Stand Management Assistant (SMA): A Tool for Forest Stand Management Analysis

Lauri Valsta, Tapio Linkosalo and Olli Salminen

Finnish Forest Research Institute
Unioninkatu 40 A, 00170 Helsinki, FINLAND

E-mail: Lauri.Valsta@Metla.Fi

1 Introduction

Silvicultural decision-making can benefit from quantitative models of forest stand development. To aid management, these models are often combined with optimization algorithms that help considering simultaneously several silvicultural operations with variable length effects on stand properties. Stand management decisions are often made at either stand level or forest level. Stand level analysis permits detailed comparisons of silvicultural alternatives and the examination of the effects of different factors, such as prices, costs and available treatments. For these questions, results of forest level analysis are often too case dependent to provide sufficient resolution.

Stand Management Assistant is a program for analyzing silvicultural and economic options of stand management. Based on deterministic or stochastic optimization, the programs allows the user to determine optimum regimes for the specified conditions and view the results of optimization as well as the sensitivity to changes in the prescription (response surface analysis). The program operates on unix operation system workstations. The current version uses a Sun SPARCSTATION and DEC Alpha workstations.

2 Program Structure

2.1 Principal Modules

The program consists of three basic functions: simulation, optimization and visualization. The mouse driven user interface in divided into 8 parts (Fig. 1). Pushing the numbered buttons produces the main window for each function. Subwindows are used in many parts for additional control. The basic operations for user input include typing numbers or text into fields, selecting and deselecting buttons, and adjusting numerical values using slide bars.

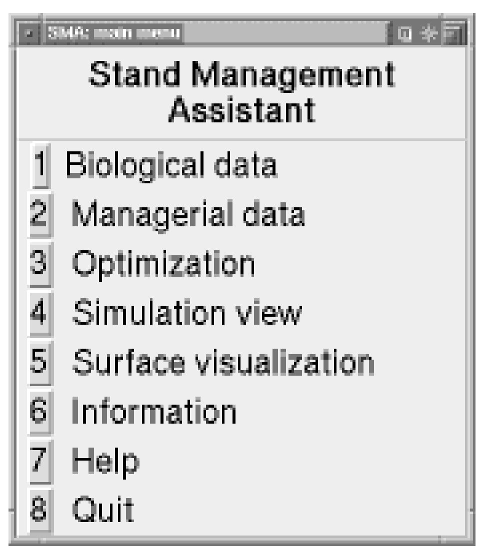

Figure 1. The basic structure of SMA.

The relation of the program elements are shown in Fig. 2. The stand simulator is called by several routines to provide responses to silvicultural regimes. The information and help modules are not shown.

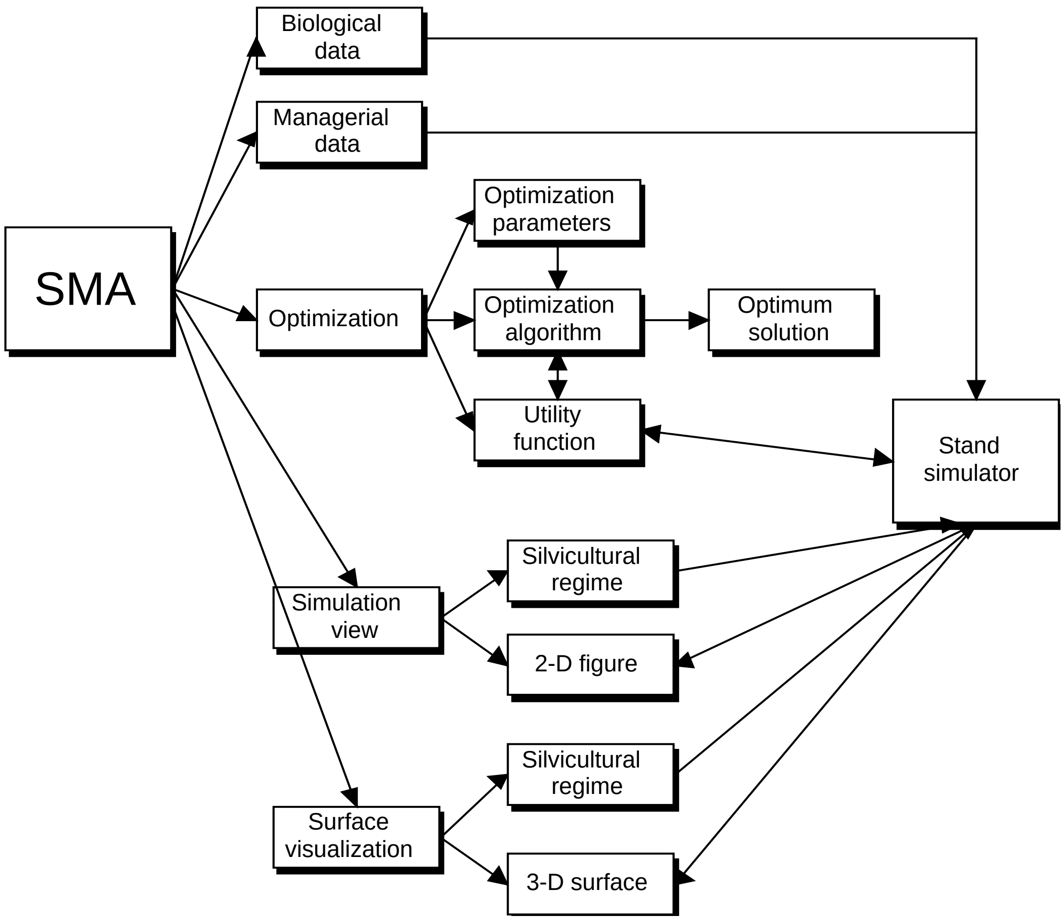

Figure 2. The relations of principal program elements.

The input data for the system are specified in two windows: biological data and economic data.

2.2 Biological Data

The window lets the user identify input files and parameters related to stochasticity in the model (Fig. 3). The scenarios used in stochastic optimization contain all probabilistic phenomena. Therefore, the stochasticity of round wood prices is conveniently entered here. The version described in this paper is customized for the stand simulator by Kellomaki et al. 1991.

