Modeling Partial Differential Equations in VHDL-AMS

Pavel V. Nikitin, C.-J. Richard Shi, and Bo Wan
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500, USA.
Email: {nikitin, cjshi, wanbo}@ee.washington.edu

***Abstract*— This paper discusses a problem of modeling distributed physics effects described by partial differential equations (PDE's) in VHDL-AMS, a powerful modeling language for mixed-signal systems. First, we summarize the requirements for PDE support. Second, we demonstrate with the example of a distributed transmission line how to model PDE's in an existing VHDL-AMS by applying spatial discretization to system equations. Third, we propose a language extension needed to support PDE's. Our work should be perceived as a first step towards an accurate description and modeling of coupled multi-physics systems in VHDL-AMS.** [1]

I. Introduction

An IEEE standard, VHDL-AMS is a powerful hardware description language that allows one to model the behavior of mixed-signal (analog and digital) and multi-physics (mixed electrical, electromagnetic, thermal, mechanical, etc.) systems [1], [2], [3]. VHDL-AMS specifies what system of equations is to be used at each simulation time but the choice of a solution technique is left to an implementor. Continuous parts of the system can currently be described in VHDL-AMS using differential and algebraic equations (DAE's). Due to the complexity, the support for partial differential equations (PDE's) was intentionally left out in VHDL-AMS [4]. This limits the accurate modeling of system blocks that include distributed physics effects.

Such blocks are currently modeled in VHDL-AMS exclusively via equivalent circuit approach [5] or reduced order models [6] imported from an accurate solution obtained by an external domain-specific simulator [7].

A proposition to extend the capability of VHDL-AMS to support full-wave modeling of distributed RF and microwave components has recently appeared in the literature [8]. This is a challenging task and to the best of our knowledge no other publications have followed yet. The only other published work in this direction was an earlier paper by Zhou et al. [9] who solved a steady-state PDE in VHDL-AMS with a neural network algorithm.

The purpose of this paper is to define the first step towards modeling distributed physics effects in VHDL-AMS – to introduce a language support for PDE's. The importance of such support in a universal hardware description language cannot be overestimated and has been discussed earlier during the development of a microwave hardware description language (MHDL) [10], [11].

[1]This research was supported by DARPA NeoCAD Program under Grant No. N66001-01-8920 and NSF CAREER Award under Grant No. 9985507

We present an example of a distributed transmission line connected to a circuit and show how to model such system in an existing language. We also demonstrate how it can be modeled using a language extension proposed by us.

The remainder of the paper is organized as follows. Section II summarizes the requirements for PDE support. Implementation of PDE's in an existing VHDL-AMS standard for the transmission line example is shown in Section III. Section IV discusses a VHDL-AMS extension needed for PDE support. Conclusions are given in Section V.

II. Requirements for PDE support

To include a block described by PDE's into a VHDL-AMS system simulation, one needs to define:

1) PDE's that describe the physics of a problem
2) Parameters of the PDE's
3) Boundary conditions
4) Contact interface with the rest of the system

For example, a one-dimensional PDE can have a form:

$$\frac{\partial A}{\partial x} + \alpha(x,t)\frac{\partial A}{\partial t} = f(x,t), \quad (1)$$

where $A(x,t)$ is the quantity of interest, $\alpha(x,t)$ is the parameter, $f(x,t)$ is the excitation, x is a spatial variable, and t is time. To solve (1), we need to know $\alpha(x,t)$, which contains the information about material properties and geometry of the system, and the boundary conditions for $A(x,t)$, which also include the initial conditions.

If the system described by (1) is connected to a circuit, we need to define how the quantity $A(x,t)$ interacts with circuit quantities. Exact definition of the contact interface depends on the physics of the problem and may involve a translation, e.g., between electric and magnetic fields and voltages and currents [12], [13]. In VHDL-AMS, such interaction can be implemented using port and terminal definitions.

III. PDE's in existing VHDL-AMS

Since current VHDL-AMS does not support partial derivatives, the only way to implement PDE's in existing language is to discretize the equations with respect to spatial variables and leave the time derivatives to be handled by VHDL-AMS. The idea of a stand-alone spatial discretization has been used by several researchers before for solving PDE problems by creating and then solving equivalent circuits with SPICE and its likes [14], [15]. Using VHDL-AMS approach allows one to bypass the equivalent circuit step. It also makes

0-7803-8182-3/03/$17.00 ©2003 IEEE

possible a concurrent simulation of mixed-technology multi-physics problems, where PDE's and lumped circuits are mixed together. Below, we present an example that demonstrates this concept.

