Eurographics Workshop on Natural Phenomena (2005)
E. Galin, P. Poulin (Editors)

A Preliminary Approach of 3D Simulation of Soil Surface Degradation by Rainfall

Gilles Valette[1], Michel Herbin[1], Laurent Lucas[1†] and Joël Léonard[2]

[1] CReSTIC / LERI / MADS, Université de Reims Champagne Ardenne, Reims, France
[2] Unité d'Agronomie INRA de Laon-Reims-Mons, France

Abstract

Soil surface structure and morphology deeply influence a lot of processes of high agronomic and environmental relevance, such as mass and heat transfer through the soil-atmosphere interface, runoff and erosion, seed germination and seedling emergence. The soil surface structure of agricultural field is in continuous evolution: it is strongly affected by tillage, and in between tillage operations, erosion by rainfall and runoff causes a progressive degradation of the structure whose intensity and speed partly depend on the initial state associated to tillage modalities. A soil surface degradation model could allow to predict this evolution of the soil surface structure, and even to help choosing adequate tillage practices and sowing dates. Erosion modelling has been addressed by soil scientists but also by computer graphic scientists in order to add realism to virtual landscapes. Mixing both of these points of view would be interesting to simulate and visualize the evolution of the soil surface of a cultivated soil. In this paper, we present our project of a simulator of soil surface degradation by rainfall at a small spatial scale (1 m^2 or less), including visualization, and which is mainly based on a 3D cellular automata approach with a specific type of cell. The choices made for the implementation of our model are discussed in the light of the results found in the literature with different modelling approaches.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Physically based modeling

1. Introduction

Soil surface structure and morphology deeply influence a lot of processes of high agronomic and environmental relevance, such as mass and heat transfer through the soil-atmosphere interface, runoff and erosion, seed germination and seedling emergence. The soil surface of arable soils is in constant evolution both because of farming operations and climate. Major aspects of this evolution are the formation of soil crusts and the development of cracks: presence of aggregates and crust due to rainfalls can mechanically inhibit seedling emergence whereas presence of cracks can allow seedlings to break through the soil surface. A model predicting soil structure under different initial soil conditions and climatic scenarios would be an efficient way to select adequate tillage and sowing practices. A portion of land of metric size seems to be specially adequate for such a study. This scale allows clods and cracks observation and a precise study of the solid particles transfer associated with rainfall and runoff.

There has been many studies of erosion, at scales ranging from small plots to entire catchments, under various climatic conditions,by soil scientists. Often, the main objective of the developed models is to allow soil loss evaluation rather than to analyse the evolution of the soil surface and its relief. Besides, during the last two decades considerable progress has been made towards developing efficient models for generating approximations to natural terrain. Although this kind of models generally focused on large scale relief evolution, they are potentially useful and should not be neglected. A popular type of model simulates erosion of the entire landscape using transport equations for the removal of solid material. We think that it can be profitable to bring together

† Correspondence to : laurent.lucas@univ-reims.fr, Rue des Crayères, B.P. 1035, 51687 REIMS Cedex 2.

both of these fields of research by blending the objectives of a pedological simulation and those of a realistic visualization, i.e. physical accuracy and visually plausible results.

The objective of our work is therefore to develop and validate a dynamic simulation of soil erosion on a metric scale, keeping in mind a constant care for visualization. This is a long term and ambitious project and we present here its early beginning. To ensure the correction of our model, we must keep in mind a preoccupation with a validation by the images and numerical data that simulation will provide. For that purpose collaborations with experts in this field of research are essential and we do cooperate with an INRA laboratory (Agronomy unit Laon-Reims-Mons, France).

This paper is organized as follows: the next section 2 describes the erosion processes; section 3 presents a brief overview of related work; section 4 deals with the main ideas for our work; section 5 shows our first results and the last section 6 concludes the paper.

2. Description of the processes involved in soil erosion and surface crusting

In our literature review we have encountered three main kinds of soil erosion: wind erosion, thermic erosion and hydraulic erosion. Although thermal weathering, due to thermal shocks and gravity, influences the topological aspect of the soil and has been modeled for several visual simulations [MKM89, Nag97, BF01a, BF01b], we will concentrate on hydraulic erosion which is the main process responsible for the evolution of the soil surface of agricultural soil. Hydraulic erosion can be caused by rainfall or running water and affects the aspect of a landscape. It can be divided into five processes: detachment by raindrops (splash effect), detachment by runoff, transport by raindrops, transport by runoff and deposit by runoff (figure 1).

Rainfall and runoff cause structural reorganization by the formation of crusts. Crusts are thin soil surface layers more compact and hard, when dry, than the material directly beneath. Generally, two main types of crust are distinguished by their mode of formation: structural crusts and depositional crusts. A structural crust develops in situ and is the result of gradual coalescing of aggregates caused both by particle translocation and by raindrop compaction, whereas a sedimentary crust is formed by deposition of the particles suspended in overland flow [BB90]. Crusts hamper seedling emergence, reduce infiltration and favour runoff and erosion. An example of formation of both types of crust is shown figure 2.

