CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. (2008)
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cpe.1323

A Grid-enabled problem-solving environment for advanced reservoir uncertainty analysis

Zhou Lei[1,*,†], Gabrielle Allen[1], Promita Chakraborty[1],
Dayong Huang[1], John Lewis[1], Xin Li[2] and Christopher D. White[2]

[1]*Center for Computation & Technology, Louisiana State University, Baton Rouge, LA 70803, U.S.A.*
[2]*Department of Petroleum Engineering, Louisiana State University, Baton Rouge, LA 70803, U.S.A.*

SUMMARY

Uncertainty analysis is critical for conducting reservoir performance prediction. However, it is challenging because it relies on (1) massive modeling-related, geographically distributed, terabyte, or even petabyte scale data sets (geoscience and engineering data), (2) needs to rapidly perform hundreds or thousands of flow simulations, being identical runs with different models calculating the impacts of various uncertainty factors, (3) an integrated, secure, and easy-to-use problem-solving toolkit to assist uncertainty analysis. We leverage Grid computing technologies to address these challenges. We design and implement an integrated problem-solving environment *ResGrid* to effectively improve reservoir uncertainty analysis. The ResGrid consists of data management, execution management, and a Grid portal. Data Grid tools, such as metadata, replica, and transfer services, are used to meet massive size and geographically distributed characteristics of data sets. Workflow, task farming, and resource allocation are used to support large-scale computation. A Grid portal integrates the data management and the computation solution into a unified easy-to-use interface, enabling reservoir engineers to specify uncertainty factors of interest and perform large-scale reservoir studies through a web browser. The ResGrid has been used in petroleum engineering. Copyright © 2008 John Wiley & Sons, Ltd.

Received 16 July 2007; Revised 29 January 2008; Accepted 10 February 2008

KEY WORDS: problem-solving environment; reservoir uncertainty analysis; Grid computing; task farming; load balancing

*Correspondence to: Zhou Lei, Center for Computation & Technology, Louisiana State University, 350 Johnston Hall, Baton Rouge, LA 70803, U.S.A.
†E-mail: zlei@cct.lsu.edu

Contract/grant sponsor: U.S. Department of Energy; contract/grant number: DE-FG02-04ER46136
Contract/grant sponsor: Board of Regents, State of Louisiana; contract/grant number: DOE/LEQSF(2004-07)

1. INTRODUCTION

In order to help decision making, reservoir studies are adopted to obtain reservoir performance prediction under different development and production scenarios. However, accurately predicting reservoir performance is difficult due to different sources of uncertainty, each of which may seriously impact reservoir performance.

Reservoir uncertainty analysis through simulation [1] is viewed as an effective tool to improve performance prediction. The steps to conduct reservoir uncertainty analysis are creating a reservoir base model, generating loadable models by integrating uncertainty factors with the base model, executing flow simulations with different loadable models, and then obtaining different impacts on reservoir performance with different uncertainty factors through a comparison of simulation results.

From the computation perspective, the major challenges in reservoir uncertainty analysis are

(1) The need for data set management. Data sets involved in the generation of the base model include geological & geophysical (G&G) data and well-logging data with large scale (terabyte, even petabyte) and geographical distribution characteristics.
(2) The need to rapidly and repeatedly perform hundreds or thousands of time-consuming simulations with different loadable models to identify and/or quantify the impacts of uncertainty factors.
(3) The lack of an integrated, secure, and easy-to-use problem-solving toolkit to assist uncertainty analysis. Reservoir engineers have to manually handle all stages of uncertainty analysis, including resource discovery, staging, job submission and monitoring, result retrieval, and post-processing, without unified security support.

Grid computing is an active research area to provide solutions for large-scale computing-intensive and data-intensive scientific and engineering applications. It takes advantage of worldwide network computation facilities to support coordinated resource sharing within a distributed, dynamic, and heterogeneous virtual organization. There have been many efforts to research and develop various Grid computing middleware. Technologies such as Grid security infrastructure (GSI) [2], Globus toolkit [3], Condor-G [4], GridSphere [5], and simple application programming interface (API) for Grid application (SAGA) [6] have been (or are becoming) the *de facto* standards. GSI is a specification for secure and authenticated communication in the Grid computing environment. Globus toolkit is an open-source toolkit for building Grids, which integrates or implements GSI, remote resource allocation, data location service, information infrastructure, etc. Condor-G provides a job submission queue system across a Grid. GridSphere is an open-source portal framework, offering web-based user management, access control, and data & execution integration. SAGA is the standardization effort for Grid application programming abstraction, pursued through the Global Grid Forum Research Group on Programming Models.

Our research focuses on the design and development of an integrated problem-solving environment (PSE)—*ResGrid* for advanced reservoir uncertainty analysis, leveraging state-of-the-art Grid computing technologies and contemporary flow simulation software. We have implemented a data management tool, task-farming-based computation management software, and an easy-to-use Grid portal in the ResGrid. Existing data Grid tools, such as metadata, replica, and data transfer services, are integrated to meet massive size and geographically distributed characteristics of reservoir study

DOI: 10.1002/cpe

data sets. Load balancing is adopted to select resources. A task-farming methodology dispatches jobs with different models and configurations on various selected computing resources. Finally, a Grid portal integrates data and computation management services into a unified user interface, which enables reservoir engineers to specify uncertainty factors of interest and perform large-scale reservoir uncertainty analysis through a web browser. Some results have been published in [7–9].

This paper outlines our efforts in terms of problem understanding, Grid-enabled data management and execution management, Grid portal, and real applications. It is organized as follows. Section 2 explains what reservoir uncertainty analysis is and the challenging issues involved therein. Section 3 demonstrates the Grid-enabled solutions to advanced uncertainty analysis from data management, execution management, and portal perspectives. A use case study is described in Section 4. Concluding remarks are made in Section 6.

2. RESERVOIR UNCERTAINTY ANALYSIS

2.1. Overview

Prior to investments, petroleum exploration and production engineers must be able to identify reservoir characteristics and uncertainty factors, and then assess and quantify the impacts of these factors. Uncertainty analysis plays a key role in reservoir assessment and performance prediction. Experimental design and response surface methodology are the fundamental mechanisms for providing inference with a number of reservoir simulations, as well as quantifying the influences of uncertainty factors on production and economic forecasts [10,11]. The concepts of importance are explained as follows to better understand uncertainty analysis.

