

Magnetic Resonance Imaging (MRI) Simulation on a Grid Computing Architecture

Hugues Benoit-Cattin, Fabrice Bellet, Johan Montagnat, Christophe Odet

▶ **To cite this version:**

Hugues Benoit-Cattin, Fabrice Bellet, Johan Montagnat, Christophe Odet. Magnetic Resonance Imaging (MRI) Simulation on a Grid Computing Architecture. 2010. hal-00476894

HAL Id: hal-00476894
https://hal.science/hal-00476894

Preprint submitted on 27 Apr 2010

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.

MAGNETIC RESONANCE IMAGING (MRI) SIMULATION ON A GRID COMPUTING ARCHITECTURE

H. BENOIT-CATTIN, F. BELLET, J. MONTAGNAT, C. ODET

CREATIS, UMR CNRS #5515
INSA, Bât. B. Pascal, 69621 Villeurbanne, FRANCE

Abstract

In this paper, we present the implementation of a Magnetic Resonance Imaging (MRI) simulator on a GRID computing architecture. The simulation process is based on the resolution of Bloch equation [1] in a 3D space. The computation kernel of the simulator is distributed to the grid nodes using MPICH-G2 [2]. The results presented show that simulation of 3D MRI data is achieved with a reasonable cost which gives new perspectives to MRI simulations usage.

1. INTRODUCTION

The simulation of Magnetic Resonance Imaging (MRI) is an important counterpart to MRI acquisitions. Simulation is naturally suited to acquire theoretical understanding of the complex MR technology. It is used as an educational tool in medical and technical environments [3, 4]. By offering an analysis independent of the multiple parameters involved in the MR technology, MRI simulation permits the investigation of artifact causes and effects [5, 6]. Likewise simulation may help in the development and optimization of MR sequences [7].

With the increased interest in computer-aided MRI image analysis methods (segmentation, data fusion, quantization ...), there is a greater need for objective methods of algorithm evaluation. Validation of in vivo MRI studies is complicated by a lack of reference data (gold standard) and the difficulty of constructing anatomical realistic physical phantoms. In this context, an MRI simulator provides an interesting assessment tool [8] since it generates 3D realistic images from medical virtual objects perfectly known.

In 1984, Bittoun et al. built one of the first 1D MRI simulator by numerically solving the Bloch equations [1] at each point in an object [9]. The same technique was later applied to 2D and 3D objects by Summers et al. [10] and Olsson et al. [6]. Brenner et al. have proposed a distributed implementation of these computationally intensive simulations [11]. In the domain of brain imaging, Kwan et al. [8, 12] have limited the computation time by using templates of the main brain tissues. Nevertheless, MRI simulation is still under investigation in order to simulate high resolution MR images including MRI artifacts (Magnetic susceptibility, chemical shift, diffusion, …) that may impact the quantitative measurements required for several MRI applications.

In this context, we develop, in collaboration with the CNRS LRMN-MIB[1] lab and with CEMAGREF/TEA[2] research unit, a 3D MRI simulator named SIMRI that is designed to simulate realistic high resolution 3D MR images. Since simulation of the MR physics is computationally very expensive [11], parallel implementation is mandatory to achieve performances compatible with the target applications. By offering a virtually unlimited computing power, grid technologies appear to be promising.

In this paper, we focus on the grid implementation of this simulator. Parts II gives the main components of the simulation process. Part III details the "gridification" of the computation kernel and presents the main results.

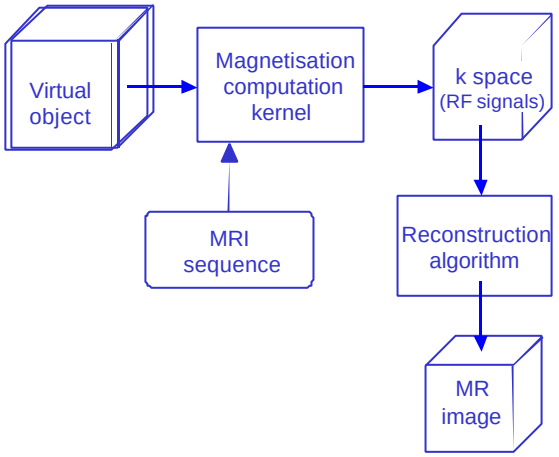


Figure 1 : SIMRI Simulator overview.