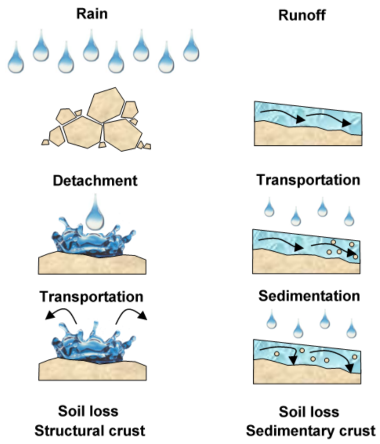
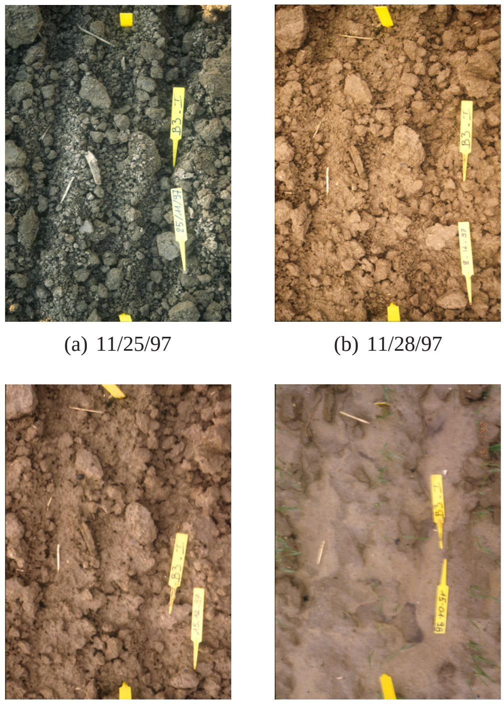


Figure 1: *The different effects of rain and runoff.*



(a) 11/25/97 (b) 11/28/97 (c) 12/15/97 (d) 01/15/98

Figure 2: *Formation of crusts in the field (photos C. Dürr, INRA)*

3. Related works

3.1. Computer graphics models

Musgrave, Kolb and Mace [MKM89] demonstrate a new method for creating mountain fractal terrains with the use of height fields. They suggest two erosion algorithms which simulate hydraulic erosion by flowing water and thermal weathering which chips away steep inclines and forms talus slopes. The hydraulic erosion algorithm supposes that the running water is dissolving some material and moving it to different location. This material is then deposited. Material at every point has some parameters classifying these abilities. The water flows in the direction of the highest gradient. This algorithm ignores evaporation and infiltration and needs some constants which are not well defined. For thermal weathering the material is dissolved because of changes in temperature. Due to the thermal shocks the small parts of the terrain are break-up and fall down. Depending on the consistence of the material, this process is faster or slower. The eroded part falls down in the direction of the greatest gradient as in the previous case. The thermal weathering algorithm is quite simple and efficient, giving realistic results.

Kelley, Malin and Nielson [KMN88] produce images of realistic-looking terrain with an algorithm consisting of two distinct steps. The first steps represent the creation of a drainage network of rivers according to geologically known data and behavior. In the second step the terrain is modeled as a set of surfaces under tension fitting previously generated data.

Beneš and Forsbach [BF01a] introduce new data structure for visual simulation of 3D terrains: the representation is based on horizontal stratified layers consisting of one material. They run the previous algorithm simulating thermal erosion on this representation. In [BF02] they divide the hydraulic erosion process into four independent steps that can be applied independently to achieve high level of realism: new water appears, water erodes the underlying terrain and captures the material, water and the suspended material are transported and finally, because of the evaporation, water deposits the material at another location. The runoff, evaporation and deposit simulation gives convincing pictures, but the step of material capture is oversimplified. Besides, these papers lack visual results of hydraulic erosion.

In order to create some realistic landforms, Roudier et al. [RPP93] simulate geologically contrasted terrains and apply deterministic erosion processes to them. The erosion on any point of the landsurface is therefore related to local geological parameters. Any height field may be chosen as an initial topographic surface. A 3D model defines the geological parameters of each point according to its elevation. The method is iterative: at each step, rock removal and possible alluvial deposition are computed at each point of the landsurface. The available erosion laws simulate mechanical erosion, chemical dissolution and alluvial deposition. This model allows the authors to create some authentic landforms for mountainous regions with geological outlines and stream network. Nagashima [Nag97] creates eroded valley and mountain terrains with layer traces by using geological parameters and simulating the physical erosion of river flow, precipitation, and thermal weathering.

Marak et al. [MBS97] try to formalize some of the algorithms used for simulation of erosion under one algorithm that is based on rewriting matrices. A height field representing the terrain is considered as the matrix, and context sensitive rewriting of this matrix represents the erosion process. The authors define classes of matrices that are used for different erosion algorithms and demonstrate their ability to simulate existing techniques.

Chiba et al. [CMF98] present a simple "quasi physically based" method for simulating the topography of eroded mountains based on velocity fields of water flow. In this method, infiltration and evaporation are ignored, the velocity fields of water flowing down the face of a mountain are calculated by simulation of the motion of "water particles", and these velocity fields are used for simulating erosion. Visual results are impressive and some present various types of layers corresponding to different geologic strata.

