Proceedings of the 1994 Winter Simulation Conference
ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila

MODEL SENSITIVITY ANALYSIS IN ENVIRONMENTAL EMERGENCY MANAGEMENT: A CASE STUDY IN OIL SPILL MODELING

William Lehr
Donna Calhoun
Robert Jones
Andrzej Lewandowski*
Roy Overstreet
National Oceanic and Atmospheric Administration
Seattle, Washington, 98155, U.S.A.

ABSTRACT

When simulation models are used as decision support systems in environmental emergency response, it is important to know the confidence of simulation results generated by a model. The Hazardous Material Response and Division (HAZMAT) at NOAA uses the oil weathering software package named Automated Data Inquiry for Oil Spills (ADIOS) to investigate the oil weathering process. This model combines a database of physical and chemical properties of oils, with a simulation model computing mass losses, density, viscosity, and water content of an oil spill due to evaporation, dispersion, and emulsification.

The sensitivity analysis of the ADIOS model with respect to environmental conditions and oil physical parameters is discussed in the paper. To perform the sensitivity analysis, the Object-Oriented Modeling System was used, together with the Sensitivity Analysis Laboratory, both being developed at NOAA. The preliminary results of sensitivity analysis of evaporation process are presented in the paper. According to these preliminary results it is expected that it will be possible to provide the taxonomy of oils with respect to the reliability of results generated by the ADIOS model. Sensitivity analysis of the complete ADIOS model is in progress.

1 INTRODUCTION

Stochastic methods can be used in solving environmental models characterized by only a few parameters. However, most models contain relatively large numbers of parameters and this technique is not feasible for emergency use because of prohibitive computer time. Hence, emergency response models are generally deterministic in formulation. Explicit statements of the uncertainty in these models' output become very important, especially when this must be translated into decisions about response or mitigation measures. However, this element of environmental prediction modeling is often overlooked, even in cases where the level of uncertainty in the input data is known. In other words, the output of these models represents essentially the response to the nominal values of input data and model parameters, but contains little information about their confidence level.

Reckhow and Chapra (1983) also recognize another important aspect of this problem:

> "... *Uncertainty is a problem because analysts do not like to admit that uncertainty is present in their work. To the uninitiated, the existence and admission of uncertainty in a study implies that the study has less validity and utility than does a study which exhibits no mention of uncertainty. Ironically, often just the opposite is true. Reluctance on the part of analysts to consider uncertainty is exacerbated by the fact that decision makers often cannot deal with uncertainty....*"

The application of sensitivity analysis to deterministic simulation is essential, and it provides a wide range of tools for the systematic evaluation of environmental emergency models. These areas include model behavior and validation; estimating model uncertainties; investigating the decision-making process in the face of models producing uncertain results; and identifying both fruitful and fruitless research areas.

As applied here, sensitivity analysis consists of the examination of changes in the output of a dynamical system (set of model algorithms) in response to specified changes in the system's parameters, input, and initial conditions. A fundamental element of sensitivity analysis is differentiating dependent variables of interest with respect to the model parameters, and then expansion of these variables into Taylor series about the nominal (mean) values of the parameters (Section 3 provides more detailed discussion of these

*On leave from Wayne State University, Detroit, Michigan

topics). This procedure, which generates so-called sensitivity functions, can be interpreted simultaneously as both a measure of a model's sensitivity to its parameters and its level of uncertainty (Frank, 1978; Wierzbicki, 1984; Karnavas et al., 1993; Calhoun and Overstreet, 1993). Obviously, the well-known error-propagation equation of statistics (Taylor, 1982) and the sensitivity function of system analysis are closely related and convey essentially the same information. Although the terms *sensitivity analysis* and *error analysis* are often used interchangeably here, strictly speaking, the former has more general applicability.

The application of sensitivity analysis to deterministic models is an attractive technique for model evaluation. However, the cost can be large, because it requires calculating of large numbers of derivatives of the dependent variables with respect to the model parameters. It has only been recently that efficient, accurate, and automatic differentiation methods have become available for the routine applications of sensitivity analysis (Calhoun and Overstreet, 1993; Calhoun and Lewandowski, 1994).