[1] LRMN-MIB UMR CNRS 5012, Lyon, France. http://jade.univ-lyon1.fr/

[2] CEMAGREF TEA Research Unit, Rennes, France. http://www.rennes.cemagref.fr/tere/tere.htm

2. MRI SIMULATION

2.1 Simulator overview

MR imaging [15] involves the nuclear spin system of an object to be imaged, the magnetic fields generated by the imager, the resonance and relaxation phenomena arising from the interaction nuclei / magnetic fields, and the signal detection and reconstruction. All these components are taken into account by the SIMRI simulator (Figure 1). From a virtual object and a MRI sequence, the magnetisation computation kernel computes the RF signals named k-space. The final MR image is reconstructed from the k-space with generally a 3D fast Fourier transform [15].

2.2 Virtual object description

The virtual 3D object is a discrete description of a real object spin system [15]. Each voxel of the virtual object contains a set of physical values necessary to computes with the Bloch equations the local spin magnetization. These values are the proton density noted r and the two relaxation constants T_1 and T_2. Basically, a virtual object describes the nuclear spin system of one isochromat, the water hydrogen atom. Note that SIMRI is able to deal with many isochromats in order to simulate chemical shift artifact [15] (Figure 2) or partial volume effect. In this case, each voxel contains the r, T_1, T_2 values of each isochromat.

A local main field homogeneity is also attached to each virtual object voxel. This value is used to simulate inhomogeneous main field and/or susceptibility artifact.

A virtual 3D object of size $S= X.Y.Z$ voxels is then described by $(3p+1).S$ values where p is the number of isochromats modeling to each object point.

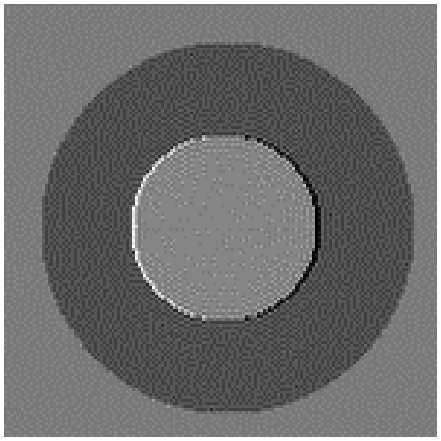


Figure 2 : 256² simulation of the chemical shift artifact using a two isochromats (water and fat) virtual object. The white and black part at the left and the right of the central circle correspond to the artifact.

2.3 MRI Sequence

During an MRI experience, an object is placed in a static magnetic field named B_0 and is excited by electro-magnetic events of two types: Radio Frequency (RF) pulse referred as B_1 field and magnetic fields gradients. Excitation steps are followed by an RF acquisition step. It corresponds to the MRI system antennas acquisition of the magnetization state of the object which is stored as a complex signal in the k-space [15]. The time chaining (Figure 3) of the magnetic events with RF signal acquisitions is named MRI sequence. It defines the way of filling the k-space and determines the final image characteristics.

It must be noted that a MRI sequence in the simulator has the same components and the same time schedule than in a real system.

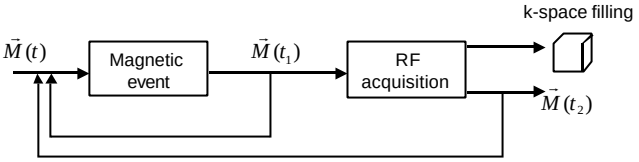


Figure 3 : MRI sequence time chaining reproduced by the simulator computation kernel.

