USER FRIENDLY APPLICATIONS OF THE FORCYTE ECOSYSTEM MODEL ON A MICROCOMPUTER

M.J. Apps[1], W.A. Kurz[2], J.P. Kimmins[2] and K.A. Scoullar[3]

ABSTRACT. The ecosystem model FORCYTE-11 has been transferred successfully from a mainframe environment to microcomputers equipped with a 32-bit coprocessor board. A user friendly software package, PROBE, was developed to facilitate the preparation, execution and analysis of multiple runs of simulation models for applications such as sensitivity analysis and management gaming. PROBE was used to conduct preliminary sensitivity analysis of FORCYTE-11.

INTRODUCTION

Until recently, large computer simulation models have been restricted t high speed, large memory, mainframe computers. New advances in microcomputer technology have changed this situation. The advent of the faster 32-bit microprocessor, the removal of the 640 kilobyte memory limitation, and the development of microcomputer compilers capable of processing very large programs in commonly used languages such as FORTRAN have made the economical desktop computer a suitable environment for large models. Utilization of the full computational power of this new generation of desktop computers may be enhanced by the development of user-oriented supervisory software to facilitate multiple runs and analysis of the large output data sets that are produced. Such software can also partly automate the otherwise time consuming and tedious job of conducting sensitivity analysis on large, complex models.

In this paper we describe the successful transfer of FORCYTE-11 (Kimmins et al., this volume) from the University of BC mainframe (Amdahl 580) to a PC, and the subsequent development of a user friendly software package (PROBE) to permit the preparation, execution and analysis of multiple runs of FORCYTE for activities such as management gaming and sensitivity analysis. Originally conceived to assist in probing the sensitivity of FORCYTE to various input parameters, PROBE has much wider applicability and can be used to enhance the model's use in basic research, teaching, and as a predictive forest management tool. A simple example will be used to illustrate PROBE's application in sensitivity analysis of the model.

---

[1] Research Scientist, Northern Forestry Centre, Canadian Forestry Service, Edmonton, Alberta, Canada.
[2] Graduate Student and Professor, Dept. of Forest Sciences, Faculty of Forestry, University of British Columbia, Vancouver, B.C., Canada.
[3] Research Consultant, Life Sciences Programming, Vancouver, B.C.

This work was supported (in part) by the Federal Panel on Energy R&D (PERD) through the ENFOR (ENergy from the FORest) program of the Canadian Forestry Service.

Presented at the IUFRO Forest Growth Modelling and Prediction Conference, Minneapolis, MN, August 24-28, 1987.

FORCYTE

The philosophy and approach of FORCYTE-11 are presented elsewhere in these proceedings, and only a few relevant points will be repeated here. FORCYTE-11 presently consists of three separate FORTRAN programs (FORSOILS, TREEGROW and MANAFOR), each containing several thousand lines of source code (additional modules are in preparation for the simulation of shrubs, herbs and mosses).

The FORCYTE modules FORSOILS and TREEGROW define and describe various aspects of a forest ecosystem and make this information available to the MANAFOR module, a program which provides a simulation of user-specified management strategies for this ecosystem. These three FORCYTE modules are typically executed in sequence, and each requires a user-prepared input data file. MANAFOR also requires binary input files (TREND-files), which are generated by the other two modules. Once the appropriate TREND files have been established, any number of runs of the MANAFOR module can be executed without repeating the runs of the preceeding modules.

Each FORCYTE module can produce large quantities of diagnostic output which can be reviewed by the user. Prior to the development of PROBE, each run of each module was treated as a separate entity requiring editing of the input data files and execution of the program module by the user. Direct comparison of the results of a series of runs was performed by manually extracting the results of interest from the various output files.

THE HARDWARE

The personal computers used in this project (IBM PC/XT/AT and compatibles) are equipped with a 32-bit coprocessor board (DEFINICON DSI-32) with its own on-board RAM memory. The execution of FORCYTE-11 requires at least 2 Mb of RAM. In the FORTRAN source code only the filename declarations had to be modified to permit the successful compilation of the programs with a FORTRAN 77 compiler available for the DSI coprocessor board.

