Proceedings World Geothermal Congress 2010
Bali, Indonesia, 25-29 April 2010

Array Computing Based Implementation of Water Reinjection in Geothermal Structure

Simeon G. Kostianev, S. Kocsárdi, Z. Nagy, P. Szolgay, S. Akin

University of Mining and Geology – Dept. of Mathematics, Sofia – 1700, Bulgaria

simeon44@yahoo.co.uk

Keywords: Partial Differential Equations, Water Heating, Cellular Neural Networks, Array computing.

ABSTRACT

A method is given here to model the processes of filtration and heat transfer in a three-dimensional geothermal medium. The main goal of the analysis was to find and prove the correlation between pressure and temperature in a geothermal structure. The Cellular Neural/Nonlinear Networks paradigm is a natural framework that describes the behavior of locally interconnected dynamical systems that have an array structure. Emulated digital implementation of the CNN-Universal Machine can use space variant templates (weighted interconnections) and multi-layered structures on different array processing architectures. It is desired to find the optimal computational architecture which satisfies the functional requirements, using the minimal precision and nevertheless achieving maximum computing power. To meet these requirements, it is desired to process the spatial-regions with the highest possible parallelism.

INTRODUCTION

The Cellular Neural Network is a non-linear dynamic processor array. Its extended version, the CNN Universal Machine (CNN-UM), was invented in 1993 [1]. The CNN paradigm is a natural framework to describe the behavior of locally interconnected dynamical systems that have an array structure. Thus, it is quite straightforward to use CNN to compute the solution of different PDEs and to use the highest order of time derivatives [2], [3], [5]. However, the results cannot be used in practical applications because of the limitations of the analog CNN-UM chips, such as low precision and the restricted usability by applications operating with space-variant templates in a multi-layered structure. By implementing the CNN-UM architecture on an array processor, it is possible to modify the cell model and simulate its behavior in a very short time.

A method is given here to model the processes of filtration and heat transfer in a three-dimensional geothermal medium. The main goal of the analysis was to find and prove the correlation between pressure and temperature in a geothermal structure.

Knowing the partial differential equations of a valid geothermal problem, it is necessary to compute complex spatio-temporal dynamics.

It is desired to find the optimal computational architecture which satisfies the functional requirements, while using the minimal required precision and nevertheless achieving maximum computing power. To meet these requirements, it is desired to process the spatio-regions with the highest possible parallelism.

1. THE GEOTHERMAL MODEL

A mathematical model of filtration and thermal processes of the surveyed region (the "Kazichene-Ravno pole") has been developed with a focus on geothermal energy production. The mechanism of thermodynamic processes is strictly defined by Darcy's law of filtration and Fourier's law of heat transfer [4], so it is expressed using differential equations and supplemented with boundary conditions corresponding to the specific problem.

1.1. Boundary Value Problem of Filtration Process

For underground water movement, the main differential equation referring to the filtration in the surveyed stratum, is given in Equation 1:

$$\frac{\partial}{\partial x}\left(T\frac{\partial H}{\partial x}\right)+\frac{\partial}{\partial y}\left(T\frac{\partial H}{\partial y}\right)=0 \qquad (1)$$

$$T = k_f m$$

where H is water pressure measured from an unspecified reference plane, T is stratum conductivity (which is function of co-ordinates x and y), k_f is the filtration coefficient, and m is the stratum thickness (which is a function of the co-ordinates x and y). The filtration rate at each point in the filtration field is given in Equation (2):

$$V_x = -k_f\frac{\partial H}{\partial x},\quad V_y = -k_f\frac{\partial H}{\partial y} \qquad (2)$$

1.2. Boundary Value Problem of Heat Transfer

As the heat transfer takes place in three layers with different hydro-geological and thermo-physical characteristics, the model is described by three differential equations. The structure of the layers is shown in Figure 1.

