SAMT/DESIRE – A New Software for Landscape Analysis

H. Vakilzadian[1], G.A. Korn[2], R. Wieland[3], X. Holtma[4]

[1]Dept of Electrical Eng., University of Nebraska-Lincoln, email hvakilzadian@unl.edu
2 G.A. and T.M. Korn Industrial Consultants, email: gatmkorn@aol.com
[3]Leibniz-Centre for Agricultural Landscape and Land Use Research (ZALF), email: rwieland@zalf.de
[4] ZALF email: holtmann@zalf.de

Abstract

SAMT/DESIRE is integrated and interactive simulation software for dynamic and spatial computations, specifically for landscape modeling and analysis. It permits the integration of fuzzy models and neural networks with specific features for dynamic model development. It is intended to integrate different landscape models into a single system.

The software facilitates application of dynamic-system models from different fields such as economics and ecology to a large set of points on a landscape grid. It includes a grid-based geographical information system (GIS) and the fast interactive dynamic-system-simulation program, DESIRE, for data handling and landscape modeling. This paper describes the features of this combined tool along with an application example.

1.0 Introduction

The land-use problems are multidisciplinary. The interaction of many factors change landscape features and use of land. Analytical tools for simulation of land-use should account for ecology and economy related parameters to provide a framework for integration of different models. SAMT/DESIRE goes beyond the existing geographical information system software and allows concurrent interactive modeling of conditions at various points of a simulated landscape. Using this software package, one can construct relations among landscape data using expert knowledge (fuzzy-set and neural-network) models.

These techniques are, moreover, combined with improved data-analysis programs like 3D-visualization (NURBS [3], Splatter technique [9] etc.) to permit convenient interactive modeling.

DESIRE, which is a user-friendly environment for very fast interactive modeling and simulation of dynamic systems, formulates models in differential- or difference-equation forms. This software then vectorizes the model for many points of a landscape grid and executes the vectorized (replicated) models in a single simulation run.

DESIRE and SAMT are freely available under the Open Software Foundation's General Public License (GPL).

2.0 SAMT

The Spatial Analysis and Modeling Tool (SAMT) [4] is a simple geographical information system (GIS) that is used to declare and to store landscape properties at a grid of geographical points as a set of arrays of numbers. Each array contains data values that represent:

- Landscape properties (landscape features) at each grid point,
- Altitude, physical data like temperature, soil moisture, biomass of a species, etc.

SAMT is a grid based system that is able to read and write grids in ASCII in ARCGIS formats, performing all operations on grids. It has a connection to a *mysql* database. An HDF system is used as a spatial database. SAMT includes some simple grid functions (SGRID), like set a grid to a constant value, construct a random grid, multiply a grid with constant and so on. In addition, it has features for advanced grid functions (AGRID), like scaling, interpolation, distances, and focal functions.

The structure of SAMT is shown in Figure 1. The components are grouped around the interactive control unit, which controls all the operations of SAMT. The control unit is a graphical user interface, written using the graphical library qt (qt, 2004). The control unit manages the interaction between the user and all other units. The spatial data management organizes import and export to the GIS (ARCGIS) via

ASCII-grids. These grids include information about the region (size, location etc.) and the data matrix. Internally SAMT uses the HDF-Library (hdf, 2004) to store spatial data. This method is faster than using ASCII-grids and can store additional information like modeling method, creation time etc.

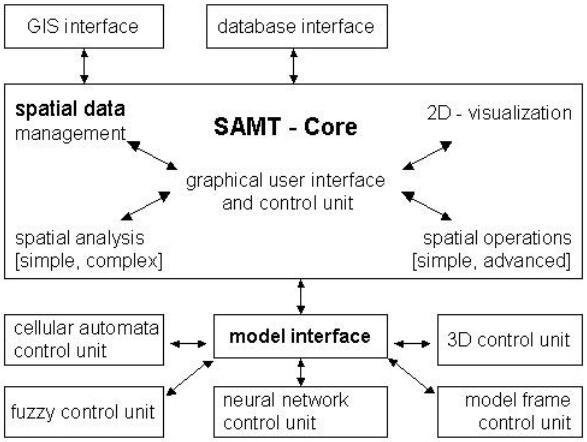

Figure 1. Main Components of SAMT

In addition to simple grid operation functions, the analysis functions, such as histogram and statistic, have also been embedded in SAMT. Additional analysis techniques like the sensitivity analyses, semi variogram or power spectra also are part of this package. Functions for edge detection; rotation of grid with a 90, 180, and 270 degrees; some point algorithms (edit point theme, transformation of a point theme to a grid) and special functions: flood fill algorithm and Hasse diagram technique are also integrated into this system.

