Monte Carlo validation of a mu-SPECT imaging system on the lightweight grid CiGri

Joe Aoun, B. Bzeznik, Vincent Breton, M. Leabad, J. Dimastromatteo, Y. Georgiou, O. Richard, P. Neyron, L. Desbat

Elsevier Editorial System(tm) for Future Generation Computer Systems
Manuscript Draft

Manuscript Number:

Title: Monte Carlo validation of a μ-SPECT imaging system on the lightweight grid CiGri

Article Type: SS: Medical Imaging on Grids - Montagnat

Keywords: Lightweight GRID, GATE, CIMENT GRID, Monte Carlo Simulation, Small Animal SPECT.

Corresponding Author: Mr Joe Aoun, Ph.D. Student

Corresponding Author's Institution: University of Grenoble

First Author: Joe Aoun, PhD student

Order of Authors: Joe Aoun, PhD student; Bruno Bzeznik, Research engineer; Vincent Breton, Researcher; Mehdi Leabad, Engineer; Julien Dimastromatteo, PhD student; Yiannis Georgiou, PhD student; Olivier Richard, University Professor and researcher; Pierre Neyron, Engineer; Laurent Desbat, University Professor and researcher

Abstract: Monte Carlo Simulations (MCS) are nowadays widely used in the field of nuclear medicine for system and algorithms designs. They are valuable for accurately reproducing experimental data, but at the expense of a long computing time. An efficient solution for shorter elapsed time has recently been proposed: grid computing. The aim of this work is to validate a small animal gamma camera MCS and to confirm the usefulness of grid computing for such a study. Good matches between measured and simulated data were achieved and a crunching factor up to 70 was attained on a lightweight campus grid.

Laboratoire TIMC – IMAG
(UMR CNRS 5525)
Institut d'Ingénierie de
l'Information de Santé

FACULTE DE MEDECINE DE GRENOBLE
UNIVERSITE JOSEPH FOURIER
38706 LA TRONCHE cedex
FRANCE

December, 7th, 2008

Peter Sloot, Editor-in-chief
Future Generation Computer Systems
University of Amsterdam
Kruislaan 403
1098 SJ Amsterdam, The Netherlands

Dear Mr Sloot,

Please find herewith a submission to Future Generation Computer Systems entitled, "Monte Carlo validation of a µ-SPECT imaging system on the lightweight grid CiGri". This manuscript is an extended and improved version of the "MICCAI-Grid Workshop" paper which is original. (*http://www.i3s.unice.fr/~johan/MICCAI-Grid/pdf/aounMICCAIG.pdf*).

The paper introduces a lightweight grid which exploits the idle resources of a set of computing resources on the Grenoble university campus (France). It also shows the interest of grids for heavy computations in medical imaging Monte Carlo Simulation.

Please feel free to address all correspondence concerning this paper to me or get in touch through e-mail (*Joe.Aoun@imag.fr*).

Sincerely,

Joe Aoun
PhD student
Université Joseph Fourier, Faculté de medicine
Domaine de la Merci
38706 La Tronche
Tel. +33 (0)4 56 52 00 12
Fax +33 (0)4 56 52 00 55
Sec. +33 (0)4 56 52 00 07
E-mail: Joe.Aoun@imag.fr

Title
Monte Carlo validation of a μ-SPECT imaging system on the lightweight grid CiGri

Authors
Joe Aoun [a,b], Bruno Bzeznik [c,d], Vincent Breton [b], Mehdi Leabad [e], Julien Dimastromatteo [f], Yiannis Georgiou [d], Olivier Richard [d], Pierre Neyron [g], Laurent Desbat [a]

Affiliation
[a] Laboratoire des Techniques de l'ingénierie Médicale et de la Complexité – IMAG, In3S/UMR CNRS 5525, Domaine de la Merci, 38710 La Tronche, France

[b] Laboratoire de Physique Corpusculaire, IN2P3/UMR CNRS 6533, 24 avenue des Landais, 63177 Aubière, France

[c] CIMENT, TOUR IRMA, 51 rue des Mathématiques, 38400 Saint Martin d'Hères, France

[d] Laboratoire d'Informatique de Grenoble, ENSIMAG - antenne de Montbonnot, ZIRST 51 avenue Jean Kuntzmann, 38330 Montbonnot Saint Martin, France

[e] Biospace LAB, 10 rue Mercoeur, 75011 Paris, France

[f] INSERM U877, Radiopharmaceutique biocliniques, Domaine de la Merci, 38700 La Tronche, France

[g] INRIA Grenoble - Rhône-Alpes - Service Experimentation et Developpement Logiciel, Inovallée, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier Cedex, France

Abstract
Monte Carlo Simulations (MCS) are nowadays widely used in the field of nuclear medicine for system and algorithms designs. They are valuable for accurately reproducing experimental data, but at the expense of a long computing time. An efficient solution for shorter elapsed time has recently been proposed: grid computing. The aim of this work is to validate a small animal gamma camera MCS and to confirm the usefulness of grid computing for such a study. Good matches between measured and simulated data were achieved and a crunching factor up to 70 was attained on a lightweight campus grid.

Keywords
Lightweight GRID, GATE, CIMENT GRID, Monte Carlo Simulation, Small Animal SPECT.

Main text

1. Introduction

Preclinical small animal researches have become a major focus in nuclear medicine [1]. New therapeutic and diagnostic studies are first investigated and validated on mice or rats before their application to patients [2]. In consequence, we can find many dedicated small field of view scanners for SPECT (Single Photon Emission Computed Tomography) [3-7] and PET (Positron Emission Tomography) [8,9] which have been designed in the last decade for these purposes. Before simulated data can be trusted, the simulation model must be validated by comparing it with the corresponding experimental results. Scanner models should incorporate all the physical effects (attenuation, scatter…) in order to improve the quantification and the quality of SPECT images [10]. When imaging with ^{125}I (27.5 keV) a concentration of radioactivity placed in the centre of a rat-sized cylinder of water, the scatter-to-primary ratio can reach up to 30% and the photon attenuation can reduce the activity quantitation by almost 50% (25% with ^{99m}Tc - 140 keV) [11]. Such degradations are very difficult to compensate for by an experimental or analytical approach [12]. Few researchers have tried to assess an analytical and experimental estimation of photon distributions in SPECT projections [13,14]. Monte Carlo Simulations (MCS) have greatly contributed to solve these problems thanks to their accuracy and the possibility of modelling all the physical processes in an independent way [15,16]. Since the 1960s, Monte Carlo methods have become an essential tool in the field of nuclear medicine [17]. Unfortunately, the use of MCS has limitations in computing time. Different strategies have been suggested to reduce the computing time such as the acceleration techniques [18]. Another solution to speed up MCS is to combine Monte Carlo and non-Monte Carlo modelling [18]. A third option that has recently been explored is the deployment of computing grids, also known as the parallelization of the MCS [19]. This consists in sub-dividing a long simulation into short ones. Each MCS needs a Random Number Stream (RNS) to produce the physical interactions in question. The distribution of MCS on multiple computing resources requires the generated streams to be independent [20].

A popular way of exploiting computing power for scientific research on a single site is the utilisation of networks of computers as a single unified computing resource, called cluster computing. On those systems, the resource allocation to users and computation tasks is handled through batch queuing software such as PBSPro [21], Torque, LSF, SLURM [22], OAR [23] and SGE [24]. Larger scale scientific research, linking geographically distributed computational resources, is achieved by technologies called computational grids or grid computing. The project with the greatest visibility on grid computing is Globus [25]. This mainstream approach provides a software infrastructure that enables applications to handle distributed heterogeneous computing resources as a single virtual machine. However, the installation, configuration, customization and maintenance of a system like Globus, is a rather complicated task and requires a highly skilled support team, which only a few laboratories are willing to afford. The opportunity of performing large computations at low-cost has motivated scientists to come up with another technological solution. This approach works through the exploitation of idle cycles of desktop PCs, known as voluntary computing or desktop grid. Other alternative technologies, based on the idea of harvesting unused resources of multiple distinct administrative domains that want to share their resources, are provided by the projects OurGrid [26] and Condor [27]. Under these circumstances, this paper introduces two grid tools, CiGri (CIMENT Grid) [28,29] and ComputeMode [30], used on the computing resources of the University of Grenoble, to report the validation of the Biospace small field of view gamma camera using the GATE Monte Carlo Simulations toolkit.

The paper is organized as follows:
Chapter 2 presents the materials and methods used in this study: we introduce CiGri, ComputeMode, the GATE MCS toolkit, the gamma camera we studied and the methods we used to deploy the simulations on the grid and to validate the simulated results compared to experimental measurements. Chapter 3 offers an analysis of the simulation results as well as an overall view of the grid performances. Last and not least, chapter 4 presents some conclusions and perspectives.

2. Materials and Methods

2.1. CiGri grid

2.1.1. A lightweight grid

CiGri (CIMENT Grid, *https://ciment.imag.fr/CiGri*) is a simple, lightweight, scalable and fault tolerant grid system which exploits the unused resources of a set of computing clusters. CiGri middleware is a free open source software under GPL licence developed at the University of Grenoble (France) and used upon the production clusters of the CIMENT project [28,29] since 2002. Working discreetly on the interconnected clusters (without influencing their normal functionality), it makes use of a transparent technique to harness the idle cluster resources for executing large scale scientific computations. The platform supports only a specific type of applications, called bag-of-tasks, which represent a set of independent sequential tasks. Hence, by tracking this specific kind of applications in a computing centre like CIMENT (counting half a dozen of clusters) and by running those applications at this transparent grid level, we "smooth" the overall load of the clusters (cf. Figure 1). Bag-of-tasks applications are removed from the local peak of an overloaded cluster and spread on other clusters that have idle resources.

The concept of a lightweight grid concerns infrastructures that simplify the general problem of the grids. This simplification generally goes through a certain homogeneity of services and administration procedures by adopting the same services and configurations on all interconnected clusters. Our main interest on CIGRI system is to provide a platform that focuses on the research and development of specific problems that come along with the execution of tasks (scheduling, fault tolerance...) on distributed grid environments. Our approach does not deal with critical problems inherent to computational grids like security, authentication mechanisms and resource location.

Fig.1

2.1.2. Best-effort computing with OAR

One of the simplifications assumed by the lightweight grid concept is that CiGri relies on the OAR batch scheduler [23]. A resources management software (or batch scheduler) handles the resource allocation to users and tasks on a computing cluster. OAR and CiGri are developed by the same research team. It provides an interesting feature used to not influence the normal functionality of the interconnected clusters on a CiGri platform. This is the concept of "best-effort" tasks: these are jobs which have a zero execution priority and are submitted only if there is a free resource. Moreover, if during the execution of those jobs the

resource is requested by a local cluster user, the “best-effort” job is killed by the local batch scheduler and CiGri grid server is notified (cf. Figure 2).

Fig.2

This CiGri/OAR collaboration does not mean that OAR is mandatory for the local users’ tasks submissions. It means that OAR must be installed on all the clusters of a CiGri platform to provide the “best-effort” interface for CIGRI remote jobs; but in the same time, it may be coupled to another batch scheduler for the local cluster jobs. This coupling option has already been tested with PBSPro, Torque and SGE.

2.1.3. CiGri Fault Treatment

Considering this best-effort feature and the aim of scalability (large bag-of-tasks applications means > 100k tasks), it is straightforward that CiGri has to provide support for error management. In CiGri fault treatment mechanism, the initial concern is to be able to locate, log and categorize the different errors that are possible to occur (ex. Cluster momentarily not responding). To ensure the platforms’ reliability, it provides automatic actions to be taken for every different category.

Some types of events can be automatically and transparently treated. This happens for example, in the case of the interference failures where the grid best-effort task is killed by the local cluster batch scheduler, so that the resource can be occupied, by a higher priority local cluster job.

On such a grid infrastructure, the walltime of the tasks is a very important parameter as the life expectancy depends on the load and the turnaround time of the cluster jobs. Thus, the tasks have to be either relatively short or we have to use specific Checkpointing mechanisms. Such mechanisms are already implemented and are currently under evaluation [28]. Concerning the application discussed in this paper, we were able to generate rather small jobs (less than one hour) to have a good probability of ending without “being killed” (see the resubmission percentage in 3.5).

Furthermore, jobs length must not be too short because of the overhead time generated by the submission. Currently, CiGri submits jobs sequentially. In case we have hundreds of free resources, it may take several minutes to fill them all with best-effort jobs. Thus, with very short jobs (of the order of minutes), we may spend more time submitting than computing. This is a known issue, and OAR is currently evolving to offer an “array-job” functionality that will help a lot minimizing the submission overhead of CiGri.

2.1.4. CiGri architecture

The CIGRI software is composed of a server which communicates with all the batch schedulers of the clusters. It acts like a normal user, submitting minimum priority jobs on the best-effort queue of the batch schedulers. There is no specific CIGRI software installed on the clusters, except OAR. Nevertheless, it just takes advantage of the classic system tools (ssh, bash, sudo, cat, ls, scp, tar...) and the NFS file system of the cluster. An optional LDAP client may be used for the authentication, but it is not mandatory. Users may have different login accounts on the different clusters. Then CiGri makes the binding between the grid login and the clusters’ logins.

