Dynamic Decision and Data-Driven Strategies for the Optimal Management of Subsurface Geo-systems*

Manish Parashar[1], Hector Klie[2], Tahsin Kurc[3], Umit Catalyurek[3], Joel Saltz[3], Mary F. Wheeler[2]

[1] TASSL, Dept. of Electrical & Computer Engineering, Rutgers, The State University of New Jersey, New Jersey, USA
parashar@caip.rutgers.edu
[2] CSM, ICES, The University of Texas at Austin, Texas, USA
{klie,mfw}@ices.utexas.edu
[3] Dept. of Biomedical Informatics, The Ohio State University, Ohio, USA
{kurc,umit,jsaltz}@bmi.osu.edu

Abstract. Effective geo-system management involves understanding of the interplay between surface entities (e.g., locations of injection and production wells in an oil reservoir) and appropriately effecting subsurface characteristics. This in turn requires efficient integration of complex numerical models of the environment, optimization procedures, and decision making processes. The dynamic, data-driven application systems (DDDAS) paradigm offers a promising framework to address this requirement. To achieve this goal, we have developed advanced multi-physics, multi-scale, and multi-block numerical models and autonomic systems software for dynamic, data-driven applications systems. This work has enabled a new generation of data-driven, interactive and dynamically adaptive strategies for subsurface characterization and management. These strategies have been applied to different aspects of subsurface management in strategically important application areas, including simulation-based optimization for the optimal oil well placement and the data-driven management of the Ruby Gulch Waste Repository. This paper summarizes the key outcomes and achievements of our work, as well as reports ongoing and future activities focused on uncertainty estimation and characterization.

* The research presented in this paper is supported in part by the National Science Foundation Grants ACI-9984357, EIA-0103674, EIA-0120934, ANI-0335244, CNS-0426354, IIS-0430826, ACI-9619020 (UC Subcontract 10152408), ANI-0330612, EIA-0121177, SBR-9873326, EIA-0121523, ACI-0203846, CNS-0643969, ACI-0130437, CCF-0342615, CNS-0406386, CNS-0426241, ACI-9982087, CNS-0305495, NPACI 10181410, Lawrence Livermore National Laboratory under Grant B517095 (UC Subcontract 10184497), Ohio Board of Regents BRTTC BRTT02-0003, and DOE DE-FG03-99ER2537.

Introduction

The dynamic, data driven application systems (DDDAS) paradigm is enabling a new generation of end-to-end multidisciplinary applications that are based on seamless aggregation of and interactions between computations, resources, and data. An important class of applications in this paradigm includes simulations of complex physical phenomena that symbiotically and opportunistically combine computations, experiments, observations, and real-time data to provide important insights into complex systems.

As part of the "Instrumented Oil-Field" project [1-8], we have developed several key DDDAS technologies to enable a new generation of data-driven, interactive and dynamically adaptive strategies for subsurface characterization and reservoir management. This project aimed at completing the symbiotic feedback loop between measured data and the computational models to provide more efficient, cost-effective and environmentally safer production of oil reservoirs, which can result in enormous strategic and economic benefits. The project has led to conceptual and infrastructure solutions, which include advanced multi-physics, multi-scale and multi-block numerical models as well as a DDDAS software stack. The software stack provides a middleware for autonomic DDDAS applications and consists of a Grid-based execution engine that supports self-optimizing, dynamically adaptive applications, distributed data management services for large scale data management and processing, and self-managing middleware services for seamless discovery, access, interactions and compositions of services and data on the Grid.

In this paper, we summarize these computational techniques and infrastructure components for the dynamic data-driven management and optimization of subsurface geo-systems. The overall approach is based on the Integrated Parallel Accurate Reservoir Simulator (IPARS), which supports a multi-block approach for the scalable simulation of multi-physics, multi-scale and multi-algorithm reservoir applications, and its interplay with Discover/Automate [9, 10] (a decentralized and autonomic Grid middleware substrate), STORM/DataCutter [11-18] (a data subsetting and processing middleware infrastructure), Seine [19] (an adaptive multiblock computational engine), and two very efficient stochastic optimization algorithms [2], the SPSA (Simultaneous Perturbation Stochastic Approximation) and the VFSA (Very Fast Simulated Annealing). Together, these components have enabled new paradigms and strategies for subsurface management in strategically important application areas, including multiphysics applications that imply the coupling of flow, geomechanics, petrophysics and seismics, and simulation-based optimization for well placement, data-driven management of subsurface contaminants at the Ruby Gulch Waste Repository, and more recently, efficient subsurface characterization.

This work stresses how the conjunction of the aforementioned components offers the possibility of developing large-scale efficient approaches to perform uncertainty characterization and management studies, leading to opportune reservoir management decisions. We believe that the approaches discussed here can also be applied to other fields such as environmental remediation and biomedical tissue engineering.

The rest of this paper is organized as follows. First, we provide an overview of DDSMF (Dynamic Data-Driven Subsurface Management Framework) showing the orchestration of the key components required for the management of subsurface sys-

tems. Next, we proceed with a description of the models, methods and middleware supporting DDSMF. We devote a section to show the capabilities of the framework and highlight its ability to efficiently tackle a set of different subsurface applications. We end the paper with some concluding remarks and a summary of on-going and future research avenues.

