Intelligent Grid Enabled Services for Neuroimaging Analysis

Richard McClatchey[a], Irfan Habib[a], Ashiq Anjum[b], Kamran Munir[a], Andrew Branson[a], Peter Bloodsworth[c], Saad Liaquat Kiani[a], and the neuGRID Consortium

[a]*Centre for Complex Cooperative Systems, University of the West of England, UWE, Bristol, BS16 1QY, UK*
[b]*School of Computing and Mathematics, University off Derby, Derby, DE221GB, UK*
[c]*School of Electrical Engineering & Computer Science, National University of Science and Technology (NUST), Islamabad, Pakistan*

Abstract

This paper reports our work in the context of the neuGRID project in the development of intelligent services for a robust and efficient Neuroimaging analysis environment. neuGRID is an EC-funded project driven by the needs of the Alzheimer's disease research community that aims to facilitate the collection and archiving of large amounts of imaging data coupled with a set of services and algorithms. By taking Alzheimer's disease as an exemplar, the neuGRID project has developed a set of intelligent services and a Grid infrastructure to enable the European neuroscience community to carry out research required for the study of degenerative brain diseases. We have investigated the use of machine learning approaches, especially evolutionary multi-objective meta-heuristics for optimising scientific analysis on distributed infrastructures. The salient features of the services and the functionality of a planning and execution architecture based on an evolutionary multi-objective meta-heuristics to achieve analysis efficiency are presented. We also describe implementation details of the services that will form an intelligent analysis environment and present results on the optimisation that has been achieved as a result of this investigation.

Keywords: Intelligent Services, Machine Learning and Genetic Algorithms, Grid Enabled Planning and Execution, Service Oriented Architecture, Neuroimaging Analysis

1. Introduction

Alzheimer's disease is a progressive, degenerative and irreversible brain disorder that causes intellectual impairment, disorientation and eventual death. It is the most common cause of dementia, accounting for around two thirds of cases in the elderly. It is estimated that 2-5% of people over 65 years of age and up to 20% of those over 85 years of age suffer from the disease. The study of Alzheimer's disease (AD), its causes, its symptoms and especially its early diagnosis is now a major driver in the provision of healthcare for the elderly. Early diagnosis is beneficial for several reasons. Having an early diagnosis and starting treatment in the early stages of the disease can help preserve function for months to years and can aid caring strategies and support networks.

Distributed computing infrastructure based workflows are being utilised in a wide range of scientific research domains [1, 2]. Alzheimer's clinical researchers are currently seeking the assistance of large-scale information technology resources to enable them to study masses of neuroimaging data being accumulated across the older patient community so that early onset indicators such as cortical thinning can be studied [3] [4]. Rapid advances in neuroimaging technologies such as PET, SPECT, MR spectroscopy, DTI and fMRI have offered a new vision into the pathophysiology of AD [5] and, consequently, new increasingly powerful data analysis methods have been developed [6]. Since the beginning of the new century the development of innovative techniques for ROI-based volumetry, automated voxel based morphometry, cortical thickness measurement, basal forebrain volumetry, and multivariate statistics have emerged [7, 8]. The availability of large image data repositories to the neuroimaging community has necessitated the development of distributed data and processing infrastructures to access data and online image analysis tools and to assess longitudinal brain changes [9, 10, 11, 12].

Many efforts have been directed at creating brain image repositories including the recent US Alzheimer Disease Neuroimaging Initiative (ADNI) [13]. Numerous efforts, such as NeuroLOG [14]and Neurogrid[15], have been conducted which focus on providing grid infrastructures that support neuroimaging application [16]. At present, however, these applications tend to be either focused on specific pathologies or are directed at supporting a subset of neuroimaging applications. Moreover, these solutions are tightly bound to specific platforms, which may limit their wider adoption across neuroscience. neuGRID is an effort which targets the limitations of existing neuroimaging based Grid infrastructures and aims to provide an infrastructure and a set of complementary analysis services that are designed to support and enhance research. neuGRID is an EC-funded effort which will allow the collection and archiving of large amounts of imaging data paired with services, Grid-based algorithms and computational resources. The major benefit will be faster discovery of new disease markers that will be valuable for earlier diagnosis and development of innovative drugs.

It needs to be stressed that some of the presently available algorithms can take many hours per brain to run on a state-of-the-art workstation [17]. The modus operandi today is that of scientists physically migrating image data to remote imaging centres where they can find expertise and computational facilities for analysing small personal datasets (a few hundreds of images at most). Typically, a research fellow can spend months at an image analysis centre where he/she learns the use of the algorithms on personal image data, then returns to the original research group, where he/she can install all or part of the procedure and run jobs either in house or remotely on the image analysis centre servers. This scenario is becoming unsustainable and it needs to change radically in the near future. Conventional file sharing mechanisms e.g. peer-to-peer file sharing, can be used to share image and clinical data, however such mechanisms still require the researchers to feed in the data to computational analysis programmes. The benefits of such data sharing on a Grid based infrastructure include the fact that the data remains *online*, it can be shared across organisational boundaries through the concept of virtual organisations in the Grid, better resource utilisation through Grid scheduling and better access control.

Neuroimaging researchers require infrastructures that can enable the large-scale computation of standardised pipelines on large data sets provided by the major data repositories. Domain researchers also require an infrastructure that enables collaborative studies that may involve multiple geographically dispersed research centres. However efficiently optimising the neuroimaging pipelines that are both compute and data intensive on an e-Science infrastructure poses various challenges. First, these pipelines consist of a large number of tasks. The CIVET pipeline [18], for instance, can consist of 108 tasks and the workflow turn-around time is around eight hours for a single brain scan. Secondly, these pipelines can generate a large amount of data. CIVET has been shown to produce ten times more data than it consumes [19]. This can add up to several terabytes for larger studies and several months of computations. Thirdly, neuroimaging pipelines consist of a large number of fine-grained tasks that have shown to severely affect the turn-around time of the workflow. However workflow optimisation methods have not kept pace with the rise of complexity in workflows, hence researchers have called for new approaches to optimising, managing and enacting them. Moreover, they need multi-criteria optimisation methods that can effectively optimise workflows for computation.

To achieve a low turn-around time (compute optimisation), computations within a workflow must be distributed in order to benefit from parallelism. On the other hand, to achieve data efficiency computations must be localised in order to limit expensive network transfers. We used a multi-objective meta-heuristic to optimise scientific workflows and evaluated through a number of real world scientific workflows – focusing on the CIVET [18] workflow in particular.

The domain of multi-objective meta-heuristics has been an active area of research [20] and various successful applications have been reported. For instance, several multi-objective evolutionary approaches have been used to optimise distributed computing capabilities such as scheduling [21] and classification [22]. However their use in the optimisation of scientific workflows has not been explored. Since the compute and data performance may be dependant on various factors, the search space of all possible optimised workflow plans may be large. An evolutionary meta-heuristic, being a stochastic population-based search algorithm, enables the simultaneous exploration of a search space as members of a population can be randomly distributed across the search space. Moreover, the genetic operations of mutation and crossover can enable the fine-grained control of the balance between exploitation (the ability to leverage characteristics of known solutions) and exploration (the ability to explore new parts of the search space). Multi-objective evolutionary algorithms (MOEAs) regarded as state-of-the-art include the Non-dominated Sorting Genetic Algorithms II (NSGA-II) [23], Strength Pareto Evolutionary Algorithm 2 (SPEA2) [24], Indicator based Evolutionary Algorithm (IBEA) [25] and HyPE [26].