Factor: A factor is defined as an object that might affect the performance of a reservoir. Factors can be classified by the nature of change. *Controllable* factors can be varied by process implementers (e.g. well location specified by development programs). *Observable* factors can be measured relatively accurately, but cannot be controlled (e.g. the depth to the top of a structure). *Uncertain* factors can be neither measured accurately nor controlled (e.g. permeability far from wells).

Response: Decisions are based on *responses* obtained by simulation. Reservoir studies examine responses that affect project value (e.g. water breakthrough time, peak oil rate, and cumulative oil recovery). *Controllable* factors are selected to maximize project value subject to *observable* factors and over the ranges of *uncertain* factors.

Response surface model: A response surface model is an empirical fit of responses. The responses are usually measured or computed at factor combinations specified by an experimental design (explained in Section 2.2). The model is usually a polynomial fit with a linear regression. Each regressor in the polynomial is a function of one or more factors. The coefficients in the polynomial are factor effects and interactions.

Reservoir model: A reservoir model has many large, interacting data elements that contain information describing the characteristics of a reservoir. Many features of models affect study design and execution, such as interdependence of data elements, permeability array assignment and saturation table selection, geostatistical models for permeability, etc.

Flow numerical simulation: Numerical flow simulation (i.e. reservoir simulation) [12] is the main approach to characterizing a reservoir in planning and evaluation of sequential development phases.

DOI: 10.1002/cpe

A reservoir can be represented by a mathematical model by applying the mass conservative law (i.e. Darcy's law) in a differential equation

$$\nabla \cdot (\rho_m K \lambda_m \nabla P_m) - q_m = \frac{\partial(\phi \rho_m S_m)}{\partial t}$$

where m represents oil, water, or gas; ρ_m the density; K the permeability; λ_m the mobility; P_m the pressure; q_m the production rate; ϕ the porosity; S_m the saturation; and t the time. To obtain an analytical solution to a reservoir, numerical simulation is required.

2.2. Workflow and methodologies

There are four major steps to conduct an uncertainty analysis [13,14]: (1) seismic inversion; (2) reservoir modeling; (3) flow numerical simulation; and (4) post-processing. Figure 1 shows the workflow.

The workflow starts with sparse spike inversion and conventional correlated wavelet extraction. Sparse spike inversion gives a preliminary estimate of net rock volume and fluid probabilities. It helps in building a layer-based model framework. To obtain what uncertainty of seismic data is, wavelet derivation needs to produce an estimate of seismic noise, that is, the part of seismic data which does not correlate with the synthetic of well log. Then, probabilistic wavelet derivation based on Bayesian concepts is used to predict the noise level. Given the wavelet with uncertainty, trace-based probabilistic model-based inversion is performed. Large-scale data sets, such as G&G data and well-logging data, are involved in this phase.

The second phase is reservoir modeling. This step needs to deal with massaging seismic inversion results into simulation Grid, enforcing spatial correlations, and decorating models with stratigraphic

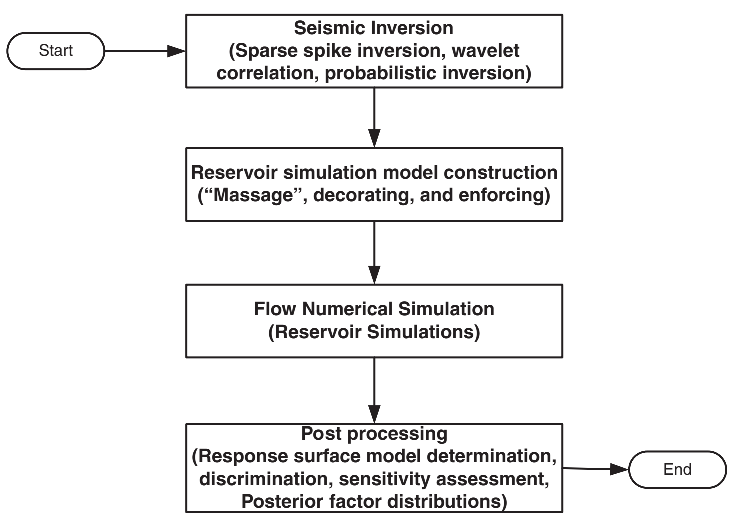

Figure 1. Reservoir uncertainty analysis workflow. There are four major steps involved: (1) seismic inversion; (2) reservoir modeling; (3) flow numerical simulation; and (4) post-processing.

DOI: 10.1002/cpe

subseismic structure to allow reservoir simulation to assess flow effects of subseismic heterogeneity. The output of model-based inversion is an ensemble of models for each trace on the seismic Grid.

Flow simulation is adopted to characterize a reservoir described by a simulation model. A typical flow simulation consists of the following steps: (1) geostatistical realizations are generated to sample the uncertainty of geological parameters; (2) reservoir engineers combine geology and fluid and flow parameters, along with the well locations and other engineering factors to constitute a base model; (3) this model is simulated to obtain production profiles and recovery factors for a chosen recovery process; (4) economic performance indicators, such as return on investment and net present value, are calculated. To conduct uncertainty analysis, loadable models need to be generated, each of which is the product of the base model and a combination of considered uncertainty factors with different levels. One loadable model needs one execution of flow simulation.

To perform post-processing, the first thing is to determine a response surface model. Response surfaces for stochastic, kriged, and uniform permeability fields are usually computed with stepwise regression. The process activities include discriminating models, computing sensitivities, measuring posterior factor distributions, etc. With Monte Carlo simulation, the differences in response surfaces among various loadable models can be analyzed with the x^2-likelihood test.

As reservoir uncertainty analysis is a time-consuming process that integrates extensive geoscience and engineering data with complex process models to simulate and examine reservoir behaviors, experimental design [15] is adopted to optimize uncertainty analysis. It is an efficient tool in diverse engineering and scientific areas, such as aerospace and electronics, for analysis and optimization of complex and nonlinear systems. The basic idea of experimental design is to identify the optimum settings for different factors that affect analysis process. In reservoir uncertainty analysis, an experimental design framework selects relevant models, records factor settings for models, creates data files, controls execution, gathers summary data, and creates response models.

2.3. Challenges

There are many factors involved in reservoir performance prediction. The more the factors are considered, the more accurate the prediction is. However, it is challenging to conduct a large-scale uncertainty analysis. A single high-performance computing facility cannot satisfy the requirements of massive reservoir simulation runs. Large-scale data storage is required for both modeling-related data and simulation results.