A Framework for Dynamic Data-Driven Subsurface Management

The Dynamic Data-Driven Subsurface Management (DDSMF) framework comprises accurate, multi-resolution, multi-physics models derivable from diverse data sources, coupled with dynamic data-driven optimization strategies for uncertainty estimation and decision-making (see Figure 1). Traditionally, the estimation of model parameters and the optimization of decision parameters have been treated separately in decision-making applications. Moreover, most optimization frameworks have been built under the assumption of perfect knowledge of (noise-free) data, forcing specialists to further tune the data when results do not describe the phenomenon under study. This process is unreliable and inefficient in practice, and does not provide, in most cases, a fully unbiased measurement of uncertainty. DDSMF aims at generating a functional and closely connected feedback loop between data and simulation, driven by optimization. It is composed of three major components: The Dynamic Decision System (DDS), the Dynamic Data-Driven Assimilation System (DDA), and the Autonomic Grid Middleware (AGM). The orchestration of these components provides the computational feedback between data and the model through optimization (Figure 1).

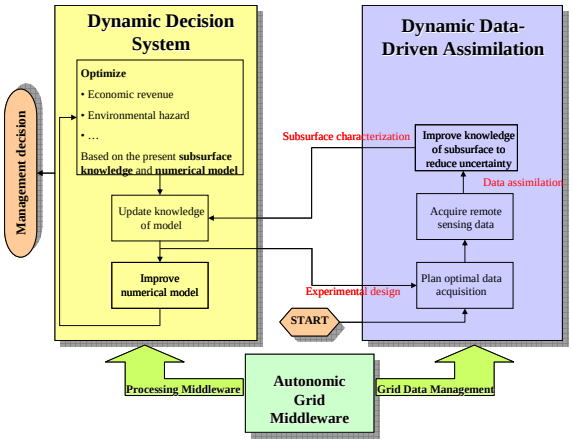

Figure 1: Conceptual architecture of the Dynamic Data-Driven Subsurface Management Framework (DDSMF).

Dynamic Decision System: This module utilizes as its input the current knowledge of a subsurface system, and initiates the decision-making process. It also includes estimates of the reliability and accuracy of proposed strategies, taking into account numerical errors as well as uncertainty in subsurface characterization. This involves the formulation of objective functions, forward numerical simulation models and optimization algorithms.

The goals include (1) the optimal scheduling, design and deployment of observations (e.g., wells) to optimize a desired response (e.g., optimum economic revenue,

minimum bypassed oil); and (2) fitting numerical model output to field measured values (e.g., history matching, seismic data and/or resistivity data fitting together with well constraints). Finding the optimum in realistic reservoir scenarios generally leads to an ill-posed problem. Furthermore, due to the complexity and cost of running forward models, optimization methods must minimize the number of function evaluations. Methods that have been used with particular success are a variant of the Simulated Annealing algorithm, i.e., the Very Fast Simulated Annealing (VFSA) and the Simultaneous Perturbation Stochastic Approximation (SPSA) method [2]. In order to cope with the curse of dimensionality associated to the optimization, it is almost mandatory to develop smart parameterization and meta-modeling strategies to reduce the size and associated costs of the search space. Each update of the reservoir model implies a systematic improvement of the quality and complexity of the numerical model to accommodate multi-physics, multi-scale and multi-algorithmic capabilities. This in turn motivates the generation of specialized solvers and robust error indicators to guide mesh adaptivity, time stepping, and the strength of the coupling dictated by the required accuracy of multiple processes and scales.

Dynamic Data-Driven Assimilation and Analysis System: From the data management and integration perspective, the simulation and optimization components interact with both, the data generated by simulations and stream of data collected by field sensors. Data is also collected in a less regular (but more conventional) basis through seismic surveys, core samples, and well observations. Simulation and optimization components are driven by the assimilation and analysis of these datasets. For example, if the information available about the subsurface is insufficient, then all simulations will be unreliable and will result in large error margins. Subsurface characterization with geophysical and fluid measurements involves the quantitative assessment of the 3D spatial distribution of material properties such as density, P- and S-wave velocities, electrical resistivity, permeability, porosity, magnetic polarization, pressures, or temperatures, from a finite set of noisy measurements. 4D seismic surveys are of increasing use in industry for reservoir characterization. Pressure and production data can be matched with greater confidence only by combining seismic data and reservoir simulation results, because ambiguities due to fault transmissibility and sand body connectivity are reduced.

Datasets in large scale dynamic data driven subsurface management applications are characterized by the large volume of data and the distributed nature of the datasets. Moreover, data are acquired at different time and spatial scales and resolutions. High-end sensor technologies enable the measurement of properties of a given field rapidly and at high resolution. A single seismic survey, for example, is capable of generating multi-terabytes of data from one field. These surveys are carried out multiple times to observe structural changes in the field and changes in rock properties. The complexity of numerical models necessitates the execution of simulations on high-performance, massively parallel systems distributed in a Grid environment, thus generating large volumes of distributed data. These simulations can easily generate Terabytes of simulation output on multi-scale, multi-resolution meshes. The dynamic data assimilation and analysis component provides support for data management, data integration, and data processing for the analysis, interpretation, storage and retrieval of these large and heterogeneous data sets.

Autonomic Grid Middleware: Dynamic data-driven subsurface management presents challenging middleware and runtime requirements. Distributed computation engines and adaptive runtime management strategies are required to support efficient and scalable implementations of adaptive geophysical and flow reservoir simulations in heterogeneous, widely distributed and highly dynamic Grid environments. Control networks with embedded software sensors and actuators are required to enable computational components to be accessed and managed externally, both interactively and using automated policies, to support runtime monitoring, dynamic data injection and control. Self-managing middleware services are necessary to enable seamless interactions where the application components, Grid services, resources (systems, CPUs, instruments, storage) and data (archives, sensors) can interact symbiotically and opportunistically as peers. These middleware services must support autonomic behaviors so that their interactions and adaptations can be driven by high-level policies. The autonomic Grid middleware component provides core capabilities and services to address these requirements.

