Eurographics Workshop on Natural Phenomena (2006)
E. Galin, N. Chiba (Editors)

A Generalized Cracks Simulation on 3D-Meshes

Gilles Valette[1], Stéphanie Prévost [1] and Laurent Lucas[1†]

[1] CReSTIC / LERI / MADS, University of Reims Champagne Ardenne, Reims, France

Figure 1: *Three examples of results obtained by our method.*

Abstract

This article describes a method for simulating the formation and the development of cracks on the surface of a shrinking volume. The simulated cracks are applied afterwards to any surface provided with a parameterization. The 2D path of the cracks is automatically precalculated by an appropriate algorithm which gives a graph of discrete ways. We newly propose to take into account a possibly inhomogeneous thickness of the shrinking layer by using a watershed transformation to compute this path. The propagation of one crack is then based on the respect of the primary orientation of the crack. Another originality of our method is the calculation of the enlargement of each crack by a discrete shrinkage volume propagation. We consider the shrinking layer as a set of cubic cells which contain volumes of matter and pores. During the dessiccation process, the matter shrinks, creating what we call a "shrinkage volume". We propagate this shrinkage volume among the cells up to the cracked ones, and we deduce the width of the cracks from the resulting shrinkage volume in these cells. In this paper, this method is presented in detail and we give images obtained from different simulations. Initially designed to help for the prediction of seedlings emergence in an agronomic environment, the method we present can also be applied to enhance the realism of virtual 3D objects.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Physically based modeling I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism I.6.8 [Simulation and Modeling]: Visual

1. Introduction

Cracking is a very common phenomenon on many different types of surface on several scales. Thus cracks formation is of great interest for scientists from different disciplines. In the computer graphics (CG) field, visually plausible crack patterns can enhance the realism of a natural scene. Our work is focused on cracks caused by the shrinking of a material layer frustrated by its deposition on a non-shrinking substrate (produced, for example, in the glaze of ceramics or in a desiccating soil). Such cracks can be considered as quasi-

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

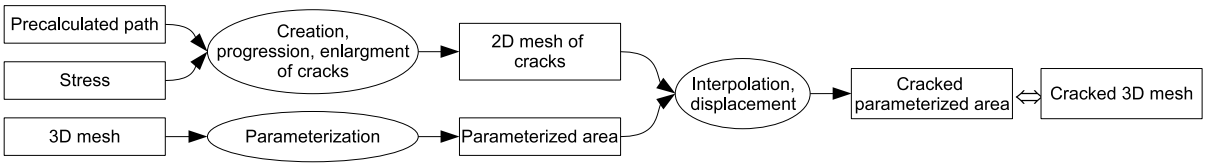


Figure 2: *The principle of our method.*

static, and no external force is needed to generate them. Other works propose methods for creating shattering cracks and breaking objects into fragments [OH99, MGDA04], and very impressive results can also be obtained with a designing stage where the user defines the crack pattern and profile curves [DGA05].

The models of quasi-static cracks in the literature can be roughly subdivided into three main classes. Geometric models propose algorithms to get cracks pattern close to those produced by nature. Perrier *et al.* [PMRdM95] applied a fractal fragmentation algorithms based on Dirichlet tessellation to generate crack patterns at hierarchical scales. MacVeigh [Mac95] created patterns based on line segments and circular arcs, with respect to two characteristics: the cracks intersect at right angles and they are such so as to minimize their lengths. Horgan and Young [HY00] proposed a two-dimensional (2D) stochastic geometric model for 2D crack patterns in clay soil which produces realistic patterns. However, some particularities are often not presented by this type of model, such as varied widths, the three-dimensional (3D) aspect, the presence of dead branches or the dynamics of cracking.

On the contrary, physical approaches propose models which tend to faithfully reproduce this dynamics. Their main drawback is their slowness. Furthermore, in general, their principal aim is not to get visually interesting results. Skjeltorp and Meakin [SM88] used a 2D lattice where the nodes are connected through Hookean springs which break when a critical strain is exceeded. Hirota *et al.* [HTK98, HTK00] also applied a spring-network model to compute 2D and 3D cracks formation in a material. Vogel *et al.* [VHLR05] expressed the water evaporation, which causes the formation of cracks, as a slow contraction of springs. Federl [Fed02] adapted the method of Skjeltorp and Meakin by incorporating growth. He showed that mass-spring models are limited in their capacity to successfully model fracture formation on growing domains. Therefore Federl and Prusinkiewicz [FP04] used another approach to fracture modeling, introduced for modeling inelastic deformation by Terzopoulos and Fleischer [TF88] and based on solid mechanics and finite element models. This approach is more accurate but slower than the spring-network approach.