In this paper we present work on the set of intelligent services in the neuGRID project that has been specified in consultation with its user community and developed to facilitate neuroimaging analysis, such as Alzheimer's studies. The services, using machine learning approaches, can intelligently plan, execute and 'glue' a spectrum of user applications to a range of available Grid platforms thereby creating a foundation for pervasive cross-platform services for neuroimaging analysis and promoting interoperability between diverse projects in this domain. This paper provides the background for understanding the characteristics of scientific analyses, highlights the issues that influence their optimisation and presents an approach for their intelligent planning and execution.

2. A Service Oriented Analysis Environment in NeuGRID

In order to facilitate analysis and collaboration that can address the community's requirements, a service oriented analysis environment has been proposed in neuGRID in which high-level distributed services such as querying, workflow management, provenance, and anoymization services [27] coordinate and interact to support user analyses. Such services will help the users in sharing data and knowledge and should enrich medical decision support systems [28]. The preferred approach is to implement a service oriented architecture (SOA) [29]. The service layer in neuGRID was implemented using the SOA paradigm in order to have a flexible and reusable medical services layer, which can be customized for various applications. The following paragraphs illustrate how the services in the neuGRID analysis environment, will coordinate to facilitate the neuroimaging analysis process, using SOA principles. In the later sections we emphasise how these services can intelligently support the analyses and improve the planning and execution process on distributed infrastructures.

The first action in the neuroimaging analysis cycle, shown in Figure 1, is to register images in the neuGRID store that have been collected from the hospital data acquisition systems or have been imported from other research projects. The border in Figure 1 denotes the limit of the neuGRID project infrastructure as determined from users' requirements. As an example, consider a new clinical site that may wish to make use of the neuGRID infrastructure to share data within a wider research community. Existing data would be put through a process that enforces quality control, anonymisation and ethical compliance. The data is then integrated with the neuGRID data model, which enables other researchers to access it and carry out their research. As new data sets are acquired they go through a local quality control step before passing through the same system-wide quality control, formatting, ethical compliance and data model integration processes.

Once the data has been registered, the next step in the analysis process is to make the data browsable through automated querying tools. Consequently, an appropriate data access mechanism needs to be put in place. For example, a researcher may be interested in a rare form of a disease and may want to carry out a statistically significant analysis. However, the researcher's institution may not have sufficient images to enable this. The user would interact with the system using the neuGRID store, to identify an appropriate set of images from a group of hospitals that match the required criteria. At this stage access controls and ethical policies are fully enforced to protect sensitive data. The researchers then use the system to submit the study set for analysis through a workflow.

Once the data has been imported into the neuGRID system, the users may want to carry out studies and data analyses to find results of interest. Workflow development is a methodology that can be used to represent user preferences for an intelligent analysis of data. Users may create workflows and then execute them on distributed resources provided by the Grid. For example, a researcher may wish to run a comparative analysis using a study set of 3000 MRI scans stored in geographically distributed medical centres. The user would interact with the system to choose a study set of 3000 images, would select the pipeline or workflow through which the analysis will take place and would start the analytical process. Users are not limited to using previously specified workflows and study samples, they can also construct new workflows.

It is important that results, as and when required, should be reproduced and reconstructed using past information. The verification of the results using audit trail information is known as 'provenance' verification [30]. The validation of results using provenance data is an important aspect in the analysis process. Often it may be necessary to validate and, if required, reproduce the workflow that has been used to obtain the results. This makes users confident not only on the results that have been produced but also on the process that led them to generate these results. After results have been produced, the user can examine the provenance to check that each stage of the analysis has been completed correctly. The raw results can then be exported into the user's preferred analysis tool and the whole process can be added to the researcher's history for future reference. Without the mechanism to validate workflows, it would not be possible to correct erroneous processes and generate accurate results. Once a workflow has been developed and verified, a user should be able to share it with other researchers. These analysis tracking and refinement techniques are, of course, not specific to the neuroimaging domain, nor indeed to the medical domain. Therefore any services that are produced to handle them should be generalizable across (at least) the medical domain.

A service oriented analysis environment can enable the construction of a catalogue of reusable services that can be customized and reused across domains and applications with minimal change. The users can use part of, or whole services to conveniently build their applications by exploiting well defined sets of interfaces that come with each service. This suggests that a SOA is a very viable distributed environment on which to design and build the analysis environment for neuGRID.

3. Architecture and Philosophy

In order to enhance the reusability of the neuGRID services, one of the major design considerations was to develop them in a manner that keeps them independent from the underlying Grid middleware. The neuGRID services (as shown in Figure 2) have been designed to be middleware agnostic and to hide the heterogeneity of underlying distributed resources through a common abstraction layer. Commonly, service interfaces would need to be reconfigured with each new Grid middleware release in order to cope with the software evolution. Services in neuGRID will not have to undergo these changes since the abstraction layer will shield the services from evolution of the underlying middleware. This design philosophy will increase the flexibility of the neuGRID service layer and allow it to make use of emerging advances in Grid (and Cloud) technologies with minimal changes to the service interfaces.

The lower-level neuGRID services hide the peculiarities of a specific Grid technology from the upper layers, thereby providing application independence and en-

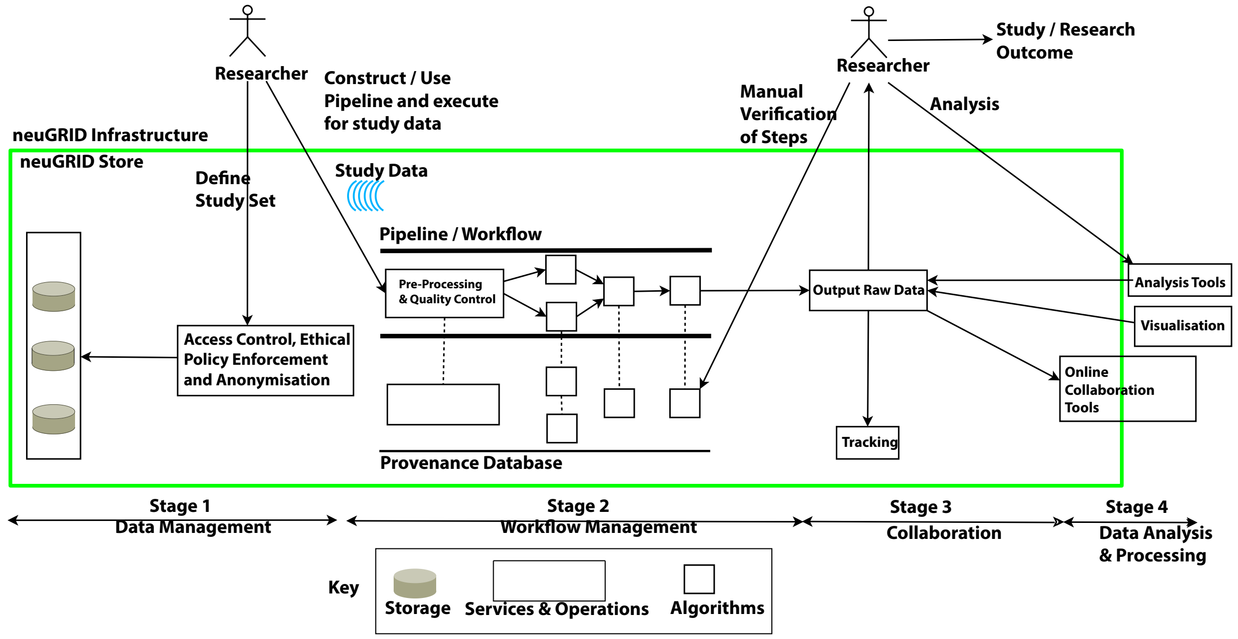


Figure 1: An end-to-end example of the neuGRID analysis environment

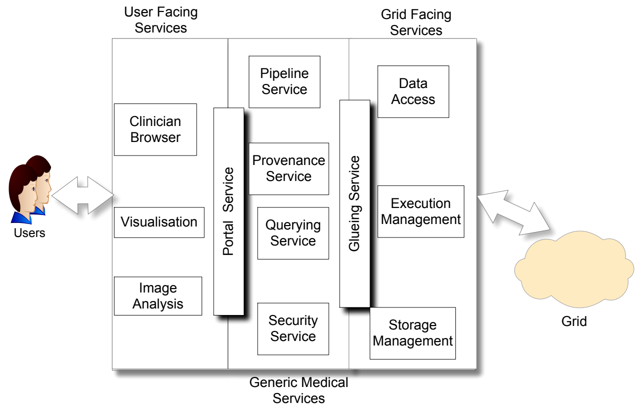


Figure 2: The neuGRID layered services architecture

abling the selection of 'fit-for-purpose' infrastructures. These services, as shown in Figure 3, glue a wide range of user applications to the available Grid platforms creating a foundation of cross-community and cross-platform services.