3.2. Soil erosion models

In the field of soil science, prediction of soil erosion generally means prediction of soil loss. Current models for soil erosion by water such as WEPP (Water Erosion Prediction Project) and EUROSEM (European Soil Erosion Model) consider an eroding hillslope to be conceptually partitioned into rill and interrill areas. Soil loss in rill and interrill areas is calculated separately and summed to yield a prediction of total soil erosion. Favis-Mortlock et al. [FMBPL00] have developed the "RillGrow" model which applies simple rules at a millimetre scale, to govern the iterative interaction between microtopography, runoff routing and soil loss. In this model emphasis is put on the evolution of the soil surface rather than on sediment exportation. No distinction is made between interrill and rill areas. Soil is decomposed in a regular grid of cells and runoff is considered to be discretised into packets which move from cell to cell. Splash erosion, deposition and infiltration are not modelled. Patterns of erosive flow generated by RillGrow compared well with observations.

As soil erosion can be considered as a system evolving exclusively by means of local interactions, D'Ambrosio et al. [DGGG01, DGI03] as Avolio et al. [ACD*03] suggest a Cellular Automata (CA) model for soil erosion by water. This model involves a larger number of states, including altitude, water depth, total head, vegetation density, infiltration, erosion, sediment transport and deposition. The first applications to the small catchment of the Fiumara Armaconi, Calabria, Southern Italy, gave encouraging results, reproducing the pattern of river network and evaluating convincing erosion values for an intense rainfall event. A similar CA approach, based on the model of "*precipitons*" introduced by Chase [Cha92], is used by Luo et al [LDP*03] for the WILSIM project. In order to simulate landscape erosion and deposition Haff [Haf01] uses a cellular automaton model called "*waterbot*". Waterbots are discrete units of runoff that are moving autonomously and asynchronously across the landscape and are able to pick up and deposit sediment.

Servat [SLPT99] suggests a description of flows in terms of heterogeneous agents that interact in a continuous space

	Evaporation	Infiltration	Detachment by runoff	Detachment by splash	Transport by runoff	Transport by splash	Deposition	Crusting
Musgrave et al.			✓		✓		✓	
Beneš et al.	✓		?	?	✓		✓	
D'Ambrosio et al.		✓	✓		✓		✓	
Servat		✓	✓	✓	✓	✓	✓	✓
Roudier et al.		✓	✓		✓		✓	
Marak et al.			?	?	?	?	?	
Chiba et al.			✓		✓		✓	

Table 1: *Comparison of some computer models of erosion in relation to processes we have to simulate: evaporation, splash and crusting are quasi absent. "?" means that it is not said if the model simulates the process.*

and whose various laws of interaction enable to account for the coupling of concurrent processes. This research has led to the development of the RIVAGE simulator of runoff and infiltration, with some additional trials to incorporate erosion processes. RIVAGE means *Ruissellement et Infiltration Vus par des AGEnts* (runoff and infiltration seen by agents). Numerical results obtained with this model, for the simulation of runoff, compare well to results obtained using a finite-difference solution to the classical Saint-Venant equations. However, the particle approach becomes more and more resources consuming as the number of particles increases: when the rainfall intensity is high, when a lot of water is trapped in depressions, when soil particles are introduced to deal with erosion.

3.3. Comparison

Table 1 shows a comparison between different hydraulic erosion models. The most complete is Servat's multi-agent model. However their approach does not seem to be adapted to our objective because it is centered on the water particles and their mobility and not on the soil. It is worth noting that evaporation, splash and crusting are quasi absent of these simulations. This can be easily explained by the fact that these simulations generally aim to create landscapes or to study soil erosion on a large scale for which these processes can be neglected. Techniques based on a cellular automata model seem to be of great interest in order to obtain accurate results in simulation of erosion and they are largely used in the field of pedology.

4. Our approach

4.1. Soil model

The usual techniques for simulating erosion work with one kind of data structures: height fields. Height fields simulate surface erosion only but algorithms running on this representation are usually faster. For our project we need a volumic model because we must take into account the structure of the subsurface and its evolution and therefore we discretize the space in small elementary cubes (2 mm side) which are our basic cells.

The soil system is composed by three different kinds of components: soil, liquid and gas. We use a system where these three phases are additive. Such a system is provided by the use of specific volumes. We can write the total volume of a soil sample as:

$$V_{cell} = V_s + V_l + V_g$$

where V_s is the volume of soil, V_l is the volume of liquid and V_g is the volume of gas. As our 3D-grid is regular, the height of each component in a cell can be considered equivalent to its corresponding volume, so we can do calculations on heights:

$$H_{cell} = H_s + H_l + H_g$$

where H_s is the height of soil, H_l is the height of liquid and H_g is the height of gas. As H_{cell} is a fixed constant (2mm), H_g is easily computed with the knowledge of H_s et H_l.

