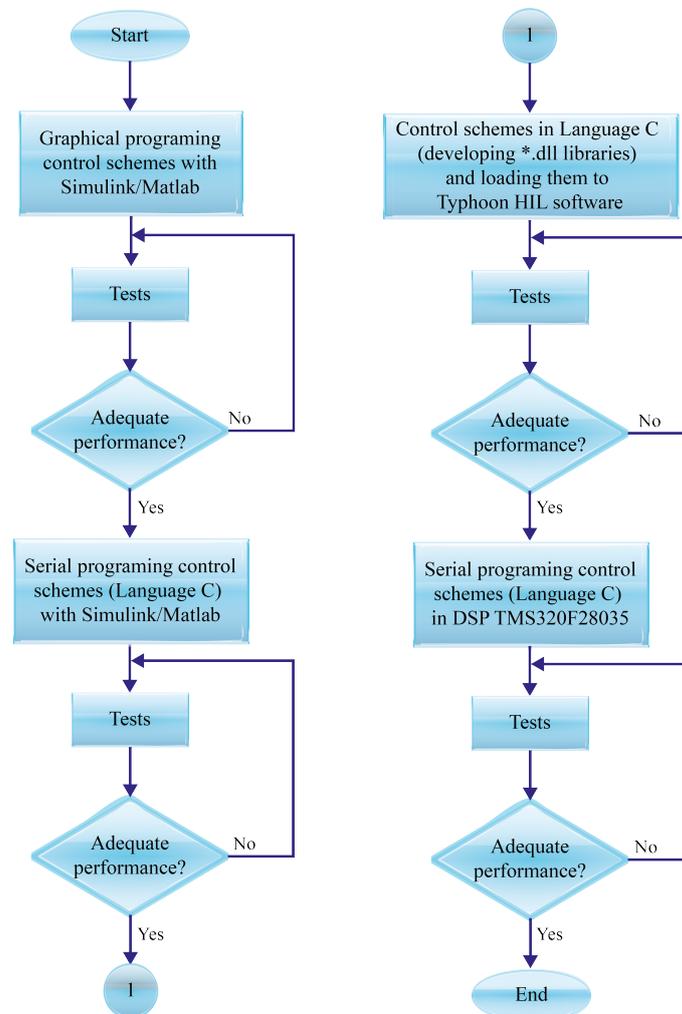


Figure 6. Implementation processes flow diagram.

This repetitive process allowed debugging the variable speed controllers in each step and incrementing the practical systems' reliability. The six variable speed strategies with SIMULINK have C functions and the *.dll libraries on the Virtual HIL Typhoon software are not reported due to space issues. The induction motor and controller parameters are shown in Appendix A.

The drive consisted of a sinusoidal PWM controlled voltage source inverter (VSI) feeding a squirrel cage three-phase induction machine. The DC-link of the inverter was connected to an ideal DC voltage source, and the stator three-phase voltages were generated by the PWM VSI. The VSI consisted of a classical three-leg configuration, one per phase, with two switches with antiparallel diode each, and only one switch per leg was turned on at any time. The switches were controlled by sinusoidal PWM phase shifted by 120° for each phase, generated by a SIMULINK Simscape block.

3.2. HIL-DSP Test Bed Considerations

After the *.dll libraries were verified on simulations, the next step was to migrate those to C, so the DSP could execute them. A backward Euler method was selected to implement

the overall digital controllers. It is worth mentioning that some control techniques require smaller sampling time for execution, for instance IFOC (16 μ s) or DTC (15 μ s). Therefore, appropriate sampling time and discrete controller gains were calculated.

For the practical implementation, a HIL DSP 100 conditioning card was used between Typhoon HIL 402 and DSP TMS320F28035 (Figure 7). The Typhoon HIL 402 handles analog outputs up to ± 10 V, so the HIL DSP 100 Interface card conditioned it to DSP voltage levels of 0–3 V. It is necessary to mention that the Typhoon HIL 402 ADC's output scaling signal can be independently adjusted between output channels. Indeed, the Typhoon HIL 402 emulates the IM dynamic and sends three-phase voltages, three-phase currents and a reference and measured speed to eight DPSs' analog-to-digital converter inputs, ADCINA1–A4 and ADCINB1–B4, respectively. Once the DSP executes the applicable controller technique, it sends back the PWM switching signals to the HIL's digital inputs 1–6.

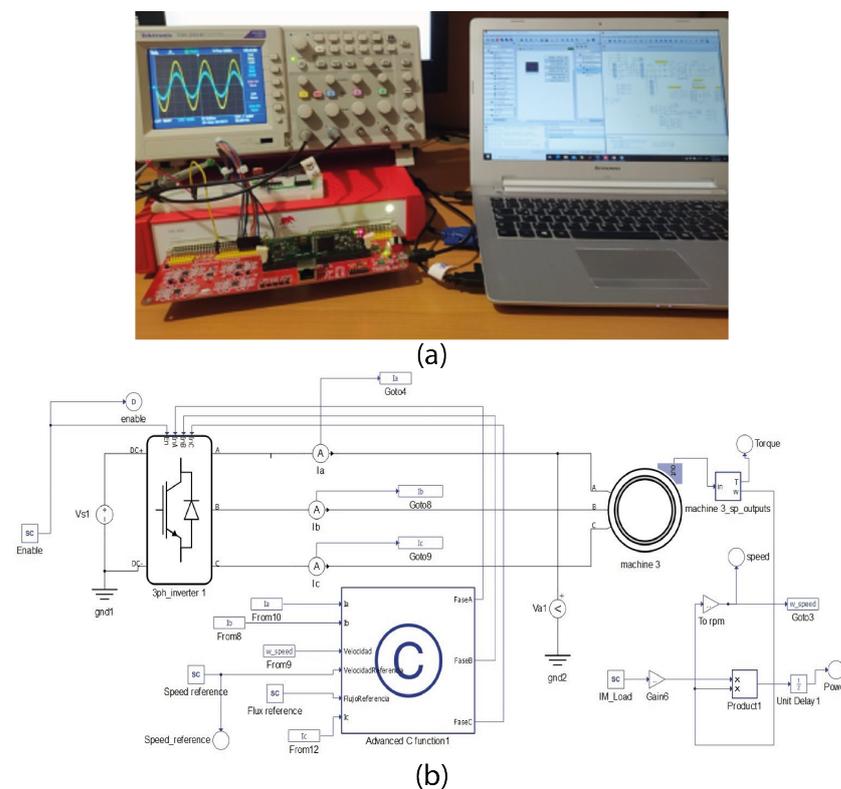


Figure 7. Interface DSP: (a) DSP 100 and Typhoon HIL 402; and (b) Typhoon HIL schematic editor.

Another fundamental parameter for the experimental implementation was the selection of variable type. In the numerical simulations, a float variable was used because the programmed sampling time is considered in the numerical results and there is no time delay in the mathematical operations due to the type of variable. In contrast, the DSP process must be accomplished within the sampling time. Thus, even though the float variables offer better resolution in the DSP TMS320F28035, they take a long time to execute, so a Q-format fixed point format was used for the overall experimental tests. Indeed, these fixed-point mathematical operations were executed in the DSP RAM's memory instead of FLASH, since they require fewer clock cycles to execute.

3.3. Practical Validation

Figure 8 shows the results of the MATLAB SIMULIK block functions and DSP-HIL of traditional IFOC. Figure 8a shows that IFOC has a good speed response for both systems during step speed transitions from clockwise to counterclockwise speed directions. The highest overshoot is achieved during a transition of 100 rad/s in 2 s. A load change of -10 Nm was applied in 9 s and removed in 10 s; as can be seen, a small overshoot is

reflected in the speed. A speed zoom is also shown in the same figure; it can be noticed that a neglectable speed variation between reference and practical responses was found, and both systems follow the overall reference signal.