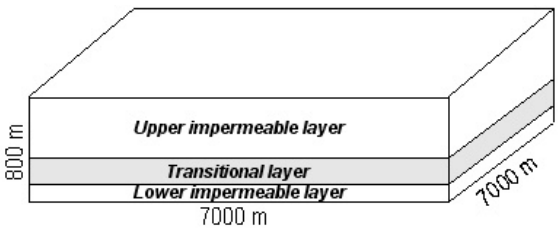

Figure 1. Model of the examined region

In conformity with the key prerequisites of the heat transfer model, the main differential equations can be expressed in the following way:

$$\frac{\partial}{\partial x}\left(\lambda_1\frac{\partial t_1}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda_1\frac{\partial t_1}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda_1\frac{\partial t_1}{\partial z}\right)=\rho_1 c_1\frac{\partial t_1}{\partial \tau} \qquad (3)$$

$$\frac{\partial}{\partial x}\left(\lambda_2\frac{\partial t_2}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda_2\frac{\partial t_2}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda_2\frac{\partial t_2}{\partial z}\right)-m\rho cV_x\frac{\partial t_2}{\partial x}$$
$$-m\rho cV_y\frac{\partial t_2}{\partial y}=m\rho_2 c_2\frac{\partial t_2}{\partial \tau} \quad (4)$$

$$\frac{\partial}{\partial x}\left(\lambda_3\frac{\partial t_3}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda_3\frac{\partial t_3}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda_3\frac{\partial t_3}{\partial z}\right)=\rho_3 c_3\frac{\partial t_3}{\partial \tau} \quad (5)$$

where the symbols are as follows:

t_1, t_2 and t_3 denote temperatures in the 1st, 2nd and 3rd areas, respectively, and are functions of x, y and z.

λ_1, λ_2 *and* λ_3 are the coefficients of conductivity in the same areas and are functions of x, y and z.

c_1, c_2 and c_3 are the heat capacities of the rocks in the respective areas.

ρ_1, ρ_2 and ρ_3 are the rock thicknesses in the respective areas.

ρ and c are the density and heat capacity of water.

The Equations (3) and (5) describe the heat transfer in the upper and lower argillaceous, impermeable layer, while Equation (4) denotes the process in the transitional, water saturated calcareous layer.

1.3. Discretization of PDEs in Time and Space

The process described by the governing equation of filtration is truly a boundary value problem, which does not depend on time. However, computations suggest that the filtration terms are space dependent.

In the implementation of the governing equations of heat transfer on different hardware units, it is necessary to discretize the system of equations in accordance with both space and time. The first order forward Euler formulation has been used to perform this operation, and a set of explicit, coupled finite-difference equations have been derived corresponding to Equations (3)-(5) and can be described by the following formulas, dubbed Equations (7) and (8):

$$t_{1_{x,y,z}}^{k+1}=t_{1_{x,y,z}}^{k}+\frac{\Delta\tau}{\rho_1 c_1}\frac{1}{\Delta x^2}\left(\lambda_{1_{x-1,y,z}}t_{1_{x-1,y,z}}^{k}-\left(\lambda_{1_{x-1,y,z}}+\lambda_{1_{x,y,z}}\right)t_{1_{x,y,z}}^{k}+\lambda_{1_{x,y,z}}t_{1_{x+1,y,z}}^{k}\right)$$
$$+\frac{\Delta\tau}{\rho_1 c_1}\frac{1}{\Delta y^2}\left(\lambda_{1_{x,y-1,z}}t_{1_{x,y-1,z}}^{k}-\left(\lambda_{1_{x,y-1,z}}+\lambda_{1_{x,y,z}}\right)t_{1_{x,y,z}}^{k}+\lambda_{1_{x,y,z}}t_{1_{x,y+1,z}}^{k}\right) \quad (6)$$
$$+\frac{\Delta\tau}{\rho_1 c_1}\frac{1}{\Delta z^2}\left(\lambda_{1_{x,y,z-1}}t_{1_{x,y,z-1}}^{k}-\left(\lambda_{1_{x,y,z-1}}+\lambda_{1_{x,y,z}}\right)t_{1_{x,y,z}}^{k}+\lambda_{1_{x,y,z}}t_{1_{x,y,z+1}}^{k}\right),$$