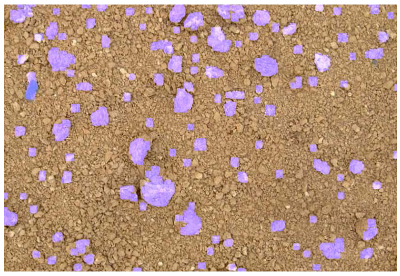
During rainfall, the size of the fragments produced by the desaggregation processes varies with the initial size distribution of particles, the hydric state of the soil, the intensity of the stresses exerted by raindrops. In addition, sedimentary crusts are defined by the fact that particles are sorted according to their size during sedimentation. The size of soil fragments determines if runoff will be able to transport them or not. It is thus very important to have and to keep some information about particle size. This is one originality of this simulator. So, in addition to these different heights we keep in a cell the numbers of particles of certain given sizes, the total volume of which is now given by:

$$V_{cell} = A_{cell}.(H_s + H_l + H_g) + \sum_{i=1}^{NP} N_i.V_i$$

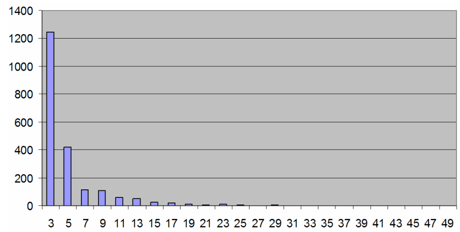
where A_{cell} is the area of a cell (i.e. 4mm^2), NP the number of different particle sizes, V_i the volume of a particle of size S_i, N_i the number of particles of that size. We don't need to keep the specific value H_g because it can be obtained by a simple operation, even if it's less immediate than previously. In the current implementation we use 5 classes of cells corresponding to sizes of 1000, 500, 250, 100 and 50 μm which are common sizes for sieve analysis. One cell state represents 64 bits, and typically a volume size is 500x500x100mm, which means, if cells are 2x2x2mm, about 24 Mo memory occupation for one volume.

4.2. Soil initialization

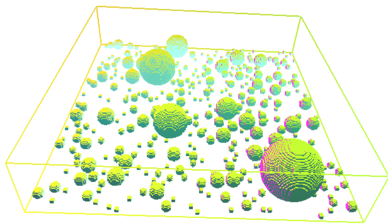
One first important step of our work is to generate an initial volume of soil whose properties reflect the state of the real soil we want to study. For example, we want this initial soil to have the correct topography, but also a correct size distribution of aggregates because this size distribution will allow to make a direct link between the state of the soil and soil tillage, seedbed preparation and sowing techniques. Therefore we need to generate a 3D virtual terrain with realistic characteristics and in particular to take into account the differences in aggregates shape, size and spatial organization. Different types of input data are available: digital elevation maps (by laser profile metering), photographs, statistical data concerning aggregates and physical measures. A digital elevation map is very easy to use and gives a very realistic virtual terrain, but lacks a lot of information, especially concerning the aggregates. To overcome this problem the digital elevation map can be completed with photographs of the same terrain. In this case, image processing can be used to obtain information about aggregate number and size. For example figures 3(a) and 3(b) show a result obtained by granulometry [Ser88] and figure 3(c) shows a corresponding virtual distribution of aggregates (represented here by spheres), with a technique based on the method of Bertuzzi et al [BGSC*95]. One of our objectives in soil generation is to succeed in mixing a digital elevation map and results of granulometry. A possible source for generating initial soil using variables characterizing soil structure is the seedbed generator included in the program SIMPLE developed by our partners of INRA [DAR*01] which creates three-dimensional seedbeds using the input variables describing the aggregate shape, number and spatial organization. Finally we need to feed our model with data concerning moisture and porosity which can be obtained from field measurements.



(a) Detection of aggregates by granulometry on a photograph. The blue colour shows the aggregates detected by this method: their size is equal or larger than the size of the chosen structuring element.



(b) Histogram of aggregate sizes deducted from granulometry of figure (a). Because of image resolution, aggregates smaller than 3mm cannot be counted.



(c) Virtual distribution of aggregates, statistically equivalent to histogram represented in figure (b) and therefore to the terrain represented in figure (a).

Figure 3: *Example of a soil generation with granulometry (images A.Criquy).*

4.3. Erosion model

4.3.1. Cellular Automata

A complex phenomenon such hydraulic erosion could be modeled by a reductionist modeling approach. This approach aims to identify the governing mechanical processes and represents them by systems of differential equations, which cannot be easily solved without making substantial simplifications. Wolfram [Wol94] showed that Cellular Automata (CA) with simple underlying rules can produce highly complex behavior and can mimic many features of what we see in nature. The principle of CA is not to describe a complex system with complex equations, but to let the complexity emerge by interaction of simple individuals following simple rules.

CA are based on a regular division of space in cells, each one embedding identical finite automata, the input of which is given by the states of the neighbouring cells. At the time $t = 0$, cells are in arbitrary states (initial soil) and the CA evolves changing the state of all cells simultaneously at discrete times, according to the transition function. On the one hand, our soil model (section 4.1) gives an explicit regular division of space in cubic cells, and on the other hand, soil erosion can be considered as a system evolving exclusively by means of local interactions. For both these reasons CA could apply successfully and that is our basic approach. Furthermore, CA models have been employed in the field of computer graphics in order to simulate stone weathering [DEJ*99], corrosion [MDG01] or crack patterns [GC01].

4.3.2. Extensions

Nevertheless, classical CA present certain limits in simulating complex macroscopic phenomena. In order to overcome these limits and to simulate lavas flows, debris/mud flows and contamination of soils, Di Gregorio and Serra [DGS99] developed an empirical method based on CA. In order to deal with specific aspects of the processes we want to simulate, we use some of the extensions they have introduced:

- Classical CA are based upon elementary automata with few states and a simple transition function. We allow a large number of different states and a more complicated transition.
- The state of a cell is decomposed into several substates.
- Transition function is decomposed into elementary subfunctions which can be internal transformations as well as interactions with the neighbourhood.

