Application Driven Design for a Large-Scale, Multi-Purpose Grid Infrastructure

NC STATE UNIVERSITY

Featured Application:

Simulation-Optimization for Threat Management in Urban Water Systems

ABSTRACT: This application incorporates dynamic water-usage data, in real-time, into a simulation-optimization process for contamination threat management in drinking water distribution systems. The nature of this work is highly compute-intensive and requires multi-level parallel processing via computer clusters and high-performance computing architectures such as SURAgrid. The optimization component uses evolutionary computation based algorithms and the simulation component uses EPANET, a water distribution simulation code originally released by USEPA. Simulation-Optimization with EPANET is part of a multi-disciplinary, three-year NSF-funded DDDAS (Dynamic Data-Driven Application Systems) research project to develop a cyberinfrastructure system that will both adapt to and control changing needs in data, models, computer resources and management choices facilitated by a dynamic workflow design. Project Partners: North Carolina State University; University of Chicago; University of Cincinnati University of South Carolina

Understanding the Science

Contamination threat management in drinking water distribution systems involves real-time characterization of the source and extent of the contamination, identification of control strategies, and design of incremental data sampling schedules. This requires dynamic integration of time-varying measurements of flow, pressure and contaminant concentration with analytical modules including models to simulate the state of the system, statistical methods for adaptive sampling, and optimization methods to search for efficient control strategies. The goal of this multi-disciplinary research project (NSF-funded from Jan 2006 to Dec 2008) is to develop a cyberinfrastructure system that will both adapt to and control changing needs in data, models, computer resources and management choices facilitated by a dynamic workflow design.

Application Characteristics

The analytical modules (composed of thousands to millions of simulation instances that are driven by optimization search algorithms) used to simulate realistic water distribution systems are highly compute-intensive and require multi-level parallel processing via computer clusters. While data often drive the analytical modules, data needs for improving the accuracy and certainty of the solutions generated by these modules dynamically change when a contamination event unfolds. Since such time-sensitive threat events require real-time responses, the computational needs must also be adaptively matched with available resources. Grid environments composed of independent or loosely coupled computer clusters (e.g., the TeraGrid, SURAgrid) are ideal for this application as the simulation instances can be easily clustered (or bundled) into semi-independent sets, often requiring synchronization at various stages, that can be effectively executed in these environments through an intelligent allocation and monitoring mechanism which is currently being implemented as a middleware feature.

SURAgrid Deployment

The integrated simulation-optimization system developed through this project is intended to be used by the project team members during the two-year development phase of this project. Team members include application engineers at North Carolina Statue University (NCSU) and the University of Cincinnati, optimization methodology developers (NCSU and the University of South Carolina), and computer scientists (NCSU and the University of Chicago). The application engineers will test and analyze various water distribution contamination problem scenarios using realistic networks. The methodology developers will investigate various optimization search algorithms for source characterization, demand uncertainty and sensor sampling design.

The computer scientists will undertake the grid implementation, integration of various components, and performance testing in different grid environments and computer clusters, including SURAgrid. The team is using SURAgrid as an “on-ramp” to the TeraGrid. Citing specific SURAgrid benefits such as compute resource heterogeneity and low overhead to participate, the team plans to ready the application for porting to the TeraGrid by uncovering and addressing potential programming and workflow issues on SURAgrid.

Grid Workflow

To be able to run jobs on SURAgrid, the NCSU user applies for an affiliate user certificate issued by SURAgrid site Georgia State University (GSU), who has a Certificate Authority (CA) that has been cross-certified with the SURAgrid Bridge CA (BCA). Cross-certification enables SURAgrid resource sites to trust the user certificate being presented by the NCSU user and, when the SURAgrid User Administrator at GSU also creates a SURAgrid account for the NCSU user, the user essentially has single-sign-on access to SURAgrid resources at cross-certified SURAgrid sites[1]. After they've authenticated to the SURAgrid resource, the user invokes the optimization method on the client workstation that initiates the middleware that directly communicates with the specific SURAgrid resource (authenticated through ssh keys) for job submission and intermediate file movement. Currently the application needs to be pre-staged by the user, but this functionality will be integrated into the middleware. The middleware, which uses public key cryptography, will provide a seamless, python-based application interface for staging initial data and executables, data movement, job submission, and real-time visualizations of application progress. The interface uses passwordless ssh commands to create the directory structure necessary to run the jobs and handles all data movement required by the application. It launches the jobs at each site in a seamless manner, through their respective batch commands. The middleware is able to minimize resource queue time by querying the resource at a given site to determine the size of resource to request. Most of the middleware functionality has been implemented at least at a rudimentary level and efforts are now focused on better integration and sophistication.

In addition to the middleware interface described above, the application consists of two major components: one for optimization, one for simulation. The optimization component presently used on the SURAgrid is called JEC (Java Evolutionary Computation toolkit), This is the client side that drives the simulation component by calling the middleware interface. Evolutionary algorithms call multiple instances of simulations (typically hundreds) at each generation (or iteration) and require synchronization at each generation as the simulation results have to be processed before beginning the next generation. Everything on the server side (middleware, simulation component, and the grid resources) is transparent to the client.

The simulation component is an MPI C wrapper written around EPANET that does a number of things. It bundles multiple simulations (typically hundreds) and performs simultaneous execution of these on a single cluster via a coarse-grained MPI-based parallelism feature. The wrapper saves a considerable amount of processing time by not duplicating I/O and parts of simulations that are common to all simulation instances. It also has a persistent capability such that, once an EPANET job is launched, it does not need to exit until all simulation instances have been completed across all generations of an evolutionary algorithm (i.e., once the simulation outputs are written for a given generation, it can maintain a wait state until the next set of evaluations arrives from the middleware). The output files are moved back to the client workstation as the simulation progresses on the resource side. A python/TK real-time visualization tool developed by NCSU then enables visualization of the progress of the algorithm on the water distribution network. The visualization tool also creates PNG files of various stages of the output.

[1] NCSU users may also need to obtain a local account from cross-certified resource sites that have additional authorization policies.

Lessons Learned

The NCSU team has demonstrated their application and workflow on a local cluster, SURAgrid, and the TeraGrid. Testing however, has primarily been done on their local cluster due to several very challenging issues related to dynamic resource discovery and allocation that still need to be addressed. Due to a lack of more robust grid resource discovery mechanisms, commands such as "showbf" are used to estimate available grid resources. This is a cumbersome method and presents a major challenge to the application user; if available grid resources at any given time cannot be accurately predicted, it is impossible to minimize job queue wait time - the single most bottleneck for this type of application. The ability to launch and monitor multiple jobs at multiple sites is also underdeveloped at this point. Since application users cannot predict the order in which their jobs will start, they need additional middleware features in order to accommodate different launch scenarios.

Conclusion

The NCSU simulation-optimization team's ultimate goal is to produce a cyberinfrastructure that will be able to harness multiple grid resources in order to perform real-time, on-demand computing to solve water distribution threat management problems. This infrastructure will integrate various components such that the measurement system, the analytical modules and the computing resources can mutually adapt and steer each other. In the future, the team plans to improve application's functionality in grid environments by focusing on the dynamic aspects of resource demand, availability, and allocation.

For more information about Simulation-Optimization for Threat Management in Urban Water Systems contact:

Sarat Sreepathi - sarat_s@ncsu.edu NC STATE UNIVERSITY
Kumar Mahinthakumar - gmkumar@ncsu.edu

SURAgrid is developing an infrastructure for access to heterogeneous resources and support for a dynamic and diverse application set.

For more information, or to join SURAgrid:

- http://www.sura.org/SURAgrid
- maryfran@sura.org