Gobron and Chiba [GC01] proposed a completely different "intuitive-physical" approach, based on their 3D cellular automaton (CA) model, for simulating realistic propagation of various types of cracks on any triangulated surface. Paquette *et al.* [PPD02] applied this model to flat surfaces in order to reproduce paint cracking and peeling. These methods permit to visualize realistic cracks, but their parameterization seems to be difficult. Moreover, they are not directly linked with physical quantities, like those of a desiccating soil.

In this paper, we propose a method based on a similar CA approach to compute cracks and to apply them on any surface provided with a parameterization. We consider the shrinking layer as a set of cubic cells which contain volumes of matter and pores. During the dessiccation process, the matter shrinks, creating what we call a "shrinkage volume". One of the originalities of our method is the calculation of the enlargement of each crack with a propagation of this volume by summing it from the farthest to the nearest cell (according to the distance to the cracks). In their progression, cracks follow a precalculated path, generated by an appropriate algorithm, with respect to their orientation. We have implemented three different algorithms to compute the path of the cracks. In particular, one of the originalities of our approach is the use of a watershed transformation in order to compute this path, which permits to take into account the inhomogeneous thickness of the shrinking layer. Our method is based on a physical approach, and it produces visually interesting images of 3D cracks, with different characteristics controlled by a set of intuitive parameters. In this paper, we present this method in detail and we give images obtained from different simulations.

2. Approach overview

As is shown in Fig. 2, our method is a two-stage process: in the first stage, we compute a 2D mesh of cracks (a collection of facets in a planar area with an information of depth for each vertex), and in the second stage, we mix these cracks with a 3D mesh by means of a parameterization. In the first stage, we consider that cracks formation depends on two main parameters: a stress function and a precalculated path.

2.1. Cracks formation

2.1.1. The stress function

We define the stress function with the thickness of the shrinking layer of the material and a shrinkage function. The

thickness of the shrinking layer is given in a discrete way by a heightmap, which can be understood as a grey image: white is the highest value, and black the lowest value. Except for the particular case of a constant value, this heightmap can be either randomly generated or based on a particular property of the surface, like the gaussian curvature.

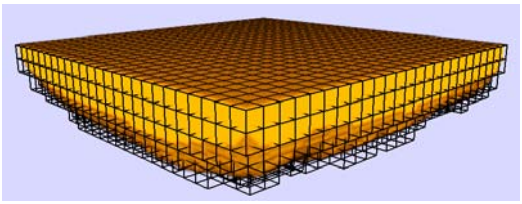


Figure 3: *Representation of the shrinking layer with its elementary cubic cells (a visible cell in this image is actually* $4^3 = 64$ *times bigger than a real one).*

We represent the shrinking layer as a volume of elementary cubic cells (see Fig. 3). We consider that the heightmap used for the shrinking layer thickness has the same resolution. Therefore the pixel (i, j) of this map corresponds to the cells column (i, j) of the layer (and to the particular first cell of this column which is on the surface). Each cell holds two main substates which are the volume of matter and the volume of pores it contains. We can write the total volume of a cell $V_{cell} = V_m + V_p$, where V_m is the volume of matter and V_p is the volume of pores. 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: $H_{cell} = H_m + H_p$, where H_m is the height of matter and H_p the height of pores.

The shrinkage function must give the volume of matter of one cell at an instant t. In our method, we simply consider that we can compute a coefficient of shrinkage C_s which depends on the time and on the relative depth of the cell. For that coefficient we use the following empirical formula:

$$C_s(t,z) = k_0 + (1-k_0)\exp\left(-k_1 \times t^2 \times (1-z_r)\right) \qquad (1)$$

with k_0 the minimum value for this coefficient, k_1 another coefficient permitting the control of the speed of the shrinkage process, t the elapsed time, and $z_r = z/\max(z)$ the relative depth. For this method we have:

$$V_m(t) = V_m(t-1) \times C_s(t,z) \qquad (2)$$

It is worth noticing that any other formula giving the volume of matter could be used and an example is shown in section 3.

2.1.2. The precalculated path

Our method uses a pre-existing 2D path for the cracks. This path is stored as an undirected graph G: the nodes are points of intersection (or one final point of a path), the edges contain the sequence of discrete points connecting two nodes, each point corresponding to a surface cell of the shrinking layer. Any binary image with one pixel wide discrete lines can be used as an input for our method. Nevertheless, in order to get realistic results, we have implemented three methods to compute the cracks path. The first two methods are not new in this context: they are the empirical stochastic model of Horgan and Young and the Dirichlet tessellation. As the third approach, we introduce a novelty in the computing of cracks path by using a watershed model. Examples of these three types of path are given in Fig. 8, 10 and 11.