$$t_{2_{x,y,z}}^{k+1}=t_{2_{x,y,z}}^{k}+\frac{\Delta\tau}{m_{x,y}\rho_2 c_2}\frac{1}{\Delta x^2}\left(\lambda_{2_{x-1,y,z}}t_{2_{x-1,y,z}}^{k}-\left(\lambda_{2_{x-1,y,z}}+\lambda_{2_{x,y,z}}\right)t_{2_{x,y,z}}^{k}+\lambda_{2_{x,y,z}}t_{2_{x+1,y,z}}^{k}\right)$$
$$+\frac{\Delta\tau}{m_{x,y}\rho_2 c_2}\frac{1}{\Delta y^2}\left(\lambda_{2_{x,y-1,z}}t_{2_{x,y-1,z}}^{k}-\left(\lambda_{2_{x,y-1,z}}+\lambda_{2_{x,y,z}}\right)t_{2_{x,y,z}}^{k}+\lambda_{2_{x,y,z}}t_{2_{x,y+1,z}}^{k}\right)$$
$$+\frac{\Delta\tau}{m_{x,y}\rho_2 c_2}\frac{1}{\Delta z^2}\left(\lambda_{2_{x,y,z-1}}t_{2_{x,y,z-1}}^{k}-\left(\lambda_{2_{x,y,z-1}}+\lambda_{2_{x,y,z}}\right)t_{2_{x,y,z}}^{k}+\lambda_{2_{x,y,z}}t_{2_{x,y,z+1}}^{k}\right) \quad (7)$$
$$-\frac{\Delta\tau}{\rho_2 c_2}\frac{\rho c}{\Delta x}V_{x_{x,y}}\left(t_{2_{x,y,z}}^{k}-t_{2_{x+1,y,z}}^{k}\right)$$
$$-\frac{\Delta\tau}{\rho_2 c_2}\frac{\rho c}{\Delta y}V_{y_{x,y}}\left(t_{2_{x,y,z}}^{k}-t_{2_{x,y+1,z}}^{k}\right)$$

and

$$t_{3_{x,y,z}}^{k+1}=t_{3_{x,y,z}}^{k}+\frac{\Delta\tau}{\rho_3 c_3}\frac{1}{\Delta x^2}\left(\lambda_{3_{x-1,y,z}}t_{3_{x-1,y,z}}^{k}-\left(\lambda_{3_{x-1,y,z}}+\lambda_{3_{x,y,z}}\right)t_{3_{x,y,z}}^{k}+\lambda_{3_{x,y,z}}t_{3_{x+1,y,z}}^{k}\right)$$
$$+\frac{\Delta\tau}{\rho_3 c_3}\frac{1}{\Delta y^2}\left(\lambda_{3_{x,y-1,z}}t_{3_{x,y-1,z}}^{k}-\left(\lambda_{3_{x,y-1,z}}+\lambda_{3_{x,y,z}}\right)t_{3_{x,y,z}}^{k}+\lambda_{3_{x,y,z}}t_{3_{x,y+1,z}}^{k}\right) \quad (8)$$
$$+\frac{\Delta\tau}{\rho_3 c_3}\frac{1}{\Delta z^2}\left(\lambda_{3_{x,y,z-1}}t_{3_{x,y,z-1}}^{k}-\left(\lambda_{3_{x,y,z-1}}+\lambda_{3_{x,y,z}}\right)t_{3_{x,y,z}}^{k}+\lambda_{3_{x,y,z}}t_{3_{x,y,z+1}}^{k}\right)$$

where $\Delta\tau$ is the time step, and Δx, Δy and Δz are the distance of grid points in the x,y and z directions, respectively.

2. CELL PROCESSOR ARCHITECTURE

The Cell Broadband Engine Architecture (CBEA) is designed to achieve high computing performance with better area/performance and power/performance ratios than the conventional multi-core architectures. The CBEA defines a heterogeneous multi-processor architecture where general purpose processors called Power Processor Elements (PPE) and SIMD processors called Synergistic Processor Elements (SPE) are connected via a high speed on-chip coherent bus called an Element Interconnect Bus (EIB). The CBEA architecture is flexible, and the ratios of the different elements can be defined according to the requirements of different applications. The first implementation of the CBEA is the Cell Broadband Engine (Cell BE or Cell), which was designed for the Sony Playstation 3 game console and contains 1 PPE and 8 SPEs. A block diagram of the Cell is shown in Figure 2.

The PPE is a conventional dual-threaded 64 bit PowerPC processor that can run existing operating systems without modification and can control the operation of the SPEs. To simplify processor design and achieve higher clock speed, instruction reordering is not supported by the PPE. The EIB is not a bus as suggested by its name but a ring network which contains 4 unidirectional rings, where two rings run counter to the direction of the other two. The dual-channel Rambus XDR memory interface provides very high memory bandwidth (25.6 GB/s). I/O devices can be accessed via two Rambus FlexIO interfaces where one of them (the Broadband Interface (BIF)) is coherent and makes it possible to directly connect two Cell processors.

