

AN APPLICATION OF PDE TO PREDICT BRAIN TUMOR GROWTH USING HIGH PERFORMANCE COMPUTING SYSTEM

Md. Rajibul Islam¹, Norma Alias² and Siew Young Ping³

¹Faculty of Information Science and Technology, Multimedia University, Malaysia

²Ibnu Sina Institute, Faculty of Science, University Technology Malaysia, Johor

³Mathematics Department, Faculty of Science, University Technology Malaysia, Johor

E-mail: md.rajibul.islam05@mmu.edu.my, norma@ibnusina.utm.my, siew.y.p@gmail.com

Abstract: This study is to predict two-dimensional brain tumors growth through parallel algorithm using the High Performance Computing System. The numerical finite-difference method is highlighted as a platform for discretization of two-dimensional parabolic equations. The consequence of a type of finite difference approximation namely explicit method will be presented in this paper. The numerical solution is applied in the medical field by solving a mathematical model for the diffusion of brain tumors which is a new technique to predict brain tumor growth. A parabolic mathematical model used to describe and predict the evolution of tumor from the avascular stage to the vascular, through the angiogenic process. The parallel algorithm based on High Performance Computing (HPC) System is used to capture the growth of brain tumors cells in two-dimensional visualization. PVM (Parallel Virtual Machine) software is used as communication platform in the HPC System. The performance of the algorithm evaluated in terms of speedup, efficiency, effectiveness and temporal performance.

Keywords: Partial Differential Equation (PDE), parabolic equation, explicit method, Red Black Gauss-Seidel, Parallel Virtual Machine (PVM), High Performance Computing (HPC), Brain Tumor.

1. Introduction

Brain tumours can either develop within the brain (primary) or develop from cancer cells that spread to the brain (metastatic or secondary). Primary tumours can be grouped into non-cancerous (benign) and cancerous (malignant). Benign brain tumours usually grow slowly and can often be removed by surgery depending upon their specific location in the brain. Almost half of all brain tumours are non-cancerous. Malignant brain tumours are commonly called brain cancer tend to grow rapidly spreading into the surrounding brain

tissue and often cannot be entirely removed surgically. Because there is no known cause of brain tumours, there is no way to clearly prevent them. For the simulation of brain tumours growth using parabolic equations, it needs us to see the definition of partial differential equations [1]. Mathematically, parabolic PDEs serve as a transition from the hyperbolic PDEs to the elliptic PDEs. Physically, parabolic PDEs tend to arise in time dependent diffusion problems, such as the transient flow of heat in accordance with Fourier's law of heat conduction.

Mathematical modeling will play an increasingly important role in helping biomedical researchers to understand and gain useful insight into different aspects of solid tumor growth. In this study, the two-dimensional model is deduced by a plane square lattice scheme [2] for the advection-diffusion equation to describe the evolution of brain tumor growth in term of partial differential equation systems. This evolution equation can be extended to become a system on a cubic lattice [3].

This research will focus on the study of parabolic equation in two space dimensions. An efficient finite difference discretization method [4] is used to solve the parabolic equations. The explicit finite-difference method has been studied. The explicit method has used to solve the parabolic equations in this research. The finite-difference equations are converted into matrix forms and solved by Red Black Gauss-Seidel iterative.

The using of heterogeneous parallel computer system in solving the mathematical problems by parabolic equations in two space dimensions will be introduced. Parallel computing is a software of using

heterogeneous or homogeneous parallel computer system as a counting source, which coordinate together and been connected with cluster system. Parallel computing is the simultaneous use of multiple compute resources to solve a computational problem [5, 6]. The heterogeneous PC cluster system contains 6 Intel Pentium IV CPUs (each with a storage of 40GB, speed 1.8MHz and memory 256 MB) and two servers (each with 2 processors, a storage of 40GB, speed AMD-Athlon (tm) MP processor 1700++ MHz and memory 1024 MB) are connected with internal network Intel 10/100 NIC under RetHat Linux 9.2 operation are used in this study as [7]. A software package namely Parallel virtual Machine (PVM) has used. PVM as a communication platform permits a heterogeneous collection of Linux or Window computers hooked together by a network to be used as a single large parallel computer [8]. PVM is designed to link computing resources and provide with a parallel platform for running their computer applications, irrespective of the number of different computers are used and where the computers are located. Besides, the PVM model is a set of message passing routine, which allows data to be exchanged between tasks by sending and receiving messages. This study will analyze the performance of the parallel computer in respect of speedup, efficiency, effectiveness, and temporal performance.

