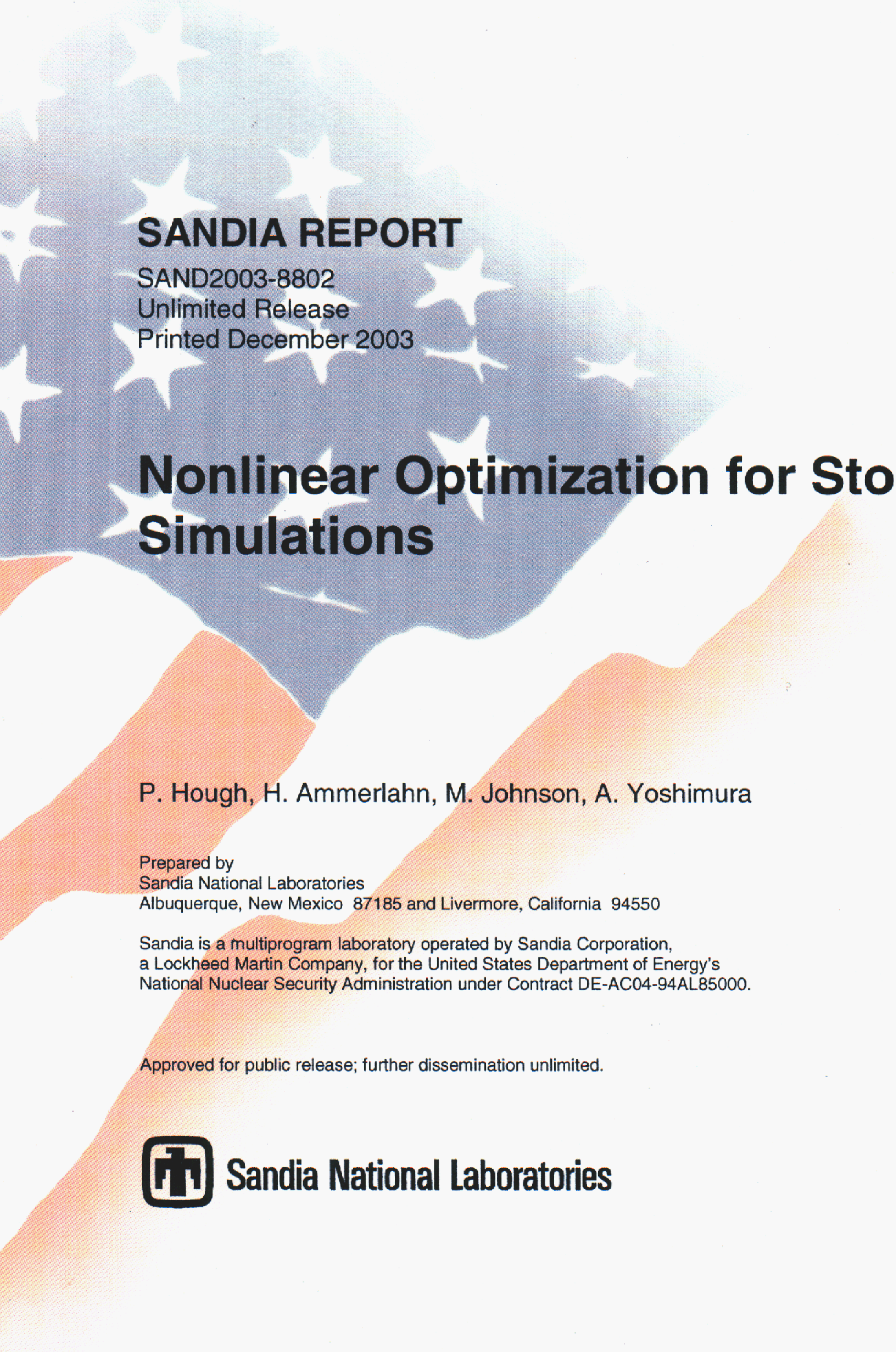


SANDIA REPORT

SAND2003-8802
Unlimited Release
Printed December 2003

Nonlinear Optimization for Stochastic Simulations

P. Hough, H. Ammerlahn, M. Johnson, A. Yoshimura

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2003-8802
Unlimited Release
Printed December 2003

Nonlinear Optimization for Stochastic Simulations

Patricia Hough
Computational Sciences and Mathematics Department

Heidi Ammerlahn and Ann Yoshimura
Systems Research Department

Michael Johnson
Systems Studies Department

Sandia National Laboratories
P. O. Box 969
Livermore, California 94551-9201

Abstract

This report describes research targeting development of stochastic optimization algorithms and their application to mission-critical optimization problems in which uncertainty arises. The first section of this report covers the enhancement of the Trust Region Parallel Direct Search (TRPDS) algorithm to address stochastic responses and the incorporation of the algorithm into the OPT++ optimization library. The second section describes the Weapons of Mass Destruction Decision Analysis Center (WMD-DAC) suite of systems analysis tools and motivates the use of stochastic optimization techniques in such non-deterministic simulations. The third section details a batch programming interface designed to facilitate criteria-based or algorithm-driven execution of system-of-system simulations. The fourth section outlines the use of the enhanced OPT++ library and batch execution mechanism to perform systems analysis and technology trade-off studies in the WMD detection and response problem domain.

