School of Computer Science and Software Engineering
Monash University

Bachelor of Software Engineering Honours
Clayton Campus

Dramatic Landscapes

Cellular Automata Modeling of Landscape Phenomena

Thesis 2002

Adam Koh
Supervisor: Alan Dorin

November 11, 2002

Declaration of Originality

I, Adam Koh, declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given in the bibliography.

Adam Koh

11-11-2002

Abstract

In the field of computer graphics, automated generation of terrain featuring natural landscape phenomena has traditionally employed techniques that are highly specific to the effects being sought. Effects are often difficult to combine into a single scene, due to differing, specialised terrain representations and varying dynamic characteristics of the models employed.

This thesis presents a generalised approach to creating landscape effects that attempts to address these issues by combining *cellular automata*-based systems with *height field* representations of terrain. The technique is used to simulate stream erosion and desert dune formation.

Acknowledgments

I wish to thank my supervisor Dr Alan Dorin for his guidance, support, and insight during the course of this project, and also fellow student Suzannah Molloy, with whom ideas were shared and problems tackled.

I also wish to acknowledge the support of our course leader Dr Sita Ramakrishnan, and my Honours colleagues, all of whom underwent the many trials and tribulations called for this year, to become part of a precedence for Monash software engineering.

Contents

1 Introduction 1
1.1 Overview 1

2 Landscape Phenomena 4
2.1 Stream Erosion 4
2.1.1 Particle Transport by Water 4
2.1.2 Tributary Networks 5
2.2 Desert Dunes 5
2.2.1 Dune Archetypes 5
2.2.2 Sand Supply and Wind Characteristics 6
2.2.3 Sand Stability 6
2.3 Fallen Snow 7
2.3.1 Snow Accumulation 7
2.3.2 Wind Model 7

3 Terrain Representation 8
3.1 Height Fields 8
3.2 Limitations 9

4 Effect Representation 9
4.1 Cellular Automata 10
4.2 Self-Organised Criticality 11

5 Experiments and Results 12
5.1 Stream Erosion Model 12
5.1.1 Methodology 12
5.1.2 Results 13
5.1.3 Extensions 16
5.2 Desert Dune Model 17
5.2.1 Methodology 17
5.2.2 Results 18

6 Conclusion 20

7 References 21

A Appendices 23
A.1 Software 23

List of Figures

Figure 1: Applying the stream erosion effect to a landscape.................................2
Figure 2: Progression of the stream erosion model...3
Figure 3: Barchan dunes (top and side view)...5
Figure 4: Transverse dunes (top and side view)..6
Figure 5: Uniform and non-uniform height fields..8
Figure 6: Height field representations of landscape..9
Figure 7: 8-neighbourhood of a cellular automaton..10
Figure 8: Complexity in Conway's Game of Life..11
Figure 9: Stream erosion initial height field...13
Figure 10: Stream erosion initial height field rendering......................................14
Figure 11: Stream erosion height field after 1 pass..15
Figure 12: Stream erosion height field rendering after 1 pass............................16
Figure 13: Erosion with a ground softness map...16
Figure 14: Wind model testing..18
Figure 15: Dune formation height field after 300 iterations...............................19
Figure 16: Rendered representation of dune formation......................................20

Stream erosion on a fractal landscape

1 Introduction

Evocative and expressive, landscapes are commonly featured as the subject of computer graphics artists. The ability to easily create landscapes of dramatic scale and disposition is exploitable in a number of situations, including the creation of backdrops for composited imagery; the generation of environments for simulation purposes; and in images for the visual appeal of the landscapes themselves.

The field of animation has also revealed a need for techniques to reproduce the dynamic aspects of landscapes, namely, the types of effects (both natural and imaginative) that cause a landscape to vary over time. Many such effects, like snowfall accumulation and sand dune formation, involve prohibitively complex (and not well understood) interactions between numerous physical factors. In some cases, this complexity is handled by reducing the physics of the effect in question to numerical descriptions ([LM93]), producing solutions that rely on partial differential equations, for example. Other techniques ([Fea00]) circumvent the dynamic aspects, using non-simulatory approaches to approximate a final scene. The latter approach, in particular, may not be suitable if the subject of the animation happens to be the dynamic aspects.

In addition, techniques designed to produce specific effects tend to rely on particular underlying representations of the landscape. For any set of effects, the disparity in terrain representations can mean that combining those effects – to produce, say, a landscape featuring both glacial and stream erosion – is difficult or impossible to do.

This thesis presents an alternative approach to modeling a range of landscape effects, using *cellular automata*-based systems with a common *height field* representation of terrain. The use of a standard terrain representation, along with a flexible approach to modeling effects, allows a given piece of terrain to be run through a sequence of effects, thus easily accommodating combinations of effects. The choice of using cellular automata systems also affords advantages in the potential for parallelism, and in simplicity when deriving the rules to drive a particular effect.

1.1 Overview

The approach presented herein represents terrain using a height field. Application of a particular landscape effect involves varying the height values using a cellular automaton – that is, by repeatedly examining the state (height) of each point on the field and updating its value according to some function of its state and the state of its immediate neighbours. Depending on the desired effect, the state information of each cell may be expanded to include more than a mere height value. The stream erosion model we develop, for example, also includes flags to indicate the presence of water at each cell.

An example of a working application based on this approach can be seen in Figure 1. Height values are shown with darker shades corresponding to lower regions, and brighter shades marking the higher regions. When running, the display is updated after each time-step of the cellular automaton.