Figure 8b shows the torque response for the same speed reference. At 2 s, the highest step load variation and difference between SIMULINK and DSP-HIL responses can be noticed. Despite the great difference, the torque performance is not as severely affected. This last is a notorious inaccuracy caused by noise observed in the torque monitoring signal from the Typhoon HIL 402. Meanwhile, torque exhibits similar behavior on both simulation platforms, as shown in Figure 8b (zoom). Figure 8c,d shows the stator current phase A and stator flux, respectively. A good agreement between both platforms can be seen, with a slight time delay caused by the sampling and DSP delay time.

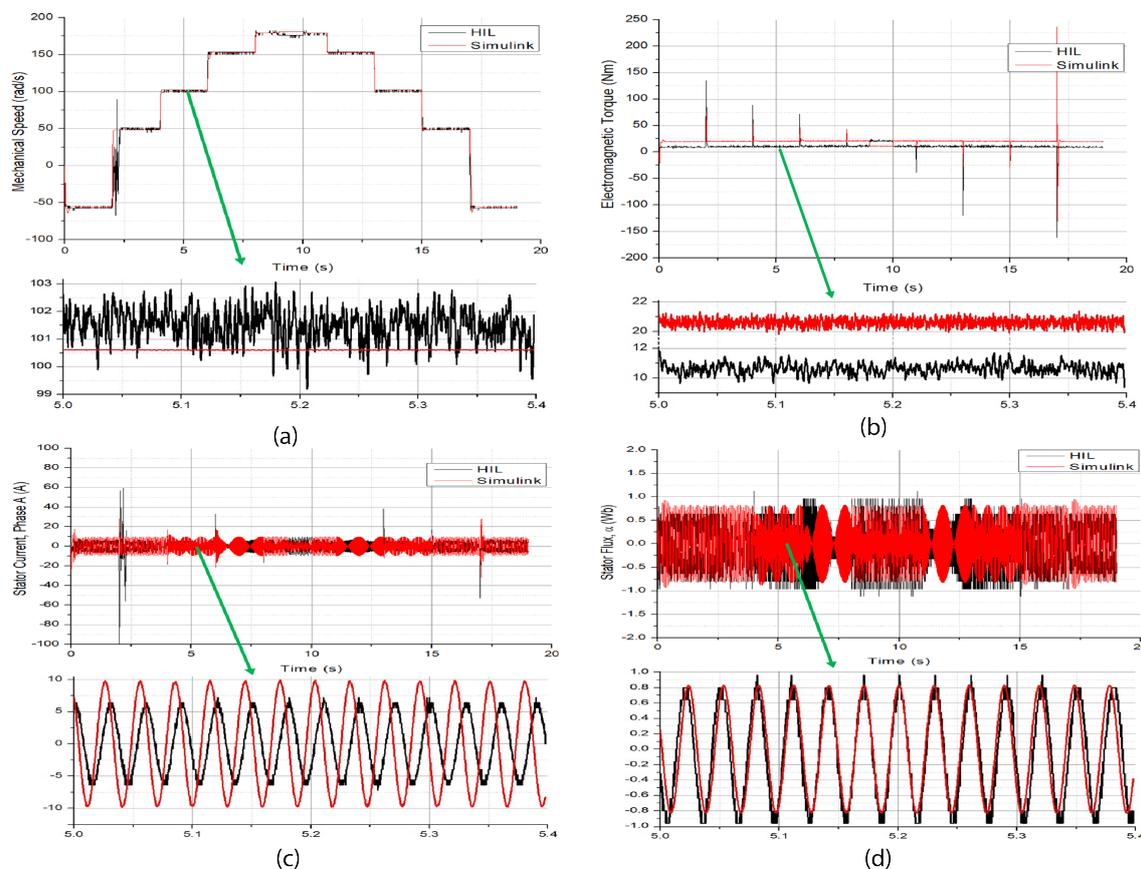


Figure 8. Numerical and practical results of the IFOC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

Figure 9 shows the results of the MATLAB SIMULINK block functions and DSP-HIL of DIFOC. Figure 9a shows that DIFOC has a good speed response for both systems during step speed transitions from clockwise to counterclockwise speed directions. The highest speed error tracking is achieved during a transition of 100 rad/s in 2 s. A load change of -10 Nm was applied in 9 s and removed in 10 s; as can be seen, no overshoot is reflected in the speed. A speed zoom is also shown in the same figure. A neglectable speed variation between reference and practical responses was found, and both systems follow the overall reference signal.

Figure 9b shows the torque response for the same speed reference. At 17 s, the highest step load variation and difference between SIMULINK and DSP-HIL responses can be noticed. Despite the notorious great difference, the torque performance is not as severely affected. This last once again is a mismatch caused by noise observed in the torque monitoring signal. Figure 9c,d shows the stator current phase A and stator flux,

respectively. A poor agreement between both platforms can be noticed. Indeed, the current in steady state is bigger in the SIMULINK compared to the DS-HIL response, with a slight time delay caused by the sampling and DSP processing time.

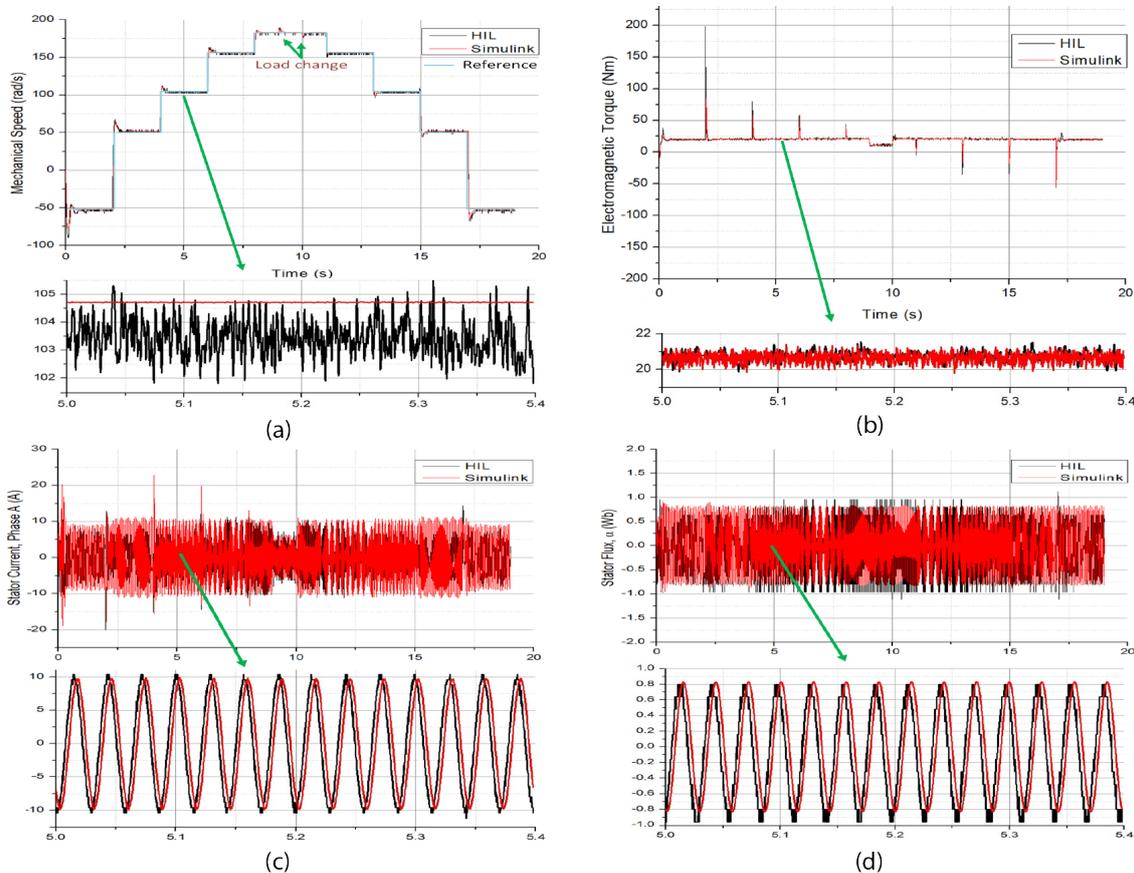


Figure 9. Numerical and practical results of the DIFOC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

