Progress in Nuclear Energy 89 (2016) 197–203

Contents lists available at ScienceDirect

Progress in Nuclear Energy

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

Simulation of nuclear reactor dynamics equations using reconfigurable computing

Ahmad Pirouzmand*, Abolhasan Nabavi

Department of Nuclear Engineering, School of Mechanical Engineering, Shiraz University, Shiraz, Iran

ARTICLE INFO

Article history:
Received 8 July 2015
Received in revised form
24 January 2016
Accepted 18 February 2016
Available online 27 February 2016

Keywords:
Discrete-time cellular neural network (D-T CNN)
FPGA
Nuclear reactor dynamics equations
Reconfigurable computing

ABSTRACT

This paper describes the application of a multilayer discrete-time cellular neural network (DT-CNN[1]) and its hardware implementation on a field programmable gate array (FPGA[2]) to model and simulate the nuclear reactor dynamics equations. A new computing architecture model based on FPGA and its detailed hardware implementation are proposed for accelerating the solution of nuclear reactor dynamics equations. The proposed FPGA-based reconfigurable computing platform is implemented on a Xilinx FPGA device and is utilized to simulate step and ramp perturbation transients in typical 2D nuclear reactor cores. Properties of the implemented specialized architecture are examined in terms of speed and accuracy against the numerical solution of the nuclear reactor dynamics equations using MATLAB and C programs. Steady state as well as transient simulations, show a very good comparison between the two methods. Also, the validity of the synthesized architecture as a hardware accelerator is demonstrated.

1. Introduction

Good understanding and prediction of the nuclear reactor dynamics are essential parts of correct system simulation for overall nuclear power plant performance and safety during transients (Duderstadt and Hamilton, 1976). Previous studies have proposed generally simplified models to simulate the nuclear reactor dynamics. These models utilize point kinetics equations for reactor power calculations (Espinosa-Paredes et al., 2011; Nahla, 2008, 2010; Ray and Patra, 2012, 2013). There are also some works that utilize more accurate models such as the multi-point reactor model, the neutron diffusion equation, and the neutron transport equation (Aboanber and Hamada, 2008, 2009; Aboanber and Nahla, 2006). However, all of these models are based on numerical methods that are time-consuming and suffer from such weaknesses as vulnerability to transient phenomena, accumulation of round-off errors and floating-point overflows (Roska et al., 1995; Chedjou et al., 2001, 2008).

Conventionally, to speed up scientific or engineering computation programs on general-purpose computers, one may elect to use faster CPUs, more memory, systems with more efficient (though complicated) architecture, better software compilers, or even coding with assembly languages. A novel approach to high performance computing that is growing in credibility is the use of reconfigurable hardware to form custom hardware accelerators for numerical computations. This can be achieved through the use of field programmable gate arrays (FPGAs) which are large arrays of programmable logic gates and memory. FPGAs can be reconfigured in real time to provide large parallel arrays of simple processors that can solve a problem much faster and more efficiently. FPGAs have evolved rapidly in the last few years and have now reached sufficient speed and logic density to implement highly complex systems. FPGAs are being applied in many areas to accelerate algorithms that can make use of massive parallelism (He, 2007; Hu, 2011). The rapid improvement in hardware capabilities in the last few years has steadily widened the range of prospective application areas. One promising application area is the use of FPGA-based reconfigurable hardware to form custom hardware accelerators for numerical computations.

This study is a continuation of previous works in which we effectively simulated the dynamics of a typical nuclear reactor with a semi-analog medium using the neutron diffusion and transport equations to compute the reactor power distribution in steady state and transient conditions (Hadad and Piroozmand, 2007; Hadad et al., 2008; Pirouzmand and Hadad, 2012, 2011; Pirouzmand et al., 2011). Herein, the nuclear reactor dynamics equations are first modeled using a five-layer discrete-time cellular neural

* Corresponding author.
E-mail address: pirouzm@shirazu.ac.ir (A. Pirouzmand).
[1] Discrete-time cellular neural network.
[2] Field programmable gate array.

http://dx.doi.org/10.1016/j.pnucene.2016.02.018
0149-1970/

network (DT-CNN). Then, the DT-CNN model is implemented on a Xilinx FPGA device using VHSIC hardware description language (VHDL[3]) and is utilized to simulate step and ramp perturbation transients in typical 2D nuclear reactor cores.

2. Theory

2.1. Nuclear reactor dynamic model

Nuclear reactor core dynamics are stated by linear and nonlinear differential equations with varying coefficients, which are functions of the core operating conditions (power level, coolant and fuel temperatures, coolant density, poison buildup, burn-up rate, etc.) (Duderstadt and Hamilton, 1976). In this study a simplified dynamic model based on two-group diffusion equations, an effective one group delayed neutron precursor, and fuel temperature feedback is applied.