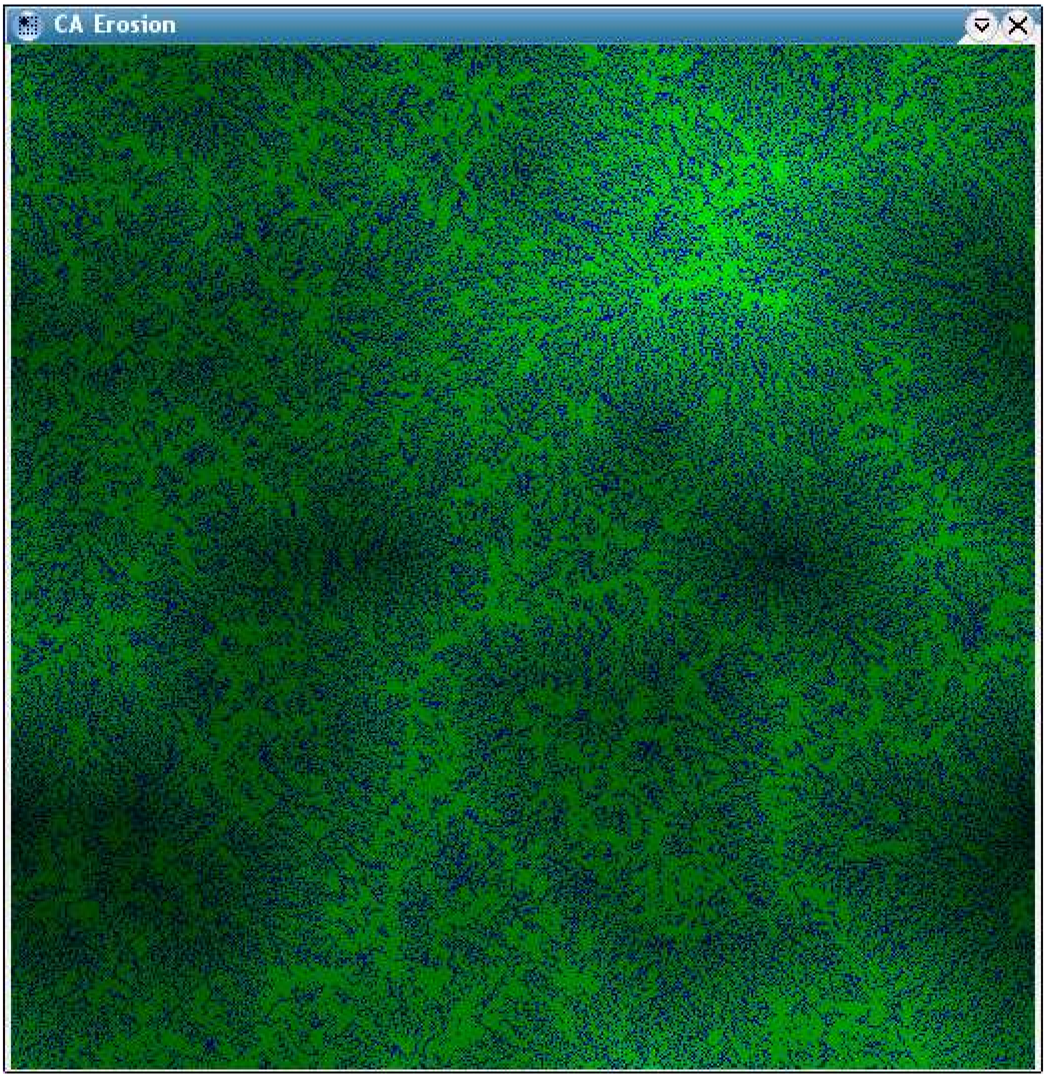
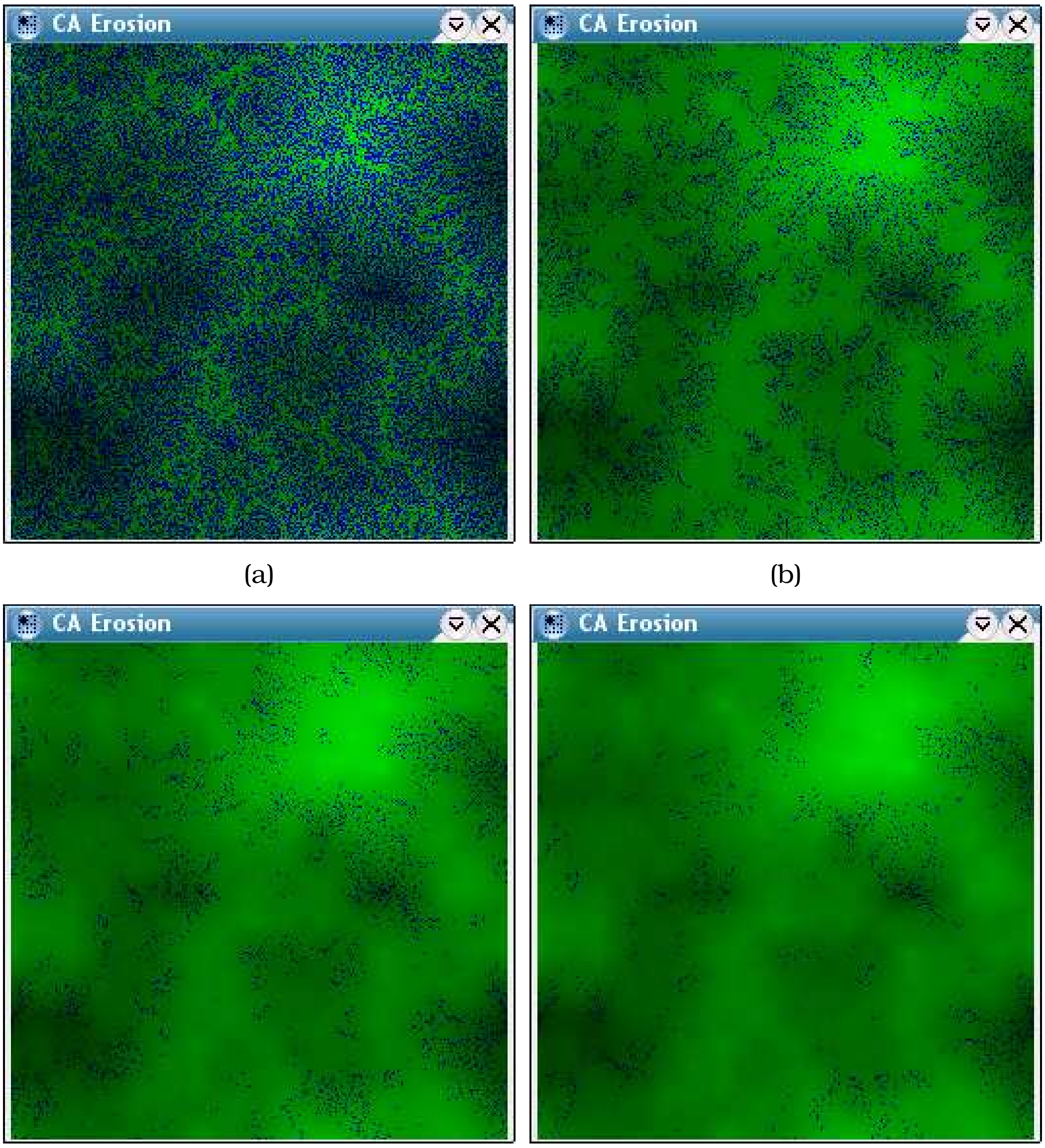


Figure 1: Applying the stream erosion effect to a landscape

Additional cell state specific to an effect is shown in varying ways, dependent on the effect. In Figure 1, the stream erosion model is being applied. Green areas mark terrain without water, while blue areas indicate the presence of water. User controls are also introduced as appropriate, to support the operation of a particular effect as well as to provide basic facilities, such as saving the output. Our example above allows water to be introduced to the terrain on command.

Configurable parameters are supplied via a text configuration file, in which the initial terrain may be specified (eg. flat, fractally generated, loaded from an image file), along with any parameters related to the effects.

The rate of display updating is a function of the terrain size, effect complexity, and the hardware being employed, but is generally fast enough for the action of the cellular automaton upon the landscape to be perceived as a continuous process. Figure 2 shows the progression of the stream erosion model over time.



(a) (b)

(c) (d)

Figure 2: Progression of the stream erosion model

Effects can either be truly continuous (possessing no definite “finishing” point, eg. dune formation), or they can operate in *passes*. A pass of a particular effect is thus defined as being the process of stepping the cellular automaton until it reaches a stable state, where the terrain is no longer modified by its actions until further external intervention takes place.

Figure 2a shows the state of the erosion system a few moments after the user has introduced the water. Figure 2b and 2c are taken in successive periods of time afterwards. By the time of Figure 2d, nearly all of the water has been absorbed, and one pass of the stream erosion effect is almost complete.