Figure 10 shows the results of the MATLAB SIMULINK block functions and DSP-HIL of PCC. Figure 10a shows that PCC has a good speed response for both systems during step speed transitions from clockwise to counterclockwise speed directions. Similar to the previous controllers, the highest speed error tracking is achieved during a transition of 100 rad/s in 2 s. A load change of -10 Nm was applied in 9 s and removed in 10 s. The system is able to come back to the reference in less than 1 s. A speed zoom is also shown in the same figure. A low speed variation between reference and practical responses was found, and both systems follow the desired signal.

Figure 10b shows the torque response for the same speed reference. At 2 s, the highest step load variation and difference between SIMULINK and DSP-HIL responses can be noticed. Despite the notorious difference, the torque performance follows a similar behavior. Figure 10c,d shows the stator current phase A and stator flux, respectively. There is a modest agreement between both platforms. Indeed, the current in steady state is bigger in the SIMULINK compared to the DS-HIL response, with a slight time delay caused by the DSP's sampling processing time.

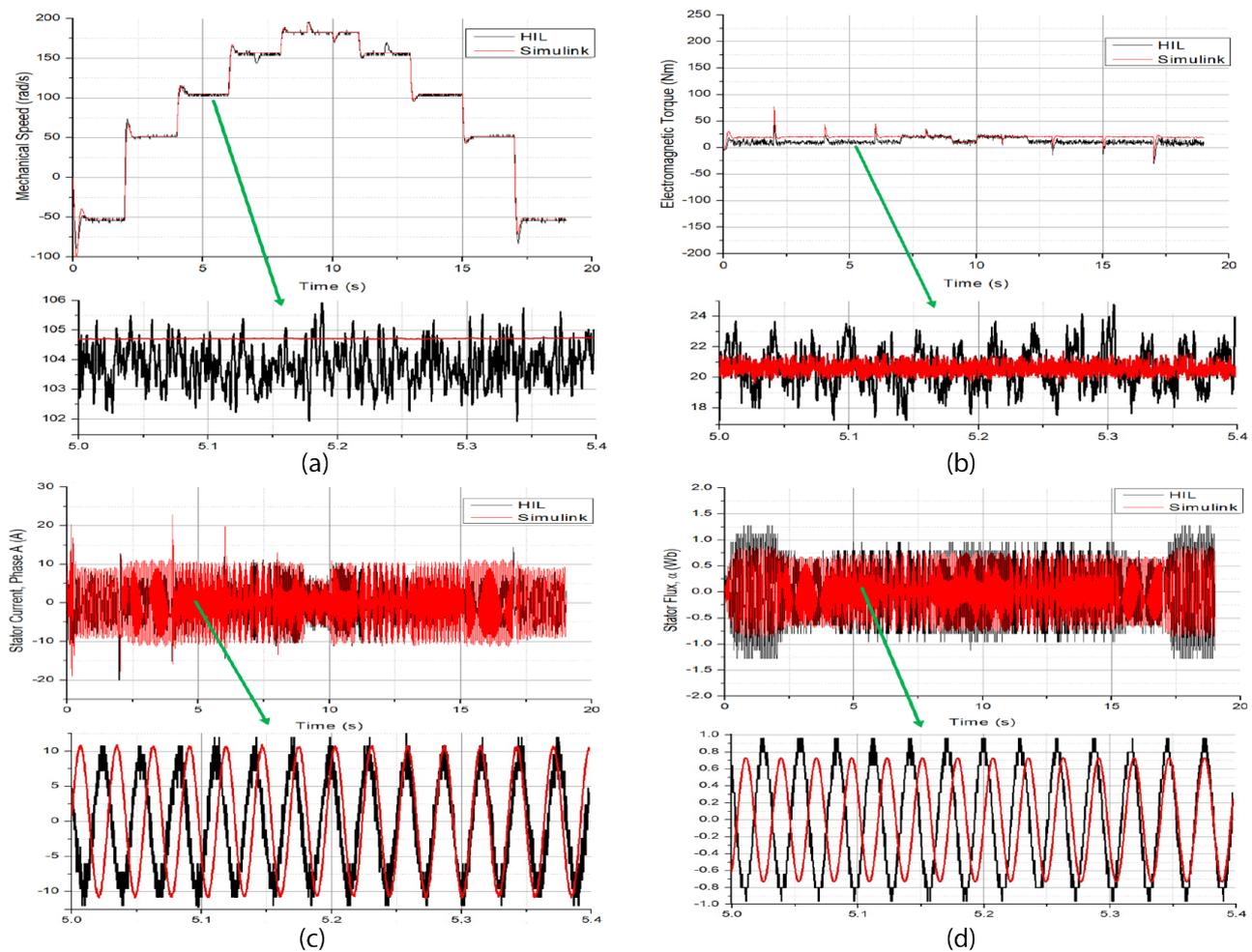


Figure 10. Numerical and practical results of the PCC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

Figure 11 shows the results of the MATLAB SIMULINK block functions and DSP-HIL of DTC. Figure 11a shows that DTC has a good speed response for both systems during step speed transitions from clockwise to counterclockwise speed directions. Similar to the previous controllers, the highest speed error tracking is achieved during a transition of 100 rad/s in 2 s. A load change of -10 Nm was applied in 9 s and removed in 10 s. The system is able to come back to the reference in less than 1 s. A speed zoom is also shown in the same figure. A low speed variation between reference and practical responses was found, and both systems follow the desired signal.

Figure 11b shows the torque response for the same speed reference. At 2 s, the highest step load variation and difference between SIMULINK and DSP-HIL responses can be noticed. Despite the notorious difference, the torque performance follows a similar behavior, and the DSP-HIL error, twice the expected one, is due to numerical inaccuracy. Figure 11c,d shows the stator current phase A and stator flux, respectively. Similar to the previous results, a low equivalence between both platforms and results is observed, with a notorious phase shift delay caused by the DSP's sampling processing time.