A. Transmission Line Example

Consider a system that consists of a distributed transmission line connected to a circuit as shown in Fig. 1.

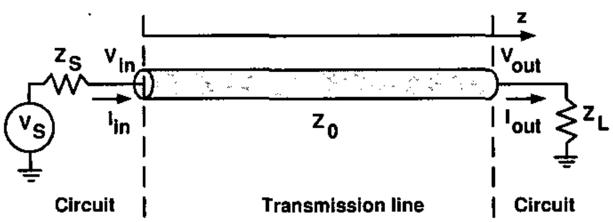

Fig. 1. Transmission line connected to a circuit.

The transmission line can represent an integrated circuit interconnect. The signal propagation on a transmission line can be described with the wave equation, which is a second-order PDE. The circuit is described by Kirchoff's current and voltage law equations.

Coupled problems similar to this one are usually treated by extracting an equivalent port model network for the transmission line and then using SPICE-like circuit simulator to solve for the whole system as a circuit [16] or by interfacing electromagnetic solver and circuit simulator [17].

The interaction between the transmission line and the circuit happens through the terminal voltages and currents: I_{in}, V_{in}, I_{out}, and V_{out}. In this example, the internal quantities of the distributed physics part (voltages and currents) are the same as circuit variables; thus no translation is needed. Boundary conditions that describe an interface to the circuit are:

$$V_{in} = V_S - I_{in} Z_S \tag{2}$$

$$V_{out} = I_{out} Z_L \,. \tag{3}$$

If the line is lossless, the wave equation has the form:

$$-\frac{\partial^2 V}{\partial z^2} - \beta^2 \frac{\partial^2 V}{\partial t^2} = 0\,, \tag{4}$$

where V is the voltage on the transmission line and $\beta = \sqrt{l\,c}$ is the propagation constant (l and c are the inductance and the capacitance per unit length). The same problem can be equivalently formulated in terms of two Telegrapher's equations [18]:

$$\begin{cases} -\dfrac{\partial V}{\partial z} = l \dfrac{\partial I}{\partial t}\,, \\[2ex] -\dfrac{\partial I}{\partial z} = c \dfrac{\partial V}{\partial t}\,. \end{cases} \tag{5}$$

To discretize these equations with respect to z, one can use a classical central difference formula used in many finite difference techniques [19]. If the length of the transmission line is d, a spatial step of Δz results in $N+1$ points where $N = d/\Delta z$ and the voltage and the current need to be determined at each point. A set of two PDE's given by (5) can be converted into the following set of $2N$ ODE's:

$$\begin{cases} -\dfrac{V_n - V_{n-1}}{\Delta z} = l\, I_n', \quad n = 1 ... N \\[2ex] -\dfrac{I_{n+1} - I_n}{\Delta z} = c\, V_n', \quad n = 1 ... N \end{cases} \tag{6}$$

where V_n and I_n are currents and voltages at spatial points as shown in Fig. 2 and prime (') denotes a derivative with respect to time. Two additional equations are given by (3) and (3).

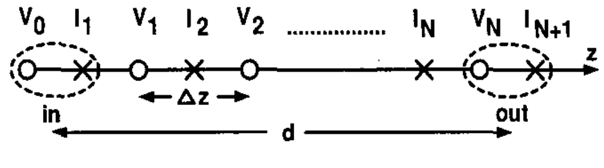

Fig. 2. Spatial finite difference grid.

Note that in this discretization scheme voltages and currents are not defined at the same points in space. This causes matching errors and may give rise to reflections on the transmission line even when all impedances are perfectly matched. The magnitude of the error depends on the discretization step. This effect is known [16] and usually requires an introduction of correction elements to eliminate the errors.

One can see that discretized equations for V and I on the transmission line are equivalent to circuit equations describing the equivalent N-section LC-ladder network shown in Fig. 3, where $L = l\Delta z$ and $C = c\Delta z$ are the inductance and the capacitance of each segment of the transmission line. N-section ladder network is usually valid only for a certain frequency range. The number of sections in the network depends on the transmission line length with respect to the minimum wavelength of interest. As the frequency increases, more stages need to be added. For digital circuits with sharp signal transitions, N needs to be large to accurately reproduce a wide-band response of the transmission line.