2 Landscape Phenomena

For this project, three main types of landscape phenomena were examined: stream erosion on firm terrain; desert dune formation; and the transport and accumulation of fallen snow. Essentially, all three can be generalised as being variations on particle transport, though the nature of that transport differs greatly. In particular, desert dune formation and snow accumulation can both be characterised as forms of particle transport in a fluid field. These phenomena were deemed sufficiently varied, and of suitable complexity, to serve as candidates for the proposed modeling approach.

The relevant literature is comprised chiefly of writing from two main categories: accounts and analyses of observable characteristics of these phenomena in real-world environments; and descriptions of techniques used to model or capture specific aspects of those characteristics. Where the latter is concerned with the dynamic properties of a phenomenon, fluvial modeling techniques are almost always discussed.

The following sections further describe aspects of modeling the three phenomena.

2.1 Stream Erosion

A great amount of work has been carried out in the study of erosion and transport by water, from both physical and chemical perspectives. However, from a terrain modeling perspective, it is the physical effects of sedimentation that are most noticeable in landscapes, so we focus on erosion from a purely physical perspective. Specifically, the emphasis is on reproducing the distinct forked gullies produced by stream networks on uneven terrain.

2.1.1 Particle Transport by Water

Twenhofel describes three principle modes of transport by water as being traction (movement by rolling or small jumps), suspension (movement by floating), and solution [Twe39]. The first two concepts incidentally agree with corresponding principles in the literature on particle transport by wind, which popularly defines creep (rolling movement along the surface), saltation (movement in small leaps), and suspension (higher altitude movement, lifted by wind) [Bag73, Lan89]. The general conjecture appears to be that particles of significant size (larger than “colloidal” dimensions, according to Twenhofel) are repeatedly removed from the terrain surface, and replaced at short intervals before being able to travel very far. Particles of smaller sizes become suspended in the stream, and are carried for distances that may be much longer. In terms of our model, this is represented by factors which control the rate of particle removal and deposition at each cell.

2.1.2 Tributary Networks

Modeling of the actual flow and formation of stream tributaries has been accomplished by Kelley, Malin, and Nielson [KMN88]. Their technique uses a polygonal surface representation of the un-eroded terrain, applying empirical data from geomorphological models to form tributary networks. They present a method of representing the surface as being under tension, so that the initial form of the landscape may dictate the forms of the resulting eroded valleys. In our model, a similar effect is accounted for by the cellular automaton rules, which cause water to favour steeper slopes, in turn causing erosion activity to be more pronounced on mountainsides and in valleys.

2.2 Desert Dunes

Perhaps because of their innate curiosity, dunes have garnered significant attention from researchers. The literature indicates that desert dune activity simultaneously features two separate physical actions: the transportation of sand by wind, and the collapse (by gravity) of the sand as it forms piles.

2.2.1 Dune Archetypes

Lancaster describes the processes of desert dunes [Lan89], confirming the existence of clear dune archetypes that are also identified by other researchers in the field [PF00]. He states that there is little evidence that grain size and sorting characteristics are related to the type of dunes formed. Among the common dune archetypes are the horseshoe-shaped *barchan* dunes, and the parallel *transverse* dunes.

The former possess a windblown slope (the “outside” of the crescent) and a collapsing front with two “horns” which point downwind. The dunes are additionally observed to creep forward in the windward direction over the course of time, occasionally catching and merging with dunes of other sizes.

(a) (b)

Figure 3: Barchan dunes (top and side view)

Transverse dunes, on the other hand, are identified by their long striations which run perpendicularly to the wind direction. They, too, possess a collapsing front and windblown back.

(a) (b)

Figure 4: Transverse dunes (top and side view)

2.2.2 Sand Supply and Wind Characteristics

In relation to barchans, Lancaster suggests that their occurrence demands consistent sand volume (without greatly fluctuating supply), and "almost unidirectional winds." Both factors are appealing for their simplicity, lending credence to our use of particles of uniform size, and to the wind simplifications described later for our model.

Additionally, Lancaster proposes that the eventual appearance of a dune is the result of an equilibrium between "sand supply and removal at each point on the dune, summed for the dune as a whole." Such equilibrium behaviour is frequently encountered in the literature as *self-organised criticality*, and is discussed further in this report. Our model incorporates this concept using sets of rules that dictate the displacement of sand over multiple points on the terrain.

2.2.3 Sand Stability

Closely related to sand stability, the characteristics of wind movement on the surface of dunes is explored in detail by Burkinshaw and Rust [BR93]. Their observations on sand removal rates and the related gradients of the dune surface serve as bases for empirical comparison of the results of our model. Of note is the fact that they state a 32—34° slope on the collapsing face of the dune, which is consistent with the *angle of repose* of dry sand reported throughout the literature.

The angle of repose, or critical angle, is defined as being the steepest sloping angle naturally held by dry sand. Attempting to form steeper piles invariably causes the sand to collapse in chained avalanches so that the angle of repose is maintained, consistent with *self-organised criticality* (discussed later in this report).

2.3 Fallen Snow

Due to the complexity of packing, collapsing, and wind interaction of snow, documented efforts to capture its behaviour are comparatively scarce. Overwhelmingly, the literature in this area is primarily concerned with snow as a source of water [Pea73, BP73]. One point of interest in this is that snow and rainfall are closely related, only significantly differing (from our perspective) in how each reaches the ground. Rogers and Yau [RY96] expand on this relationship, stating that snow flakes, composed by the aggregation of ice crystals, account for fallen snow in its familiar form, and that snow flake data is usually measured in terms of particle mass.

2.3.1 Snow Accumulation

The work of Chopard and Masselot [CM99, MC72], which aims to model snow transport as the combination of a fluid and particle field, shows that falling snow shares at least some similarity to windblown sand. Notable differences are that snow is able to "stick" without stable support, drifts much further and with less regularity on the wind, and accumulates more by falling rather than creeping. The model they propose is in many ways very similar to the approach described herein, although it acts upon discretised particles rather than height values.

Fearing [Fea00] describes a method for automatic computation of the amount and form of fallen snow over a polygonal scene. Many aspects of falling snow are considered, including the proper distribution of snow volume over the scene, the nature of falling flakes (and flake "flutter"), and accumulation based on occlusion by other structures. His use of a two-stage system (one to compute the volume of snowfall, and a second to handle the collapse of unsteady snow) is reflected in our sand stability model.

2.3.2 Wind Model

Wejchert and Haumann [WH91], and Yaeger, Upson, and Meyers [YUM86] independently propose different models for simulating wind flow and atmospheric disturbance. Some concepts from their systems are drawn upon for our implementation in the dune formation model, although the resulting wind model is greatly simplified.

Unfortunately, the lack of an easily identifiable angle of repose (the critical angle for snow varies considerably as the snow itself varies from mush to hard packed), and the complex collapsing behaviour (snow can collapse in sliding sheets, or stick in apparently unstable formations) meant that, under the time constraints, developing a model for snow was not feasible.