The application of the parabolic equation with numerical finite-difference methods has applied to solve a mathematical model in medical field. The mathematical model is converted to matrices form using the finite-difference methods. Then, parallel computing system has chosen to solve mathematical problems. The growth of the brain tumor will be presented in a graph to predict the pattern of this cancer cell growth.

2. The Mathematical Model

The model represents both the avascular and the vascular phase of tumor evolution, and is able to simulate when the transition occurs. The evolution problems can be written as a free-boundary problem in parabolic type joined with an initial-boundary value problem in a fixed domain. The two classes of model dependent variables characterize the physical state of the biological system in both the tumor mass and the outer environments are cell

populations and chemical species. They are different fundamentally. The cell size is much larger than that of the chemical factors and macromolecules. The cells are delimited by a membrane and cannot penetrate each other; they occupy actual physical space. By contrast, the chemical species consist of macromolecules that may diffuse in the intercellular space, attach to the cell membrane or penetrate it, such that they actually do not take up physical space [9]. The cell populations considered important for the process and the chemical factors that influence their motion and proliferation [10].

The mathematical model consists in an evolution equation for the variable $u = u(t, x)$ considered to describe, in time, t and space, x , the physical state of the system. The variable u includes both cell population and chemical factors produced in the environment by interacting cells. The derivation of the model here described is developed on the basis of mass balance equations, also supported by a random walk scheme [3].

Under suitable regularity assumptions one can expand use N, P, Q and R , as well as use $N_{i,j}(t) \approx u(t, x_i, y_j) \Delta V_{i,j}$, and write the word equation above mathematically [2] as:

$$\frac{\partial u}{\partial t} = -\frac{\partial(Pu)}{\partial x} - \frac{\partial(Ru)}{\partial y} + \frac{\partial}{\partial x}(Q \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(Q \frac{\partial u}{\partial y}) + \Gamma - Lu, \quad (1)$$

With $\Gamma(t, x_i, y_j) = \Gamma_{i,j}(t) / \Delta V_{i,j}$ and where the indices (i, j) have been substituted with the dependence of u and of all coefficients on the space variable. The elementary volume centered in the node (i, j) is denoted by $V_{i,j}$ and its volume by $\Delta V_{i,j}$. Finally, all cells in $V_{i,j}$ are considered concentrated in the node (i, j) . While the number of a certain type of cells (or chemical factors) is denoted by $N_{i,j}(t)$ found in the node (i, j) at the time t . Equivalently, one can then write the following general balance law in local form [2, 3, 9, 10, 11] as:

$$\frac{\partial u}{\partial t} = -\nabla \cdot (Wu) + \nabla \cdot (Q \nabla u) + \Gamma - Lu, \quad (2)$$

$\Gamma = \Gamma(u)$ is the generation (proliferation/production) coefficient $L = L(u)$ is the death

/decay coefficient Q is the diffusion coefficient W is the drift velocity field and in two dimensions, $W = (P, R)$.

2.1 Explicit Method

A forward finite divided difference is used to approximate the time derivative.

Consider the two-dimensional of parabolic equations

$$\frac{\partial u}{\partial t} = c \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad c \text{ constant} \quad (3)$$

For numerical solution, one introduces a grid in the (x, y) plane. Let $u_{i,j}^{(n)}$ denote the value of u at the grid point (i, j) at time step n . We consider $I + 1$ grid points in x and $J + 1$ points in y . The boundary points are at $i = 0, I$ and $j = 0, J$. When $i = 1, I - 1$, or $j = 1, J - 1$, boundary values are required, and their value from prescribed boundary conditions are used. Application of any explicit *numerical* method is very straightforward. For example, consider explicit Euler in conjunction with second order finite-difference approximation for the spatial derivatives

$$\frac{u_{i,j}^{(n+1)} - u_{i,j}^{(n)}}{\Delta t} = c \left[\frac{u_{i+1,j}^{(n)} - 2u_{i,j}^{(n)} + u_{i-1,j}^{(n)}}{\Delta x^2} + \frac{u_{i,j+1}^{(n)} - 2u_{i,j}^{(n)} + u_{i,j-1}^{(n)}}{\Delta y^2} \right] \quad (4)$$

