UNCERTAINTY ANALYSIS FOR ENVIRONMENTAL MANAGEMENT MODELS: APPLICATION OF THE GENERALIZED SENSITIVITY ANALYSIS

by

ANDREWS KWABENA TAKYI

A thesis
presented to the University of Manitoba
in fulfilment of the
thesis requirement for the degree of
Master of Science
in
Civil Engineering

Winnipeg, Manitoba, Canada 1991

ISBN 0-315-76787-1

UNCERTAINTY ANALYSIS FOR ENVIRONMENTAL MANAGEMENT MODELS: APPLICATION OF THE GENERALIZED SENSITIVITY ANALYSIS

BY

ANDREWS KWABENA TAKYI

A thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

To My Dear Dad, Mom, and Grandma

I hereby declare that I am the sole author of this thesis.

I authorize the University of Manitoba to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the University of Manitoba to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

The University of Manitoba requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

ABSTRACT

An approach is developed for computing sensitivity indices for correlated input data in an environmental management model. The method modifies the Generalized Sensitivity Analysis technique for simulation models to make it applicable to analyses in which both optimization and simulation models are applied.

This approach is demonstrated for a risk equivalent seasonal BOD waste load allocation program for multiple dischargers for a case study based on the Willamette River in Oregon. The case study considers uncertainty in summer and winter temperatures and flows at five monitored stations on the river. The analysis was done for three DO standards and six different summer and winter season length combinations.

The results obtained indicate that seasonal waste management programs should identify and accommodate the uncertainty in the input data bases. The sensitivity indices obtained for the seasonal flow and temperature data were different for the different management decisions analysed. Higher DO standards produced higher sensitivity indices and the mean of the sensitivity indices of the uncertain input data for each DO standard peaked at a summer season length of 3 months. It is demonstrated that for the Willamette River, summer conditions are more important than those of winter for maintaining water quality under such programs. It is also shown that the system design is more sensitive to the three most downstream gauging stations on the river, rather than the two upstream stations.

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my supervisor, Dr. Barbara J. Lence for her financial support for my M.Sc program and for her deep interest and guidance throughout the preparation of this thesis. I would also like to thank Mike Bender, Kwame Agyare, and Hassen Zghal for their assistance. I am also grateful to Drs. D. H. Burn, G. M. Hornberger, D. S. Meek and S. P. Simonovic for their useful suggestions during various stages of this research.

Throughout this study, my wife has been a source of strength and emotional support. She also helped in typing the manuscript. Peace, for all those sleepless nights, thanks for being there.

Finally, I would like to acknowledge the assistance of all my office colleagues during the period of my M.Sc. program.

Contents

List of Tables

List of Figures

Chapter 1

INTRODUCTION

1.1 General

There has been a rapid increase in the development of mathematical models for a wide range of applications in the past three decades as a result of the large improvement in computational accuracy and time offered by computers. Advances in the field of operations research have also contributed to the sudden rise in mathematical modelling. Many of these models employ very complicated or advanced computational techniques aimed at improving the efficiency of planning and management of the systems they represent. For example, well represented models such as those for business, industry, and national defence have proved to be highly efficient and relatively reliable, and have been highly utilized. In fact, optimization models used to enhance planning and management in industry have been very successful and greatly accepted by decision makers. In two different surveys of several top corporations in the United States, Turban (1972), and Thomas and DaCosta (1979) indicated that nearly half of the companies reporting had special departments engaged mainly in operations research. Turban surveyed a sample from Fortune's 1969 list of the top

500 companies, while Thomas and DaCosta surveyed a sample from Fortune's 1975 list of the top 500 corporations, the largest 100 industrial firms in California, and the financial institutions in California.

Despite their claim to closely represent the systems they describe and their acceptance as management aids in business, industry, and national defence, mathematical models are seldom used by planners and managers in other areas of application. Models of environmental systems are in the class of complex models which planners and managers have rarely used over the years. The widespread rejection of the seemingly efficient environmental models is due in part to the uncertainty in their output. When reviewing uncertainty analysis in water quality models, Beck (1987) wrote:

> "Most of the evidence suggests that the current models of water quality, in particular, the larger models, are easily capable of generating predictions to which little confidence would be attached. Yet there have been few case studies of the truly large scale models, and the challenge that they present has not been adequately met. In this respect, much remains to be done."

This thesis develops a technique for computing an index of uncertainty associated with input data for complex environmental management models. The overall approach determines the contribution of the uncertainty in input data to the model output uncertainty. This approach is demonstrated for an innovative water quality management model for the Willamette River in Oregon.

1.2 Uncertainty in environmental systems

Uncertainty in environmental model output is caused by inexact or poor representation of the system and by incomplete or inaccurate input data. Poor definition of the

structure of an environmental model may arise from the lack of complete knowledge of the physical, chemical, and biological relationships between the output and the input data and from the existence of unmodelled, unknown, or unquantifiable components or aspects of the system. Simplifying assumptions made in the model formulation, such as the use of a one-dimensional transport model to represent a two-dimensional system, may also introduce random variance in the model output. The problem of poorly defined systems can be addressed by obtaining a better understanding of the system mechanics, avoiding unnecessary assumptions, and using widely accepted methods of modelling.

Uncertainty in input data is caused by measurement errors and by the stochastic nature of most environmental systems. Measurement errors are attributed to: variation in replicate laboratory or in-situ measurement; spatial variation at a given sampling location (e.g., flow or dissolved oxygen measurement across the width of a river or through the water column); temporal variation during the sampling period, and in the case of a river, longitudinal variation over a specified length. Most environmental models are built using limited data and lump sum reaction parameters. The actual system behaviour, therefore, is often different from what is produced by the model. The model output may result in failure to meet a management objective or in gross over-design of the management solution. Even under situations where input data quality is very good, there is still the uncertainty due to the effects of stochasticity and of possible climatic change on environmental systems.

In water quality simulation models, the sources of uncertainty include insufficient data and errors or stochasticity in water discharge and temperature measurements. There is also some uncertainty in the estimates of reaction coefficients and river velocity obtained from empirical relationships. Given the stochastic nature of these parameters and the limited available data, a management decision based on the output from a deterministic water quality simulation model could result in a high rate of

policy failure.

Water quality data collection is expensive and data is sometimes difficult or even impossible to obtain. Therefore, available data for design will often be limited and inaccurate. In such cases, it would be useful to estimate the degree of certainty of model outcome. Evaluating the sensitivity of model output to uncertain input data is one way of estimating certainty of model outcome. Traditionally, the sensitivity of environmental models to uncertain input data is computed with one of the following approaches:

1. First Order Error Analysis (FOEA),
2. Sensitivity Analysis; and
3. Monte Carlo Simulation (MCS).

Although, these techniques have been extensively used by researchers in environmental system studies (see, e.g., Voelkel, 1978; Lettenmaier and Richey, 1979; Whitehead and Young, 1979; Burn and McBean, 1985; Uber et al., 1985; and Song and Brown, 1990), they have some weaknesses. Detailed descriptions of these approaches and the advantages, disadvantages, and limitations of them are given in Chapter 3. Spear and Hornberger (see, e.g., Spear and Hornberger, 1980; and Hornberger and Spear, 1981), have developed and applied a fourth technique for uncertainty analysis for complex environmental simulation models called the Generalized Sensitivity Analysis (GSA). This thesis develops a modified version of the basic GSA technique which may be applied to environmental management problems. GSA gives quantitative estimates of the sensitivity of model output to input data and parameter uncertainty, which, in addition to estimates of costs and benefits, is useful information for the environmental manager in assessing alternative management policies.

1.3 Innovative water quality management models

In many aspects of water pollution control, public concern and consequently public policy have demanded strict, often conservative, or unrealistic management strategies. In an effort to reduce the social cost of waste management, researchers have attempted to design water quality management policies that reduce pollution and minimize the total cost of waste treatment. Traditionally, non-seasonal water quality management programs have been designed to assist decision makers in managing quality in river systems. A non-seasonal program is a management plan which is the same for all periods in a year and is based on annual critical river conditions. These programs do not take advantage of the increased assimilative capacity of a river at certain times of the year.

Recent work in water quality management has focussed on seasonal waste treatment programs that reduce pollution abatement costs further by taking advantage of the dynamic nature of the assimilative capacity of a stream. Under these programs, different waste removal levels are required in different seasons of the year based on the assimilative capacity of the stream in those seasons. Some of the innovative management programs are, Seasonal Transferable Discharge Permit, Seasonal Least Cost, and Seasonal Uniform Treatment Programs (see, e.g., Hargett and Seagraves, 1979; O'Neil, 1980; Reheis et al., 1982; Boner and Furland, 1982; Ferrara and Dimino, 1985; Kilgore, 1985; Eheart et al., 1987; Rossman, 1989; Lence et al., 1990). These management models have demonstrated that seasonal programs achieve substantial cost savings relative to non-seasonal programs in many river basins. There has however, been concern about the ability of these innovative models to achieve water quality levels comparable to the traditional non-seasonal ones.

Seasonal risk equivalent models (Rossman, 1989) are designed to give a risk of water quality violation which is equivalent or comparable to the non-seasonal pro-

grams in addition to achieving significant cost efficiency in waste treatment. One such program which is designed for river basins with several dischargers is the Minimum Average Uniform Treatment Risk Equivalent Seasonal Discharge Program (Lence et al., 1990). The modified GSA approach which is developed in this thesis is demonstrated using a case study based on the risk equivalent seasonal program designed by Lence et al. (1990).

1.4 Research objectives and scope of work

This thesis presents an approach for analysing the effects of uncertainty in input data on the uncertainty in the predictions of environmental management models. The work has the following objectives:

1. develop a modified GSA technique for correlated input data,
2. develop an approach for using the GSA in a management context, and
3. demonstrate the approaches developed in objectives (1) and (2) with a seasonal Biochemical Oxygen Demand (BOD) management program for the Willamette River Basin in Oregon.

The contributions of the uncertainty in the individual input data to model output uncertainty is computed in the form of sensitivity indices for seasonal BOD management models. These indices are analysed for systems under different management assumptions, such as different water quality goals and different season length combinations. The sensitivity indices obtained in this analysis could help improve decision making for the control of water pollution in the Willamette River Basin by identifying the input data that are important to the model output.

The literature in the two areas of uncertainty analysis in environmental systems and innovative water quality management models is reviewed in Chapter 2. Chapter 3 gives brief explanations, advantages, and disadvantages of traditional methods of uncertainty analysis for environmental models. The GSA technique for environmental simulation models is presented in Chapter 4. Next, the modified version of this technique which was developed in this research is presented. This technique is applicable to environmental management models which are based on optimization and simulation and which have correlated input data. In Chapter 5, the Minimum Average Uniform Treatment Risk Equivalent Seasonal Discharge Program for the Willamette River Basin is discussed and used as a case study for the methodology developed in Chapter 4. Results obtained for this case study are discussed in Chapter 6, and a summary and conclusions are given in Chapter 7.

He concludes that although much research has been conducted on the analysis of uncertainty in the development of mathematical models for water quality, the problem has not been totally addressed. He also discusses the future directions of uncertainty analysis, and speculates on the development of novel techniques such as artificial intelligence in knowledge representation and manipulation for use in model structure identification and model output prediction.

Most of the studies on uncertainty in water quality models utilize First Order Error Analysis (FOEA), Sensitivity Analysis, or Monte-Carlo Simulations. FOEA estimates model output mean and variance from input parameter means and variances. The model output variance is then used as an index of model output uncertainty. Sensitivity Analysis denotes the contribution of an uncertain input parameter to the uncertainty in the model prediction as the rate at which the model output changes with this uncertain parameter. Monte-Carlo Simulation develops statistical properties of the model output, which are used to characterize the uncertainty in the model response, by randomly generating a number of model realizations from selected input data. The random input data are selected from their fitted probability density functions.

FOEA was first proposed for use with water quality models by Burges and Lettenmaier (1975). They apply FOEA to a stream dissolved oxygen-biochemical oxygen demand (DO-BOD) model. Since then, it has been widely used by many researchers in modelling water quality for different types of water bodies and different pollutants (Lettenmaier and Richey, 1979; Reckhow, 1979; Scavia, 1980; Scavia et al., 1981a, 1981b; Beck, 1981, 1983; Chadderton et al., 1982; Walker, 1982; van Straten, 1983; and Malone et al. 1984).

FOEA is applied to an optimization model by Berthouex and Polkowski (1970). They predict system output uncertainty for an optimization model of an activated sludge waste water treatment plant, as a result of input parameter uncertainty using

the method of variance propagation (FOEA). A 7% to 12% increase in total cost is obtained when uncertainty is included in the design. Burn and McBean (1985) employ chance constraint programming to optimize a water quality planning model and characterize the uncertainty inherent in the random inputs using FOEA.

Sensitivity Analysis has been widely used for uncertainty studies in wastewater treatment plant design (see, e.g., McBeath and Eliassen, 1966; Chen et al., 1970; Tarrer et al., 1976; Tyteca, 1981). Chang (1967) develops sensitivity equations for models that could be represented by sets of algebraic, differential, and difference equations using the mathematical definition of sensitivity. To illustrate the use of the sensitivity equations developed, he applies them to a model of a simple chemical reactor which has an uncertain kinetic rate coefficient. His work also includes a study of the design of robust systems by incorporating sensitivity constraints in an optimization model of the reactor.

Tarrer et al. (1976) expand the optimization model of the activated sludge wastewater treatment plant developed by Berthouex and Polkowkis (1970). They perform sensitivity and perturbation analysis on the expanded model and indicate that sensitivity analysis is useful in determining which direction input parameters should be perturbed to obtain a robust design.

Voelkel (1978) uses a type of sensitivity analysis called perturbation analysis to calculate sensitivity coefficients for a waste water treatment plant model. He ranks the importance of the perturbed parameters by the magnitudes of their sensitivity coefficients, the degree of nonlinearity of the model output with respect to the uncertain input parameters, and the number of nonfeasible points encountered during the analysis.

Uber and Brill (1990) develop a general sensitivity constrained nonlinear programming (SCNLP) technique for nonlinear optimization models. This technique

optimizes a collection of sensitivity based design criteria including a first order estimate of variance, reliability, and robustness of the system performance. The authors observe that the solutions obtained with SCNLP are only approximations to system performance uncertainty but could be used for (a) generating alternative solutions, (b) exposing the tradeoff between system output and uncertainty which is not obvious in complex systems, and (c) gaining more insight into the model performance. The technique is applied to designing a simple hypothetical heat exchanger in which the objective is to minimize the effluent temperature sensitivity subject to a constraint on the total heat exchanger area. The results indicate that minimizing the sensitivity of the heat exchanger design is significantly different from the traditional design approach of minimizing heat exchanger area. An application of the SCNLP approach to the design of wastewater treatment plant is also discussed.

The SCNLP technique is subsequently applied to the design of a typical wastewater treatment plant (Uber et al., 1991b). Uber et al. (1991c) then apply this approach to the design of a comprehensive, activated sludge treatment plant (the non-linear minimum cost wastewater treatment plant model developed by Tang et al., 1987). The robustness measure evaluated in this work considers uncertainty in 55 model parameters. The SCNLP technique is used to generate a trade-off curve between total system cost and robustness. The results obtained are consistent with typical designs developed in practice. For example, in order to obtain a favourable robustness for the variance in effluent BOD, longer activated sludge mean solids residence time is required in contrast to the short solids residence times which are typically chosen in least cost designs.

Two of the many advantages offered by computers are speed and accuracy in mathematical computations. These two features have made MCS the most popular technique for uncertainty analysis in environmental systems (see, e.g., Argentesi and Olivi, 1976; Lettenmaier and Richey, 1979; DiToro and van Straten, 1979; Beck et al.

1979; Gardner et al. 1980; Fedra, 1983; Gardner and O'Neill, 1983).

Whitehead and Young (1979) use an MCS to derive probability distributions for daily BOD and DO levels at two points in the Bedford Ouse River in Central England. The sensitivities of the resultant DO and BOD values at selected locations to input parameter uncertainty are assessed by comparing the probability distributions for stochastic simulations of the water quality with those observed at corresponding locations. It is shown that the output is insensitive to input parameter uncertainty when the latter is estimated using the multivariable Instrumental Variable-Approximate Maximum Likelihood (IVAML) time series algorithm. However, results from simulations with a 10% increase in the reaeration coefficient show a positive shift in the cumulative probability distribution of DO of approximately 1.5 mg/l compared to the observed distribution. The authors further show that the BOD and DO at two downstream sites, Olney and Clapham, are sensitive to the effluent discharge rate at Milton Keynes, which is about 55 km upstream of Clapham.

Results of applications of First Order Error Analysis for uncertainty analysis in environmental systems have very often been compared with those from MCS to evaluate their efficacy (e.g., Burges and Lettenmaier, 1975; Hornberger, 1980; Walker, 1982; Scavia et al., 1981b; Song and Brown, 1990). Song and Brown (1990) compare results of an FOEA and MCS analysis for a modified Streeter-Phelps dissolved oxygen model. They conclude that results from these two uncertainty analysis methods compare favourably with each other. The mean values of the predicted DO deficit for the two techniques are not significantly different at short travel times, and the largest difference between the standard deviations is about 25%.

Correlation among input parameters has a significant effect on model output. In fact, if the correlation structure among uncertain input parameters or data and their effects are ignored, results obtained from uncertainty analysis may grossly underestimate the actual uncertainty in the model. Song and Brown (1990) also investigate

the effects of input data uncertainty on model output uncertainty using FOEA, Sensitivity Analysis, and Monte-Carlo Simulations. The authors perform uncertainty analysis for both correlated and uncorrelated input parameters for a hypothetical river. It is shown that the standard deviation of the predicted DO deficit is 20-40% larger for correlated inputs than it is for uncorrelated inputs. Although only a general estimate of the correlation among the input parameters is used, the results give a clear indication of the effects of correlated uncertain input data on model output uncertainty.

Ponnambalam and McBean (1990) use a chance constrained programming model of a hypothetical acid rain abatement program to show that the optimal program cost (i.e., the optimization model output) is a non-linear function of the significance level for the chance constraint, and the variances and covariances among the uncertain input transfer coefficients. With this example, they show that correlation among input parameters should not be ignored in uncertainty analysis.

Spear and Hornberger (1980) develop an innovative approach for the analysis of uncertainty in environmental simulation systems called the Generalized Sensitivity Analysis (GSA). They use the GSA to compute the indices of sensitivity of observed system behaviour to uncertain input parameters. The sensitivity indices obtained are used to identify critical uncertainties in a phosphorus based model for eutrophication in the Peel Inlet in Western Australia. The method utilizes MCS, a classification algorithm, and a statistical analysis. The MCS is used to simulate *Cladophora* and phosphorus concentrations given uncertainty in 19 input parameters. A qualitative binary classification definition is used to categorize each simulation output and its associated uncertain input parameter set. The classification algorithm determines whether or not a particular Monte-Carlo Simulation output mimics the known system behaviour (e.g., actual concentrations from sampling programs). Simulation results that resemble the known system behaviour are called a behaviour and the others

are called a non-behaviour. The parameter sets are then classified as belonging to a behaviour or a non-behaviour class according to the outcome of their corresponding simulated model response. The Kolmogorov-Smirnov (K-S) goodness of fit test is then used to test for the separation between the cumulative distribution functions of the behaviour and non-behaviour classes for each uncertain parameter. The authors assume zero induced covariance among the uncertain input parameters since there is no significant correlation among them. They conclude that a parameter is important in simulating results in either of the two classes if the cumulative distribution functions of that parameter for the behaviour and non-behaviour groups are significantly different. The K-S sample statistic which measures the separation between the two behavioural data groups for each parameter is used as a sensitivity index to rank the importance of that uncertain input parameter in the model. Based on the results from the GSA, Spear and Hornberger (1980) conclude that seven of the 19 uncertain parameters in the eutrophication simulation model are not important for defining the binary classification. They suggest that future research areas for the Peel Inlet study should include investigation of mechanisms of deposition, release, and remineralization of nutrients in the sediment, and of certain aspects of the physiology of *Cladophora*.

The GSA technique for simulation models is restated and explained by Hornberger and Spear (1981). They present further analysis required when the assumption of zero induced covariance (i.e., no correlation) among input data is not valid. The authors use a principal components transformation to remove any correlation among uncertain input data before proceeding with the analysis of the separation of the transformed parameters. However, they do not develop any sensitivity indices for the original uncertain parameters.

The extended GSA technique with induced covariance among uncertain input parameters is illustrated for a simple hypothetical problem with a three-dimensional

parameter space. The hypothetical example has significant correlation between two of the three parameters. However, a univariate analysis (i.e., an analysis based on the original untransformed parameters) indicates that only the uncorrelated parameter is significant at the 99% confidence level. Subsequent use of a multivariate analysis (principal components analysis) shows that the other two parameters are also important in simulating a behaviour or a non-behaviour. This technique is also applied to the phosphorus model for the eutrophication of Peel Inlet. The authors conclude that because of the small induced covariance among the 19 input parameters, much of the discriminating information between the behaviour and non-behaviour groups for the Peel Inlet problem is concentrated in mean shift, making univariate analysis adequate.

Humphries et al. (1984) apply the GSA technique to a nitrogen based model of the eutrophication in the Peel Inlet. Their work is a preliminary analysis to identify processes in the nitrogen cycle that are important for simulating the eutrophic behaviour of the Peel Inlet, and consequently to plan research studies. It is shown that nitrogen limitation of *Cladophora* growth rarely occurs. Results obtained from this study and from field and laboratory data collected after 1978 confirm conclusions from an earlier study of the phosphorus based model (Hornberger and Spear, 1981; Spear and Hornberger, 1980), hence indicating the appropriateness of the GSA technique.

Spear and Hornberger (1983) use the GSA to select a set of design parameters for a waste treatment lagoon which discharges into a river. The parameters selected maximize the probability of maintenance of the DO level in the river. Spear and Hornberger adopt a single reach DO model for the River Cam (developed by Young and Beck, 1974), and define the binary classification as whether or not a specified DO criterion is met. By changing the ranges for the design parameters, the authors show that the acceptable behaviour of the controlled process can be attained with

a probability of 84%. However, they observe that significant effects on performance could result from modest levels of process parameter uncertainty, and suggest that additional attention be given to the design of robust parameters for environmental systems.

GSA is subsequently applied to a water quality and hydrological model for the Shenandoah National Park, in Virginia (Hornberger and Cosby, 1985b). It is demonstrated that (i) simulation models can be "tentatively calibrated", (ii) the data most likely to provide a stringent test of the model can be identified, and (iii) potential problems with model identifiability can be exposed in a preliminary analysis using the GSA technique.

Regional calibration of a sulfate dynamics model for the Shenandoah National Park is also performed with a GSA (Hornberger et al., 1986). The authors indicate that, when data is sparse, traditional methods of parameter estimation for environmental systems are inferior to the GSA approach.

2.2 Innovative water quality management models

Traditionally, waste discharge management programs have been designed using the same design conditions for the whole year. These programs, referred to as non-seasonal programs, use some annual critical low flow (e.g., the 7-day average low flow which has a return period of 10 years), temperature (e.g., the highest temperature on record), and background condition (e.g., typical non-point source pollution contributions) as input data for the design of the management model.

There has been a recent move by researchers to design seasonal or dynamic waste discharge management programs that utilize changing stream assimilative capacity and achieve environmental quality at least cost. For example, in spring when flows

are high and temperatures and decay rates are low, streams have the capacity to assimilate a larger amount of BOD waste than they do in summer months. Designing different waste removal levels for the different seasons generally results in higher total waste discharge and possible cost savings compared to those that result under non-seasonal waste discharge programs.

The management strategies that have been studied by water quality researchers can be classified as direct regulation or incentive based. The former consists of uniform or zoned uniform treatment and least cost programs while the latter include programs that employ subsidies, effluent charges, and transferable discharge permits (TDPs). Seasonal variations of the above management strategies have been developed to take advantage of the dynamic assimilative capacity of the receiving water body (see, e.g., Hargett and Seagraves, 1979; O'Neil, 1980; Reheis et al., 1982; Boner and Furland, 1982; Kilgore, 1985; Ferrara and Dimino, 1985; Eheart et al., 1987; Rossman, 1989; Lence et al., 1990). Each of these programs exhibit cost efficiency when compared to similar non-seasonal management strategies.

O'Neil (1980) uses a linear programming model to determine the optimal discharge policy for a hypothetical seasonal TDP management strategy for the Lower Fox River in Wisconsin. The results indicate that, for a water quality goal of 6.2 mg/l, cost savings of approximately $23 million could be achieved by using the seasonal TDP program instead of a non-seasonal uniform treatment policy.

Boner and Ferland (1982) use a seasonal discharge program to study the management of BOD, nitrogenous, and thermal wastes in a stretch of the Chattahoochee River near the City of Atlanta. The study shows that substantial cost savings over a non-seasonal program could be achieved by utilizing the dynamic assimilative capacity of the river. About 13% of the total year-round nitrification and secondary treatment cost, 10% of the total capital cost, and 23% of the user-charge cost (i.e., expansion debt service and coverage for local capital costs and all additional oper-

ation and maintenance costs) could be saved if a seasonal program is implemented instead of the static one proposed.

Lence (1985) develops methods of analysis and assessment for Dynamic Transferable Discharge Permit programs. She identifies three TDP programs capable of exploiting dynamic environmental conditions for the control of BOD waste. These programs are Seasonal Static, Dynamic Feedforward, and Dynamic Feedback Permits. Seasonal Static Permits allow a constant rate of discharge during a time period in the year designated as a 'season'. Dynamic Feedforward Permits index the allowable discharge to observed environmental conditions (e.g., flow and temperature) while the Dynamic Feedback Permits relate allowable discharge to the official water quality indicators which influence and are influenced by the pollutants (e.g., the available DO in streams). She raises questions regarding the combination of possible policy choices that would yield the highest cost efficiency, the type of control mechanism, and the location of monitoring devices (stream gauges or DO meters) that would produce the most efficient and reliable DO control system. Kilgore (1985) shows that a two season TDP program could achieve a cost savings of up to 30% over a single season TDP program for the Willamette River in Oregon.

Although there is no dispute about the superior cost efficiency of seasonal waste management programs compared to non-seasonal ones, there are concerns related to the increased risk of water quality violation, the increased amount of information requirements, the administrative complexity, and the operational difficulty of these programs. Nonetheless, it has been argued that the significant cost savings from the dynamic programs would justify continued investigation of this management approach (Eheart et al., 1987; O'Neil, 1980; Lence, 1985; Kilgore, 1985).

The basis for comparison between the seasonal and non-seasonal programs is only valid if the risk of water quality violation is the same for both types of programs. The acclaimed cost efficient seasonal models allow a larger waste discharge into the stream

and therefore are more likely to cause a violation of the water quality standard or goal. Most of the other concerns about dynamic waste management programs (i.e., increased information needs, administrative complexity, and operational difficulties), can be expressed in monetary terms. Risk of water quality violation or certainty of model outcome however, cannot be quantified in the same monetary terms. Risk Equivalent seasonal programs have therefore been developed for dynamic programs (Rossman, 1989). These programs refer to risk as the probability of incurring a water quality violation, and are designed to maintain the same risk and level of water quality standard as non-seasonal programs.

Rossman (1989) presents a waste discharge program that varies with the natural assimilative capacity of the receiving river body and maintains the same risk of water quality violation as static water quality management programs for the single discharger case. He defines risk as the probability of incurring one or more water quality violations in a given year, and uses a non-linear programming model to find the seasonal discharge limits that minimize total cost while keeping an acceptable risk of water quality violation. The model is applied to controlling ammonia toxicity and BOD, chlorine, and lead effluents in the Quinnipiac River in Connecticut and the Uncompahgre River in Colorado. In both river basins it is demonstrated that the risk-based waste load allocation is more cost efficient than the existing practice (i.e., non-seasonal management program) for that system.

Two Risk Equivalent seasonal waste discharge programs for river basins with several dischargers are developed by Lence et al. (1990). They adopt the definition of risk from Rossman (1989). The two management programs have objectives of minimizing average uniform treatment and maximizing total waste discharge. BOD waste management for the Willamette River in Oregon is used for a case study, and it is demonstrated that both seasonal programs have significant cost savings compared with a non-seasonal model with the same risk of water quality violation. The authors

investigate effects on the total waste treatment cost due to (1) the summer season length for a two-season program and (2) the DO standard. They show that the optimal length of the summer season ranges from two to five months depending on the water quality goal. Although a high cost efficiency is demonstrated for the Risk Equivalent seasonal programs, the authors express concern about a possible increase in environmental damage relative to the corresponding non-seasonal program, and suggest that future research be directed towards developing new measures that account for environmental damage. This damage can then be incorporated in the design of Risk Equivalent seasonal discharge programs.

There is still some pessimism about the performance of the risk equivalent seasonal programs, and uncertainty analysis for these models would be desirable. Seasonal programs are generally more complex and require more information than non-seasonal programs. For example, while year-round BOD waste discharge programs use design flows from low flow seasons and design temperatures from summer, seasonal models need design information for all seasons of the year. Some of this design information is either unavailable or unreliable for certain periods of the year (e.g., winter flows). There is therefore a higher level of uncertainty in the input parameters for seasonal programs than for non-seasonal ones. Model output uncertainty is also expected to be higher for the former. To date, no work has been conducted on uncertainty analysis for the innovative risk equivalent seasonal waste management programs.

Chapter 3

TRADITIONAL METHODS OF UNCERTAINTY ANALYSIS

Three common methods have been applied for analysing uncertainty in environmental system models. These methods are:

- First Order Error Analysis (FOEA);
- Sensitivity Analysis; and
- Monte-Carlo Simulation.

This chapter describes these techniques and discusses their advantages and disadvantages.

3.1 First order error analysis

This method estimates model output mean and variance from input data means and variances. The estimated output variance is taken as an index of the uncertainty in

the model prediction resulting from uncertain input data. The technique uses a first order Taylor Series approximation of the functional relationship between the model output and the input data, and the means and variances of the input data, to estimate the model output mean and variance. The variances of the input data indicate the uncertainty in the input data and are assumed to explain the uncertainty in the model output. The method assumes infinitesimal changes in input data about their mean values. Normally, it estimates only the first two moments of the model output, and consequently may not give the a complete probability distribution of the output. For a mathematical model defined by:

$$Y = Z(\mathbf{X}) \tag{3.1}$$

where

Y = model output;

$\mathbf{X}$ = a vector of input data; and

Z = functional relationship between input data and model output,

the matrix form of the original FOEA expressions which includes correlation among the uncertain input data is given as:

$$E[Y] = Z(\overline{\mathbf{X}}) \tag{3.2}$$

$$E[YY^T] = \mathbf{a^T} E[\mathbf{X}\mathbf{X^T}]\mathbf{a} \tag{3.3}$$

where

$\overline{\mathbf{X}}$ = vector of means of $\mathbf{X}$;

$E[Y]$ = expectation of Y;

$\mathbf{a}$ = vector of partial derivatives of Z with respect to $\mathbf{X}$; and

$\mathbf{B^T}$ = transpose of a vector or a matrix $\mathbf{B}$.

3.1.1 Advantages and disadvantages - FOEA

FOEA is a useful tool in instances when only approximate values of the means and variances of the model output are required from estimates of means and variances of input data, and when the model is approximately linear. It is not computationally intensive and does not need much computer time and memory. Although this method of uncertainty analysis has been extensively used, it has some limitations, which are discussed below.

FOEA assumes a linear relationship between output function and input data. This is not true for most environmental systems. If the second derivative of the response function (i.e., the functional relationship between the model output and inputs) with respect to the uncertain input data is not approximately zero, significant errors in the input data will cause the estimated model output mean and variance to be biased and large errors may result. Using Taylor Series expansions to propagate the variances of the input data assumes that these variances are very small. However, variances of uncertain input data that are encountered in environmental systems are significantly greater than those which would produce acceptable results using Taylor Series expansion. Expected errors or variances in environmental systems model input data are generally greater than 10%. In fact, input data errors up to 1000% have been observed for some analyses (van Straten, 1983).

The error propagation technique used in FOEA estimates the variance of the model output based on the variances of input data and the relationships between the rate of change of output to the given input data at one point in the input data space. This may give large errors in the estimation of the output variance for a highly non-linear model, where the rate of change of model output is very different for different points in the input data space.

This technique requires the use of the derivatives of the model output with respect

to input data to estimate the output variance. This makes it impossible to apply the FOEA method to mathematical models which are not differentiable. If the model is mathematically complex, numerical differentiation may have to be used. This can cause serious problems with the application of FOEA.

3.2 Sensitivity analysis

Sensitivity of a model response to input data is defined as the change in the model output per unit change in a particular input datum. This definition is the basis for both the analytical and numerical approaches for computing sensitivity coefficients or indices. The numerical approach is normally called perturbation analysis. In most applications these two techniques are used to compliment each other.

3.2.1 Analytical sensitivity analysis

This approach computes uncertainty as the partial differential of model output with respect to the individual uncertain input datum. Like FOEA, the basic expression for an index of uncertainty is derived from Taylor Series expansion. For a mathematical model defined by:

$$Y = Z(\mathbf{X}) \tag{3.4}$$

the sensitivity of the model outcome to input datum X_i, is given by:

$$S_i = \frac{\partial Z(\mathbf{X})}{\partial X_i}. \tag{3.5}$$

From *Equation* 3.5 the sensitivity, S_i, computed with this approach is for a single point in the input data space.

The analytical sensitivity approach has been extended by some researchers to analyze uncertainty in optimization models (see, e.g., Chang, 1967 and Uber et al., 1985). For details of these modifications and their applications, the interested reader is referred to Chang (1967), and Uber et al. (1985, 1991a). Modifications to this approach include weighting the sensitivity coefficients obtained to reflect concerns about uncertainty in various input data.

3.2.2 Advantages and disadvantages - Analytical sensitivity analysis

The analytical sensitivity approach is an approximate but simple method for estimating model output sensitivity to input data. It is a useful method when only model output uncertainty about a point in the model output space is required, and when the probability distributions or uncertainties in the input data are not known. For differentiable mathematical models, if the uncertainty in the input data cannot be quantified, it is the most useful of the three traditional methods discussed in this chapter.

Since this approach is calculus based, it also has many of the disadvantages that were discussed for FOEA. The sensitivity obtained for an input datum is evaluated at one point in the uncertain input data space. Therefore it reflects the significance of that input datum at the point where it is evaluated and does not give the overall importance of the input data in the entire input data range for non-linear models. Furthermore, the degree of uncertainty inherent in the input data is not related to the sensitivity, S_i, computed.

Evaluation of the differentials can pose considerable problems for large complex environmental system models especially management models with a large number of constraints. Uber et al. (1985) developed sensitivity equations for a non-linear

optimization model for a waste treatment plant design. When describing the solution technique used in the case study they wrote:

> "The coefficient matrix for the system of linear equations to be solved may be large and very sparse, depending on the model structure. The coefficient matrix for the waste water treatment plant model considered in this work contained less than 5.0% non-zero elements. In this case decomposition of the coefficient matrix by standard Gaussian elimination techniques may not proceed well. Also, a poorly scaled coefficient matrix will result if the NLP is poorly scaled and this could affect the solution accuracy."

The authors also noted that, if the model variables have a high interaction or are poorly scaled the mathematical model may be ill-conditioned and the optimization process would proceed poorly and often stop at a non-optimal point. Such a sub-optimal solution may result in sensitivity coefficients which are grossly in error.

3.2.3 Perturbation analysis

This method of sensitivity analysis is more common than the analytical approach because it is straight forward and avoids the problems associated with evaluating differentials in complex environmental models. For this method, model realizations are obtained by varying input data values in small amounts. To obtain a complete picture of the model sensitivity to a given input datum, the input datum may be varied in the entire uncertain range for all possible combinations of the other uncertain data. Normally, these results are presented in a graphical form for analysis where model output would be plotted against each uncertain input datum.

