Electrical Power and Energy Systems 99 (2018) 650–658

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

A new approach for FPGA-based real-time simulation of power electronic system with no simulation latency in subsystem partitioning

Chen Liu*, Rui Ma, Hao Bai, Franck Gechter, Fei Gao

FEMTO-ST (UMR CNRS 6174), Energy Department, Univ. Bourgogne Franche-Comte, UTBM, Rue Thierry Mieg, F-90010 Belfort Cedex, France

ARTICLE INFO

Keywords:
Real-time simulation
Circuit partitioning
Power electronic system
Parallel calculation
Traction system
FPGA

ABSTRACT

In real-time Hardware-in-the-Loop (HIL) test applications for power electronic systems, the main hurdle is to tackle with the mathematical models of variable topology of complex and high frequency driven converter. The most widespread solution is to separate the whole system into subsystems. However, partitioning method usually introduces simulation time step latency between different subsystems, which causes numeric instabilities especially when stiff situation occurs. In this paper, we propose a novel parallel simulation approach which has no time step latency in the whole system division, from which a numerically stable system modeling can be realized. Its numerical accuracy of the solution, the architecture design, and the issue pertaining to the parallel calculation are discussed in detail in this paper. The pertinence of the developed solution is also tested using a case study relating to a traction system power electronic application. For this case study, implementations are made both on a 3 GHz Xeon CPU of RT LAB real-time simulator with a 2 μs simulation step and a Field Programmable Gate Arrays (FPGA) Kintex-7 embedded in National Instruments FlexRIO PXIe-7975 enabling a simulation step below 50 ns. Besides, comparison with results obtained from Simpower system in Matlab allows to evaluate the accuracy of our proposed modeling approach.

1. Introduction

For a real-time system, the correctness of the behavior does not only depend on the result of the computation, but also on the physical time required to produce it. One example of such a system is a Hardware-in-the-Loop test setup, in which one or several power electronic devices are simulated using an embedded system [1]. In this context, the feature of power electronic system is simulated using embedded system. Nowadays, real-time simulation provides an effective substitute, at the early design stages, to the expensive downscale approaches. It works as an effective tool allowing the device and system being tested and controlled without building the physical system first. In order to ensure the accuracy and consider the complexity of the power electronic topology with high switching frequency, a very small simulation step (nanoseconds to microseconds) is generally required for power electronic system real-time simulation [2].

Broadly speaking, there are two ways to tackle with this requirement: using specific hardware or improving the simulation modeling method. Between these two possibilities, using a hardware solution with up to date processors is the most efficient. Using central processing unit (CPU) or graphics processing unit (GPU), it is possible to simulate within a reference time of several microsecond. Moreover, simulation step can reach nanosecond level with FPGA [3]. As compared to power systems, in which a 50 μs time-step is sufficient for simulating electrical grid elements, an accurate power electronic system simulation with a PWM control strategy should be under 1 μs [4]. Some papers [5] discuss interpolation methods to reduce the effect of multi-switching events, but the involved algorithms are complex. The most efficient way is parallel calculation based on FPGA to realize a simulation in nanoseconds.

To this end, complex power electronic systems are usually decomposed into smaller subsystems. Both fixed-rate and multi-rate strategy are used. In fixed-rate simulation, state-space equation method and latency insertion method (LIM) can be found in literature. They both use continuous first order functions of time, but they treat semiconductor device differently. Compared with idealized switch characteristic in state-space method, LIM methods formulate update equations for voltages and currents based on branch inductors and shunt capacitors. It can be used to simulate large networks [6]. Since a nanosecond simulation step is required, LIM method not only introduces possible circuit simulation oscillations but also can result in model numerical instability for a real-time implementation. The other method is Multi-rate (MR) approach, it uses a combination of iterative implicit and explicit solvers to eliminate these artificial delays and thus reduce

* Corresponding author.
E-mail address: chen.liu@utbm.fr (C. Liu).

https://doi.org/10.1016/j.ijepes.2018.01.053
Received 21 August 2017; Received in revised form 26 January 2018; Accepted 30 January 2018

instability issues. In paper [7], it divides the whole system into subsystem by applying an explicit Forward Euler integration to selected large energy storage elements. Then an implicit Backward Euler is used to ensure its stability. And in paper [8], it uses transmission line as a separator. A combination of back forward and forward Euler solvers is used so as to calculate the state of the transmission line from subsystem B to subsystem A. In their work, there is no artificial delay introduced in the data communication from B to A, however extra calculation burden is needed.

The drawback of previous method is that calculations are executed in series. The partitioning element first calculates with the explicit method, and then it applies implicit method to the rest subsystems. Moreover, the computation time depends largely on the size of the power electronic systems.

In contrast, this paper proposes a predictor-corrector circuit modeling approach, which utilizes the particular simplification in the predictor step. With this method, the circuit partitioning scheme is performed to reduce the computation power requirement and the memory cost. Relative work have reported converter simulation below 500 ns in a FPGA environment [9,10], but it considers only the converter, no other circuit elements or electrical motor model is considered. The main objective of this work is to accelerate the calculation speed in FPGA. This paper describes the basic implementation for the whole power electronic system which includes one rectifier, one inverter and one motor. The proposed method presents the following advantages:

1. It can achieve small simulation time step as low as 50 ns using FPGA. There is no need to consider the inter-simulation time step switching event.
2. The solver can remain stable even for a stiff system, like electrical machine.