To justify the design philosophy, consider a service previously deployed using the Grid middleware gLite [31] and a user wanting to use a different middleware. In the current scenario, this transition from gLite to another middleware is not straightforward and requires changes in the software code, its recompilation and redeployment of the resulting applications/services on the new middleware. If a mechanism is developed whereby users are not concerned about the Grid fabric and functionality (with Grid details remaining abstracted), this should help not only to make the Grid use more attractive, but also to enable the users to concentrate on their research analyses. Applications and services developed for one platform can then be deployed on any other Grid middleware without significant user effort thereby enabling neuGRID to make use of existing resources that are running a range of middleware. This is one of the core design objectives of the neuGRID project where generalised services have been developed that can run on any middleware.

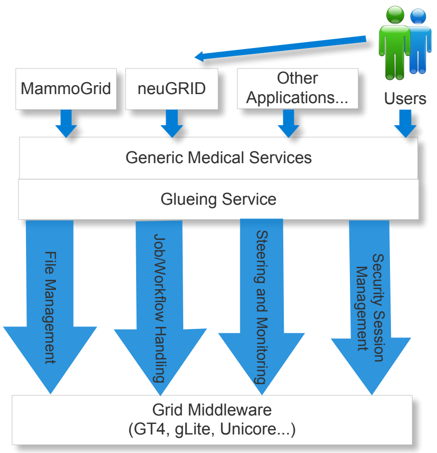


Figure 3: Middleware agnostic service

4. NeuGrid Services and Functionality

User requirements have been distilled into a set of services that help neuGRID provide an enabling analysis

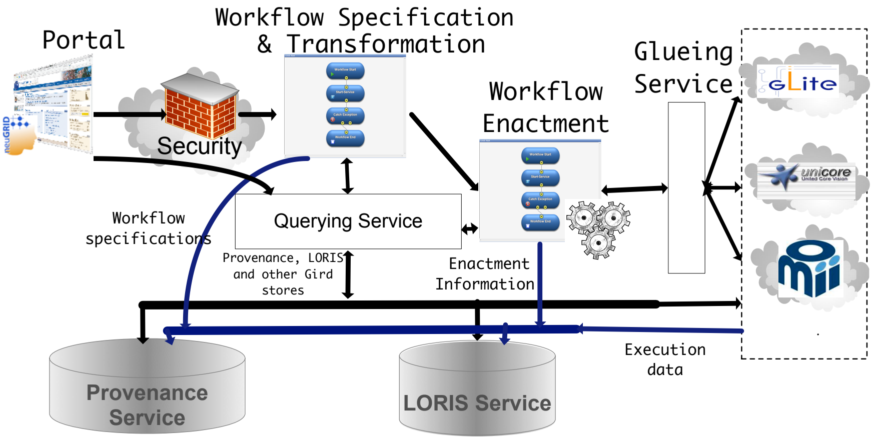


Figure 4: neuGRID Service Oriented Analysis Environment

environment to the neuroimaging community. The services cover each requirement depicted in the end-to-end requirements diagram shown in Figure 1. The services are self contained and loosely coupled entities that exist independently and can support the user analysis process. As stated earlier, the services are divided into three groups: i) User-facing Services ii) Analysis Services and iii) Grid facing Services. The user facing services include those that are accessed by a user for his/her day-to-day activities. They provide the interfaces that are necessary to enable the user to leverage the underlying neuGRID services to support her analyses. The user will interact with the system through the Portal Service which encapsulates and abstracts the complexity of the underlying neuGRID services and presents their functionality in an easy-to-use web-based portal. This service is supported by a single sign-on authentication service that enables the user to access the underlying neuGRID and Grid services without repeated authentication. A Security Service that is embedded in the Portal Service manages the authentication. This service is responsible for all the authentication, authorisation, access controls and policy enforcement issues. The Security Service, in association with an anonymisation service, is also responsible for the anonymisation and privacy protection of the datasets which will be studied during an analysis. The anonymisation service addresses any data format conversion issues (for example that from MINC [32] to DICOM[33] data formatting) as well.

The next set of services shown in the services architecture (see Figure 4) is designed to provide general-purpose analysis facilities. This set of services focuses on providing functionality for managing and executing workflows, querying and managing provenance information as well as facilitating information querying and retrieval from both image datasets and other clinical data. They have been implemented in such a way that a variety of applications and Grid middleware can be supported.

The first of the generic services is a workflow specification and transformation service called the Pipeline Service. Through this service users can specify their pipelines (or workflows, connected collections of algorithms). This service is designed to support all major neuroimaging workflow authoring environments (such as LONI Pipeline [34], Kepler [35] etc.) and with the help of the Glueing Service it can enable their enactment on a number of underlying infrastructures. The user can then submit the workflow for enactment through the PortalService and can retrieve the results in conjunction with the Provenance Service.

The Provenance Service, as depicted in Figure 4, can capture steps in an analysis workflow and store the workflow and any associated meta-data generated during its enactment. The Provenance Service allows users to query analysis information, to regenerate analysis workflows, to detect errors in past analyses and to validate analyses. After the execution of a workflow all the information that was initially provided and that which was generated during an analysis is stored in the Provenance store. This store can be queried by the user to verify results or improve and fine-tune pipelines and acts as a rich knowledge base of accumulated analysis steps and outcomes for users to consult.

The neuGRID Provenance Service is based on the orchestration characteristics of the CRISTAL [36] software. CRISTAL is a data and workflow tracking (i.e. provenance) system, which is being used to track the construction of large-scale experiments such as the CMS project at the CERN LHC [1]. It uses the so-called description-driven nature of the CRISTAL models [37] to act dynamically on process instances already running and can thus intervene in the actual process instances during execution.