3.2.4 Advantages and disadvantages - Perturbation analysis

Perturbation analysis is not constrained by the differentiability of a mathematical model. Two-dimensional graphs of model output against each uncertain input datum can be obtained. The graphical representation gives a good picture of the model output characteristics within the input data ranges considered. This method of uncertainty analysis can be used even if the mathematical model is not differentiable or is discontinuous. It is therefore useful for analysing model output uncertainty when the model is complex and when there are few input data.

The sensitivity of model output to an input datum does not only depend on the particular input datum which is being perturbed but also on the values of the others which have been kept constant. Therefore, a large number of analyses and results are required to completely describe how model output changes with respect to the uncertain input datum for environmental systems. Perturbation analysis completely disregards any correlations that might exist among uncertain input data. Inferences made from results obtained with this technique for highly correlated input data could therefore be inaccurate. Although large changes in input data can be examined by this method, it only gives approximate values for the rates of change of output to uncertain input data.

3.3 Monte-Carlo Simulation (MCS)

This technique is the most widely accepted and applied approach for uncertainty analysis in environmental system modelling. In fact, most researchers compare results obtained from the other common methods of uncertainty analysis with those obtained from Monte-Carlo Simulation. The analysis involves fitting distribution functions for the uncertain input data and generating a number of model output

realizations by randomly selecting model inputs from the fitted parent probability distribution functions. The random selection of input data from the fitted distributions ensures that all uncertain input data used in the analysis maintain their parent probability distributions and reflect the degree of uncertainty in them. One common procedure for selecting the input data is to break each random input datum into two terms, a fixed mean and a random component which is a function of its variation. The random component is generated by multiplying the standard deviation of the datum by a normalized random number obtained from a normal random number generator. The uncertain input data used for the simulation can therefore be represented mathematically as:

$$X_{ij} = \overline{X}_i + e_{ij} \tag{3.6}$$

where

X_{ij} = value of input datum i generated during simulation j;

$\overline{X}_i$ = mean value of input datum i (from its parent distribution); and

e_{ij} = random component for input datum i during simulation j.

After the appropriate number of simulations have been performed, summary statistics are evaluated or a distribution function is fitted to the model output to obtain an idea of the uncertainty in the model realizations. The technique is simple and easy but capable of giving a good overall picture of model response uncertainty (i.e., the range and distribution of simulated response) to all the input data distributions used. The usefulness of MCS becomes very apparent for highly non-linear and complex models where analytical solutions for indices of model output uncertainty are difficult or even impossible to obtain.

3.3.1 Number of simulations

An important criterion of the MCS approach is that the distribution functions for the input data samples selected for the Monte-Carlo analysis are similar to the respective parent distributions. The similarity between these sample distribution functions and the unknown but "true" parent distributions for the input data is a function of the number of simulation runs (Whitehead and Young, 1979). The adequate number of simulations depends on the following:

- the type and complexity of the model and output distribution;
- the number of uncertain input data; and
- the distributions or ranges of the uncertain data.

The more complicated a model and its output distribution are, the larger the number of simulations that will be required. Of course, increases in the number of uncertain input data used in the analysis will result in increases in the number of simulations. If the range of an input data is widened, a larger number of Monte-Carlo simulations will be necessary to obtain a distribution similar to the fitted parent distribution. There is no standard method for obtaining the appropriate number of Monte-Carlo simulations. Normally, an experiment is conducted to obtain the number of simulations necessary to give statistically stable output results. Spear and Hornberger (1980) used the Kolmogorov-Smirnov and Renyi statistics to determine the number of adequate simulations. However, some researchers have chosen the number of simulations qualitatively by experience. For example, Gardner et al. (1980) specified 500 simulations as sufficient, and Fedra (1983) used 10000 runs for his analysis.

3.3.2 Limitations of MCS technique

Despite its versatility, MCS has some weaknesses. There could be difficulties encountered with the random number generators required to select the values of input data. Random number generation is an essential component of the technique. It is difficult to obtain random number generators or the appropriate transformation for some types of distributions. Also, less powerful computers are very poor in generating random numbers.

Another drawback is the large computer disc space and time required. The method requires a large amount of computations, and its use could be limited in cases where computer disc space and time are expensive. However, this problem and the one discussed in the previous paragraph are not very serious limitations since large and powerful computers continue to be built and are available to many professionals.

The error variance estimate from Monte-Carlo simulations is difficult to interpret, if the sample distribution of the output exhibits significant skewness or is bimodal, i.e., if the first and the second moments poorly characterize the full distribution (Scavia, 1981b). An important drawback which may affect the utility of MCS in management models is its inability to indicate the sources of input data uncertainty crucial to the overall management strategy. Although some researchers have tried to rank the importance of input data uncertainty to the model output uncertainty (O'Neill et al., 1980; Gardner et al., 1980; Kohberger et al., 1978), their techniques are not adequate especially when model output is multivariate or when there are significant correlation among the uncertain input data.

3.4 Comparison OF FOEA and MCS

Many researchers have applied FOEA and compared the results with those obtained from the widely accepted Monte-Carlo simulations. The agreement between these two methods of uncertainty analysis is reported to be high, although most of the applications violate the basic assumptions of FOEA. There have not been very good explanations for the similarity in results obtained. In an attempt to explain differences in results between the two procedures, Scavia et al. (1981b) concluded that the interpretation of error variances from FOEA and MCS should be considered fundamentally different and argued that the former refers to predictions about the future behaviour of a typical representative of a population, whereas the latter refers to population as an ensemble.

The types of models used may be responsible for the high agreement between uncertainty indices obtained with FOEA and MCS in the literature. If the mathematical models are simple and not highly nonlinear, the two techniques could produce comparable results. Nevertheless, it is not advisable to use FOEA for detailed analysis of uncertainty in nonlinear models because of possible violation of the underlying assumptions. Gardner and O'Neill (1983) caution the use of FOEA for making conclusions about solutions to the problem of reducing prediction uncertainty, although the assumptions of the method do not appear to cause serious problems.

Chapter 4

GENERALIZED SENSITIVITY ANALYSIS

The Generalized Sensitivity Analysis technique is an innovative method that evolved from the ubiquitous Monte-Carlo simulation procedure for uncertainty analysis in complex environmental systems. It was developed by Spear and Hornberger (Spear and Hornberger, 1980; and Hornberger and Spear, 1981), and has since been applied to several environmental system models(see, e.g., Spear and Hornberger, 1983; Humphries et al., 1984; Hornberger and Cosby, 1985a, 1985b; and Hornberger et al., 1986). It was initially used at an early stage of research into poorly defined environmental system models, to assist in model structure identification by isolating critical uncertainties in knowledge of the system under study, and in directing future data collection and research.

Although GSA is a fairly new technique, it has been demonstrated to be very useful especially for complex systems. There are indications that it will be widely used by researchers in environmental system modelling. When concluding his review on uncertainty analysis in water quality models, Beck (1987) wrote:

> "The contributions of over a decade of research into the analysis of uncertainty in water quality modelling have been many and varied. Perhaps the most innovative has been the HSY algorithm *(the GSA algorithm)* in Section 3, a conceptually simple means of generating preliminary hypothesis about the behaviour of a system under conditions of sparse field data and gross uncertainty in the prior theoretical knowledge of the system's behaviour."

The early applications of GSA were for identifying plausible model structures and selecting the best among them, and the task was accomplished by evaluating sensitivity indices of input parameters. The technique has also been used for model calibration or parameter estimation, and for computing the contributions of uncertain input parameters to total model output uncertainty (Hornberger and Cosby, 1985b; Hornberger, Cosby and Galloway, 1986).

GSA is used to develop insight into environmental system behaviour that cannot be easily obtained with traditional uncertainty analysis techniques. It has all the characteristic advantages of MCS. The procedure is simple and the statistical analysis used is flexible and capable of utilizing non-standard goodness-of-fit criteria. The GSA approach is discussed in this Chapter under two main sections: one related to uncorrelated input parameters, and the other dealing with correlated input parameters. The former is presented in Section 4.1 and the latter is presented in Section 4.2. For the correlated parameters, a relationship used to compute sensitivity indices for the uncertain parameters is developed in this research and is also presented in Section 4.2. Section 4.3 defines a suitable behavioural classification algorithm for management models that incorporate simulation and optimization.

4.1 GSA approach for uncorrelated inputs

Figure 4.1 outlines the four major steps involved in the GSA method when the uncertain input parameters are independent of each other. They are:

1. Binary classification definition;
2. Monte-Carlo Simulations (MCS);
3. Classification of parameter sets for the MCS results based on Step 1; and
4. Statistical analysis of the parameter sets for the two classes obtained in Step 3. For each parameter, analyse whether the cumulative distribution function for the two classes are significantly different. The K-S statistic which evaluates the separation of the Cumulative Distribution Functions (CDFs) for the two classes is referred to as the sensitivity index for a given parameter.

With the exception of Step 2 which is essentially the same as the MCS described in Section 3.3, these steps are explained in the following sections.

4.1.1 Binary classification definition

The definition used for the classification of model output is based on the expected system response (i.e., the expected system output values and their pattern). Simulation results that mimic important system output characteristics, either measured or reported, are classified as Behaviours **(B)** and those that do not are classified as Non-Behaviours **(NB)**. For example, when applying GSA to a phosphorus model for cultural eutrophication in the Peel Inlet in Australia, Spear and Hornberger (1980) used information obtained about the physical system from a sampling program (Atkins et al., 1977) to define the behavioural classification. They defined system behaviour

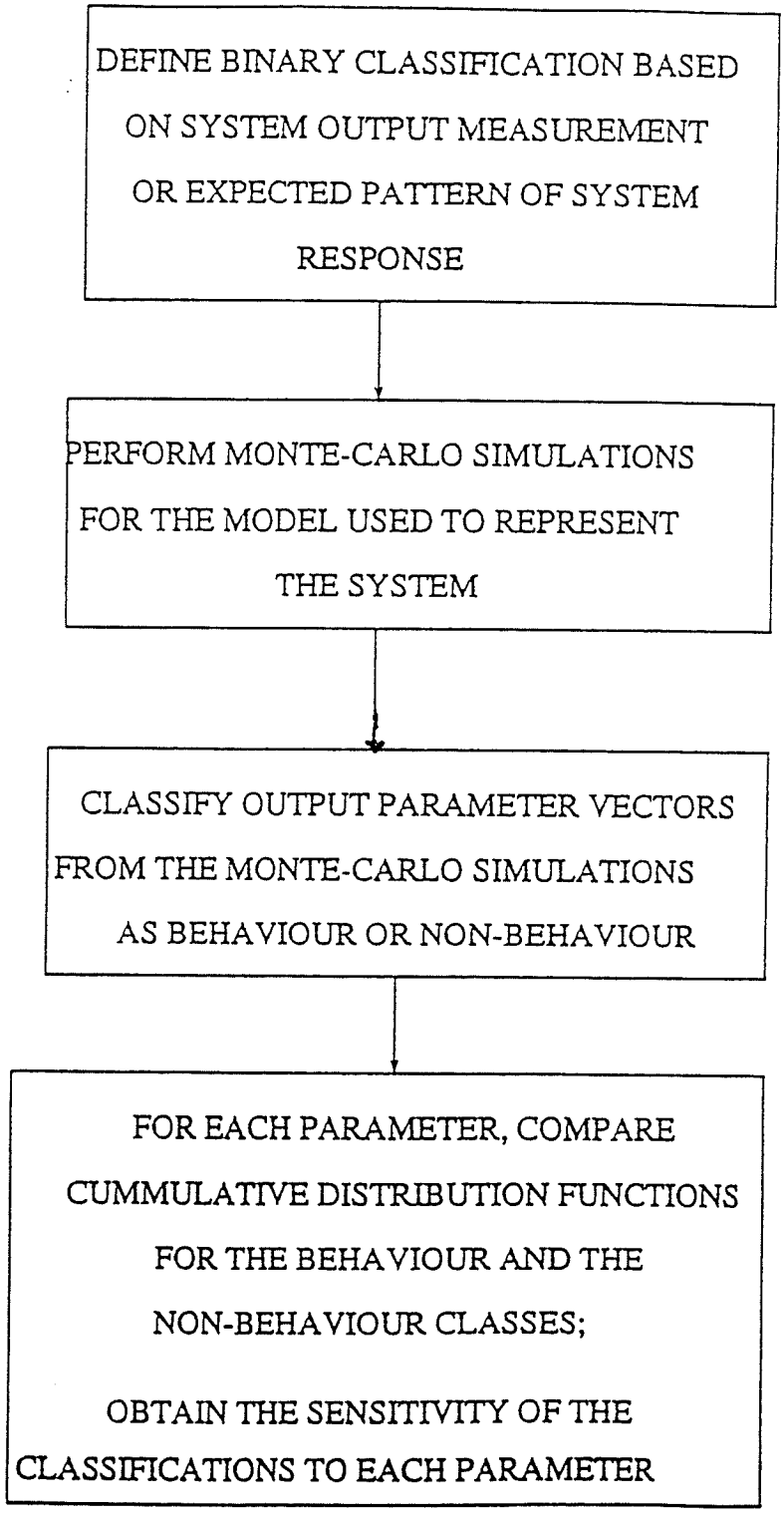

Figure 4.1: Flow diagram for the GSA

based on whether the simulated *Cladophora* biomass and phosphorus concentrations resembled values were reported by Atkins et al. (1977).

4.1.2 Classification of parameter sets for the MCS

This step of the procedure follows the classification definition and the MCS steps. If the results for a particular simulation mimic observed system characteristics, the parameter set which produced this model output becomes a member of the **B** group and represents a column of the Behaviour Matrix. Similarly, for simulation results that do not mimic observed characteristics the vector of values of the parameter set for that simulation constitutes a column of the Non-Behaviour Matrix. This classification process, therefore, divides the matrix of parameter sets for all the simulations performed into two matrices, each representing one behavioural class. Assuming for a given environmental model that there are p uncertain parameters and l simulations, and that the classification definition results in m behaviours and n non-behaviours, the dimension of the matrices will be given as:

Behaviour Matrix	$p \times m$;
Non-Behaviour Matrix	$p \times n$; and
Unclassified Matrix	$p \times l$.

Row k ($1 \leq k \leq p$) of the Unclassified Matrix represents values of parameter k from all of the simulations, and column j ($1 \leq j \leq l$) contains values of the parameter set used for simulation j. Similar definitions for the rows and columns apply to the Behaviour and Non-Behaviour Matrices in which case $1 \leq j \leq m$ and $1 \leq j \leq n$, respectively.

4.1.3 Statistical analysis and determination of sensitivity indices

The main reason for this analysis which is performed on parameter matrices obtained from the classification step, is to assess the degree to which the probability distributions of the Behaviour and the Non-Behaviour sets for each parameter differ from each other. The separation of the Cumulative Probability Distribution Functions (CDFs) of the two classes for each parameter gives an indication of the importance of that parameter for simulating a behaviour or a non-behaviour. Spear and Hornberger (1980) proved that the degree of separation of the CDFs for each parameter could be used as a sensitivity index for that parameter. They used the Kolmogorov-Smirnov (K-S) two sample test statistic, d_{mn}, to measure the degree of separation, and to represent the sensitivity of the behavioural classification to uncertainties in the input parameters. The K-S statistic for the two classes of parameter X_i, $d_{mn}^{x_i}$ is defined as:

$$d_{mn}^{x_i} = \sup_{X_i} |F_n(X_i) - F_m(X_i)| \tag{4.1}$$

where

$F_n(X_i)$ = sample CDF for the Behaviour class of parameter X_i; and

$F_m(X_i)$ = sample CDF for the Non-Behaviour class of parameter X_i.

The K-S statistic gives the maximum vertical distance between the cumulative distribution functions of the two behavioural classes for a given parameter. The value of the d_{mn} statistic can be used to test the hypothesis that the Behaviour and Non-Behaviour CDFs are the same at a given significance level. The d_{mn} for each parameter represents the sensitivity index of that input parameter and reflects both differences in central tendency and in the behavioural distribution functions. If the two behavioural CDFs of an input parameter are significantly different a large d_{mn}

value is determined. If the Behaviour and Non-Behaviour distributions for a certain parameter are not different, then the potential simulation of either of the two classes can be considered as random and therefore less sensitive to that parameter. While the magnitude of the d_{mn} that constitutes a large value has not yet been determined, for a given simulation model the relative values of d_{mn} for different parameters may be used to rank the relative importance of these parameters to the model output. Figures 4.2 and 4.3 show the CDFs and the d_{mn} statistic for two hypothetical input parameters which are significant and insignificant to a behavioural definition respectively.

The analysis and inferences discussed above are valid only when the uncertain input parameters are uncorrelated. The presence of strong correlation between parameters can result in an erroneous underestimate of the d_{mn} value and an inability to identify significant input parameters. Analyses required for correlated uncertain input parameters are discussed in the next section.

4.2 GSA approach for correlated input data

The separation of the behavioural cumulative distribution functions, evaluated with the d_{mn} statistic, give an index of the importance of a given parameter for simulating a behaviour. The basis for evaluating the separation between two distribution functions along the original axes of the parameter (i.e., with the original parameter values) is not valid if there is strong correlation among the uncertain input parameters. In the case of correlated inputs, significant separation of the two behavioural CDFs still indicates that the ability to simulate a behaviour is sensitive to the particular parameter, however, the converse is not always true. It is possible that a parameter shows no separation of the behavioural CDFs, yet is important in simulating either of the two classes because of its high correlation with another uncertain parameter.

Hornberger et al. (1986) observed that even if there is no significant correla-

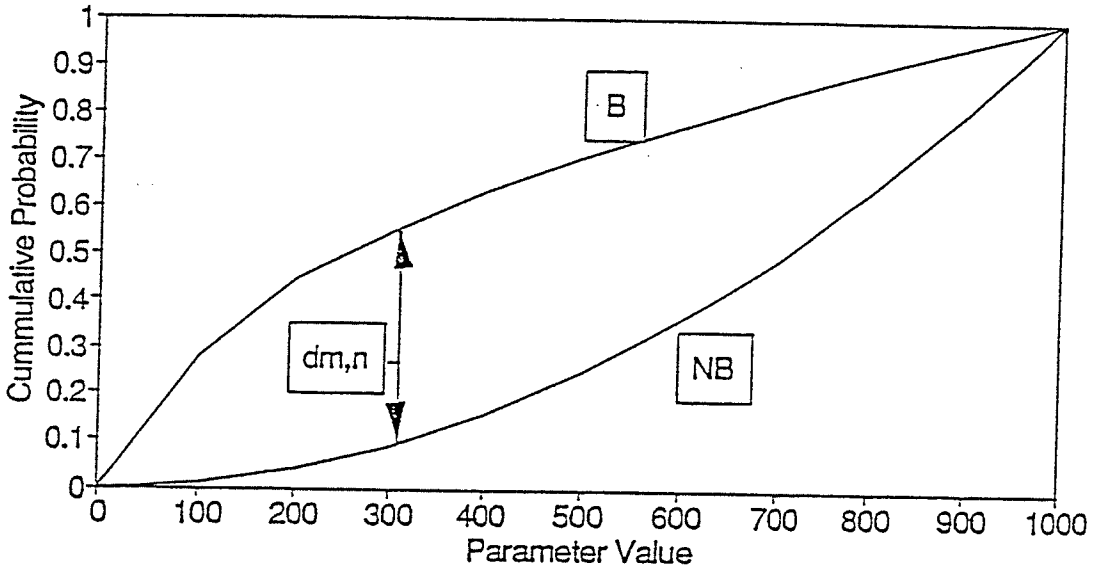

Figure 4.2: Graphical representation of the K-S statistic (significant separation)

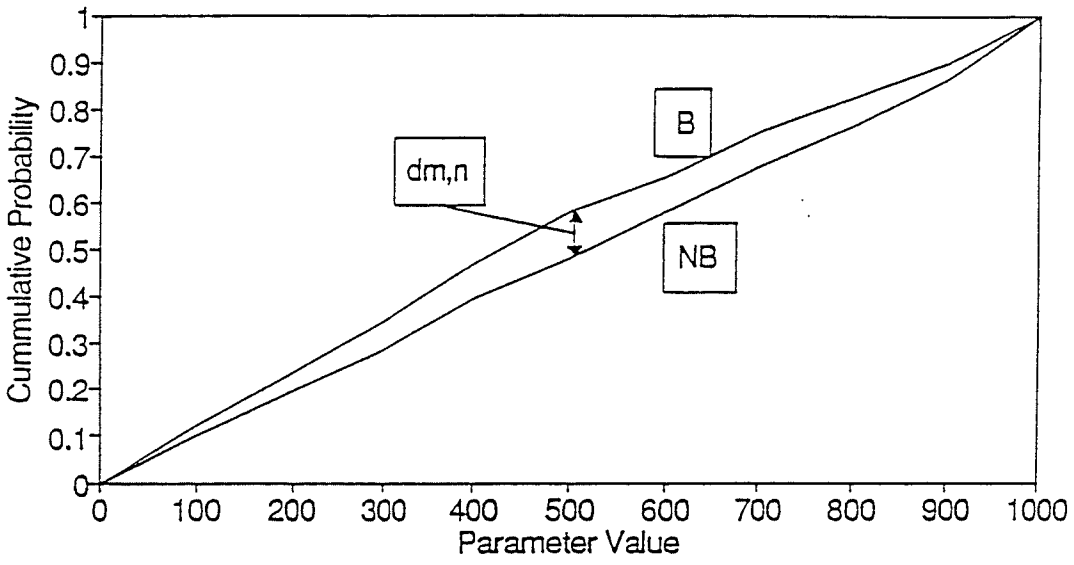

Figure 4.3: Graphical representation of the K-S statistic (insignificant separation)

tion among input parameters, there might still be the need for modification of the GSA for uncorrelated inputs since the uncertain parameters are not independently important in simulating acceptable results. This is a reasonable observation because the uncertain input parameters do not always appear independent of one another in model equations, and model simulation results are based on a combination of input parameter values rather than on the individual parameter values alone.

Principal components analysis is used with the GSA technique when the input parameters or data are correlated or when the univariate analysis discussed in the last section is not applicable. The multivariate (principal components) analysis has been used by some researchers to identify general indicators of sensitivity when correlation between input parameters exists (see, e.g., Hornberger et al., 1986; and Hornberger and Spear, 1981). However, none of these authors have developed or obtained sensitivity indices for the original uncertain input parameters. One of the two main contributions of this thesis is the development of an approach for computing sensitivity indices for individual uncertain input parameters which are correlated and require multivariate analysis. The approach uses the basic univariate GSA technique, simultaneous diagonalization of a function of the variance-covariance matrices of the Unclassified, Behaviour, and Non-Behaviour Parameter Matrices, and a classical definition of sensitivity to derive sensitivity indices for each of the uncertain input parameters. Simultaneous diagonalization is applied to obtain a matrix which can be used to transform the original correlated parameters to uncorrelated variables and to derive sensitivity indices for the original input parameters from sensitivity indices calculated for the uncorrelated variables.

4.2.1 Transformation of original parameters

This section describes how the original correlated input parameters are transformed to an independent variable set. This transformation is comprised of normalization of the original parameters, simultaneous diagonalization of the variance-covariance matrices of the Unclassified, Behaviour, and Non-Behaviour Matrices, and the transformation of the normalized correlated input parameters to an uncorrelated variable set.

Normalization of original input data set

Because of the possibility of different units and scales of the input data, the original parameters must be normalized to the same scale before the transformation is performed. The normalization is described by the following equation:

$$X_{ij} = \frac{Z_{ij} - \overline{Z_i}}{\sigma_{z_i}} \tag{4.2}$$

where

i = parameter index;

j = simulation index;

X_{ij} = normalized value for parameter i and simulation j;

Z_{ij} = original value for parameter i and simulation j;

$\overline{Z_i}$ = mean of parameter i for the Unclassified Simulations; and

σ_{z_i} = standard deviation of parameter i for the Unclassified Simulations.

The sensitivity index obtained for a given input parameter using the GSA approach does not change when the parameter is normalized using *Equation* 4.2 since this normalization process only changes the scale of the abscissa of the CDF for that

parameter. This process will therefore not affect the relative importance of the original parameters.

Simultaneous diagonalization of functions of the covariance matrices

The purpose of this procedure is to obtain a matrix which can be used to transform the original normalized correlated parameters to uncorrelated variables. The variables obtained are linear combinations of the original parameters. The univariate analysis discussed under GSA with uncorrelated input parameters, can then be applied to the transformed variables. Assume $\mathbf{X}$ is a matrix of the original normalized parameters for all simulations performed. A matrix of an independent variable set can then be obtained from:

$$\mathbf{Y} = \mathbf{AX} \tag{4.3}$$

where

$\mathbf{Y}$ = matrix of transformed, uncorrelated variable set for all simulations; and

$\mathbf{A}$ = transformation matrix.

If there are p parameters and l simulations, $\mathbf{Y}$ and $\mathbf{X}$ are both $p \times l$ matrices and $\mathbf{A}$ is a $p \times p$ matrix.

The variance-covariance matrix of $\mathbf{Y}$, $\mathbf{M_y}$, can be defined as:

$$\mathbf{M_y} = E[\mathbf{YY^T}] \tag{4.4}$$

where

$E[\mathbf{D}]$ = expectation of $\mathbf{D}$; and

$\mathbf{D^T}$ = transpose of a matrix $\mathbf{D}$.

Substituting $\mathbf{AX}$ for $\mathbf{Y}$ (see *Equation* 4.3), $\mathbf{M_y}$ becomes:

$$\mathbf{M_y} = E[\mathbf{AX}(\mathbf{AX})^{\mathbf{T}}] \tag{4.5}$$

which simplifies to:

$$\mathbf{M_y} = \mathbf{A}E[\mathbf{XX}^{\mathbf{T}}]\mathbf{A}^{\mathbf{T}}. \tag{4.6}$$

But the variance-covariance matrix of $\mathbf{X}$ is given by:

$$\mathbf{M_x} = E[\mathbf{XX}^{\mathbf{T}}]. \tag{4.7}$$

Therefore *Equation* 4.6 can be written as:

$$\mathbf{M_y} = \mathbf{AM_xA}^{\mathbf{T}}. \tag{4.8}$$

Since the $\mathbf{Y}$ variables are uncorrelated, $\mathbf{M_y}$ is equal to the identity matrix $\mathbf{I}$ and *Equation* 4.8 becomes:

$$\mathbf{M_y} = \mathbf{AM_xA}^{\mathbf{T}} = \mathbf{I}. \tag{4.9}$$

In this case, it is not only necessary to remove correlation in the original set of parameters, but also to remove correlation in each of the two classified parameter matrices, the Behaviour and Non-Behaviour Matrices. The variance-covariance matrices of the Behaviour, Non-Behaviour, and Unclassified parameter sets are related by:

$$E[\mathbf{XX}^{\mathbf{T}}] = P(B)E[\mathbf{XX}^{\mathbf{T}}/B] + P(NB)E[\mathbf{XX}^{\mathbf{T}}/NB] \tag{4.10}$$

where

$P(B)$	=	probability of a behaviour simulation;
$P(NB)$	=	probability of a non-behaviour simulation;

$E[\mathbf{X}\mathbf{X}^{\mathbf{T}}/B]$ = variance-covariance matrix for parameters in the behaviour class; and

$E[\mathbf{X}\mathbf{X}^{\mathbf{T}}/NB]$ = variance-covariance matrix for parameters in the non-behaviour class.

Equation 4.10 can be simplified to:

$$\mathbf{M_x} = \mathbf{M'_b} + \mathbf{M'_{nb}} \tag{4.11}$$

where

$$\mathbf{M'_b} = P(B)E[\mathbf{X}\mathbf{X}^{\mathbf{T}}/B] \tag{4.12}$$

$$\mathbf{M'_{nb}} = P(NB)E[\mathbf{X}\mathbf{X}^{\mathbf{T}}/NB] \tag{4.13}$$

$$\mathbf{M_x} = E[\mathbf{X}\mathbf{X}^{\mathbf{T}}] \tag{4.14}$$

Substituting *Equation* 4.11 in *Equation* 4.9 and simplifying yields:

$$\mathbf{A}\mathbf{M'_b}\mathbf{A}^{\mathbf{T}} + \mathbf{A}\mathbf{M'_{nb}}\mathbf{A}^{\mathbf{T}} = \mathbf{A}\mathbf{M_x}\mathbf{A}^{\mathbf{T}} = \mathbf{I}. \tag{4.15}$$

$\mathbf{M'_b}$, $\mathbf{M'_{nb}}$, and $\mathbf{M_x}$ are all symmetric matrices, therefore a real matrix $\mathbf{A}$ exists. $\mathbf{A}$ is the matrix that simultaneously diagonalizes $\mathbf{M'_b}$, $\mathbf{M'_{nb}}$, and $\mathbf{M_x}$.

From matrix algebra $\mathbf{A}^{\mathbf{T}}$ can be obtained from *Equation* 4.15 as:

$$\mathbf{A}^{\mathbf{T}}\Lambda = \mathbf{M_x^{-1}}\mathbf{M'_b}\mathbf{A}^{\mathbf{T}} = \mathbf{M_x^{-1}}\mathbf{M'_{nb}}\mathbf{A}^{\mathbf{T}} \tag{4.16}$$

where

$\mathbf{M_x^{-1}}$ = inverse of the variance-covariance matrix of the Unclassified Parameter Matrix;

Λ = a diagonal matrix of eigenvalues for $\mathbf{M_x^{-1}}\mathbf{M'_b}$ or $\mathbf{M_x^{-1}}\mathbf{M'_{nb}}$;

and

$$\mathbf{A^T} = \text{eigenvector matrix of } \mathbf{M_x^{-1}M'_b} \text{ or } \mathbf{M_x^{-1}M'_{nb}}.$$

The transformation matrix **A**, is therefore the transpose of the eigenvector matrix of the product of $\mathbf{M_x^{-1}}$ and $\mathbf{M'_b}$ or $\mathbf{M_x^{-1}}$ and $\mathbf{M'_{nb}}$ (Fukunaga, 1972).

Transformation of original parameter set

The transformation matrix **A** can be used to transform the correlated input parameters into an uncorrelated variable set. The independent variable set can then be classified according to the binary classification of the corresponding simulations, and sensitivity indices may be obtained for them using the univariate analysis described in Section 4.1. It should be noted that the transformation does not affect the binary classification definition or the classification procedure. The class of each simulation remains the same after the transformation of the input parameters. The parameter set for a given simulation and the transformation matrix **A** are used to produce an independent variable set for that simulation. For example, if there are p correlated input parameters in a given model, the transformation process will produce p uncorrelated variables for each simulation.

4.2.2 Derivation of sensitivity indices for original correlated parameters

The transformation of the correlated input parameter sets produces sensitivity indices for the transformed variables, but not the original parameters which are of interest to the environmental quality manager. The purpose of uncertainty analysis is not only to get a measure of the overall uncertainty in the model output, but also to know the sources of these uncertainties. Knowledge of the contribution from individual input

data uncertainty to total model uncertainty is important for model structure identification, parameter estimation, and system management. In this thesis an approach is developed for evaluating sensitivity indices for the original input parameters.

Suppose the mathematical model of interest is represented by the following simple equation:

$$G = Q(\mathbf{X}) = Q(X_1, X_2, X_3, \ldots, X_i, \ldots, X_p) \tag{4.17}$$

where

G = model output;

Q = functional relationship between model output and normalized uncertain input parameters;

$\mathbf{X}$ = a vector of normalized uncertain input parameters; and

p = number of uncertain input parameters.

If there is no functional relationship among the X_i values, then from *Equation* 4.17,

$$\frac{\partial G}{\partial X_i} = \frac{\partial Q}{\partial X_i} \tag{4.18}$$

If $X_i = X_i(\mathbf{Y}) = X_i(Y_1, Y_2, \ldots, Y_p)$, *Equation* 4.18 would be written as:

$$\frac{\partial G}{\partial X_i} = \sum_{t=1}^{p} \frac{\partial Q}{\partial Y_t} \frac{\partial Y_t}{\partial X_i} \tag{4.19}$$

A sensitivity index for parameter X_i, S_i, can therefore be defined as:

$$S_i = \max \left| \frac{\partial Q}{\partial X_i} \right| = \max \left| \sum_{t=1}^{p} \frac{\partial Q}{\partial Y_t} \frac{\partial Y_t}{\partial X_i} \right| \tag{4.20}$$

Equation 4.20 can be simplified to:

$$S_i = \sum_{t=1}^{p} \left| \frac{\partial Q}{\partial Y_t} \frac{\partial Y_t}{\partial X_i} \right| \tag{4.21}$$

For an analytical sensitivity analysis, $\frac{\partial Q}{\partial X_i}$ and $\frac{\partial Q}{\partial Y_t}$ represent the sensitivity indices of the model output to parameter X_i and to variable Y_t, respectively. Similarly, the sensitivity indices, d_{mn}, obtained for the uncorrelated Y_t variables using a univariate GSA can be used as surrogates for the sensitivity index $\frac{\partial Q}{\partial Y_t}$. Since $d_{mn}^{Y_t} \geq 0$, substituting $d_{mn}^{Y_t}$, for $\frac{\partial Q}{\partial Y_t}$ in *Equation* 4.21 becomes:

$$S_i = \sum_{t=1}^{p} d_{mn}^{Y_t} \left| \frac{\partial Y_t}{\partial X_i} \right| \tag{4.22}$$

Here, $|\frac{\partial Y_t}{\partial X_i}|$ is the magnitude of the i-th component of the t-th transformation vector (the t-th column of the transformation matrix **A**). It represents the relative contribution of parameter X_i to the separation obtained for the transformed variable Y_t. These values can be normalized (i.e., forced to have a sum of unity) so that they indicate the proportion of the separation in the CDFs of the behavioural classes of transformed variable Y_t, which is attributed to parameter X_i. This is achieved by dividing the RHS of *Equation* 4.22 by the sum of the magnitudes of the t-th column vector of matrix **A**. That is:

$$S_i = \sum_{t=1}^{p} d_{mn}^{Y_t} \frac{|\frac{\partial Y_t}{\partial X_i}|}{\sum_{k=1}^{p} |\frac{\partial Y_t}{\partial X_k}|} \tag{4.23}$$

where

$\frac{\partial Y_t}{\partial X_i}$ = i-th component of the t-th vector of matrix **A**.

Setting $\frac{\partial Y_t}{\partial X_i} = a_{ti}$, *Equation* 4.21 becomes:

$$S_i = \sum_{t=1}^{N} d_{mn}^{Y_t} \frac{|a_{ti}|}{\sum_{k=1}^{N} |a_{tk}|} \tag{4.24}$$

Equation 4.24 can be used to obtain sensitivity indices for the original input parameters. Although calculus has been used to derive the expression for the sensitivity indices for correlated input parameters, *Equation* 4.24 does not include a derivative

and is applicable even if the model is not differentiable in the interval of interest. Therefore these sensitivity indices are not subjected to the disadvantages and limitations of the traditional analytical sensitivity analysis technique. The analytical sensitivity indices (i.e., the derivative of model output with respect to uncertain input parameters) are replaced by surrogate sensitivity indices from the modified GSA. The sensitivity indices obtained from *Equation* 4.24, therefore, exhibit all the favourable attributes of the GSA technique.

4.3 Behavioural definition for environmental optimization models

This section presents the second major contribution of this work. It is a modification of the criteria for defining the behaviour classification which makes the GSA applicable to both optimization and simulation models. Spear and Hornberger's behavioural definition for simulation models uses observed system response obtained from measurements, literature, and professional judgement to define whether or not a simulation result can be classified as a behaviour. Generally, the output from an optimization model does not have an observed system response that can be used to define a behaviour or a non-behaviour. For example, an optimization based management model to control sulphur-dioxide (SO_2) pollution will give the amount of SO_2 that each polluter may emit in order to achieve a specified level of air quality at minimum cost. However, the expected emission levels that will achieve the management goals cannot be obtained from observed system measurements or any way other than through model prediction.