Fig.3

The system is designed with a modular architecture based upon two high level components: an SQL database and a scripting language (in fact, two scripting languages: Perl and Ruby). The database is used to holding all the internal data and works as the only communication medium between the modules (cf. Figure 3). It is used to store information such as: the state of clusters' nodes, the state of submitted jobs, the various errors that can be obtained and logged, logging of all activities, information about the users, etc. The choice of a database as the central component ensures the management of big amounts of information and guarantees the reliability of the platform.

The central part of the CIGRI software is made of a collection of independent modules. Each of them is in charge of a specific task such as jobs scheduling, jobs monitoring, error logging, jobs submitting, statistics updating, and results collecting. The whole system is managed by a central module which is in charge of calling the other modules to perform either regular tasks (such as monitoring) or on-demand tasks (such as submission).

Another important functionality is the collection of results from the clusters, along with the cleaning of temporary files. A specially designed module auxiliary to the system is responsible for periodically collecting the result files, archiving them on the grid server and at the same time erasing them from the cluster. Hence, the user can collect the results of its computation on one centralised place and at the same time, the cluster storage resources are not overbooked by grid application data. Moreover, a private web portal exists to monitor the jobs on the grid.

2.1.5. Jobs submission

A grid user logs on a central host, through ssh and from there, jobs are prepared, submitted and collected. The job is described into a simple JDL (Job Description Language) file. Parameters are simply listed into a second file. One list of parameters per line corresponds to a single job. Each first parameter on a line is by convention the name of the output file or directory that will be collected once the job is ended. The most difficult part of the work, for a user, is to have the application installed and ready for running on every cluster of the grid he plans to use. This is generally achieved with the help of the local administrators (here again, we stick to the concept of a "lightweight" grid, where the administrators are easily reachable). Once the application is installed everywhere, the user generally needs to "feed" it with the input data. But, all the data have to be available for every job upon all clusters, even if only a subset will be used as the jobs are spread all over the grid. This is because we don't know in advance which task is going to be executed on which cluster and because there's no easy way to transfer only the needed files at the beginning of the job without creating a bottleneck. Studies are currently made for facilitating and optimizing the input data supply. One of the most probable solutions is to deploy a distributed file system that can easily work over tunnels, like GlusterFS [31] for example. A distributed file system could also be an answer to the problem of Checkpoint files availability.

Nevertheless, during this study, the input data were small enough (a few megabytes) to be entirely copied upon every cluster before each experiment. That was achieved by using a simple loop from the central CiGri host.

2.2. Exploiting desktop PCs for nightly computing: ComputeMode

ComputeMode is another useful tool that was used through CiGri to provide additional computing resources during this study. ComputeMode is a free software that creates a computing cluster by rebooting idle workstations in an intranet to a diskless operating system at periodic scheduled times (mostly during the night). Initially developed in 2004, it is now co-maintained by the developers of CiGri since it is a valuable tool used in conjunction with this grid. The ComputeMode service is provided by a single server connected to the same LAN as the PCs that are targeted to be used for computation when not used for another matter. ComputeMode core components include a special PXE boot manager, an OAR server/front-end, a NFS server providing diskless operating systems and home directories, and a powerful web interface for the management of the resources and schedules. Using that interface, administrators can create schedules (calendar) for automatically booting PCs on specific systems at specific times. In fact this schedule capability is not only useful for managing volatile computing systems but can also serve to handle the start-up process of machine for a third party cloning system, for nightly automatic re-installation for instance. That is how the tool is actually used in one of the department of Grenoble University, hence proving being helpful both to the local administrators of the department for the management of the pool of workstations, and to the CIMENT project by providing extra computing resources during nights and weekends.

Like CiGri and OAR, ComputeMode is built on top of a SQL database that holds the information with regard to the workstations it manages: MAC addresses, available operating systems and schedules. The administration interface is written in PHP and interacts with the system components through the database. A complex configuration of DHCP server and a modified TFTP server allow workstations to be automatically registered in the database and to smartly load the judicious operating system at boot time, based on the schedules and possible actions from a user that might be in front of the machine unexpectedly. A very light HTTP agent is also provided for both Windows and Linux, to run as a service of the normal operating system of the workstation while not in the computation mode, polling the server to know when it is time to reboot in order to switch operating system. On the other hand, while in the computation mode, the machine poll the server as well, in order to switch back at the end of a computation scheduled, but will also automatically switch to the normal operating system at reboot if a user shows up and need the machine, without requesting more interaction than power cycling the machine using the keyboard or the power switch.

From the grid resource management point of view, new workstations are automatically recorded into the batch scheduler and a simple click only is necessary in the administration interface to configure a group of workstations to be in the computation mode at some schedule. It is important to notice that when a workstation is in the computation mode, the local disk is not activated (diskless operating system, remote file system), which ensures the owner or local user of the workstation that the local operating system and data won't be modified and can't be compromised. The computing operating system is a diskless GNU Linux system which is hence easily configurable from the ComputeMode server in a centralized manner, allowing the grid administrators to install applications and libraries very efficiently, since no deployment phase is needed.

From the job management point of view, resources of a ComputeMode cluster are volatile: they may show up or disappear at some time of the day and nobody can really know in advance if a job will be completed in time before the resource actually disappear. Moreover, in such a cluster, it is quite common to use PCs from classrooms or offices which places are not as reliable as machine rooms, hence where random failures can happen like bad network connectivity, power failures or unexpected reboots. This makes the use of CiGri even more

helpful since it is indeed designed to manage such "best-effort" systems. Furthermore, one can notice that often, only one or two of the ComputeMode resources are available during the day and can be used for the preparation or the testing of a job. All other resources appear in CiGri as unavailable, just like any resource of the grid which is used by a local user or in a failure state. Then if a grid campaign is running, as soon as a PC handled by ComputeMode switch to the computation mode, for instance at the end of the day, CiGri sends jobs to it just like to any other idle resource of any other classical cluster. CiGri will as well get back into the bag-of-tasks any job that was not finished when related ComputeMode resources become unavailable, for instance in the morning when workstations must operate their normal system for serving users. Those uncompleted jobs are then resubmitted to other clusters of the grid that are available during the day. For CiGri, ComputeMode provides indeed a temporary but auto-managed pool of numerous resources for the grid, by just optimizing the utilization of existing resources of the University.

2.3. Monte Carlo simulation toolkit

GATE (Geant4 Application for Tomographic Emission) [32] is a Monte Carlo simulation toolkit based on the general-purpose simulation package Geant4 [33]. It was developed in order to model treatment and diagnosis examinations such as radiotherapy, SPECT and PET in the field of nuclear medicine (and recently CT-scan exams). GATE is an open source code which uses approximately 200 C++ classes from Geant4. User does not have to program in C++ thanks to an extended version of Geant4 scripting version. Thus, the user can easily build a simulation (macro) by using a script language. Many researchers have been using GATE for its flexibility and its accurate modelling of different detector designs [34] and very complex geometries [35]. The major drawback of GATE is the computation time especially when simulating realistic configurations such as voxelized emission and attenuation maps. Several solutions were proposed to accelerate some of the GATE simulations: (*i*) setting thresholds for the production of secondary electrons, x-rays and delta-rays, (*ii*) limiting the emission angle to a certain range, (*iii*) replacing the disintegration scheme by a source of monoenergetic gammas, (*iv*) parametrizing replicas for the collimator hole arrays (SPECT), (*v*) compressing voxelized phantom [36], (*vi*) techniques such as variance reduction, bootstrapping and jackknifing, (*vii*) splitting the simulation on a cluster [37], and recently on the grid [38].

Nowadays, GATE is largely used on grids. Indeed, Monte Carlo Simulations can be packed in independent bags of tasks. They are thus easily and efficiently computed on distributed computers. To distribute a long sub-divided simulation on multiple processors, one should also sub-divide the associated long Random Number Stream (RNS) into short ones. However, these short RNS have to be independent because any intra or inter-sequence correlation could lead to inaccurate results [39]. The parallelization of the RNS was accomplished by using the sequence splitting method [38] (also known as the random spacing method). The pseudo random numbers generator (PRNG) James Random (period 2^{144}), implemented by default in GATE, was replaced by the Mersenne Twister PRNG due to his huge period (2^{19937}) [39].

Tab.1

The work described in this paper was done by using GATE 3.1.1 built upon Geant4.8.1. Table 1 shows the number of clusters/CPUs on which GATE was installed. The GATE macro

contains a unique random number status and an output filename. These two parameters are renamed according to their appropriate value during the distribution process. Output files were analysed with the ROOT object oriented data analysis framework (*http://root.cern.ch/*).

2.4. The γ Imager: system configuration

The Biospace γ Imager (*http://www.biospacelab.com/index.html*) is a high resolution planar scintigraphic camera which associates a 4 mm thick NaI(Tl) crystal and a PSPMT Hamamatsu R3292 which leads to a circular 100 mm diameter field of view. While most high resolution cameras use a pixelated crystal, the scintillation crystal of the γ Imager has a continuous 120 mm diameter with a 100 mm circular diameter active area. An aluminium protection layer, 0.8 mm thick, is placed in front of the crystal. The 5 Inch diameter PSPMT is equipped with a bialkali photocathode, 11 multiplication dynodes, 1 reflective dynode and a multiwire anode with 28 (X) + 28 (Y) wires. The readout of the 56 anode signals enables to calculate the spatial position (X, Y) and the energy for each event. A removable low-energy high-resolution parallel hole collimator with 35 mm thickness is used. The flat-to-flat distance of the hexagonal holes is 1.3 mm and the septum thickness in all directions is 0.2 mm. The whole detection head is surrounded with a 15 mm thick lead shielding. The γ Imager consists of two detector heads, only one head was used in this study.

2.5. Simulation of the Biospace gamma camera

Fig.4

The geometry of the γ Imager was accurately described in GATE. Figure 4 (a) shows the model of the detector head. The dimensions and the material of each part of the real camera were modelled as realistically as possible. The detector head simulation was thus composed of the lead collimator, a 1.2 mm air gap space, the aluminium protection, the NaI(Tl) crystal and the lead shielding of the whole head. As the plate scintillation crystal is 4 mm thick, about 38% of the 140 keV incident photons pass through the crystal without interacting. Thus, a significant number of photons will not be detected if we do not take into account the back-compartment. A 10 μm thick gold layer is placed behind the crystal, followed by a 0.39 mm Epoxy layer which separates it from the 1 mm light guide made of quartz. Another 0.15 thick Epoxy layer is positioned behind the light guide followed by the PSPMT, modelled as a 2 mm borosilicate glass entrance window and a 110 mm nickel back-part. The detector back-compartment was ended by a 15 mm rear lead shielding. Figure 4 (b) illustrates the relative scattering contribution of each layer in the model (source at 2 cm from the collimator), by counting the scattered events in each particular part divided by the total number of scatter events. During the acquisitions, a solution containing ^{99m}Tc, a photon emitter at 140 keV, was put in a glass capillary of 1.4 mm inner diameter and 1.8 mm outer diameter and 6 mm length. The capillary was held by a Pyrex Ruler (150 mm x 40 mm x 3 mm). The Ruler and the capillary were also simulated.

2.6. Validation of the small animal camera

The validation of the small animal SPECT camera is based on the comparison of four parameters measured experimentally with the corresponding simulation data; Energy Spectra, Spatial Resolution (FWHM), Sensitivity and Image of a capillary phantom. The three first

parameters represent three basic features of a gamma camera, while the image of an inhomogeneous phantom allows a visual comparison between experimental and simulated data.

2.6.1 Experimental set-up

In this study, 16 planar acquisitions were made altogether. The radioactive background was first measured during 30 minutes without any activity and was subtracted from all the other measurements after normalizing it to the same acquisition duration. Then, 14 acquisitions were performed using a liquid source of ^{99m}Tc (140 keV) which was put in a thin capillary (1.4 mm inner diameter and 6 mm long). Total activity was 70 µCi (2.59 MBq). The capillary was first placed in the air at 2, 7, 10 and 16.5 cm from the detector's surface. Then, the capillary at 16.5 cm was positioned above a cylindrical beaker (10.5 cm inner diameter and 15.6 cm long) which was put on the collimator. The beaker was used three times: empty, filled with 400 ml (4.62 cm height) and with 1000 ml (11.55 cm height) of water. All the seven configurations were performed in a first set of measurements with the source at the centre of the camera Field Of View (FOV), and a second set was performed with the source 2 cm off-centred. The sixteenth configuration was performed so as to evaluate the image of the capillary phantom. The phantom consisted of four parallel capillaries (1.4 mm inner diameter and 31.5 mm long), with a capillary-to-capillary distance of 5 mm. The capillaries were filled with ^{99m}Tc solutions of different activities (from the least (right) to the most (left) radioactive) (cf. Figures 8 (a) and (b)): 81 µCi, 129 µCi, 220 µCi and 611 µCi. The phantom was 5 cm far from the scintillation camera. Events were recorded in an energy window between 40 and 186 keV for a duration of 5 minutes for each acquisition. The size of the projections was 256 x 256 pixels (pixel size ~ 0.39 mm).