Models, Methods and Middleware

We have developed simulators for complex numerical models, optimizations methods, and a suite of middleware services to enable the functionality of the DDSMF. In this section we provide brief descriptions of these components.

The Integrated Parallel Accurate Reservoir Simulator (IPARS): IPARS represents a new approach to parallel reservoir simulator development, emphasizing modularity, code portability to many platforms, ease of integration and inter-operability with other software. It provides a set of computational features such as memory management for general geometric grids, portable parallel communication, state-of-the-art non-linear and linear solvers, keyword input, and output for visualization. A key feature of IPARS is that it allows the definition of different numerical, physical, and scale models for different blocks in the domain (i.e., multi-numeric, multi-physics, and multi-scale capabilities). A more technical description of IPARS capabilities can be found in [20]. Recent developments have been made in solvers and error estimators to be able to rely with efficient and robust criteria to update the subsurface model in DDS [21-25].

Optimization algorithms: Novel stochastic optimization algorithms have been included in the framework, namely, the Very Fast Simulated Annealing (VFSA), the Finite Difference Stochastic Approximation (FDSA) and the Simultaneous Perturbation Stochastic Approximation (SPSA). The VFSA and SPSA, in particular, are amenable to large scale implementations [2, 3].

A critical issue in optimization is that parameter estimation may involve several hundreds of thousands of variables and the objective function evaluation is a highly demanding process since it involves a full simulation run. This also rules out the possibility for using gradient-based methods, especially in a multi-model environment that systematically aims at different physics and algorithms for which sensitivity coefficients are not trivial either to compute or reformulate. We have recently developed parameterization and meta-modeling strategies to cope with the curse of dimensionality associated with parameter estimation [26-28]. The parameterization was based on

the combination of singular value decomposition (SVD) and the wavelet transform to perform a multi-scale assessment of the most relevant features in the parameter space. The meta-modeling strategy allows for replacing the simulation model by a surrogate model in the neighborhood of the optimal solution. The meta-model offers the alternative to estimate derivatives (i.e., sensitivities) and even switch to a fast local optimization method if necessary [26].

Storage and Management of Large Volumes of Data: Effective solutions to problems targeted in this work involve gleaning and extracting information from results of optimization processes, output of complex numerical simulations, and data gathered by field sensors. Datasets generated by an optimization run consists of the values of the input and output parameters along with the output from simulations of a numerical model of the physical domain. Datasets gathered from field sensors consist of readings obtained from each sensor, the location of the sensor, and the date of the reading. This information provides a dynamically updated (as more readings are obtained) historical record of the field under study. Both simulation and sensor datasets can be generated and stored at multiple locations in a Grid environment. By maintaining simulation and sensor datasets, a large-scale, distributed knowledge base can be created. This knowledge base can be used to speed up the execution of optimization runs, to carry out post-optimization analyses, to refine numerical models using field data, and to control where and how much field data should be collected, thus implementing a dynamic, data-driven application system approach. Common types of queries against these datasets include computing data subsets via range queries, aggregations such as counts, averages on over regions of meshes, and differences between regions of interest on multi-resolution datasets. Several core functions need to be supported to create and manage this type of a knowledge base in a Grid environment. These functions include support for virtualization of file based datasets and for data subsetting and data product generation (e.g., data aggregates from data subsets).

Virtualization of Large Scale File-based Datasets. Datasets generated by simulations or field measurements are stored in files, because most simulations and sensor systems write data to files, and staging multi-terabyte volumes of data into a database system takes very long time. The datasets are typically stored in a wide range of file formats, making it difficult to search for and retrieve the data of interest from a dataset. This creates a major obstacle to effective integration of these datasets in analysis and decision making processes. On the hardware front, systems built from commodity disks have become the platform of choice for storage of scientific datasets. We have developed a service-oriented framework, called STORM [15-18], that provides a database abstraction on datasets stored in files on disk-based parallel and distributed storage platforms. It provides basic database support for 1) selection of the data of interest from dataset files, and 2) transfer of selected data from storage nodes to compute nodes for processing. STORM can perform parallel I/O on distributed data and execute data selection and data filtering operations on cluster platforms. The abstractions implemented by STORM include virtual tables based on object-relational database models, select queries, and distributed data descriptors. The virtual table abstraction represents the contents of a dataset as a big table. The rows of the table correspond to data elements in the dataset. The STORM middleware can be used to support SQL-style select queries against these virtual tables. The client program that requests the data may be a parallel program implemented using a distributed-memory

programming paradigm. In this case, the distribution among client nodes of the data elements returned as the result of the query can be represented as a distributed array. The distributed data descriptor abstraction is utilized to specify how data elements selected from the database are to be distributed across the nodes of the client program.

Data Product Generation. The support for data product generation (e.g., data aggregates from datasets, visualization of data subsets) is provided by the DataCutter system [11-14]. DataCutter is a middleware system designed to support processing of large datasets in a distributed environment. A DataCutter application consists of a network of interacting application-specific components, called filters, one or more filter groups. Filters are connected through logical streams and collectively realize the processing structure of the application. A logical stream denotes a uni-directional data flow from one filter (i.e., the producer) to another (i.e., the consumer). DataCutter allows for combined use of task-parallelism, data-parallelism, and pipelining for reducing execution time of data processing and analysis applications.