A modification of the behavioural definition is hereby proposed for optimization based environmental management models. A deterministic solution of the manage-

ment model of interest is obtained. This solution is based on parameters that are estimated from known or measured input data. The deterministic model output is what would be used to obtain a management policy for the system if there were no uncertainties in the input parameters or data. The modified behavioural definition assumes that the deterministic solution is just adequate for maintaining the environmental quality goal. Then the binary classification is based on whether or not the results of a given MCS would violate the environmental quality goal by comparing the management solution from the MCS to that of the deterministic model. Simulations which give management strategies that will result in worse environmental quality than the deterministic solution are classified as Non-Behaviours and the others are classified as Behaviours. Professional judgement can also be incorporated in the modified behavioural definition. For instance, suppose that in the SO_2 pollution example given above there is only one source of pollution and the deterministic solution is that 40 $KT\ SO_2/yr$ may be emitted by this source. The truth set for the behaviour class can be defined as:

$$B = \{30 \leq EL \leq 50\}$$

or

$$B = \{0 \leq EL \leq 40\}$$

where

B = set of model output that falls in the behaviour class; and

EL = emission level for the polluter, obtained from a given MCS of the optimization based management model ($KT\ SO_2/yr$).

The first of the two definitions indicates that emission levels within 25% of the deterministically based optimal emission level will not result in environmental objective failure and are therefore acceptable. This definition addresses the concerns

that the management policy should neither be under nor over designed by an amount greater than 25%. The second definition for the SO_2 example reflects the concern that the management policy should not be under-designed. This definition reflects a strong emphasis on satisfying environmental quality goal. Sensitivity indices can be calculated for several definitions of a behaviour. A GSA can also be performed for different management concerns such as the level of air quality and the total cost for the air pollution example given above.

Chapter 5

CASE STUDY

The modified GSA for correlated input parameters and optimization based environmental management models developed in Chapter 4 is demonstrated for the Minimum Average Uniform Treatment Risk Equivalent Seasonal Discharge (MAUT) Program (developed by Lence et al., 1990) and applied to the Willamette River Basin in Oregon. This chapter gives a brief description of the MAUT and the Willamette River Basin and details of the procedure for applying the techniques developed in this thesis.

5.1 The Minimum Average Uniform Treatment Risk Equivalent Seasonal Discharge Program

The MAUT program for controlling BOD discharges and ambient DO levels specifies the uniform percent BOD removal level in each season that minimizes the total annual cost of waste treatment for the region and maintains a specified level of risk. The risk criterion limits the probability of incurring one or more water quality violations in a given year. The solution approach is a two step process.

In the first step, linear water quality management models are solved for each season, s, of each year, y, in the history of record. For these models, strict water quality constraints maintain the desired DO standard at all locations in the stream and the objective function minimizes the uniform percent removal level required to satisfy the water quality constraints. Water quality impact coefficients for the water quality constraints are based on results from simulation models for steady-state stream conditions for each period. The solutions to the series of LP models define the historical series of uniform percent removal levels, UT_{sy}, for each season s and year y.

In the second step, a non-linear model is solved which selects a design set of seasonal uniform percent removal levels that minimizes the annual cost of waste treatment and limits the frequency of years that contain one or more events in which the design uniform percent removal level is less than the historical value, UT_{sy}. The formulation for this model is:

$$\min z = \sum_{s=1}^{R} day_s \sum_{d=1}^{D} c_d \xi_{ds} \tag{5.1}$$

subject to

uniform treatment requirement:

$$\xi_{ds} - \nu_s = 0 \qquad \forall_s = 1, \ldots, R, \; d = 1, \ldots, D \tag{5.2}$$

limit on the frequency of years which contain one or more water quality violations:

$$\frac{\sum_{y=1}^{T} \prod_{s=1}^{R} [1 - V(UT_{sy} > \nu_s)]}{T} + P - 1 \geq 0 \tag{5.3}$$

limits on the allowable treatment level at any given time:

$$\xi_{ds} \geq \xi L_{ds} \qquad \forall_s = 1, \ldots, R, \; d = 1, \ldots, D \tag{5.4}$$

$$\xi_{ds} \leq \xi U_{ds} \qquad \forall_s = 1,\ldots, R, \; d = 1,\ldots, D \qquad (5.5)$$

where

R = number of seasons per year;

day_s = number of days in season s;

ξ_{ds} = design percent removal level for discharger d in season s;

c_d = daily cost per unit of percent removal for discharger d;

ν_s = design uniform percent removal level in season s;

T = years of stream flow and temperature record;

UT_{sy} = uniform percent removal level required based on water quality levels in season s of year y;

P = acceptable probability of incurring one or more water quality violations in any given year;

$V(\mathbf{X})$= 1 if condition $\mathbf{X}$ is true and = 0 if condition $\mathbf{X}$ is false;

ξL_{ds} = lower limit on ξ_{ds}; and

ξU_{ds} = upper limit on ξ_{ds}

The first constraint represents the uniform treatment requirements for the dischargers in any season s. The second constraint is referred to as the risk equivalency condition. It limits the number of years, on average, in which at least one water quality violation would occur by comparing the design uniform percent removal level in each season, ν_s, with the historical series of uniformpercent removal levels, UT_{sy}. The third constraint set represents the technological limits of waste treatment.

For the mathematical formulation and details about the Minimum Average Uniform Treatment Seasonal Discharge Program, the interested reader is referred to Lence et al. (1990).

5.2 The Willamette River case study

A 298 *km* segment of the Middle Fork of the Willamette River in Oregon was used for this case study. This portion of the river has six river gauging stations, eight main tributaries, and ten major BOD waste dischargers. The gauging stations are the Middle Fork, Coast Fork, Harrisburg, Albany, Salem, and Portland Stations. Thirty years of mean daily streamflow data from 1954 to 1984 for the six gauging stations and twenty-three years of mean daily temperature data from 1961 to 1984 for the Harrisburg gauging station were obtained from the United States Geological Survey and were used in this analysis. Since the Coast Fork tributary joins the Middle Fork of the river upstream of the segment of interest and of Springfield, Oregon, the flow records for the Middle Fork and Coast Fork gauging stations were added and assigned to what is referred to as the Springfield Station. Five flow stations were consequently used for this study. Their locations relative to Springfield (obtained from Kilgore, 1985) and the station numbers adopted for this thesis are shown in Table 5.1. All gauging stations, except for the Harrisburg Station had short records for temperature values. Temperature records at Harrisburg were therefore assumed to represent temperature data in the portion of the river basin used for the case study.

Piecewise-linearized treatment cost data for the ten major BOD waste dischargers along the Middle Fork, developed by Kilgore (1985), were utilized. The treatment cost data are in 1978 dollars and are shown in Table 5.2. Also listed are the dischargers' locations, influent BOD concentrations and design waste flows. For all reaches except the final 48 *km* segment which is estuarine, Worley (1963) developed a functional relationship between velocity and streamflow, and reaeration rates and streamflow for the Willamette River. Liebman (1965) assumed a velocity which is proportional to streamflow and a constant reaeration rate for all streamflows for the estuarine portion. There are no existing estimates of the benthic oxygen demand and the

Table 5.1: Flow stations used for analysis

Station	Number	Location (km)
Springfield	1	0.00
Harrisburg	2	10.13
Albany	3	108.00
Salem	4	160.00
Portland	5	254.70

Table 5.2: Discharger waste load characteristics and treatment cost options

Discharger	No.	Location (km)	Design flow (L/S)	Infl. *BOD** (mg/L)	BOD removal(%)	Total cost ($ m/yr)
Springfield	1	0.00	301.0	248	35.0	1.60
					67.0	1.80
					93.0	1.95
					98.0	4.30
Eugene	2	10.2	752.6	284	35.0	2.94
					68.0	3.25
					94.0	3.60
					98.0	5.51
American Can Co	3	57.8	235.1	101	35.0	0.89
					66.0	1.16
					89.0	2.08
					98.0	3.47
Evans Products	4	79.4	181.6	118	35.0	0.40
					47.0	0.48
					70.0	0.85
					88.0	1.35
					98.0	2.95
Corvallis	5	85.8	312.6	121	35.0	1.68
					63.0	1.85
					94.0	2.13
					98.0	3.19
Albany	6	105.2	382.1	93	35.0	1.96
					63.0	2.16
					85.0	2.39
					98.0	4.81
Western Kraft	7	108.3	185.5	240	35.0	1.15
					68.0	1.44
					88.0	2.52
					98.0	5.72
Boise Cascade	8	161.2	689.5	130	35.0	0.85
					46.0	0.94
					72.0	2.14
					90.0	4.32
					98.0	6.59
Salem	9	163.1	775.8	408	35.0	2.92
					69.0	3.23
					96.0	3.57
					98.0	4.67
Crown Zellerbach	10	254.9	415.4	279	35.0	1.65
					68.0	2.19
					90.0	4.02
					98.0	6.58

*Biochemical oxygen demand (BOD), 5 days at 20°C.

Adapted from Kilgore (1985)

background DO deficit for the river and its tributaries. The benthic oxygen demand was therefore assumed to be zero, and the background DO deficit was set to 1.0 mg/l for the river and its tributaries. The dischargers' effluent temperature and DO concentration were set at $15°C$ and 0.0 mg/l respectively for both summer and winter seasons.

5.3 Methodology used for analysis

This section discusses the management decisions and uncertain input data considered in this research, the procedure for selecting the input data for the MCSs, and the application of GSA to the MAUT Program.

5.3.1 Management decisions and input data analysis

The resultant uncertainty under two main management decisions were investigated. These decisions are the season length combinations and the water quality standard chosen by the manager. Lence et al. (1990) investigated the cost efficiency of the MAUT program under both of these decisions in the deterministic optimization models they developed. They concluded that it was not necessary to examine seasonal discharge programs with more than two seasons for the Willamette River. Hence, a two-season, summer and winter, discharge schedule was investigated in this research.

The seasonal flow and temperature data chosen for each year were selected from the 7-day average low flow values and the mean temperatures for each month in that year. For a given season, the most critical flow and the highest monthly average temperature was selected. The most critical flow condition was determined by ranking the monthly 7-day average low flow values for a given season and year and for each

gauging station in an ascending order of magnitude. The critical flow condition was the flow in the month with the lowest mean rank for all five stations.

Six season length combinations were examined in which the summer season length ranged from one to ten months. Months included in the summer season were determined by first selecting the month with the most severe flow and temperature conditions in the period of record (a one month summer season), and then including the month with the next most severe water quality conditions for each successive summer season duration through ten months. Since the month that was added each time was also adjacent to the months in the previous summer season (of one month shorter duration), it was not necessary to examine seasonal discharge programs with more than two seasons. A summary of the six two-season length combinations analysed are given in Table 5.3.

Different station flow and temperature data were obtained for the different season length combinations. Thirty years of low flow data were developed for each gauging station and each season. However, only twenty-three years of seasonal temperature values could be calculated from the Harrisburg temperature records. In order to base MAUT on thirty years of records, the mean of the seasonal temperatures obtained for the twenty-three years were assumed to be equal to the corresponding seasonal temperatures for each of the seven years without data. This approach is reasonable because the seasonal temperatures had a very small range (less than $4^{o}C$ for all season length combinations considered).

Dissolved oxygen (DO) standards of 6.00, 7.00, and 7.25 mg/l were examined in this study. DO standards less than 6.00 mg/l were not analysed because Monte-Carlo simulations carried out for these standards gave the same output results for all simulations. Non-varying simulation output result in all solutions being classified as behaviours and thus do not lend themselves to the GSA technique. Monte-Carlo simulations of the MAUT program for DO standards greater than 7.50 mg/l also

Table 5.3: Months in summer season

Summer Season Length (months)	Months in Summer Season
1	August
2	July - August
3	July - September
4	June - September
6	May - October
10	February - November

produced a large amount of infeasible solutions for some season length combinations. The probability of one or more water quality violations in a given year was set at 10% in accordance with current practice for non-seasonal BOD waste discharge programs. This allowed the same risk of water quality violations (risk equivalence) for both the seasonal and the non-seasonal waste management programs.

The uncertain input data or parameters in the MAUT model include the seasonal stream flows at each gauging station, the temperature values for each season, the river velocity, the reaeration coefficient, the decay coefficient, the benthic oxygen demand, and the stream background BOD and DO deficit concentrations. Other sources of uncertainty are the waste discharge flows and concentrations and the treatment cost coefficients. To reduce the size of this problem only uncertainty in seasonal stream flows at every station and stream temperatures are considered since they are generally the most important input data in terms of the effect on DO in the stream. Also, there is enough data to characterize the uncertainty (i.e., develop probability distributions) in the flow and temperature.

Eighteen reaches were defined for the water quality model for the Willamette River. As shown in Figure 5.1, the confluence of the tributaries and the points of waste discharge of the polluters were used to divide the river into these reaches. The portion of the river downstream of Crown Zellerbach defines Reach 18, and the most upstream segment denotes Reach 1.

5.3.2 Monte-Carlo simulations for the MAUT Program

Probability distributions were fit for the selected seasonal low flow data for each gauging station and the seasonal temperatures for the Harrisburg Station. The station flows and temperatures for both seasons fit the two-parameter lognormal distribution. In order to maintain the same correlation structure as the historical seasonal station

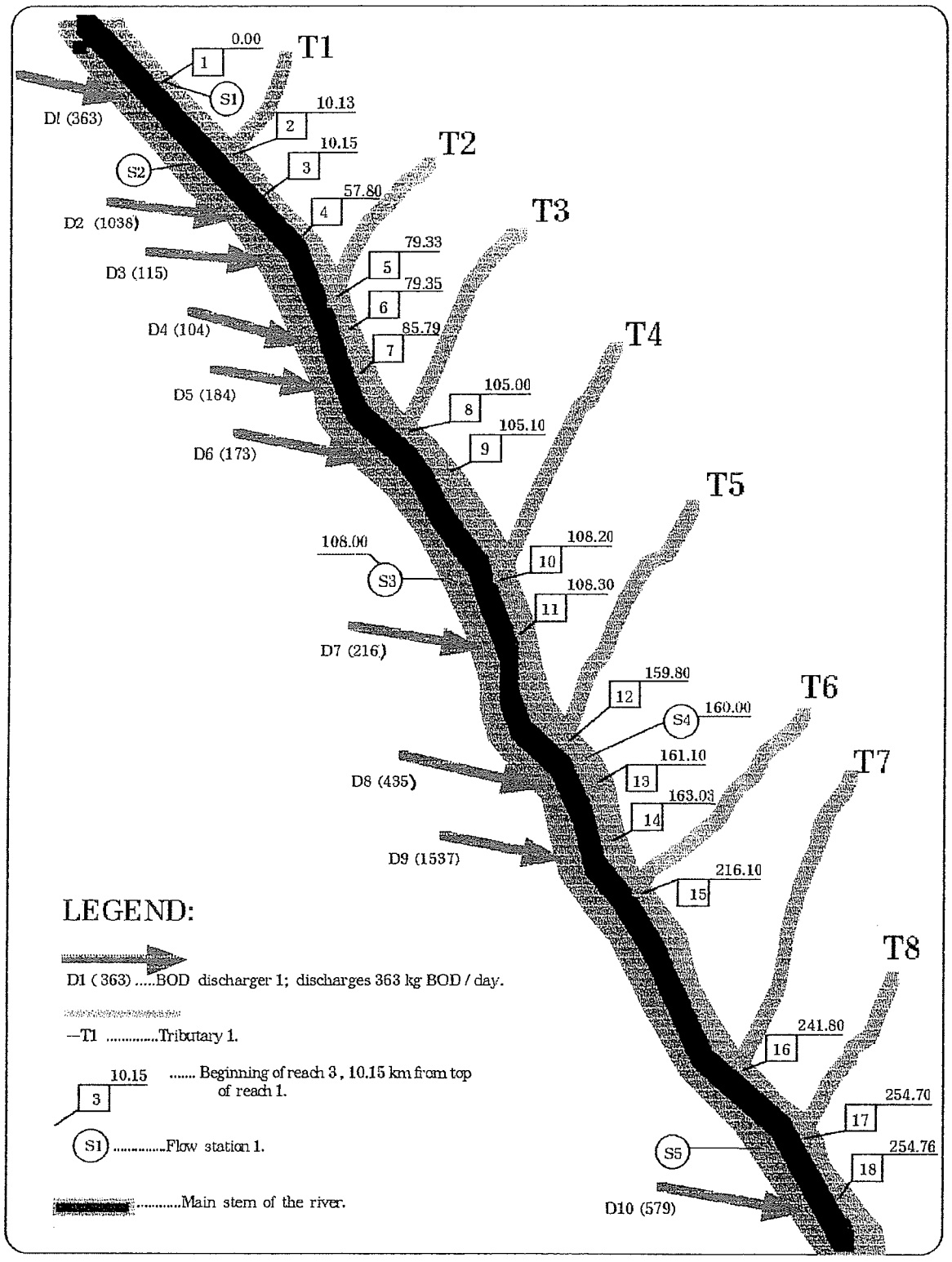

Figure 5.1: Layout of the Willamette River basin

flows and to reduce computational time for the MCS analysis, the flow at Station 3 was used as a basis for generating flows at the other four stations. Station 3 was selected as a base station because its flows were the best predictor of the other station flows. It was also the most central among the five flow stations along the stretch of the Willamette River used for the case study. Seasonal flows for Stations 1, 2, 4, and 5 were regressed against corresponding seasonal flows at Station 3. The mean squared error (MSE) obtained for each of the regression equations is the variance of the predicted flow about the mean response of the dependent station flow. For each season of each year and each Monte-Carlo simulation, the distribution function for Station 3 was used to generate the random flow at Station 3. The flow for Station 3 and the regression equations were used to determine mean response flows, corresponding to the flow at Station 3, for the other four stations. A summary of the regression analysis is given in Appendix A. The MSEs for the regression equations were used to determine a random error about the mean response flows at Stations 1, 2, 4, and 5. The random errors obtained were added to the mean response flows to obtain the random flows for each of the four stations. An additional requirement placed on the generation of flows was that the upstream flow could not be greater than a downstream flow for a given time. Since only one temperature station was used, its seasonal values for any Monte-Carlo simulation were simply generated from its fitted distribution.

Each Monte-Carlo simulation generates thirty years of summer and winter seasonal low flows for each of the five river flow stations, and thirty years of seasonal temperatures for the entire system. Design uniform treatment levels for both seasons are determined based on all thirty years of generated flows and temperatures and the goals of the MAUT program. The seasonal station flows and temperatures which correspond to the design treatment levels are recorded and are considered as the input data to be analysed with the GSA. Therefore the uncertainty analysis was performed

for the design flow and temperature conditions and not on all the generated input data.

Three thousand (3000) Monte-Carlo simulations were performed for each season length combination and DO standard considered. An adequate number of MCS runs was obtained by gradually increasing the number of simulation from 500 until the statistical properties of the input data and output realizations converged for an experimental trial.

5.3.3 Analysis of MCS results

The historical seasonal data were used to obtain deterministic seasonal uniform treatment levels for each season length combination and water quality goal of interest. If there was no uncertainty in the input data, information from these deterministic solutions are what the water quality manager would use to control pollution in the river. The binary classification definition was based on whether the design removal level for a particular Monte-Carlo simulation realization would cause a violation of DO standard or not, assuming the deterministic solution was correct. A simulation output and its corresponding design flows and temperatures were consequently classified as a Behaviour if neither of the removal levels in the summer and winter seasons was less than the corresponding removal levels for the deterministic case. All other simulations were grouped as Non-Behaviours.

The classification described above yielded Behaviour and Non-Behaviour data matrices consisting of seasonal temperature and station flow design conditions. The correlation matrix of the Unclassified design flows and temperatures was determined. The variance-covariance matrices of the Behaviour, Non-Behaviour, and Unclassified design temperatures and flows were also calculated for each management scenario investigated (i.e., each season length combination and DO standard). The modified

GSA technique developed in Chapter 4 was then applied to the design temperature and flow values for each management scenario.

Normalization of K-S statistic

The statistical significance of a d_{mn} value (the K-S statistic) depends on the magnitudes of m and n, the number of behaviours and non-behaviours for the GSA application. The d_{mn}s obtained from the GSA applied to the transformed uncorrelated variables are normalized for comparison of the results from different management scenarios. Normalized d_{mn} values are obtained by dividing the particular d_{mn} by the theoretical value at a selected significance level. The normalized d_{mn} for a given transformed variable Y_t is given by:

$$D_{mn}^{Y_t} = \frac{d_{mn}^{Y_t}}{d_{mn}^{\alpha}} \tag{5.6}$$

where

$d_{mn}^{Y_t}$	=	K-S statistic obtained from the behaviour and non-behaviour classes for variable Y_t;
d_{mn}^{α}	=	theoretical K-S statistic for m behaviours and n non-behaviours at the α significance level; and
$D_{mn}^{Y_t}$	=	normalized $d_{mn}^{Y_t}$.

Substituting the normalized $D_{mn}^{Y_t}$, for $d_{mn}^{Y_t}$ in *Equation* 4.24, the normalized sensitivity index for input datum i is given as:

$$S_i = \sum_{t=1}^{N} D_{mn}^{Y_t} \frac{|a_{ti}|}{\sum_{k=1}^{N} |a_{tk}|} \tag{5.7}$$

In computing the sensitivity indices for the seasonal design temperatures and station flows, the 5% significance level value for the K-S statistic ($d_{mn}^{5\%}$) was used to normalize sample d_{mn} values obtained.

5.3.4 Solution algorithms used

A modified form of the Fortran code developed for the MAUT by Lence et al. (1990) was adapted for the Monte-Carlo simulation analysis. A random number generator on a SUN-OS 486 workstation was used to generate normally distributed random numbers for the simulations. Fortran programs were developed to accomplish the behavioural classification and to obtain the correlation and variance-covariance matrices to construct the transformation matrix **A**. The transformation matrix **A** and the sample d_{mn}s were evaluated with Fortran subroutines from EISPAC's Matrix Eigensystem Routines (Garbow et al., 1977) and from Numerical Recipes (Press et al., 1986). A listing of all computer codes developed for this research are included in Appendix B. Aside from fitting probability distributions to the flows and temperatures, and obtaining regression equations for the station flows, which were performed on PCs, all other analyses were carried out on APPOLO and SUN-OS workstations or on the UNIX network at the University of Manitoba.

Chapter 6

DISCUSSION OF RESULTS

The results of the case study described in Chapter 5 are presented in this chapter. The factors that affect input data sensitivity indices determined with the GSA technique are also identified. The sensitivity results from the Willamette River Basin example are discussed with reference to these factors.

6.1 Simulation and binary classification results

Henceforth, each set of water quality management decisions is referred to as a scenario. For example, the one month summer and eleven month winter season length combination for the 6.00 mg/l DO standard constitutes a scenario. Eighteen scenarios, consisting of six summer and winter season length combinations for each of the three DO standards (6.00, 7.00, and 7.25 mg/l), were analysed.

Summary statistics for the Monte-Carlo simulation output are given in Table 6.1. The statistics listed are the mean, the standard deviation, the maximum, the minimum, and the range of seasonal removal levels. The deterministic solution obtained for the historical seasonal flow and temperature data are also given in the table.

Table 6.1: Summary statistics for simulation output results (removal fraction)

	SUMMER SEASON LENGTH											
	1		2		3		4		6		10	
	W*	S**	W	S	W	S	W	S	W	S	W	S
DO STANDARD=6.00 mg/l												
DS	0.45	0.44	0.37	0.47	0.35	0.43	0.35	0.45	0.35	0.46	0.35	0.46
Mean	0.42	0.39	0.36	0.42	0.35	0.41	0.35	0.42	0.35	0.42	0.35	0.44
SD.	0.023	0.033	0.012	0.032	0.001	0.024	0.001	0.022	0.001	0.022	0.000	0.021
Max.	0.50	0.52	0.43	0.55	0.38	0.50	0.37	0.50	0.38	0.50	0.35	0.51
Min.	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.37
Range	0.15	0.17	0.08	0.20	0.03	0.15	0.02	0.15	0.03	0.15	0.00	0.14
DO STANDARD=7.00 mg/l												
DS	0.67	0.66	0.61	0.66	0.54	0.67	0.44	0.68	0.35	0.68	0.35	0.68
Mean	0.66	0.64	0.58	0.68	0.50	0.68	0.37	0.68	0.35	0.67	0.35	0.68
SD	0.014	0.025	0.020	0.019	0.032	0.018	0.028	0.016	0.003	0.012	0.000	0.013
Max.	0.71	0.73	0.65	0.76	0.61	0.77	0.50	0.74	0.40	0.72	0.35	0.72
Min.	0.62	0.57	0.50	0.61	0.39	0.63	0.35	0.60	0.35	0.63	0.35	0.63
Range	0.09	0.16	0.15	0.15	0.22	0.14	0.15	0.14	0.05	0.09	0.00	0.09
DO STANDARD=7.25 mg/l												
DS	0.73	0.76	0.67	0.74	0.60	0.74	0.49	0.74	0.35	0.74	0.35	0.74
Mean	0.73	0.71	0.65	0.74	0.58	0.74	0.43	0.74	0.35	0.74	0.35	0.74
SD	0.013	0.023	0.017	0.018	0.030	0.017	0.043	0.016	0.006	0.011	0.000	0.012
Max.	0.77	0.80	0.70	0.82	0.67	0.83	0.59	0.80	0.45	0.78	0.35	0.78
Min.	0.69	0.64	0.58	0.68	0.48	0.68	0.35	0.67	0.35	0.70	0.35	0.70
Range	0.08	0.16	0.12	0.14	0.19	0.15	0.24	0.13	0.10	0.08	0.00	0.08

* Winter ** Summer DS Deterministic Solution SD Standard Deviation

The most important of these statistics for describing uncertainty are the standard deviation and the range of the seasonal waste treatment levels. From Table 6.1 it can be seen that these two statistics have a similar pattern, therefore only the range will be used to explain the pattern of uncertainty in the waste treatment levels for different scenarios. For the 6.00 mg/l DO standard the range of removal levels is generally wider for summer than for winter. This suggests a higher level of uncertainty associated with summer removal levels for this standard. There is a trend in the removal levels. As the summer season length increases from one month to three months, the range for winter removal levels decreases from 0.15 to 0.03. It stays fairly constant up to the scenario with the six month summer season, after which it drops to zero for the scenario with the ten month summer season. A range of removal levels equal to zero implies that all the 3000 simulations give the same removal levels for a given season and therefore there is no uncertainty with regard to the design removal level. On the other hand, the range for summer removal levels varies from 0.14 to 0.20 for all scenarios having the 6.00 mg/l standard.

For the 7.00 mg/l standard, the range for winter removal levels increases from 0.09 to a maximum of 0.22 as the summer season length increases from one to three months. It then decreases steadily to a minimum of zero for the scenario with the ten month summer season. The summer removal level range is fairly constant at about 0.15 for the scenarios with summer season lengths from one to four months. It decreases to a minimum of 0.09 for the six month summer season scenario, and then stays constant up to the ten month summer season combination. The 7.25 mg/l DO standard has the same pattern for the winter and summer removal level ranges, except that the winter range peaks at the case with a four month summer season length.

It should be noted that the range in the removal levels alone does not entirely explain the uncertainty in the output of the MAUT Program. There are other impor-

tant concerns such as the variance of the simulated treatment levels. Furthermore, uncertainty in treatment levels does not imply the same level of uncertainty in the design treatment cost.

Although the ranges and standard deviations of the simulation output give an idea of the uncertainty in the model realizations, they do not indicate the sources of these uncertainties. It is natural to assume that uncertainties in winter and summer removal levels are attributed to uncertainties in input data for winter and summer, respectively. However, for the risk equivalent seasonal program analysed here, this assumption is not generally true. For an MAUT Program based on thirty years of data, a 10% risk requirement specifies that only three years of record may exist in which a violation in one or both seasons occurs. This causes the design removal levels in either season to be not only dependent on that season's input data but also on those of the other season. The dependence of one season's removal level on the other season's input data will be examined further in Section 6.3.

The percentage of simulations that compose the behaviour group are shown in Table 6.2. For the 6.00 and 7.25 mg/l standards, the proportion of behaviours decrease from the one month summer season scenario to a minimum at the three month summer season combination. For the 7.00 mg/l DO standard however, the proportion of behaviours reaches a minimum at the two month summer season scenario. For all three DO standards, the percentage of behaviours increase from the minimum values to a maximum for the case with the six month summer season, and drops slightly for the ten month summer scenario. Comparing the number of behaviours across the water quality goals, the 6.00 mg/l DO standard has the highest proportion of behaviours and the 7.00 mg/l DO standard has the lowest proportion of behaviours for all season length combinations.

Table 6.2: Percentage of behaviours

Standard	Summer season Length (months)					
(mg/l)	1	2	3	4	6	10
6.00	89.6	87.6	81.1	92.6	97.2	86.2
7.00	68.0	26.9	33.1	68.2	81.2	76.6
7.25	74.9	66.3	47.5	57.5	78.9	77.7

6.2 Factors affecting the GSA input data sensitivity indices

This research identifies three main factors which affect the values of the sensitivity indices for the model input data. They are:

1. the degree of uncertainty associated with an input datum;
2. the relative importance or significance of the input data in producing the acceptable model output; and
3. the definition of a behaviour used.

The uncertainty in a given input datum is defined by the range, variance, and distribution of that input datum. The wider the range or variance of possible values, the higher the uncertainty and the larger the sensitivity index is expected to be. This is reasonable since the uncertainty in simulating a behaviour or a non-behaviour is propagated from uncertainties in the input data. In fact if there is no uncertainty in an input datum (i.e., only one value is associated with that datum and it is known), the sensitivity index is at its minimum value of zero. This means that the probability distributions for the two behavioural classes are exactly the same for that datum.

The degree to which an input datum affects the model output also contributes to the sensitivity computed with the GSA method. This factor is actually the classical sensitivity index identified with analytical sensitivity analyses. If the gradient of the model output with respect to an input datum is steep within the range of data considered, small changes in this input could result in significant differences in model realizations. Hence, the cumulative probability distributions of the behaviour and non-behaviour classes for input data of higher relative importance to model output

will be very different and will result in a large d_{mn} statistic. On the other hand, input data which do not significantly affect model output will have small K-S two sample statistics. If model response does not change significantly with respect to a given input datum, obtaining a model realization based on this input datum can be regarded as a random process and the probability distributions of the behavioural classes can be assumed to come from the same parent distribution.

The qualitative binary classification definition on which the GSA approach is based evaluates the input data sets according to whether they affect the important system characteristics of the problem at hand. It is therefore expected to affect the sensitivity values of the uncertain input data and their relative importance. Different input data affect the model response behaviour differently. For example, while a given input parameter may affect initial values of a time series model response, another may affect the mean of the output series. Input data sensitivities obtained for such a model with the technique developed in this work will therefore be different for classification definitions based on initial system characteristics or based on mean system response or both.

6.3 Results of the modified GSA

Since the sensitivity indices were normalized by dividing the sample values by the 5% significance value of the d_{mn}, a sensitivity index of 1.00 for a given input data indicates that this input data is significant in simulating the model output at 95% confidence. On the other hand, a sensitivity index less than 1.00 means the particular input datum is not important in simulating a Behaviour or a Non-Behaviour at 95% confidence. The magnitude of the sensitivity indices that significantly affect the model output has not been determined. Sensitivity indices calculated are therefore used only for ranking the importance of input data uncertainty to model output uncertainty for

the same type of model. It is cautioned that sensitivity indices obtained for different types of models using the modified GSA technique developed in this thesis should not be compared because of lack of knowledge about the magnitude of the indices that constitute a significant value.

The sensitivity indices for the stream temperature and the design flows for all stations in summer and winter were calculated for each season length combination and water quality goal. Figures 6.1, 6.2, and 6.3 show the sensitivity indices for the temperatures and the mean of the sensitivity indices for the station flows for winter and summer for the 6.00, 7.00, and 7.25 mg/l DO standards, respectively. From Figure 6.1, it may be observed that the winter sensitivity indices generally decrease as the summer season length increases from one month to four months for the 6.00 mg/l DO standard, after which they stay approximately constant. The 7.00 and 7.25 mg/l standards have a similar pattern for winter sensitivity indices, except that the sensitivity indices stay approximately constant after the six month summer scenarios. The summer input data sensitivity indices increase steadily to a maximum at a summer season length of three or four months for all DO standards analysed. Then they drop slightly and stay fairly constant up to the ten month summer season scenario.

Naturally, it is expected that winter input data sensitivities should decrease as the summer season length increases. The importance of the winter input data in determining adequate design treatment levels decreases as the winter season length decreases. This is because the proportion contribution of winter waste treatment cost to the total annual treatment cost decreases with increases in summer season length due to shorter winter season length and lower winter removal levels. Similarly, the increasing sensitivity indices of the summer input data can be explained by the increased importance of summer treatment cost to the total annual waste removal cost. However, the design winter and summer removal levels do not depend solely

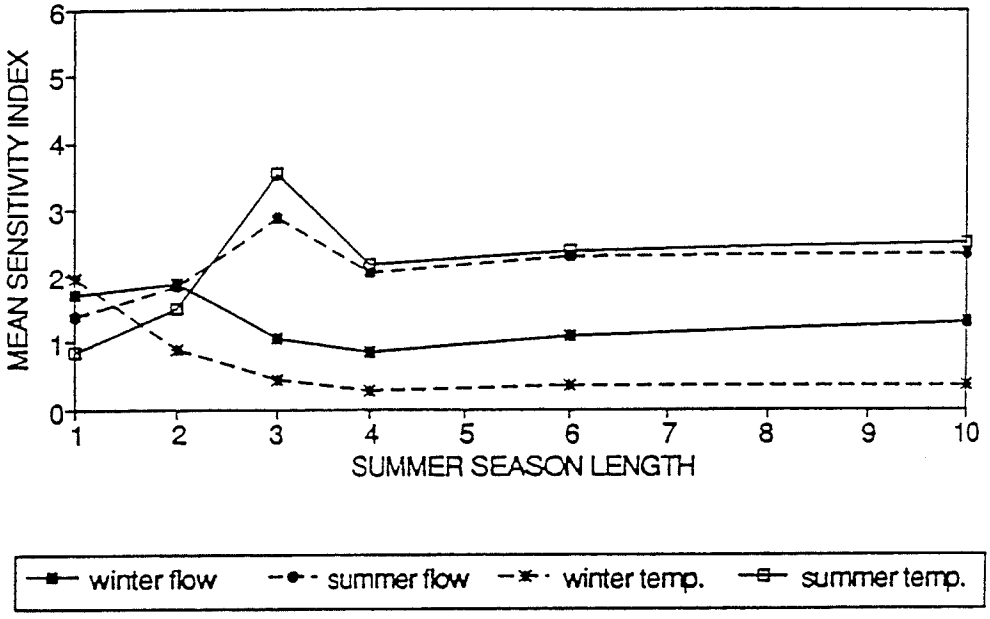

Figure 6.1: Seasonal temperature and mean of station flow sensitivity indices (6.00 mg/l)

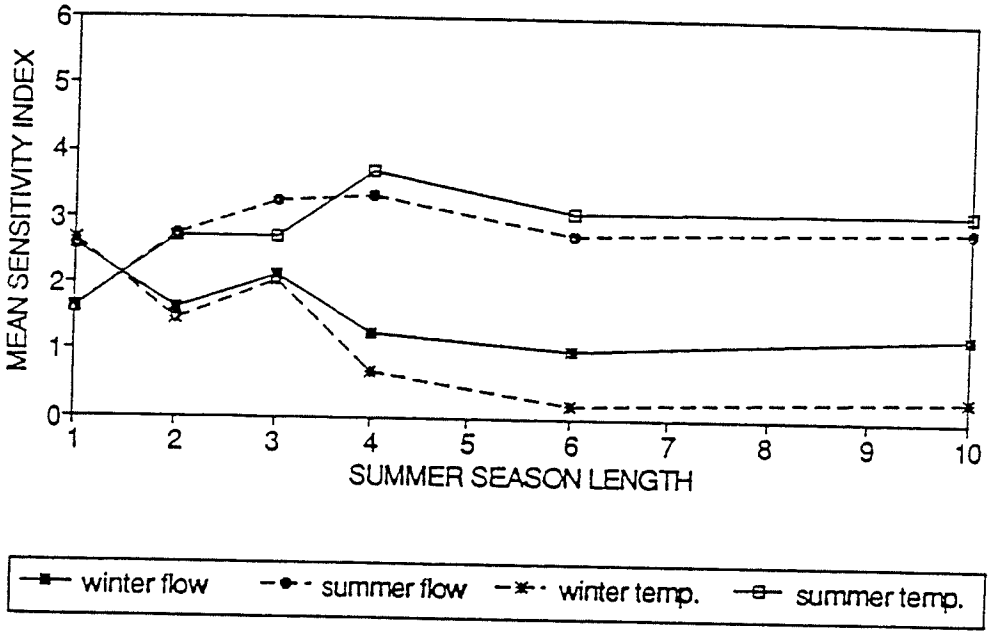

Figure 6.2: Seasonal temperature and mean of station flow sensitivity indices (7.00 mg/l)

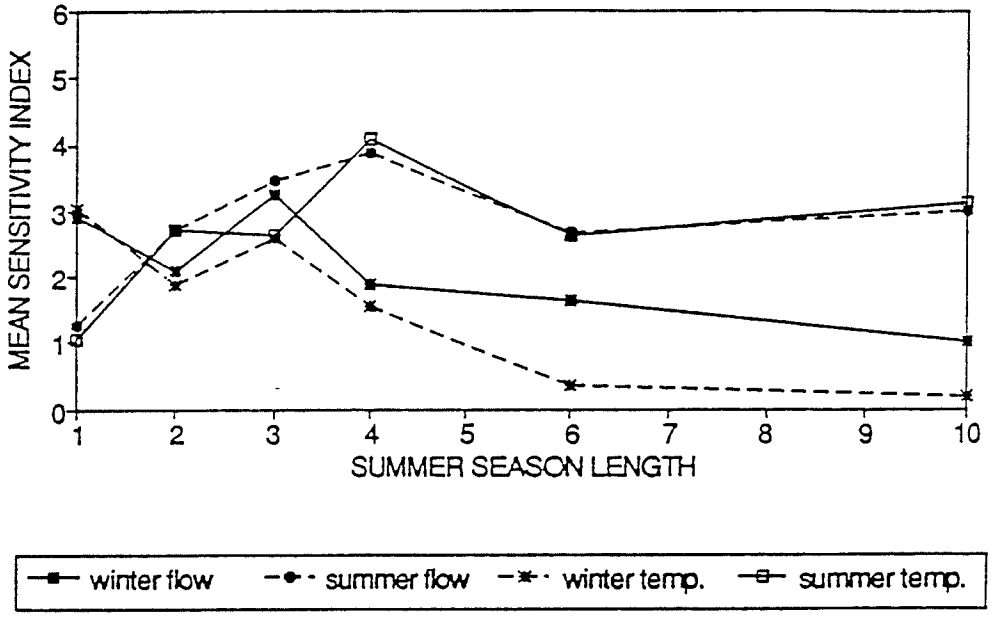

Figure 6.3: Seasonal temperature and mean of station flow sensitivity indices (7.25 mg/l)

on the winter and summer input data, respectively, because the constraint on risk relates the design winter removal level to the design summer removal level for each Monte-Carlo simulation.

