

NOTE TO USERS

Page (s) not included in the original manuscript is unavailable from the author or university. The manuscript was microfilmed as received.

136

This reproduction is the best copy available.

UMI[®]

EVALUATION OF MODELLING UNCERTAINTY
AND THE ROLE OF MODEL COMPLEXITY IN RISK ASSESSMENT

By

SPENCER SNOWLING, B.Eng.Mgt

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Doctor of Philosophy

McMaster University

© Copyright by Spencer Snowling, June 2000

**EVALUATION OF MODELLING UNCERTAINTY AND
THE ROLE OF MODEL COMPLEXITY IN RISK ASSESSMENT**

DOCTOR OF PHILOSOPHY (2000)
(Geology)

McMaster University
Hamilton, Ontario

TITLE: Evaluation of Modelling Uncertainty and the Role of Model Complexity in Risk Assessment

AUTHOR: Spencer Snowling (McMaster University)

SUPERVISOR: Dr. J.R. Kramer

NUMBER OF PAGES: vii, 48 + appendices

Abstract

Numerical simulations are used to investigate the behaviour of a system, based on a set of initial conditions and assumptions about the processes involved. The complexity of a model's structure influences its output, which in turn affects predictive performance. The confidence accorded to a model's results is directly associated with the input uncertainties propagated and transformed through the model's structure. Understanding the relationship between modelling uncertainty and model complexity is important when using numerical simulation for decision-making purposes such as environmental risk assessment.

Components of uncertainty are defined by modelling error and modelling sensitivity. Modelling error is defined as the difference between a model's predictions and actual observations from the system. Modelling sensitivity is defined as the change in model output, given a change in model input.

Error and sensitivity are related to complexity in different ways. More complex models have more detailed mathematical descriptions of the system being simulated. They also have less error, but greater sensitivity, due to larger numbers of inputs (degrees of freedom) and interactions within the model's structure. Therefore, error decreases and sensitivity increases, with increasing model complexity.

Model utility, U , is used to select among models of different complexities. U is defined by combining evaluations of error and sensitivity into a single, quantitative characteristic. Utility is weighted by the modeller to reflect the relative importance of error and sensitivity.

The uncertainty/complexity relationship is determined for two systems (a simple sorption system, and a more complex 3-dimensional groundwater tracer transport system). Moderately complex models are the most utile of those studied. For the more complex groundwater system however, all models performed equally well (no difference in error), indicating that sensitivity was the only significant contributor to utility measurements. The uncertainty/complexity relationship derived for this system indicates that all the models studied might be more complex than the system warrants, and that simpler models should be investigated.

While the uncertainty/complexity relationship exists for all models, inter-disciplinary models that combine two or more discrete systems (such as environmental-economic models) are of particular interest, due to the presence of discontinuities and incompatibilities between the different model types. An inter-disciplinary model is developed, involving the integration of an input-output economic model (which describes the flow of money to and from various sectors of an economy) with a physical model that describes the environmental impact of economic activity. Simple, linear input-output models, both with and without environmental extensions, are evaluated and compared for modelling uncertainty. The effect of adding an environmental extension (and therefore changing overall model complexity) is an increase in uncertainty of model output. Uncertainty is largest at low levels of economic activity.

The uncertainty/complexity relationship is a useful diagnostic tool for the purposes of selecting among models of differing complexity. The trends of error and sensitivity can define the optimal threshold of complexity, where the improvement in fit of moving to

more complex models is not worth the increased sensitivity. If the optimal complexity threshold is not within the set of models studied, the results can be used to determine whether to move to more or less complicated models before repeating the process. This way, the methods can be used iteratively to arrive at the “best” model for the job.

Acknowledgments

While the goal of completing a Ph.D degree is to carry out independent research, it is rarely possible for a person to do so without the help of others. I would like to acknowledge the assistance of those who have helped me get to where I am today.

I would like to thank the members of my committee for their valuable feedback: Dr. D. Grundy, Dr. P. Kanaroglou, Dr. G. Patry, and especially Dr. Imre Takács, for all of his help and technical assistance with the modelling software. I would also like to thank Dr. Joe Depinto (SUNY-Buffalo) for his help with the uncertainty/complexity project. My thanks go to Dr. Pierre Brassard, for many interesting conversations over the years that have stimulated my mind and helped my research. Lastly I would like to thank my supervisor, Dr. James Kramer, for bringing me to McMaster Geology and taking me on as a grad student, as well as for all of his help over the years.

Thanks go to my close friends, Scott Smith, Nicholas Adams, Helen Manolopoulos, and Robyn Jamieson for making my working environment so friendly, and for putting up with my complaints about how slow my computer was. I also thank Pam Collins and Jim McAndrew for all their technical help, and their wisdom and perspective on many matters.

There are also three individuals who have greatly contributed to my love of science, and whose teaching skills and enthusiasm have greatly influenced my academic and career paths. I would like to thank Dennis Skowronski, Bill Stark and Earl Slack for the time they took in the past to help me see how interesting everything around us really is.

I offer my love and thanks to my parents Peter and Melva Snowling, my grandparents Mel and Thelma Swart, and parents-in-law Bill and Judy Burgess, who have all supported me both spiritually and financially, and enabled me to stay in school and be a student for so long.

And finally, my sincerest thanks and love go to my beautiful wife Marnie, without whose sacrifices I could not have completed this work. Your enthusiasm, support, love and occasional kicks in the butt have made it possible for me to achieve my most important goals. I love you and thank you for everything.

Table of Contents

1.0 Introduction	1
2.0 Thesis Objectives and Summary	3
2.1 Structure of Thesis	5
3.0 Background	8
3.1 Importance of Modelling Uncertainty Evaluation	9
3.1.1 Interpretation of Model Output	9
3.1.2 Ranking of the Importance of Model Inputs	11
3.1.3 Evaluation of Model Structure	11
3.2 Importance of Choosing the Best Model Structure	12
3.2.1 Modelling Uncertainty and Model Structure	12
3.2.2 Simulation Cost and Model Structure	13
3.3 Model Selection for Inter-Disciplinary Models	14
4.0 The Uncertainty/Complexity Relationship	16
4.1 Modelling Uncertainty as a Function of Model Complexity	16
4.2 Characterization Methods	17
4.2.1 Characterization of Model Complexity	18
4.2.2 Characterization of Model Error	19
4.2.3 Characterization of Model Sensitivity	21
4.2.4 Characterization of Model Utility	24
5.0 Results	25
5.1 Case Study I - Sorption of Metals onto Sediments	26
5.2 Case Study II - Transport of a Tracer in a Homogenous Sand Aquifer	28
5.3 Inter-disciplinary Models - Uncertainty in Environmental/Economic Modelling	32
6.0 Discussion and Conclusions	36
6.1 The Uncertainty/Complexity Relationship	37
6.2 Uncertainty/Complexity in Inter-disciplinary Modelling	42
6.3 Model Utility and the Optimal Complexity Threshold	43
6.4 Using Model Utility to Select the Best Model	43
6.4 Conclusions	46
7.0 Future Work	47
8.0 References	49

1.0 Introduction

Modelling is the use of representative constructs to simulate the behaviour of a given system. Models may be of a physical or theoretical nature, and are commonly used to investigate a system's behaviour under various conditions. Mathematical models of environmental systems can be used to evaluate different options with regard to environmental management and protection.

Using a model in a decision-making process, such as risk analysis, often involves simulating the system in question under various scenarios to determine the consequences of different courses of action. This process requires not only suitable definition of the given problem (along with appropriate methods of system evaluation), but also proper data acquisition and the selection of a suitable model. Selecting an appropriate model is not always an easy task, as the choice can directly influence the amount of data needed, and therefore, the amount of resources required to complete the risk assessment.

The use of models requires an assessment of the credibility of results just as with any scientific study. An important part of this assessment is uncertainty analysis. The uncertainty associated with a model's results is introduced by elements of input parameter variability, and is then propagated through the structure of the model. Uncertainty needs to be evaluated to have a frame of reference for the evaluation of output credibility.

Selecting one model over another can vary the amount of uncertainty encountered in a modelling exercise. How is uncertainty related to the structure of a model? This thesis investigates the relationship between model uncertainty and the complexity of a model's structure, with the goal of using uncertainty evaluation as a method for model

selection.

Before proceeding further, it is important to define terms that are frequently used throughout this text.

A *model* is a representative construct that is used to simulate the behaviour of a system. While models can vary in nature from physical to hypothetical, typically the term here refers to a series of mathematical equations that describe the flow of matter and energy to and from various states in a system.

Modelling uncertainty describes the degree to which a model's output is "known", or to what extent confidence can be attached to simulation results. It describes the propagation of variability in inputs through the model's structure, as reflected in model output. Modelling uncertainty is defined here as a combination of how close the model's predictions are to reality (goodness-of-fit), as well as the variability in the results.

Model *complexity* describes the level of detail in the relationships describing the processes in a system. Model complexity is higher in models with greater numbers of state variables, parameters and processes, and fewer simplifying assumptions.

Model *sensitivity* is defined as the amount of change in model output resulting from a change in model input. The sensitivity of a model to changes in its individual parameters is dependent on several factors, including the number and value of other parameters, the mathematical nature of the equation in which the parameter is incorporated, and the range over which the change is made.

Model *error* is defined as the quality of the output. It is a "goodness-of-fit" measurement of how well a model can simulate a given system. Error may be measured in

different ways, depending on the way a particular problem or system is defined. An example, however, would be to consider the sum of absolute differences between model predictions and observed data.

Model *structure* is defined as the series of variables, equations, and assumptions used to describe a system.

2.0 Thesis Objectives and Summary

The objective of this thesis is to characterize the relationship between modelling uncertainty and model complexity, with the purpose of using this information as a tool for model selection. Understanding the relationship between the complexity of a model's structure and its predictive performance is useful in determining how complex a structure is required for a given simulation task. A method is proposed for evaluating and selecting the "best" model structure among models of varying complexity.

Knowing the consequences of choosing a more complex model over a simpler one gives a modeller valuable information for the selection process. It can aid the modeller in answering the question, "Is it worth going to a more complex model, when a simpler (and less costly) one might adequately define the system?"

Investigation of the nature of the relationship between complexity and uncertainty requires their proper characterization. Quantitative methods to evaluate these two model attributes in a meaningful way are essential.

Inter-disciplinary models are models that involve the integration of two or more models of different disciplines. Incompatibilities and discontinuities in the different model

structures can have significant impacts on modelling uncertainty, given that overall model structure can be substantially altered by the integration of discrete modelling concepts.

The specific goals to be achieved in this research are outlined below:

1) *Development of a method for measuring model complexity.*

Model complexity must be defined in a useful and rigorous manner to evaluate the relationship between model uncertainty and the complexity of a model's structure. An index of model complexity is proposed, and used in the characterization of the uncertainty/complexity relationship.

2) *Development of a method for evaluating uncertainty based on the complexity of model structure.*

To investigate the relationship between uncertainty and model complexity, the method of uncertainty evaluation has to isolate only the uncertainty associated with structure. This can be done by minimizing contributions from elements such as input data variability and solution method effects. A method to rigorously differentiate models of varied structural complexity, based on modelling uncertainty, is proposed and used for the characterization of the uncertainty/complexity relationship.

3) *Assessment of uncertainty with real environmental systems and dynamic models.*

A series of simple sorption models and more complex groundwater transport models (of varying complexities) are evaluated against real systems to determine the relationship between uncertainty and model complexity for each system. The

uncertainty evaluation methods cited above in objective 2 are used.

Environmental-economic inter-disciplinary models are a specific subset of the overall scope of models in general. The study of the relationship between modelling uncertainty and the complexity of model structure also focuses on these models in order to highlight the relationship where two or more models of different disciplines are combined. In particular, the objective related specifically to inter-disciplinary models is:

- 4) *Characterization of the effects on modelling uncertainty of integrating environmental models into economic models, using the Leontief input-output model as a case study.*

Environmental issues relating to economic impact are studied through the use of Leontief input-output (I/O) models which describe the flow of money to and from various sectors of the economy. Adding an extension to this model, which describes environmental processes both in terms of physical and economic units, relates environmental and economic impact. The Monte Carlo Method is used to evaluate the uncertainty of regular input-output models, and to evaluate how uncertainty changes when this type of environmental extension is added to a traditional Leontief input-output model.

2.1 Structure of Thesis

The above topics are addressed by drawing on results and conclusions of four

papers included as appendices I-IV. These papers are referred to by their roman numerals in the thesis text, and are listed below along with a statement of the major contribution of each paper. The papers consider:

- I. creating an index of model complexity
- II. characterizing the relationship between modelling uncertainty and model complexity
- III. integrating an environmental model into an economic model, and
- IV. evaluating the effects on uncertainty of integrating environmental models into economic models.

These four papers address the objectives outlined earlier. Paper I describes the methods used to characterize complexity that are employed in the study in Paper II. The input-output model developed in Paper III uses the environmental extensions that are investigated in Paper IV, and the model itself is evaluated in this thesis using the methods outlined in Paper IV. Lastly, the results obtained from Papers II and IV show how changing model structure alters the way uncertainty is generated in model results. Knowledge of the relationship between model structure and modelling uncertainty can be used as a tool to aid in selecting a model of appropriate complexity for a given task.

I. Index of Model Complexity

This paper develops a new method for quantifying model complexity. Model complexity considers model structure and the level of detail in the mathematics describing

each process. I_c , the index of model complexity, is calculated from the equation,

$$I_c = \sum_{j=1}^N \sum_{i=1}^{n_j} p_i r_i$$

where N = number of state variables in the model, n_j = number of processes associated with state variable j , p_i = number of parameters used to describe process i , and r_i = number of mathematical operations used to describe process i . Several biological wastewater models of varying complexity are used to illustrate and to evaluate I_c . The index of model complexity effectively differentiates between models of different structures (e.g. carbon-nitrogen models have lower I_c values than carbon-nitrogen-phosphorus models), and is well correlated with computational requirements for performing computer simulations.

II. Evaluating Modelling Uncertainty for Model Selection

This paper investigates the relationship between modelling uncertainty and model complexity (as defined in Paper I). Methodology developed for isolating and measuring the uncertainty associated with model structure is used to assess models of varying complexity for two environmental systems: the sorption of metals in a sediment solution, and a 3-dimensional tracer transport in a homogeneous sand aquifer. An index of model utility is proposed to quantify the uncertainty information for the purposes of model selection. In both case studies, moderately complex models were shown to have the highest utility of those tested.

III. Input-Output Modelling and Groundwater Remediation

This paper expands the use of a model feedback effect for use in the agricultural

sector, through an environmental extension to a standard input-output model. Pollution cost (from pesticide use) is incorporated back into the economy by evaluating the production associated with remediation of agricultural pollutants. When the model is applied to a Southern Ontario agricultural region, up to a 5% increase in production of the services sector is predicted when remediation is included. While this paper does not directly deal with uncertainty, it develops the basis for the environmental feedback extensions used in the uncertainty study in Paper IV.

IV. Uncertainty Analysis in Input-Output Modelling: Effects of Internal Feedbacks

By adding a feedback effect to a standard input-output model (such as the model in paper III), uncertainty is increased, depending on the level of exogenous (external to the model) demand. In this paper, the Monte Carlo method is used to investigate changes in uncertainty with the integration of environmental extensions into a hypothetical Leontief input-output model. Modelling uncertainty is highest at low levels of demand, where remediation production is equal to or greater than exogenous demand.

3.0 Background

Modelling uncertainty can be defined in many different ways, but simply put, it describes how well the solution to a model is “known”. Uncertainty is a property of several factors, including model structure, variability of input data, and solution methods. It can be measured in different ways, such as simple statistical measures of variance,

analytical calculation of propagation of error, or the Monte Carlo method (Kalos and Whitlock, 1986).

Modelling uncertainty directly affects the ability to use a model for purposes such as environmental risk assessment. It is important to take into account the uncertainty of results when using model output to draw conclusions about the behaviour of a system.

3.1 Importance of Modelling Uncertainty Evaluation

Evaluating modelling uncertainty can be useful in three ways; for interpretation of model output, as a tool to determine which model inputs are most important (enabling resource allocation for data acquisition to be optimized), and for assessment of model structure for model evaluation or selection. Each of the uses of uncertainty information differs in purpose and methodology, and is described in the following sections.

3.1.1 Interpretation of Model Output

Using models in risk assessment necessitates the evaluation of uncertainty in order to quantify the variability of the model output. Dakins, et al. (1994) outlined a decision framework for risk-based environmental remediation where the evaluation of uncertainty plays a key role. The authors noted that full uncertainty analyses are not commonly performed in the decision making process. Incorporation of an uncertainty analysis into risk assessment can have a substantial economic value, as shown with an on-going case study.

Reckhow (1994) expanded upon this idea by discussing the role of uncertainty in management decisions, and evaluated several water quality models with respect to their

predictive abilities and the difficulties encountered in uncertainty analyses. He considered the notion that models must be complicated in order to achieve accurate simulations. This idea is in contrast to Beck's (1987) comment that "Most of the evidence suggests that current models of water quality, in particular, the larger models, are easily capable of generating predictions to which little confidence would be attached". In addition, Reckhow stated that the presence of a quantified uncertainty measurement enhances the quality of model output. He also noted that uncertainty evaluation is most useful in directing modelling efforts when both the modellers and decision-makers use it to structure analysis, and as a context in which to present results.

Decision-makers, however, need to be careful in their interpretation of "validated" model results. Several methods exist for the validation of model output, although the outcome of the use of such methods has been questioned. Oreskes et al. (1994) described the difference between model validation, verification and calibration. They noted that these terms are often used synonymously, which is incorrect. Verification usually refers to the confirmation of numerical solutions to models, whereas validation and calibration are methods used to compare simulated results to observed data. The use of these terms to imply that models are in some way true representations of reality was also questioned.

Konikow and Bredehoeft (1992) noted that the terms "verified" and "validated" are often incorrectly construed by users to mean that models are "true". Harremoës and Madsen (1999) supported this idea by warning that current calibration/verification methods can give an exaggerated sense of certainty regarding model results.

3.1.2 Ranking of the Importance of Model Inputs

Dakins, et al. (1994) identified modelling uncertainty analysis as a suitable method to identify which parameters are most important in a risk assessment. A case study from New Bedford Harbour, Massachusetts illustrated that incorporation of an uncertainty analysis in risk assessment could be used to estimate the amount of resources that should be allocated to reduce the study's uncertainty. This assessment also showed how the resources might be best spent.

3.1.3 Evaluation of Model Structure

The structure of a model plays a role in the evaluation of its associated uncertainty. Beck (1987) provided an excellent review of uncertainty analysis in water quality modelling. The author reviewed methods for uncertainty analysis, but more importantly, investigated the role of model identifiability in uncertainty. Model identifiability refers to the ability of a model to uniquely define the behaviour of a system. Beck concluded that meaningful calibration or parameter estimation is difficult for models that lack model identifiability. The inability to explicitly falsify constituent model hypotheses makes validation of increasing complex models difficult.

Therefore, following the principle of Ockham's Razor, simpler models are often chosen in favour of more complex ones, partly because they are easier to validate (Pearl, 1978). Morgan and Henrion (1990) argued however, that the development of large, complex models can aid in identifying and directing research in areas where knowledge is lacking.

3.2 Importance of Choosing the Best Model Structure

Beck, Reckhow and Dakins all pointed out the need for proper evaluation of modelling uncertainty when simulation is used as a decision-making tool in environmental risk assessment. The amount of uncertainty propagated through a model is a property of its structure. Given that sometimes there are many models of varying structures for a system, modellers are often faced with the task of selecting among models that describe the behaviour of a system to varying levels of complexity. This choice may not be a simple one. The choice of model structure influences not only the amount of uncertainty encountered in the simulation exercise, but also the amount of data required to properly calibrate the model to the system being simulated.

3.2.1 Modelling Uncertainty and Model Structure

The uncertainty associated with model structure can influence the quality of model results. If a model is far more complex than the system it is simulating, the variability of the output can be significantly large compared to the answer it provides. For example, highly sensitive models which produce output with a variability of $\pm 100\%$ have only limited application.

Beck (1987) noted in his review of several extant water quality models that there is the capability for larger, more complicated models to generate output which is accorded little confidence. He stated that simpler, well-identified models reflect only observed past behaviour, and therefore can be accredited with some amount of confidence. Conversely, complex models, which may possess excellent predictive abilities on a case-by-case basis, can rely on “apparently redundant or ambiguous hypotheses”, and therefore cannot be

accorded much confidence. The resulting hypothetical paradox is that simple models can confidently provide a “wrong” answer (or “right” answer for the wrong system), whereas complex models can provide a “right” answer with very little confidence. To be able to select an appropriate model from the middle ground is the goal of the model selection process (Smith et al., 1999).

3.2.2 Simulation Cost and Model Structure

The cost of data acquisition is directly proportional to the amount of input parameters and state variable initial conditions. A more complex model has more parameters and state variables, and more processes to calibrate. The most costly aspect of modelling is the collection and analysis of field data which is used as input or to calibrate the model. The CPU requirements of a model have been considered a cost in the past, but due to the cheap computing power currently available (relative to the costs of manual or automated field data gathering), CPU time is generally no longer considered a significant cost.

Usually data gathering is governed by the complexity of a model’s structure, in that the model will only be as complex as the data used to calibrate it. For example, kinetic data as well as steady-state data is required to calibrate a kinetic model. Calibrating only at steady-state would limit the effective use of the model, because kinetic behaviour would not be observed, and kinetic parameters would be essentially arbitrary.

In summary, choosing a simpler model can effectively mean a decrease in the cost of collecting and analyzing data needed to prepare the model to capture the behaviour of the system.

3.3 Model Selection for Inter-Disciplinary Models

It is hypothesized that the relationships between modelling uncertainty, complexity and cost are also valid for the subset of interdisciplinary models. Adding environmental extensions to economic models alters the structure of the overall model, thereby changing the way uncertainty is generated, as well as increasing the amount of data required for a simulation. For this study, an environmental model is incorporated into the Leontief input-output (I/O) model (Miller and Blair (1985)).

A standard input-output model describes the relationships between sectors in an economy, and can be used to predict the amount of overall production associated with a given level of exogenous demand for each sector's product. Miller and Blair (1985) presented a concise review of input-output modelling, including the development of extensions for environmental analysis. Linear relationships between production and various other economic factors are used to estimate total levels of employment, resource use, and pollution generation.

Leontief (1970) developed the use of environmental feedback extensions to regular I/O models. Leontief's proposed environmental extension determines the amount of pollution generated from production in the economy, and then compares it to an "acceptable" level of pollution. The amount of pollution in excess of the acceptable level is removed from the environment by an environmental remediation sector. The production from that sector is then added back into the economy. This new production then generates further production in other sectors, and the result is more pollution. This pollution is then remediated, and the cycle is repeated until the system reaches equilibrium.

Feedbacks were further expanded by Chen (1973), who developed a block diagram system of illustrating input-output models. Chen also developed methods for internalizing the costs of pollution control, whereby the cost of pollution control methods are passed on to the customer.

Xu, et al. (1994) extended Chen's ideas for use in urban stormwater quality planning. The production associated with the remediation of pollution generated by various types of land use was incorporated into the economy. This same type of approach is used in Paper III, where the pollution generated from the agricultural sector is remediated, and the production associated with the remediation is then incorporated into the economy. Xu et al. (1994) and Paper III both incorporate Chen's box diagram illustration method.

Adding an environmental feedback effect necessitates the acquisition of data describing remediation production (in addition to data on the amount of pollution generated at various levels of production). The addition of the feedback effect changes the overall model complexity, and therefore, also affects the uncertainty of the new, coupled model.

In summary, the relationship between model structure and modelling uncertainty is important for model selection. Both the amount of uncertainty in a model's results and the cost of providing input data are a property of the complexity of a model's structure. When choosing among several models of different complexities, selecting one structure over another has an impact both on predictive performance as well as simulation cost.

4.0 The Uncertainty/Complexity Relationship

The predictive performance of a model is influenced by many factors, including: complexity of model structure, input data variability, calibration data variability and numerical solution methods. Following the ideas of Beck and Reckhow, uncertainty and complexity are used to evaluate predictive performance.

4.1 Modelling Uncertainty as a Function of Model Complexity

It is important to properly describe uncertainty to identify and characterize the relationship between modelling uncertainty and model complexity. In this study, uncertainty is characterized by two properties of a simulation: sensitivity and error.

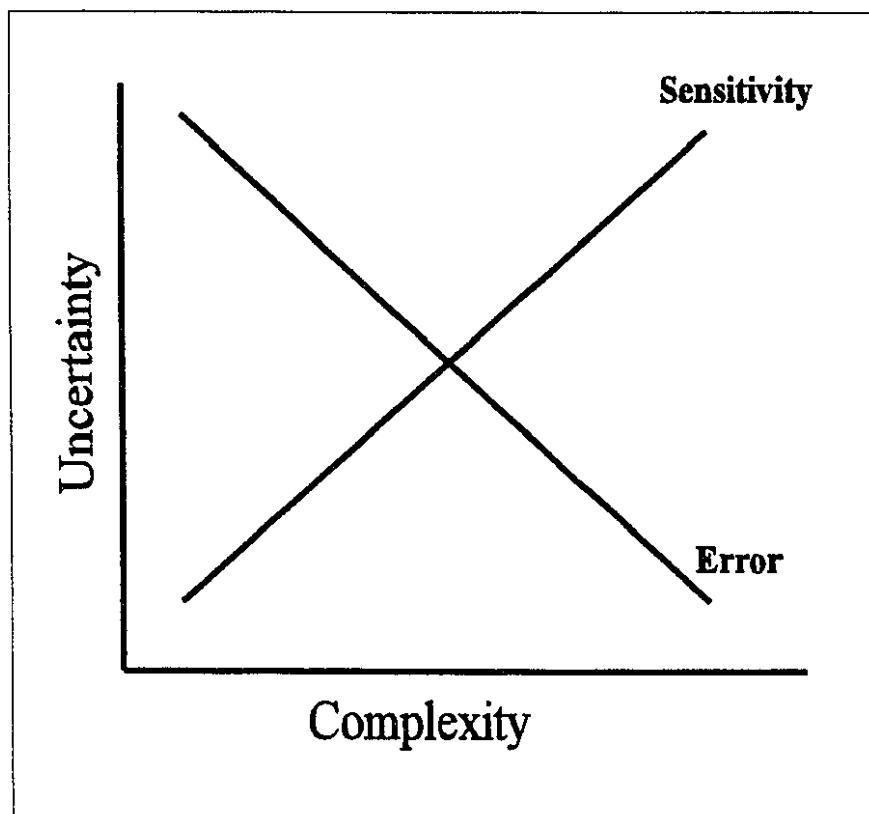


Figure 1. Hypthetical Uncertainty-Complexity Relationship.

Methods used to quantify these characteristics are described in section 4.3.

Figure 1 schematically summarizes the hypothesized relationship between uncertainty and complexity, by illustrating the trends of sensitivity and error with increasing model complexity. As model complexity increases, models have more parameters, state variables and more complex mathematics. They also have more degrees of freedom, and therefore should be more sensitive. More complex models are hypothesized to simulate reality better since more mechanisms are considered, and in a more sophisticated manner. Therefore, error (the difference between simulation and observation) should decrease with increasing complexity. In summary, the hypothesized relationship proposes that simpler models give a less realistic simulation, but with a smaller envelope of sensitivity, whereas more complex models can give a more realistic simulation, but with a greater envelope of sensitivity. The trends shown in Figure 1 are a general hypothesis only. The trends may have different slopes (including zero), and may also be non-linear.

The optimally complex model for any given system is defined by the relationships shown in Figure 1. Minimizing both model error and sensitivity improves simulation (reduces uncertainty), so therefore the optimal model will lie somewhere in the middle of the ranges of complexity shown in Figure 1. Methods of using sensitivity and error information to determine the optimally complex model are included in section 4.2.4.

4.2 Characterization Methods

Methods for the quantitative characterization of uncertainty and complexity are

developed as tools for model evaluation and selection. Quantification of model complexity, model error and model sensitivity is carried out through the methods described in sections 4.2.1 to 4.2.3. In each case, the objective is to differentiate between models for model selection. Therefore, the methods are designed to produce *relative* measurements, rather than *absolute* measurements.

4.2.1 Characterization of Model Complexity

Smith and Vaughan (1980), Halfon (1983) and Kohler, et al. (1996) performed simple assessments of models to address the issue of an appropriate model complexity. In these studies however, the complexity of the models was not explicitly characterized. Few methods exist in the literature to characterize the complexity of a dynamic model, and no one method has been accepted as a standard. Bosserman's index of connectivity, c (Bosserman, 1980), was used by Halfon (1983) to measure the complexity of ecosystem models. It is calculated by determining the number of direct and indirect links between states in an ecosystem. It does not, however, take into account the complexity of the mathematics involved in the model.

Paper I outlines a method to characterize complexity for any type of dynamic model. The index of complexity, I_c , quantifies both the structure of the model, as well as the mathematical complexity. I_c is shown in Eq.1 below:

$$I_c = \sum_{j=1}^N \sum_{i=1}^{n_j} p_i r_i \quad (1)$$

where N = number of state variables
 n_j = number of processes flowing to or from state variable j
 p_i = number of parameters used to describe process i .
 r_i = number of mathematical operations used to describe process i .

By summing across each state variable, processes that are reversible between two state variables are counted twice, to emphasize the complexity of such processes. The most straightforward method of calculating I_c is through the use of the Petersen matrix (Petersen, 1965, Henze, 1987), which uses a simple matrix notation to illustrate the relationships between model flows and processes.

I_c is calculated for a series of wastewater models of varying complexity (specific details of the models and calculations can be found in Paper I). Models of similar type are grouped together when evaluating $\log(I_c)$. Wastewater models consisting only of carbon components had a $\log(I_c)$ value of approximately 1.7. The addition of nitrogen components increases $\log(I_c)$ to the 3.0 to 3.6 range. Addition of phosphorus components to the carbon-nitrogen models increases $\log(I_c)$ values to the 3.9 - 4.0 range.

Computational requirements (in the form of CPU usage) for model solutions are well correlated with I_c . A linear relationship between $\log(I_c)$ and CPU time required (for a standard dynamic simulation) was shown for the models used in the study ($r^2 = 0.97$). The use of this index as a measurement of complexity allows for appropriate characterization of models, as well as a rough estimate of computational effort required to solve the model.

4.2.2 Characterization of Model Error

It is necessary to isolate model structure as an independent variable to evaluate the uncertainty that is specifically associated with structural complexity. Several models of

different structural complexities are used to simulate the same system in the case studies in Paper II. The only difference among the models is their structural complexity. All other model properties are held the same (such as boundary conditions, spatial resolution, and integration methods). Differences in error evaluation among the different models are then a property of structural complexity only.

The Jackknife method (Miller, 1974) is used to evaluate the error aspect of modelling uncertainty. The method is useful for situations where there is minimal data for a system. In typical calibrations, one set of data is used for calibration of the model, and a second, independent set of data (from the same system) is used for model evaluation. The time series from the Borden Data Set (Roberts and McKay, 1990), used in case study II in Paper II, are highly spatially resolved, but there are data for only one event, so the Jackknife method is used. For each of the five temporally coincident data sets (representing the concentration at five different points in the system over time), the model is calibrated with the other four data sets, and then evaluated against the fifth. This is repeated four times, switching the sites used for calibration and evaluation, so that all the data have been used for both calibration and evaluation. The overall error measurement is equal to the average value of the objective function (which describes the difference between the observed value and the model prediction) during the five evaluations.

There is only a single time series of data for case study I. Error evaluation is equal to the value of the objective function used in the optimization with observed data. The objective function is equal to the squared differences between prediction and observation, summed across all data points in the time series.

4.2.3 Characterization of Model Sensitivity

Several models of different structural complexities are used to simulate the same system during the sensitivity characterization (similarly to error characterization). The only difference among the models is their structural complexity. All other model properties are held the same (such as boundary conditions, spatial resolution, and integration methods). Differences in sensitivity evaluation among the different models is then a property of structural complexity only.

Recalling that sensitivity is the amount of variability in model output caused by a change in model input, sensitivity characterization is based on quantifying this output variability. The Monte Carlo method estimates model sensitivity through the use of repeated simulations to observe the behaviour of a model. For each model input parameter, a distribution of values is assumed (arbitrarily, from empirical data, literature, or otherwise). A series of values is independently sampled from each of the distributions. The model is then solved using each of the sets of values. The output from each simulation is collected, and the combined data are evaluated in order to determine the spread across all the runs. A typical evaluation of spread might involve calculating the mean and standard deviation for each time step in the model's output (similar methods are described in Kalos and Whitlock, 1986, Dakins et al., 1994).

As described above, the results of a Monte Carlo analysis depend on the parameter distributions used. As stated earlier, the purpose of sensitivity characterization in this study is to quantify changes in sensitivity in models of different complexities. The choice of parameter distributions should serve to distinguish models from each other, based on

structural complexity. Arbitrarily chosen parameter distributions may only describe model behaviour in specific ranges that are limited by the distributions. These limited ranges of behaviour may or may not cover the optimal behaviour of the calibrated model as seen in the error characterization. A much better method to determine appropriate parameter distributions is the “reverse” Monte Carlo method.

The reverse Monte Carlo method exchanges the positions of input and output in the analysis. Instead of starting with a distribution of input parameter values and then determining a series of resultant output, the reverse method starts with many sets of output data and determines input. Often these sets of output are created by adding randomly generated noise to a single data set many times, to create a series of noisy output data sets. The model is then fit to each one of the sets of output data by varying the input parameters being studied. The result is a set of parameter values for each of the noisy output data sets. The values for each of the parameters are collected, and a distribution is calculated.

The sensitivity characterization performed in the case studies in Paper II uses the reverse Monte Carlo method to determine parameter distributions. The noisy output data sets are created by adding randomly generated noise to the “clean” time series data as taken from the Borden Data Set. The noise added to the original data set incorporates both independent random noise (added to each point), as well as a random, noisy bias across multiple data points.

The model is fitted to each of the noisy sets, and the parameter distributions are determined. These parameter distributions are then independently sampled to create a

series of input parameter values. The parameter values are used in the normal “forward” Monte Carlo test, where the mean and standard deviation are calculated for each timestep of the multiple series of output data. An output variability “envelope” can be plotted as plus/minus one standard deviation around the mean, at each timestep in the series. The overall sensitivity measurement is proportional to the area of the envelope.

Performing a reverse Monte Carlo test followed by a regular Monte Carlo test highlights the differences in sensitivity between models with different structures. A simple example illustrates this concept. If a deterministic model is fitted to a time series of data by adjusting a single parameter, and that resultant parameter value is then used as input, the output that is generated will be exactly the same as the original time series. If a series of values for the parameter are generated using the reverse Monte Carlo method as described earlier, and those values are then used as input (for repeated simulations), the model will produce a series of output data sets that exactly match the original series.

If however, the input values are not exactly those that were recorded in the reverse Monte Carlo method, but rather sampled from a distribution determined from the recorded values, the output from the model will be similar to the original noisy output data. The variability in the results is a property of the structure of the model. Comparing the output results from different models highlights the differences in sensitivity that are caused by the differences in model structure. This concept is then extended to a deterministic model with more than one input parameter.

The model sensitivity measured with this method is relative (for comparing models of different structural complexities), and reflects the sensitivity of a model calibrated to

reflect the system used in the study. This method is used in both of the case studies in Paper II.

4.2.4 Characterization of Model Utility

An index of model utility is proposed to combine the error and sensitivity evaluations into a single, quantitative measurement. Model utility, U_i , is calculated for each of the models in the two case studies, to provide a tool for selecting the “best” model. U_i is calculated as the weighted and normalized combination of error (E), and sensitivity (S), as shown in Eq. (2):

$$U_i = 1 - \sqrt{\frac{k_s S_i^2 + k_e E_i^2}{(k_s + k_e)}} \quad (2)$$

where U_i = utility index for model i
 S_i = sensitivity value for model i (relative to maximum sensitivity)
 E_i = error value for model i (relative to maximum error)
 k_s, k_e = weighting constants for sensitivity, error

k_s and k_e are chosen to weigh sensitivity and error relative to each other. If k_s is a larger value than k_e , sensitivity will be given a greater weighting in the utility measurement. For all studies in this thesis, sensitivity and error are weighted equally. A higher value of U_i indicates a model with low error and sensitivity, compared to others in the same study. Therefore, the “best” model (based on the proposed uncertainty evaluation) has the highest U_i value.

5.0 Results

The studies carried out in Papers II and IV quantify the relationship between model complexity and uncertainty by analyzing models of different complexities. The case studies in Paper II consider the uncertainty/complexity relationship in dynamic environmental models, while the study in Paper IV assesses environmental-economic models. Paper II is somewhat similar to the works by Smith and Vaughan (1980), Halfon (1983) and Kohler et al. (1996) and the water quality model comparisons of Reckhow (1994). Paper II however, isolates and evaluates the differences in uncertainty that are associated specifically with the changes in complexity, by employing the characterization methods described earlier.

Paper III illustrates the use of an environmental feedback effect, which expands upon the traditional Leontief input-output model, and assesses the effects of increasing the complexity of a standard input-output model. Paper IV extends the methods presented by Bullard and Sebald (1988) and West (1986) for use in comparisons between models of different complexities. Such comparisons were done with dynamic models by Smith and Vaughan (1980), Halfon (1983), Kohler et al. (1996) and in Paper II.

In summary, Papers I and III present methods and techniques that were used (along with other methods described in this thesis) in the studies carried out in Papers II and IV. Papers II and IV address the objectives of the study on the relationship between modelling uncertainty and complexity. This section presents the results of those studies.

5.1 Case Study I - Sorption of Metals onto Sediments

Case study I uses three different models to evaluate the uncertainty/complexity relationship for a system of metals in solution sorbing onto sediments. The data are taken from experiments performed by LeBeouf (1992). The three models differ in the complexity of the sorption mechanism only. The simplest model is a 2-state (sorbed and solute) equilibrium model, in which the only parameter employed in the Monte Carlo tests is the distribution coefficient. The moderately complex model (of the 3 models) is a 2-state model that describes the sorption of the metals with a kinetic, reversible relationship. Two parameters are used in the Monte Carlo tests; the distribution coefficient, and the

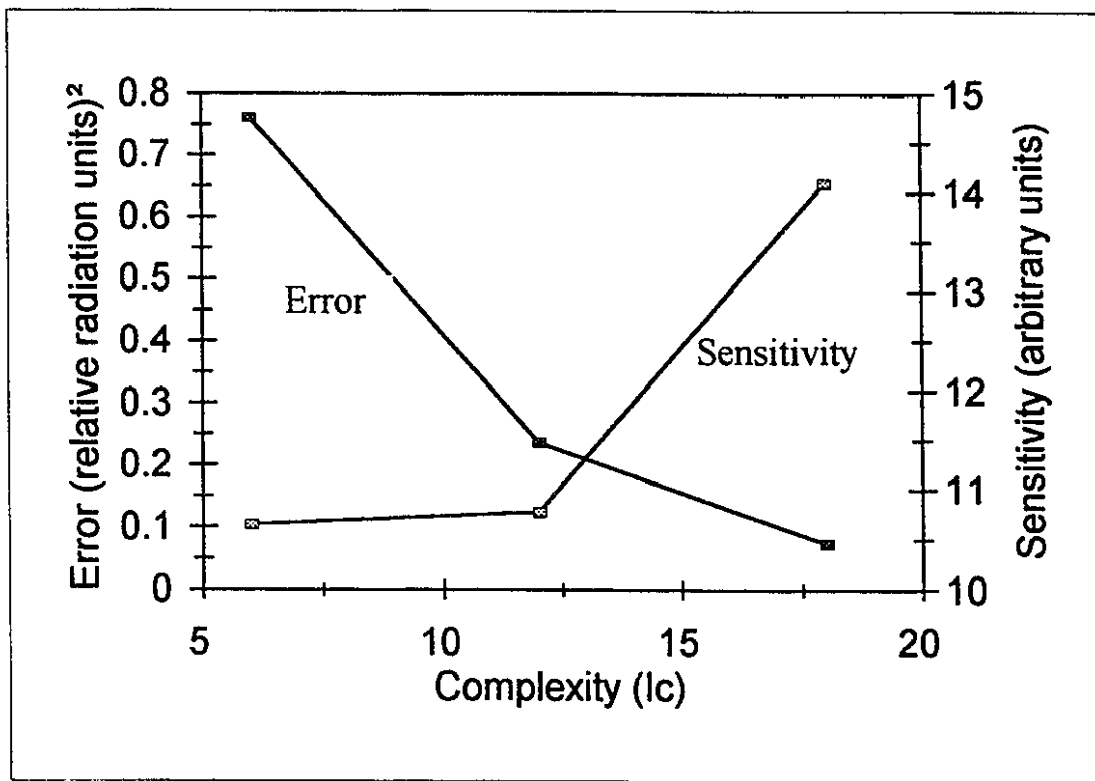


Figure 2. Uncertainty vs. Complexity - Case Study I.

kinetic constant that describes the rate at which the sorption reaction goes to completion. The most complex of the three models uses a 3 state system, which has one solute phase, and two sorbed phases. Both equilibrium and kinetic reversible rate equations are employed, and three parameters are used in the Monte Carlo analysis; the equilibrium distribution coefficient, the forward kinetic rate constant, and the reverse kinetic rate constant. Please see Appendix V for more details on the models, including the parameter distributions and the calculation of the complexity index, I_c .

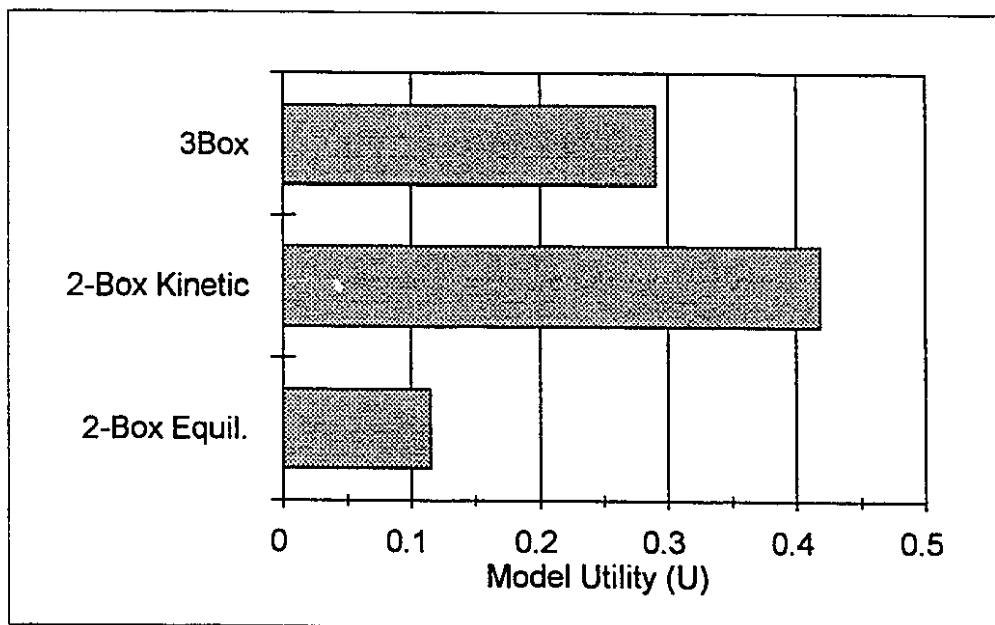


Figure 3. Model Utility - Case Study I.

Figure 2 shows the uncertainty/complexity relationship obtained for case study I. Similar to the hypothesized relationship, the most complex model had the least error, while also being the most sensitive. Figure 3 presents the utility evaluation for the three models (increasing complexity on the y-axis), and shows that the moderately complex model is the most utile.

5.2 Case Study II - Transport of a Tracer in a Homogenous Sand Aquifer

Case study II uses a series of seven 3-dimensional models which describe the transport of an organic tracer (carbon tetrachloride) in a homogeneous sand aquifer. The data are taken from the Borden Data Set, a highly spatially resolved set of data that describes the movement of several tracers in an aquifer over a period of three years at the Canadian Forces Base in Borden, Ontario.

The seven models used in the study vary in complexity through differences in the simulation of sorption (adherence to soil particles which causes retardation of transport) and degradation (transformation of the tracer into a different product, essentially removing it from the system being modelled). The transport model simulates the movement and dilution of the tracer with simple convective and dispersive flow algorithms.

The simplest model consists only of solute transport, with the assumption of no sorption and degradation. In the other models, (in addition to the solute transport) degradation was modelled as either no reaction, or as a first-order kinetic reaction. Sorption is described by either equilibrium reactions (progressing to completion at an infinitely fast rate), or at a rate described by a rate constant. In addition, the liquid-soil sorption function varied in complexity, from a linear relationship to a non-linear relationship to a Monod-type equation. Details on the models, the parameter distributions, and the calculation of I_c for each model can be found in Appendix VI, whereas Table 1 summarizes the nature of the seven models in case study II.