Autonomic Grid Middleware Substrate: Emerging knowledge-based and DDDAS applications, such as the subsurface management applications described in this paper, combine computations, experiments, observations, and real-time data, and are highly heterogeneous and dynamic in their scales, behaviors, couplings and interactions. Furthermore, the underlying enabling computational and information Grid is similarly heterogeneous and dynamic, globally aggregating large numbers of independent computing and communication resources, data stores and sensor networks. Together, these characteristics result in complexities and challenges that require a fundamentally different approach to how the applications are formulated, developed and managed - one in which applications are capable of managing and adapting themselves in accordance with high-level guidance from the experts based on their state, the available information and their execution context [7].

AutoMate [10], an Autonomic Computational Engine for management and control, investigates conceptual models and implementation architectures to address these challenges and enable the development and execution of such self-managing Grid applications. Specifically, it investigates the development of autonomic computational engines and runtime system that can support efficient and scalable implementations of adaptive multi-physics, multi-model, and multi-scale simulation models in heterogeneous, widely distributed and highly dynamic Grid environments. The engines enable control networks to support computational components to be accessed and managed externally, both interactively and using automated policies, to support runtime monitoring, dynamic data injection and control. Furthermore, self-managing middleware services enable seamless interactions where the application components, Grid services, resources (systems, CPUs, instruments, storage) and data (archives, sensors) can interact symbiotically and opportunistically as peers. This middleware supports autonomic behaviors so that the interactions and feedback between scales, models, simulations, optimization services sensor data and data archives can be orchestrated using high-level policies and rules to navigate the parameter space and optimize design. Key components include:

- The *Seine Computational Engine* [19] that implements a dynamic geometry-based shared space interaction model to support the dynamic and complex communication and coordination patterns resulting from the multi-physics, multi-numerics, multi-scale and multi-domain couplings required by the multi-block

parallel multi-block simulations. Seine complements existing parallel programming frameworks such as MPI and OpenMP.

- The *Accord Programming System* [29] that enables the definition of autonomic components and the dynamic composition, management and optimization of these components using externally defined rules and constraints. Autonomic components in *Accord* export sensors and actuators for external monitoring, control and adaptation.
- The *Autonomic Runtime Environment* [30] provides policies and mechanisms for both "system sensitive" and "application sensitive" runtime adaptations to manage the heterogeneity and dynamism of the applications as well as Grid environments. The former are driven by the current system state and system performance predictions while the latter are based on the current state of application.
- The *Content-based Grid Middleware Substrate* [10] that supports autonomic application behaviors and interactions, and to enable simulation components, sensors/actuators, data archives and Grid resources and services to seamlessly interact as peers. Key components of the middleware include the Meteor, a decentralized infrastructure for decoupled associative interactions, the Squid content-based routing engine and decentralized information discovery service, and the Pawn peer-to-peer messaging substrate.
- The *Discover Collaboratory* [9] that provides a collaborative problem solving environment and enables geographically distributed scientists and engineers to collaboratively monitor, interact with, and control high performance applications in a truly pervasive manner using portals.

Putting it to Work: Applications of DDSMF

Simulation-based Optimization of Optimal Well Placement [3, 5]: The determination of optimal well locations is a challenging problem since it depends on geological and fluid properties as well as on economic parameters. This leads to a very large number of potential scenarios that must be evaluated using several numerical reservoir simulations. The high costs of simulation make an exhaustive evaluation of all these scenarios infeasible. As a result, the well locations are traditionally determined by analyzing only a few scenarios. However, this *ad hoc* approach may often lead to incorrect decisions with a high economic impact. In this application we employ the DDSMF to address these challenges for optimization of well placement and configuration.

This DDSMF application involves: (1) the coupling of IPARS with VFSA, FDSA and SPSA algorithms and their execution on the Grid; (2) distributed data archives for historical, experimental (e.g., data from field sensors), and simulated data; (3) Grid services that provide secure and coordinated access to the resources and information required by the simulations; (4) external services that provide data, such as current oil market prices, relevant to the optimization of oil production or the economic profit; and (5) the actions of scientists, engineers and other experts, in the field, the laboratory, and in management offices.

Furthermore, in a Grid environment, data analysis programs need to access data subsets on distributed storage systems. This need is addressed by STORM and DataCutter. Large datasets, generated from previous optimization runs and potentially stored at distributed storage systems, are queried using STORM to extract data subsets. For example, a query may request simulation data between time steps T_1 and T_2 and regions of the mesh where saturation of oil is above a user-defined threshold. The results of the query are processed by DataCutter to calculate average oil saturation and maximum oil saturation in those regions and between the time steps. Such queries and data products can be used to compare output from multiple simulations and assess whether a particular set of numerical model and optimization parameter values lead to expected (or desired) output. This assessment can be used to drive the initial set of parameter values for the numerical models and optimization methods for the current optimization run.

The AutoMate autonomic Grid middleware provides the support for items 3, 4, and 5 in autonomic oil reservoir optimization process [3]. The overall scenario is illustrated in Figure 2. The components include: IPARS providing sophisticated simulation components that encapsulate complex mathematical models of the physical interaction in the subsurface, and execute on distributed computing systems on the Grid; IPARS Factory responsible for configuring IPARS simulations, executing them on resources on the Grid and managing their execution; Optimization Service (e.g. VFSA and SPSA); and Economic Modeling Service that uses IPARS simulation outputs and current market parameters (oil prices, costs, etc.) to compute estimated revenues for a particular reservoir configuration.