SAMT has a prototype to communicate with external models using sockets (linked model).

SAMT can assign and calculate functions relating landscape features at different grid points, say

q1 = q2 + q3

q1 = cos (q2)

q1 = calc (q2q3)

Functions like calc (q2q3) can be numerical expressions, or they may be regression functions obtained, e.g. from neural-network or fuzzy-set models. SAMT can also assign grid-point data values that depend on data at other grid points, such as the distance of each grid point to another selected grid point, which might represent the location of city or a bird's nest; it also could find the shortest distance to a river or road. Furthermore SAMT can accumulate statistics (mean, standard derivation, histogram etc.) over a set of grid points. Last but not least, SAMT can draw maps that show grid point data values in different colors, or show contour lines and three-dimensional plots of different landscape features.

3.0 DESIRE

To study how landscape features change with time, SAMT needs landscape feature values only at fairly widely spaced sampling times (communication times) $t = t_0$, t_0 + COMINT, t_0 + *2COMINT; the communication interval COMINT might be a day, a month, a year, etc. DESIRE, however, can increment the time in smaller steps DT to emulate continuous changes. DESIRE relates current and future data values at each grid point by ordinary differential equations:

$$\frac{d}{dt}q_i = f(q_1, q_2, ..., p_1, p_2, ..., a_1, a_2, ...) \forall i \in (1..n) \quad (1)$$

For each grid point, the feature values q_1, q_2 are state variables starting from a given initial values. p_1, p_2 are feature values related to the state variables by defined-variable assignments

$$p_1 = g1(p_2, p_3, ...; q_1, q_2, ...; b1, b2, ...; t)$$
$$p_2 = g2(p_1, p_3, ...; q_1, q_2, ...; b_1, b_2, ...; t)$$

This must not involve recursive algebraic loops. a_1, a_2, b_1, b_2 are fixed parameter values associated with each grid point. The differential equation system (1) for each grid point is solved by numerical integration to produce time histories of the feature values: $q_i = q_i$ (t) and $p_i = p_i$ (t). Such an equation system might, for instance, model the growth of a crop, or the population dynamics of competing plant species at a point of the landscape.

SAMT lets you enter, display, and manipulate the database of grid-point features qi and pi , but the program was not designed to generate landscape-feature time histories by solving differential equations. This is left to DESIRE, a computer program specifically designed to solve large differential-equation systems. SAMT and DESIRE are loaded independently and communicate through data arrays and command codes in semaphore-protected shared memory. SAMT sends system parameters and initial values of landscape features to DESIRE; DESIRE then solves the differential equations and returns landscape-feature values to SAMT at periodic sampling times. Both DESIRE and SAMT can generate time-history graphs and listings, and SAMT can produce time-variable maps. The entire simulation of an evolving landscape can be controlled by DESIRE commands, by SAMT dialog-window controls or by an interpreted experiment protocol script.

4.0 Analysis Methods in SAMT

SMAT provides seven analysis methods:

- Simple Analysis
- Correlation
- Advanced Analysis
- Complex Analysis
- Radial Basis Function Network (RBF)
- Additional analysis methods (Variance, Hasse Diagrams)
- Sensitivity Analysis

4.1 Simple Analysis

This method includes statistic and a histogram functions. For example, the "STAT" function calculates statistical values of a grid such as minimum, maximum, mean, standard deviation etc.

