Simulation Tool to Evaluate Fault Diagnosis Techniques for DC-DC Converters

Acacio M. R. Amaral 1,* and Antonio J. Marques Cardoso 2

1 CISE, Department of Informatics and Systems, Polytechnic Institute of Coimbra, ISEC, Quinta da Nora, P-3030-199 Coimbra, Portugal
2 CISE—Electromechatronic Systems Research Center, University of Beira Interior, Calçada Fonte do Lameiro, P-6201-001 Covilhã, Portugal
* Correspondence: amramaral@ieee.org

Abstract: This paper presents a tool that evaluates the applicability of a fault diagnostics technique (FDT) for Switch-Mode Non-Isolated DC-DC converters (SMNI DC-DC). The proposed tool simulates the operation of a Type II controller for a buck converter under different operating conditions and, simultaneously, implements the FDT. In this way, it is possible to evaluate the applicability of the FDT in real-time. The proposed tool is implemented in Python, and requires the partitioning of the code into different modules. The first module is responsible for modeling the different states of the converter. The second module contributes to the production of the PWM signal. The concepts of symmetry and asymmetry are implicit to the signal processing methods used in the design of the FDTs that compose the third module. Finally, all of the modules are properly included in the main function, which represents the fourth module. The validation of the proposed tool is carried out by comparing the results with the LTspice simulator and with an experimental prototype built for this purpose. Different operating conditions are considered. In addition to the proposed tool, an algorithm for the design of both the power section and the converter compensator is presented.

Keywords: condition monitoring; fault diagnosis; aluminum electrolytic capacitors (Al-Caps); equivalent series resistance (ESR); buck converter; type II controller; simulation; python

1. Introduction

Power converters are systems that convert energy from one form (source) to another form (load), with high performance and efficiency. These systems must ensure compliance with electromagnetic interference (EMI) standards [1] and high reliability [2–5]. Most power converters lack redundancy [2], such that a failure in one component can lead to the system shutdown. Outages resulting from unexpected failures can decrease system security and increase operating costs [2,3]. For instance, in electric or hybrid electric vehicles (HEVs), a failure in the electric propulsion system can reduce the fuel economy and, under certain circumstances, may reduce safety [2,3].

Power converters can be subdivided into four categories: AC/AC, AC/DC, DC/DC and DC/AC. This paper focuses on DC/DC converters, which are widely used in several critical systems, such as DC micro grids, electric vehicles, data centers, medical equipment, aerospace power systems, and others [2–5]. It is important to highlight that a loss of functionality or an unexpected stoppage of these systems can result in a catastrophe.

In order to improve the reliability of DC-DC converters, two solutions can be implemented: the design of redundant systems [6,7] and the implementation of an online FDT [8,9] during the converter design. In both cases, the proposed tool can be fundamental. The design of DC-DC converters requires a set of fundamental concepts, such as the identification of component specifications, choosing the best gate drive circuit and the best heat sinks for the power switch, designing inductors, and solving EMI problems, among others. On the other hand, if the system to be developed fits into a critical application,
the development of redundant circuits or the implementation of online FDT is another fundamental aspect of the design process.

In the design process, simulation tools play a key role, as they allow the testing of the proposed solution in a very simple and flexible way. Therefore, before developing the experimental prototype, it is important to carry out a set of tests in a computer simulation environment to verify the feasibility of the proposed solution.

One of the most commonly used tools to simulate DC-DC converters is MATLAB/Simulink [3,10–13]. The simplicity in the design of the DC-DC converters and the user-friendly graphical interface makes MATLAB/Simulink very attractive, playing a fundamental role in carrying out research tasks [3,10–13]. However, as MATLAB is a proprietary programming language, its algorithms are proprietary; thus, it is not possible to access its code. This, particularly, can significantly limit the implementation of control methods, as well as the development of fault diagnostic techniques [14,15].

A good alternative to MATLAB is the Python programming language [15–17]. Despite the fact that it does not allow a model-based design like MATLAB, it has the computational power and tools to simulate DC/DC converters [15]. On the other hand, Python is an open-source programming language that has several machine learning algorithms that can be very useful in the development of fault diagnostic techniques for DC-DC converters [8,18,19]. It is important to highlight the fact that simulation techniques based on open-source tools are a very recent topic of investigation [15–17], and their application in the scope of FDTs has not been studied. This work intends to demonstrate that the simulation techniques mentioned above can evaluate the applicability of FDTs in DC-DC converters without the need to design an experimental prototype.

The proposed tool simulates the DC-DC converter operating in closed loop, and simultaneously implements the FDT in real time. The FDT used in the scope of this paper uses Fast Fourier Transform (FFT), which is a signal processing method that implicitly uses the concepts of symmetry and asymmetry. The proposed simulation tool can be implemented in a high-level open-source programming language oriented to numerical analysis, such as Python, Scilab or Octave.

The programming language used to implement this tool will be Python, and the DC-DC converter under analysis will be a buck converter operating in closed loop, with a type II compensator.