The Hazardous Materials Response and Assessment Division of the National Atmospheric and Oceanic Administration (HAZMAT) has developed, and is using, a number of environmental emergency models that estimate the chemical properties of spilled oils and their pathways in the environment. These models include hydrodynamic models that predict the movement and dispersion of oil in coastal and estuarine waters; atmospheric dispersion of toxic vapors; and correlation models that estimate physical and chemical properties as functions of environmental parameters. The model described in this paper estimates the spreading, weathering, evaporation, and vertical dispersion of spilled oil. The model is treated as a dynamical system whose uncertainty is treated as a sensitivity function having a time-dependent trajectory.

2 MATHEMATICAL MODELING OF OIL SPILL WEATHERING

One of the models developed by HAZMAT is an intelligent computer database and simulation system, named ADIOS (Lehr, et al., 1992, NOAA; 1993), containing properties for almost 1000 oils and oil products combined with a model for oil weathering. Environmental data are provided by the user through a menu-driven graphic interface. The format of the output is designed to provide immediate information relevant for clean up decisions. Typical output information includes changes in key physical parameters (density, viscosity, water content of emulsions) over time and mass losses due to evaporation or dispersion into the water column.

While not true object-oriented software, ADIOS was written following certain object-oriented protocols. The computational engine of ADIOS, which contains the weathering algorithms, is isolated from the oil database and platform-dependent user interface. The ADIOS system consists of the following components:

1. The oil database,
2. Preprocessor modules that perform necessary computations associated with calculating model coefficients that depend on the oil's chemical parameters,
3. Computational engine that performs simulation of oil spreading and weathering,
4. User interface and presentation module.

ADIOS simulates the following four weathering processes: evaporation, spreading, vertical dispersion of oil into the water, and emulsification of the oil by forming a water-in-oil emulsion.

2.1 Spreading Model

Spreading is unique from the other processes in that it is not linear in spill volume. That is, twice the volume of a spill does not translate into twice the surface area. The spreading rate algorithm is based on the Fay (1971) gravity-viscous spreading formula modified to include wind effects.

The area at which transition from gravity-inertial to gravity-viscous spreading takes place is considered as the initial condition for the spreading algorithm. It is assumed that, at this phase of oil distribution, the slick forms a circle whose area can be calculated by the following formula

$$A_0 = d_1 \left(\frac{V_0^5 g \Delta}{\nu_w^2} \right)^{\frac{1}{6}}$$

where V_0 is the initial volume of the spill, ν_w is the water viscosity and Δ depends on the relation between water density ρ_w and oil density ρ_o

$$\Delta = \frac{\rho_w - \rho_o}{\rho_w}$$

At the second stage of the spreading process, gravity and viscous forces dominate. Due to the impact of wind the oil spreads elliptically. The area of the oil

slick during this phase is calculated as a solution of the following differential equation

$$\frac{dA(t)}{dt} = \frac{d(\pi R_W R_F)}{dt}$$

where

$$R_F = d_2 \left(\frac{\Delta g V^2 t^{\frac{3}{2}}}{\sqrt{\nu_w}} \right)^{\frac{1}{6}}$$

and

$$R_W = R_F + d_3 U t$$

In the above formulas, R_W and R_F denote the major and the minor axes of the ellipse formed by the slick, and U denotes the wind speed.

A suspected major source of error in calculations involving the formula is the inability to estimate accurately the time-varying, over-water wind speeds that must be input by the user. Based on studies of actual spill data versus earlier spreading models (Lehr et al., 1984), it is anticipated that the algorithm would provide at best a rough approximation of the actual area of any real spill. This large uncertainty in the area has direct effects on the accuracy of the estimated dispersion and evaporation rates that depend on it.

2.2 Evaporation Model

The oil parameters that determine the evaporation rate are the initial bubble point of the oil and the rate at which the bubble point changes with the fraction of oil evaporated. These parameters are calculated from any distillation data that may be available. The basic formula (Stiver and Mackay, 1984) is

$$\frac{df(t)}{dt} = \frac{k_M^e A_S}{V_0} \exp\left(A - \frac{B}{T}(T_B + T_G f(t)) \right) \quad (1)$$

where $f(t)$ is the fraction of oil evaporated, A and B are empirically fit constants, T is the oil temperature, T_B is initial bubble point, T_G is the rate at which the bubble point changes with the fraction of oil evaporated, k_M^e is the effective mass transfer coefficient corrected according to water content, A_S is the area of a spill, and V_0 is the initial volume of a spill.