This page intentionally left blank

Contents

Introduction

Nonlinear optimization algorithms have traditionally been designed to solve deterministic optimization problems. In these problems, none of the optimization variables are random and when given the same set of parameters, the function produces the same value. In the context of complex, system-of-systems or enterprise modeling and simulation applications, these assumptions no longer hold. While stochastic linear programs and stochastic integer programs are well understood and a class of methods based on those described in Birge and Louveaux [1] exist for solving them, there are significant challenges associated with developing algorithms for nonlinear stochastic programs. These are described in Wets [2]. While some algorithms, such as those described by Romero [3], do exist, they require a large number of function evaluations in order keep confidence intervals small. This is not always reasonable when the function evaluation is based on a simulation. In this work, we try to address some of these challenges and describe a simulation application in which these algorithms can be evaluated.

Optimization Algorithm Development

The general problem of interest is to minimize the average response of interest. In particular,

$$\min E(f(\mathbf{x})),$$

where E(-) is the expectation as defined in the standard statistical sense, $f(\mathbf{x})$ is the stochastic response to given deterministic input parameters $\mathbf{x}$. The algorithm described here is based on the TRPDS (Trust Region Parallel Direct Search) algorithm described in Hough and Meza [4] and basic frequentist, or sampling-based, statistical principles as can be found in most introductory statistics books such as Gonick and Smith [5].

There are two main points of interest that we consider in modifying the TRPDS algorithm to handle stochastic responses. The first concerns the gradient. Because the optimization problems of interest are based on simulations, analytic gradients are usually not available. Thus, the gradients are computed using finite differences. In addition to the numerical errors inherent in gradients computed in this manner, it is necessary to also overcome new inaccuracies introduced by the stochastic nature of the response. This is accomplished by a combination of two approaches. The first approach is computing the gradient multiple times, with the functions used in the finite-difference calculation being recomputed each time. The result is a range of gradient directions that can be used to define the search. The second approach involves sampling within the trust region using a space-filling method such as Latin hypercube or orthogonal array. This choice of directions is in contrast with the simplex-based set of search directions used in the original TRPDS algorithm. The combined group of directions provides a more robust set of directions, as they better enable capturing the randomness of the function.

The other algorithmic aspect addressed in this work is decision making. Throughout the course of the TRPDS algorithm, as with any optimization algorithm, there are a number of decision points. These decisions are based on comparisons of two values in order to determine things such as whether or not a trial iterate is acceptable and whether or not convergence has been achieved. These decisions are typically made in a deterministic manner; however, that approach is inadequate for stochastic simulations. The conversion to statistics-based decisions is fairly straightforward. The first step is to ensure that the stochastic function is evaluated multiple times at each set of trial points. Each input is independent and the same simulation is used at each input, so it is reasonable to assume that the distribution of the response at each trial point is very similar. So given a confidence level specified by the user, average responses can be compared and a determination made as to whether of not the difference between those responses is significant. There is an obvious trade-off between the confidence level and the number of simulations that are required. Allowing the user to specify that confidence will enable him to control the cost of solving the optimization problem.

As a final note, there are numerous issues that must still be addressed in the context of this algorithm. The first concerns computational expense. While this approach is less costly than other approaches, there is still a potentially unacceptable computational cost associated with this algorithm. Part of that cost can be mitigated by taking advantage of the inherent parallelism of the algorithm and multi-processor execution. Another question concerns the theoretical convergence properties of the algorithm. This will be addressed in a forthcoming technical paper. Finally, there are a number of research efforts exploring alternate approaches to handling the statistics, including other sampling methods and Bayesian statistics.

The TRPDS algorithm mentioned above is currently implemented within the OPT++ software [6]. That implementation is for deterministic problems; however, the modifications needed to accommodate stochastic functions in the manner described are minimal. In particular, the logic remains the same with the deterministic quantities replaced by statistical quantities. The required statistical samples and quantities are obtained by interfacing with the DDACE (Distributed Design and Analysis of Computer Experiments) software [7]. The more substantial software modifications are actually required for the enterprise simulation software. In particular, the OPT++ software expects to be able to execute a simulation whenever it deems necessary with whatever values it generates for the parameters of interest. This interaction should happen in an automated way. For the purposes of exploring this interaction, the Weapons of Mass Destruction Decision Analysis Center (WMD-DAC) suite of applications was identified as a potential avenue for integration. The WMD-DAC enterprise applications were designed to be interactive tools, thus relying on input or feedback from a user in order to progress. A component of this work, therefore, entailed developing a version of WMD-DAC that runs in batch mode and as a result, can be used in conjunction with OPT++. The development of the batch version of a WMD-DAC simulation application, integration with the OPT++ library, and the use of this capability are described in the following sections.

