HAL
open science

Diffusion MRI Simulation with the Virtual Imaging Platform

Lihui Wang, Sorina Camarasu-Pop, Glatard Tristan, Yue-Min Zhu, Isabelle E. Magnin

To cite this version:

Lihui Wang, Sorina Camarasu-Pop, Glatard Tristan, Yue-Min Zhu, Isabelle E. Magnin. Diffusion MRI Simulation with the Virtual Imaging Platform. journées scientifiques mésocentres et France Grilles 2012, Oct 2012, Paris, France. hal-00766081

HAL Id: hal-00766081

https://hal.science/hal-00766081v1

Submitted on 17 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Diffusion MRI simulation with the Virtual Imaging Platform

Lihui Wang, Sorina Camarasu-Pop, Tristan Glatard, Yue-Min Zhu, Isabelle E. Magnin
{prenom.nom}@creatis.insa-lyon.fr, Université de Lyon, CREATIS ; CNRS UMR5220 ; Inserm U1044 ; INSA-Lyon ; Université Lyon 1.

Overview

Myocardial fiber architecture plays an important role in ensuring normal mechanical and electrical properties of the heart. However, due to limitations of existing imaging techniques, little is known about the fiber structure of *in vivo* human heart. Diffusion magnetic resonance imaging (dMRI) is one of the most potential techniques for mapping in-vivo cardiac fiber architecture. However, in the absence of the ground truth information and with influence of MRI scanner noise and artifacts, it is difficult to evaluate how well the diffusion characteristics calculated from experimental dMRI reflect the actual cardiac fiber microstructure properties. To cope with this issue, we propose a dMRI simulator that simulates the diffusion of water molecules within a virtual cardiac fiber model with a Monte-Carlo approach, and calculates the diffusion images using the spin quantum theory. The noise-free and non-artifacts images from this simulation enable the evaluation of dMRI image processing algorithms and the optimization of imaging parameters based on the model ground truth.

To support its computing needs, the simulator was ported to the European Grid Infrastructure (EGI) through the Virtual Imaging Platform (VIP). The simulation of 16 billion water molecules, which would require 8 CPU years on a representative machine of the biomed virtual organization (VO), can now be achieved in one month. Much manual intervention is still required to compute such simulations though.

Challenge and computing needs

Despite decades of intensive cardiovascular research in basic and clinical sciences, very little information exists on the intrinsically three-dimensional (3D) fiber architecture of the human heart, which is however fundamental for a comprehensive understanding of the relations between mechanical function, hemodynamic, and adaptive structural changes in cardiac diseases. The only reason for this situation is that there is currently no means to access such *in vivo* 3D fiber architecture. Diffusion magnetic resonance imaging (dMRI) [1] including diffusion tensor imaging (DTI) [2] appears as the new and perhaps the only way to access the 3D fiber architecture non-invasively for the human heart.

However, due to the tissue property itself of the myocardium and the motion sensitivity of dMRI technique, it is currently not possible to obtain *in vivo* cardiac fiber architecture. Moreover, even for the ex vivo dMRI experiments, because of the limits of MRI scanner, the image resolution is about 2 mm which is not sufficient for investigating cardiac fiber properties [3]. To tackle this problem, we propose here a radically different approach which consists in modeling the 3D fiber architecture using multi-physical data from different imaging modalities working at different spatial resolutions and simulating the diffusion magnetic resonance images for both *ex vivo* and *in vivo* human heart.

To this end, the myocardium of the human heart was imaged using polarized light imaging (PLI) [4] which gives us a map of fiber orientation with spatial resolution of 100×100×500 µm. Based on this information, an *ex vivo* virtual cardiac fiber structure (VCFS) was modeled. The diffusion behavior of water molecules in this VCFS was simulated by means of Monte-Carlo method [5]-[7], and the diffusion images in different directions were calculated using spin quantum theory.

It is well known that Monte-Carlo simulation usually requires important computation resources to guarantee the simulation accuracy, but this requirement becomes even more stringent in our case. Knowing that for one fetus heart there are 707,672 fibers constructed in our VCFS model, and that 24,000 water molecules are simulated for each fiber, the simulation time is estimated at approximately 8 CPU years on a machine representative of the biomed VO. Grid computing provides an interesting solution to reduce the computation time.

Simulation parallelization

The straightforward static parallelization approach for Monte-Carlo simulations equally splits the simulation into p jobs, each job simulating m molecules in a subset of f cardiac fibers among the total number of fibers F. This results in Algorithm 1, executed on the computing nodes (workers).

Worker:

```
Input: n1, ..., nf: the f indices of the cardiac fibers to simulate
Download input data and simulation code
for i from 1 to f
    for j from 1 to m
        randomly choose one initial molecule position in fiber ni
        simulate diffusion of molecule and contribution to DWI signal
    end
```

```
8.  end
9.  write and upload molecule positions and signal contributions
```

Algorithm 1: static parallelization of the simulation.

