SDM98 – Manual

Congratulations on your download of SDM98! The following manual briefly describes the features of the model, input and parameter files, and execution and output.

History

The cellular automaton model was originally written in 1996 in Turbo-Pascal for an MSc thesis: "Stochastic Dune Model for the Simulation of Dune Landscapes under Desert and Coastal Conditions" (Baas, 1996, University of Amsterdam), based on the original Werner (1995) model and extended with the effects of vegetation in the model space.

The code was rewritten in C^{++} in 1998 (hence the name SDM98) to compile and run under Unix. This is the version of the cellular automaton model described in Baas (2002) "Chaos, fractals, and self-organization in coastal geomorphology: simulating dune landscapes in vegetated environments" (*Geomorphology* 48, 309-328).

The current model has been modified and debugged in 2006 to run on Windows XP and Windows 2000 Professional by J.M. Nield and A.C.W. Baas.

Features

The manual presumes familiarity with the basic model algorithm as described in Baas (2002). The model-space is a grid with user-defined dimensions, *x* in downwind (streamwise) and *y* in transverse (spanwise) direction, with periodic boundaries connecting opposite edges of the grid (a self-contained model space). A wind regime sequence of up to 12 transport trajectories can be imposed, with trajectories defined as jump lengths in *x* and *y*, each with an individual duration in iterations. Oblique wind directions can thus be approximated by defining jump lengths in both directions to different degrees. The length of a vegetation cycle (growth season), and the total length of the scenario (simulation) are independently defined in iterations as well. The model contains four (pre-defined) selectable vegetation growth functions (see below), as well as the option to exclude vegetation altogether, i.e. **bare-sand** simulations.

Files

SDM98.exe	the executable, it can be activated from a Windows file-folder
scenario_filenames.txt	the file that defines all other filenames (exact name match required)
*windregimes.txt**	specifies transport trajectories, growth season duration, and scenario duration
*parameters.txt**	specifies model parameters
*grid_elevation.txt**	starting DEM of the model space
*grid_vegetation.txt**	starting vegetation coverage of the model space
*grid_balance.txt**	sedimentation balance across the model space

* can be renamed (name is defined in *scenario_filenames.txt*)

The precise content of these text-files is explained below as well as commented in the example files supplied in the download package

Running the model

Run the *SDM98.exe* file (e.g. double-click in Windows Explorer). The model requires a text file in the same directory with a preset name, "scenario_filenames.txt". This file lists all the other input files that define the scenario. The model then reads in these input files and executes the scenario. Different scenarios can be implemented by changing the *grid_* files (easiest done in MS Excel) and altering parameters and wind regimes (e.g. in MS Notepad).

Output

Output consists of the three *grid_* files preceded by the term ‘output’. In the example files the output will consist of the following three files:
- *outputgrid_elevation.txt*
- *outputgrid_vegetation.txt*
- *outputgrid_balance.txt*

Input File Content

The input files should ideally be located in the same directory as the executable and should have the following content conventions (actual content shown in Arial font):

scenario_filenames.txt:

```
windregimes.txt
parameters.txt
grid_elevations.txt
grid_vegetation.txt
grid_balance.txt
```

This file contains the names and extensions for the five data files required to run a simulation [strings].

windregimes.txt:

```
5  0  1
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0
0  0  0

100
10
```

Up to 12 different wind regimes may be entered (the first 12 lines of the file):
- the first column corresponds to the transport length in the *x* direction [integer]
- the second column corresponds to the transport length in the *y* direction [integer]
- the third column defines the duration of that trajectory in number of iterations [integer]

The second to last number (here: 100) represents the total duration of the scenario (simulation) in number of iterations [integer]
The last number (here: 10) is the duration of the vegetation cycle in number of iterations [integer], i.e. the length of time between updating the vegetation effectiveness according to the growth function.

parameters.txt:

```
0.400  0.600
1
33.000  40.000
15.000
100  100
0.1
```

This file contains the environmental parameters. In line-order, the values contained in this file represent:
- probability of deposition for (1) sandy surface and (2) bare ground [real numbers]
- vegetation growth function type number [integer]
- angle of repose (1) without and (2) with vegetation (in degrees) [real numbers]
- Shadow zone angle (in degrees) [real number]
- number of cells in the (1) downwind and (2) transverse direction [integers]
- slab height ratio (relative to cell-width) [real number]

Note:
Vegetation type (1) represents a burial dependent pioneer species.
Vegetation type (2) represents a burial loving pioneer species.
Vegetation type (3) represents a high tolerance successor.
Vegetation type (4) represents a low tolerance successor.
(see below for graphs of the vegetation growth functions)

To run the model in **bare-sand mode**, a vegetation type of 0 (zero) should be entered. The user should also be aware that for the **bare-sand mode**, the *grid_vegetation.txt* file should contain values of 0.0 only, as the vegetation values will not be altered in this mode.

grid_elevations.txt

This file contains the number of slabs on each cell, according to the grid-dimensions specified in *parameters.txt*, with values separated by spaces along the *x*-direction and lines along the *y*-direction. Values must be integers ≥ zero.

grid_vegetation.txt

This file contains the vegetation effectiveness on each cell, according to the grid-dimensions specified in *parameters.txt*, with values separated by spaces along the *x*-direction and lines along the *y*-direction. Values must be real numbers between [0,1]. For the **bare-sand mode** all values in this file should be set to zero.

grid_balance.txt

This file contains the sedimentation balance in number of slabs since that last growth cycle on each cell, according to the grid-dimensions specified in *parameters.txt*, with values separated by spaces along the *x*-direction and lines along the *y*-direction. These can be negative or positive integers (or zero). Note that this file is mostly for more advanced use so that a simulation can be resumed from an arbitrary starting point after ending a prior scenario in mid-season (and information on partial sedimentation balance is required).