Let us take an example. Uncertainty analysis is applied to a single-well water-drive gas reservoir with a radial geometry [16]. Fourteen factors are considered: 11 geologic factors and three engineering factors. The simulation runs for a full factorial design would be $4^6 \times 3^8 = 26\,873\,856$ if there are six factors each of which has four levels and eight factors each of which has three levels. Conservatively assuming that a single simulation run with a Grid block size of 20 m feet for a middle-scale reservoir consumes 6 min of CPU time, the total execution time would be 2 687 386 h (or over 100 days on a 1024 processor cluster). Meanwhile, large-scale data are involved in such a study. G&G data set and well-logging data set, with a size of terabytes and even petabytes, are geographically distributed. The resulting data set of a single simulation depends on the configuration. An average size reaches up to 50 M or so. Massive simulations lead to storage needs that cannot be easily accommodated with a typical storage resource.

DOI: 10.1002/cpe

The lack of an integrated solving environment is another issue that hinders reservoir uncertainty analysis studies. An engineer needs to manually handle all stages of the process, including resource discovery, staging, visualization, result retrieval, and quantification analysis. There is no integrated and ease-to-use environment for use.

Security concerns make it difficult to form effective collaborations among reservoir study communities as exploration and production data sets are very sensitive with potential commercial benefits.

Constraint by the lack of easy and effective ways of accessing multiple high-end computing resources, reservoir engineers are often forced to minimize the number of uncertainty factors and factor levels when they conduct uncertainty analysis process, which may often cause the loss of correct conclusions.

3. A GRID-ENABLED PSE FOR ADVANCED RESERVOIR UNCERTAINTY ANALYSIS

We leverage Grid computing technologies to address the issues mentioned above. Our effort is to provide a PSE, *ResGrid*, to support advanced reservoir uncertainty analysis. It allows a user conveniently to collect G&G data and well-logging data, specify the uncertainty parameter space, invoke numerical reservoir simulations, and analyze and visualize simulation results in a Grid environment. All these operations are completed via a Grid portal, interacting with various Grid services and resources. GSI ensures high security among all the processes and data transfer.

3.1. Usage scenario

The ResGrid usage scenario is illustrated in Figure 2. Typically, there are the following steps:

1. A user logs into the ResGrid portal and retrieves a GSI certificate from a proxy server. The certificate authenticates and authorizes the user to access the Grid resources and achieve secure data transfer.
2. The user specifies the uncertainty factor parameters with levels and the size of the reservoir Grid block, which are used for reservoir model construction and result analysis.
3. By clicking on the 'Job Submission' button, the user invokes the execution of the ResGrid services.
4. The reservoir modeling service triggers a data replica tool and analyzes the uncertainty factor parameter space specified by the user in Step 2.
5. The modeling service constructs reservoir models and starts the resource brokering service.
6. The resource brokering service captures the resource information from the external information services provided by the Grid, selects the proper resource for each single simulation run, and then calls the massive simulation execution service.
7. The simulation executions are invoked on all the selected resources across the Grid.
8. Once all the simulation runs have been completed, the result analysis service is activated to analyze the simulation results.
9. The visualization service visualizes the simulation results.
10. The user views the results on the ResGrid portal.

DOI: 10.1002/cpe

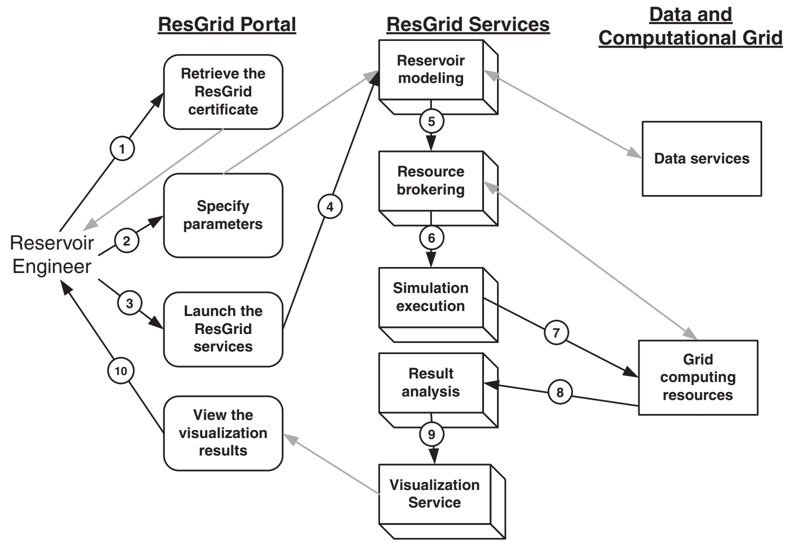

Figure 2. Usage scenario. What a reservoir engineer needs to do is only to interact with the Grid portal. The backend ResGrid services handle security issue, data acquisition, resource management complexity, result analysis and visualization, etc.

What a reservoir engineer needs to do in this scenario is only to interact with the Grid portal. The ResGrid services handle security issue, data acquisition, resource management complexity, result analysis and visualization, etc.

3.2. Data management

Data management acquires distributed modeling-related data, constructs reservoir models, and archives simulation results. Figure 3 shows the structure of the ResGrid data management, in which data manipulation and model generation play key roles.

A data replica tool is designed for data acquisition for various data sources. The Grid application toolkit (GAT) [17] is used to hide Grid middleware heterogeneity. Meanwhile, a data format converter is employed to represent complex data objects. A model generator takes investigated uncertainty factors and uniform data representation from the data format converter to create loadable models.

3.2.1. Data manipulation

Modeling-related data including G&G data, exploration well data, and production well data are geographically distributed with the size of terabytes, even petabytes. The seismic data size, for example, would be up to 200 terabytes to record 1 h flow motion by a modern seismographic system, given a 20 km reservoir with 5 m × 5 m × 5 m resolution. The seismic data and engineering data are generated at different locations.