The rest of this paper is organized as follows. Section 2 describes the implementation of the proposed tool (the first, second and fourth blocks of the code) for a buck converter operating in closed loop. Section 3 presents the converter design process, and in Section 4 a comparison is carried out between the proposed tool and LTspice for validation purposes. Section 5 presents the implementation of the FDT (the third block of code) using the proposed tool and several experimental results. Finally, Section 6 presents the conclusions.

2. Simulation Closed-Loop Operation

As mentioned in the previous section, the proposed tool will be implemented in Python, which is a high-level, general-purpose programming language. Therefore, at first it is essential to make the development environment suitable for simulating DC-DC converters.

Python has a vast set of libraries, the most important, in this context, are Numpy and Matplotlib. The Numerical Python package (Numpy) is used for manipulating arrays and matrices, and the Matplotlib package is used for plotting. These libraries make the development environment suitable for numerical analysis.

In the following, it will be described how the proposed tool can be implemented in Python. For this purpose, it will be necessary to divide the code into four modules:

1. The first module will be responsible for modelling the different states of the converter.
2. The second module generates a saw-tooth signal that is fundamental for the control section.
3. The third module implements the FDT, and it will be described in Section 5.
4. The fourth module represents the main function, which integrates all of the modules and implements the control section.
2.1. First Module

Figure 1 shows an electrical schematic of a buck converter.

During the conduction state, the transistor, \( M_{ps} \), conducts in contrast to the diode, \( D_{ps} \). Consequently, current will flow through the inductor, \( L_{ps} \), which stores energy in the form of a magnetic field; simultaneously, the capacitor, \( C_{ps} \), is charged.

During the non-conduction state, \( M_{ps} \) is turned off, which forces the current to freewheel around the path consisting of \( L_{ps} \), \( C_{ps} \) and \( D_{ps} \).

After analyzing the equivalent circuits corresponding to each of the states, equations must be discretized and represented in the form of a function (Figure 2). This procedure can be found in detail in [15].

The functions represented in Figure 2 have the following elements as input arguments: the input voltage \( V_{in} \), \( L_{ps} \) resistance \( (R_{l}) \), \( L_{ps} \) inductance \( (L) \), \( M_{ps} \) drain-source resistance \( (R_{a}) \), \( D_{ps} \) forward voltage drop \( (V_{d}) \), \( D_{ps} \) internal resistance \( (R_{d}) \), \( C_{ps} \) capacitance \( (C) \), \( C_{ps} \) internal resistance \( (ESR) \), load resistance \( (R_{Load}) \), initial time \( (t_{0}) \), sample period \( (PA) \), \( L_{ps} \) initial condition \( (i_{10}) \), and \( C_{ps} \) initial condition \( (v_{c0}) \).

The functions represented in Figure 2 return the following elements: the inductor current \( (i_{L}) \), capacitor current \( (i_{C}) \), output voltage \( (v_{O}) \), output voltage component due exclusively to \( C \) \( (v_{C}) \), and time \( (t) \), respectively.

The control system designed for the converter assumes that it operates in Continuous Conduction Mode (CCM). However, for short periods of time, namely during the transient regime, the converter might operate in Discontinuous Conduction Mode (DCM). In DCM, the inductor current will be canceled because it cannot reverse polarity due to \( D_{ps} \). This situation was taken into account in the code, as can be seen in Figure 2b.
2.2. Second Module

After creating the first module, it is necessary to create the second one, which is essential for the generation of the Pulse Width Modulator (PWM) signal. At first, it is important to analyze the control circuit (Figure 3).

![Control circuit schematics](image)

**Figure 3.** Control circuit schematics.

As it can be seen in Figure 3, the control circuit is composed of three fundamental elements: the PI controller, the Comparator, and the Saw-Tooth Generator.

The first two elements will be integrated into the main function (fourth module). Therefore, the second module is only responsible for generating the saw-tooth signal, which is responsible for defining the converter operating frequency.

The saw-tooth generator must be implemented in a specific function (Figure 4). The function should receive as input arguments the amplitude of the Saw-tooth waveform (Amp), the Period (P), the final simulation time (tₚₚₜ), and the total number of iterations, TAM. In turn, the function returns the time vector (t), as well as a saw-tooth waveform with period P (sinal).

```python
def dente_serra(Amp, P, tₚₚₜ, TAM):
    PA = Amp/float(TAM); sinal = np.zeros(TAM); tempo = 0; t₀ = 0; j = 0
    while t₀ < P:
        t₀ += PA; j += 1
        Vₛ = Amp / float(t₀); t₀ = 0; Valor = 0; tempo = 0
        while k < TAM:
            if t₀ <= P:
                t₀ = 0; valor = Qsinal[i] = valor
            else:
                t₀ = PA; valor = VSsinal[i] = valor
            tempo += PA[tₜ]; tempoji += 1
        return [t[sinal]]
```

**Figure 4.** Second module.

2.3. Fourth Module

The fourth module corresponds to the main function, which must properly integrate the previous modules and, simultaneously, implement the PI controller and the comparator.