The coprocessor board, with a 12.5 MHz clockspeed (faster ones are now available), executes an 80-year simulation with MANAFOR in approximately 2 to 4 minutes. The time varies with the input/output facilities of the PC (i.e. availability of a RAM disk and the speed of the harddisk).

In operation, the coprocessor board provides an entirely separate and independent background computing environment within the PC. Several PC software packages are available to provide a foreground/background operation in which the DSI coprocessor can be executing one task in the background while the PC is being actively used in the foreground for some other task. The PROBE user can exploit this feature by conducting a series of FORCYTE runs on the coprocessor board, while concurrently planning the next series or analyzing a previous one in the PC foreground.

PROBE

PROBE consists of several programs and files which assist the user in preparing, executing and analyzing multiple runs of a simulation model. Primary objectives in the development of PROBE were to permit the unattended execution of large numbers of runs and to facilitate comparison of the results of such runs.

Although developed for FORCYTE, PROBE in its generic form can be used with any simulation model which can be controlled from the PC or MS-DOS environment. In addition, although currently using the DSI coprocessor board, it can be used to control programs executing in the DOS environment (e.g. the DOS version of FORCYTE-10). This may be of increasing importance as new operating systems and faster CPUs become available for personal computer systems.

PREPARATION OF MULTIPLE RUNS

PROBE gives the user full control over the sequence in which the different FORCYTE modules will be executed and over the input data that are used for these runs. To achieve this flexibility, the user constructs a simple sequence file which PROBE uses to determine the order of execution and the data files to be used. Each element of the sequence consists of the FORCYTE module name and the data case number. For example, a user might run FORSOILS and TREEGROW to initialize the simulation of a particular ecosystem, and then apply different management regimes, such as a series of different levels of thinning, to this ecosystem. The sequence "S1 T1 M1 M2 M3 M4" signals the execution of FORSOILS and TREEGROW with a particular set of data in the SOILDATA (case 1) and TREEDATA (case 1) input files, followed by four successive runs of MANAFOR for the different thinning levels defined in MANADATA (cases 1 through 4). All four MANAFOR runs use the same TREND curves generated by S1 and T1.

Each FORCYTE module case run requires a separate input data file describing that case. Because successive cases often differ only in a few data, PROBE uses a data overlay technique so that only one default data set is maintained for each module. Case changes are stored in separate files which the user prepares with the assistance of an interactive PROBE utility program. This program also reduces the risk of operator-introduced structural changes to the input data file.

Sequences of virtually any length and complexity can be constructed by the user. Figure (1) illustrates a sequence in which the four MANAFOR treatments are run following each of two separate cases of TREEDATA. Such a sequence is the first step in performing a sensitivity analysis for the model, as discussed below.

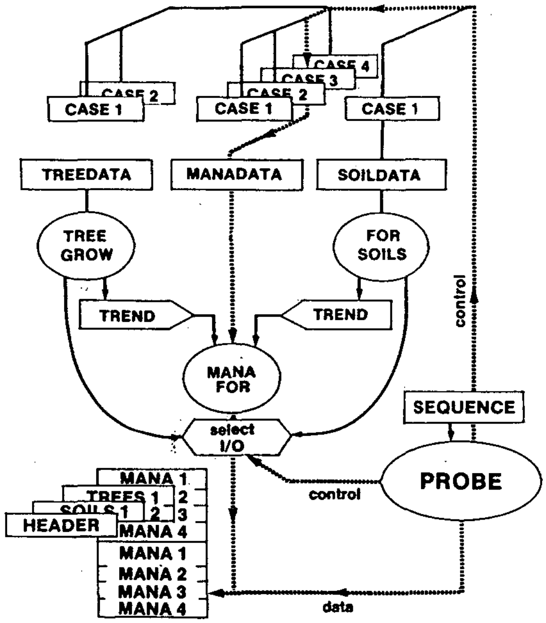


Figure 1. PROBE supervises the execution of multiple runs of the FORCYTE modules FORSOILS, TREEGROW, and MANAFOR. In this example, case 3 of MANADATA (dotted line) is prepared by PROBE. MANAFOR is then executed, and the selected results are added to the output file which contains the previous cases' results.