Given an initial condition on the grid points $u_{i,j}^{(0)}$, for each one i and j simply marches forward in time to obtain the solution at subsequent time steps. The stability properties of this scheme can be analyzed in the same manner as in the one-dimensional case. Approximating the partial derivatives with centered second differences gives the 5-point discrete Laplacian.

Let $r_x = \frac{c\Delta t}{\Delta x^2}$ and $r_y = \frac{c\Delta t}{\Delta y^2}$, the equation (4)

becomes,

$$u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + r_x [u_{i+1,j}^{(n)} - 2u_{i,j}^{(n)} + u_{i-1,j}^{(n)}] + r_y [u_{i,j+1}^{(n)} - 2u_{i,j}^{(n)} + u_{i,j-1}^{(n)}] \quad (5)$$

It is convenient to let $\Delta x = \Delta y = h$. We shall use the convention of using an order pair of indices (i, j) to denote the point $(i\Delta x, j\Delta y)$ where $i = 0, \dots, \Delta x$ and $j = 0, \dots, \Delta y$.

Solution equation (2.24) for $u_{i,j}^{(n+1)}$ gives

$$u_{i,j}^{(n+1)} = r_x u_{i-1,j}^{(n)} + (1 - 2r_x) u_{i,j}^{(n)} + r_x u_{i+1,j}^{(n)}$$

Leads to the equations

$$\begin{aligned} u_{1,j}^{(n+1)} &= 0 + (1 - 2r_x) u_{1,j}^{(n)} + r_x u_{2,j}^{(n)} \\ u_{2,j}^{(n+1)} &= r_x u_{1,j}^{(n)} + (1 - 2r_x) u_{2,j}^{(n)} + r_x u_{3,j}^{(n)} \\ &\vdots \\ u_{N-1,j}^{(n+1)} &= r_x u_{N-2,j}^{(n)} + (1 - 2r_x) u_{N-1,j}^{(n)} + 0 \end{aligned}$$

These can be written as

$$\begin{bmatrix} (1 - 2r_x) & r_x & \cdots & \cdots & 0 \\ r_x & (1 - 2r_x) & r_x & & \vdots \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & r_x & (1 - 2r_x) & r_x \\ 0 & \cdots & \cdots & r_x & (1 - 2r_x) \end{bmatrix} \begin{bmatrix} u_{1,j}^{(n)} \\ u_{2,j}^{(n)} \\ \vdots \\ \vdots \\ u_{N-1,j}^{(n)} \end{bmatrix} = \begin{bmatrix} u_{1,j}^{(n+1)} \\ u_{2,j}^{(n+1)} \\ \vdots \\ \vdots \\ u_{N-1,j}^{(n+1)} \end{bmatrix}$$

Where $r_x = \frac{c\Delta t}{\Delta x^2}$. or as $Au_j = u_{j+1}$

2.2 Red Black Gauss-Seidel Iteration Method

Red Black Gauss Seidel algorithm is used to implement the parallel algorithm in solving the finite difference equation. The iterative method contains 2-sub domain, Ω^R and Ω^B . There is a communication between Ω^R and Ω^B [5]. The calculation of this method is shown as follow:

i Grid calculation at Ω^R :

$$u_i^{(n+1)} = r[u_{i-1}^{(n+1)} + u_i^{(n+1)}] + b_i, \quad i = 1, 3, 5, \dots, n \quad (6)$$

ii Grid calculation at Ω^B :

$$u_i^{(n+1)} = r[u_{i-1}^{(n+1)} + u_i^{(n+1)}] + b_i, \quad i = 2, 4, 6, \dots, n + 1 \quad (7)$$

The simple Gauss-Seidel update strategy is more appropriate in sequential program while Red Black Gauss-Seidel parallel algorithm is more appropriate for the solution of large mathematical problem compares to the Gauss-Seidel with sequential algorithm.