Figure 5 presents a flowchart that shows how to implement the main function in Python. CI represents the current iteration, and t₀, i_L, V_C, V_O, i_C, SignalsST and Signalcont represent the vectors referring to the following quantities: the time, inductor current, output voltage component due exclusively to C, output voltage, capacitor current, saw-tooth signal and PI controller output (v_control), respectively.
2.3. Fourth Module

The fourth module corresponds to the main function, which must properly integrate the previous modules and, simultaneously, implement the PI controller and the comparator.

Figure 5 presents a flowchart that shows how to implement the main function in Python.

CI represents the current iteration, and t, iL, vC, vO, iC, SignalST and Signalcont represent the vectors referring to the following quantities: the time, inductor current, output voltage component due exclusively to C, output voltage, capacitor current, saw-tooth signal and PI controller output (vcontrol), respectively.

3. Buck Converter Design

In this section, the process of designing a buck converter will be presented. At first, the power section will be designed using the following specifications.

Table 1 shows that the converter must operate in CCM, that is, the Lps inductance (L) must be greater than its critical value (Lcritical). In order to compute Lcritical, it is necessary to perform the following steps:

- First, a set of rules must be considered, namely, the converter must be operating in a steady-state regime, the inductor average voltage must be zero, the average current in the capacitor should also be zero, all components should be considered ideal, and the capacitor capacitance should be large enough.
- Following this, after analyzing the theoretical waveform of the inductor current, it is possible to obtain the mathematical relationship between the inductor ripple current and the input and output voltage.
- Finally, we should consider the following condition: the average value of the inductor current is equal to half of its ripple (critical conduction mode or transition mode); it is possible to calculate the Lcritical as shown in (1).
\[ L > L_{\text{critical}} = \frac{1 - D}{2 \times f} \times R = \frac{1 - \frac{8}{25}}{2 \times 25k} \times \frac{25}{20} = 14.6 \, \mu H \]  

(1)

Table 1. Power section specification.

| Input Voltage | Output Voltage | Output Power \((P_O)\) | Output Voltage Ripple \((\Delta V_O)\) | Inductor Ripple Current \((\Delta i_L)\) | Operating Frequency |
|---------------|----------------|------------------------|--------------------------------|--------------------------------|-------------------|
| 12 V          | 5 V            | 20 W                   | Defined by Capacitor ESR       | CCM                                | 25 kHz            |

Taking into account condition (1), an inductor with an inductance of 22 \( \mu H \) (11 A) was chosen. In order to reduce the output voltage ripple, an aluminum electrolytic capacitor (Al-Cap) was used, as it has a high capacitance value. Therefore, an Al-Cap with 1000 \( \mu F \) (25 V) capacitance was chosen.

The converter output voltage ripple is essentially due to the capacitor ESR, because its operating frequency is close to the capacitor resonance frequency [20]. The Al-cap mathematical model is composed of a series connection of a resistor (ESR), an inductor (ESL), and a capacitor (C). When the Al-cap operates near its resonant frequency, the ESL effect cancels the C effect. Thus, using the method proposed in [21], it is possible to obtain the capacitor ESR experimentally (ESR = 0.077 \( \Omega \)).

Following this, the converter control section will be designed. For this purpose, the specifications presented in Table 2 will be applied.

Table 2. Controller specifications.

| Crossover Frequency \((f_C)\) | Desired Phase Margin \((PM_{\text{desired}})\) | Steady-State Error |
|-----------------------------|---------------------------------|------------------|
| 5 kHz                       | 45°                             | 0                |

In order to design the controller, it is necessary to obtain a mathematical model that represents the dynamic behavior of the system to be controlled (dynamic model). The dynamic model can be represented through state space representation or through the transfer function. The first one can be obtained through a set of differential equations that model the system, and the second one (transfer function) can be obtained through state space representation (first one). From the controller’s point of view, its input signal corresponds to the converter output voltage, and its output is the control signal (duty cycle). For this reason, the buck converter transfer Equation (2) relates the output voltage \( (v_o) \) of the converter with the duty cycle \( (d) \).

\[
\frac{v_o}{d} (s) = \frac{V_{\text{in}} \times (1 + s \times \text{ESR} \times C)}{1 + s \times \left( \frac{1}{R} + C \times \text{ESR} \right) + s^2 \times L \times C \times \left( 1 + \frac{\text{ESR}}{R} \right)}
\]  

(2)

The effect of the complex conjugate poles of transfer Equation (2) will cause the gain curve to drop \(-40 \text{ db}\), and tries to bring the phase curve towards \(-180^\circ\). A few frequencies later, the zero effect resulting from the ESR arises, that is, there is a reduction in the slope of the gain curve to \(-20 \text{ db}\), and the phase curve shifts to \(-90^\circ\) (Figure 6).
In the controller design (Type II), the k-factor method will be used. The type II controller provides a region between its zero \((w_Z)\) and its pole \((w_P)\), in which the gain value is constant and the phase undergoes a shift from \(-90^\circ\) to \((\varphi_{\text{boost}}=-90^\circ)\). Therefore, it is possible to introduce the desired phase margin \((\text{PM}_{\text{desired}})\) at the crossover frequency \((f_C)\) of the system (buck converter plus the controller) if \(w_Z\) and \(w_P\) are properly chosen. The k-factor method computes the \(w_Z\), \(w_P\) and gain \((K_C)\) values that ensure the initial specifications (Table 2).