The Querying Service provides methods to enable the efficient querying of heterogeneous data in neuGRID. The primary data sources in neuGRID comprise clinical data sets that include images and associated metadata. SOAP is used for communication between disparate services;

[1] CERN's Large Hadron Collider, http://lhc.web.cern.ch/lhc/

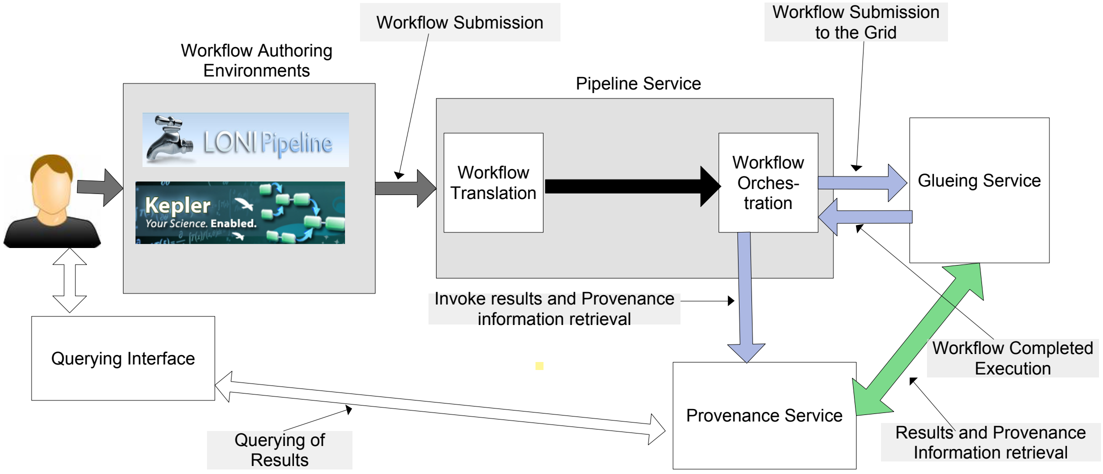


Figure 5: The Pipeline Service in neuGRID

the SOAP stack adopted for the project is the Apache Axis 2 Framework[2]. The Apache Tomcat container is used to host the web services. The basic single sign-on infrastructure used in the neuGRID project is the Central Authentication Server (CAS)which is a widely used Open Source SSO implementation in Java. To extend the single sign-on facility to the Grid environment, CAS has been integrated with the MyProxy Service [38]. The basic computational infrastructure in neuGRID is based on gLite.

The Pipeline Service (as shown in Figure 5) enables clients to perform various functions such as the submission of workflows, tracking progress and control functions to monitor workflows. The Pipeline Service supports multiple workflow specification formats and therefore a unified format is required for processing workflows that have been authored in different environments. For this purpose an object-oriented workflow API has been designed and implemented. The translation component implements an API which allows the translation of various workflow specification formats to a common format. When a workflow is submitted to the Pipeline Service, at first an appropriate translator for the format is instantiated dynamically. The format specific translator translates the workflow into a common object-oriented workflow model. This model is then forwarded to the Pipeline Service planner for workflow optimisation to enable efficient enactment.

Once a workflow has been enacted in the Grid (using a so-called Glueing Service), the Provenance Service coordinates the retrieval of all final data outcomes as well as intermediate data that was produced during the lifetime of the workflow. The Glueing Service hides the encapsulation of Grid middleware complexities from the neuGRID services. Using the Glueing Service neuGRID services can be deployed on various Grid middleware that includes gLite, Globus[39], Unicore [40] or any other SAGA [41] supported Grid middleware, thus promoting interoperability. It offers a mechanism to access any deployed Grid middleware through an easy-to-use service. It provides a service-based approach to shield users and applications from writing complex Grid specific functionality. The user requires a minimum set of Grid specific APIs and the rest of the functionalities are managed by the service. Users can, using the Glueing Service, *gridify* their applications without installing and maintaining too many Grid specific libraries.

In the next section, we describe the use of these services in enabling an intelligent and optimal analysis environment for the neuroimaging analysis.

5. Multi-Objective Scientific Workflow Optimisation

Another complexity dimension is the increasing number of tasks in a workflow. Due to the increase in the amount of data to be processed and the tasks in a workflow, the resources a single workflow consumes will also scale up. This, coupled with the fact that the nature of tasks greatly varies for scientific workflows, means there will be real scalability issues when it comes to optimising the workflows. Current state of the art optimisation techniques provide best effort optimisations, where the focus is either on low workflow turn-around time or maximising data efficiency [42]. Often one is achieved at the expense of the other. Existing optimisation techniques do not deal with multi-criteria optimisation which may be essential for various scientific domains.

e-Science workflows are compute and data intensive. Table 1 shows the compute and data characteristics of some of the workflows that we studied for evaluation. The CIVET neuroimaging workflow may appear to have a lower data footprint than other e-Science workflows, but it is data intensive because it has a large intermediate data usage cost (40 times more than the input [19])

[2] Apache Axis 2 Framework, http://ws.apache.org/axis2/

per brain scan. In order to achieve compute and data efficiency a multi-objective approach must be considered since both compute and data efficiency are mutually conflicting objectives. Consequently this is a suitable application for evolutionary multi-objective meta-heuristics. However the process of optimising a scientific workflow using an evolutionary meta-heuristic involves the evaluation of hundreds or thousands of candidate workflow plans. An extensive search may not be feasible as the evaluation of individual workflow plans may be expensive. Therefore, an approach is proposed which attempts to detect convergence based on the hyper volume and terminates the search. In the following paragraphs, we introduce the novel termination criteria and its application to the SPEA2 meta-heuristic.

Workflow	Workflow Turn-around Time	Data Footprint
CIVET Workflow	9.6 hr	403.75 MB
Cybershake Workflow	0.86 hr	318.047 GB
Epigenomics Workflow	16.2 hr	124.13 GB
LIGO Inspiral Workflow	0.85 hr	2.7 GB
SIPHT Workflow	2.07 hr	2.04 GB

Table 1: Characteristics of Scientific Workflows

5.1. *Hypervolume-Based Termination Algorithm*

The proposed meta-heuristic is shown in Algorithm 1. The meta-heuristic follows the pattern of a standard evolutionary algorithm. At the beginning of the algorithm the population of candidate workflow clustering solutions is initialised. After the initialisation the meta-heuristic proceeds into a loop which will be iterated until the termination conditions become valid. Within the loop initially the objective vector for each candidate solution is evaluated, followed by the calculation of the fitness of each candidate solution. The aim of this study is to optimise scientific workflows to achieve compute and data optimisation. Therefore, the objectives used in this approach are the workflow turn-around time and the data footprint, which can be used to indicate the compute and data efficiency of a workflow.

The fitness calculation procedure is followed by the update of the external archive which contains all the non-dominated solutions which have been discovered by meta-heuristic up to this point. The termination criterion procedure is invoked from the third generation of the evolutionary search, because the calculation procedure calculates the change in the hypervolume over the last three generations. The termination criterion is checked as soon as an archive of non-dominated solutions for the generation `t` has been determined. The termination criterion is detailed in Algorithm 2. The final steps within the loop prepare the next generation candidate solutions. At first candidate solutions for the next generation are selected using the binary tournament approach, followed by the of the application crossover operator on a subset of the population determined by crossover probability ρ_c. Finally, a mutation operator is applied on a subset of the population, as determined by the probability of mutation ρ_m.

Algorithm 1 SPEA2 meta-heuristic Algorithm Main Loop

```
Input :
$N$, Size of the population
$\overline{N}$, Size of the external set
$T$, Max number of generations
$\rho_c$, Crossover probability
$\rho_m$, Mutation probability
$\alpha$, Change threshold
$n$, Generation count threshold
Returns :
$\overline{P}$, Set of non-dominated solutions of size $\overline{N}$
procedure SPEA2($N, \overline{N}, T, \rho_c, \rho_m$)
    $t \leftarrow 0$
    $P_0^D \leftarrow initPopulation()$
    while $t \leq T \wedge termCondition \neq true$ do
        $P_t^O \leftarrow evalObjectives(P_t^D)$
        $[P^F, \overline{P}^F] \leftarrow evalFitness(P_t, \overline{P}_t)$
        $\overline{P}_{t+1} \leftarrow UpdateArchive(P_t, \overline{P}_t, \overline{N})$
        if $t \geq 3$ then
            $termCondition \leftarrow termCheck(t, \overline{P}_t, n, \alpha)$
        end if
        $P_{t+1} \leftarrow Selection(P_{t+1})$                ▷ Binary Tournament
        $P_{t+1} \leftarrow Crossover(P_{t+1}, \rho_c)$
        $P_{t+1} \leftarrow Mutation(P_{t+1}, \rho_m)$
        $t \leftarrow t + 1$
    end while
    return $\overline{P}_t$                                   ▷ Final Archive Set
end procedure
```