2.4 Magnetization kernel

The magnetization computation kernel called during the simulation of an MRI sequence is based on the solving of the 3D Bloch equation [1]. It is a linear differential equation describing the time evolution of the spin magnetization and it is given by :

$$\frac{d\vec{M}}{dt} = g.(\vec{M} \times \vec{B}) - \begin{pmatrix} M_x / T_2 \\ M_y / T_2 \\ (M_z - M_0)/T_1 \end{pmatrix} \quad (1)$$

where $\vec{M} = (M_x, M_y, M_z)^T$ is the spin magnetization vector, M_0 is the spin magnetization equilibrium value which depends of the proton density r, (T_1, T_2) are the relaxation constants and g is the gyromagnetic constant of the considered isochromat (42.58 MHz/T for the Hydrogen proton). The magnetic field B is modeled as follows :

$$\vec{B} = B_0\vec{z} + \Delta\vec{B} + (\vec{G}(t).\vec{r}).\vec{z} + \vec{B}_1(t) \quad (2)$$

where B_0 is the main static magnetic field, $\Delta\vec{B}$ are the field inhomogeneities, $\vec{G}(t)$ is the applied field gradient, $\vec{B}_1(t)$ is the RF pulse and $\vec{r} = (x, y, z)^T$ is the spatial position.

The simulation kernel implements a discrete time solution [9] of the Bloch equation with the means of rotation matrices and exponential scaling which depend of the magnetic events of the MRI sequence. The magnetization vector evolution is computed from its previous time value according to the following equation:

$$\vec{M}(\vec{r}, t + \Delta t) = Rot(q_g).Rot(q_i).R_{relax}.R_{RF}.\vec{M}(\vec{r}, t) \quad (3)$$

where $Rot(a)$ is a rotation matrix defined by:

$$Rot(a) = \begin{pmatrix} \cos a & \sin a & 0 \\ -\sin a & \cos a & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4)$$

where φ_g is linked to the applied gradient $\vec{G}(t)$ by:

$$q_g = g.\vec{r}.\int_t^{t+\Delta t} \vec{G}(t)dt \quad (5)$$

where φ_i is linked to the field inhomogeneities by:

$$q_i = g.\Delta B(\vec{r}).\Delta t \quad (6)$$

where R_{relax} describes the relaxation effect by:

$$R_{relax} = \begin{pmatrix} e^{-\frac{\Delta t}{T_2(\vec{r})}} & 0 & 0 \\ 0 & e^{-\frac{\Delta t}{T_2(\vec{r})}} & 0 \\ 0 & 0 & e^{-\frac{\Delta t}{T_1(\vec{r})}} \end{pmatrix} \quad (7)$$

and where R_{RF} represents the rotating effect of an RF pulse of phase angle f leading to a flip angle a defined by:

$$R_{RF} = \begin{pmatrix} \cos^2 f + \sin^2 f \cos a & \sin f \cos f(1-\cos a) & -\sin f \sin a \\ \sin f \cos f(1-\cos a) & \sin^2 f + \cos^2 f \cos a & \cos f \sin a \\ \sin f \sin a & -\cos f \sin a & \cos a \end{pmatrix} \quad (8)$$

To design an MRI sequence, the magnetization computation kernel (based on equation 3) offers three functions which are application of RF pulse, application of gradient, and waiting which corresponds to the spin magnetization relaxation.

One can notice in equation 3 that the magnetization vector at a given localization depends only on the previous magnetization vector at the same position. This property will be used by the gridification of the simulation kernel.

The last functionality offered by the kernel is the RF signal acquisition (Figure 3) which corresponds to the acquisition by two antennas in the x,y plane of the magnetization state of the object after a given excitation. The RF signal is a one dimensional discrete complex signal that will fill in respect with the excitation sequence one line of the k-space volume.