The values of the summer and winter input data sensitivity indices are approximately constant for the scenarios with six and ten month summer seasons. For these cases, all of the months with summer characteristics have been removed from the winter season and the sensitivity exhibited is due to the level of uncertainty associated with data for the true summer and winter months. This observation indicates that once the summer and the winter seasons have been adequately separated, the sensitivity of the MAUT Program to winter data is reduced to its minimum and both summer and winter sensitivity indices are stabilized.

For scenarios with summer season lengths of six and ten months, the sensitivity indices of the summer flows and temperature data remain approximately constant and high. Since summer removal levels are high, these removal options are located on steeper portions of waste treatment cost curves. The seasonal design removal levels are more sensitive to input data on the steep segments of the cost curve and result in high summer sensitivity indices for scenarios with summer season lengths greater than six months.

After all months with summer characteristics have been effectively removed from the winter season, the mean winter flow increases rapidly. The high winter flows and low temperatures result in low winter removal levels. These low removal levels correspond to the mildly sloped portions of the waste treatment cost curve (i.e., the less sensitive portions of the cost curve). The sensitivity indices obtained for the winter input data therefore stay fairly low and constant when all true summer months are removed from the winter season.

Explanations for trends in winter and summer temperature sensitivities are sim-

ilar to those for flow given above. For example, the mean temperature decreases significantly after the four month summer season scenario, raising the assimilative capacity of the river and allowing low waste treatment levels. The low sensitivity index of winter temperature is due to the reduced sensitivity associated with low waste treatment levels.

As shown in Table A.3, the variances of the seasonal flows and temperatures have different trends than the seasonal sensitivity indices for flow and temperature described in Figures 6.1, 6.2, and 6.3. For example, winter flows and temperatures always have higher variance than the corresponding summer data, yet the GSA generally produced lower sensitivity indices for winter input data. Furthermore, the trends and the magnitudes of the sensitivity indices for the station flows and temperatures are quite consistent. These two factors suggest that for the Willamette River Basin the management decisions of the MAUT have a greater effect on sensitivity indices than the uncertainty in the input data themselves.

A plot of the mean of the sensitivity indices for all twelve input data (flows and temperatures in both seasons) versus the summer season length is shown in Figure 6.4. For each DO standard the mean sensitivity indices generally reach a maximum at the three month summer season scenario. They then decrease steadily to attain minimum values at the four or six month summer season scenario, after which they remain fairly constant. In general, the input data sensitivities increase with an increase in the DO standard. This is expected since a higher waste removal level in both seasons is required to meet the water quality goal as the DO standard is increased. As explained earlier, these high treatment levels lie on the steep portions of the treatment cost function and produce results which are more sensitive to flow and temperature than low removal levels. Figure 6.4 shows that the difference between the mean of the input data sensitivity indices for the different DO standards is large for summer season lengths between two and six months. This indicates that the uncertainty in

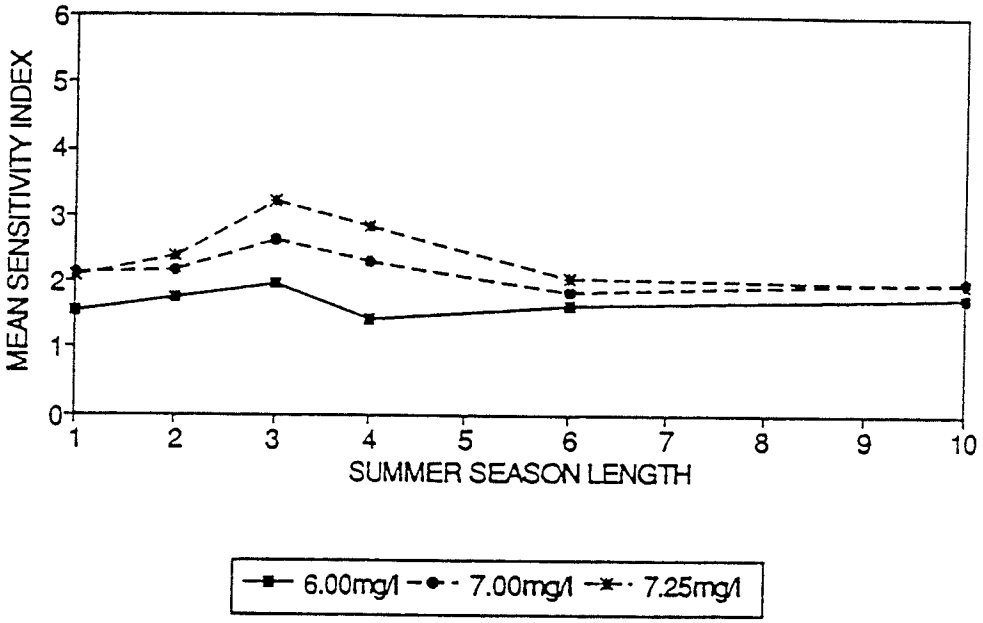

Figure 6.4: Mean of input data sensitivity indices (6.00, 7.00, and 7.25 mg/l)

the results of the MAUT model for the Willamette River are more sensitive to the water quality goal for summer season lengths between two and six months than for the other season length combinations studied in this work.

The individual sensitivity indices for each of the station flows and the stream temperature are shown in Figures 6.5 to 6.10. Figures 6.5, 6.7, and 6.9 show sensitivity indices for summer data and Figures 6.6, 6.8, and 6.10 show those for winter data for the 6.00, 7.00, and 7.25 mg/l DO standards, respectively. Sensitivity indices obtained for summer flows for all season length combinations, and for winter flows for scenarios with summer season lengths less than six months are generally high (mostly greater than 1.00).

For the scenarios with short summer seasons, the design flows in the summer and winter are generally low. Under these scenarios, as discussed before, the winter season selected includes months with typical summer flow characteristics. Figures 6.5 to 6.10 demonstrate that for such scenarios, flows at the three most downstream gauging stations, Stations 3, 4, and 5, have relatively higher sensitivity indices than those at the two most upstream stations.

For summer season lengths longer than than six, four, and three months for the 7.25, 7.00, and 6.00 mg/l DO standards, respectively, the winter flow sensitivity indices are low (generally less than 1.00), and winter flows for gauging Stations 3, 2, and 4 are more important than those for the other two stations. It is interesting to note that Station 5, which is identified as one of the most important stations when design flows are low, is no longer significant when summer season lengths increase and the winter design flows are high.

The patterns observed for station flow sensitivity indices can be explained by the magnitude of the low flow for a given season, the position of the critical point for the DO sag curve relative to the gauging stations, and the sensitivity of the DO deficit

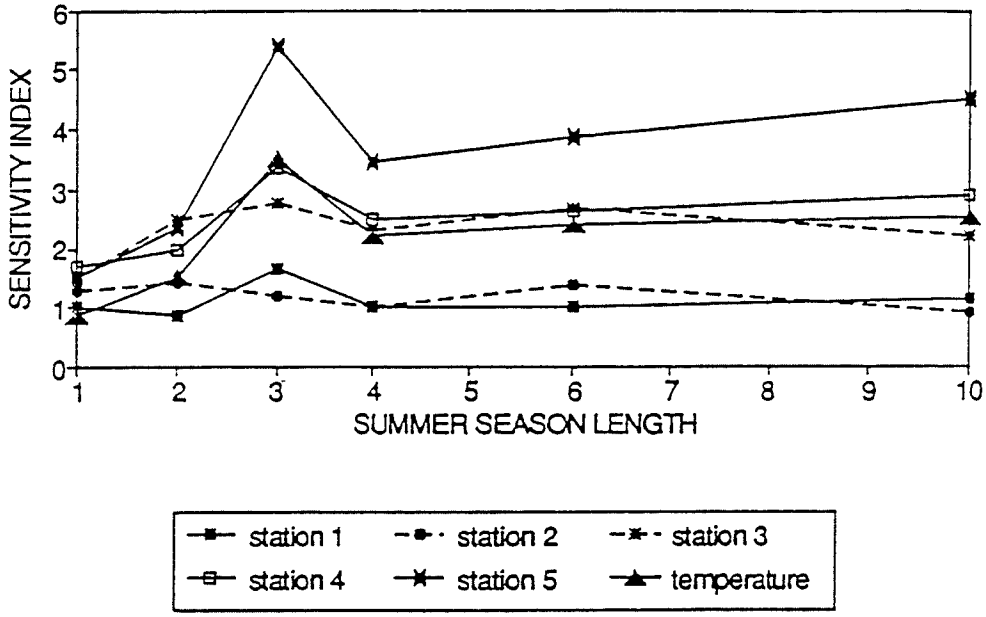

Figure 6.5: Summer temperature and station flow sensitivity indices (6.00 mg/l)

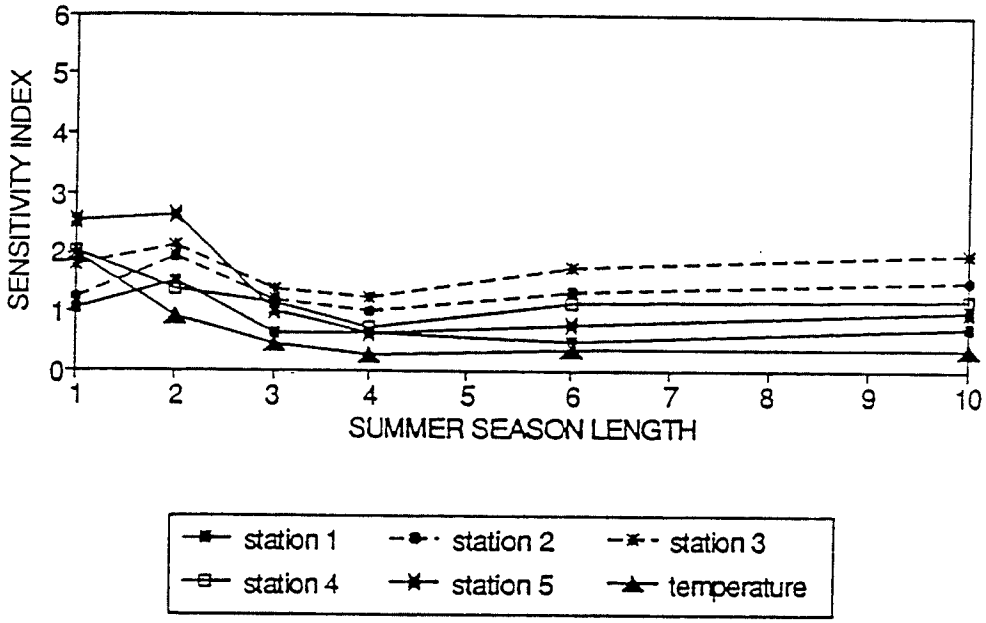

Figure 6.6: Winter temperature and station flow sensitivity indices (6.00 mg/l)

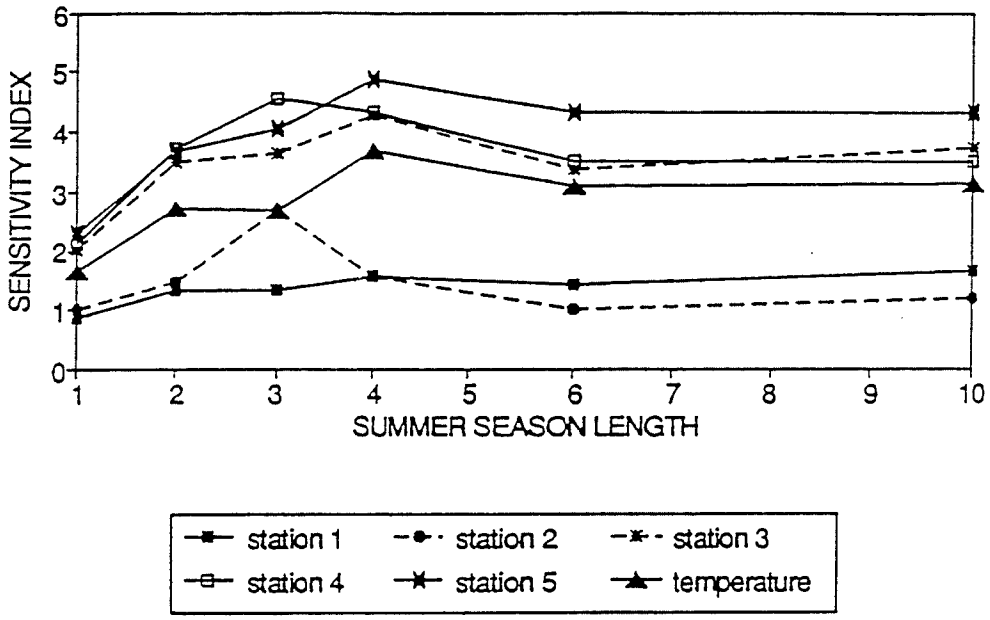

Figure 6.7: Summer temperature and station flow sensitivity indices (7.00 mg/l)

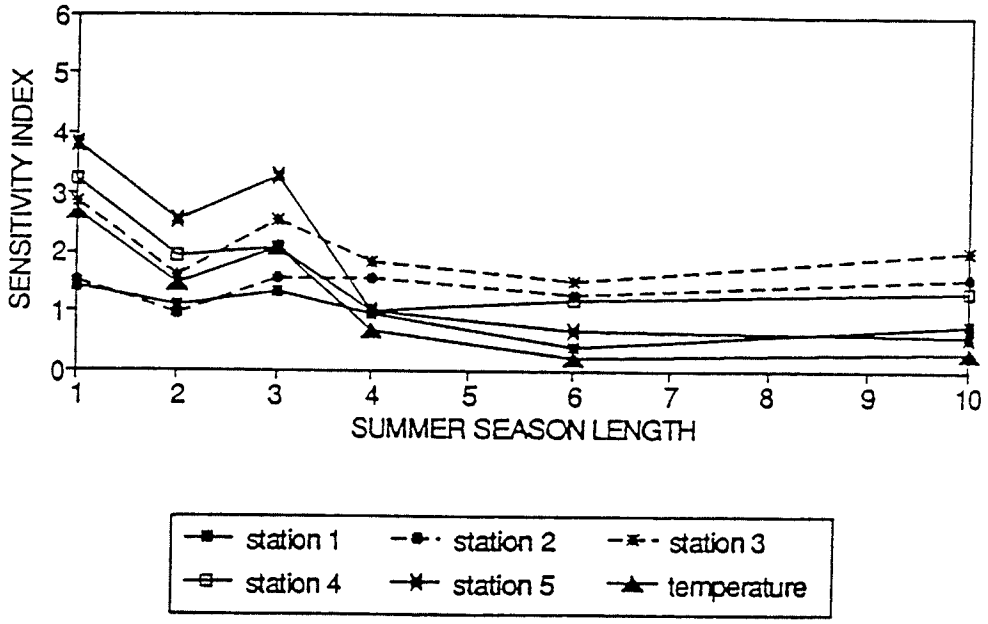

Figure 6.8: Winter temperature and station flow sensitivity indices (7.00 mg/l)

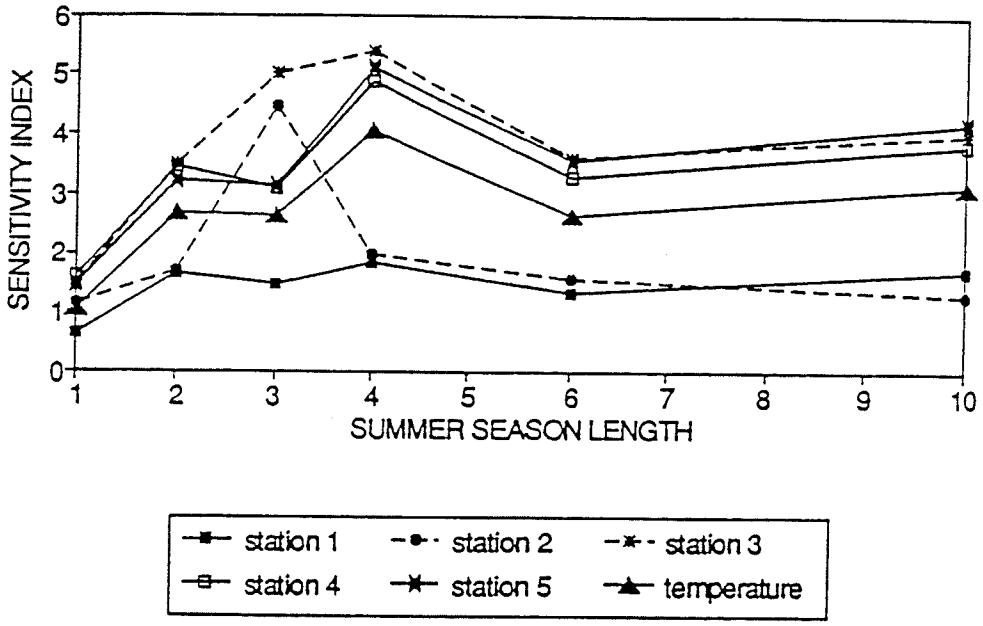

Figure 6.9: Summer temperature and station flow sensitivity indices (7.25 mg/l)

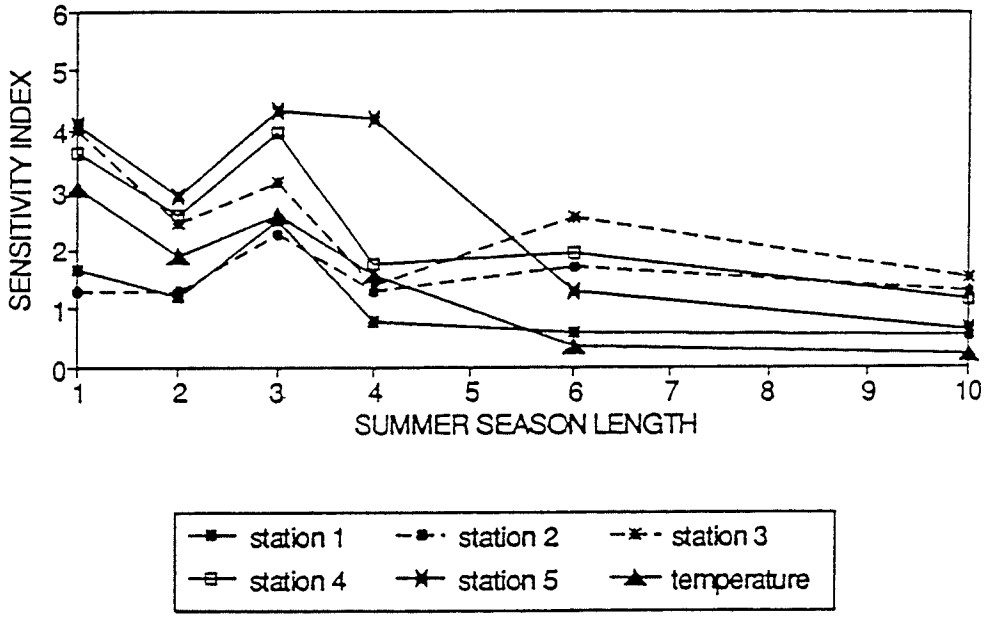

Figure 6.10: Winter temperature and station flow sensitivity indices (7.25 mg/l)

at this critical point to the flow magnitude. The critical DO point for the Willamette River always occurs downstream of Station 5 (this was also observed by Lence et al., 1990 and Kilgore, 1985). Hence when flows are low, upstream BOD waste takes a long time to travel to the critical point. The range of DO deficit contributed by upstream dischargers is therefore low compared with that for dischargers close to the critical point. Also, flows at stations near the critical point are generally more important for determining design uniform treatment levels than those at stations far away from the critical point. However, when flows are high, the upstream velocities are high and the waste is carried quickly downstream creating a relatively large DO deficit impact at the critical point. This decreased residence time is manifested in an increase in the relative importance of the upstream gauging stations during high flow periods (see Eheart, 1988 for an analysis of the decreased residence time effect).

The effect of flow magnitude on the sensitivity indices can be clearly demonstrated with winter and summer flow patterns obtained for Stations 1 and 5 for the 7.00 mg/l DO standard (see, Figures 6.11 and 6.12). Flows at Station 1 are the least important flows in both seasons for all season length combinations analysed. The low magnitude of flows at Station 1 does not significantly affect the DO deficit at the critical point, and therefore the sensitivity indices obtained for both seasons are generally lower than those computed for other stations. The sensitivity index for winter flows at Station 5 decreases from a high value to a very low value as the summer season length increases. This drastic drop from being important to being insignificant is the result of the large increase in the magnitude of the winter low flow at Station 5. Winter flows at Station 5 become so high that uncertainty in the value does not greatly affect the design removal levels and the associated cost.

Summer temperatures and the flows at Stations 3, 4, and 5 are very important to the model outcome. The water quality manager should therefore be concerned with these input data. In fact, if more data are going to be obtained to update the model

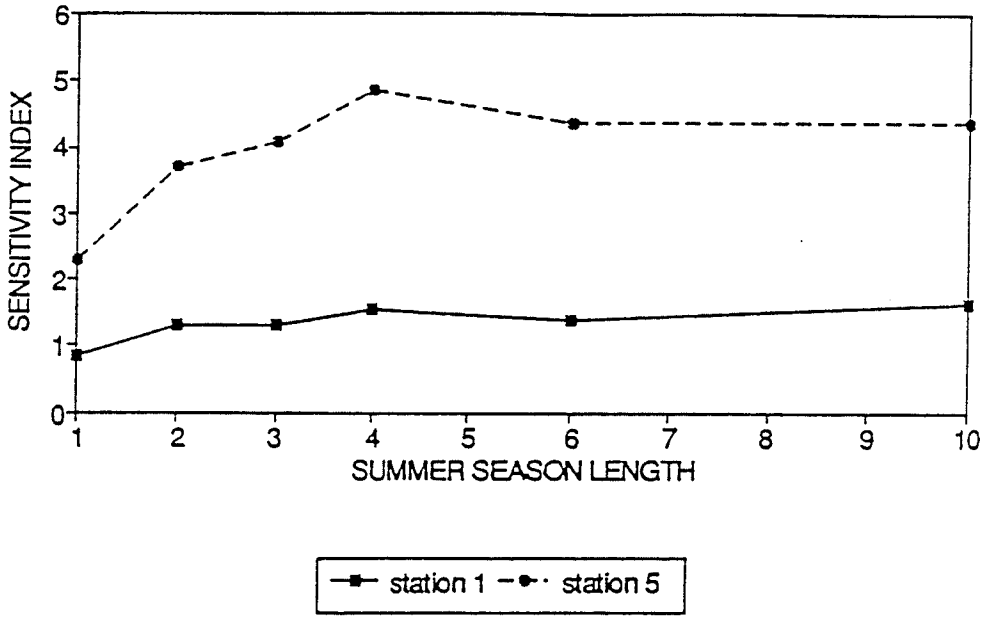

Figure 6.11: Summer flow sensitivity indices for Stations 1 and 5 (7.00 mg/l)

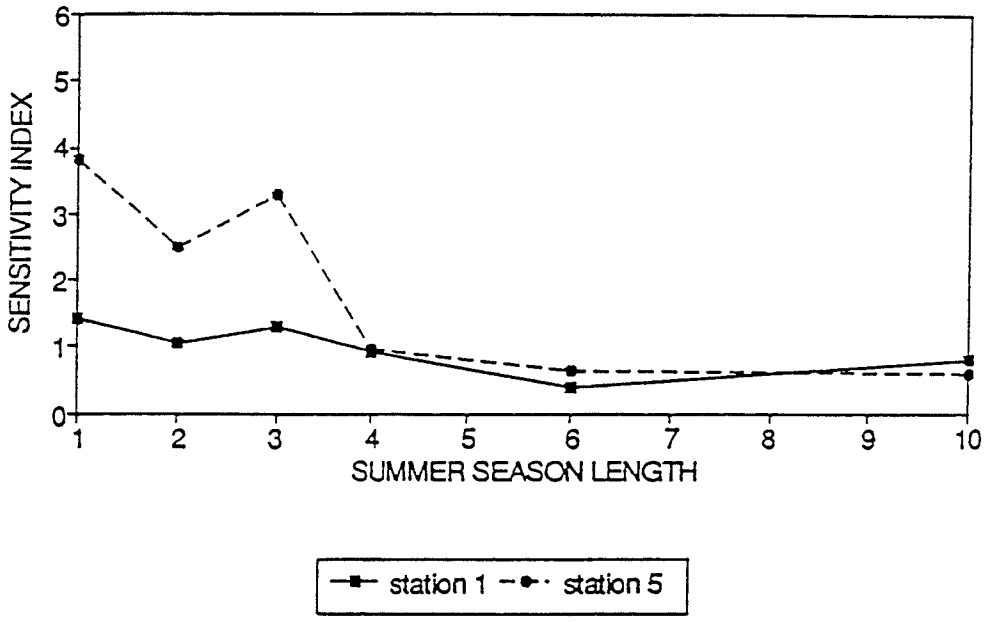

Figure 6.12: Winter flow sensitivity indices for Stations 1 and 5 (7.00 mg/l)

output, then more attention should be paid to temperature records and flow data at Stations 3, 4, and 5 from May to October, followed by flow data at Stations 1 and 2 from May to October. The winter and summer input data sensitivity indices (see Figures 6.5 to 6.10) indicate that these months form the "true" summer period for BOD waste management for the river basin studied.

Lence et al. (1990) and Kilgore (1985) show that, for a two-season discharge schedule, the optimal length of the summer season with respect to cost efficiency ranges from two to five months, depending on the water quality goal. Figure 6.13 shows the annual waste treatment cost for the deterministic solutions obtained from the historical data used for this work. It can be seen that the pattern of waste treatment cost obtained is similar to that reported by Lence et al. (1990) and Kilgore (1985). The minimum annual cost for the 6.00 mg/l DO standard occurred at the three month summer season scenario and it occurred at the four month summer season scenario for the 7.00 and 7.25 mg/l DO standards.

The mean input data sensitivities obtained in this research (see Figure 6.4) indicate that for each DO standard, a potential trade-off exists between the annual waste treatment cost (see Figure 6.13) and model output uncertainty. For example, the minimum sensitivity index for the 7.00 mg/l DO standard occurs at the six month summer season length while the four month summer season scenario had the minimum annual treatment cost. This finding is interesting since it cautions the environmental system manager not to consider only economic efficiency and environmental quality goal, but also the uncertainty in the management model output, when formulating management policies.

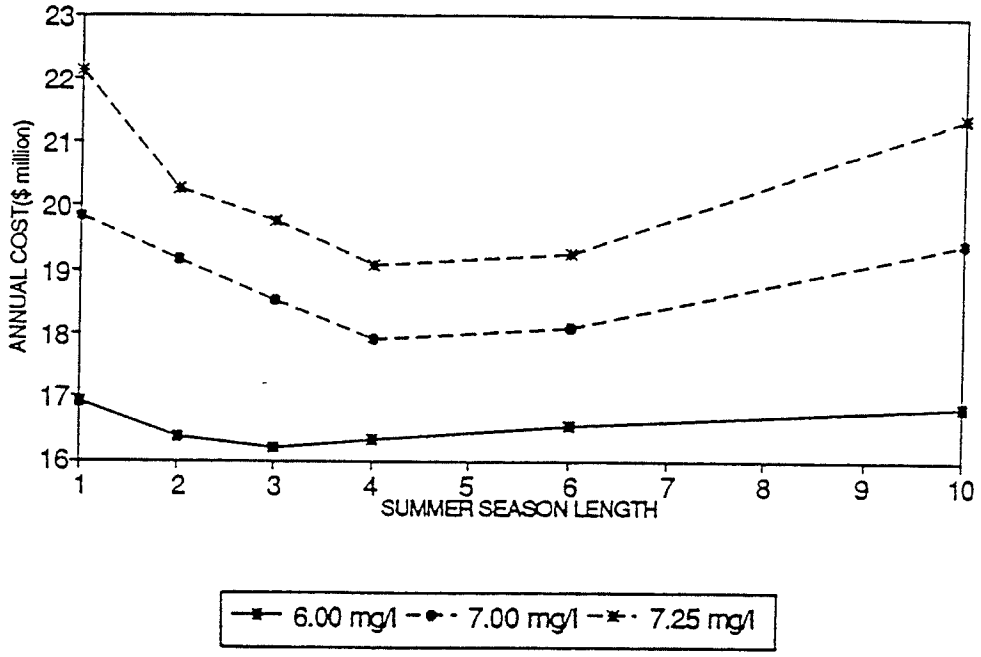

Figure 6.13: Total annual cost of deterministic solutions (6.00, 7.00, and 7.25 mg/l)

Chapter 7

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

In this research, an approach is developed for evaluating the contribution of input data uncertainty to output uncertainty for an optimization based environmental management model. The approach modifies the GSA technique to account for correlated input data. It also modifies the binary classification definition for simulation based models which may be compared to actual system measurements to make the approach useful for optimization based models which do not have actual system measurements. These two aspects are combined and demonstrated for a risk equivalent seasonal discharge program for the Willamette River in Oregon.

It has been shown that the uncertainty in the output for the Minimum Average Uniform Treatment Risk Equivalent Seasonal Discharge Program for the Willamette River increases with increases in the water quality goal. The results also reveal the importance of summer flows and temperatures in the design of a seasonal waste allocation program for the Willamette River. The sensitivity indices for the various input data, combined with information about the quality of that input data, would give a

water quality manager insight into the confidence that can be placed on decisions obtained from a management model. For example, for the case study analysed, if the water quality manager has confidence in summer flow and temperature data but not in those for winter, results from the management model can be utilized as a guide for managing the water quality of the Willamette River Basin. On the other hand, if the summer data used are unreliable, then the manager has to be very concerned with using the model output as an aid for managing the water quality.

It has been demonstrated that flow and temperature data for months having true summer characteristics are more important than flow and temperature data for the other months to the design of treatment levels for the MAUT Program for the Willamette River. This observation is reasonable since the true summer season has more critical water quality conditions than the winter season for this case study. The high sensitivity indices obtained for Stations 3, 4, and 5 for seasons which include the months of May through October indicate that flow data for these months significantly affect the output of the seasonal program analysed. Water quality management decisions for the Willamette River based on the MAUT Program should therefore be more concerned with summer flow data downstream of Station 2, the Harrisburg station, than with other flow records. It has been shown that for each DO standard, a potential trade-off exists between total annual treatment cost and model output uncertainty.

The question of which value of the sensitivity index is significant for a given model can be answered by comparing the sensitivity indices computed for the uncertain input data for that model. At this stage, the sensitivities estimated with the approach developed here should be used exclusively for comparing sensitivity indices for a single model only. Comparison of sensitivity values calculated for different models using this method should be avoided until a more complete understanding of the technique and the sensitivity indices have been gained.

The effect of other uncertain input data such as velocity, reaeration, and decay coefficients on model output uncertainty under the MAUT Program should be investigated using the modified GSA approach. Research should also be conducted on the effect of fitting other distribution functions to the uncertain input data and the use of other techniques such as disaggregation to generate the seasonal station flows.

Further research on the GSA techniques should be conducted to assess the relevance of the magnitudes of sensitivity indices obtained, so that they can be effectively compared for different models. Work should also be focussed on how this technique can be used or modified to design robust management policies for environmental systems. The technique should be applied to other types of environmental management programs and to different behavioural definitions such as those that do not allow a certain level of over-design.

REFERENCES

[1] Atkins, R. P., D. M. Gordon, A. J. McComb, and J. S. Pate, Botanical contributions to the Peel/Harvey estuarine study. Preliminary report of results obtained during the first year of the study. Dept. of Botany, University of Western Australia, Perth, 1977.

[2] Beck, M.B., Operational estimation and prediction of nitrification dynamics in an activated sludge process, *Water Res., 15,* 1313-1330, 1981.

[3] Beck, M.B., Uncertainty, system identification and the prediction of water quality, in *Uncertainty and Forecasting of Water Quality,* edited by M. B. Beck and G. van Straten, pp. 3-68, Springer Verlag, New York, 1983.

[4] Beck, M.B., Water quality modeling: A review of the analysis of uncertainty, *Water Resour. Res., 23(8),* 1393-1442, 1987.

[5] Berthouex, P. M., and L. B. Polkowski, Optimal waste treatment plant design under uncertainty, *Jour. Water Pollution Control Federation, 42(9),* 1589-1613, 1970.

[6] Boner, M. C., and L. P. Furland, Seasonal treatment and variable effluent quality based on assimilative capacity, *Jour. Water Pollution Control Federation, 54(10),* 1408-1416, 1982.

[7] Burges, S. J., and D. P. Lettenmaier, Probabilistic methods in stream quality management, *Water Resour. Bull., 11,* 115-130, 1975.

[8] Burn, D. H., and E. A. McBean, Optimization modeling of water quality in an uncertain environment, *Water Resour. Res., 21(7),* 934-940, 1985.

[9] Chadderton, R. A., A. C. Miller, and A. J. McDonnell, Uncertainty analysis of dissolved oxygen model, *Proc. Am. Soc. Civ. Eng. Jour. Environ. Eng. Div., 108*(EE5), 1003-1013, 1982.