As shown in [12], this approach is notably inefficient on large, heterogeneous, unreliable infrastructures such as EGI for the following reasons:

- The simulation time is bound by the performance of the slowest job.
- Failed jobs have to be resubmitted.
- Failed jobs do not contribute to the simulation (results are lost).

To cope with these issues, the master-slave proposed in [12] was implemented. Checkpointing is also used, to avoid losing important amounts of computation in case the jobs are killed after several hours of execution. Algorithm 2 shows the pseudo-code of the parallelized application. It has the following properties:

- The number of molecules simulated by a job is adjusted to the speed of its machine (see while loop in steps 3-11 of worker algorithm).
- Failed jobs can be ignored: failures would only lead to increased numbers of simulated molecules by successful jobs.
- Failed jobs contribute to the simulation thanks to checkpointing (step 8 of worker algorithm).

Each job adapts its checkpointing period (see step 10 of worker algorithm) as follows: $C = NJ * T * \theta$, where C is the checkpointing period expressed in molecules, NJ is the number of parallel computing jobs, T is the time in seconds needed to transfer the checkpointed result to/from the storage element, and θ is the computing speed of the worker node expressed in molecules per second. This ensures that checkpointed results can be downloaded and merged before the next checkpoint. The checkpointing frequency is thus adapted so that the merger can download and merge the results checkpointed by all parallel computing jobs between two checkpoints.

Master:

```
1.  N = F x m
2.  n = 0
3.  while ( n < N ) do
4.    n = number of water molecules simulated by running and completed tasks
5.  end
6.  send stop signal to all tasks
7.  cancel queued tasks
```

Worker:

```
1.  Download input data and simulation code
2.  N = F x m
3.  while stop signal not received and n < N do
4.      randomly select one cardiac fiber from uniform distribution
5.      randomly select one initial molecule position in the previously selected cardiac fiber
6.      simulate diffusion of the molecule and contribution to DWI signal
7.      every C iteration:
8.              checkpoint molecule positions and signal contributions
9.              report number of simulated molecules to master
10.             adjust checkpointing frequency C
11. end
```

Algorithm 2: dynamic parallelization of the simulation with checkpointing.

Note that Algorithm 2 does not guarantee that an equal number of molecules is simulated in each cardiac fiber. Step 4 of the worker algorithm only guarantees that they are uniformly distributed in cardiac fibers.
Results are finally merged together using simple additive operations. The merging order is not constrained as merging operations are associative and commutative. Thanks to checkpointing, results can be incrementally merged during the simulation. Merging is a critical task since partial results have to be transferred from sites distributed world-wide.

Tools used and difficulties encountered

The simulator was ported to the VIP portal described in [8]. This environment is a web-portal accessible at http://vip.creatis.insa-lyon.fr, which gives access to some 10 simulators and counts more than 200 registered users. In a few clicks, users can launch parallel executions, monitor their status and download the final results when they are ready. The

simulators available in the portal are described as MOTEUR workflows [9], from which tasks are generated and executed using DIRAC [10] pilot jobs on the resources available to the biomed VO within the EGI grid. All computing resources supporting the biomed VO are available for the computation (about 100 clusters). DIRAC is responsible for matching simulation jobs to computing nodes. Synchronization between master and workers is ensured by a specific service developed in the DIRAC framework.

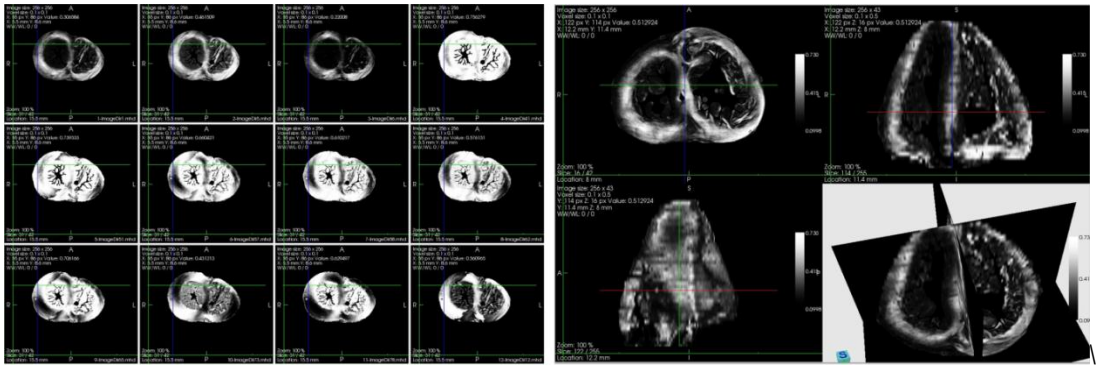
The simulation code was significantly instrumented to support checkpointing, and to properly handle random generators to ensure statistical independence of all checkpoints. The simulator is developed in Matlab. To avoid licensing issues on distributed computing nodes, the code is compiled using the Matlab Compiler and deployed on the fly on the computing nodes with the Matlab Compiler Runtime (MCR).