Model Number	Sorption Submodel	Degradation Submodel	Model Complexity, I_c
1	None	None	16
2	Linear Isotherm Equilibrium Sorption	None	22
3	Linear Isotherm Kinetic Sorption	None	28
4	Linear Isotherm Equilibrium Sorption	1 st Order Reaction	24
5	Linear Isotherm Kinetic Sorption	1 st Order Reaction	29
6	Non-Linear Isotherm Kinetic Sorption	1 st Order Reaction	41
7	Monod-type Isotherm Kinetic Sorption	1 st Order Reaction	47

Table 1. Models used in Case Study II.

Unlike case study I, the models used in case study II have spatial resolution. The resolution is chosen as a balance between the quality of the output (too coarse a resolution can lead to difficulties with numerical dispersion) and computational effort. The spatial resolution is held constant for all models in the study. Values for several parameters in the tracer transport models are either taken from the literature (from the Borden Data report), or are estimated from fitting the raw data without noise. These and other model parameters relating to the simulation (such as the integration method and time step) are held constant across all seven models. This is done to isolate the differences in structural complexity as the distinguishing factor among models.

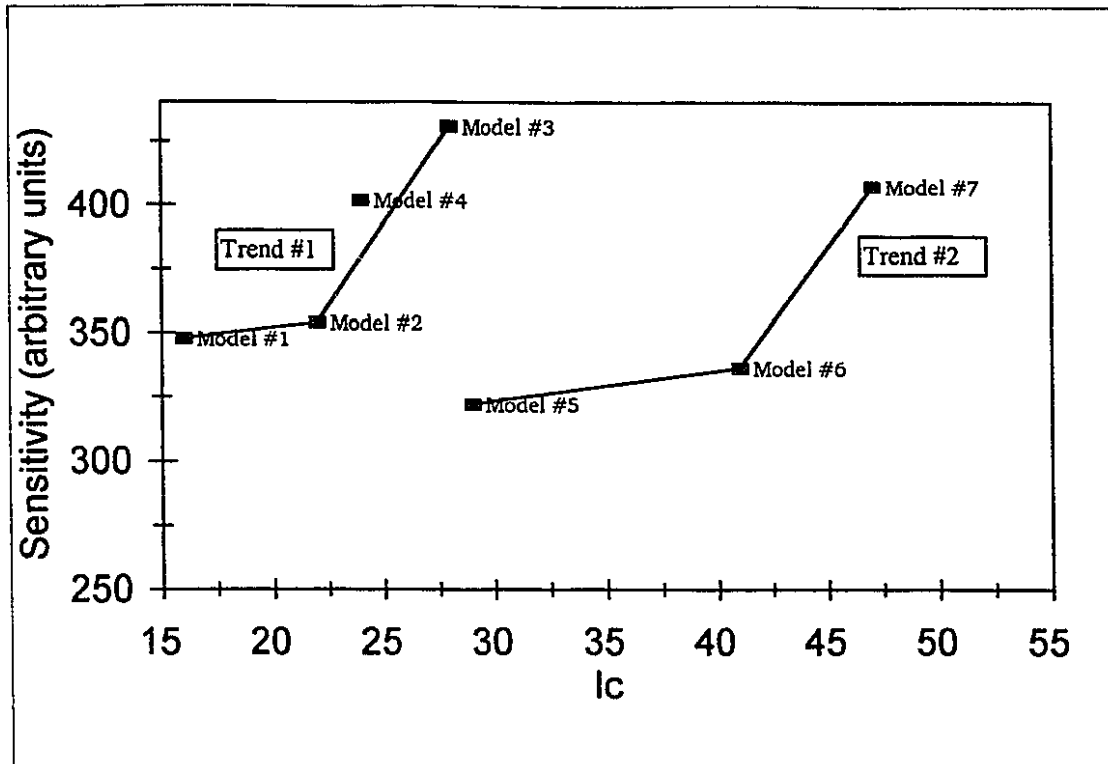


Figure 4. Sensitivity vs. Complexity - Case Study II. Two trends are shown, both indicating increasing sensitivity with increasing complexity.

The results for the sensitivity and error evaluations for case study II are shown in Figures 4 and 5. Trends are evident, although they are not as simple those shown in case study I. Figure 4 shows two trends of increasing sensitivity with increasing complexity; one for models with degradation processes, and one for models without degradation processes. Increasingly complex sorption isotherms (Trend #1 - no sorption/equilibrium sorption/kinetic sorption) and increasingly complex models with degradation (Trend #2 - linear/non-linear/Monod sorption) follow the hypothesis. Error however, appears to be independent of complexity (Figure 5). Significance testing (T-test) indicates that the slope of a regression of the points is indistinguishable from zero at $p = 0.005$. Thus, with respect to error, more complex models do not fit the data better than the simpler models.

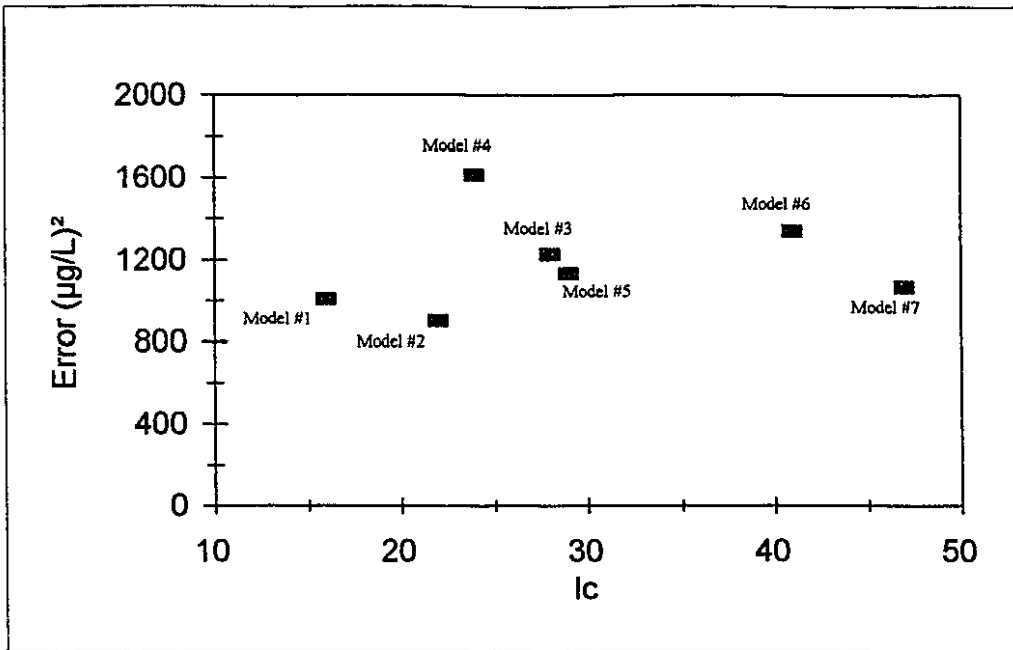


Figure 5. Error vs. Complexity - Case Study II.

Utility for the various models in case study II is summarized in Figure 6. Two of the moderately complex models have relatively low utility values, while the most utile model is the second-least complex. In this case, variations in utility reflect mostly sensitivity rather than both sensitivity and error (even though they are evenly weighted in utility calculations), because error is relatively similar across all models.

In this case, where all models had relatively similar error evaluations, the conclusion that all models are adequately complex can be useful information. This is an indication that one or more of the processes in the model may dominate the system, and that the further processes included in the model do not improve error.

In general, the trends show that the most complex models are not always the best choice. Depending on the definition of utility (sensitivity and error can be weighted according to the user's preference), the increases in sensitivity can outweigh any gains in

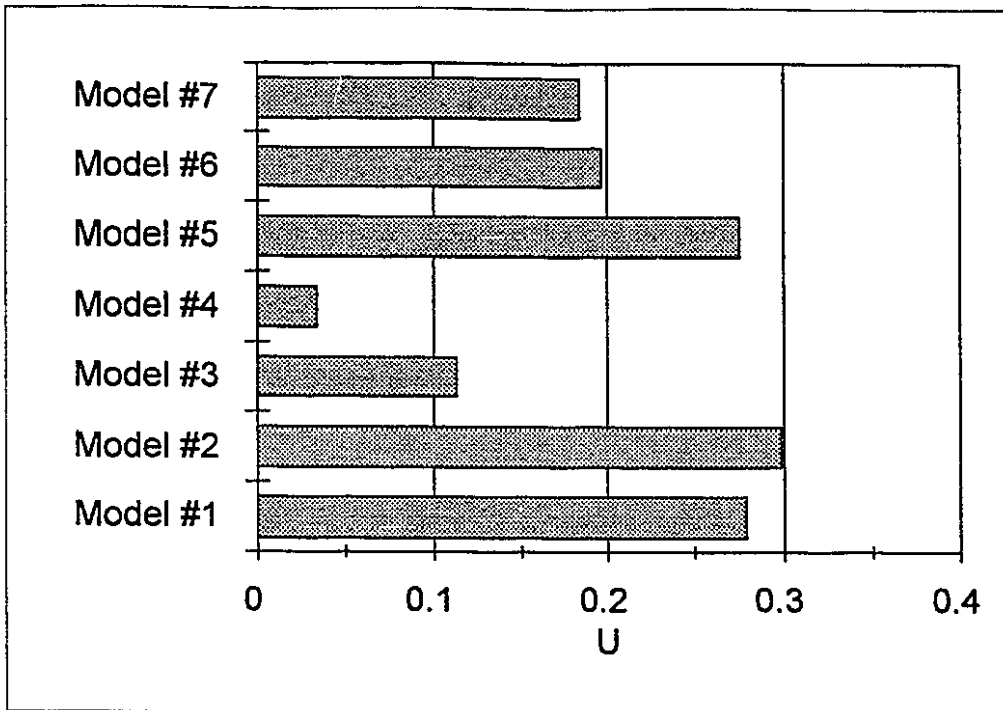


Figure 6. Model Utility - Case Study II. Model #2 has the greatest utility of those studied.

fit achieved through use of more complex models.

5.3 Inter-disciplinary Models - Uncertainty in Environmental/Economic Modelling

The relationship between modelling uncertainty and model structure is investigated further by studying the specific subset of environmental models that are interdisciplinary in nature. This study attempts to quantify the effects on uncertainty of adding an environmental feedback extension. This is similar to the goals of the study in Paper II

An environmental-economic model is developed for the purpose of investigating how uncertainty changes when an economic extension is added to a traditional Leontief input-output model. Environmental impacts from agricultural activities were chosen as the system to be simulated in this study. This expands upon the work of Chen (1973) and Xu et al. (1994), who developed extensions for other environmental systems.

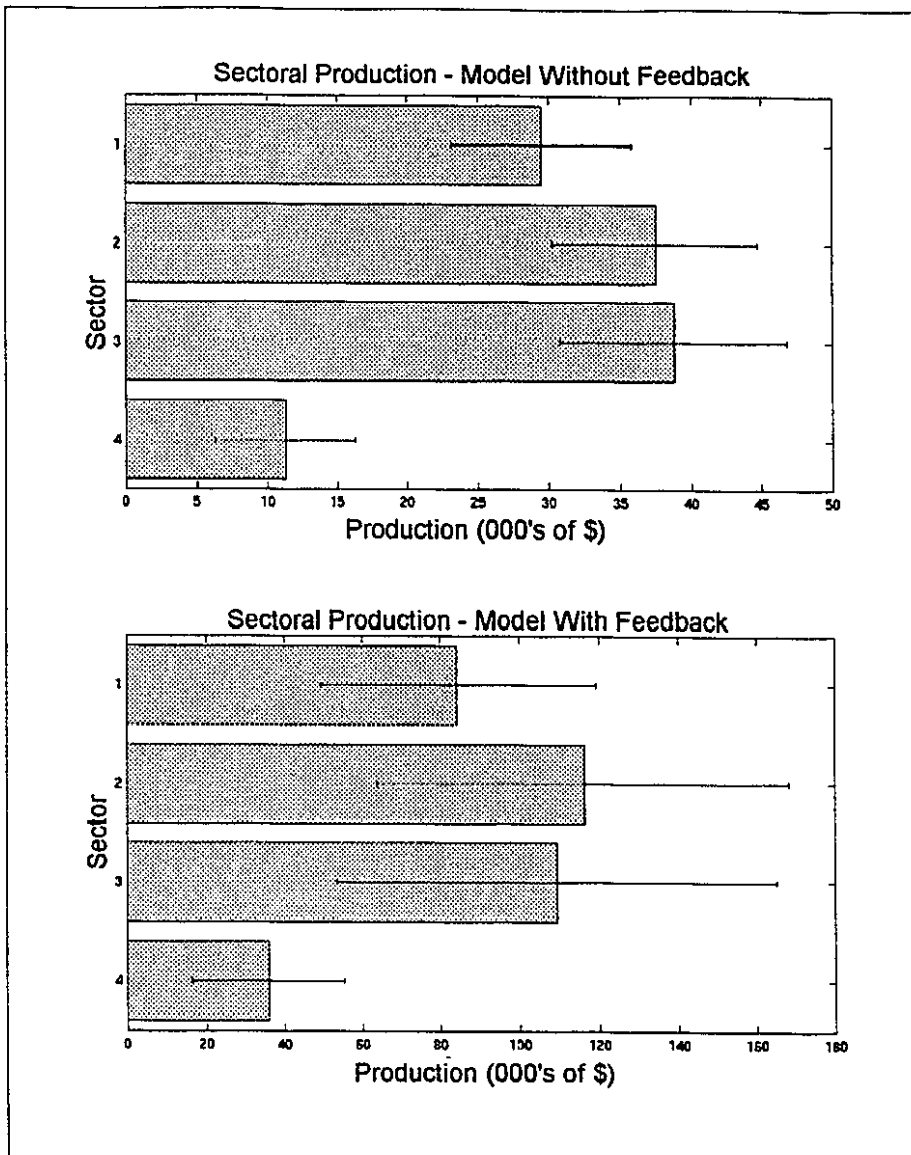


Figure 7. Monte Carlo analyses of I/O models with and without environmental feedback extensions. Note that the model with feedback has higher relative uncertainty.

The Leontief input-output model is a system of linear equations that can be used to evaluate the inter-relationships between different sectors of an economy. Environmental extensions to the basic economic model can be used to investigate the environmental impact associated with various levels of economic production. The use of a feedback

effect as proposed by Leontief (1970) incorporates the cost of remediating pollution back into an economy.

A feedback effect can be added to a standard input-output model to study the relationship between agricultural pesticide regulation and economic impact. The costs of remediating agricultural pesticide pollution are first estimated, and then put back into the economy. Paper III outlines the development of the Pesticide Impact Input-Output Model.

Incorporating an environmental extension into a standard input-output model changes the way uncertainty is generated, because the overall structure of the model is altered. The environmental feedback effect used in the model in Paper III significantly changes the structure of the model by including more equations and processes to describe environmental impact.

In the past, Monte Carlo analysis (as described in section 4.2) has been incorporated to analyze input-output models (West, 1977, Bullard and Sebald, 1988). A standard Monte Carlo analysis is performed on a hypothetical input-output model that contains four economic sectors. Random noise is added to each of the 16 coefficients describing flow to and from each sector. Monte Carlo analysis results for the hypothetical input-output model, both with and without a feedback effect, are shown in Figure 7 (details of the hypothetical model are shown in paper IV). The relative size of the error bars, compared to the magnitude of the production, is significantly greater for the model with feedback effect. The average size of the bars (relative to the magnitude of production), increases approximately 80% when the feedback is added.

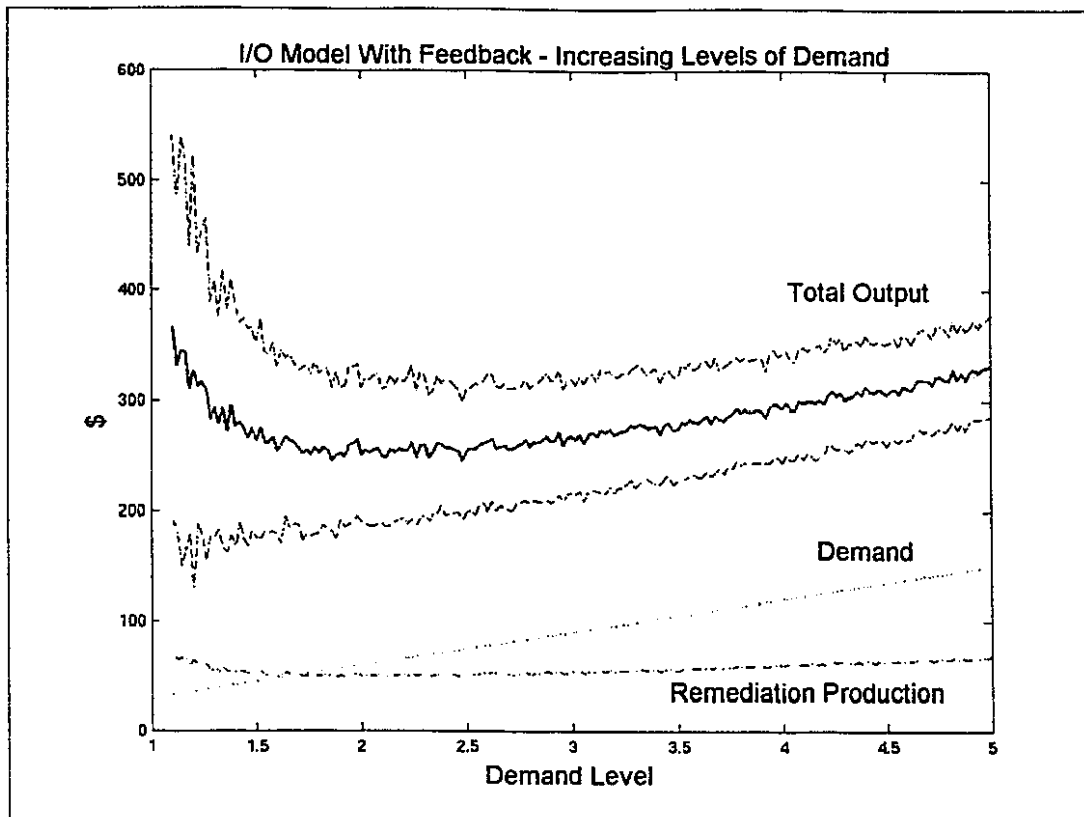


Figure 8. I/O Model with Feedback - Uncertainty is shown by the dashed lines at ± 1 standard deviation, at several different levels of demand. Note that uncertainty is largest when demand is equal to or less than remediation production.

Another way to assess uncertainty is to analyze Monte Carlo results at different levels of exogenous demand. Figure 8 illustrates the total production level (sum of production across all four sectors) at different levels of demand, for models with the feedback effect. Demand level (x-axis) is a scalar factor by which the initial sectoral demand is multiplied. The total output is shown as a thick solid line, and the uncertainty envelope (\pm one standard deviation) is shown with dashed lines. Also shown are the total amounts of demand and remediation production. When the remediation production is significantly less than the total amount of demand, the uncertainty associated with total

production is constant. When remediation production is equal to or greater than the total amount of demand, the uncertainty is significantly increased. At low levels of demand, the new production from pollution remedation (which then causes more pollution, and more remediation, and so on) causes the model solution to converge slowly. At higher demand, the amount of new remediation production is very small compared to the original (demand-based) production, and the model solution converges quickly.

The studies carried out in Paper IV show an example of how a changing model structure (by adding an extension onto a simple, well-understood model) changes the amount of uncertainty of the results. While the studies related to a very specific example (Leontief models), the results show that not only did the new model have higher levels of uncertainty, but also that the uncertainty changed with different levels of output.

A modeller using Leontief-style models may be faced with choosing between a standard input-output model and one with extensions that relate environmental (or other) issues to economic impact. Simple input-output models have smaller uncertainty compared to more complex (but possibly more realistic) input-output models with environmental extensions. The modeller should take into account the balance between realistic simulation and output uncertainty when choosing between the models.

6.0 Discussion and Conclusions

Selecting the correct model for a given simulation effort is important. Too complex a structure can lead to overly-sensitive models that are difficult to properly calibrate and very dependent on data quality. If the model structure is too simple, the

effectiveness of the simulation may be limited to only those situation where the model's simplifying assumptions are valid. Finding the "middle ground" of suitably complex models can be difficult. The following discussion considers this "middle ground" to determine the most suitably complex model.

6.1 The Uncertainty/Complexity Relationship

Case studies I and II in Paper II characterize the uncertainty/complexity relationship for two real systems. The results of case study I follow the hypothesis (Figure 1) closely, showing that the more complex models are more sensitive, but also fit the data better. The results of case study II, however, show that all models produced approximately the same error. In addition, a discontinuity in the sensitivity trend was observed when degradation was introduced.

The results of the two case studies highlight three significant elements of the uncertainty/complexity relationship, and the series of models used in the case studies:

- 1) ***Importance of Model Structure in Sensitivity Evaluation:*** The trend of increasing sensitivity with increasing sorption complexity holds in both case studies, but there is a break when degradation complexity is added in case study II. For each of the two trends shown in the graph of sensitivity vs. complexity (Figure 4), the relationship between sensitivity and complexity is similar. Both Trend #1 and Trend #2 show increasing sensitivity with increasing complexity, while error is essentially the same across all complexities.

This pattern of increasing sensitivity is related to the increasingly complex algorithms used to describe processes (such as sorption) in the model. The discontinuity is due to adding a new process to the model (in this case, degradation). Each of these two elements - using more complex algorithms and adding more processes - results in an increase in overall model complexity, however they affect uncertainty in different ways. Changing the *complexity of the algorithms* produces a trend that follows the original hypothesis, as proposed in Figure 1. Increasing the complexity of the algorithm alone increases sensitivity in a continuous manner. Changing the *structure of the model*, by adding a new process, introduces a discontinuity into the sensitivity trend, as shown in case study II.

An analogy can be drawn with regards to the relationship of sensitivity and complexity, as hypothesized by the upwards-sloping line in Figure 1. The complexity of the algorithms used to describe processes in the model defines the slope (and/or curvature) of the line, and the complexity of the structure of the model (number of processes) defines the intercept. A set of models of a given structure (all having the same state variables and processes) will produce a continuous trend of increasing sensitivity with increasing algorithm complexity. If the structure of the set of models is changed (by adding a new process), a new continuously increasing sensitivity/complexity trend will be found, at a level higher or lower than the original trend.

The sensitivity/complexity trend can be summarized as a series of upward-sloping lines. Changing algorithm complexity slides the model up and down a continuous sensitivity/complexity slope, and changing structural complexity jumps to another, different slope.

2) ***Error Trend Becomes Flat at Higher Ranges of Complexities:*** The models used in each of the two case studies varied in complexity to determine error at different points across a spectrum of complexities (i.e. across a range of I_c values). The models used in the two case studies, however, do not capture the same range of complexities. With respect to error, the models in case study I covered the range from simple (low I_c values) to complex (high I_c values) well. The models in case study II however, mostly fell toward the complex end of the spectrum, as shown by the flat trend of error vs. complexity. All the models were equally good at simulating the system, indicating that there were no models that could be improved upon by being more complex (i.e. there were no models that were “too simple”).

3) ***Relative Importance of Sensitivity and Error in Determining***

Uncertainty: The results of the two case studies are different, but both indicate that sensitivity and error (and therefore uncertainty) are a function of model complexity. For a given set of models, as complexity increases,

the trend of error will eventually become flat, since there is a point at which moving to a more complex model will not reduce error (i.e. they are all sufficiently complex). Beyond this point, uncertainty is a function of sensitivity only. Sensitivity behaves as described above, where it is a property of both algorithm complexity and the structure of the model.

Therefore, uncertainty is a property of algorithm complexity and structural complexity, and is dependent on error and sensitivity in differing amounts across the complexity spectrum.

Given the results of the two case studies in Paper II, and the results of Paper IV, the original hypothesis, as proposed in Figure 1, can be revised. The clearest way to do this is to deal with each trend – error and sensitivity – separately.

The original hypothesis regarding error and complexity stated that error decreased with increasing complexity. Results indicate that this is true only up to a certain threshold, beyond which the error trend becomes flat. At I_c values above this threshold, more

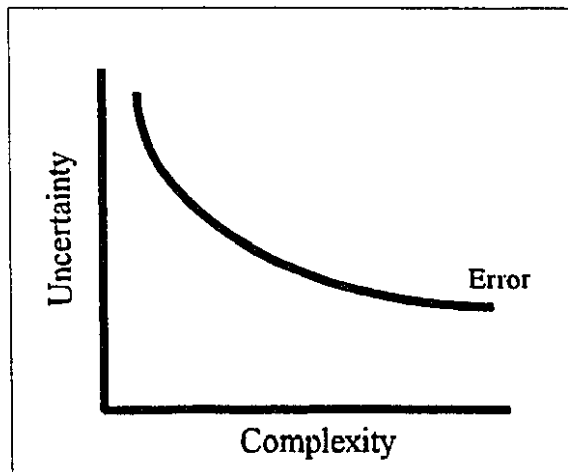


Figure 9. Revised Error/Complexity Relationship.

complex models do not do a better job of simulation (i.e. do not have less error) than simpler ones. Therefore, the revised hypothesis can be shown as in Figure 9.

The sensitivity/complexity relationship can be revised from that shown in Figure 1 to include the effects shown in the case studies. For a given model structure (number of processes and state variables held constant), there is an increasing continuous trend in sensitivity with increasing algorithm complexity. This relationship may be of a linear or exponential nature. In addition, there is a discontinuity observed when structural complexity is increased (such as by adding a new process). Figure 10 shows a revised hypothesis for sensitivity and complexity.

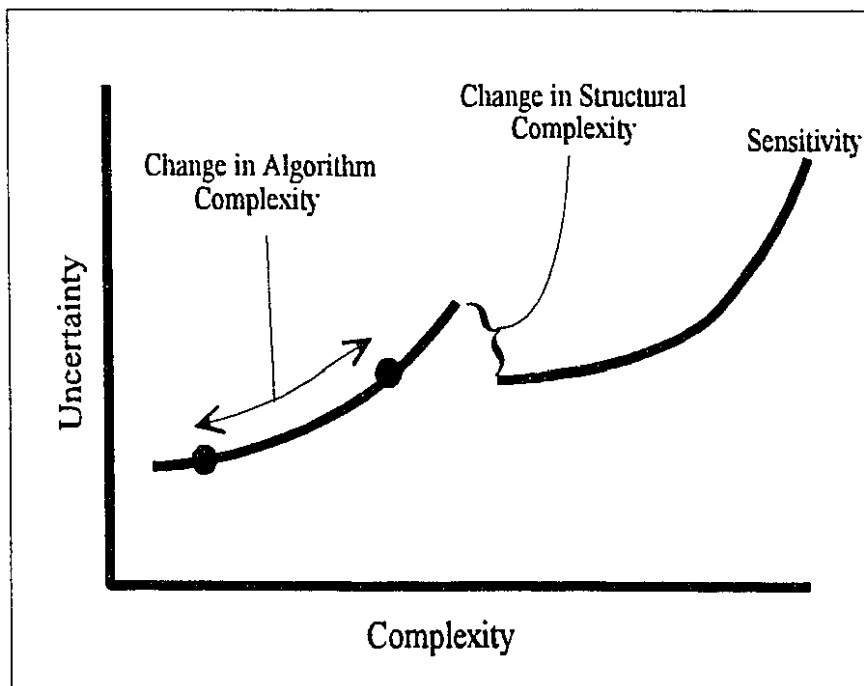


Figure 10. Revised Sensitivity/Complexity Relationship.

6.2 Uncertainty/Complexity in Inter-disciplinary Modelling

The study on Leontief models is a very specific example of the effects of changing structural complexity. Altering even the most simple, linear models can have a very significant effect (under certain circumstances) on the uncertainty of the results. In this particular case, the results show a significant increase in uncertainty with the addition of a feedback effect.

This change is due to the fact that a feedback changes the way the Leontief model solution is determined. Without the feedback, the Leontief model is a "black box" that can be solved explicitly. However, adding the feedback effect changes the "black box" solution into an iterative process, which is much more sensitive to parameter input.

While the inter-disciplinary study carried out in this thesis is a very specific example (in that there are many other types of inter-disciplinary models), the results can be used similarly to those of Paper II to determine whether it is better to use a more complex model (in this case, one with an environmental extension), or a simpler one. The results from Paper IV indicate that the Leontief model with an environmental extension is more sensitive than the standard (non-extended) model, for the ranges of output shown. Therefore, the utility of the more complex model might be less than that of the simpler model, depending on the purpose of the modelling exercise. This trend agrees with the sensitivity trend discussed earlier. Altering the structural complexity of the model (by adding a feedback process) significantly changes sensitivity.

6.3 Model Utility and the Optimal Complexity Threshold

The methods presented in this thesis identify the threshold point where moving to more complex models is not worth the increase in sensitivity for the improvement in error gained. Given data limitations, this threshold defines a point of optimum model complexity. The calculation of utility gives the modeller a method to weigh the gains and costs of error and sensitivity against one another. This balancing of sensitivity and error can be used to determine the optimum model complexity threshold.

The optimum complexity threshold was determined for each case study through evaluation of model utility, where error and sensitivity are weighted equally. Case study I defines the threshold point well, because the models suitably spanned this point (from too simple to too complex), whereas they did not span it well in case study II.

For the models in case study II, improvements in error would probably be best achieved through changes in spatial resolution or other methods. Increasingly complex algorithms may not be the best choice for improvement, since the existing models are all sufficiently complex for the system. Models of lower complexity could be included as part of the test in order to better define the simpler (lower I_c values) end of the trend. These new models could include simpler algorithms, or be structurally simpler models with fewer state variables or processes.

6.4 Using Model Utility to Select the Best Model

Model utility evaluation (as presented in this thesis) can be used to select among models of varying complexity. If however, as in case study II, the models are all

sufficiently complex for the system being simulated, it is possible that error will not contribute to the equation (i.e. all error evaluations are the same since all models do an equally good job of simulation). Evaluation of utility then simplifies to an evaluation of sensitivity. If this happens, utility evaluation could indicate that the simplest model is the best one. The modeller however, should conclude from this evaluation that there might be simpler models that weren't included in the study that might have higher utility. These models should be investigated.

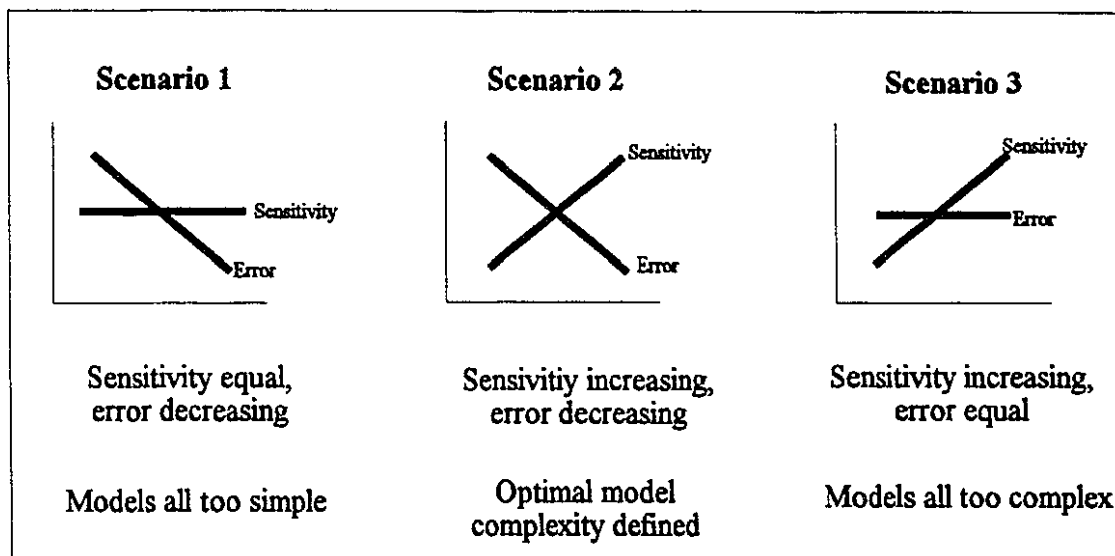


Figure 11. Different scenarios of uncertainty/complexity relationship.

A more general conclusion can be drawn about the uncertainty/complexity relationship by looking at different possible outcomes of model evaluations. Generally, different possible uncertainty/complexity relationships (determined from the methods described in this thesis) can be grouped into three distinct scenarios, as shown in Figure 11. Each scenario provides the user with information that can be used for model selection.

Scenario 2 in Figure 11 shows results where the models span the optimal complexity threshold, and the relationship appears as originally hypothesized (as shown in Figure 1). Case study I falls into this category.

Scenario 3 shows the instance where all the models in the study are sufficiently complex (in error evaluation) for the system being simulated. They can all be fitted to the data equally well for error (as shown by the flat error trend). The user can determine from this that simpler models should be investigated, in order to fully determine the optimal complexity threshold. Case study II falls into this category.

Scenario 1 shows the instance where all the models in the study are much simpler than the system being simulated. This case could be illustrated by fitting a series of simple linear models to data from a complex system (including the simplest model of all, $x(t) = \text{constant}$). In this case, the results indicate that the models are all too simple, and that there may be more complex models that could better simulate the system. These models should be investigated to fully determine the optimal complexity threshold, by determining the point where moving to more complex models is not worth the associated gain in sensitivity.

In the most general sense, performing this type of study on any series of models and systems can help the user determine which of the models is closest to the optimal complexity. It will also show if the models chosen do not fully define this point. In the latter case, the results can be used to help choose a new set of models, and in an iterative process, achieve the goal of selecting the optimally complex model for the job.

6.4 Conclusions

Uncertainty evaluation is important when using a model as a decision-making tool. It provides a context within which to evaluate the confidence of output or help to allocate resources for efficient data gathering. It can also be used to help select a model of optimal complexity, through the analyses presented in this thesis.

Evaluation of model error and sensitivity can be used to determine the optimal complexity threshold, beyond which it is not better to move to a more complex model. The uncertainty/ complexity relationship does not follow that proposed in Figure 1 for each and every possible set of models and systems. The relationship however, once determined, can provide very useful information to a modeller faced with choosing among models of different complexities. The relationship may show flat trends for either sensitivity or error, which tells the modeller that the optimal level of complexity does not lie within the set of models studied.

The uncertainty/complexity relationship for two real systems indicated that moderately complex models can be the most utile. Case study I illustrated the instance where the modeller should choose a two-state model over a three-state model, because even though the three-state model can simulate the system more closely, the two-box model does almost as good a job with far less sensitivity. Case study II showed that the two simplest models simulated the system as well as the more complex ones, but with less sensitivity. In this case however, it might be possible to find even simpler models (lower I_c values) with even less sensitivity, since the results do not define the optimal complexity point very well. None of the models in case study II are so simple as to have greater error

than any of the other models, and therefore do not bound the optimal complexity threshold from the lower I_c side. In this case, the modeller should choose to repeat the study with some simpler models included in the evaluation.

7.0 Future Work

The use of uncertainty evaluation for model selection can be expanded beyond the examples presented in this thesis. Specifically, the development of methods to analyze the effect of spatial resolution (and/or complexity) of models on uncertainty could be addressed.

Analyzing the relationship between spatial resolution and uncertainty would first necessitate the development of methods to include spatial resolution in the index of complexity (I_c) calculations presented in Paper I. If a 3-dimensional model with 1,000 spatially distinct cells is considered to be more complex than a 1-cell reactor with equivalent kinetic complexity, the I_c value should reflect this.

Because sensitivity is dependent on both algorithm complexity and structural complexity, studying these two elements separately would be beneficial. Currently, I_c combines these two types of complexity together into one value. Therefore, development of independent algorithm and structural complexity measurements is needed. Then, the relationship between sensitivity and each of algorithm and structural complexity could be further characterized.

The uncertainty/complexity relationship determined for the case studies in Paper II could also be determined for other kinds of models. The method is not model specific,

and could be applied to any series of models that can be described adequately by a Petersen chart.

8.0 References

- Beck, M. (1987) Water Quality Modeling: A Review of the Analysis of Uncertainty. *Water Resources Research*, **23**: 1393-1442.
- Bosserman, R.W. (1980) Complexity Measures for Assessment of Environmental Impact in Ecosystem Networks. In: *Proceedings of the Pittsburgh Conference on Modeling and Simulation*. Pittsburgh, PA., Apr. 20-23, 1980.
- Bullard, C.W., and Sebald, A.V. (1988) Monte Carlo Sensitivity Analysis of Input-Output Models, *The Review of Economics and Statistics*, **70**: 708-712.
- Chen, K. (1973) Input-Output Economic Analysis of Environmental Impact, *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-3**: 539-547.
- Dakins, M.E., Toll, J.E., and Small, M.J. (1994) Risk-Based Environmental Remediation: Decision Framework and Role of Uncertainty, *Environmental Toxicology and Chemistry*, **13**:1907-1915.
- Haimes, Y.Y., and Bear, J. (1989) Preface. In: *Groundwater Contamination: Use of Models in Decision Making*, IGWMC Conference 1987, Kluwer Academic Publishers, The Netherlands, 656 pgs.
- Halfon, E. (1983) Is There a Best Model Structure?: 1. Modeling the Fate of a Toxic Substance in a Lake. *Ecological Modelling*, **20**: 135-152.
- Harremoës, P., and Madsen, H., (1999) Fiction and reality in the modelling world - balance between simplicity and complexity, calibration and identifiability, verification and falsification. *Water Science and Technology*, **39**: 1-8.
- Henze, M., Grady, C.P.L., Gujer, W., Marais, G v R, and Matsuo, T. (1987) *Activated sludge model no. 1*. IAWPRC Scientific and Technical Report No. 1, IAWPRC, London.
- Kalos, M.H., and Whitlock, P.A. (1986) *Monte Carlo Methods: Volume I: Basics*. John Wiley & Sons, Inc., Toronto. 186 pgs.
- Kohler, M. (1996) Experimental investigation and modeling of uranium(VI) transport under variable chemical conditions. *Water Resources Research*, **32**: 3539-3551.
- Konikow, L. and Bredehoeft, J. (1992) Ground-water models cannot be validated. *Advances in Water Resources*, **15**: 75-83.

- LeBeuf, M. (1992) *Kinetics of Trace Metal Interactions with Lake Sediments*, Ph.D Thesis, McMaster University, Hamilton, Ontario.
- Leontief, W. (1970) Environmental Repercussions and the Economic Structure: An Input-Output Approach, *The Review of Economics and Statistics*, 57: 262-271.
- Miller, R. (1974) The jackknife - a review. *Biometrika*, 61: 1-15.
- Miller, R.E., and Blair, P.D., (1985) *Input-Output Analysis: Foundations and Extensions*, Prentice-Hall, New Jersey, 464 pgs.
- Morgan, M.G., and Henrion, M. (1990) *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*, Cambridge University Press, New York, N.Y.
- Oreskes, N., Shrader-Frechette, K., and Belitz, K. (1994) Verification, Validation, and Confirmation of Numerical Models in the Earth Sciences. *Science*, 263: 641-646.
- Palmini, D.J., 1982. The Secondary Impact of Nonpoint-Pollution Controls: A Linear-Programming-Input/Output Analysis, *Journal of Environmental Economics and Management*, 9: 263-278.
- Pearl, J. (1978) On the Connection Between the Complexity and Credibility of Inferred Models. *International Journal of General Systems*, 4: 255-264.
- Petersen, E. (1965) *Chemical Reaction Analysis*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- Reckhow (1994) Water Quality Simulation Modeling and Uncertainty Analysis for Risk Assessment and Decision Making. *Ecological Modelling*, 72: 1-20.
- Smith, V., and Vaughan, W. (1980) The implications of model complexity for environmental management. *Journal of Environmental Economics and Management*, 7: 184-208.
- West, G.R., (1986) A Stochastic Analysis of an Input-Output Model. *Econometrica*, 54: 363-374.
- Xu, P., Tsanis, I.K., Anderson, W.P., and Kanaroglou, P. (1994) An Economic Input-Output Analysis for Urban Stormwater Quality Planning, *Water Resources Management*, 8: 155-170.

Appendix I: Index of Model Complexity

Abstract

An index of model complexity, I_c , is proposed to obtain a unified measure of model complexity. $I_c = \sum_{j=1}^N \sum_{i=1}^{n_j} p_i r_i$ where p_i is the number of parameters, and r_i is the number of mathematical operations, collectively used to describe a process in the model. These values are then summed across all processes and state variables in the model. The index is easily calculated from a Petersen Matrix. I_c is calculated for seven wastewater models, to determine their relative complexity. The index can be used for categorization of model structures as well as complexity characterization in studies relating structure to model performance.

Keywords: complexity, index, model, wastewater, Petersen matrix.

Nomenclature

I_c	= index of model complexity
N	= number of state variables in the model
n_j	= number of processes associated with state variable j
p_i	= number of parameters used to describe process i
r_i	= number of mathematical operations used to describe process i .

Introduction

When comparing models for an evaluation such as fit, efficiency, speed or predictive ability, there is a need to be able to differentiate models based on their compositional properties and internal characteristics. The concept of a model's complexity is nebulously defined, but is generally conceived to be a useful general description of model

structure. How can model complexity be rigorously defined and quantified as a tool for model characterization? This paper proposes the index of model complexity, I_c for that specific purpose.

Model developers and model users have estimated the "complexity" of a particular model based on the number of state variables, processes, and the number of stoichiometric and kinetic parameters. This information on model complexity has practical importance when choosing which model to use, and when considering the effort involved in model coding, data collection, and usage.

The index of complexity was developed for ordinal comparisons among models, to complement the above characteristics and in an effort to provide a unified measure of "model complexity". I_c is representative of both the complexity of the structure of the model, as well as the complexity of the mathematical relationships used to describe the processes. However, the use of I_c does not supercede any issues related to the applicability of a model based on its development. The value is readily obtained by working with a Petersen matrix of the model (Petersen, 1965; Henze, 1987).

Quantifying Complexity

The complexity of a model is a characteristic that could be quantified by assessing various aspects of the model, including calculation time, number of variables, etc. There is no accepted standard method for complexity characterization in the modelling literature, although some methods have been suggested. For example, Halfon (1983) used the Bosserman index of connectivity, c , to measure the complexity of ecosystem models.

Bosserman's index is calculated by determining the number of direct and indirect links between states in an ecosystem. It does not, however, take into account any complexity characteristics of the mathematics involved in the model.

The technique presented here combines the number of mathematical operations and number of parameters to obtain a characteristic complexity measurement. Eq. (1) describes the index, I_c , by summing the complexity values for each model process equation. For each process in the model, the number of parameters is multiplied by the number of mathematical operations in the equations describing the process. This calculation reflects both the complexity of the mathematics, as well as the number of degrees of freedom for that particular process. This number is then summed across all processes for each of the state variables.

$$I_c = \sum_{j=1}^N \sum_{i=1}^{n_j} p_i r_i \quad (1)$$

Several simple equations are presented in Table 1 to illustrate how the number of mathematical operations defines complexity.

Equation	Number of Operations, r_i	Number of Parameters, p_i	I_c
$y = C$ (2)	0	1	0
$y = kX$ (3)	1	1	1
$y = kX^N$ (4)	2	2	4
$y = \frac{kX}{R + X}$ (5)	3	2	6

Table 1. Complexity of Several Equations.

In moving from the least complex relationship, Eq.(2), to the most complex, Eq.(5), note that the number of mathematical operations increases from zero to three, justifying the use of operations as a measurement of complexity.

The number of parameters used to describe a mathematical relationship can be viewed as the number of degrees of freedom of a model, and therefore also as an evaluation of complexity. For example, eq.(3) has one degree of freedom, k , whereas the more complex Freundlich-type equation, eq.(4), has two degrees of freedom, k and N . The number of degrees of freedom increases with the increasing complexity of a relationship. Both mathematical operations and degrees of freedom are incorporated into the overall evaluation of model complexity, I_c .

This characterization of complexity through evaluation of process equations defines the “kinetic complexity” of the model. There are other types of model complexity that are analogous concepts. For example, “spatial complexity” could be characterized as a property of the dimensionality or spatial resolution of a model. However, for the purposes of this paper, only the kinetic complexity of model is considered.

Calculation of I_c from a Petersen Matrix

A Petersen matrix offers a straightforward way to calculate I_c . The values r_i and p_i are calculated for each process rate equation listed on the Petersen matrix, as well as for each entry in the stoichiometry table. The process rate equations can be calculated as a stoichiometry term multiplied by the corresponding process rate, and then summed down the column of the Petersen matrix. Therefore, to calculate I_c , the term $p_i r_i$ is calculated for

each element of the table (stoichiometry multiplied by process rate), and totalled for each column. Each column total is then added up to calculate the total I_c . This “double-summing” is equivalent to totalling all the flows for each state variable (down the columns) across all state variables (across the columns), as described in Eq.(1).