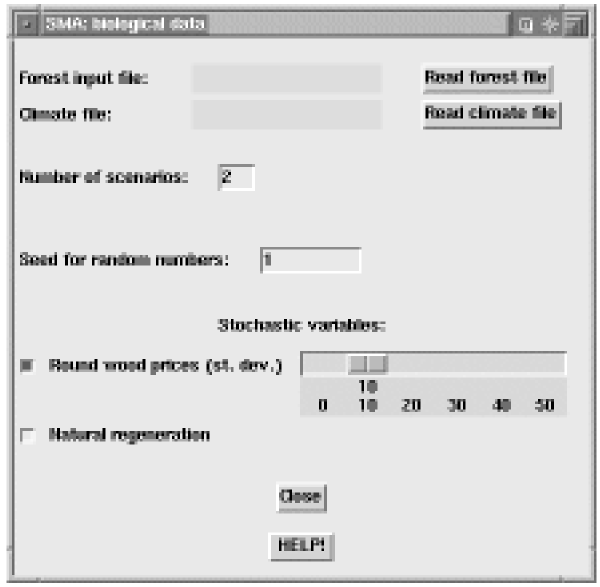

Figure 3. Window for biological data.

Elements

Forest input file

This lets the user identify an input file for the initial stand structure. This information is often needed by a stand simulator. The button [Read forest file] opens a list of files to choose from. The file names listed have the tag ".met".

Climate file

The climate file sets the climate parameters. This is also specific to the simulator by Kellomaki et al. 1991. The button [Read climate file] opens a list of files to choose from. The file names listed have the tag ".ymp".

Number of scenarios

The number is scenarios is required for the stochastic stand developments and (if specified) the stochastic prices. Serious analyses require up to 100 scenarios and then computations will be severely slowed down.

Seed for random numbers

The random number seed is used by the (pseudo) random number generator. By rerunning analyses with other seeds, the user can see how dependent the results are on the particular set of realizations of stochastic processes used in a run.

Round wood prices

Round wood prices can be made stochastic by clicking the square on the left. The slider is used to input percentage standard deviation of yearly prices. The underlying price model is of AR(1) type.

Natural regeneration

Simulating natural regeneration can be turned on/off be clicking the square on the left. Turning natural regeneration off improves the optimization process.

2.3 Managerial Data

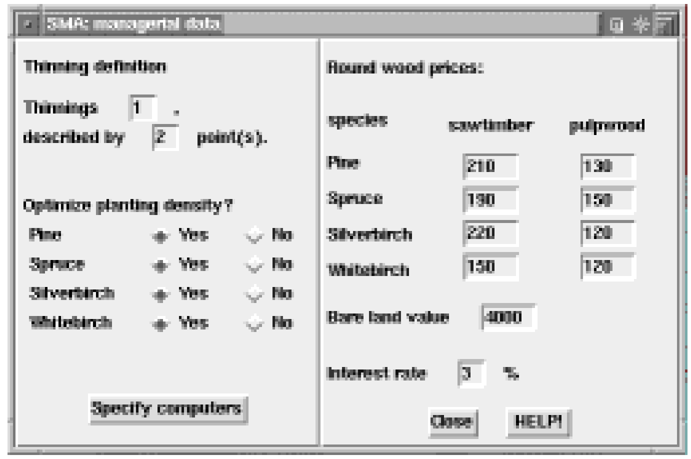

Figure 4. The window for managerial data.

This is the module for entering managerial data for the simulator (Fig. 4). The workstations to be used in computing are also specified here.

Elements

Thinning definition

The optimization requires that the number of thinnings and the number of thinning points be specified. Each number of thinnings forms a different optimization problem with different variables to be optimized. Reasonable values are 1-3. If a smaller number of thinnings provides a higher objective function value, the optimization may drive thinning percentages to zero for unwanted thinnings.

The thinning points enable varying thinning types (thinning from below/above) in simulation. The thinning rate in relation to tree diameter by species is defined by a piecewise linear function. The flexibility of the line depends on the number of parameters, the thinning points. In view of optimization, a reasonable number of points is 1-5.

Just one thinning point produces constant thinning rate over diameter classes. Figure 5 shows optimum solutions for the same problem with varying thinning definition (Valsta 1992). Note how the depiction based on just a few thinning points approximates the exact solution (12 points which match the 12 diameter classes in the stand at time of thinning).

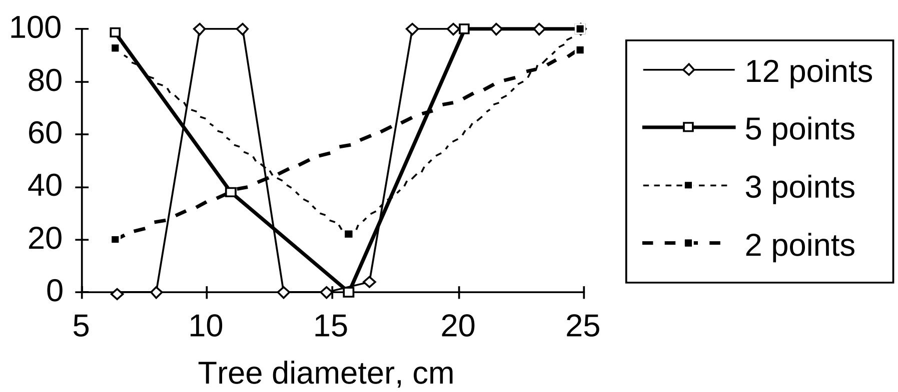

Figure 5. Thinning rates in relation to diameter in optimum solutions to four different thinning specifications.

Planting density

The underlying stand simulator features four species. The user can specify for each species independently whether planting density is optimized. Planted trees are added to whatever seedlings may occur as a result of natural regeneration or an initial tree list. Establishment costs are adjusted according to planting density.

Specify computers

The workstation(s) to be used for computations (simulations) are identified here (Fig. 6). There is only one workstation for optimization but the computations for surface generation can be parallelized onto up to four workstations. The buttons for the "Optimizer" and the "Surface generator" behave differently when pushed.