Simulation Architecture Integration

Weapons of Mass Destruction Decision Analysis Center (WMD-DAC)

In parallel with the development of a stochastic optimization algorithm, a set of interfaces were implemented to support algorithm use with Weapons of Mass Destruction Decision Analysis Center (WMD-DAC) applications. These distributed, interactive, human-in-the-loop simulations support exploration of detection and response strategies for terrorist attacks using weapons of mass destruction.

The WMD-DAC tool suite includes the following applications:

- WMD-DAC Biological Defense application,
- WMD-DAC Nuclear Defense application,
- WMD-DAC Facility Protection application, and
- WMD-DAC Borders Defense application.

These applications can be used for:

- Training/Education: Because these interactive applications allow user decisions to impact the outcome of the simulation scenario, these tools have provided the environment for several simulation-based tabletop exercises. Through these exercises, participants think through the scenarios presented, make realistic decisions, and observe the results of their actions.

- Trade-off Studies: The WMD-DAC applications are comprised of modular software elements representing constitutive components of the problem domain. These elements can include multi-fidelity models of technologies, processes, concepts of operation, etc. These models support tradeoff studies around the implementation of technologies and policies. For example, the relationship between early warning sensors that detect the presence of a biological agent attack and improved medical surveillance capabilities can be evaluated.

- Analysis: Subject to the fidelity and validation levels of the underlying models, the WMD-DAC suite of applications can be used to evaluate and optimize detection and response actions for a given scenario. To explore the problem space of these complex scenarios, a batch execution capability and an optimization capability are required.

Because of the complex, system-of-systems nature of these WMD-DAC applications, the underlying computational models incorporate statistical uncertainties representative of the corresponding real-world systems. Thus, the capabilities developed in this research can facilitate tradeoff studies and systems analyses conducted through the WMD-DAC suite of applications.

Simulation/Algorithm Integration Approach

The WMD-DAC applications were initially designed to require interactive input from the simulation user. Thus, the first requirement for the use of stochastic optimization techniques with WMD-DAC applications was the development of a generic batch execution mechanism. This mechanism is applicable to all applications developed in the WMD-DAC framework. The second requirement was the integration with optimization and sensitivity analysis algorithms. The WMD-DAC Biological Defense application was chosen to demonstrate the integrated batch execution and optimization capabilities developed through this research.

The WMD-DAC Biological Defense Application simulates an anthrax attack in a metropolitan region. An application user, playing the role of a regional Public Health Officer (PHO), is able to make decisions throughout the course of the simulation and observe the effects of those decisions. The simulation allows the user to conduct a simple epidemiological investigation through access to various public health reports such as morbidity and death reports, implement a prophylaxis strategy by activation of a simulated Strategic National Stockpile and prophylaxis distribution centers, as well as direct people to seek treatment. In addition to models representing PHO policies and data, the simulation utilizes supporting models to characterize and quantify the evolution of the event and response actions. These supporting models included threat characterization and dispersion, population behavior, historical disease trends, and resource utilization models for health care providers, prophylaxis supplies, etc. Figure 1 illustrates the models included in the application.