DOI: 10.1002/cpe

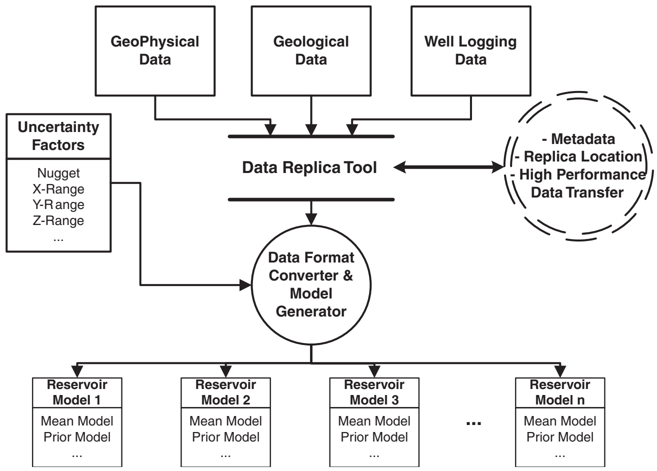

Figure 3. Data management. Data replica and model generation play key roles in this structure.

The data replica tool is built on GAT, which provides a single and easy-to-use Grid API, hiding the complexity and diversity of the actual Grid resources and their middleware layers. GAT is composed of two parts, a GAT engine and a set of GAT adaptors. The GAT engine exposes a set of APIs to an application programmer and the GAT adaptors are a set of lightweight software elements that provide access to specific Grid capabilities. Taking advantage of GAT, the data replica tool demonstrates high flexibility and portability across a Grid. The implementation of the replica tool includes data-related adaptors development and high-level logic development.

There are three modules in this tool: metadata service, replica location service, and high-performance data transfer service. Correspondingly, three functional categories of GAT adaptors are involved in metadata adaptors, logical file adaptors, and physical file adaptors, which provide interactions with metadata service, replica location service, and high-performance data transfer service, respectively. Given the description information (i.e. metadata) describing the required data, the metadata service retrieves logical file names, the replica location service locates the physical files that map to the logical filenames, and then these physical files are relocated via high-performance data transfer.

Consider a case of how to archive simulation results by this tool. A user intends to obtain simulation results for impact investigation on two levels of x range. What needs to be done is to specify the values of x range and archiving data category (i.e. simulation results). Then, automatically, the tool identifies where the related simulations were conducted (data location), chooses appropriate transfer protocol(s), and updates operation progress.

3.2.2. Model generation

With the help of the data replica tool, a base model is generated by extracting modeling-related data. Uncertainty factors and factor levels are provided and parameter space is made. Based on the base model and the parameter space, massive reservoir loadable models are constructed, each

DOI: 10.1002/cpe

of which is associated with a combination of uncertainty factors and factor levels. The number of models depends on the parameter space. Typically, it is up to thousands. These models are the inputs of massive reservoir simulation executions.

Simulation models need to be dispatched across a Grid for execution and load balancing are employed for this process. First, the information on every participating resource is collected. Second, the number of models assigned to a single resource is calculated. Finally, models are transferred to resources. There is a table to record the map between models and resources. More details are discussed in the following section.

3.3. Execution management

Execution management is incharge of simulation workflow, resource allocation, and simulation invocation.

3.3.1. Design considerations

We identify the characteristics of reservoir uncertainty analysis and investigate it from the computation perspective.

- A number of time-consuming simulation executions.
- Sequential job for each execution, i.e. one CPU for a single run.
- No communication among executions.
- Large-scale simulation output data sets.

We adopt a task-farming-based framework to meet these criteria. It dispatches a number of simulation jobs on different computing resources with security assurance. After the executions have been done, the results are collected for post-processing. We also address concerns on scalability, load balancing, and heterogeneity to design such a Grid-enabled framework. In addition, the workflow of each simulation is designed to be customized flexibly by a user without the impact of the whole execution management.

3.3.2. Architecture

The architecture of execution management is illustrated in Figure 4. Task farming is used as the framework that takes reservoir models as inputs, checks a resource broker for resource allocation, and invokes massive executions. The simulation workflow, i.e. computation model, integrates geostatistics algorithms with flow simulation. Data conversion is developed between geostatistics algorithm and simulation execution. The definition of such a computational workflow is open to allow a user to customize his/her own computational model with no change in other components.

This architecture consists of four modules: resource brokering module, staging in/out module, invocation module, and status monitoring module.

The resource brokering module is employed to manage resources for load sharing across a Grid. It accesses external information services and extracts resource information of interest into a list.

DOI: 10.1002/cpe

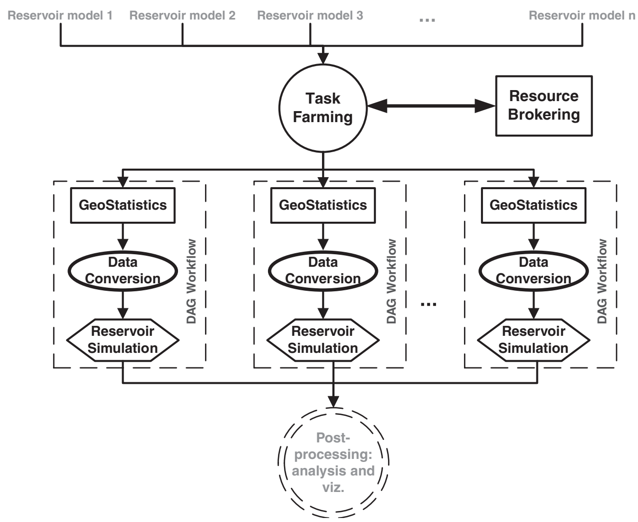

Figure 4. Execution management. Task farming is used as the framework that takes reservoir models as inputs, checks a resource broker for resource allocation, and invokes massive executions.

Each item of this list represents a resource available in a Grid. Execution management functions, such as load balancing and staging in/out, depend on this resource list.

Two major elements of a resource are considered to dispatch simulations: computational capability and architecture. There is a metric adopted to describe the features of a resource, which includes CPU number, CPU speed, CPU load averages, network bandwidth, memory size, queue length, etc. Based on these features, a measurement is taken to calculate the computational capability of a resource. The architecture element is used to decide the type of binaries of geostatistics algorithms and reservoir simulators that should be uploaded. In our current use case (described in Section 4), CPU number and CPU speed of a resource are critical because the adopted reservoir simulator and geostatistics algorithms are sequential programs. The computational capability of a resource C_i is measured as follows:

$$C_i = \text{CPU number} \times \text{CPU speed}$$

The load-balancing strategy aims at dispatching a certain number of simulations to a resource based on its computational capability. The following equation is used to calculate the number of simulations N_i on a resource i:

$$N_i = T \times \frac{C_i}{\sum_{k=1}^{n} C_k}$$

where n is the number of all available resources, C_k is the computational capability of a resource k, and T is the total simulation number of uncertainty analysis.