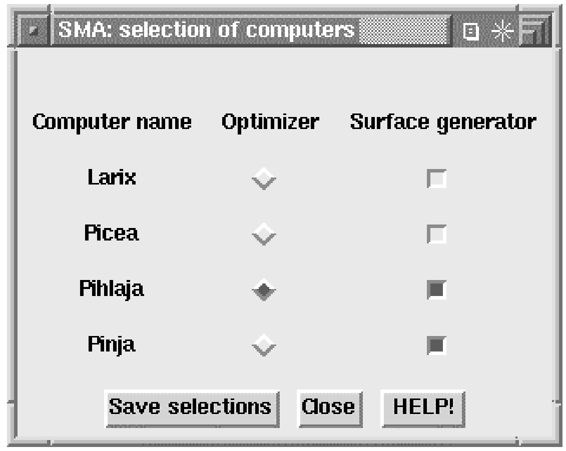

Figure 6. The window for selecting the computers.

Round wood prices

The wood prices refer to prices for wood delivered to road side in the forest, per unit volume (FIM/cu.m.). The default values are recorded in the file sima.laj (at the end), located in ../sima directory.

Bare land value

The bare land value is used when computing discounted net revenues of stand management. It is used for approximating the value of future rotations, required for computing the soil expectation value. The bare land value depends on the interest rate. Reasonable values for Finnish conditions are given below:

Interest rate, %	Bare land value, FIM/ha	
0	1 000	(Corresponds to average annual cash flow)
1	70 000	
2	25 000	
3	10 000	
4	3 000	

Interest rate

Discounting is based on the interest rate, here defined as a real, tax tree rate of interest. The present value criterion implies, strictly speaking, that unlimited amounts of money are available and can be invested at the rate given as interest rate. Interest rates greater than 4 % may lead to negative soil expectation values and disrupt optimization.

2.4 Optimization Data

This module handles optimization parameters, the initial solution, the utility function, and control of the optimization process. The optimization algorithm is a modification of the Hooke and Jeeves' direct search method, adopted from Osyczka (1984). This algorithm can be classified as a derivative-free, unconstrained, nonlinear programming algorithm. It is suitable for being used in connection with individual-tree growth models as they produce a set of models for which it is complicated or impossible to derive derivatives with respect to the state or control variables of the model. An example of the steps of the algorithm is seen in Fig. 8.

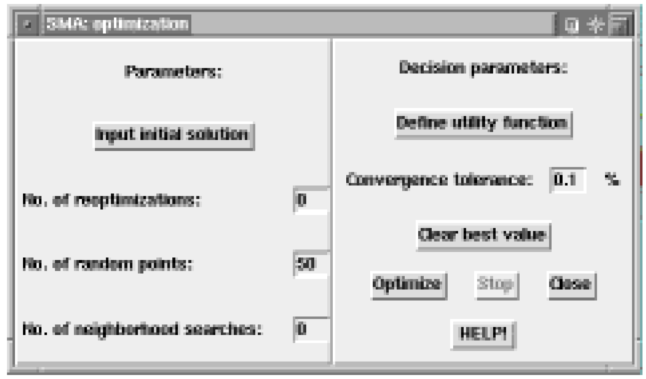

Figure 7. The window for optimization data.

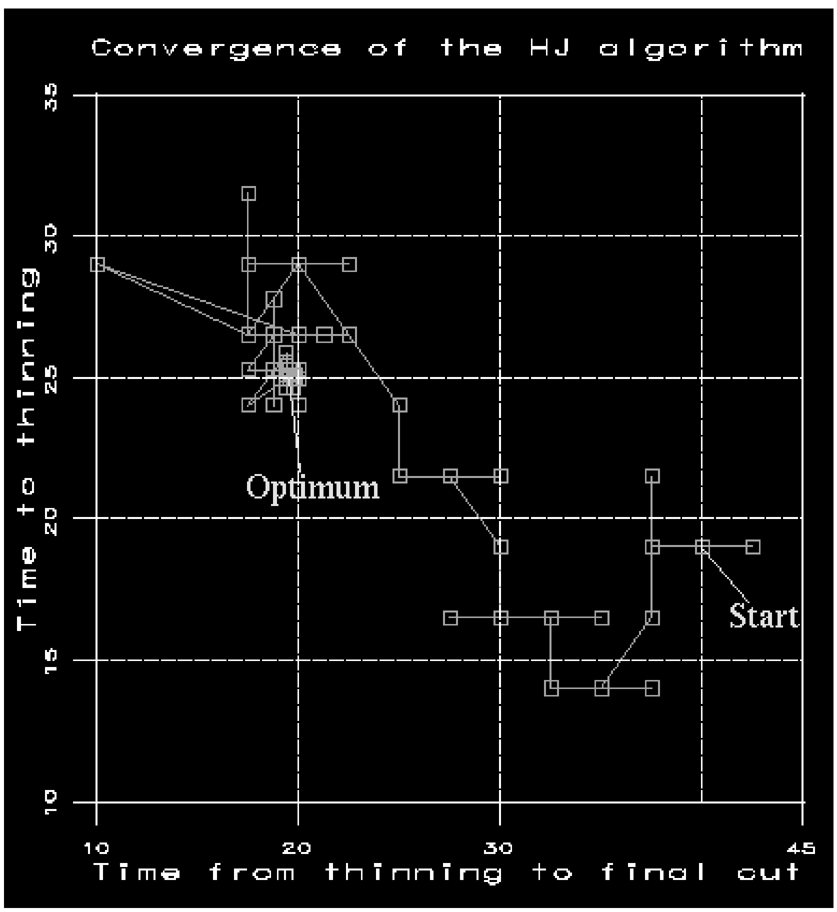

Figure 8. An example of the points generated by the Hooke and Jeeves' direct search algorithm.

Elements

Input initial solution

The optimization requires an initial solution from which the algorithm starts improving step by step. This button brings up a window (Fig. 9) that lets the user specify the initial solution, i.e., a silvicultural regime for the whole rotation, using slide bars. This can be used during optimization or visualization. The cuts and planting densities are given initial values. The structure of the window depends on the number of thinnings and thinning points.