2.6.2. GATE simulations

The fifteen configurations mentioned above were accurately reproduced using GATE. Monoenergetic gamma rays (140 keV) were emitted in 4 π steradians. The physical processes involving photon interactions (photoelectric effect, Compton scattering and Rayleigh scattering) were modelled using the low-energy electromagnetic package of GEANT4, while gamma-conversion was disabled. The energy resolution Full Width Half Maximum (FWHMe) of the camera was modelled by the convolution of the output data using a Gaussian distribution with user defined mean and FWHM as stated by the manufacturer (11.5% FWHMe at 140 keV). The intrinsic spatial resolution FWHMi was also modelled in the same way (2.2 mm FWHMi at 140 keV).

2.6.3. Parallelization of the simulations

Many models of the camera have been tested in order to obtain the most realistic detector. A big simulation of one billion events was generated for each configuration. The simulation was split into 1000 small simulations, 1 million emitted events each, and was then distributed on CiGri. The 1000 small output files were collected from the grid, merged into one file (on a local CPU) and finally analysed using ROOT. Simulating 1 million events takes 10 minutes on a local CPU (Pentium IV, 3.2 GHz, 1 Go RAM) and requires ~ 30 million random numbers. Thus, a global sequence of more than 30 billion PRN was used for each simulation (which is small for the used PRNG Mersenne Twister).

2.6.4. Comparison parameters

Experimental and simulated data were compared basing them on the following quantities:

• *Energy Spectra*. The energy spectra of events recorded in the whole FOV (40-186 keV) was sketched for five configurations: (*i*) in air: source placed at 2 and 10 cm from the collimator, (*ii*) in water: source placed at 16.5 cm from the collimator above the empty beaker which then is filled with 4.62 and 11.55 cm of water.

• *Spatial Resolution*. The spatial resolution was estimated by measuring the FWHM of profiles perpendicular to the axis of the line source. The FWHM was calculated for the 14 configurations by drawing a 6 pixel thick profile (~ 2.345 mm) through the source in a plane perpendicular to the cylinder axis (X-axis) in the Y-axis direction. Events were detected in the photopeak window (126-154 keV) to reduce the noise effects.

• *Sensitivity*. The system sensitivity, defined as the number of detected events divided by the number of emitted events, was evaluated for the 7 configurations where the source is placed in the centre of the FOV. Data were acquired in 3 energy windows: (*i*) photopeak window 126-154 keV, (*ii*) Compton window 92-125 keV and (*iii*) total FOV window 40-186 keV.

• *Image of a capillary phantom*. The image of the capillary phantom was acquired in the whole FOV in the energy window 40-186 keV. A 6 pixels thick profile was also drawn through the 4 line sources seen on the image (cf. Figures 8 (a) and (b)).

3. Results and discussions

3.1. Energy Spectra

Figure 5 shows the experimental and simulated energy spectra of the ^{99m}Tc point source placed in the air at 2 and 10 cm from the collimator: contributions of the simulated photons scattered within different components of the detector head and within the capillary and the ruler are also plotted. The experimental and simulated energy spectra were normalized to the same number of counts detected at 140 keV.

As illustrated in figures 5 (a) and (b), including a back-compartment model was essential as well as the simulation of the capillary and the ruler to obtain a good agreement between measured and simulated spectra between 80 and 120 keV.

Fig.5

Good agreements between experimental and simulated energy spectra can be noticed especially in the photopeak and Compton windows. Differences observed in figures 5 (a) and (b) between 60 and 100 keV could come from the imperfect estimated shape and material of the ruler and from the unawareness of a layer which was not modelled behind the crystal. Considering the PSPMT as a continuous layer of Nickel could also lead to such differences [40].

Energy spectra obtained for the ^{99m}Tc source at 16.5 cm from the collimator, with 0, 4.62 (400 ml) and 11.55 cm (1000 ml) water thickness are shown in figure 6. The scattered photons contributions were also drawn.

The scattered photons within the phantom have increased, as expected, proportionally to the water thickness. Good agreements between experimental and simulated energy spectra could be observed for the two configurations with the empty beaker which then is filled with 4.62 cm thick of water.

The same differences with the source in air between 60 and 100 keV were maintained in figures 6 (a) and (b), which confirms that the back-compartment was not accurately modelled. But it is not the case in figure 6 (c) because backscattered photons (scattered within the ruler) have less influence on the energy spectra. It was reduced by the presence of an 11.55 cm thickness of water.

The disagreement in figure 6 (b) (4.62 cm of water) between 75 and 85 keV and between 110 and 120 keV is lower than in figure 6 (c) (11.55 cm of water) and doesn't figure at all when the beaker is empty (cf. Figure 6 (a)). Thus, these differences are proportional to the water thickness and could be the result of the impurity of the real water used in the measurements.

Fig.6

3.2. Spatial resolution

Experimental FWHM for a centred and off-centred source were well reproduced in all configurations by the simulations where discrepancies are below 2% and 5% respectively, except the source at 2 cm. The 5% differences might be due to the imperfect modelling of the PSPMT non-uniform response in GATE [6]. The approximate modelling of the ruler was also noticed as the disagreements of the FWHM for the source centred and off-centred at 2 cm is clearly higher than the other configurations.

Tab.2

Several SPECT and PET scanner models have been validated in the literature using MCS. Some of these works have performed a comparison between experimental and simulated spatial resolutions and have illustrated the following differences: 8.3% in μ-SPECT [5], 6% in μ-PET [8], 4.5% [34] and 2% [41] in SPECT and 11-18% in PET [42]. It should be known that the results below cannot be compared to those in our study because each validation was performed differently (source, collimator, crystal…), even in the case of the μ-SPECT system. Nevertheless, it could be useful to assert that a 2% or 5% difference is well acceptable.

3.3. Sensitivity

The Biospace small animal camera sensitivity values obtained in Air in three energy windows (total FOV 40-186 keV, photopeak window 126-154 keV and Compton window 92-125 keV) with GATE compared to the measured values are plotted in figure 7 (a) for the 4 source-collimator distances 2, 7, 10 and 16.5 cm. Relative differences between experimental

and calculated values were respectively: (*i*) 40-186 keV: 6.8%, 4.6%, 4.1% and 5.7%; (*ii*) 126-154 keV: 7.9%, 6%, 5.9% and 7.2%; (*iii*) 92-125 keV: 8.1%, 4%, 1.4% and 5.9%.

Results of the system sensitivity with the phantom (empty, 400 ml and 1000 ml) derived from the experiment measurements and the simulations are drawn in figure 7 (b) for the source placed at 16.5 cm from the collimator where events were detected in the same three energy windows. Relative differences between experimental and simulated values for the different water thickness (0, 4.62 and 11.55 cm) were: (*i*) 40-186 keV: 1.5%, 3.8% and 3.1%; (*ii*) 126-154 keV: 3.2%, 4.4% and 1.9%; (*iii*) 92-125 keV: 0.3%, 0.5% and 5.6%.

Fig.7

The simulation was able to reproduce the system sensitivity in the three energy windows for all the configurations within an 8% error. This difference could result from the simple modelling of the inhomogeneous materials and the complex shapes of the different detector components which only the constructor accurately knows. The 8% differences between experimental and simulated system sensitivities are relatively a good result if we look at other illustrated disagreements: 2% in μ-SPECT [6], 3% in μ-PET [8], 6.1% [34] and 9% [41] in SPECT and 10% in PET [42].

3.4. Image of a capillary phantom

Fig.8

Figure 8 shows a visual comparison between (a) experimental and (b) simulated images of the capillary phantom as well as the (c) horizontally drawn profiles (6 pixels thick ~ 2.345 mm) through these images. The comparison of the two images and profiles shows a good agreement. However, the simulation was unable to perfectly reproduce the local distortion which could account for the 5% differences between experimental and simulated FWHM for an off-centred source (cf. Table 2). [6] have modelled the non-uniform PSPMT response (CsI(Tl) array small gamma camera) using a corrective map obtained experimentally. This simple approach has partially corrected the local distortion from the GATE images. It has also reduced the errors between experimental and simulated FWHM and system sensitivity values.

3.5. Calculation time

Table 3 contains the computing time of a long simulation (1000 jobs) on a local CPU (Pentium IV, 3.2 GHz, 1 Go RAM) and on CiGri: during the day, during nights and weekends and an estimated average of the CPUs availability. CiGri has reduced the duration of a simulation compared to 1 CPU computation by a factor of 42, 67 and 56 respectively. Resubmitted jobs rate on CiGri, defined as the number of resubmitted jobs divided by the number of executed jobs, is also represented at the right of table 3. 16.9% of the jobs were killed and then were resubmitted by CiGri during the day, which is as expected higher than the 10.2% resubmission rate during the nights and weekends (more users during the day). The estimated average of the resubmitted jobs is 13.4%.

Each long simulation corresponds to one configuration. The material composition and the thickness of many components of the detector weren't exactly known (PSPMT, Gold

layer…). In consequence, about 200 simulations were carried out in order to optimise the model of the Biospace camera. All these simulations would have taken 4 years of computation time on a local CPU, but on CiGri it took only 25 days (cf. Table 3). In the ideal case, where no job is killed, the 200 simulations would have taken about 22 days on CiGri. Thus, the 200 simulations would not have been feasible on a local CPU. Reducing the number of simulations would have altered the quality of the results. So the deployment of the Grid was crucial in this study to get accurate results.

Tab.3

4. Conclusions and perspectives

Nuclear medical imaging needs a better physical model to improve the quantification and the quality of the reconstructed images. Our study has brought into light that the Monte Carlo Simulations toolkit GATE was able to accurately model the Biospace small field of view γ Imager. The camera model can thus be used for future researches such as developing new attenuation and diffusion correction algorithms. This paper has also proved the interest of grids in order to obtain accurate MCS results in SPECT within reasonable elapsed time. The CIMENT Grid, powered by CiGri, was an essential tool to develop and assess our MCS of the γ Imager on a grid.

In this paper, we have presented the lightweight grid CiGri. We have experimented CiGri on a real life medical imaging application. Compared to other grid systems like Condor, Ourgrid or Globus, the lightweight grid CiGri is very simple to install, to administrate and to use. CiGri is well suited for parametric applications involving a large bag of independent tasks. It was not designed to support parallel tasks with strong dependences. This simplification allowed us however to build a grid very easy to grasp. Beginners can earn their spurs on a local grid before trying to use an international grid. Users needing a reasonable computation time for bag-of-tasks applications are able to quickly obtain results. Moreover, the use of campus clusters can be optimized as their idle CPU cycles can be dedicated to parametric computations thanks to CiGri instead of being wasted. Additionally, CiGri offers a way to efficiently exploit volatile resources like desktop workstations managed by a tool like ComputeMode.

Planned future works concerning CiGri, include the integration of the new “array jobs” feature, supported upon OAR, which will minimize the submission time overhead of a large number of jobs; a distributed file system usage to facilitate and optimize the input data transfers; and the Checkpointing integration using BLCR (Berkeley Lab Checkpoint/Restart) [43] for better global efficiency of the grid and particularly a minimization of the resubmission percentage [29].

Furthermore, we are also working upon the use of virtualization technologies to facilitate the applications deployment. It will allow us to abstract the diversity of operating systems. Virtualization can also be used as another way to perform Checkpointing.

Concerning the medical imaging application, our future study will be the development of an iterative reconstruction algorithm in which long GATE simulations will be carried out at each iteration. For such studies, a higher scale of computation power is needed. The EGEE (Enabling Grid for E-sciencE) [44] Grid with a next multi-core processor generation should be able to meet our requests.

Acknowledgement

The authors are very grateful to Professor Daniel Fagret of the INSERM U877 Laboratory for providing us with the γ Imager. We would also like to thank Serge Maitrejean of Biospace LAB for his technical support.
CiGri has been supported by the Region Rhône-Alpes (grant CPER CIRA).