[10] Chang, T. M., *Sensitivity analysis in optimal process design,* Ph.D. Dissertation, Chemical Engineering Dept., West Virginia University, Morgantown, W. Va., 1967.

[11] Eheart, J. W., Effects of streamflow variation on critical water quality for multiple dischargers of decaying pollutants, *Water Resour. Res., 24(1),* 1-8, 1988.

[12] Eheart, J. W., E.D. Brill, Jr., B. J. Lence, J. D. Kilgore, and J. G. Uber, Cost efficiency of time-varying discharge permit programs for water quality management, *Water Resour. Res., 23(2),* 245-251, 1987.

[13] Fedra, K., A Monte-Carlo approach to estimation and prediction, in *Uncertainty and Forecasting of Water Quality,* edited by M. B. Beck and G. van Straten, pp. 259-291, Springer Verlag, New York, 1983.

[14] Ferrara, R. A., and M. A. Dimino, A case study analysis for seasonal nitrification: economic efficiency and water quality preservation, *Jour. of Water Pollution Control Federation, 57(7),* 763-769, 1985.

[15] Fukunaga, K., and W. L. G. Koontz, Application of the Karhunen-Loeve expansion to feature selection and ordering, *IEEE Transactions on Computers, C-19(4),* 1970.

[16] Fukunaga, K., *Introduction to Statistical Pattern Recognition,* Academic Press, New York, 1972.

[17] Garbow, B. S., J. M. Boyle, J. J. Dongarra, C. B. Moler, *Lecture notes in computer science: Matrix eigensystem routines - EISPACK guide extension,* edited by G. Goos and J. Hartmanis, Springer-Verlag, Berlin, Heidelberg, New York, 1977.

[18] Gardner, R. H., and R. V. O'Neill, Parameter uncertainty and model predictions: A review of Monte-Carlo results, in *Uncertainty and Forecasting of Water Quality,* edited by M. B. Beck and G. van Straten, pp. 245-257, Springer Verlag, New York, 1983.

[19] Gardner, R. H., R. V. O'Neill, J. B. Mankin, and D. Kumar, Comparative error analysis of six predator-prey models, *Ecology, 61(2),* 323-332, 1980.

[20] Gere, J. M., and W. Weaver, Jr., *Matrix algebra for engineers* Litton Educational Publishing, Inc., New York, 1965.

[21] Hargett, D., and J. A. Seagraves, *Benefits and costs of seasonal effluent permits in North Carolina.* North Carolina Water Resource Research Institute, Raleigh, N.C. 1979.

[22] Hornberger, G. M., Uncertainty in dissolved oxygen prediction due to variability in algae photosynthesis, *Water Res., 14,* 355-361, 1980.

[23] Hornberger, G. M., and B. J. Cosby, Evaluation of model of long-term response of catchments to atmospheric deposition of sulphate, in *Identification and System Parameter Estimation,* edited by H. A. Barker and P. C. Young, pp 229-234, Pergamon, New York, 1985a.

[24] Hornberger, G. M., and B. J. Cosby, Selection of parameter values in environmental models using sparse data: A case study, *Appl. Math. Comput., 17(4),* 335-355, 1985b.

[25] Hornberger, G. M., and R. C. Spear, Eutrophication in Peel Inlet I, Problem-defining behaviour and a mathematical model for the phosphorous scenario, *Water Res., 14,* 29-42, 1980.

[26] Hornberger, G. M., and R. C. Spear, An approach to the preliminary analysis of environmental systems, *Jour. Environ. Manage., 12* 7-18, 1981.

[27] Hornberger, G. M., and B. J. Cosby, and J. N. Galloway, Modeling the effects of acid deposition: Uncertainty and spatial variability in estimation of long-term sulfate dynamics in a region, *Water Resour. Res., 22(8),* 1293-1302, 1986.

[28] Humphries, R. B., G. M. Hornberger, R. C. Spear, and A. C. McComb, Eutrophication in Peel Inlet III, A model for the nitrogen scenario and a retrospective look at the preliminary analysis, *Water Res., 18(4),* 389-395, 1984.

[29] Kilgore, J. D., Seasonal static transferable discharge permits for the control of biochemical oxygen demand in the Willamette River, *Working Paper No. 2, NSF Award PRA 81-21692,* Dept. of Civ. Eng., University of Illinois, Urbana, III, 1985.

[30] Kite, G. W., *Frequency and risk analysis in hydrology,* Water Resources Publications, Littleton, Colorado, 1988.

[31] Kohberger, R. C., D. Scavia, and J. W. Wilkinson, A method for parameter sensitivity analysis in differential equation models, *Water Resour. Res., 14(1),* 25-29, 1978.

[32] Lence, B. J., Dynamic transferable discharge permits for the control of biochemical oxygen demand, *Working Paper No. 1,* Dept. of Civ. Eng., University of Illinois, Urbana, 1985.

[33] Lence, B. J., J. W. Eheart, and E. D. Brill, Jr., Risk equivalent seasonal discharge programs for multidischarger streams, *Proc. Am. Soc. Civ. Eng. Jour. Water Resour. Planning and Manage. Div, 116(2),* 170-186, 1990.

[34] Lettenmaier, D. P., and J. E. Richey, Use of first order analysis in estimating mass balance errors and planning sampling activities, in *Theoretical Systems Ecology: Advances and Case Studies,* edited by E. Halfon, pp 80-106, Academic, Orlando, Fla., 1979.

[35] Liebman, J. C., *The optimal allocation of stream dissolved oxygen resources,* thesis presented to Cornell University at Ithaca, N.Y., in partial fulfillment of the requirements for the degree of Doctor of Philosophy, 1965.

[36] Malone, R. F., H. Saidi, and K. Wegener, Predictive accuracy determination applied to a linear model of phosphorous loading resulting from urban runoff, *Appl. Math. Modell., 8,* 81-88, 1984.

[37] O'Neill, R. V., R. H. Gardner, and J. B. Mankin, Analysis of parameter error in a nonlinear model, *Ecol. Modell., 8,* 297-311, 1980.

[38] Ponnambalam, K., E. A. McBean, and T. E. Unny, Impacts of meteorological variations on acid rain abatement decisions, *Jour. of Environ. Eng., ASCE, 116(6),* 1063-1075, 1990.

[39] Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., *Numerical Recipes: The Art of Scientific Computing,* Cambridge University Press, Cambridge, 1986.

[40] Reckhow, K. H., The use of a simple model and uncertainty analysis in lake management, *Water Resour. Bull., 15,* 601-611, 1979.

[41] Reheis, H. F., J. C. Dozier, D. M. Word, J. R. Holland, Treatment costs savings through monthly variable effluent limits, *Jour. Water Pollution Control Federation, 54(8),* 1224-1230, 1982.

[42] Rossman, L. A., Risk equivalent seasonal waste load allocation, *Water Resour. Res., 25(10),* 2083-2091, 1989.

[43] Scavia, D., Uncertainty analysis of a lake eutrophication model, Ph.D. dissertation, University of Mich., Ann Arbor, 1980.

[44] Scavia, D., R. P. Canale, W. F. Powers, and J. F. Moody, Variance estimates for a dynamic eutrophication model of Saginaw Bay, Lake Huron, *Water Resour. Res., 17(4),* 1115-1124, 1981a.

[45] Scavia, D., W. F. Powers, R. P. Canale, and J. L. Moody, Comparison of first-order error analysis and Monte-Carlo simulation in time-dependent lake eutrophication models, *Water Resour. Res., 17(4),* 1051-1069, 1981b.

[46] Spear, R. C., and G. M. Hornberger, Eutrophication in Peel Inlet, II, Identification of critical uncertainties via generalised sensitivity analysis, *Water Res., 14,* 43-49, 1980.

[47] Spear, R. C., and G. M. Hornberger, Control of DO level in a river under uncertainty, *Water Resour. Res., 19(5),* 1266-1270, 1983.

[48] Song, Q., and L. C. Brown, DO model uncertainty with correlated inputs, *Jour. of Environ. Eng., ASCE 116(6),* 1164-1180, 1990.

[49] Tang, C. C., E. D. Brill, Jr., and J. T. Pfeffer, Comprehensive model of activated sludge wastewater treatment system, *Jour. of Environ. Eng., ASCE, 113(5),* 952-969, 1987.

[50] Tarrer, A. R., C. P. L. Grady, Jr., H. C. Lim., and L. B. Koppel, Optimal activated sludge design under uncertainty, *Jour. of the Environ. Eng. Div., ASCE, 102*(EE3), 657-673, 1976.

[51] Thomas, G., and J. DaCosta, A sample survey of corporate operations research, *Interfaces, 9,* 102-111, 1979.

[52] Tumeo, M. A., and G. T. Orlob, Stochastic methods for environmental quality modeling, *Proc. of the International Conference on Water Resources Systems Application,* in Winnipeg, Manitoba, Canada, edited by S. P. Simonovic, I. C. Goulter, D. H. Burn, and B. J. Lence, pp. 442-451, 1990.

[53] Turban, E., A sample survey of operations research activities at the corporate level, *Operations Research, 20,* 708-721, 1972.

[54] Uber, J. G., and E. D. Brill, Jr., Design optimization sensitivity constraints, *Eng. Opt., 16,* 15-28, 1990.

[55] Uber, J. G., E. D. Brill, Jr., and J. T. Pfeffer, A nonlinear programming model of a wastewater treatment system: Sensitivity analysis and a robustness constraint, *Water Resour. Centre Report No. 196,* Dept. of Civ. Eng., University of Illinois, Urbana, 1985.

[56] Uber, J. G., E. D. Brill, Jr., and J. T. Pfeffer, Use of mathematical programming methods for complex systems: Illustrations using wastewater plant design, Accepted for publication in the *Jour. Environ. Eng. Div., ASCE,* 1991a.

[57] Uber, J. G., E. D. Brill, Jr., and J. T. Pfeffer, Robust optimal design for wastewater treatment: A general approach, Accepted for publication in the *Jour. Environ. Eng. Div., ASCE,* 1991b.

[58] Uber, J. G., J. T. Pfeffer, and E. D. Brill, Jr., Robust optimal design for wastewater treatment: An application, Accepted for publication in the *Jour. Environ. Eng. Div., ASCE,* 1991c.

[59] van Straten, G., Maximum likelihood estimation of parameters and uncertainty in phytoplankton models, in *Uncertainty and Forecasting of Water Quality,* edited by M. B. Beck and G. van Straten, pp 157-171, Springer Verlag, New York, 1983.

[60] Walker, W. W., A sensitivity analysis and error analysis framework for lake eutrophication modelling, *Water Resour. Bull., 18,* 53-60, 1982.

[61] Whitehead, P. G., and P. C. Young, Water quality in river systems: Monte-Carlo analysis, *Water Resour. Res., 15(6),* 1305-1312, 1979.

[62] Worley, J. L., *A system analysis method for water quality management by flow augmentation,* thesis presented to Oregon State University, Corvallis, Oreg., in partial fulfillment of the requirements for the degree of Master of Science, 1963.

[63] Young, P. C., and B. Beck, The modelling and control of water quality in a river system, *Automatica, 10,* 455-468, 1974.

Appendix A

STATISTICAL ANALYSIS PERFORMED ON INPUT DATA

The results of statistical analysis performed on flow and temperature data are presented in this appendix.

A.1 Distributions fitted

The Kolmogorov-Smirnov and Chi-Squared goodness of fit tests for the seasonal temperatures for the river and the seasonal flows at Station 3 are given below.

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 1-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	7	11	13	17	18	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.075

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	7	6	5	5
EXPECTED	6.6	6.6	6.6	3.3

CHI-SQUARED TEST STATISTIC = 1.348

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 1-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	10	11	13	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.149

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	6	5	5	7
EXPECTED	3.3	6.6	6.6	6.6

CHI-SQUARED TEST STATISTIC = 3.022

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 2-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	2	9	10	15	17	18	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.106

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	9	6	8
EXPECTED	6.6	6.6	9.9

CHI-SQUARED TEST STATISTIC = 1.297

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 2-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	8	12	14	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.118

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	8	8	7
EXPECTED	6.6	9.9	6.6

CHI-SQUARED TEST STATISTIC = 0.688

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 3-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	5	9	10	15	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.106

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	5	5	6	7
EXPECTED	3.3	6.6	6.6	6.6

CHI-SQUARED TEST STATISTIC = 1.348

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 3-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	8	12	14	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.118

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	8	8	7
EXPECTED	6.6	9.9	6.6

CHI-SQUARED TEST STATISTIC = 0.688

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 4-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	7	10	14	16	18	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.075

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	10	8	5
EXPECTED	9.9	9.9	3.3

CHI-SQUARED TEST STATISTIC = 1.246

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 4-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	8	12	14	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.118

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	8	8	7
EXPECTED	6.6	9.9	6.6

CHI-SQUARED TEST STATISTIC = 0.688

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 6-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	5	8	11	15	17	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.081

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	5	6	6	6
EXPECTED	3.3	6.6	6.6	6.6

CHI-SQUARED TEST STATISTIC = 1.043

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 6-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	5	7	11	14	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.075

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	5	6	5	7
EXPECTED	3.3	6.6	6.6	6.6

CHI-SQUARED TEST STATISTIC = 1.348

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER TEMPERATURE: 10-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	8	13	13	15	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.137

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	8	7	8
EXPECTED	6.6	9.9	6.6

CHI-SQUARED TEST STATISTIC = 1.449

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER TEMPERATURE: 10-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	8	12	14	16	19	23
EXPECTED	3.3	6.6	9.9	13.1	16.4	19.7	23.0

DN = 0.118

CRITICAL VALUES : C.V.(.10) = 0.255 C.V.(.05) = 0.283

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	8	8	7
EXPECTED	6.6	9.9	6.6

CHI-SQUARED TEST STATISTIC = 0.688

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 1-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	8	14	17	20	26	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.048

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	8	6	6	10
EXPECTED	8.6	4.3	8.6	8.6

CHI-SQUARED TEST STATISTIC = 1.733

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 1-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	13	14	19	19	25	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.148

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	13	6	6	5
EXPECTED	8.6	12.9	4.3	4.3

CHI-SQUARED TEST STATISTIC = 6.750

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 2-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	3	7	15	18	23	25	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.071

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	7	8	8	7
EXPECTED	8.6	4.3	8.6	8.6

CHI-SQUARED TEST STATISTIC = 3.833

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 2-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	11	15	16	20	26	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.081

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	6	5	9	10
EXPECTED	4.3	4.3	12.9	8.6

CHI-SQUARED TEST STATISTIC = 2.200

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 3-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	3	10	15	17	21	26	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.071

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	10	5	6	9
EXPECTED	8.6	4.3	8.6	8.6

CHI-SQUARED TEST STATISTIC = 1.150

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 3-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	10	14	16	19	27	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.081

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	6	8	5	11
EXPECTED	4.3	8.6	8.6	8.6

CHI-SQUARED TEST STATISTIC = 2.900

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 4-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	3	9	14	18	18	27	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.114

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	9	9	12
EXPECTED	8.6	12.9	8.6

CHI-SQUARED TEST STATISTIC = 2.550

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 4-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	6	11	13	15	19	27	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.081

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	6	7	6	11
EXPECTED	4.3	8.6	8.6	8.6

CHI-SQUARED TEST STATISTIC = 2.433

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 6-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	1	11	18	27	30	30	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.329

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	11	7	12
EXPECTED	8.6	4.3	17.1

CHI-SQUARED TEST STATISTIC = 3.950

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 6-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	0	1	10	16	28	30	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.252

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3
OBSERVED	10	6	14
EXPECTED	12.9	4.3	12.9

CHI-SQUARED TEST STATISTIC = 1.422

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR WINTER FLOW: 10-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	4	8	12	17	23	26	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.052

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4
OBSERVED	12	5	6	7
EXPECTED	12.9	4.3	4.3	8.6

CHI-SQUARED TEST STATISTIC = 1.150

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

GOODNESS OF FIT TEST FOR SUMMER FLOW: 10-MONTH SUMMER SEASON SCENARIO

TESTING FIT OF 2 PARAMETER LOGNORMAL DISTRIBUTION

KOLMOGOROV-SMIRNOV TEST

CLASS	1	2	3	4	5	6	7
OBSERVED	2	8	14	19	20	24	30
EXPECTED	4.3	8.6	12.9	17.1	21.4	25.7	30.0

DN = 0.076

CRITICAL VALUES : C.V.(.10) = 0.223 C.V.(.05) = 0.248

CHI-SQUARED TEST

CLASS	1	2	3	4	5
OBSERVED	8	6	5	5	6
EXPECTED	8.6	4.3	4.3	8.6	4.3

CHI-SQUARED TEST STATISTIC = 3.017

CRITICAL VALUES : C.V.(.10) = 2.706 C.V.(.05) = 3.841

A.2 Means and standard deviations of flows and temperatures

The means and standard deviations of the temperatures for the river and flows at Station 3, are given in Tables A.1 and A.2 respectively.

A.3 Results of regression analysis

For each summer season length combination, simple linear regressions were fit between the seasonal flows at Stations 1, 2, 4, and 5 and the corresponding seasonal flows at Station 3. The general regression equation is of the form:

$$FLOW_j = A + B * FLOW_3 \tag{A.1}$$

where

$FLOW_j$	=	seasonal flow at Station j (j = 1, 2, 4, and 5),
$FLOW_3$	=	seasonal flow at Station 3,
A	=	$FLOW_j$ intercept of the regression equation, and
B	=	the slope of the regression equation.

A summary of the regression analysis is given in Table A.3.

Table A.1: Statistics of temperatures (^{o}C)

SUMMER SEASON LENGTH = 1 MONTH				
	WINTER		SUMMER	
	Untransformed	Ln Trans.*	Untransformed	Ln Trans.*
Mean	18.08	2.893147	17.70	2.872299
Variance	1.007657	0.00304	0.708775	0.002213
SUMMER SEASON LENGTH = 2 MONTHS				
Mean	16.59	2.807645	18.24	2.902308
Variance	0.70166	0.002561	0.897183	0.002658
SUMMER SEASON LENGTH = 3 MONTHS				
Mean	15.79	2.757167	18.24	2.902308
Variance	1.270947	0.004972	0.897183	0.002658
SUMMER SEASON LENGTH = 4 MONTHS				
Mean	13.98	2.636276	18.24	2.902308
Variance	0.657603	0.00333	0.897183	0.002658
SUMMER SEASON LENGTH = 6 MONTHS				
Mean	10.81	2.378279	18.24	2.902308
Variance	0.588589	0.004867	0.897183	0.002658
SUMMER SEASON LENGTH = 10 MONTHS				
Mean	7.39	1.994226	18.24	2.902308
Variance	0.655097	0.011666	0.897183	0.002658
* Natural Log. Transformed				

Table A.2: Statistics of flows at Station 3 (*cusecs*)

SUMMER SEASON LENGTH = 1 MONTH				
	WINTER		SUMMER	
	Untransformed	Ln Trans.*	Untransformed	Ln Trans.*
Mean	3976	8.275245	4940	8.499568
Variance	399468	0.027333	293776	0.326924
SUMMER SEASON LENGTH = 2 MONTHS				
Mean	4231	8.335069	4489	8.399895
Variance	554950	0.032409	403675	0.019492
SUMMER SEASON LENGTH = 3 MONTHS				
Mean	4614	8.414765	4335	8.36606
Variance	961887	0.047029	327593	0.017497
SUMMER SEASON LENGTH = 4 MONTHS				
Mean	4937	8.482697	4194	8.329569
Variance	1071293	0.046592	435836	0.025116
SUMMER SEASON LENGTH = 6 MONTHS				
Mean	6152	8.685844	4044	8.293162
Variance	2923491	0.283278	402388	0.155482
SUMMER SEASON LENGTH = 10 MONTHS				
Mean	11125	9.18956	3977	8.273715
Variance	40727090	0.253288	475856	0.031189
* Natural Log. Transformed				

Table A.3: Summary of regression analysis for seasonal flows at Stations 1, 2, 4, and 5

SUMMER SEASON LENGTH = 1 MONTH								
Station	Winter				Summer			
	A	B	$\sqrt{MSE}$	R^2	A	B	$\sqrt{MSE}$	R^2
1	154.7566	0.403072	258.7068	0.501	2071.327	0.064311	324.2009	0.011
2	106.4874	0.934546	155.9704	0.937	331.4989	0.882207	237.9500	0.807
4	323.0905	1.312274	407.9946	0.811	-336.508	1.350616	335.1715	0.832
5	454.1811	1.481862	855.2376	0.554	113.7452	1.487217	875.7437	0.467
SUMMER SEASON LENGTH = 2 MONTHS								
1	-156.618	0.480554	268.5666	0.650	994.5115	0.230367	309.9935	0.186
2	134.5672	0.906055	211.9524	0.913	485.5986	0.845352	203.9581	0.878
4	-554.781	1.595877	558.4564	0.824	297.0946	1.240789	351.6179	0.839
5	-401.157	1.761411	1004.911	0.638	488.2159	1.417777	798.9789	0.568
SUMMER SEASON LENGTH = 3 MONTHS								
1	-459.391	0.529804	276.9537	0.785	1032.263	0.218113	294.9645	0.156
2	176.2259	0.862857	302.7148	0.890	282.6183	0.892933	208.2617	0.862
4	131.7262	1.519668	621.7668	0.856	-100.331	1.334161	361.9228	0.822
5	463.9734	1.732016	1292.286	0.642	-115.895	1.527621	818.2451	0.542
SUMMER SEASON LENGTH = 4 MONTHS								
1	-340.979	0.494084	351.5636	0.687	766.2603	0.257145	226.2975	0.368
2	149.2789	0.840467	341.2152	0.871	296.0544	0.887573	197.1514	0.901
4	65.53959	1.610640	924.7873	0.771	-1.25904	1.321708	349.5089	0.866
5	-173.522	1.942772	1765.609	0.573	117.8890	1.489518	809.8212	0.604
SUMMER SEASON LENGTH = 6 MONTHS								
1	177.0149	0.339629	412.2190	0.673	556.6705	0.312515	225.3317	0.453
2	40.65101	0.865441	359.2578	0.946	145.6524	0.924545	168.6637	0.928
4	22.65736	1.733208	785.0530	0.937	-22.3663	1.338933	345.4437	0.866
5	-491.842	2.494312	2301.377	0.781	-122.174	1.531277	743.0378	0.647
SUMMER SEASON LENGTH = 10 MONTHS								
1	-777.309	0.418387	1054.790	0.869	405.6662	0.343778	233.2368	0.517
2	99.80095	0.789148	948.4704	0.967	163.0812	0.918914	166.2812	0.938
4	2445.072	1.471581	1822.706	0.965	194.2988	1.290095	356.9297	0.866
5	5101.776	2.056622	5119.690	0.872	375.1344	1.415109	767.4048	0.626

Appendix B

LISTING OF FORTRAN PROGRAMS

B.1 Include file for the MAUT program

```
C*************************************************************
C  THIS PROGRAM IS AN INCLUDE FILE FOR THE MAUT PROGRAM    *
C                                                          *
C*************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
      DIMENSION TTRIB(12,50,10),COSTMIN(12),DISMIN(12)
      DIMENSION RVEL(20,2,30),RR(20,2,30),RITEMP(20,2,30),
     1          TT(2,30)
      DIMENSION CC(20,5,12),FCST(20),AVEL(20),BVEL(20),
     2          XK1(20),AK2(20),BK2(20),XS(20),
     3          NDISNO(20),
     4          NTRIBNO(20),XMEAN(20),XVAR(20)
      DIMENSION C1(12),T1(12),DO1(12),BOD2(20,12),
```

```
     2              Q2(20,12),T2(20,12),FEASC(901),
     3              DO2(20,12),W2(20,12),DIST2(21),DINC2(20)
      DIMENSION  DAYS(12),STD(12),C2L(20,12),C2U(20,12),
     2              CT2L(12),CT2U(12),
     3              SEGU(20,5,12),SEGL(20,5,12),XDAY(12),
     4              IX(12),IS(12),M1(12),M2(12),NSEG(20,12)
      DIMENSION XQ(12,50),C(12,50),XQTRIB(12,50,10),
     2              CCOST(12,50),AIJ(20,140),BDO(140),S3(12,50),
     3              XX(20,12,50),XTEMP(12,50),QQ(12,50),
     4              UTLEVL(2,30),JNOS(8),S1(12,50),S2(12,50),
     5           S4(12,50),S5(12,50)
      DIMENSION RHS(200),CCOEF(320),ACOEFF(200,320),
     2              NCOD(200)
      COMMON/STUFF/ TTRIB,COSTMIN,DISMIN,
     2              NY
      COMMON/MSTUFF/ RVEL,RR,RITEMP,TT
      COMMON/PARAM/ CC,FCST,AVEL,BVEL,
     2              XK1,AK2,BK2,XS,
     3              NDIS,NFIJ,IRC,NTRIB,NDISNO,
     4              NTRIBNO,XMEAN,XVAR
      COMMON/MEANS/ C1,T1,DO1,BOD2,
     2              Q2,T2,FEASC,
     3              DO2,W2,DIST2,DINC2
      COMMON/SEAS/  DAYS,STD,C2L,C2U,
     2              CT2L,CT2U,
     3              SEGU,SEGL,XDAY,
     4              NP,IX,IS,M1,M2,NSEG
      COMMON/RECORD/ XQ,C,XQTRIB,
     2              CCOST,AIJ,BDO,S3,S1,S2,S4,S5,
     3              XX,XTEMP,QQ,
     4              UTLEVL,JNOS
      COMMON/LPVAR/ RHS,CCOEF,ACOFF,
     2              NCOD,KP1,MP1,NVAR,KVAR,
```

```
     3              MCONS,NGET,NLET,NET,NTYPE,NP1,NC,NC1,INDEXG,
     4              INDEXL,INDEXE,NFLAG,NBASIC,NOPTSL,NOPT,ISEEDN
```