Algorithm 2 Termination Condition Handling Procedure

```
Input :
$n$, Generation count threshold
$I_\Delta$, Ordered tuple of change in $I_{HV}$ indicator values where $I_\Delta^i$ is the element $i$ of $I_\Delta$
$\alpha$, Change threshold for $I_\Delta$
$\overline{P}_t$ external set at generation $t$
Returns :
termCondition Sets the termination condition
procedure TERMCHECK($t, \overline{P}_t, n, \alpha$)
    if $|I_\Delta| = n$ then
        $(\frac{\sum_{i=1}^{|I_\Delta|} I_\Delta^i}{|I_\Delta|} \leq \alpha) \rightarrow (termCondition \leftarrow true) \wedge \neg(\Delta I_\Delta \leq \alpha) \rightarrow (I_\Delta \leftarrow I_\Delta - I_\Delta^0)$
    end if
    $I_\Delta \leftarrow I_\Delta + \Delta_{HV}(\overline{P}_t, \overline{P}_{t-1}, \overline{P}_{t-2})$
end procedure
```

For the termination procedure, the following approach is used in order to evaluate the hypervolume of the approximate pareto fronts of two generations. The approximate pareto front at generation t, is denoted as $\overline{P}_t$. It is

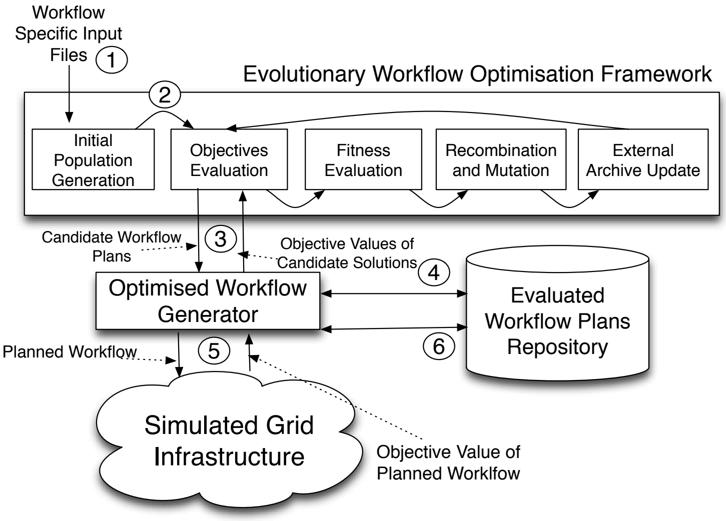

Workflow Specific Input Files
1
Evolutionary Workflow Optimisation Framework
2
Initial Population Generation
Objectives Evaluation
Fitness Evaluation
Recombination and Mutation
External Archive Update
Candidate Workflow Plans
3
Objective Values of Candidate Solutions
4
Optimised Workflow Generator
Evaluated Workflow Plans Repository
Planned Workflow
5
6
Simulated Grid Infrastructure
Objective Value of Planned Worklfow

Figure 6: Overview of the Simulation Architecture

assumed that $\overline{P}_t \geq \overline{P}_{t-1}$. In order to calculate the hypervolume of $\overline{P}_{t-1}$, the maximum values for all objectives in $\overline{P}_t$ are used as the nadir point. Therefore, formally:

$$\Delta_{HV}(\overline{P}_t, \overline{P}_{t-1}, \overline{P}_{t-2}) \leftarrow \frac{I_{HV}(\overline{P}_{t-1}, \overline{P}_t) - I_{HV}(\overline{P}_{t-2}, \overline{P}_t)}{I_{HV}(\overline{P}_{t-2}, \overline{P}_t)} \quad (1)$$

The procedure is detailed in Algorithm 2. A list of past hypervolume changes is maintained, denoted as I_Δ. The element i in the list is denoted as I^i_Δ. During a search, if the size of I_Δ equals the user specified generation count threshold n then the following conditional is invoked (line 3). If the average hypervolume change over the past n generations is below the user specified change threshold α then the termination condition, *termCondition* is set and the meta-heuristic terminates in this generation. However, if the average change in I_Δ is not below α, then the oldest element, I^0_Δ is removed from the list I_Δ. Afterwards, in line 5 the change in hypervolume, $\Delta(\overline{P}_t, \overline{P}_{t-2})$, over the last two generations is added to the I_Δ.

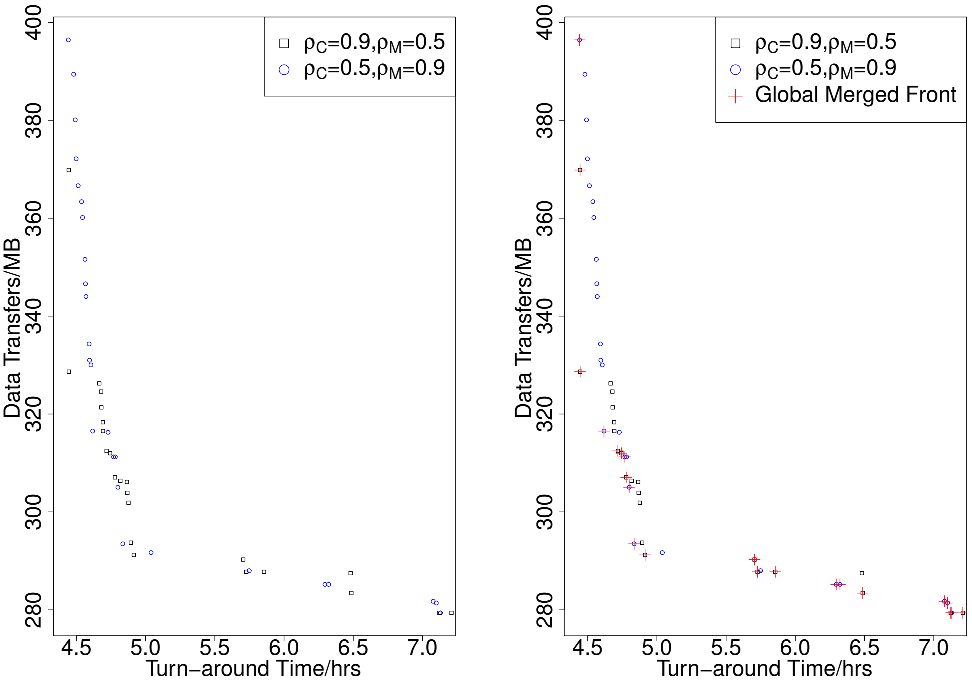
5.2. Experimental Setup for Optimising Scientific Workflows

The simulation framework (see Figure 6) used to implement the meta-heuristic and optimise scientific workflows uses JMetal [43] and SimGrid [44]. The framework consists of three components. The first component is the Evolutionary Workflow Optimisation Framework (EWOF) that implements the approach. The second component is the Optimised Workflow Generator (OWG), which is a component that receives a candidate solution plan from the EWOF and generates the optimised workflow instance. The optimised workflow instance is then executed in the simulated grid infrastructure (SGI).

