Decoupling Technique for Real-Time Simulation of Power Electronic Using Short Line

L.-A. Grégoire, Y. Bouzid, H. F. Blanchette, K. Al-Haddad

***Abstract*--This paper proposes a method to decouple and subdivide electrical circuits in order to achieve fast and accurate real-time simulation. Using this method, each state variable can be discretized using different solvers. By combining implicit and explicit ordinary differential equation solvers, state-space equations are decoupled while remaining accurate and stable. Unlike most decoupling methods proposed in previous literature, no artificial delay or supplementary states are added to decouple the system. Furthermore, this technique is meant to be implemented with already commercially available software by modifying only the point of coupling of the circuit. Finally, stability, accuracy and gain in computation speed are demonstrated through an example and poles study.**

***Keywords*: decoupling, tearing, real-time simulation, discrete system, micro-grid.**

I. Introduction

NOWADAYS, traditional and real-time simulation has become an essential development tool for engineers. Simulation must always be accurate and stable, but when it comes to real-time simulation it also requires a small computation time. Different aspects can be considered to reduce computation time, from the method used to generate the circuit equations [1], the integration method or the type of hardware used. As discretization of the equations can take time to complete, one common method is to pre-compute the different sets of equations from the circuit and store them in memory. Circuits containing a large number of states or multiple power switches can eventually lead to a memory usage too large for hardware available. Therefore, because of the increasing complexity of simulated system, and also the development of parallel computing, decoupling of electrical circuit equations has been the focus of many researches. Decoupling electrical circuits reduces the size of their equations making computation more time efficient. Also, isolating power switches through decoupling techniques reduces the size of the equations that need to be stored in memory. In the case of large power system, propagation delay in distribution line can be used for decoupling purposes [2]. Unfortunately, this method cannot be applied for short-lines where propagation delays are smaller than the simulation time-step, unless parasitic shunt capacitances are added [3]. Another method for a decoupling system is to add a delay on a slow varying state, such as a capacitor on a DC-bus [4]. This method works well on circuits with DC-buses but is limited this type of topology in practical applications.

This paper proposes to use implicit and explicit solvers to decouple electrical circuits. In section II, the operational substitution technique is summarized for the most common discrete integration. The multi-solver technique is presented in section III followed by a numerical example in section IV showing the implementation and the real-time simulation results for traditional and the proposed method. Finally conclusions are drawn in section V.

II. Operational Substitution

Keeping in mind that an integration is the area under a signal, Fig. 1 shows three of the most popular approximations:

- Forward Euler (FE)
- Trapezoidal (TR)
- Backward Euler (BE)

Power electronic systems are highly non-linear, and for this reason single-step integration methods are preferred. In Fig. 1, the filled region represents the local truncation error (LTE) yield by each method. Naturally, a smaller T increases accuracy of the results as it reduces the area of the region. FE is known for being under-damped, with its error under the expected value, BE for being over-damped, with the error over the expected value, and TR, with its error distributed over and under the expected value, usually gives the most accurate results.

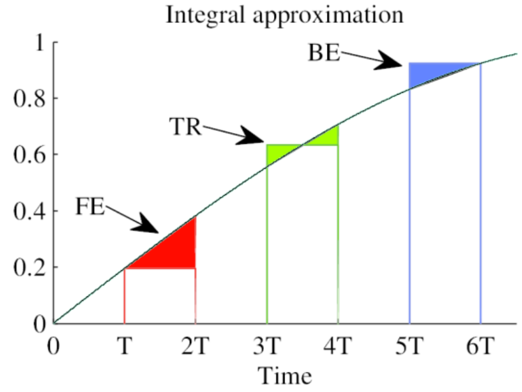

Fig. 1. Most common one-step integration.

This work was supported in part FQRNT, CRSNG, OPAL-RT technologies and Canada Research Chairs (CRC).

L.-A. Grégoire is with the Department of Electrical Engineering, École de technologie supérieure, Montréal, QC Canada (e-mail of corresponding author: luc-andre.gregoire.1@ens.etsmtl.ca).