B.2 The MAUT program

```
C*************************************************************
C THIS PROGRAM PERFORMS MONTE-CARLO SIMULATION FOR THE    *
C MINIMUM AVERAGE UNIFORM TREATMENT RISK EQUIVALENT SEASONAL *
C DISCHARGE PROGRAM  *
C*************************************************************
C LAST UPDATE BY:  A. K. TAKYI
C ON DATE:         May 28 (8:00 pm)
C
C This program determines seasonal discharge loads and design
C conditions for several dischargers
C for a single pollutant so that a desired return
C period on water quality excursions is achieved at least
C level of treatment effort or least cost, and with or without
C a uniform treatment constraint.
C
C Variables in COMMON/PARAM/
C-----------------------------------------------------------------------
C       NDIS  =  number of reaches being managed, maximum = 20
C       NFIJ  =  number of waste dischargers in the river < or = NDIS
C NDISNO(LL)  =  the locations of the waste dischargers,
C                there are NFIJ of these
C      NTRIB  =  number of tributaries in the river = NDIS-NFIJ
CNTRIBNO(LL)  =  the locations of the tributaries,
C                there are NTRIB of these
C        IRC  =  risk criterion indicator
C CC(LL,5,K)  =  vector of total treatment cost coefficients
```

```
C                for discharger LL
C   FCST(LL)  =  fixed cost on discharger LL's cost curve
C     XK1(L)  =  deoxygenation rate coeff. at 20 deg. C for reach L,constant yearround
C AK2,BK2(L)  =  reaeration rate expression coeffs. for reach L,constant yearround
C      XS(L)  =  benthic oxygen demand at 20 deg. C for reach L,constant yearround
CAVEL,BVEL(L) =  velocity expression coeffs for reach L,constant year round
C-----------------------------------------------------------------------
C
C Variables in COMMON/MEANS/
C-----------------------------------------------------------------------
C     C1(12)  =  vector of seasonal means of ambient pollutant concentration
C     T1(12)  =  vector of seasonal means of ambient stream temperature
C    DO1(12)  =  vector of seasonal means of ambient stream dissolved oxygen
C   Q2(L,12)  =  vector of seasonal means of discharge flow for reach L
C   W2(L,12)  =  vector of seasonal means of withdrawal flow for reach L
C   T2(L,12)  =  vector of seasonal means of discharge temperature
C                for reach L
C  DO2(L,12)  =  vector of seasonal means of discharge dissolved oxygen
C                for reach L
C BOD2(L,12)  =  vector of seasonal means of discharge BOD, mg/L, for each
C                reach L, these are included so that tributary BOD loads
C                may be accounted for, they are zero for dischargers
C   DIST2(L)  =  distance of reach L from the starting point, kms
CDIST2(NDIS+1)=  distance to the last reach from the starting point, kms
C   DINC2(L)  =  distance increment (in kms) between the top of reach L and
C                the top of reach L+1--used to calculate the incremental DO
C                MINIMUM = [DIST2(L+1)-DIST2(L)]/12
C-----------------------------------------------------------------------
C
C  Variables in COMMON/SEAS/
C-----------------------------------------------------------------------
C         NP  =  number of seasons per year
C      IX(P)  =  season corresponding to decision variable P
```

```
C      M1(K)  =  starting month of season K
C      M2(K)  =  number of months in season K
C     DAYS(K)  =  number of days in season K
C      STD(K)  =  water quality concentration criterion in season K
C   C2L(LL,K)  =  lower limit on discharge concentration in season K
C                 for discharger LL
C   C2U(LL,K)  =  upper limit on discharge concentration in
C                 for discharger LL
C     CT2L(K)  =  lower limit on total discharge (lbs/day) in season K
C     CT2U(K)  =  upper limit on total discharge (lbs/day) in season K
C NSEG(LL,K)   =  number of segments in discharger LL's cost curve in season K
CSEGU(LL,5,K) =  upper limit on each segment of discharger LLs cost curve
C                 in season K
CSEGL(LL,5,K) =  lower limit on each segment of discharger LLs cost curve
C                 in season K
C-----------------------------------------------------------------------
C
C Variables in COMMON/RECORD/
C-----------------------------------------------------------------------
C      XQ(K,J) =  historical low flow in season K of year J
CXQTRIB(K,J,I)=  historical low flow in season K of year J for
C                 tributary I
C TTRIB(K,J,I)=  historical high temperature in season K of year J for
C                 tributary I
C XX(LL,K,J)   =  historical optimal load for season K of year J
C                 and discharger LL
C      C(K,J)  =  historical optimal total load for season K of year J
C CCOST(K,J)   =  historical optimal total cost for season K of year J
C COSTMIN(K)   =  minimum of the optimal total costs for season K over
C                 all years
C   AIJ(LL,N)  =  water quality impact coefficient for discharger LL on point N
C                 for a given temperature and flow
C      BDO(N)  =  background DO at point N for a given temperature and flow
```

```
C         NY  =  number of years in historical flow record
C-----------------------------------------------------------------------
C Local Variables in MAIN segment
C  DISMIN(K)  =  minimum of the historical optimal total load for season K
C     C2(K)   =  design allowable load or design total cost assigned to season K
C     ISTAT   =  status code which signifies an error in flow input
C  MONTH(K)   =  month of season K chosen to establish design conditions
C        NP   =  number of seasons
C     Q1(K)   =  design streamflow for season K
CDISCH1(K,LL) =  design waste load for discharger LL for season K
CFLOW1(K,LL)  =  design flow for discharger LL for season K
CREM1(K,LL)   =  design percent removal for discharger LL for season K
CCOST1(K,LL)  =  design cost for discharger LL for season K
C         R   =  return period on water quality excursion events, yr
C-----------------------------------------------------------------------
        include 'ins2'
C
       DIMENSION     C2(12),Q1(12),MONTH(12),DISCH1(12,20),
     2               FLOW1(12,20),REM1(12,20),COST1(12,20),TOTCOST(12)
C
       OPEN(3,FILE='ut7725.data')
       OPEN(8,FILE='nonfea725')
       OPEN(UNIT=2, FILE='inflo5', ERR=80)
       OPEN(UNIT=4, FILE='trib5.data')
       OPEN(UNIT=5, FILE='f1725')
       OPEN(UNIT=9, FILE='temp.data')
C
C Read seed value for the generation of the Uniform Random deviate
CC      WRITE(*,*) 'WHAT IS THE SEED VALUE - INTEGER'
CC      READ(*,*) NSEED
ISEEDN=123455
C
C Read problem input data and historical seasonal low flow record
```

```
C
      CALL INPUT(R,NUMBOSIM,STANDARD)
      CALL FLOW(ISTAT)
      IF (ISTAT .EQ. -1) GOTO 80
C
      DO 7 K=1,NP
7     STD(K)=STANDARD
C
      XDAY(1)=31.
      XDAY(2)=28.
      XDAY(3)=31.
      XDAY(4)=30.
      XDAY(5)=31.
      XDAY(6)=30.
      XDAY(7)=31.
      XDAY(8)=31.
      XDAY(9)=30.
      XDAY(10)=31.
      XDAY(11)=30.
      XDAY(12)=31.
C
      CALL SEASON
C
      DO 70 NSIM =1,NUMBOSIM
C
      ICHK=NY/IDINT(R)
      IEXC=0
C
C Construct historical record of seasonal total and individual allowable waste
 loads
C
      CALL ASCAP(R,ICHK,IEXC)
      IF(IEXC.GT.ICHK) GO TO 70
```

```
C
C Search for allowable pollutant loads that minimize treatment effort
C subject to desired risk of water quality excursions.
C
      CALL SEARCH(R,C2,ICHK,IEXC)
C
C Associate design conditions with allowable loads and display results.
C
      CALL DESIGN(C2,Q1,MONTH,DISCH1,FLOW1,REM1,COST1,TOTCOST)
70    CONTINUE
80    STOP
      END
C
C-------------------------------------------------------------------------
C
      SUBROUTINE INPUT(R,NUMBOSIM,STANDARD)
C
C This routine reads in problem input data.
C Passed Variables:
C    R  =  return period on years with water quality violations
C-------------------------------------------------------------------------
       include 'ins2'
C
C    Read in the number of reaches, the number of dischargers, where they are
C    located with respect to the reach numbers,the number of tributaries and their
C    locations, velocity coefficients, decay coefficients, reaeration coefficients,
C    and benthic coefficients.
C
      READ(3,*) NUMBOSIM
      READ(3,*) STANDARD
      READ(3,*) NP
      READ(3,*) (M1(I),I=1,NP)
      READ(3,*)NDIS
```

```
      READ(3,*)NFIJ
      READ(3,*) (NDISNO(I),I=1,NFIJ)
      READ(3,*)NTRIB
      READ(3,*) (NTRIBNO(I),I=1,NTRIB)
      DO 55 I=1,NDIS
55    READ(3,*) AVEL(I),BVEL(I)
      DO 65 I=1,NDIS
65    READ(3,*) XK1(I)
      DO 75 I=1,NDIS
75    READ(3,*) AK2(I),BK2(I)
      DO 85 I=1,NDIS
85    READ(3,*) XS(I)
90    CALL INPUT1
      CALL INPUT2
      CALL INPUT3
      CALL INPUT4(R)
      CALL INPUTT
      RETURN
      END
C
C-----------------------------------------------------------------
C
      SUBROUTINE INPUT1
C
       include 'ins2'
C
C   Read in the distance to each reach, the distance to the last point, the
C   distance increments, the discharge (l/sec), the withdrawal (l/sec),
C   temperature, DO and BOD (mg/l) loads of the dischargers.
C
      DO 32 J=1,NDIS
32    READ(3,*) DIST2(J)
      READ(3,*) DIST2(NDIS+1)
```

```
      DO 36 J=1,NDIS
36    READ(3,*) DINC2(J)
C
C  SUBSTITUTE ABOVE BLOCK WITH THIS
C
      DO 42 J=1,NFIJ
42    READ(3,*) (Q2(NDISNO(J),I),I=1,NP)
      DO 44 J=1,NFIJ
44    READ(3,*) (W2(NDISNO(J),I),I=1,NP)
      DO 65 J=1,NDIS
65    READ(3,*) (T2(J,I),I=1,NP)
      DO 75 J=1,NDIS
      DO 75 I=1,NP
75    DO2(J,I)=(14.652-(0.41022*T2(J,I))+(0.007991*(T2(J,I)**2))-
     1 (0.000077774*(T2(J,I)**3)))-1.0
      DO 76 J=1,NFIJ
 76   READ(3,*) (DO2(NDISNO(J),I),I=1,NP)
      DO 77 J=1,NDIS
77    READ(3,*) (BOD2(J,I),I=1,NP)
110   RETURN
      END
C
C-------------------------------------------------------------------------
C
      SUBROUTINE INPUT2
C
       include 'ins2'
C
C   Read in the initial (upstream of first discharger) BOD concentration,
C   DO concentration, and temperature
C
      READ(3,*) (C1(I),I=1,NP)
      READ(3,*) (T1(I),I=1,NP)
```

```
      DO 75 I=1,NP
75    DO1(I)=(14.652-(0.41022*T1(I))+(0.007991*(T1(I)**2))-
     1(0.000077774*(T1(I)**3)))-1.0
C
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE INPUT3
C
       include 'ins2'
C
C   Read in the number of seasons, starting month of each season, and
C   the upper and lower limit on BOD effluent concentration
C   for each discharger(mg/l).  Convert the concentrations to lb/day.
C   Read in the number of segments and the upper and lower limits
C   on concentration (mg/l) for each segment of each dischargers' cost
C   curve, and each season of the year.  Convert the limits to lb/day.
C   Add up the total maximum BOD loads, in lb/day, for each month,
C   given limits read in here.
C
      DO 85 J=1,NFIJ
85    READ(3,*) (C2U(J,I),I=1,NP)
      DO 95 J=1,NFIJ
95    READ(3,*) (C2L(J,I),I=1,NP)
      DO 105 J=1,NFIJ
      DO 105 I=1,NP
      C2U(J,I)=(C2U(J,I)*Q2(NDISNO(J),I))*.190512
105   C2L(J,I)=(C2L(J,I)*Q2(NDISNO(J),I))*.190512
      DO 107 J=1,NFIJ
      DO 107 I=1,NP
      READ(3,*) NSEG(J,I)
```

```
      READ(3,*) (SEGU(J,M,I),M=1,NSEG(J,I))
      READ(3,*) (SEGL(J,M,I),M=1,NSEG(J,I))
      DO 107 M=1,NSEG(J,I)
      SEGU(J,M,I)=(SEGU(J,M,I)*Q2(NDISNO(J),I))*.190512
107   SEGL(J,M,I)=(SEGL(J,M,I)*Q2(NDISNO(J),I))*.190512
      DO 110 I=1,NP
      CT2L(I)=0.0
110   CT2U(I)=0.0
      DO 120 J=1,NFIJ
      DO 120 I=1,NP
      CT2U(I)=CT2U(I)+C2U(J,I)
120   CT2L(I)=CT2L(I)+C2L(J,I)
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE INPUT4(R)
C
       include 'ins2'
C
C   Read the cost coefficients for each segment of each discharger's cost
C   curve, for each season of the year($/mg/l).  Read the fixed costs for each discharger
C   (million $ per year).
C   Read in the time period for counting excursions
C   and the return period for excursions.
C
      DO 10 J=1,NFIJ
      DO 10 I=1,NP
      READ(3,*) (CC(J,M,I),M=1,NSEG(J,I))
      DO 10 M=1,NSEG(J,I)
10    CC(J,M,I)=CC(J,M,I)/(Q2(NDISNO(J),I)*.190512)
      DO 20 J=1,NFIJ
```

```
20      READ(3,*) FCST(J)
        READ(3,*) IRC
        READ(3,*) R
        RETURN
        END
C
C-------------------------------------------------------------------------
C
SUBROUTINE INPUTT
include 'ins2'
C
C Read mean temperature for each season and year (oC)
C
DO 10 J=1,30
10 READ(9,*) (TT(I,J),I=1,2)
RETURN
END
C-------------------------------------------------------------------------
C
        SUBROUTINE FLOW(ISTAT)
C
C This routine reads in seasonal low flow values off a file called FLOW.DAT.
C
         include 'ins2'
C
       DIMENSION X(2)
C
       NY=0
5      READ(2,*,ERR=60,END=50) X
       NY=NY+1
       DO 20 I=1,NP
20     XQ(I,NY)=X(I)
       GOTO 5
```

```
50      CLOSE(UNIT=2)
        DO 55 N=1,NTRIB
        DO 55 J=1,NY
        DO 55 I=1,NP
55      READ(4,*)XQTRIB(I,J,N)
        ISTAT=0
        RETURN
60      ISTAT=-1
        RETURN
        END
C
C-------------------------------------------------------------------------
C
        SUBROUTINE SEASON
C
C This routine divides the year into seasons.
C-------------------------------------------------------------------------
         include 'ins2'
C
C For each season do the following:
C
        DO 10 K=1,NP
        DAYS(K)=0.
        M2(K)=0
C
C Find the start month and end month of the season,
C
        M=M1(K)
        IF (K .LT. NP) M3=M1(K+1)
        IF (K .EQ. NP) M3=M1(1)
C
C And find the number of days in the season.
C
```

```
5       IS(M)=K
        DAYS(K)=DAYS(K)+XDAY(M)
        M2(K)=M2(K)+1
        M=M+1
        IF (M .EQ. 13) M=1
        IF (M .NE. M3) GOTO 5
10      CONTINUE
        RETURN
        END
C
C-------------------------------------------------------------------------
C
        SUBROUTINE ASCAP(R,ICHK,IEXC)
C
C This routine finds the assimilative capacity of the river for each
C month of each year of the flow record and converts this into an
C allowable pollutant loading for each season of each year.
C-------------------------------------------------------------------------
         include 'ins2'
        DO 20 J=1,NY
        DO 10 I=1,NP
C
C Subroutine FIJS calculates the water quality impact coefficients
C for each discharger and the background DO in month I and year J.
C
        CALL FIJS (I,J)
C
C Subroutine LPSUBS calculates the waste load allocations in month I and
C year J using the simplex method.
C
        CALL LPSUBS(I,J,ICHK,IEXC)
        IF(NFLAG.NE.2) GO TO 10
        IEXC=IEXC+1
```

```
      IF(IEXC.GT.ICHK) THEN
      WRITE(5,*) 'NO SOLUTION'
      RETURN
      ENDIF
      CCOST(I,J)=31750000.
      C(I,J)=CT2L(I)
      DO 5 NN=1,NFIJ
5     XX(NN,I,J)=C2L(NN,I)
10    CONTINUE
20    CONTINUE
      CALL COMPRS
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE FIJS(I,J)
C
C Calculates the water quality impact coefficients
C for each discharger and the background DO in month I and year J,
C These values are then passed to LPSUBS to solve LP allocations.
C Passed Variables:
C      I  =  month of year
C      J  =  year of historical flow record
C Local Variables:
C     RK  =  deoxygenation rate coeff., 1/day (XK1 corrected for temp)
C     RR  =  reaeration rate coefficient, 1/day (corrected for temp)
C BENTIC  =  benthic oxygen demand, mg/L/day (XS corrected for temp)
C    RDO  =  reach initial dissolved oxygen, mg/L
C    RLO  =  reach initial BOD, mg/L
C RDODEF  =  reach initial DO deficit, mg/L
C    RCS  =  reach DO saturation concentration, mg/L
C   RVEL  =  reach average velocity, mtrs/sec (converted from ft/sec,.3048)
```

```
C  RTEMP  =  reach average temperature, C
C  RFLOW  =  flow rate of water entering reach, L/sec
C RITEMP  =  temperature of water entering reach, DEG C
C     GT  =  coefficient of BOD in S-F deficit eqn.
C     FT  =  coefficient of DO in S-F deficit eqn.
C BENTHO  =  benthic effect in S-F deficit eqn.
C    DEF  =  background DO deficit, all dischargers at 0 mg/L BOD, mg/L
C DEFDOT  =  DO deficit for a 1.0 mg/L change in BOD for each discharger, mg/L
C   DIST  =  distance from start to a given checkpoint
C    BOD  =  BOD at a given checkpoint
C
C-----------------------------------------------------------------------
C
        include 'ins2'
      DIMENSION RFLOW(20), RTEMP(20),
     1 DOBAC(20,12),RK(20), RCS(20), RLO(20), RDO(20),
     2 DEF(20,12), TIME(20,12), GT(20,12), NPT(21),
     3 FT(20,12),RDODEF(20),DIST(20,12), BENTIC(20), FIJ(20,12),
     4 NPOINT(20), BOD(20,12), DEFDOT(20,12),TMEAN(2),TVAR(2),
     5 S3LMEAN(2),S3LSTD(2),S1STD(2),S2STD(2),S4STD(2),S5STD(2),
     6 S1A(2),S1B(2),S2A(2),S2B(2),S4A(2),S4B(2),S5A(2),S5B(2)
       REAL U(10)
C
C Pollutant is ultimate oxygen demand
C
C
C for season 1 (winter) generate tributary and initial upstream flows
C
      NRAN=10
      U(10)=0.0
C
C reaeration, velocity constants
C
```

```
C temperature constants
C
      TMEAN(1)=2.636276
      TVAR(1)=0.057706152
C
      TMEAN(2)=2.902308
      TVAR(2)=0.051555795
C
C flow constants
C
      S3LMEAN(1)=8.482697
      S3LSTD(1)=0.215851
      S1STD(1)=351.5636
      S2STD(1)=341.2152
      S4STD(1)=924.7873
      S5STD(1)=1765.609
      S1A(1)=-340.979
      S1B(1)=0.494084
      S2A(1)=149.2789
      S2B(1)=0.840467
      S4A(1)=65.53959
      S4B(1)=1.61064
      S5A(1)=-173.522
      S5B(1)=1.942772
C
      S3LMEAN(2)=8.329569
      S3LSTD(2)=0.158480282
      S1STD(2)=226.2975
      S2STD(2)=197.1514
      S4STD(2)=349.5089
      S5STD(2)=809.8212
      S1A(2)=766.2603
      S1B(2)=0.257145
```

```
      S2A(2)=296.0544
      S2B(2)=0.887573
      S4A(2)=-1.25904
      S4B(2)=1.321708
      S5A(2)=117.889
      S5B(2)=1.489518
C  FIJS   ---------
C
C set up for current season (1 or 2)
C
C      IF (I.EQ.1) THEN
171     BB1=RAN(ISEEDN)
            CALL RANDY(U,NRAN,BB1)
            TEMPVAR=(TMEAN(I)+U(10)*TVAR(I))
            IF (TEMPVAR.LT.0.) GO TO 171
IF (I.EQ.1) THEN
   BB1=RAN(ISEEDN)
   CALL RANDY(U,NRAN,BB1)
   TEMPVAR=(U(10)*0.020489268+1.0)*DEXP(TEMPVAR)
ELSE
   BB1=RAN(ISEEDN)
   CALL RANDY(U,NRAN,BB1)
   TEMPVAR=(U(10)*0.018285671+1.0)*DEXP(TEMPVAR)
END IF
C            RITEMP(1,I,J)=DEXP(TEMPVAR)
            RITEMP(1,I,J)=TEMPVAR
    T1(I)=RITEMP(1,I,J)
         DO1(I)=(14.652-(0.41022*T1(I))+(0.007991*(T1(I)**2))-
     1(0.000077774*(T1(I)**3)))-1.0
C
    BB1=RAN(ISEEDN)
            CALL RANDY(U,NRAN,BB1)
            S3L=S3LMEAN(I)+U(10)*S3LSTD(I)
```

```
      IF (I.EQ.1) THEN
    BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
    IF (U(10).LT.0.)THEN
    S3(I,J)=(U(10)*0.073059518+1.0)*EXP(S3L)
    ELSE
    S3(I,J)=(U(10)*0.078817917+1.0)*EXP(S3L)
    END IF
        ELSE
    BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
    IF (U(10).LT.0.)THEN
    S3(I,J)=(U(10)*0.054178701+1.0)*EXP(S3L)
    ELSE
    S3(I,J)=(U(10)*0.057282175+1.0)*EXP(S3L)
    END IF
         END IF
199     BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
          S1(I,J)=(S1A(I)+S1B(I)*S3(I,J)+U(10)*S1STD(I))
    BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
          S2(I,J)=(S2A(I)+S2B(I)*S3(I,J)+U(10)*S2STD(I))
       IF ((S2(I,J).LT.S1(I,J)).OR.(S2(I,J).GT.S3(I,J))) GO TO 199
201     BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
          S4(I,J)=(S4A(I)+S4B(I)*S3(I,J)+U(10)*S4STD(I))
       IF (S4(I,J).LT.S3(I,J)) GO TO 201
202     BB1=RAN(ISEEDN)
          CALL RANDY(U,NRAN,BB1)
          S5(I,J)=(S5A(I)+S5B(I)*S3(I,J)+U(10)*S5STD(I))
       IF (S5(I,J).LT.S4(I,J)) GO TO 202
C
```

```
            XQ(I,J)=S1(I,J)
C
            XQTRIB(I,J,1)=S2(I,J)-S1(I,J)
            XQTRIB(I,J,2)=(S3(I,J)-S2(I,J))/2
            XQTRIB(I,J,3)=(S3(I,J)-S2(I,J))/2
            XQTRIB(I,J,4)=(S4(I,J)-S3(I,J))*0.8
            XQTRIB(I,J,5)=(S4(I,J)-S3(I,J))*0.2
            XQTRIB(I,J,6)=(S5(I,J)-S4(I,J))*0.241
            XQTRIB(I,J,7)=(S5(I,J)-S4(I,J))*0.12
            XQTRIB(I,J,8)=(S5(I,J)-S4(I,J))*0.639
C
      RLO(1)=C1(I)
      RDO(1)=DO1(I)
      RFLOW(1)=XQ(I,J)*28.32
      DO 200 M=1,MTRIB
           W2(MTRIBNO(M),I)=0.0
           Q2(MTRIBNO(M),I)=XQTRIB(I,J,M)*28.32
           T2(MTRIBNO(M),I)=RITEMP(1,I,J)
200   CONTINUE
      DO 280 MI=1,MFIJ
280        BOD2(MDISNO(MI),I)=0.0
C
      DO 1000 MR=1,MDIS
C
C  calc temp, DO sat, decay, reaeration, benthic coef
C
      RTEMP(MR)=((RFLOW(MR)-W2(MR,I))*RITEMP(MR,I,J)+Q2(MR,I)*T2(MR,I))/
     1 (RFLOW(MR)-W2(MR,I)+Q2(MR,I))
      RCS(MR)=14.652-(0.41022*RTEMP(MR))+(0.007991*(RTEMP(MR)**2))-
     1 (0.000077774*(RTEMP(MR)**3))
      RK(MR)=XK1(MR)*(1.047**(RTEMP(MR)-20.))
C
C generate reaeration, velocity values
```

```
C
      TEMPVAR=(RFLOW(NR)-W2(NR,I)+Q2(NR,I))/28.32
17    RVEL(NR,I,J)=AVEL(NR)*TEMPVAR**BVEL(NR)
      RVEL(NR,I,J)=RVEL(NR,I,J)*.3048
      IF (RVEL(NR,I,J).LT.0.OR.RVEL(NR,I,J).GT.(10.0)) GO TO 17
19    RR(NR,I,J)=AK2(NR)*TEMPVAR**BK2(NR)
      RR(NR,I,J)=RR(NR,I,J)*(1.024**(RTEMP(NR)-20.0))
      IF (RR(NR,I,J).LT.0.OR.RR(NR,I,J).GT.(5.0)) GO TO 19
C
      BENTIC(NR)=XS(NR)*(1.065**(RTEMP(NR)-20.))
C
C MASS BALANCE
C
300   RLO(NR)=((RFLOW(NR)-W2(NR,I))*RLO(NR)+Q2(NR,I)*BOD2(NR,I))/
     1 (RFLOW(NR)-W2(NR,I)+Q2(NR,I))
      RDO(NR)=((RFLOW(NR)-W2(NR,I))*RDO(NR)+Q2(NR,I)*DO2(NR,I))/
     1 (RFLOW(NR)-W2(NR,I)+Q2(NR,I))
C
C
C TRAVEL TIME
C
      TB=(DIST2(NR+1)-DIST2(NR))/(RVEL(NR,I,J)*86.4)
C
C  CALCULATE BOD AND FLOW RATE ENTERING REACH NR+1
C
      RLO(NR+1)=RLO(NR)*DEXP(-TB*RK(NR))
      RFLOW(NR+1)=RFLOW(NR)-W2(NR,I)+Q2(NR,I)
      RITEMP(NR+1,I,J)=RTEMP(NR)
C
C  CALCULATE TIME INCREMENT AND INITIAL DEFICIT IN REACH NR
C
      TI=DINC2(NR)/(RVEL(NR,I,J)*86.4)
      RDODEF(NR)=RCS(NR)-RDO(NR)
```

```
      IF(DABS(RK(NR)-RR(NR,I,J)).LT.0.0001)RK(NR)=RR(NR,I,J)+.0001
C
C  FOR EACH INTERVAL WITHIN REACH NR,DETERMINE TIME AND
C      DEFICIT USING STREETER-PHELPS EQUATION
C
  750 T=0
      IFLAG=0
      NPOINT(NR)=0
      DO 900 NI=1,12
      NPOINT(NR)=NPOINT(NR)+1
      IF(T.LT.(TB-1.0E-6))GO TO 800
      T=TB
      IFLAG=1
  800 GT(NR,NI)=RK(NR)/(RR(NR,I,J)-RK(NR))*(DEXP(-RK(NR)*T)
     1 -DEXP(-RR(NR,I,J)*T))
      FT(NR,NI)=DEXP(-RR(NR,I,J)*T)
      BENTHO=(1.-FT(NR,NI))*BENTIC(NR)/RR(NR,I,J)
      DEF(NR,NI)=RLO(NR)*GT(NR,NI)+RDODEF(NR)*FT(NR,NI)+BENTHO
              IF (DEF(NR,NI).GT.RCS(NR)) DEF(NR,NI)=RCS(NR)
C
      DOBAC(NR,NI)=RCS(NR)-DEF(NR,NI)
      TIME(NR,NI)=TTOT+T
      DIST(NR,NI)=DIST2(NR)+T*RVEL(NR,I,J)*86.4
      BOD(NR,NI)=RLO(NR)*DEXP(-T*RK(NR))
              IF (IFLAG.EQ.1) GO TO 950
      T=T+TI
  900 CONTINUE
  950 TTOT=TTOT+TB
C
C  CALCULATE DO ENTERING REACH NR+1
C
      N1=NPOINT(NR)
      RDO(NR+1)=RCS(NR)-DEF(NR,N1)
```

```
 1000 CONTINUE
C
      NPT(1)=0
      DO 1998 JJ=1,NDIS
      N1=NPOINT(JJ)
      DO 1997 K=1,N1
      IF(JJ.EQ.1) THEN
      MCONS=K
      ELSE
      MCONS=K+NPT(JJ-1)
      ENDIF
      BDO(MCONS)=DOBAC(JJ,K)
 1997 CONTINUE
      IF(JJ.EQ.1) THEN
      NPT(JJ)=N1
      ELSE
      NPT(JJ) = NPT(JJ-1)+N1
      ENDIF
 1998 CONTINUE
      NIT=NDISNO(1)
      DO 9996 IT=1,NFIJ
      BOD2(NIT,I)=0.0
            RFLOW(1)=XQ(I,J)*28.32
      RLO(1)=C1(I)
      RDO(1)=DO1(I)
      NIT=NDISNO(IT)
      BOD2(NIT,I)=1.0
C
      DO 9995 NR=1,NDIS
C
C MASS BALANCE
C
 3000 RLO(NR)=((RFLOW(NR)-W2(NR,I))*RLO(NR)+Q2(NR,I)*BOD2(NR,I))/
```

```
     1(RFLOW(NR)-W2(NR,I)+Q2(NR,I))
      RDO(NR)=((RFLOW(NR)-W2(NR,I))*RDO(NR)+Q2(NR,I)*DO2(NR,I))/
     1(RFLOW(NR)-W2(NR,I)+Q2(NR,I))
C
C TRAVEL TIME
C
      TB=(DIST2(NR+1)-DIST2(NR))/(RVEL(NR,I,J)*86.4)
C
C  CALCULATE BOD AND FLOW RATE ENTERING REACH NR+1
C
      RLO(NR+1)=RLO(NR)*DEXP(-TB*RK(NR))
      RFLOW(NR+1)=RFLOW(NR)-W2(NR,I)+Q2(NR,I)
C
C  CALCULATE TIME INCREMENT AND INITIAL DEFICIT IN REACH NR
C
      TI=DINC2(NR)/(RVEL(NR,I,J)*86.4)
      RDODEF(NR)=RCS(NR)-RDO(NR)
      IF(DABS(RK(NR)-RR(NR,I,J)).LT.0.0001)RK(NR)=RR(NR,I,J)+.0001
C
C  FOR EACH INTERVAL WITHIN REACH NR,DETERMINE TIME AND
C      DEFICIT USING STREETER-PHELPS EQUATION
C
 7500 T=0
      IFLAG=0
      DO 9000 NI=1,12
      IF(T.LT.(TB-1.0E-6))GO TO 8000
      T=TB
      IFLAG=1
 8000 GT(NR,NI)=RK(NR)/(RR(NR,I,J)-RK(NR))*(DEXP(-RK(NR)*T)
     1-DEXP(-RR(NR,I,J)*T))
      FT(NR,NI)=DEXP(-RR(NR,I,J)*T)
      BENTHO=(1.-FT(NR,NI))*BENTIC(NR)/RR(NR,I,J)
      DEFDOT(NR,NI)=RLO(NR)*GT(NR,NI)+RDODEF(NR)*FT(NR,NI)+BENTHO
```

```
              IF (DEFDOT(NR,NI).GT.RCS(NR)) DEFDOT(NR,NI)=RCS(NR)
C
      TIME(NR,NI)=TTOT+T
      DIST(NR,NI)=DIST2(NR)+T*RVEL(NR,I,J)*86.4
C
C CALCULATE THE IMPACT ON D.O. PER LB/DAY, MG/L
C
      FIJ(NR,NI)=((DEFDOT(NR,NI)-DEF(NR,NI))/(Q2(NIT,I)*BOD2(NIT,I))
     1 /.190512)*1.E4
C      FIJ(NR,NI)=((DEFDOT(NR,NI)-DEF(NR,NI))/BOD2(NIT,I))*1.E4
      IF(FIJ(NR,NI).LE.1.E-10) FIJ(NR,NI)=0.0
      IF(IFLAG.EQ.1) GO TO 9500
      T=T+TI
 9000 CONTINUE
 9500 TTOT=TTOT+TB
C
C CALCULATE DO ENTERING REACH NR+1
C
      N1=NPOINT(NR)
      RDO(NR+1)=RCS(NR)-DEFDOT(NR,N1)
 9995 CONTINUE
C
      NPT(1)=0
      DO 9998 JJ=1,NDIS
      N1=NPOINT(JJ)
      DO 9997 K=1,N1
      IF(JJ.EQ.1) THEN
      MCONS=K
      ELSE
      MCONS=K+NPT(JJ-1)
      ENDIF
      AIJ(NIT,MCONS)=FIJ(JJ,K)
9997 CONTINUE
```

```
      IF(JJ.EQ.1) THEN
      NPT(JJ)=N1
      ELSE
      NPT(JJ) = NPT(JJ-1)+N1
      ENDIF
 9998 CONTINUE
 9996 CONTINUE
      RETURN
      END
C
C-------------------------------------------------------------------------
C
       SUBROUTINE RANDY(U,NRAN,BB1)
C
       REAL  V(10),U(10)
       REAL*8 DS,D2,D3
C
       DS=99999.0*BB1+1.0
       IF(DS.EQ.0.) DS=DS+1.0
C
       D2=2147483647.D0
       D3=2147483648.D0
10     DO 1 I=1,NRAN
       DS=MOD(16807.D0*DS,D2)
 1     V(I)=SNGL(DS/D3)
IF(V(NRAN).EQ.0.)GO TO 10
       U(NRAN)=SQRT(2*ALOG(1.0/V(NRAN)))*SIN(2*3.14159*V(NRAN-1))
C
       RETURN
       END
C-------------------------------------------------
C
      SUBROUTINE LPSUBS (I,J,ICHK,IEXC)
```

```
C
C      I,J ARE SEASON AND YEAR RESPECTIVELY
C      A(L,K) IS IMPACT COEFF.AT CHECK PT. L BY DISCHARGER K
C      B(L) IS RHS COEFF. OF CONSTRAINT L
C      W(K) IS WASTE LOAD LEVEL OF DISCHARGER K
C      UTLEVL(I,J) IS UT LEVEL FOR SEASON I AND YEAR J
C      CALCULATE UT LEVEL REQ'D IN A GIVEN I,J TO SATISFY W.Q
C      CRITERIA AT EACH CHECK PT. L
C
       include 'ins2'
      DIMENSION A(40,15), B(40), W(15),UNT(40)
C
C-----------------------------------------------------------------------
C
      MCONS2=MCONS-1
      NFLAG=0
      DO 15 JJ=1,NFIJ
15    XX(JJ,I,J)=0.0
C
C Constraints
C
C   Water Quality Constraints
C
      DO 105 II=1,MCONS2
      B(II)=BDO(II+1) - STD(I)
      DO 105 JJ=1,NFIJ
      A(II,JJ) = AIJ(NDISNO(JJ),II+1)
105   CONTINUE
      DO 110 II=1,NFIJ
110   W(II) = (C2U(II,I)/.65)/1.E4
       DO 20 L=1,MCONS2
C      SET INITIAL UT LEVEL=0.40
       UT=0.40
```

```
   22  WL=1.00-UT
C      SALHS SUMS THE IMPACT OF DISCHARGERS AT A GIVEN L
       SALHS=0.00
       DO 25 K=1,10
C      ALHS COMPUTES THE IMPACT OF DISCHARGER K AT L
       ALHS=A(L,K)*WL*W(K)
       SALHS=SALHS+ALHS
   25  CONTINUE
C      COMPARE TOTAL IMPACT AT L WITH RHS OF CONSTRAINT 2
       IF(SALHS-B(L))30,35,40
C      INCREASE UT LEVEL IF THERE IS VIOLATION
   40  UT=UT+0.10
       IF(UT.GT.1.00)GO TO 56
       GO TO 22
C      DECREASE UT LEVEL IF THERE IS OVER DESIGN
   30  UT=UT-0.09
   41  WL=1.00-UT
       SALHS=0.00
       DO 26 K=1,10
       ALHS=A(L,K)*WL*W(K)
       SALHS=SALHS+ALHS
   26  CONTINUE
       IF(SALHS-B(L))35,35,50
C      COMPUTE 2ND DECIMAL PLACE FOR UT LEVEL
   50  UT=UT+0.01
       IF(UT.GT.0.98)GO TO 56
       GO TO 41
C      STORE UT LEVEL FOR A GIVEN I,J,L
   35  IF(UT.LT.0.35)GO TO 45
       UNT(L)=UT
       GO TO 55
   45  UNT(L)=0.35
   55  GO TO 20
```

```
   56  UNT(L)=1.00
   20  CONTINUE
C      SELECT THE MAX. UT LEVEL FOR A GIVEN I,J
       UTLIJ=0.00
       DO 80 L=1,MCONS2
       IF(UNT(L).GE.UTLIJ)GO TO 85
       UTLIJ=UTLIJ
       GO TO 80
   85  UTLIJ=UNT(L)
   80  CONTINUE
       UTLEVL(I,J)=UTLIJ
       C(I,J)=0.0
       IF(UTLEVL(I,J).EQ.1.) THEN
       WRITE (8,*) I, J
       WRITE (8,940)
940    FORMAT(///' A FEASIBLE SOLUTION DOES NOT EXIST')
       NFLAG=2
       ENDIF
C
       RETURN
       END
C
C-----------------------------------------------------------------------
C
       SUBROUTINE COMPRS
C
C This routine sorts out the removal levels for the 8 worst years and
C compresses the seasonal entries in matrices XCY and XTY into
C minimum seasonal values for matrices XXY and CY, based on total load
C or total cost and selects the minimum total discharge for each season in
C each year of record.
C Passed Variables:
C       XX  =  triple subscripted array of seasonal discharge values for
```

```
C                each year of record and each discharger
C         C  =  matrix of seasonal total discharge values for each year of
C                record
C-----------------------------------------------------------------------
        include 'ins2'
C
C
      DIMENSION AUT(30),BUT(30),TA(30),TB(30),JNO(30),
     1 JJNO(30),COST2(12,20),PREM(12,20)
C
C
      DO 5 J=1,NY
      AUT(J)=UTLEVL(1,J)
5     BUT(J)=UTLEVL(2,J)
      DO 9 JJ=1,NY
      JNO(JJ)=JJ
9     CONTINUE
      DO 30 J=1,4
      DO 20 I=1,(NY-1)
      IF(AUT(I).GT.AUT(I+1)) THEN
      TA(I)=AUT(I+1)
      TB(I)=BUT(I+1)
      JJNO(I)=JNO(I+1)
      AUT(I+1)=AUT(I)
      BUT(I+1)=BUT(I)
      JNO(I+1)=JNO(I)
      AUT(I)=TA(I)
      BUT(I)=TB(I)
      JNO(I)=JJNO(I)
      ENDIF
20    CONTINUE
      JNOS(J)=JNO(31-J)
      UTLEVL(1,JNOS(J))=AUT(31-J)
```

```
      UTLEVL(2,JNOS(J))=BUT(31-J)
30    CONTINUE
      DO 36 J=5,8
      DO 25 I=1,(NY-5)
      IF(BUT(I).GT.BUT(I+1)) THEN
      TB(I)=BUT(I+1)
      TA(I)=AUT(I+1)
      JJNO(I)=JNO(I+1)
      BUT(I+1)=BUT(I)
      AUT(I+1)=AUT(I)
      JNO(I+1)=JNO(I)
      BUT(I)=TB(I)
      AUT(I)=TA(I)
      JNO(I)=JJNO(I)
      ENDIF
25    CONTINUE
      JNOS(J)=JNO(31-J)
      UTLEVL(2,JNOS(J))=BUT(31-J)
      UTLEVL(1,JNOS(J))=AUT(31-J)
36    CONTINUE
C
      DO 981 J=1,8
      DO 982 I=1,NP
      DO 980 II=1,NFIJ
      XX(II,I,JNOS(J)) = (C2U(II,I)/.65)*(1-UTLEVL(I,JNOS(J)))
      C(I,JNOS(J))=C(I,JNOS(J))+XX(II,I,JNOS(J))
980   CONTINUE
C
C      WRITE(*,*) I,J,UTLEVL(I,J)
      CCOST(I,JNOS(J))=0.
      DO 911 NN=1,NFIJ
      COST2(I,NN)=0.0
      PREM(I,NN)=0.0
```

```
911     CONTINUE
        DO 915 NN=1,NFIJ
        PREM(I,NN)=C2U(NN,I)-XX(NN,I,JNOS(J))
        DO 13 MSEG=1,NSEG(NN,I)
        IF((PREM(I,NN)-SEGU(NN,MSEG,I)).GT.0.) THEN
        COST2(I,NN)=COST2(I,NN)+(SEGU(NN,MSEG,I)*
     1 CC(NN,MSEG,I))
        PREM(I,NN)=PREM(I,NN)-SEGU(NN,MSEG,I)
        ELSE
        COST2(I,NN)=COST2(I,NN)+(PREM(I,NN)*
     1 CC(NN,MSEG,I))
        PREM(I,NN)=0.0
        ENDIF
13      CONTINUE
        CCOST(I,JNOS(J))=CCOST(I,JNOS(J))+COST2(I,NN)
915     IF(CCOST(I,JNOS(J)).LT.0.00000000000001) CCOST(I,JNOS(J))=0.0
        CCOST(I,JNOS(J))=CCOST(I,JNOS(J))*1.E4
982     CONTINUE
981     CONTINUE
        DO 80 K=1,NP
        XCOSMIN=1.E20
        DO 70 J=1,8
        IF (CCOST(K,JNOS(J)).GE.XCOSMIN) GOTO 70
        XCOSMIN=CCOST(K,JNOS(J))
70      CONTINUE
        COSTMIN(K)=XCOSMIN
80      CONTINUE
        DO 45 K=1,NP
        XDISMIN=1.E20
        DO 35 J=1,8
        IF(C(K,JNOS(J)).LT.XDISMIN) XDISMIN=C(K,JNOS(J))
35      CONTINUE
        DISMIN(K)=XDISMIN
```

```
45      CONTINUE
90      RETURN
        END
C
C-------------------------------------------------------------------------
C
        SUBROUTINE SEARCH(R,C2,ICHK,IEXC)
C
C This routine sets up the seasonal allowable loads C2 into decision
C variables passed to the optimization routine COMPLX.
C Passed Variables:
C     NY  =  number of years of flow record
C      R  =  return period on water quality excursions
C     C2  =  optimized seasonal allowable loadings
C Local Variables:
C    NAP  =  number of seasonal discharge loads to be optimized
C      P  =  cumulative probability associated with return period R
C      X  =  vector of decision variables passed to optimization routine
C-------------------------------------------------------------------------
C
         include 'ins2'
        DIMENSION C2(12),X(12),G(12),BL(13),BU(13)
C
        NAP=0
C For each season, establish whether allowable load can be varied and
C if so, then add it to the input to the optimization routine.
        DO 10 K=1,NP
        X(K)=0.0
        C2(K)=CT2U(K)
        IF (CT2U(K) .LE. DISMIN(K)) GOTO 10
        NAP=NAP+1
        IX(NAP)=K
        BL(NAP)=CT2L(K)
```

```
      BU(NAP)=CT2U(K)
10    CONTINUE
      IF (NAP .EQ. 0) RETURN
C Define constraint limits for water quality excursion frequency.
      BL(NAP+1)=0.
      BU(NAP+1)=DBLE(NY)/R
C Call Total Enumeration optimization routine if problem is
C min cost and Box's Complex Direct Search optimization method
C if problem is max discharge.
      CALL ENUM(X,NAP,BL,BU,G,ICHK,IEXC)
      IT=0
      DO 40 L=1,NAP
40    C2(IX(L))=X(L)
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE ENUM(X,NAP,BL,BU,G,ICHK,IEXC)
       include 'ins2'
      DIMENSION X(12),G(12),BL(13),BU(13)
      IF(NAP.EQ.1) THEN
      CALL NAP1(X,NAP,BL,BU,G,ICHK,IEXC)
      ELSE
      CALL NAP2(X,NAP,BL,BU,G,ICHK,IEXC)
      ENDIF
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE NAP1(X,NAP,BL,BU,G,ICHK,IEXC)
       include 'ins2'
```

```
      DIMENSION X(12),G(12),BL(13),BU(13),XMIN(31)
      FMIN=1.E20
      JJ=0
      DO 10 J=1,8
      X(1)=CCOST(IX(1),JNOS(J))
      CALL FUNC2(X,NAP,F,G,ICHK,IEXC)
      IF(G(1).LT.BL(NAP+1)) GOTO 10
      IF(G(1).GT.BU(NAP+1)) GOTO 10
      JJ=JJ+1
      FEASC(JJ)=F
      XMIN(JJ)=X(1)
10    CONTINUE
      DO 30 L=1,JJ
      IF(FEASC(L).GE.FMIN) GOTO 30
      FMIN=FEASC(L)
      X(1)=XMIN(L)
30    CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE NAP2(X,NAP,BL,BU,G,ICHK,IEXC)
       include 'ins2'
      DIMENSION X(12),G(12),BL(13),BU(13),XMIN(2,901)
      FMIN=1.E20
      JJ=0
      DO 20 I=1,8
      X(1)=CCOST(IX(1),JNOS(I))
      DO 10 J=1,8
      X(2)=CCOST(IX(2),JNOS(J))
      CALL FUNC2(X,NAP,F,G,ICHK,IEXC)
      IF(G(1).LT.BL(NAP+1)) GOTO 10
```

```
      IF(G(1).GT.BU(NAP+1)) GOTO 10
      JJ=JJ+1
      FEASC(JJ)=F
      XMIN(1,JJ)=X(1)
      XMIN(2,JJ)=X(2)
10    CONTINUE
20    CONTINUE
      DO 30 L=1,JJ
      IF(FEASC(L).GE.FMIN) GOTO 30
      FMIN=FEASC(L)
      X(1)=XMIN(1,L)
      X(2)=XMIN(2,L)
30    CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE FUNC2(X,NAP,F,G,ICHK,IEXC)
C
C This function computes the cost and
C the number of water quality excursions over the historical streamflow
C record of a seasonal allowable loading policy.
C Passed Variables:
C     X  =  vector of seasonal allowable minimum cost values
C   NAP  =  number of active seasons
C     F  =  total treatment cost
C     G  =  number of water quality excursion events
C Local Variables:
C  Y(K)  =  allowable least cost value of season K
C   NV1  =  number of monthly excursions
C   NV2  =  number of seasonal excursions
C   NV3  =  number of annual excursions
```

```
C
C-----------------------------------------------------------------------
C
        include 'ins2'
       DIMENSION X(12),G(12),Y(12),IV2(12)
C
C Extend allowable total costs from active seasons (X) to all seasons (Y).
C
       DO 10 K=1,NP
10     Y(K)=COSTMIN(K)
       DO 20 L=1,NAP
20     Y(IX(L))=X(L)
C
C Compute cost of treatment.
C
       F=0.
C
C Compute the average daily cost over entire year.
C
52     DO 54 I=1,12
54     F=F+( Y(IS(I)) ) /12.
C
C Count up number of water quality excursions.
58     NV1=IEXC
       NV2=IEXC
       NV3=IEXC
       DO 90 J=1,8
       IV3=0
       DO 60 K=1,NP
60     IV2(K)=0
       DO 70 I=1,12
C
C Add to excursion count when design cost is lower than
```

```
c historical allowable cost.
C
      CCOST(IS(I),JMOS(J))=CCOST(IS(I),JMOS(J))-.0000001
      IF (Y(IS(I)) .GE. CCOST(IS(I),JMOS(J))) GOTO 69
      MV1=MV1+1
      IV2(IS(I))=1
      IV3=1
69    CCOST(IS(I),JMOS(J))=CCOST(IS(I),JMOS(J))+.0000001
70    CONTINUE
      DO 80 K=1,MP
80    MV2=MV2+IV2(K)
      MV3=MV3+IV3
90    CONTINUE
      IF (IRC .EQ. 1) G(1)=DBLE(MV1)
      IF (IRC .EQ. 2) G(1)=DBLE(MV2)
      IF (IRC .EQ. 3) G(1)=DBLE(MV3)
      RETURN
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE DESIGN(C2,Q1,MONTH,DISCH1,FLOW1,REM1,COST1,TOTCOST)
C
C This routine determines which month of the historical allowable
C load record comes closest to matching the design allowable loads
C in each season of the year.
C Passed Variables:
C     C2(K)  =  design allowable load/or cost for season K
C     Q1(K)  =  design streamflow flow for season K
C  MONTH(K)  =  month on which design conditions are based for season K
C  XX1(M,K)  =  design waste load for discharger M for season K
C Local Variables:
C   DIFF(K)  =  minimum difference between design and historical
```

```
C               allowable loads in season K
C-----------------------------------------------------------------------
        include 'ins2'
       DIMENSION C2(12),Q1(12),MONTH(12),DIFF(12),DISCH1(12,20),
     2           FLOW1(12,20),REM1(12,20),COST1(12,20),PREM(12,20),
     3           TOTCOST(12),XQT(12,20),RV(12,20),RR2(12,20),RT(12),
     4 Q11(12),Q12(12),Q14(12),Q15(12)
C
       DO 10 K=1,NP
       DIFF(K)=1.E20
       MONTH(K)=0
       Q1(K)=0.
       RT(K)=0.0
       DO 5 II=1,NFIJ
       FLOW1(K,II)=0.
       COST1(K,II)=0.
       REM1(K,II)=0
       PREM(K,II)=0.
       DISCH1(K,II)=0.
5      CONTINUE
       DO 7 II=1,NDIS
       RV(K,II)=0.
       RR2(K,II)=0.
7      CONTINUE
       DO 6 JJ=1,NTRIB
       XQT(K,JJ)=0.
6      CONTINUE
10     CONTINUE
       DO 30 J=1,8
       DO 20 M=1,NP
       T=DABS(C2(M)-CCOST(M,JNOS(J)))
       IF (T .GT. DIFF(M)) GOTO 20
       DIFF(M)=T
```

```
      MONTH(M)=M1(M)
      Q1(M)=S3(M,JNOS(J))
Q11(M)=S1(M,JNOS(J))
Q12(M)=S2(M,JNOS(J))
Q14(M)=S4(M,JNOS(J))
Q15(M)=S5(M,JNOS(J))
      TOTCOST(M) = 0.0
      RT(M)=RITEMP(1,M,JNOS(J))
C
C    THIS SECTION PREPARES DATA FOR BEHAVIOR/NBEHAVIOR RESULTS
C
      DO 8 II=1,NDIS
      RV(M,II)=RVEL(II,M,JNOS(J))
      RR2(M,II)=RR(II,M,JNOS(J))
8     CONTINUE
      DO 9 II=1,NTRIB
9     XQT(M,II)=XQTRIB(M,JNOS(J),II)
      DO 15 NN=1,NFIJ
      COST1(M,NN)=FCST(NN)
      FLOW1(M,NN)=Q2(NDISNO(NN),M)
      DISCH1(M,NN)=XX(NN,M,JNOS(J))
      PREM(M,NN)=C2U(NN,M)-DISCH1(M,NN)
      DO 13 MSEG=1,NSEG(NN,M)
      IF((PREM(M,NN)-SEGU(NN,MSEG,M)).GT.0.) THEN
      COST1(M,NN)=COST1(M,NN)+(SEGU(NN,MSEG,M)*
    1 CC(NN,MSEG,M))
      PREM(M,NN)=PREM(M,NN)-SEGU(NN,MSEG,M)
      ELSE
      COST1(M,NN)=COST1(M,NN)+(PREM(M,NN)*
    1 CC(NN,MSEG,M))
      PREM(M,NN)=0.0
      ENDIF
13    CONTINUE
```

```
      TOTCOST(M)=TOTCOST(M)+COST1(M,NN)
15    REM1(M,NN)=1-(DISCH1(M,NN)*.65/C2U(NN,M))
20    CONTINUE
30    CONTINUE
      WRITE(5,40) REM1(1,1),REM1(2,1),Q11(1),Q12(1),Q1(1),
     1 Q14(1),Q15(1),Q11(2),Q12(2),Q1(2),Q14(2),Q15(2),
     2 RT(1),RT(2)
call flush(5)
40    FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F5.2,2X),/)
      RETURN
      END
```