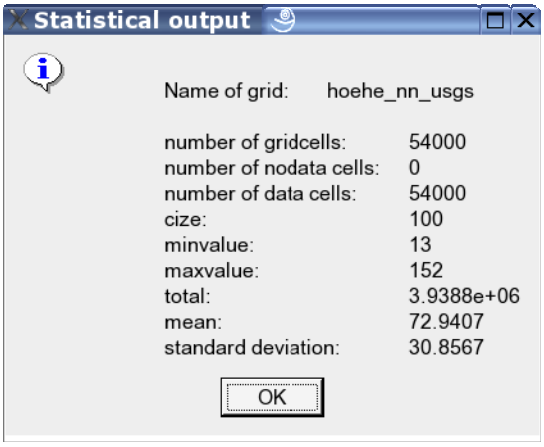

Figure 2. A Sample Statistical Output Window

Figure 3 below is the histogram output generated by "HIST" function. Number of bins in graph is *parameter1* (default 250 bins). This function provides an overview of a selected grid.

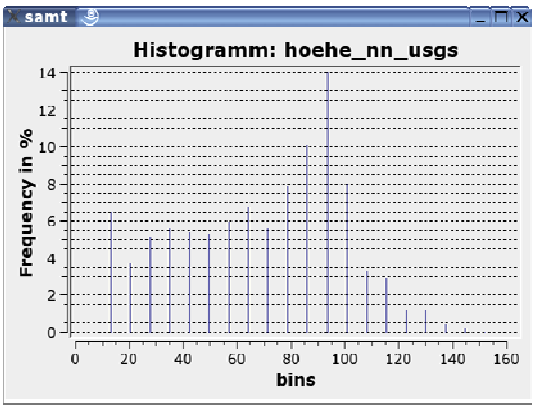

Figure 3. A Histogram Output

4.2 Correlation

This function is useful to compare two different maps. The function, first calculates the covariance of the two maps:

cov (map1,map2)=sum((map1[i,j]-m1)*(map2[i,j]-m2)) with m1=mean(map1) and m2=mean(map2)

The covariance and the variances of map1 and map2 is then used to calculate the correlation:

cov(map1,map2)/(sqrt(variance(map1))*sqrt(variance(map2))

or all design entry forms. The compiler generates various files for the result in P2=variance(map1) and P3=variance(map2).

As an example, the correlation of two grids, P1 and p2, is sometimes interesting. The correlation analysis method calculates correlation of the grid in P1 and the grid in P2 and shows the correlation in P1 and the variance of map1 in P2 and variance of map2 in P3.

4.3 Advanced Analysis

This method permits using the three-dimensional visualization software. The 3-D graphs provide special views to the data, like a simple elevation model and a splatter model. The use of 3D-visualization software currently is an extra step. This software runs independently from SAMT. The exchange of data between 3-D visualization software and SAMT is performed through a temporary hdf-file.

4.4 Complex Analysis

Complex analysis functions include a cluster algorithm and a kohonen feature map. This function is performed by an external program that communicates over an interface socket with SAMT.

The cluster algorithm is a simple method to group up to three grids and put it in a selected number of cluster (*parameter1*). The algorithm performs the following steps:

- selects n random points in the space (the cluster centers)
- for every grid cell, it calculates the Euclidian distance to the centers
- assigns the grid cells to the centers for minimum distance
- moves the centers into its center of gravity
- terminates if the centers are stable

In the kohon feature map, the cluster centers are modified depending on the location of activation.

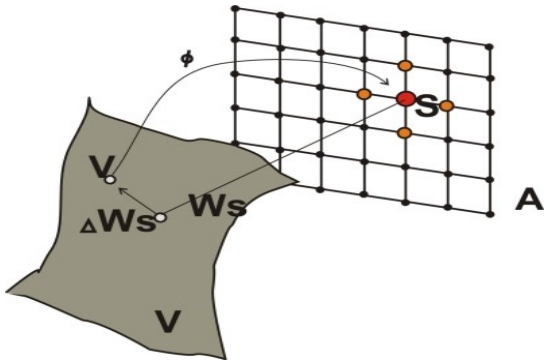

Figure 4 Shows the Cluster Number of the Center

5.0 Analysis Using DESIRE

DESIRE [5, 13] can read typed or programmed equations and differential equations like Eq. (1) in a natural mathematical notation. On a drun command, DESIRE immediately compiles and solves the problem with a choice of integration rules and produces time-history graphs or listings. While similar programs have long been used for simulating dynamic systems like aerospace vehicles, DESIRE is particularly convenient for interactive modeling [6]. Fast code for time-history solutions is compiled at runtime, but interpreted commands like "**reset**" lets the user modify the model and then try a new simulation run. Such interpreted commands can also be combined into useful experiment-protocol programs (scripts) that control multirun simulation studies, e.g. statistics evaluation (Monte Carlo simulation), and for parameter optimization or model identification.