The mass transfer coefficient depends on the wind speed U and the diameter of the oil slick D

$$k_M = d_2 D^{-\frac{1}{9}} U^{\frac{7}{9}} S_c^{-\frac{2}{3}}$$

In the above formula S_c denotes the Schmidt number. The mass transfer coefficient is affected by the emulsification process. If oil is mixed with water, the value of the mass transfer coefficient decreases proportionally to the water content

$$k_M^e = k_M(1 - Y)$$

where Y denotes the water content in oil.

2.3 Dispersion Model

Vertical dispersion of oil into the water column is estimated by using a hydraulic model developed by Delvigne and Sweeney (1988). This model assumes that breaking waves at the water surface causes the oil to disperse into droplets of various sizes that are driven into the water column. Droplets smaller than a certain size are presumed to stay in the water column due to natural turbulence, rather than resurface like the larger droplets. The mass flux Q associated with the dispersion process is calculated using the following formula:

$$Q = a_1 D^{0.57} F_W (d_{max}^{1.7} - d_{min}^{1.7})$$

where d_{max} and d_{min} are the diameter of the largest and smallest oil droplets, respectively. These values are determined experimentally. In the above formula, D denotes the dissipation of energy from breaking waves calculated as

$$D = a_2 \rho_w g H_{RMS}^2$$

where H_{RMS} is the RMS wave height

$$H_{RMS} = \frac{H_s}{\sqrt{2}}$$

and H_s is the significant wave height

$$H_s = a_3 \frac{U^2}{g}$$

The value F_W is the fraction of waves breaking relative to the wave period and is calculated as

$$F_w = \frac{F_B}{T_p}$$

where T_p is the period of spectral peak of waves

$$T_p = a_4 U$$

and F_B is the fraction of waves breaking calculated as

$$F_B = a_5 U^{3.5}$$

The dispersion model depends on the choice of the maximum droplet size suspended in the water column and the estimated viscosity of the oil, both potential sources of error. However, probably the least accurately estimated variable in the model is the fraction of breaking to non-breaking waves for a given wind speed, since experiments to measure this number show wide variance in the results.

2.4 Emulsification Model

ADIOS incorporates a simple, first-order rate law proposed by Mackay (1980) to describe formation of a stable water-in-oil emulsion

$$\frac{dY(t)}{dt} = p_E \left(1 - \frac{Y(t)}{Y_f}\right)$$

where $Y(t)$ is the water fraction of the emulsion, and Y_f is the final water fraction of the fully emulsified oil. The emulsification constant, p_E, depends on the fraction of oil evaporated, the type of oil, and the wind speed. It is also affected by the evaporation process:

$$p_E = \begin{cases} 0 & \text{if } F \leq \hat{F} \\ c_3 U^2 & \text{if } F > \hat{F} \end{cases}$$

where U denotes wind speed and the value of $\hat{F}$ depends on the properties of oil. The value of this constant is an area of major uncertainty in the model.

3 METHODS OF MODEL SENSITIVITY ANALYSIS

The most commonly accepted method for sensitivity analysis of mathematical models of systems is based on Taylor series expansion of the solution of a model with respect to unknown parameters. If the system being investigated is formulated in terms of ordinary differential equations

$$\frac{dx(\alpha,\beta;t)}{dt} = f(x(\alpha,\beta;t),t;\alpha), \quad x(\alpha,t_0) = \beta \quad (2)$$

where α and β are parameters, then the sensitivity equations can be formulated as follows

$$\frac{dS_\alpha(t)}{dt} = \frac{\partial f(x(t;\alpha,\beta),t;\alpha)}{\partial x} S_\alpha(t) + \frac{\partial f(x(t;\alpha,\beta),t;\alpha)}{\partial \alpha} \quad (3)$$

$$S_\alpha(t_0) = 0$$

and

$$\frac{dS_\beta(t)}{dt} = \frac{\partial f(x(t,\alpha,\beta),t;\alpha)}{\partial x} S_\beta(t), \quad S_\beta(t_0) = 1$$