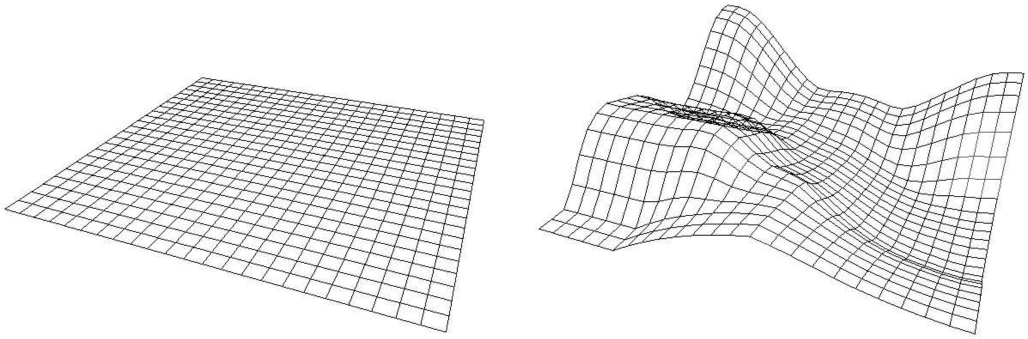
3 Terrain Representation

As described in the preceding section, existing techniques for modeling landscape phenomena rely heavily on specific underlying representations of the landscape. The terrain tension model for stream erosion [KMN88], for example, does not translate well to Fearing's snowfall model [Fea00]. For our needs, this was less than satisfactory. Since our goal was to allow different effects to operate on the same landscape, a common terrain representation was adopted instead.

3.1 Height Fields

Height fields are a common technique for representing terrain in computer graphics. Their widespread use stems from their simplicity, as well as their being flexible enough to represent most terrain forms.

Essentially, height fields are grids of values that represent the height of the terrain surface across the grid area. The surface formed by points at those heights defines the terrain. Figure 5a and 5b show visualisations of two-dimensional height fields with uniform and non-uniform heights, respectively.



(a) (b)

Figure 5: Uniform and non-uniform height fields

For greatest flexibility, height values should ideally be in floating-point form. However, the POVRay renderer used to produce the visualisations limits height field values to 16-bit integers. Alternatives were explored, such as generating primitives in place of POVRay height fields, but problems with performance and stability were encountered when attempting to render landscapes with over 512x512 points.

3.2 Limitations

The simplicity of using height fields nevertheless poses some limitations. Most noticeably, the terrain cannot feature overhangs of any sort. This is illustrated in Figure 6a and 6b, for a one-dimensional height field in profile. Note that in in centre of Figure 6b, where the region of the landscape overhangs itself, the height values are misrepresented.

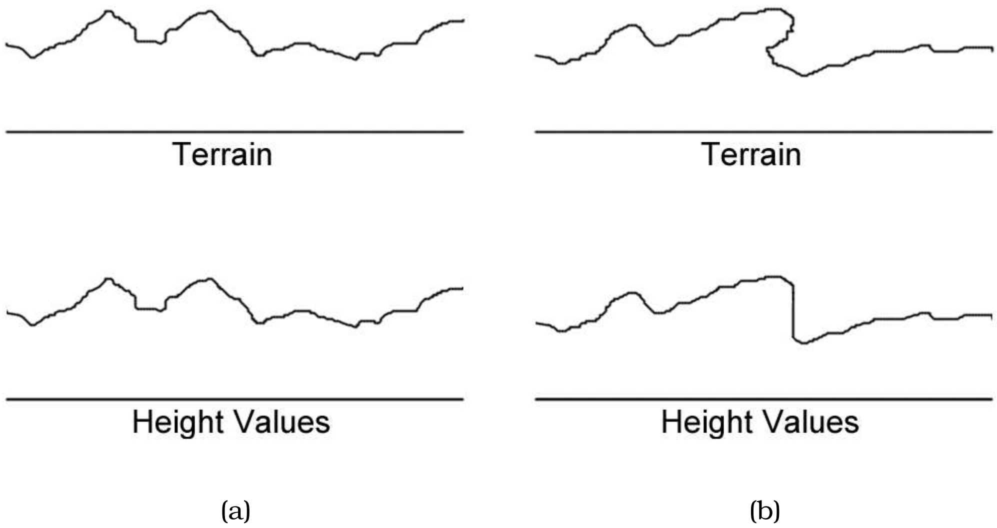


Figure 6: Height field representations of landscape

The stability behaviour of sand guarantees that overhangs cannot occur. It would therefore seem that only the stream erosion model suffers from the height field limitation. However, the literature strongly suggests that stream erosion processes are largely surface interactions, as described by the three processes of traction, suspension, and solution [Twe39], where the great majority of particle transport occurs by traction, and to a lesser extent by suspension. We argue, therefore, that these limitations do not significantly affect either of our models.

4 Effect Representation

With a fixed representation of the terrain, it falls to the method of representing effects to be general enough to accommodate different landscape phenomena. The desired properties of such a system are simplicity, to interact efficiently with potentially very large height fields; and flexibility, so that effects can be modeled at a sufficient level of complexity. Cellular automata served this role in our model.

4.1 Cellular Automata

Cellular automata are simple systems which operate on a lattice of cells. Cells can be in various states, and have a *neighbourhood* of surrounding cells which depend on the topology and dimensionality of the automaton in question. For instance, the neighbourhood $n_{1..8}$ of a cell c in a rectangular two-dimensional cellular automaton appears in Figure 7.

n_1	n_2	n_3
n_4	c	n_5
n_6	n_7	n_8

Figure 7: 8-neighbourhood of a cellular automaton

At each time step, all the cells of an automaton update their state in parallel, according to some function F where

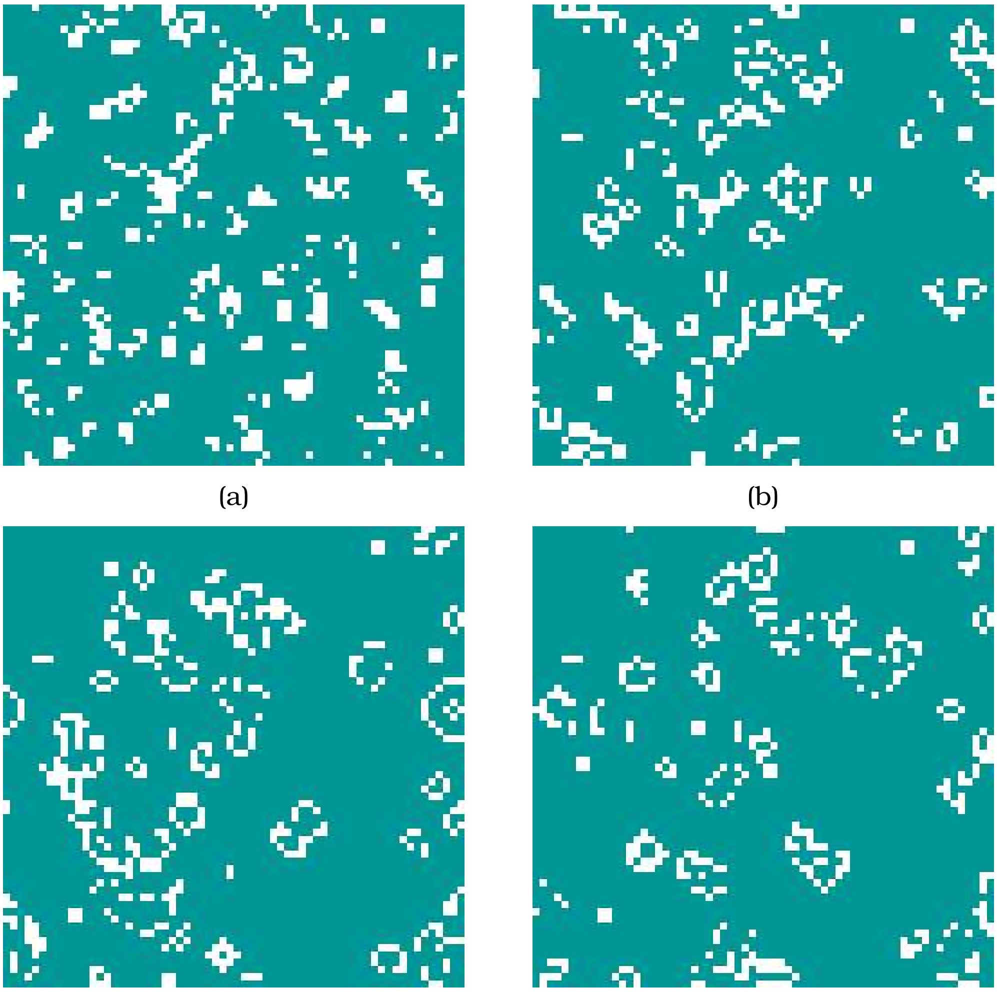
$$future\_state(c) = F(c, n_1, \dots, n_8)$$