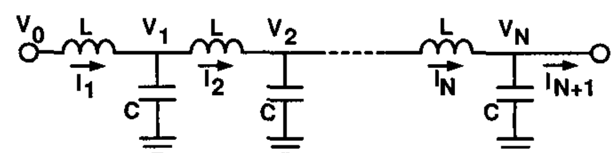

Fig. 3. Equivalent LC-ladder network.

B. VHDL-AMS Implementation

For the transmission line system of equations shown above, we have used two different VHDL-AMS implementations with two different VHDL-AMS simulators: freely available *Hamster*[2] and our own in-house *MCAST* [20].

[2] *Hamster* is now part of *Simplorer*, trademark of *Ansoft Corp.*

Current VHDL-AMS standard includes a language construct called "GENERATE" that in theory allows one to create a large set of simultaneous equations whose terms are array elements and their time derivatives. Boundary conditions and PDE parameter dependence on variables can be defined using array initialization. This is critical for efficient VHDL-AMS implementation of spatial discretization algorithms, which involve a transformation of a small set of PDE's into a large set of ODE's. Unfortunately, to the best of our knowledge the support for the simultaneous statements loop ("GENERATE" construct) is currently missing in many existing VHDL-AMS simulators.

Below we show the VHDL-AMS implementation of the transmission line model, which consists of an entity and an architecture. The entity and the first part of the architecture contain the description of line ports and parameters and are the same for both *Hamster* and *MCAST*. The part of the architecture that describes discretized transmission line equations is different for *Hamster* and *MCAST*.

In our example, PDE parameters l and c are constant, which corresponds to a homogeneous medium. The boundary conditions are given by (2) and (3), and no quantity conversion is needed because quantities of interest (voltages and currents) are the same for both circuit and transmission line.

```
-------- Transmission line begins ---------
LIBRARY DISCIPLINES; LIBRARY IEEE;
 USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;
 USE IEEE.MATH_REAL.ALL;
ENTITY transmission_line IS
 PORT (TERMINAL a, b, g : ELECTRICAL);
END;
ARCHITECTURE behav OF transmission_line IS
 CONSTANT Lz     : REAL := 1.0;
 CONSTANT Cz     : REAL := 1.0;
 CONSTANT Length: REAL := 0.1;
 CONSTANT N      : REAL := 5.0;
 CONSTANT dz     : REAL := Length / N;
 CONSTANT C      : REAL := Lz * dz;
 CONSTANT L      : REAL := Cz *dz;
 QUANTITY Vin  ACROSS Iin  THROUGH a TO g;
 QUANTITY Vout ACROSS Iout THROUGH b TO g;
.........
```

One can see that the transmission line has three terminals: a, b, and g, which correspond to input, output, and ground. The terminal across and through quantities are voltages and currents: $V_{in} = V_0$, $I_{in} = I_1$, $V_{out} = V_N$, and $I_{out} = I_{N+1}$, where $N = 5$. For simplicity, the source and the load resistors were assumed to be $Z_S = 1$ Ohm and $Z_L = 1$ Ohm. The transmission line was taken to be 0.1 m long and have the parameter values $l = 1.0$ H/m and $c = 1$ F/m, which resulted in a characteristic impedance of $Z_0 = 1$ Ohm.

Below we present two implementations of the equations part of the transmission line problem. For *Hamster*, each equation had to be explicitly written out. *MCAST* supports simultaneous loops, which is advantageous for large N. Note that in both implementations I_{out} has a negative sign in front of it because of the VHDL-AMS definition of current flowing into a terminal.

Hamster implementation:

```
.........
 QUANTITY V1, V2, V3, V4: REAL;
 QUANTITY I2, I3, I4, I5: REAL;
BEGIN
 -(V1-Vin) == L * Iin'dot;
 -(I2-Iin) == C * V1'dot;
 -(V2-V1) == L * I2'dot;
 -(I3-I2) == C * V2'dot;
 -(V3-V2) == L * I3'dot;
 -(I4-I3) == C * V3'dot;
 -(V4-V3) == L * I4'dot;
 -(I5-I4) == C * V4'dot;
 -(Vout-V4) == L *I5'dot;
 -(-Iout-I5) == C * Vout'dot;
END;
-------- Transmission line ends ---------
```