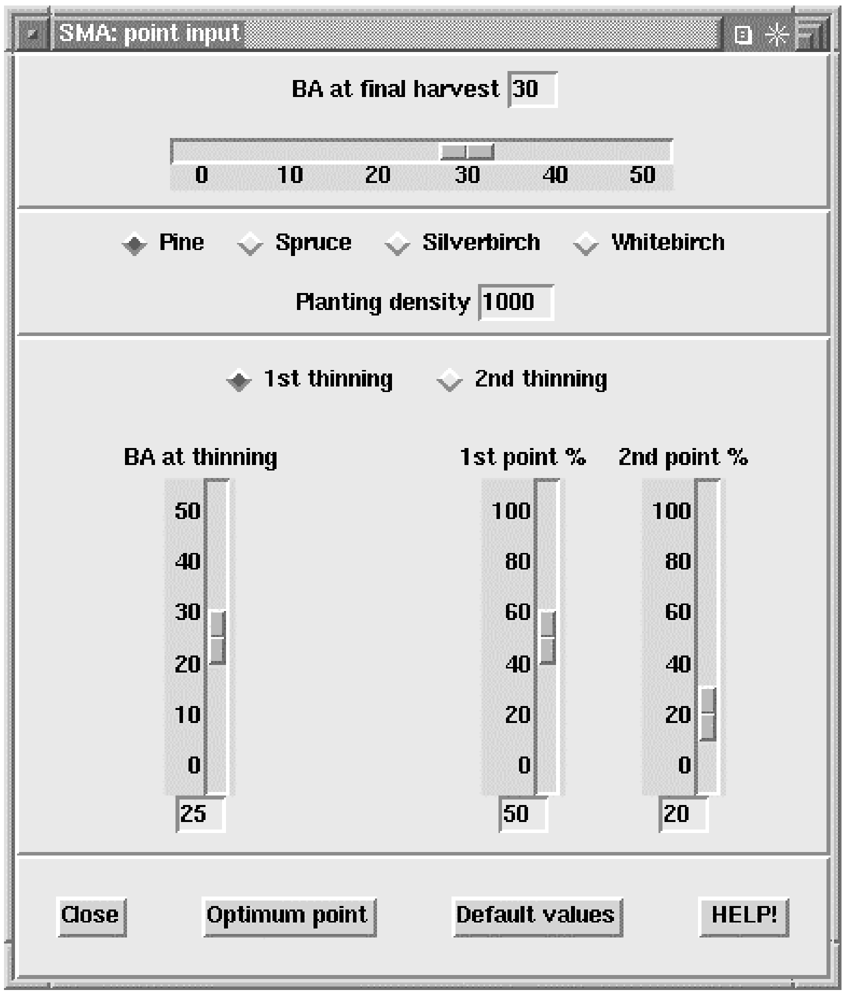

Figure 9. The window for specifying an initial solution for optimization or a base point for visualization.

Number of reoptimizations

Because of the non-convexity of the optimization problem, locally optimal solutions may be reported in optimization runs. To improve solutions, optimization may be restarted several times from randomly generated initial solutions. The number of additional runs is given here.

Number of random points

Randomly generated points are used in two phases of the runs: To compose an initial solution for reoptimization; to produce additional points in the vicinity of a candidate optimum solution (the so-called neighborhood search).

Number of neighborhood searches

A neighborhood search is initiated in the vicinity of a candidate solution identified by the Hooke and Jeeves algorithm. The neighborhood search produces randomly a number of points around the candidate solution with a spread of 1/10 of the feasible range of variables. If the simulator creates a lot of local optima, it is possible that neighborhood searches would be restarted several times. This parameter sets a maximum number of neighborhood searches available.

Define utility function

This module (Fig. 10) is used for defining the utility function used in stochastic optimization. The basic measure of utility is the expected value of the objective function. If the decision maker is risk averse and wishes to consider also the variation of the objective function value, this model enables inclusion of risk aversity.

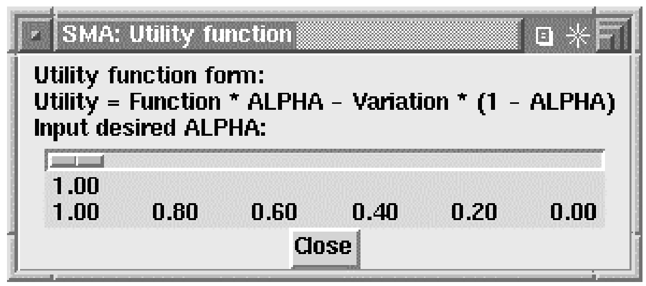

Figure 10. Specifying the degree of risk aversion in the utility function.

The model uses one parameter, ALPHA, to change the degree of risk aversity. When ALPHA = 1, the expected value is given full weight in the utility and the decision maker is risk neutral. With decreasing values of ALPHA, the expected value is given less importance and risk aversion increases. When ALPHA = 0, the decision maker minimizes variation and does not care at all about the expected objective function value, showing extreme risk aversion.

Convergence tolerance

The termination criterion for optimization is given by the convergence tolerance. If the change in variables between successive iterations of the Hooke and Jeeves algorithm is less than the tolerance, the algorithm reports a solution. The tolerance is relative to the feasible ranges of variables.

Clear best value

This button clears the best objective function value that is used in the window showing the changes is in variables and function value during an optimization run.

Optimize

This button starts the optimization. During the run, a dynamic window (Fig. 11) pops up showing the changes in the objective function value and the variables optimized. When optimization ends, a new window emerges showing the optimum solution (Fig. 12).

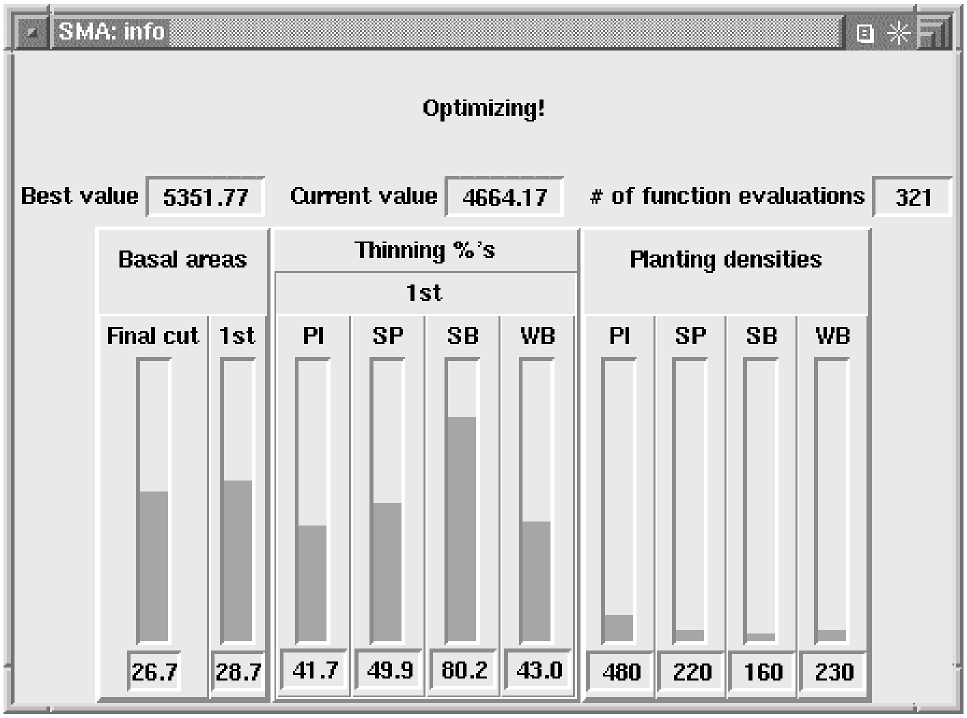

Figure 11. The dynamic window showing the course of the optimization run.