In the above equations, $S_\alpha(t)$ is the *sensitivity trajectory* with respect to the parameter α, interpreted as the derivative of the solution of equation (2) with respect to this parameter

$$S_\alpha(t) = \frac{\partial x(\alpha,\beta;t)}{\partial \alpha}$$

calculated around the nominal trajectory $x(\alpha,\beta;t)$ which is a solution of the model equation (2). The sensitivity trajectory can be used to estimate the deviation of the state variable from its nominal value when the parameter changes

$$\delta x(\alpha;t) = S_\alpha(t)\delta\alpha$$

More frequently, the *relative sensitivity trajectory* is used

$$\Delta_\alpha(t) = \frac{\alpha S_\alpha(t)}{x(\alpha;t)}$$

The relative sensitivity trajectory links the relative changes of parameter and the state trajectory

$$\frac{\delta x(\alpha;t)}{x(\alpha;t)} = \Delta_\alpha(t)\frac{\delta\alpha}{\alpha}$$

The basic technical difficulty associated with calculating the sensitivity trajectories $S_\alpha(t)$ and $S_\beta(t)$ is the necessity to implement a computer code that calculates the right-hand side of equation (3). Sensitivity analysis has an exploratory character; therefore, the model formulation is frequently changed during these experiments. Each change of a model requires calculating these derivatives and revising the code calculating the right-hand side of the sensitivity equation.

3.1 Object-Oriented Modeling System (OOMS)

Although there are many simulation tools and languages available on the market that simplify system simulation by providing means for graphically defining simulated systems (Cellier, 1991), no tools are available that provide a direct support for sensitivity analysis. Therefore, the programming environment supporting system simulation and sensitivity analysis has been designed and implemented at NOAA. The design assumption was that this environment should:

1. Provide tools for defining a system being simulated. This system is to be defined in terms of *state variables*, *submodels*, and *ports* linking these submodels, following the concepts of object-oriented simulation formulated by Zeigler (1990),
2. Provide tools for sensitivity analysis, including analytical calculation of derivatives necessary to formulate the sensitivity equations. These equations are automatically generated and augment the original equations without involvement of the user,
3. Provide tools for building the user interface, presenting of results, and experimenting with models.

The methodology for computer-supported system sensitivity analysis presented in this paper is based on the table algorithm for analytical differentiation, implemented in C++ by overloading standard arithmetic operators (Jerell, 1990). This concept is presented by Calhoun and Overstreet (1994), further extended by Calhoun and Lewandowski (1994), and implemented as the *Object-Oriented Modeling System (OOMS)*.

A mathematical model of a dynamical physical system can be modularized according to the structure of the physical system and patterns of information flow between subsystems. This makes it possible to split the equations describing such systems into a collection of interacting blocks of smaller dimensionality. These blocks are called *submodels*.

Coupled submodels describe parts of a system being modeled. This representation of systems has been explored by Zeigler in the context of *object-oriented, multifaceted system simulation* (Zeigler, 1990) and resulted in the methodological background for development of the OOMS, which supports simulation of systems consisting of the following components:

1. *Dynamical subsystems*, each described by a set of ordinary differential equations. Each dynamical subsystem has state variables $x(t) \in R^N$, input variables $u(t)$, and output variables $y(t)$,

2. *Static subsystems*, each described by a function linking input variables $u(t)$ and output variables $y(t)$,

3. *Connecting equations*, each of which defines links between selected output and input variables of different subsystems,

4. *Initial conditions connecting equations* that make it possible to compute the initial conditions of a dynamical subsystem if initial conditions for other dynamical subsystems are known.

Both static and dynamic subsystems *can depend on parameters* which determine initial conditions of the state vector components and properties of the system being modeled.

The simulation system is implemented in C++ as a library of classes supporting hierarchical system specification. Since *state variables* characterize the behavior of each dynamical system, the state variable object can be used as the smallest building blocks from which models of dynamical systems can be assembled. As objects, all state variables are instances of the **StateVar** class.

The objects corresponding to physical components of the system being simulated are instances of the **Model** class and are called *models*.

Since the system being simulated consists of many cooperating components, models corresponding to these components must exchange information about their state variables. To make this possible, *input ports* and *output port* objects can be used. These objects are instances of the **Inport** and **Outport** classes. An output port can calculate a value and make this value available for input ports defined in other models. An input port can receive a value and make it available within a model in which this port is defined.

The model and port mechanism allow the user to build models in a hierarchical way: a network consisting of models connected through ports can be encapsulated within a model with its own input and output ports.