5.1 Evolution at multiple grid points: the vectorizing compiler

Landscape evolution involves equations similar to (1) with different parameters and initial conditions at many points of a landscape grid. If there are not too many grid points (say up to 1000), DESIRE can define an n-dimensional array (vector) for each state variable q_i, for each defined variable p_i (landscape feature vectors), and for each parameter a_i and b_i (parameter vectors) with array declarations

STATE $P_1[n]$, $P_2[n]$, …. (2)

ARRAY $q_1[n]$, $q_2[n]$, …., $a_1[n]$, $a_2[n]$,….., (3)

One can effectively replicate each model equation n times by programming vector equations [7] corresponding to each scalar equation (1),

$$\text{Vectr } \frac{d}{dt}q_i = f_i(q_1, q_2, ...; p_1, p_2, ...; a_1, a_2, ...; t) \; i = 1, 2, \quad (4)$$

$$\text{Vector } p_1 = g_1(p_2, p_3, ...; q_1, q_2, ...; b_1, b_2, ...; t) \; i = 1, 2, \quad (5)$$

$$\text{Vector } p_2 = g_2(p_1, p_3, ...; q_1, q_2, ...; b_1, b_2, ...; t) \; i = 1, 2, \quad (6)$$

DESIRE's vectorizing compiler automatically compiles each vector equation (4) into **n** corresponding scalar assignments, one for each grid point, for example

$$\frac{d}{dt}q_i[k] = f_i(q_1[k], q_2[k], ...; p_1[k], p_2[k], ...; a_1[k], a_2[k], ...; t)$$

i = 1, 2, … k = 1, 2, .., n (7)

All the derivative values **d/dt(qi)** are fed to a selected integration routine. Vectorizing compilers produce efficient code, since there is no vector-loop overhead. Such compilers were originally developed for fast Monte Carlo simulation on supercomputers [8]. DESIRE brought this technique to inexpensive personal computers, where it also replicates neurons in artificial neural-network simulations and solves other problems. A simple personal computer can often solve thousands of differential equations in a fraction of a second.

5.2 Complexity of Input Data

A grid based GIS, though, may have to handle several millions of grid cells, too many for DESIRE's vectorization. But spatial data often contain a lot of redundancy. In practice, data are often quantified [11], so that only a discrete set of values are available. This is, for example, the case for fields in an agricultural used region (with a set of products like wheat, rye, corn, or potatoes); for soil types (every soil type is linked with a row of a table containing soil information like texture, profile data, etc.); and for climate regions like "arid", "semiarid" etc. Such a lumping into classes of input data can be used to reduce the amount of calculations dramatically. The number of required calculations is:

$n = nx_1 * nx_2 * nx_n$ with nx_i is the number of elements in the set x_i ..(8)

For three inputs with 8 fruits, 6 soil classes and 2 climate regions, we need for example:

n = 8 * 6 * 2 = 96 calculations. The number of grid cells was 54,000 (with 100m * 100m size), for this region.

Sometimes such quantization is not possible. Every grid cell represents a floating point number. This is true, for instance, when different geographical elevations are stored in grid cells. Another example is the representation of distances between grid cells and points (nesting places for birds, location of wind power stations etc.) or distances between grid cells and lines (roads, rivers etc.). A final important example involv-

ing continuous data demonstrates the so called moving-window technique [12], which calculates

$$q_i = f(x,y) = \iint_{A(x,y)} g(x,y)dxdy \qquad (9)$$

Here A(x, y) is an area around the point (x, y), and g(x, y) is a function depending on spatial modeling problem. For simple problems, g(x, y) can be the mean or the median for all points in a region A(x, y). This technique produces a spatial abstraction of data at one point.