One point k of the RF signal is obtained by summation of the local magnetization over all the virtual object (Eq. 9). The next point is obtained after an evolution of the local magnetization respecting equation 3 with a time step *Dt* equal to the sample period of the signal. This linearity property will be exploited by the grid architecture.

$$s[k] = \sum_n \vec{M}_n(t).\vec{x} + j\sum_n \vec{M}_n(t).\vec{y} \quad (9)$$

From equation 3 and equation 9, it appears that the simulation of one magnetization evolution is proportional to the virtual object dimension (*X.Y.Z*), while the simulation of the whole k-space acquisition is proportional to the object size and to the MRI image size (*N.M.P*). The whole simulation time of a 3D sequence (T_s) is then given by:

$$T_s = c.(X.Y.Z).[(M.N.P) + N_{ex}] \quad (10)$$

where *c* is a constant and N_{ex} is the number of magnetic events simulated before an acquisition.

In a MRI sequence, N_{ex} is negligible comparing to the MRI image size. So, as an example, multiplying by two all the dimensions of the virtual object and the MR image leads to a simulation time multiplication by 16 in two dimensions and by 64 in three dimensions.

If the simulation of small images (128x128 pixels) is done within a minute on a Pentium IV PC, equation 10 shows that high resolution MRI 3D simulation is not practicable on such a computer. Therefore, we turn toward Grid technologies that promise a virtually unlimited computing power.

3. GRID IMPLEMENTATION AND RESULTS

The whole code of the SIMRI simulator is written in ANSI C language. One module concerns virtual objects manipulation, one set of modules implements MRI sequences, one module deals with image reconstruction and the final set of modules is related to the magnetization computation kernel.

Profiling the execution of a SIMRI simulation indicates the percentage of time spent in each part of the simulation code. It appears clearly that most of the time (more than 90 %) is spent in the computation kernel specifically on the rotational and exponential scaling needed by the Bloch equation resolution. Consequently, we focused our parallelisation efforts on the kernel code in order to have a grid implementation of the simulator. This modification has to be almost transparent for MRI sequence developers .The gridification of the kernel must be done using the MPI [2] version of GLOBUS [16] since the middleware targeted for this application is the European DataGrid project [13] built on top of GLOBUS.

3.1 Gridification of the magnetization computation kernel

Because all the spin magnetization vectors are independents (Equation 3) and because the signal acquisition process is linear (Equation 9), many solutions for parallelisation can be considered.

One solution consists in distributing a minimal task which is to compute the new magnetization sate of a single spin vector. In this case, the granularity is maximized but the time of node communication will be predominant compared to the computation time.

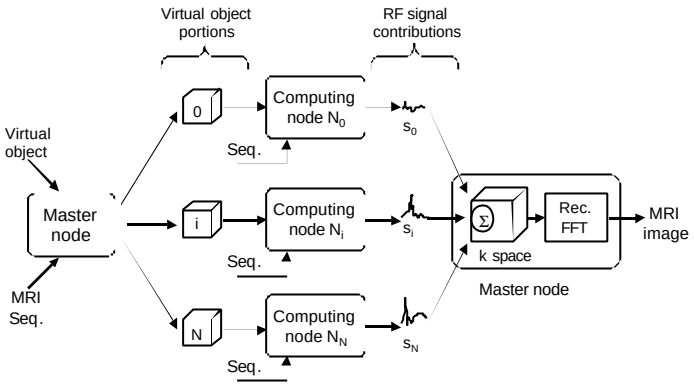


Figure 4 : Data and Process distribution to the grid nodes.

The second approach consists in distributing the magnetization computation of a subset of spin vectors. This subset can be fixed to a given size or adapted to the number of active nodes. We have selected this last solution and since each computation node has a set of magnetization vectors, this node will compute the contribution of these vectors to the RF signal generated during an acquisition step.

The data and process distribution to the grid nodes is given in Figure 4. All the computation nodes have the MRI sequence knowledge and they receive from the master node a part of the virtual object. They compute the magnetization evolution of the corresponding spin vector subset. At the end of each acquisition step, the master node collects and add all the RF signal contributions and stores the RF signal in the k-space. At the end of the MRI sequence, the master node applies the reconstruction algorithm to generate the MRI simulated image.