The SPEs are SIMD only processors that are designed to handle streaming data. Therefore, they do not perform well in general purpose applications and cannot run operating systems.

2.1. Cell Blade Systems

The third generation blade system is the IBM Blade Center QS22 equipped with new generation PowerXCell 8i processors manufactured using 65nm technology. The double precision performance of the SPEs are significantly improved providing extraordinary computing density – up to 6.4 TFLOPS single precision and up to 3.0 TFLOPS double precision in a single Blade Center house. These blades are the main building blocks of the world's fastest supercomputer at Los Alamos National Laboratory which first broke the "petaflop barrier" of 1,000 trillion operations per second.

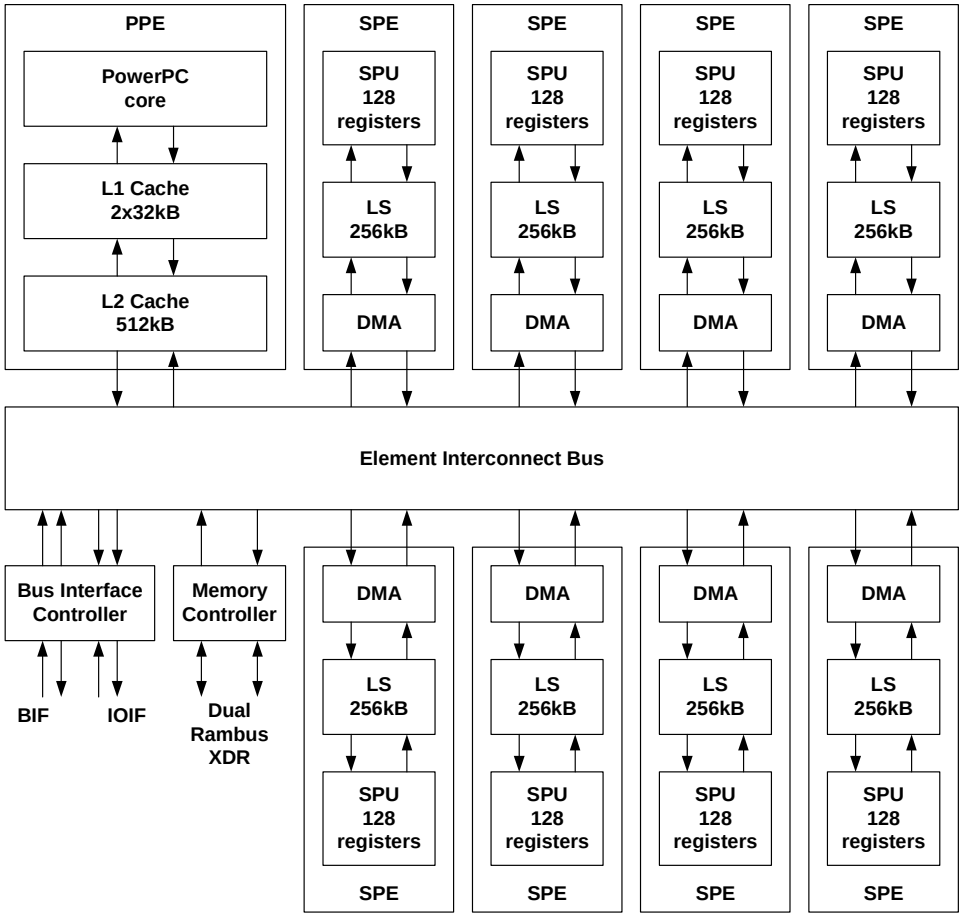

Figure 2. Block diagram of the Cell Broadband Engine

3. SOLUTION ON A CNN ARCHITECTURE

To model the process of reinjection on emulated digital CNN architecture [8], a space-variant CNN model was developed based on Equations (6)-(8), which is operating with 3 5-dimensional templates. The second equation, which describes the behavior of the water saturated transitional layer, contains two additional parts that were derived from the time-independent filtration equation and make the connection between the processes of filtration and heat transfer.

For the process of filtration, only the temperature values have to be calculated and updated during the iterations, so zero-input CNN templates can be used, where the given initial values dictate the initial state of the template that is running. To design a space-variant, non-linear template for the three-dimensional medium, 3 coupled 2D templates were designed using an r=1 neighborhood for every three physical layers so that every feedback template-threefold contains 27 elements.