Horgan and Young [HY00] proposed an empirical stochastic model to define the geometry of 2D cracks, such as will form in a thin layer of soil. We use the results produced by this model in two ways: statically, as a final pattern from which we extract nodes and edges of our graph, or dynamically, by adding to the edges of our graph information concerning the order of creation of the crack which owns this edge. Further explanations of using this information in our method are given in the next section.

Inspired by the work of Perrier *et al.* [PMRdM95], we use a Dirichlet tessellation (also called Voronoï tessellation) to get a space partition of the terrain in polygonal zones (the Voronoï regions, also called in hydrology and geography the Theissen polygons). This method starts from a set S of m points s_i called seeds in the Euclidean plane $\mathbb{R}^2$. These seeds are randomly chosen. The points in the interior of a Voronoï region $V(i)$ are the points of the plane that are closer to the seed point $s_i \in S$ than to any other seed point s_j of S, $(j \neq i)$. The borders B_v of the Voronoï regions are called *skeletons by influence zones* (SKIZ). They contain the points that are equidistant from two seeds, and in our method they form the possible path of the cracks.

The drawback of these techniques is that they do not consider the thickness of the shrinking layer. Making the assumption, based on our own observation of cracked soils, that cracks appear and propagate where the shrinking layer is the thickest, we apply a watershed transform to get the possible cracks path from the corresponding heightmap.

The watershed transform treats the image as a relief: the pixel (i, j) is considered as a point with a vertical coordinate equal to its level of gray. This way, all the pixels define some landscape. An algorithmic definition of the watershed transform by simulated immersion was given by Vincent and Soille [VS91]. For this algorithm, the watershed transform fills up this landscape with water starting at the local minima and, at points where water coming from different basins would meet, builds dams. As a final result, the landscape is partitioned into regions (or "catchment basins") separated by dams, called watershed lines or simply watersheds: these lines represent exactly the path we need for the cracks. It is well known that the watershed method produces an oversegmentation of the image, *i.e.*, many small basins are produced due to many local minima in the input image. According to the scale-space theory for discrete signals, sig-

nificant image features must be stable with respect to variations in scale [Lin94]. Thus, this drawback can be avoided by smoothing the image before applying the watershed algorithm. In our case, that means that we can control the number of possible ways for the cracks by more or less smoothing the shrinking layer thickness map.

It is worth noticing that these three methods do not need any user interaction, except for one possible initial parameterization. The cracks path is automatically generated and no design stage is required, contrary to other methods [DGA05].

2.1.3. Creation and progression of cracks

In this section we describe the creation of cracks and the progression of active cracks.

The potential start points of the cracks are all the middle points of each edge of the graph. To select the points from which a new crack will actually start, we arrange them in order from the highest down to the lowest distance to the existing cracks. On this purpose we maintain a distance map computed from the skeleton of the cracks by an External Euclidean Distance Transformation [ST94]. When a start point is selected, it is removed from the list of potential start points.

Two parameters are used to decide if cracks need to be created: a minimum number of active cracks m, and a maximum number of active cracks M. A crack is considered as "active" if it keeps growing at least at one of its extremities. If the number of active cracks is below m, cracks are created until their number reaches M. If one chooses $m = 0$, the creation proceeds by entire generations. On the contrary, if one chooses $m = M$, the creation proceeds by individuals. Of course, creation is possible only if the list of potential start points is not empty. Creation can be stopped if a predefined number of cracks is reached, or if the cracks exceed a maximal total length.

When a crack is created, it grows simultaneously at each extremity. Each extremity grows by adding the points stored in the current edge up to the distance corresponding to the speed of the crack multiplied by the step duration, and by choosing the next edge when it reaches a node. The principle of this choice is to respect direction φ of the crack, which is determined by the edge containing the start point. A global limit of tolerance α is given, so a crack tries to find an edge whose angle is the closest to φ and in the range $[\varphi - \alpha, \varphi + \alpha]$. It stops because either such an edge does not exist, or the current node is an extremity, or it is attracted by another crack. This attraction occurs if the distance between the current node and a node from another crack is below a threshold. A crack can also be stopped if it arrives at a place where the shrinking layer thickness is below another threshold.

Another criterion is taken into account when the edges have stored an order of creation (as for the Horgan and Young model). In this case, the start points are arranged with respect to this order, prior to their distance to cracks. In the same way, during the progression of a crack, the next edge is the edge whose angle is in the desired range and whose order has the minimal value amongst the available edges.