Since equations representing the right-hand side of a differential equation being solved can depend on state variables, ports and parameters can appear in the arithmetic expressions defining these equations. Therefore, the standard arithmetic must be extended to support operations involving these elements. To achieve this, the **StateVar**, **Port**, and **Parm** classes are defined as subclasses of the same abstract class **ModelElements**. The **ModelElements** class provides necessary extensions to the standard arithmetic. The same mechanism can be used to implement other types of arithmetic, in particular, to support the sensitivity analysis by automatic differentiation (Jerell, 1990; Calhoun and Overstreet, 1992).

To perform a simulation, it is necessary to coordinate the behavior of all state variables and ports and to integrate all these components with the ODE solver. The *model manager* object performs these functions. This object is an instance of the **ModelManager** class.

Sensitivity analysis is supported by the **Parm** class. Instances of this class are *parameters*, each of which has a value and can be either *active* or *non-active*. Non-active parameter behave like a constant with defined value. Active parameter, except having a value, cause the system to generate sensitivity equations associated with this parameter. Deactivation of a parameter can be performed by sending the `deactivate()` message to a parameter object. This allows the user to specify parameters for which the sensitivity analysis will be performed. Since overloaded arithmetic operators cause additional computational overhead that is not necessary when simulation without sensitivity analysis is performed, a compile time switch is used to activate and deactivate the

code associated with extended arithmetic during the program compilation.

3.2 The Sensitivity Analysis Laboratory (SAL)

In addition to C++ classes supporting simulation and sensitivity analysis with automatic calculation of derivatives, an environment is necessary to specify values of model parameters, provide quantitative information about their uncertainty, and present the results of simulation and sensitivity analysis. Such an environment, called a *Sensitivity Analysis Laboratory*, is being developed at NOAA. This consists of a collection of software tools that provides the functionality necessary to perform sensitivity analysis in the Microsoft Windows environment.

The input data for SAL consists of the following elements:

1. A C++ program written using the OOMS classes and which describes the system to be simulated, including specification of parameters for sensitivity analysis,
2. A *problem description file*, which defines the parameters for sensitivity analysis, their ranges as well as details regarding presentation of results, like labels of boxes and sliders, plotting colors and styles,
3. Predefined *screen templates* written in VisualBasic,
4. A user interface and presentation *program template* written in VisualBasic,
5. Scripts for the AWK language interpreter to convert the C++ simulation program in a DLL Microsoft Windows library.

The first two elements are prepared by the user, and all other elements are parts of the SAL system. The following actions are performed by the SAL system:

1. The DLL interface code is appended to the source code of a simulation program written in C++ using the OOMS library. This step is performed by the AWK language interpreter, which processes the source code of the C++ program according to the information specified in the problem description file,
2. The C++ program is compiled and, as the result, the DDL library is generated,
3. The VisualBasic program is generated by the AWK language interpreter. This program is generated using the predefined templates and the problem description file,
4. The VisualBasic program is compiled.

As the result of the above procedure, the Microsoft Windows interface is generated, together with tools necessary to link this interface with the simulation program.

4 SENSITIVITY ANALYSIS OF EVAPORATION OF SPILLED OIL

Sensitivity analysis is being performed for evaporation model of the ADIOS system. The goal of this stage of research is to determine the set of critical parameters that have most strongly influence on the behavior of the oil spill evaporation model.

In the table below, a sample of results of sensitivity analysis performed for 40 oils taken from ADIOS database is presented. The sensitivity coefficients presented in this table are the average relative sensitivity coefficients for the fraction of oil evaporated, calculated according to equation (1). Results obtained for all 40 oils are presented in Figure 1 and Figure 2. It can be seen from these plots that the critical parameter influencing the sensitivity properties of oil weathering model is the initial bubble point T_B. There is no correlation between the slope of distillation curve T_G and sensitivity parameters.

