Simulating Water Diffusion in Tissues with an Improved Finite Difference Method

J. Xu[1], M. D. Does[1], J. C. Gore[1]

[1]Institute of Imaging Science, Vanderbilt University, Nashville, TN, United States

Introduction: The Finite difference (FD) method [1] is well suited for numerical studies of water diffusion in tissues because it performs better and is computationally more efficient than the Monte Carlo method [2]. In addition it can be adapted to study systems that are more realistic than those based on simple geometrical structures [3]. However, there are two main limitations to the use of the FD method; the edge effect, which causes large errors, especially with long diffusion times, and the computing time, which is still long for complex tissues. We have developed a novel hybrid discretization scheme algorithm (HDSA) to eliminate the edge effect, and a tightly-coupled FD model that uses parallel computing to decrease the computing time. To test those two improvements, we implemented a simulation of water diffusion in a grid of dimension 100×100×100. The results show that the HDSA constrains the computing errors to a level of less than 5% for diffusion times as long as 100ms, and the tightly-coupled FD model decreases the computing time from 16 hours to 24 minutes. The explanation and limitation of HDSA have also been evaluated.

Methods and Results: The edge effect in simulating water diffusion with the FD method was noted by Hwang et al. [1]. They found that the apparent diffusion close to the boundaries was highly restricted in the FD simulation. They chose the unaffected central one third grid points only for a diffusion time of 37 ms. Unfortunately, if the diffusion time becomes longer, even those points are not reliable and the only solution is to choose even fewer points or, equivalently, a larger computational domain. This is either computationally expensive or a source of significant errors. In this work, we developed an HDSA to eliminate the edge effect and obtained reasonable results. HDSA employs both the explicit FTCS [4] and a fully implicit discretization scheme but has quite different update rules from the hopscotch algorithm [5]. It divides the whole grid into three pools: interior points (green in Fig.1), edge (pink) and corner (blue). Each pool has a different discretization scheme. The updating rules of HDSA in each time step can be briefly expressed in the following three steps: 1. Update all interior points using the explicit FTCS or the hopscotch algorithm. This part is conventional and the same as Hwang's method. 2. Update the edge points except the corners by solving the fully implicit discretized Bloch-Torrey equations, namely,

$$M_{i,j,k-1}^{n+1} = \frac{dz^2}{D_z dt}[(1+i\gamma g(t)xdt + 2\frac{D_x dt}{dx^2} + 2\frac{D_y dt}{dy^2} + 2\frac{D_z dt}{dz^2})M_{i,j,k}^{n+1} - M_{i,j,k}^{n} - \frac{D_x dt}{dx^2}M_{i+1,j,k}^{n+1} - \frac{D_x dt}{dx^2}M_{i-1,j,k}^{n+1} - \frac{D_y dt}{dy^2}M_{i,j+1,k}^{n+1} + -\frac{D_y dt}{dy^2}M_{i,j-1,k}^{n+1} - \frac{D_z dt}{dz^2}M_{i,j,k+1}^{n+1}],$$ where

M is the local magnetization, subscripts denote the grid indices, superscripts the time indices. Notice that an edge point is solved in a 7-point stencil, instead of being updated from all its nearest neighbors. 3. To update the remaining points at corners, the Cartesian coordinate is rotated clockwise by 45° and step_2 is repeated in the rotated coordinate. For biological tissues, the diffusion coefficients are no longer uniform and jump probabilities must be used to describe the movement of water molecules between grid points. Notice that the information needed to update all points is totally inside the computational domain in HDSA and the artificial computing boundaries are virtually removed, which explains why HDSA eliminates the edge effect.

Although it is more efficient than the Monte Carlo method, the FD method is still very time-consuming, especially for 3D heterogeneous tissues. We developed a tightly-coupled FD with parallel computing to address this problem. This model manipulates multiple CPU's and arranges them into a 3D Cartesian topology. All of the CPU's work synchronously and each CPU deals only with a sub-block of the computing domain. To avoid the influence of the artificial boundaries of sub-blocks, CPU's swap boundary information among neighbors in every time step.

To test these two improvements to the conventional FD method, we implemented them to simulate the diffusion of water in a grid of dimension 100×100×100. All programs were written in C with MPI and performed on a high performance cluster. The simulated signals in the x-direction using a conventional FD and our improved FD methods are shown in Fig.(2). The conventional FD method mistakenly increases the simulated signals with respect to the diffusion times. Using our modified approach, the edge effect is mostly eliminated even for very long diffusion time (100ms). Fig.(3) shows how the computing time decreases as the number of CPU's used in the simulation is increased.

Discussion: HDSA actually serves as a kind of boundary condition in solving the Bloch-Torrey equations. However, we found out that HDSA is more sensitive to the amplitudes of the gradients than the conventional FD method. Empirically, HDSA produces larger errors when a criteria factor (b×dx×dy×dz) is larger than 100ms•μm. Fortunately, in the range of interest for space step size 0.1μm - 1μm, HDSA only produces larger errors if b > 10^7 – 10^{10}s/cm^2 that is far beyond the practical values. So we can use HDSA freely in our simulation except for the extremely high b value imaging. It should also be noted that for a certain problem, the tightly-coupled FD with parallel computing technique can not decrease the total computing time indefinitely as more CPU's are used. There is a limit because of the time required for communication between the CPU's for swapping information.

References: (1) Hwang, MRM, 50:373-82 (2003). (2) Szafer, MRM, 33:697-712(1995). (3) Ford, MRM, 37:387-94(1997). (4) Fletcher, Computational techniques for fluid dynamics, Springer-Verlag, 1988.p217. (5) Mitchell & Griffiths, The finite difference method in partial differential equations, John Wiley&Sons, 1980. p77-81

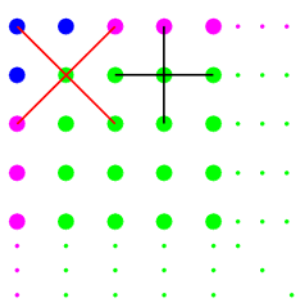

Fig.1 Schematic diagram of HDSA

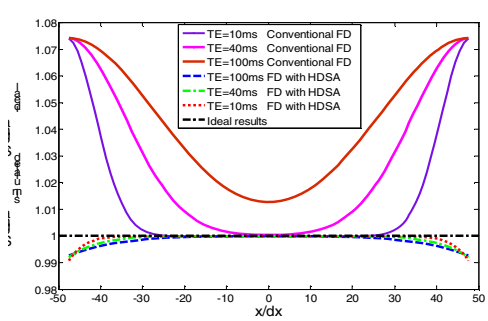

Fig.2. Comparison of the simulated results of the conventional FD and the FD with HDSA

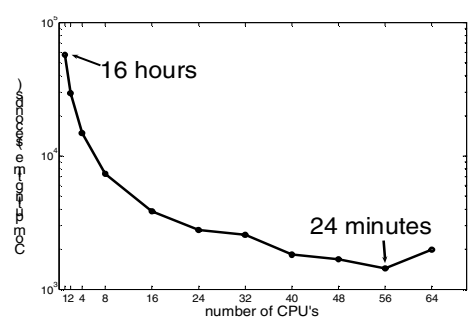

Fig.3. Computing time versus the number of CPU's.