These entities dynamically discover and interact with one another as peers to achieve the overall application objectives. Figure 2 illustrates the overall workflow and the key interactions involved. For example: (1) The experts use pervasive portals to interact with the Discover middleware and Grid services to discover and allocate appropriate resource, and to deploy the IPARS Factory, Optimization Service, and Economic model peers. (2) The IPARS Factory discovers and interacts with the Optimization Service to configure and initialize it. (3) The experts interact with the IPARS Factory and Optimization Service to define application configuration parameters. (4) The Optimization algorithm is seeded using DataCutter/STORM. This seed can be obtained by querying previously executed simulations. (5) The IPARS Factory then interacts with the Discover middleware to discover and allocate resources and to configure and execute IPARS simulations. (6) The IPARS simulation now interacts with the Economic model to determine current revenues, and discovers and interacts with the Optimization Service when it needs optimization. (7) The Optimization Service provides IPARS Factory with an improved well location, which then (8) launches new IPARS simulations with updated parameters. (9) Experts can at anytime discover, collaboratively monitor and interactively steer IPARS simulations, configure the other services and drive the scientific discovery process.

The optimization of well locations using the VFSA and SPSA optimization algorithms for two different scenarios are presented in Figure 3. The goal is to maximize profits for a given economic revenue objective function. The well positions plots (3(a) left and 3(b) right) show the oil field and the positions of the wells. Black circles represent fixed production wells and a gray square at the bottom of the plot is a fixed injection well. The plots also show the sequence of guesses for the position of the other

injection well returned by the optimization service (shown by the lines connecting the light squares), and the corresponding normalized cost value (3(a) right and 3(b) left). Further details can be found [3, 31].

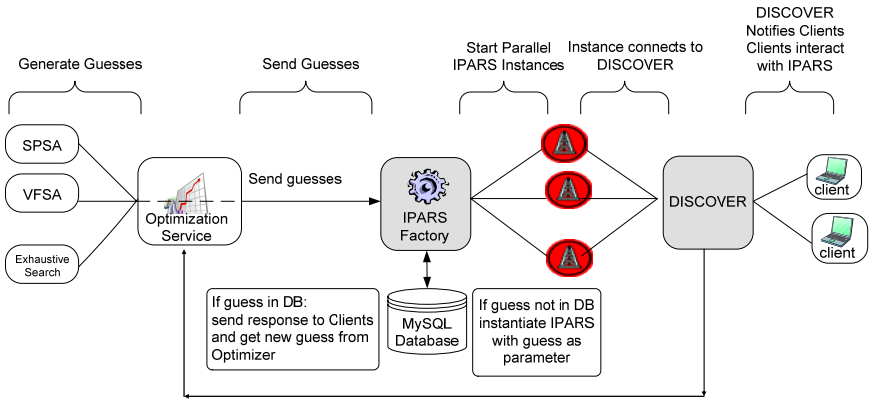

Figure 2: Autonomic oil reservoir optimization using DDSMF.

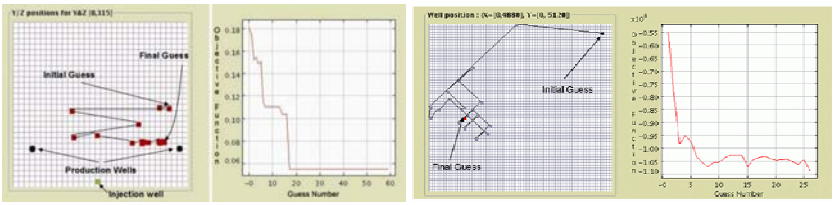

Figure 3: Convergence history for the optimal well placement in the Grid using (a) VFSA algorithm and (b) SPSA algorithm.

The overall process described above is data-driven and autonomic in that the peers involved automatically detect sub-optimal oil production behaviors at runtime based on dynamically injected data, and orchestrate interactions among themselves to correct this behavior. Further, the detection and optimization process is achieved using policies and constraints that minimize human intervention. The interactions between instances of peer services are opportunistic, based on runtime discovery and specified policies, and are not predefined.

The Dynamic Data-Driven Waste Management: The Ruby Gulch Waste Repository[32]: The Gilt Edge Mine is located near Deadwood, South Dakota. Mining for gold and silver at this site occurred from 1880-1999. In 1999 the Dakota Mining Corporation (which operated the mine through its subsidiary, Brohm Mining Company) declared bankruptcy, and the site reverted to the state of South Dakota. Mining activities had resulted in several negative environmental impacts on the site. One of the main environmental issues was the presence of multiple sources of ARD (Acid Rock Drainage). The primary source was the Ruby Waste Rock Repository. This repository is a valley in which mine rock was disposed of post leaching. It contains approximate 11 million cubic yard of waste rock. As ARD flowing from this repository

would severely impact drinking water quality downstream of the site, this ARD needs to be captured and treated.

In order to minimize the amount of water coming out of this repository, a ROD (Record of Decision) for this repository called for the emplacement of a cap over the site. As EPA had interest in the monitoring of the performance of this cap, a monitoring system was designed and installed by scientists from the Idaho National Laboratory (INL) [33]. The monitoring system autonomously collects continuous data using the following sensors: (1) a weather station operated by SDENR (South Dakota Department of Environment and Natural Resources); (2) an outflow meter at the bottom of the Ruby Repository; (3) temperature sensors in four well boreholes; (4) advanced tensiometers are located within boreholes and measure matrix potential (related to water saturation); and (5) a multi-electrode resistivity system.