C Dufour is with OPAL-RT Technologies Inc., Montréal, QC Canada (e-mail:Christian.dufour@opal-rt.com)

H. F. Blanchette is with the Department of Electrical Engineering, École de technologie supérieure, Montréal, QC Canada (e-mail:handy.fortin-blanchette@etsmtl.ca).

K. Al-Haddad is with the Department of Electrical Engineering, École de technologie supérieure, Montréal, QC Canada (e-mail:kamal.al-haddad@etsmtl.ca).

Paper submitted to the International Conference on Power Systems Transients (IPST2015) in Cavtat, Croatia June 15-18, 2015

This has also been demonstrated mathematically using Taylor series yielding the following equations (1) to (3) for the LTE of the different integration methods presented. Further discussions and numerical analyses are discussed in [6].

$$LTE_{FE} = \frac{1}{2}T^2\ddot{x} + O(T^3) \quad (1)$$

$$LTE_{TR} = \frac{1}{6}T^3\dddot{x} + O(T^4) \quad (2)$$

$$LTE_{BE} = -\frac{1}{2}T^2\ddot{x} + O(T^3) \quad (3)$$

When electrical circuits are represented by state-space equations, the Laplace variable s can be replaced by an expression in *z* domain using those methods; this approach is called operational substitution. Equations (4)-(6) give the approximation of s for different methods shown in Fig. 1.

$$s_{FE} = \frac{z-1}{T} \quad (4)$$

$$s_{TR} = \frac{2(z-1)}{T(z+1)} \quad (5)$$

$$s_{BE} = \frac{z-1}{Tz} \quad (6)$$

Although equations (4) to (6) can be obtained graphically, they could also be obtained by approximating (7) using Taylor series.

$$s = \frac{ln(z)}{T} \quad (7)$$

If (7) is evaluated using another approximation method, like Padé [7], various operational substitution methods can be obtained. The three methods shown in this paper are associative, meaning that they can replace either *s* or s^{-1}. Using higher order of the Taylor series or Padé approximation can result in non-associative, which can only replace s^{-1}. Interested readers can refer to chapter 3 of [5]. Also, in nodal admittance based EMT solvers, using generalized branch definition described by state-space equations enable the use of high-order L-stable matrix exponential approximants during the formation of the solver discrete equations [8].

Using operational substitution, each state of a system can be discretized using a different method. This may seem counter intuitive but by combining an explicit method, like FE, with an implicit one, like BE, allows decoupling states from a system and resolving them in parallel. When doing so, one must be very careful since explicit methods can lead to numerical instability. This will be demonstrated in the next section, as well as the method to verify the error introduced by the discretization.

III. Multi-solver method

In this paper, the term solver refers to the integration method used during discretization. The term multi-solver refers to using different approximations, or operational substitution, within one system. Taking the example of the second order state-space system in (8), with the continuous matrices A_c and B_c, can be discretized using a multi-solver method, obtaining discrete matrices A_d and B_d. The first step is to rewrite each state as an individual equation, yielding (9) for the state x_1 and (10) for the state x_2.

$$\begin{bmatrix} x_1 s \\ x_2 s \end{bmatrix} = \overbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}^{A_c} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u \quad (8)$$

$$x_1 s = a_{11}x_1 + a_{12}x_2 + b_1 u \quad (9)$$

$$x_2 s = a_{21}x_1 + a_{22}x_2 + b_2 u \quad (10)$$

Secondly, Laplace variable *s* in (9) is replaced by the expression in z of the equation (4) and the s from (10) by the one from (6); this will then be referred as a FEBE solver. Using algebra, variables multiplied by *z* are isolated, and equations (11) and (12) are obtained.

$$x_1 z = (Ta_{11} + 1)x_1 + Ta_{12}x_2 + Tb_1 u \quad (11)$$

$$x_2 z = \left(\frac{T^2 a_{21} a_{11} + Ta_{21}}{Ta_{22} - 1}\right) x_1 - \left(\frac{T^2 a_{21} a_{12} + 1}{Ta_{22} - 1}\right) x_2 - \left(\frac{T^2 a_{21} b_1}{Ta_{22} - 1}\right) u - \left(\frac{Tb_2}{Ta_{22} - 1}\right) uz \quad (12)$$