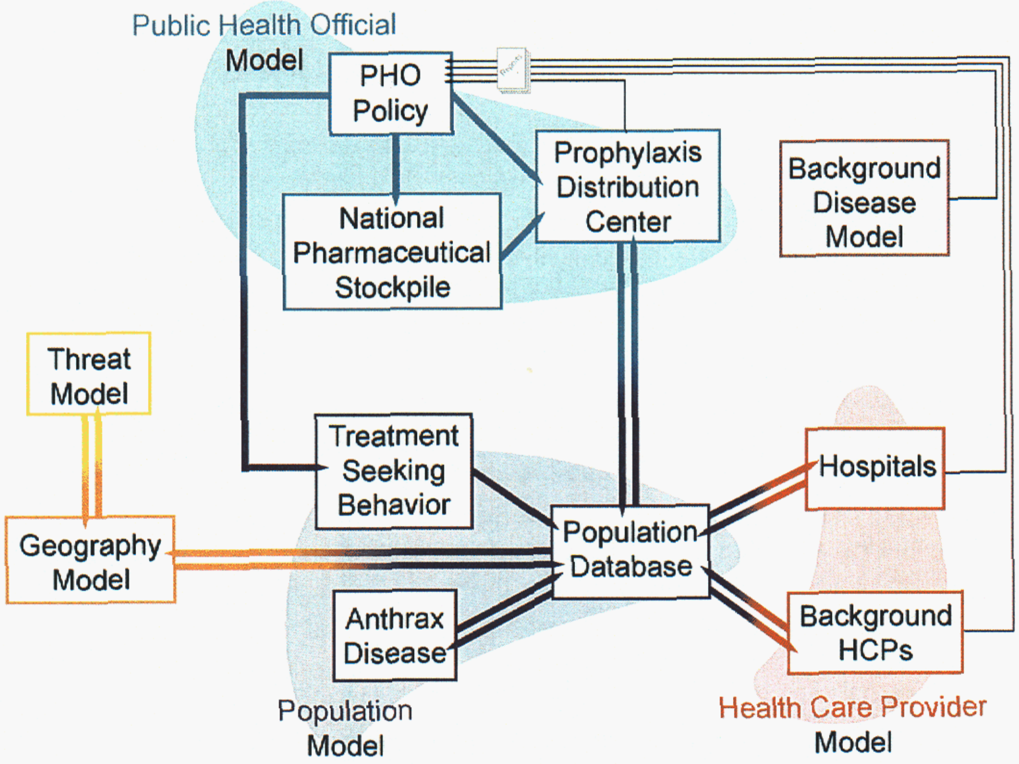


Figure 1: WMD-DAC Biological Defense Application Models

WMD-DAC Enterprise Simulation Batch Mode

Design Considerations

The WMD-DAC enterprise simulation framework was initially designed to support interactive applications. Execution of these interactive applications involves three components: (1) A collection of models, referred to as a simulation, which represents the state and behavior of the enterprise of interest, (2) a human user who interacts with the simulation during application runtime, and (3) a graphical user interface (GUI) that functions as the interface between simulation and user.

In designing a batch execution mode for the WMD-DAC framework, it was clear that the human user needed to be replaced with a new component that represents the user's behavior during runtime. What was not clear was how the new component should interact with the simulation, that is, whether or not the GUI should be included as a component of the batch mode.

If the GUI is retained, the new component would interact with the GUI much as a human user would. Input for the simulation would be generated by manipulating GUI components such as buttons, check boxes, and geographical displays. Output would be accomplished by interpreting charts, text messages, and geographical displays. This approach has the disadvantage of requiring a "double transformation" during these interactions: Output generated by the simulation in programmatic format is transformed into human-readable graphical format for the GUI, then must be transformed back before the new component can make use of the information. This double transformation would also occur for input. On the other hand, the advantage of this approach is that post-run analyses of batch runs could be greatly facilitated by archiving and "playing back" the interactions through the graphical display. This GUI playback would appear as if an invisible user were manipulating the GUI, closely emulating an interactive run of the application.

If the GUI is not retained, the new component could interact directly with the simulation, using the same programming interface that the GUI does. This eliminates the awkwardness of the double transformation and allows for more efficient runs. While the GUI playback capability would be especially advantageous in training situations, the efficiency of the batch runs is critical when performing sensitivity analyses for which very large numbers of runs are required. Since optimization is the focus of this project, the non-GUI approach was implemented.

Batch Implementation

The new component that was developed to represent user behavior is called a `BatchJob`. A `BatchJob` replaces the GUI and interacts directly with the simulation. It does this by implementing the GUI interface methods that are expected by (and invoked by) the simulation, and in return invokes methods on the simulation that would normally (in interactive mode) be invoked by the GUI.

A `BatchJob` defines setup parameters as well as runtime behavior for a single run of the simulation. Setup parameters include the name of the application, the role of the user, and a set of simulation configuration files (known as a "scenario"). Runtime behavior is embodied in a set of `Rules` and `Flags`.

Rules

A `Rule` is an "if-then" statement that represents user logic. The "if" part of the `Rule` is encoded in a `Condition` which defines a specific state that must exist for the `Condition` to be satisfied. During batch runs, `Conditions` can access all simulation information that is available to users during interactive runs through the GUI. For example, a `Condition` may be satisfied at a certain simulation time, or when a particular variable in the simulation reaches a certain value.

A `Condition` may also be a compound `Condition` that relies on the status of other `Conditions`. There are three types of such `Conditions`: `AND`, `OR`, and `NOT`. An `AND` `Condition` is satisfied when all of its related `Conditions` are satisfied, an `OR` `Condition` is satisfied when any of its related `Conditions` are satisfied, and a `NOT` `Condition` is satisfied when its related `Condition` is not satisfied. Hierarchical use of compound and simple `Conditions` enables the specification of arbitrarily complex root `Conditions` for `Rules`.