Avolio et al. [ACD*03] complete this model for surface flows modelling:

- Di Gregorio and Serra have used substates of type "outflows" to describe a movement of a certain amount of fluid from a cell towards another cell. In order to avoid time discrepancy (outflows shift is effective only on the successive step), Avolio et al. extend the definition of elementary process: they don't use "outflow" substate and compute in the same step the new value of substate "fluid" in a certain cell by subtraction of the outflow and the new value of this substate in another cell by addition of the same outflow.
- They allow input from the external world to certain cells of the CA, in order to describe an external influence which cannot be represented by local rules. For our simulation these external influences are raindrops and particles projected by splash.

4.3.3. Formalization

As Avolio et al. we add some formalism to our CA model and define it by the set (C, Q, Γ, h, F, σ, P, Φ):

- $C \subset \mathbb{Z}^3$ is the 3D matrix representing the locations of the cubic cells in a discrete space.
- $Q = (Q_1, \ldots, Q_p)$ is a p-tuple of sets of substates, each substate corresponding to a characteristic of a cell, such as moisture or number of particles. Inside a cell a substate value is considered as constant. One important substate defines if a cell is on the surface or not, i.e. undergoes to the influences of the external world and receives raindrops or soil particles transported by splash. For each substate, permitted values must form a finite set. As they represent mostly physical quantities, they are continuous variable and therefore they have to be discretized with a sufficient number of digits.
- the application $h : \mathbb{N} \times C \to Q$ gives the state of C at the instant t; $h(t,c)$ is noted $h_t(c)$; h_0 defines the initial state.
- F is not one transition function, but a matrix $k \times m$ of transition functions (m is the number of processes, k is the maximal number of available simulating functions for one process); in order to be able to inhibit some processes we have $\forall i \; F_{1,i} = Id$.
- Γ is also a matrix $k \times m$ of neighbourhoods $\Gamma_{j,i}$ corresponding to $F_{j,i}$ with $\forall i \; \Gamma_{1,i} = (0,0,0)$ (this neighbourhood also defines internal transformations).
- σ : one of the originalities of our model is to permit to choose between different simulating function for each process; σ is an application such as $\sigma(i) = j$ if and only if we choose to use the function j for the process i.
- P is the finite set of the global parameters which have an effect upon the transition functions. Fundamental global parameters are the cell size and the time step.
- Φ is the function (or set of functions) which computes the external influences on the state of the cells.

We can now write the system evolution between t and $t+1$:

$$\forall c \in C \;\; h_{t+1}(c) = \sum_{i=1}^{m} F_{\sigma(i),i}\left(h_t\left(\Gamma_{\sigma(i),i}(c)\right)\right)$$

$\sum$ is not a sum here but means the succession from 1 to m (the number of processes) of the transition functions.

4.4. Simulated processes

Since hydraulic erosion is a very complex physical phenomenon, it seems essential to divide it into elementary (interacting) phenomena which will be simpler to simulate: soil desaggregation by raindrop impact and slaking, transport by splash and runoff and sediments deposition. Infiltration through the soil must be taken into account, with a strong coupling of infiltrability and the local properties of the developing crust (type of crust, porosity, thickness for example).

4.4.1. Rain generation

A rainfall event can be defined by a hyetogram which represents rain intensity variations within a given duration. We generate several successive rainfall events. The size of raindrops can vary, and raindrops are distributed randomly in space according to an uniform distribution.

4.4.2. Detachment

We consider two distinct mechanisms for soil detachment [BB90]. The first one is disaggregation by entrapped air compression when fast moistening of a dry aggregate occurs. The dryer the soil, the more intense this process. The detached mass is given at a first approximation by the formula: $dm = \alpha\, dh$, with dh the unit of rain height and α an index of soil sensibility, function of its moisture and composition. The second one is the disaggregation without bursting, that is the one due to the impact of raindrops which is able to detach particles from the mass. The detached mass is given again at a first approximation by the formula: $dm = \beta\, ke\, dh$, with ke the kinetic energy of the raindrop and β a sensibility factor, function of its wetness and compactness.

As rain kinetic energy is not a commonly measured meteorological parameter, empirical laws relating the rain kinetic energy to the more easily available rain intensity have been proposed [SPST02]. We use the empirical linear-log formulation of Wischmeier and Smith: $ke = a + b\log(I)$ where $a = 11.9$ and $b = 8.7$ for $I \leq 76$ mm/h and $ke = 28.3$ for $I > 76$ mm/h [RFW*97]. With the assumptions that raindrops size is never greater than the cell size and that the products of detachment are particles smaller than a cell, disaggregation is considered as a local transformation. In our implementation, in each cell which is touched by a raindrop particles are created, their number and size depending on the intensity of the disaggregation processes.