The structure of the coupled templates for one physical layer can be seen in Figure 3, where *n* denotes the described physical layer.

The coupled templates of the second layer that was determined from Equation (7) are as follows:

$$A_{21} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta z^2}\lambda_{x,y,z-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A_{23} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta z^2}\lambda_{x,y,z} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A_{22} = \begin{bmatrix} 0 & \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta x^2}\lambda_{x-1,y,z} & 0 \\ \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta y^2}\lambda_{x,y-1,z} & 1-\dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta x^2}\left(\lambda_{x-1,y,z}+\lambda_{x,y,z}\right) & \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta y^2}\lambda_{x,y,z} \\ & -\dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta y^2}\left(\lambda_{x,y-1,z}+\lambda_{x,y,z}\right) & \\ & -\dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta z^2}\left(\lambda_{x,y,z-1}+\lambda_{x,y,z}\right) & +\dfrac{\Delta\tau}{\rho_2 c_2}\dfrac{\rho c}{\Delta y}V_{y_{x,y}} \\ & -\dfrac{\Delta\tau}{\rho_2 c_2}\dfrac{\rho c}{\Delta x}V_{x_{x,y}}-\dfrac{\Delta\tau}{\rho_2 c_2}\dfrac{\rho c}{\Delta y}V_{y_{x,y}} & \\ 0 & \dfrac{\Delta\tau}{m_{x,y}\rho_2 c_2}\dfrac{1}{\Delta x^2}\lambda_{x,y,z}+\dfrac{\Delta\tau}{\rho_2 c_2}\dfrac{\rho c}{\Delta x}V_{x_{x,y}} & 0 \end{bmatrix}$$

The space-variant templates for the first and third physical layers can be determined similarly, but the appropriate ρ and c multiplier coefficients must be used.

By using the previously described discretization method, a C based solver was developed and optimized for the SPEs of the Cell architecture.

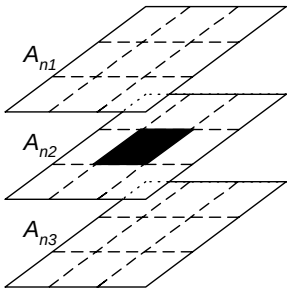

Figure 3. Structure of coupled template for the 3D heat transfer (r=1)

The large (128-entry) register file of the SPE makes it possible to store the neighborhood of the currently processed cell during the solution of the governing equations. The number of load instructions can be decreased significantly. Since the SPEs cannot address the global memory directly, the user's application running on the SPE is responsible for the data transfer between the local memory of the SPE and the global memory via DMA transactions.

The relatively small local memory of the SPEs does not allow storage of all the data required for the computation. Therefore, an efficient buffering method is required to save memory bandwidth. In this solution, a belt of 6 slices is stored in the local memory from the array: 3 slices are required to form the local neighborhood of the currently processed row, one slice is required for data synchronization, and two slices are required to allow overlap of the computation and communication, as shown in Figure 4. During implementation, the environment of the CNN simulation kernel was used [8]. Template operations were optimized according to discretized Equations (6)-(8) to improve performance. The optimized kernel requires about 32 KB of memory from the local store of the SPE, leaving approximately 224 KB for the slice buffers. Therefore, the size of the buffer is at maximum 3584 grid points (59x59 array), while the number of slices is only limited by the size of the main memory.

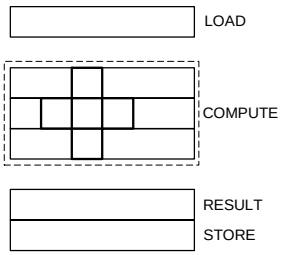

Figure 4. Local store buffers

The SPEs in the Cell architecture are SIMD-only units; hence, the state values of the cells should be grouped into vectors. The size of the registers is 128 bit, and 32 bit floating point numbers are used during the computation. Accordingly, the vectors used here contain 4 elements. The state value of the i^{th} cell is denoted by *si*.

It seems obvious to pack 4 neighboring cells into one vector {*s5*,*s6*,*s7*,*s8*}. However, constructing the vector that contains the left {*s4*,*s5*,*s6*,*s7*} and right {*s6*,*s7*,*s8*,*s9*} neighbors of the cells is somewhat complicated because 2 "rotate" instructions and 1 "select" instruction are needed to generate the required vector, as illustrated in Figure 5. This limits the utilization of the floating-point pipeline, because 3 integer instructions (rotate and select) must be carried out to generate the left and right neighborhoods of the cell before a floating point instruction can be issued.