Following this, the k-factor method will be applied. At first, it is necessary to obtain the phase of the buck converter \((2)\) at the crossover frequency \((f_C)\) (Figure 6).

From the previous figure, it is possible to obtain the phase of plant \((2)\) at \(f_C\) \((\varphi_{\text{system}} \approx -104.5^\circ)\).

The required phase boost, \(\varphi_{\text{boost}}\), can be computed as follows [22]:

\[
\varphi_{\text{boost}} = \text{PM}_{\text{desired}} - \varphi_{\text{system}} - 90^\circ \Rightarrow \varphi_{\text{boost}} = 45^\circ - (-104.5^\circ) - 90^\circ \equiv 59.5^\circ \tag{3}
\]

Using the computed \(\varphi_{\text{boost}}\), it is possible to obtain the controller \(w_P\) and \(w_Z\) as follows [22]:

\[
k = \tan\left(\frac{\varphi_{\text{boost}}}{2}+45^\circ\right) \equiv 3.67 \Rightarrow \begin{cases} w_Z = \frac{w_C}{k} \\ w_P = \frac{w_C}{k} \times k \end{cases} \Rightarrow \begin{cases} w_Z \equiv 8560 \\ w_P \equiv 115,296 \end{cases}, w_C = 2 \times \pi \times f_C \tag{4}
\]

Equation (4) introduces a \(\varphi_{\text{boost}}\) shift at the geometric mean of \(w_Z\) and \(w_P\), which is \(w_C\).

The controller transfer function (Figure 3) can be represented by the following equation:

\[
G_{\text{ctrl}}(s) = \frac{K_C}{s} \times \left(1 + \frac{s}{w_Z}\right) \times \left(1 + \frac{s}{w_P}\right)^{-1} \tag{5}
\]

In order to compute \(K_C\), it is necessary to represent the Bode magnitude plot of the set of the controller \((K_C = 1)\), modulator and buck converter.

The value of \(K_C\) is equal to the inverse of the magnitude value of the graph of Figure 7 at the crossover frequency \((-95.74 \text{ db})\).

\[
K_C = 10^{-\frac{(-95.74 \text{ db})}{20}} \equiv 61,235 \tag{6}
\]
Finally, the controller transfer function is obtained.

\[ G_{\text{ctrl}}(s) \approx \frac{(61,000 + 7.13 \times s)}{(s + 8.7 \times 10^{-6} \times s^2)} \quad (7) \]

The following Figure 8 represents the frequency response of the set of the controller (\( K_C = 61 \, k \)), modulator and buck converter.

![Figure 8](image-url)

**Figure 8.** Frequency response of the final system (controller (\( K_C = 61 \, k \)), modulator and buck converter) obtained in Scilab.

The previous figure shows that the final system behaves according to the following specifications (Table 2): PM\(_{\text{desired}} = 180^\circ\) — (final system phase at \( f_C = 45^\circ \).

Alternatively, it is possible to obtain the following transfer function for the PI controller (type II) represented in Figure 3.

\[ G_{\text{ctrl}}(s) = \frac{1 + R_2 \times C_2 \times s}{R_1 \times (C_1 + C_2) \times s + (R_1 \times R_2 \times C_2 \times C_1) \times s^2} \quad (8) \]

Using (5) and (8), it is possible to write the following:

\[ C_2 = \frac{1}{w_2 \times R_2} \land R_1 = \frac{1}{K_C \times C_2} \times \left( 1 - \frac{1}{w_p \times R_2 \times C_2} \right) \land C_1 = \frac{1}{K_C \times w_p \times R_2 \times C_2 \times R_1} \quad (9) \]

Assuming \( R_2 = 1 \, k\Omega \), it is possible to calculate the remaining controller elements: \( C_2 \approx 117 \, nF, R_1 \approx 129 \, \Omega \) and \( C_1 \approx 9.4 \, nF \). In this way, the following standard values were chosen: \( R_1 = 100 \, \Omega \), \( R_2 = 1 \, k\Omega \), \( C_1 = 10 \, nF \) and \( C_2 = 100 \, nF \).

### 4. Proposed Tool versus LTspice

In this section, a comparison will be performed between the proposed tool and the LTspice simulator. Figure 9 shows the implementation of the converter in LTspice.

![Figure 9](image-url)

**Figure 9.** Schematic diagram of the circuit implemented in LTspice.

It is important to mention that LTspice is one of the most frequently used pieces of simulation software in both industry and academia [23–25]. However, LTspice does not
allow the implementation of FDT; for that reason, the third module will be presented in Section 5.

In order to assess the feasibility of the proposed tool, three different operating conditions are considered: steady state (Figure 10), load variation (Figure 11) and input voltage variation (Figure 12).

![Comparison between LTspice (blue curve) and Python (red curve)](image1)

**Figure 10.** Waveforms of $v_O$, $i_L$ and $i_C$ obtained in the steady-state regime ($R_{\text{Load}} = 1.3 \, \Omega$), with LTspice (blue curve), and with the proposed tool (red curve).