DOI: 10.1002/cpe

The staging in/out module uploads model data sets and executables to and downloads simulation results from a particular resource. To upload executables, this module needs to check the resource brokering module to obtain the type of the operating system on a particular remote resource. In this manner, the module decides which kind of executable binaries is needed. Retrieving load-balancing calculation, the staging in/out module is aware of how many and which simulation models should be run on a resource. This module also needs to figure out work directory provided by the resource. Once obtaining the required information, this module transfers the data sets with security. The staging out procedure is similar to the staging in procedure. It downloads the simulation results from remote sites.

The invocation module needs to handle remote execution. This module is used to communicate with various local resource management systems (LRMS) on remote resources and invoke simulation executions on the corresponding *LRMS* queues.

The status monitoring module is incharge of the communication with *LRMS*. There are two levels of queues for status monitoring: resource queue on submission machine and *LRMS* job queue on each remote resource. Each resource that is running simulations has an entry in the resource queue. On a particular resource, the job queue of *LRMS* is checked periodically. Once all the simulations dispatched to a resource have been accomplished, the corresponding resource entry in the resource queue is removed.

3.3.3. Implementation

Condor-G and Globus GRAM are employed to handle simulation executions on remote resources. Condor-G allows one to submit jobs into the resource queue and has a log detailing the life cycle of the jobs along with anything else expected from a job queuing system. Considering data transfer performance and large-scale storage requirement of simulation results, GridFTP [18] is used instead of the input and output management provided by Condor-G (i.e. Globus GASS). Globus GRAM provides underlying software to allocate Grid resources, such as authentication and remote program execution.

For flexibility and portability, we adopt *Perl* scripts for functionality implementations: simulation workflow is described; the resource broker queries the external information service for which machines are available, how the machines should be utilized, and when a machine is no longer available; a load-balancing strategy is used to share massive simulation runs across a Grid; etc.

3.4. Grid portal

A Grid portal, built on top of GridSphere and GridPortlets, provides an entry point to conduct reservoir uncertainty studies. Firstly, a GSI certificate is retrieved from a proxy to provide authentication for accessing Grid resources. Secondly, the portal provides web pages for users to specify input data sources, flow simulators, computation resources, etc. In the end, a user submits the job and views the results via this portal.

Let us take job submission to demonstrate how the ResGrid portal works. Figure 5 shows the screenshots mentioned below. An active proxy credential is required for job submission. The portal allows a user to retrieve the credential from a MyProxy [19] server via the credential

DOI: 10.1002/cpe

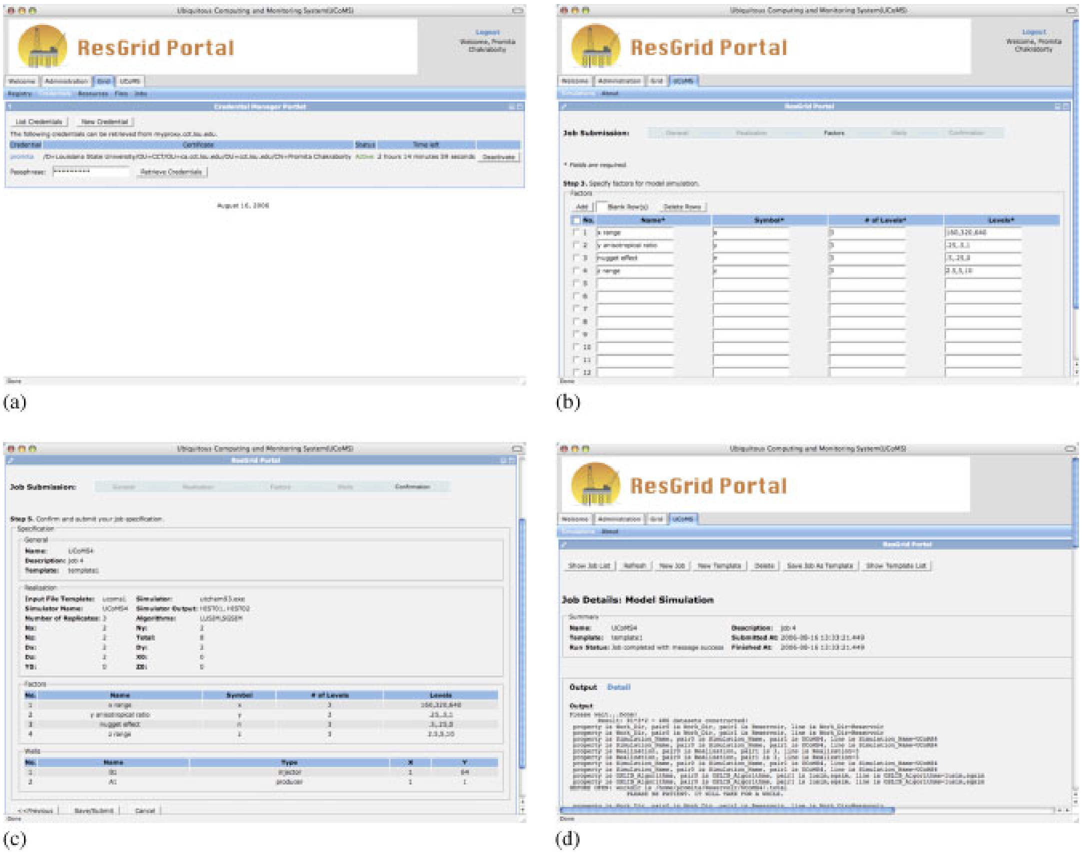

Figure 5. Screenshots of the ResGrid portal. First of all, a user retrieves a credential (a) and specifies the application, such as parameters (b). Once application specification is confirmed (c), the job is submitted to run. The application status details can be monitored (d).

management portlet. The job submission service automatically uses the retrieved credential for authentication with Grid resources. The premise for credential retrieval is that a user needs to create a proxy credential into the MyProxy server via a MyProxy client tool. Then, the job portlet allows the user to specify the parameters, including resource selection, uncertainty factors, factor levels, realizations, well description, etc. The user can also save a parameter set as a template for later use. Finally, once the confirmation has been done, the user can click the 'Submission' button. The portal will invoke the ResGrid's services automatically. The status details, such as reservoir modeling, stage in, job execution, and stage out, will be shown on the portal.