References

[1] A. Del Guerra, N. Belcari, State-of-the-art of PET SPECT and CT for small animal imaging, *Nucl. Instr. and Meth.* A **583** (2007) 119-124.
[2] A. Constantinesco, C. Goetz, V. Israel-Jost, P. Choquet, Quel avenir pour l'imagerie TEMP du petit animal ? = What does the future hold for small animal SPECT imaging?, *Médecine nucléaire* **31** (2007) 183-192.
[3] L. Smith, Z. He, D. Wehe, G. Knoll, S. Wilderman, Design and modelling of the Hybrid Portable Gamma Camera system, *IEEE Tr. Nucl. Sci.* **45** (1998) 963-969.
[4] J. Berthot, V. Breton, P. Brette, S. Crespin, N. Giokaris, D. Lazaro, J. Maublant, L. Méritet, Monte Carlo simulation of gamma cameras using GEANT, *Proceedings of the third IEEE Nucl. Sci. Symposium and Medical Imaging Conference*, Lyon France, (2000) 20/110-20/113.
[5] F. Garibaldi et al, Optimization of compact gamma cameras for breast imaging, *Nucl. Instr. and Meth. in Phys.* Res A **471** (2001) 222-228.
[6] D. Lazaro et al, Validation of the GATE Monte Carlo simulation platform for modelling a CsI(Tl) scintillation camera dedicated to small-animal imaging, *Phys. Med. Biol.* **49** (2004) 271-285.
[7] F. J. Beekman, B. Vastenhouw, Design and simulation of a high-resolution stationary SPECT system for small animals, *Phys. Med. Biol.* **49** (2004) 4579-4592.
[8] C. Merheb, Y. Petegnief, J.N. Talbot, Full modelling of the MOSAIC animal PET system based on the GATE Monte Carlo simulation code, *Phys. Med. Biol.* **52** (2007) 563-576.
[9] E. Vandervoort, M. Camborde, S. Jan, V. Sossi, Monte Carlo modelling of singles-mode transmission data for small animal PET scanners, *Phys. Med. Biol.* **52** (2007) 3169-3184.
[10] M.A. King, S.J. Glick, P.H. Pretorius, R.G. Wells, H.C. Gifford, M. Narayan, T. Farncombe, Attenuation scatter and spatial resolution compensation in SPECT, in: M.N. Wernick, J. Aarsvold (Eds.), *Emission Tomography: The Fundamentals of PET and SPECT*, Elsevier, 1st ed. San Diego, 2004, pp. 473-498.
[11] A.B. Hwang, B.L. Franc, G.T. Gullberg, B.H. Hasegawa, Assessment of the sources of error affecting the quantitative accuracy of SPECT imaging in small animals, *Phys. Med. Biol.* **53** (2008) 2233-2252.
[12] I. Buvat, I. Castiglioni, Monte-Carlo methods in PET and SPECT, *Quarterly J. Nucl. Med.* **46** (2002) 48-61.
[13] T.A. Riauka, R.H. Hooper, Z.W. Gortel, Experimental and numerical investigation of the 3D SPECT photon detection kernel for non-uniform attenuating media, *Phys. Med. Biol.* **41** (1996) 1167-1189.
[14] R.G. Wells, A. Celler, R. Harrop, Analytical calculation of photon distributions in SPECT projections, *IEEE Trans. Nucl. Sci.* **44** (1997) 3202-3214.
[15] D. Lazaro, Z. El Bitar, V. Breton, D. Hill, I. Buvat, Fully 3D Monte Carlo reconstruction in SPECT: a feasibility study, *Phys. Med. Biol.* **50** (2005) 3739-3754.

[16] Z. El Bitar, D. Lazaro, V. Breton, D. Hill, I. Buvat, Fully 3D Monte Carlo image reconstruction in SPECT using functional regions, *Nucl. Instr. Meth. Phys. Res.* A **569** (2006) 399-403.
[17] C. Morel, La simulation Monte Carlo en médecine nucléaire = Monte Carlo simulation in nuclear medicine, *Médecine Nucléaire* **31** (2007) 160-164.
[18] I. Buvat, D. Lazaro, Monte Carlo simulations in emission tomography and GATE: An overview, *Nucl. Instr. and Meth.* A **569** (2006) 323-329.
[19] V. Breton, R. Medina, J. Montagnat, DataGrid Prototype of a Biomedical Grid, *Meth. Inf. Med.* **42** (2003) 143-147.
[20] M. Mascagni, A. Srinivasan, Parameterizing parallel multiplicative lagged-Fibonacci generators, *Parallel Comput.* **30** (2004) 899-916.
[21] B. Nitzberg, J.M. Schopf, J.P. Jones, PBS Pro: Grid computing and scheduling attributes, in: J. Nabrzyski, J.M. Schopf, J. Weglarz (Eds.), *Grid resource management: state of art and future trends*, Kluwer Academic Publishers, Norwell USA, 2004, pp. 183-190.
[22] A.B. Yoo, M.A. Jette, M. Grondona, SLURM: Simple Linux Utility for Resource Management, in: D. Feitelson, L. Rudolph, U. Schwiegelshohn (Eds.), *Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing*, Springer, Seattle USA, **2862** (2003) 44-60.
[23] N. Capit et al, A batch scheduler with high level components, *Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid* **2** (2005) 776-783.
[24] W. Gentzsch, Sun Grid Engine: towards creating a compute power grid, *Proceedings of the First IEEE/ACM International Symposium on Cluster Computing and the Grid* (2001) 35-36.
[25] I. Foster, C. Kesselman, Globus: A metacomputing infrastructure toolkit, *International Journal of Supercomputer Applications and High Performance Computing* **11** (1997) 115-128.
[26] W. Cirne, F. Brasileiro, N. Andrade, L. Costa, A. Andrade, R. Novaes, M. Mowbray, Labs of the world, unite!!!, *J. Grid Comput.* **4** (2006) 225-246.
[27] A.R. Butt, R. Zhang,Y.C. Hu, A self-organizing flock of condors, *Journal of parallel and distributed computing*, **66** (2006) 145-161.
[28] Y. Georgiou, O. Richard, N. Capit, Evaluations of the lightweight grid CiGri upon the grid5000 platform, *Proceedings of the Third IEEE International Conference on e-Science and Grid Computing*, Washington USA, (2007) 279-286.
[29] Y. Georgiou, N. Capit, B. Bzeznik, O. Richard, Simple, fault tolerant, lightweight grid computing approach for bag-of-tasks applications, *3rd EGEE User Forum*, Clermont-Ferrand France, (2008).
[30] F. Dupros, F. Boulahya, J. Vairon, P. Lombard, N. Capit, J-F. Méhaut, IGGI, a computing framework for large scale parametric simulations: Application to uncertainty analysis with toughreact, *TOUGH Symposium 2006 proceedings*, California USA, (2006).
[31] M. Marinov, GlusterFS, Openfest 2007 Fifth Annual Conference, Sofia Bulgaria, (2007).
[32] S. Jan et al, GATE: a simulation toolkit for PET and SPECT, *Phys. Med. Biol.* **49** (2004) 4543-4561.
[33] S. Agostinelli et al, GEANT4: A simulation toolkit, *Nucl. Instrum. Methods* A **506** (2003) 250-3003.
[34] S. Staelens, D. Strul, G. Santin, S. Vandenberghe, M. Koole, Y. D'Asseler, I. Lemahieu, R. Van de Walle, Monte Carlo simulations of a scintillation camera using GATE: validation and application modelling, *Phys. Med. Biol.* **48** (2003) 3021-3042.
[35] N. Sakellios, J.L. Rubio, N. Karakatsanis, G. Kontaxakis, G. Loudos, A. Santos, K. Nikita, S. Majewski, GATE simulations for small animal SPECT/PET using voxelized

phantoms and rotating-head detectors, *IEEE Nucl. Sci. Symposium Conference Record* **4** (2006) 2000-2003.
[36] R. Taschereau, A.F. Chatziioannou, Compressed voxels for high-resolution phantom simulations in GATE, *Mol. Imaging Biol.* **10** (2007) 40-47.
[37] J. De Beenhouwer, S. Staelens, Y. D'Asseler, I. Lemahieu, Optimizing the Scalability of Parallelized GATE Simulations, *IEEE Nucl. Sci. Symposium Conference Record* **6** (2006) 3904-3908.
[38] L. Maigne, D. Hill, V. Breton, R. Reuillon, P. Calvat, D. Lazaro, Y. Legré, D. Donnarieix, Parallelization of Monte Carlo simulations and submission to a grid environment, *Parallel Process. Lett.* **14** (2004) 177-196.
[39] R. Reuillon, D.R.C. Hill, Z. El Bitar, V. Breton, Rigorous Distribution of Stochastic Simulations Using the DistMe Toolkit, *IEEE Trans. Nucl. Sci.* **55** (2008) 595-603.
[40] J. Aoun, V. Breton, L. Desbat, B. Bzeznik, M. Leabad, J. Dimastromatteo, Validation of the Small Animal Biospace Gamma Imager Model Using GATE Monte Carlo Simulations on the Grid, in: S. Olabarriaga, D. Lingrand, J. Montagnat (Eds.), Proceedings MICCAI-Grid Workshop, Medical imaging on grids: achievements and perspectives, New York, 2008, pp. 75-84.
[41] K. Assié, I. Gardin, P. Véra, I. Buvat, Validation of the Monte Carlo simulator GATE for indium-111 imaging, *Phys. Med. Biol.* **50** (2005) 3113-3125.
[42] F. Lamare, A. Turzo, Y. Bizais, C. Cheze Le Rest, D. Visvikis, Validation of a Monte Carlo simulation of the Philips Allegro/GEMINI PET systems using GATE, *Phys. Med. Biol.* **51** (2006) 943-962.
[43] P.H. Hargrove, J.C. Duell, Berkeley Lab Checkpoint/Restart (BLCR) for linux clusters, *J. Phys.: Conf. Ser.* **46** (2006) 494-499.
[44] N. Jacq, J. Salzemann, F. Jacq, Y. Legré, E. Medernach, J. Montagnat, A. Maass, M. Reichstadt, H. Schwichtenberg, M. Sridhar et al, Grid enabled virtual screening against malaria, *Journal of Grid Computing* **6** (2008) 29-43.

Vitae

Joe Aoun received the MSc degree in particle physics from the Blaise Pascal University, Clermont-Ferrand, France, in 2005.
He is currently a PhD candidate in the Laboratory of Techniques of medical engineering and complexity – Informatics, mathematics and application of Grenoble at the Joseph Fourier University (TIMC-IMAG, Grenoble, France) and in the Laboratory of Particle Physics (LPC) at the Blaise Pascal University.
He is involved in Single Photon Emission Computed Tomography and CT-scan (SPECT/CT) images registration with the Data Consistency Conditions and Monte Carlo simulations for SPECT imaging on Grids.

Bruno Bzeznik currently works as a Research Engineer (Grenoble University) on the CIMENT project (*http://ciment.ujf-grenoble.fr*). He is the gridmaster of the local grid and acts as a UNIX and network expert on the HPC platforms of the University. During another half of its working time, he collaborates in the Laboratory of Informatics of Grenoble (LIG) in the MESCAL team, as a contributor to the administration of the national IT grid Grid'5000 and co-developer of the grid tools OAR, OARGRID and CiGri. This latter activity helps Bruno Bzeznik in being a gateway between HPC-research and HPC-production in the local area.

Vincent Breton received his Engineer degree from Ecole Centrale de Paris in 1985 and his PhD in Nuclear Physics from the University of Paris XI-Orsay in 1990. Since 1990, he has been a Researcher at the French National Centre for Scientific Research (CNRS).
In 2001, he founded a research group (*http://clrpcsv.in2p3.fr*) on the application to biomedical sciences of the IT technologies and tools used in high energy physics. Co-founder of the GATE collaboration (*http://opengate.in2p3.fr*) gathering more than 20 research laboratories around the world, co-founder of the Healthgrid (*http://www.healthgrid.org*) and WISDOM (*http://wisdom.healthgrid.org*) initiatives, he has been leading the French national GLOP (ACI GRID 2002) and regional Auvergrid grid projects as well as coordinating work packages in several European projects within FP5 (DataGrid) and FP6 (EGEE, BioInfoGrid, Embrace, SHARE, EGEE-II) Framework Programmes.

Mehdi Leabad received the B.S. degree in Physics from Grenoble Polytechnics National Institute (INPG), Grenoble, France, in 2004.
After working as a technical engineer in multimodal preclinical imagery at the Laboratory of Electronics and Information Technologies (LETI), he is currently product engineer of Biospace Lab, an international preclinical imagery company. He manages principally the developments of SPECT/CT and autoradiography imagers.

Julien Dimastromatteo received the MSc degree in medical imaging, in 2005, from the University of Grenoble, Grenoble, France.
He's currently a PhD student in the Bioclinical Radiopharmaceutics Laboratory, INSERM U877, Grenoble, France. His research interests include medical imaging and experimental models of cardiovascular diseases.

Yiannis Georgiou received the B.S. degree in Electronic & Computer Engineering from the Technical University of Crete, Chania, Greece in 2003 and the M.S degree in Systems and Software from the Joseph Fourier University, Grenoble, France in 2005.
He is currently a Ph.D candidate in the Joseph Fourier University, Grenoble, France.
His work is supported by LIG (Laboratory of Informatics of Grenoble) and BULL (R&D HPC department).
His research interests focus on High Performance Computing systems architectures and Resource Management upon Clusters and Grids.