![Comparison between LTspice (blue curve) and Python (red curve)](image2)

**Figure 11.** Waveforms of $v_O$, $i_L$ and $i_C$ obtained during a load variation (from 10 A to 5 A) at 2.5 ms, with LTspice (blue curve), and with the proposed tool (red curve).

The previous graphs show that the proposed tool presents very similar results to LTspice regarding the quantities under analysis ($v_O$, inductor current—$i_L$, and capacitor current—$i_C$) for the different operating conditions (steady state, load variation and input voltage variation).

The $L_{\text{ps}}$ inductance value of the experimental prototype was experimentally calculated (27 $\mu$H). For this reason, a 27 $\mu$H value was used in the simulations instead of 22 $\mu$H.
Figure 11. Waveforms of $v_O$, $i_L$ and $i_C$ obtained during a load variation (from 10 A to 5 A) at 2.5 ms, with LTspice (blue curve), and with the proposed tool (red curve).

Figure 12. Waveforms of $v_O$, $i_L$ and $i_C$ obtained during an input voltage variation (from 9 V to 15 V) at 2.5 ms, with LTspice (blue curve), and with the proposed simulation technique (red curve).

5. Fault Diagnostics Technique—Third Module

This paper does not intend to present a new fault diagnostics technique for DC-DC converters; rather, it presents a tool that can be very useful in its development. However, it is important to introduce the issue of fault diagnosis such that the relevance of the proposed tool can be understood.

About 30% of converter failures are caused by capacitor degradation [26], which makes this component one of its most vulnerable elements [8,9,26]. In the literature, several fault diagnostic methods are proposed that allow the evaluation of the health condition of capacitors used in DC-DC converters [8,9,27–33].

During the aging process, Al-Caps lose part of their electrolytes, which results in an increase in the ESR. Therefore, it can be concluded that the identification of this electrical parameter allows the assessment of Al-Cap health status. The estimation of the ESR will be performed by computing the capacitor impedance at the converter switching frequency (csf), as the converter operates very close to the capacitor’s resonant frequency [33].

The calculation of the capacitor impedance is carried out by computing the ratio between the first harmonic ($h_{csf}^{csf}v_C$) of the capacitor voltage ($h_{csf}^{csf}v_C$) and current ($h_{csf}^{csf}i_C$) [33]. In order to compute the $h_{csf}^{csf}$ value of the capacitor voltage and current, an FFT algorithm will be used. The NumPy library provides a function that computes the fft of a signal, which was used in the development of the third module.

The third module is composed of a function that receives as input arguments $t$, $v_O$ and $i_C$. The above signals must have a duration of the converter switching period (csp), and must be properly handled such that the NumPy fft function can correctly compute the $h_{csf}$ of both signals. Following this, the ESR is calculated by computing the ratio between $h_{csf}^{csf}v_C$ and $h_{csf}^{csf}i_C$. Finally, the function returns the ESR value.

Module 3 can be easily integrated into the algorithm of Figure 5. At first, it is necessary to obtain the input arguments of the function: $t$, $v_O$ and $i_C$. Therefore, for each csp, the three vectors must be obtained. As a result, the function will return the ESR value for that specific csp.

In order to evaluate the applicability of the third module, four different operating conditions will be considered, which are summarized in Table 3.
During the aging process, Al-Caps lose part of their electrolytes, which results in an increase in the ESR. Therefore, it can be concluded that the identification of this electrical parameter allows the assessment of Al-Cap health status. The estimation of the ESR will be performed by computing the capacitor impedance at the converter switching frequency (csf), as the converter operates very close to the capacitor’s resonant frequency [33].

In order to evaluate the applicability of the third module, four different operating conditions will be considered, which are summarized in Table 3.

### Table 3. Four Operating Conditions—Third Module Evaluation.

| Situation | Operating Conditions |
|-----------|----------------------|
| first     | V_{in} = 9 V, R_{Load} = 1.3 Ω, csf = 26.5 kHz |
| second    | V_{in} = 9 V, R_{Load} = 2.2 Ω, csf = 26.5 kHz |
| third     | V_{in} = 12 V, R_{Load} = 1.3 Ω, csf = 26.5 kHz |
| fourth    | V_{in} = 15 V, R_{Load} = 1.3 Ω, csf = 26.5 kHz |

Figures 13–16 show the v_{O}, i_{L} and i_{C} waveforms for the four situations represented in Table 3. From these figures, it is possible to compare the waveforms obtained from the proposed tool with the experimental ones. The experimental results were obtained from an experimental prototype with characteristics similar to the circuit of Figure 9 (considering the operating conditions in Table 3).

![Figure 13](image1.png)

**Figure 13.** Waveforms of v_{O}, i_{L} and i_{C} obtained for the first situation described in Table 3. The left plots represent the experimental results, and the right plots represent the waveforms obtained from the proposed tool.

![Figure 14](image2.png)

**Figure 14.** Waveforms of v_{O}, i_{L} and i_{C} obtained for the second situation described in Table 3. The left plots represent the experimental results, and the right plots represent the waveforms obtained from the proposed tool.
Figure 14. Waveforms of $v_O$, $i_L$ and $i_C$ obtained for the second situation described in Table 3. The left plots represent the experimental results, and the right plots represent the waveforms obtained from the proposed tool.