F is generally represented as a set of rules relating the future state of a cell to its current state and the state of its neighbours.

Cellular automata in general are well documented by a string of researchers, beginning with Stanislaw Ulam, and extended by others such as Von Neumann [vN51, vN52], Holland [Bur70], Conway [Gar70], and Wolfram [Wol84, Wol02]. A recurring theme in the literature is that cellular automata are capable of exhibiting highly complex global behaviour while being rooted in very simple local rules. This behaviour is demonstrated by Conway's Game of Life, which uses the following rules:

1. Cells have boolean state: *dead* or *alive*.
2. If a cell is dead and has 3 living neighbours, it becomes alive (reproduction).
3. If a cell is alive and has 2 or 3 living neighbours, it stays alive (survival).
4. Otherwise, cell dies (from overcrowding or starvation) or stays dead.

The simplicity of these rules belies the complexity which follows from them, as demonstrated by the following progression, beginning with a randomly initialised grid in Figure 8a, and stepping in 10 iterations each through Figure 8b, 8c, and 8d. Despite being only 10 iterations apart, little correlation is apparent between each frame.



(a) (b)

(c) (d)

Figure 8: Complexity in Conway's Game of Life

The ability to produce complex behaviour from simple rules makes cellular automata a logical choice for our effects model. A rectangular cellular automaton maps trivially onto a height map, and the particular ruleset chosen thus corresponds to the effect being modeled.

4.2 Self-Organised Criticality

The occurrence of *self-organised criticality* – an effect analogous to (and often described as) the unpredictable, chained avalanches that keep the slopes of a sand pile at a consistent angle – in cellular automaton systems has been examined by Malamud and Turcotte [MT00]. Their paper is of particular interest in that it specifically discusses self-organised criticality in relation to cellular automata. Self-organised critical behaviour is evinced by many natural systems, in which an inherent state of balance is reached through continual reorganisation.

The exhibiting of self-organised critical behaviour appears to play two influential parts in desert dune processes: firstly, the balance between sand removal and deposition rates (which are themselves functions of wind speed) dictates the eventual size of the dune; and secondly, the shape of the dune is dictated by the collapsing behaviour which maintains the critical slope angle.

However, it should be stressed that no scientific evidence exists to suggest that self-organised criticality itself is a controlling factor of natural systems – rather, it is a convenient concept for discussing the critical-state balancing that occurs in many systems.

In terms of our model, self-organised criticality suggests that dune sizes will be limited at some critical point, related to the wind and sand transport parameters.

5 Experiments and Results

Based on the approach described thus far, we developed two models: one to reproduce the effects of stream erosion, and one to model the formation of sand dunes. Since the underlying terrain representation is the same in each case (ie. a height field), the models are capable of acting on arbitrary terrains, and do not depend on any particular initial state. This is consistent with our desire for combinable effects.

The following sections detail the models and present some of their results.

5.1 Stream Erosion Model

The stream erosion model aimed to reproduce the forked gullies formed by streams on sloping terrain. For the purposes of running the model, landscapes were generated as height fields using a basic *diamond-square* fractal subdivision algorithm [Sha00].

5.1.1 Methodology

The cellular automaton state data for this model consisted of the height values and flags indicating the presence of water at each cell. The effect operates in separate passes. Each pass begins by flooding the landscape with water (setting the water-present flag for all cells). Cells are then updated as follows:

1. If a cell has water and lower neighbours, it moves water to the lowest neighbour (flowing down steepest slope). The height at that cell is decremented.
2. If a cell has water and no lower neighbours, water is removed (absorbed).

A minor difficulty was presented in correctly updating the water-present flag when water is passed from cell to cell. Since cells only update their own state, and not that of their neighbours, passing water to a neighbouring cell cannot be done by directly setting that neighbour's water-present flag. The solution was to introduce a set of flags (one for each neighbour) to the cell state data, representing whether water is moved to any of those neighbours at each time step. Cells are then able to check the corresponding flag of their neighbours and update their own water-present state accordingly.

5.1.2 Results

The results of one pass are illustrated here. Figure 9 shows the initial height field generated using the diamond-square algorithm with low fractal noise. Figure 10 is a POVRay rendered representation of the same height field. Figure 11 is the resulting height field after a complete pass of the stream erosion effect, and figure 12 shows the rendered result, with erosion gullies clearly visible.