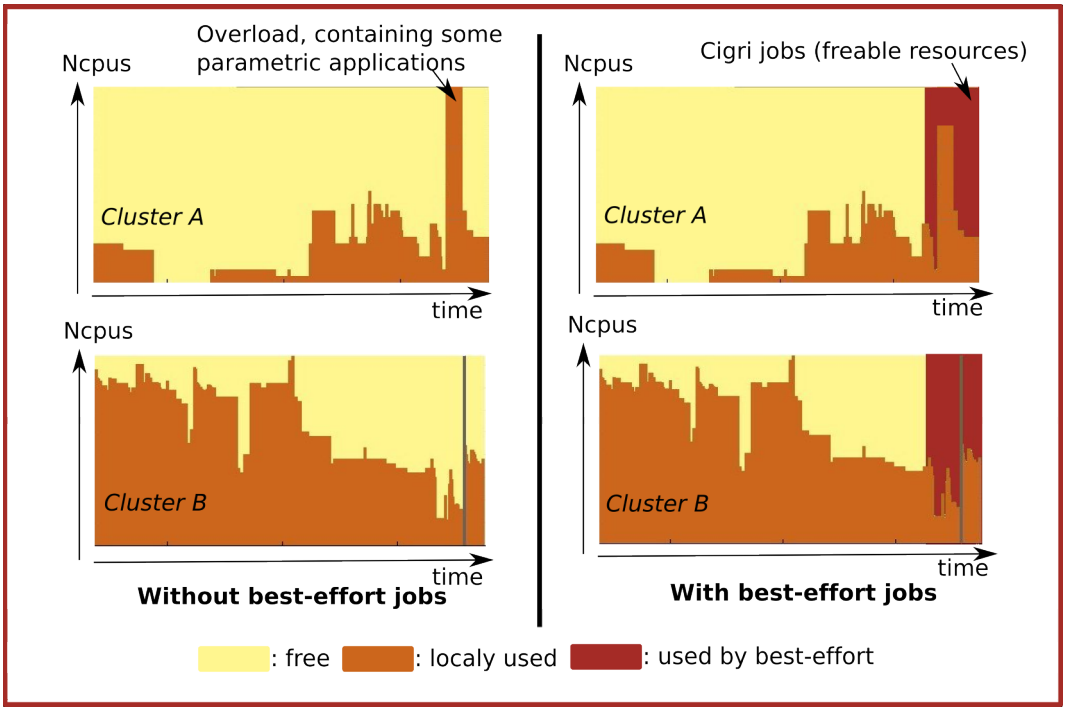
Olivier Richard received the M.D. and Ph.D. degrees from the University of Paris-Sud, Paris, France in 1995 and 1999, respectively.
He is currently an Associate Professor at the University of Joseph Fourier of Grenoble, France. He leads his research at the Laboratory of Informatics of Grenoble (LIG). His research interests include workload and resources management on cluster, grid and P2P platforms, and design of platforms dedicated to experimentation like Grid'5000 or Planetlab.

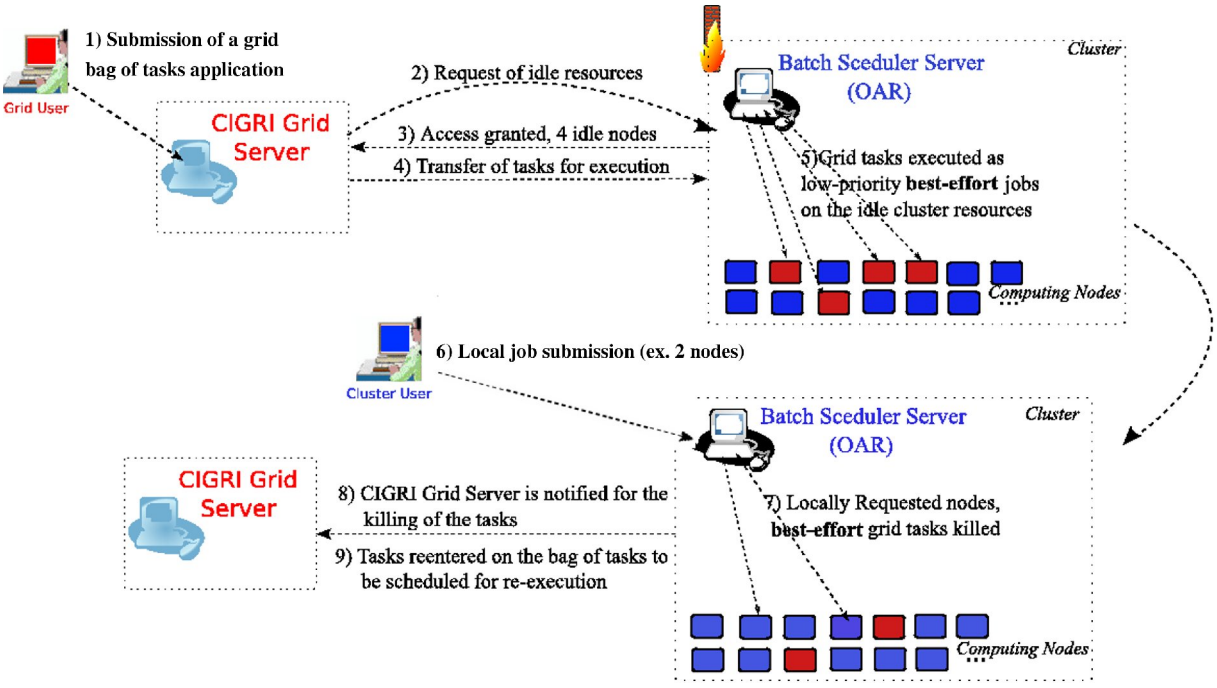
Pierre Neyron developed the ComputeMode software while member of the Icatis company. In 2005, he joined INRIA (The French National Institute for Research in Computer Science and Control) to work as a research engineer and took the head of Grid'5000 technical committee. He is recently working at Yahoo.

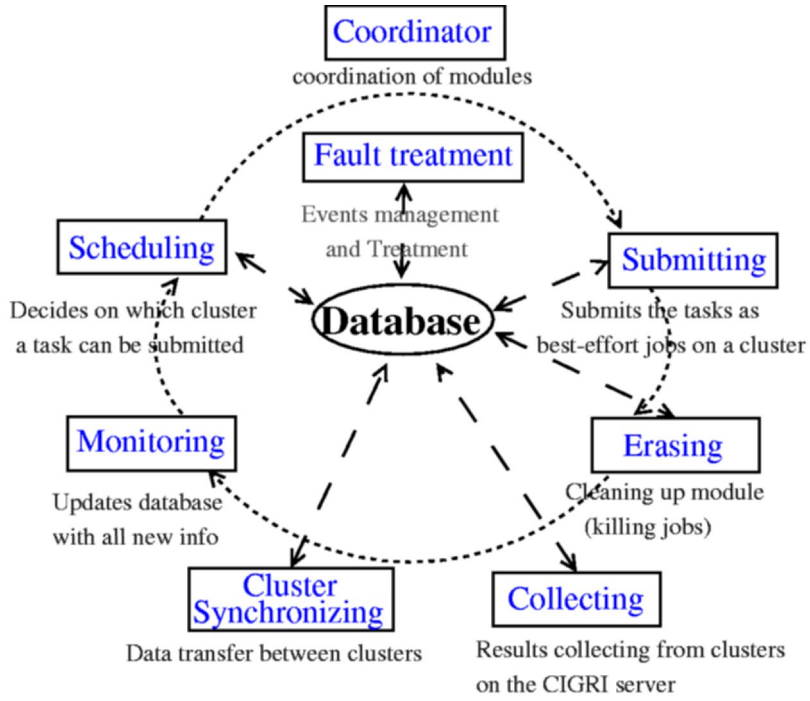
Laurent Desbat received the computer science and applied math engineer degree at ENSIMAG/INPG in 1987, and the MD and PhD degrees at UJF Grenoble in 1987 and 1990, respectively.

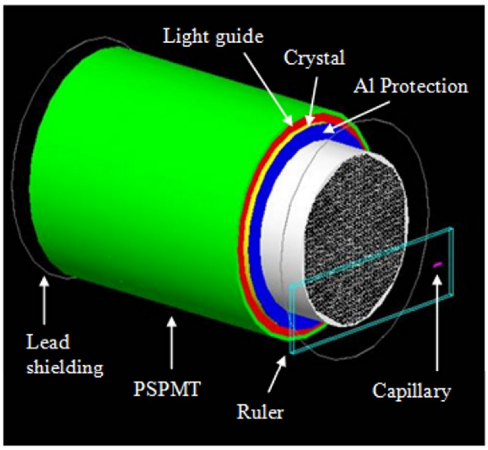
He is currently applied maths Professor at UJF Grenoble. His research at TIMC-IMAG concerns inverse problem and medical imaging and more particularly tomography, nuclear imaging, dynamic CT, sampling. He is responsible for the intensive scientific computing project CIMENT at Grenoble University for ten years.

Figures (in color on the web and in black and White for printing)

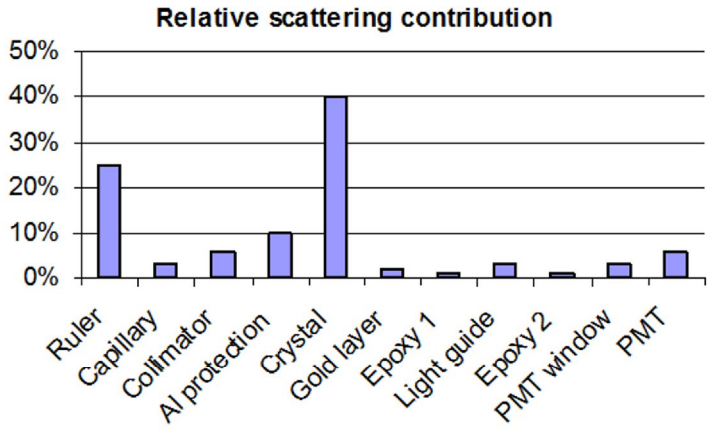




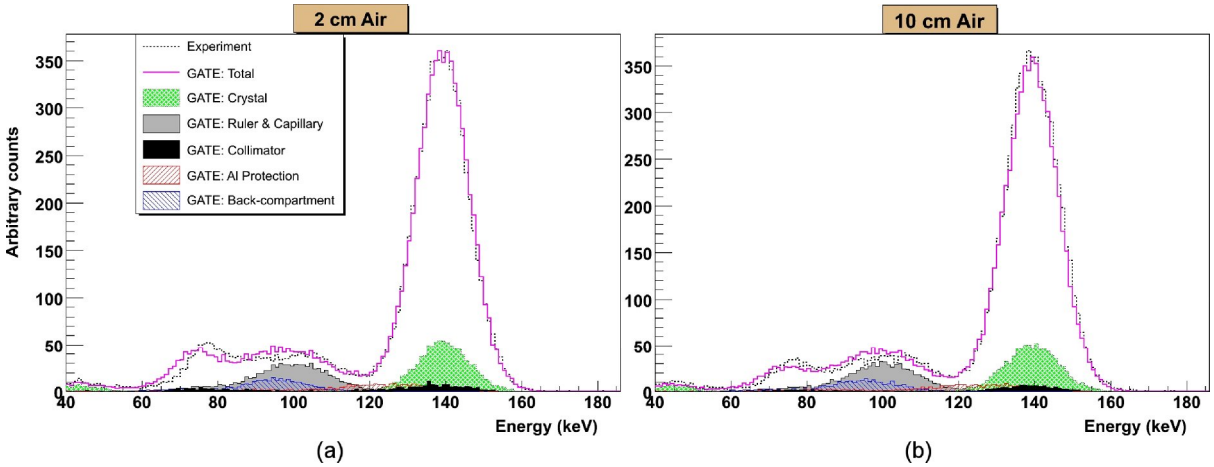

Coordinator
coordination of modules
Fault treatment
Events management
and Treatment
Scheduling
Decides on which cluster
a task can be submitted
Database
Submitting
Submits the tasks as
best-effort jobs on a cluster
Monitoring
Updates database
with all new info
Erasing
Cleaning up module
(killing jobs)
Cluster
Synchronizing
Data transfer between clusters
Collecting
Results collecting from clusters
on the CIGRI server


Light guide
Crystal
Al Protection
Lead
shielding
PSPMT
Ruler
Capillary

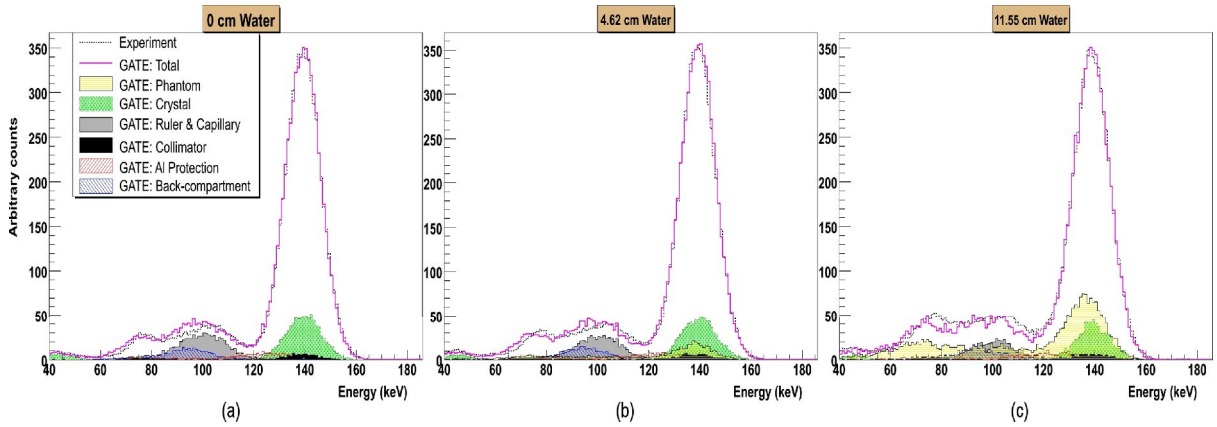
(a)