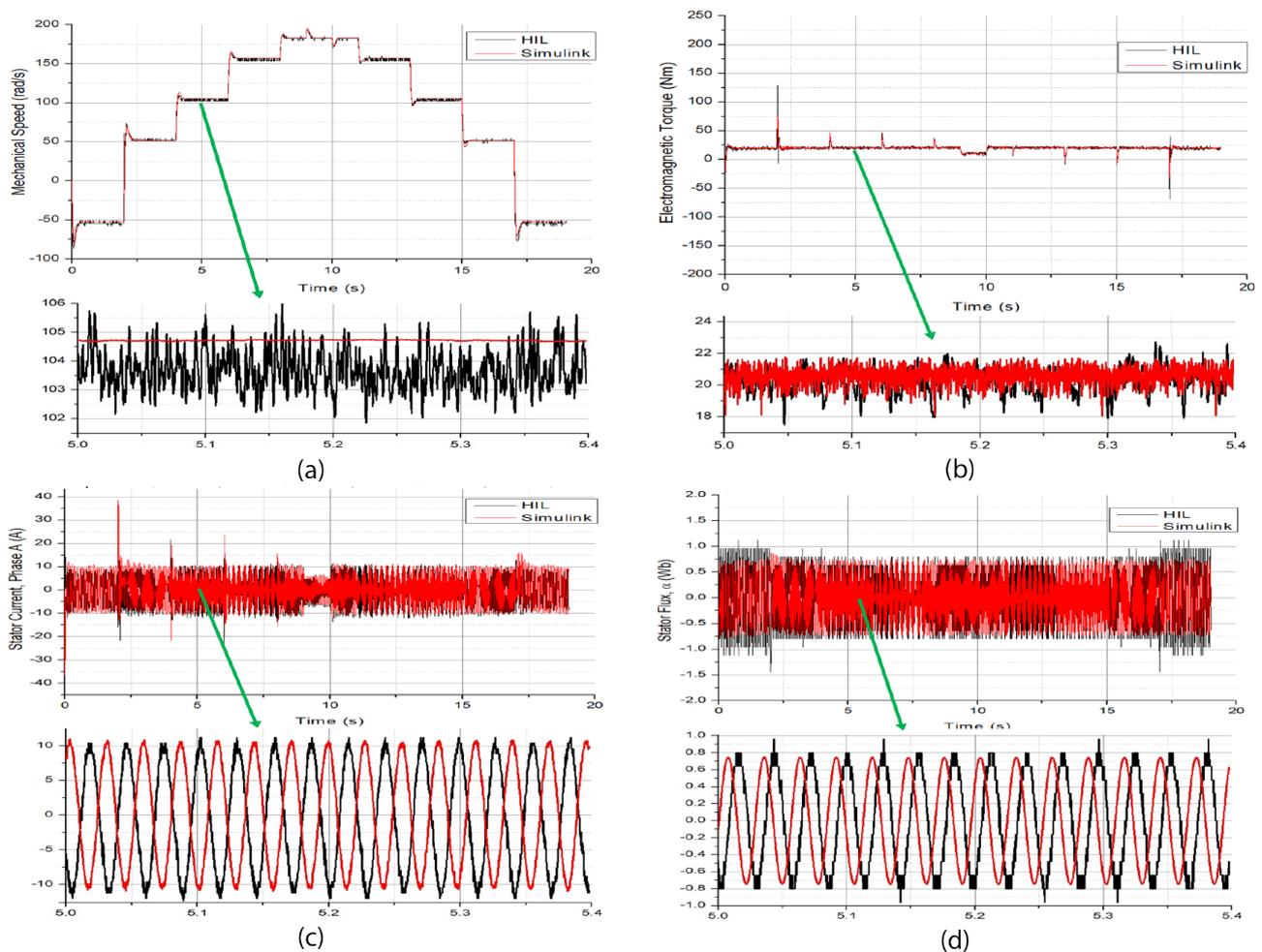


Figure 11. Numerical and practical results of the DTC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

Figure 12 shows the results of the MATLAB SIMULINK block functions and DSP-HIL of FTC. Figure 12a shows that FTC has a good speed response agreement for both systems during step speed transitions on both directions. In contrast to previous controllers, there is a neglectable speed error tracking in the overall speed transitions. Once again, a load change of -10 Nm was applied in 9 s and removed in 10 s, and the system came back to the reference in less than 1 s for both cases. Additionally, Figure 12b shows the torque response for the same speed reference. A torque zoom is also shown in the same figure. There is a low torque variation between SIMULINK and DSP-HIL responses.

Figure 12c,d shows the stator current phase A and stator flux, respectively. Similar to the previous results, a low equivalence between both platforms and results is observed, with a notorious phase shift delay in the phase caused by the DSP's sampling processing time.

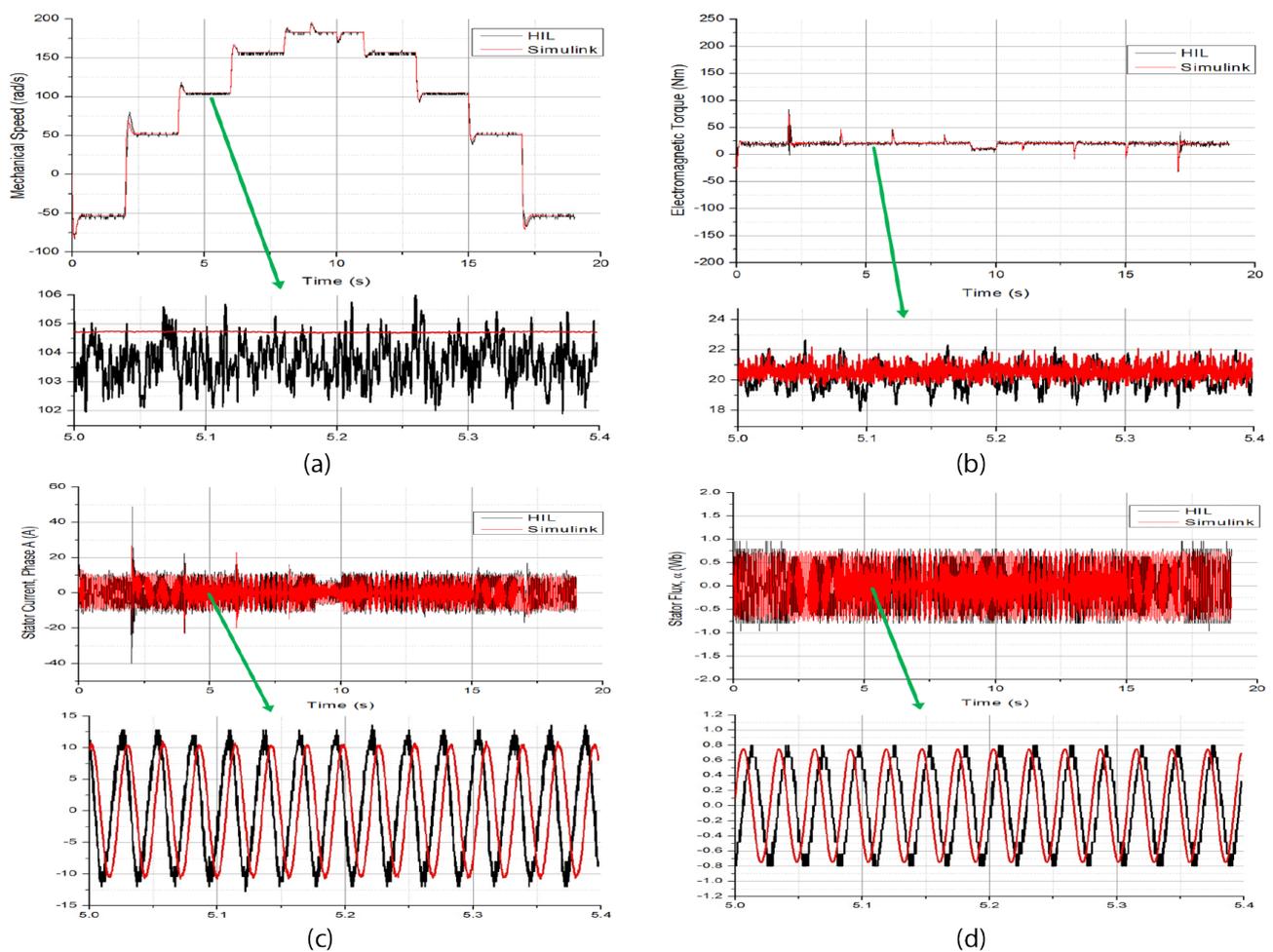


Figure 12. Numerical and practical results of the FTC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

The final tests were on PTC algorithm. Figure 13a shows that PTC also has a good speed response agreement for both systems during step speed transitions on both directions. Additionally, there is a neglectable speed error tracking in the overall speed transitions. Once again, a load change of -10 Nm was applied in 9 s and removed in 10 s, and the system came back to the reference in less than 1 s for both cases. Additionally, Figure 13b shows the torque response for the same speed reference. Similar to FTC response, a low torque variation between SIMULINK and DSP-HIL is noticed.