5.3 Data exchange between SAMT and DESIRE

DESIRE obtains landscape-feature data from SAMT and simulates a dynamic process at each grid point. To avoid overloading DESIRE with data from too many closely-spaced grid points, one can feed DESIRE with data lumped into classes and return the result to SAMT. This lumping into classes means:

$$dq = (\hat{q} - \check{q})/(n-1) \text{ with } \hat{q} = max\{q_i\} \text{ and } \check{q} = min\{q_i\} \forall i \qquad (10)$$

With n number of classes, dq is the difference between the classes. SAMT interpolates between these data. For the two-dimensional case, SAMT employs the following interpolation algorithm:

(11)

$$a1 = q_{i,j} + (a - x_i)/(x_{i+1} - x_i) * (q_{i+1,j} - q_{i,j}) \qquad (12)$$

$$a2 = q_{i,j+1} + (a - x_i)/(x_{i+1} - x_i) * (q_{i+1,j+1} - q_{i,j+1})$$

$$q = a1 + (b - y_j)/(y_{j+1} - y_j) * (a2 - a1) \qquad (13)$$

With: $q_{i,j} = f(x_i, y_i)$ calculated value at the begin of the interval $[(x_i, x_{i+1})(y_j, y_{j+1})]$; $q_{i+1,j} = f(x_{i+1}, y_j)$ etc. ; a and b are values in the interval $[(x_i\ x_{i+1})(y_j, y_{j+1})]$; The values a and b represent the inputs. SAMT takes these inputs from every grid cell and interpolates, using the results from DESIRE, the output grid.

6.0 Example

The example below demonstrates the interaction between SAMT and DESIRE. A simple function to model the growth process of biomass is:

d/dt x = K*x (B-x)

x represents the biomass, K is a growth constant (growth effective temperature that is related to the elevation) and B describes a limitation of the growth process (the soil quality). These parameters are stored in SAMT.

The input grids are indicated in Figure 5, while Figure 6 shows the results. The soil map exhibits regions with equal soil quality. As described above, we used pre-classified soil data and we reduced computations by feeding DESIRE with elevation data lumped into classes. The K value was derived from elevation data and a classification from 0.0008 …. 0.0013 with 10 classes was used.

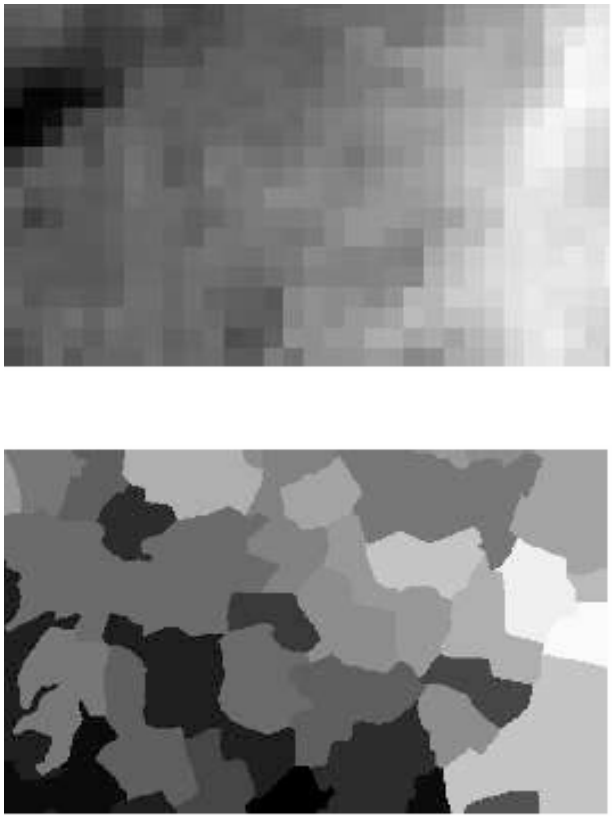

Figure 5: K-parameter and soil quality (brighter are higher values)