PROBE EXECUTION AND OUTPUT

A complex batch program controls the case overlay of data files, orders the FORCYTE module execution according to the user-specified sequence, and performs certain error-checking functions to ensure the proper execution of the modules which were called. It also redirects the selected FORCYTE module output to an output file which provides a complete record of the PROBE run. As schematically indicated in Figure 1, the FORCYTE data output for each case is saved and ordered in blocks for each of the modules executed. Only a subset of the potentially available FORCYTE output information is retained by PROBE. However, each case record includes sufficient data to exactly repeat the simulation if more detail is needed at a later time.

To provide for long unattended (overnight or background) execution, PROBE uses the special sequence "Hn" to allow the user to specify a new output file and to preface it with a descriptive header block. Using a new header for each, a series of computer experiments can be preconstructed and set into operation for overnight or background execution. A PROBE utility automatically archives the results of the experiments in successive files PROBEOUT.n (n = 0, 1, 2,...). There is no theoretical limit to the number of runs that can be executed in one session, although the amount of storage space available to archive the result files will act as practical limit.

ANALYSIS OF MULTIPLE RUNS

The output file contains two types of variables: dynamic and static. The status of dynamic variables (e.g. annual stemwood biomass) is written to the output file at user-specified time intervals. The status of static variables is reported for a particular time (eg. stemwood biomass at harvest) or for a time average (eg. mean annual increment over a rotation).

PROBE's output file is structured to facilitate its transfer into a spreadsheet. In addition to the functions normally performed by common spreadsheet programs (such as SYMPHONY TM), several menu-driven routines have been developed for use with SYMPHONY. These assist in the analysis and interpretation of both static and dynamic variables. Options presently available include the automated importation of PROBE result files, the compilation of summary tables and the graphical display of both dynamic and static variables.

In addition, a special routine has been developed to extract 2 way variation tables in which a user-chosen output variable is tabulated for the different management treatments (e.g. MANADATA cases representing various management regimes) and for different input parameter values (eg. TREEDATA cases using different values of ecosystem parameters). These tables provide the starting point for a specialized form of sensitivity analysis, a simple example of which is given below.

THE USE OF PROBE WITH FORCYTE-11: SENSITIVITY ANALYSIS

The objectives of sensitivity analysis must be clearly distinguished from those of model validation. While validation attempts to verify that the simulation results are close to or identical with the "real" observed values, sensitivity analysis investigates the model's response to changes in the input parameters. A review of the various approaches to and applications of sensitivity analysis is beyond the scope of this paper.

Sensitivity analysis of a model is typically performed to investigate the question "How do selected output parameters respond to changes in selected input parameters?" PROBE permits us, however, to take the next step and ask "how are the comparative predictions for a series of management strategies affected by these input parameter variations?" This question requires the assessment of the model's response to changes of parameters along two axes: the varied input parameter axis and the management treatment axis.

To illustrate the use of PROBE for such a second order sensitivity analysis we considered a relatively simple example. We chose five levels of a single input parameter which describes the light level below the canopy at the maximum overstory foliage biomass level of the stand. The five levels are the default value (15% of the light level above the canopy) multiplied by 0.5, 0.9, 1.0, 1.1, and 2.0 for cases 1 through 5 of TREEDATA, respectively. For each of the 5 runs of TREEGROW, four different management regimes were simulated. These describe four levels of thinning intensity in which 0%, 20% 40% and 80% of the stemwood biomass are removed at age 20 for cases 1 through 4, respectively.

Five runs of TREEGROW and twenty runs of MANAFOR were executed with PROBE to obtain the results for this example. The results were then analyzed using the spreadsheet routines developed for use with PROBE. As a demonstration only, Figure 2 shows the results for stemwood mean annual increment (per ha) at the end of each of the twenty runs.