Best value is the highest objective function value reported since starting the program or clearing the best value in the optimization window. Current value corresponds to the values of the variables in optimization.

The number of function evaluations shows how many times the stand simulator has been called during this run of the Hooke and Jeeves algorithm (including randomly generated points).
The basal areas (sq.m./ha) at cuts, thinning percentages and planting densities (trees/ha) define the silvicultural regime. The symbols are the names of the tree species supported by the simulator :

PI = (Scots) pine
SP = (Norway) spruce
SB = Silver birch
WB = White birch

The window for the optimum solution (Fig. 12) shows the result of optimization. This is the overall best solution found during possible reoptimizations defined by the respective parameter. Each push of the button [Optimize] produces this window eventually. Windows for consecutive runs will be numbered, in the example the window resulting from the second run is shown (Fig. 12). The objective function value is the soil expectation value (the current rotation discounted net present value plus bare land value discounted from the end of the current rotation). In a stochastic analysis, the objective function value is the expected soil expectation value, possibly adjusted by the risk aversion factor.

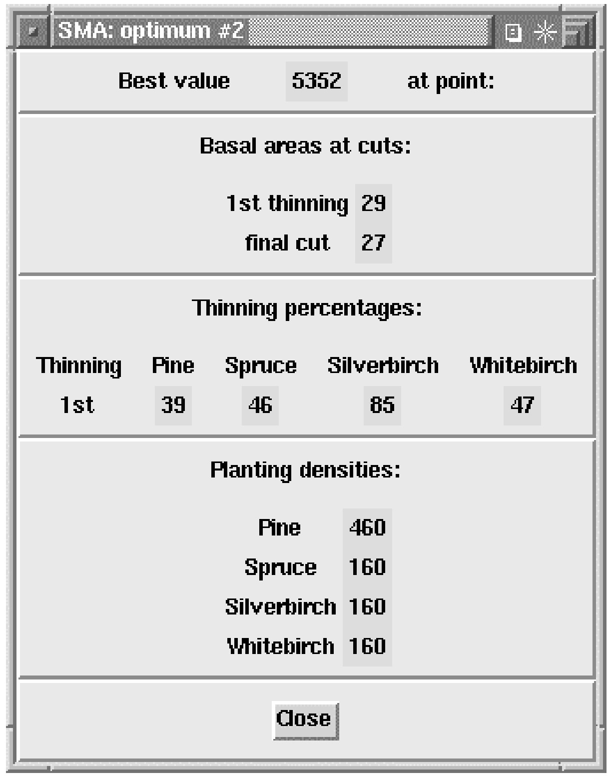

Figure 12. The window reporting the optimum solution found in optimization.

Stop

The stop button kills the optimization run.

2.5 Simulation View

This module produces graphs about the development of selected stand variables over the rotation. A detailed textual output is also produced on the shell window and also to a temporary file, from which it can be saved for later inspection.

If nothing else is defined, the regime simulated is the last result of optimization. The user can also simulate any desired regime; in this case the silvicultural operations are defined using the point input window (Fig. 9).

Two examples of the main window for simulation view are shown in Fig. 13. Most of buttons are disabled in the left hand size window. They become enabled after a simulation is calculated by clicking the Calculate button. Before that the silvicultural regime may be changed by clicking Plot point (see point input about that).

Two plot types are available: Average refers to values computed by averaging scenario-wise developments, All scenarios produces a line for each scenario (or the first 10 scenarios if there are more than that of them).

The list of picture types corresponds to the information available by the stand simulator used. A header for the picture can be typed in to the accompanying box.

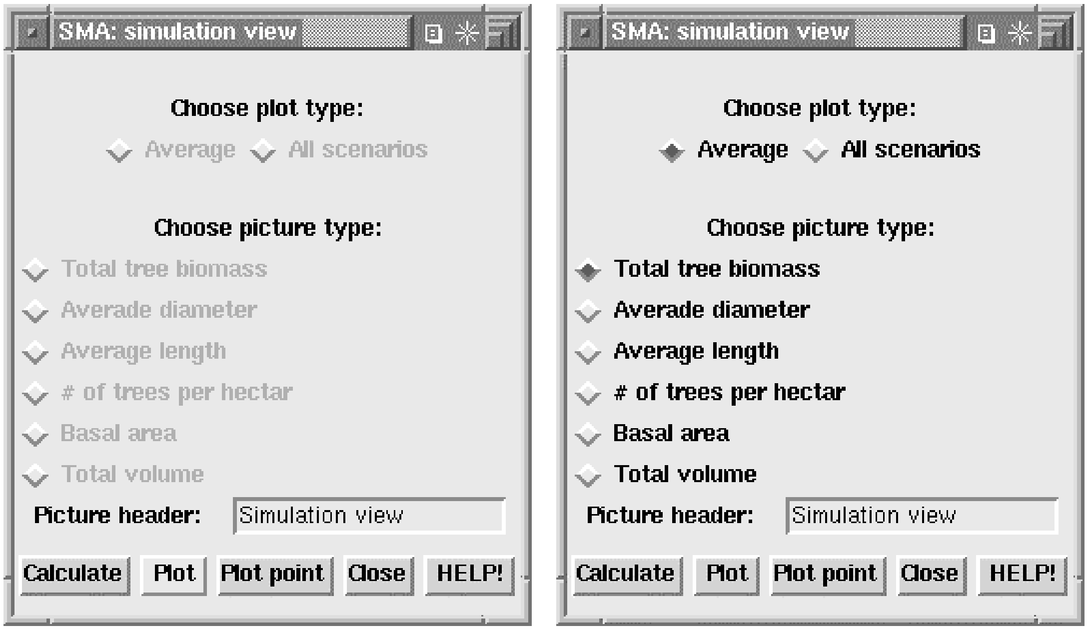

Figure 13. The main window for simulation view at two instances: before a silvicultural regime has been simulated (left) and after that (right).