The "then" part of the `Rule` is implemented as a set of `Actions` to be performed when the `Condition` is satisfied. `Actions` can affect the simulation in every way that a user can affect it during interactive runs through the GUI. If the `Rule` has a specified delay time, the `Actions` will not be performed until the delay (in simulated time) has passed. During this delay period the `Rule` is in a "deactivated" state; it cannot be triggered. When the `Actions` are performed, the `Rule` is reactivated and may be triggered again at a future time.

Flags

While `Rules` represent the logical reasoning of a user, `Flags` represent the user's persistent memory during the simulation. A `Flag` contains a value which can be accessed or set by any of the `Conditions` or `Actions` belonging to any of the `Rules` in the `BatchJob`. There can be an arbitrary number of `Flags`, each identified by a unique name. Since they persist for the duration of the simulation run, `Flags` can be used to record events and status of the simulation and of the `Rules` themselves for future reference.

Scheduling

In interactive mode, a user can manipulate the GUI to affect the simulation at any arbitrary simulation time. However, a `BatchJob` can affect the simulation only at the time at which the `Rules` are evaluated. This results in a temporal discretization of simulated user actions in the batch mode.

In the WMD-DAC Biological Defense simulation, for example, the status of most simulation variables is updated in the GUIs on a daily basis, so user reactions typically take effect with an approximate resolution of one day. In order to implement a similar resolution in batch mode, the `Rules` are evaluated at the beginning of each simulated day, giving them the opportunity to affect the simulation on a daily basis.

Execution

`BatchJobs` can be defined by an XML file that conforms to the DTD. The XML file is parsed by the `EMFBatchClient`, a stand-alone application that acts as a client to the `EMFServer`. After parsing the XML file, the `EMFBatchClient` runs the entire set of `BatchJobs` with no further user intervention. Output from each `BatchJob` is captured in a text file identified with the `BatchJob` name and date of execution.

Optimization Library Interface

Because simulations such as the WMD-DAC Biological Defense application are not deterministic – the highly variable human response element coupled with the underlying probabilistic nature of the treatment seeking and disease evolution models produce non-linear and sometimes emergent behavior – mechanisms for evaluating optimal solutions for defined threat scenarios must be robust in the presence of uncertainty.

OPT++

Sandia National Laboratories has developed, and made available as open source, a library of non-linear optimization algorithms called OPT++ [6]. The library is available at http://csmr.ca.sandia.gov/opt++/ and was developed as an environment for the rapid prototyping and development of new optimization algorithms. The library focuses on robust and efficient algorithms for problems in which the function and constraint evaluations require the execution of an expensive computer simulation. Currently, OPT++ includes classic Newton methods, a nonlinear interior-point method, parallel direct search, etc. Between these methods, a wide range of problems can be solved, e.g. with or without constraints, with or without analytic gradients, simulation based, etc. As previously discussed in this report, the OPT++ library provided the basis for stochastic optimization algorithm development.

Java Application Programming Interface

The OPT++ library is written in C++ and supports use in most UNIX-based environments. The WMD-DAC suite of applications is written in the Java programming language and supports execution on heterogeneous computing platforms. Thus, to integrate OPT++ with enterprise simulations such as WMD-DAC (and to make it available to other Java-based applications), the library interfaces were "wrapped" to allow invocation of algorithm methods from a Java API.

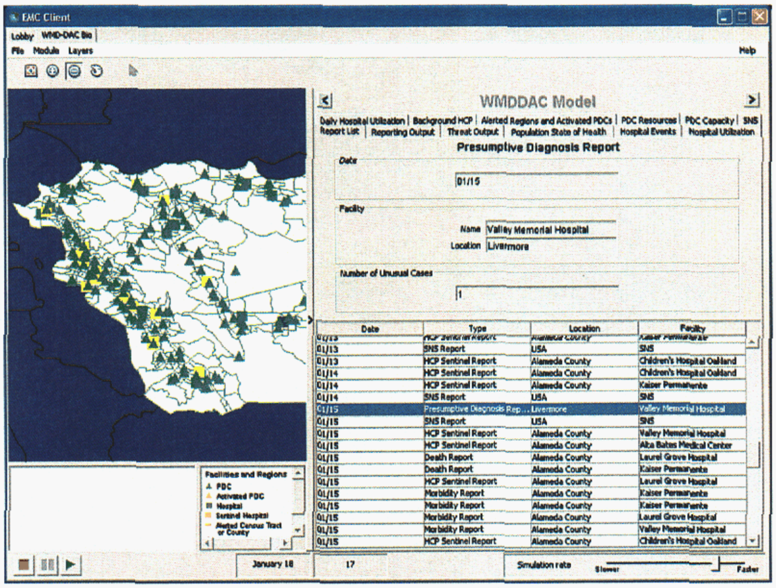
The wrapper was implemented using the Java Native Interface, a set of libraries provided by Sun Microsystems to facilitate Java-C/C++ interoperability. The Java wrapper for the OPT++ was generated using SWIG, the Simplified Wrapper and Interface Generator tool [8]. This automated process has the benefits of minimizing the amount of Java code required to implement the interface and automating the documentation process. Furthermore, changes in OPT++ are easily incorporated into the Java wrapper by simply rebuilding the code, and debugging of C++ and Java code can be done simultaneously.