In (12), there is now an input u multiplied by *z*, which is inherent to implicit method; it requires the input at the next step to compute state value at the next step. Equation (11), being an explicit solution, only needs the present values of the system and therefore no *z* is found on the right side of the equation. Finally, those equations are translated in a discrete state-space system using recurrent equations in (13).

$$\begin{bmatrix} x_{1_n} \\ x_{2_n} \end{bmatrix} = \overbrace{\begin{bmatrix} (Ta_{11} + 1) & Ta_{12} \\ \left(\frac{T^2 a_{21} a_{11} + Ta_{21}}{Ta_{22} - 1}\right) & -\left(\frac{T^2 a_{21} a_{12} + 1}{Ta_{22} - 1}\right) \end{bmatrix}}^{A_d} \begin{bmatrix} x_{1_{n-1}} \\ x_{2_{n-1}} \end{bmatrix} + \begin{bmatrix} Tb_1 \\ -\left(\frac{T^2 a_{21} b_1}{Ta_{22} - 1}\right) \end{bmatrix} u_{n-1} + \begin{bmatrix} 0 \\ -\left(\frac{Tb_2}{Ta_{22} - 1}\right) \end{bmatrix} u_n \quad (13)$$

Looking at the matrices values of A_d, B_{d1} and B_{d2} in (13), the following observations can be made.

- The system now has two inputs, u^n and u^{n+1}, because of the implicit equation.
- Variable x^n_1, being explicit, only requires the input value at the present step, u^n.
- Values required to compute x^n_1 in (13) are only dependent of the values used to compute x_1s in (8) and time-step T.
- Values required to compute x^n_2 in (13) are dependent of the values used to compute both x_1s and x_2s from (8) and of the time-step T.

Stability of the system using FEBE can be demonstrated, like any discretized system, by studying the eigenvalues of the matrix A_d. Furthermore, eigenvalues or poles of (8) can be evaluated as discrete ones using (14)

$$z = e^{sT} \quad (14)$$

Replacing *s* in (14), by the numerical values of the poles from A_c, the exact discrete poles are obtained. This is demonstrated through a numerical example, with its implementation in Matlab/Simulink SimPowerSystems (SPS), in the next section.

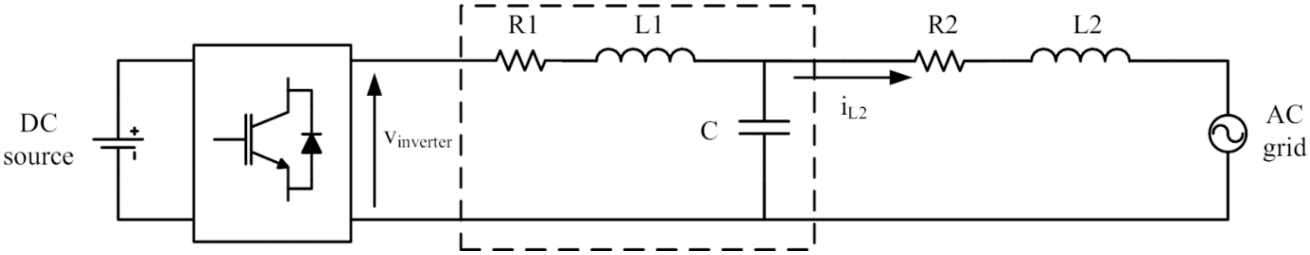

Fig. 2. Single phase inverter circuit with filter.

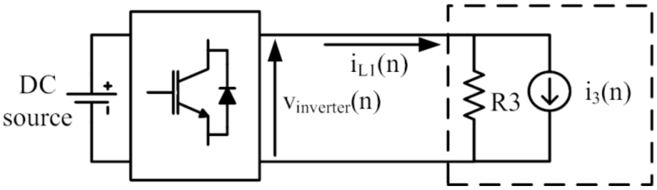

Fig. 3. Left side of the single phase inverter circuit with filter.