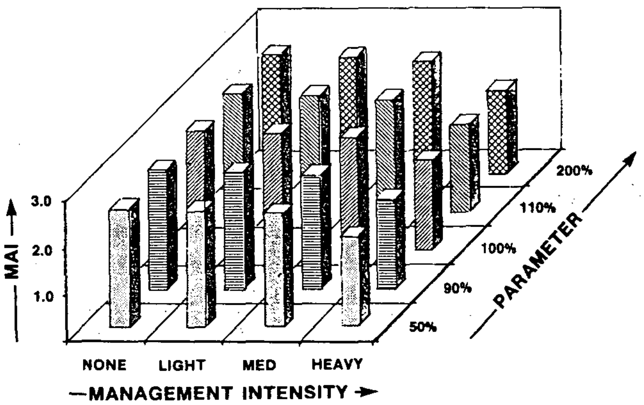


Figure 2. Stemwood mean annual increment (MAI) at the end of an 80-year rotation in a hypothetical aspen ecosystem. Each of the twenty bars represents the result obtained for a particular combination of thinning intensity and input parameter value.

As anticipated, the mean annual increment of the stand declined with increasing thinning intensity. Variations of the input parameter had little effect on MAI with no thinning and light thinning (maximum deviation from the default value was 0.3%, Figure 3). In the heavy thinning regime, differences in the output variable due to variations of the input variable were more pronounced and reached a maximum deviation from the default value of -4.3%. A full explanation of the different simulated processes which were affected by the changes to the input parameter is beyond the scope of this paper.

The above example illustrates the importance of conducting sensitivity analyses of a complex model by varying both the input parameters and the processes simulated with those input data. Although there was no effect of the variation of the input parameter at one management regime, at another regime the differences could be important. Such an application-oriented sensitivity analysis is particularly important with models, such as FORCYTE, which attempt to make management predictions.

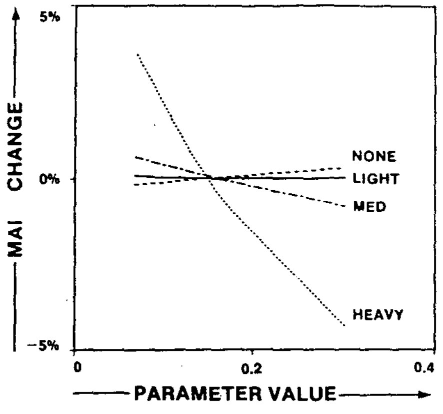


Figure 3. The data of Figure 2 expressed as percent deviation from the result obtained for the default input parameter.

Using PROBE, such an analysis can be prepared, excecuted and analysed efficiently. Compilation of the resulting graphs and tables will give future users of FORCYTE an indication of its sensitivity to changes in input parameters.

Future research plans include a systematic investigation of the model's response to variations in different input parameters using the approach outlined in this paper. The model's input data can then be classified into broad classes of sensitivity which can be used in deciding on the required levels of accuracy of the input data.

CONCLUSIONS

The transfer of FORCYTE-11 to a microcomputer equipped with a special coprocessor board has made the model available to users without access to mainframe computing facilities. The personal computer environment provided opportunities for the development of user friendly software which can enhance the utility of FORCYTE. The underlying concept and the software of PROBE, which were originally developed for FORCYTE, can be applied to other simulation models on personal computers. Sensitivity analysis, management gaming and other applications which require repeated runs of the same model can be performed with PROBE.

The example of the application of PROBE to preliminary sensitivity analysis of FORCYTE-11 emphasizes the need to conduct second order sensitivity analysis, in which the model's input parameters are varied, and a range of management scenarios are simulated with each of the varied input parameters.