The Plot button invokes the program xprism2 of the Khoros visualization system with input data produced by the stand simulator. Two sample graphs are shown below (Figs 14 and 15), one for the average values and another for all scenarios.

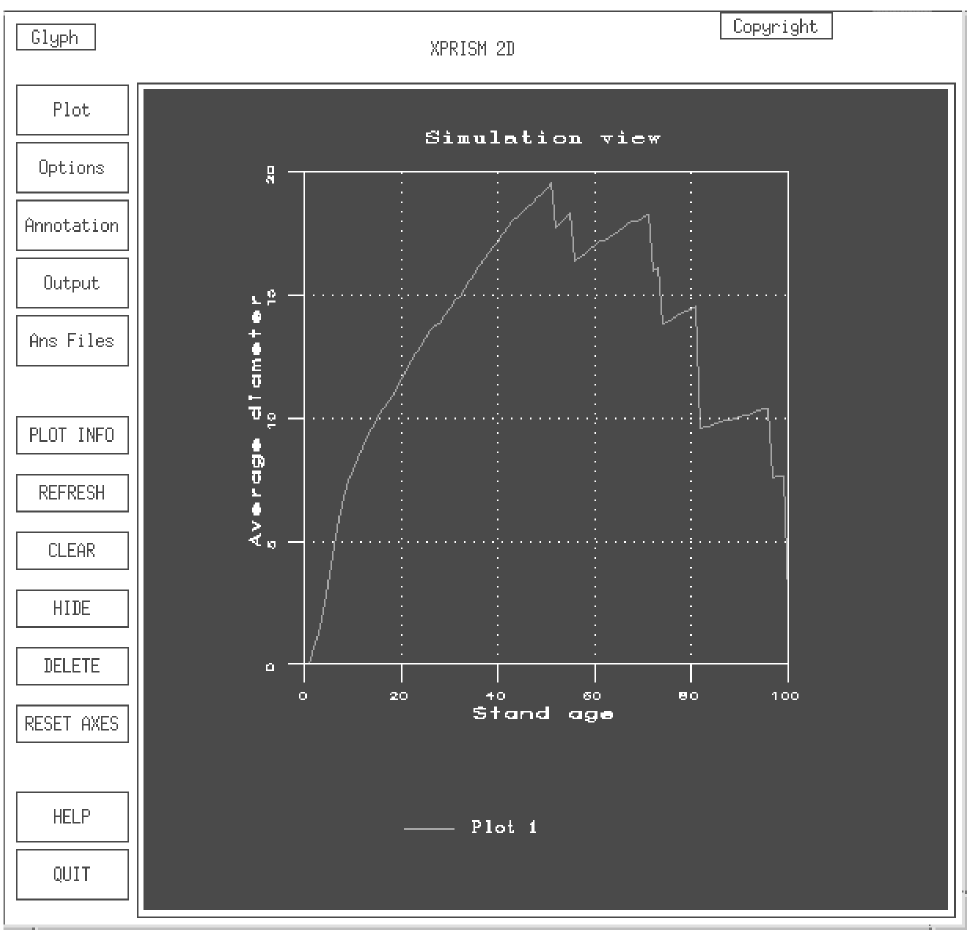

Figure 14. Simulated development of tree diameter, averaged over five scenarios.

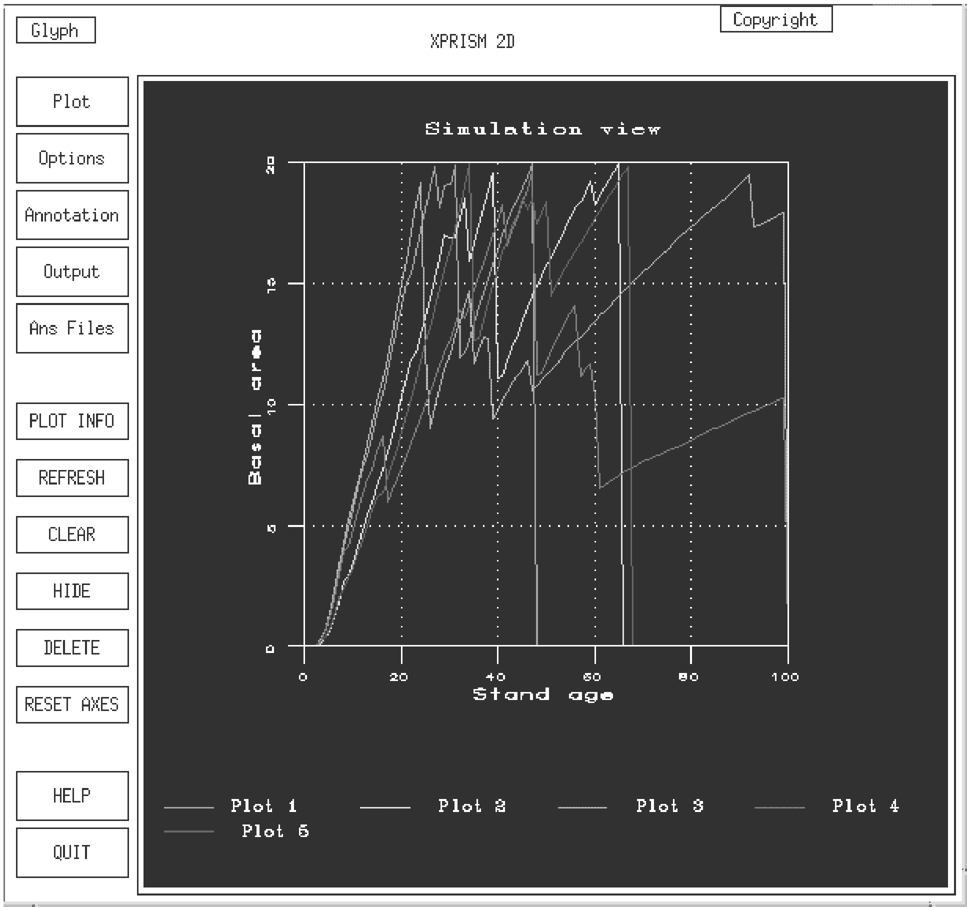

Figure 15. Simulated development of stand basal area in five scenarios.