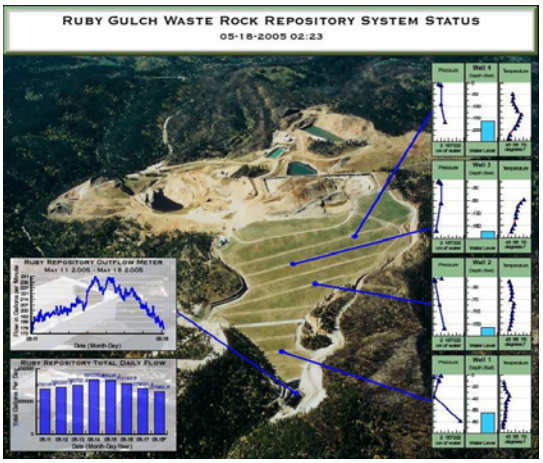

Figure 4: Instrumentation at the Rube Gulch Waste Repository.

In order to make better predictions from the measurements at the Gilt Edge site, the first task is to develop a model that suitably explains the observations from the experiments at the waste repository. For instance, it was observed that there exists a diurnal/seasonal variation in the outflow measurements. Using IPARS, a system of modified air-water equations can be used to model the problem. This solution takes into account that water can exist in the air phase as vapor, and can explain the diurnal variations qualitatively.

Once the prediction model is calibrated, the next challenge is to determine the physical parameters of the site, such as permeability, porosity and capillary pressure, in order to reproduce the exact measured outflows at the site. This task can be modeled as a parameter estimation problem using the numerical model of the environment. This is where an efficient optimization method such as SPSA or VFSA plays a significant role. At present, the SPSA method is being implemented on the hydrology model in IPARS. Using the autonomic computational engine, the execution of IPARS and the optimization methods can be dynamically orchestrated in a Grid environment.

To perform parameter estimation a mismatch function, based on the difference of measured and calculated outflow of water at a specified location at the site, is posed. Similarly, an objective function based on maximum cleanup rate can be designed for optimal site management. To extend these optimizations for data assimilation, the numerical models implemented using IPARS is refined using dynamic data from outflow measurements at the site. To achieve this, an optimal control scheme should be formulated to accommodate dynamic changes to the parameters (i.e., properties) and state variables (e.g., saturations, pressures, temperatures) of the model. Increasing understanding of the physical model, and the need to respond more quickly to observations, leads to meta-models (surrogate models) or reduced models. These simpler models mimic the behavior of the original predictive model given by IPARS.

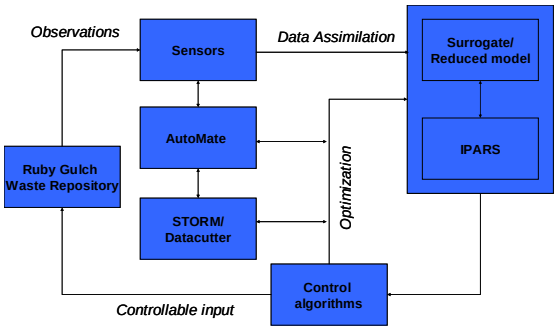

Figure 5: Dynamic Data-Driven Waste Management using the DDSMF.

Accurate model prediction and optimization capabilities in conjunction with the Grid middleware and data management tools described in Section 2 makeup the fundamental components of the dynamic data-driven waste management workflow (see Figure 5). The workflow uses the DDSMF to add autonomic decision making and control capabilities to the monitoring process.

The AutoMate autonomic computational engine and middleware services provide the infrastructure for: (1) enabling the efficient, large scale, dynamically adaptive multi-block IPARS simulations; (2) for discovering, aggregating and assimilating data from the sensors at the remote Gilt-Edge site and dynamically injecting it into the simulation processes as required; (3) selecting and invoking appropriate optimization services; (4) enabling dynamic composition of services to realize data driven workflows on the Grid; and (4) enabling remote collaborative and interactive access to the simulations and the data using pervasive portals.

The estimation of physical properties of the environment involves search of a parameter space and requires data from outflow measurements to dynamically drive the simulation of the numerical models and parameter space search. A knowledge base can be created from the datasets that are generated (via simulations or field measurements) and referenced in this application to speed up the optimization process. At any given step during optimization, the knowledge base can be queried to see if a given step, or a subset of numerical simulations at that step, has been already evaluated. STORM and DataCutter can be employed to support queries into distributed collections of large datasets stored as a collection of files. For instance, in post-optimization analyses, a user may want to compare and correlate a subset of results obtained from one optimization run with results from another set of optimization runs. In that case, the user can submit a query to STORM to find the subsets of data of interest from dis-

tributed datasets. The results can further be processed in the DataCutter system using application-specific filters to compare data values or regions of the mesh obtained from different optimization runs.

Concluding Remarks and Future Work

Dynamic, Data-driven approaches coupled with Grid computing provide a promising approach for tackling large-scale oil engineering and subsurface management applications. However, due to the significant complexity of processes and scale taking place in real subsurface geo-systems, it is a true challenge to accommodate computations "on-demand" fashion. In this paper, we have provided an overview of recent computational strategies and systems software that we have developed to facilitate the incorporation of more complex processes, data, interaction and understanding of the subsurface, and more specifically the oil reservoir. We have proposed the DDSMF to support such dynamic data-driven subsurface management applications, developed its core components, and successfully applied in different applications.