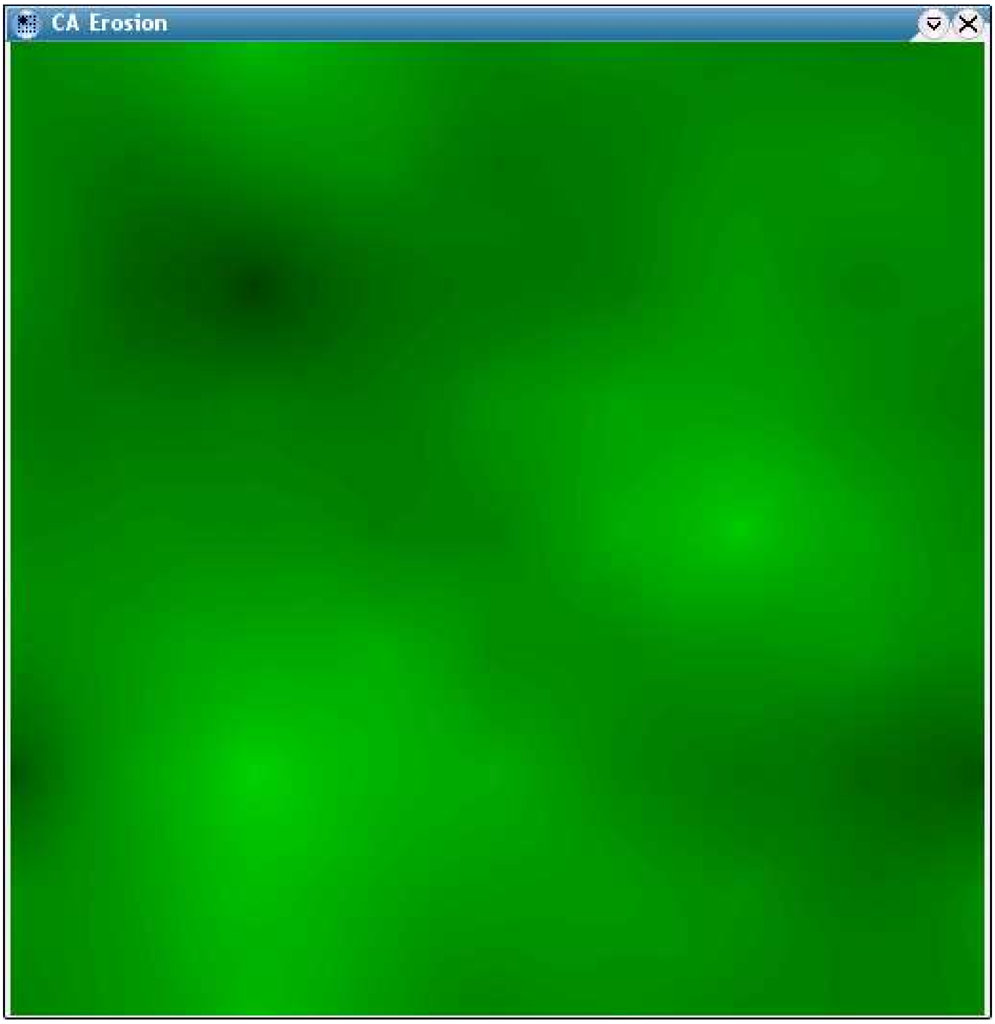


Figure 9: Stream erosion initial height field

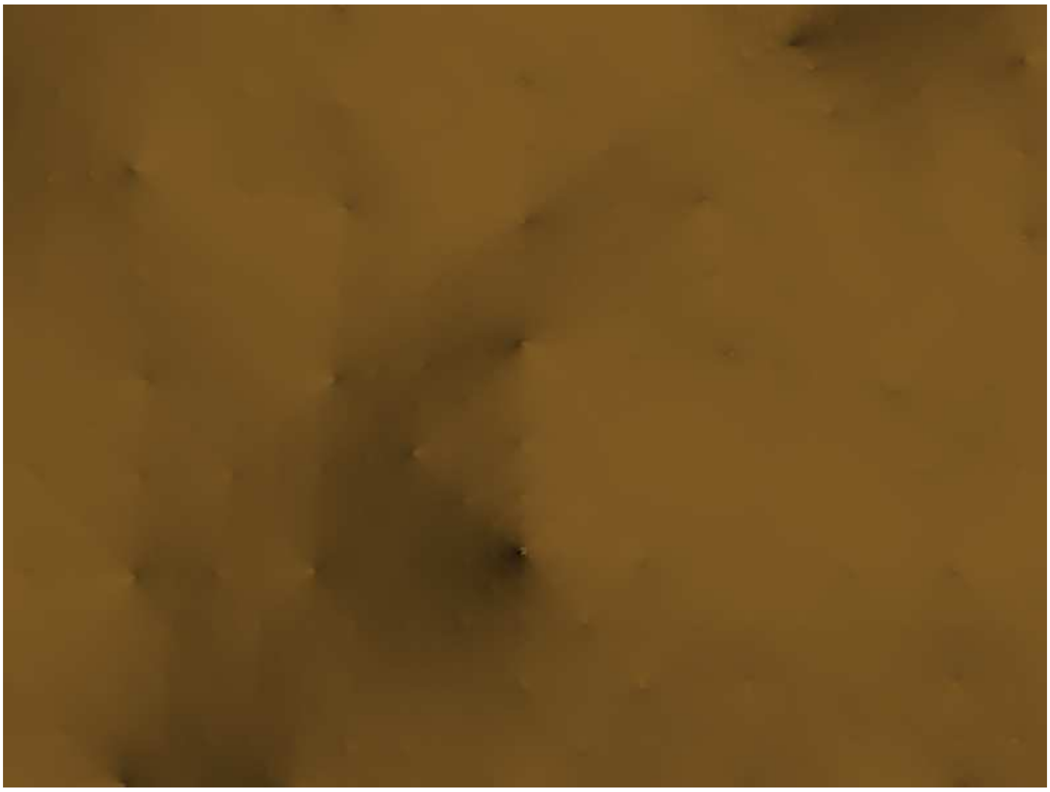


Figure 10: Stream erosion initial height field rendering

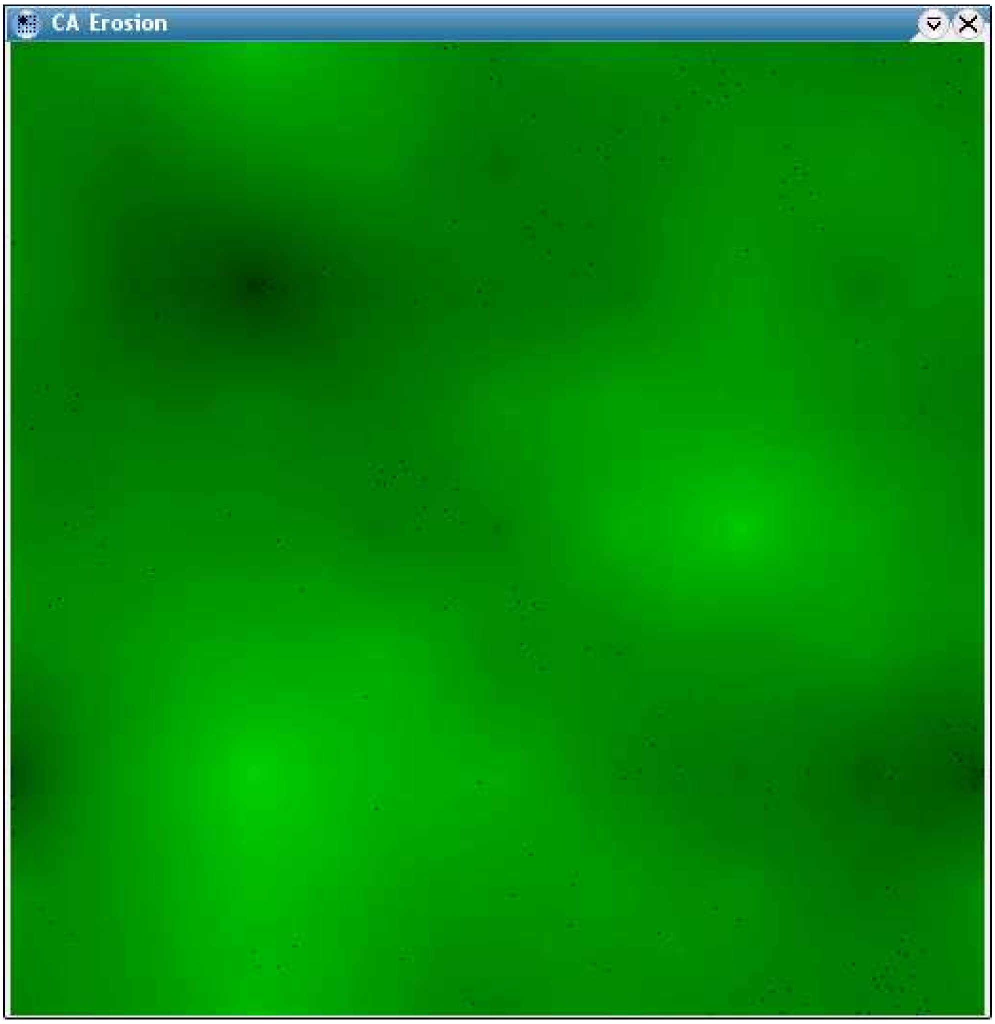


Figure 11: Stream erosion height field after 1 pass

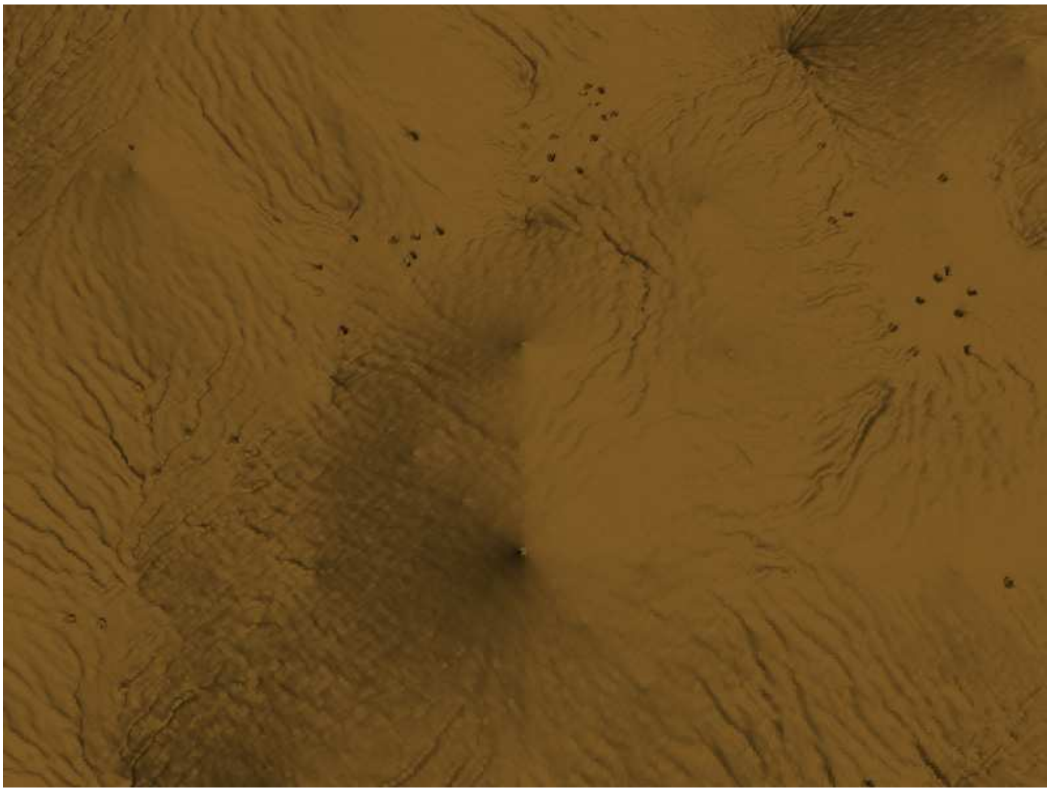


Figure 12: Stream erosion height field rendering after 1 pass

5.1.3 Extensions

Some extensions were also explored, including the use of a ground "softness map", which specified the rate of erosion at each cell, to simulate the presence of soil of varying densities and compositions. A whimsical example of the result obtained is shown in Figure 13, in which an image of a smiley face was used for the softness map.

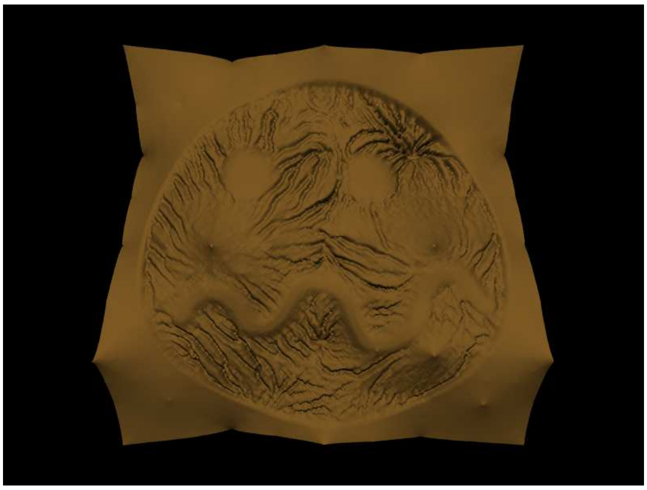


Figure 13: Erosion with a ground softness map

5.2 Desert Dune Model

The model for dune formation was considerably more complex, attempting to reproduce the formation and movement of desert dunes (along with their coalescing