MCAST implementation:

```
..........
 QUANTITY V:real_vector(0 to N-1);
 QUANTITY I:real_vector(2 to N);
BEGIN
 -(V(1)-Vin) == L * Iin'dot;
 -(I(2)-Iin) == C * V(1)'dot;
FOR i IN 2 TO N GENERATE
  -(V(i)-V(i-1)) == L * I(i)'dot;
  -(I(i+1)-I(i)) == C * V(i)'dot;
END GENERATE;
 -(Vout-V(N-1)) == L * I(N)'dot;
 -(-Iout-I(N)) == C * Vout'dot;
END;
-------- Transmission line ends ---------
```

Fig. 4 shows the input voltage V_{in} and the output voltage V_{out} for the transmission lines with $N = 5$ and $N = 20$ (other parameters are the same as described before) simulated both in *Hamster* and *MCAST*. The differences are due to truncation errors and the fact that two simulators use different integration methods. One can also see that the average delay of the response is approximately the same for both $N = 5$ and $N = 20$, but more high frequencies are present in the transient for $N = 20$, as one would expect.

IV. Extension for PDE support

Based on the example considered above, an extension for PDE support in VHDL-AMS can be considered. The extension would include a language operator $'dot(x)$, where x is a spatial variable. Such operator is currently non-existent in VHDL-AMS language standard. Choice of spatial discretization technique will be left to an implementor of the VHDL-AMS simulator, as it is now the case with time discretization. Equations for the transmission line example using such language extension would look as follows:

```
-V'dot(z) == Lz * I'dot
-I'dot(z) == Cz * V'dot
```

In perspective, one can also consider including in VHDL - AMS some generic operators, such as nabla operator ($\vec{\nabla}$). Together with vector and scalar multiplication operations and a coordinate system specification, this would enable one to cast

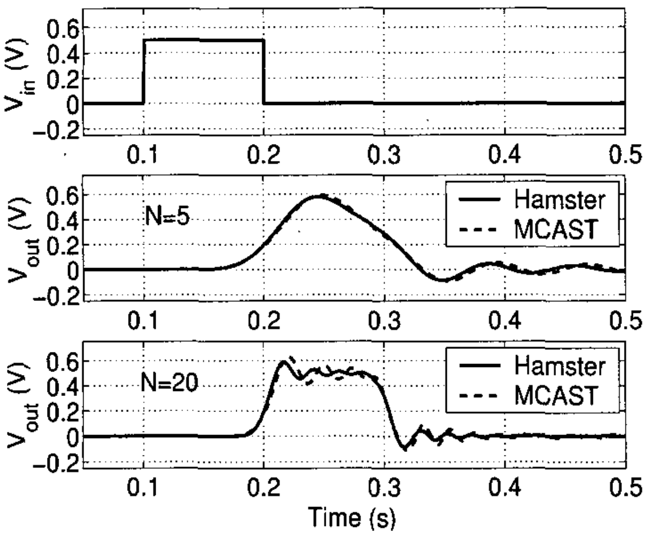

Fig. 4. Input and output voltages for the transmission line shown in Fig. 1 with $N = 5$ and $N = 20$ excited by a rectangular pulse.

many physical equations in VHDL-AMS in a very intuitive form.

Boundary conditions and PDE parameter dependence on variables can be defined using functional description. The contact interface to circuit in VHDL-AMS part can be defined via ports and terminals. Depending on the physics of the problem, the internal quantities of distributed physics blocks may need to be converted into voltages and currents.

The challenge in VHDL-AMS PDE modeling is going to be a realization of a simulator with built-in discretization schemes and solution techniques for different PDE's. Many numerical PDE solvers have already been developed for PDE's of different types and in different physical domains such as *FEMLAB*[3]. The results of PDE solvers research should definitely be used in development of the simulator. In order to be useful to the CAD industry, such simulator must be accurate and fast when applied to large scale multi-physics problems with many unknowns.

V. Conclusion

In this paper, we discussed the problem of modeling and simulation of distributed physics systems described by PDE's in VHDL-AMS. We summarized the requirements for PDE support, demonstrated with the example how to model PDE's in the existing language, and proposed a language extension to support simple PDE's.