By combining this wrapper and the batch execution mechanism described above, optimization algorithms can be used to drive simulation execution and vary input parameters to derive feasible solutions. Based on a specified objective function that encapsulates the scenario analysis objective, the feasible solutions can be ranked and the optimal solution(s) identified.

Simulation-Based Analysis

Thus, with the batch processing, optimization library interface, and newly developed algorithms that support problem domains in which uncertainty arises, automated execution of simulation applications to determine feasible and optimal solutions becomes possible.

This capability will be demonstrated through tradeoff studies performed with the WMD-DAC Biological Defense Application. Through geographic agility – the rapid modification of the application to incorporate data and models specific to a geographic region – this application has primarily been used to model real urban areas, including the San Francisco Bay Area, California (see Figure 2), and the greater Albuquerque area, New Mexico.



Presumptive Diagnosis Report

Date: 01/15

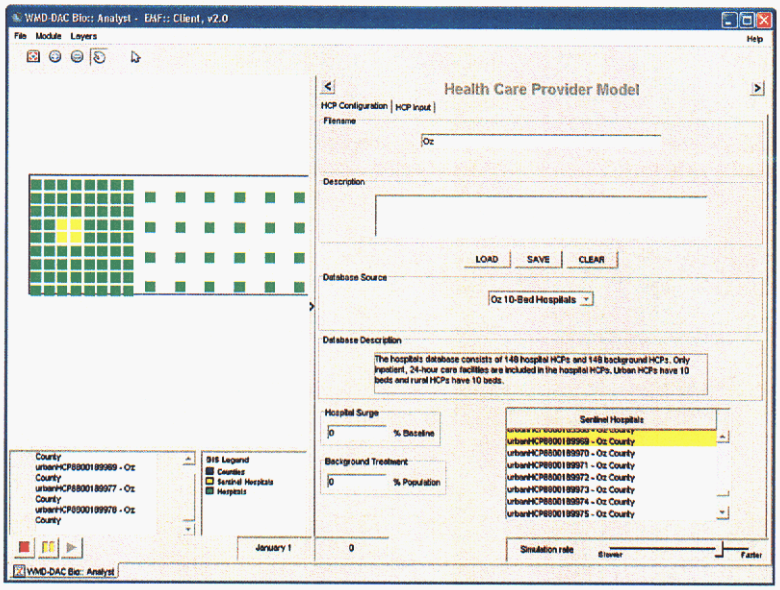
Facility
Name: Valley Memorial Hospital
Location: Livermore

Number of Unusual Cases: 1

Date	Type	Location	Facility
01/13	SNS Report	USA	SNS
01/13	HCP Sentinel Report	Alameda County	Children's Hospital Oakland
01/14	HCP Sentinel Report	Alameda County	Children's Hospital Oakland
01/14	HCP Sentinel Report	Alameda County	Kaiser Permanente
01/14	SNS Report	USA	SNS
01/15	Presumptive Diagnosis Rep...	Livermore	Valley Memorial Hospital
01/15	SNS Report	USA	SNS
01/15	HCP Sentinel Report	Alameda County	Valley Memorial Hospital
01/15	HCP Sentinel Report	Alameda County	Alta Bates Medical Center
01/15	Death Report	Alameda County	Laurel Grove Hospital
01/15	Death Report	Alameda County	Kaiser Permanente
01/15	HCP Sentinel Report	Alameda County	Laurel Grove Hospital
01/15	Morbidity Report	Alameda County	Kaiser Permanente
01/15	Morbidity Report	Alameda County	Kaiser Permanente
01/15	Morbidity Report	Alameda County	Laurel Grove Hospital
01/15	Morbidity Report	Alameda County	Valley Memorial Hospital
01/15	HCP Sentinel Report	Alameda County	Children's Hospital Oakland

Figure 2: Healthcare Reports for Alameda and Contra Costa Counties

These regional representations contain accurate demographic information, as well as attributes reflecting public health policies and procedures in those areas. To provide a flexible and easily reconfigurable environment for trade-off and systems analysis studies, however, a “generic urban region” was required. As shown below, a new region in which population demographics, health care resources (see Figure 3), public health policies (see Figure 4), and other resource and concepts of operation (CONOPS) could be easily expressed was created.