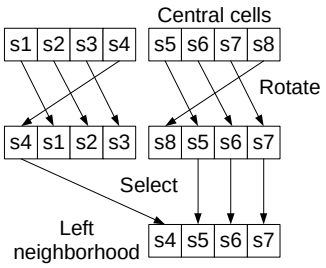

Figure 5. Generation of the left neighborhood

This limitation can be removed by slicing the cell array into 4 vertical stripes and rearranging the cell values. In the above case, the 4-element vector contains data from the 4 different slices, as shown in Figure 6. This makes it possible to eliminate the shift and shuffle operations to create the neighborhood of the cells in the vector. The rearrangement should be carried out only once (at the beginning of the computation) and can be carried out by the PPE. Although this solution improves the performance of the simulation data, data dependency between the floating-point instructions may still cause pipeline stalls. In order to eliminate this dependency, the inner loop of the computation must be rolled out. Instead of waiting for the result of the first floating-point instruction, the computation of the next group of cells is started. The level of unrolling is limited by the size of the register file.

To utilize the power of the Cell architecture, computation work should be distributed between the SPEs. In spite of the large memory bandwidth of the architecture, the memory bus can be easily saturated. Therefore, an appropriate arrangement of data between SPEs can greatly improve computing performance. One possible solution is to distribute grid data between the SPEs. In this case, each SPE is working on a narrow horizontal slice of the grid, as shown in Figure 7. Communication between the SPEs is required only during the computation of the first and last row of the slice (gray areas) and can be efficiently carried out by a single DMA transaction.

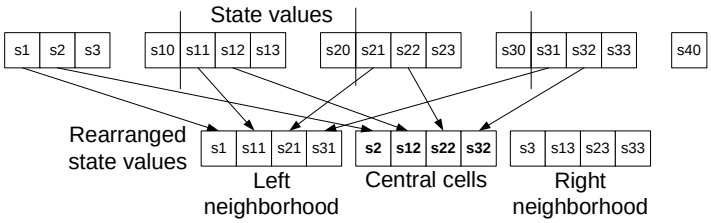

Figure 6. Rearrangement of the state values

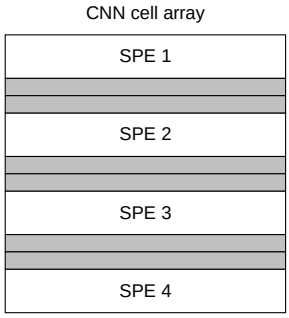

Figure 7. Data distribution between SPEs: slicing

However, the above data arrangement is well suited for the architecture of the array processors and simplifies the inter-processor communication. The SPEs access main memory in parallel, which might require very high memory bandwidth. If few instructions are executed on large data sets, then the memory system becomes saturated, resulting in low performance.

Static timing analysis of the optimized CFD solver kernel showed that a grid point can be updated in approximately 63 clock cycles. Each update requires movement of 24 byte data (1x4byte state value, 1x4byte updated state value, 4x4byte mask value) between the main memory and the local store of the SPE. The Cell processor is running at a 3.2 GHz clock frequency; therefore, in an ideal case the expected performance of the computation kernel using one SPE is 203 million update/s. The estimated memory bandwidth requirement is 4.8 GB/s, which is less than 20% of the available memory bandwidth. Therefore, grid data can be distributed between 5 SPEs, and each of them can work on its own slice in parallel without running into a memory bottleneck.

RESULTS, PERFORMANCE

To show the efficiency of our solution, a complex test case was accomplished in a model having 77x77x17 grid points. The input and the result of the 32 bit floating-point computation of water saturated layer after 1 year of simulation time with a 2048 s timestep are shown in Figures 8 and 9, respectively. The computation time was 221 seconds on an Intel Core 2 Duo 2 GHz microprocessor. This is equivalent to approximately 6.9 million cell update/s.