This work should be perceived as a first step towards further extension of VHDL-AMS, which would allow one to model and simulate mixed-technology multi-physics systems, consisting of both distributed-physics and lumped-circuit parts. Such capability would facilitate portability, distribution, and exchange of various models between different designers even if a designer is not an expert in, e.g., electromagnetic or thermal modeling. This should greatly speed up an automated synthesis of complex systems-on-chips and can hopefully lead to a new language standard in the CAD industry.

[3]Trademark of *The COMSOL group*

References

[1] E. Christen and K. Bakalar, "VHDL-AMS – a hardware description language for analog and mixed-signal applications," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, pp. 1263–1272, October 1999.

[2] B. F. Romanowicz, "Methodology for the modeling and simulation of microsystems," *Kluwer Academic Publishers*, 1998.

[3] R. Shi, E. Christen, P. Liebmann, S. Krolikoski, and W. Zhou, "VHDL-A: analog extension to VHDL," *Proceedings of Seventh Annual IEEE International ASIC Conference and Exhibit*, pp. 19–23, September 1994.

[4] C.-J. Shi and A. Vachoux, "VHDL-AMS design objectives and rationale," *Current Issues in Electronic Modeling, Kluwer Academic Publishers*, vol. 2, pp. 1–30, 1995.

[5] C. Lallement, F. Pecheux, and Y. Herve, "VHDL-AMS design of a MOST model including deep submicron and thermal-electronic effects," *Proceedings of the Fifth IEEE International Workshop on Behavioral Modeling and Simulation*, pp. 91–96, 2001.

[6] X. Huang and H. A. Mantooth, "Event-driven electrothermal modeling of mixed-signal circuits," *Proceedings of IEEE/ACM International Workshop on Behavioral Modeling and Simulation*, pp. 10–15, 2000.

[7] L. Starzak, M. Zubert, and A. Napieralski, "The new approach to the power semiconductor devices modeling," *Proceedings of Fifth International Conference on Modeling and Simulation of Microsystems*, pp. 640–644, 2002.

[8] J. Willis and J. Johnson, "Language design requirements for VHDL-RF/MW," *IEEE MTT-S International Microwave Symposium Digest*, vol. 3, pp. 2093–2095, 2002.

[9] X. Zhou, B. Liu, and B. Jammes, "Solving steady-state partial differential equation with neural network," *Proceedings of 8th International Conference on Neural Information Processing*, November 2001.

[10] B. Cohen, "Partial differential equations in an MHDL," *MHDL requirements document*, 1991.

[11] D. L. Barton and D. D. Dunlop, "An introduction to MHDL," *IEEE MTT-S International Microwave Symposium Digest*, vol. 3, pp. 1487–1490, 1993.

[12] I. A. Tsukerman, A. Konrad, G. Meunier, and J. C. Sabonnadiere, "Coupled field-circuit problems: trends and accomplishments," *IEEE Transactions on Magnetics*, vol. 29, pp. 1701–1704, August 1992.

[13] N. Orhanovic and N. Matsui, "FDTD-SPICE analysis of high-speed cells in silicon integrated circuits," *Proceedings of Electronic Components and Technology Conference*, pp. 347–352, 2002.

[14] G. Kron, "Numerical solution of ordinary and partial differential equations by means of equivalent circuits," *Journal of Applied Physics*, vol. 16, pp. 172–186, March 1945.

[15] W. R. Zimmerman, "Time domain solutions to partial differential equations using SPICE," *IEEE Transactions on Education*, vol. 39, pp. 563–573, November 1996.

[16] W. Gwarek, "Analysis of arbitrarily shaped two-dimensional microwave circuits by finite-difference time-domain method," *IEEE Transactions on Microwave Theory and Techniques*, vol. 36, pp. 738–744, April 1988.

[17] V. Jandhyala, Y. Wang, D. Gope, and C.-J. Shi, "A surface-based integral-equation formulation for coupled electromagnetic and circuit simulation," *IEEE Microwave and Optical Technology Letters*, vol. 34, pp. 103–106, July 2002.

[18] M. N. Sadiku and L. C. Agba, "A simple introduction to the transmission-line modeling," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 991–999, August 1990.

[19] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations," *IEEE Transactions on Antennas and Propagation*, vol. 14, pp. 302–307, 1966.

[20] B. Wan, B. Hu, L. Zhou, and C.-J. R. Shi, "MCAST: An abstract-syntax-tree based model compiler for circuit simulation," *Proceedings of IEEE Custom Integrated Circuits Conference*, 2003.