The time-dependent two-group diffusion equations and a precursor concentrations equation are given by Eqs. (1) and (2), respectively (Duderstadt and Hamilton, 1976):

$$\frac{1}{\upsilon^{(1)}}\frac{\delta\phi^{(1)}}{\delta(t)} = D^{(1)}\nabla^2\phi^{(1)} - \Sigma_r^{(1)}\phi^{(1)} + (1-\beta)\sum_{g=1}^{2}\vartheta\Sigma_f^{(g)}\phi^{(g)} + \lambda C \tag{1-a}$$

$$\frac{1}{\upsilon^{(2)}}\frac{\delta\phi^{(2)}}{\delta(t)} = D^{(2)}\nabla^2\phi^{(2)} - \Sigma_r^{(2)}\phi^{(2)} + \Sigma_s^{(12)}\phi^{(1)} \tag{1-b}$$

$$\frac{\partial C(t)}{\partial t} = -\lambda C(t) + \beta\sum_{g=1}^{2}\vartheta\Sigma_g^{(f)}\phi^{(g)} \tag{2}$$

where g is the energy group, v(g) neutron velocity of group g in (cm/s), $\phi^{(g)}$ neutron flux of group g in (neutron/cm^2 s), $\Sigma_r^{(g)}$ removal and $\Sigma_f^{(g)}$ fission macroscopic cross sections of group g in (cm^{-1}), respectively, $\Sigma_s^{(12)}$ macroscopic scattering cross section from energy group 1 to 2 in (cm^{-1}), $D^{(g)}$ neutron diffusion coefficient of group g in (cm), β the fraction of delayed neutrons, υ average neutrons produced per fission, λ decay constant of precursors in (s^{-1}), and C is the one group delayed neutron precursors concentration in (nuclei/cm^3). Also, the feedback model (an adiabatic fuel heat up thermal model and feedback Doppler effects) is given by Eq. (3) (Aboanber and Hamada, 2009):

$$\frac{\partial}{\partial t}T(r,t) = \alpha\varepsilon\sum_{g=1}^{G}\Sigma_{fg}\phi_g(r,t) \tag{3-a}$$

$$\Sigma_a^1(r,t) = \Sigma_a^1(r,0)\left[1+\gamma\left(\sqrt{T(r,t)}-\sqrt{T(r,0)}\right)\right] \tag{3-b}$$

$\alpha = 1.1954$ K cm^{3}j^{-1} Conversion factor from power to fuel temperature. $\gamma = 0.003034$ K$^{-1/2}$ Feedback constant. $\varepsilon = 0.3204 \times 10^{-10}$j Energy release per fission.where T is the fuel temperature in [K].

2.2. Architecture of DT- CNN

Modeling of physical phenomena often involves the use of complex system of equations whose computational solution has demanding requirements in terms of memory and computing power. Among the different techniques proposed to alleviate this problem, the DT-CNN has been proved to be a powerful tool as it has the advantage of a feasible hardware implementation that can significantly speed up the computations.

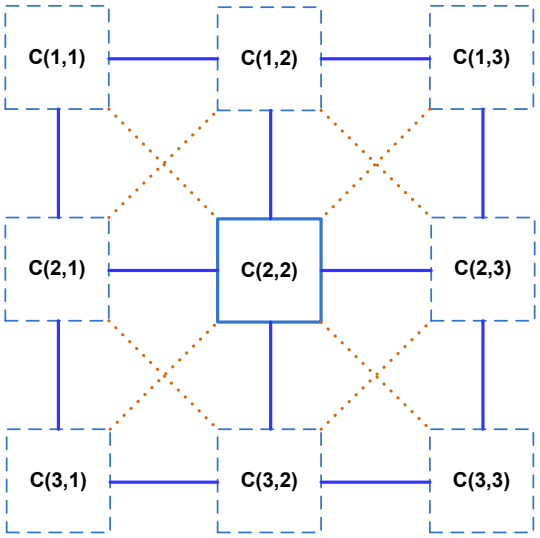

Fig. 1. A 2-D cellular neural network of size 3 × 3. Links between the cells indicate interactions between them.

Fig. 1 shows an $M \times N$ cellular neural network (CNN[4]), having M rows and N columns. We call the cell on the ith row and jth column, cell (i, j) and denote it by C(i, j). Each cell has input, output, and the equation of state of its own. Governing state equation of a cell at location (i, j) in a discrete time network is as follows (Harrer and Nossek, 1992):

$$x_{i,j}[n+1] = \sum_{k,l\in\mathcal{N}(i,j)} A\bullet y_{k,l}[n] + \sum_{k,l\in\mathcal{N}(i,j)} B\bullet u_{kl} + z \tag{4}$$

In Eq. (4), u_{ij}, x_{ij} and y_{ij} are input, state and output variables of a cell at location (i,j), respectively. After discretizing the reactor dynamics equations (i.e. Eqs. (1)–(3)) in space and time and a comparison with Eq. (4), the DT-CNN model of reactor dynamics is achieved. The dynamic behavior of the DT-CNN model is determined by the pattern of two matrices A and B and a scalar quantity z (bias term). Since the state of each node at each time step is only dependent to the state of that node and its adjacent nodes in the previous time step, the matrix B and the bias term z are equal to zero. In this study, the network model can be written as follows:

$$A = \begin{pmatrix} a_{0,0} & a_{0,1} & a_{0,2} \\ a_{1,0} & a_{1,1} & a_{1,2} \\ a_{2,0} & a_{2,1} & a_{2,2} \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad z = 0$$

where the coefficients $a_{i,j}$ are determined from the CNN[5] model of nuclear reactor dynamic equations. It is worth mentioning here that cellular neural networks, which were introduced above can be generalized and expanded to a multi-layer cellular neural network by considering multi-state variable for each network cell (Hadad and Piroozmand, 2007).

3. DT-CNN model

To solve the above reactor dynamics equations and simulate the steady state and transient behaviors of neutron flux distribution in

[3] VHSIC hardware description language.

[4] Cellular neural network.

[5] Application-specific integrated circuits.