Filename: Oz

Description:

Database Source: Oz 10-Bed Hospitals

Database Description: The hospitals database consists of 148 hospital HCPs and 148 background HCPs. Only inpatient, 24-hour care facilities are included in the hospital HCPs. Urban HCPs have 10 beds and rural HCPs have 10 beds.

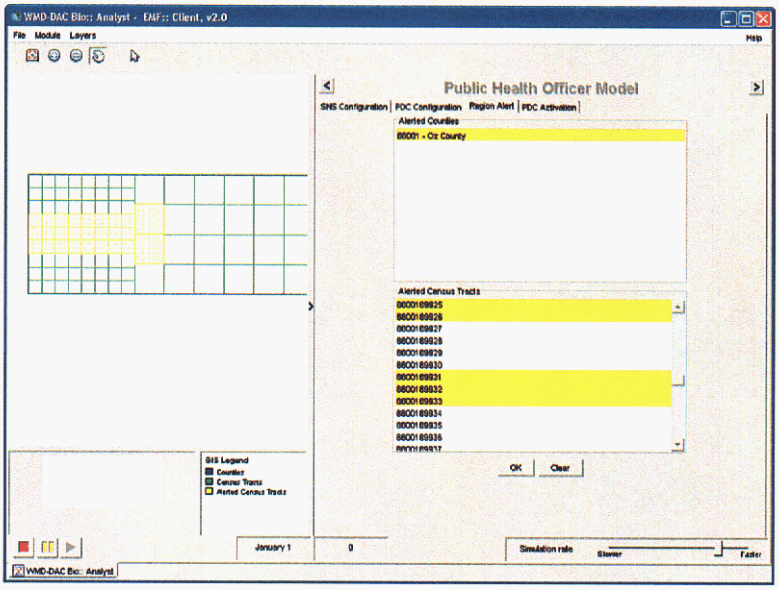
Hospital Surge: 0 % Baseline

Background Treatment: 0 % Population

Sentinel Hospitals:
urbanHCP8800189969 - Oz County
urbanHCP8800189970 - Oz County
urbanHCP8800189971 - Oz County
urbanHCP8800189972 - Oz County
urbanHCP8800189973 - Oz County
urbanHCP8800189974 - Oz County
urbanHCP8800189975 - Oz County

County
urbanHCP8800189969 - Oz County
urbanHCP8800189977 - Oz County
urbanHCP8800189976 - Oz County

Figure 3: Health Care Provider Configuration Interface

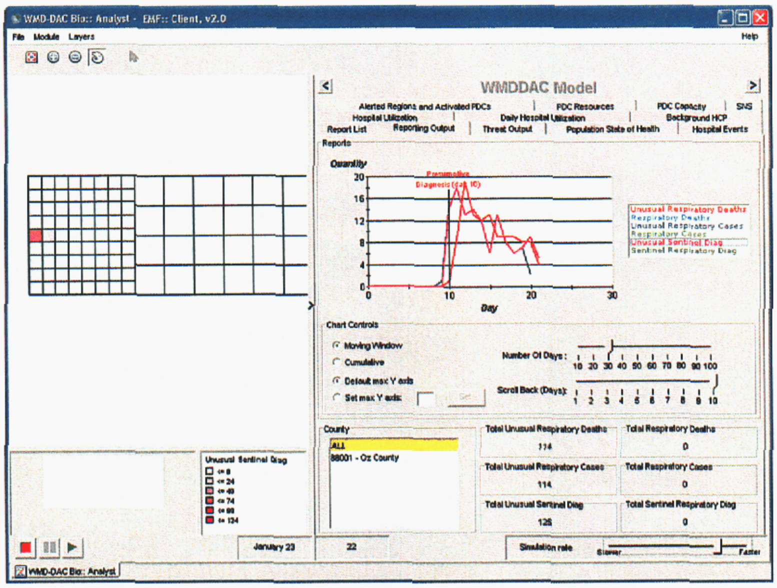


Alerted Counties:
88001 - Oz County

Alerted Census Tracts:
8800189925
8800189926
8800189927
8800189928
8800189929
8800189930
8800189931
8800189932
8800189933
8800189934
8800189935
8800189936
8800189937

Figure 4: PHO Response Input Interface

A set of studies using this application and focused on medical surveillance, detect-to-warn and detect-to-treat technologies, and optimal detection architecture and response strategies have been identified. The elements underlying these studies, to include human behavior, treatment, and disease models, and public health officer response actions, exhibit statistical variance making traditional deterministic optimization techniques ineffective. By allowing stochastic optimization algorithms to drive the execution of the simulation, factors as technology cost, number of fatalities, timeliness of attack detection, or other simulation metrics (as shown in Figures 5 and 6) can be used to identify effective technologies and strategies for a broad array of attack scenarios.