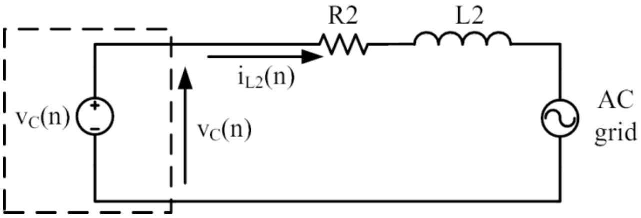

Fig. 4. Right side of the single phase inverter circuit with filter.

In this section, an example of a single phase inverter is feeding the grid through an LCL filter [9], as shown in Fig 2. Parameters used for simulation are given in Table I.

TABLE I
NUMERICAL EXAMPLE SIMULATION PARAMETERS

L1	3.50	mH
R1	0.10	Ω
C	15	μF
L2	1.50	mH
R2	0.05	Ω
PWM	5	kHz
SAMPLING TIME	10	μS

A. Proposed Implementation

The first step in decoupling the system is to choose state-variables used. In this example, the inverter will be decoupled from the grid using the elements inside the dash-box of Fig. 2; current from L1 and the voltage from C. $v_{inverter}$ is the voltage measured on the AC-side of the inverter of Fig. 2 and i_{L2} is the current flowing in the inductance L2. Considering only those two state-variables yields the continuous state-space system of (15) which becomes (16) once discretized.

$$\begin{bmatrix} v_C s \\ i_{L1} s \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L1} & -\frac{R1}{L1} \end{bmatrix} \begin{bmatrix} v_C \\ i_{L1} \end{bmatrix} + \begin{bmatrix} 0 & -\frac{1}{C} \\ \frac{1}{L1} & 0 \end{bmatrix} \begin{bmatrix} v_{inverter} \\ i_{L2} \end{bmatrix} \quad (15)$$

$$\begin{bmatrix} v_{C_n} \\ i_{L1_n} \end{bmatrix} = \begin{bmatrix} 1 & 0.6667 \\ -0.0029 & 0.9975 \end{bmatrix} \begin{bmatrix} v_{C_{n-1}} \\ i_{L1_{n-1}} \end{bmatrix} + \begin{bmatrix} 0 & -0.6667 \\ 0 & 0.0019 \end{bmatrix} \begin{bmatrix} v_{inverter_{n-1}} \\ i_{L2_{n-1}} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.0029 & 0 \end{bmatrix} \begin{bmatrix} v_{inverter_n} \\ i_{L2_n} \end{bmatrix} \quad (16)$$

Using the same method as in section III, v_C and i_{L1} are discretized using FE and BE respectively. Again, the inputs at the previous step and at the present step are required to solve the system. This complicates the process since those are the measured values from SPS. These values are only available once v_C and i_{L1} are evaluated, since they are required to compute $v_{inverter}$ and i_{L2}. This is solved by adding the element from the coupling matrix of (16) to the SPS circuit, as shown in Fig. 3. The value of i_3, in Fig. 3, is the value of i_{L1} at the present time, minus the contribution of $v_{inverter}$, or by the previous value of the different measurements of the circuit, as shown in equation (17). Adding R3 in shunt with the current source i_3, allows the computation of i_{L1} at the present time with the contribution of $v_{inverter}$. Likewise, since no element of the coupling matrix contributes to compute v_C, only a controlled voltage source is required in the right side circuit of Fig. 4.

$$i_{3_n} = i_{1_n} - 0.0029 v_{inverter_n}$$
$$i_{3_n} = -0.0029 v_{C_{n-1}} + 0.9975 i_{L1_{n-1}} + 0.0019 i_{L2_{n-1}} \quad (17)$$
$$R3 = \frac{1}{0.0029}$$

The circuit of Fig. 2 in now decoupled into two sub-circuits that can be solved in parallel. In power electronics, when a switching event occurs, it requires recomputing the state-equation, since the circuit is modified. In this case, switching event on the left side circuit will not require modifying any equation from the right side circuit. This could also be applied to hardware-in-the-loop (HIL) and power-HIL (PHIL) simulation. The circuit of Fig. 3 could be simulated and Fig. 4 would include the real-hardware, for example. This is a major gain, but only if the results remain accurate, and they do, as demonstrated in the next subsection.