Calculating I_c for a Hypothetical Example

To illustrate the method of calculation of I_c , a box-and-arrow diagram of a simple, hypothetical model is shown in Figure 1, along with an accompanying Petersen matrix in Table 2.

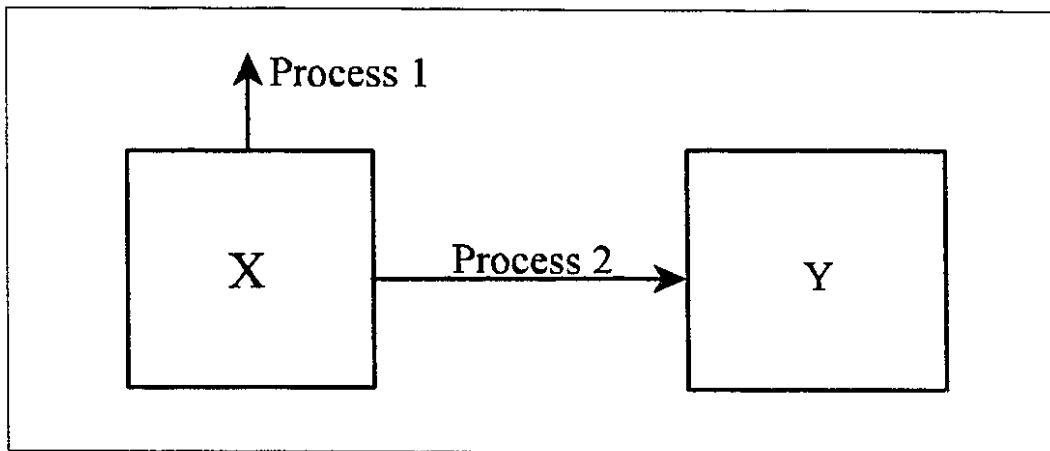


Figure 1. Box-and-Arrow Diagram of Hypothetical Model.

	X	Y	Process Rate Equation
Process 1	1		$k_1(X)$
Process 2	-a	(1-a)	$k_2(X-Y)$
Stoichiometric Parameters: a			Kinetic Parameters: k_1, k_2

Table 2. Petersen Matrix of Hypothetical Model.

According to the above Petersen matrix, the process rate equation for X and Process 1 is assumed to be simply $k_1(X)$, which has 1 parameter (k_1) and 1 mathematical operation (1 multiplication). Therefore, $p_1r_1 = 1 \times 1 = 1$. For X and Process 2, the process rate equation is $-ak_2(X-Y)$, which has 2 parameters (a and k_2) and 3 mathematical operations (2 multiplications and 1 subtraction). Therefore, $p_2r_2 = 2 \times 3 = 6$. Lastly, for Y and Process 2, the process rate equation is $(1-a)k_2(X-Y)$. This equation has 2 parameters (a and k_2) and 4 mathematical operations (2 multiplications and 2 subtractions). Therefore, $p_2r_2 = 2 \times 4 = 8$. Summing down the columns, the complexity value for X is equal to $1 + 6 = 7$, and the complexity value for Y is equal to 8. Totalling across the rows, $I_c = 7 + 8 = 15$.

Calculating I_c for Real Models

Mathematical modelling of the activated sludge process has been the subject of intensive research over the last ten years. The development of the ASM1 model (Henze, et al., 1987) standardized nomenclature, model presentation and to a lesser extent, model structure. The basic model structure has remained consistent in the large number of variations of ASM1 that have since been published in the literature. The publication of the ASM2 model for phosphorus removal (Henze, et al., 1995) brought forth a new family of more complex models. Since then, ASM3 (Gujer, et al., 1998) and ASM2d (Henze, et al., 1998), as well as several other structurally different models (Jeppsson and Olsson, 1993), have been published. Even though these models are more complex than their predecessors, there has also been a tendency to develop simpler models. The main benefit of simpler

models is easier and less costly parameter identification.

Table 3 summarizes results obtained for I_c for a series of often used wastewater models. The models are described below.

Carbon Removal Models

Monod-Herbert: Simple model often used as a theoretical illustration, to explain the Petersen matrix model documentation format. The model only deals with organic carbon removal (Herbert, 1958).

Carbon and Nitrogen Removal Models

ASM1: The standard model for most of the last decade, which introduced the death-regeneration hypothesis for simulation of endogenous conditions. The model was one of the first to use the now standard influent fractionation of COD into different biodegradable and non-biodegradable fractions (Henze, et al., 1987).

Reduced Order Model: A simplified version of the ASM1 model, developed primarily for control applications and easier identifiability. The model is in fact two separate models in one: one model structure deals with aerobic environment, while another one

describes anoxic processes (Jeppsson and Olsson, 1993).

ASM3: Reflects new knowledge gained since the release of ASM1, as well as an attempt to keep model structure simplified. The major difference from ASM1 is more emphasis on storage of organic substrate before oxidation, the simplified description of the endogenous state, and simplified handling of nitrogen fractions (Gujer, et al., 1998).

Biological Phosphorus Removal Models

BEPR+ASM1: The first model to describe biological P removal coupled to a full carbon-nitrogen model (BEPR+ASM1 contains a slightly modified version of ASM1) (Dold, 1990).

ASM2: Second generation model describing biological P removal from the IAWQ Task Group (Henze, et al., 1995). In addition to the biological reactions which affect P removal, the model also includes spontaneous or induced chemical precipitation of phosphorus.

ASM2d: Small modification of ASM2, including anoxic denitrification by phosphate-accumulating heterotrophic organisms (Henze,

et al, 1998).

Model	Components	Processes	Parameters	I_c	$\log(I_c)$
MONOD-HERBERT (1958)	3	2	4	49	1.7
REDUCED (1993)	5	4	7	82	1.9
ASM1 (1987)	13	8	19	1050	3.0
ASM3 (1998)	13	12	43	3515	3.6
BEPR+ASM1 (1990)	17	28	49	7724	3.9
ASM2d (1998)	19	21	55	9368	4.0
ASM2 (1995)	17	17	49	9402	4.0

Table 3. Index of Complexity Values for Several Wastewater Models.

Several conclusions are drawn from evaluating I_c for each of the models. The addition of new components to simpler models significantly increases model complexity. For example, the MONOD-HERBERT model has only carbon components. The addition of nitrogen components to these simpler models (resulting in the ASM1 and ASM3 models) increases $\log(I_c)$ values from the 1.7 range up to the 3.0 - 3.6 range. The addition of phosphorus components (ASM2, ASM2d, and BEPR+ASM1 models) increases $\log(I_c)$ values up to 3.9 - 4.0 range.

The REDUCED model has an overall I_c value of 82 for the entire model as a whole. Calculating I_c separately for each of the two model structures results in a value of 37 for the anoxic component, and 45 for the aerobic component. Using Monod terms in the process rate equations causes a high penalty in terms of the complexity index. The REDUCED model, with linear rates and two separate environments (instead of continuous switching for DO), is significantly simpler than the source model it was derived from,

ASM1, which incorporates Monod terms. Note that Monod terms become effectively less complex at low and high concentrations, where (respectively) linear and constant behaviour is dominant.

More complex models with greater numbers of processes require larger amounts of CPU time to perform simulations. The CPU usage (for a standard simulation) for several of these models was correlated to I_c , and a linear relationship was observed ($r^2 = 0.97$).

While there is a tendency for the complexity of mathematical models to increase with their development over time ($I_c \approx 50$ in the 1950's, $I_c \approx 10000$ in the 1990's) a definite trend to keep the models simple can be observed.

How to Use I_c Effectively

I_c can be best used as an independent variable for ordinal comparisons between models. For example, I_c would be an effective characterization of complexity in studies correlating complexity and predictive ability, computational requirements, or modelling efficiency. The index could also be used for categorization of model structures, as shown with the wastewater models above. Models with similar structure types would group together, and the addition of new components are shown by significant differences in I_c .

It is important to acknowledge that I_c is not an evaluation in any way of model performance. It only characterizes model mathematics and structure, and does not address issues such as numerical integration, parameter distribution, or identifiability.

Conclusion

The index of complexity, I_c , is a characteristic measurement of the composition of a model, incorporating both the model structure and the mathematics involved. It can be used for comparison or categorization of models and model structures.

References

- Dold, P.L. (1990) Incorporation of biological excess phosphorus removal in a general activated sludge model. Proc. 13th International Symposium on Wastewater Treatment. November, Montreal, Canada. 83-113.
- Gujer, W., Henze, M., Mino, T., and van Loosdrecht, M.C.M., (1998). Activated sludge model no. 3. Proceedings of the Specialize Conference on Modelling and Microbiology of Activated Sludge Processes, Kollokollo, Denmark, 16-18 March, 1998. To be published in *Water Science and Technology*.
- Halfon, E. (1983) Is There a Best Model Structure?: 1. Modeling the Fate of a Toxic Substance in a Lake. *Ecological Modelling*, **20**, 135-152.
- Henze, M., Grady, C.P.L., Gujer, W., Marais, G v R, and Matsuo, T. (1987). Activated sludge model no. 1. IAWPRC Scientific and Technical Report No. 1, IAWPRC, London.
- Henze, M., Gujer, W., Mino, T., Matsuo, T., Wentzel, M., and Marais, G v R. M.C.M. van Loosdrecht (1998) Outline - Activated sludge model no. 2d. Proceedings of the Specialized Conference on Modelling and Microbiology of Activated Sludge Processes, Kollokollo, Denmark, 16-18, March, 1998. To be published in *Water Science and Technology*.
- Henze, M., Gujer, W., Takahashi, M., Matsuo, T., Wentzel, M., and Marais, G v R. (1995). Activated sludge model no. 2. IAWQ Scientific and Technical Report No. 3, IAWPRC, London.
- Herbert, D.(1958) Some principles of continuous culture. In: Recent Progress in Microbiology. G. Tunevall (Ed.). Almquist and Wiksell, Stockholm, 381-396.
- Jeppsson, U. and Olsson, G. (1993), Reduced Order Models for on-line Parameter Identification of the Activated Sludge Process. *Water Science and Technology* **28**

(11-12) 173-183.

Petersen, E. (1965), *Chemical Reaction Analysis*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Appendix II: Evaluating Modelling Uncertainty for Model Selection

Abstract

Modelling uncertainty is evaluated with respect to model complexity, sensitivity and error. The hypothesis that more complex model simulate reality better, but with more sensitivity and less error, is tested. An Index of Complexity is proposed. Improvements in fit of more complex models are weighed against the increase in model sensitivity. A simple index of utility is then proposed for model selection. The index of utility evaluates model sensitivity (response to changes in input) and model error (closeness of simulation to measurement). Model utility is evaluated for several models in two case studies, a simple system involving sorption of metals on sediments, and a more complex system, involving groundwater transport of a hydrophobic contaminant. Moderately complex models are found to be the more utile of those tested.

Keywords: Modelling, uncertainty, complexity, utility, sensitivity.

Nomenclature:

I_c	= index of model complexity
N	= number of state variables
n_j	= number of processes flowing to or from state variable j
p_i	= number of parameters used to describe process i .
r_i	= number of mathematical operations used to describe process i .
U_i	= utility index for model i
S_i	= sensitivity value for model i (relative to maximum sensitivity)
E_i	= error value for model i (relative to maximum error)
k_s, k_e	= weighting constants for sensitivity, error
a, k_1, k_2, X, Y	= hypothetical model components (for example purposes only)

Introduction

Modelling is a commonly-used tool in the research and management of environmental systems. The ability to use a mathematical construct to simulate the behaviour of a system has allowed environmental researchers and managers to predict how different scenarios and stimuli will affect various environmental systems. However, there are many different model structures available for any given problem, which begs the question, “which structure is best?”

The usefulness of a model can be assessed by looking at modelling uncertainty, a measure of how well the output of the model is “known”. Uncertainty is inherent in the modelling process, and is a property of a model’s structure. Therefore, in order to determine which modelling structure is best for any given exercise, modelling uncertainty needs to be quantified in a meaningful way.

When users are faced with the task of selecting among several different models of varying complexity, uncertainty can be used to evaluate one model against another. Are more complex models better? Does the development of ever-increasingly complex models always benefit model users? Is there a point where the data cost and increased variability of a more complex model outweigh the gains of more accurate calibrations and predictions? These issues can be addressed by looking at the relationship between the complexity of a model’s structure and the uncertainty associated with its output.

Modelling uncertainty can be defined in many ways, ranging from statistical parameters (such as standard deviation) to analytical calculations of propagation of error, to more complicated methods of sensitivity estimation such as Monte Carlo analysis. The

evaluation of modelling uncertainty attempts to quantify the quality of output, and to give the user a framework for its interpretation. The level of uncertainty associated with a model's output is dependent on the model input, the calibration data, and the complexity of the model's structure.

A user who wishes to use uncertainty evaluation as a tool for model selection, needs a method to quantify the utility of a model. Utility involves predictive ability, uncertainty and complexity (or "cost"). The utility of a model increases as the uncertainty of its output is reduced. By evaluating the utility of several models of different complexities, the question of "which one is best" can be addressed.

The purpose of this paper is to identify and characterize modelling uncertainty as a property of the complexity of a model's structure, and to propose a method for the evaluation of model utility. Given this knowledge, a user can make an informed choice of one particular model structure over another, thereby influencing the amount of uncertainty encountered in a modelling exercise. *

Background

In order to be able to evaluate modelling uncertainty, it is important to understand how models are used, and how model output is comprehended. Each model is developed in an unique way, and has properties that influence how its output should be interpreted. Since models are a representation of reality, model output needs interpretation, and should not necessarily always be believed to be the truth. Oreskes et al (1994) distinguished between verification, validation, and calibration of numerical models. Verification and

validation are techniques that are used to provide a measure of credence or credibility to the model results. The authors point out that often, terms such as verification (testing to show that a model's numerical solution is adequately close to the analytical solution) and validation (testing to show that calibrated models produce output consistent with observed data) are used synonymously, which is incorrect. These terms are often used, even more incorrectly, to imply that a given model is an accurate representation of physical reality. This belief that models will provide accurate simulations under all conditions can lead to poor judgment when models are used as decision-making tools. Reckhow and Chapra (1983) also noted that verification or validation testing of model performance is at best confirmation (or corroboration) with empirical evidence. Konikow and Bredehoeft (1992) note that the terms "verified" and "validated" are often incorrectly construed by users to mean that the models are "true".

Beck (1987) identified the uncertainty of model structure as an important area of study. Beck points out that models of different structures serve different roles in the field of environmental simulation, and that models of increasing complexity are increasingly difficult to validate, due to the fact that it is often difficult to unambiguously falsify the model. With large numbers of hypotheses involved in the development of complex models, merely demonstrating a difference between model prediction and observations is usually inadequate for distinguishing which of the many hypotheses have been falsified. Supporting this idea, Pearl (1978) notes that the tendency to accept simpler models as more trustworthy or credible is partly based in the idea that simpler models are more falsifiable and more testable, and therefore more "confirmable". Simple models are

generally chosen, following the principle of Ockham's Razor. Oreskes et al (1994), however, argue that there is no evidence that simple models are more likely to produce more accurate results than complex models.

There are also fundamental reasons for choosing as simple a model as possible for a given endeavour. Data requirements increase with increasing model complexity. Depending on the nature of the system being modelled, data acquisition can be very expensive, and cost-prohibitive. Smith and Vaughan (1980) used three models of varying complexity to investigate cleanup costs for steel industries using various furnace types. The purpose of the study was to investigate the balance between using models of higher complexity and the cost of the associated increases in data and computational requirements. More simply put by the authors, "the issue amounts to judging whether the improved quality of the information provided by a more detailed model is worth the added costs of its development and use." They concluded that it is not possible to answer this question in a generalized way, but that it is possible to analyse the increased costs of building and utilizing progressively more complex process analysis models. For the steel industry models used in their study, there was an estimated tenfold increase in development costs between the simplest and most complex models. They conclude by saying that the increased cost of a more complex model may outweigh any benefits of greater complexity.

Halfon (1983) also addressed the issue of modelling uncertainty, by evaluating the performance of several different lake ecosystem models of varying complexity.

Various studies have been made to ascertain the "best" model in environmental

chemistry. Smith et. al. (1999), in a comprehensive assessment of chemical speciation, found increasing uncertainty with greater complexity in these relatively simple systems. They concluded whereas simple one parameter systems gave an error of log "x" of about 0.02, 7 parameters systems generated a probable error of log "x" of 2+. They also found that the mathematical techniques used and the assumptions for minimization affected the outcome.

Usunoff et. al. (1992) produced a statistical approach to select among different models, whereas Kohler et. al. (1992) used similar ideas to select a "best" model for uranium sorption.

Model Complexity

Defining the nature of model complexity is prerequisite to relating uncertainty and complexity. How can model complexity be usefully quantified, and how is it related to other properties of the model?

Wagenet and Rao (1990) discussed different types of simulation models used for various purposes, and categorized models into three basic groups, based on model complexity: research models (more complex), management models (less complex), and screening models (analytical solutions used only for relative comparisons - not part of this study). Each of these these types of models was developed for a specific purpose, and had inherent properties, assumptions and limitations based on its development.

The use of each of these types of models involves assumptions about model

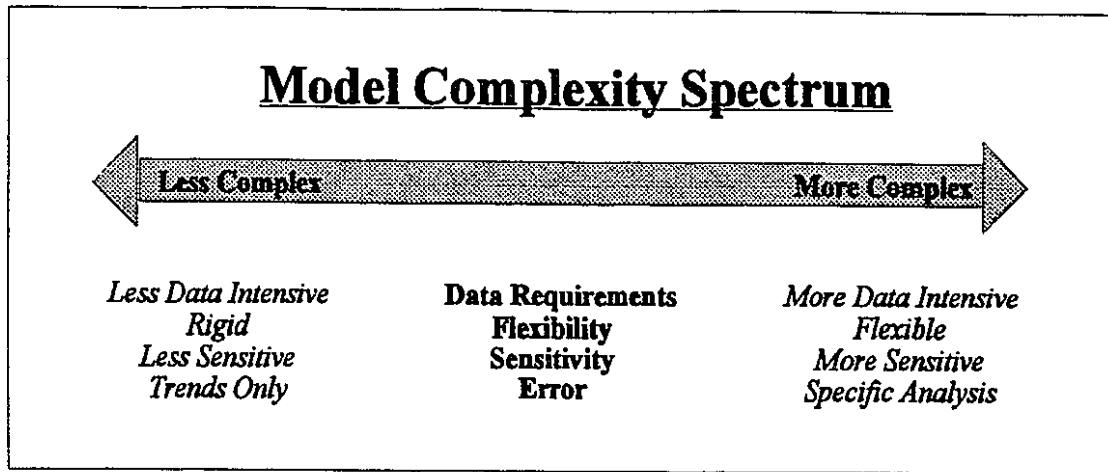


Figure 1. Model Complexity Spectrum

performance. Users often accept the hypothesis that more complex models simulate reality better than simple models, but the relative requirements (e.g. data “cost”) of them are greater. In general, users of simpler models are interested in output information such as relative trends (in time and space), and general magnitudes of the variables of interest. Conversely, users of more complex models are willing to spend the extra cost, (in terms of data, for instance) to get more realistic simulations.

Models of different complexity exhibit different properties based upon their structure. Model sensitivity and data requirements are different depending on whether the model is more or less complex. Using the concept of a spectrum of model complexity (from simple models to complex models), the general trends of several model properties can be investigated.

Figure 1 shows the properties of models across a theoretical complexity spectrum. The arrow across the top of the diagram represents the range of possible model complexities from the simplest on the left to the most complex on the right. Several

properties of the two end-member models of this spectrum are shown: data requirements, flexibility, sensitivity, and error. Each property is described below, and a generalization about the relationship between the model property and model complexity:

1. *Data Requirements*

- More complex models have more parameters and state variables, and therefore require more data. For each parameter, an observed field value, literature value or some empirical estimate is required. For each state variable, an initial condition is required. Therefore, data requirements increase with increasing complexity.

2. *Flexibility*

- Flexibility refers to the number of assumptions employed in the development of the model. Less complex models usually use more restricting assumptions, in order to reduce the number of state variables and parameters, and therefore are less flexible. These assumptions restrict the application of the model to only those situations for which the assumptions are valid. More complex models usually have fewer restrictions, and are therefore more flexible. They can be applied to a wider range of situations. Thus, flexibility tends to increase with increasing complexity.

3. *Sensitivity*

- Sensitivity refers to the amount of change in model output resulting from a

change in model input. The sensitivity of a model to changes in its individual parameters is dependent on several factors including the number and value of other parameters, the mathematical nature of the equation in which the parameter is incorporated, and the range over which the change is made. For each parameter, the sensitivity can be (but is not necessarily) unique. However, for an overall measure of a model's sensitivity, the most significant factor is the number of parameters, since each individual parameter affects sensitivity, and the overall result can be cumulative. Given that each of the parameters in a model introduces sensitivity, and the number of parameters in a model increases with increasing complexity, sensitivity therefore generally increases with increasing complexity. However, due to the fact that sensitivity is dependent on multiple factors, the above trend should be considered only a generalization.

4. *Error*

- Error refers to the quality of the output, and the way that model results should be interpreted. It is a measure of how well the model can simulate given observations. Less complex models cannot simulate real systems as well as more complex models, due to the fact that they use simplifying assumptions and therefore employ an approximation of the processes that make up the system being simulated. For this reason, the output for the model should not be interpreted as being absolutely "true". More complex

models can simulate the system in question with greater detail and with fewer simplifying assumptions, in order to investigate specific spatial or temporal relationships. Conversely, simpler models can be used to evaluate trends only. The way in which error is measured is subjective, and an example would be to measure the sum of absolute errors on fitting a time series of data. Given that modelling error is a measure of how closely the a model can simulate observed data, error is expected to decrease with increasing model complexity.

Defining modelling uncertainty as a function of the model properties above is important to investigating the relationship between modelling uncertainty and model complexity. Given that the overall goal of any simulation is to generate a representation of a system under specific conditions, modelling uncertainty should reflect the quality of that representation.

Uncertainty is defined in terms of error (the difference between simulation and observation), and sensitivity (a measurement of the response of the model). Through the choice of model, a modeller will want to minimize both error and sensitivity, in order to improve the quality of a simulation. At the same time, cost is also a significant factor, which will also influence the choice of how complex a model to use. Having information about the relationships between the sensitivity, error and complexity of a model will allow the modeller to be able to make an informed decision.

Proposing the Uncertainty/Complexity Relationship

The uncertainty/complexity relationship has been defined in terms of two of the several model properties discussed above: model sensitivity and modelling error. A general hypothesis states that more complex models can simulate reality better than simpler models (i.e. less error), and with a greater sensitivity. Simpler models provide a

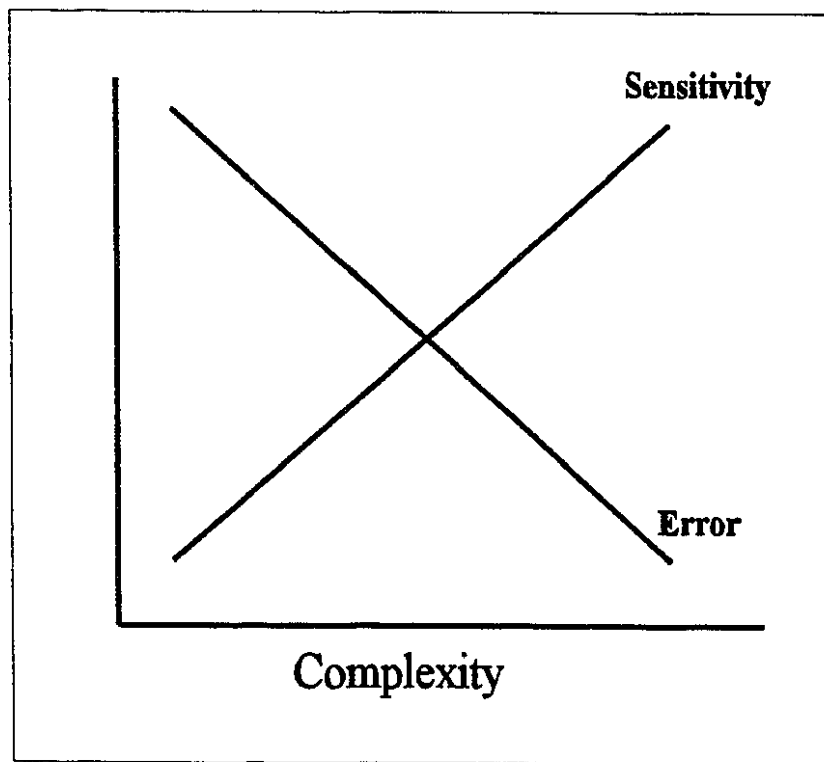


Figure 2. Hypothetical Uncertainty-Complexity Relationship. Uncertainty is defined in terms of both sensitivity and error.

more approximate simulation (i.e. more error), but with less sensitivity.

Figure 2 illustrates the hypothetical relationship between model sensitivity, modelling error and model complexity. Model sensitivity increases with increasing model complexity, due to the larger number of degrees of freedom, and the structure of the interactions between parameters and state variables. Modelling error decreases with

increasing model complexity, as the more complex models are able to better simulate reality with more processes included and fewer simplifying assumptions.

For any given system, the consequences of choosing a model of a given complexity can be illustrated on this diagram. If the model is too complex, the sensitivity will be large (too far to the right), and if the model is not complex enough, the error will be large (too far to the left). The ultimate decision of what is the best place on the complexity axis for each modelling exercise is dependent upon the needs of the modeller, and the purpose of the modelling effort.

Identifiability

Beck (1987) noted that model identifiability is an important issue with respect to model and experimental design. Identifiability is a measure of how well the system is defined by the model. Specifically, it is a measure of whether the model “over-defines” the system. If the number of degrees of freedom of the model is greater than the number of degrees of freedom of the system, then the system is said to be not well identified.

A degree of freedom is an element of change in the model which is independent from all others. For example, a model which is dependent on only two parameters has two degrees of freedom. Either of the two parameters can be changed, which allows independent, unique changes in the model output. However, if a system that is described by a model can be equally well represented by more than one set of distinctly different set of parameters, that means that the number of degrees of freedom in the model is greater than the number of degrees of freedom in the system (as represented in the output).

Changes in one parameter can be compensated by changes in another parameter to give the same output. This is undesirable, since it means that the parameters are ambiguously defined, and are not unique solutions to fitting the model to reality. Therefore the number of parameters, state variables or the complexity of the process equations is too high. With respect to this study, a model that does not identify its system well lies too far to the right on the graph of sensitivity and error vs. complexity, because it is more complex than is required.

Another issue of importance with respect to model identifiability is whether the output used to evaluate the model fully captures the behaviour of the model. The choice of state variable for model evaluation may not be obvious for complex models that have many state variables, since each process may not directly influence each state variable. Looking at the behaviour of one state variable instead of another may highlight a different process in the model. When evaluating the identifiability of a model, the choice of state variable may influence the results. Can the output being studied “see” the changes that are happening in the model? It is important to recognize that the behaviour of a model can be represented differently by looking at different state variables.

Investigating the Uncertainty/Complexity Relationship in a Real System

The hypothetical uncertainty/complexity relationship as shown in Figure 2 can be determined for a given system and set of models. The model complexity spectrum is covered through the use of a specific number of models of various complexities, each evaluated for error and sensitivity during a simulation of the given system. In each case,

the model's error and sensitivity is plotted against its complexity, to determine the uncertainty/complexity relationship.

Because the models are all evaluated against the same data set, it should be noted that this is a "fixed" evaluation of model behaviour. Measuring each model against a series of increasingly complex systems could theoretically better identify the relationship between model complexity and error. This approach, however, was not possible due to the lack of sets of equally high quality data from a series of different systems.

Two case studies were used to evaluate the uncertainty/complexity relationship for different systems. One study involved the relatively simple sorption of zinc onto a sediment, whereas the second study considered a range of models to describe the more complex groundwater transport and fate of a hydrophobic contaminant (Borden site). Model complexity, sensitivity and error were characterized for all model choices in the two case studies. This analysis emphasized the relative differences among models, rather than creating an absolute measurement or detailed assessment of the models. In addition, the models chosen are those commonly used in chemical speciation and groundwater fate and transport studies.

Characterizing Complexity

The complexity of a model is determined by several factors that relate to its structure and the level of detail in the processes that make up the model (where a process is defined as a flow to or from a state variable). The number of parameters and state variables, the sophistication of the mathematical relationships that describe each process,

and the overall number of processes in the model are all properties of the model complexity. For the purposes of this study, the Index of Model Complexity (I_c) is given as:

$$I_c = \sum_{j=1}^N \sum_{i=1}^{n_j} p_i r_i \quad (1)$$

Eq. (1) emphasizes the complexity of the relationships that make up the processes in the model. For each process in the model, the number of parameters is multiplied by the number of mathematical operations in the equations describing the process. This calculation reflects both the complexity of the mathematics, as well as the number of degrees of freedom for that particular algorithm. This number is then summed across all processes for each state variable.

I_c can be conveniently calculated from a Petersen matrix of a given model. The Petersen matrix (Petersen, 1965; Henze, 1987) is a rigorous method of model presentation, where model processes are listed in the rows of the matrix, and state variables in the columns. A Petersen matrix for a hypothetical, two-state model with 2 processes is shown in Table 1.

	X	Y	Process Rate Equation
Process 1	1		$k_1(X)$
Process 2	-a	(1-a)	$k_2(X-Y)$
Stoichiometric Parameters: a			Kinetic Parameters: k_1, k_2

Table 1. Petersen Matrix of Hypothetical Model.

In each element of the table, the stoichiometry of the process is used to describe the rate of change of each of the state variables. This stoichiometry is represented by a term which may contain one or more stoichiometric parameters. The kinetic rate of each process is described by a process rate term in the far right column of the matrix. This process rate term may contain kinetic parameters. The total rate of change for each state variable is equal to the sum (down the column) of the products of the stoichiometric and kinetic terms for each process.

A complexity term can be calculated for each element of the table. According to the above Petersen matrix, the term for X and Process 1 is assumed to be $k_1(X)$, which has 1 parameter (k_1) and 1 mathematical operation (1 multiplication). Therefore, $p_1r_1 = 1 \times 1 = 1$. For X and Process 2, the process rate equation is $-ak_2(X-Y)$, which has 2 parameters (a and k_2) and 3 mathematical operations (2 multiplications and 1 subtraction). Therefore, $p_2r_2 = 2 \times 3 = 6$. Lastly, for Y and Process 2, the process rate equation is $(1-a)k_2(X-Y)$. This equation has 2 parameters (a and k_2) and 4 mathematical operations (2 multiplications and 2 subtractions). Therefore, $p_2r_2 = 2 \times 4 = 8$. Summing down the columns, the complexity value for X is equal to $1 + 6 = 7$, and the complexity value for Y is equal to 8. Totalling across the rows, the overall model's index of complexity is $I_c = 7 + 8 = 15$.

Characterization of Modelling Error

The purpose of the characterization of modelling error is to evaluate the ability of each model (of different complexity) to simulate a real system. In order to evaluate the model in useful way, the characterization of error should reflect the model's ability to

accurately *predict* a time-series under unique circumstances, not merely *fit* an existing time series. Ideally, the best way to evaluate the predictive ability of a model is with two independent time-series datasets from the same system. One dataset is used to calibrate the model, which is then used to predict the other time-series. The fit against the second time-series dataset evaluates the *predictive ability* of the model.

However, acquiring two high-quality, independent datasets from the same system is seldom possible. In the event where only one dataset is available, other methods can be incorporated. When the data set is spatially resolved, and multiple (temporally coincident) time-series are available, a modification of the jackknife method (Miller, 1974) can be used. This method uses the evaluation of a separate fit for each time-series in the dataset. For example, if the dataset contained a time-series for each of 3 different sites in a study area, the model would be calibrated with the first 2 sites and then evaluated against the third. Then the model would be calibrated with sites 1 and 3 and evaluated against number 2, and then again calibrated with 2 and 3 and evaluated against number 1. The total evaluation of the model is an average of the evaluation at each of the 3 sites.

Typical error evaluations employ the use of objective functions, which calculate the difference between simulated results and actual observations. Typical objective function types use the absolute sum of the differences, the sum of squares of the differences, or the sum of relative differences.

The sum of the square of the differences was used for objective function values in both case studies. Due to the presence of only 1 dataset, the error evaluation for Case Study I simply was equal to the value of the objective function. Due to the presence of

multiple, temporally co-incident datasets, the jackknife method was used for error evaluation for Case Study II.

Characterization of Model Sensitivity

The purpose of the characterization of modelling sensitivity is to evaluate the amount of change in model output caused by changes in input. It is a measurement of how the output of a model is controlled by the input parameters and initial conditions. A model which shows a large change in output relative to a small change in input parameters is said to be sensitive. In order to characterize sensitivity in a manner that can be related to model complexity, the method should capture the difference in modelling sensitivity between models of different complexity.

The Monte Carlo method is a common tool in evaluating model sensitivity, which is widely used in many forms (Beck, 1987, Brattin, et al., 1996). This method uses assumed distributions for each parameter in the study. Sets of parameter values are created by sampling from each of the parameter distributions, and the model is run repeatedly with each of the parameter sets. The multiple sets of time-series model output are then evaluated to determine the spread or precision of the output.

To employ the Monte Carlo method as a sensitivity analysis for these case studies, the parameter distributions are determined by fitting, rather than by assuming some arbitrary probability density function. Because the objective of the sensitivity method is to distinguish between sensitivities in different model structures, the distributions should be a function of the model structure.

This can be done through the use of the Reverse Monte Carlo Method, using the following steps:

- Add random noise to the (output) time-series data set repeatedly, to create many sets
- Fit the model to each data set, recording the parameter values of each fit
- Calculate the distribution for each parameter from the multiple parameter values.

The parameter distributions created through this method are a function of the model structure. The rest of the sensitivity procedure follows the regular Monte Carlo Method, using the following steps:

- Independently sample many parameter values from each distribution to create multiple parameter sets
- Run the simulation using each parameter set, recording the time-series model output
- From all of the model output time-series, calculate a envelope of +/- one standard deviation around the mean at each timestep in the series
- Calculate the area in the envelope.

The overall sensitivity is a value proportional to the area of the envelope. This method, where the forward Monte Carlo analysis attempts to recreate the original noise, emphasizes the relative differences in sensitivity based on model structure. The result is a

sensitivity measurement that is characteristic of a *calibrated* model, and not a property of arbitrarily defined parameter distributions. This method was used to evaluate sensitivity in both case studies.

CASE STUDY I: Sorption of a Radioactive Tracer on Sediment Surfaces

System

A simple sorption system was established for the purposes of studying the sorption of radioactive zinc onto sediments in solution (LeBeuf, 1992). A sediment suspension was prepared, and a known amount of radioactive zinc tracer was added at t_0 . The relative amount of radioactivity in solution was measured over a period of 17 days.

Models

The relative radioactivity in solution was simulated using three simple sorption models, with flow equations of differing complexities. The two simplest models are 2-box models (where the 2 boxes are sorbed phase and solute phase metal concentrations), and differ in that one is an equilibrium sorption model, and the other uses sorption kinetics. The equilibrium model assumes that the sorption reaction happens instantaneously, and that the sorption reaction goes to completion at an infinitely high rate. In the case of the kinetic model, the sorption reaction goes to completion at a rate described by a rate coefficient. The third and most complex model is a 3-box model which has one solute phase and two sorbed phases, and both equilibrium and kinetic flows. The models are described in Table 2:

Model Name	Sorption Process	Model Complexity, I _c
2-Box Equil.	Equilibrium	6
2-Box Kinetic	Kinetic	12
3-Box	Both Equilibrium and Kinetic	18

Table 2. Case Study I Models.

For Case Study I, the objective function used for all fitting was the sum of the square of difference between each data point and the simulated value, summed across all points in the time series. The Simplex Method was used to minimize this objective function.

Results

Figure 3 shows the results of the error and sensitivity analyses for Case Study I. Error decreases with increasing complexity, as hypothesized earlier. The more complex 3-box model was able to simulate the observed relative radioactivity observations in the solute phase better than the simpler 2-box models.

Sensitivity shows an increasing trend with increasing complexity. The two simpler 2-box models have approximately the same sensitivity, while the most complex 3-box model has significantly more sensitivity. The 3-box model uses more parameters to define the distribution and rate of sorption of zinc than the two simpler 2-box models, and exhibits more sensitivity.

CASE STUDY II: Transport of a Groundwater Tracer Plume

System

A study was carried out to simulate the transport of an organic tracer in a

homogeneous sand aquifer. 7 models of varying complexity were used to model the flow of the plume over a 1-½ year period. Time-series data sets for 5 locations within the aquifer were used in the evaluation.

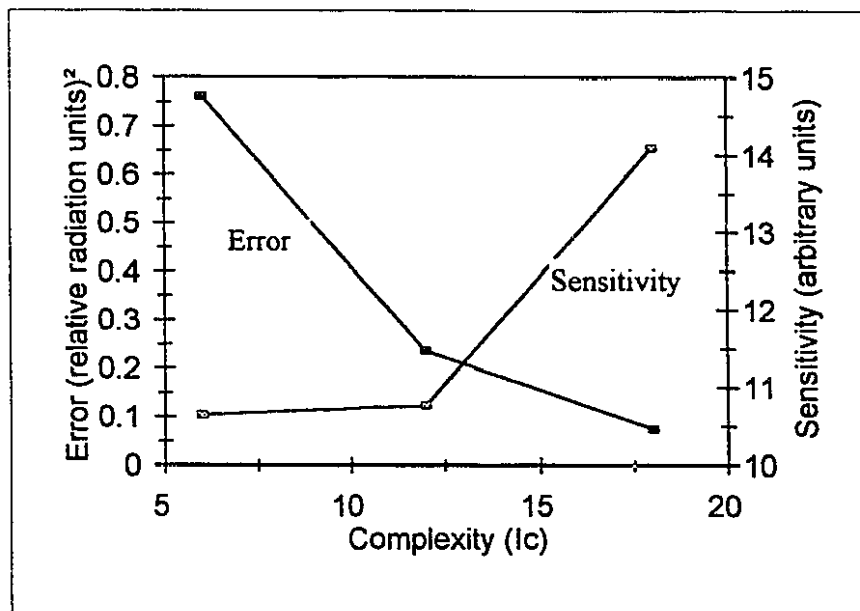


Figure 3. Uncertainty vs. Complexity - Case Study L

The plume data used in the study are from the Borden Groundwater Data Set (Roberts and McKay, 1990). This highly spatially resolved data set was created for the purposes of groundwater transport research, and consists of lengthy time series data for many sites in a homogeneous sand aquifer at the Canadian Forces Base in Borden, Ontario. A group of 7 organic and inorganic tracers were injected into the aquifer in August of 1982, and the plume was monitored for a period of over 3 years. For the purposes of this study, only the transport of the non-conservative carbon tetrachloride tracer is simulated.

Models

The models used in the study are all built upon a basic transport model (of a standard complexity) for flow in a 3-dimensional homogeneous media. To create models of different complexities, additional sorption and degradation submodels were added to the basic transport model. By adding sorption or degradation models of varying complexities, the overall model complexity was adjusted. Table 3 summarizes these models.

Model Number	Sorption Submodel	Degradation Submodel	Model Complexity I_c
1	None	None	16
2	Linear Isotherm Equilibrium Sorption	None	22
3	Linear Isotherm Kinetic Sorption	None	28
4	Linear Isotherm Equilibrium Sorption	1 st Order Reaction	24
5	Linear Isotherm Kinetic Sorption	1 st Order Reaction	29
6	Non-Linear Isotherm Kinetic Sorption	1 st Order Reaction	41
7	Monod-type Isotherm Kinetic Sorption	1 st Order Reaction	47

Table 3. Case Study II Models.

Sorption Submodels

The sorption submodel describes the retardation of flow through the media, as the solute-phase pollutant sorbs to the media particles. In order to create several sorption submodels of varying complexities, the sorption mechanism is simulated with different

types of sorption relationships. The differences between various relationships are summarized below:

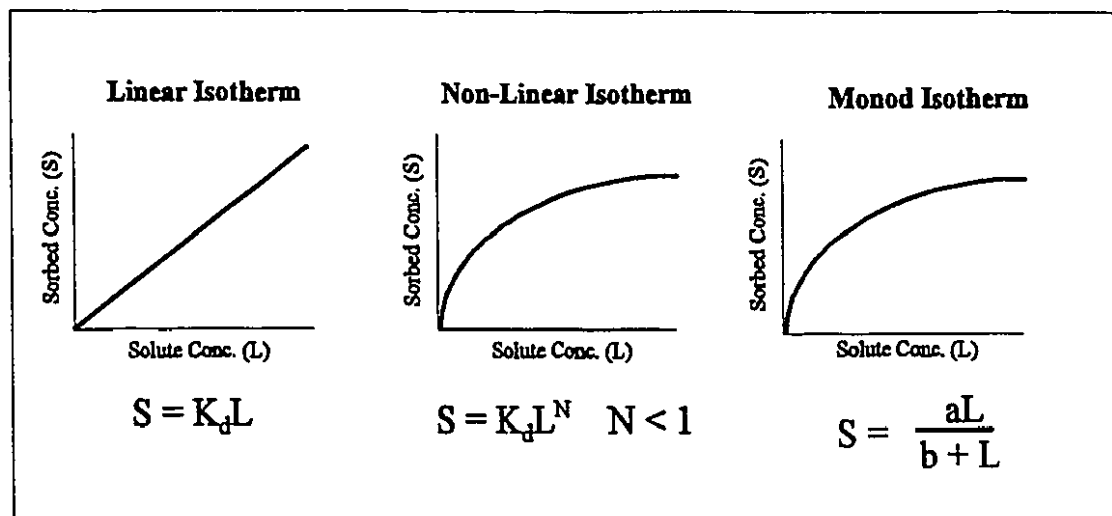


Figure 4. Linear vs. Non-Linear Isotherm Relationships.

1) *Linear vs. Non-Linear vs. Monod Sorption Isotherm*

The relationship between sorbed-phase concentration and solute-phase concentration is described through the use of a sorption isotherm. This isotherm represents the concentration of the sorbed-phase pollutant as a function of the concentration of the solute-phase concentration. Three of the most commonly used sorption isotherms are shown in Figure 4, along with the equations used to describe them.

Note that the non-linear isotherm has one more parameter used to describe it (K_d and N versus just K_d) and one more mathematical operation (multiplication and exponent versus just multiplication) than the linear isotherm. The Monod-type isotherm has two parameters and three operations. These properties make the

non-linear isotherm a more complex sorption submodel than the linear form, and the Monod-type the most complex of the three.

2) *Equilibrium vs. Kinetic Sorption*

The speed at which the sorption mechanism takes place is described by either an equilibrium or a kinetic relationship in the sorption submodel. The equilibrium relationship simulates sorption as happening instantaneously, with the sorption reaction going to completion at an infinitely high rate. In the case of the kinetic model, the sorption reaction goes to completion at a rate described by a rate coefficient. The difference in complexity between these two types of submodels is the increase in the number of parameters (the additional rate coefficient in the case of the kinetic model).

The above two elements of sorption submodels can be combined together in different ways. For example, the simplest sorption submodel is the equilibrium linear submodel, and the most complex is the kinetic Monod-type submodel.

Degradation Submodels

The degradation submodel describes the transformation of the pollutant to a different state. In the case of this study, the degradation of the non-conservative tracer removes it from the system. This process is modelled using reaction mechanisms of different complexities.

The two different degradation submodels used are:

- No Degradation: there is no degradation mechanism

- **1st Order Degradation: the rate of degradation is proportional to the concentration**

The 1st order degradation submodel is more complex than the “no degradation” format, due to the use of a degradation rate constant.

For the purposes of Case Study II, the fitting of the models to the real data was done through the use of an objective function that described the difference between the real data and the simulation run in the following manner:

- The objective function (for one site) is the sum of the square of the difference between each data point and the simulated value, summed for all data points in the time series.
- The above value is calculated for each site and summed across all sites, using weighting functions to remove the bias of any difference in numbers of data points at each site.

The model was then fit to the data using the Simplex Method to minimize the objective function as described above.