Output File Content

The three output files follow the same format as the three input *grid_* files.

Vegetation Growth Functions

A graph of the four selectable growth functions is shown below. The growth is defined as an addition to, or deduction of, vegetation effectiveness as a function of the sedimentation balance on the cell, i.e. a negative balance if the cell is lower relative to the previous growth season (eroded depth) and a positive balance if the cell is higher relative to the previous growth season (accumulated depth). The balance is quantified in units of cell-width, so that with a slab-height of 0.2 a positive balance of 6 slabs deposited since the last season equals a sedimentation of 1.2 units. Vegetation effectiveness is confined to a maximum of 1.0 and a minimum of 0.0.

Vegetation types:

(1) a pioneer species that requires a positive balance to grow and that declines at a neutral balance (a burial dependent pioneer species)

(2) a pioneer species that requires a positive balance to grow but that can remain stable at a neutral balance (a burial loving pioneer species)

(3) a successor species with optimum growth at a neutral balance, but fairly tolerant of erosion and deposition (a high tolerance successor)

(4) a successor species with optimum growth at a neutral balance, and less tolerant of erosion and deposition (a low tolerance successor)

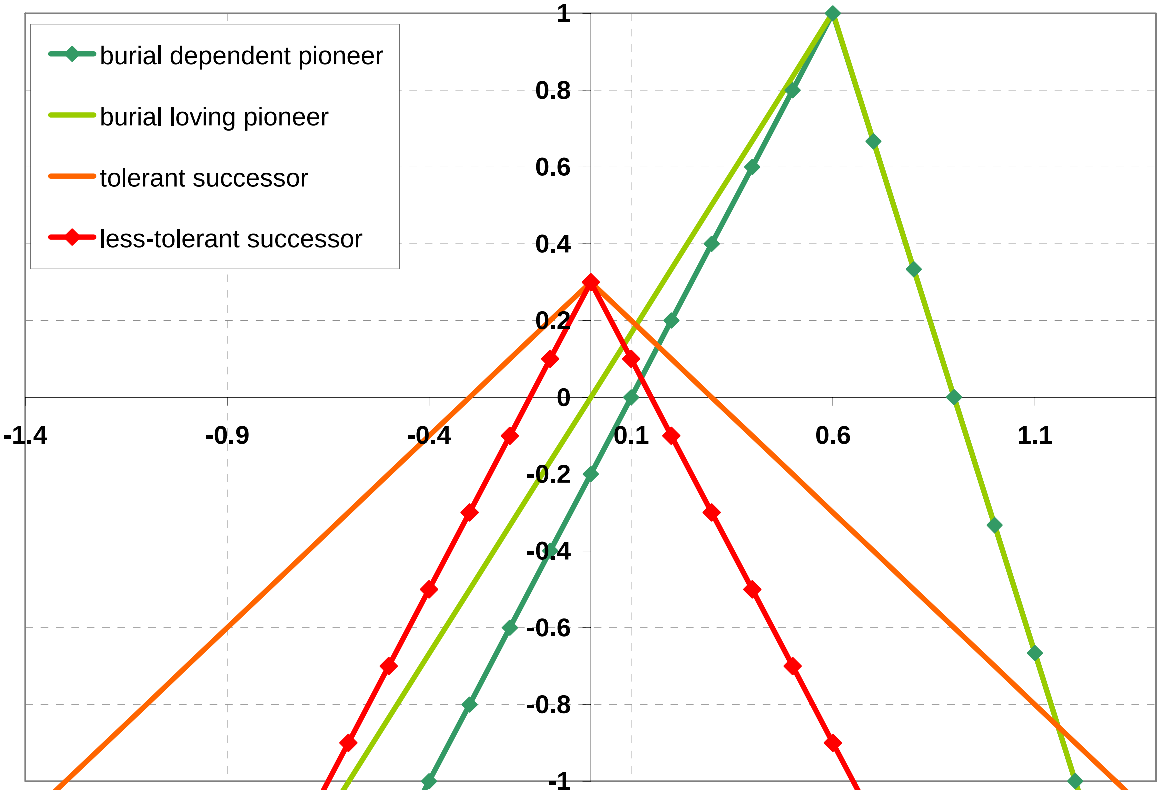

Figure 1: growth functions of the four selectable vegetation types.

© 2006 Baas & Nield
Department of Geography
King's College London
Strand, London, WC2R 2LS, UK
andreas.baas@kcl.ac.uk