B. Simulation Validation

In this subsection, values of the three state-variables, i_{L1}, v_C and i_{L2}, are used to determine accuracy of the different methods. An open-loop control is used to ensure that errors inherent to the discretization methods are not compensated by the controller. The following four implementations are studied:

- A reference without any decoupling (Ref)
- The proposed method (Case 1)
- Replacing L1 by STUBLINE method [2] (Case 2)
- Replacing C by STUBLINE method (Case 3)

Note that a STUBLINE is a Bergeron distributed parameter line model with losses in which capacitance or inductance is adjusted to obtain exactly one-time step of propagation delay sufficient to decouple the equations on each side of it in 2 parts. Phase and amplitude of the modulating signal are adjusted to achieve nominal voltage and current. Fig. 7 shows current in inductance L1. The maximum relative error is obtained with the method from case 2 with a peak value of 6%, and case 1 and 3 have a peak value of 3% and 1% respectively. Furthermore, results from case 2 are highly oscillating, which is due to the STUBLINE method as it will be discussed later.

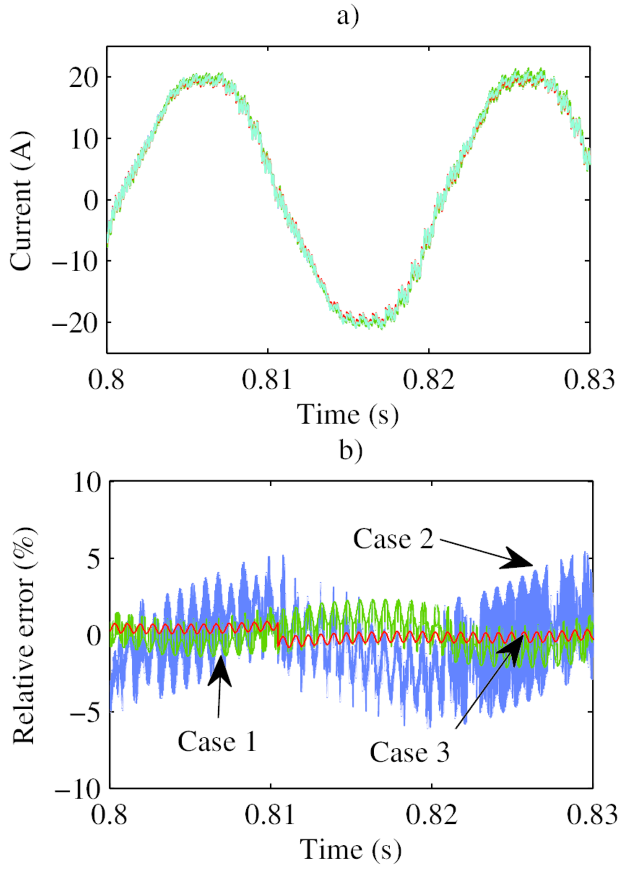

Fig. 5. Current in L1 a) for the four methods and b) relative error.

Results for i_{L2} are shown in Fig. 6, and similar absolute errors are achieved but without the high frequency oscillation. Unlike the previous state-variable, i_{L1}, or the capacitor voltage, v_C, this state is computed by the commercial software used. This means that the exact same method is used for its discretization.

So far, case 3 has yielded the most accurate results, but since the STUBLINE method is used, fast oscillation should be expected when observing v_C. Fig. 7 shows such oscillation for case 3 where the absolute error now reach 35%. Peak absolute errors for case 1 and 2 are 5% and 12%.

It is important to note that the relative error varies faster than the fundamental signal. This would mean that the relative errors are caused by higher harmonics, which could be filtered and have little impact on a controller under test.