This solution is scalable and follows a simple parallelisation scheme of type "divide & conquer" with a regular problem of space division. Two types of activities are defined: computation of subset for N-1 nodes and results collect for one node. The time dedicated to the node communication is minimized since each node received only one task. Nevertheless, this solution drawback is a low granularity introducing loss of efficiency on heterogeneous grid. Indeed, all the computation nodes have the same amount of data to process and then the lowest node will give the timing.

3.2 Results

The results presented here have been obtained on a 8 nodes cluster. Each node is a Pentium III-1Ghz with 1Go of RAM. We run the MPICH-G2 [2] version of MPI associated to Globus 2.2 [16] for the grid access control.

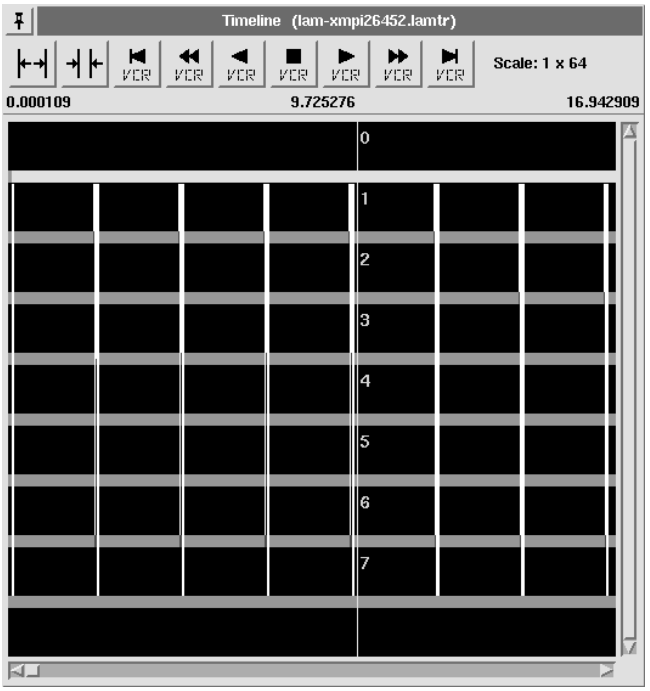


Figure 5 : Nodes activity reported by XMPI during a 256^2 2D spin echo simulation using MPI instead of MPICH. Horizontal bars correspond to the 8 nodes activity between 0 and 16.94 seconds. Vertical white bars correspond to MPI message exchange at the end of RF signal acquisition.

First of all, we have compared the efficiency of using our grid compared to a single machine. When the virtual object size is over 2^{10}, the average efficiency of each grid node is over 87 % meaning that the simulation time is divided by 7 at least. Since one node out of eight is just collecting results and reconstructing the image, this means that the seven computation nodes have an efficiency close to 100 %.

This very good efficiency is due to the homogeneity of our grid and to the kernel gridification chosen which minimizes the number and the size of the exchanged messages as shown in figure 5. For a 256^2 simulation, during an acquisition step, messages exchange and nodes synchronization takes 0.13 s compared to 2.5 s of computation time which corresponds to 5.2 % of the computation time. Table 1 shows that when the size of the virtual object increases this percentage decreases until a limit of 1.5 %. This limit is due to one node of our grid which is systematically late compared to the others.

Object size	32^2	64^2	128^2	256^2	512^2	1024^2
Ts/Tc (%)	33.1	11.4	5.2	1.5	1.5	1.5

Table 1 : percentage of synchronization and communication time (Ts) under computation time (Tc) function of the virtual object size.

The grid implementation of our simulator have been tested on 2D and 3D MRI Spin echo sequences applied on synthetic virtual objects of different sizes (Figure 6-7). Simulation time results are given in Tables 2 and 3. One can note that after a given size (128^2 and 64^3) the time factor induced by the dimension change follows the rule of equation 10. Before these size, the communication and synchronization time is too important compared to the process time (Table 1).