Figure 15. Waveforms of $v_O$, $i_L$ and $i_C$ obtained for the third situation described in Table 3. The left plots represent the experimental results, and the right plots represent the waveforms obtained from the proposed tool.

Figure 16. Waveforms of $v_O$, $i_L$ and $i_C$ obtained for the fourth situation described in Table 3. The left plots represent the experimental results, and the right plots represent the waveforms obtained from the proposed tool.

The previous figures contain data for two csp, such that by using module 3 it is possible to estimate the ESR value for two csp.

At first, module 3 was applied to the simulation results (Python results—proposed tool). Thus, considering the four situations described in Table 3, the maximum error obtained was less than 0.35% (ESR = 0.077265 Ω).

Next, module 3 was applied to the experimental waveforms. As a result, the following ESR values were obtained (Table 4).

| Situation | ESR (Ω) | ERROR (%) |
|-----------|---------|-----------|
| 1st       | 0.0803  | 4.3%      |
| 2nd       | 0.0768  | 0.3%      |
| 3rd       | 0.0783  | 1.7%      |
| 4th       | 0.0760  | 1.3%      |

The previous results (ESR estimation from the experimental and simulation results) clearly show that the proposed tool can be very useful. Thus, at first, it is possible—with a
high degree of certainty and using the proposed tool—to predict the reliability of an FDT. It should be noted that, in this first step (ESR estimation from simulation results), it is not necessary to design an experimental prototype.

The relevance of the proposed tool becomes even more evident when the system is subjected to external disturbances such as load variation or input voltage variation. The proposed tool can easily evaluate the performance of an FDT during transients, as shown in Figures 17 and 18.

The previous results (ESR estimation from the experimental and simulation results) in Figures 17 and 18.

The previous figure shows that during the transient, the maximum error was greater than 110%.

The proposed solution was implemented in Python, as it is a high-level open-source programming language oriented to numerical analysis, unlike MATLAB/Simulink, which is a proprietary language, and is for that reason much less flexible. In this particular, it is important to mention that proprietary languages makes the implementation of FDT much more difficult. By comparing the proposed tool with LTspice and experimental results, it was possible to prove its applicability. Therefore, it was demonstrated that the proposed tool can easily evaluate the performance of an FDT during transients, as shown in Figures 17 and 18.

Figure 17. FDT response when the converter is subjected to a load variation from 20 A to 5 A at 2.5 ms. $v_O$, $i_L$, and $i_C$ represent the waveforms of the output voltage, inductor current and capacitor current, respectively. The ESR represents the value of the equivalent series resistance of the Al-Cap estimated from module 3 (proposed tool).

Figure 18. FDT response when the converter is subjected to an input voltage variation from 19 V to 9 V at 2.5 ms. $v_O$, $i_L$, and $i_C$ represent the waveforms of the output voltage, inductor current and capacitor current, respectively. The ESR represents the value of the equivalent series resistance of the Al-Cap estimated from module 3 (proposed tool).

Figure 17 shows the response of the FDT when the converter is subjected to a load variation (from 20 A to 5 A at 2.5 ms).

The previous figure shows that during the transient, the maximum error was greater than 110%.

Figure 18 shows the response of the FDT when the converter is subjected to an input voltage variation (from 19 V to 9 V at 2.5 ms).
The previous figure shows that during the transient, the maximum error was greater than 40%.

In order to understand the meaning of previous errors in the context of an FDT, it is important to delve deeper into the issue of Al-Caps failures.

The Al-Caps present two types of failures: catastrophic and parametric failures. The catastrophic failures lead to the destruction of the component through a short or an open circuit. Thus, the capacitor completely loses its function. In the parametric failure, the capacitor does not completely lose its function; however, its electrical characteristics deteriorate (the ESR increases). Usually, capacitor manufacturers define the end life limit of Al-Caps as being when the ESR doubles with respect to its initial value [8]. The previous threshold defines a significant increase in the occurrence of a catastrophic failure.

In this way, using the proposed tool, it was possible to conclude that the analyzed FDT can lead to an error in assessing the health status of the Al-Caps during the transient regime. This situation is particularly serious in critical applications, which is why a much more conservative threshold is defined for the ESR [8].

6. Conclusions

This paper presented a tool that simulates the operation of a closed-loop SMNI DC-DC converter and, simultaneously, implemented an FDT in real time.

The proposed solution was implemented in Python, as it is a high-level open-source programming language oriented to numerical analysis, unlike MATLAB/Simulink, which is a proprietary language, and is for that reason much less flexible. In this particular, it is important to mention that proprietary languages makes the implementation of FDT much more difficult. By comparing the proposed tool with LTspice and experimental results, it was possible to prove its applicability. Therefore, it was demonstrated that the proposed tool allows the evaluation of the reliability of FDTs for SMNI DC-DC converters at different operating conditions. In this regard, it is important to highlight one of the great advantages of the proposed tool: it allows us to easily verify the deficiencies of the FDT without having to design an experimental prototype. Another important aspect of the proposed tool is the fact that it allows the evaluation of the applicability of the FDT in real time. In the case of the analyzed FDT, the estimated ESR value corresponds to the previous csp, which is in the order of the tens of µs. Therefore, it is possible do conclude that the system responds in real time.