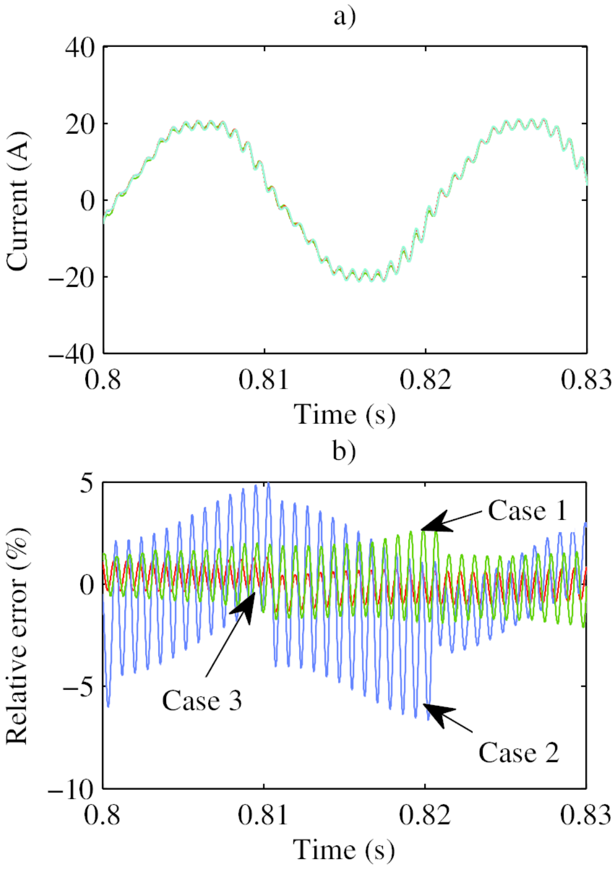

Fig. 6. Current in L2 a) for the four methods and b) relative error.

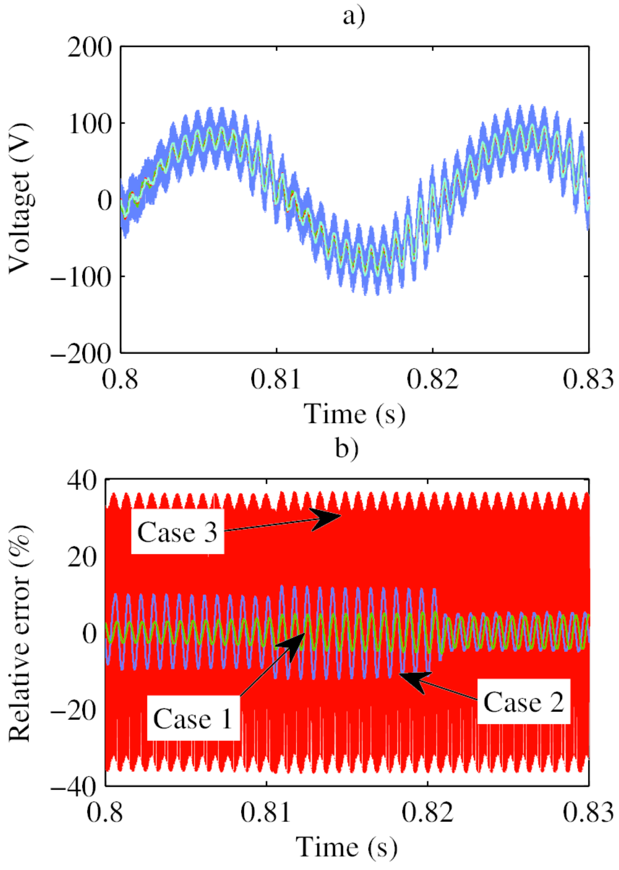

Fig. 7. Voltage on C a) for the four methods and b) relative error.

A. Harmonics Content Validation

Applying the fast Fourier transform to the different signals from Fig. 5 to Fig. 7, shows that relative errors are mainly caused by the higher harmonic. Table II shows the harmonics with the highest amplitude for the reference and their relative errors with the three different cases for the current in L1. Similar results are shown in Table III and IV for the current in L2 and voltage across C. The fundamental frequency is 50 Hz, the 25th harmonic is the natural frequency of the filter, the 100th is due to the carrier frequency and the 1000th to the sampling time of the simulation. As expected from the graphical results, relative errors for the 1st harmonic of the different method are low. For the 1000th one, since the reference is always close to 0, the relative errors become very big as soon as there is a slight variation. But when it becomes much larger than 100%, this is caused by the STUBLINE method. By observing the pole location for each method, even the error on the 25th harmonic could have been expected as demonstrated in the next subsection.