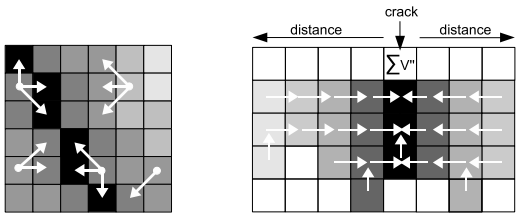
2.1.4. Enlargement of cracks

We assume that there is a loss of volume of matter during the shrinkage process. We call this loss of volume the "shrinkage volume". In one cell (whose volume is constant), this volume must be replaced by some matter: this matter must come from a neighbor cell, which also needs some matter to replace its own shrinkage volume, and so on. We assume that this phenomenon would explain the formation and enlargement of cracks: a cell which is part of a crack has lost all its matter which was distributed to other cells. We conclude that we can deduce the volume of the cracks from a calculation of the shrinkage volume accumulated in the cracked cells. Thus, in our method, the enlargement of cracks consists in three stages: the calculation of the shrinkage, the propagation of the shrinkage volume inside the cells, and finally the calculation of the cracks enlargement.

By means of the shrinkage function, in each cell c, at iteration $t+1$, we calculate a new volume of pores, and we deduce from it the shrinkage volume V_{sh} for this cell:

$$V_{sh}(c,t+1) = V_{sh}(c,t) + V_p(c,t) - V_p(c,t+1) \qquad (3)$$

Notice first that as V_p increases, V_{sh} also increases, and second that $V_{sh}(c,t+1)$ will be modified by the next stage of our method, *i.e.*, the shrinkage volume propagation.



(a) Distance map. (b) Vertical section.

Figure 4: *The principle of the shrinkage volume propagation: (a) on the distance map, volumes are propagated towards the closer cells (b) the volumes are propagated horizontally and, if not possible, vertically, up to a crack.*

The aim of the shrinkage volume propagation is to express the exchange of matter and pores between the cells. This propagation uses the same distance map than the creation of cracks (section 2.1.3) and starts from the cells whose distance is equal to D, initially the maximum of this map. At each iteration, we decrement the distance D in order to treat closer cells, until reaching the cracked cells, which are at distance zero (Fig. 4(a)). If a cell is isolated at its depth, *i.e.*, there is no closer cell of the shrinking layer in its horizontal

neighboring, its shrinkage volume is transmitted vertically (Fig. 4(b)). Otherwise, first we calculate the volume $V'_{sh}(c)$ which will be distributed from the cell to the valid neighbor cells:

$$V'_{sh}(c) = \begin{cases} (1-\frac{1}{r_s}) \times V_{sh}(c) \times (1-\sqrt{\frac{d}{d_{max}}}) & \text{if } d \leq d_{max} \\ 0 & \text{if } d > d_{max} \end{cases} \quad (4)$$

with $V_{sh}(c)$ the shrinkage volume of the cell c, d the distance of the cell to the cracks, d_{max} the maximum distance of volume exchange for one crack, r_s the geometry factor: in order to take care of vertical shrinkage (also called subsidence), we use this "geometry factor" r_s, which equals 3 in case of isotropic shrinkage and equals 1 in case of subsidence only. If vertical shrinkage dominates cracking, we have $1 < r_s < 3$, if cracking dominates vertical shrinkage, we have $r_s > 3$. The distance d_{max} can be fixed or can evolve as the maximum of the distance map. We use this value $V'_{sh}(c)$ to update $V_{sh}(c)$ for the next iteration:

$$V_{sh}(c) = V_{sh}(c) - V'_{sh}(c) \quad (5)$$

Then, we calculate the proportion $V''_{sh}(i)$ of the volume $V'_{sh}(c)$ which will be distributed to a valid neighbor cell i:

$$V''_{sh}(i) = V'_{sh}(c) \times \frac{d-d_i}{\sum_{j \in N(c)}(d-d_j)} \quad (6)$$

where $N(c)$ is the set of valid cells from the horizontal neighboring of c. When all the cells have been visited, the cells which are on the cracks path have received the shrinkage volume they must transform in void, so we base our enlargement calculation on this information. This is the next stage of our method.

A crack is composed of a succession of links between the adjacent surface cells of its path. To avoid unnatural regularity, the extremities of these links are slightly randomly displaced from the center of the cells. The portion of the crack between two adjacent surface cells is considered as a triangular prism, optionally truncated. At the surface, this prism is seen as a rectangle. We deduce from the volume of this prism V_{Pr}, which is deduced from the shrinkage volume, its width W and its depth H:

$$W = \sqrt{\frac{2V_{Pr}}{LR} \times \frac{1}{1-r^2}} \qquad H = RW \times (1-r) \quad (7)$$

where L is the distance between the two extremities, w the width of the base of the crack, R and r two ratios defining the proportions of the crack: $R = H/W$ and $r = w/W$. In the case of a triangular shape, we have $w = 0$ and thus $r = 0$.