2.3 The Discretization of the Model

Equations

Based on central finite difference method, the discretization is shown as follow,

Red Black Gauss-Seidel algorithm can be implemented the parallel algorithm in solving the evolution equation for brain tumor prediction as,

$$\begin{aligned} & \frac{N_{ij}^{(k+1)}(t + \Delta t) - N_{ij}^{(k)}(t)}{\Delta t} \\ &= [P_{i-1}^j N_{i-1,j}^{(k+1)}(t) - P_i^j N_{ij}^{(k)}(t)] + [R_{i,j-1}^{j-1} N_{i,j-1}^{(k+1)}(t) - R_i^j N_{ij}^{(k)}(t)] \\ &+ [Q_{i-1}^j N_{i-1,j}^{(k+1)}(t) - (Q_{i-1}^j + Q_{i+1}^j) N_{ij}^{(k)}(t) + Q_{i+1}^j N_{i+1,j}^{(k)}(t)] \\ &+ [(Q_{i-1}^{j-1} N_{i,j-1}^{(k+1)}(t) - (Q_{i-1}^{j-1} + Q_i^{j+1}) N_{ij}^{(k)}(t) + Q_i^{j+1} N_{i,j+1}^{(k)}(t)] \\ &+ \Gamma_{ij} - L_{ij} N_{ij}^{(k)}(t). \end{aligned} \quad (8)$$

We have formulated the discrete model that involves using a discretized form of the PDEs. The entire fixed computational domain $0 \leq x \leq 1$ and $0 \leq y \leq 1$ are each discretized using equally spaced meshes, the interface is a mesh point, corresponding both to $x = 1$ and $y = 1$. The domain occupied by the tumor is embedded into a larger fixed, time-independent, computational domain D that is discretized using a uniform Cartesian mesh with $\Delta x = \Delta y = h$.

Assume that the tumor has grown approximately independent on the initial cell density ($u \approx 200 \pm 20 \mu\text{m}$). Initial conditions are given in the form of a discrete set of Gaussian bumps of width of the order of the average cell diameter ($\sim 30 \mu\text{m}$), placed at random position with uniform probability over a square surface [10]. The small initial proliferating cell density (0.01) at the centre of the tumor corresponds to the early stages of tumor growth [12]. While the death coefficient from 10^{-7} to 10^{-5} are taken.

3. High Performance Computing System

High Performance Computing System [13] computing provides infrastructure for solving distributed problem by sharing, selection and aggregation of distributed resources at runtime depending on their availability, performance, cost and users' quality of service requirements. High Performance Computing is a computer system that made up of many specialized processing units working together in parallel. Parallel computing is an execution of the same task on multiple processors in order to obtain

results faster. The idea is based on the fact that the process of solving a problem usually can be divided into smaller tasks, which may be carried out with some coordination. A huge number of software systems have been designed for the programming of HPC, both for the operating system and programming language level [14]. Parallel Virtual Machine (PVM) is one of them to develop parallel programs executable in networked UNIX computers [8]. PVM allows a heterogeneous collection of workstations and supercomputers to function as a single high-performance parallel machine [8]. PVM is designed to link computing resources and provide users with a parallel platform for running their computer application. It is capable of connecting the combined resources of typically heterogeneous networked computing platforms to deliver high levels of performance and functionality.

3.1 The Prediction of the Brain Tumor Growth through Explicit

In this study, we have studied the finite difference approximation that is explicit, in solving the two-dimensional parabolic equation. Here the explicit method is applied in solving the mathematical model by using a suitable sequence iteration method of Red Black Gauss-Seidel parallel algorithms with some minimize requirements such as PVM, C programming and Linux environment on HPC system. The results from the parallel algorithm using C programming are shown in Table 1 for 30 days.