Apps

United States Department of Agriculture

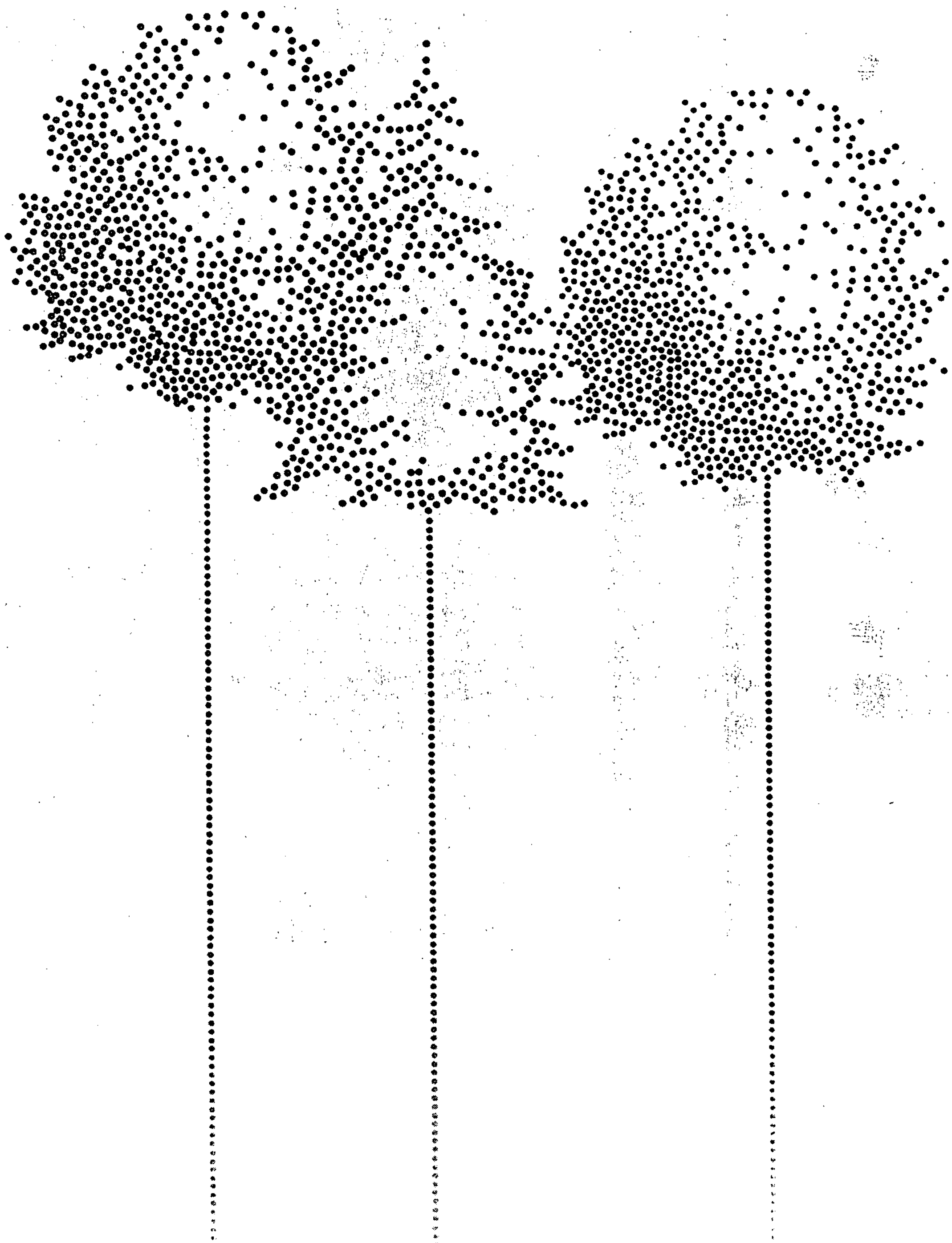
Forest Service

North Central Forest Experiment Station

General Technical Report **NC-120**

Forest Growth Modelling and Prediction

Volume 1



IUFRO

SOCIETY OF AMERICAN FORESTERS 1900

UNIVERSITY OF MINNESOTA

Forest Growth Modelling and Prediction
Volume 1

Proceedings of the IUFRO Conference
August 23-27, 1987, Minneapolis, Minnesota

Editors

Alan R. Ek
Stephen R. Shifley
Thomas E. Burk

Sponsors

International Union of Forestry Research Organizations (IUFRO)
Subject Group S4.01, Mensuration, Growth and Yield

College of Forestry, University of Minnesota

North Central Forest Experiment Station
United States Department of Agriculture, Forest Service

Cosponsors

IUFRO Subject Group S6.02
Statistical Methods, Mathematics
and Computers

Society of American Foresters
Biometrics Working Group
Inventory Working Group
Silviculture Working Group
Forest Ecology Working Group

Midwest Mensurationists

Western Biometricians

West Gulf Coast States
Biometricians

Association of Southern
University Biometricians

Society of American Foresters Publication Number SAF-87.12