The paper also presented an algorithm that can be very useful in the design of buck converters.

Author Contributions: Conceptualization, A.M.R.A.; Data curation, A.M.R.A.; Formal analysis, A.M.R.A. and A.J.M.C.; Funding acquisition, A.M.R.A. and A.J.M.C.; Investigation, A.M.R.A. and A.J.M.C.; Methodology, A.M.R.A.; Project administration, A.M.R.A. and A.J.M.C.; Resources, A.M.R.A.; Software, A.M.R.A.; Supervision, A.M.R.A. and A.J.M.C.; Validation, A.M.R.A.; Visualization, A.M.R.A. and A.J.M.C.; Writing—original draft, A.M.R.A.; Writing—review & editing, A.M.R.A. and A.J.M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Regional Development Fund (ERDF) through the operational program for competitiveness and internationalization (COMPETE 2020), under project POCI-01-0145-FEDER-029494, and by national funds through the FCT—Portuguese Foundation for Science and Technology, under projects PTDC/EEI-EEE/29494/2017, UIDB/04131/2020 and UIDP/04131/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

1. Narayanasamy, B.; Luo, F. A Survey of Active EMI Filters for Conducted EMI Noise Reduction in Power Electronic Converters. *IEEE Trans. Electromagn. Compat.* 2019, 61, 2040–2049. [CrossRef]

2. Song, Y.; Wang, B. Survey on Reliability of Power Electronic Systems. *IEEE Trans. Power Electron.* 2013, 28, 591–604. [CrossRef]

3. Zhang, J.; Yao, H.; Rizzoni, G. Fault Diagnosis for Electric Drive Systems of Electrified Vehicles Based on Structural Analysis. *IEEE Trans. Veh. Technol.* 2017, 66, 1027–1039. [CrossRef]

4. Khan, S.; Wen, H. A Comprehensive Review of Fault Diagnosis and Tolerant Control in DC-DC Converters for DC Microgrids. *IEEE Access* 2021, 9, 80100–80127. [CrossRef]

5. Jhu, J.; Lin, Y.; Chen, C. Single-Stage Interleaved Buck-Boost-LLC Resonant Converter with Wide Input Voltage Range for Railway and Medical Applications. In Proceedings of the IEEE International Future Energy Electronics Conference, Taipei, Taiwan, 16–19 November 2021.

6. Shi, H.; Wen, H.; Chen, G.; Bu, Q.; Chu, G.; Zhu, Y. Multiple-Fault-Tolerant Dual Active Bridge Converter for DC Distribution System. *IEEE Trans. Power Electron.* 2022, 37, 1748–1760. [CrossRef]

7. Faraji, R.; Ding, L.; Rahimi, T.; Keshht, M. Application of Soft-Switching Cell with Inherent Redundancy Properties for Enhancing the Reliability of Boost-Based DC–DC Converters. *IEEE Trans. Power Electron.* 2021, 36, 12342–12354. [CrossRef]

8. Amaral, A.; Sahraoui, M. Capacitors. In *Diagnosis and Fault Tolerance of Electrical Machines Power Electronics and Drives*; Cardoso, A.J.M., Ed.; IET-The Institution of Engineering and Technology: London, UK, 2018; Chapter 5; pp. 195–286.

9. Zhao, Z.; Davari, P.; Lu, W.; Wang, H.; Blaabjerg, F. An Overview of Condition Monitoring Techniques for Capacitors in DC-Link Applications. *IEEE Trans. Power Electron.* 2021, 36, 3692–3716. [CrossRef]

10. Stala, R.; Waradzyn, Z.; Bolmer, S. Input Current Ripple Reduction in a Step-Up DC–DC Switched-Capacitor Switched-Inductor Converter. *IEEE Access* 2020, 10, 19890–19904. [CrossRef]

11. Prag, K.; Woolway, M.; Celik, T. Data-Driven Model Predictive Control of DC-to-DC Buck-Boost Converter. *IEEE Access* 2021, 9, 101902–101915. [CrossRef]

12. Nupur Nath, S. Effect of Coupling on Discontinuous Conduction Mode of Coupled Inductor SIDO Boost Converter. *IEEE Trans. Power Electron.* 2022, 37, 4991–5002. [CrossRef]

13. Navamani, J.; Geetha, A.; Almakhles, D.; Lavanya, A.; Ali, J. Modified LUO High Gain DC–DC Converter with Minimal Capacitor Stress for Electric Vehicle Application. *IEEE Access* 2021, 9, 122335–122350. [CrossRef]

14. Ranjani, J.; Sheela, A.; Meena, K. Combination of NumPy, SciPy and Matplotlib/Pylab—A good alternative methodology to MATLAB—A Comparative analysis. In Proceedings of the 1st International Conference on Innovations in Information and Communication Technology, Chennai, India, 25–26 April 2019.