B.3 The binary classification program

```
C*********************************************************
C  THIS PROGRAM CLASSIFIES MONTE-CARLO SIMULATIONS AS     *
C   BEHAVIOUR OR NON-BEHAVIOURS                           *
C*********************************************************
      DIMENSION UT1S(3000),UT2S(3000),UT1NB(3000),UT2NB(3000),
     1XQ1(3000,5),XQ2(3000,5),XQ1NB(3000,5),XQ2NB(3000,5),
     2RT1(3000),RT2(3000),RT1NB(3000),RT2NB(3000),XQ1B(3000,5),
     3XQ2B(3000,5),RT1B(3000),RT2B(3000),UT1B(3000),UT2B(3000)
      OPEN(3,FILE='f1')
      OPEN(5,FILE='f2')
      OPEN(6,FILE='f
3')
      READ(3,15)UT1,UT2
15    FORMAT(2(F4.2,2X))
      DO 5 I=1,3000
      READ(3,25) UT1S(I),UT2S(I),(XQ1(I,K),K=1,5),
     1 (XQ2(I,K),K=1,5),RT1(I),RT2(I)
```

```
 25    FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F5.2,2X),/)
  5    CONTINUE
       JN=0
       JB=0
       DO 100 I=1,3000
       IF(UT1S(I).GT.UT1)GO TO 10
       IF(UT2S(I).GT.UT2)GO TO 10
C**********************************
C      NON-BEHAVIOUR OCCURS       *
C**********************************
       JN=JN+1
       UT1NB(JN)=UT1S(I)
       UT2NB(JN)=UT2S(I)
       DO 20 K=1,5
       XQ1NB(JN,K)=XQ1(I,K)
       XQ2NB(JN,K)=XQ2(I,K)
 20    CONTINUE
       RT1NB(JN)=RT1(I)
       RT2NB(JN)=RT2(I)
       GO TO 100
C**********************************
C      BEHAVIOUR OCCURS           *
C**********************************
 10    JB=JB+1
       UT1B(JB)=UT1S(I)
       UT2B(JB)=UT2S(I)
       DO 40 K=1,5
       XQ1B(JB,K)=XQ1(I,K)
       XQ2B(JB,K)=XQ2(I,K)
 40    CONTINUE
       RT1B(JB)=RT1(I)
       RT2B(JB)=RT2(I)
100    CONTINUE
```

```
      NNBEHA=JN
      NBEH=JB
      IF(NNBEHA.LT.20)GO TO 110
      IF(NBEH.LT.20)GO TO 120
      GO TO 130
110   WRITE(*,*)'TOO MANY BEHAVIOURS'
      GO TO 130
120   WRITE(*,*)'TOO MANY NON BEHAVIOURS'
130   WRITE(5,135)NNBEHA
135   FORMAT(I4)
      DO 140 J=1,JN
      WRITE(5,25)UT1NB(J),UT2NB(J),(XQ1NB(J,K),K=1,5),
     1 (XQ2NB(J,K),K=1,5),RT1NB(J),RT2NB(J)
140   CONTINUE
C*********************************
C     WRITE STMTS FOR BEHAVIORS  *
C*********************************
      WRITE(6,135)NBEH
      DO 180 J=1,JB
      WRITE(6,25)UT1B(J),UT2B(J),(XQ1B(J,K),K=1,5),
     1 (XQ2B(J,K),K=1,5),RT1B(J),RT2B(J)
180   CONTINUE
      STOP
      END
```

B.4 Program for the calculation of correlation and covariance matrices

```
C*************************************************************
C THIS PROGRAM COMPUTES VAR-COV AND CORRELATION MATRICES  *
```

```
C FOR THE BEHAVIOUR, NON-BEHAVIOUR AND UNCLASSIFIED       *
C PARAMETERS MATRICES                                      *
C***********************************************************
PROGRAM covcor
C
C CALCULATES THE EIGENVECTORS FOR THE TRANSFORMED VAR-COV
MATRIX
C
PARAMETER(NP=12,N=12)
DIMENSION PSINB(3000,NP),PSIB(3000,NP),COV(NP,NP),
     1 CNB(NP,NP),CB(NP,NP),XMEANNB(NP),INDX(NP),
     2 XMEANB(NP),XMEAN(NP),VARNB(NP),VARB(NP),
     3 VAR(NP),XMBNB(NP),XMB(NP),XMNB(NP),
     4 COVBNB(NP,NP),COVB(NP,NP),COVNB(NP,NP),
     5 AA(NP,NP),BB(NP,NP),CC(NP,NP),UT1B(3000),UT2B(3000),
     6 CONB(NP,NP),COB(NP,NP),Y(NP,NP),
     7 UT1NB(3000),UT2NB(3000)
C
OPEN(2,FILE='f2')
OPEN(3,FILE='f3')
OPEN(4,FILE='f4')
OPEN(5,FILE='f5')
C READ IN RAW DATA
READ(2,135)NNBEHA
135 FORMAT(I4)
DO 101 J=1,NNBEHA
READ(2,25)UT1NB(J),UT2NB(J),(PSINB(J,I),I=1,NP)
101 CONTINUE
READ(3,135)NBEH
DO 15 J=1,NBEH
READ(3,25)UT1B(J),UT2B(J),(PSIB(J,I),I=1,NP)
15 CONTINUE
C CALCULATE MEANS AND VARIANCES OF RAW DATA
```

```
      DO 903 I=1,NP
      XMEANNB(I)=0.0
      XMEANB(I)=0.0
      DO 902 J=1,NNBEHA
  902 XMEANNB(I)=XMEANNB(I)+PSINB(J,I)
      DO 802 J=1,NBEH
  802 XMEANB(I)=XMEANB(I)+PSIB(J,I)
      XMEAN(I)=XMEANNB(I)+XMEANB(I)
      XMEAN(I)=XMEAN(I)/FLOAT(NNBEHA+NBEH)
  903 CONTINUE
      DO 905 I=1,NP
      VARNB(I)=0.0
      VARB(I)=0.0
      DO 906 J=1,NNBEHA
  906 VARNB(I)=VARNB(I)+(PSINB(J,I)-XMEAN(I))**2.0
      DO 806 J=1,NBEH
  806 VARB(I)=VARB(I)+(PSIB(J,I)-XMEAN(I))**2.0
      VAR(I)=VARNB(I)+VARB(I)
      VAR(I)=VAR(I)/(FLOAT(NNBEHA+NBEH)-1.0)
  905 CONTINUE
      C
      C CALCULATE VAR-COVAR MATRICES FOR BEH,NON-BEH AND
      C UNCLASSIFIED DATA
      DO 2 K=1,NP
      XMBNB(K)=0.0
      XMB(K)=0.0
      XMNB(K)=0.0
      DO 2 J=1,NP
      COVBNB(K,J)=0.0
      COVB(K,J)=0.0
    2 COVNB(K,J)=0.0
      DO 10 I=1,NBEH
      DO 4 K=1,NP
```

```
PSIB(I,K)=(PSIB(I,K)-XMEAN(K))/SQRT(VAR(K))
XMB(K)=XMB(K)+PSIB(I,K)
4 CONTINUE
10 CONTINUE
DO 11 I=1,NNBEHA
DO 6 K=1,NP
PSINB(I,K)=(PSINB(I,K)-XMEAN(K))/SQRT(VAR(K))
XMNB(K)=XMNB(K)+PSINB(I,K)
6 CONTINUE
11 CONTINUE
DO 12 K=1,NP
XMBNB(K)=XMB(K)+XMNB(K)
XMBNB(K)=XMBNB(K)/FLOAT(NBEH+NNBEHA)
XMB(K)=XMB(K)/FLOAT(NBEH)
12 XMNB(K)=XMNB(K)/FLOAT(NNBEHA)
DO 40 J=1,NP
DO 40 K=1,NP
DO 41 I=1,NBEH
COVB(J,K)=COVB(J,K)+(PSIB(I,J)-XMB(J))*
     1 (PSIB(I,K)-XMB(K))
COVBNB(J,K)=COVBNB(J,K)+(PSIB(I,J)-XMBNB(J))*
     1 (PSIB(I,K)-XMBNB(K))
41 CONTINUE
DO 42 I=1,NNBEHA
COVNB(J,K)=COVNB(J,K)+(PSINB(I,J)-XMNB(J))*
     1 (PSINB(I,K)-XMNB(K))
COVBNB(J,K)=COVBNB(J,K)+(PSINB(I,J)-XMBNB(J))*
     1 (PSINB(I,K)-XMBNB(K))
42 CONTINUE
COVB(J,K)=COVB(J,K)/FLOAT(NBEH-1.0)
COVNB(J,K)=COVNB(J,K)/FLOAT(NNBEHA-1.0)
40 COVBNB(J,K)=COVBNB(J,K)/FLOAT(NBEH+NNBEHA-1.0)
C
```

```
C CALCULATE CORRELATION MATRICES
DO 50 J=1,NP
DO 50 K=1,NP
AA(J,K)=COVNB(J,K)/SQRT(COVNB(J,J)*COVNB(K,K))
BB(J,K)=COVB(J,K)/SQRT(COVB(J,J)*COVB(K,K))
CC(J,K)=COVBNB(J,K)/SQRT(COVBNB(J,J)*COVBNB(K,K))
50 CONTINUE
DO 619 I=1,NP
DO 619 K=1,NP
AA(I,K)=XMNB(I)*XMNB(K)
BB(I,K)=XMB(I)*XMB(K)
CC(I,K)=XMBNB(I)*XMBNB(K)
619 CONTINUE
TOTAL=FLOAT(NNBEHA+NBEH)
APNB=FLOAT(NNBEHA)/TOTAL
APB=FLOAT(NBEH)/TOTAL
DO 8121 I=1,NP
DO 8121 K=1,NP
COVNB(I,K)=(COVNB(I,K)+AA(I,K))*APNB
COVB(I,K)=(COVB(I,K)+BB(I,K))*APB
COVBNB(I,K)=COVBNB(I,K)+CC(I,K)
8121 CONTINUE
C
C PREPARE COVARIANCE MATRICES FOR COMPUTATION OF EIGENVECTOR
DO 4903 K=1,NP
DO 4903 J=1,NP
CONB(K,J)=COVNB(K,J)
COB(K,J)=COVB(K,J)
4903 COV(K,J)=COVBNB(K,J)
DO 199 L=1,NP
199 WRITE(5,230) (COV(L,K),K=1,NP)
C
C CALL MATCOPY(COV,NP,A)
```

```
DO 102 I=1,N
   DO 201 J=1,N
Y(I,J)=0.
201    CONTINUE
   Y(I,I)=1.
102 CONTINUE
CALL LUDCMP (COV,N,NP,INDX,D)
DO 103 J=1,N
   CALL LUBKSB (COV,N,NP,INDX,Y(1,J))
103 CONTINUE
DO 66 I=1,NP
DO 77 K=1,NP
SUM=0.
DO 88 L=1,NP
88 SUM=SUM+Y(I,L)*CONB(L,K)
CNB(I,K)=SUM
77 CONTINUE
66 CONTINUE
DO 100 L=1,NP
100 WRITE(4,230) (CNB(L,K),K=1,NP)
DO 626 I=1,NP
DO 727 K=1,NP
SUM=0.
DO 828 L=1,NP
828 SUM=SUM+Y(I,L)*COB(L,K)
CB(I,K)=SUM
727 CONTINUE
626 CONTINUE
DO 98 L=1,NP
98 WRITE(4,230) (CB(L,K),K=1,NP)
230 FORMAT(4(/,3(E13.5,2X)))
220 FORMAT(/,1X,'CORRELATION MATRICES-NONBEHA,BEHA,UNCLASS.')
25    FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F5.2,2X),/)
```

```
      STOP
      END
C
      SUBROUTINE LUDCMP (A,N,NP,INDX,D)
      PARAMETER (NMAX=100,TINY=1.0E-20)
      DIMENSION A(NP,NP),INDX(N),VV(NMAX)
      D=1.
      DO 12 I=1,N
        AAMAX=0.
        DO 11 J=1,N
        IF (ABS(A(I,J)).GT.AAMAX) AAMAX=ABS(A(I,J))
11      CONTINUE
      IF (AAMAX.EQ.0.) PAUSE 'Singular matrix.'
      VV(I)=1./AAMAX
12    CONTINUE
      DO 19 J=1,N
        DO 14 I=1,J-1
      SUM=A(I,J)
      DO 13 K=1,I-1
        SUM=SUM-A(I,K)*A(K,J)
13    CONTINUE
      A(I,J)=SUM
14      CONTINUE
        AAMAX=0.
        DO 16 I=J,N
      SUM=A(I,J)
      DO 15 K=1,J-1
        SUM=SUM-A(I,K)*A(K,J)
15    CONTINUE
      A(I,J)=SUM
      DUM=VV(I)*ABS(SUM)
      IF (DUM.GE.AAMAX) THEN
        IMAX=I
```

```
      AAMAX=DUM
 ENDIF
 16       CONTINUE
 IF (J.NE.IMAX) THEN
    DO 17 K=1,N
 DUM=A(IMAX,K)
 A(IMAX,K)=A(J,K)
 A(J,K)=DUM
 17    CONTINUE
   D=-D
   VV(IMAX)=VV(J)
 ENDIF
 INDX(J)=IMAX
 IF(A(J,J).EQ.0.)A(J,J)=TINY
 IF(J.NE.N)THEN
    DUM=1./A(J,J)
    DO 18 I=J+1,N
A(I,J)=A(I,J)*DUM
18    CONTINUE
ENDIF
19 CONTINUE
RETURN
END
C
SUBROUTINE LUBKSB(A,N,NP,INDX,B)
DIMENSION A(NP,NP),INDX(N),B(N)
II=0
DO 12 I=1,N
   LL=INDX(I)
   SUM=B(LL)
   B(LL)=B(I)
   IF (II.NE.0)THEN
DO 11 J=II,I-1
```

```
   SUM=SUM-A(I,J)*B(J)
11 CONTINUE
   ELSE IF (SUM.NE.0.) THEN
II=I
   ENDIF
   B(I)=SUM
12 CONTINUE
DO 14 I=N,1,-1
    SUM=B(I)
    IF(I.LT.N) THEN
DO 13 J=I+1,N
   SUM=SUM-A(I,J)*B(J)
13 CONTINUE
    ENDIF
    B(I)=SUM/A(I,I)
14 CONTINUE
RETURN
END
```