Finally, we have evaluated the performance of the ResGrid from scalability, applicability, portability, and easy-to-use perspectives. The ResGrid demonstrated excellent scalability by our experiments across the LONI cyberinfrastructure [20]. Although it is designed for reservoir

DOI: 10.1002/cpe

studies, the ResGrid provides a generic framework for optimization applications [9]. The design and implementation based on the GAT and Grid portal offer good portability and easy-to-use features.

4. CASE STUDY

The first application is to identify and quantify the geological factor influences on reservoir performance through four different stochastic simulation algorithms. In this section, the application is introduced in terms of study purpose, design of the experiment, and model discrimination.

4.1. Study purpose

The primary purpose is to identify and quantify geological factor influences on the determination of effective properties and production behaviors through four different stochastic algorithms [21], i.e. LU decomposition simulation (LUSIM), sequential Gaussian simulation (SGSIM), LU-SGSIM hybrid (HYBRID), and spectral simulation (SPECSIM).

It is a common method in geological modeling to create permeability fields by stochastic simulation algorithms, which honors the available information at wells and reproduces the pattern of spatial variability among wells. Stochastic simulations can be categorized into direct (*LUSIM*) and sequential approaches (*SGSIM*). *LUSIM* is rigorous but slow. *SGSIM* is quicker but may overestimate the spatial discontinuity. The overestimation may mislead decision making of a project because oil and gas recovery efficiency is underestimated. To improve geological modeling, a hybrid simulation *HYBRID* is created to take advantage of direct and sequential approaches. *SPECSIM* is another kind of simulation, which is based on the *fast Fourier transform*. It is a global method like *LUSIM* in the sense that a global density spectrum is calculated once from the known variogram model and the inverse Fourier transform is performed once to generate the petrophysical fields. *SPECSIM* is fast and accurate in nature.

To provide the comparison of these algorithms, we examine their variability in oil and gas recovery efficiency predictions, caused by geologic heterogeneity and uncertainty through experimental design. We adopt flow numerical simulation to investigate the effects of geologic heterogeneity. Experimental design is used to maximize the information derived from the simulations of various geological models. Empirical response surface models are employed to assess different stochastic algorithms with geologic factors.

4.2. Design of the experiment

Four geologic factors are selected, whose ranges and scales are shown in Table I. Variogram range (r) describes the spatial continuity of permeability. Variogram nugget effect (n) is related to the sources of variation that operate over distances of the shortest sampling interval. Variogram geometric anisotropy ratio (a) is the directional variogram (covariance) that has the same shape and sill but different range values. Variogram vertical range (v) describes the spatial continuity of vertical permeability.

DOI: 10.1002/cpe

Table I. Four factors examined with ranges and scales.

Name	Symbol	Units	L0	L1	L2	L3
Nugget	n	Fraction of sill	0.75	0.50	0.25	0
Range x	r	m	160	320	640	1280
Geometric anisotropy	a	Ratio, fraction	0.25	0.5	0.75	1
Range z	z	m	1.785	3.75	7.5	15

Full factorial experimental design has been adopted as it is the most accurate method to study the joint impacts of factors on a response. The number of simulation runs in this study is $4^4 \times 5 \times 4 \times 4 = 20\,480$ (four factors, each of which has four levels, with five realizations, four geostatistics algorithms, and four well patterns). The flow is simulated through these models in three directions by using different well patterns.

Three responses, i.e. upscaled permeability, breakthrough time, and sweep efficiency, are investigated. Upscaled permeability is defined as the ratio of the flow rate to the pressure computed from simulation results. Breakthrough time is a dimensionless time in pore volumes, which is the total tracer injection volume when the outlet tracer concentration exceeds 1%. Sweep efficiency is the fraction of initial tracer-free water recovered after 1 pore volume ($1\,pv$) of injection. These responses are commonly used to assess the hydraulic effect of permeability field in reservoir engineering. They can be calculated by extracting simulation results. Based on the 20 480 full factorial designed simulation results, a least-square method is used to build the main effects and the interactive polynomial response surface model. The factor sensitivity, uncertainty assessment, and model discriminations are based on this response surface model.

The Grid testbed is provided by the Center for Computation and Technology (CCT) at Louisiana State University and the Center for Advanced Computer Studies (CACS) at the University of Louisiana. High-performance computing facilities include a 128×2 2.0 GHz CPU Linux cluster at CCT, a 12×2 3.0 GHz CPU Linux cluster at CACS, and a 32 1.5 GHz SGI Prism Single System Image server, connected by the Internet. The core services of the testbed are provided by the Globus toolkit as well as other services developed at CCT as part of the Grid computing research initiative.

UTCHEM [22], an open-source software package developed by the University of Texas at Austin, is adopted as the reservoir simulator. GSLIB [23] provides the code of *LUSIM* and *SGSIM*. *HYBRID* algorithm, *SPECSIM* algorithm, and flow model builder are the inhouse code.

4.3. Model discrimination

Figures 6 and 7 show the analysis results of the fraction flow simulations on selected factor combinations. The breakthrough time and the sweep efficiency are different against the four algorithms (taken at the factor space [3, 3, 3, 3]). We can see in Figure 6 that the water front breakthrough (when the well tracer concentration reaches 5%) of *SGSIM* and *HYBRID* models is quicker than that of *LUSIM* and *SPECSIM* models. As breakthrough time depends only on the continuity of the fastest flow paths, the permeability fields of *SGSIM* and *HYBRID* are more fluctuated. Figure 7 is the tracer flow charts of the four algorithms. Water front of the SGSIM model has reached the horizontal well at the right-hand side at the upward of the field, which is a high permeability zone,

DOI: 10.1002/cpe

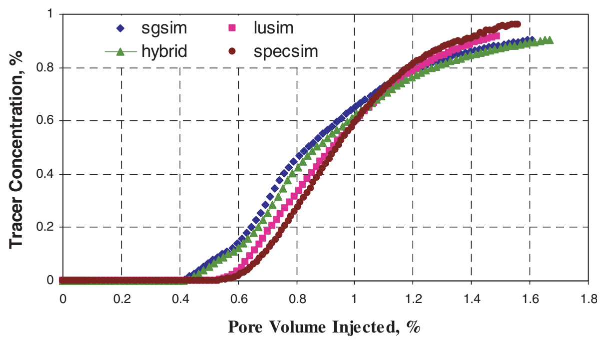

Figure 6. Tracer fraction flow comparison. Water front breakthrough of *SGSIM* and *HYBRID* models is quicker than that of *LUSIM* and *SPECSIM* models. As breakthrough time depends only on the continuity of the fastest flow paths, the permeability fields of *SGSIM* and *HYBRID* are more fluctuated.

but is far away from the well in the downward low permeability zone. After tracer breakthrough, the curve goes up and the slope means the water sweep efficiency. The steeper the slope, the higher the sweep efficiency of the model. The curve slopes of *SGSIM* and *HYBRID* are smaller than those of *LUSIM* and *SPESCSIM*. After $1pv$ injection, injected water sweeps larger area in *LUSIM* and *SPECSIM* models in contrast to *SGSIM* and *HYBRID* models. Again, the fluctuation feature of *SGSIM* and *HYBRID* appears to exert the influence on the flow behavior of sweep efficiency. As for *LUSIM* and *SPECSIM*, the permeability field is more homogeneous and the flow front geometry is more regular.