Results

The results of the sensitivity method for all 7 models are shown in Figure 5. There is not a continuous linear relationship as hypothesized in Figure 2, however several interesting features are shown. For example, two distinct trends are visible - one for models without degradation (Trend #1), and one for models with degradation (Trend #2). Note that Model #2 does not fit the trend of models with degradation. In progressing from less complex to more complex models, both trends show increasing sensitivity. The models shown in Trend #1 range from no sorption to equilibrium sorption to kinetic sorption. As the complexity of the model increases, the sensitivity increases, as hypothesized. The same is also true in Trend #2, where the increase in complexity of

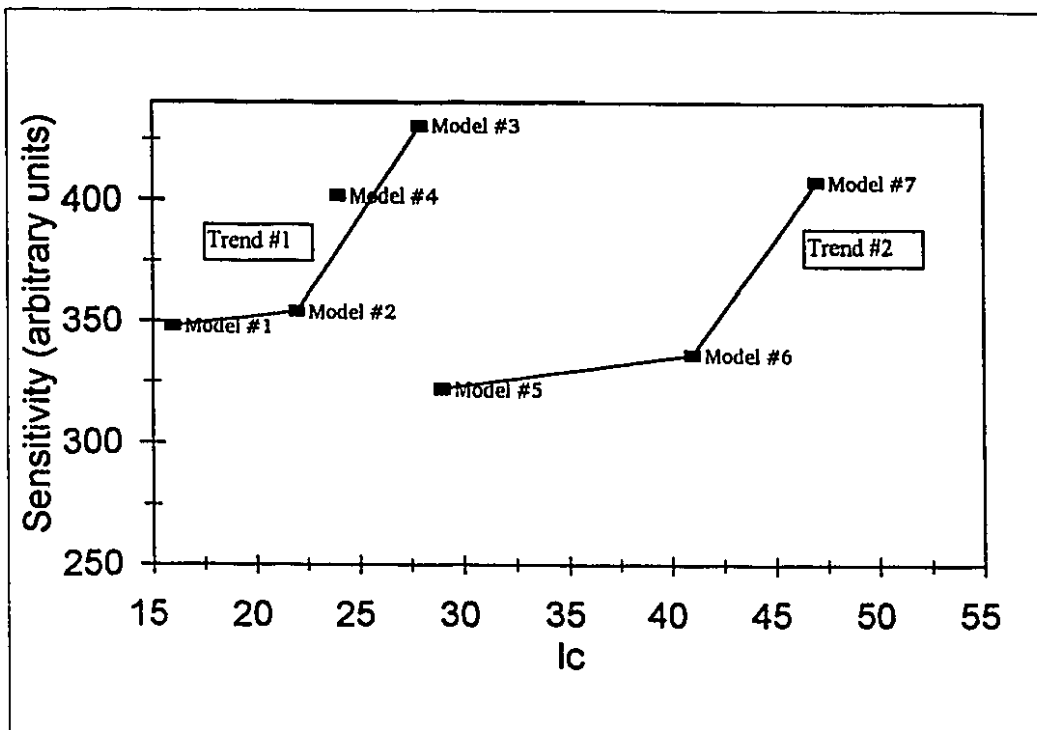


Figure 5. Sensitivity vs. Complexity - Case Study II.

kinetic sorption models with degradation goes from linear to non-linear to Monod-type equations. In this case, the sensitivity also increases with increasing complexity.

There is not a clear trend present when comparing (otherwise similar) equilibrium and kinetic models. The two linear sorption models with no degradation (Models #2 and #3) differ only in the equilibrium/kinetic sense. The kinetic model has significantly higher sensitivity, according to the results shown in Figure 5. However, the opposite is true for these same models with degradation. The equilibrium model (Model #4) has a higher sensitivity than the kinetic model (Model #5).

It can be concluded from these results that the hypothesized relationship of increasing sensitivity with increasing model complexity holds true for certain types of complexity increases. The sensitivity increases for increasingly complex sorption isotherms (linear vs. non-linear vs. Monod - Trend #2), and also for increasingly complex models without degradation (Trend #1). However, there is no consistent trend in sensitivity when comparing equilibrium models to kinetic models, or comparing models with no degradation to models with degradation. There is no overall trend when viewing all 7 models as a group, due to the discontinuity present when switching from models without degradation to those with degradation.

Figure 6 shows error vs. complexity for the seven groundwater models used in Case Study II. There is no clear trend regarding error as was shown in Case Study I. Several of the models show roughly the same error (in the range of 1000-1300 mg/L), while one model (Model #4, $I_c = 24$) has considerably higher error. While there is no general consistent trend, it can be noted that the more complex models do not provide

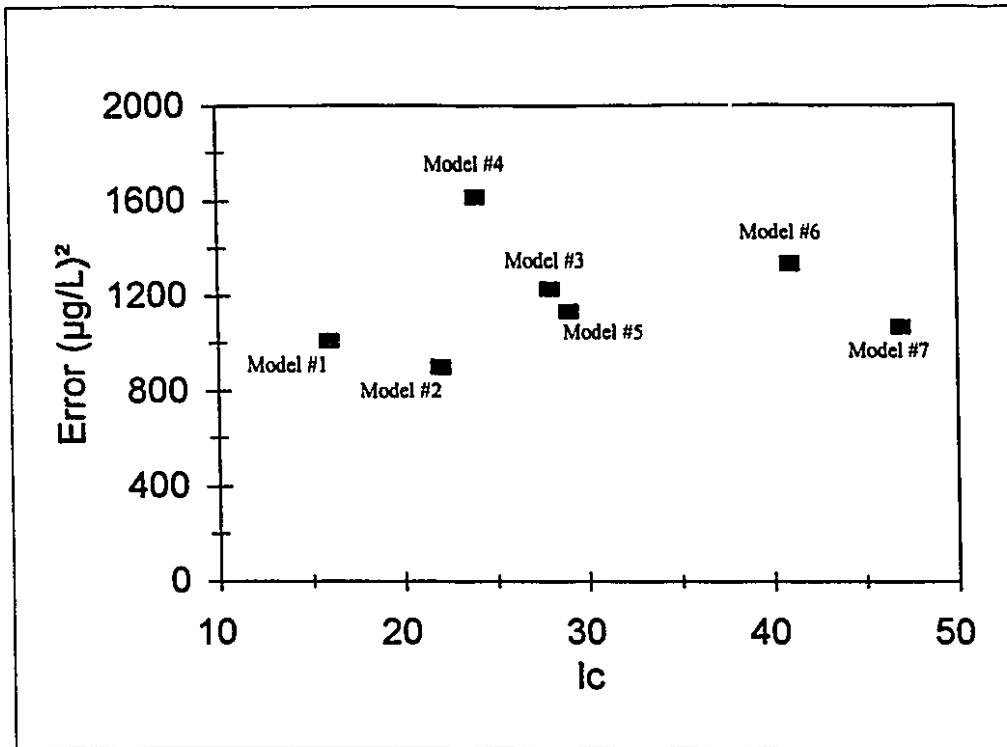


Figure 6. Error vs. Complexity - Case Study II.

significantly better fits than simpler models, for the models used in this Case Study II.

Overall Model Utility

It is possible to use the information presented in Figures 3, 5 and 6 to draw conclusions about the use of the models in the study. In order to make a decision about which is the “best” model for any given modelling exercise, a combination error/sensitivity statistic can be calculated. This statistic measures the “utility” of the model, with respect to modelling uncertainty as defined in this study. A modeller wishes to minimize both error and sensitivity, so a combined measure U is needed, which increases as error and sensitivity decrease. The following equation proposes an index of utility by measuring a

scaled “distance from origin” on a graph of sensitivity vs. error.

$$U_i = 1 - \sqrt{\frac{k_s S_i^2 + k_e E_i^2}{(k_s + k_e)}} \quad (2)$$

U varies between 0 and 1, where the larger the value of U, the greater the model utility. The values of S and E for each model should be normalized to 1 by dividing all values by the maximum sensitivity and error value, respectively. The use of a statistic like the one calculated in Eq.(2) requires an assumption about how the modeller values error and sensitivity, relatively to each other. If error and sensitivity are valued equally, then k_s and k_e should both be set to 1. Setting k_s or k_e to a value greater than one will emphasize

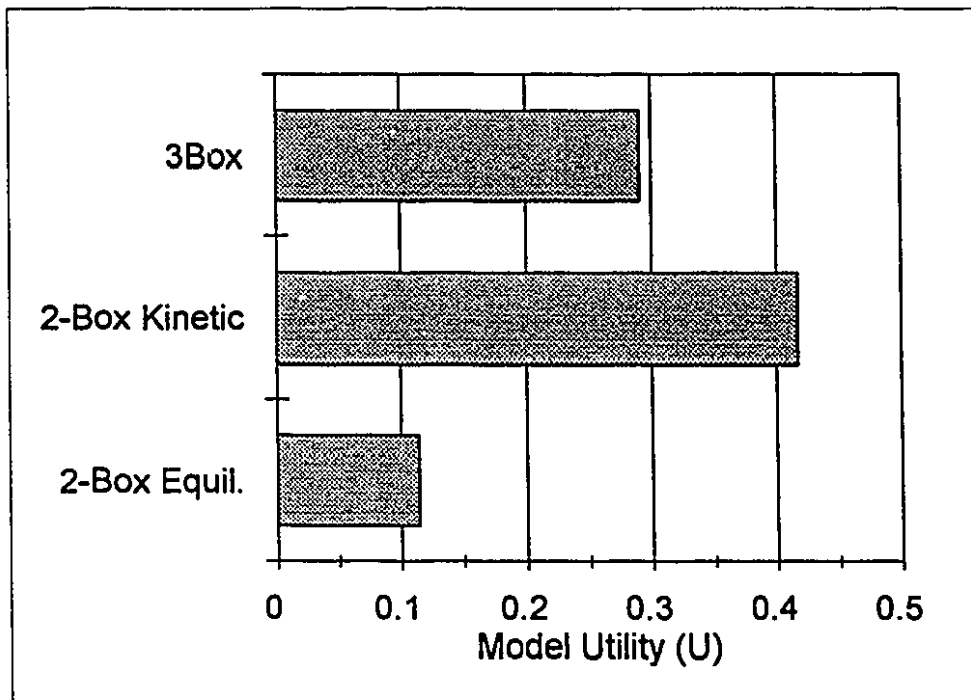


Figure 7. Model Utility - Case Study I.

that particular characteristic. The values of U_i are relative, and can only be used to compare models within the same study.

For illustrative purposes, U has been calculated for the two case studies. Error and sensitivity were valued equally, so k_e and k_s were set to 1. The results for Case Study I are shown in Figure 7. The 2-box kinetic sorption model is the “best” model, according to the definition of utility in Eq.(2). The medium complexity model in this case has the best combination of error and sensitivity.

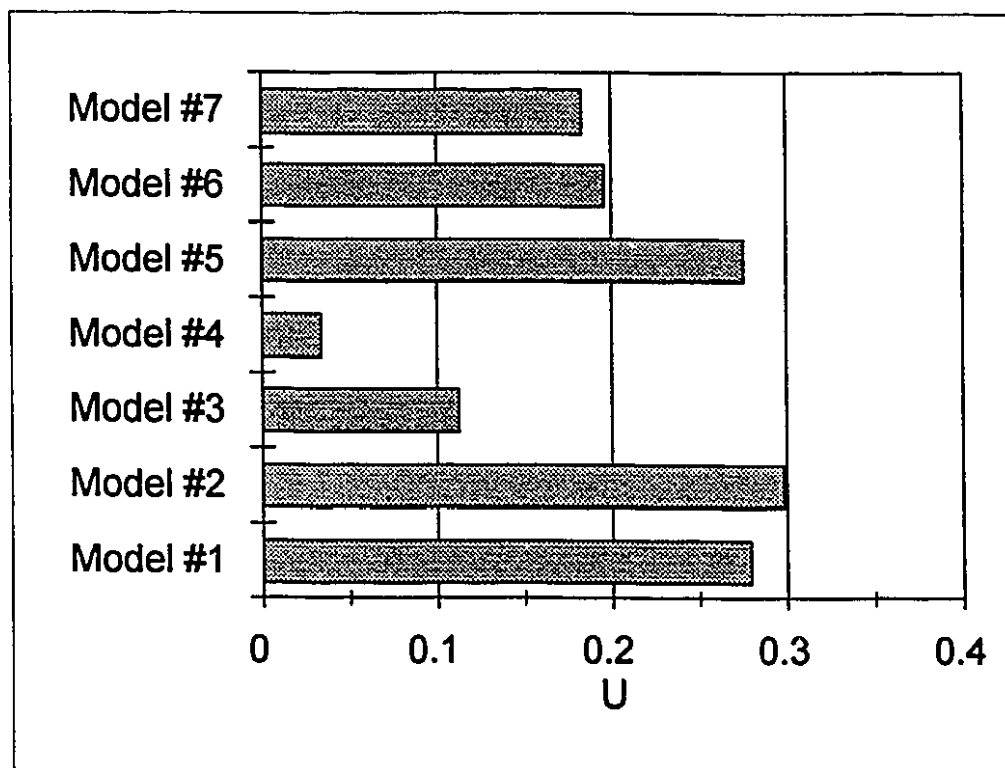


Figure 8. Model Utility - Case Study II.

The results from Case Study II are shown in Figure 8. Due to the fact that the sensitivity of Model #3 and Model #4 are relatively high, without significantly less error than the other models, they stand out as the 2 lowest utility models in the study.

Amongst all 7 models, Model #2 has the best model utility.

Conclusions

The goal of this study was to identify and characterize modelling uncertainty as a function of the complexity of a model's structure. The methods presented can do this in a relative manner, and the case studies show that in some cases more complex models are not necessarily better. It is not possible to relate uncertainty and complexity in a completely general way, but empirical studies can be useful in selecting the appropriate model for the job.

As a general observation about the utility of models in both of the case studies presented, it appears that the increased sensitivity associated with more complex models outweighs the benefits of the marginally (if at all) better fits these models provide. The utility ratings presented in Case Studies I and II are specific to the models and data sets used, but the methods used for characterization of modelling uncertainty and complexity can be extended to use with models and data sets of any kind. Interpretation of the results is dependent on how the modeller chooses to define error, and the relative value or importance assigned to error and sensitivity. However, the evaluation of a model, based on its properties related to uncertainty, is a valuable task. By being able to use information such as a measurement of model utility, the selection of an appropriate model can help to reduce not only simulation issues such as sensitivity and error, but cost as well, by reducing the tendency to employ overly-complex models.

Acknowledgements

S. Snowling would like to acknowledge the valuable assistance of Imre Takács and Hydromantis, Inc., for their help with the GPS-X modelling software, and Joe DePinto of SUNY-Buffalo for his help in the formulation of the project.

References

- Beck, M. (1987) Water Quality Modeling: A review of the analysis of uncertainty. *Water Resources Research*, **23**: 1393-1442
- Brattin, W., Barry, T., and Chiu, N. (1996) Monte Carlo modeling with uncertain probability density functions. *Human and Ecological Risk Assessment*, **2**: 820-840.
- Halfon, E. (1983) Is there a best model structure?: 1. Modeling the fate of a toxic substance in a lake. *Ecological Modelling*, **20**: 135-152.
- Henze, M., Grady, C.P.L., Gujer, W., Marais, G v R, and Matsuo, T. (1987). *Activated sludge model no. 1*. IAWPRC Scientific and Technical Report No. 1, IAWPRC, London.
- Kohler, M. (1996) Experimental investigation and modeling of uranium(VI) transport under variable chemical conditions. *Water Resources Research*, **32**: 3539-3551.
- Konikow, L. and Bredehoeft, J. (1992) Ground-water models cannot be validated. *Advances in Water Resources*, **15**: 75-83.
- LeBeuf, M. (1992) *Kinetics of Trace Metal Interactions with Lake Sediments*, Ph.D Thesis, McMaster University, Hamilton, Ontario.
- Miller, R. (1974) The jackknife - a review. *Biometrika*, **61**: 1-15.
- Oreskes, N., Shrader-Frechette, K., and Belitz, K. (1994) Verification, validation, and confirmation of numerical models in the earth sciences. *Science*, **263**: 641-646.
- Pearl, J. (1978) On the connection between the complexity and credibility of inferred models. *International Journal of General Systems* **4**: 255-264.

- Petersen, E. (1965) *Chemical Reaction Analysis*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- Reckhow, K., and Chapra, S. (1983) Confirmation of water quality models. *Ecological Modelling*, 20:113-133.
- Roberts, P., and McKay, D. (Eds.) (1990). *Borden Data Set: Natural Gradient Experiment on Solute Transport in a Sand Aquifer*. International Ground Water Modeling Center, Colorado School of Mines, Golden, CO.
- Smith, D.S., Adams, N.W.H. and Kramer, J.R. (1999) Resolving uncertainty in chemical speciation determinations, *Geochim Cosmochim Acta.*, 63, Nov (in press).
- Smith, V., and Vaughan, W. (1980) The implications of model complexity for environmental management. *Journal of Environmental Economics and Management*, 7: 184-208.
- Usunoff, E., Carrera, J., and Mousavi, S.F. (1992) An approach to the design of experiments for discriminating among alternative conceptual models, *Advances in Water Resources*, 15: 199-214.
- Wagenet, R., and Rao, P. (1990) Modeling Pesticide Fate in Soils. In: *Pesticides in the Soil Environment: Processes, Impacts, and Modeling*, (Cheng, H., ed.), Soil Science Society of America Book Series (No. 2), S.S.S.A., Inc., Madison, Wisconsin, pp. 351-399.

Appendix III: Input-Output Modelling and Groundwater Remediation

Abstract:

To study the relationship between agricultural pesticide regulation and economic impact, a new input-output model structure is proposed which incorporates the production associated with groundwater remediation as an economic sector. Pesticide use is related to agricultural production through the use of non-linear relationships in order to estimate the amount of economic production associated with pesticide contamination removal. The relationships between economic impact and pesticide regulation can be investigated for various types of agricultural and economic scenarios. A numerical example of the application of the model to a Southern Ontario study area estimates a maximum 5.8% increase in sectoral production with the inclusion of remediation production into the economy.

Key Words: Input-output analysis, modelling, pesticide, groundwater, remediation.

Introduction

Synthetic pesticide use has become a major component in agricultural management over the last fifty years (Madhun and Freed, 1990). Adverse impact on the natural environment from the use of pesticides has been recorded since the early 1960's (Carson, 1962), and continues to the present day.

Groundwater is a very important resource in rural areas. It has been estimated to be the sole drinking water source for 50% of the general population, and over 95% of the rural population in the U.S. (Gladwell, 1989). Impacts on this resource by non-point

source contaminants, such as agricultural pesticides, can be significant and have warranted regulated control of pesticide use.

To study the relationship between pesticide regulation and economic impact, this paper proposes the use of a Leontief-style input-output model linked with an environmental model such as EPIC (Erosion/Productivity Impact Calculator, Sharpley and Williams, 1990), to estimate the impacts of various levels of pesticide use control on economic production. This model can be used to estimate the economic production of all sectors of the economy, based on the allowable pesticide groundwater concentration and economic information. Although feasible under certain assumptions, no attempt is made in this paper to link production to employment impacts.

Relating economic impact and environmental impact through the use of input-output models allows for investigation into the regional effects of environmental management of pesticide use. While other studies have addressed this issue at a farm-level scale (Johnson, et al, 1991), input-output models are most effective at illustrating economy-wide trends.

The purpose of this paper is to introduce an input-output model structure which expands upon previous economic-environmental I/O models, and is specifically developed for agricultural pesticide applications. The use of an environmental feedback effect is included in the model, which adds to the traditional Leontief model structure, and a non-linear relationship is included to more realistically simulate groundwater contaminant behaviour under different levels of application.

The paper outlines the development of the new input-output model structure, and

an example of model output (for a Southern Ontario agricultural area) is shown to demonstrate the difference between the newly proposed model structure and that of the traditional input-output model.

Review of Input-Output Models

The input-output model, as originally proposed by Wassily Leontief in the 1930's, uses data on the amounts of goods and services exchanged between various sectors in the economy to relate economic production to exogenous final demand.

There are many examples in the literature of environmental input-output models. Leontief (1970) and Chen (1973) relate environmental impact and input-output analysis in a general, comprehensive manner. Lonergan and Cocklin (1985) summarize several approaches to environmental planning with input-output models, and present a good analysis of economic-ecologic models proposed by several authors. Heslin and Hobbs (1990a) studied the impact of oil and gas brine disposal regulations on the economy of Ohio using Leontief models. Gay and Proops (1993) assessed carbon dioxide production in the UK economy with input-output analysis, and Helsin and Hobbs (1990b) analysed the economic impacts of SO₂ emission reduction strategies. Anaman (1994) studied the secondary economic effects of a screwworm fly invasion of Australia. In each case, an input-output model was used in conjunction with environmental models and/or field observations to relate the environmental and economic impacts.

In order to investigate the use of input-output models, a review of their structure is useful. Table 1 shows an example of a simple input-output transactions table.

Table 1: Simple Input-Output Transactions Table.

		Purchasing Sector					
		1	2	i	n
Selling Sector	1	z_{11}	z_{12}	z_{1i}	z_{1n}
	2	z_{21}	z_{22}	z_{2i}	z_{2n}
	⋮	⋮	⋮	⋮	⋮
	i	z_{i1}	z_{i2}	z_{ii}	z_{in}
	⋮	⋮	⋮	⋮	⋮
	n	z_{n1}	z_{n2}	z_{ni}	z_{nn}

An element of the transactions table, z_{ij} , represents the monetary value of goods or services flowing from sector i (shown on the left side of the table) to sector j (shown across the top of the table). Table 1 shows the economy for n sectors, and collectively the elements of the table are referred to in matrix format as the *transactions matrix*, Z .

The total output produced by sector i , can be calculated from the following relationship:

$$x_i = \sum_{j=1}^n z_{ij} + y_i \quad i = 1, \dots, n \quad (1)$$

where x_i = total output (of goods and services) of sector i
 y_i = total final demand for output of industry i .

This equation shows that a sector's production is made up of two parts - the *indirect* demand, which is the sale of goods and services to other sectors, and *final* demand, which is the sale of goods and services to final consumption, investment, export, or to the public sector.

A fundamental assumption of input-output models is the linear relationship that is assumed between the amount of goods produced by a given sector (output) and the amount of goods used by that sector in its production process (input). This linear relationship means that if the amount of output is doubled, then the amount of each of the input products required is also doubled. Such an assumption assumes that the technology and input mix is constant over time and over all levels of output, implying that economies of scale are ignored.

To make use of this assumption, an $n \times n$ matrix A is calculated as follows:

$$a_{ij} = \frac{z_{ij}}{x_j} \quad (2)$$

The element a_{ij} is known as the *technical coefficient* (or *direct input coefficient*) between sectors i and j , and represents the dollar value of input from sector i used in the production of one dollar's worth of output of sector j . Rearranging (2) and substituting into (1) results in the following relationship:

$$x_i = \sum_{j=1}^n a_{ij}x_j + y_i \quad i = 1, \dots, n \quad (3)$$

Letting $n \times 1$ column vectors X and Y represent the production and demand values for each of n sectors, (3) can be rewritten in matrix form as:

$$X = AX + Y \quad (4)$$

To be able to solve for each sector's production, X , (4) can be rearranged as:

$$X = (I - A)^{-1}Y \quad (5)$$

$(I - A)^{-1}$ is referred to as the *Leontief Inverse*, and is used to determine the production of each sector from the demand for the products of all other sectors in the economy. Elements of the $n \times n$ Leontief Inverse matrix are referred to as α_{ij} . Included in the production calculation is the portion of the sector's product going to final demand, as well as the portion of the sector's product becoming inputs to other sectors (Miller and Blair, 1985).

Environmental Input-Output Analysis

The traditional input-output analysis method can be augmented with additional information, to study the relationship between the economy and the environment. The most straightforward way is to assume simple, linear relationships that relate resource use and pollutant generation to sectoral output. For example:

v_{kj} = amount of pollutant of type k , generated per dollar worth of output in sector j .

V is an $m \times n$ matrix, where m is the number of pollutants under consideration.

Therefore, the $m \times 1$ vector of total pollutant levels, V^* , would be:

$$V^* = VX \tag{7}$$

The elements of which, v^*_k , represent the total amount of pollutant k produced by all sectors. Then, using the traditional Leontief model, the pollutant levels can be directly calculated as a function of demand:

$$V^* = [V(I - A)^{-1}] Y \tag{8}$$

A more complex and detailed way of relating environmental and economic effects

involves the use of “ecological commodities”, which become part of the input commodities mix that are used in the production of each sector’s output. Victor (1972) proposed a comprehensive approach to the use of ecological commodities, involving the flow of such commodities both to and from the environment.

Pesticide Impact Input-Output Model

The analysis of economic and environmental impact in agricultural sectors is one that has been approached by several methods (and different model structures) over the last decade (Johnson, et al, 1991; Mapp, et al., 1994 for example). This paper however, deals only with the use of input-output models for economic-environmental studies.

The traditional Leontief input-output model structure forms the basis of the proposed Pesticide Impact Input-Output Model. The use of input-output models to study pesticide regulation has been addressed before by Palmieri (1982), who analysed the effects of non-point source pollution controls on small, regional economies. Several different scenarios were analysed, including switching to a environmentally safer, but more expensive, alternative pesticide. The model showed that agricultural revenues (more specifically, return over variable costs) were reduced only by 0.7%, whereas other pollution controls such as nitrogen restriction, reduced revenues by 2.7%.

Rather than looking at switching to alternative pesticides, the Pesticide Impact Input-Output Model proposed in this paper will relate back into the economy the production generated by pesticide contamination removal. The amount of production from the agricultural sector is used to determine the amount of pesticide present in the

groundwater. The amount of pesticide present in excess of the allowable amount is used to determine the production associated with groundwater pollution abatement, which is then linked back into the overall economy. This method was proposed by Leontief (1970), expanded by Chen (1973) and used in Xu et al (1994) for stormwater quality planning. The model will then be able to estimate the production of all sectors of the economy based on physical characteristics of the study region, as well as the allowable amount of pesticide in the groundwater. The effect of changing the regulated amount on the economy can be studied with different levels of regulation, environmental or agricultural characteristics, or groundwater remediation costs. As such, these data requirements are simpler than those of Victor (1972), which requires monetary valuation of flows to, from, and within the environment. The data requirements of the proposed model are straightforward input-output tables, and standard environmental data used in models such as EPIC (Erosion/Productivity Impact Calculator), developed by the U.S. Department of Agriculture (Sharpley and Williams, 1990).

The following shows a summary of the structures which relate pollution and production, from the proposed model as well as previous models :

Leontief (1970): **Production → Pollution**

Xu, et al (1994): **Production → Land Use → Pollution**

***Pesticide
Impact I/O Model:*** **Production (Agriculture) → Pesticide Use → Pollution**

The most significant difference between the methods used in models such as Leontief (1970) and Xu et al (1994) and the model proposed in this paper is in the way that pollutant generation is estimated. As noted earlier, sectoral output has been used to estimate the amount of pollutant generated by various sectors of the economy through the use of linear relationships such as those shown in eq (7). A variation on this idea is to linearly relate pollution and production through one or more steps. Xu, et al (1994) do this by relating production to land use, and then land use to pollution. In the proposed Pesticide Impact Input-Output Model, production is used to estimate pollution through two steps, including a non-linear relationship. Production (from the agricultural sector only) is non-linearly related to pesticide use, which is in turn related to pesticide concentration in groundwater.

While the ultimate goal of these relationships is the same - to estimate pollution generation from sectoral production - the steps by which this is done are different. Each model uses different assumptions and relationships to link each part of the chain.

Pesticide Impact Input-Output Model: General Structure

This section outlines in detail the development of the Pesticide Impact Input-Output Model. There are two distinct types of relationships found in the model - the economic relationships described by the traditional input-output model, and those used to relate production and pollution. Application of the model requires some preliminary analysis of the study area, to determine the non-linear relationship between agricultural production and pesticide use. This relationship can be determined from empirical

research, or can be estimated from an agricultural model.

To illustrate the structure of the economic input-output model, the block diagram style proposed by Chen (1973) will be used. Chen's block diagram style uses boxes and arrows to represent different parts of the model structure. An arrow represents a vector or matrix of information about the economy, and a box represents the multiplication (or other transformation) of the matrix. For example, if an arrow representing a matrix B entered a box labelled X , the resulting transformation is the matrix multiplication BX .

In the block diagram schematic, an open circle is a node which is used in forming an equation in the model. Each arrow leaving the circle represents a term on the left-hand side of the equation, and each arrow entering the circle is a term on the right hand side of the equation. Each of the arrows is accompanied by a positive or negative sign to indicate whether that term is added or subtracted from that side of the equation. If no sign is given, positive is assumed. A small, closed circle represents a point where the arrow splits so that the matrix may be used in more than one calculation.

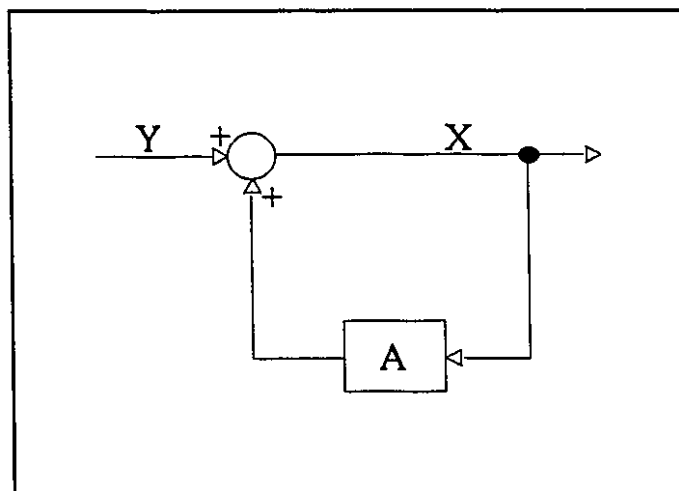


Figure 1. Simple Leontief Model.

The development of the model starts with a standard, industry-by-industry Leontief input-output model, with an economy of n sectors. Figure 1 illustrates the model in the block diagram format.

By deriving the equation from the circle on the left-hand side of the diagram, the Leontief equation as shown in (4), is achieved:

$$X = AX + Y \quad (9)$$

The production of the agricultural sector is then used to estimate the amount of pesticide applied to farms in the study region. EPIC is a model that can be used to estimate the amount of crops grown under different agricultural practices. The environmental characteristics of the study region are used along with several different levels of pesticide application to estimate crop yield (in monetary units). The results are then tabulated in a lookup table, which is labelled N in the schematic block diagram. For each level of agricultural crop yield (in dollars), the model can use the lookup table to find the corresponding level of pesticide application (in kg/ha).

In order to properly incorporate the lookup table into to matrix algebra used in this model, a $1 \times n$ row-vector N is used. The row vector contains all zeros, except for the element corresponding to the agricultural sector. This corresponding element will contain the appropriate value from the lookup table (according to the value of agricultural production). For example, a 4 sector economy with the first sector being the agricultural sector, would have a corresponding row-vector N as follows:

$$N = [n_j \quad 0 \quad 0 \quad 0] \quad (10)$$

The element n_j is the appropriate value from the lookup table (according to the agricultural production, X_j) in units of kg/ha·\$. When the column-vector X is pre-multiplied by the row-vector N , the result will be a scalar, which represents the amount of pesticide application (in units of kg/ha).

The concentration of pesticide found in the groundwater in the study region is assumed to be linearly related to the amount of pesticide applied to the fields. Let the scalar M represent this relationship:

$$M = \frac{\text{Groundwater pesticide concentration (mg / L)}}{\text{Amount of pesticide applied (kg / ha)}} \quad (11)$$

Using the pesticide application determined from the look-up table calculation, the concentration of pesticide in groundwater in the study area is estimated by multiplying by the scalar M . The value of M (in units of mg·ha/L·kg) is a property of the hydrological

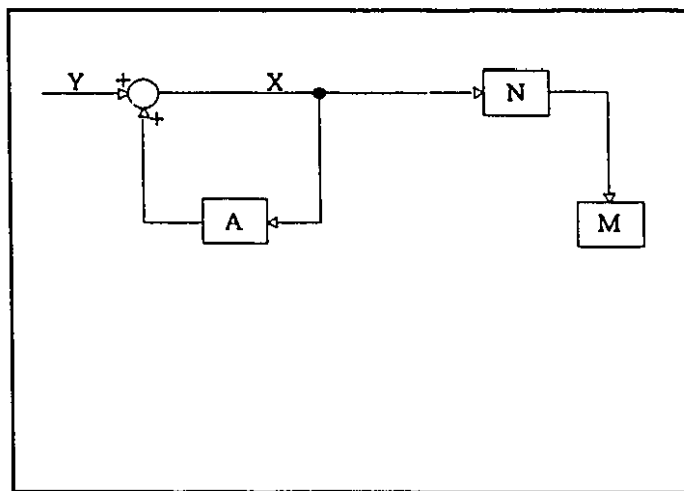


Figure 2. Amount of Pesticide.

and geological characteristics of the study region. Again, EPIC is capable of estimating this relationship given the various groundwater parameters, or the value of M could be estimated empirically from field observations. Figure 2 illustrates the block schematic diagram of the model with the N, and M structures in place.

The concentration of pesticide allowed in groundwater under water quality regulations is represented by the scalar S (in units of mg/L). The difference between the amount of pesticide present in the groundwater and the regulated allowable amount is the amount that must be removed, which is labelled R in the schematic. Figure 3 shows the block schematic with the calculation which determines R.

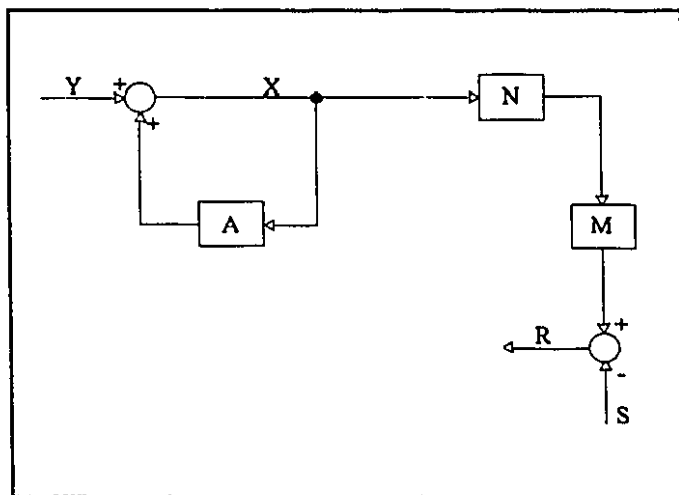


Figure 3. Amount of Pesticide to be Removed.

The production associated with the pesticide removal is a function of the amount of excess pesticide. K is an $n \times 1$ column vector, where each element, k_i , is the amount of production of that sector's product associated with the removal of 1 concentration unit of pesticide from all of the groundwater across the entire study region. The units of K are

(concentration unit/\$), such that multiplying K by R results in KR , an $n \times 1$ column vector of total remediation production in all sectors. This remediation production is then added back into the economy by adding the arrow to the circle in the upper left-hand side of the diagram. This method of including production associated with environmental remediation is similar to Chen (1973) and Xu, et al. (1994). Figure 4 shows the complete model block schematic.

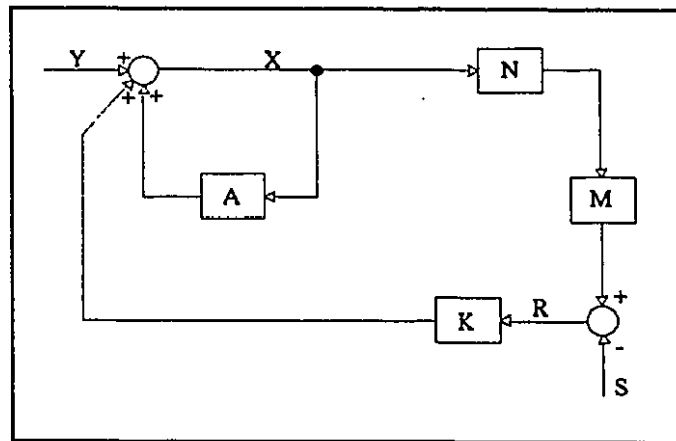


Figure 4. Complete Pesticide Impact Input-Output Model

The values of the elements of K can be estimated from observed economic production associated with groundwater cleanup efforts. Environmental consulting sectors which provide groundwater cleanup services would have the largest k_i values, however other sectors may also have non-zero values (such as the insurance or finance services sectors associated with environmental cleanup liability).

Also, note that this model structure is valid for evaluation of more than one groundwater contaminant. N (as well as M , S and K) can be modified to include information for a series of pesticides or fertilizers, or any other pollutant that results from

agricultural production. However, for the purposes of clarity of explanation, only one pesticide will be shown in the calculations in this paper.

Pesticide Impact Input-Output Model Equations

Equations for the model can be derived from the diagram as discussed earlier.

Starting with the circle on the upper left-hand side of the diagram, the following equation is derived:

$$X = AX + Y + KR \quad (12)$$

Deriving the equation from the lower right-hand corner of the diagram in Figure 4:

$$R = MNX - S \quad (13)$$

where N represents the lookup table, and all other variables are as shown earlier.

Substituting (13) into (12) gives:

$$X = AX + Y + KMNX - KS \quad (14)$$

which can be rearranged to give:

$$X = [I - A - KMN]^{-1}(Y - KS) \quad (15)$$

Equation (15) shows the model equation solved for the production, X. The total amount of remediation production (in dollars) can be calculated for each sector as the column vector KR.

A set of example calculations for this model, using assumed and real data from Southern Ontario, can be found in the Sample Calculations appendix below.

Interpretation of Model Results

Each model has a number of assumptions inherent in its development that affect the way the model behaves. Therefore, these assumptions need to be considered when interpreting the results. This section outlines issues relating to Pesticide Impact Input-Output Model results.

The Pesticide Impact Input-Output model incorporates back into the economy the production associated with removing pesticide contamination from groundwater. The additional sectoral production associated with groundwater remediation requires the additional input of products from other sectors, which in turn require additional inputs from other sectors, and so on. The additional clean-up production is incorporated not only into the sectors directly related to groundwater remediation, but also those sectors that provide inputs to remediation as well. Therefore, both the direct and indirect remediation production is estimated.

The interpretation of the model results should consider the cost of the cleanup as not being borne by any specific sector of the economy. Rather, the cleanup revenue can be considered to be the result of “exogenous cleanup demand” by an institution outside the market economy, such as government. The incorporation of the cost of cleanup into the “input recipe” for each sector’s product (to reflect remediation costs borne by the pollution-generating sector) is an extension of this model that could be considered for future research.

Conclusions

The Pesticide Impact Input-Output Model can be used to investigate the relationship between economic production in various sectors of the economy and agricultural pesticide regulation. The additional sectoral production associated with groundwater remediation is incorporated into the economy, to reflect the removal of pesticide concentrations in excess of regulated limits. The use of environmental models or empirical observation to relate the amount of pesticide contamination to the amount of production allows the model to reflect the local environmental characteristics of the study region.

Appendix: Sample Calculations

For the purposes of illustrating the use of the Pesticide Impact Input-Output Model, a study area in southwestern Ontario is selected. The area is illustrated in Figure 5, and encompasses parts of Wellington, Perth, Oxford, Elgin, Middlesex and Huron Counties, Waterloo Regional Municipality, and the City of London. The study region encompasses areas mostly used for agricultural production.

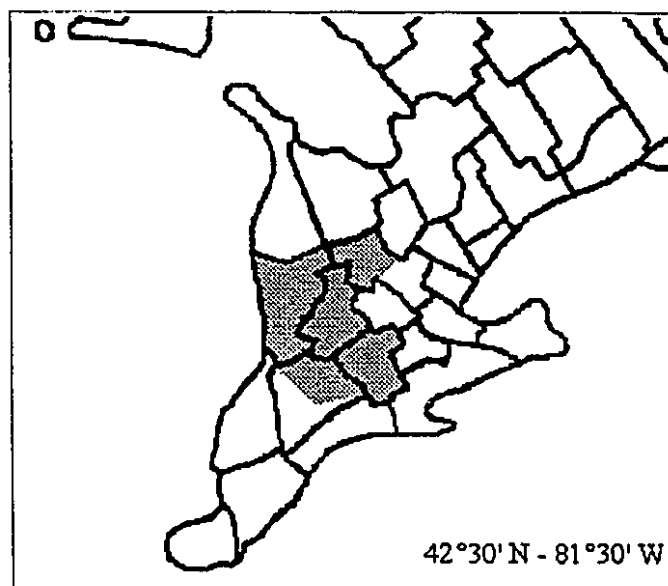


Figure 5. Southern Ontario Study Area

Table 2 shows the technical coefficients (A) table for the study area, which is determined from a StatsCanada Ontario Technical Coefficients Table from 1986, scaled down to the study area using location quotients (LQ) generated from employment data (see Miller and Blair (1985) for this method).

Table 2: Technical Coefficients Table (A).

Sector	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.218	0.000	0.003	0.001	0.032	0.003	0.000	0.000	0.001	0.008	0.000	0.002	0.017
2	0.000	0.008	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.001
3	0.000	0.000	0.119	0.000	0.006	0.000	0.000	0.000	0.001	0.000	0.000	0.000	0.000
4	0.002	0.004	0.001	0.033	0.066	0.008	0.006	0.082	0.001	0.003	0.002	0.001	0.004
5	0.239	0.239	0.066	0.098	0.417	0.337	0.097	0.029	0.073	0.048	0.009	0.234	0.224
6	0.011	0.000	0.014	0.016	0.004	0.001	0.021	0.021	0.002	0.006	0.041	0.002	0.005
7	0.014	0.011	0.079	0.026	0.022	0.020	0.395	0.014	0.029	0.012	0.002	0.040	0.011
8	0.016	0.000	0.005	0.035	0.018	0.004	0.020	0.025	0.042	0.045	0.033	0.038	0.033
9	0.020	0.013	0.009	0.018	0.018	0.044	0.009	0.003	0.013	0.003	0.001	0.029	0.018
10	0.005	0.006	0.005	0.002	0.002	0.009	0.004	0.003	0.002	0.001	0.001	0.020	0.008
11	0.028	0.002	0.057	0.041	0.015	0.015	0.018	0.021	0.066	0.087	0.065	0.031	0.037
12	0.021	0.001	0.118	0.074	0.060	0.066	0.048	0.043	0.112	0.081	0.071	0.110	0.041
13	0.002	0.001	0.013	0.004	0.003	0.006	0.009	0.006	0.004	0.002	0.003	0.018	0.005

Sectors 1 to 13 in the above table are described below:

<u>Sector #</u>	<u>Description</u>	<u>Sector #</u>	<u>Description</u>
1	Agriculture and Related Service Industries	9	Wholesale Trade Industries
2	Fishing and Trapping Industries	10	Retail Trade Industries
3	Logging and Forestry Industries	11	Finance, Insurance and Real Estate Industries
4	Mining, Quarrying and Oil Well Industries	12	Community, Business and Personal Services
5	Manufacturing Industries	13	Accommodation, Food and Beverage Industries
6	Construction Industries		
7	Transportation and Storage Industries		
8	Communications and Utilities Industries		

Note that the agricultural sector in this economic table is defined as sector #1. The following crop yield lookup table was determined by using EPIC and an assumption regarding the shape of the hypothetical crop yield relationship, which is also shown in Figure 6.

<u>Crop Yield (1,000's of \$)</u>	<u>Pesticide Application (kg/ha)</u>
10,000	0.0
17,000	1.0
22,000	2.0
26,000	3.0
30,000	4.0
34,500	5.0
37,000	6.0
38,000	7.0
39,000	8.0
39,500	9.0
40,000	10.0

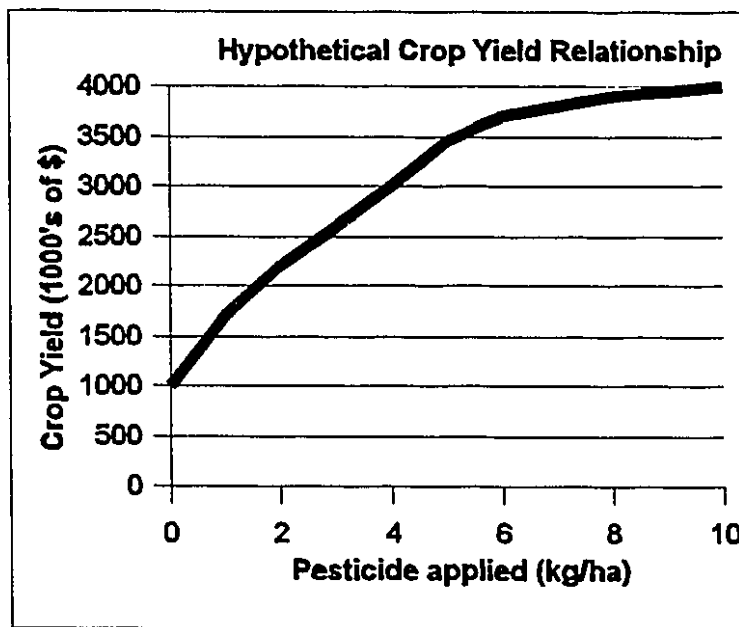


Figure 6. Hypothetical Crop Yield Relationship.

M relates the amount of pesticide applied to the amount of pesticide found in the groundwater, and is assumed to be equal to 0.001 mg/L per kg/ha, or converted to 0.001 mg·ha/L·kg (see eq. 11).