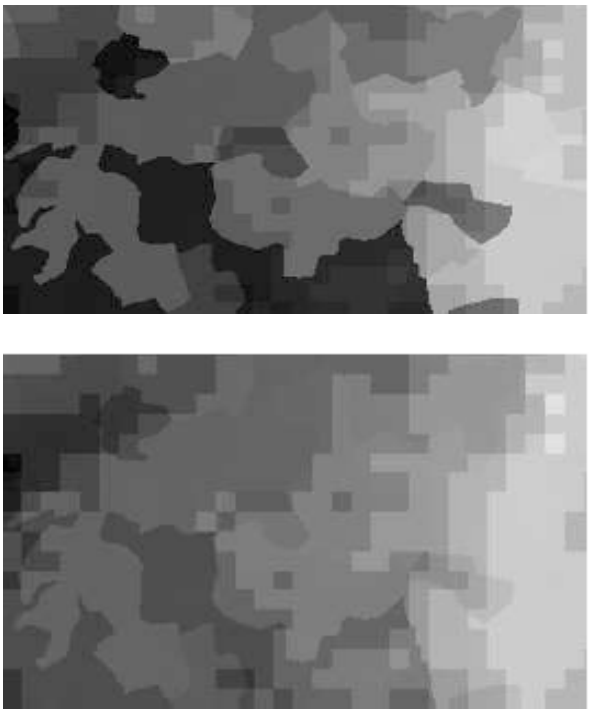

Figure 6: Biomass after 150 days and after 200 days

The result shows how the biomass varies with time. In the first map, the soil quality was dominant; in the second map we can also see the effect of temperature profile. Such maps can serve as inputs for follow-on simulations, say of wildlife-habitat quality [13].

6.0 Conclusions

DESIRE simulations can help to develop (spatial) models, estimate parameters and make sensitivity analysis of models, simulate processes along a line (streams, rivers), and simulate dynamics (see example)

DESIRE's efficient vector compiler handles simulations at 1000 or more landscape grid points. DESIRE's time-history displays let one check the progress of the calculation and verify results. This is far easier than modeling with C++ programs. The scripting language of DESIRE's experiment-protocol program, much like old-fashioned Basic, is easy to learn. SAMT/DESIRE added a new graphical user interface for DESIRE; this includes a new editor with syntax highlighting, improved file management, jump to the line with the error, and improved graph plotting.

SAMT/DESIRE serendipitously combines geographical information and dynamic-system models. We can also use SAMT's neural-network and fuzzy-logic subsystems [1] as independent toolboxes or to generate inputs for DESIRE differential-equation models.

6.0 Application and Future

SAMT has been successfully applied to projects in Germany, Switzerland and Mexico. In particular, SAMT was used to optimize the location of wind power plants with a view to reduce their detrimental effects on bird populations. Another SAMT application investigated possible landscape changes due to renewable primary products. Future applications of the new SAMT/DESIRE combination will investigate effects of new technologies (e.g. example genetically modified plants, alternative energy production) on the environment. One plan is to investigate the migration of copper compounds from Swiss house roofs into streams and rivers, considering diffuse-matter transport from the roofs, matter transport in streams or rivers, and chemical, biological processes affected by the copper migration.

7.0 References

[1] http://www.samt-lsa.org
[2] http://members.aol.com/gatmkorn/
[3] Piegl, L.; Tiller, W.: The NURBS book Berlin (Springer) 1997
[4] Wieland, R.; Voss, M.: Spatial Analysis and Modeling Tool - structure possibilities and applications in IASTED: Environmental Modeling and Simulation, Nov. 2004
[5] Korn, G.A.: Interactive Dynamic system Simulation, Gordon and Breach, London, 1998.
[6] Korn, G.A.: A Simulation-Model Compiler For All Seasons, Simulation Practice and Theory, 9:21-25, 2001.
[7] Korn, G.A.: Model-replication Techniques for Parameter-influence Studies and Monte Carlo Simulation with Random Parameters, Math. and Computers in Simulation, 67/6, 501-513, 2004.
[8] Korn, G.A.: Fast Monte Carlo Simulation of Noisy Dynamic Systems on Small Digital Computers, Simulation News Europe, 2002.
[9] Schroeder, W.; Martin, K.; Lorenzen, B.: The Visualization Toolkit; Prentice Hall PTR, 1998
[10] http://www.esri.com/software/arcgis/
[11] A. Glaton: Qualitative Spatial Change; Oxford University Press, 2000
[12] Kiesel J.; Lutze G.: \Skalierbare Generalisierungsmethoden zur Ausgrenzung von Landschaftseinheiten" - In: Wittmann J, Wieland R. [HRSG.]: \Simulation in Umwelt- und Geowissenschaften 2004" Shaker Verlag 2004.
[13] G. A. Korn and H. Vakilzadian, "Beyond Desire: Pascal Implementation of Direct-executing Simulation Languages," Mathematics and Computers in Simulation, 28 (1986) pp. 129 – 139.