Many of the properties of the graph can be adjusted via Khoros control functions. The graphs may be saved to files or plotted on PostScript color or monochrome printers. There is also an extensive help facility associated.

2.6 Surface Visualization

The response surface around the optimum or a user defined point can be viewed as 3-dimensional figures. This provides a visual sensitivity analysis.
The user specifies two variables, one for the x axis and the other for the y axis (Fig. 16). These variables are given minimum and maximum values and their values are changed accordingly when the data points for the 3-dimensional graph are produced. Other variables are kept at the values of the current base point. The slide bar plot intervals specifies the number of grid points for both the x and y axis.

The buttons that are disabled become enabled when planting density or thinning percentage are selected.

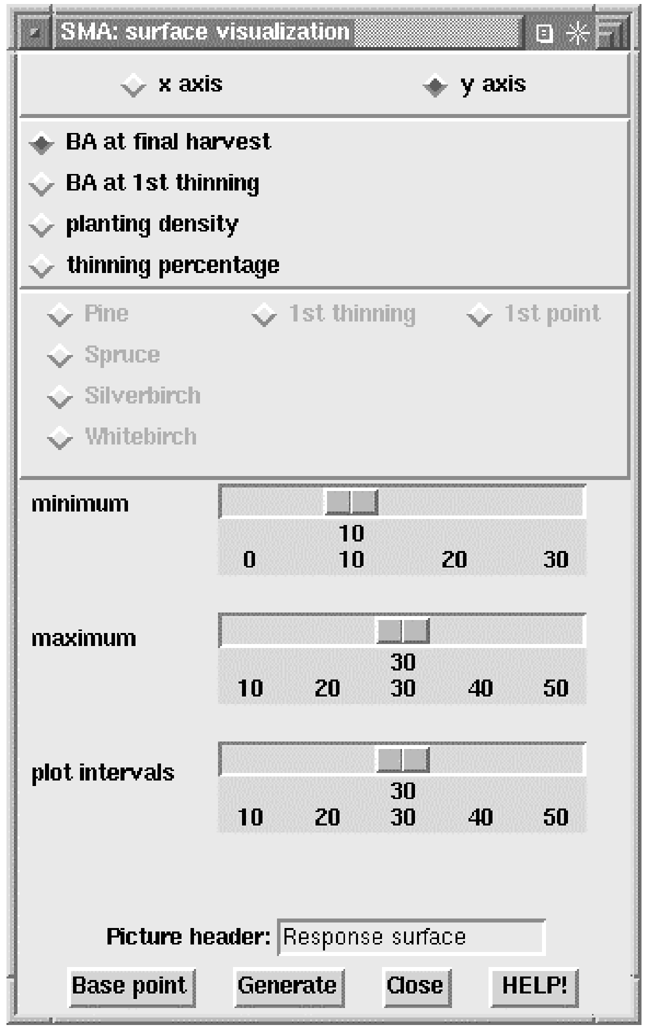

Figure 16. The window for specifying response surface graphs.

The button Base point invokes the window with which other variables of the silvicultural regime can be set. The Generate button invokes the program xprism3 of the Khoros visualization system with input data produced by the stand simulator. A sample graph is shown in Fig. 17.

Many of the properties of the graph can be adjusted via Khoros control functions. The graphs may be saved to files or plotted on PostScript color or monochrome printers. There is also an extensive help facility associated.