In case of flat surface, as our method does not change the relief at all, the visual result can be unnatural. To improve this point, we have implemented a vertical shrinkage.

2.1.5. Vertical shrinkage

This shrinkage uses the same volume shrinkage V_{sh} as for the cracks enlargement, but it does not affect the cracks, only the height of the surface. We mix two types of calculation: the first one involves one column of cells, depending on its distance to the cracks (the farther it is, the more it goes down), the second one involves each region enclosed by cracks, depending on its area (the larger it is, the more it goes down). The vertical volume shrinkage of a cell c is given by:

$$V'_{vsh}(c) = \frac{1}{r_s}\left(k\frac{\sum_{i \in R(c)} V_{sh}(i)}{\max(area)} + (1-k)\frac{d}{D_{max}}V_{sh}(c)\right) \quad (8)$$

where r_s is the geometry factor, $R(c)$ the region of c, $\max(area)$ the maximal area of the regions (in pixels), d the distance of c to a crack, D_{max} the maximum of the distance map, and k a coefficient which determines the contribution of both the column and region calculations. We update $V_{sh}(c)$ as in equation (5). To get the change of the height H_{sh} of one point P of the surface, we divide this volume by the cell area, and sum the result for the column $C(P)$ corresponding to this point:

$$H_{sh}(P) = -\frac{\sum_{c \in C(P)} V'_{vsh}(c)}{A_{cell}} \quad (9)$$

An example of the effect of the vertical shrinkage on an originally flat surface is given in Fig. 8. The cracks on the painting in Fig. 1 were also obtained from a Dirichlet tessellation with both horizontal and vertical shrinkage, and Fig. 8 shows a detail of this painting.

2.1.6. 2D mesh of cracks computation

At this stage of our method, we dispose of a collection of cracks, each one defined by a sequence of points, each point storing the width and the depth of the volume portion of crack starting from this point. In order to mix these data with the parameterized area of a surface, we need to transform them into a triangulated surface. This transformation is straightforward: *(i)* we assemble consecutive portions of a crack as polygons, in order to eliminate voids, with respect to the width of each portion at its beginning; *(ii)* we compute the lines or the polygons corresponding to the bottom of the portions of the cracks; *(iii)* we transform the edges of all the polygons and the lines into vertices, adding a sufficient number of interpolated vertices in order to avoid aliasing; *(iv)* finally, we get facets from all the vertices by making a Delaunay triangulation with the freely available library Triangle [She96]. Notice that step *(i)* modifies the shape of the portions of the crack and consequently, the previous calculation of the volumes does not exactly match the final rendered cracks, but we consider that the difference is negligible.

2.2. Parameterization

In order to apply our cracks to any surface, we need a method which allows operations on a surface to be performed as if it were flat. Parameterization offers this opportunity by mapping a surface onto regions of the plane. Surface parameterization has been studied extensively in the CG field and

many techniques are available. As our cracks are represented in a continuous area, the chosen method of parameterization must unfold the surface rather than be based on charts. For example, Gu and Yau [GY03] propose a global conformal parameterization for surfaces with nontrivial topologies, with or without boundaries. Fig. 9 shows a part of the Stanford Bunny which can be cracked using this method of parameterization.

Simpler methods can be used for less complicated surfaces. For a heightmap (Fig. 6) or a surface of revolution, the method is trivial because there is a natural correspondance between the surface and the unity square. It corresponds to a planar projection which can also be used for certain non planar surfaces like a plate (Fig. 1) or parts of a wall (Fig. 7). In the same way, spherical and cylindrical projections can be used to get a parameterized area of some part of other particular surfaces, like the igea and the moai models respectively (Fig. 11).

The use of a parameterization has a drawback: the quality of the result depends on the quality of the parameterization. For example, in Fig. 7, the rough planar projection inhibits cracks to follow the surface curvature well. However, this method offers two main advantages: first, there is no relation between the resolution of the cells and the surface, thus, it would work well on a highly bumpy or irregular surface; second, it can be applied to any 3D surface with arbitrary genus.

2.3. Interpolation and displacement

Assuming that we dispose of a conformal parameterization of at least a patch of the surface we want to crack, we need to mix the mesh of cracks with the corresponding parameterized area. For that purpose we project all the points of the cracks to this parameterized area. First, we discard points which are off the area boundaries. For the other points, we determine in which triangular facet each point is projected and we use its barycentric coordinates in this facet to interpolate both its 3D cartesian coordinates and its normal. Then we modify this 3D position by moving the point along its normal by the depth stored in the cracked mesh at this point. Finally, we remove all the original points of the mesh that are inside a cracked facet and we make a new triangulation from all the remaining points (an example is given in Fig. 5).