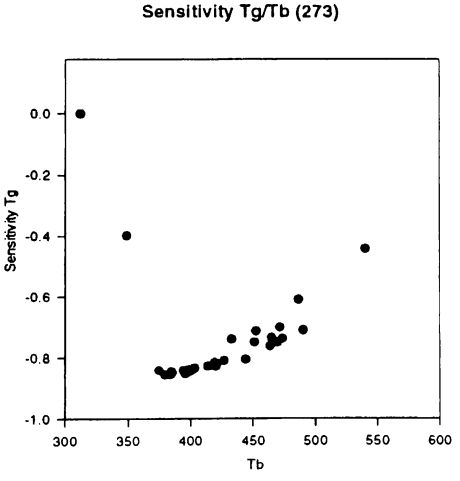

Figure 1: Dependence of the oil sensitivity parameters on the initial bubble point T_B

The above results are preliminary. A more detailed analysis of the relation between the type of oil and

Table 1: Sensitivity parameters for selected oils

Oil type	Area	Volume	T_B	T_G	Temperature	Mass Transfer
Humble	0.88	-0.88	-19.05	-0.11	19.17	0.88
Greta	0.70	-0.70	-13.71	-0.30	14.01	0.70
Yombo	0.43	-0.43	-8.08	-0.57	8.65	0.43
Prudhoe Bay	0.21	-0.21	-3.35	-0.79	4.14	0.21
Abu Dhabi	0.17	-0.17	-2.49	-0.83	3.31	0.17
Hondo	0.17	-0.17	-2.42	-0.83	3.26	0.17

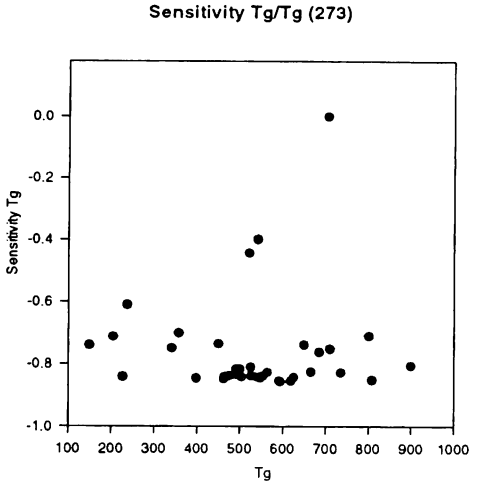

Figure 2: Dependence of the oil sensitivity parameters on the slope of the distillaction curve T_G

the model sensitivity is being performed, including the sensitivity analysis of the complete model consisting of the spreading, evaporation, emulsification and dispersion submodels. The results of sensitivity analysis will be further validated by using more oils from the NOAA oil database and by performing the analysis for various spill volume, water temperature and other environmental parameters. It is expected that the results of this analysis will make it possible to build a model linking spill parameters and oil parameters with sensitivity properties of a model, and to build a taxonomy of oils with respect to sensitivity properties of the weathering process.

5 CONCLUSIONS

It is obvious that results similar to those presented in the above table are of great importance for oil spill responders using weathering models. The simulation results generated by a weathering model for such oils as *Humble* must be interpreted with extreme care: a 1% error in determining oil temperature or properties of oil (T_G) can cause an almost 20% change in estimated evaporation rate. The situation is much better for such oils as *Hondo* where the same imprecision in oil temperature results in an error which is almost six times smaller than for *Humble*. Information about the sensitivity class of an oil could be stored in the oil database, which would allow the responder to determine the reliability of a model without performing additional, time-consuming calculations.

The results presented in Section 4 are preliminary. Research is in progress to study the sensitivity properties of a larger sample of oils. The impact of spill characteristics (amount of oil, weather conditions) and possible taxonomies of oil spills will also be studied using the NOAA spill database.

REFERENCES

Calhoun, D. and R. Overstreet. 1994. Sensitivity Analysis of a Dynamical Systems Using C++. *Scientific Computing*, Vol. 2, pp. 157-169.

Calhoun, D. and A. Lewandowski. 1994. Design of C++ Classes for Structured Modeling and Sensitivity Analysis of Dynamical Systems. Presented on the *Conference on Object-Oriented Numerics (OONSKI)*, Sunriver, Oregon, April 1994.

Cellier, F. 1991. *Continuous System Modeling.* Springer-Verlag.

Delvigne, G. and C. Sweeney. 1988. Natural Dispersion of Oil. *Oil and Chemical Pollution*, Vol. 4, pp. 281-310.

Fay, J. 1971. Physical Processes in the Spread of Oil on a Water Surface. In: *Proceedings of the Joint conference on Prevention and Control of Oil Spills*, June 15-17, 1971, American Petroleum Institute, Washington D.C., pp. 463-467

Frank, P. M. 1978. *Introduction to System Sensitivity Theory.* Academic Press.