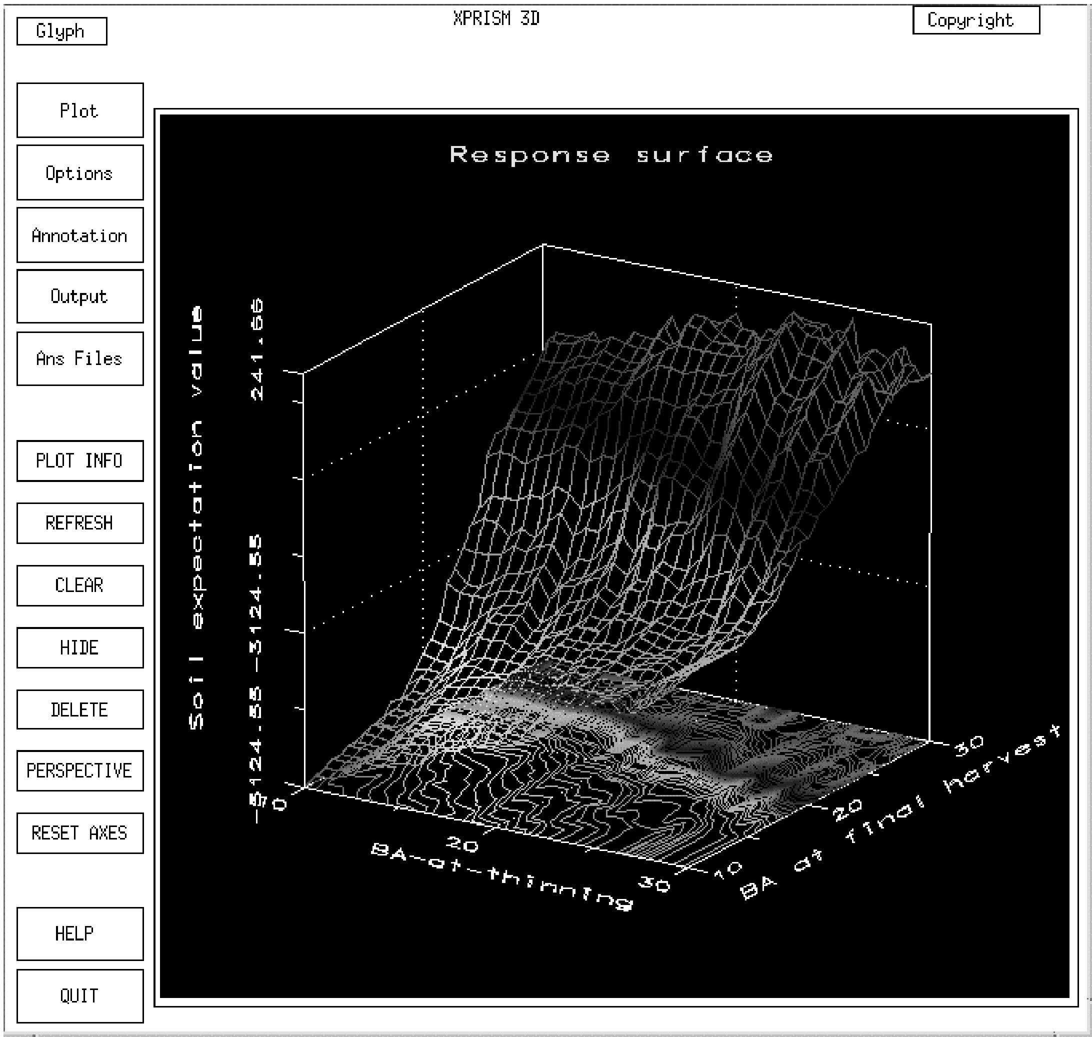

Figure 17. A response surface depicting the sensitivity of soil expectation value (the objective) to basal
area at thinning and final harvest.

2.7 Information

This button invokes a text widget that displays information about the version of SMA in use. The text is read from a text file.

2.8 Help

The help buttons bring up help pages for different program modules.

3. Software Tools

The SMA system consists of three types of elements:

- User interface programmed using Tcl/Tk
- Simulation and optimization routines programmed in FORTRAN
- Visualization parts using the Khoros system

Tcl (Tool command language) and Tk (a toolkit for X Window System) form a development environment for creating user interfaces. They were developed by Ousterhout (1994) at the University of California, Berkeley, USA, and have gained popularity both in academia/research and businesses with tens of thousands of estimated users (Strand 1994). Tcl is a scripting language for controlling and extending applications. A Tcl script is interpreted during program execution. Tcl structure allows dynamic user interfaces where, for example, the layouts of the windows can react to user input. The windows in SMA change during a session according to what treatments are defined (the number of thinnings, the number of thinning points).

Tcl is available for most unix platforms, VAX/VMS, Macintosh, DOS, MS-Windows, OS/2, AmigaDOS and NeXT.

The user interface calls FORTRAN language routines for optimization and stand simulation. C language routines could be used as well. The optimization routines are adopted from Osyczka (1984) and they are described in Valsta (1992a and 1992b). The optimization method is a modification of the Hooke and Jeeves' direct search algorithm which is classified as a derivative-free nonlinear programming algorithm

Stand simulation routines depend on the purpose of the tool. Two case studies are described below. Necessary parameters for optimization and simulation are passed as argument lists to called routines.

The user interface also calls the visualization routines for two and three dimensional graphs produced by the Khoros system, developed at the University of New Mexico, USA (Rasure and Williams 1991). The Khoros consortium has established a company, Khoral Research, Inc. to continue system development. The next version of the system will be made publicly available in August 1994.

Graphing information is obtained from disk files and argument lists. Once a graph is formed by Khoros programs, it can be extensively customized with the options provided in the programs. Printing options are also well developed. The Khoros system is available on most unix platforms for mainframe and micro computers.

4. Case Studies

The SMA system has been used in connection with two stand development models: the SIMA model by Kellomäki et al. (1991) and the MELA model by Ojansuu et al. (1991). Because the models require different input data, the user interface was slightly modified to match the models' needs.

The SIMA version was developed for a study on climate change and forest management alternatives. The growth and development of trees and ground cover are controlled by temperature and light conditions and the availability of nitrogen and water. Because of the large computational tasks involved, parallel processing is used in this version when generating data points for 3-D visualization.

The computations for surface generation can be parallelized onto up to four workstations. A status window (Fig. 18) shows dynamically how computations advance on each workstation. In this example there are three workstations in use.

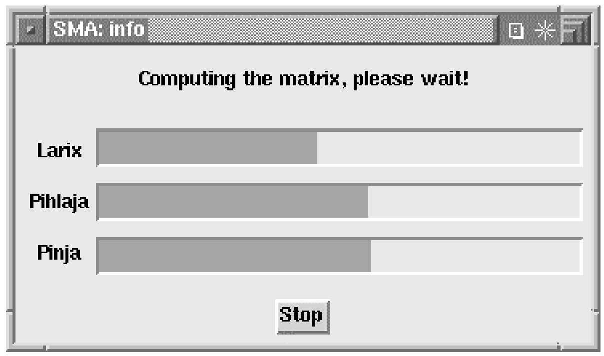

Figure 18. A status window on task completion in each workstation.

The MELA version aims at economic analysis of silvicultural options of stand management. Stand development is based on statistically estimated individual tree growth and mortality models. Both deterministic and stochastic optimization can be used to answer to questions concerning the optimum rotation, number of thinnings, timing of thinnings, thinning type, and planting density.

5. Literature

Hooke, R. & Jeeves, T.A. 1961. "Direct search" solution of numerical and statistical problems. J. Assoc. Comput. Mach. 8:212-229.

Kellomäki, S., H. Väisänen, H. Hänninen, T. Kolstrˆm, R. Lauhanen, U. Mattila and B. Pajari. 1992. A simulation model for the succession of the boreal forest ecosystem. Silva Fennica 26:1-18.

Ojansuu, R., Hynynen, J., Koivunen, J. & Luoma, P. 1991. Luonnonprosessit metsälaskelmassa (MELA) - METSÄ 2000 -versio. Metsäntutkimuslaitoksen tiedonantoja 385. 59 p.

Osyczka, A. 1984. Multicriterion optimization in engineering with FORTRAN programs. Ellis Horwood, Chichester.

Ousterhout, J.K. 1994. Tcl and the Tk Toolkit. Addison-Wesley.

Rasure, J. & Williams. 1991. An Integrated Visual Language and Software Development Environment. Journal of Visual Languages and Computing, Vol 2, pp 217-246.

Strand, E. J. 1994. A shell for X Windows? GIS World, March 1994:24,26.

Valsta, L. T. 1992a. An optimization model for Norway Spruce management based on individual tree growth models. Tiivistelmä: Kuusikon käsittelyn optimointi puittaisiin kasvumalleihin pohjautuen. Acta Forestalia Fennica 232. 20 p.

Valsta, L. T. 1992b. A scenario approach to stochastic anticipatory optimization in stand management. Forest Science 38(2):430-447.