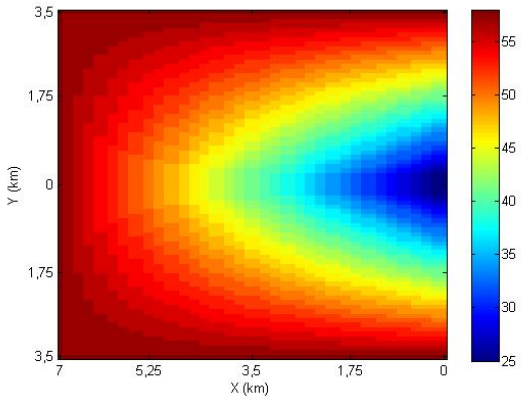

Figure 8. Input map for transitional layer

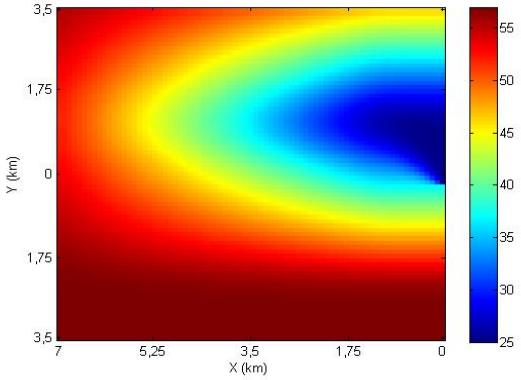

Figure 9. Temperature map after simulation

The performance of the IBM Cell based emulated digital solution of the model is very encouraging. The previous computation takes just 7.5 s using 1 SPE and 4.14 s using all 8 SPEs of the Cell processor, so the computation has been accelerated by 29.4 times using 1 SPE and 194.2 times using all 8 SPEs compared to the performance of Core 2 Duo.

CONCLUSION

The state equation of water reinjection in geothermal reservoirs with realistic input parameters was solved using a CNN architecture. The cell model of the emulated digital CNN-UM processor was modified according to the governing equations of the model of heat transfer and filtration processes. Further, the CNN-UM solution can be efficiently accelerated using re-configurable devices [6],[7],[10]. The proposed architecture was implemented on IBM Cell processor based QS22 architecture.

The solution was optimized according to the special requirements of the Cell architecture. Performance comparison showed that the calculations could be performed 29 times faster with respect to a high performance microprocessor in the single SPE solution and 194 times faster when all the 8 SPEs are utilized. It is planned to extend our solution to use multiple Cell processors in the future.

REFERENCES

T. Roska and L. O. Chua, “The CNN Universal Machine: an Analogic Array Computer”, *IEEE Transaction on Circuits and Systems-II*, vol. 40, pp. 163-173, 1993.

R. Carmona, F. Jiménez-Garrido, R. Domínguez-Castro, S. Espejo, A. Rodríguez-Vázquez, “Programmable retinal dynamics in a CMOS mixed-signal array processor chip”, Proc. Of SPIE Vol. 5119, pp. 13-23.

Z. Nagy and P. Szolgay, “Configurable Multi-layer CNN-UM Emulator on FPGA”, *IEEE Transaction on Circuit and Systems I: Fundamental Theory and Applications*, vol. 50, pp. 774-778, 2003.

Pentland, Gitirana and Fredlund, “Use of a General Partial Differential Equation Solver for Solution of Mass and Heat Transfer Problems in Geotechnical Engineering”, 4th Brazilian Symposium on Unsatured Soils, March 2001, pp. 29-45.

P. Szolgay, G. Vörös and Gy. Erőss, “On the Application of an Analog Input Dual Output CNNUM Chip in Transient Analysis of Mechanical Vibrating Systems”, Proc. Of the IEEE INES97, pp. 257-260, 1997.

Celoxica Ltd. Homepage [Online]. Available: http://www.celoxica.com

Xilinx Products Homepage [Online]. Available: http://www.xilinx.com

Z. Nagy, Zs. Vörösházi,P. Szolgay, "Emulated Digital CNN-UM Solution of Partial Differential Equations", *Int. J. CTA, Vol. 34, No. 4, pp. 445-470 (2006)*

Z. Nagy, L. Kék, Z.Kincses, A Kiss, P. Szolgay, "Toward exploitation of cell multiüprocessor array in time-consuming applications by using CNN model", *Int. J. CTA, Vol. 36, No. 5-6, pp. 605-622 (2008)*

S Kocsárdi, Z. Nagy, S Kostianev, P. Szolgay, "FPGA based implementation of water reijection in geothermal structure", *Proc. of CNNA2006, pp. 323-327, 2006,Istanbul*