Object size	32^2	64^2	128^2	256^2	512^2	1024^2
Time	0.9 s	3.4 s	43.1 s	12 m	201 m	3277 m
Time factor	-	3.8	12.7	16.7	16.8	16.3

Table 2 : 2D MRI simulation computation time on a 2D spin echo sequence function of the object size. Virtual object and simulated image have the same size.

These simulation results obtained are very encouraging since it shows that with a small cluster, MRI simulation of high resolution (1024^2) 2D objects is possible within three days. Concerning 3D images, it is not realistic to simulate on such a small set of processors over 64^3 cubic MRI images. Nevertheless, it is possible to simulate within a week 3D multislice images (16 slices of 512x512 pixels). The simulation of high resolution 3D images should be tractable on full scale grids like the ones actually in development[13].

Object Size	16^3	32^3	64^3	128^3
Time	4.9 s	3.5 m	210 m	1626 m
Time factor	-	42.9	60	7.7

Table 3 : 3D MRI simulation computation time on a 3D spin echo sequence. Virtual object and simulated image have the same size except for the 128^3 object where the simulated image is 64^3 explaining a time factor of 7.7 instead of 64.

4. CONCLUSION

In this paper, we present a grid implementation of a 3D MRI simulator based on Block equations. The gridification is done in the magnetization computation kernel of the simulator and makes it transparent at the sequence level used by MRI sequence developers. The gridification strategy makes the simulator very efficient on an homogeneous grid. Results obtained on an 8 nodes cluster show that high resolution MRI simulation is possible at a reasonable cost. This is very encouraging for the different researches using MRI simulation like artifact analysis and correction or MRI post-processing [12].

Further works under development include analysis of MRI simulation performance on a larger grid and tests of different strategies for gridifying the computation kernel. Evolutions of the MRI simulation model is also under consideration in order to simulate more complex physical phenomena.

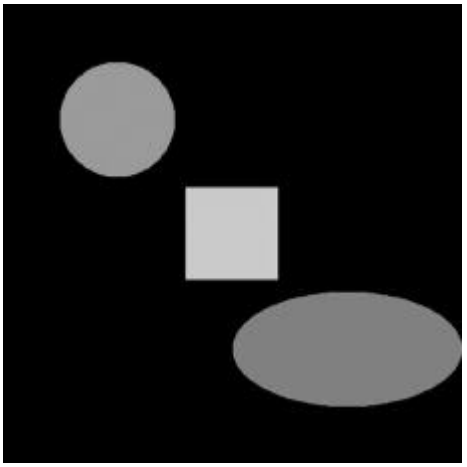


Figure 6 : Example of a simulated MRI image(512x512 pixels) using a r weighted spin echo sequence. The virtual object is composed by an elliptic zone (T1=350 ms, T2=75ms, r=60), a square zone (T1=2500 ms, T2=200 ms, r=100) and a circular zone (T1=500 ms, T2=90 ms r=70).

5. ACKNOWLEDGEMENT

This work is in the scope of the scientific topics of the PRC-GdR ISIS research group of the French National Center for Scientific Research CNRS. This work is partly supported by the IST European Data-Grid Project [13] and the French ministry for research ACI-GRID project [14]. This work has been also funded by the INSA Lyon.

We want to thank our partners in the SIMRI project, H. Saint-Jalmes and O. Beuf from the CNRS LRMN-MIB lab, G. Collewet from the CEMAGREF / TEA Research Unit and S. Balac from the CNRS MAPLY lab. The SIMRI project has

been initiated thanks to the work done by G. Soufflet and H. Saint-Jalmes on the initial 1D MRI simulator of J. Bittoun.