At first (denoted by (1) in Figure 6), from the input data set, the EWOF uses the workflow graph structure to initialise the workflow specific decision vectors and generate the initial population. In (2) after the initial population has been generated the main loop of SPEA2 meta-heuristic commences. At first the objective values for each individual candidate solution are evaluated. In order to perform this evaluation the OWG is invoked, as depicted in (3). The OWG, at first determines if the clustering strategy, denoted by the decision vector, has already been evaluated (4). If the clustering strategy has been evaluated then the objective vector obtained in the previous evaluation is returned to the EWOF. Otherwise, the OWG uses the task runtime, data set size and the workflow graph files in addition to the decision vector to formulate an optimised workflow instance. The optimised workflow instance is executed on a simulated grid infrastructure, generated by the SGI component as shown in (5). The SGI component generates a simulated Grid infrastructure as per the requirements specified in the infrastructure platform file. The objective values obtained from the execution are returned to the EWOF via the OWG. The OWG stores the results in the Evaluated Workflow Plans Repository, as shown in (6). After all the objective vectors for each decision vector are obtained the main loop of the meta-heuristic continues.

SimGrid uses a mathematical model to calculate the runtime of a workflow, similarly, in jMetal no time-sensitive calculations are performed which could impact the accuracy of the results. In order to evaluate if the performance of the meta-heuristic is impacted by the balance of mutation and crossover, two sets of evaluations will be performed for the CIVET workflow. One evaluation will be performed where the balance of the crossover and mutation probabilities is towards the crossover operation. In this configuration the probabilities are ρ_m=0.5 and ρ_c=0.9.

The evolutionary search in this configuration will be more exploitation driven, therefore this configuration is



(a) Merged Non-dominated Fronts achieved for Individual Configurations

(b) Superimposed Global Non-dominated Front

Figure 7: Global Non-dominated Solutions obtained for the CIVET workflow across all experiments

termed as the exploitation focused configuration. The other configuration will be mutation centric. This configuration will have the probabilities ρ_m=0.9 and ρ_c=0.5. Due to the higher mutations in this configuration, the nature of the evolutionary search will be more exploration driven, therefore, this configuration is termed as the exploration focused configuration. These two configurations will be compared and in order to determine if the nature of the probabilities has any impact on the efficacy of the approach.

5.3. *Optimisation of the CIVET Neuroimaging Workflow*

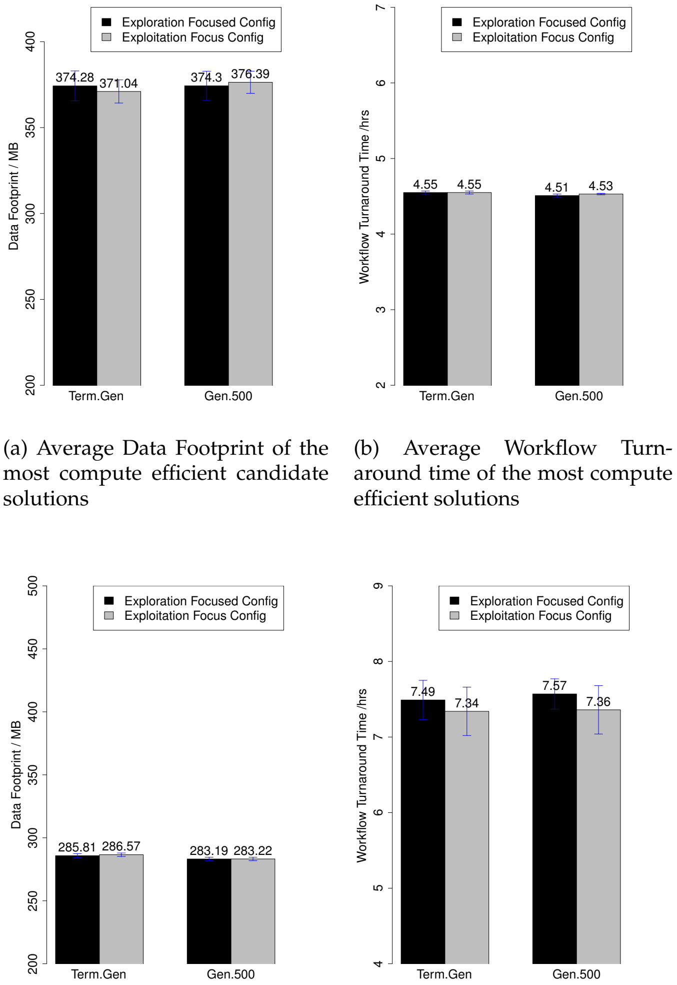
This section presents the results of the optimisation of a real world workflow in the meta-heuristic presented in this paper. The characteristics of the CIVET workflow have been presented in [19]. The workflow is both compute and data intensive. A feature of this workflow is that it is a data producing workflow that produces 4000% more data than it consumes. The standard turn-around time of the CIVET workflow in the simulated Grid infrastructure used for this study is 10:54 hours and the data footprint is 405.80MB. This performance will be the benchmark against which the optimised workflows will be evaluated. The results are presented as follows: at first, the global approximate pareto optimal front obtained will be presented. The global pareto optimal front is calculated by determining the non-dominated solutions discovered across all 20 runs of the meta-heuristic. Specifying appropriate values for the crossover and mutation probabilities is an open research issue and highly dependent on the nature of the problem. The probabilities for the genetic operations of crossover and mutation determine the balance of exploration and exploitation during an evolutionary search. Therefore, experiments were carried out where the balance between is in favour of exploration and experiments were carried out where the balance is in favour of exploitation. Consequently, two global pareto-fronts will be presented.

Figure 7a depicts the global pareto fronts achieved for the CIVET experiments. The pareto front achieved for the experiments where the balance between exploration and exploitation was set towards exploitation is depicted with a blue circle. While a black square represents the solutions that were obtained where the balance was towards exploration. We can observe that the geometric shape of the pareto front is largely convex. The solutions at the bottom right are the solutions that are the most data efficient, as they have the lowest data transfers during the lifetime of the workflow. The solutions at the top left are the solutions that are the most compute efficient, as they have the lowest workflow turn-around time. For instance, in Figure 7a the optimisation achieved is significant. The most compute efficient solution discovered has a workflow turn-around time of 4.44 hours, which is 57.87% more efficient than the unplanned CIVET instance. The most data efficient solution discovered had a data footprint of 279.38MB, which is 31.15% less than the standard unplanned workflow instance.

Several workflow plans were also discovered that have a better balance between compute and data efficiency. The most compute efficient workflow plan, for instance has a data footprint which is only 8.8% more data efficient than the standard unplanned workflow. While, the most data efficient workflow has a workflow turn-around time which is 32.44% less than the standard workflow turn-around time. We can observe that the pareto front achieved with more exploitation than exploration is more diverse. While the pareto front achieved for the other configuration is largely clustered between the data transfers values of 300 MB and 320 MB and on the other axis, around the workflow turnaround time of 4.8 hrs. However, as we can observe from Figure 7b the contribution of the non-dominated solutions from both configurations is largely equal. In fact, the quality of solutions discovered by the exploration centric configuration is better than the quality of solutions discovered by the exploitation centric configuration. The global pareto front, shown by a red triangle depicts the globally non-dominated solutions discovered in all 40 runs. In order to determine the average efficiency for a single run multi-objective indicators are used. These indicators provide a quantitative means of comparing the performance of various runs. All of these meta-heuristics compare against a reference pareto front. The reference pareto front used is the respective computed global pareto front that was presented in this section.

In order to judge how much performance, in terms of the actual workflow turn-around time and data footprint of the candidate workflow plans, was compromised we will compare the average most compute efficient solutions at the termination generation with the average most compute efficient solutions at the end of the evolutionary search. The same analysis will be performed for the average most data efficient solutions for each of the configurations. The results for the most compute efficient solutions are presented in Figure 8a and Figure 8b, which depict the average data footprint and average workflow turn-around time respectively. We can observe that the difference between the data footprints of the most compute efficient candidate workflow plans for both configurations are minor.