3. Applications and results

For the agronomy, a model of cracks formation could be useful to help in the prediction of seedling emergence, because 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. Thus, this method was initially designed for that aim, by using real or simulated input data. For example, we can obtain a heightmap by measuring the surface roughness of a real patch of soil with a laser profile meter: the result is a digital elevation map (DEM). Starting from this DEM, we can try to simulate the formation of crust during a rainfall [VHLL05] and then use the result of this simulation both for the shrinking layer and the surface to crack (examples of cracks produced by our method from these input data are given in Fig. 6).

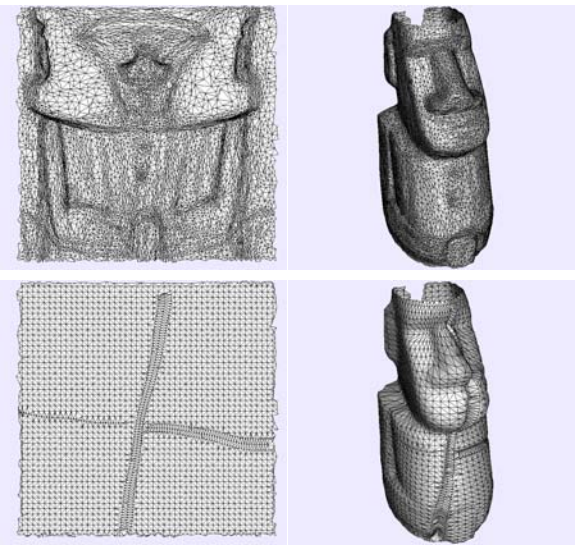


Figure 5: *Left figures show the parameterized area of the surface and right figures the corresponding 3D visualization. Top figures corresponds to the original mesh and bottom figures to the (roughly) cracked mesh.*

To get the shrinking function, in order to obtain more accurate results, we use shrinking models available in the Soil Science domain. For soil scientists, shrinkage is often divided into three stages [OH98]. In the first stage, called normal or basic shrinkage, the decrease in the volume of water in saturated soil equals the decrease in pore volume and the soil remains saturated. In the next stage, called residual shrinkage, air enters the pores and the water loss exceeds the decrease in pore volume. The last stage, called zero shrinkage, occurs when the soil has reached its densest configuration: water loss is not accompanied by any further change in volume. Soil scientists define the shrinkage characteristic as the relation between the moisture ratio v and the void ratio e, which are defined as $v = V_l/V_m$ and $e = V_p/V_m$, where V with subscripts l, m and p means volume of water, solid material and pores, respectively. For this type of modeling, as we have to take into account the presence of water in the soil, we need to add a new substate in the cell, corresponding to the moisture volume.

Olsen and Haugen [OH98] have proposed an equation of the shrinkage characteristic with three parameters λ_1, λ_2 and

λ_3 which determine the form of the shrinkage curve:

$$e(v) = \frac{1}{2}\left(\lambda_3 v + \lambda_2 + \sqrt{(\lambda_3 v + \lambda_2)^2 - 4\lambda_3\lambda_2(1-\lambda_1)v}\right) \quad (10)$$

Another equation, with four parameters, was presented by Groenevelt and Grant [GG02]:

$$e(v) = \varepsilon + k_3\left[\exp\left(\frac{-k_0}{v^n}\right) - \exp\left(\frac{-k_0}{\varepsilon^n}\right)\right] \quad (11)$$

As we have to vary the void ratio with time, we need to know how the moisture ratio is varying with the time. Water evaporation from a soil surface can be divided into two stages: the constant-rate stage and the falling-rate stage. As first-stage evaporation does not usually last long, we consider only the second-stage evaporation, by means of an equation proposed by Suleiman and Ritchie [SR03]:

$$\theta(t) = \theta_{ad} + (\theta_{dul} - \theta_{ad})\exp(-C \times t) \quad (12)$$

where θ is the volumetric water content, *i.e.*, $V_l/(V_m + V_p)$, t the elapsed time, θ_{dul} the initial uniform volumetric water content (*i.e.*, for $t = 0,\ z \geq 0$), and θ_{ad} the volumetric water content at the surface, constantly in equilibrium with the vapor pressure of the atmosphere (*i.e.*, for $t > 0,\ z = 0$). C is a conductance parameter that can vary among soils and is given by the empirical formula $C(z) = az^b$, where z is the depth, and a, b two parameters.