Table 1 shows that the data of the number of brain tumor cells had been computed using explicit methods. The values are used to predict the growth of the tumor cells more perceptibly within 30 days. This result is relatively correct, exact and the growth rate of the tumor cells can be perceived compare to actual data. The time step that has been used is 200 with the round in 4, 8, 12, 16 and 20 respectively. These caused the explicit to work properly in capturing the real image of the tumor cells growth. By using this method, the time step can be made larger without fret about excessive buildup of round off error. By using the data from the Table 1, the pattern of the brain tumor growth can be shown as the following graphs in different days.

The graph in Figure 1 shows the movement in values of the tumor cells growth increasingly across two dimensional continuous curves.

Table 1: The number of cancer cells' growth for 30 days

Days Grid	6	12	18	24	30
1	235.59	314.344	429.717	575.287	743.405
2	235.608	314.368	429.751	575.333	743.464
3	235.621	314.386	429.776	575.367	743.507
4	235.629	314.398	429.792	575.389	743.536
5	235.633	314.404	429.801	575.4	743.551
6	237.386	316.794	433.089	579.822	749.281
7	237.404	316.819	433.123	579.867	749.341
8	237.417	316.837	433.148	579.901	749.385
9	237.426	316.849	433.165	579.924	749.414
10	237.43	316.855	433.173	579.935	749.429
11	238.686	318.611	435.602	583.213	753.686
12	238.705	318.636	435.636	583.259	753.746
13	238.718	318.655	435.662	583.293	753.79
14	238.726	318.667	435.678	583.316	753.82
15	238.73	318.673	435.687	583.327	753.834
16	239.528	319.814	437.272	585.472	756.626
17	239.546	319.839	437.306	585.518	756.686
18	239.56	319.858	437.332	585.552	756.731
19	239.568	319.87	437.349	585.575	756.76
20	239.572	319.876	437.357	585.586	756.775
21	239.942	320.414	438.107	586.603	758.1
22	239.96	320.439	438.141	586.65	758.16
23	239.973	320.458	438.167	586.684	758.204
24	239.982	320.47	438.184	586.707	758.234
25	239.986	320.476	438.192	586.718	758.249

The curves shown above are smooth and relevant in describing the growth rate of the tumor cells of every 6 days until 30 days. They imply that explicit method is appropriate to apply the mathematical model in solving the parabolic problem. However according to Hang [15], Crank Nicolson method is the best technique compare to the explicit methods in solving a one-dimensional brain tumor problem with its unconditional stability. The graph formed by explicit method undoubtedly will surely produce smoother and relevant curve than the Crank Nicolson method in visualizing the growth of brain tumor cells in two-dimensional space.

According to the Table 1, we noticed that the tumor is growing from 200 cells up to about more than 700 cells within one month. This implies that the tumor grows about 500 cells per month. The growth rate is about 17 cells per day. This also shows that the evolution of the tumors cells is high and triple the time of the initial growth. However, only a small

portion of actual tumor can be predicted. For overall, we conclude that the growth of tumor cells is slow but the process of evolution is quick [9].

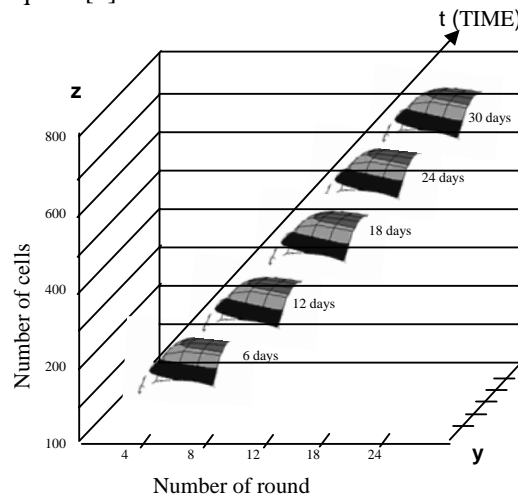


Fig. 1: Expansion rate of brain tumor

Figure 1 shows that the growth rates of tumor cells are increasing day by day. These growth rate increases till the first 24 days consistently. After 24 days, the tumor cells become highly active in evolution. The tumor cell will grow more than 800 cells after 30 days.