behaviour). Since the actual formation of dunes was one of the desired features, and because of the aforementioned observations of the sand supply characteristics required of dunes, it was decided that the landscape be initially flat, and be blown by wind into dunes.

5.2.1 Methodology

To model the interaction between sand and wind, the dune cellular automaton is combined with a discretised wind computation that occurs before each time step. Several models of wind behaviour were attempted, including a model of wind as a self-equalising pressure field, and another as a variation on thermal conductivity. However, it was found that a simpler model, operating as described below, provided suitable results.

The wind pass was updated as follows:

1. Wind is considered strictly uni-directional, having only normalised velocity at each cell between 0.0 and 1.0.

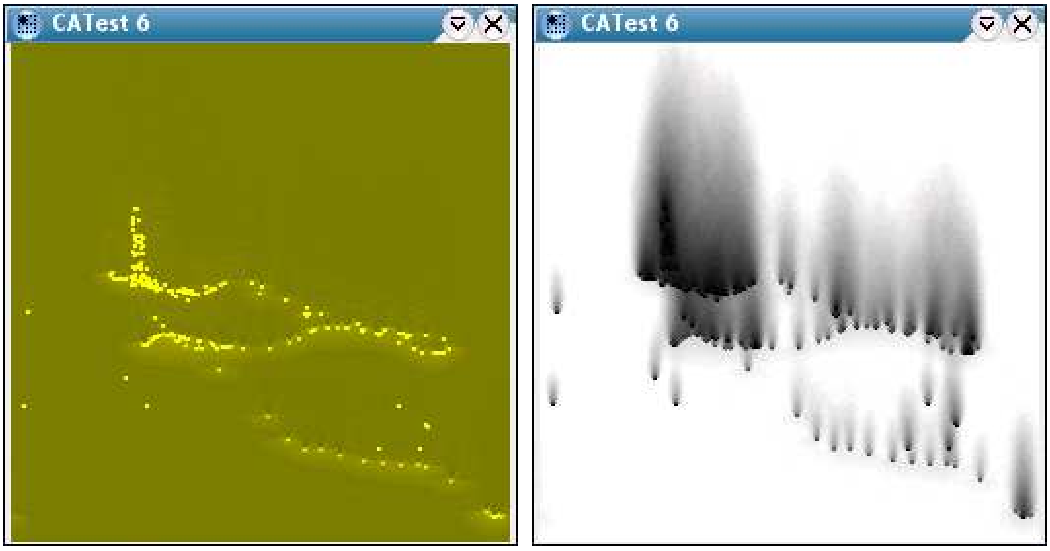
2. The row of cells closest to the wind source is initialised to have full wind (wind speed of 1.0).