Jerell, M. E. 1990. Automatic Differentiation in C++. *Journal of Object-Oriented Programming*, May/June 1990.

Karnavas, W. J., P. J. Sanchez, and A. T. Bahill. 1993. Sensitivity Analyses of Continuous and Discrete Systems in the Time and Frequency Domains. *IEEE Trans on System, Man and Cybernetics*, Vol. 23, No. 2.

Lehr, W. J., R. Overstreet, R. Jones and G. Watabayashi. 1992. ADIOS – Automated Data Inquiry for Oil Spills. In: *Proceedings of the Fifteenth Arctic and Marine Oil Spill Program Technical Seminar*, Edmonton, Alberta.

Lehr, W., H. Cekirge, R. Fraga, and M. Belen. 1984. Empirical Studies of the Spreading Oil. *Oil and Petrochemical Pollution*, Vol. 2, pp. 7-11.

NOAA/Hazardous Materials Response and Assessment Division. 1993. ADIOS (Automated Data inquiry for Oil Spills). User's Manual.

Mackay, D., F. Buist, R. Mascarenhas, and S. Paterson. 1980. Oil Spill Processes and Models. Report EE-8, Environment Canada, Ottawa, Ontario.

Reckhow, K. H., and S. C. Chapra. 1983. *Engineering Approaches for Lake Management.* Butterworth.

Stiver, W., and D. Mackay. 1984. Evaporation Rate of Spills of Hydrocarbons and Petroleum Mixtures. *Environ. Sci. Technol.*, Vol. 18, pp. 834-840.

Taylor, J. R. 1982. *An Introduction to Error Analysis.* University Science Books.

Wierzbicki, A. 1984. *Models and Sensitivity of Control Systems.* Studies in Automation and Control, Vol. 5, Elsevier.

Zeigler, B. P. (1990). Object-Oriented Simulation with Hierarchical, Modular Models. Academic Press.

AUTHOR BIOGRAPHIES

WILLIAM LEHR is an oceanographer and spill response group leader for the National Oceanic and Atmospheric Administration. He received his B.Sc. degree in Physics from Gonzaga University in 1970 and his Ph.D. in physics from Washington State University in 1976. He has been actively involved in oil spill modeling research for over fifteen years.

DONNA A. CALHOUN is a programmer/analyst for the Modeling and Simulation Studies Branch of the Hazardous Materials Response and Assessment Division of the National Oceanic and Atmospheric Administration (NOAA) in Seattle, Washington. She has been involved in developing models in the environmental fields, and more recently, been actively involved in developing software tools for automated sensitivity analysis of dynamical systems. She received her B.A. in Mathematics from Pomona College in 1985 and M.Sc. in Mathematics from the University of Washington in 1988.

ROBERT JONES is a scientist for the Modeling and Simulation Studies Branch for the Hazardous Material Response and Assessment Division of the National Oceanic and Atmospheric Administration (NOAA) in Seattle, Washington. He received a Ph.D. in Physical Chemistry from Indiana University in 1985 and B.Sc. in Physical Chemistry from Xavier University in 1976. He is active in modeling the transport and physical transformation of petroleum products in the marine environment, and the dispersion of hazardous vapor clouds from accidental releases.

ANDRZEJ LEWANDOWSKI is a professor in the Department of Electrical and Computer Engineering, Wayne State University in Detroit, Michigan. Currently on leave, working as the Operations Research Specialist for the Modeling and Simulation Studies Branch for the Hazardous Material Response and Assessment Division of the National Oceanic and Atmospheric Administration (NOAA) in Seattle, Washington. He received a Ph.D. in Electrical Engineering from Warsaw University of Technology, Warsaw, Poland in 1974. His research interest is in software engineering, simulation of systems, and decision support systems. He is a member of **IEEE** and **ORSA**.

ROY OVERSTREET is a physical oceanographer and research group leader in the Modeling and Simulation Studies Branch, Hazardous Materials and Assessment Division of the National Oceanic and Atmospheric Administration (NOAA) in Seattle, Washington. He received a B.A. in Physics from the University of California in 1958 and M.Sc. in Physical Oceanography from the University of Washington in 1965. His research interests include studies of circulation and modeling of coastal and estuarine waters, mass and heat transfer in atmospheric boundary layers, and analysis of uncertainty in environmental models.