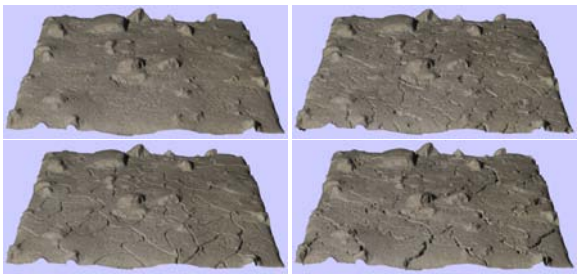


Figure 6: *Cracks applied on a simulated crusted soil. The top left image shows the original terrain, i.e., the heightmap obtained from laser profile metering followed by a simulation of crust formation. For the other images, the paths are produced by the watershed transformation on a more or less smoothed shrinking layer heightmap.*

Fig. 6 shows different types of cracks applied on a soil obtained by a laser profile metering and a simulation of crust formation. The range of the side of the terrain we simulate is typically between 25 cm and 30 cm for a depth of 10 cm, and one cell is a 2 mm side cube (corresponding to the laser profile meter resolution). For the dessiccation process, we use the equation proposed by Groenevelt and Grant, with the following parameters: $k_0 = 9.988$, $k_3 = 6101$, $n = 0.233$, $\varepsilon = 1.546$ corresponding to a clay soil [GG02]. The evaporation is modeled by the equations of Suleiman and Ritchie with these values: $a = 0.58$, $b = -1.98$, $\theta_{ad} = 0.05$ [SR03].

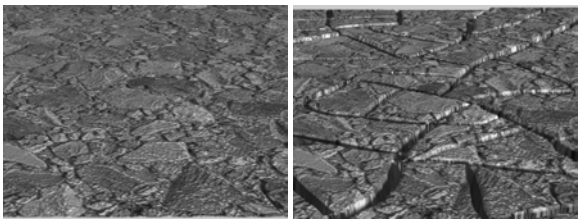


Figure 7: *Cracks applied on a rocky soil, with paths coming from the Horgan and Young model.*

Fig. 7 and 10 show results obtained with paths computed with the model of Horgan and Young [HY00] and a planar projection. Fig. 11 shows results obtained with paths computed with a watershed transformation. On the left, this transformation is based on the gaussian curvature of the model with a cylindrical projection and on the right, it is based on a purely random method (the fault algorithm) with a spherical projection. These examples show that our method gives various depths and widths, highlighting their order of creation, and thus permits plausible natural cracks. Furthermore, even if the possible path is a connected graph, our method can produce "dead branches", *i.e.*, cracks can be interrupted before joining another crack, which is close to reality.

In terms of time complexity, the duration of the cracks formation stage varies considerably with the number of the cells of the shrinking layer and the total length of cracks, *i.e.*, the number of cells for which a calculation of width and depth is required. This stage can be very short, because a few iterations suffice to give an interesting result, but it can also require a longer time. On the contrary, the time of the interpolation-displacement stage can be neglected: during our tests, it has ranged only from a few seconds to a maximum of one minute (examples are given Table 1).

	(without cracks)		(with cracks)		
Model	Vertices	Triangles	Vertices	Triangles	Timing
Soil Fig. 7	4	2	121389	241696	40 s
EG Fig. 1	33114	65484	99087	132534	1 min 20 s
Moai Fig. 11	10002	20000	92038	164212	18 min
Igea Fig. 11	134359	268714	417056	593386	21 min

Table 1: *Examples of timings and memory costs, on a Pentium IV CPU running at 3GHz with 1024MB of RAM.*

4. Conclusion

In this paper we have detailed a new generalized method to simulate the formation of cracks due to a shrinking layer. It is generalized in two ways: first, cracks can be applied at any scale to any surface provided with a parameterization, and second, the parameters controlling the creation, progression and enlargement of the cracks can either be linked to

the original model or be purely random or even be arbitrarily chosen. Following an automatically precalculated path, cracks appear, propagate with respect to their initial direction and can be attracted by another close crack. Their width is computed from the propagation of shrinkage volumes. The input data are simple and the parameters controlling the simulation are intuitive. In order to reproduce cracks on a crusted soil, we have tried our method on virtual terrains. As this method is based on physical quantities, we hope that it could be used to help in the prediction of seedling emergence. For this purpose, we have first to validate our results, for example by comparing the appearance, the total length, the area, the tortuosity of our simulated cracks with those of real cracks. Besides, we have tried our method on different 3D models and obtained visually interesting results proving that it can help to enhance the realism of these models. In the future it could be worth trying other methods for the computation of both the cracks path and the shrinking layer thickness, as well as studying more precisely the effect of the different parameters.

Acknowledgements

Special thanks to Jonathan Shewchuk for his library Triangle and to David Xianfeng Gu for his parameterization of the Stanford Bunny and other meshes not used in this paper. This work is supported by the regions of Champagne-Ardenne and Picardie.