S (the allowable limit of pesticide) is set at 0.005 mg/L (Ontario Drinking Water

Objective for the pesticide Atrazine, Ontario Ministry of Environment and Energy, 1994), and the K (remediation production, units of thousands of dollars) vector is assumed as shown below. The K vector represents the amount of production from each of the 13 sectors required to remove one 0.001 mg/L of pesticide from the groundwater of all the farms in the study region. Note that the business services sector (#12) is the sector which contains the primary remediation industry. The assumed final demand Y for each sector (in units of thousands of dollars) is also shown below:

$$K = \begin{bmatrix} 100 \\ 0 \\ 0 \\ 100 \\ 500 \\ 250 \\ 100 \\ 0 \\ 100 \\ 0 \\ 100 \\ 5,000 \\ 500 \end{bmatrix} \qquad Y = \begin{bmatrix} 25,000 \\ 800 \\ 1,500 \\ 2,500 \\ 2,000 \\ 10,000 \\ 3,000 \\ 15,000 \\ 20,000 \\ 2,000 \\ 25,000 \\ 30,000 \\ 18,000 \end{bmatrix}$$

Using the information above, eq. (15) can be solved for the level of sectoral production, X, and the dollar value of the remediation, KR. X and KR are shown below (in units of thousands of dollars):

$$\begin{array}{l}
 \begin{array}{c}
 \left[\begin{array}{c}
 35,720 \\
 850 \\
 2,220 \\
 9,900 \\
 68,020 \\
 13,700 \\
 15,860 \\
 24,170 \\
 25,460 \\
 21,990 \\
 37,070 \\
 55,120 \\
 20,290
 \end{array} \right] \\
 X =
 \end{array}
 \end{array}
 \quad
 \begin{array}{l}
 \begin{array}{c}
 \left[\begin{array}{c}
 49 \\
 0 \\
 0 \\
 49 \\
 245 \\
 123 \\
 49 \\
 0 \\
 49 \\
 0 \\
 49 \\
 2450 \\
 245
 \end{array} \right] \\
 KR =
 \end{array}
 \end{array}$$

In order to illustrate the impact of including groundwater remediation cost into the economy, the production that is calculated from eq. (4) that does not include remediation production ($X_{\text{NO REMED}}$) is compared to the above production with remediation (X_{REMED}).

Below, the difference between the two production vectors is shown:

$$\begin{array}{l}
 \begin{array}{c}
 \left[\begin{array}{c}
 35,560 \\
 840 \\
 22,00 \\
 97,00 \\
 65,950 \\
 13,530 \\
 15,460 \\
 23,970 \\
 25,260 \\
 21,920 \\
 36,820 \\
 52,090 \\
 19,980
 \end{array} \right] \\
 X_{\text{NO REMED}} =
 \end{array}
 \quad
 \begin{array}{c}
 \left[\begin{array}{c}
 35,720 \\
 850 \\
 2,220 \\
 9,900 \\
 68,020 \\
 13,700 \\
 15,860 \\
 24,170 \\
 25,460 \\
 21,990 \\
 37,070 \\
 55,120 \\
 20,290
 \end{array} \right] \\
 X_{\text{REMED}} =
 \end{array}
 \quad
 \begin{array}{c}
 \left[\begin{array}{c}
 0.46 \\
 0.12 \\
 0.69 \\
 2.25 \\
 3.14 \\
 1.23 \\
 2.59 \\
 0.86 \\
 0.80 \\
 0.33 \\
 0.68 \\
 5.81 \\
 1.58
 \end{array} \right] \\
 \% \text{ Increase} =
 \end{array}
 \end{array}$$

Note that with the inclusion of remediation production, the business services sector (#12) showed the largest increase at 5.81%, and the fishing/trapping (#2) sector showed the smallest increase at 0.12%.

References

- Anaman, K.A. (1994). Input-output analysis of the secondary impact of a screwworm fly invasion of Australia on the economy of Queensland. *Preventative Veterinary Medicine*, 21, 1-18.
- Carson, R. (1962). "Silent Spring", Cambridge: Riverside Press.
- Chen, K. (1973). Input-Output Economic Analysis of Environmental Impact, *IEEE Transactions on Systems, Man, and Cybernetics SMC-3*, 539-547.
- Gay, P.W. and Proops, L.R. (1993). Carbon-dioxide Production by the UK Economy: An Input-Output Assessment, *Applied Energy* 44, 113-130.
- Gladwell, J.S. (1989). Groundwater Quality Management: A Multi-Dimensional Problem. In *Proceedings of the International Conference on Groundwater Contamination: Use of Models in Decision Making*, (G. Jousma, ed.), Boston: Kluwer Academic Publishers.
- Heslin, J.S. and Hobbs, B.F. (1990a). Economic Analysis of Oil and Gas Brine Regulations, *Journal of Energy Engineering*, 116, No. 1. 51-70.
- Heslin, J.S. and Hobbs, B.F. (1990b). Application of a Multiobjective Electric Power Production Costing Model to the U.S. Acid Rain Problem, *Engineering Costs and Production Economics* 20, 241-251.
- Johnson, S.L., Adams, R.M., and Perry, G.M. (1991). The On-Farm Costs of Reducing Groundwater Pollution, *American Journal of Agricultural Economics*, 73, 1063-1073.
- Leontief, W. (1970). Environmental Repercussions and the Economic Structure: An Input-Output Approach, *The Review of Economics and Statistics* 57, 262-271.
- Loneragan, S.C. and Cocklin, C. (1985). The Use of Input-Output Analysis in

- Environmental Planning, *Journal of Environmental Management* 20, 129-147.
- Madhun, Y.A. and Freed, V.H. (1990). Impact of Pesticides on the Environment. In *Pesticides in the Soil Environment: Processes, Impacts and Modeling*, S.S.S.A. Book Series No.2 (H.H. Cheng, ed.), pp. 429-458. Wisconsin: Soil Science Society of America.
- Miller, R.E. and Blair, P.D. (1985). *Input-Output Analysis: Foundations and Extensions*, New Jersey: Prentice-Hall.
- Ontario Ministry of Environment and Energy, (1994). *Ontario Drinking Water Objectives*, p. 32. Toronto: Queen's Printer for Ontario.
- Palmini, D.J. (1982). The Secondary Impact of Nonpoint-Pollution Controls: A Linear-Programming-Input/Output Analysis, *Journal of Environmental Economics and Management* 9, 263-278.
- Sharpley, A.N. and Williams, J.R. (1990). EPIC -- Erosion/Productivity Impact Calculator: 1. Model Documentation, U.S. Department of Agriculture Technical Bulletin No. 1768, Agricultural Research Service, Temple, Texas.
- Victor, P. A. (1972). "Pollution: Economy and Environment", London: Allen and Unwin.
- Xu, P., Tsanis, I.K., Anderson, W.P. and Kanaroglou, P. (1994). An Economic Input-Output Analysis for Urban Stormwater Quality Planning, *Water Resources Management* 8, 155-170.

Appendix III: Input-Output Modelling and Groundwater Remediation

Abstract

Use of models in a decision-making context requires the interpretation of results with respect to model uncertainty. Monte Carlo methods have been used to assess model uncertainty with standard Leontief input-output models. Input-output models have also been extended for use in environmental management by incorporating pollution cleanup costs back into the economy. The effect on uncertainty of adding environmental feedback extensions to standard input-output models is investigated. Monte Carlo analysis of a hypothetical model indicates that uncertainty can be increased by as much as 80% by adding a feedback effect. Results also indicate that while the amount of uncertainty is constant for different levels of demand for models without feedback, models with feedback have significantly increased uncertainty at lower levels of demand. Specifically, uncertainty is greatly increased when the total amount of remediation production is equal to or less than the total amount of exogenous demand.

Key Words: input-output models, uncertainty, Monte Carlo, feedback, environment.

Introduction

In the 1930's, Wassily Leontief proposed the input-output model, which uses information about the amount of goods and services exchanged between different economic sectors to determine an economy's behaviour under different product demand levels. Simple, linear relationships describe the relative amount of any sectors's product being used in the creation of any other sector's product. These relationships can then be used to determine the overall production in each economic sector, based on a given level of exogenous final demand. Leontief

input-output models have been extended for use in many areas, including employment, natural resource and environmental analyses (Miller and Blair, 1985).

The use of any model (economic or otherwise) requires the consideration of uncertainty when interpreting model results. The predictive ability of the model is dependent on several factors, including the structure of the model and the quality of the input data (Beck, 1987). The basic Leontief input-output model has a standard structure, the uncertainty of which has been analyzed in the past by various authors (Bullard and Sebald, 1977; West, 1986; Bullard and Sebald, 1988). However, extensions of the basic input-output model have been proposed which change the way uncertainty propagates through the model. One such extension is the inclusion of a feedback effect, where a “cost” related to the level of production in one or more sectors is then added back into the economy. Making this type of structural change to an input-output model can alter the amount of uncertainty.

The purpose of this paper is to investigate how changing the structure of an input-output model (specifically through the addition of a feedback effect) changes the uncertainty of its output. How does adding an environmental extension affect the way this type of model should be used?

The concepts of uncertainty analysis developed in this paper will be illustrated with a simple numerical example, but the methods and conclusions can be applied universally to any input-output model with feedback effects.

Uncertainty in Input-Output Modelling

When evaluating the output from input-output models, determining a measure of

uncertainty in the results provides a framework for interpretation. Uncertainty can be defined in many different ways, but for the purposes of this paper, it is defined as a measurement of the variability in the output. It is a measure of how well the output of the model is “known”. In the studies included in this project, uncertainty is measured by the standard deviation associated with the output of the model.

Various kinds of uncertainty analyses can be used to determine the amount of change in model output that comes from a change in model input. Bullard and Sebald (1977) used matrix analysis to define error bounds on input-output calculations. The authors note that parameters of input-output models are subject to two major types of uncertainty: statistical errors in compiling empirical data, and the fact that input-output coefficients do not remain constant over time. Bullard and Sebald quantified the error bounds by using matrix norm analysis and creating perturbed technical coefficients matrices.

A more complex method of calculating confidence intervals for input-output multipliers was proposed by West (1986). Assumed independent, normal distributions of input coefficients were used to analytically calculate distributions of output multipliers in an input-output model of the Queensland, Australia economy. West concluded that the results of the analysis are dependent on the assumptions made regarding the distributions of input coefficients, but that the method can be employed to analyze the uncertainty of such models.

Bullard and Sebald (1988) updated their earlier work by using the Monte Carlo method to better define the error bounds on input-output analysis. They concluded that the Monte Carlo method provides a tighter and more reasonable error estimate than the analytical methods proposed earlier. Analysis of the 1967 U.S. input-output tables showed that analytical error

tolerances are overestimated due to the fact that input data uncertainties combine or cancel one another in a manner that reduces error magnification.

Applying the Monte Carlo Method to Input-Output Models

A comprehensive method to study the sensitivity of a model is the Monte Carlo method. It has been commonly used to study the behaviour of many different kinds of models (Beck, 1987, Brattin, et al, 1996), and employed with input-output models by Bullard and Seward (1988). The basis of the Monte Carlo method is the use of assumed distributions of input parameters. Many sets of values are independently sampled from the distributions, and then the model is solved with each of the input value sets. The multiple sets of output are then analyzed to determine the variability.

The Monte Carlo Method is applied to the following hypothetical input-output model.

The transactions matrix Z and exogenous demand vector Y are shown below:

$$Z = \begin{bmatrix} 3 & 7 & 1 & 2 \\ 4 & 9 & 3 & 1 \\ 14 & 3 & 11 & 2 \\ 1 & 1 & 1 & 2 \end{bmatrix} \quad Y = \begin{bmatrix} 10 \\ 13 \\ 5 \\ 2 \end{bmatrix}$$

In order to create a distribution of values for each element in the 4×4 Z matrix, 100 new Z matrices are generated by adding normally distributed noise to the existing Z element values. Normally distributed noise is used to represent a typical distribution of coefficients. West (1986) noted that real coefficient distributions are not well defined, but that Monte Carlo I/O analyses have shown to be insensitive to distribution type. In his analysis, normally distributed coefficients

were assumed.

In this study, the noise is created such that the standard deviation is equal to 30% of the maximum original Z matrix value. The Z matrix is then corrected to make sure all elements are greater than zero. For this example, Y is not altered.

For each new Z matrix, the corresponding technical coefficients matrix A is then calculated. 100 new vectors of X are then determined using the basic Leontief equation $X = (I - A)^{-1}Y$. The mean and standard deviation of the new X values can be determined for each sector's production.

$$X_{\text{MEAN}} = \begin{bmatrix} 25.9 \\ 31.6 \\ 36.7 \\ 11.7 \end{bmatrix} \quad E = \begin{bmatrix} 6.6 \\ 7.1 \\ 7.9 \\ 5.5 \end{bmatrix}$$

X_{MEAN} is the mean value of sectoral production, and E is the standard deviation of the X

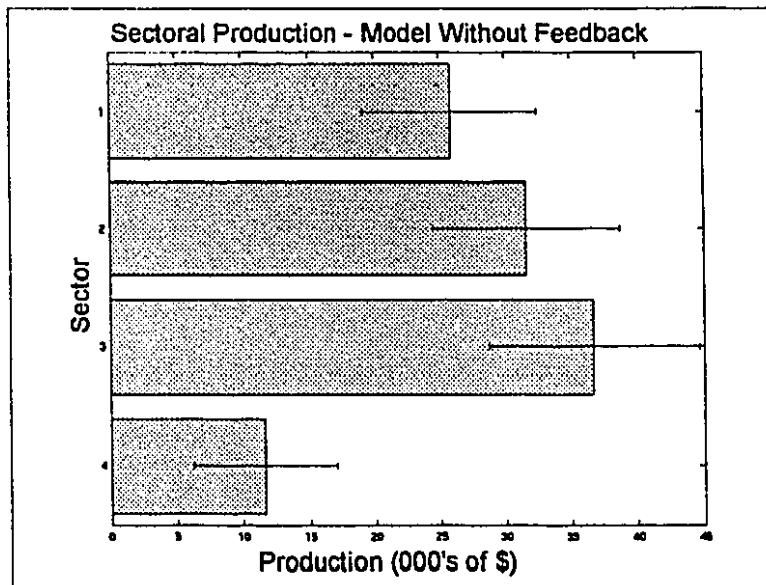


Figure 1. Monte Carlo Analysis - I/O Model Without Feedback.

element values across all 100 X vectors calculated.

Figure 1 illustrates the results of the Monte Carlo test, where the bars represent the mean value of sectoral production, and the error bars represent +/- 1 standard deviation of production values. The uncertainty of the input-output model output is represented by the size of the error bars. Note that all error bars are of similar magnitude, and that the magnitude of uncertainty does not necessarily correspond to the value of production.

Environmental Extensions of Input-Output Models

Environmental extensions of input-output modelling have been described by Miller and Blair (1985). They include the use of environmental coefficients that relate items such as resource use and demand, and also the use of "ecological commodities" which describe the flow of products to and from the natural environment.

Environmental input-output models have been developed by several authors. Leontief (1970) and Chen (1973) developed simple extensions of the basic input-output model to relate economic production and environmental impact. Input-output analysis has been applied to various types of environmental-economic studies, including waste disposal regulations (Heslin and Hobbes, 1990), pesticide regulations (Palmini, 1982), and insect vector invasions (Anaman, 1994).

The structure of the extended environmental-economic model allows the user to simulate economic production at various demand levels, while at the same time evaluating economic impact. The structure of the model also influences the uncertainty associated with the results. In order to be able to properly evaluate the output from these kinds of models, the uncertainty of

input-output models with environmental extensions must be investigated.

Internal Feedback Effects

In a standard input-output model, exogenous final demand is used to determine sectoral output through information on the transactions of money and goods between economic sectors. The basis for the Leontief input-output model is an assumption that the ratio of goods (or services) from any one sector used in the production of goods (or services) in another sector is constant, regardless of the level of that sector's production. In other words, there is a linear relationship between the amounts of input goods used and the amount of output goods produced. If a sector's production is doubled, then the amount of input goods required will also double. This linear relationship assumes that the technology and input "recipe" for a good is constant over time and over all levels of output, implying that economies of scale are ignored.

Extensions of the basic input-output model can be used to determine the behaviour of factors related to an economy, such as employment, input resource use, and pollution generation. Linear assumptions are made to relate each of these elements to production. For example, the vector V describes the amount of pollution generated per dollar of sectoral output, where:

$$v_{kj} = \begin{array}{l} \text{amount of pollutant of type } k, \text{ generated per} \\ \text{dollar worth of output of sector } j \end{array}$$

V is an $m \times n$ matrix, where m is the number of pollutants under consideration, and n is the number of sectors in the economy. Therefore, the vector of total pollutant levels V^* , would be:

$$V^* = VX \tag{eq.(1)}$$

The elements of the total pollutants vector, v^*_k , represent the total amount of pollutant k

generated by all sectors. The pollutant levels can be directly calculated as a function of demand through the use of the Leontief equation (Miller and Blair, 1985):

$$V^* = V[(I - A)^{-1}]Y \quad \text{eq.(2)}$$

Leontief (1970) proposed comparing the amount of pollution generated by various sectors to some accepted standard. If the amount of pollution is greater than the standard, then there is an amount of pollution that needs to be removed, treated or otherwise remediated. The production associated with the removal of the pollution can then be related back into the economy. This approach has been expanded by Chen (1973) and Xu, et al. (1994).

When a vector of pollutant levels is calculated and compared to a regulated “accepted” level, S , the difference R (where $R = V^* - S$) is the amount of pollutant that has to be removed (for this example, it is assumed only one pollutant is generated). The production associated with this removal can be described by K , an $n \times 1$ column vector where each element, k_i , is the amount of sector i 's product associated with the removal of 1 physical unit of pollutant. The total remediation production, KR , can then be added to the economy, creating a feedback effect.

Given that the production of an economy is defined by the equation:

$$X = AX + Y \quad \text{eq.(3)}$$

the amount of additional production from pollution remediation, KR , can be added to form the equation:

$$X = AX + Y + KR, \quad \text{eq.(4)}$$

and the amount of pollution to be removed, R , is:

$$R = VX - S. \quad \text{eq.(5)}$$

These two equations can be combined and rearranged to explicitly express production X as a function of several elements, including demand Y , pollution generation V , cleanup cost K , and allowable level of pollutant, S :

$$X = (I - A - KV)^{-1}(Y - KS) \quad \text{eq.(6)}$$

This extension causes a repeating, iterative effect. First the pollution generated from economic production is calculated, and excess pollution (above the acceptable level) is remediated. The amount of economic output generated from the remediation is added back into the economy, causing further production, and therefore further pollution. This new pollution is then remediated, causing more production and pollution, etc., and the cycle is repeated. In each iteration of the cycle, the amount of new production added is smaller, until the solution to the model converges.

Monte Carlo Analysis

To analyze the uncertainty associated with the output of a model with an environmental feedback effect such as the one described above, a Monte Carlo analysis can be used. Figure 2 shows the results of a Monte Carlo analysis performed on the hypothetical input-output model used earlier, both with and without feedback. The feedback effect uses the pollution vector V , and acceptable level of pollutant S , as shown below:

$$V = \begin{bmatrix} 2 & 2 & 3 & 1 \end{bmatrix} \quad S = [0]$$

Note that the inclusion of a feedback element not only increases the production levels, but also increases the variability of the results. The uncertainty of the two models can be compared

by looking at the size of the standard deviation as a percentage of the mean. The average percentage uncertainty (across the 4 sectors) for the input-output model without feedback is 26.4%, compared to 47.9% for the model with feedback.

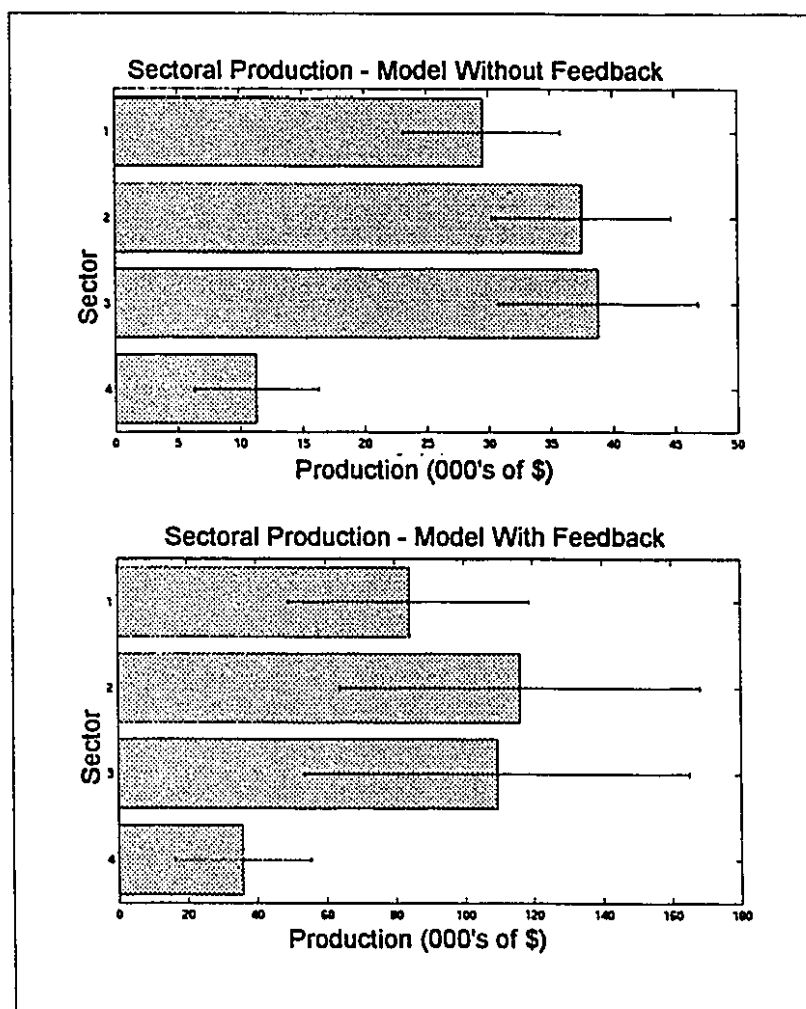


Figure 2. Comparing Uncertainty of Models With and Without Feedback.

The inclusion of a feedback effect creates an iterative loop where the uncertainty associated with sectoral production is then incorporated back into the economy through the inclusion of remediation production. Each new step brings more uncertainty. However, due to

the fact that the production (and uncertainty) added in each new iteration is smaller than the one in the previous step, the system converges. The repeated iteration of adding production back into the economy introduces more uncertainty each step.

Another way to analyze the difference between models with and without feedback effects is to compare the uncertainty associated with sectoral output over several different ranges of demand. Figure 3 shows the output and associated uncertainty (error bars equal \pm one standard

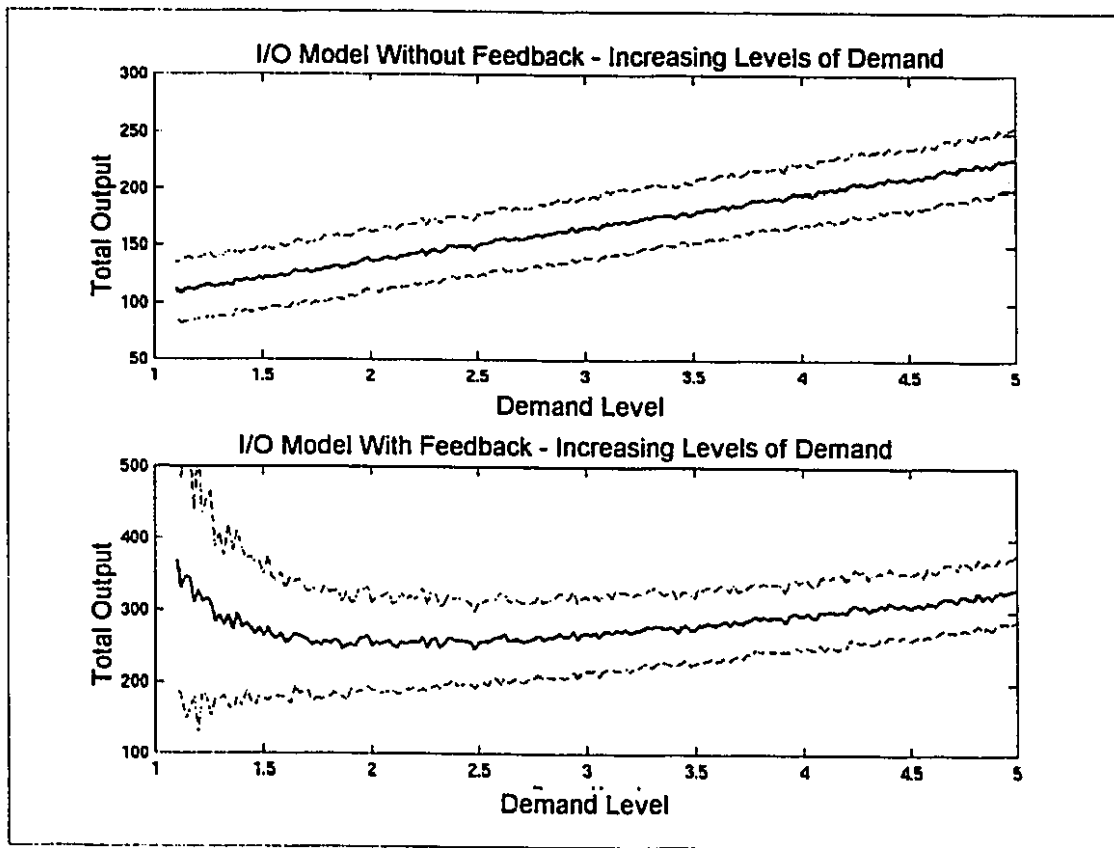


Figure 3. Comparing Models With and Without Feedback - Increasing Levels of Demand.

deviation around the mean) at several different levels of demand. The dark line represents the total production, and the thin dashed lines represent the uncertainty "envelope" associated with

NOTE TO USERS

Page (s) not included in the original manuscript is unavailable from the author or university. The manuscript was microfilmed as received.

136

This reproduction is the best copy available.

UMI[®]

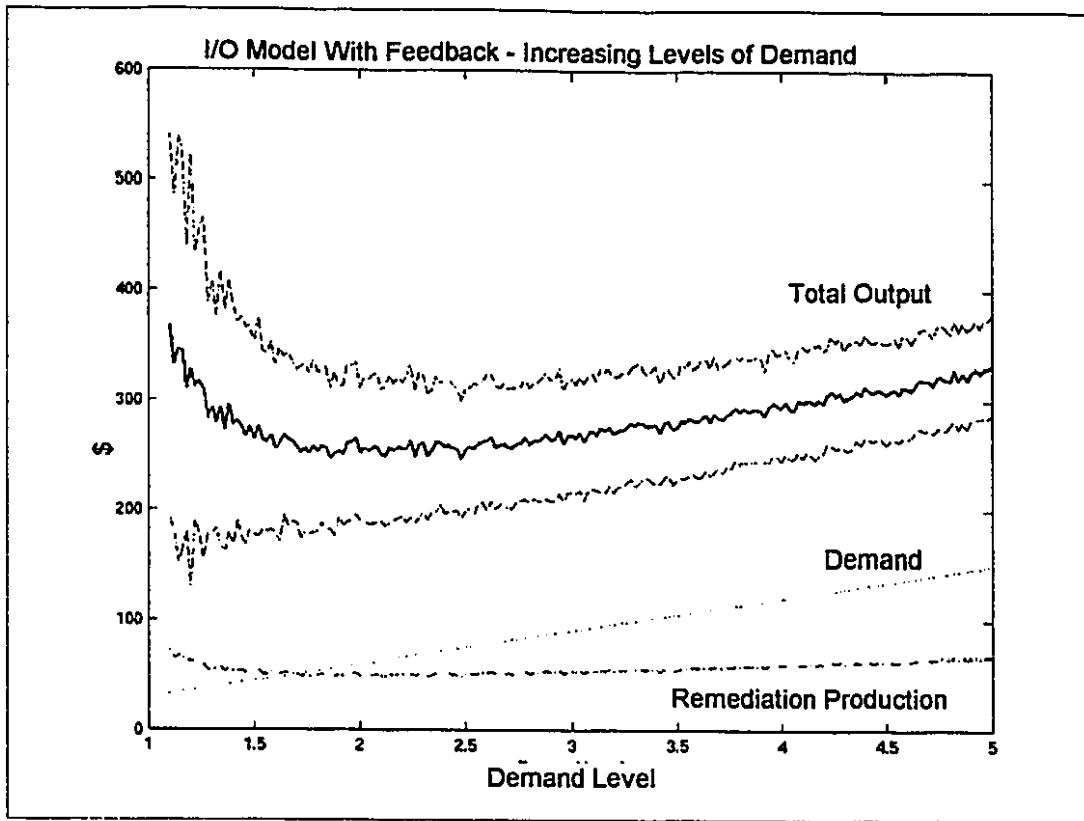


Figure 4. I/O Model with Feedback - Uncertainty at Increasing Levels of Demand.

produces more pollution, which is then remediated, which creates more pollution, etc. If the amount of new production created in each step is small with respect to the overall demand, the iterations will converge to a solution for the model. However, when the new production created each step is not significantly smaller each step, the solution does not converge quickly, and the uncertainty can be large. When the new production is larger in each step, the model cannot be solved, either iteratively, or through the use of eq.6. Therefore, the relative magnitude of the remediation production is important to defining the uncertainty in a model with feedback.

Conclusion

Input-output modelling is a commonly used tool for the evaluation of economic-environmental relationships. However, for a tool such an input-output model to be useful to an decision-maker, the amount of uncertainty associated with the model's output must be evaluated.

Input-output models have a characteristic uncertainty based upon their structure. Extensions to this basic structure affect the amount of uncertainty. Results from Monte Carlo analyses show that adding feedback effects (in order to incorporate a remediation cost borne from pollution production) significantly increases uncertainty. In addition, the uncertainty encountered in models with feedback is not constant over different levels of demand, and is generally greater when demand is equal to or less than remediation production.

References

- Anaman, K.A., (1994) Input-Output Analysis of the Secondary Impact of a Screwworm Fly Invasion of Australia on the Economy of Queensland. *Preventative Veterinary Medicine*, **21**: 1-18.
- Beck, M.B., (1987) Water Quality Modelling: A Review of the Analysis of Uncertainty. *Water Resources Research*, **23**: 1393-1442.
- Brattin, W.J., Barry, T.M., and Chiu, N. (1996) Monte Carlo Modeling with Uncertain Probability Density Functions. *Human and Ecological Risk Assessment*, **4**: 820-840.
- Bullard, C.W., and Sebald, A.V., (1977) Effects of Parametric Uncertainty and Technological Change on Input-Output Models. *The Review of Economics and Statistics*, **59**: 75-81.
- Bullard, C.W., and Sebald, A.V., (1988) Monte Carlo Sensitivity Analysis of Input-Output Models, *The Review of Economics and Statistics*, **70**: 708-712.
- Chen, K., (1973) Input-Output Economic Analysis of Environmental Impact. *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-3**: 539-547.

- Heslin, J.S., and Hobbs, B.F., (1990) Economic Analysis of Oil and Gas Brine Regulations. *Journal of Energy Engineering*, **116**: 51-70.
- Leontief, W., (1970) Environmental Repercussions and the Economic Structure: An Input-Output Approach. *The Review of Economics and Statistics*, **57**: 262-271.
- Miller, R.E., and Blair, P.D., (1985). *Input-Output Analysis: Foundations and Extensions*, New Jersey: Prentice-Hall.
- Palmini, D.J., (1982). The Secondary Impact of Nonpoint-Pollution Controls: A Linear-Programming-Input/Output Analysis. *Journal of Environmental Economics and Management*. **9**: 263-278.
- West, G.R., (1986). A Stochastic Analysis of an Input-Output Model. *Econometrica*, **54**: 363-374.
- Xu, P., Tsanis, I.K., Anderson, W.P., and Kanaraglou, P., (1994). An Economic Input-Output Analysis for Urban Stormwater Quality Planning. *Water Resources Management*, **8**: 155-170.

Appendix V: Models Used in Case Study I - Sorption of Metals onto Lake Sediments

Appendix V: Models Used in Case Study I - Sorption of Metals onto Lake Sediments

This appendix contains data regarding the three models used in the uncertainty/complexity study found in the paper “Evaluating Modelling Uncertainty for Model Selection” in Appendix II. For each model, a box-and-arrow diagram is shown, which schematically represents the flow of matter between various states in the model. Each state and flow is labelled, and the processes are briefly described. The calculation of the index of model complexity, as introduced in Appendix I, is explicitly shown for each model.

Computer code for each model is found following each box-and-arrow diagram page. This code was implemented with the modelling software GPS-X (General Purpose Simulator) by Hydromantis, Inc. Only the .usr file is shown, which contains all the relevant rate equations to describe all flows in the model.

The computer code is followed by a series of histograms, representing the parameter distributions for each parameter used in the sensitivity studies described in Appendix II. Following this, a graphical representation of the output “envelope” is shown. The thick, black line represents the mean value of the multiple output runs generated during the Monte Carlo analysis, and the thin lines represent \pm one standard deviation at each timestep. Information is presented for the following three models:

EQUILRAD - Equilibrium sorption algorithm

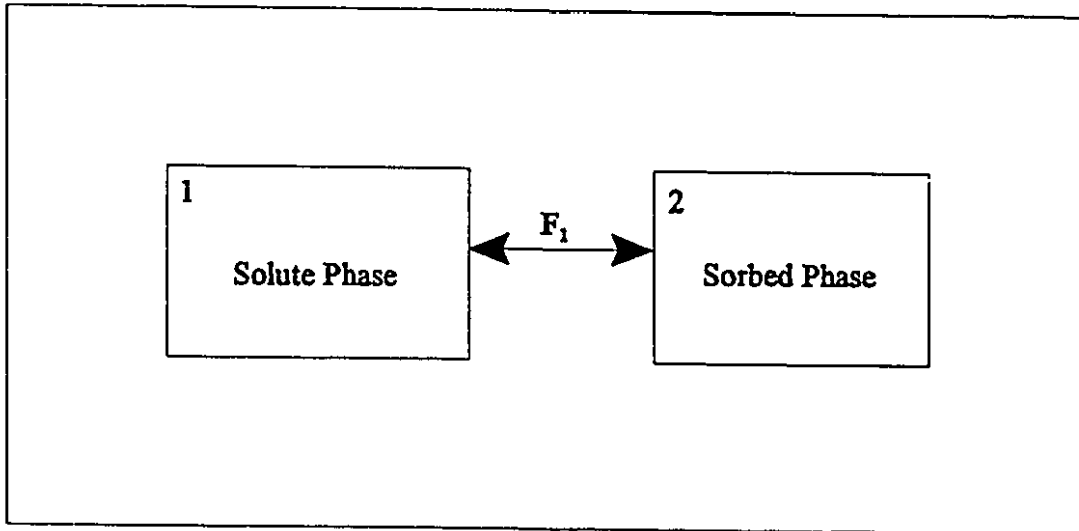
KINRAD - Kinetic sorption algorithm

3BOX - Both equilibrium and kinetic sorption algorithms

Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER: 1

Model Name: EQUILRAD



Flow Equations for State Variable 1:

F_1 = equilibrium sorption
= $f(kd)$

Number of parameters = 1

Number of operations = 3

Flow Equations for State Variable 2:

F_1 = equilibrium sorption
= $f(kd)$

Number of parameters = 1

Number of operations = 3

COMPLEXITY: $I_{kc} = (1*3) + (1*3) = 6$

FIT: = 0.7617

EQUILRAD.USR

```

*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

:macro rvardata
!   procedural(dliquid=kd)
!   dimension kd(300)
!   open (unit=1, file='equilrad.sam', status='old')
!   read (1,*) (kd(j), j=1,300)
!   close(unit=1)
!   print *,'parameter data entered'
!macro end

*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

!rvardata

macro end
*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

dliquid = (sorbed - liquid*kd)*50
dsorbed = -dliquid

      quid = integ(dliquid,initliquid)
sorbed = integ(dsorbed,initisorbed)

liquid2 = liquid

macro end
*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretes plus code to be executed every communication interval)
!Don't put macro definitions here

macro end
*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

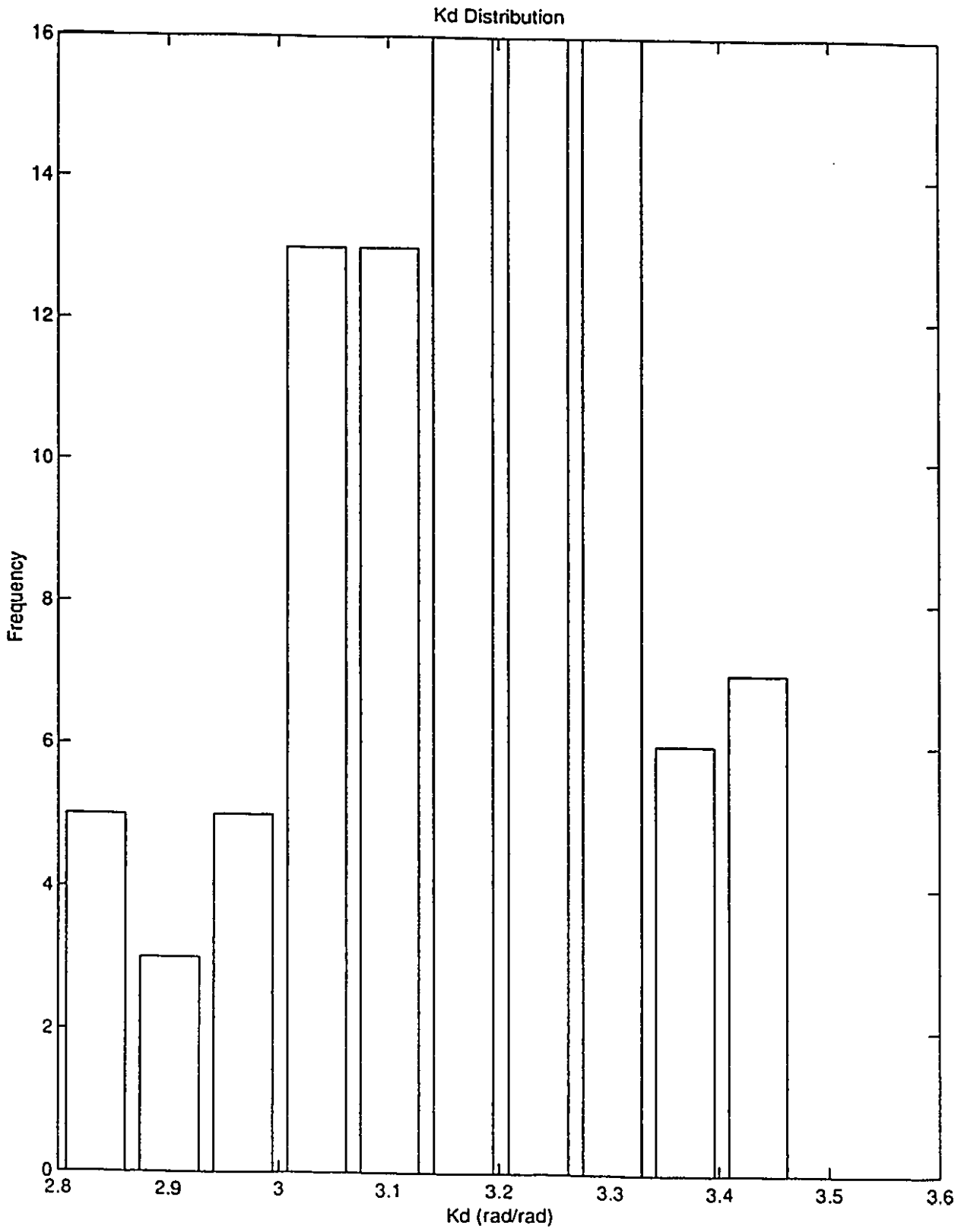
!open (unit=1,file='sens.dat',status='old')
!write (1,*) kd
!close (unit=1)
!write (*,*) 'sens.dat written.'

!write (*,*) 'kd = ',kd(rnum), '      rnum = ',rnum

      ro end
*****

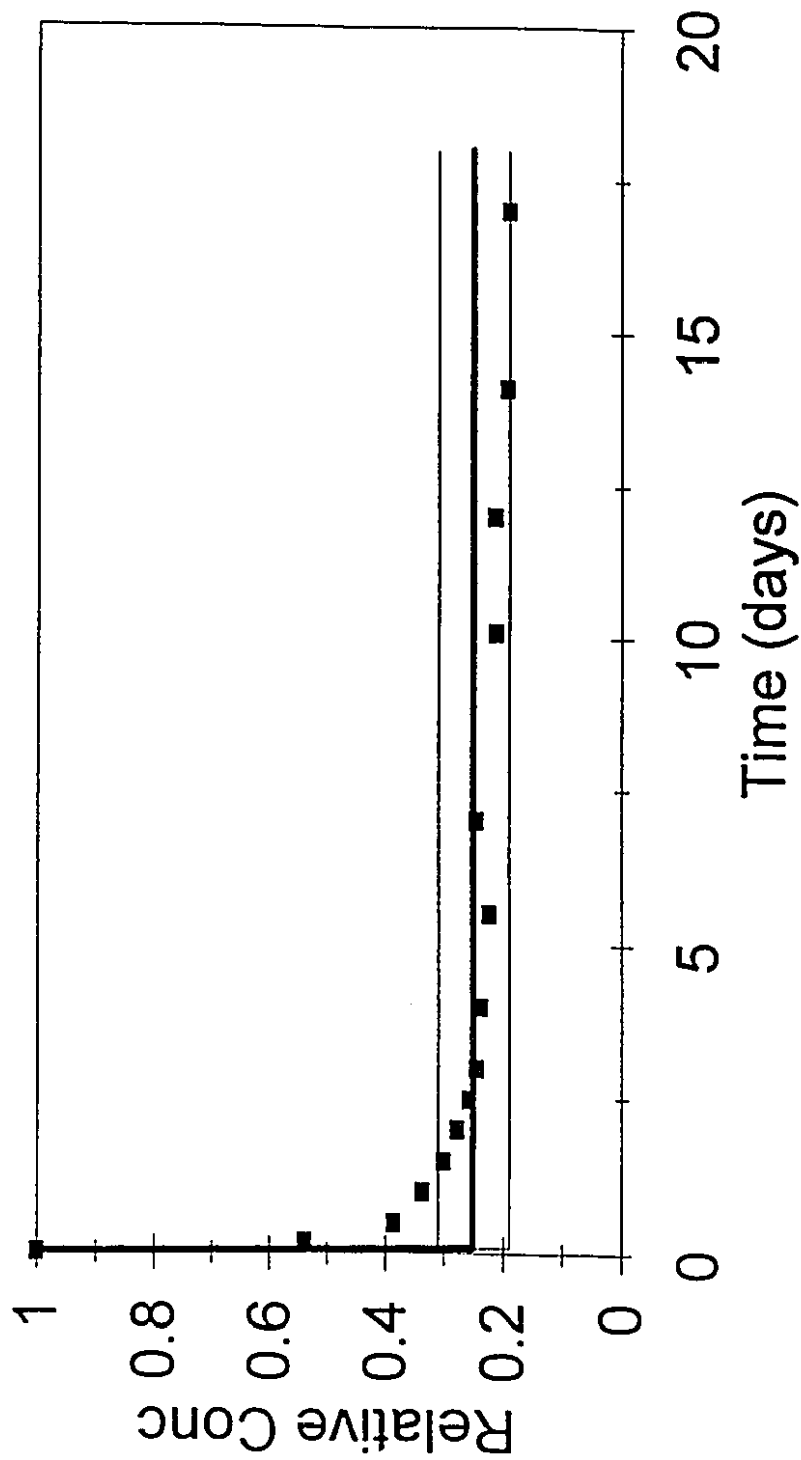
```

EQUILZAD



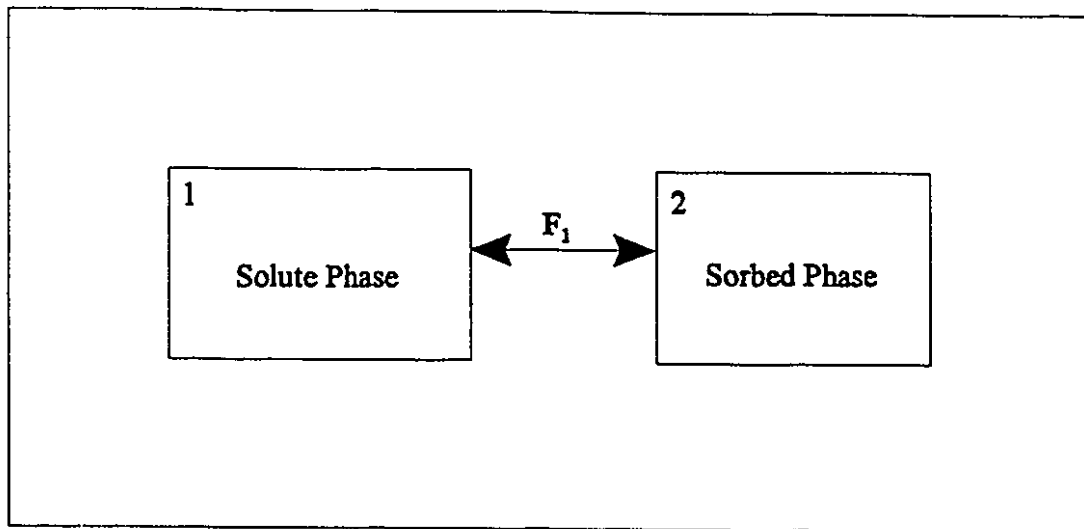
Relative Conc. vs. Time

Sensitivity Test - Equilrad



MODEL NUMBER: 2

Model Name: KINRAD



Flow Equations for State Variable 1:

F_1 = kinetic sorption
= $f(kd, \text{sorbrate})$

Number of parameters = 2

Number of operations = 3

Flow Equations for State Variable 2:

F_1 = kinetic sorption
= $f(kd, \text{sorbrate})$

Number of parameters = 2

Number of operations = 3

COMPLEXITY: $I_{kc} = (2*3) + (2*3) = 12$

FIT: = 0.236

KINRAD.USR

```
!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

:macro rvardata
!   procedural(dliquid=kd)
!   dimension kd(300), j(300)
!   open (unit=1, file='kinrad.sam', status='old')
!   read (1,*) (kd(j), rateconst(j), j=1,300)
!   close(unit=1)
!   print *, 'parameter data entered'
!   end
!macro end

!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

dliquid = (sorbed - liquid*kd)*rateconst
dsorbed = -dliquid

      liquid = integ(dliquid,initliquid)
      rsorbed = integ(dsorbed,initrsorbed)

liquid2 = liquid

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretes plus code to be executed every communication interval)
!Don't put macro definitions here

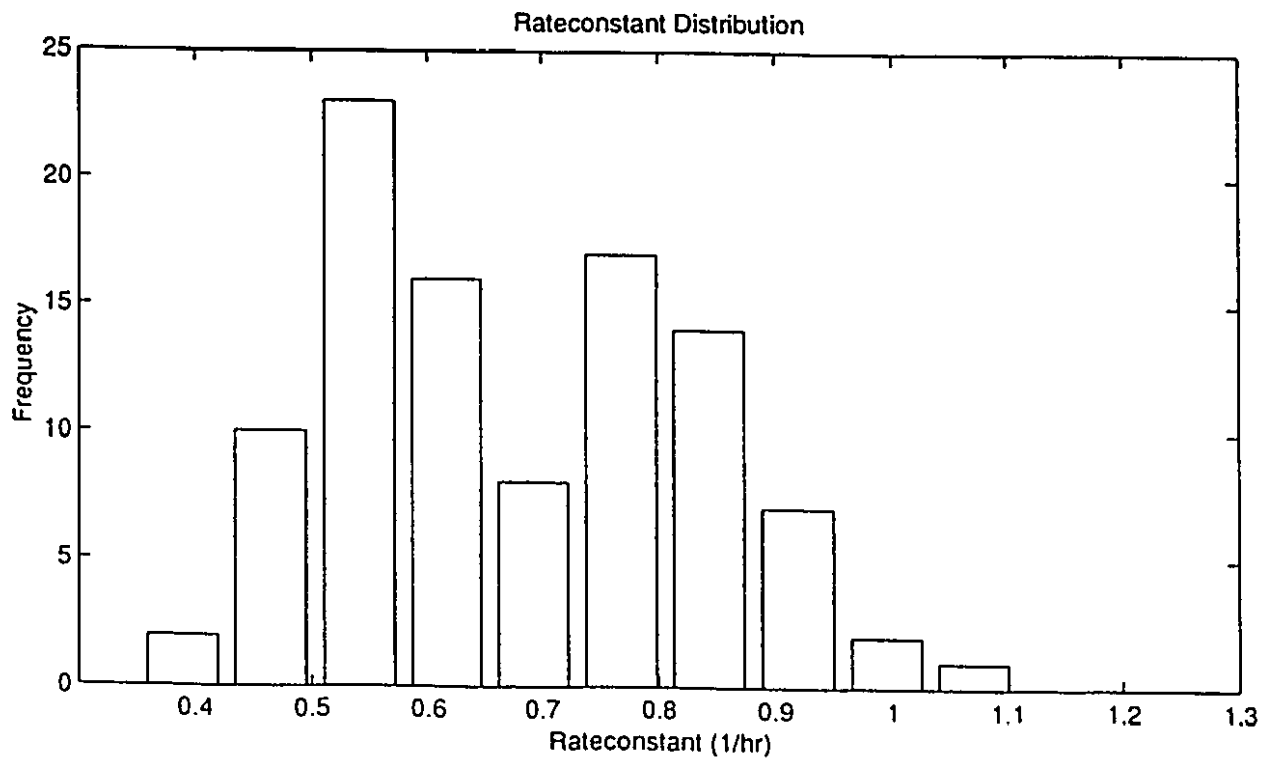
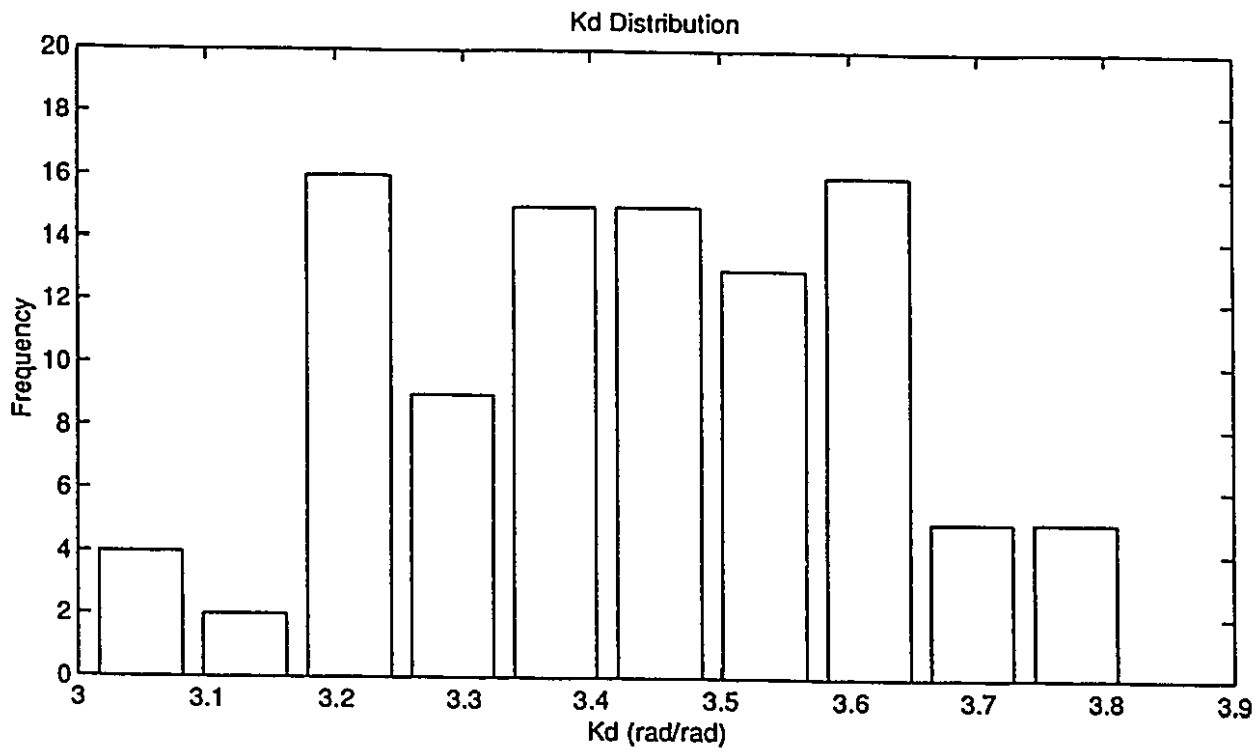
macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

!open (unit=1, file='sens.dat', status='old')
!write (1,*) kd,rateconst
!close (unit=1)
!write (*,*) 'sens.dat written.'

!print *, 'kd = ', kd(rnum), '   rateconst = ', rateconst(rnum)

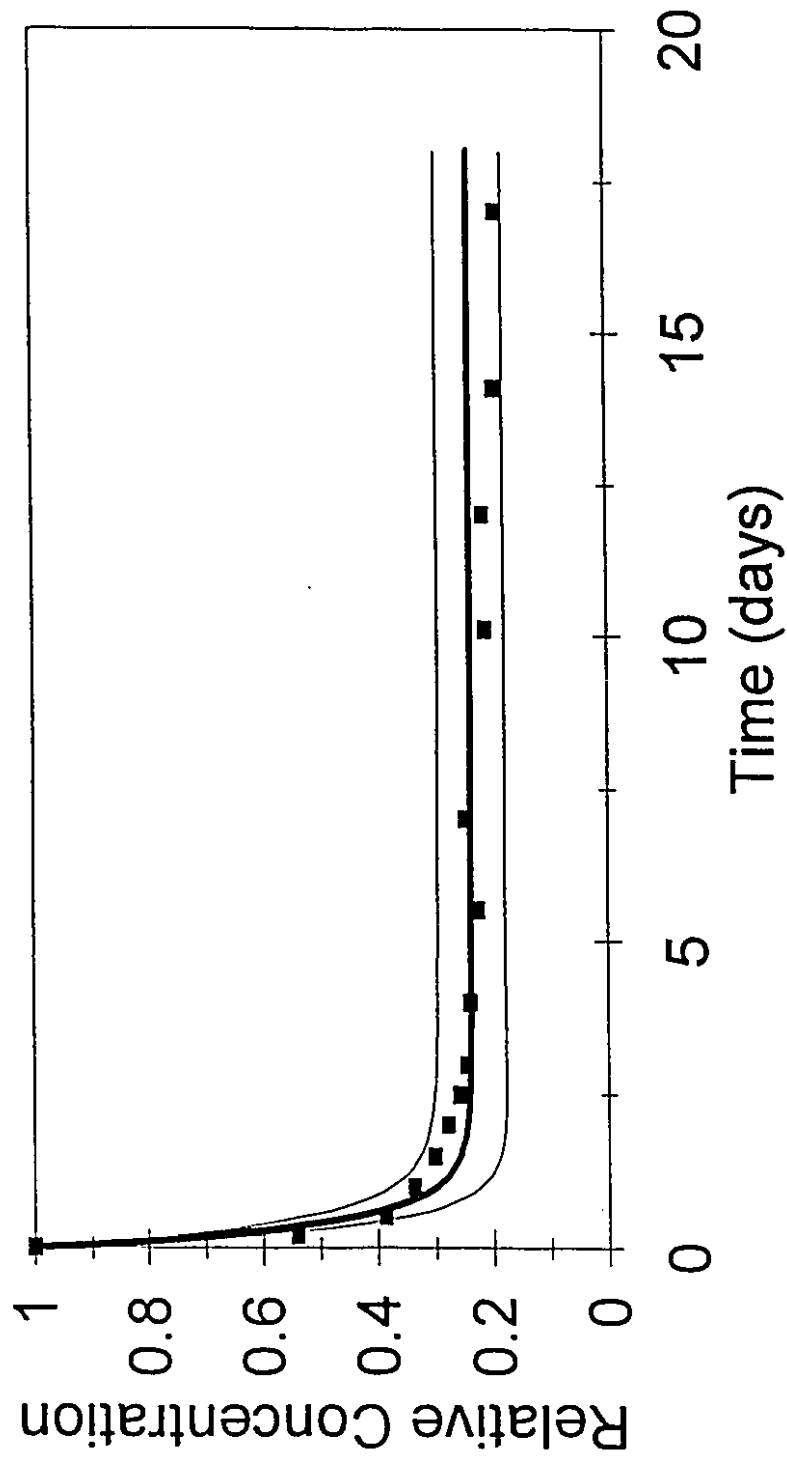
macro end
!*****
```

KINRAD



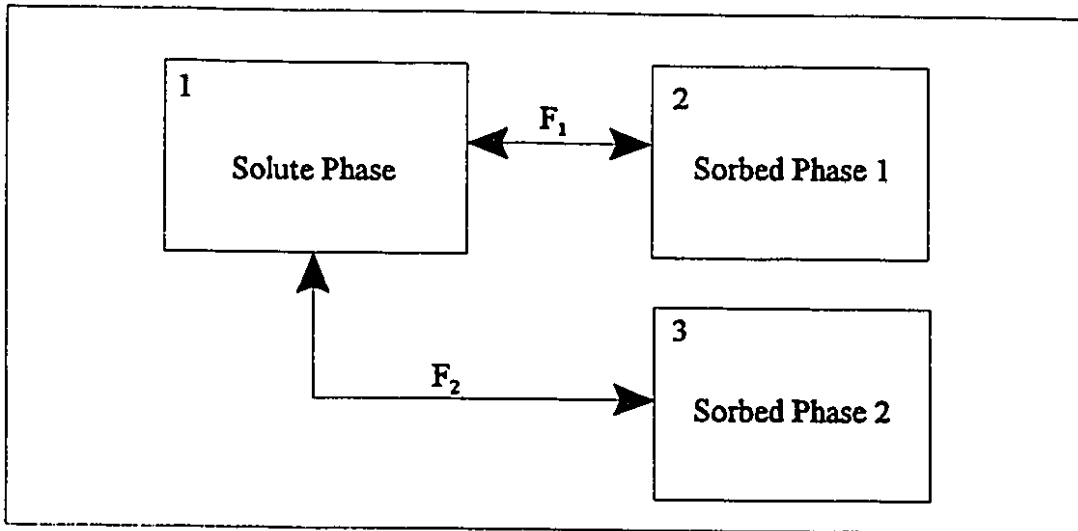
Relative Conc. vs. Time

Sensitivity Test - Kinrad



MODEL NUMBER: 3

Model Name: 3BOX



Flow Equations for State Variable 1:

F_1 = equilibrium sorption
= $f(kd)$

Number of parameters = 1

Number of operations = 3

F_2 = kinetic sorption
= $f(kf, kr)$

Number of parameters = 2

Number of operations = 3

Flow Equations for State Variable 2:

F_2 = kinetic sorption
= $f(kf, kr)$

Number of parameters = 2

Number of operations = 3

Flow Equations for State Variable 2:

F_1 = equilibrium sorption
= $f(kd)$

Number of parameters = 1

Number of operations = 3

COMPLEXITY: $I_{kc} = (1*3 + 2*3) + (1*3) + (2*3) = 18$

FIT: = 0.0749

3BOX.USR

```
!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!macro rvardata
!   procedural(dliquid=kd)
!   dimension kd(300), kf(300), kr(300)
!   open (unit=1, file='3box.sam', status='old')
!   read (1,*) (kd(j), kf(j), kr(j), j=1,300)
!   close(unit=1)
!   print *, 'parameter data entered'
!   end
!macro end

!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

!rvardata

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

equilsorb = 50*(sorbed1-liquid*kd)
kinsorb = -kf*liquid + kr*sorbed2
liquid = equilsorb + kinsorb
sorbed1 = -equilsorb
dsorbed2 = -kinsorb

liquid = integ(dliquid,initliquid)
sorbed1 = integ(dsorbed1,init sorbed1)
sorbed2 = integ(dsorbed2,init sorbed2)

liquid2 = liquid

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretes plus code to be executed every communication interval)
!Don't put macro definitions here

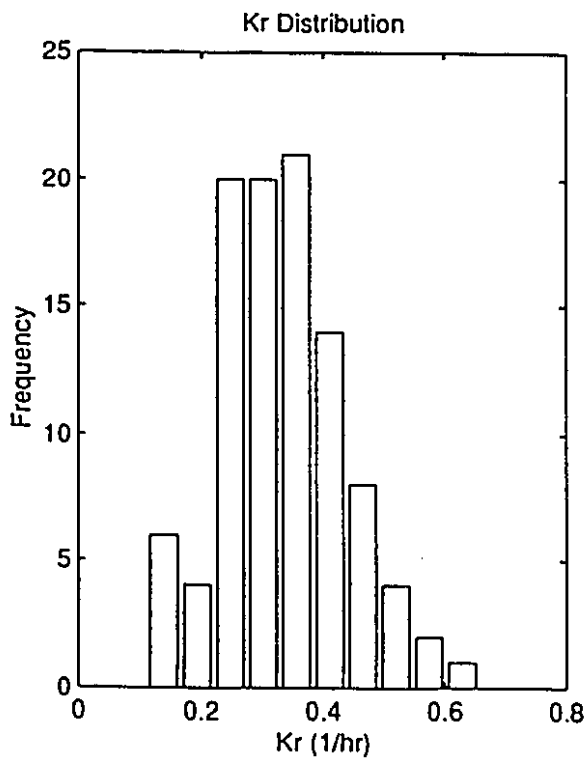
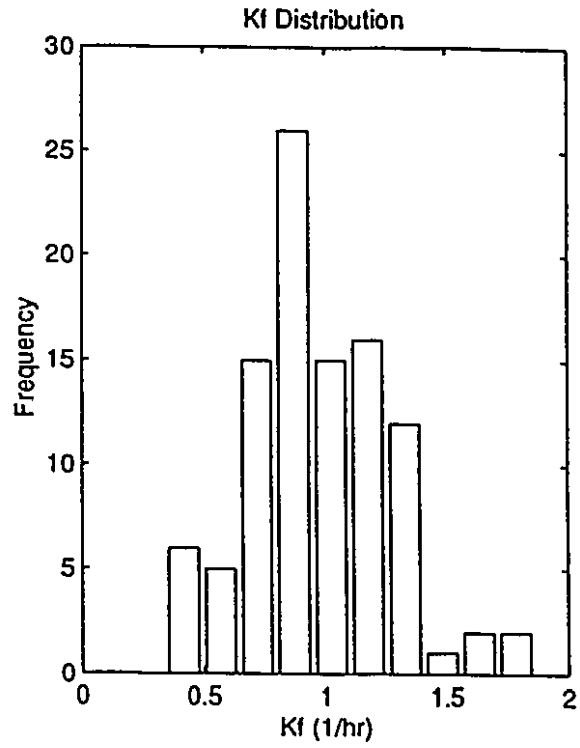
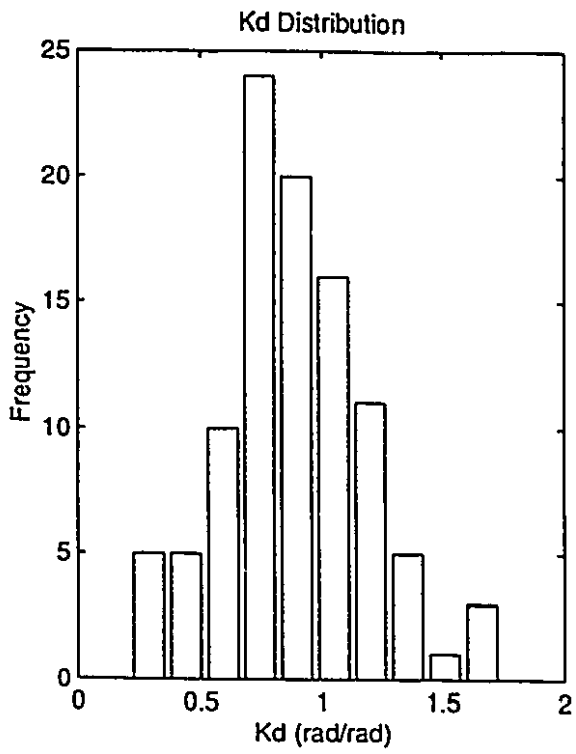
macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

!open (unit=1, file='sens.dat', status='old')
!write (1,*) kd,kf,kr
!close (unit=1)
!write (*,*) 'sens.dat written.'

!write (*,*) 'kd = ',kd(rnum), '   kf = ',kf(rnum), '   kr = ',kr(rnum)

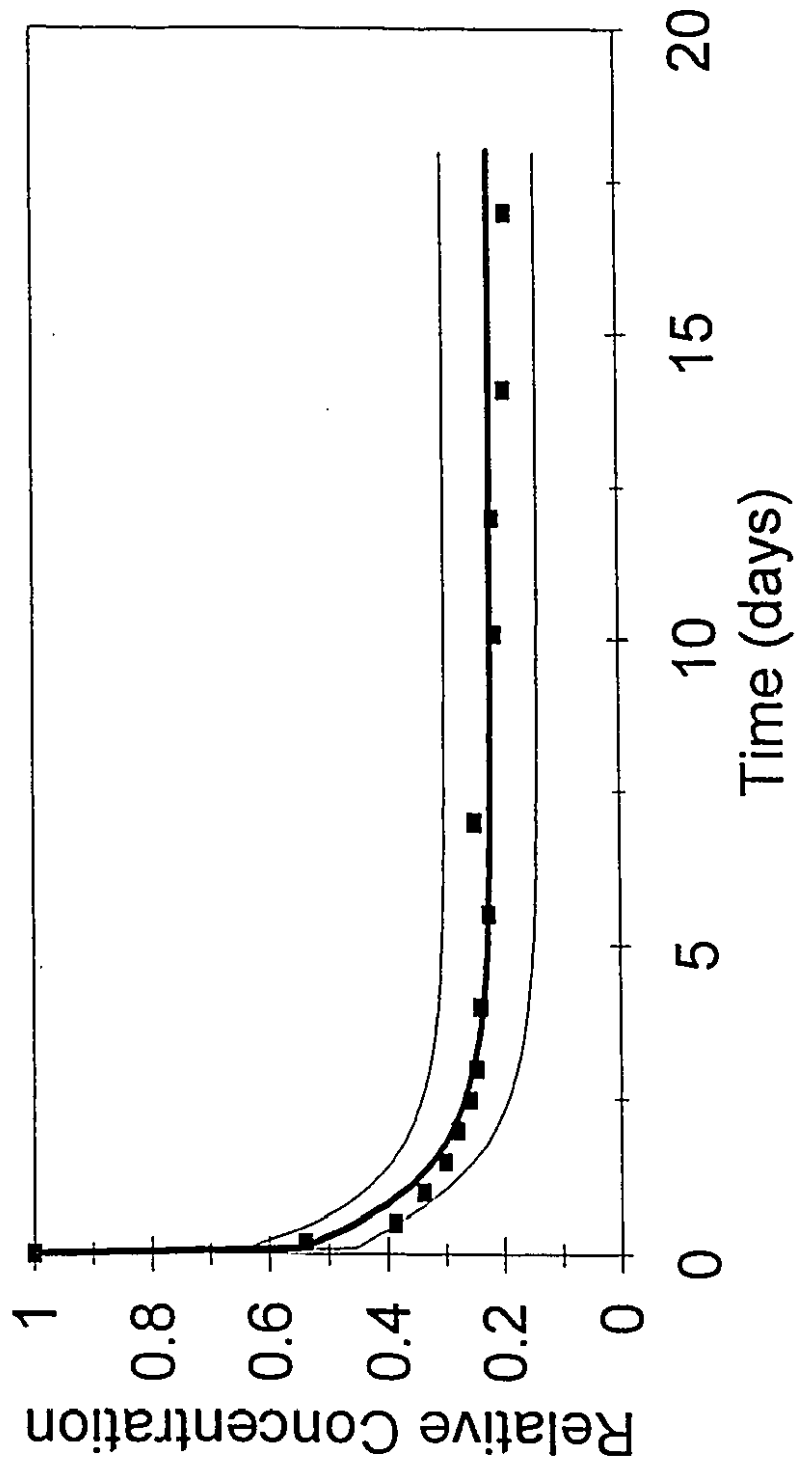
macro end
!*****
```

380K



Relative Conc. vs. Time

Sensitivity Test - 3Box



Appendix VI: Models Used in Case Study II - Transport of a Non-Conservative Tracer in a Homogeneous Aquifer

This appendix contains data regarding the seven models used in case study II of the uncertainty/complexity study found in the paper “Evaluating Modelling Uncertainty for Model Selection” in Appendix II. For each model, a box-and-arrow diagram is shown, which schematically represents the flow of matter between various states in the model. Each state and flow is labelled, and the processes are briefly described. The calculation of the index of model complexity, as introduced in Appendix I, is explicitly shown for each model.

Computer code for each model is found following each box-and-arrow diagram page. This code was implemented with the modelling software GPS-X (General Purpose Simulator) by Hydromantis, Inc. Only the .usr file is shown, which contains all the relevant rate equations to describe all flows in the model.

The computer code is followed by a series of histograms, representing the parameter distributions for each parameter used in the sensitivity studies described in Appendix II. Following this, a graphical representation of the output “envelope” is shown. The thick, black line represents the mean value of the multiple output runs generated during the Monte Carlo analysis, and the thin lines represent \pm one standard deviation at each timestep.

Information is presented for the following seven models:

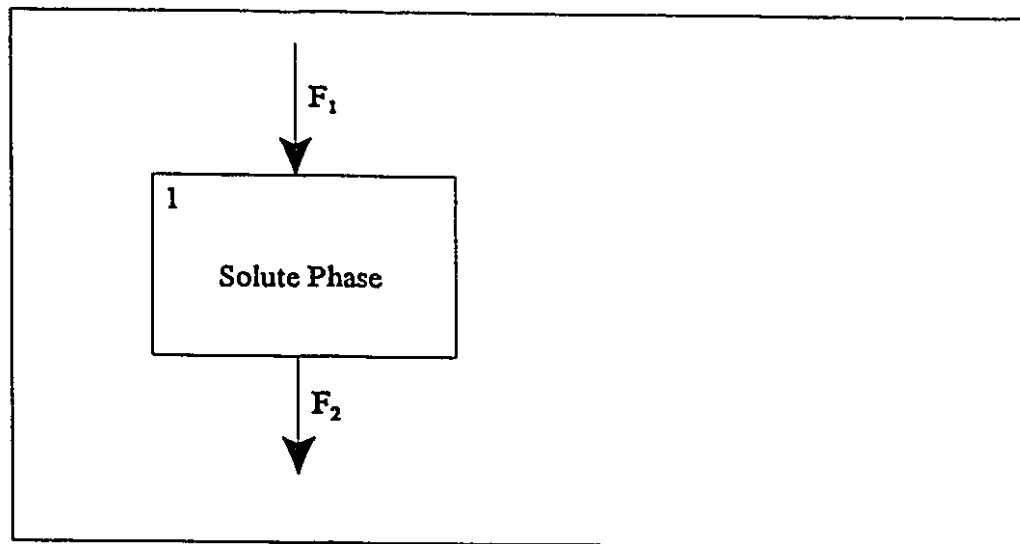
NOSORB-NODEG - no sorption or degradation
EQLINEAR-NODEG - equilibrium sorption with linear isotherm, no degradation
KNLINEAR-NODEG - kinetic sorption with linear isotherm, no degradation
EQLINEAR-1STORDER - equilibrium sorption w/ linear isotherm, 1st-order degradation

KNLINEAR-1STORDER - kinetic sorption w/ linear isotherm, 1st-order degradation
KNNON-LINEAR-1STORDER - kinetic sorption w/ non-linear isotherm, 1st-order
degradaton
KNMONOD-1STORDER - kinetic sorption w/ Monod-type isotherm, 1st-order
degradation

Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER: 1

Model Name: NOSORB/NODEG Model Location: c:\spencer3dmodel\models\nosorbnodeg\speed4.usr



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$
Number of parameters = 2
Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$
Number of parameters = 2
Number of operations = 2 (convective) + 2 (dispersive) = 4

COMPLEXITY: $I_{lc} = 2*4 + 2*4 = 16$

Dates: Start: March 24/98 End: Mar. 30/98

Sens data located: c:\spencer\3dmodel\noiseresults\nosorbnodeg

Crunched parameter values located: c:\spencer\3dmodel\noiseresults\nosorbnodeg\nsndsens.dat

Parameter Distribution located: JRK1:/snowling/sensprogram/nosorbnodeg/nsnd.ps

Output Parameters: porosity, xvel,yvel,zvel,optdif(1)

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

macro difmacro
  macro relabel 1888
  procedural(dif=dif,iiter,site,dat)
  dif = 0
  do 1888 q=1,5
    if (dat((q+1),iiter).gt.0.0) then
      dif=dif+((abs(dat((q+1),iiter)-site(q))**2.0)*w(q))
    end if
  1888..continue
end
macro end

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

!macro optprint
!  macro relabel aaa,baa,caa,daa,eea
!    if (optwarnings) then

!      open (unit=1,file='sens.dat',status='old',access='append')
!      write (1,*)
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
!      close (unit=1)
!      write (*,*) 'sens.dat written.'

!      if (termflag.eq. 'delta-p') print baa
!      baa..format(/,' Termination due to small change in parameters')
!      if (termflag.eq. 'delta-f') print caa
!      caa..format(/,' Termination due to small change in objective')
!      if (termflag.eq. 'loopleft') print daa
!      daa..format(/,' Termination caused by reaching maximum
iterations')
!      if (termflag.eq. 'OK') print eea
!      eea..format(/,' Termination due to reaching specified
objective')
!      if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
!      faa..format(/,' Termination caused by interrupt')
!      if(termflag.ne.'break'.and.termflag.ne.'brunbreak') then
!        do 101 optii=1,ndimm
!          print aaa,optii,optp(ilo,optii)
!          aaa..format(/,' Parameter 'il,' is ',f15.7)
!        101..continue
!      endif
!    endif
!macro end

macro soilprop
  watervolume=cellvolume*porosity
  soilvolume=cellvolume*(1.0-porosity)
!mL
!mL

```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz(maxz,maxx)                                     !g
    ! dimension showyz(maxz,maxy)                                     !g
    dimension sorbedamt(maxx,maxy,maxz)                               !g
    ! print*, 'iiter,t,dif'

    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar          !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)             !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
        icellmass(3,1,1)=0.367                      !g
macro end

macro calculatetransport
    !
    procedural(mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate,&
        watervolume,ratecon,xflow,yflow,zflow)
    !
    mflow=0

```

```

!      nosorb/nodeg
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
          sorbedamt(x,y,z)=0.0
          dcellmass(x,y,z)=0.0
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
          xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
          ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
          ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!
      if (x.eq.maxx) then
          xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+xtrans
!g/hr
      else
          xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
!g/hr
          dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
          ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+ztrans
!g/hr
      else
          ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
!g/hr
          dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
          if (y.eq.maxy) then
              ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
              mflow=mflow+ytrans
!g/hr
          else
              ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon

!g/hr
              dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

!g/hr
          endif

```

```

else
  if (y.eq.1) then
    ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
    !g/hr
    mflow=mflow+ytrans
  !g/hr
  else
    ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
    !g/hr
    dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
    !g/hr
  endif
endif

! This section calculates the derivatives for the liquid
! and sorbed masses.

dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans
!g/hr
dsorbmass(x,y,z)=sorbedamt(x,y,z)
!g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
  limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
  limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
  balance=integ(mflow,0.0)
  balance2=balance
macro end

macro mapviews

  site(1)=(cellmass(5,1,2)/watervolume)*1e9      !ug/L
  site(2)=(cellmass(5,2,3)/watervolume)*1e9      !ug/L
  site(3)=(cellmass(6,1,2)/watervolume)*1e9      !ug/L
  site(4)=(cellmass(17,9,3)/watervolume)*1e9     !ug/L
  site(5)=(cellmass(17,9,4)/watervolume)*1e9     !ug/L
! site(6)=(cellmass(20,9,4)/watervolume)*1e9     !ug/L
! site(7)=(cellmass(20,9,3)/watervolume)*1e9     !ug/L

macro end
!*****
macro userinitialsection

```

```

!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretizes plus code to be executed every communication interval)
!Don't put macro definitions here

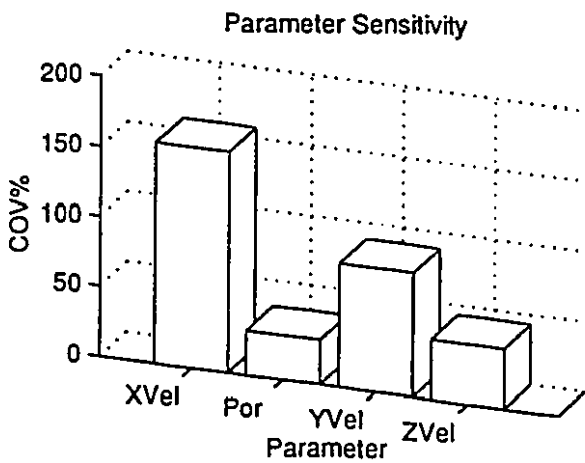
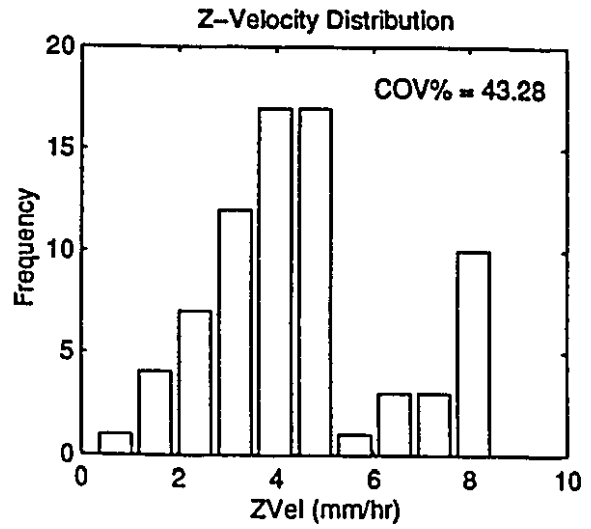
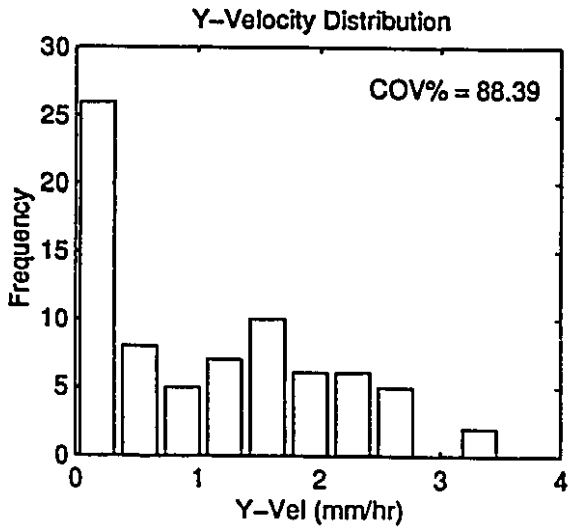
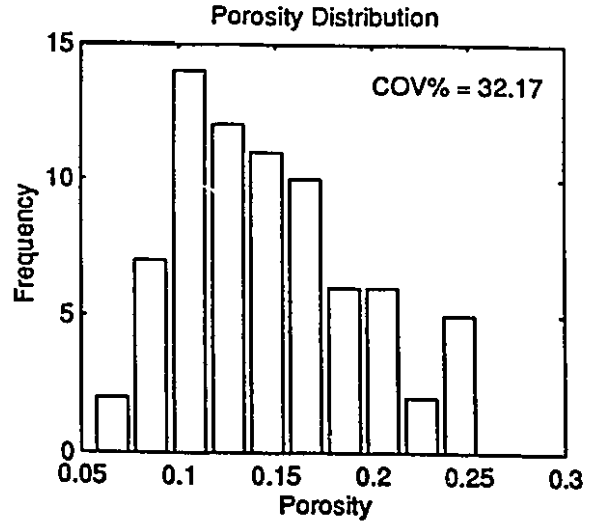
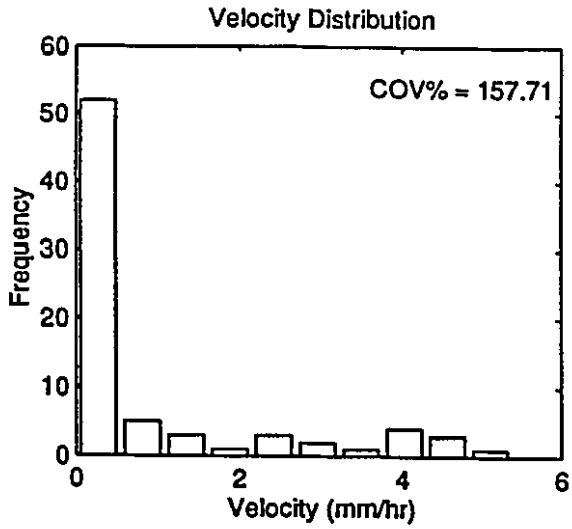
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

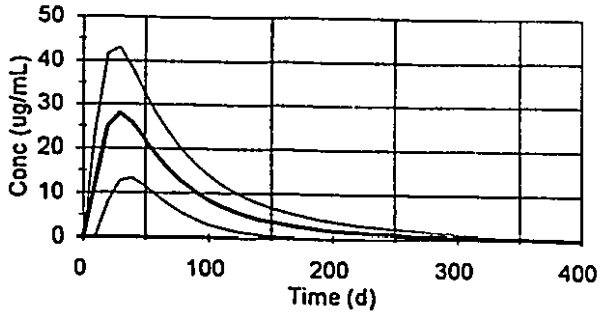
open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1)
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

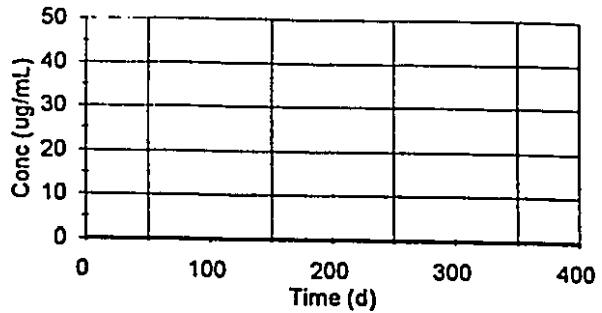
```



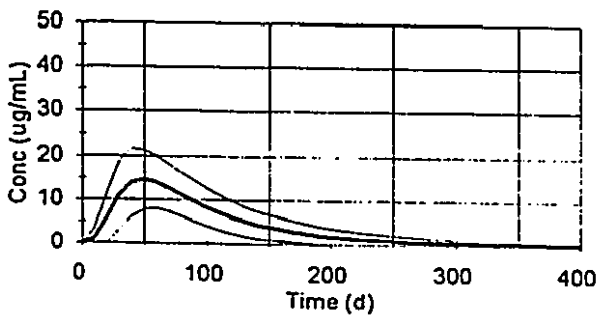
Conc. vs. Time - Site 1
NOSORBNODEG



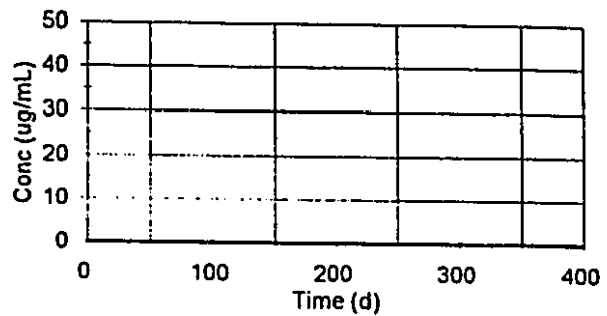
Conc. vs. Time - Site 4
NOSORBNODEG



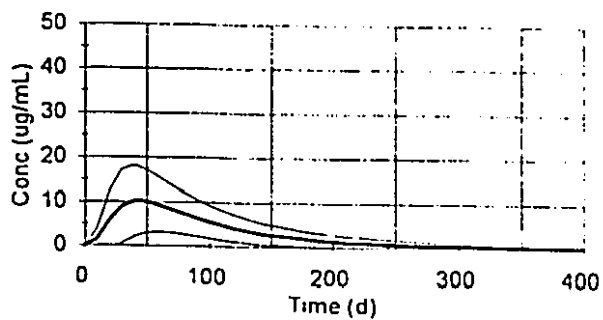
Conc. vs. Time - Site 2
NOSORBNODEG



Conc. vs. Time - Site 5
NOSORBNODEG



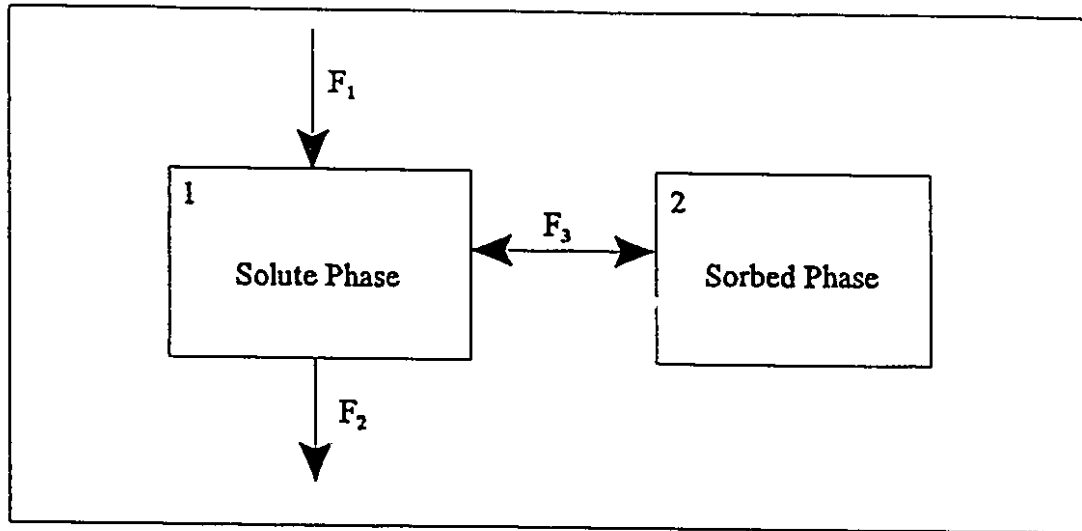
Conc. vs. Time - Site 3
NOSORBNODEG



Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER 2:

Model Name: EQLINEAR/NODEG Model Location: c:\spencer\3dmodel\models\eqlinear-nodeg\speed4.usr



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
 = $f(\text{porosity, velocity})$
 Number of parameters = 2
 Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
 = $f(\text{porosity, velocity})$
 Number of parameters = 2
 Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (linear equilibrium)
 = $f(kd)$
 Number of parameters = 1
 Number of operations = 3

Flow Equations for State Variable 2:

F_3 = sorption/desorption (linear equilibrium)
 = $f(kd)$
 Number of parameters = 1
 Number of operations = 3

Dates: Start: Aug.18/98 End: Aug.24/98
 Sens data located: c:\spencer\3dmodel\noiseresults\linear-nodeg
 Parameter values located: c:\spencer...eq\nd.dat
 Parameter Distribution located: _____

COMPLEXITY: $I_{kc} = (2*4 + 2*4 + 1*3) + (1*3) = 22$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), kd

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

macro difmacro
macro relabel 1888
procedural(dif=dif,iiter,site,dat)
dif = 0
do 1888 q=1,5
if (dat((q+1),iiter).gt.0.0) then
dif=dif+((abs(dat((q+1),iiter)-site(q)))*2.0)*w(q)
end if
1888..continue
end
macro end

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

!macro optprint
! macro relabel aaa,baa,caa,daa,eea
! if (optwarnings) then

! open (unit=1,file='sens.dat',status='old',access='append')
! write (1,*)
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
! close (unit=1)
! write (*,*) 'sens.dat written.'