Flow simulation results of the geostatistical models are analyzed using statistics methods, such as analysis of variance. Standard t- and F-statistics assess that whether or not flow responses of LU models are different from other simulation methods. Tests of t-statistics in contrast to response model coefficients show that the LU simulation responses are significantly different from the sequential Gaussian or the hybrid method for the cases examined. Furthermore, F-tests comparing variances of response models indicate a significant difference between the LU decomposition and the sequential Gaussian. The results are shown in Table II. The p-values are less than 0.05 and the *null* hypothesis that models are similar is rejected with 95% confidence. This means the distribution for effective permeability and breakthrough time are different at 95% at this direction. However, the distribution for sweep efficiency does not differ at 95%. F-tests for other two algorithms and all the other directions have been done. The flow response models of *LUSIM* are significantly different from those of *SGSIM*, *SPECSIM*, and *HYBRID*. The fluctuation of the permeability field generated by *SGSIM* significantly affects the reservoir performance prediction and the uncertainty assessment. *HYBRID* does not have very significant improvement from *SGSIM* because of insufficient conditional data generated by *LUSIM*. *SPECSIM* generates values at all the wells instead of visiting each well sequentially. It reproduces the observed variogram better than SGSIM. However, it cannot build geologic models honoring desired mean, variance, and variogram model simultaneously.

The analysis of flow model results indicate that sequential simulation may yield biased mean estimates and significantly overestimate stochastic fluctuations compared with the more rigorous

DOI: 10.1002/cpe

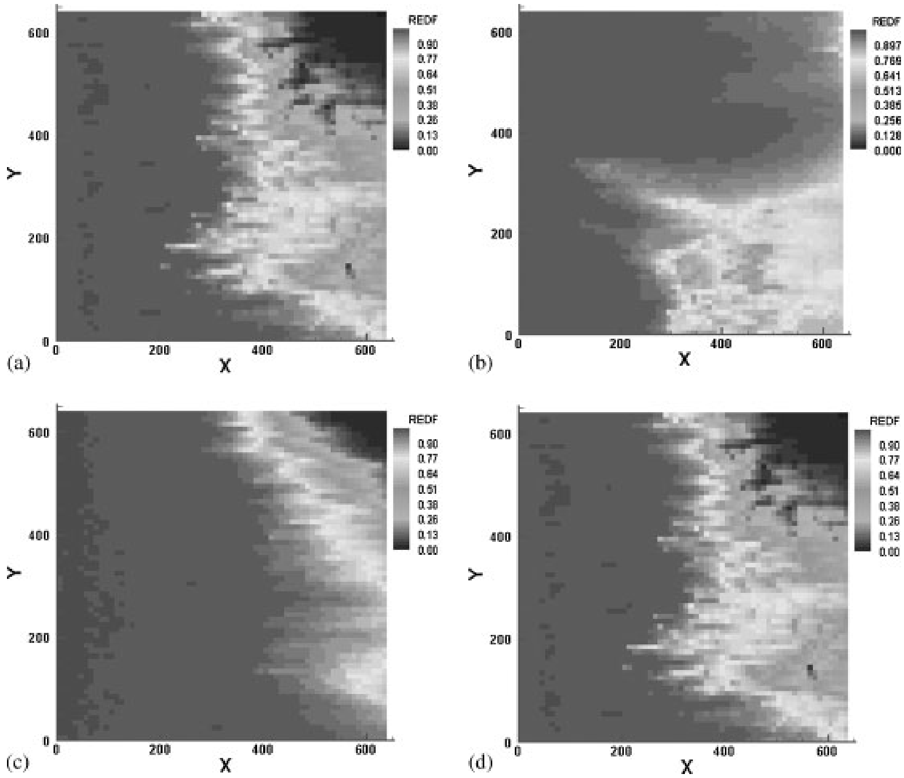

Figure 7. Tracer flow charts. The four different algorithms show the different tracer flow charts: (a) LUSIM; (b) SGSIM; (c) HYBRID; and (d) SPECSIM.

Table II. F-test results between LUSIM and SGSIM.

	Breakthrough time	Upscaled permeability
F-test	0.5125	2.0319
p-value	5.433e−05	1.903e−05
95% Confidence interval	0.371–0.707	1.471–2.80

direct methods such as LU decomposition and spectral methods. Biased estimates clearly affect resource assessments and overestimation of variance distorts uncertainty assessment. These geostatistical insights are important to engineers and geoscientists engaged in reservoir modeling and risk assessment.

DOI: 10.1002/cpe

5. PERTINENT WORK

This section reviews earlier work related to reservoir studies and highlights the contributions of the ResGrid. Although it is designed for reservoir studies, the ResGrid provides a generic framework for optimization applications. An autonomic reservoir framework [24] has been studied by W. Bangerth, H. Klie, etc. It emphasizes the optimization and the integration of high-level services of reservoir management, such as well placement and economical influence. The ResGrid focuses on different areas on optimizing reservoirs: reservoir performance prediction and uncertainty analysis based on the G&G characteristics of a reservoir. Meanwhile, because the geostatistical algorithm characteristics are presumed *a priori* of many factors (e.g. integral range and anisotropy), the sensitivity of algorithm differences to different regionalization models has been considered in the ResGrid.

COUGAR [25] is an industrial contribution on reservoir studies. It is a reservoir uncertainty analysis tool with the ability to make use of Grid resources to run a number of reservoir simulations and achieve the reduction in the individual result turnaround time. However, it does not address the large-scale G&G data integration, and its framework is tied to commercial packages, such as LSF® for execution invocation, ECLIPSE® for reservoir simulator, and security issues are not considered. The ResGrid provides an open generic framework to solve reservoir uncertainty analysis with open-source software packages and a tight security consideration.