This paper is organized as follows: in Section 2 presents the predictor-corrector method, its stability, as well as its parallel calculation method. Section 3 gives details about the circuit partitioning method used in our approach. Its performance is then validated through a case study of electrical traction system. For benchmarking, our model is compared with a Simpower System (SPS) model. The model is also implemented in a Kintex-7 FPGA embedded in FlexRIO PXIe-7975. Finally, Section 5 concludes this paper and provides some ideas for future work.

2. A parallel state-spcace approach

The circuit modeling approach proposed in this paper relies on two aspects. The first one is the discretization of state-space equation; the second is the combination of implicit and explicit solving methods.

2.1. The Predictor-Corrector method

For most cases, the power electronic system can be expressed as a state-space equation $y' = f(t,y)$ with initial values $y(t_0)$. Discretization of the original continuous system and numerical methods are utilized as an approximation of the solution shown in Eq. (1).

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(t,y(t))dt \tag{1}$$

In order to solve this formula at the time (t_{n+1}), we need to know the value f_{n+1} of the derivative function $f(t,y(t))$ at time (t_{n+1}). The solution process is a prescribed function of values at certain previous time point.

It is known that implicit method is more stable and accurate than explicit method [11]. Since ~~an~~ unknown numbers of iteration could cause overrun during real time simulation, we usually utilize a fixed one. A common strategy is the predictor and corrector method as follows.

P-step: Compute the predictor $\hat{y}_{n+1}$ by an explicit numeric solver method;

C-step: Apply the $\hat{f}_{n+1}$ using an implicit method to obtain the corrector y_{n+1};

Define h as simulation step, the combination of Backward and Forward Euler solver can be expressed in Eq. (2).

$$\begin{aligned} &\mathbf{P-step}: \hat{y}_{n+1} = y_n + h\cdot f(t_n,y_n) \\ &\mathbf{C-step}: y_{n+1} = y_n + \tfrac{h}{2}\cdot(f(t_n,\hat{y}_{n+1}) + f(t_n,y_n)) \end{aligned} \tag{2}$$

Its sequence can be seen in Fig. 1, the black curve represents the basic time line. It has two computation processes: Predictor step (blue curve) and Corrector step (black curve). Corrector step uses value $\hat{y}_{n+1}$ (at the time point t_n, red[1] circle) which depends on the result of Predictor step. The global time step h is the sum of the calculation time of these two steps.

If we can calculate both the predictor and corrector at the same time, the calculation time can be shortened by a factor of two. To accomplish this, we can take advantage of the "multistep computation" between the different equations involved in the process. That means, in our modified Predictor Corrector method, the Predictor is calculated one additional time step ahead, so that its value can be directly used for Corrector calculation at that time step.

As shown in Fig. 2, the new proposed method calculates the predictor value $\hat{y}_{n+2}$ at time point t_n within the process solving y_{n+1}. In the next step t_{n+1}, when we estimate the value of y_{n+2}, the value of $\hat{y}_{n+2}$ is already known from last step. As a result, $\hat{y}_{n+2}$ and y_{n+1} are known at the same time. This method can be effectively used when the simulation time step is relatively small, which is the case for power electronic system models.

Thus, the modified process can be written in Eq. (3),

$$\begin{aligned} &\mathbf{P-step}: \hat{y}_{n+2} = y_n + 2\cdot h\cdot f(t_n,y_n) \\ &\mathbf{C-step} y_{n+1} = y_n + \tfrac{h}{2}\cdot(f(t_n,\hat{y}_{n+1}) + f(t_n,y_n)) \end{aligned} \tag{3}$$

Proceeding the same way, we can get a different expression applying different discretization method.

2.2. Numerical stability analysis of the proposed method

The universal state-space function can be expressed as $\frac{d}{dt}x = Ax(t) + bu(t)$, A is state Matrix and B is Input-to-state matrix. Let's considered a coupled two level system in Eq. (4),

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 & 0 \\ 0 & b_2 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{4}$$

where x1 and x2 are two coupled variables in one system. Since variables are coupled, the function f is affected by both variables x1 and x2. In order to decouple the system, we split them between the two state variables by introducing the variables from predictor step and suppose $x_1 \approx \hat{x}_1$ and $x_2 \approx \hat{x}_2$:

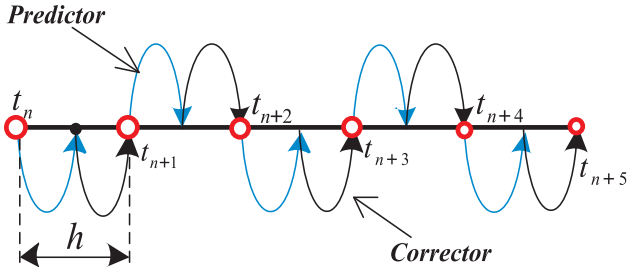

Fig. 1. Execution timeline of Predictor Corrector method.

[1] For interpretation of color in Figs. 1, and 13–15, the reader is referred to the web version of this article.