B.5 Program for the calculation of the transformation matrix

```
C*************************************************************
C  THIS PROGRAM CALCULATES THE MATRIX THAT SIMULTANEOUSLY *
C  DIAGONALIZES TWO SYMMETRIC MATRICES THAT ADD UP TO THE *
C  IDENTITY MATRIX {bf I}                                  *
C*************************************************************
  PARAMETER(NM=24,N=12)
   REAL A(NM,NM),B(NM,NM),Z(NM,NM),ALFR(NM),ALFI(NM),BETA(NM),
     1 CH(NM),TZ(NM,NM),COV(NM,NM),TSC(NM,NM),ASC(NM,NM)
```

```
C
OPEN(3,FILE='f4')
OPEN(4,FILE='f5')
OPEN(5,FILE='f6')
OPEN(6,FILE='f7')
DO 31 J=1,N
31 READ(3,230) (A(J,K),K=1,N)
C
DO 1991 L=1,N
1991 READ(4,230) (COV(L,K),K=1,N)
DO 321 J=1,N
  DO 221 K=1,N
    B(J,K)=0.0
221   CONTINUE
  B(J,J)=1.0
321 CONTINUE
EPS1=-1.0
CALL QZHES(NM,N,A,B,.TRUE.,Z)
CALL QZIT(NM,N,A,B,EPS1,.TRUE.,Z,IERR)
CALL QZVAL(NM,N,A,B,ALFR,ALFI,BETA,.TRUE.,Z)
IF (IERR.NE.0) GO TO 99999
CALL QZVEC(NM,N,A,B,ALFR,ALFI,BETA,Z)
99999 CONTINUE
C*********************************************************
C SCALE EIGENVECTORS TO DIRECTIONAL COSINES     *
C*********************************************************
        DO 7177 I=1,N
DO 7177 K=1,N
7177 TZ(K,I)=Z(I,K)
DO 666 K=1,N
DO 777 I=1,N
SUM=0.0
DO 888 L=1,N
```

```
888 SUM=SUM+COV(I,L)*Z(L,K)
TSC(I,K)=SUM
777 CONTINUE
666 CONTINUE
DO 1666 K=1,N
DO 1777 I=1,N
SUM=0.0
DO 1888 L=1,N
1888 SUM=SUM+TZ(I,L)*TSC(L,K)
ASC(I,K)=SUM
1777 CONTINUE
1666 CONTINUE
DO 6661 I=1,N
6661 CH(I)=SQRT(ASC(I,I))
DO 53 K=1,N
DO 54 J=1,N
Z(J,K)=Z(J,K)/CH(K)
54 CONTINUE
53 CONTINUE
C*********************************************************
C WRITE SCALED EIGENVECTORS AND EIGENVALUES     *
C*********************************************************
DO 199 J=1,N
199 WRITE(5,230) (Z(J,K),K=1,N)
WRITE(6,*) 'REAL EVALS'
WRITE(6,8004) (ALFR(I),I=1,N)
8004 FORMAT(1X,2(6(E9.3,2X),/))
C
WRITE(6,*) 'IMAG. EVALS'
        WRITE(6,8004) (ALFI(I),I=1,N)
230 FORMAT(4(/,3(E13.5,2X)))
STOP
END
```

```
C
      SUBROUTINE QZHES(NM,N,A,B,MATZ,Z)
C
      INTEGER I,J,K,L,N,LB,L1,NM,NK1,NM1,NM2
      REAL A(NM,N),B(NM,N),Z(NM,N)
      REAL R,S,T,U1,U2,V1,V2,RHO
      REAL SQRT,ABS,SIGN
      LOGICAL MATZ
C
      IF(.NOT.MATZ) GO TO 10
C
      DO 3 I=1,N
C
            DO 2 J=1,N
      Z(I,J)=0.0
2           CONTINUE
C
            Z(I,I)=1.0
3     CONTINUE
C*****************************************************
C REDUCE B TO UPPER TRIANGULAR FORM          *
C*****************************************************
10    IF(N.LE.1) GO TO 170
      NM1=N-1
C
      DO 100 L=1,NM1
      L1=L+1
            S=0.0
C
      DO 20 I=L1,N
            S=S+ABS(B(I,L))
20    CONTINUE
C
```

```
IF(S.EQ.0.0) GO TO 100
S=S+ABS(B(L,L))
        R=0.0
C
DO 25 I=L,N
B(I,L)=B(I,L)/S
R=R+B(I,L)**2
25 CONTINUE
C
R=SIGN(SQRT(R),B(L,L))
B(L,L)=B(L,L)+R
RHO=R*B(L,L)
C
DO 50 J=L1,N
T=0.0
C
DO 30 I=L,N
T=T+B(I,L)*B(I,J)
30 CONTINUE
C
T=-T/RHO
C
DO 40 I=L,N
B(I,J)=B(I,J)+T*B(I,L)
40 CONTINUE
C
50 CONTINUE
C
DO 80 J=1,N
        T=0.0
C
        DO 60 I=L,N
        T=T+B(I,L)*A(I,J)
```

```
 60 CONTINUE
 C
          T=-T/RHO
 C
 DO 70 I=L,N
 A(I,J)=A(I,J)+T*B(I,L)
 70 CONTINUE
 C
 80 CONTINUE
 C
 B(L,L)=-S*R
 C
 DO 90 I=L1,N
 B(I,L)=0.0
 90 CONTINUE
 C
 100 CONTINUE
 C*****************************************************
 C     REDUCE A TO UPPER HESSENBERG FORM, WHILE       *
 C        KEEPING B TRIANGULAR                        *
 C*****************************************************
 IF(N.EQ.2) GO TO 170
 NM2=N-2
 C
 DO 160 K=1,NM2
         NK1=NM1-K
 C*****************************************************
 C     FOR L=N-1 STEP -1 UNTIL K+1 DO                 *
 C*****************************************************
 DO 150 LB=1,NK1
         L=N-LB
 L1=L+1
 C*****************************
```

```
C    ZERO A(L+1,K)           *
C*****************************
        S=ABS(A(L,K))+ABS(A(L1,K))
        IF(S.EQ.0.0) GO TO 150
        U1=A(L,K)/S
        U2=A(L1,K)/S
        R=SIGN(SQRT(U1*U1+U2*U2),U1)
        V1=-(U1+R)/R
        V2=-U2/R
        U2=V2/V1
C
DO 110 J=K,N
        T=A(L,J)+U2*A(L1,J)
A(L,J)=A(L,J)+T*V1
A(L1,J)=A(L1,J)+T*V2
110 CONTINUE
C
A(L1,K)=0.0
C
DO 120 J=L,N
T=B(L,J)+U2*B(L1,J)
B(L,J)=B(L,J)+T*V1
B(L1,J)=B(L1,J)+T*V2
120 CONTINUE
C*****************************
C    ZERO B(L+1,L)           *
C*****************************
S=ABS(B(L1,L1))+ABS(B(L1,L))
IF(S.EQ.0.0) GO TO 150
U1=B(L1,L1)/S
U2=B(L1,L)/S
R=SIGN(SQRT(U1*U1+U2*U2),U1)
V1=-(U1+R)/R
```

```
      V2=-U2/R
      U2=V2/V1
C
      DO 130 I=1,L1
              T=B(I,L1)+U2*B(I,L)
      B(I,L1)=B(I,L1)+T*V1
      B(I,L)=B(I,L)+T*V2
  130 CONTINUE
C
      B(L1,L)=0.0
C
      DO 140 I=1,N
       T=A(I,L1)+U2*A(I,L)
       A(I,L1)=A(I,L1)+T*V1
       A(I,L)=A(I,L)+T*V2
  140 CONTINUE
C
      IF(.NOT.MATZ) GO TO 150
C
      DO 145 I=1,N
        T=Z(I,L1)+U2*Z(I,L)
        Z(I,L1)=Z(I,L1)+T*V1
        Z(I,L)=Z(I,L)+T*V2
  145 CONTINUE
C
  150 CONTINUE
C
  160 CONTINUE
C
  170 RETURN
      END
C
      SUBROUTINE QZIT(NM,N,A,B,EPS1,MATZ,Z,IERR)
```

```
C
      INTEGER I,J,K,L,N,EN,K1,K2,LD,LL,L1,NA,NM,ISH,ITS,KM1,LM1,
     X        ENM2,IERR,LOR1,ENORN
      REAL A(NM,N),B(NM,N),Z(NM,N)
      REAL R,S,T,A1,A2,A3,EP,SH,U1,U2,U3,V1,V2,V3,ANI,A11,A12,
     X       A21,A22,A33,A34,A43,A44,BNI,B11,B12,B22,B33,B34,
     X       B44,EPSA,EPSB,EPS1,ANORM,BNORM
      REAL SQRT,ABS,SIGN
      INTEGER MAX0,MIN0
      LOGICAL MATZ,NOTLAS
C
      IERR = 0
C*******************************
C      COMPUTE EPSA, EPSB      *
C*******************************
      ANORM = 0.0
      BNORM = 0.0
C
      DO 30 I = 1,N
      ANI =0.0
      IF (I.NE. 1) ANI = ABS (A(I,I-1))
      BNI =0.0
C
      DO 20 J = I, N
      ANI = ANI + ABS(A(I,J))
      BNI = BNI + ABS(B(I,J))
 20   CONTINUE
C
      IF (ANI.GT. ANORM) ANORM = ANI
      IF (BNI .GT.BNORM) BNORM = BNI
 30   CONTINUE
C
      IF (ANORM .EQ. 0.0) ANORM =1.0
```

```
      IF (BNORM .EQ. 0.0) BNORM =1.0
      EP = EPS1
      IF (EP .GT. 0.0) GO TO 50
C*********************************************************
C      COMPUTE ROUNDOFF LEVEL IF EPSI IS ZERO           *
C*********************************************************
      EP = 1.0
 40   EP = EP /2.0
      IF (1.0 + EP .GT. 1.0) GO TO 40
 50   EPSA = EP *ANORM
      EPSB = EP * BNORM
C*********************************************************
C      REDUCE A TO QUASI - TRIANGULAR FORM, WHILE       *
C           KEEPING B TRIANGULAR                        *
C*********************************************************
      LOR1 =1
      ENORN = N
      EN = N
C************************
C      BEGIN QZ STEP    *
C************************
 60   IF (EN .LE. 2) GO TO 1001
      IF (.NOT. MATZ) ENORN = EN
      ITS = 0
      NA = EN -1
      ENM2 = NA -1
 70   ISH = 2
C*************************************************
C      CHECK FOR CONVERGENCE OR REDUCIBILITY     *
C             FOR L=EN STEP -1 UNTIL 1 DO --     *
C*************************************************
      DO 80 LL = 1, EN
      LM1 = EN -LL
```

```
      L = LM1 +1
      IF (L.EQ. 1) GO TO 95
      IF (ABS(A(L,LM1)).LE. EPSA) GO TO 90
 80   CONTINUE
C
90    A(L,LM1) = 0.0
      IF (L.LT. NA) GO TO 95
C*****************************************
C       1 -BY-1 OR 2 - BY 2 BLOCK ISOLATED *
C*****************************************
      EN = LM1
      GO TO 60
C************************************
C       CHECK FOR SMALL TOP OF B    *
C************************************
 95   LD = L
 100  L1 = L + 1
      B11 = B(L,L)
      IF (ABS(B11) .GT. EPSB) GO TO 120
      B(L,L) = 0.0
      S = ABS(A(L,L)) + ABS(A(L1,L))
      U1 = A(L,L) / S
      U2 = A(L1,L) / S
      R = SIGN(SQRT(U1*U1+U2*U2),U1)
      V1 = -(U1+R) / R
      V2 = -U2 /R
      U2 =V2 /V1
C
      DO 110 J=L, ENORM
        T = A(L,J) +U2 * A(L1,J)
        A(L,J) = A(L,J) + T * V1
        A(L1,J) = A(L1,J) + T * V2
        T = B(L,J) +U2 * B(L1,J)
```

```
          B(L,J) = B(L,J) + T * V1
          B(L1,J) = B(L1,J) + T * V2
  110   CONTINUE
C
        IF (L.NE. 1) A(L,LM1) = -A(L,LM1)
        LM1=L
        L= L1
        GO TO 90
  120   A11 = A(L,L) / B11
        A21 = A(L1,L) /B11
        IF (ISH .EQ. 1) GO TO 140
C  ***** ITERATION STRATEGY  **********
        IF (ITS .EQ. 50) GO TO 1000
        IF (ITS .EQ. 10) GO TO 155
C *******  DETERMINE TYPE OF SHIFT  *******
        B22 = B(L1,L1)
        IF (ABS(B22) .LT. EPSB)  B22 = EPSB
        B33 = B(NA,NA)
        IF (ABS(B33) .LT.  EPSB)  B33 = EPSB
        B44 = B(EN,EN)
        IF (ABS(B44) .LT.  EPSB)  B44 = EPSB
        A33 = A(NA,NA) / B33
        A34 = A(NA,EN) / B44
        A43 = A(EN,NA) / B33
        A44 = A(EN,EN) / B44
        B34 = B(NA,EN) / B44
        T = 0.5 * (A43 * B34 - A33 -A44)
        R = T*T +A34 *A43 -A33 *A44
        IF (R .LT. 0.0) GO TO 150
C  *******  DETERMINE SINGLE SHIFT ZEROTH COLUMN OF A ******
        ISH = 1
        R = SQRT(R)
        SH = -T + R
```

```
      S = -T -R
      IF (ABS(S - A44) .LT. ABS(SH -A44)) SH = S
C  ***** LOOK FOR TWO CONSECUTIVE SMALL
C        SUB-DIAGONAL ELEMENTS OF A.
C        FOR L = EN -2 STEP -1 UNTIL LD DO --- **********
      DO 130 LL = LD, ENM2
        L = ENM2 + LD -LL
        IF (L .EQ. LD) GO TO 140
        LM1 = L -1
        L1 = L + 1
        T = A(L,L)
        IF (ABS(B(L,L)) .GT. EPSB) T = T-SH * B(L,L)
        IF (ABS(A(L,LM1)) .LE. ABS(T/A(L1,L)) * EPSA) GO TO 100
 130  CONTINUE
C
 140  A1 = A11 -SH
      A2 =A21
      IF (L .NE. LD) A(L,LM1) = -A(L,LM1)
      GO TO 160
C*****************************************************
C      DETERMINE DOUBLE SHIFT ZEROTH COLUMN OF A      *
C*****************************************************
 150  A12 = A(L,L1) / B22
      A22 = A(L1,L1) / B22
      B12 = B(L,L1) /B22
      A1 = ((A33 -A11) * (A44 - A11) - A34 * A43 +A43 * B34 * A11)
     X       / A21 + A12 -A11 * B12
      A2 = (A22-A11) - A21 *B12 - (A33 -A11) - (A44 -A11)
     X       + A43 * B34
      A3 = A(L1 +1, L1) / B22
      GO TO 160
C*******************************
C      AD HOC SHIFT            *
```

```
C*********************************
 155  A1 = 0.0
      A2 = 1.0
      A3 = 1.1605
 160  ITS =  ITS + 1
      IF (.NOT.MATZ) LOR1 = LD
C*********************************
C      MAIN LOOP                  *
C*********************************
      DO 260 K = L, NA
      NOTLAS =  K .NE. NA .AND. ISH .EQ. 2
      K1 = K + 1
      K2 = K + 2
      KM1 = MAXO(K-1,L)
      LL = MINO(EN,K1+ISH)
      IF (NOTLAS) GO TO 190
C*********************************
C      ZERO A(K+1,K-1)            *
C*********************************
      IF (K .EQ. L) GO TO 170
      A1 = A(K,KM1)
      A2 = A(K1,KM1)
 170  S = ABS(A1) + ABS(A2)
      IF (S .EQ. 0.0) GO TO 70
      U1 = A1 /S
      U2 = A2 / S
      R = SIGN(SQRT(U1*U1+U2*U2),U1)
      V1 = -(U1+R) /R
      V2 = -U2 /R
      U2 = V2 / V1
C
      DO 180 J = KM1,ENORN
        T = A(K,J) + U2*A(K1,J)
```

```
          A(K,J) = A(K,J) + T*V1
          A(K1,J) = A(K1,J) + T *V2
          T = B(K,J) + U2*B(K1,J)
          B(K,J) = B(K,J) + T*V1
          B(K1,J) = B(K1,J) + T *V2
  180   CONTINUE
C
        IF (K.NE. L) A(K1,KM1) = 0.0
        GO TO 240
C*******************************************
C       ZERO A(K+1,K-1) AND A(K+2,K-1)        *
C*******************************************
  190   IF (K .EQ. L) GO TO 200
        A1 = A(K,KM1)
        A2 = A(K1,KM1)
        A3 = A(K2,KM1)
  200   S = ABS(A1) + ABS(A2) + ABS(A3)
        IF (S .EQ. 0.0) GO TO 260
        U1 = A1 / S
        U2 = A2 /S
        U3 = A3 /S
        R = SIGN(SQRT(U1*U1+U2*U2+U3*U3),U1)
        V1 = -(U1 +R) / R
        V2 = -U2 /R
        V3 = - U3 / R
        U2 =  V2 /V1
        U3 = V3 /V1
C
        DO 210 J = KM1, ENORM
        T = A(K,J) + U2 * A(K1,J) + U3 *A(K2,J)
        A(K,J) = A(K,J) + T * V1
        A(K1,J) = A(K1,J) + T *V2
        A(K2,J) = A(K2,J) + T * V3
```

```
      T = B(K,J) + U2 * B(K1,J) + U3 * B(K2,J)
      B(K,J) = B(K,J) + T * V1
      B(K1,J) = B(K1,J) +T * V2
      B(K2,J) = B(K2,J) +T * V3
 210  CONTINUE
C
      IF (K .EQ. L) GO TO 220
      A(K1,KM1) = 0.0
      A(K2,KM1) = 0.0
C*********************************************
C       ZERO B(K+2,K+1) AND B(K+2,K)        *
C*********************************************
 220  S = ABS(B(K2,K2)) + ABS(B(K2,K1)) + ABS(B(K2,K))
      IF (S .EQ. 0.0) GO TO 240
      U1 = B(K2,K2) / S
      U2 = B(K2,K1) / S
      U3 = B(K2,K) / S
      R =SIGN(SQRT(U1*U1+U2*U2+U3*U3),U1)
      V1 = -(U1+R) / R
      V2 = -U2 / R
      V3 = -U3 /R
      U2 = V2 /V1
      U3 = V3 /V1
C
      DO 230 I = LOR1, LL
      T = A(I,K2) + U2 * A(I,K1) + U3 * A(I,K)
      A(I,K2) = A(I,K2) +T * V1
      A(I,K1) = A(I,K1) + T * V2
      A(I,K) = A(I,K) +T *V3
      T = B(I,K2) + U2 * B(I,K1) + U3 * B(I,K)
      B(I,K2) = B(I,K2) +T * V1
      B(I,K1) = B(I,K1) + T * V2
      B(I,K) = B(I,K) +T *V3
```

```
 230  CONTINUE
C
      B(K2,K) = 0.0
      B(K2,K1) = 0.0
      IF (.NOT. MATZ) GO TO 240
C
      DO 235 I = 1,N
      T = Z(I,K2) + U2 * Z(I,K1) + U3 * Z(I,K)
      Z(I,K2) = Z(I,K2) +T * V1
      Z(I,K1) = Z(I,K1) + T * V2
      Z(I,K) = Z(I,K) +T *V3
 235  CONTINUE
C*********************************
C       ZERO B(K+1,K)             *
C*********************************
 240  S = ABS(B(K1,K1)) + ABS(B(K1,K))
      IF (S .EQ. 0.0) GO TO 260
      U1 = B(K1,K1) / S
      U2 = B(K1,K) / S
      R =SIGN(SQRT(U1*U1+U2*U2),U1)
      V1 = -(U1+R) / R
      V2 = -U2 / R
      U2 = V2 /V1
C
      DO 250 I = LOR1,LL
      T = A(I,K1) + U2 * A(I,K)
      A(I,K1) = A(I,K1) +T * V1
      A(I,K) = A(I,K) + T * V2
      T = B(I,K1) + U2 * B(I,K)
      B(I,K1) = B(I,K1) +T * V1
      B(I,K) = B(I,K) + T * V2
 250  CONTINUE
C
```

```
      B(K1,K) = 0.0
      IF(.NOT.MATZ) GO TO 260
C
      DO 255 I = 1,N
      T = Z(I,K1) + U2 * Z(I,K)
      Z(I,K1) = Z(I,K1) +T * V1
      Z(I,K) = Z(I,K) + T * V2
 255  CONTINUE
C
 260  CONTINUE
C**************************
C      END QZ STEP         *
C**************************
      GO TO 70
C*********************************************************
C      SET ERROR -- NEITHER BOTTOM SUBDIAGONAL ELEMENT   *
C            HAS BECOME NEGLIGIBLE AFTER 50 ITERATIONS   *
C*********************************************************
 1000 IERR = EN
C***********************************************
C      SAVE EPSB FOR USE BY QZVAL AND QZVEC     *
C***********************************************
 1001 IF (N .GT. 1) B(N,1) = EPSB
      RETURN
      END
      SUBROUTINE QZVAL(NM,N,A,B,ALFR,ALFI,BETA,MATZ,Z)
C
      INTEGER I,J,N,EN,NA,NM,NN,ISW
      REAL A(NM,N),B(NM,N),ALFR(N),ALFI(N),BETA(N),Z(NM,N)
      REAL C,D,E,R,S,T,AN,A1,A2,BN,CQ,CZ,DI,DR,EI,TI,TR,U1,U2,
     X       V1,V2,A1I,A11,A12,A2I,A21,A22,B11,B12,B22,SQI,SQR,
     X       SSI,SSR,SZI,SZR,A11I,A11R,A12I,A12R,A22I,A22R,EPSB
      REAL SQRT,ABS,SIGN
```

```
      LOGICAL MATZ
C
      EPSB=B(N,1)
      ISW=1
C*************************************************************
C       FIND EIGENVALUES OF QUASI-TRIANGULAR MATRICES       *
C            FOR EN=N STEP -1 UNTIL 1 DO --                 *
C*************************************************************
      DO 510 NN=1,N
         EN= N+1 -NN
         NA= EN-1
         IF (ISW.EQ.2) GO TO 505
         IF (EN.EQ.1) GO TO 410
         IF (A(EN,NA)  .NE. 0.0)  GO TO 420
C*********************************************
C       1-BY-1 BLOCK, ONE REAL ROOT           *
C*********************************************
  410    ALFR(EN) = A(EN,EN)
         IF (B(EN,EN) .LT. 0.0) ALFR(EN) = -ALFR(EN)
         BETA(EN) = ABS(B(EN,EN))
         ALFI(EN) =0.0
         GO TO 510
C*******************************
C       2 -BY- 2 BLOCK        *
C*******************************
  420    IF (ABS(B(NA,NA)) .LE. EPSB) GO TO 455
         IF (ABS(B(EN,EN)) .GT. EPSB)GO TO 430
         A1 = A(EN,EN)
         A2 = A(EN,NA)
         BN = 0.0
         GO TO 435
  430    AN = ABS (A(NA,NA)) + ABS(A(NA,EN)) + ABS(A(EN,NA))
     X      + ABS(A(EN,EN))
```

```
          BN = ABS(B(NA,NA)) + ABS(B(NA,EN)) + ABS(B(EN,EN))
          A11 = A(NA,NA) / AN
          A12 = A(NA,EN) / AN
          A21 = A(EN,NA) / AN
          A22 = A(EN,EN) / AN
          B11 = B(NA,NA) / BN
          B12 = B(NA,EN) / BN
          B22 = B(EN,EN) / BN
          E = A11 /B11
          EI = A22 / B22
          S = A21 / (B11 * B22)
          T = (A22 - E * B22) / B22
          IF (ABS(E).LE. ABS(EI)) GO TO 431
          E = EI
          T = (A11 -E * B11) / B11
  431     C = 0.5 * (T-S * B12)
          D = C * C +S * (A12 -E *B12)
          IF (D.LT.0.0) GO TO 480
C*********************************************
C                TWO REAL ROOTS               *
C         ZERO BOTH A(EN,NA) AND B(EN,NA)     *
C*********************************************
          E =E + (C+SIGN(SQRT(D),C))
          A11 = A11 -E *B11
          A12 = A12 -E *B12
          A22 = A22 -E *B22
          IF (ABS(A11) + ABS(A12) .LT.
     X       ABS(A21) + ABS(A22)) GO TO 432
          A1 = A12
          A2 = A11
          GO TO 435
  432     A1 = A22
A2 = A21
```

```
C***********************************
C        CHOOSE AND APPLY REAL Z  *
C***********************************
  435    S = ABS(A1) +ABS(A2)
         U1 =A1 /S
         U2 = A2 /S
         R =SIGN(SQRT(U1*U1 + U2*U2),U1)
         V1 = -(U1 + R) / R
         V2 = -U2 /R
         U2 = V2 /V1
C
         DO 440 I = 1, EN
         T = A(I,EN) + U2 * A(I,NA)
         A(I,EN) = A(I,EN) + T *V1
         A(I,NA) = A(I,NA) + T *V2
         T = B(I,EN) + U2 * B(I,NA)
         B(I,EN) = B(I,EN) + T*V1
         B(I,NA) = B(I,NA) + T*V2
 440     CONTINUE
C
         IF (.NOT.MATZ) GO TO 450
C
         DO 445 I=1,N
         T = Z(I,EN) + U2 * Z(I,NA)
         Z(I,EN) = Z(I,EN) + T *V1
         Z(I,NA) = Z(I,NA) + T *V2
 445     CONTINUE
C
 450     IF (BN.EQ.0.0) GO TO 475
         IF (AN .LT. ABS(E)*BN) GO TO 455
         A1=B(NA,NA)
         A2 = B(EN,NA)
         GO TO 460
```

```
  455     A1=A(NA,NA)
          A2 = A(EN,NA)
C*********************************
C       CHOOSE AND APPLY REAL Q  *
C*********************************
  460     S = ABS(A1) +ABS(A2)
          IF (S.EQ. 0.0) GO TO 475
          U1 = A1 /S
          U2 =A2/S
          R = SIGN (SQRT(U1*U1+U2*U2),U1)
          V1 = -(U1+R) / R
          V2 = -U2 /R
          U2 =V2 /V1
C
          DO 470 J=NA,N
          T = A(NA,J) + U2 * A(EN,J)
          A(NA,J) = A(NA,J) + T *V1
          A(EN,J) = A(EN,J) + T *V2
          T = B(NA,J) + U2 * B(EN,J)
          B(NA,J) = B(NA,J) + T*V1
          B(EN,J) = B(EN,J) + T*V2
 470      CONTINUE
C
 475      A(EN,NA) =0.0
          B(EN,NA) =0.0
          ALFR(NA) = A(NA,NA)
          ALFR(EN) = A(EN,EN)
          IF (B(NA,NA).LT. 0.0) ALFR(NA) = -ALFR(NA)
          IF (B(EN,EN) .LT.0.0) ALFR(EN) =-ALFR(EN)
          BETA(NA) = ABS(B(NA,NA))
          BETA(EN) = ABS(B(EN,EN))
          ALFI(EN) = 0.0
          ALFI(NA) = 0.0
```

```
          GO TO 505
C*****************************
C         TWO COMPLEX ROOTS   *
C*****************************
 480      E =E +C
          EI = SQRT(-D)
          A11R = A11 - E * B11
          A11I = EI *B11
          A12R = A12 - E * B12
          A12I = EI *B12
          A22R = A22 - E * B22
          A22I = EI *B22
          IF(ABS(A11R) + ABS(A11I) + ABS(A12R) + ABS(A12I).LT.
     X      ABS(A21) + ABS(A22R) + ABS(A22I) ) GO TO 482
          A1 = A12R
          A1I = A12I
          A2 = -A11R
          A2I = -A11I
          GO TO 485
  482     A1 = A22R
          A1I = A22I
          A2 = -A21
          A2I = 0.0
C*****************************
C         CHOOSE COMPLEX Z    *
C*****************************
  485     CZ = SQRT(A1*A1+A1I*A1I)
          IF (CZ.EQ. 0.0) GO TO 487
          SZR = (A1*A2 +A1I* A2I) /CZ
          SZI = (A1*A2I - A1I* A2) /CZ
          R = SQRT(CZ*CZ + SZR *SZR +SZI*SZI)
          CZ = CZ /R
          SZR = SZR /R
```

```
          SZI = SZI /R
          GO TO 490
   487    SZR = 1.0
          SZI = 0.0
   490    IF (AM .LT. (ABS(E) +EI) *BM) GO TO 492
          A1 = CZ *B11 + SZR *B12
          A1I = SZI *B12
          A2 = SZR *B22
          A2I = SZI *B22
          GO TO 495
   492    A1 = CZ * A11 +SZR * A12
          A1I = SZI * A12
          A2 = CZ * A21 +SZR * A22
          A2I = SZI * A22
C*****************************
C         CHOOSE COMPLEX Q     *
C*****************************
   495    CQ = SQRT(A1*A1+A1I*A1I)
          IF (CQ.EQ. 0.0) GO TO 497
          SQR = (A1 * A2 + A1I * A2I) / CQ
          SQI  = (A1 *A2I - A1I * A2) / CQ
          R = SQRT(CQ*CQ+SQR*SQR+SQI*SQI)
          CQ = CQ /R
          SQR = SQR /R
          SQI = SQI /R
          GO TO 500
   497    SQR = 1.0
          SQI = 0.0
C*********************************************************
C         COMPUTE DIAGONAL ELEMENTS THAT WOULD RESULT  *
C               IF TRANSFORMATIONS WERE APPLIED        *
C*********************************************************
   500    SSR = SQR * SZR  +  SQI * SZI
```

```
          SSI = SQR * SZI - SQI * SZR
          I =1
          TR = CQ * CZ * A11 + CQ * SZR * A12 + SQR* CZ * A21
     X         + SSR * A22
          TI = CQ * SZI * A12 - SQI * CZ * A21+ SSI * A22
          DR = CQ * CZ * B11 + CQ * SZR * B12 + SSR * B22
          DI = CQ *SZI * B12 + SSI * B22
          GO TO 503
  502     I = 2
          TR = SSR * A11 - SQR * CZ *A12 - CQ * SZR * A21
     X          + CQ * CZ * A22
          TI = -SSI * A11 - SQI *CZ * A12 + CQ * SZI * A21
          DR = SSR * B11 - SQR * CZ * B12 + CQ * CZ * B22
          DI = -SSI * B11 - SQI * CZ * B12
  503     T  = TI * DR - TR * DI
          J = NA
          IF (T .LT. 0.0) J= EN
          R = SQRT( DR*DR+DI*DI)
          BETA(J)= BN *R
          ALFR(J) = AN * (TR * DR + TI * DI) / R
          ALFI(J) = AN * T /R
          IF (I .EQ. 1) GO TO 502
  505     ISW = 3 -ISW
  510 CONTINUE
C
      RETURN
      END
C
SUBROUTINE QZVEC(NM,N,A,B,ALFR,ALFI,BETA,Z)
C
INTEGER I,J,K,M,N,EN,II,JJ,NA,NM,NN,ISW,ENM2
REAL A(NM,N),B(NM,N),ALFR(N),ALFI(N),BETA(N),Z(NM,N)
REAL D,Q,R,S,T,W,X,Y,DI,DR,RA,RR,SA,TI,TR,T1,T2,W1,X1,ZZ,Z1,
```

```
     1 ALFM,ALMI,ALMR,BETM,EPSB
REAL SQRT,ABS
C
EPSB=B(N,1)
ISW=1
C*****************************************
C       FOR EN=N STEP -1 UNTIL 1 DO --    *
C*****************************************
DO 800 NN=1,N
EN=N+1-NN
NA=EN-1
IF (ISW.EQ.2) GO TO 795
IF (ALFI(EN).NE.0.0) GO TO 710
C************************************
C       REAL VECTOR                *
C************************************
   M=EN
   B(EN,EN)=1.0
   IF (NA.EQ.0) GO TO 800
   ALFM=ALFR(M)
   BETM=BETA(M)
C*****************************************
C FOR I=EN-1 STEP -1 UNTIL 1 DO -- *
C*****************************************
   DO 700 II=1,NA
      I=EN-II
      W=BETM*A(I,I)-ALFM*B(I,I)
      R=0.0
C
      DO 610 J=M,EN
610       R=R+(BETM*A(I,J)-ALFM*B(I,J))*B(J,EN)
C
      IF (I.EQ.1.OR.ISW.EQ.2) GO TO 630
```

```
      IF (BETM*A(I,I-1).EQ.0.0) GO TO 630
      ZZ=W
      S=R
      GO TO 690
630       M=I
      IF(ISW.EQ.2) GO TO 640
C**********************************
C REAL 1-BY-1 BLOCK          *
C**********************************
      T=W
      IF(W.EQ.0.0) T=EPSB
      B(I,EM)=-R/T
      GO TO 700
C**********************************
C REAL 2-BY-2 BLOCK          *
C**********************************
640       X=BETM*A(I,I+1)-ALFM*B(I,I+1)
      Y=BETM*A(I+1,I)
      Q=W*ZZ-X*Y
      T=(X*S-ZZ*R)/Q
      B(I,EM)=T
      IF(ABS(X).LE.ABS(ZZ)) GO TO 650
      B(I+1,EM)=(-R-W*T)/X
      GO TO 690
650       B(I+1,EM)=(-S-Y*T)/ZZ
690       ISW=3-ISW
700   CONTINUE
C************************************
C        END REAL VECTOR             *
C************************************
      GO TO 800
C************************************
C        COMPLEX VECTOR              *
```

```
C***********************************
710       M=NA
      ALMR=ALFR(M)
      ALMI=ALFI(M)
      BETM=BETA(M)
C*****************************************************
C        LAST VECTOR COMPONENT CHOSEN IMAGINARY SO  *
C       EIGENVECTOR MATRIX IS TRIANGULAR      *
C*****************************************************
      Y=BETM*A(EN,NA)
      B(NA,NA)=-ALMI*B(EN,EN)/Y
      B(NA,EN)=(ALMR*B(EN,EN)-BETM*A(EN,EN))/Y
      B(EN,NA)=0.0
      B(EN,EN)=1.0
      ENM2=NA-1
      IF (ENM2.EQ.0) GO TO 795
C*******************************************************
C  FOR I=EN-2 STEP -1 UNTIL 1 DO --            *
C*******************************************************
      DO 790 II=1,ENM2
I=NA-II
W=BETM*A(I,I)-ALMR*B(I,I)
W1=-ALMI*B(I,I)
RA=0.0
SA=0.0
C
DO 760 J=M,EN
  X=BETM*A(I,J)-ALMR*B(I,J)
  X1=-ALMI*B(I,J)
  RA=RA+X*B(J,NA)-X1*B(J,EN)
  SA=SA+X*B(J,EN)+X1*B(J,NA)
760 CONTINUE
C
```

```
IF(I.EQ.1.OR.ISW.EQ.2) GO TO 770
IF(BETM*A(I,I-1).EQ.0.0) GO TO 770
ZZ=W
Z1=W1
R=RA
S=SA
ISW=2
GO TO 790
770 M=I
IF(ISW.EQ.2) GO TO 780
C*******************************************
C   COMPLEX 1-BY-1 BLOCK            *
C*******************************************
TR=-RA
TI=-SA
773 DR=W
DI=W1
C*************************************************
C   COMPLEX DIVIDE (T1,T2)=(TR,TI)/(DR,DI)  *
C*************************************************
775 IF(ABS(DI).GT.ABS(DR)) GO TO 777
RR=DI/DR
D=DR+DI*RR
T1=(TR+TI*RR)/D
T2=(TI-TR*RR)/D
GO TO (787,782),ISW
777 RR=DR/DI
D=DR*RR+DI
T1=(TR*RR+TI)/D
T2=(TI*RR-TR)/D
GO TO (787,782),ISW
C*****************************************
C   COMPLEX 2-BY-2 BLOCK            *
```

```
C*****************************************
780 X=BETM*A(I,I+1)-ALMR*B(I,I+1)
X1=-ALMI*B(I,I+1)
Y=BETM*A(I+1,I)
TR=Y*RA-W*R+W1*S
TI=Y*SA-W*S-W1*R
DR=W*ZZ-W1*Z1-X*Y
DI=W*Z1+W1*ZZ-X1*Y
IF (DR.EQ.0.0.AND.DI.EQ.0.0) DR=EPSB
GO TO 775
782 B(I+1,NA)=T1
B(I+1,EN)=T2
ISW=1
IF(ABS(Y).GT.ABS(W)+ABS(W1)) GO TO 785
TR=-RA-X*B(I+1,NA)+X1*B(I+1,EN)
TI=-SA-X*B(I+1,EN)-X1*B(I+1,NA)
GO TO 773
785 T1=(-R-ZZ*B(I+1,NA)+Z1*B(I+1,EN))/Y
T2=(-S-ZZ*B(I+1,EN)-Z1*B(I+1,NA))/Y
787 B(I,NA)=T1
B(I,EN)=T2
790      CONTINUE
C*************************************
C    END COMPLEX VECTOR         *
C*************************************
795 ISW=3-ISW
800 CONTINUE
C*********************************************************
C END BACK SUBSTITUTION                          *
C TRANSFORM TO ORIGINAL COORDINATE SYSTEM        *
C FOR J=N STEP -1 UNTIL 1 DO --                  *
C*********************************************************
DO 880 JJ=1,N
```

```
J=M+1-JJ
C
DO 880 I=1,M
        ZZ=0.0
C
        DO 860 K=1,J
860 ZZ=ZZ+Z(I,K)*B(K,J)
Z(I,J)=ZZ
880 CONTINUE
C*************************************************************
C       NORMALIZE SO THAT MODULUS OF LARGEST                  *
C COMPONENT OF EACH VECTOR IS 1                          *
C       (ISW IS 1 INITIALLY FROM BEFORE)                      *
C*************************************************************
DO 950 J=1,M
D=0.0
IF(ISW.EQ.2) GO TO 920
        IF(ALFI(J).NE.0.0) GO TO 945
DO 890 I=1,M
IF(ABS(Z(I,J)).GT.D) D=ABS(Z(I,J))
890 CONTINUE
C
DO 900 I=1,M
900 Z(I,J)=Z(I,J)/D
C
GO TO 950
C
920 DO 930 I=1,M
  R=ABS(Z(I,J-1))+ABS(Z(I,J))
  IF(R.NE.0.0)R=R*SQRT((Z(I,J-1)/R)**2+
     X   (Z(I,J)/R)**2)
  IF(R.GT.D) D=R
930 CONTINUE
```

```
C
DO 940 I=1,N
   Z(I,J-1)=Z(I,J-1)/D
   Z(I,J)=Z(I,J)/D
940 CONTINUE
C
945 ISW=3-ISW
950 CONTINUE
C
RETURN
END
```

B.6 Program for transforming correlated data to uncorrelated variables

```
C*************************************************************
C THIS PROGRAM NORMALIZES THE CORRELATED INPUT DATA AND    *
C TRANSFORMS THEM TO UNCORRELATED VARIABLES                *
C*************************************************************
PARAMETER(NP=12)
DIMENSION UT1NB(3000),UT2NB(3000),PSINB(3000,NP),
      1 UT1B(3000),UT2B(3000),PSIB(3000,NP),ZR(NP,NP),
      2 ZRT(NP,NP),CNB(3000,NP),CB(3000,NP),XMEANNB(NP),
      3 XMEANB(NP),XMEAN(NP),VARNB(NP),VARB(NP),VAR(NP)
OPEN(2,FILE='f2')
OPEN(3,FILE='f3')
OPEN(4,FILE='f6')
OPEN(5,FILE='f8')
OPEN(6,FILE='f9')
C READ IN RAW DATA
```

```
READ(2,135)NNBEHA
135 FORMAT(I4)
DO 101 J=1,NNBEHA
READ(2,25)UT1NB(J),UT2NB(J),(PSINB(J,I),I=1,NP)
101 CONTINUE
25      FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F5.2,2X),/)
READ(3,135)NBEH
DO 15 J=1,NBEH
READ(3,25)UT1B(J),UT2B(J),(PSIB(J,I),I=1,NP)
15 CONTINUE
DO 199 J=1,NP
199 READ(4,230) (ZR(J,K),K=1,NP)
230 FORMAT(4(/,3(E13.5,2X)))
C CALCULATE MEANS AND VARIANCES OF RAW DATA
DO 903 I=1,NP
XMEANNB(I)=0.0
XMEANB(I)=0.0
DO 902 J=1,NNBEHA
902 XMEANNB(I)=XMEANNB(I)+PSINB(J,I)
DO 802 J=1,NBEH
802 XMEANB(I)=XMEANB(I)+PSIB(J,I)
XMEAN(I)=XMEANNB(I)+XMEANB(I)
XMEAN(I)=XMEAN(I)/FLOAT(NNBEHA+NBEH)
903 CONTINUE
DO 905 I=1,NP
VARNB(I)=0.0
VARB(I)=0.0
DO 906 J=1,NNBEHA
906 VARNB(I)=VARNB(I)+(PSINB(J,I)-XMEAN(I))**2.0
DO 806 J=1,NBEH
806 VARB(I)=VARB(I)+(PSIB(J,I)-XMEAN(I))**2.0
VAR(I)=VARNB(I)+VARB(I)
VAR(I)=VAR(I)/(FLOAT(NNBEHA+NBEH)-1.0)
```

```
905 CONTINUE
C
C CALCULATE VAR-COVAR MATRICES FOR BEH,NON-BEH AND
C UNCLASSIFIED DATA
DO 10 I=1,NBEH
DO 4 K=1,NP
PSIB(I,K)=(PSIB(I,K)-XMEAN(K))/SQRT(VAR(K))
4 CONTINUE
10 CONTINUE
DO 11 I=1,NNBEHA
DO 6 K=1,NP
PSINB(I,K)=(PSINB(I,K)-XMEAN(K))/SQRT(VAR(K))
6 CONTINUE
11 CONTINUE
DO 166 I=1,NP
DO 166 L=1,NP
ZRT(I,L)=ZR(L,I)
166 CONTINUE
DO 66 I=1,NNBEHA
DO 66 K=1,NP
SUM=0.
DO 88 L=1,NP
88 SUM=SUM+PSINB(I,L)*ZR(L,K)
CNB(I,K)=SUM
66 CONTINUE
DO 661 I=1,NBEH
DO 661 K=1,NP
SUM=0.
DO 881 L=1,NP
881 SUM=SUM+PSIB(I,L)*ZR(L,K)
CB(I,K)=SUM
661 CONTINUE
WRITE(5,135)NNBEHA
```

```
DO 1010 J=1,NNBEHA
WRITE(5,27)UT1NB(J),UT2NB(J),(CNB(J,I),I=1,NP)
1010 CONTINUE
WRITE(6,135)NBEH
DO 150 J=1,NBEH
WRITE(6,27)UT1B(J),UT2B(J),(CB(J,I),I=1,NP)
150 CONTINUE
27      FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F6.2,2X),/)
C27 FORMAT(2(F4.2,2X),6(/,6(E10.4,2X)),/,F8.2,/,F6.2,//)
STOP
END
```

B.7 Program for calculating normalized sensitivity indices

```
C*************************************************************
C  THIS PROGRAM CALCULATES NORMALIZED (d_{mn})s FOR THE *
C  BEHAVIOUR AND NON-BEHAVIOUR CLASSIFICATIONS           *
C*************************************************************
C ******************************************************
C *                                                    *
C *          KOLMOGOROV-SMIRNOV TEST                   *
C *                                                    *
C ******************************************************
C
      PROGRAM KSTWO
C
C
C Kolmogorov-Smirnov Test, where D is Dm,n
C to test for a difference in parameter distributions
```

```
C
C DATA1 = #1 set of data
C DATA2 = #2 set of data
C N1    = length of DATA1
C N2    = length of DATA2
C D     = Dm,n (test parameter)
C PROB  = cdf probability
C
C
      IMPLICIT REAL (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
      DIMENSION DATA1(9999),DATA2(9999),DD(100),
     1DATANB(3000,100),DATABH(3000,100),PP(100),
     2EIG(12,12),DEIG(12),SEIG(12)
C
C
      OPEN(5,FILE='f8')
      OPEN(6,FILE='f9')
      OPEN(7,FILE='f6')
      OPEN(8,FILE='f10')
C     OPEN(7,FILE='F1.DAT')
C     OPEN(8,FILE='F2.DAT')
      WRITE (8,90)
      DO 144 J=1,12
144   READ(7,230) (EIG(J,K),K=1,12)
      READ(5,450)JN
      DO 500 I=1,JN
      READ(5,550)(DATANB(I,NP),NP=1,14)
500   CONTINUE
      READ(6,450)JB
      DO 600 I=1,JB
      READ(6,550)(DATABH(I,NP),NP=1,14)
600   CONTINUE
```

```
      DO 400 NP=1,14
      DO 700 I=1,JN
      DATA1(I)=DATANB(I,NP)
700   CONTINUE
      N1=JN
      DO 800 I=1,JB
      DATA2(I)=DATABH(I,NP)
800   CONTINUE
      N2=JB
C
C     WRITE (*,92)
      CALL SORT(N1,DATA1,NP)
C     WRITE (*,94)
      CALL SORT(N2,DATA2,NP)
C
C     WRITE (*,95)
C     DO 50 I=1,N1
C     WRITE (7,250) DATA1(I)
C50   CONTINUE
C     DO 60 I=1,N2
C     WRITE (8,250) DATA2(I)
C60   CONTINUE
C
C     WRITE (*,96)
      EN1=N1
      EN2=N2
      J1=1
      J2=1
      FO1=0.
      FO2=0.
      D=0.
1     IF (J1.LE.N1.AND.J2.LE.N2) THEN
      IF (DATA1(J1).LT.DATA2(J2)) THEN
```

```
      FN1=J1/EN1
      DT=AMAX1(ABS(FN1-FO2),ABS(FO1-FO2))
      IF (DT.GT.D) D=DT
      FO1=FN1
      J1=J1+1
      ELSE
      FN2=J2/EN2
      DT=AMAX1(ABS(FN2-FO1),ABS(FO2-FO1))
      IF (DT.GT.D) D=DT
      FO2=FN2
      J2=J2+1
      END IF
      GO TO 1
      END IF
C
C
      PROB=PROBKS(SQRT(EN1*EN2/(EN1+EN2))*D)
      PP(NP)=PROB*100
      DD(NP)=D*100
C     WRITE (*,120) D*100.
C
C
400   CONTINUE
      DMN1=DD(1)
      DMN2=DD(2)
      DO 122 I=3,40
      DD(I-2)=DD(I)
      PP(I-2)=PP(I)
122   CONTINUE
      DO 444 K=1,12
      SUM=0.0
      DO 443 J=1,12
443   SUM=SUM+ABS(EIG(J,K))
```

```
      SEIG(K)=SUM
444   CONTINUE
      DO 1111 J=1,12
      SUM=0.0
      DO 1112 K=1,12
C     SUM=SUM+(EIG(J,K)**2.0)*DD(K)
      SUM=SUM+ABS(EIG(J,K))/SEIG(K)*DD(K)
1112  CONTINUE
      DEIG(J)=SUM
1111  CONTINUE
      DO 2222 J=1,12
      D=DEIG(J)/100.0
      PROB=PROBKS(SQRT(EN1*EN2/(EN1+EN2))*D)
      PP(J)=PROB*100.0
2222  CONTINUE
      SNN=SQRT(EN1*EN2/(EN1+EN2))
      DO 1201 KKI=1,12
1201  DEIG(KKI)=DEIG(KKI)/1.36*SNN/100
      SUM=0.0
      DO 3333 J=1,12
3333  SUM=SUM+DEIG(J)
      DMEAN=SUM/12.0
      WRITE(8,332)JB,JN
      WRITE(8,133) DMN1,DMN2
      WRITE(8,333)
      DO 334 NP=1,12
      WRITE(8,335)NP,DEIG(NP),PP(NP)
 334  CONTINUE
      WRITE(8,552) DMEAN
C
      WRITE (8,*) 'significance level    critical D value'
      DN=1.63/SNN*100
      NA=99
```

```
      WRITE (8,260) NA,DN
      DN=1.36/SNN*100
      NA=95
      WRITE (8,260) NA,DN
      DN=1.22/SNN*100
      NA=90
      WRITE (8,260) NA,DN
      DN=1.07/SNN*100
      NA=80
      WRITE (8,260) NA,DN
C
C 400   CONTINUE
C
90    FORMAT ('Kolmogorov - Smirnov test of 2 distributions'/)
C 92    FORMAT ('sorting #1 ...')
C 94    FORMAT ('sorting #2 ...')
C 95    FORMAT ('writing sorted data sets to F?.DAT files ...')
C 96    FORMAT ('calculating Dm,n ...')
C 120   FORMAT ('value of Dm,n = ',F8.4,' %')
120   FORMAT (F8.4)
250   FORMAT (F15.8)
260   FORMAT (10X,I2,10X,F7.4,' %')
450   FORMAT (I4)
550   FORMAT(2(F4.2,2X),2(/,5(F8.2,2X)),/,2(F6.2,2X),/)
C550   FORMAT (2(F4.2,2X),6(/,6(E10.4,2X)),/,F8.2,/,F6.2,//)
332   FORMAT (2X,'m=',I4,3X,'n=',I4)
333   FORMAT (/,2X,'PARAMETER',5X,'Dm,n')
335   FORMAT (5X,I2,8X,F5.2,2X,F5.1)
133   FORMAT ('DMN1=',F5.1,3X,'DMN2=',F5.1)
552   FORMAT (5X,'MEAN',6X,F5.2)
230   FORMAT(4(/,3(E13.5,2X)))
C
      STOP
```

```
      END
C
C
      SUBROUTINE SORT(N,RA,NP)
C
C sorts parameter values for calculating the cdf
C
      DIMENSION RA(9999)
      L=N/2+1
      IR=N
10    CONTINUE
      IF (L.GT.1) THEN
      L=L-1
      RRA=RA(L)
      ELSE
      RRA=RA(IR)
      RA(IR)=RA(1)
      IR=IR-1
      IF (IR.EQ.1) THEN
      RA(1)=RRA
      RETURN
      END IF
      END IF
      I=L
      J=L+L
20    IF (J.LE.IR) THEN
      IF (J.LT.IR) THEN
      IF (RA(J).LT.RA(J+1)) J=J+1
      END IF
      IF (RRA.LT.RA(J)) THEN
      RA(I)=RA(J)
      I=J
      J=J+J
```

```
      ELSE
      J=IR+1
      END IF
      GO TO 20
      END IF
      RA(I)=RRA
      GO TO 10
      END
C
C
      FUNCTION PROBKS(ALAM)
C
      A2=-2.*ALAM**2
      FAC=2.
      PROBKS=0.
      TERMBF=0.
      DO 11 J=1,100
        TERM=FAC*EXP(A2*J**2)
        PROBKS=PROBKS+TERM
        IF (ABS(TERM).LT.0.001*TERMBF) RETURN
        FAC=-FAC
        TERMBF=ABS(TERM)
11    CONTINUE
      PROBKS=1.
      RETURN
      END
```