6. CONCLUDING REMARKS

Our work focuses on developing an integrated, secure, and easy-to-use PSE *ResGrid* for reservoir uncertainty studies across a Grid. Prior to investments, uncertainty analysis plays a key role in reservoir assessment and performance prediction. Although it is designed for reservoir studies, the ResGrid provides a generic framework for various optimization applications. This paper describes our efforts in terms of data management, execution management, Grid portal, and practical applications.

The essential part of data management is a data replica tool, which has been implemented on top of the GAT. Based on this data tool and uncertainty factor parameter space generation mechanism, reservoir modeling is implemented. To conduct execution management, a task-farming framework has been developed. The resource brokering module captures resource information and uses load-balancing strategies to dispatch reservoir simulations on resources. The invocation module is used to invoke reservoir simulation runs combined with geostatistics algorithms. A GridSphere-based Grid portal has been deployed for use. It provides a web-based user-friendly interface for large-scale uncertainty analysis. The reservoir researchers from the Petroleum Engineering Department at LSU have adopted the ResGrid for their studies.

However, there remains much work ahead on further research and development. The visualization component is under development, which will provide easy-to-understand images to users. Efforts are underway to provide monitoring and steering capabilities at runtime during the execution of a given simulation run and to provide fault tolerance assurance if an error occurs. Another direction in our future work is how to implement history matching mechanisms into the ResGrid.

DOI: 10.1002/cpe

ACKNOWLEDGEMENTS

This project is sponsored by the U.S. Department of Energy (DOE) under award number DE-FG02-04ER46136 and the Board of Regents, State of Louisiana, under contract no. DOE/LEQSF(2004-07).

REFERENCES

1. Bu T, Damsleth E. Errors and uncertainties in reservoir performance prediction. *SPE Formation Evaluation* 1996; **11**(3):124–202.
2. Foster I, Kesselman C, Tsudik G, Tuecke S. A security architecture for computational grids. *Proceedings of the 5th ACM Conference on Computer and Communications Security*, San Francisco, CA, 1998; 83–92.
3. Foster I, Kesselman C (eds). Globus: A Toolkit-based Grid Architecture. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1999; 259–278.
4. Thain D, Tannenbaum T, Livny M. Condor and the Grid. *Grid Computing*: *Making the Global Infrastructure a Reality*. Wiley: New York, 2003.
5. Novotny J, Russell M, Wehrens O. GridSphere: A portal framework for building collaborations. *Concurrency and Computation*: *Practice and Experience* 2004; **16**(5):503–513.
6. Global Grid Forum. Simple API for Grid Applications. http://forge.gridforum.org/projects/saga-rg/ [2004].
7. Lei Z, Huang D, Kulshrestha A, Pena S, Allen G, Li X, Duff R, Kalla S, White C, Smith J. Leveraging Grid technologies for reservoir uncertainty analysis. *Proceedings of High Performance Computing Symposium* (*HPC06*), Huntsville, AL, 2–6 April 2006.
8. Lei Z, Huang D, Kulshrestha A, Pena S, Allen G, Li X, Duff R, Kalla S, White C, Smith J. ResGrid: A Grid-aware toolkit for reservoir uncertainty analysis. *Proceedings of IEEE International Symposium on Cluster Computing and the Grid* (*CCGrid06*), Singapore, 16–19 May 2006.
9. Pena S, Huang D, Li X, Lei Z, Allen G, White C. A generic task-farming framework for reservoir analysis in a Grid environment. *The 8th Workshop on High Performance Scientific and Engineering Computing* (*HPSEC-06*), Columbus, OH, 18 August 2006.
10. Box G, Draper NR. *Empirical Model Building and Response Surface*. Wiley: New York, 1987; 699.
11. Myers RH, Montgomery DC. *Response Surface Methodology*: *Process and Product Optimization Using Designed Experiments*. Wiley: New York, 1995; 700.
12. Aziz K, Settari A. *Petroleum Reservoir Simulation*. Applied Science Publishers Ltd.: London, 1979.
13. White C, Willis B, Dutton S, Narayanan K. Identifying and estimating significant geologic parameters with experimental design. *SPE Journal* (SPE 74140) 2001; **40**:311–324.
14. Willis B, White C. Quantitative outcrop data for flow simulation. *Journal of Sedimentary Research* 2000; **70**(4):788–802.
15. White C, Royer S. Experimental design as a framework for reservoir studies. *2003 SPE Reservoir Simulation Symposium*, Houston, TX, February 2003.
16. Kalla S, White CD. Efficient design for reservoir simulation studies for development and optimization. *2005 SPE Annual Technical Conference and Exhibition*, Dallas, TX, U.S.A., 9–12 October 2005.
17. Allen G, Davis K, Goodale T, Hutanu A, Kaiser H, Kielmann T, Merzky A, Nieuwpoort R, Reinefeld A, Schintke F, Schutt T, Seidel E, Ullmer B. The GridLab Grid application toolkit: Towards generic and easy application programming interfaces for the Grid. *Proceedings of the IEEE* 2005; **93**(3):534–550.
18. Allock W. GridFTP protocol specification. *Technical Report*, Global Grid Forum, GridFTP Working Group, March 2003.
19. Basney J, Humphrey M, Welch V. The MyProxy online credential repository. *Software—Practice and Experience* 2005; **35**(9).
20. Louisiana Optical Network Initiative. http://www.loni.org/ [2007].
21. Goovaerts P. *Geostatistics for Natural Resources Evaluation* (*Geostatistics Series*). Oxford University Press: Oxford, 1997.
22. UTCHEM User Guide. http://www.cpge.utexas.edu/utchem/UTCHEM_Users_Guide.pdf [July 2000].
23. Deutsch CV, Journel AG. *GSLIB Geostatistical Library and User's Guide*. Oxford University Press: Oxford, 1998.
24. Bangerth W, Klie H, Matossian V, Parashar M, Wheeler M. An autonomic reservoir framework for the stochastic optimization of well placement. *The Journal of Networks*, *Software Tools*, *and Applications* 2005; **8**(4):255–269 (Special Issue on Challenges of Large Applications in Distributed Environments).
25. COUGAR. http://www.slb.com/media/services/software/reseng/ressim_ondemand.pdf [2007].