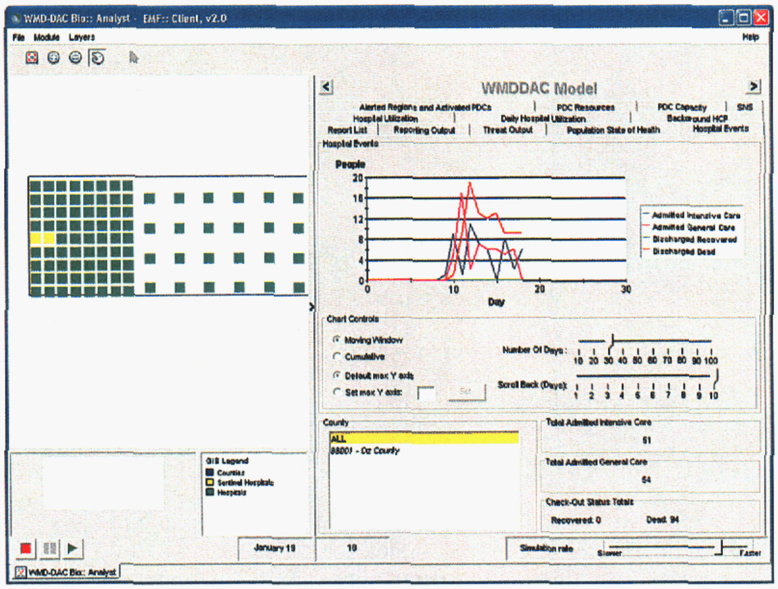
● Moving Window
○ Cumulative
● Default max Y axis
○ Set max Y axis:

County
ALL
88001 - Oz County

Total Unusual Respiratory Deaths: 114
Total Respiratory Deaths: 0
Total Unusual Respiratory Cases: 114
Total Respiratory Cases: 0
Total Unusual Sentinel Diag: 125
Total Sentinel Respiratory Diag: 0

January 23
22

Figure 5: Spatial and Temporal Distribution of Detected BioTerrorism Victims



● Moving Window
○ Cumulative
● Default max Y axis
○ Set max Y axis:

County
ALL
88001 - Oz County

Total Admitted Intensive Care: 51
Total Admitted General Care: 54
Check-Out Status Totals
Recovered: 0
Dead: 94

January 19
18

Figure 6: Hospital Resource Use Statistics

Summary

In order to reach research goals, three primary tasks were completed: (1) the development of an optimization algorithm in which deterministic decision points were replaced by probabilistic decisions in order to account for variabilities in a stochastic simulation, (2) the implementation of a batch execution mechanism for the WMD-DAC enterprise simulation framework, and (3) the creation of a Java wrapper and integration methodology that enables use of a C++ optimization library with the Java-based WMD-DAC suite of applications.

Finally, a technical implementation roadmap and set of threat scenarios were developed for demonstrating these tools and techniques in trade-off studies and systems analyses performed through the WMD-DAC Biological Defense Application.

References

[1] J. R. Birge and F. Louveaux, “Introduction to Stochastic Programming”, Springer-Verlag New York, Inc., New York, 1997.

[2] R. J.-B. Wets, “Challenges in Stochastic Programming”, working paper, International Institute for Applied Systems Analysis, 1994.

[3] V. J. Romero, “Probabilistic Ordinal Optimization versus Surrogate-Based Methods for Industrial-Scale Optimization Under Uncertainty”, presented at SIAM Conference on Optimization, 2002.

[4] P. D. Hough and J. C. Meza, “A Class of Trust-Region Methods for Parallel Optimization”, *SIAM Journal on Optimization*, Volume 13, Number 1, pp. 264-282, 2002.

[5] L. Gonick and W. Smith, “The Cartoon Guide to Statistics”, HarperCollins Publishers, Inc., New York, 1993.

[6] OPT++ Software for Nonlinear Optimization, http://csmr.ca.sandia.gov/projects/opt++.

[7] DDACE Software for Design and Analysis of Computer Experiments, http://csmr.ca.sandia.gov/projects/ddace.

[8] Simplified Wrapper and Interface Generator http://www.swig.org

Distribution

1	MS 0323	D. Chavez, LDRD Office, 1011

3	MS 9018	Central Technical Files, 8945-1
1	MS 0899	Technical Library, 9616
1	MS 9021	Classification Office, 8511 for Technical Library, MS 0899, 9616
		DOE/OSTI via URL
2	MS9201	Heidi Ammerlahn, 8112
1	MS9201	Larry Brandt, 8112
1	MS9201	Patricia Falcone, 8110
1	MS9001	Jim Handrock, 8960
5	MS9217	Patricia Hough, 8962
1	MS9201	Michael Johnson, 8114
1	MS9217	Steve Thomas, 8962
1	MS9003	Ken Washington, 8900
1	MS9201	Ann Yoshimura, 8112

This page intentionally left blank