Although the simulator was ported to VIP, which should allow end-users to autonomously launch their computations, the simulation still had to be handled by grid experts with much manual intervention. The main difficulties were:

1) The very long duration of the simulation jobs (a few weeks if not interrupted). Simulation jobs were progressively killed by sites or system errors. New jobs had to be resubmitted periodically to maintain a sufficient number of running jobs. The simulation was launched in 10 different job batches, each with 500 jobs. The workflow had to be adapted so that the different job batches shared the same logical folders and that partial results could be merged all together to provide a single final output.
2) The merging of partial results distributed across the sites where they were produced. Each batch of 500 jobs produced on average 2000 files of approximately 500 megabytes each. Due to scheduled and unscheduled downtimes of biomed storage sites, file transfers had to be retried several times, sometimes over a few days. In addition, the merging process was memory-intensive and consumed very little CPU. It was thus manually launched on a local dedicated machine with 4 GB of RAM.

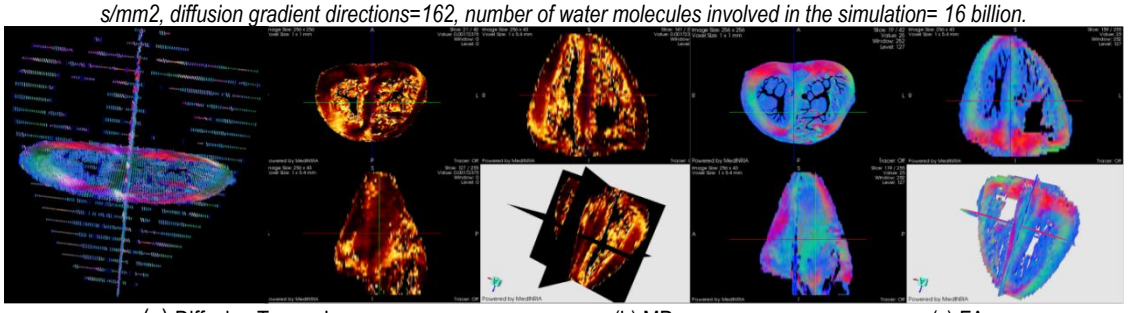
Simulation results

A 40-week fetus heart was imaged using the PLI system and, based on its angle map, the cardiac fiber was modeled by the orientated cylinder. The motion of each water molecule in this kind of structure was mimicked by the random walks and its displacement during one certain period was tracked, from which the phase shift was derived and diffusion signal attenuation caused by the diffusion was calculated. The sub-figure (a) in Fig.1 shows the simulated diffusion images in 12 different directions, sub-figure (b) is the 3D diffusion weighted (DW) image along z direction. From these DW images, the diffusion tensor field, fractional anisotropy (FA) and mean diffusion coefficient (MD) were calculated accordingly, as shown in Fig.2 The combination of PLI and DWI provides us not only the fiber orientation distribution but also FA and MD information. It takes advantage of the merit of high spatial resolution of PLI which compensates the insufficiency of experimental dMRI.



(a) DW images in 12 directions for one slice (b) DW image for the whole heart in one direction

Fig. 1 The simulated diffusion weighted images for ex-vivo fetus heart. Simulation parameters: diffusion time=200 ms, b value= 2288 s/mm2, diffusion gradient directions=162, number of water molecules involved in the simulation= 16 billion.



(a) Diffusion Tensor Image (b) MD (c) FA

Fig. 2 Diffusion Tensor, MD and FA images. Diffusion coefficient used in the simulation is 10^{-3} mm^2/s, the simulated MD ranges from 4×10^{-4} mm^2/s to 7×10^{-4} mm^2/s, FA changes from 0.31 to 0.96.

These results correspond to 8 CPU years and were obtained in about one month using the resources available to the biomed VO on the EGI grid. The simulation was executed in production conditions using the VIP portal. The speed-up of each batch of simulation jobs was of approximately 140, but if we take into account the total time elapsed between the job batches, the average speed-up of the complete simulation is close to 80.

Perspectives

The simulated noise-free and non-artifact images can be used for evaluating diffusion image processing algorithms, such as denoising, k-space reconstruction, and motion correction. In addition, the simulation can help optimize the imaging parameters for high-b value imaging and q-space imaging. Furthermore, *in vivo* cardiac fiber models can be constructed based on PLI data and motion information from MRI cine sequences and as a result *in vivo* DW images can be simulated.