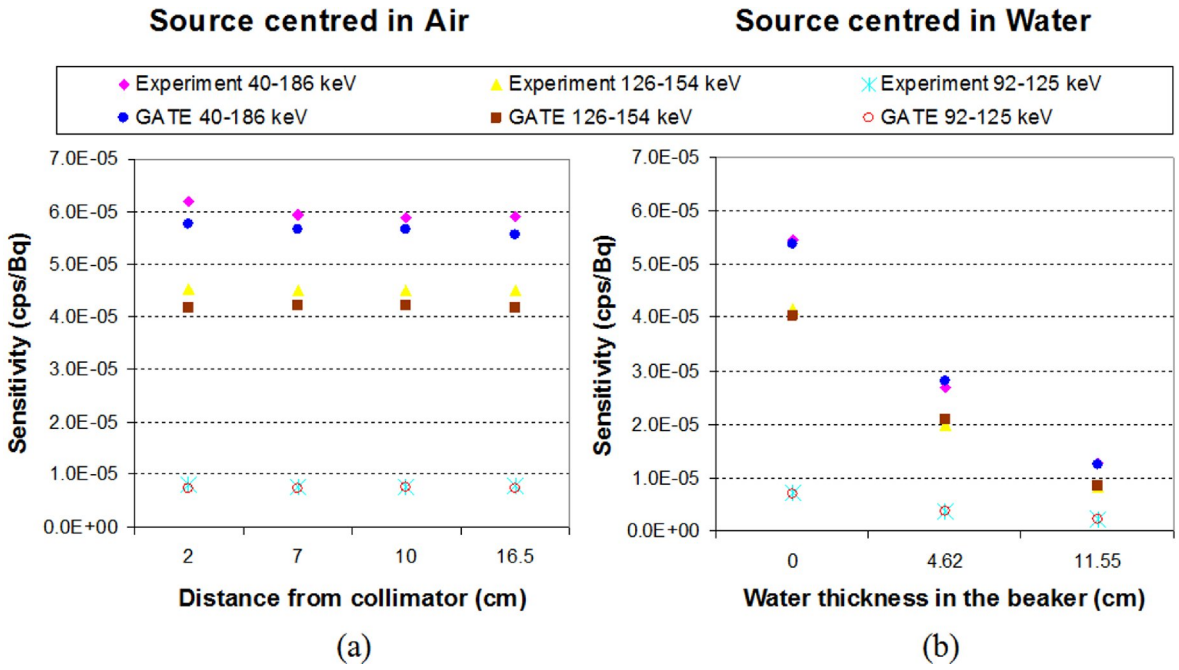

Relative scattering contribution
50%
40%
30%
20%
10%
0%
Ruler
Capillary
Collimator
Al protection
Crystal
Gold layer
Epoxy 1
Light guide
Epoxy 2
PMT window
PMT

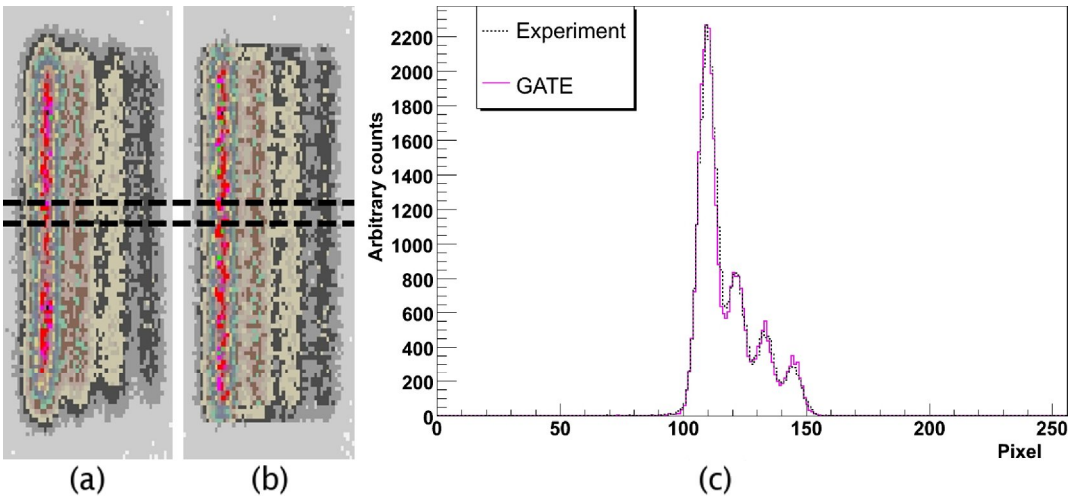
(b)


2 cm Air
10 cm Air
Experiment
GATE: Total
GATE: Crystal
GATE: Ruler & Capillary
GATE: Collimator
GATE: Al Protection
GATE: Back-compartment
Arbitrary counts
Energy (keV)
Energy (keV)

(a) (b)


0 cm Water
4.62 cm Water
11.55 cm Water
Experiment
GATE: Total
GATE: Phantom
GATE: Crystal
GATE: Ruler & Capillary
GATE: Collimator
GATE: Al Protection
GATE: Back-compartment
Arbitrary counts
Energy (keV)
(a)
(b)
(c)


Source centred in Air
Source centred in Water
Experiment 40-186 keV
Experiment 126-154 keV
Experiment 92-125 keV
GATE 40-186 keV
GATE 126-154 keV
GATE 92-125 keV
Sensitivity (cps/Bq)
Distance from collimator (cm)
Water thickness in the beaker (cm)
(a)
(b)


Experiment
GATE
Arbitrary counts
Pixel
(a)
(b)
(c)

Legends

Figure 1. Example of load balancing of a grid composed of 2 clusters.

Figure 2. Best-effort jobs.

Figure 3. CiGri modules and their operation.

Figure 4. (a) The Biospace small animal camera modelled with GATE, (b) contribution of the scattered photons in each component of the simulation where the source is placed at 2 cm from the collimator.

Figure 5. Experimental and simulated energy spectra for a ^{99m}Tc source positioned in air at (a) 2 cm and (b) 10 cm from the camera.

Figure 6. Experimental and simulated energy spectra for a ^{99m}Tc source positioned at 16.5 cm from the collimator above a beaker filled with (a) 0 cm, (b) 4.62 cm and (c) 11.55 cm of water.

Figure 7. Comparison between experimental and simulated system sensitivities in three energy windows for the centred source placed at: (a) 2, 7, 10 and 16.5 cm in air; (b) 16.5 cm with three water thickness 0, 4.62 and 11.55 cm.

Figure 8. Comparison between (a) experimental and (b) simulated images of a four capillaries phantom filled with different ^{99m}Tc concentrations and (c) horizontal profiles through the middle of these images.

Table 1. Number of clusters and CPUs available to use GATE on CiGri.

Table 2. Comparison between experimental and simulated spatial resolutions for the centred and off-centred sources placed at: 2, 7, 10 and 16.5 cm in air; 16.5 cm with three water thickness 0, 4.62 and 11.55 cm.

Table 3. Comparison between computing time of 1 and 200 simulations on a local CPU (Pentium IV, 3.2 GHz, 1 Go RAM) and on the CiGri Grid with the percentage of the resubmitted jobs.

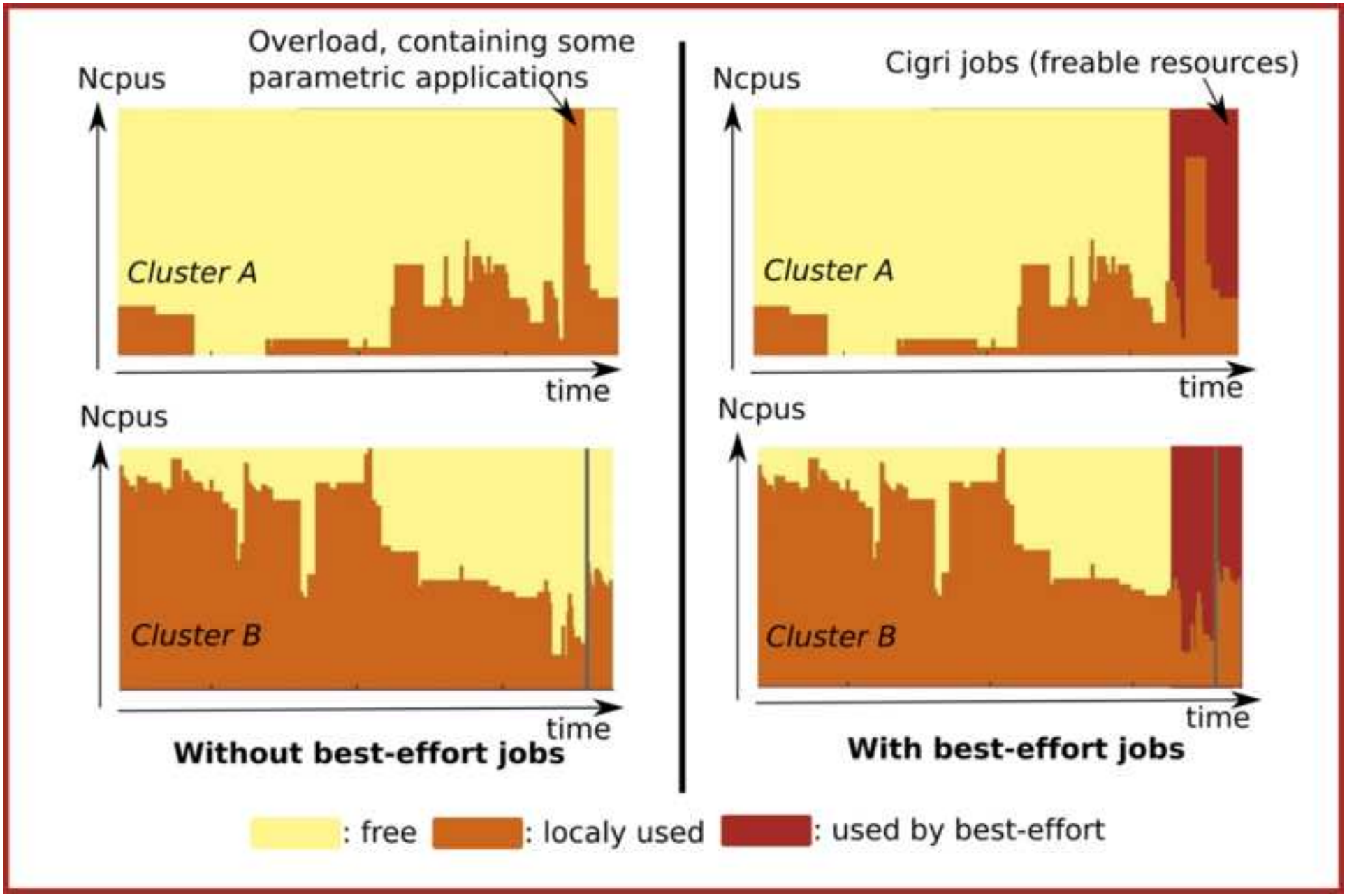
Tables

	Total		GATE availability (max)		GATE availability (average)	
	Clusters	CPUs	Clusters	CPUs	Clusters	CPUs
Day	11	886	7	430	7	125
Night and Weekend	11	886	7	555	7	215

Source-collimator distance (water thickness)	Centred source			Off-centred source		
	Experiment FWHM (mm)	Simulation FWHM (mm)	Difference FWHM (%)	Experiment FWHM (mm)	Simulation FWHM (mm)	Difference FWHM (%)
2 cm	3.34	3.22	3.64	3.58	3.20	10.5
7 cm	4.53	4.54	1.89	4.75	4.53	4.49
10 cm	5.32	5.40	1.47	5.61	5.39	3.95
16.5 cm	7.25	7.34	1.30	7.64	7.37	3.49
16.5 cm (0 cm)	7.27	7.39	1.63	7.63	7.38	3.32
16.5 cm (4.62 cm)	7.26	7.32	0.91	7.62	7.26	4.69
16.5 cm (11.55 cm)	7.58	7.65	0.91	7.83	7.61	2.95