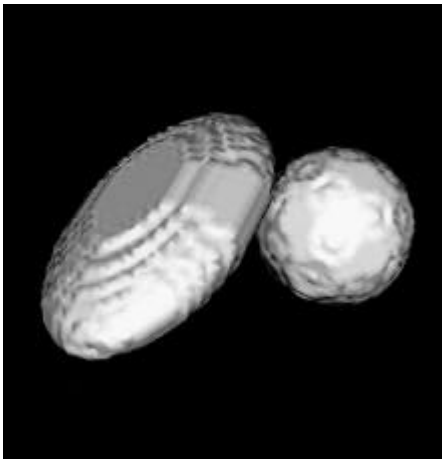


Figure 7 : 3D visualization of a simulated MRI image (64x64x64 voxels) using a T1 weighted 3D spin echo. The virtual object is composed by an elliptic zone (T1=500ms, T2=90 ms, r =70) and a spherical zone (T1=350 ms, T2=375 ms, r =60).

6. REFERENCES

[1] F. Bloch, "Nuclear induction", *Physical Review*, vol. 70, pp. 460-474, 1946.

[2] MPICH "http://www-unix.mcs.anl.gov/mpi/mpich/".

[3] G. Torheim, P. A. Rinck, R. A. Jones, and J. Kvaerness, "A simulator for teaching MR image contrast behavior", *MAGMA*, vol. 2, pp. 515-522, 1994.

[4] D. Rundle, S. Kishore, S. Seshadri, and F. Wehrli, "Magnetic resonance imaging simulator: A teaching tool for radiology", *Journal of Digital Imaging*, vol. 3, pp. 226-229, 1990.

[5] A. R. Brenner, S. Kannengiesser, J. Kürsch, and T. G. Noll, "Removal of artifacts in steady-state multi-pulse sequences", *MAGMA*, vol. 4, pp. 164-165, 1996.

[6] M. B. E. Olsson, R. Wirestam, and B. R. R. Persson, "A computer simulation program for MR imaging: Application to RF and static magnetic field imperfections", *Magnetic Resonance in Medicine*, vol. 34, pp. 612-617, 1995.

[7] A. R. Brenner, A. Glowinski, J. Kürsh, M. Drobnitzky, and R. W. Günther, "Optimization of ultrafast multiple sequences for dynamic MR sequences", *In Proc. of Society of Magnetic Resonance, SMR*, vol. 1, pp. 638, 1995.

[8] R. K.-S. Kwan, A. C. Evans, and G. B. Pike, "An extensible MRI simulator for post-processing evaluation", *In Proc. of International Conference on Visualization in Biomedical Computing, VBC'96*, pp. 135-140, 1996.

[9] J. Bittoun, J. Taquin, and M. Sauzade, "A computer algorithm for the simulation of any nuclear magnetic resonance (NMR) imaging method", *Magnetic Resonance Imaging*, vol. 3, pp. 363-376, 1984.

[10] R. M. Summers, L. Axel, and S. Israel, "A computer simulation of nuclear magnetic resonance imaging", *Magnetic Resonance in Medicine*, vol. 3, pp. 363-376, 1986.

[11] A. R. Brenner, J. Kürsch, and T. G. Noll, "Distributed large-scale simulation of magnetic resonance imaging", *MAGMA*, vol. 5, pp. 129-138, 1997.

[12] R. K. S. Kwan, A. C. Evans, and G. B. Pike, "MRI simulation based evaluation of image processing and classification methods", *IEEE Trans. on Medical Imaging*, vol. 18, n° 11, pp. 1085-1097, 1999.

[13] European DataGrid project "http://www.edg.org".

[14] MEDIGRID, "ACI-GRID project, French Research Ministry" "http://www.creatis.insa-lyon.fr/medigrid, http://www-sop.inria.fr/aci/grid/public/".

[15] Z. P. Liang and P. C. Lauterbur, *Principles of magnetic resonance imaging. A signal Processing perspective* New York: IEEE Press, 2000.

[16] Globus, "http://www.globus.org".