4.4.3. Transport by splash

Soil transport by rainsplash has been measured using a variety of approaches, including splash cups, trays, and boards. The most frequent assumption in the literature is that the spatial distribution of particles splashed from a point source can be described by a negative exponential function [vDMB02, LLBI04]. The intensity of raindrops impact, the size of particles and the local topography determine projection distance. To estimate the size of the particles ejected, we use a fuzzy logic approach based on simple rules which take the kinetic energy of the rain and the size of the raindrops as inputs. The knowledge of the flux of matter and the size of particles then gives the number of particles ejected from the cell where the raindrop falls. The distance is randomly chosen in a greater interval for smaller particles, with an exponential distribution, and the point of destination is accepted if it is lower than the source point.

4.4.4. Infiltration and runoff

Proving the versatility of our simulator the implementations of infiltration and runoff are very different. Infiltration is modelized with the Green-Ampt method which makes the following assumptions: wetting front advances at constant rate, volumetric water contents remain constant above and below the wetting front as it moves and soil water suction below the wetting front remains constant. It needs four parameters: the saturated hydraulic conductivity $\mathcal{K}_s$, the saturated and initial volumetric water contents θ_s and θ_i, and the soil water suction at the wetting front Ψ. The potential infiltration rate is given by:

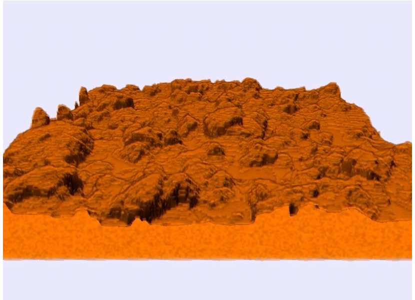
$$f = \mathcal{K}_s \left(1 + \frac{\mathcal{H}_s + \Psi}{\mathcal{L}}\right) \quad \text{with} \quad \mathcal{L} = \frac{\mathcal{H}_i}{\theta_s - \theta_i}$$

with $\mathcal{H}_s$ the local height of water and $\mathcal{H}_i$ the height of infiltrated water. On the contrary we don't use any physically-based equation for runoff but we simply calculate the report of water between the surface cells, using the fact that water flows in the direction of the highest gradient.

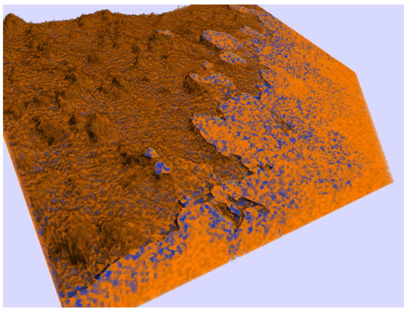
4.5. Visualization

Simulation produces data volume(s) which can be processed to give numerical results (e.g. mass of displaced soil) but also images or animations. For that result we choose to keep the structure of the 3D grid and therefore we use volume visualization and precisely direct volume rendering (DVR). DVR is a powerful technique for visualizing the structure of volume datasets. The most attractive aspect of DVR is actually its defining characteristic: it maps directly from the dataset to a rendered image without any intermediate geometric calculations. This is in contrast to traditional isosurface rendering, where the rendering step is proceeded by the calculation of an isosurface for a particular data value. DVR can also avoid the time-consuming processes of segmentation and classification, because visualization can be done without any high-level information about the material content at each voxel. The basic benefit of DVR over isosurface rendering is that it provides much greater flexibility in determining how the voxels contribute to the final image. Voxels over a range of values can all contribute to the image, with varying amounts of importance, depending on the transfer function.

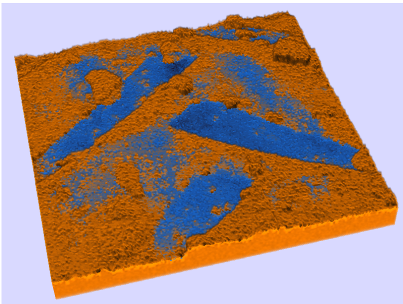
DVR is based on the premise that the data values in the volume are themselves a sufficient basis for creating an informative image. What makes this possible is a mapping from the numbers which comprise the dataset to the optical properties that compose a rendered image, such as opacity and color. This critical role is performed by the transfer function. Because the transfer function is central to the DVR process, picking a transfer function appropriate for the dataset is essential to creating an informative, high-quality rendering. We need to transform every cell into a voxel, that means to reduce the amount of data contained in a cell to an appropriate value. This value will be either a byte (with choice of a global color table) or a 24 bits RGB value. This



(a) Surface and matter.



(b) Matter and moisture (here two clipping planes are used to cut through the volume).



(c) Surface, matter and water (this is a terrain with an artificially created tyre print).

Figure 4: *Three types of visualization.*

method gives the opportunity of showing different aspects of the same data volume (e.g. moisture, particles of certain size) by retaining only certain data for the cell to voxel transformation as shown in figure 4. The drawback is the difficulty to find an accurate transfer function which can give realistic images of the soil.

The algorithm we used, described by Benassarou et al. [BBJL05], is based on an accelerated slicing algorithm for interactive volume rendering of structured grids. It requires a small amount of memory and provides adaptive rendering for improved image accuracy as well as progressive rendering for rapid feedback at interaction time. The slicing methods use the 3D texture mapping hardware: the volume data (our voxels computed from the cells) is first converted to a 3D texture. The texture coordinates in parametric object space are assigned to each vertex of the clipped polygon. During rasterization, fragments in the slice are trilinearly interpolated from 3D texture and projected onto the image plane using adequate blending operations.