TABLE II
FAST FOURIER TRANSFORM FOR I_{L1}

HARMONIC	REF	CASE 1	CASE 2	CASE 3
1	18.89 A	1.10 %	1.11 %	0.11 %
25	0.40 A	5.46 %	11.56 %	0.13 %
100	0.67 A	1.19 %	2.41 %	0.05 %
1000	≈0 A	23.75 %	> 100 %	4.88 %

TABLE III
FAST FOURIER TRANSFORM FOR I_{L2}

HARMONIC	REF	CASE 1	CASE 2	CASE 3
1	18.90 A	0.02 %	1.13 %	0.12 %
25	1.03 A	4.96 %	11.34 %	0.09 %
100	0.03 A	3.49 %	2.33 %	2.35 %
1000	≈0 A	2.16 %	1.02 %	0.80 %

TABLE IV
FAST FOURIER TRANSFORM FOR V_C

HARMONIC	REF	CASE 1	CASE 2	CASE 3
1	74.06 V	≈0 %	1.13 %	≈0 %
25	12.17 V	5.06 %	11.30 %	0.03 %
100	1.48 V	2.67 %	2.44 %	2.24 %
1000	≈0 V	1.65 %	22.93 %	> 100 %

B. Pole Location Validation

Observations made from the simulation results could also have been extrapolated from analyzing the poles of the different method. Table V shows the different poles for each method used. For example, when using the STUBLINE method, a parasitic state is added, adding an extra pole; which is responsible for the fast oscillation at every time-step. Fig. 8 shows the right side of the unity circle. All the poles are within the unity circle which ensures system stability. Poles from the reference and from the case 1 are superimposed. The pole on the real axe, Fig. 8 shows that case 2 and 3 would have a different damping, either faster or slower than the reference and case 1. Also, since the poles with an imaginary part are different, the oscillation frequency and amplitude would be different, because they have different angles and absolute values.

TABLE V
POLES FOR DIFFERENT METHODS

REF	CASE 1	CASE 2	CASE 3
0.9994	0.9994	0.9996	0.9980
0.9965± *i*0.0796	0.9965± *i*0.0796	0.9941± *i*0.0730	0.9964± *i*0.0561
-	-	-0.9992	-0.9984

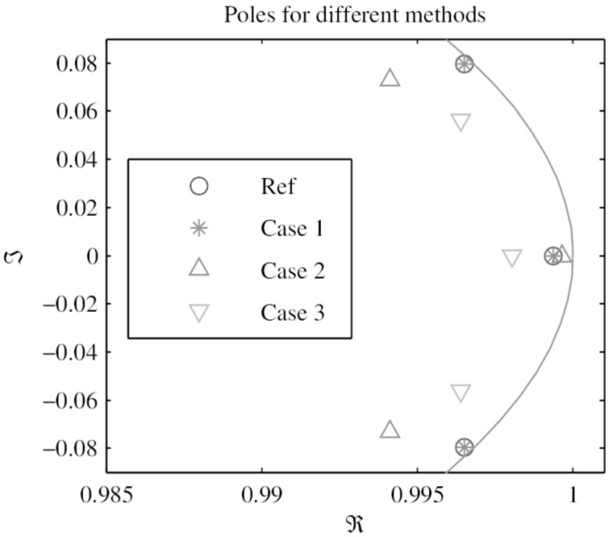

Fig. 8. Poles location for different methods.

Stability and accuracy of the proposed method have been demonstrated through simulation results and poles analysis. Studying the poles of the system allows identifying every possible pole, even those that might not be excited in every simulation scenario. Furthermore, knowing the exact reference pole location, using equation (14), accuracy of the different methods can be compared. Also, one must keep in mind that when the main goal of decoupling the circuit is to achieve real-time simulation capability, for which application a relative error of 5 % for higher harmonic is well within an acceptable range. [10-12]

C. Real-time Simulation

Real-time simulation results were obtained using the real-time platform OP4500 from OPAL-RT Technologies [13] using a 2.4GHz i7 quad-core processor. OPAL-RT uses MATLAB/Simulink SimPowerSystems library to simulate power electronic circuits but also offers an additional library to achieve parallel/real-time simulation. The reference model is computed using a single core and case 1 used two cores where the circuit is divided as shown in Fig. 3 and 4. Case 3 and 4 are also computed using two cores and OPAL-RT STUBLINE in their library. Acceleration factors are calculated on the average computation time for each method and is presented in table VI.