3. Successive rows in the wind direction are calculated in turn. Each cell's wind strength is the average of the three neighbouring cells that are in the wind direction.

4. The wind at each cell is reduced according to the difference in height values of the cell and the next cell in the wind direction, by a parameterised factor, to represent occlusion by the landscape.

This wind model was evaluated using an interactive implementation, in which obstacles could be “drawn” with the mouse over a height field, to observe the corresponding effects of the wind. A screen from the running application is shown in Figure 14a as a height field, and its corresponding wind velocity field in Figure 14b. Wind flow is from bottom to top, and wind strength is shown with darker shades representing less strength, and lighter shades representing higher strength.

Some sand piling can also be seen in Figure 14a, since the test implementation also incorporated an initial version of the sand drift model described below.



(a) (b)

Figure 14: Wind model testing

With the wind being dynamically updated each time step, the normalised wind velocities are used as probabilities for sand movement. The total volume of sand was conserved by: 1) not introducing additional sand (as might be the case in a real desert, when distant sandstone cliffs are eroded), and 2) using a toroidal-space cellular automaton, where the cells at edge rows have their neighbours mapped to the opposite side of the field.

Sand is then transported as follows:

1. If wind strength at a cell is greater than a parameterised threshold, sand can be transported. Sand is then removed if the wind strength is greater than a random percentage, but only if the slope (difference in height to the three neighbouring cells in the windward direction) is sufficiently shallow (another parameterised value).
2. If sand has been determined as being transported, the lowest of the three neighbours in the wind direction is found, and chosen as the destination for the sand.

Once again, the movement of sand from cell to cell is carried out in similar fashion to that described for the movement of water in the stream erosion model. Many variations on the rules for sand transport were attempted, but the above provided the most pleasing results.

5.2.2 Results

The results after 300 iterations (beginning with a flat landscape) are shown below. Figure 15 is the height field representation, clearly showing the parallel striations characteristic of transverse dunes. Figure 16 is a POVRay rendered image of the same height field, from a lowered perspective. When animated over a sequence of iterations, the dunes can be observed to creep forwards in the wind direction, occasionally coalescing into each other or breaking into smaller dunes. By the 300th iteration, self-organised critical behaviour is observed in the dune sizes and intervals, both of which are retained indefinitely.

Unfortunately, a model for the formation of barchan dunes was never completed, due in part to time limitations. It is especially regretful since preliminary work in "growing" dunes with barchan-like horns appeared to produce good results.

Figure 15: Dune formation height field after 300 iterations

Figure 16: Rendered representation of dune formation

6 Conclusion

We have devised an approach for generalised modeling of landscape effects, using cellular automata-based mechanisms with a common terrain representation. The limitations imposed by incompatible terrain representations are thus overcome. The technique has been successfully applied in reproducing two different effects: stream erosion and desert dune formation.

A possible extension could see to generating the landscape texture in a similar manner, so that cliffs and erosion gullies could be rendered more realistically.

Because multiple effects can be applied to a single landscape, the approach established here provides some interesting avenues of exploration. One possibility is that a flexible terrain generation application could be built, employing a library of effects that can be selectively applied (genetic selection could prove to be interesting). Similarly, a detailed simulation of the formation of some hypothetical landscape is made possible. Such a simulation could employ empirical models (or even input from sources such as text or music) to select the locality and type of effect to apply, drawing from a bank of effects to “evolve” a landscape.

7 References

[Bag73] R. A. Bagnold. *The Physics of Blown Sand and Desert Dunes*. Chapman and Hall, 1973.

[BP73] K. Blyth and R. B. Painter. Analysis of snow distribution using terrestrial photogrammetry. In *Advanced Concepts and Techniques in the Study of Snow and Ice Resources*, pages 679—687. National Academy of Sciences, 1973.

[BR93] J. R. Burkinshaw and I. C. Rust. *Aeolian Sediments: Ancient and Modern*, pages 13-21. Blackwell Scientific Publishing, 1993.

[Bur70] Arthur W. Burks, editor. *Essays on Cellular Automata*. University of Illinois Press, 1970.

[CM99] B. Chopard and A. Masselot. Cellular automata and lattice Boltzmann methods: a new approach to computational fluid dynamics and particle transport. Technical report, University of Geneva, Computer Science Department, March 1999.

[Fea00] P. Fearing. Computer modeling of fallen snow. In *Proceedings of SIGGRAPH*. ACM Press, 2000.

[Gar70] M. Gardner. Mathematical games. *Scientific American*, 223:120—123, October 1970.

[KMN88] A. D. Kelley, M. C. Malin, and G. M. Nielson. Terrain simulation using a model of stream erosion. In *Proceedings of SIGGRAPH*, pages 263—268. ACM Press, 1988.

[Lan89] N. Lancaster. *The Namib Sand Sea: Dune Forms, Processes and Sediments*. A. A. Balkema, 1989.

[LM93] X. Li and J. M. Moshell. Modeling soil: Realtime dynamic models for soil slippage and manipulation. In *Proceedings of SIGGRAPH*, pages 361—368. ACM Press, 1993.

[MC72] A. Masselot and B. Chopard. Snow transport and deposition. *Contributions from the Institute of Low Temperature Science*, A(24):1—58, 1972.

[MT00] B. Malamud and D. Turcotte. Cellular automata models applied to natural hazards. *Computing in Science and Engineering*, 2(3), May/June 2000.

[Pea73] Gregory L. Pearson. Snowpack and related trends in mountains of the southwestern United States. In *Advanced Concepts and Techniques in the Study of Snow and Ice Resources*, pages 22—29. National Academy of Sciences, 1973.

[PF00] Pauline and Philippe De Flers. *Egypt: Civilization In The Sands*, chapter 3, pages 144—155. Konemann Publishing, 2000.

[RY96] R. R. Rogers and M. K. Yau. A Short Course in Cloud Physics. Butterworth-Heinemann, 1996.

[Sha00] J. Shankel. Fractal terrain generation – midpoint displacement. *Game Programming Gems*, 4.18, pages 503—507. Charles River Media, 2000.

[Twe39] W. H. Twenhofel. *Principles of Sedimentation*, pages 185—195. McGraw-Hill, 1939.

[vN51] John von Neumann. The general and logical theory of automata, 1951.

[vN52] John von Neumann. The theory of automata: Construction, reproduction, and homogeneity, 1952.

[WH91] J. Wejchert and D. Haumann. Animation aerodynamics. *Computer Graphics*, 25(4):19—22, July 1991.

[Wol84] S. Wolfram. Universality and complexity in cellular automata, 1984.

[Wol02] S. Wolfram. A New Kind of Science. Wolfram Media, 2002.

[YUM86] L. Yaeger, C. Upson, and R. Meyers. Combining physical and visual simulation – creation of the planet Jupiter for the film 2010. *Computer Graphics*, 20(4):85—93, July 1986.

A Appendices

A.1 Software

The attached CD-ROM contains snapshots of the code produced for this project, along with some sample images and animations that resulted. Code was built for GCC 3.2 running on the x86 platform, using the SDL and OpenGL APIs. Additionally, the code requires Elements, a custom image library (also included).