5. First results

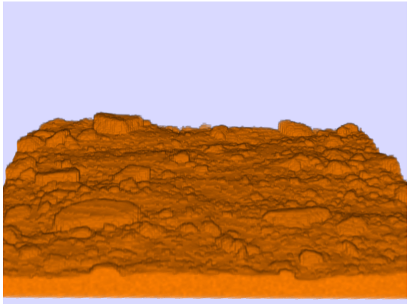
As a beginning, we have tested our model on the simulation of the splash phenomenon which appears during a rainfall and which is at the origin of the first stage of soil crusting. In our implementation we generate a rainfall and then we decompose the splash into two main processes: the detachment or the dislodgement of the particles from the surface mass and the ejection of detached fragments.

For the moment our results are only qualitative and visual. The resulting images show that the simple rules simulating the splash on our soil model give plausible results (figure 5) comparable with those of a rain-simulator experiment - even if our simulated erosion is acting too quickly (it's mostly due to the fact the we don't consider for the moment protection against splash which is constituted by surface water). This is encouraging, but the transition rules were very crude in our tries of simulation. We have to improve the existing rules and develop other rules on the basis of the knowledge of the physical processes at work.

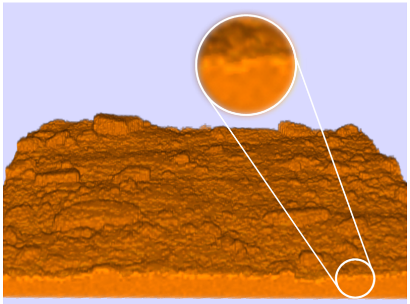
6. Conclusion

We are developing a simulator of the evolution of soil surface structure of cultivated soils under rainfall, at the metric scale. This is an important issue as it has both theoretical and practical interest. Knowledge has been accumulated by soil scientists about the various physical processes controlling the degradation and crusting of soil surfaces submitted to rainfall. However, there exist few tools that combine these processes to allow the simulation of the evolution of the structure of the soil surface. Thus, our simulator will first have a role in the aggregation of the different pieces of existing - and future - knowledge. This integration of the different processes in a model will also offer the possibility of an alternative validation of the description of the processes, by comparing simulated and observed soil surface evolution. In addition, as soil surface degradation and soil crusting control key variables such as hydraulic conductivity and mechanical resistance of the soil surface, we expect our simulator to have an important applied interest, to establish links between the state of the soil surface associated to tillage, sowing operations, and important processes such as runoff and erosion or seedling emergence.

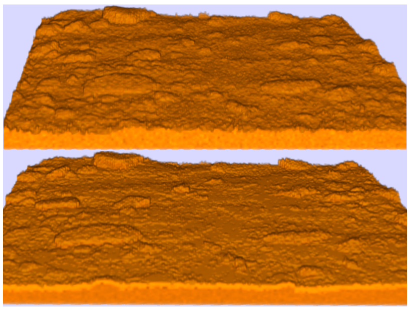
Our project is still in a very early stage, and some key issues have to be addressed, such as how to define in the context of the model the different types of crusts observed in the field. But the first results obtained on disaggregation and



(a) The initial soil created from a digital elevation map.



(b) Result of our splash simulation after 600 iterations (corresponding to 10mn), crusts formed by particles are visible because we have chosen a transfer function which shows where the particles are concentrated.



(c) Real terrain after 80mn of a 30mm/h rain in a rain-simulator and below our virtual simulation after 40mn with a 30mm/h rain.

Figure 5: *Visual comparison between a result of the simulation of splash and a result of real erosion.*

transport by splash, despite the crude assumptions made in the description of the processes involved, are encouraging.

References

[ACD*03] Avolio M. V., Crisci G. M., D'Ambrosio D., Di Gregorio S., Iovine G., Rongo R., Spataro W.: An extended notion of Cellular Automata for surface flows modelling. *WSEAS Transactions on Computers 2* (2003), 1080–1085.

[BB90] Bresson L. M., Boiffin J.: Morphological characterization of soil crust development stages on an experimental field. *Geoderma 47* (1990), 301–325.

[BBJL05] Benassarou A., Bittar E., John N., Lucas L.: MC Slicing for Volume Rendering Applications. In *Computer graphics and geometric modeling* (Atlanta, GA, USA, may 2005), Springer Verlag, pp. 314–321.

[BF01a] Beneš B., Forsbach R.: Layered Data Representation for Visual Simulation of Terrain Erosion. In *EEE Proceedings of SCCG volume 25(4)* (2001), pp. 80–86.

[BF01b] Beneš B., Forsbach R.: Parallel implementation of terrain erosion applied to the surface of Mars. In *Proceedings of the 1st international conference on Computer graphics, virtual reality and visualisation* (2001), ACM Press, pp. 53–57.

[BF02] Beneš B., Forsbach R.: Visual Simulation of Hydraulic Erosion. *Journal of Winter School of Computer Graphics* (2002).

[BGSC*95] Bertuzzi P., Garcia-Sanchez L., Chadoeuf J., Guerif J., Goulard M., Monestiez P.: Modeling Surface Roughness by a Boolean Approach. *European Journal of Soil Science 46* (1995), 215–220.

[Cha92] Chase C. G.: Fluvial landsculpting and the fractal dimension of topography. *Geomorphology 5* (1992), 39–57.