The authors believe that computational frameworks and systems to support dynamic, data-driven strategies, such as the DDSMF, will proliferate with increasing deployment of sensor technology and availability of computing power. Furthermore, these technological advances should further facilitate human and software integration towards more reliable decision-making in reservoir engineering, and in geo-scientific applications in general. In our research, we have outlined strategies that demonstrate how Grid middleware and data management tools can enable and support multi-physics, multi-scale and multi-algorithm computations. Such capabilities are especially important in uncertainty quantification and management and as a result, increasing the reliability of decision-making in subsurface planning.

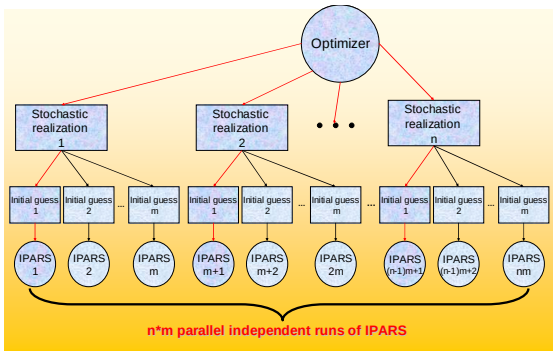

Figure 6: Levels of parallelism in uncertainty analysis.

Due to the lack of data, subsurface characterization via parameter estimation and uncertainty analysis must rely on a considerable number of simulations on different scenarios (i.e., realizations). Computationally, this creates opportunities for the use of massive parallel and distributed systems, autonomic middleware solutions and large-scale data management. These components can collectively exploit the concurrency and asynchrony in the problem to support its efficient solution, as illustrated in Figure 6. For example, based on our particular framework, we can identify that (1) simulations for different scenarios are independent of each other, (2) different initial guesses

(perturbations) may be performed for a particular class of realizations, (3) stochastic gradients (as in SPSA) could be computed with a variable number of function evaluations, (4) different optimizers could be executed in parallel to improve robustness in the estimations and, finally, and (5) each function evaluation or simulation using a high-performance simulator such as IPARS can run in parallel.

There are other components that deserve further development, including: (1) robust parameterization, metamodeling and model reduction methods to mitigate the overall burden of computation [26, 34]; (2) incorporation of state-of-the-art non-intrusive stochastic methods that are capable of dealing with multipoint (non-stationary) statistics and are more efficient than conventional Monte Carlo simulations (see e.g., [35-37]); (3) integration of data-assimilation tools capable of dealing with multiple scales and physics [38, 39]; (4) smart strategies for information extraction to define "on-demand" criteria for executing the most appropriate simulation and inverse model in a particular space and time frame [40]. Part of these developments have been already initiated and are expected to be reported on in forthcoming publications.

References

1. Bangerth, W., et al., *An Autonomic Reservoir Framework for the Stochastic Optimization of Well Placement.* Cluster Computing, 2005. **8**(4): p. 255-269.
2. Bangerth, W., et al., *On Optimization Algorithms for the Reservoir Oil Well Placement Problem.* Computational Geosciences, 2006. **10**(3): p. 303-319.
3. Bangerth, W., et al., *An Autonomic Reservoir Framework for the Stochastic Optimization of Well Placement.* Cluster Computing: The Journal of Networks, Software Tools, and Applications, 2005. **8**(4): p. 255-269.
4. Klie, H., et al., *Models, methods and middleware for Grid-enable multiphysics oil reservoir management.* Engineering with Computers, 2006. **22**: p. 349-370.
5. Matossian, V., et al., *Autonomic Oil Reservoir Optimization on the Grid.* Concurrency and Computation: Practice and Experience, John Wiley and Sons, 2005. **17**(1): p. 1-26.
6. Parashar, M., et al., *Application of grid-enabled technologies for solving optimization problems in data-driven reservoir studies.* Future Generation of Computer Systems, 2005. **21**(1): p. 19-26.
7. Parashar, M., et al., *Application of Grid-Enabled Technologies for Solving Optimization Problems in Data-Driven Reservoir Studies.* Journal of Future Generation Computer System, Special Issue on Engineering Autonomic Systems, 2005. **21**(1): p. 19-26.
8. Parashar, M., et al., *Enabling Interactive Oil Reservoir Simulations on the Grid.* Concurrency and Computation: Practice and Experience, 2005. **17**(11): p. 1387-1414.
9. Mann, V. and M. Parashar, *DISCOVER: A Computational Collaboratory for Interactive Grid Applications*, in *Grid Computing: Making the Global Infrastructure a Reality*, F. Berman, G. Fox, and T. Hey, Editors. 2003, John Wiley and Sons. p. 727-744.
10. Parashar, M., et al., *AutoMate: Enabling Autonomic Grid Applications.* Cluster Computing: The Journal of Networks, Software Tools, and Applications, Special Issue on Autonomic Computing, 2006. **9**(2): p. 161--174.
11. Beynon, M., et al., *Processing Large-Scale Multidimensional Data in Parallel and Distributed Environments.* Parallel Computing, 2002. **28**(5): p. 827-859.
12. Beynon, M., et al., *DataCutter: Middleware for Filtering Very Large Scientific Datasets on Archival Storage Systems*, in *Proceedings of the Eighth Goddard Conference*

on Mass Storage Systems and Technologies/17th IEEE Symposium on Mass Storage Systems. 2000: College Park, MD.