However the analysis of the evolutionary search dynamics suggests significant differences. For instance, the difference in terms of the compute footprint between the exploitation and exploration centric configuration is 4 MB. While the differences in the workflow turn-around times of the most compute efficient configurations is 0.01 hrs or 36 seconds. We can observe that when compared to the data footprint of the 500th generation of the evolutionary search to the termination generation pareto fronts, a difference of 0.02MB is noted for the exploitation centric configuration and a larger difference of 6.18 MB is noted. In terms of the comparison of the workflow turn-around times in Figure 8b, we note that the difference of the most compute efficient workflow solution at the termination generation of the exploration centric configuration and the most compute efficient solution at the end of the evolutionary search is 114 seconds, while the difference is only 36 seconds for the exploration centric configuration.



(a) Average Data Footprint of the most compute efficient candidate solutions

(b) Average Workflow Turn-around time of the most compute efficient solutions

(c) Average Data Footprint of the most data efficient candidate solutions

(d) Average Workflow Turn-around time of the most data efficient candidate solutions

Figure 8: Comparison of the data footprint and workflow turn-around time of the average most data efficient and most compute efficient candidate solutions at the termination generation and end of the evolutionary search

Figure 8c and Figure 8d depict the average most data efficient candidate workflow plans at the termination generation and the end of the evolutionary search. We can observe from Figure 8c that the differences between the most data efficient candidate solution for the exploitation-centric and exploration-centric configuration is 1 MB. While the average most data efficient solution in the final generation of the evolutionary search is more data efficient by 2.49 MB for the exploitation-centric configuration and the difference is marginally larger by 3.13 MB for the exploitation-centric configuration. In terms of the workflow turn-around time of the most data efficient workflow solution the differences are more significant. Although in terms of data footprint the difference is on average 1 MB, in terms of the workflow turn-around time the difference is of 8.4 minutes. Compared to the workflow turn-around time of the most data efficient candidate solutions, the difference for the exploitation focused configuration is 3 minutes and 1.3 minutes for the exploration-centric configuration.

From the analysis of the dynamics of the evolutionary search it may appear that the exploration centric configuration is more suitable for optimising the CIVET workflow. Significant differences existed in the dynamics of the search as the exploration centric configuration was able to achieve convergence with the configuration specific pareto-front more quickly and achieved a higher spread of solutions. However, the analysis of the actual workflow turnaround time and the data footprint of the candidate workflow solutions revealed that less significant differences existed between the configurations. In fact, on average the most compute efficient candidate solution for the exploration centric configuration was only 36 seconds more efficient than the most compute efficient candidate solution of the exploitation focused configuration of the meta-heuristic. The average most data efficient candidate solution discovered by the exploitation centric configuration had a lower footprint by 1 MB. Meta-heuristic quality indicators provide a quantitative value for an entire front. As depicted in Figure 7a the pareto front of the exploration centric configuration had a better spread and diversity therefore it had higher indicator values for the spread and the hypervolume indicator. While the pareto front of the exploitation centric configuration was not evenly spread and was clustered in certain locations which yielded a poorer indicator values.

The termination criterion was able to successfully detect search stagnation and terminate the search with a minor loss in the fitness of the candidate solutions. The worst compromised performance for the workflow turn-around time was 3 minutes (in the case of the average most data efficient candidate workflow for the exploitation focused configuration in Figure 8d). The worst compromised performance for the data footprint was 6.18 MB (in the case of the average most compute efficient candidate workflow for the exploration focused configuration in Figure 8c).

6. Conclusions

By taking Alzheimer's disease as an exemplar, the neuGRID project has developed a set of analysis services and an infrastructure which can enable the European neuroscience community to carry out research required for the study of degenerative brain diseases. Using the services in the neuGRID infrastructure, neuroscientists should be able to identify neurodegenerative disease markers through the analysis of 3D magnetic resonance brain images. The set of services has been designed and developed using the SOA paradigm and the services can thereby be reusable both across Grid-based neurological data and for wider medical analyses. The services have been developed following the SOA approach which provides the basis for extensibility and inter-operability.

We investigated the use of an evolutionary multi-objective meta-heuristic for optimising scientific workflows for distributed infrastructures. The study was motivated by the fact that e-Science workflows, which are the principle means of conducting distributed scientific analyses on distributed computing infrastructures, are growing in scale and complexity. This paper also explored the use of an adaptive hyper volume based termination criterion for detecting search stagnation. The search approach is relevant to real world applications, specifically those, such as neuroimaging workflows, where the evaluation of objectives may be expensive and an extensive evolutionary search may be infeasible. Due to its adaptive nature, the approach can be tuned by the decision maker to achieve the desired balance between finding more optimal solutions at the cost of increased evaluations or efficient solutions with an acceptable level of compromise on the performance. The results indicate that a multi-objective approach is feasible for the optimisation of scientific workflows and significant performance gains can be achieved by the application of this approach.

Acknowledgement

The authors would like to thank Professor Nik Bessis, University of Derby, for his valuable contributions to the article.

References

[1] F. Pop, C. Dobre, V. Cristea, Decentralized dynamic resource allocation for workflows in grid environments, in: Proceedings of the 2008 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC '08, IEEE Computer Society, Washington, DC, USA, 2008, pp. 557–563. doi:10.1109/SYNASC.2008.15.

[2] A. Costan, F. Pop, C. Dobre, V. Cristea, A workflow management platform for scientific applications in grid environments, in: Proceedings of the 2010 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC '10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 261–268. doi:10.1109/SYNASC.2010.82.

[3] G. B. Frisoni, P. h. Scheltens, S. Galluzzi, F. M. Nobili, N. C. Fox, P. H. Robert, H. Soininen, L.-O. Wahlund, G. Waldemar, E. Salmon, Neuroimaging tools to rate regional atrophy, subcortical cerebrovascular disease, and regional cerebral blood flow and metabolism: consensus paper of the eadc, Journal of Neurology, Neurosurgery & Psychiatry 74 (10) (2003) 1371–1381. doi:10.1136/jnnp.74.10.1371.

[4] M. López, J. Ramírez, J. Górriz, I. Álvarez, D. Salas-Gonzalez, F. Segovia, R. Chaves, P. Padilla, M. Gómez-Río, Principal component analysis-based techniques and supervised classification schemes for the early detection of alzheimer's disease, Neurocomputing 74 (8) (2011) 1260–1271.

[5] G. B. Frisoni, N. C. Fox, C. R. Jack, P. Scheltens, P. M. Thompson, The clinical use of structural MRI in Alzheimer disease, Nature Reviews Neurology 6 (2) (2010) 67–77. doi:10.1038/nrneurol.2009.215.

[6] A. Caroli, G. B. Frisoni, Quantitative evaluation of Alzheimer's disease, Expert Rev Med Devices 6 (5) (2009) 569–588.

[7] S. J. Teipel, T. Meindl, L. Grinberg, H. Heinsen, H. Hampel, Novel MRI techniques in the assessment of dementia, Eur. J. Nucl. Med. Mol. Imaging 35 Suppl 1 (2008) 58–69.

[8] J. Ashburner, J. G. Csernansky, C. Davatzikos, N. C. Fox, G. B. Frisoni, P. M. Thompson, Computer-assisted imaging to assess brain structure in healthy and diseased brains, Lancet Neurol 2 (2) (2003) 79–88.

[9] A. Otte, U. Halsband, Brain imaging tools in neurosciences, Journal of Physiology-Paris 99 (4â"6) (2006) 281 – 292. doi:10.1016/j.jphysparis.2006.03.011.

[10] G. B. Frisoni, Structural imaging in the clinical diagnosis of Alzheimer's disease: problems and tools, J. Neurol. Neurosurg. Psychiatr. 70 (6) (2001) 711–718.