Figure 13c,d shows the stator current phase A and stator flux, respectively. A low equivalence between both platforms and results is observed, with a small phase shift delay in the phase.

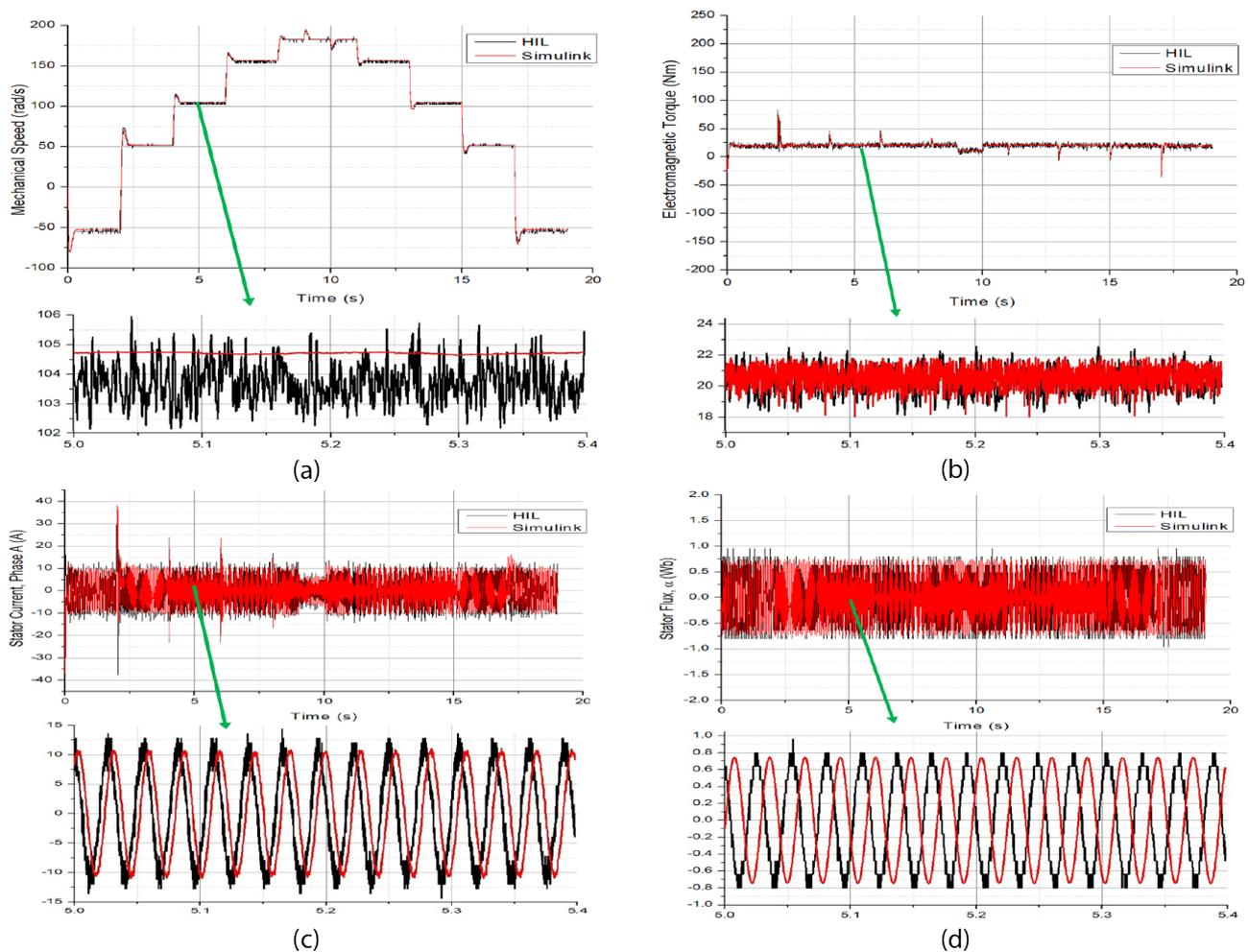


Figure 13. Numerical and practical results of the PTC variable speed controller: (a) speed; (b) torque; (c) stator current; and (d) stator flux.

4. Discussion

Table 1 provides the experimental data from the six different motor controllers. The top half of the table shows some of the main characteristics of the dynamic performance, followed by the hardware resources and general characteristics on the bottom.

DIFOC has the minimum settling time compared with the others. Additionally, it is apparent from these data that very little settling time improvement is achieved by the three torque controllers compared with current controllers. The smallest overshoot is reached by DIFOC. On the other hand, a neglectable speed variation between desired and practical response was achieved, with a maximum of 3% on the six controllers. One of the most interesting aspects of this graph is the IFOC response, which obtained the minimum torque ripple in the steady state and the minimum harmonic distortion on flux and current. It is also interesting to note that only two controllers behave with a fixed switching frequency, which helps to decrease the flux and current harmonic distortion. The final dynamic performance characteristic evaluated in this table refers to low speed behavior. There is a significant difference between the six controllers, and only two attained excellent behavior and the rest poor behavior. Note that a scale of poor, good and excellent score was used to evaluate this parameter.

Table 1. Comparative table of developed controllers at $T_l = 10$ Nm and speed of 1000 RPM. The dynamic performance, hardware resources and general characteristics are indicated with the colors green, light yellow and light orange, respectively.

Characteristics	IFOC	DIFOC	PCC	DTC	FTC	PTC
Settling time	~200 ms	~50 ms	~400 ms	~250 ms	~250 ms	~250 ms
Overshoot speed (%)	6.9	No overshoot	12.7	5.8	8.9	8.9
Speed variation [minimum peak, maximum peak]	[98.43%,100.08%]	[97.53%,101.47%]	[99.86%,100.23%]	[99.76%,100.23%]	[99.71%,100.23%]	[99.74%,100.23%]
Torque response time	~200 ms	~50 ms	~400 ms	~250 ms	~250 ms	~250 ms
Electromagnetic torque ripple in steady state (%)	4.01	24.62	7.81	4.49	11.12	14.44
Total harmonic distortion of flux (%)	0.2393	0.2843	0.3142	0.6994	0.9737	0.9698
Total harmonic distortion of current (%)	0.5607	1.1675	1.5693	7.355	7.2173	9.344
Switching frequency	Constant 40 kHz	Constant 40 kHz	Variable	Variable	Variable	Variable
Low speed behavior	Excellent	Excellent	Excellent	Poor	Poor	Poor
Microcontroller resources	GPIO, Timers PWM, ADC	GPIO, Timers PWM, ADC	GPIO, Timers ADC	GPIO, Timers ADC	GPIO, Timers ADC	GPIO, Timers ADC
Microcontroller memory used (%)	37.79	52.47	37.79	36.47	46.87	36.77
Execution time ($F_{clk} = 60$ MHz)	~16 μ S	~28 μ S	~25 μ S	~15 μ S	~31 μ S	~42 μ S
Clock cycles	911	1700	1497	879	1855	2494
Modulation Controllers	Necessary 3 PI	Necessary 2 PI and 1 PD Fuzzy	Unnecessary 1 PI	Unnecessary 1 PI and 2 hysteresis	Unnecessary 1 PI	Unnecessary 1 PI
Parameters sensitivity	R_r and L_r	R_r and L_r	All motor parameters	R_s	R_s	All motor parameters
Controlled variables	Currents	Currents	Currents	Torque and flux	Torque and flux	Torque and flux
Transformations	$abc \rightarrow \alpha\beta \rightarrow dq$ $dq \rightarrow \alpha\beta \rightarrow abc$	$abc \rightarrow \alpha\beta dq$ $dq \rightarrow \alpha\beta abc$	$abc \rightarrow \alpha\beta$ $dq \rightarrow \alpha\beta$	$abc \rightarrow \alpha\beta$	$abc \rightarrow \alpha\beta$	$abc \rightarrow \alpha\beta$