! if (termflag.eq. 'delta-p') print baa
! baa..format(/,' Termination due to small change in parameters')
! if (termflag.eq. 'delta-f') print caa
! caa..format(/,' Termination due to small change in objective')
! if (termflag.eq. 'looplimit') print daa
! daa..format(/,' Termination caused by reaching maximum
iterations')
! if (termflag.eq. 'OK') print eea
! eea..format(/,' Termination due to reaching specified
objective')
! if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
! faa..format(/,' Termination caused by interrupt')
! if(termflag.ne.'break'.and.termflag.ne.'brunbreak') then
! do 101 optii=1,ndimm
! print aaa,optii,optp(ilo,optii)
! aaa..format(/,' Parameter 'il,' is ',f15.7)
! 101..continue
! endif
! endif
!macro end

macro soilprop
watervolume=cellvolume*porosity !mL
soilvolume=cellvolume*(1.0-porosity) !mL

```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz(maxz,maxx)                                     !g
    ! dimension showyz(maxz,maxy)                                     !g
    dimension sorbedamt(maxx,maxy,maxz)                                !g
    ! print*, 'iter,t,dif'
    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar          !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)             !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
    icellmass(3,1,1)=0.367                            !g
macro end

macro calculatetransport
    !
    procedural (mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate, &
    ,
    watervolume,ratecon,xflow,yflow,zflow)
    mflow=0

```

```

!      eqlinear/nodeg
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
          sorbedamt(x,y,z)=50*(kd*cellmass(x,y,z)-sorbmass(x,y,z))
!degradation
          dcellmass(x,y,z)=0.0-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
          xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
          ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
          ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
          xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+xtrans
!g/hr
      else
          xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
!g/hr
          dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
          ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+ztrans
!g/hr
      else
          ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
!g/hr
          dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
          if (y.eq.maxy) then
              ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
              mflow=mflow+ytrans
!g/hr
          else
              ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
!g/hr
              dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
        endif
    else
        if (y.eq.1) then
            ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
            !g/hr
            mflow=mflow+ytrans
            !g/hr
        else
            ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
            !g/hr
            dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
            !g/hr
            endif
        endif
    endif

! This section calculates the derivatives for the liquid
! and sorbed masses.

dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans
    !g/hr
dsorbmass(x,y,z)=sorbedamt(x,y,z)
    !g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
    limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
    limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
    balance=integ(mflow,0.0)
    balance2=balance
macro end

macro mapviews
    site(1)=(cellmass(5,1,2)/watervolume)*1e9      !ug/L
    site(2)=(cellmass(5,2,3)/watervolume)*1e9      !ug/L
    site(3)=(cellmass(6,1,2)/watervolume)*1e9      !ug/L
    site(4)=(cellmass(17,9,3)/watervolume)*1e9     !ug/L
    site(5)=(cellmass(17,9,4)/watervolume)*1e9     !ug/L
! site(6)=(cellmass(20,9,4)/watervolume)*1e9     !ug/L
! site(7)=(cellmass(20,9,3)/watervolume)*1e9     !ug/L

macro end

```

```

!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretizes plus code to be executed every communication interval)
!Don't put macro definitions here

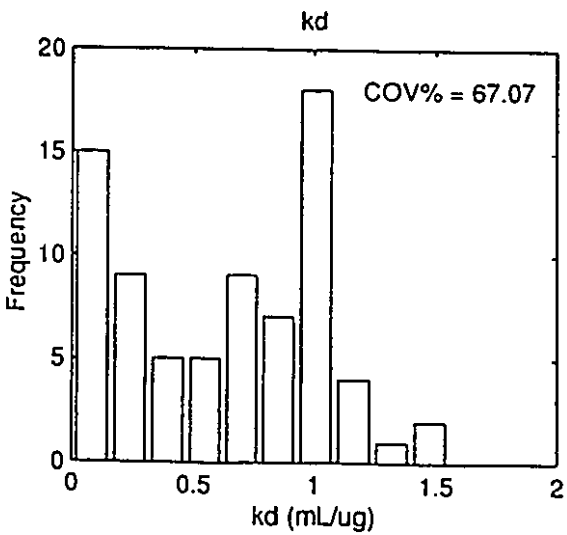
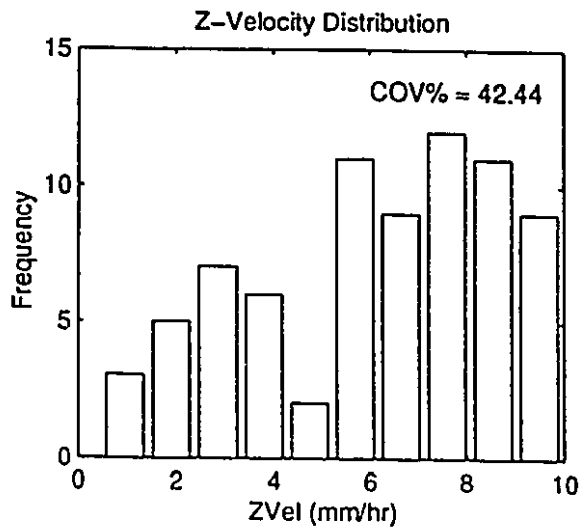
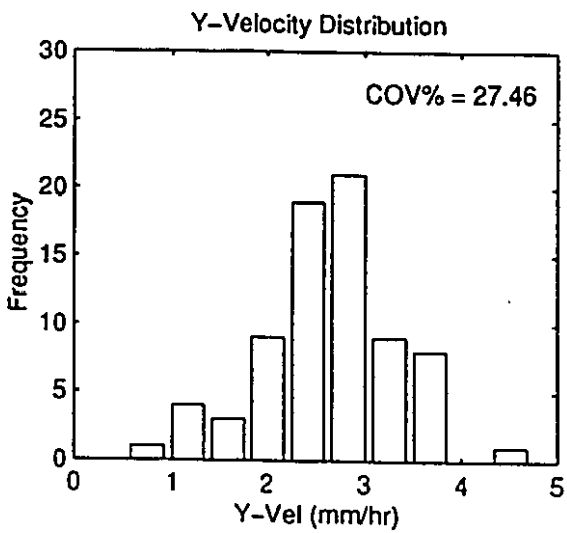
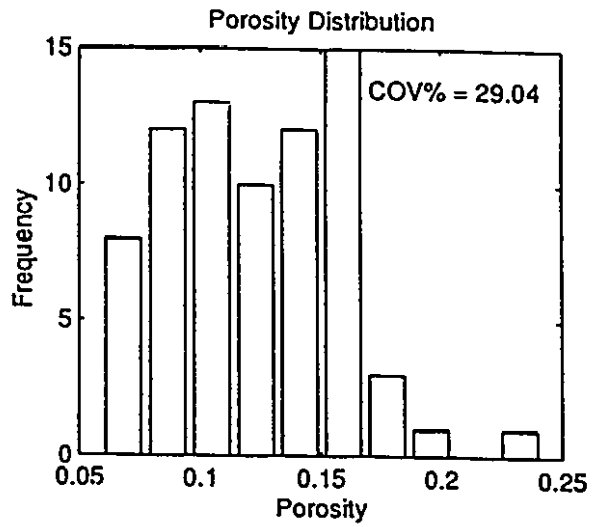
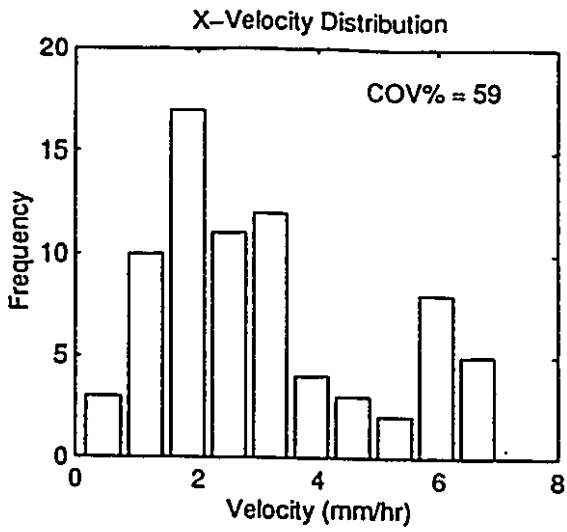
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

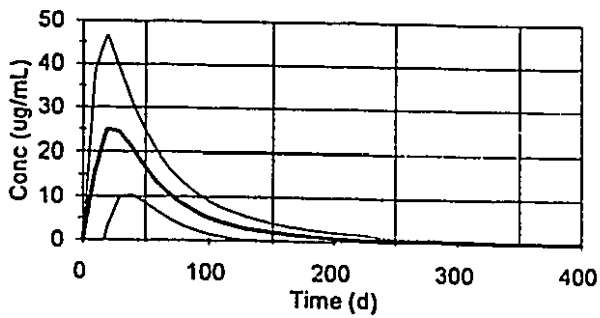
open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1),kd
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

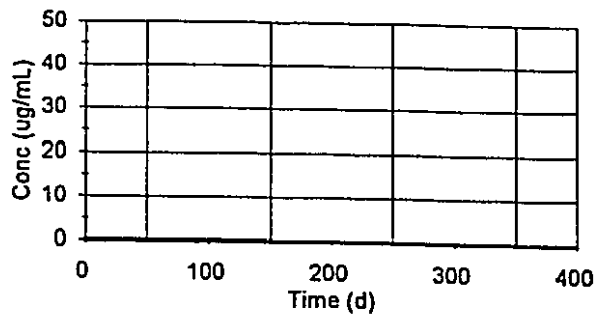
```



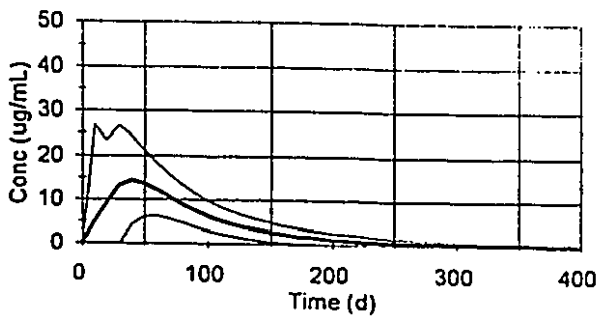
Conc. vs. Time - Site 1
EQLINEAR-NODEG



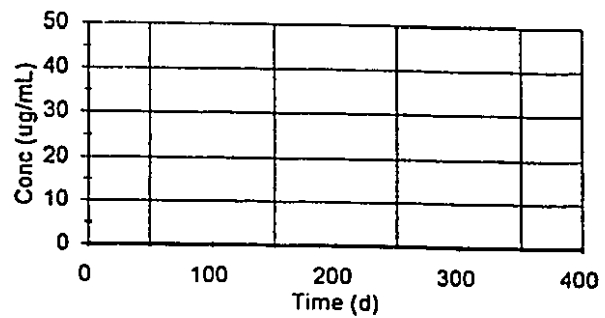
Conc. vs. Time - Site 4
EQLINEAR-NODEG



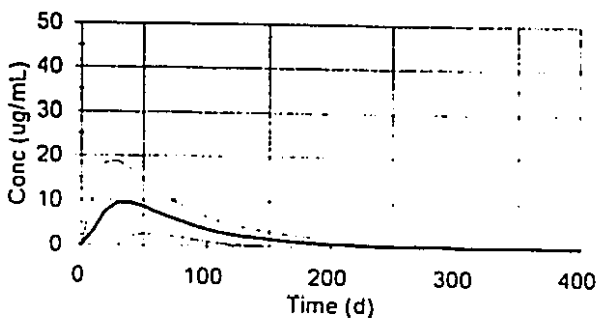
Conc. vs. Time - Site 2
EQLINEAR-NODEG



Conc. vs. Time - Site 5
EQLINEAR-NODEG



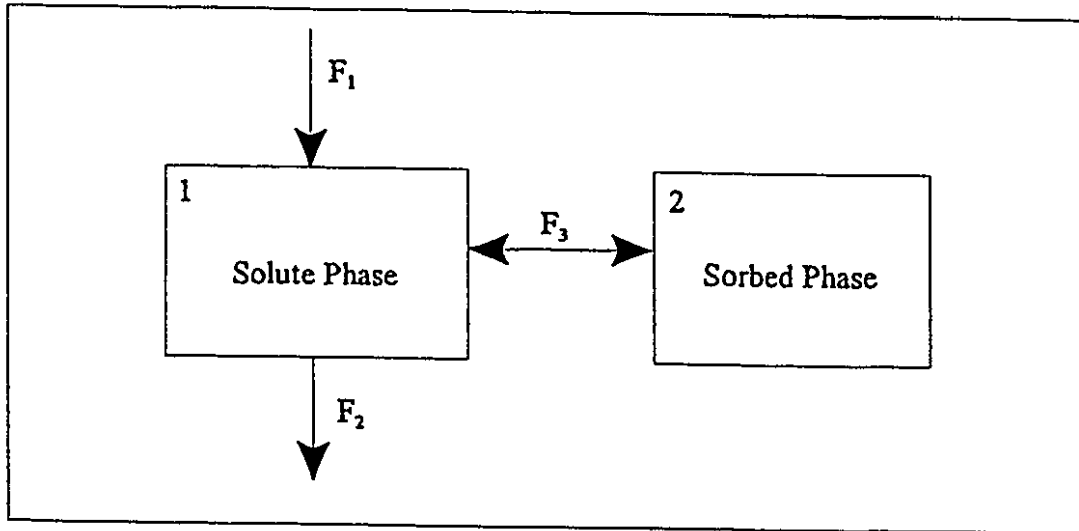
Conc. vs. Time - Site 3
EQLINEAR-NODEG



Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER: 3

Model Name: KNLINER/NODEG Model Location: c:\spencer\3dmodel\models\knliner-nodeg\spced4.usr



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
 = $f(\text{porosity, velocity})$
 Number of parameters = 2
 Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
 = $f(\text{porosity, velocity})$
 Number of parameters = 2
 Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (linear kinetic)
 = $f(kd, \text{sorption rate constant})$
 Number of parameters = 2
 Number of operations = 3

Flow Equations for State Variable 2:

F_3 = sorption/desorption (linear kinetic)
 = $f(kd, \text{sorption rate constant})$
 Number of parameters = 2
 Number of operations = 3

Dates: Start: _____ End: _____
 Sens data located: c:\spencer\...knliner-nodeg\
 Crunched parameter values located: ...knl.in.dat
 Parameter Distribution located: _____

COMPLEXITY: $I_{kc} = (2*4 + 2*4 + 2*3) + (2*3) = 28$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), kd, sorbrate

```
!*****
```

```
!PUT HERE USER DEFINED MACROS
```

```
!This section is for definitions, it will be executed
```

```
!only during loading of the program
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
! speed4.usr - !
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
macro difmacro
```

```
macro relabel 1888
```

```
procedural(dif=dif,iiter,site,dat)
```

```
dif = 0
```

```
do 1888 q=1,5
```

```
if (dat((q+1),iiter).gt.0.0) then
```

```
dif=dif+((abs(dat((q+1),iiter)-site(q))**2.0)*w(q)
```

```
end if
```

```
1888..continue
```

```
end
```

```
macro end
```

```
!*****
```

```
! this macro prints out the parameters when the optimizer
```

```
! is terminated
```

```
!*****
```

```
!macro optprint
```

```
! macro relabel aaa,baa,caa,daa,eea
```

```
! if (optwarnings) then
```

```
! open (unit=1,file='sens.dat',status='old',access='append')
```

```
! write (1,*)
```

```
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
```

```
! close (unit=1)
```

```
! write (*,*) 'sens.dat written.'
```

```
! if (termflag.eq. 'delta-p') print baa
```

```
! baa..format(/,' Termination due to small change in parameters')
```

```
! if (termflag.eq. 'delta-f') print caa
```

```
! caa..format(/,' Termination due to small change in objective')
```

```
! if (termflag.eq. 'loopleft') print daa
```

```
! daa..format(/,' Termination caused by reaching maximum
```

```
iterations')
```

```
! if (termflag.eq. 'OK') print eaa
```

```
! eaa..format(/,' Termination due to reaching specified
```

```
objective')
```

```
! if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
```

```
! faa..format(/,' Termination caused by interrupt')
```

```
! if(termflag.ne.'break'.and.termflag.ne.'brunbreak') then
```

```
! do 101 optii=1,ndimm
```

```
! print aaa,optii,optp(ilo,optii)
```

```
! aaa..format(/,' Parameter 'il,' is ',f15.7)
```

```
! 101..continue
```

```
! endif
```

```
! endif
```

```
!macro end
```

```
macro soilprop
```

```
watervolume=cellvolume*porosity
```

```
!mL
```

```
soilvolume=cellvolume*(1.0-porosity)
```

```
!mL
```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz(maxz,maxx)                                     !g
    ! dimension showyz(maxz,maxy)                                     !g
    dimension sorbedamt(maxx,maxy,maxz)                               !g
    ! print*, 'iiter,t,dif'
    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar           !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)              !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
        icellmass(3,1,1)=0.367                      !g
macro end

macro calculatetransport
    !
    procedural(mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate,&
        watervolume,ratecon,xflow,yflow,zflow)
    !
    mflow=0

```

```

!      knlinear/nodeg
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
      sorbedamt(x,y,z)=sorbrate*(kd*cellmass(x,y,z)-
sorbmass(x,y,z))
!degradation
      dcellmass(x,y,z)=0.0-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
      xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
      ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
      ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
      xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
      mflow=mflow+xtrans
      !g/hr
      else
      xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
      !g/hr
      dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
      ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
      mflow=mflow+ztrans
      !g/hr
      else
      ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
      !g/hr
      dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
      if (y.eq.maxy) then
      ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
      mflow=mflow+ytrans
      !g/hr
      else
      ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
      !g/hr
      dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
        endif
    else
        if (y.eq.1) then
            ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
            !g/hr
            mflow=mflow+ytrans
            !g/hr
        else
            ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
            !g/hr
            dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
            !g/hr
        endif
    endif

!       This section calculates the derivatives for the liquid
!       and sorbed masses.

        dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans

            !g/hr
            dsorbmass(x,y,z)=sorbedamt(x,y,z)

            !g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
    limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
    limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
    balance=integ(mflow,0.0)
    balance2=balance
macro end

macro mapviews

    site(1)=(cellmass(5,1,2)/watervolume)*1e9           !ug/L
    site(2)=(cellmass(5,2,3)/watervolume)*1e9           !ug/L
    site(3)=(cellmass(6,1,2)/watervolume)*1e9           !ug/L
    site(4)=(cellmass(17,9,3)/watervolume)*1e9          !ug/L
    site(5)=(cellmass(17,9,4)/watervolume)*1e9          !ug/L
!   site(6)=(cellmass(20,9,4)/watervolume)*1e9          !ug/L
!   site(7)=(cellmass(20,9,3)/watervolume)*1e9          !ug/L

```

```

macro end
!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretizes plus code to be executed every communication interval)
!Don't put macro definitions here

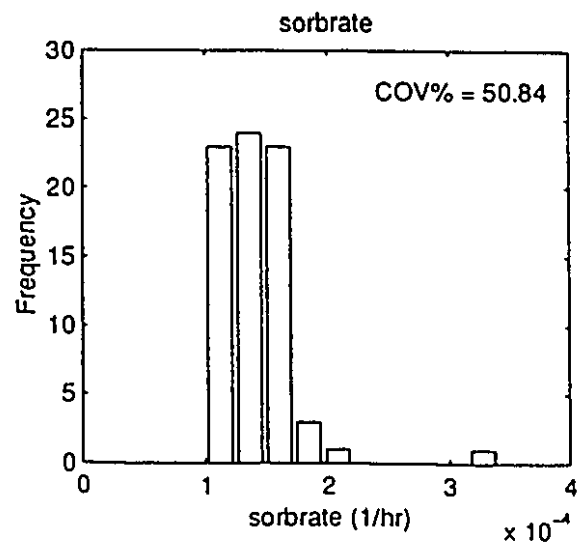
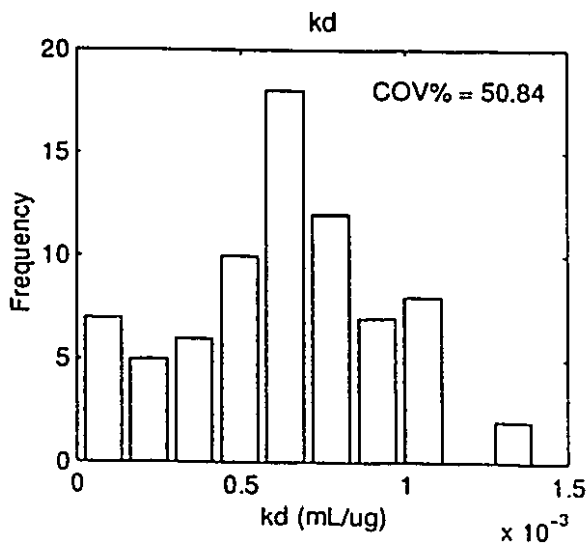
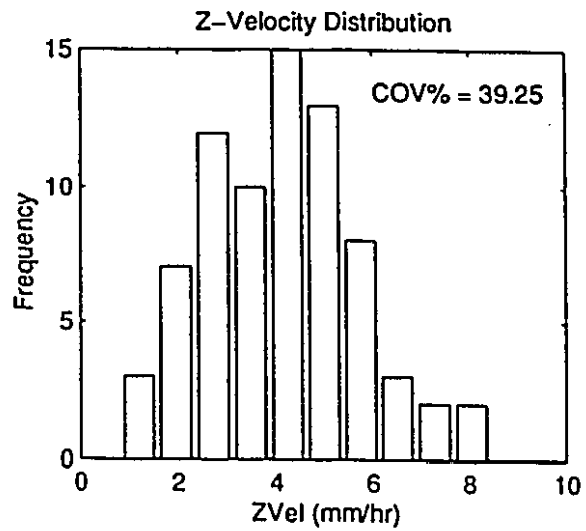
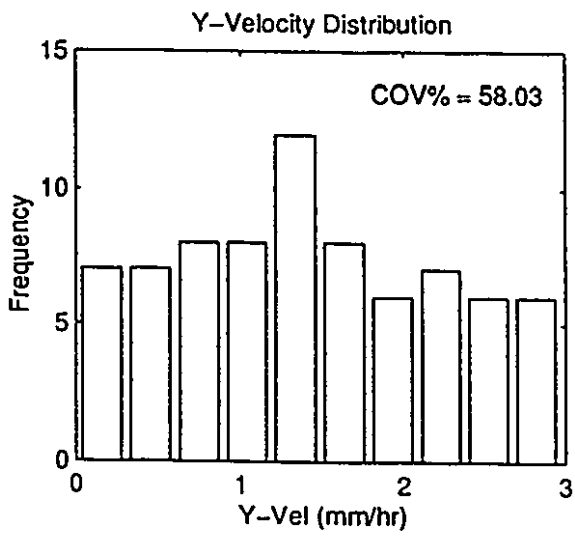
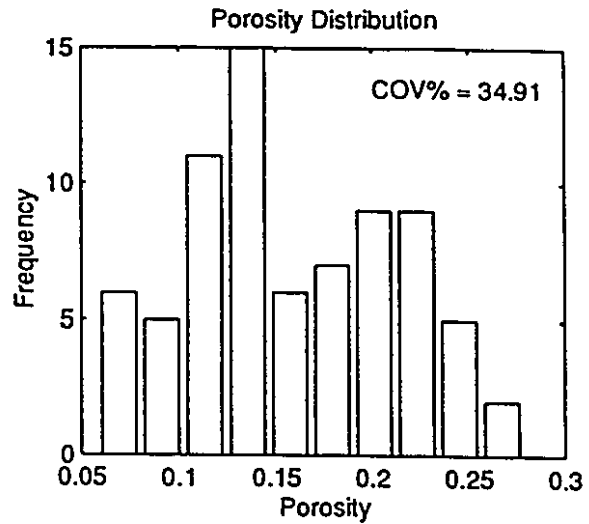
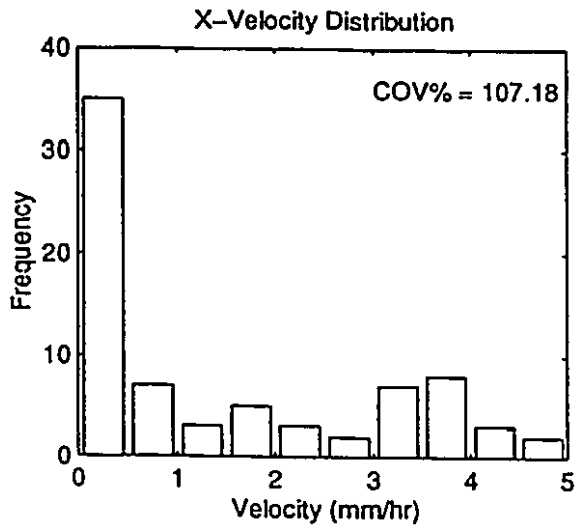
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1),kd,sorbrate
close (unit=1)
write (*,*) 'sens.dat written.'

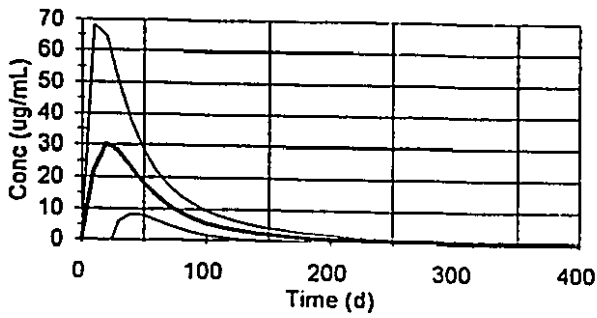
macro end
!*****

```



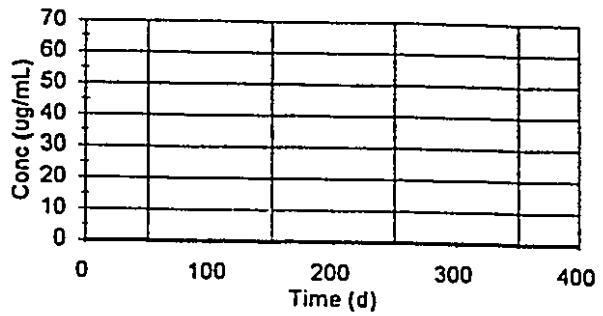
Conc. vs. Time - Site 1

KNLINEAR-NODEG



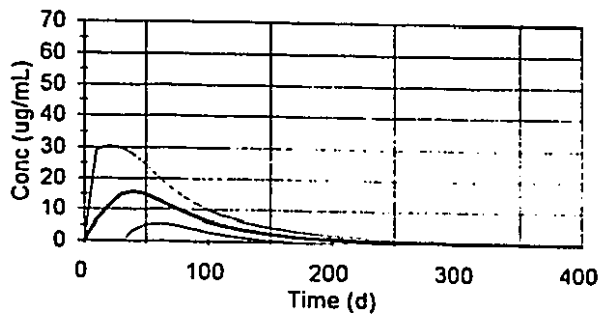
Conc. vs. Time - Site 4

KNLINEAR-NODEG



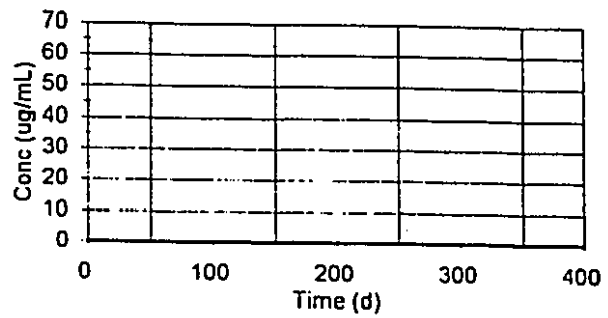
Conc. vs. Time - Site 2

KNLINEAR-NODEG



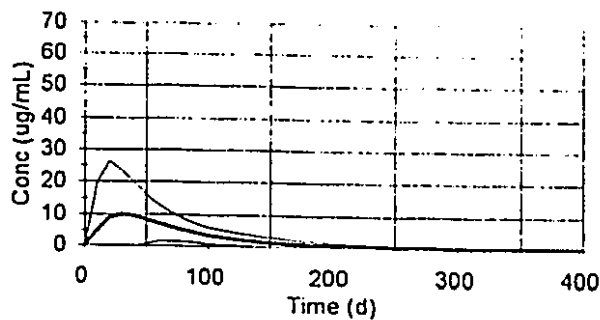
Conc. vs. Time - Site 5

KNLINEAR-NODEG



Conc. vs. Time - Site 3

KNLINEAR-NODEG

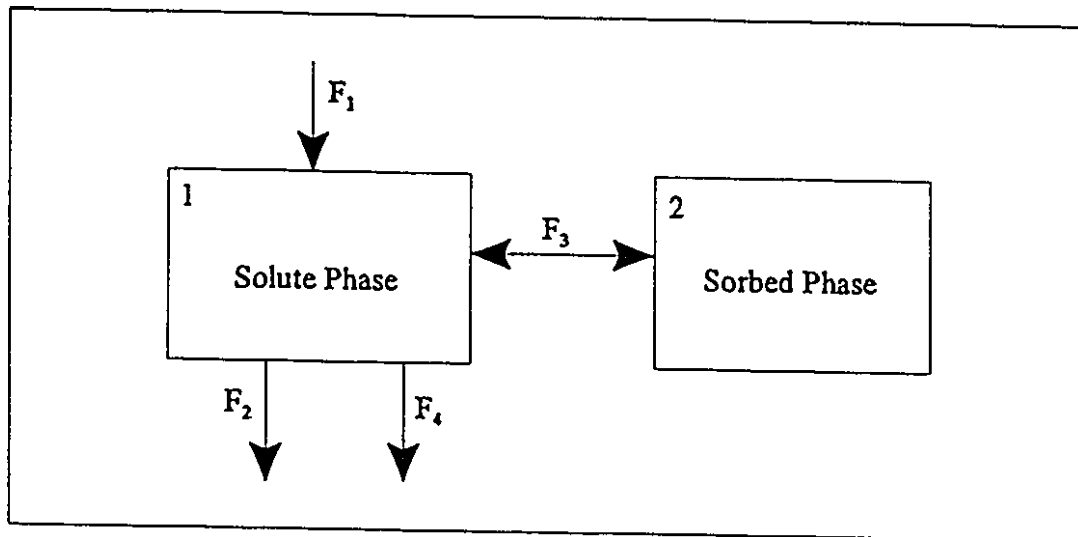


Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER 4:

Model Name: EQLINEAR/ISTORDER

Model Location: c:\spencer3dmodel\models\ eqlinear-1st



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (linear equilibrium)
= $f(kd)$

Number of parameters = 1

Number of operations = 3

F_4 = degradation (1st order)

= $f(\text{degradation constant (sorbed)})$

Number of parameters = 1

Number of operations = 1

Flow Equations for State Variable 2:

F_3 = sorption/desorption (linear equilibrium)
= $f(kd)$

Number of parameters = 1

Number of operations = 3

Dates: Start: _____ End: _____

Sens data located: _____

Parameter values located: _____

Parameter Distribution located: _____

COMPLEXITY: $I_{tc} = (2*4 + 2*4 + 1*3 + 1*1) + (1*3 + 1*1) = 24$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), kd, muliq1

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

macro difmacro
  macro relabel 1888
    procedural (dif=dif,iiter,site,dat)
    dif = 0
    do 1888 q=1,5
      if (dat((q+1),iiter).gt.0.0) then
        dif=dif+((abs(dat((q+1),iiter)-site(q)))*2.0)*w(q)
      end if
    1888..continue
  end
macro end

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

!macro optprint
!  macro relabel aaa,baa,caa,daa,eea
!  if (optwarnings) then

!    open (unit=1,file='sens.dat',status='old',access='append')
!    write (1,*)
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
!    close (unit=1)
!    write (*,*) 'sens.dat written.'

!    if (termflag.eq. 'delta-p') print baa
!    baa..format(/,' Termination due to small change in parameters')
!    if (termflag.eq. 'delta-f') print caa
!    caa..format(/,' Termination due to small change in objective')
!    if (termflag.eq. 'looplimit') print daa
!    daa..format(/,' Termination caused by reaching maximum
iterations')
!    if (termflag.eq. 'OK') print eaa
!    eaa..format(/,' Termination due to reaching specified
objective')
!    if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
!    faa..format(/,' Termination caused by interrupt')
!    if (termflag.ne.'break'.and.termflag.ne.'brunbreak') then
!      do 101 optii=1,ndimm
!        print aaa,optii,optp(ilo,optii)
!        aaa..format(/,' Parameter 'il,' is ',f15.7)
!      101..continue
!    endif
!  endif
!macro end

macro soilprop
  watervolume=cellvolume*porosity
  soilvolume=cellvolume*(1.0-porosity)
!mL
!mL

```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz(maxz,maxx)                                     !g
    ! dimension showyz(maxz,maxy)                                     !g
    dimension sorbedamt(maxx,maxy,maxz)                               !g
    ! print*, 'iiter,t,dif'
    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar          !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)             !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
    icellmass(3,1,1)=0.367                            !g
macro end

macro calculatetransport
    !
    procedural (mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate,&
                watervolume,ratecon,xflow,yflow,zflow)
    !
    mflow=0

```

```

!      eqlinear/lstorder
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
          sorbedamt(x,y,z)=50*(kd*cellmass(x,y,z)-sorbmass(x,y,z))
!degradation
          dcellmass(x,y,z)=muliq1*cellmass(x,y,z)-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
          xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
          ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
          ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
          xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+xtrans
!g/hr
      else
          xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
!g/hr
          dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
          ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+ztrans
!g/hr
      else
          ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
!g/hr
          dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
          if (y.eq.maxy) then
              ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
              mflow=mflow+ytrans
!g/hr
          else
              ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
!g/hr
              dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
      endif
    else
      if (y.eq.1) then
        ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
        !g/hr
        mflow=mflow+ytrans
      !g/hr
      else
        ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
        !g/hr
        dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
        !g/hr
      endif
    endif

!      This section calculates the derivatives for the liquid
!      and sorbed masses.

      dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans

      !g/hr
      dsorbmass(x,y,z)=sorbedamt(x,y,z)

      !g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
  limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
  limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
  balance=integ(mflow,0.0)
  balance2=balance
macro end

macro mapviews

  site(1)=(cellmass(5,1,2)/watervolume)*1e9      !ug/L
  site(2)=(cellmass(5,2,3)/watervolume)*1e9      !ug/L
  site(3)=(cellmass(6,1,2)/watervolume)*1e9      !ug/L
  site(4)=(cellmass(17,9,3)/watervolume)*1e9     !ug/L
  site(5)=(cellmass(17,9,4)/watervolume)*1e9     !ug/L
!   site(6)=(cellmass(20,9,4)/watervolume)*1e9   !ug/L
!   site(7)=(cellmass(20,9,3)/watervolume)*1e9   !ug/L

macro end

```

```

!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretess plus code to be executed every communication interval)
!Don't put macro definitions here

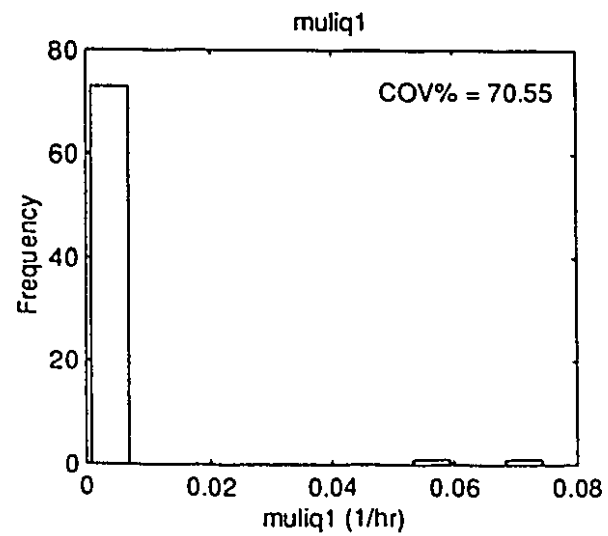
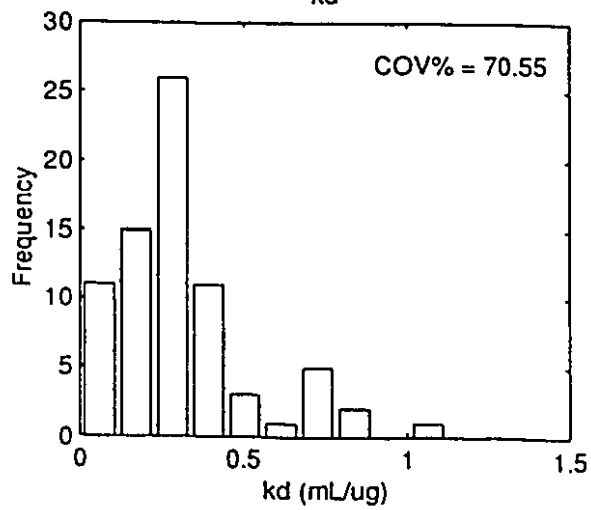
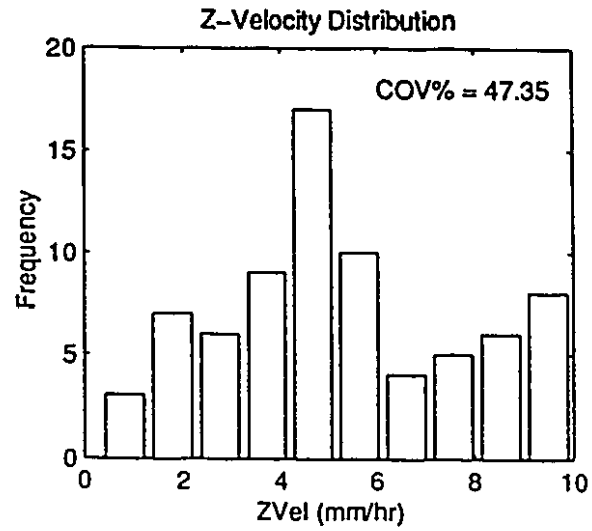
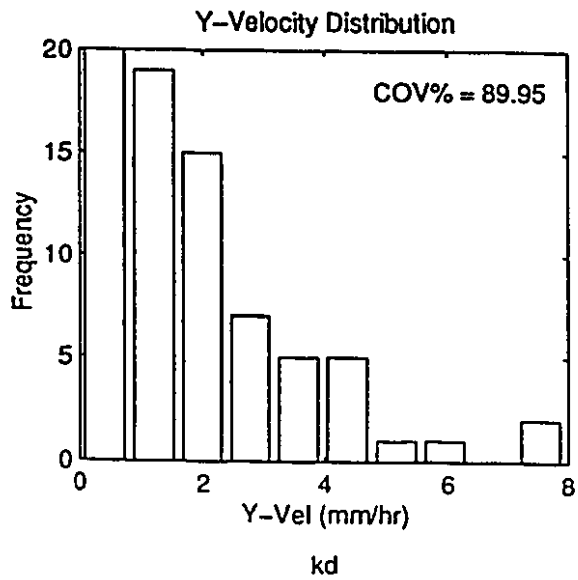
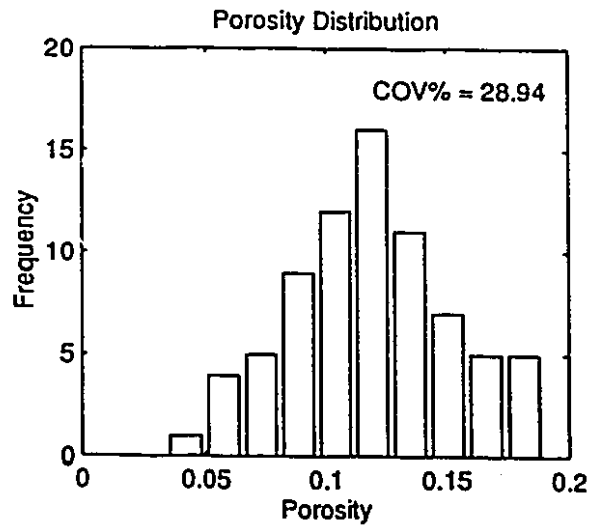
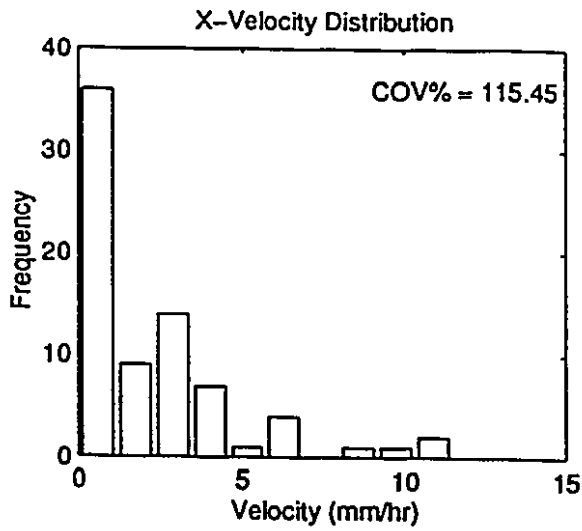
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

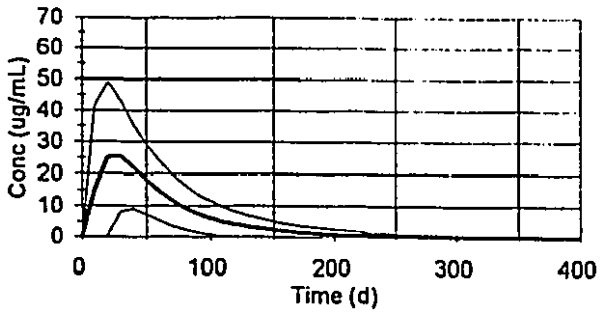
open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1),kd,muliql
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

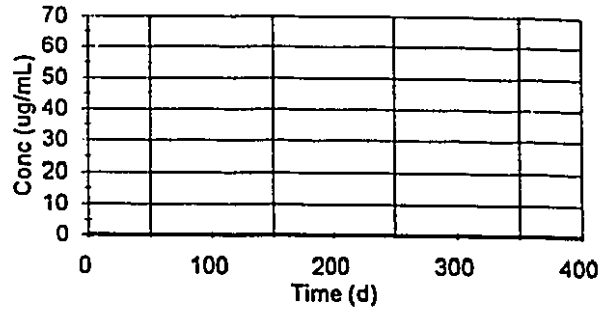
```



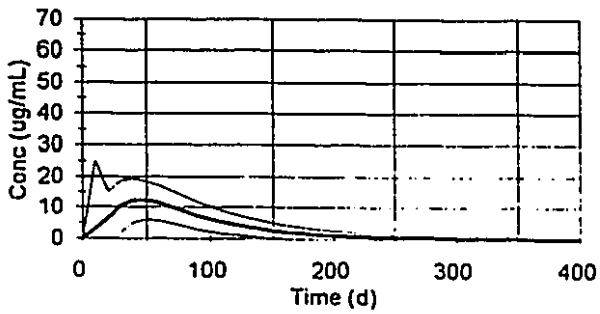
Conc. vs. Time - Site 1
EQLINEAR-1STORDER



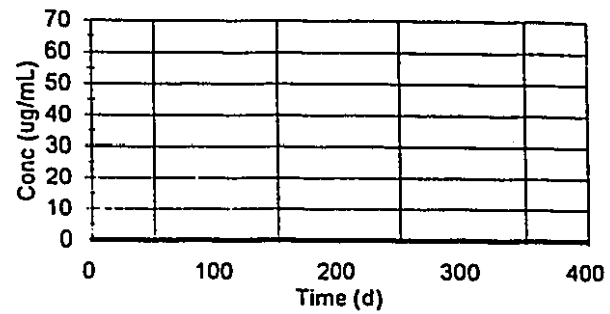
Conc. vs. Time - Site 4
EQLINEAR-1STORDER



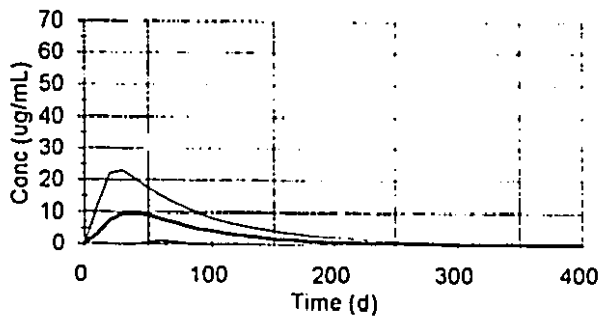
Conc. vs. Time - Site 2
EQLINEAR-1STORDER



Conc. vs. Time - Site 5
EQLINEAR-1STORDER



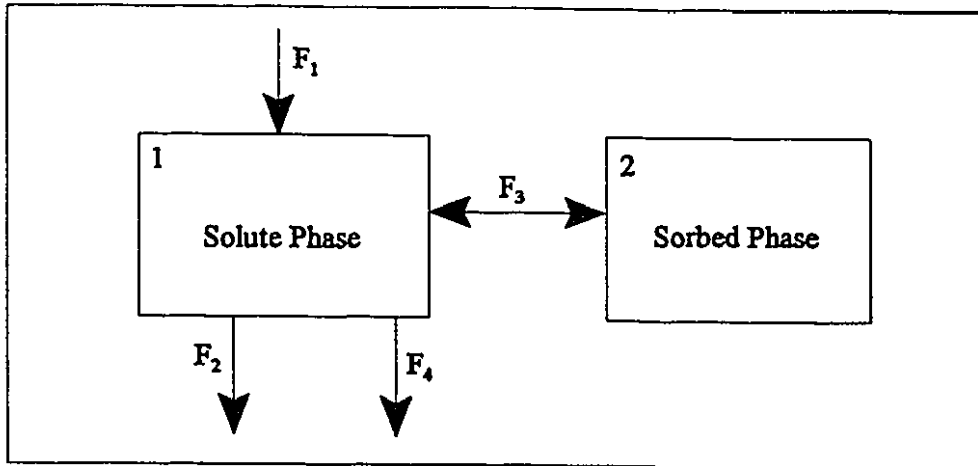
Conc. vs. Time - Site 3
EQLINEAR-1STORDER



Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER 5:

Model Name: KNLINER/1STORDER Model Location: c:\spencer\3dmodel\models\knliner-1st\speed4.usr



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
= $f(\text{porosity, velocity, dispersion coefficient})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
= $f(\text{porosity, velocity, dispersion coefficient})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (linear kinetic)
= $f(kd, \text{sorbrate})$

Number of parameters = 2

Number of operations = 3

F_4 = degradation (1st order)
= $f(\text{degradation constant (liquid)})$

Number of parameters = 1

Number of operations = 1

Flow Equations for State Variable 2:

F_3 = sorption/desorption (linear kinetic)
= $f(kd, \text{sorbrate})$

Number of parameters = 2

Number of operations = 3

Dates: Start: _____ End: _____

Sens data located: _____

Parameter values located: _____

Parameter Distribution located: _____

COMPLEXITY: $I_{kc} = (2*4 + 2*4 + 2*3 + 1*1) + (2*3) = 29$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), kd, muliq1, sorbrate

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr - for testing to speed this model up      !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

macro difmacro
  macro relabel 1888
    procedural (dif=dif,iiter,site,dat)
    dif = 0
    do 1888 q=1,5
      if (dat((q+1),iiter).gt.0.0) then
        dif=dif+((abs(dat((q+1),iiter)-site(q))**2.0)*w(q))
      end if
    1888..continue
  end
macro end

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

!macro optprint
!  macro relabel aaa,baa,caa,daa,eea
!  if (optwarnings) then

!    open (unit=1,file='sens.dat',status='old',access='append')
!    write (1,*)
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
!    close (unit=1)
!    write (*,*) 'sens.dat written.'

!    if (termflag.eq. 'delta-p') print baa
!    baa..format(/,' Termination due to small change in parameters')
!    if (termflag.eq. 'delta-f') print caa
!    caa..format(/,' Termination due to small change in objective')
!    if (termflag.eq. 'loopleft') print daa
!    daa..format(/,' Termination caused by reaching maximum
iterations')
!    if (termflag.eq. 'OK') print eaa
!    eaa..format(/,' Termination due to reaching specified
objective')
!    if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
!    faa..format(/,' Termination caused by interrupt')
!    if (termflag.ne.'break'.and.termflag.ne.'brunbreak') then
!      do 101 optii=1,ndimm
!      print aaa,optii,optp(ilo,optii)
!      aaa..format(/,' Parameter 'il,' is ',f15.7)
!      101..continue
!    endif
!  endif
!macro end

macro soilprop
  watervolume=cellvolume*porosity           !mL
  soilvolume=cellvolume*(1.0-porosity)      !mL