[11] C. R. Jack Jr., M. A. Bernstein, N. Fox, P. Thompson, The alzheimer's disease neuroimaging initiative (adni): Mri methods, Journal of Magnetic Resonance Imaging 27 (4) (2008) 685–91.

[12] L. Ogiela, M. Ogiela, Cognitive Techniques in Visual Data Interpretation, Vol. 228 of Studies in Computational Intelligence, Springer-Verlag, Berlin, Heidelberg, 2009.

[13] F. Segovia, J. M. Górriz, J. Ramírez, D. Salas-Gonzalez, I. Álvarez, M. López, R. Chaves, A comparative study of feature extraction methods for the diagnosis of alzheimer's disease using the adni database, Neurocomputing 75 (1) (2012) 64 – 71. doi:10.1016/j.neucom.2011.03.050.

[14] J. Montagnat, A. Gaignard, D. Lingrand, J. Balderrama, P. Collet, P. Lahire, Neurolog: a community-driven middleware design, Stud Health Technol Inform 138 (2008) 49–58.

[15] J. M. Wardlaw, P. Bath, P. Sandercock, D. Perry, The neurogrid stroke exemplar clinical trial protocol, International Journal of Stroke 2 (1) (2007) 63–69.

[16] A. P. Zijdenbos, R. Forghani, A. C. Evans, Automatic "pipeline" analysis of 3-D MRI data for clinical trials: application to multiple sclerosis, IEEE Trans Med Imaging 21 (10) (2002) 1280–1291.

[17] S. Balla-Arabé, X. Gao, Image multi-thresholding by combining the lattice Boltzmann model and a localized level set algorithm, Neurocomputing 93 (2012) 106–114.

[18] Y. Ad-Dab'bagh, D. Einarson, O. Lyttelton, J.-S. Muehlboeck, K. Mok, O. Ivanov, R. D. Vincent, C. Lepage, J. Lerch, E. Fombonne, A. C. Evans, The civet image-processing environment: A fully automated comprehensive pipeline for anatomical neuroimaging research, 12th Annual Meeting of the Organization for Human Brain Mapping (OHBM).

[19] I. Habib, A. Anjum, P. Bloodsworth, R. McClatchey, Neuroimaging analysis using grid aware planning and optimisation techniques, in: E-Science Workshops, 2009 5th IEEE International Conference on, 2009, pp. 102 –109. doi:10.1109/ESCIW.2009.5407988.

[20] E.-G. Talbi, Metaheuristics: From Design to Implementation, Wiley Publishing, 2009.

[21] S. Huang, Y. Zhu, NSGA-II based grid task scheduling with multi-QoS constraint, in: Proceedings of the 2009 Third International Conference on Genetic and Evolutionary Computing, WGEC '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 306–308. doi:10.1109/WGEC.2009.211.

[22] P. Liu, R.and Zhang, L. Jiao, Y. Li, Supervised immune clonal evolutionary classification algorithm for high-dimensional data, Neurocomputing Available online 6 June 2012.

[23] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, in: M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. Merelo, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature PPSN VI, Vol. 1917 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2000, pp. 849–858, 10.1007/3-540-45356-383.

[24] E. Zitzler, M. Laumanns, L. Thiele, Spea2: Improving the strength pareto evolutionary algorithm, Tech. rep., ETH Zurich (2001).

[25] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: in Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII, Springer, 2004, pp. 832–842.

[26] J. M. Bader, Hypervolume-Based Search for Multiobjective Optimization: Theory and Methods, CreateSpace, Paramount, CA, 2010.

[27] F. Estrella, T. Hauer, R. McClatchey, M. Odeh, D. Rogulin, T. Solomonides, Experiences of engineering grid-based medical software, International Journal of Medical Informatics 76 (8) (2007) 621 – 632. doi:10.1016/j.ijmedinf.2006.05.005.

[28] A. Anjum, R. McClatchey, N. Bessis, A service oriented analysis environment for neuroimaging studies, in: Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2012, Palermo, 2012, pp. 29 –36.

[29] T. Erl, Service-oriented architecture, Prentice Hall, 2004.

[30] S. Miles, P. Groth, M. Branco, L. Moreau, The requirements of using provenance in e-science experiments, Journal of Grid Computing 5 (2007) 1–25. doi:10.1007/s10723-006-9055-3.

[31] A. Kretsis, P. Kokkinos, E. Varvarigos, Developing scheduling policies in glite middleware, in: Cluster Computing and the Grid, 2009. CCGRID '09. 9th IEEE/ACM International Symposium on, 2009, pp. 20 –27. doi:10.1109/CCGRID.2009.54.

[32] R. D. Vincent, A. Janke, J. G. Sled, P. Neelin, A. C. Evans, Introduction minc 2.0: A modality independent format for multidimensional medical images, in: Proc. 10th Annual Meeting Organization for Human Brain Mapping, 2004.

[33] P. Mildenberger, M. Eichelberg, E. Martin, Introduction to the DICOM standard, Eur Radiol 12 (4) (2002) 920–927.

[34] D. Rex, The loni pipeline processing environment, NeuroImage 19 (3) (2003) 1033–1048.

[35] I. Altintas, O. Barney, Z. Cheng, T. Critchlow, B. Ludaescher, S. Parker, A. Shoshani, M. Vouk, Accelerating the scientific exploration process with scientific workflows, Journal of Physics: Conference Series 46 (1) (2006) 468.

[36] R. McClatchey, J.-M. Goff, N. Baker, W. Harris, Z. Kovacs, A distributed workflow and product data management application for the construction of large scale scientific apparatus, in: A. Doüaß, L. Kalinichenko, M. T. ñzsu, A. Sheth (Eds.), Workflow Management Systems and Interoperability, Vol. 164 of NATO ASI Series, Springer Berlin Heidelberg, 1998, pp. 18–34.

[37] F. Estrella, Z. Kovacs, J.-M. Le Goff, R. McClatchey, T. Solomonides, N. Toth, Pattern reification as the basis for description-driven systems, Software and Systems Modeling 2 (2003) 108–119, 10.1007/s10270-003-0023-0.

[38] J. Novotny, S. Tuecke, V. Welch, An online credential repository for the grid: Myproxy, in: Proceedings of the 10th IEEE International Symposium on High Performance Distributed Computing, HPDC '01, IEEE Computer Society, Washington, DC, USA, 2001, pp. 104–.

[39] I. Foster, Globus toolkit version 4: Software for service-oriented systems, in: H. Jin, D. Reed, W. Jiang (Eds.), Network and Parallel Computing, Vol. 3779 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2005, pp. 2–13. doi:10.1007/11577188_2.

[40] M. Romberg, The unicore architecture: seamless access to distributed resources, in: High Performance Distributed Computing, 1999. Proceedings. The Eighth International Symposium on, 1999, pp. 287 –293. doi:10.1109/HPDC.1999.805308.

[41] T. Goodale, S. Jha, H. Kaiser, T. Kielmann, P. Kleijer, Saga: A simple api for grid applications. high-level application programming on the grid, Computational Methods in Science and Technology.

[42] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, K. Blackburn, A. Lazzarini, A. Arbree, R. Cavanaugh, S. Koranda, Mapping abstract complex workflows onto grid environments, Journal of Grid Computing 1 (1) (2003) 25–39. doi:10.1023/A:1024000426962.

[43] J. J. Durillo, A. J. Nebro, E. Alba, The jmetal framework for multi-objective optimization: Design and architecture, in: CEC 2010, Vol. 5467 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Barcelona, Spain, 2010, pp. 4138–4325.

[44] H. Casanova, A. Legrand, M. Quinson, SimGrid: a Generic Framework for Large-Scale Distributed Experiments, in: 10th IEEE International Conference on Computer Modeling and Simulation, 2008.