15. Amaral, A.; Cardoso, A. Using Python for the Simulation of a Closed Loop PI Controller for a Buck Converter. *Signals* 2022, 3, 313–325. [CrossRef]

16. Amaral, A.; Cardoso, A. Regulated Step-Up Power Supplies: Design and Simulation. In Proceedings of the 18th International Multi-Conference on Signals, Signals & Devices, Monastir, Tunisia, 22–25 March 2021; pp. 801–807.

17. Amaral, A.; Cardoso, A. Design, Analysis and Simulation of AC-DC Converters with Python. In Proceedings of the 3rd International Conference on Electrical Engineering Research, Yogyakarta, Indonesia, 29–30 July 2021; pp. 13–18.

18. Sun, Q.; Wang, Y.; Jiang, Y. A Novel Fault Diagnostic Approach for DC-DC Converters Based on CSA-DBN. *IEEE Access* 2018, 6, 6273–6285. [CrossRef]

19. Xia, Y.; Xu, Y. A Transferrable Data-Driven Method for IGBT Open-Circuit Fault Diagnosis in Three-Phase Inverters. *IEEE Trans. Power Electron.* 2021, 36, 13478–13488. [CrossRef]

20. Amaral, A.; Cardoso, A. A Simple Offline Technique for Evaluating the Condition of Aluminum–Electrolytic–Capacitors. *IEEE Trans. Ind. Electron.* 2009, 56, 3230–3237. [CrossRef]

21. Amaral, A.; Cardoso, A. An Economic Offline Technique for Estimating the Equivalent Circuit of Aluminum Electrolytic Capacitors. *IEEE Trans. Instrum. Meas.* 2008, 57, 2697–2710. [CrossRef]

22. Ghosh, A.; Banerjee, S.; Sarkar, M.; Dutta, P. Design and implementation of type-II and type-III controller for DC–DC switched-mode boost converter by using K-factor approach and optimisation techniques. *IET Power Electronics* 2008, 9, 938–950. [CrossRef]

23. Yang, X.; Xu, M.; Li, Q.; Wang, Z.; He, M. Analytical Method for RC Snubber Optimization Design to Eliminate Switching Oscillations of SiC MOSFET. *IEEE Trans. Power Electron.* 2022, 37, 4672–4684. [CrossRef]

24. Zhao, H.; Dalal, D.; Jørgensen, J.; Bech, M.; Wang, X.; Munk-Nielsen, S. Behavioral Modeling and Analysis of Ground Current in Medium-Voltage Inductors. *IEEE Trans. Power Electron.* 2021, 36, 1236–1241. [CrossRef]

25. Stas, A. Application of LTSpice Modeling to Vishay Temperature Sensors. Application Note-Document Number: 29170. Vishay BC Components. 2016. Available online: https://www.vishay.com/docs/29170/ltspice.pdf (accessed on 1 March 2022).

26. Wang, H.; Liserre, M.; Blaabjerg, F. Toward Reliable Power Electronics: Challenges, Design Tools, and Opportunities. *IEEE Ind. Electron. Mag.* 2013, 7, 17–26. [CrossRef]

27. Zhang, X.; Liu, Y.; Yue, K.; Wang, H. Online Capacitance Estimation Method in Buck Converters with Characteristic Frequency Injection for Optimal Sensitivity. In Proceedings of the 2022 IEEE Applied Power Electronics Conference and Exposition, Houston, TX, USA, 20–24 March 2022; pp. 319–323.

28. Li, C.; Yu, Y.; Yang, Z.; Liu, Q.; Peng, X. ESR Estimation for Aluminum Electrolytic Capacitor of Power Electronic Converter Based on Compressed Sensing and Wavelet Transform. *IEEE Trans. Ind. Electron.* 2022, 69, 1948–1957. [CrossRef]
29. Dang, H.-L.; Kwak, S. Review of Health Monitoring Techniques for Capacitors Used in Power Electronics Converters. *Sensors* 2020, 20, 3740. [CrossRef] [PubMed]

30. Ren, L.; Zhang, L.; Gong, C. ESR Estimation Schemes of Output Capacitor for Buck Converter from Capacitor Perspective. *Electronics* 2020, 9, 1596. [CrossRef]

31. Zhao, Z.; Lu, W.; Davari, P.; Du, X.; Iu, H.; Blaabjerg, F. An Online Parameters Monitoring Method for Output Capacitor of Buck Converter Based on Large-Signal Load Transient Trajectory Analysis. *IEEE J. Emerg. Sel. Top. Power Electron.* 2021, 9, 4004–4015. [CrossRef]

32. Laadjal, K.; Sahraoui, M.; Cardoso, A. On-Line Fault Diagnosis of DC-Link Electrolytic Capacitors in Boost Converters Using the STFT Technique. *IEEE Trans. Power Electron.* 2021, 36, 6303–6312. [CrossRef]

33. Amaral, A.; Cardoso, A. State condition estimation of aluminum electrolytic capacitors used on the primary side of ATX power supplies. In Proceedings of the 35th Annual Conference of IEEE Industrial Electronics, Porto, Portugal, 3–5 November 2009; pp. 442–447.