In more details, growth coefficients can be evaluated by looking at the mitotic rate of cells, and death coefficient, L by looking at the rate at which cells die in the absence of mitosis. The amount of nutrient which promotes proliferation or that cause death can be evaluated by looking at the behaviour of the cells for decreasing amounts of nutrient [2]. It is slightly more difficult to obtain direct measurements of diffusion coefficients, Q and drift coefficients, $W = (P, R)$. However we obtain both of these results by data experiment. The values we have taken to run the algorithm

are, 10^{-5} and 10^{-7} for the drift coefficients of P and R respectively while the diffusion coefficient, Q is 10^{-3} . For each of the proliferation coefficient, Γ and death coefficient, L the values which have been taken are 10^{-2} and 10^{-8} .

4. Performance Analysis and Discussion

There is a master task and a number of slave tasks in the PVM performance of the modeling codes. The master task is responsible to divide the model domain into sub domains and allocate them to slave tasks. The slave tasks that involve the actual computation will execute time matching and communicate after each time step. The performances of parallel algorithm with the sequence algorithm are compared in Table 2.

Table 2: Time, convergence and number of iteration for parallel algorithm and sequence algorithm

	Red-Black Gauss Seidel with PVM (8 CPU)	Gauss Seidel with Sequence Algorithm (1 CPU)
Time (second)	10.90019	83.153291
Convergence	2.3911e-2	2.3911 e-2
Number of iteration	200	200

Table 2 shows that the executive time, convergence and number of iteration for both the Red-Black Gauss-Seidel (GSRB) with PVM and Gauss Seidel (GS) with sequence algorithm in solving the mathematical model. The table shows that the executive time for GSRB with PVM is about 8 times faster than GS with sequence algorithm. This signifies that parallel algorithm is better than sequence algorithm. Besides, the convergence and number of iteration performed by both of the algorithms are same.

After running the parallel computing based on 8 numbers of CPU, the parallel performance will be analyzed from the aspect of time execution, speedup, efficiency, effectiveness and temporal performance. The following outcomes show that the increasing of the number of processors comes with the decreasing of time execution for speedup, efficiency, effectiveness and temporal performance.

Table 3: Time execution, speedup, efficiency, effectiveness, and temporal performances against different number of processors.

Number of processor	Time execution (Second)	Speedup	Efficiency	Effectiveness	Temporal Performance
1	83.153291	1	1	0.012025982	0.012025982
2	41.84082	1.987372	0.993686	0.023749205	0.023900195
3	27.8878808	2.981708	0.993903	0.035639322	0.035857963
4	20.98909	3.961739	0.990435	0.047188072	0.047643800
5	16.9	4.920313	0.984063	0.058228557	0.059171598
6	14.262847	5.830063	0.971677	0.068126448	0.070112229
7	12.362682	6.726153	0.960879	0.077724154	0.080888597
8	10.90019	7.628609	0.953576	0.087482527	0.091741520