[CMF98] Chiba N., Muraoka K., Fujita K.: An Erosion Model Based On Velocity Fields For The Visual Simulation Of Mountain Scenery. In *The Journal of Visualization and Computer Animation* (1998), vol. 9, pp. 185–194.

[DAR*01] Dürr C., Aubertot J.-N., Richard G., Dubrulle P., Duval Y., Boiffin J.: SIMPLE: a Model for SIMulation of PLant Emergence Predicting the Effects of Soil Tillage and Sowing Operations. *Soil Science Society of America Journal 65* (2001), 414–442.

[DEJ*99] Dorsey J., Edelman A., Jensen H. W., Legakis J., Pedersen H. K.: Modeling and rendering of weathered stone. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1999), ACM Press/Addison-Wesley Publishing Co., pp. 225–234.

[DGGG01] D'Ambrosio D., Gregorio S. D., Gabriele S., Gaudio R.: A Cellular Automata Model for Soil Erosion by Water. *Physics and Chemistry of the Earth, Part B 26*, 1 (2001), 33–40.

[DGI03] D'Ambrosio D., Gregorio S. D., Iovine G.: Simulating debris flows through a hexagonal Cellular Automata model: Sciddica S3-hex. *Natural Hazards and Earth System Sciences 3* (2003), 545–559.

[DGS99] Di Gregorio S., Serra R.: An empirical method for modelling and simulating some complex macroscopic phenomena by cellular automata. *Future Generation Computer Systems 16* (1999), 259–271.

[FMBPL00] Favis-Mortlock D., Boardman J., Parsons A., Lascelles B.: Emergence and erosion: a model for rill initiation and development. *Hydrological Processes 14*, 11-12 (2000), 2173–2205.

[GC01] Gobron S., Chiba N.: Crack Pattern Simulation Based on 3D Surface Cellular Automata. *The Visual Computer 17* (June 2001), 287–309.

[Haf01] Haff P. K.: *Waterbots In: Landscape Erosion and Evolution Modeling, Harmon R. S. and Doe, W. W. III (Eds)*. Kluwer Academic/Plenum Publishers, New York, 2001.

[KMN88] Kelley A. D., Malin M. C., Nielson G. M.: Terrain simulation using a model of stream erosion. *SIGGRAPH Comput. Graph. 22*, 4 (1988), 263–268.

[LDP*03] Luo W., Duffin K. L., Peronja E., Stravers J. A., Henry G. M.: A Web-based Interactive Landform Simulation Model (WILSIM). *Submitted to Computers and Geosciences* (2003).

[LLBI04] Legout C., Leguédois S., Bissonnais Y. L., Issa O. M.: Splash distance and size distributions for various soils. *Geoderma (in press)* (2004).

[MBS97] Marak I., Beneš B., Slavik P.: Terrain Erosion Based on Rewriting of Matrices. In *Proceedings of The Fifth International Conference in Central Europe on Computer Graphics and Visualization* (1997), pp. 341–351.

[MDG01] Merillou S., Dischler J.-M., Ghazanfarpour D.: Corrosion: simulating and rendering. In *GRIN'01: No description on Graphics interface 2001* (Toronto, Ont., Canada, Canada, 2001), Canadian Information Processing Society, pp. 167–174.

[MKM89] Musgrave F. K., Kolb C. E., Mace R. S.: The synthesis and rendering of eroded fractal terrains. In *Proceedings of the 16th annual conference on Computer graphics and interactive techniques* (1989), ACM Press, pp. 41–50.

[Nag97] Nagashima K.: Computer generation of eroded valley and mountain terrains. *The Visual Computer 13* (1997), 456–464.

[RFW*97] Renard K., Foster G., Weesies G., McCool D., Yoder D.: Predicting Soil Erosion By Water: A Guide to Conservation Planning with the Revised Universal Soil Loss Equation (RUSLE). *U.S. Department of Agriculture Handbook 703* (1997), 404 pp.

[RPP93] Roudier P., Peroche B., Perrin M.: Landscapes Synthesis Achieved through Erosion and Deposition Process Simulation. *Comput. Graph. Forum 12*, 3 (1993), 375–383.

[Ser88] Serra J. (Ed.): *Image Analysis and Mathematical Morphology: Theoretical Advances*, vol. 2. Academic Press, London, England, 1988.

[SLPT99] Servat D., Léonard J., Perrier E., Treuil J.: The rivage-project: a new approach for simulating runoff dynamics. In *Modelling of transport processes in soils at various scales in time and space*, Feyen J., Wiyo K., (Eds.). Wageningen Pers, Wageningen, The Netherlands, 1999, pp. 592–601.

[SPST02] Salles C., Poesen J., Sempere-Torres D.: Kinetic energy of rain and its functional relationship with intensity. *Journal of Hydrology 257* (2002), 256–271.

[vDMB02] van Dijk A. I. J. M., Meesters A. G. C. A., Bruijnzeel L. A.: Exponential distribution theory and the interpretation of splash detachment and transport experiments. *Science Society of America Journal 66* (2002), 1466–1474.

[Wol94] Wolfram S.: *Cellular Automata and Complexity: Collected Papers*. Addison-Wesley, New York, 1994.