TABLE VI
ACCELERATION FACTOR FOR DIFFERENT METHODS

	REF	CASE 1	CASE 2	CASE 3
Comp. time	1.30μs	1.00μs	1.11μs	1.20μs
ACC. FACTOR	100%	130%	108%	117%

Difference between the methods can be explained by the number of signals exchanged between the two cores and the size of each decoupled circuit. Because of the simplicity of the circuit, gains are marginal. Acceleration factors would be much larger for larger circuits since the computational burden grows exponentially with the number of states. The proposed method is slightly faster since it has fewer signals exchanged than the one using STUBLINE method.

IV. Conclusions

This paper proposes a decoupling technique for power electronic circuits. This method can be applied to commercially available software with little effort. Simulations made in this paper using traditional decoupling methods were proven less accurate than the one proposed. The z-domain pole location analysis of the proposed method also supported this claim because its resulting poles are closer to the reference case than STUBLINE-based methods. This technique proved most useful in real-time simulation applications, where faster-parallel computing is required. This method could also be used in hardware-in-the-loop (HIL) and power-HIL (PHIL) simulation where decoupling is needed between the simulated model and the physical hardware. Future work will aim to apply this method for multi-rate systems, where the integration time-step may vary between the different states. Another future work will be to automate the application of the new method in larger systems. In particular, it will be desirable to refine the method to find the best decoupling points.

V. References

[1] L. A. Grégoire, J. Bélanger, C. Dufour, H. Blanchette, K. Al-Haddad *et al.*, “Real-time simulation of modular multilevel converters (mmcs),” in *Power electronics for renewable energy systems, transportation and industrial applications*. John Wiley & Sons, Ltd, pp. 591–607.

[2] N. Watson and J. Arrillaga, *Power systems electromagnetic transients simulation*. Iet, 2003, no. 39.

[3] M. Hong, S. Horie, Y. Miura, T. Ise, and C. Dufour, “A method to stabilize a power hardware-in-the-loop simulation of inductor coupled systems,” in *Int. Conf. on Power Systems Transients (IPST2009) in Kyoto, Japan*, 2009.

[4] C. Dufour and J. Belanger, “A pc-based real-time parallel simulator of electric systems and drives,” in *Parallel Computing in Electrical Engineering, 2004. PARELEC 2004. International Conference on*, Sept 2004, pp. 105–113.

[5] T. T. Hartley, G. O. Beale, and S. P. Chicatelli, *Digital Simulation of Dynamic Systems: A Control Theory Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1994.

[6] F. N. Najm, *Circuit simulation*. John Wiley & Sons, 2010.

[7] G. Wanner and E. Hairer, “Solving ordinary differential equations ii,” 1991.

[8] C. Dufour, J. Mahseredjian, J. Bélanger, J. L. Naredo, “An Advanced Real-Time Electro-Magnetic Simulator for Power Systems with a Simultaneous State-Space Nodal Solver”, IEEE/PES T&D 2010 - Latin America, São Paulo, Brazil, Nov. 8-10, 2010

[9] E. Twining and D. G. Holmes, “Grid current regulation of a three-phase voltage source inverter with an lcl input filter,” *Power Electronics, IEEE Transactions on*, vol. 18, no. 3, pp. 888–895, 2003.

[10] H. Blanchette, T. Ould-Bachir, and J.-P. David, “A state-space modeling approach for the fpga-based real-time simulation of high switching frequency power converters,” *Industrial Electronics, IEEE Transactions on*, vol. 59, no. 12, pp. 4555–4567, Dec 2012.

[11] A. Benigni and A. Monti, “A parallel approach to real-time simulation of power electronics systems,” *Power Electronics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[12] L.-A. Gregoire, W. Li, J. Belanger, and L. Snider, “Validation of a 60-level modular multilevel converter model-overview of offline and realtime hil testing and results,” 2011.

[13] OPAL-RT Technologies Inc., *OP4500 RT-LAB-RCP/HIL system user guide*, 2014, available from http://www.opal-rt.com.