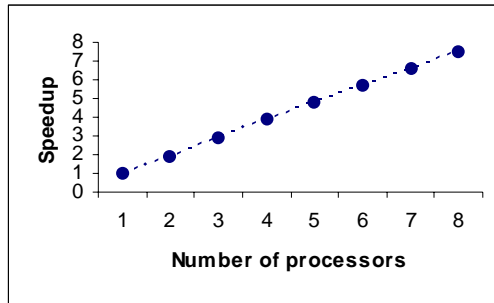


Fig. 2: Speedup vs. number of processors

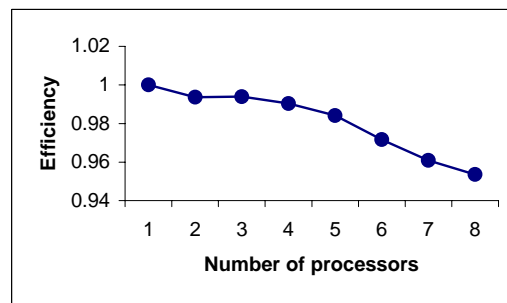


Fig. 3: Efficiency vs. number of process

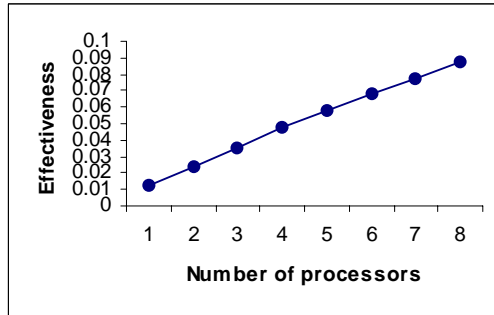


Fig. 4: Effectiveness vs. number of processors

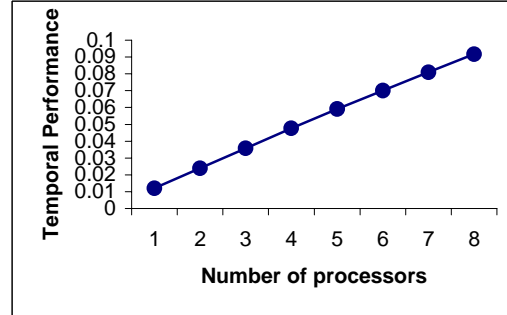


Fig. 5: Temporal performance vs. number of processors

Based on the table 3, the speedup increases when the numbers of CPU increase. Actually the real graph of speedup against the number of processors is not a straight line. It is due to the effect if the communication between the processors. Since the number of processors (8 processors) is limited, a straight line is obtained in this research. Besides, the distributed memory hierarchy causes the reduction of the time consuming access to a cluster of workstations. According to Amdahl's Law, the speedup increases with the number if processors increase up to the certain level.

Fig. 2 shows that the efficiency is decrease and less than 1 when the numbers of processors increase due to the communication involved. The factors that cause the decrease of efficiency are because of the imbalance workload, which are distributed among the different processors. The idle time, time startup and waiting time of all the processors to complete the computations are also the factors of the decrease of efficiency.

Table 3 shows that the effectiveness increases with the increasing number of CPU. The achievement of result for the increasing effectiveness is based on the increasing of the speedup. Moreover the effectiveness graph increases when the number of processors is

added. A straight line has been formed for the graph due to the communication of the 8 processors in this parallel computing system.

Fig. 5, shows that the temporal performance increases while the number of processors increases. The graph shows a straight line due to the decreasing of execution time exceedingly versus with the number of processors.

As summarise to this study, the analysis shows that the performance of the parallel algorithm is enhanced by increasing of the number of processors from the aspect of speedup, efficiency, effectiveness and temporal performance. Parallel computers become more famous since these computers provide many order of magnitude raw computing power than traditional supercomputers at much lower cost. Parallel computers consisting of thousands of processors are now commercially available. Systems with thousands of such processors are known as massively parallel. The availability of massively parallel computers has created a number of challenges. They open up new border in the application of computers, by which many unsolvable (previously) problems can be solved effectively.

The results of the analysis for the performance measurements have proved that parallel algorithms are considerably better than the

sequential algorithms from the aspect of speedup, efficiency, effectiveness and temporal performance. The Red-Black Gauss Seidel is found to be suitable for parallel implementation on the PVM efficiently [16]. Besides, the communication of processors and computing times are always effected the results of speedup, efficiency, effectiveness and temporal performance.

The computing of Two-Dimensional parabolic equation of brain tumor growth is well suite in using High Performance Computing System because it involved a large space of matrix algorithm.

5. Conclusion

In the last twenty years, most of the models of solid tumor growth were written and developed by several mathematical models, using different approaches. This indicates that the main directions for future mathematical research in this field.

In this study, regarding the growth of brain tumors, a 2-Dimensional parabolic model has been chosen to solve this problem by using standard finite difference method.

Explicit method is a numerical finite difference method, which had been used to solve the evolution equation in this study. From the graphically presented result, the explicit method can be used to show the relevant curve correctly to predict the growth of brain tumors compare to actual data [10].

From the analysis, the parallel computing with PVM system can be concluded as a well suite performance tools in solving the grand challenge of mathematical problem. The parallel processing has saved a lot of execution time compare to the sequence processing.

In this study, the explicit method has been used to predict the growth of brain tumor cells in two-dimensional space within 30 days. This is an extended research of one-dimension brain tumor cells from Hang [15].