```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz (maxz,maxx)                                     !g
    ! dimension showyz (maxz,maxy)                                     !g
    dimension sorbedamt (maxx,maxy,maxz)                              !g
    ! print*, 'iiter,t,dif'
    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar           !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)             !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
    icellmass(3,1,1)=0.367                            !g
macro end

macro calculatetransport
    !
    procedural (mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate,&
        ,
        watervolume,ratecon,xflow,yflow,zflow)
    mflow=0

```

```

!      knlinear/1storder
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
      sorbedamt(..,y,z)=sorbrate*(kd*cellmass(x,y,z)-
sorbmass(x,y,z))
!degradation
      dcellmass(x,y,z)=-muliq1*cellmass(x,y,z)-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
      xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
      ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
      ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
      xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
      mflow=mflow+xtrans
      !g/hr
      else
      xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
      !g/hr
      dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
      ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
      mflow=mflow+ztrans
      !g/hr
      else
      ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
      !g/hr
      dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
      if (y.eq.maxy) then
      ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
      mflow=mflow+ytrans
      !g/hr
      else
      ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
      !g/hr
      dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
      endif
    else
      if (y.eq.1) then
        ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
        !g/hr
        mflow=mflow+ytrans
      !g/hr
      else
        ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
        !g/hr
        dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
        !g/hr
      endif
    endif

! This section calculates the derivatives for the liquid
! and sorbed masses.

dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans

!g/hr
dsorbmass(x,y,z)=sorbedamt(x,y,z)

!g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
  limintvfv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
  limintvfv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
  balance=integ(mflow,0.0)
  balance2=balance
macro end

macro mapviews
  site(1)=(cellmass(5,1,2)/watervolume)*1e9      !ug/L
  site(2)=(cellmass(5,2,3)/watervolume)*1e9      !ug/L
  site(3)=(cellmass(6,1,2)/watervolume)*1e9      !ug/L
  site(4)=(cellmass(17,9,3)/watervolume)*1e9     !ug/L
  site(5)=(cellmass(17,9,4)/watervolume)*1e9     !ug/L
! site(6)=(cellmass(20,9,4)/watervolume)*1e9     !ug/L
! site(7)=(cellmass(20,9,3)/watervolume)*1e9     !ug/L

```

```

macro end
!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discreted plus code to be executed every communication interval)
!Don't put macro definitions here

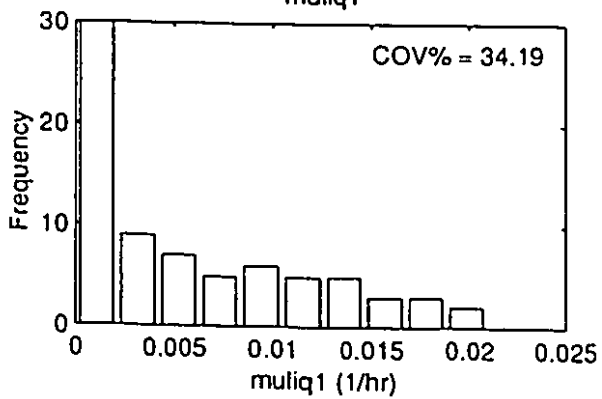
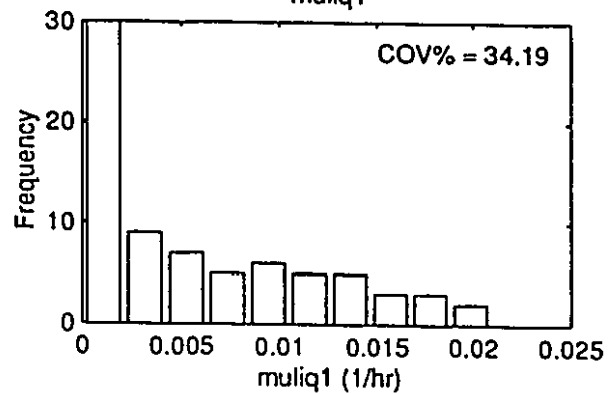
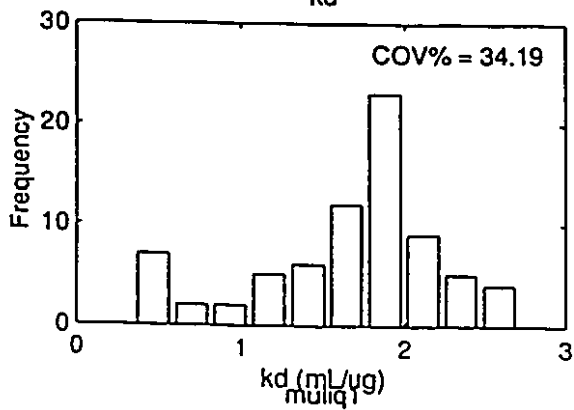
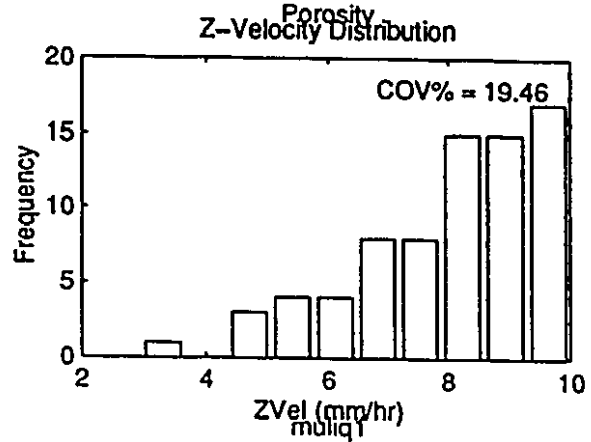
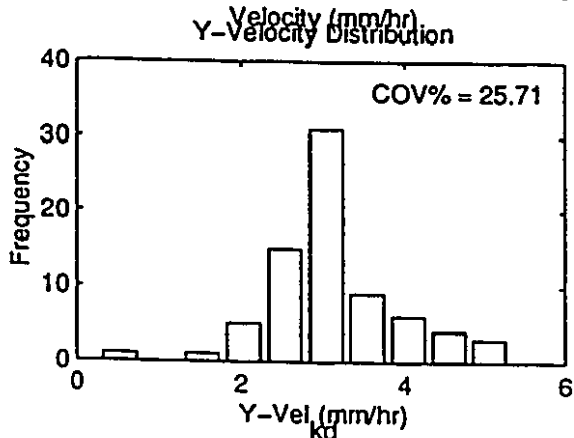
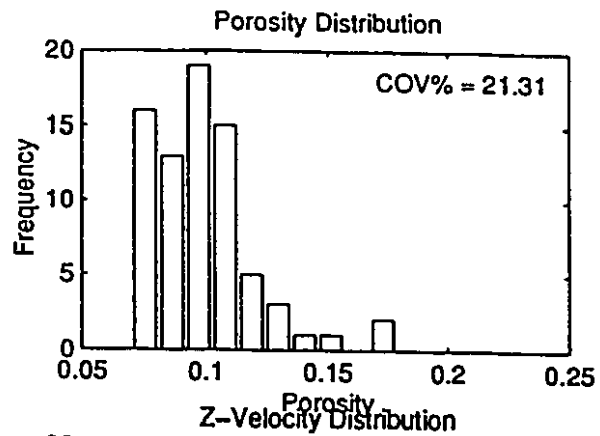
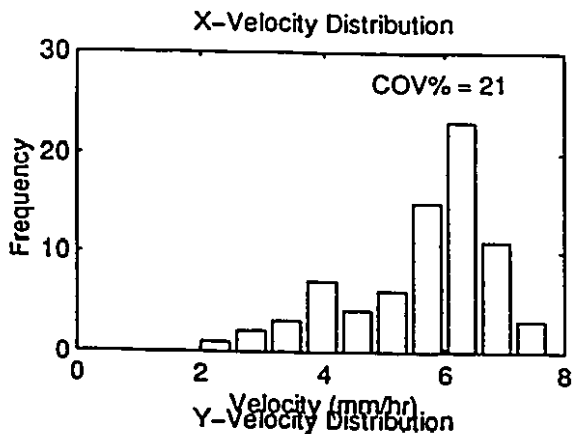
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

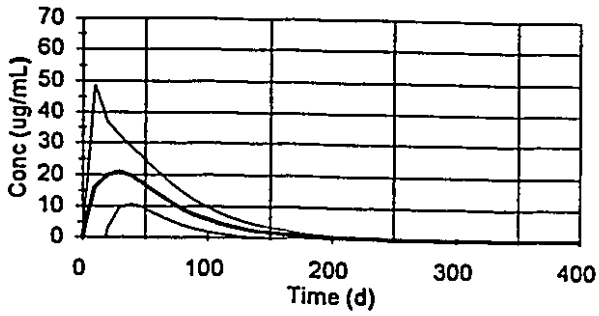
open (unit=1,file='sens.dat',status='old')
write (1,*)
porosity,xvelocity,yvelocity,zvelocity,optdif(1),kd,muliq1,sorbrate
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

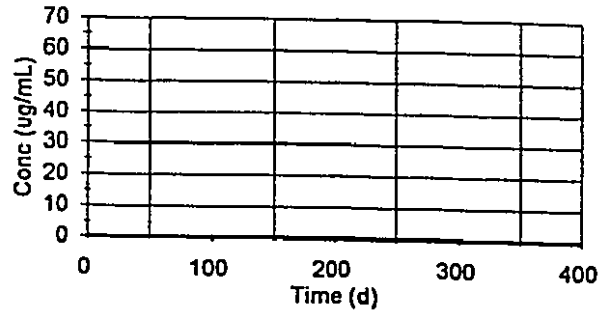
```



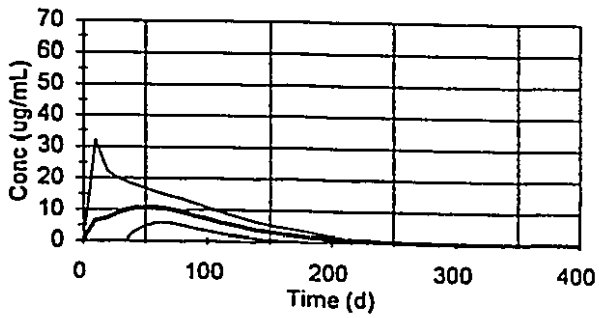
Conc. vs. Time - Site 1
KNLINEAR-1STORDER



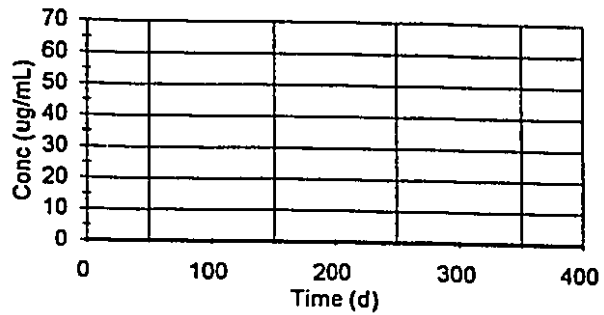
Conc. vs. Time - Site 4
KNLINEAR-1STORDER



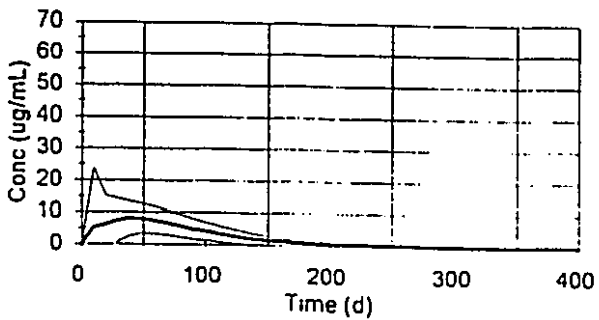
Conc. vs. Time - Site 2
KNLINEAR-1STORDER



Conc. vs. Time - Site 5
KNLINEAR-1STORDER



Conc. vs. Time - Site 3
KNLINEAR-1STORDER

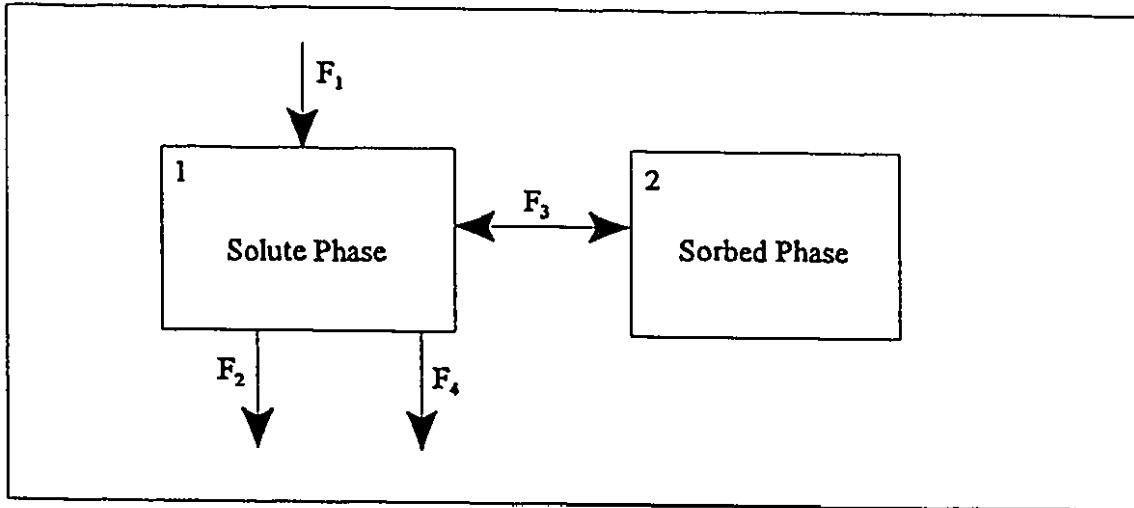


Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER 6:

Model Name: KN-NONLIN/ISTORDER

Model Location: c:\spencer3model\models\knn-lin1st



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (non-linear kinetic)
= $f(kd, nl, \text{sorbrate})$

Number of parameters = 3

Number of operations = 4

F_4 = degradation (1st order)
= $f(\text{degradation constant (liquid)})$

Number of parameters = 1

Number of operations = 1

Flow Equations for State Variable 2:

F_3 = sorption/desorption (non-linear kinetic)
= $f(kd, nl, \text{sorbrate})$

Number of parameters = 3

Number of operations = 4

Dates: Start: _____ End: _____

Sens data located: _____

Parameter values located: _____

Parameter Distribution located: _____

COMPLEXITY: $I_{kc} = (2*4 + 2*4 + 3*4 + 1*1) + (3*4) = 41$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), kd, muliq1, sorbrate, nl

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr - for testing to speed this model up      !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```

macro difmacro
  macro relabel 1888
  procedural(dif=dif,iiter,site,dat)
  dif = 0
  do 1888 q=1,5
    if (dat((q+1),iiter).gt.0.0) then
      dif=dif+((abs(dat((q+1),iiter)-site(q))**2.0)*w(q))
    end if
  1888..continue
end
macro end

```

```

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

```

```

!macro optprint
!  macro relabel aaa,baa,caa,daa,eea
!  if (optwarnings) then

!    open (unit=1,file='sens.dat',status='old',access='append')
!    write (1,*)
porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
!    close (unit=1)
!    write (*,*) 'sens.dat written.'

!    if (termflag.eq. 'delta-p') print baa
!    baa..format(/,' Termination due to small change in parameters')
!    if (termflag.eq. 'delta-f') print caa
!    caa..format(/,' Termination due to small change in objective')
!    if (termflag.eq. 'loopleft') print daa
!    daa..format(/,' Termination caused by reaching maximum
iterations')
!    if (termflag.eq. 'OK') print eaa
!    eaa..format(/,' Termination due to reaching specified
objective')
!    if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
!    faa..format(/,' Termination caused by interrupt')
!    if(termflag.ne.'break'.and.termflag.ne.'brunbreak') then
!      do 101 optii=1,ndimm
!        print aaa,optii,optp(ilo,optii)
!        aaa..format(/,' Parameter 'il,' is ',f15.7)
!      101..continue
!    endif
!  endif
!macro end

```

```

macro soilprop
  watervolume=cellvolume*porosity           !mL
  soilvolume=cellvolume*(1.0-porosity)      !mL

```



```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    ! dimension showxz(maxz,maxx)                                     !g
    ! dimension showyz(maxz,maxy)                                     !g
    dimension sorbedamt(maxx,maxy,maxz)                                !g
    ! print*, 'iiter,t,dif'
    ! This section is used to weight the sites equally for the
    ! calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar           !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)              !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0                !g
                isorbmass(i,j,k)=0.0                !g
            30.. continue
        20.. continue
    10.. continue
        icellmass(3,1,1)=0.367                        !g
macro end

macro calculatetransport
    !
    procedural (mflow,dcellmass,dsorbmass=cellmass,kd,sorbmass,sorbrate,&
                watervolume,ratecon,xflow,yflow,zflow)
    !
    mflow=0

```

```

!      kn-nonlinear/1storder
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
          sorbedamt(x,y,z)=sorbrate*(kd*cellmass(x,y,z)**nl-
sorbmass(x,y,z))
!degradation
          dcellmass(x,y,z)=-muliq1*cellmass(x,y,z)-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
          xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
          ytrans=cellmass(x,y,z)/watervolume*abs(yflow) !g/hr
          ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
          xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+xtrans
!g/hr
      else
          xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
!g/hr
          dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
          ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+ztrans
!g/hr
      else
          ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
!g/hr
          dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
          if (y.eq.maxy) then
              ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
              mflow=mflow+ytrans
!g/hr
          else
              ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
!g/hr
              dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
        endif
    else
        if (y.eq.1) then
            ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
            !g/hr
            mflow=mflow+ytrans
            !g/hr
        else
            ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
            !g/hr
            dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
            !g/hr
            endif
        endif
    endif

!      This section calculates the derivatives for the liquid
!      and sorbed masses.

    dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans

        !g/hr
        dsorbmass(x,y,z)=sorbedamt(x,y,z)

        !g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
    limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
    limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
    balance=integ(mflow,0.0)
    balance2=balance
macro end

macro mapviews

    site(1)=(cellmass(5,1,2)/watervolume)*1e9           !ug/L
    site(2)=(cellmass(5,2,3)/watervolume)*1e9           !ug/L
    site(3)=(cellmass(6,1,2)/watervolume)*1e9           !ug/L
    site(4)=(cellmass(17,9,3)/watervolume)*1e9          !ug/L
    site(5)=(cellmass(17,9,4)/watervolume)*1e9          !ug/L
!    site(6)=(cellmass(20,9,4)/watervolume)*1e9         !ug/L
!    site(7)=(cellmass(20,9,3)/watervolume)*1e9         !ug/L

```

```

macro end
!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretizes plus code to be executed every communication interval)
!Don't put macro definitions here

mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

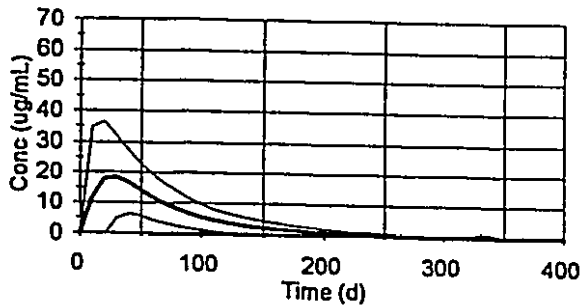
open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1),kd
write (1,*) muliql,sorbrate,nl
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

```

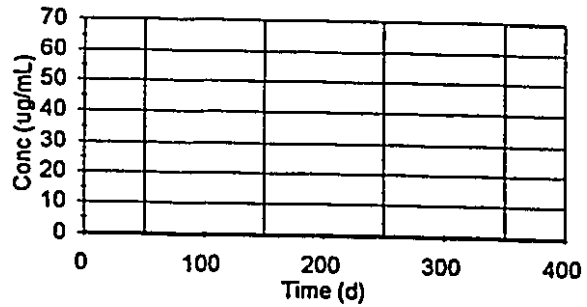
Conc. vs. Time - Site 1

KNNONLINEAR-1STORDER



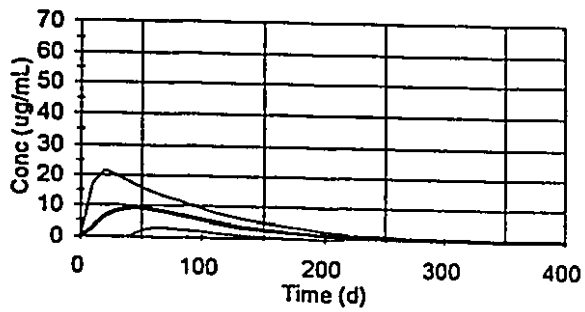
Conc. vs. Time - Site 4

KNNONLINEAR-1STORDER



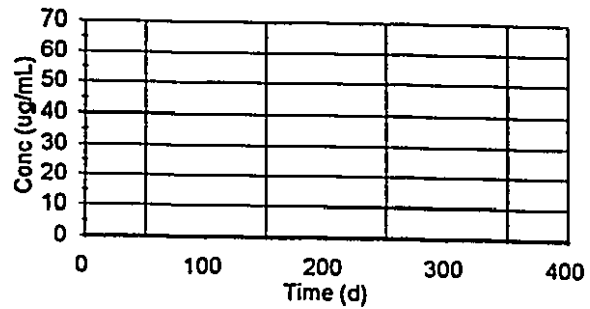
Conc. vs. Time - Site 2

KNNONLINEAR-1STORDER



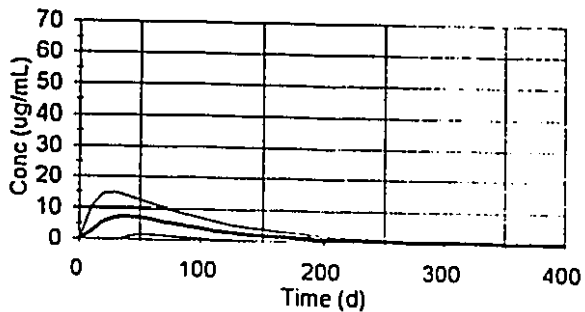
Conc. vs. Time - Site 5

KNNONLINEAR-1STORDER



Conc. vs. Time - Site 3

KNNONLINEAR-1STORDER

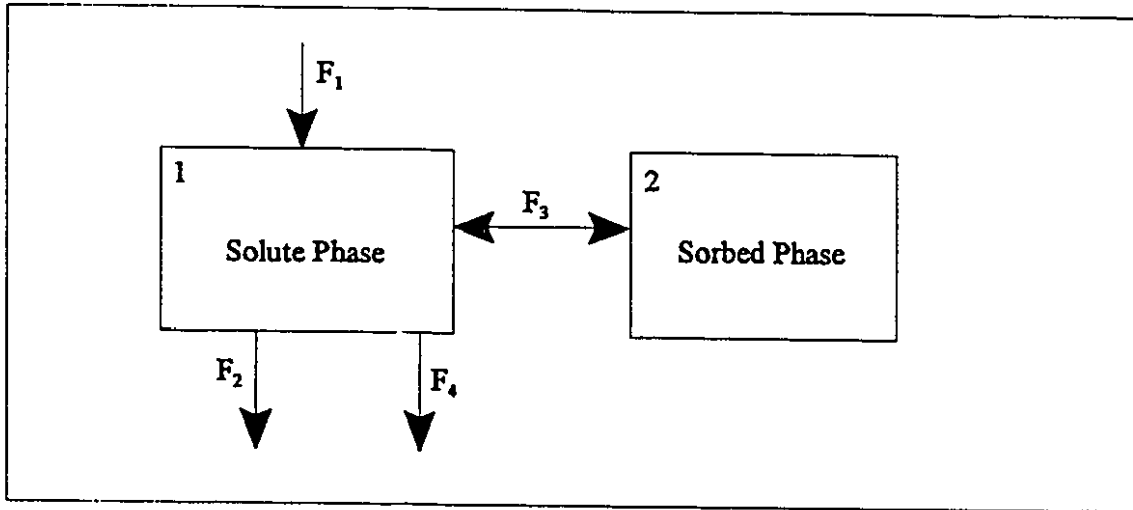


Uncertainty/Complexity Study - Sensitivity Analysis

MODEL NUMBER 7:

Model Name: KNMONOD/ISTORDER

Model Location: c:\spencer3model\models\knmonod-1st



Flow Equations for State Variable 1:

F_1 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_2 = convective flow + dispersive flow
= $f(\text{porosity, velocity})$

Number of parameters = 2

Number of operations = 2 (convective) + 2 (dispersive) = 4

F_3 = sorption/desorption (monod kinetic)
= $f(\text{monoda, monodb, sorbrate})$

Number of parameters = 3

Number of operations = 5

F_4 = degradation (1st order)
= $f(\text{degradation constant (liquid)})$

Number of parameters = 1

Number of operations = 1

Flow Equations for State Variable 2:

F_3 = sorption/desorption (monod kinetic)
= $f(\text{monoda, monodb, sorbrate})$

Number of parameters = 3

Number of operations = 5

Dates: Start: _____ End: _____

Sens data located: _____

Parameter values located: _____

Parameter Distribution located: _____

COMPLEXITY: $I_{kc} = (2*4 + 2*4 + 3*5 + 1*1) + (3*5) = 47$

Output Parameters: porosity, xvel, yvel, zvel, optdif(1), muliq1, sorbrate, monoda, monodb

```

!*****
!PUT HERE USER DEFINED MACROS
!This section is for definitions, it will be executed
!only during loading of the program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! speed4.usr - for testing to speed this model up      !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

macro difmacro
  macro relabel 1888
    procedural(dif=dif,iiter,site,dat)
    dif = 0
    do 1888 q=1,5
      if (dat((q+1),iiter).gt.0.0) then
        dif=dif+((abs(dat((q+1),iiter)-site(q)))*2.0)*w(q)
      end if
    1888..continue
  end
macro end

!*****
! this macro prints out the parameters when the optimizer
! is terminated
!*****

!macro optprint
!  macro relabel aaa,baa,caa,daa,eea
!  if (optwarnings) then

!    open (unit=1,file='sens.dat',status='old',access='append')
!    write (1,*)
!    porosity,kd,xvelocity,yvelocity,zvelocity,iiter,optdif(1)
!    close (unit=1)
!    write (*,*) 'sens.dat written.'

!    if (termflag.eq. 'delta-p') print baa
!    baa..format(/,' Termination due to small change in parameters')
!    if (termflag.eq. 'delta-f') print caa
!    caa..format(/,' Termination due to small change in objective')
!    if (termflag.eq. 'looplimit') print daa
!    daa..format(/,' Termination caused by reaching maximum
iterations')
!    if (termflag.eq. 'OK') print eea
!    eea..format(/,' Termination due to reaching specified
objective')
!    if (termflag.eq.'break'.or.termflag.eq.'brunbreak') print faa
!    faa..format(/,' Termination caused by interrupt')
!    if(termflag.ne.'break'.and.termflag.ne.'brunbreak') then
!      do 101 optii=1,ndimm
!      print aaa,optii,optp(ilo,optii)
!      aaa..format(/,' Parameter 'il,' is ',f15.7)
!      101..continue
!    endif
!  endif
!macro end

macro soilprop
  watervolume=cellvolume*porosity           !mL
  soilvolume=cellvolume*(1.0-porosity)      !mL

```

```

        soilmass=soilvolume*gamma/1000                                !g
macro end

macro flowprop
    xflow=cellsidearea*xvelocity/1000                                !mL/hr
    yflow=cellsidearea*yvelocity/1000                                !mL/hr
    zflow=cellsidearea*zvelocity/1000                                !mL/hr
    dimension site(7)                                                !g
    dimension showxz(maxz,maxx)                                       !g
    dimension showyz(maxz,maxy)                                       !g
    dimension sorbedamt(maxx,maxy,maxz)                                !g
    print*, 'iiter,t,dif'
!
!   This section is used to weight the sites equally for the
!   calculation of the objective function.

    dimension w(5)
    w(1)= 1.0
    w(2)= 0.929
    w(3)= 1.0
    w(4)= 2.76
    w(5)= 2.6

macro end

macro preinitialization
    logical preinitial
    preinitial=.true.        !to mark that we are in the preinitial section
    tlast = maxpar           !initialize saved time
    goto preinitloc         !do one run through initial
    backfrominitial..continue
    reset("eval")
    block(save)              !backup initial values in o&var
    procedural;preinitial=.false.;end
    displaylib
macro end

macro clearcells
    tot = 0
    do 10 i=1,maxx
        do 20 j=1,maxy
            do 30 k=1,maxz
                icellmass(i,j,k)=0.0    !g
                isorbmass(i,j,k)=0.0    !g
            30.. continue
        20.. continue
    10.. continue
        icellmass(3,1,1)=0.367        !g
macro end

macro calculatetransport
!
!   procedural(mflow,dcellmass,dsorbmass=cellmass,sorbmass,sorbrate,&
!           watervolume,ratecon,xflow,yflow,zflow,monoda,monodb)
!
    mflow=0

```



```

!      kn-monod/1storder
      do xend x=1,maxx
      do yend y=1,maxy
      do zend z=1,maxz
!sorption
          term=((cellmass(x,y,z)*monoda)/(monodb+cellmass(x,y,z)))
          sorbedamt(x,y,z)=sorbrate*(term-sorbmass(x,y,z))
!degradation
          dcellmass(x,y,z)=-muliq1*cellmass(x,y,z)-sorbedamt(x,y,z)
zend..continue
yend..continue
xend..continue

      do 50 x=1,maxx
      do 60 y=1,maxy
      do 70 z=1,maxz

!
!      Convective Flow - trans is equal to the amount leaving the cell
!
          xtrans=cellmass(x,y,z)/watervolume*xflow      !g/hr
          ytrans=cellmass(x,y,z)/watervolume*abs(yflow)  !g/hr
          ztrans=cellmass(x,y,z)/watervolume*zflow      !g/hr
!
!      Dispersive Flow
!

      if (x.eq.maxx) then
          xtrans=xtrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+xtrans
          !g/hr
      else
          xtrans=xtrans+(cellmass(x,y,z)-cellmass((x+1),y,z))*ratecon
          !g/hr
          dcellmass((x+1),y,z)=dcellmass((x+1),y,z)+xtrans!g/hr
      endif

      if (z.eq.maxz) then
          ztrans=ztrans+(cellmass(x,y,z)-0)*ratecon      !g/hr
          mflow=mflow+ztrans
          !g/hr
      else
          ztrans=ztrans+(cellmass(x,y,z)-cellmass(x,y,(z+1)))*ratecon
          !g/hr
          dcellmass(x,y,(z+1))=dcellmass(x,y,(z+1))+ztrans!g/hr
      endif

      if (yflow.gt.0) then
          if (y.eq.maxy) then
              ytrans=ytrans+(cellmass(x,y,z)-0)*ratecon !g/hr
              mflow=mflow+ytrans
          !g/hr
          else
              ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
          !g/hr
              dcellmass(x,(y+1),z)=dcellmass(x,(y+1),z)+ytrans

```

```

        !g/hr
      endif
    else
      if (y.eq.1) then
        ytrans=ytrans+(cellmass(x,y,z)-
cellmass(x,(y+1),z))*ratecon
        !g/hr
          mflow=mflow+ytrans
        !g/hr
      else
        ytrans=ytrans+(cellmass(x,y,z)-cellmass(x,(y-
1),z))*ratecon
        !g/hr
          dcellmass(x,(y-1),z)=dcellmass(x,(y-1),z)+ytrans
        !g/hr
      endif
    endif

!   This section calculates the derivatives for the liquid
!   and sorbed masses.

dcellmass(x,y,z)=dcellmass(x,y,z)-xtrans-ytrans-ztrans

!g/hr
dsorbmass(x,y,z)=sorbedamt(x,y,z)

!g/hr

70..continue
60..continue
50..continue

end
macro end

macro integration
  limintvvv(cellmass,dcellmass,icellmass,maxx,maxy,maxz,0.0,1e6)
  limintvvv(sorbmass,dsorbmass,isorbmass,maxx,maxy,maxz,0.0,1e6)
macro end

macro massbalance
  balance=integ(mflow,0.0)
  balance2=balance
macro end

macro mapviews

  site(1)=(cellmass(5,1,2)/watervolume)*1e9      !ug/L
  site(2)=(cellmass(5,2,3)/watervolume)*1e9      !ug/L
  site(3)=(cellmass(6,1,2)/watervolume)*1e9      !ug/L
  site(4)=(cellmass(17,9,3)/watervolume)*1e9     !ug/L
  site(5)=(cellmass(17,9,4)/watervolume)*1e9     !ug/L
!   site(6)=(cellmass(20,9,4)/watervolume)*1e9   !ug/L
!   site(7)=(cellmass(20,9,3)/watervolume)*1e9   !ug/L

```

```

macro end
!*****
macro userinitialsection
!INITIAL SECTION
!Macros called here will be executed in the initial section
!Don't put macro definitions here

soilprop
flowprop
clearcells

macro end
!*****
macro userderivativesection
!DERIVATIVE SECTION
!Macros called here will be executed in the derivative section
!Don't put macro definitions here

calculatetransport
integration
massbalance

macro end
!*****
macro userdynamicsection
!DYNAMIC AND DISCRETE SECTIONS
!Macros called here will be executed in the dynamic section.
!(discretes plus code to be executed every communication interval)
!Don't put macro definitions here

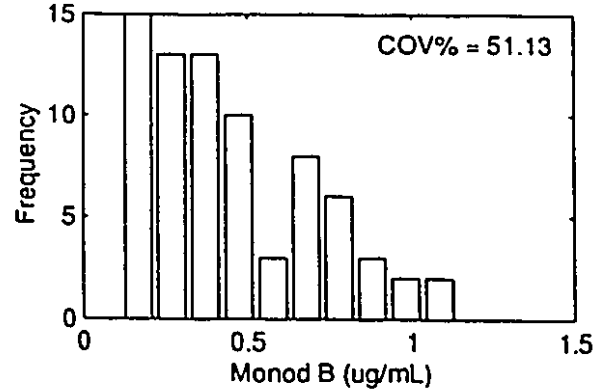
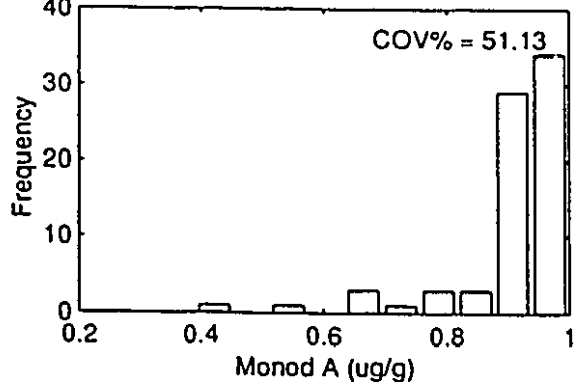
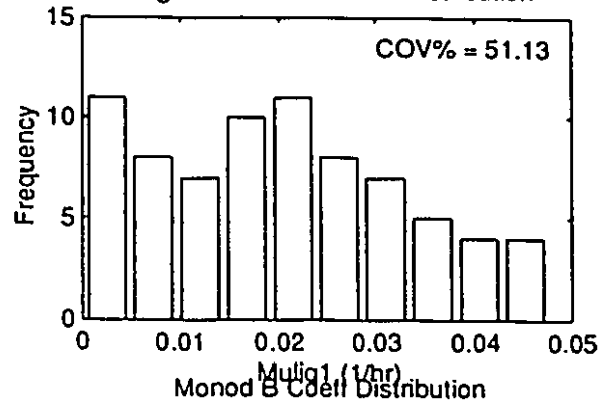
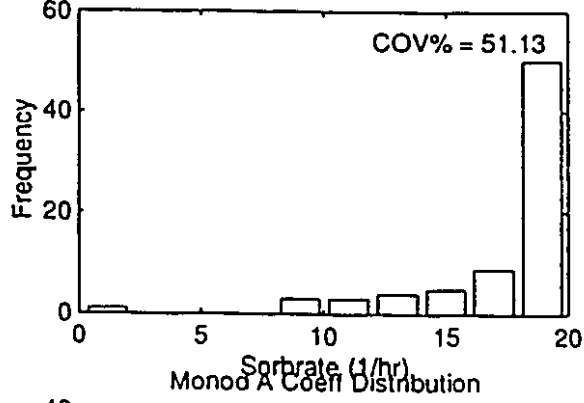
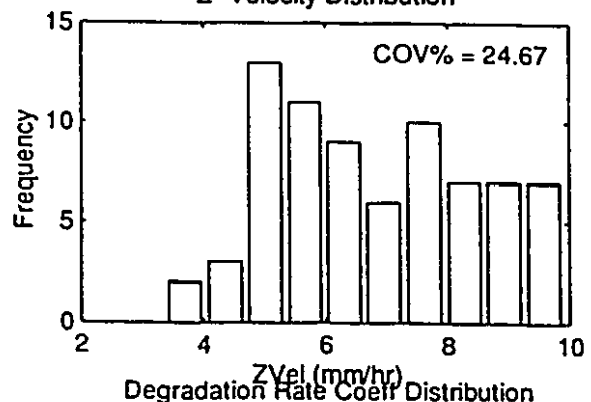
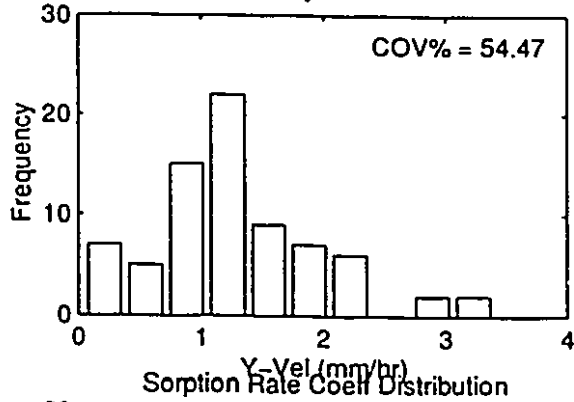
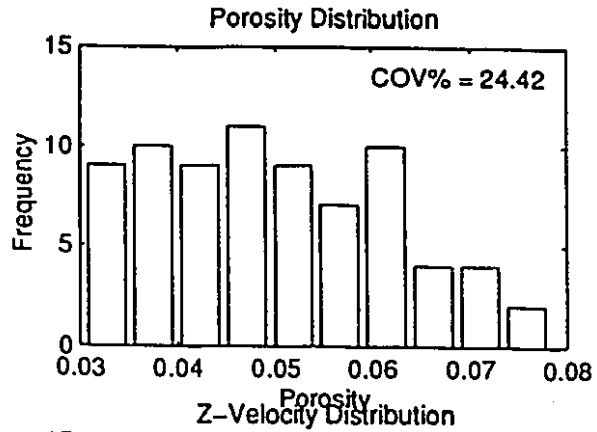
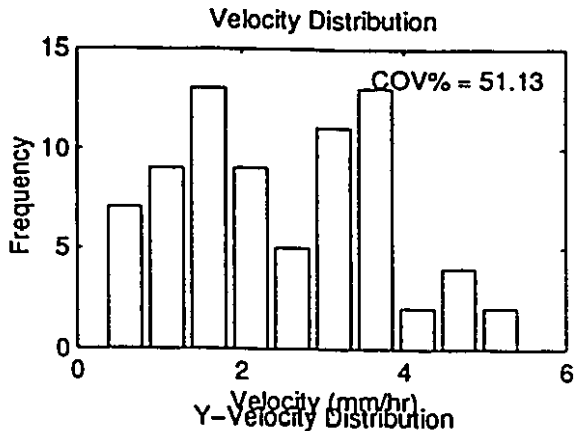
mapviews

macro end
!*****
macro userterminalsection
!TERMINAL SECTION
!Macros called here will be executed in the terminal section
!Don't put macro definitions here

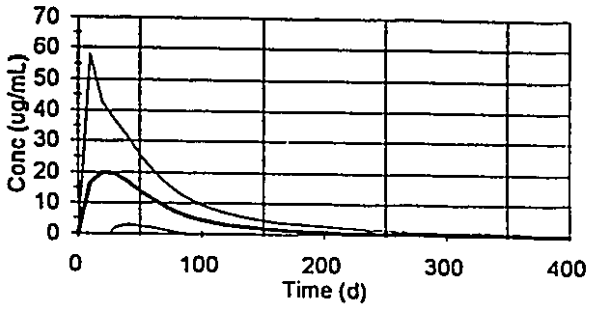
open (unit=1,file='sens.dat',status='old')
write (1,*) porosity,xvelocity,yvelocity,zvelocity,optdif(1)
write (1,*) muliql,sorbrate,monoda,monodb
close (unit=1)
write (*,*) 'sens.dat written.'

macro end
!*****

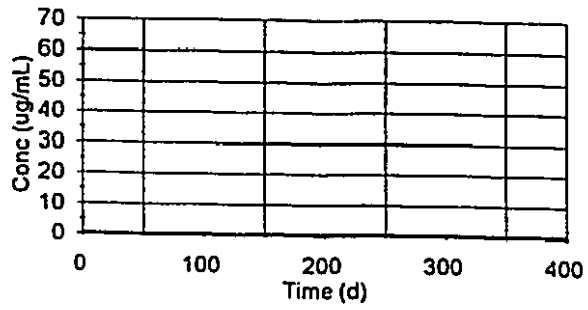
```



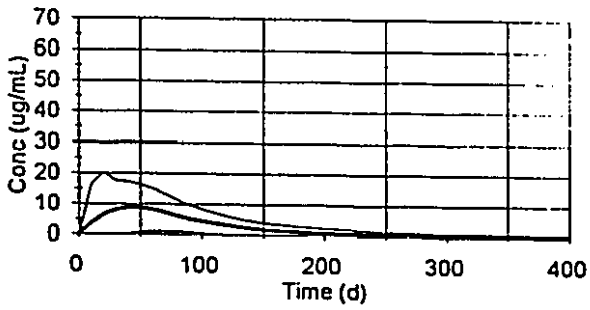
Conc. vs. Time - Site 1
KNMONOD-1STORDER



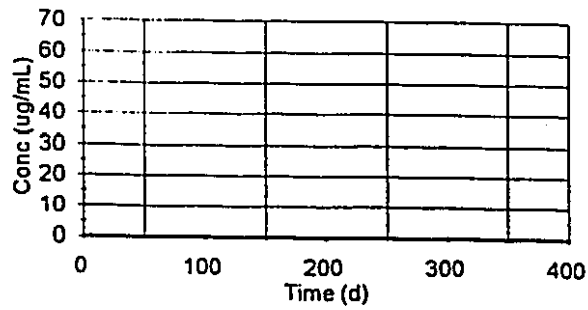
Conc. vs. Time - Site 4
KNMONOD-1STORDER



Conc. vs. Time - Site 2
KNMONOD-1STORDER



Conc. vs. Time - Site 5
KNMONOD-1STORDER



Conc. vs. Time - Site 3
KNMONOD-1STORDER