13. Beynon, M., et al., *Distributed Processing of Very Large Datasets with DataCutter*. Parallel Computing, 2001. **27**(11): p. 1457-2478.
14. Beynon, M.D., et al., *Efficient Manipulation of Large Datasets on Heterogeneous Storage Systems*, in *Proceedings of the 11th Heterogeneous Computing Workshop (HCW2002)*. 2002, IEEE Computer Society Press.
15. Narayanan, S., et al., *Applying Database Support for Large Scale Data Driven Science in Distributed Environments*, in *Proceedings of the Fourth International Workshop on Grid Computing (Grid 2003)*. 2003: Phoenix, Arizona. p. 141--148.
16. Narayanan, S., et al., *Database Support for Data-driven Scientific Applications in the Grid.* Parallel Processing Letters, 2003. **13**(2): p. 245-273.
17. Saltz, J., et al., *Driving Scientific Applications by Data in Distributed Environments*, in *Proceedings of Workshop on Dynamic Data Driven Application Systems (International Conference on Computational Science)*. 2003, Springer-Verlag.
18. Weng, L., et al., *An Approach for Automatic Data Virtualization*, in *Proceedings of the 13th IEEE International Symposium on High-Performance Distributed Computing (HPDC-13)*. 2004: Honolulu, Hawaii. p. 24-33.
19. Zhang, L. and M. Parashar, *Seine: A Dynamic Geometry-based Shared Space Interaction Framework for Parallel Scientific Applications.* Concurrency and Computations: Practice and Experience, 2006. **18**(15): p. 1951-1973.
20. Wheeler, M.F., M. Peszynska, and C.N. Dawson. *Multiphysics couplings for environmental problems*. in *Proceedings of the DOD Users Group Conference*. 2000. Albuquerque, New Mexico.
21. Klie, H. and M.F. Wheeler. *Krylov-Secant Methods for Accelerating the Solution of Fully Implicit formulations*. in *SPE Reservoir Simulation Symposium*. 2005: SPE paper No. 92863, Houston, Texas.
22. Klie, H., et al. *Deflation AMG solvers for highly ill-conditioned reservoir simulation problems*. in *SPE Reservoir Symulation Symposium*. 2007. Houston, Texas: SPE paper No. 105820.
23. Stuben, K., et al. *Algebraic Multigrid Methods (AMG) for the Efficient Solution of Fully Implicit Formulations in Reservoir Simulation*. in *SPE Reservoir Symulation Symposium*. 2007. Houston, Texas: SPE paper No. 105832.
24. Wheeler, M.F. and I. Yotov, *A multipoint flux mixed finite element method.* SIAM J. Num. Anal., 2006. **44**(5): p. 2082-2106.
25. Wheeler, M.F. and I. Yotov, *A posteriori error estimates for the mortar mixed finite element method.* SIAM J. Numer. Anal., 2005. **43**(3): p. 1021-1042.
26. Banchs, R.E., et al., *A neural stochastic multiscale optimization framework for sensor-based parameter estimation.* Integrated Computer-Aided Engineering, 2007. **14**(3): p. 213-223.
27. Klie, H., et al. *Assessing the Value of Sensor Information in 4-D Seismic History Matching*. in *76th SEG International Exposition & Annual Meeting*. 2006. New Orleans.
28. Rodriguez, A., et al. *Assessing Multiple Resolution Scales in History Matching with Stochastic Neural Metamodels*. in *SPE Reservoir Symulation Symposium*. 2007. Houston, Texas: SPE paper No. 105820.
29. Liu, H. and M. Parashar, *Accord: A Programming Framework for Autonomic Applications.* IEEE Transactions on Systems, Man and Cybernetics, Special Issue on Engineering Autonomic Systems, 2006. **36**(3): p. 341-352.
30. Chandra, S., et al., *Investigating Autonomic Runtime Management Strategies for SAMR Applications.* International Journal of Parallel Programming, 2005. **33**(2-3): p. 247--259.

31. Klie, H., et al. *Parallel well location optimization using stochastic algorithms on the grid computational framework*. in *9th European Conference on the Mathematics of Oil Recovery (ECMOR)*. 0030: EAGE.
32. Parashar, M., et al. *Towards dynamic data-driven management of the Ruby Gulch Waste Repository*. in *6th International Conference Computational Science*. 2006. Reading, UK: Springer Verlag.
33. Versteeg, R., et al., *A structured approach to the use of near-surface geophysics in long-term monitoring*. Expert Systems with Applications, 2004. **23**(7): p. 700-703.
34. Gildin, E., et al. *Projection-based Approximation Methods for the Optimal Control of Smart Fields*. in *10th European Conference on the Mathematics of Oil Recovery (ECMOR)*. 0004: EAGE.
35. Klie, H. and M.F. Wheeler. *An Efficient Krylov-Karhunen-Loeve Subspace Projection Method for Stochastic Finite Element Analysis of Flow in Highly Heterogeneous Porous Media*. in *MAFELAP*. 2006. Brunel University, UK.
36. Li, H. and D. Zhang, *Probabilistic Collocation Method for Flow in Porous Media: Comparisons with other Stochastic Methods.* Water Resources Research, submitted, 2007.
37. Xiu, D., *Efficient collocational approach for parametric uncertainty analysis.* Communications in Computational Physics, 2007. **2**(2): p. 293-309.
38. Evensen, G., *Data Assimilation: The Ensemble Kalman Filter*. 2007: Springer.
39. Ristic, B., S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* 2004, London: Artech House.
40. Pearson, R., *Imperfect Data: Dealing with Contamination and Incomplete Records.* 2005: SIAM.