The next part of Table 1 is concerned with hardware resources. IFOC and DIFOC used more resources than the others, mainly due to the PWM function. DTC has the lowest memory used and execution time, contrasting with DIFOC and PTC, which have the highest. DTC uses the minimum number of clock cycles, followed by IFOC and PTC with the maximum. To calculate the processing time, it is necessary to divide the clock cycles by the clock frequency, i.e., $879/60$ MHz = 15 μ S. Having discussed the dynamic performance and hardware resources, we next address the overall characteristics. For example, modulation is only needed on IFOC and DIFOC, which utilize three compensators. By contrast, FTC and PTC use only one. PCC and PTC are techniques with the most sensitive motor parameters, being DTC and FTC the most robust. IFOC, DIFOC and PCC use only current information, while DTC, FTC and PTC use torque and flux. Indeed, only one transformation is requested by the torque techniques, compared to the first three, which use two.

4.1. Main Achievements Traditional Controllers

Considering the implemented drivers in detail, IFOC presents better behavior in terms of current and electromagnetic torque. Therefore, for control that is referenced with currents, these will be very well detailed compared to other types of control. Being a function of the current, the electromagnetic torque will also have a reduced ripple. The disadvantage of IFOC is that the implementation and philosophy to carry out the control is more complex than the others, e.g., DTC, mainly due to changing the coordinates. Furthermore, equations used in its control are difficult to find in the literature with greater implementation details. On the other hand, DTC displays considerable curls, which are mainly due to hysteresis drivers. Therefore, the nature of this type of regulator always introduces oscillations in the control variables.

4.2. Main Achievements Advance Controllers

Fuzzy methods consist of different stages initiated by the rule selection, followed by membership functions, the number of input and output fuzzy sets and their degree of intersecting, association, connection operations and defuzzification method. Open research

directions in fuzzy control techniques are reducing the computational load and increasing feasibility; both aims can be achieved by minimizing the number of memberships and using different defuzzification functions.

For instance, the implemented DIFOC consists of a diffuse Mamdani type with a defuzzification based on center of gravity (COG). This controller consists of a set of fuzzy “IF-THEN” rules, where the system takes as input the values of the fuzzification and applies them to the related. The AND and OR operators are used to obtain a single number that represents the evaluation result. This number (the truth value) is applied to the outcome. By using this controller, a neglectable speed variation between reference and practical responses was found and both systems follow the overall reference signal.

In contrast, the FTC has a similar behavior to DTC, that is, it practically presents the same magnitudes of current ripple and electromagnetic torque. This is because this work considers the characteristics of the DTC, that is, the limits of torque and flux hysteresis are the same as for DTC, e.g., the number of sectors, translated into the fuzzy logic philosophy. The main disadvantage of this controller is that it consumes a lot of storage memory, thus it cannot be applied if resources are limited.

On the other hand, predictive techniques are emerging in the field of power electronics, and these techniques are being adapted to conventional, DTC and IFOC controllers, with promising improvements for the future replacement of conventional control systems. Their downside is that they consume a significant processing time, and, according to our findings, for a horizon time greater than one, the response in ripple current and torque are greatly reduced in simulations. However, the calculation load grows even more, which made it impossible to use this method experimentally, at least in this DSP.

In particular, it was found experimentally that PCC has a problem with maximum load at low speeds (it gets out of control), which does not happen with simulations and has not been reported in the literature. This result suggests that the implementation considerations to adapt them to the DSP interfere with the performance of this controller. This surprising finding contrasts with previously studies [16–18], in which PCC experiments were performed with real-time controllers. A possible explanation can involve the restricted range used in DSP.

4.3. Other Strategies

Optimization techniques for IM are emerging for use in scalar, vector and direct induction motor drives. Hannan et al. [17] used biology- and physics-based optimization algorithms, and their numerical and real time results show promising outcomes. In general, optimization techniques can be applied to speed, current or torque controllers to find the best parameters. Indeed, it is claimed that they can be applied to minimize the error, flux changes of the torque, current ripple, noise and frequency variation. Additionally, optimization systems can achieve robustness, damping capability, enhanced transient responses and significant speed reduction responses in terms of overshoot, steady-state error and settling time [38]. However, optimization techniques have limitations of global minimum, trial-and-error tuning procedure, local minima and optimal trapping. They also have limitations in varying the algorithms and computational time to achieve the best optimization operation. Practical implementation into a microcontroller, DSP, FPGA or other digital platforms is still limited.

A recent trend in the PE community is the application of fractional-order (FO) theory in the control of power electronic converters [39] and, recently, motor drives. Claimed advantages of FO are its possibility to more accurately represent the nonlinear characteristic of passive element and the achievement of faster controllers compared with integers one [40].

An interesting effort to bring FO to the engineering classrooms was reported by Khubalkar et al. [41], who reported the FO implementation of analog fractional integrator, a digital fractional integrator, a FO-PID controller for a DC motor, a FO-PID controller for hardware in the loop MagLev system, a FO-PID controller for a brushless direct current

motor and a digital FO-PID controller for a direct current motor. A five-level space vector pulse width modulated multipoint clamped (MPC) inverter fed direct torque-controlled IM drive with FO PI controllers was reported by Chandra Sekhar and Lakhimsetty [42]. They claimed that the proposed drive system achieves full-load torque control with fast response and reduced torque ripple for a wide range of speed variations compared with traditional structure. Another FO PI controller-based direct torque control scheme for an induction motor (IM) drive was reported by Chandra Sekhar et al. [43]. They measured the IM performance during starting, running and speed reversal along with the step change in the load torque. Numerical and practical results in a 2 HP IM are reported, but no details of the FO tuning stability or implementation details are given.

In general, a non-integer rational transfer function (RTF) can include fractional integer and fractional derivative terms, which can be positive or negative [44]. To practically implement RTF in a physical testbed, they must first be approximated to a rational function of two polynomials with integer order inside a desired frequency band. Once the approximation process is done, the following step is synthesizing the RTF. After the approximation and synthesis process finishes, an electrical circuit can be obtained. To this end, there are two main approaches, analog and digital [45]. The first one uses passive components and operational amplifiers (OPAM), while the second one employs FPGA boards.

FO is an interesting approach, and it can be easily integrated into traditional FOC or DTC controllers. Up to date, some brief theory and practical results are reported, but a systematic procedure to bring FO modeling and control to the motor drive community is still lacking. This topic is open, and it can be foreseen that it will be a hot topic soon in the motor drives community, mainly due to three reasons: (1) it does not require too much processing capabilities compared with optimization techniques; (2) it can be easily integrated into current motor drives techniques; and (3) traditional linear control theory can be extended to FO controllers, e.g., Laplace operators and frequency response.

5. Conclusions

This study set out to determine a fair comparison between several adjustable speed drives. It is clear from the current study that their behaviors were, in general terms, as reported in the literature. Taken together, these results suggest that the performance of FOC controllers can be better if their control scheme parameters are adjusted adequately.

The FOC and DTC fuzzy controllers are based completely on the ideology of conventional schemes, incorporating the experience and conceptualization of whoever builds them. Because of this, their implementation was not complicated. However, it required knowledge about IM dynamics to avoid a cumbersome tuning.