As our future research, to improve the speed and performance of the parallel processing systems, the number of processors that are used to solve the mathematical model can be increased. Besides, MPI (message-passing interface) can be used in replace of PVM to solve the large scale problems in High Performance Computing systems. The numerical schemes such as AGE (Alternating Group Explicit) and IADE (Iterative Alternating Decomposition Explicit) can be

useful for numerical parallel processing as well, since they are convergent and have unconditional stability.

Acknowledgement

This research is supported in part by the Ministry of Science, Technology and Innovation Malaysia through e-Science funding (Grant No: 78075) and the authors are grateful to Ibnu Sina Institute, University Technology Malaysia for the excellent support to this research.

References

- [1] N. Alias, M. R. Islam, and N. S. Rosly, "A Dynamic PDE Solver for Breasts' Cancerous Cell Visualization on Distributed Parallel Computing Systems", *Proceedings of the 8th International Conference on Advances in Computer Science and Engineering (ACSE 2009)*, Phuket, Thailand, 2009, pp. 138-143.
- [2] E. De. Angelis, and L. Preziosi, "Advection-Diffusion Models for Solid Tumour Evolution in Vivo and Related Free Boundary Problem", *Mathematical Models and Methods in Applied Sciences*, vol. 10, No. 3, 2000, pp. 379-407.
- [3] N. Bellomo, and L. Preziosi, "Modelling and Mathematical Problems Related to Tumor Evolution and Its Interaction with the Immune System", *Mathematical and Computer Modelling*, Vol. 32, 2000, pp.413-452.
- [4] A. R. A. Anderson, and M. A. J. Chaplain, "Continuous and discrete mathematical models of tumour-induced angiogenesis", *Bull. Math. Biol.* Vol. 60, 1998, pp. 857-899.
- [5] B. Wilkinson,, and M. Allen, *Parallel Programming Techniques & Applications Using Networked Workstations & Parallel Computers*, (2nd edition), Pearson Education Inc., 2004.
- [6] C. Xavier, and S. S. Iyengar, *Introduction to Parallel Algorithms*, John Wiley & Sons, Inc., 1998.
- [7] V. Kumar, A. Grama, A. Gupta, and G. Karypis, *Introduction to Parallell Computing Design and Analysis of Algorithms*. Redwood City, California, The Benjamin/ Cummings Publishing Company, Inc., 1994.
- [8] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam, *PVM: Parallel Virtual Machine, A User's Guide and Tutorial for Network Parallel Computing*, Massachusetts, London, England, The MIT Press, Cambridge, 1994.
- [9] C. S. Hoge, B. T. Murray, and J. A. Sethian, *Simulating complex tumor dynamics from avascular to vascular growth using a general*

- level-set method*. ICES Report no. 0529, 2005.
- [10] L. Preziosi, *Modeling tumor growth and progression*. Progress in Industrial Mathematics at ECMI 2002. In: A. Buikis, R. Ciegis, A.D. Fitt, (Eds.), Springer, 2004, pp. 53-66.
- [11] D. Ambrosi, and L. Preziosi, "On The Closure of Mass Balance Models for Tumor Growth", *Mathematical Models and Methods in Applied Sciences*, Vol. 12, No. 5, 2002, pp. 737-754.
- [12] T. L. Soon, and A. K. Cheng, "A Numerical Simulation of Avascular Tumor Growth", *Anziam J.*, Vol. 46, No. E, 2005, pp. C902-C917.
- [13] F. C. Kesselman, *The Grid: Blueprint for a new computing infrastructure*, SF, CA, Morgan Kaufmann Publication, 1999.
- [14] F. Michael J., *Computer Architecture: Pipelined and Parallel Processor Design*. Boston: Jones & Bartlett, 1995.
- [15] H. S. Pheng, *High Performance Simulation for brain Tumors Growth Using Parabolic Equation on Heterogeneous Parallel Computer System*, Universiti Teknologi Malaysia, Master Degree Thesis, 2005.
- [16] M. Silva, "Cache Aware Data Laying For The Gauss-Seidel Smoother", *Electronic Transactions on Numerical Analysis*, Vol. 15, 2003, pp. 66-77.