As detailed in [12], load-balancing among computing resources is close to optimal for the simulation phase. However, there is still room for improving performance, especially in the merging phase: (i) the merging phase could be improved by using multiple parallel mergers as done in [13], (ii) the influence of the checkpointing frequency on the merging phase could be studied, (iii) the placement of partial simulation results on different storage locations could be investigated to reduce the cost of the merging phase. These could reduce the amount of manual intervention by grid experts which, to date, is still required to conduct such experiments.

Acknowledgement

We specially thank P.S. Jouk and Y. Usson for making the set of foetal heart data available for this study. The authors also acknowledge the support of EGI and in particular France Grilles for providing computing resources on the grid infrastructure.

References

[1] L. Minati and W. P. We, “Physical Foundations , Models , and Methods of Diffusion Magnetic Resonance Imaging of the Brain: A Review,” vol. 30, no. 5, pp. 278-307, 2007.

[2] D. Le Bihan, JF. Mangin, C. Poupon, C.A. Clark, S. Pappata, N. Molko and H. Chabriat, “Diffusion tensor imaging: concepts and applications.,” *Journal of magnetic resonance imaging: JMRI*, vol. 13, no. 4, pp. 534-46, Apr. 2001.

[3] D. Le Bihan, C. Poupon, A. Amadon, and F. Lethimonnier, “Artifacts and pitfalls in diffusion MRI.,” *Journal of magnetic resonance imaging: JMRI*, vol. 24, no. 3, pp. 478-88, Sep. 2006.

[4] P.S. Jouk, A. Mourad, V. Milisic, G.Michalowicz, A. Raoult, D.Caillerie and Y. Usson, “Analysis of the fiber architecture of the heart by quantitative polarized light microscopy. Accuracy, limitations and contribution to the study of the fiber architecture of the ventricles during fetal and neonatal life.,” *European journal of cardio-thoracic surgery: official journal of the European Association for Cardio-thoracic Surgery*, vol. 31, no. 5, pp. 915-21, May 2007.

[5] E. Fieremans, Y.De Deene, S.Delputte, M.S.Ozdemir, Y.D’Asseler, J.Vlassenbroeck, K.Deblaere, E. Achten and I. Lemahieu, “Simulation and experimental verification of the diffusion in an anisotropic fiber phantom.,” *Journal of magnetic resonance (San Diego, Calif.: 1997)*, vol. 190, no. 2, pp. 189-99, Feb. 2008.

[6] L.H. Wang, Y.M. Zhu, H.Y. Li, W.Y. Liu, and I. E. Magnin, “Multiscale modeling and simulation of the cardiac fiber architecture for DMRI.,” IEEE transactions on bio-medical engineering, vol. 59, no. 1, pp. 16-9, Jan. 2012.

[7] D. Le Bihan, “The ‘wet mind’: water and functional neuroimaging.,” *Physics in medicine and biology*, vol. 52, no. 7, pp. R57-90, Apr. 2007.

[8] R. Ferreira da Silva, S. Camarasu-Pop, B. Grenier, V. Hamar, D. Manset, J. Montagnat, J. Revillard, J. R. Balderrama, A. Tsaregorodtsev, T. Glatard, “Multi-infrastructure workflow execution for medical simulation in the virtual imaging platform”, in: *HealthGrid 2011*, Bristol, UK, 2011.

[9] T. Glatard, J. Montagnat, D. Lingrand, X. Pennec, “Flexible and effcient workflow deployement of data-intensive applications on grids with MOTEUR”, *International Journal of High Performance Computing Applications* (IJHPCA) 22 (3) (2008) 347-360.

[10] A. Tsaregorodtsev, M. Bargiotti, N. Brook, A. C. Ramo, G. Castellani, P. Charpentier, C. Cioffi, J. Closier, R. G. Diaz, G. Kuznetsov, Y. Y. Li, R. Nandakumar, S. Paterson, R. Santinelli, A. C. Smith, M. S. Miguelez, S. G. Jimenez, “Dirac: a community grid solution”, *Journal of Physics*: Conference Series (2008) 119 (6).

[11] J. S. Rosenthal, “Parallel computing and Monte-Carlo algorithms”, *Far East Journal of Theoretical Statistics* 4 (1999) 207-236.

[12] S. Camarasu-Pop, T. Glatard, J. T. Moscicki, H. Benoit-Cattin, D. Sarrut, “Dynamic partitioning of Gate Monte-Carlo simulations on egee”, *Journal of Grid Computing* 8 (2) (2010) 241-259.

[13] S. Camarasu-Pop, T. Glatard, R. Ferreira da Silva, P. Gueth, D. Sarrut, and H. Benoit-Cattin, "Monte-Carlo Simulation on Heterogeneous Distributed Systems: a Computing Framework with Parallel Merging and Checkpointing Strategies", Future Generation Computer Systems, accepted for publication on 3/9/2012.