On the other hand, it is also well known that predictive controllers make use of the exact system model, and this can often be a limitation. However, thanks to advances in digital electronics, today their implementation is possible in a limited integration time horizon; otherwise, the hardware/software resources increase greatly. A consideration of PTC controllers is that they use a weighting value that becomes a, sometimes rustic, heuristic process.

Overall, for the schemes, it is concluded that, with a small sampling time, the response will be better. However, this choice has a limit related to the DSP's processing speed. Similarly, the number of bits of the analog-to-digital converter directly affected the samples taken. It is important to note that, in most studies related to variable speed drives compared in this work, more powerful digital devices are used for the experimental results. However, this does not limit the ability to implement all the strategies on the DSP-HIL platform.

Author Contributions: Conceptualization, L.E.O.-G. and F.J.P.-P.; methodology, L.E.O.-G. and F.J.P.-P.; software, L.E.O.-G.; validation, L.E.O.-G., D.R.-S. and F.J.P.-P.; formal analysis, J.C.N.-P. and Y.S.-I.; investigation, L.E.O.-G.; resources, J.C.N.-P. and Y.S.-I., data curation, F.J.P.-P.; writing—original draft preparation, D.R.-S.; writing—review and editing, D.R.-S. and F.J.P.-P.; visualization, L.E.O.-G.; supervision, J.C.N.-P.; project administration, Y.S.-I.; and funding acquisition, J.C.N.-P. and Y.S.-I. All authors have read and agreed to the published version of the manuscript.

Funding: The authors thank IPN for its support provided through the project SIP-20210345 and Tecnológico Nacional de México, Instituto Tecnológico de Celaya, through the project 9040.20-P. The work of Luis E. Ortega-Garcia was supported by CONACYT through a M.Sc. grant. The work of Daniela Rodriguez-Sotelo was supported by CONACYT through a Ph.D. grant. In addition, the authors would like to express their gratitude to COFAA for its financial support.

Acknowledgments: The authors thank IPN, Tecnológico Nacional de México, Instituto Tecnológico de Celaya, CONACYT and COFAA.

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

Abbreviations

The following nomenclature is used in this manuscript:

R_s	Stator resistance	i_{rdq}	Rotor current
i_s	Stator current	L_m	Magnetizing inductance three-phase
V_s	Stator voltage	L_s	Self-induction of stator windings
Ψ_s	Stator flux	\bar{L}_r	Total inductance of the three-phase rotor
R_r	Rotor resistance	$i_{s\alpha}^*$	Real reference stator current component
i_r	Rotor current	R_σ	Resistance of linkage factor
Ψ_r	Rotor flux	$i_{s\beta}$	Imaginary stator current component
i_α	Real current component	$i_{s\beta}^*$	Imaginary reference stator current
i_β	Imaginary current component	$\Psi_{s\beta}$	Imaginary stator flux
Ψ_{rdq}	Rotor flux	Ψ_s^*	Flux stator reference
τ_r	Rotor constant time	τ_e^*	Electromagnetic torque reference
P_p	Pair of poles	i_β^*	Imaginary reference current component
λ	Weighting factor	i_α^*	Real reference current component
k	Number of sample	k_s	Relation of mutual-inductance with stator inductance
I_s	Stator current	$i_{s\alpha}$	Real stator current
T_s	Sampling time	τ_σ	Time constant of the linkage factor
σ	Linkage factor	k_r	Relation of mutual-inductance with rotor inductance
ω_s	Stator speed	Ψ_{rd}	Real rotor flux component
ω_r	Rotor speed	V_{rdq}	Rotor voltage
τ_e	Electromagnetic torque	$\Psi_{s\alpha}$	Real stator flux
$\Psi_{s\beta}$	Real rotor flux	τ_r	Rotor constant time
R_σ	Resistance of linkage factor		

Appendix A

The induction motor's parameters for numerical and HIL are shown in Table A1.

Table A1. IM parameters.

Parameter	Magnitude
Stator resistance	1.115 Ω
Stator inductance	5.974 mH
Rotor resistance	1.083 Ω
Rotor inductance	5.974 mH
Magnetizing inductance	203.7 mH
Friction	0.005752 Nms
Pole number	4
Nominal power	5 HP
Nominal frequency	60 Hz
Nominal voltage	460 V
Moment of inertia	0.02 kgm ²

References

1. Trzynadlowski, A.M. *Control of Induction Motors*; Academic Press: Cambridge, MA, USA, 2011. [[CrossRef](#)]
2. Bose, B.K. *Modern Power Electronics and AC Drives*, 1st ed.; Pearson: London, UK, 2001; p. 711.
3. Liu, H.; Gao, D. A novel flux oriented V/f control method of induction motor based industrial adjustable speed drives. In Proceedings of the 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 31 May–2 June 2018; pp. 1739–1744. [[CrossRef](#)]