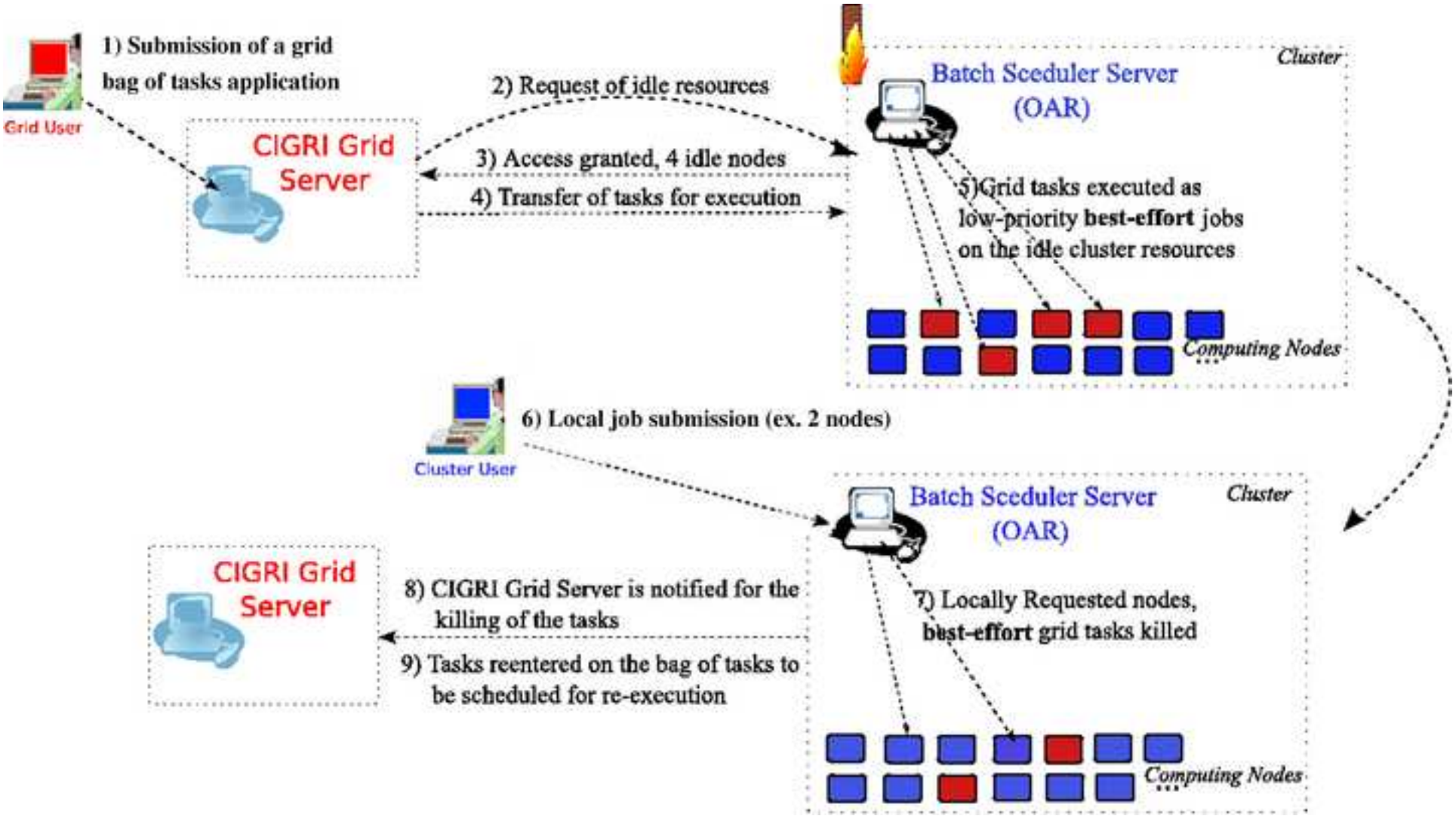
	1 simulation (1000 jobs)	200 simulations	Gain	Resubmission percentage (%)
Local CPU	167 h	1392 days (~4 years)	1	0
CiGri – day	4 h	37 days	42	16.9
CiGri – night and weekend	2.5 h	21 days	67	10.2
CiGri – estimated average	3h	25 days	56	13.4

Figure(s) 1
Click here to download high resolution image

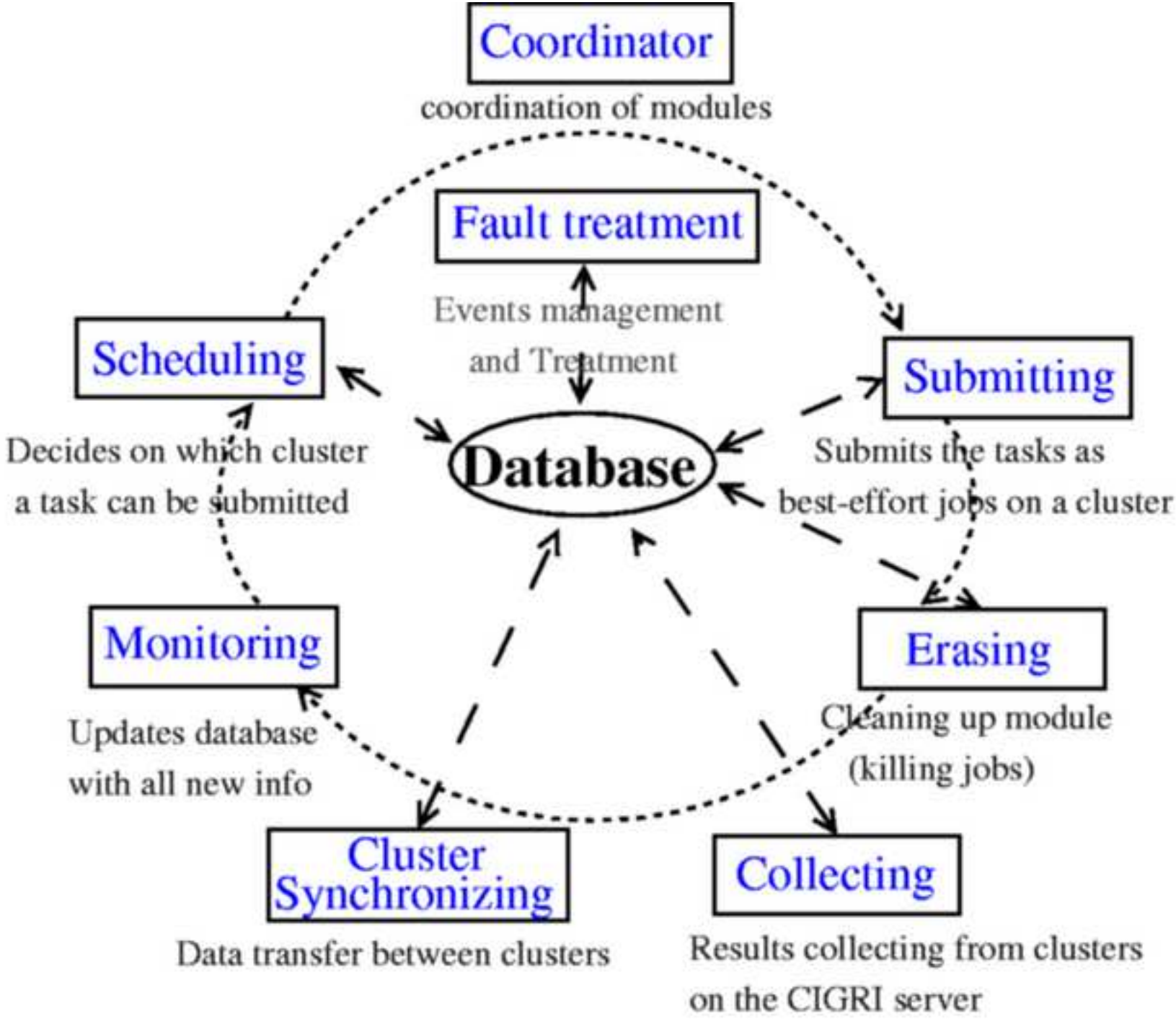


Figure(s) 2

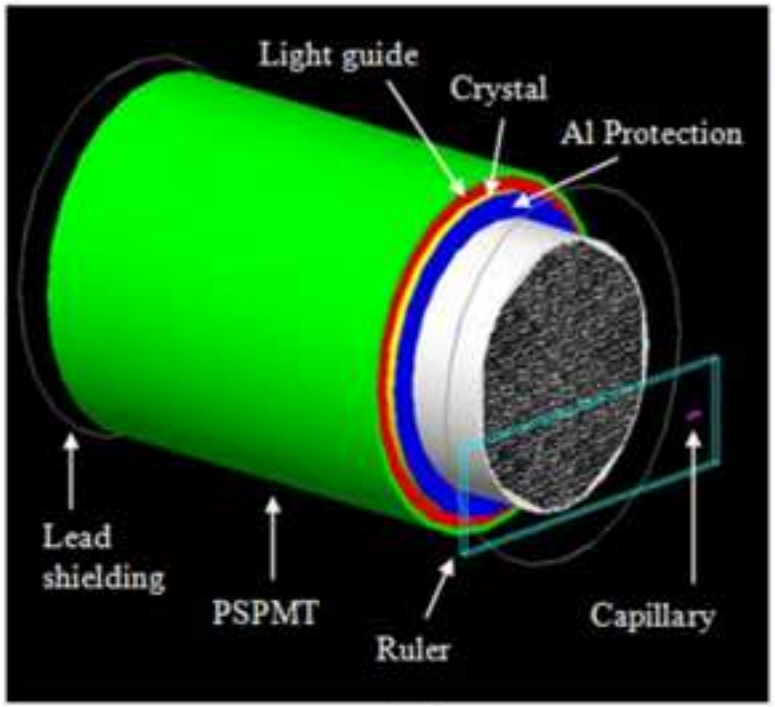
Click here to download high resolution image



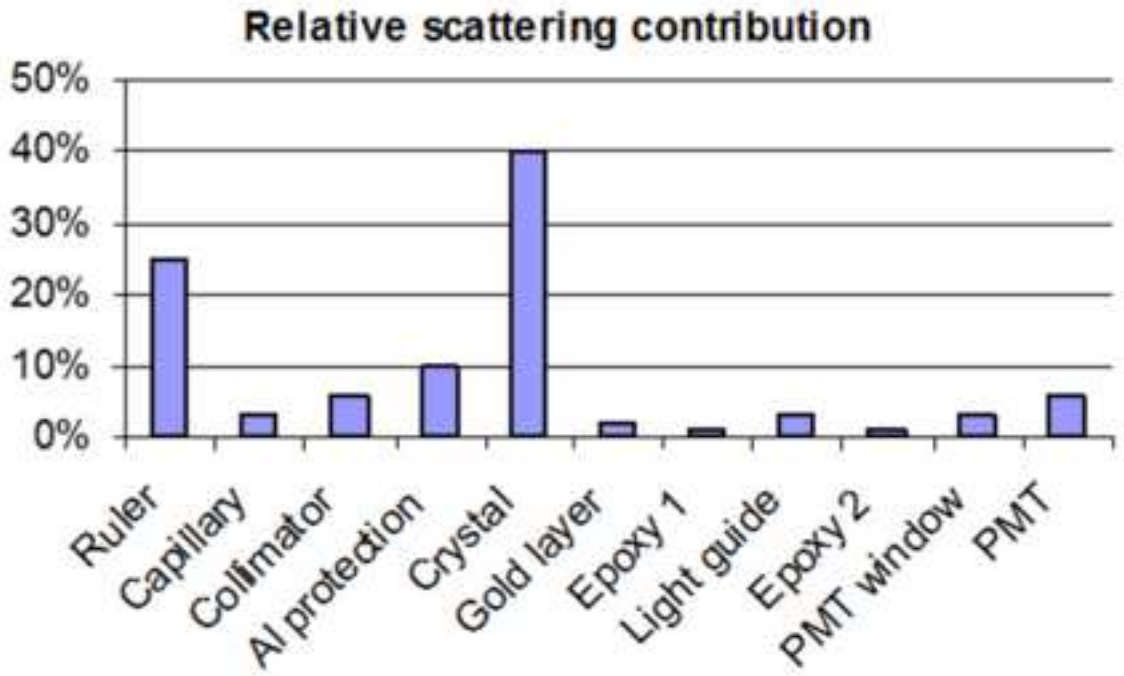
Figure(s) 3
Click here to download high resolution image



Figure(s) 4
Click here to download high resolution image



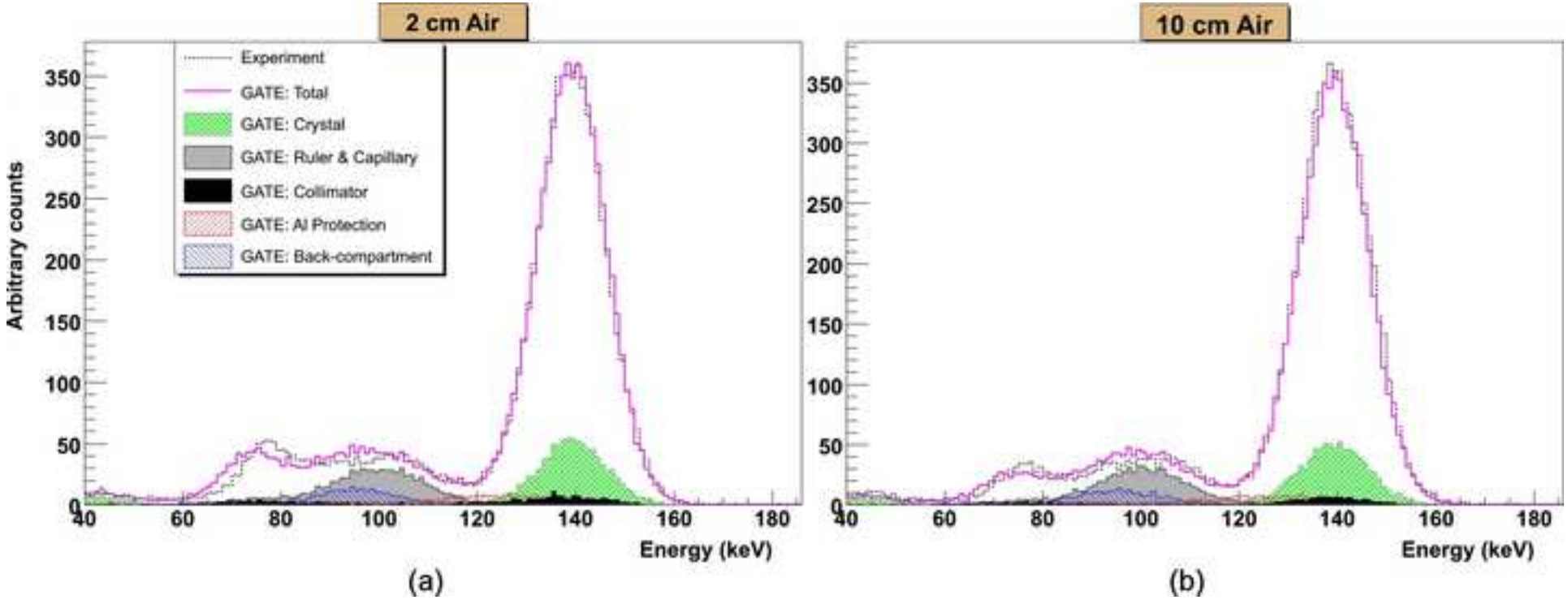
(a)