References

[DGA05] DESBENOIT B., GALIN E., AKKOUCHE S.: Modeling cracks and fractures. *The Visual Computer 21*, 8-10 (Sept. 2005), 717–726.

[Fed02] FEDERL P.: *Modeling Fracture Formation on Growing Surfaces*. PhD thesis, University of Calgary, Alberta, Canada, Sept. 2002.

[FP04] FEDERL P., PRUSINKIEWICZ P.: Finite Element Model of Fracture Formation on Growing Surfaces. *Lecture Notes in Computer Science 3037* (May 2004), 138–145.

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

[GG02] GROENEVELT P., GRANT C.: Curvature of Shrinkage Lines in Relation to the Consistency and Structure of a Norwegian Clay Soil. *Geoderma 106* (2002), 235–245.

[GY03] GU X., YAU S.-T.: Global conformal surface parameterization. In *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (2003), pp. 127–137.

[HTK98] HIROTA K., TANOUE Y., KANEKO T.: Generation of Crack Patterns With a Physical Model. *The Visual Computer 14* (1998), 126–137.

[HTK00] HIROTA K., TANOUE Y., KANEKO T.: Simulation of Three-Dimensional Cracks. *The Visual Computer 16* (2000), 371–378.

[HY00] HORGAN G., YOUNG I.: An Empirical Stochastic Model for the Geometry of Two-Dimensional Crack Growth in Soil (with discussion). *Geoderma*, 96 (2000), 263–276.

[Lin94] LINDEBERG T.: *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, Norwell, MA, USA, 1994.

[Mac95] MACVEIGH D. T.: Emulation of Uniform Cracking. In *Proceedings of Compugraphics '95* (Dec. 1995), Santo H. P., (Ed.), pp. 218–227.

[MGDA04] MARTINET A., GALIN E., DESBENOIT B., AKKOUCHE S.: Procedural Modeling of Cracks and Fractures. In *Proceedings of Shape Modeling International* (2004).

[OH98] OLSEN P., HAUGEN L.: A New Model of the Shrinkage Characteristic Applied to Some Norwegian Soils. *Geoderma 83*, 1 (Apr. 1998), 67–81.

[OH99] O'BRIEN J. F., HODGINS J. K.: Graphical Modeling and Animation of Brittle Fracture. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (1999), ACM Press/Addison-Wesley Publishing Co., pp. 137–146.

[PMRdM95] PERRIER E., MULLON C., RIEU M., DE MARSILY G.: Computer Construction of Fractal Soil Structures: Simulation of their Hydraulic and Shrinkage Properties. *Water Resources Research 31* (1995), 2927–2943.

[PPD02] PAQUETTE E., POULIN P., DRETTAKIS G.: The Simulation of Paint Cracking and Peeling. In *Graphics Interface 2002 Conference Proceedings* (2002), pp. 59–68.

[She96] SHEWCHUK J. R.: Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In *Applied Computational Geometry: Towards Geometric Engineering*, Lin M. C., Manocha D., (Eds.), vol. 1148 of *Lecture Notes in Computer Science*. Springer-Verlag, May 1996, pp. 203–222. From the First ACM Workshop on Applied Computational Geometry.

[SM88] SKJELTORP A. T., MEAKIN P.: Fracture in Microsphere Monolayers Studied by Experiment and Computer Simulation. *Nature 335* (Sept. 1988), 424–426.

[SR03] SULEIMAN A. A., RITCHIE J. T.: Modeling Soil Water Redistribution during Second-Stage Evaporation. *Soil Science Society of America Journal 67*, 2 (2003), 377–386.

[ST94] SAITO T., TORIKAWI J.: New Algorithms for Euclidean Distance Transformation of an n-Dimensional Digitalized Picture with Applications. *Pattern Recognition 27*, 11 (1994), 1551–1565.

[TF88] TERZOPOULOS D., FLEISCHER K.: Modeling Inelastic Deformation: Viscoelasticity, Plasticity, Fracture. In *SIGGRAPH 1988, Computer Graphics Proceedings, Atlanta, GA* (Aug. 1988), vol. 22, pp. 269–278.

[VHLL05] VALETTE G., HERBIN M., LUCAS L., LÉONARD J.: A Preliminary Approach of 3D Simulation of Soil Surface Degradation by Rainfall. *Eurographics Workshop on Natural Phenomena, Dublin, Ireland* (Aug. 2005), 41–50.

[VHLR05] VOGEL H.-J., HOFFMANN H., LEOPOLD A., ROTH K.: Studies of crack dynamics in clay soil II. A physically based model for crack formation. *Geoderma 125*, 3-4 (Apr. 2005), 213–223.

[VS91] VINCENT L., SOILLE P.: Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence 13*, 6 (1991), 583–598.