4. Talla, J.; Leu, V.Q.; Šmídl, V.; Peroutka, Z. Adaptive Speed Control of Induction Motor Drive With Inaccurate Model. *IEEE Trans. Ind. Electron.* **2018**, *65*, 8532–8542. [[CrossRef](#)]
5. Akin, Ö.; Alan, I. The use of FPGA in field-oriented control of an induction machine. *Turk. J. Elec. Eng. Comp. Sci.* **2010**, *18*, 943–962. [[CrossRef](#)]
6. Naganathan, P.; Srinivas, S.; Ittamveetil, H. Five-level torque controller-based DTC method for a cascaded three-level inverter fed induction motor drive. *IET Power Electron.* **2017**, *10*, 1223–1230. [[CrossRef](#)]
7. Ponce Cruz, P.; Herrera, A. *Inteligencia Artificial con Aplicaciones a la Ingeniería*; Alfaomega: Kissimmee, FL, USA, 2010; p. 348.
8. Ross, T.J. *Fuzzy Logic with Engineering Applications*; Wiley: Hoboken, NJ, USA, 2016; p. 583.
9. Zhang, Y.; Xia, B.; Yang, H.; Rodriguez, J. Overview of model predictive control for induction motor drives. *Chin. J. Electr. Eng.* **2016**, *2*, 62–76. [[CrossRef](#)]
10. Raviteja, B.; Madhuri, N. Predictive torque control of induction motor. *Int. J. Adv. Res. Sci. Eng. Technol.* **2014**, *1*, 318–329.
11. Holkar, K.S.; Waghmare, L.M. An overview of model predictive control. *Int. J. Control. Autom.* **2010**, *3*, 47–64.
12. Wróbel, K. Finite Set Model Predictive Speed Control of Induction Motor with Long Horizon. *Power Electron. Drives* **2016**, *1*, 117–126.
13. Eswar, K.M.R.; Kumar, K.V.P.; Kumar, T.V. Enhanced Predictive Torque Control with Auto-Tuning Feature for Induction Motor Drive. *Electr. Power Components Syst.* **2018**, *46*, 825–836. [[CrossRef](#)]
14. Norambuena, M.; Rodriguez, J.; Zhang, Z.; Wang, F.; Garcia, C.; Kennel, R. A Very Simple Strategy for High-Quality Performance of AC Machines Using Model Predictive Control. *IEEE Trans. Power Electron.* **2019**, *34*, 794–800. [[CrossRef](#)]
15. Hemavathy, K.; Pappa, N.; Kumar, S. Comparison of indirect vector control and direct torque control applied to Induction Motor drive. In Proceedings of the IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, 8–10 May 2014; pp. 192–197. [[CrossRef](#)]
16. Wang, F.; Zhang, Z.; Mei, X.; Rodríguez, J.; Kennel, R. Advanced Control Strategies of Induction Machine: Field Oriented Control, Direct Torque Control and Model Predictive Control. *Energies* **2018**, *11*, 120. [[CrossRef](#)]
17. Hannan, M.; Ali, J.A.; Mohamed, A.; Hussain, A. Optimization techniques to enhance the performance of induction motor drives: A review. *Renew. Sustain. Energy Rev.* **2018**, *81*, 1611–1626. [[CrossRef](#)]
18. Karlovsky, P.; Lettl, J. Induction Motor Drive Direct Torque Control and Predictive Torque Control Comparison Based on Switching Pattern Analysis. *Energies* **2018**, *11*, 1793. [[CrossRef](#)]
19. Nauel, Y.; Todd, R. Digital control HIL comparison for adjustable speed drives. In Proceedings of the 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 2754–2759.
20. Rauber, A.; den Bakker, P. A Comparison of Adjustable-Speed Drive Systems: Voltage Source Inverters and Load-Commutated Inverters for High-Power Applications. *IEEE Ind. Appl. Mag.* **2020**, *26*, 56–66. [[CrossRef](#)]
21. Lauss, G.; Strunz, K. Accurate and Stable Hardware-in-the-Loop (HIL) Real-time Simulation of Integrated Power Electronics and Power Systems. *IEEE Trans. Power Electron.* **2020**. [[CrossRef](#)]
22. Prabakar, K.; Palmintier, B.; Pratt, A.; Hariri, A.; Mendoza, I.; Baggu, M.M. Improving the Performance of Integrated Power-Hardware-in-the-Loop and Quasi-Static Time-Series Simulations. *IEEE Trans. Ind. Electron.* **2020**. [[CrossRef](#)]
23. Iranian, M.E.; Mohseni, M.; Aghili, S.; Parizad, A.; Baghaee, H.R.; Guerrero, J.M. Real-Time FPGA-based HIL Emulator of Power Electronics Controllers using NI PXI for DFIG Studies. *IEEE J. Emerg. Sel. Top. Power Electron.* **2020**. [[CrossRef](#)]
24. Ahmad, J.; Pervez, I.; Sarwar, A.; Tariq, M.; Fahad, M.; Chakraborty, R.K.; Ryan, M.J. Performance Analysis and Hardware-In-the-Loop (HIL) Validation of Single Switch High Voltage Gain DC-DC Converters for MPP Tracking in Solar PV System. *IEEE Access* **2020**, *9*, 48811–48830. [[CrossRef](#)]
25. Liang, T.; Liu, Q.; Dinavahi, V.R. Real-Time Hardware-in-the-Loop Emulation of High-Speed Rail Power System With SiC-Based Energy Conversion. *IEEE Access* **2020**, *8*, 122348–122359. [[CrossRef](#)]
26. Chakraborty, S.; Mazuela, M.; Tran, D.; Corea-Araujo, J.A.; Lan, Y.; Loiti, A.A.; Garmier, P.; Aizpuru, I.; Hegazy, O. Scalable Modeling Approach and Robust Hardware-in-the-Loop Testing of an Optimized Interleaved Bidirectional HV DC/DC Converter for Electric Vehicle Drivetrains. *IEEE Access* **2020**, *8*, 115515–115536. [[CrossRef](#)]
27. Luo, Y.; Awal, M.A.; Yu, W.; Husain, I. FPGA Based High Bandwidth Motor Emulator for Interior Permanent Magnet Machine Utilizing SiC Power Converter. *IEEE J. Emerg. Sel. Top. Power Electron.* **2020**. [[CrossRef](#)]
28. Estrada, L.; Vázquez, N.; Vaquero, J.; de Castro, Á.; Arau, J. Real-Time Hardware in the Loop Simulation Methodology for Power Converters Using LabVIEW FPGA. *Energies* **2020**, *13*, 373. [[CrossRef](#)]
29. Khooban, M.; Gheisarnejad, M.; Vafamand, N.; Jafari, M.; Mobayen, S.; Dragicevic, T.; Boudjadar, J. Robust Frequency Regulation in Mobile Microgrids: HIL Implementation. *IEEE Syst. J.* **2019**, *13*, 4281–4291. [[CrossRef](#)]
30. Bai, Y.; Zhuang, H.; Wang, D. *Advanced Fuzzy Logic Technologies in Industrial Applications*; Springer: London, UK, 2006; p. 334. [[CrossRef](#)]
31. Shaw, I.S. *Fuzzy Control of Industrial Systems: Theory and Applications*, 1st ed.; Springer: New York, NY, USA, 1998; p. 192. [[CrossRef](#)]
32. Castillo, O.; Melin, P. *Type-2 Fuzzy Logic: Theory and Applications*, 1st ed.; Studies in Fuzziness and Soft Computing; Springer: Berlin/Heidelberg, Germany, 2008; Volume 223, p. 244. [[CrossRef](#)]
33. Wang, F.; Mei, X.; Rodriguez, J.; Kennel, R. Model predictive control for electrical drive systems-an overview. *CES Trans. Electr. Mach. Syst.* **2017**, *1*, 219–230. [[CrossRef](#)]

34. Uddin, M.; Mekhilef, S.; Nakaoka, M.; Rivera, M. Model predictive control of induction motor with delay time compensation: An experimental assessment. In Proceedings of the IEEE Applied Power Electronics Conference and Exposition (APEC), Charlotte, NC, USA, 15–19 March 2015; pp. 543–548. [[CrossRef](#)]
35. Seborg, D.E.; Thomas, F. Edgar Mellichamp, D.A.; Doyle, F.J., III *Process Dynamics and Control*; Wiley: Hoboken, NJ, USA, 2016; p. 512.
36. Rossiter, J. *Model-Based Predictive Control: A Practical Approach*; Control Series; CRC Press: Boca Raton, FL, USA, 2003.
37. Gan, L. Model Predictive Control of Induction Motor Drive with Constraints. Ph.D. Thesis, RMIT University, Melbourne, Australian, 2014.
38. Sharma, F.; Kapoor, S. *Recent Soft Computing Strategies for Induction Motor Analysis and Control: A Survey*; Springer: Berlin/Heisenberg, Germany, 2021; pp. 67–71. [[CrossRef](#)]
39. Radwan, A.G.; Emira, A.A.; AbdelAty, A.M.; Azar, A.T. Modeling and analysis of fractional order DC-DC converter. *ISA Trans.* **2018**, *82*, 184–199. [[CrossRef](#)] [[PubMed](#)]
40. Tavazoei, M. Time response analysis of fractional-order control systems: A survey on recent results. *Fract. Calc. Appl. Anal.* **2014**, *17*. [[CrossRef](#)]
41. Khubalkar, S.; Junghare, A.; Aware, M.; Das, S. Unique fractional calculus engineering laboratory for learning and research. *Int. J. Electr. Eng. Educ.* **2020**, *57*, 3–23. [[CrossRef](#)]
42. Chandra Sekhar, O.; Lakhimsetty, S. Direct torque control scheme for a five-level multipoint clamped inverter fed induction motor drive using fractional-order PI controller. *Int. Trans. Electr. Energy Syst.* **2020**, *30*, e12474. [[CrossRef](#)]
43. Chandra Sekhar, O.; Lakhimsetty, S.; Bhat, A.H. A comparative experimental analysis of fractional order PI controller based direct torque control scheme for induction motor drive. *Int. Trans. Electr. Energy Syst.* **2021**, *31*, e12705. [[CrossRef](#)]
44. Vinagre, B.; Podlubny, I.; Hernández, A.; Feliu, V. Some approximations of fractional order operators used in control theory. *Fract. Calc. Appl. Anal.* **2000**, *3*, 231–248.
45. Tlelo-Cuautle, E.; Pano-Azucena, A.; Guillén-Fernández, O.; Silva-Juárez, A. *Analog/Digital Implementation of Fractional Order Chaotic Circuits and Applications*; Springer: Berlin/Heisenberg, Germany, 2019.