(b)

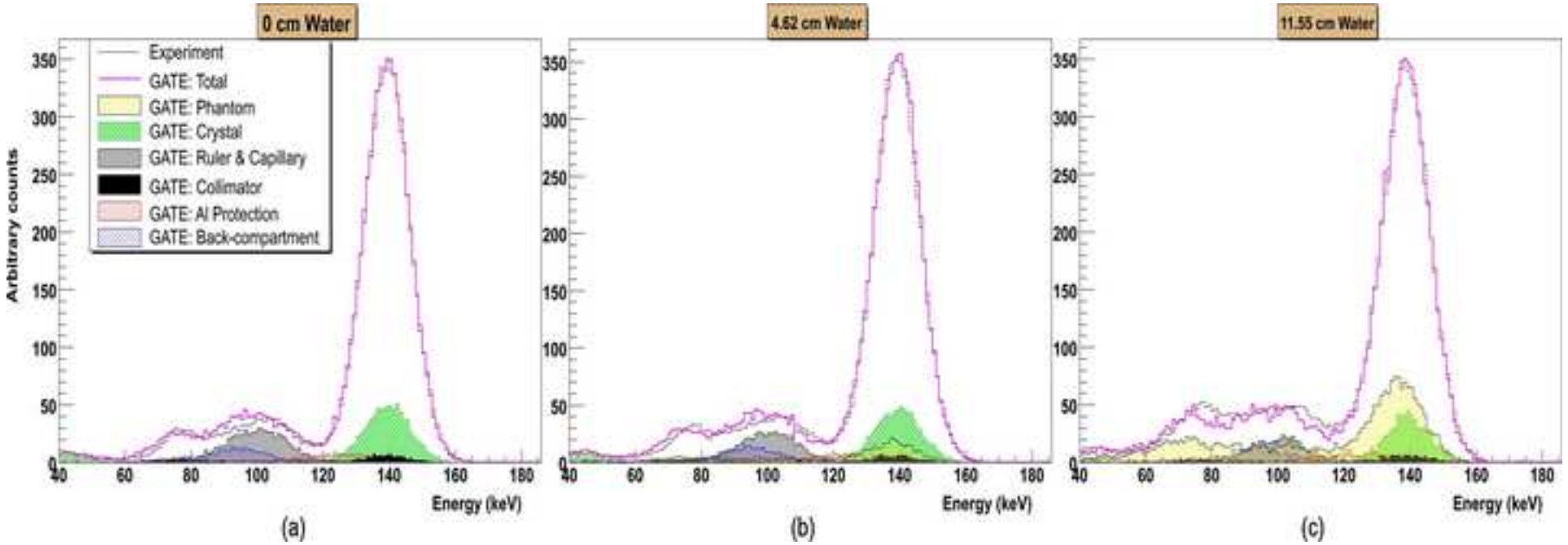
Figure(s) 5

Click here to download high resolution image



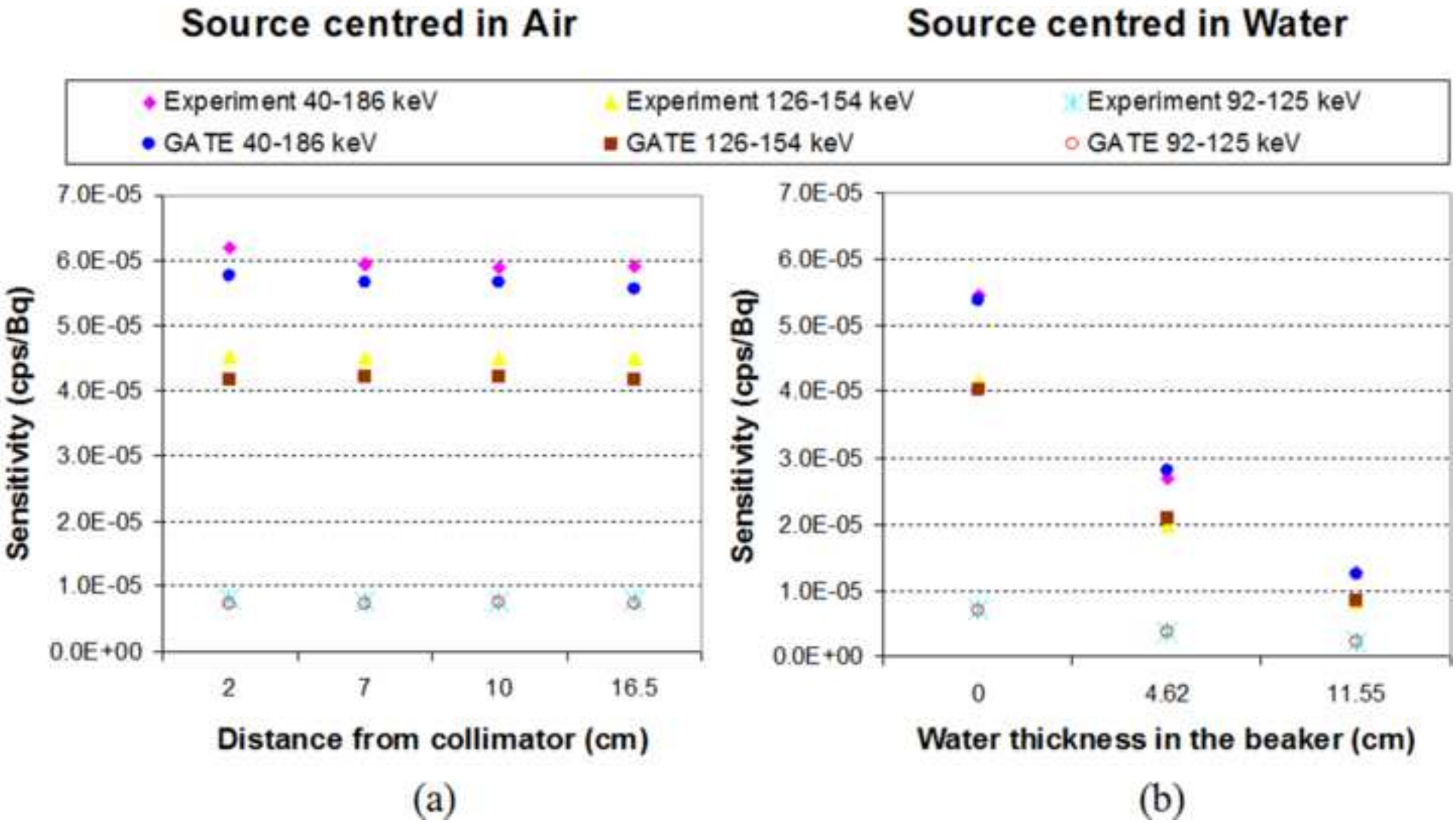
Figure(s) 6

Click here to download high resolution image

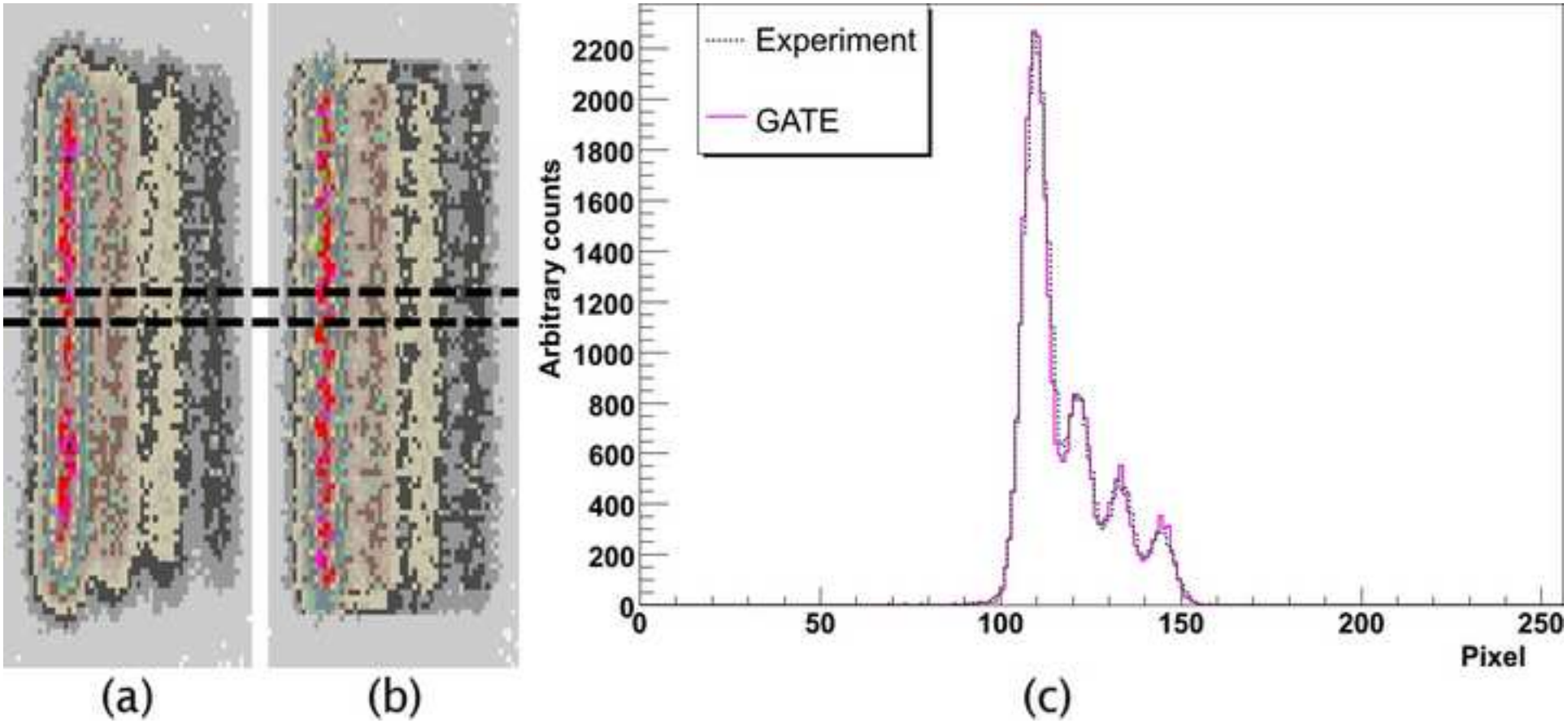


Figure(s) 7

Click here to download high resolution image



Figure(s) 8
Click here to download high resolution image



Table(s) 1

	Total		GATE availability (max)		GATE availability (average)	
	Clusters	CPUs	Clusters	CPUs	Clusters	CPUs
Day	11	886	7	430	7	125
Night and Weekend	11	886	7	555	7	215

Table(s) 2

Source-collimator distance (water thickness)	Centred source			Off-centred source		
	Experiment FWHM (mm)	Simulation FWHM (mm)	Difference FWHM (%)	Experiment FWHM (mm)	Simulation FWHM (mm)	Difference FWHM (%)
2 cm	3.34	3.22	3.64	3.58	3.20	10.5
7 cm	4.53	4.54	1.89	4.75	4.53	4.49
10 cm	5.32	5.40	1.47	5.61	5.39	3.95
16.5 cm	7.25	7.34	1.30	7.64	7.37	3.49
16.5 cm (0 cm)	7.27	7.39	1.63	7.63	7.38	3.32
16.5 cm (4.62 cm)	7.26	7.32	0.91	7.62	7.26	4.69
16.5 cm (11.55 cm)	7.58	7.65	0.91	7.83	7.61	2.95

Table(s) 3

	1 simulation (1000 jobs)	200 simulations	Gain	Resubmission percentage (%)
Local CPU	167 h	1392 days (~4 years)	1	0
CiGri – day	4 h	37 days	42	16.9
CiGri – night and weekend	2.5 h	21 days	67	10.2
CiGri – estimated average	3h	25 days	56	13.4