Physically based Simulation of Cracks on Drying 3D Solid

Kimiya Aoki, Ngo Hai Dong, Toyohisa Kaneko, and Shigeru Kuriyama
Toyohashi University of Technology, Department of Information and Computer Sciences
1-1, Tempaku-cho, Toyohashi, Aichi, Japan 441-8580
{aoki, dong, kaneko, kuriyama}@vcl.ics.tut.ac.jp

Abstract

This paper describes a new physically based method for synthesizing 3D cracking patterns in clay solids by incorporating a moisture model. We measured temporal transitions of the physical parameters used in the model. Then we describe a new approach to synthesize cracking patterns in a 3D freeform object by its approximate quantized shape and the mesh relation of its original shape. With this new method, animated synthesis of not only external cracks but also internal cracks of 3D freeform solids is possible.

1 Introduction

Cracking is a natural phenomenon commonly observed on the surface of such objects as concrete, clay walls, mud, pottery, and tree barks. These cracks may be created naturally or in some cases by external force. Cracks on clay walls, mud etc. are due to shrinking volume as they are drying up. Successful modeling of cracks will enable the visualization of not only static cracking scenes but also dynamic scenes as animations. Cracking objects need to be of any freeform shapes. The objective of this paper is to simulate cracks from a physically based approach for meeting these objectives.

The synthesis of crack patterns has attracted attention of researchers in computer graphics (CG)[1, 2, 3, 4, 5, 6, 7]. Here we will be concerned with modeling and visualizing cracks on the surface of 3D drying clay solids. The material used here is flour-based clay. All the past works did not take into account the fact that cracks appear due to shrinking volume of clay caused by water evaporation. With using water evaporation, cracks can be observed at the cross section of object. This prompted us to introduce a moisture model.

2 Method

2.1 Geometrical Model

Given a 3D object, we obtain its geometrical model by quantization. Here we employ a uniform size δ for quantization along the x, y, and z direction. Each small cubic element will be called a cell. (In our experiments we employed $\delta = 2[mm]$.) The entire physical simulation is carried out based on this quantized model. However, after simulation based on cells is carried out, the resulting visual representation is derived based on the original smooth shape derived by using the relationship between the quantized model and the original smooth model.

2.2 Mechanical Model

Each cubic cell can be broken into five tetrahedrons. We allocate a spring to each edge of a tetrahedron. The mass of an element is divided into each node. m_i represents the mass of node i. The final mechanical model is represented by a spring network model whose nodes are on the grids of the quantized model. The simulation is based upon the entire collection of these tetrahedrons.

The velocity and the displacement of each node are calculated by solving the motion equation on nodes. In this case, the force committed to each node ($\mathbf{F}_i$) is expressed with the following equation:

$$\mathbf{F}_i = -\sum_j k_{ij} \frac{\mathbf{x}_i - \mathbf{x}_j}{|\mathbf{x}_i - \mathbf{x}_j|}(|\mathbf{x}_i - \mathbf{x}_j| - l_{oij}) + m_i \mathbf{g} \qquad (1)$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the positions of node i, neighboring node j, l_{oij} and k_{ij} are the natural length and spring constant of the spring connected to the node i and j, $\mathbf{g}$ is the gravity acceleration.

Each spring is cut when it is stretched beyond the maximum strain, and then cracks are visualized. Once an edge of a tetrahedron is cut, it is assumed that the tetrahedron is divided into two, three, or four parts.

2.3 Moisture Model

As the water evaporates from a surface node, water moves from its internal neighboring nodes toward the surface node. In order to describe the movement of water, we define moisture content ω_i at node i. Let ω represent the rate of the mass of the water. Then the motion of water

Proceedings of the 10 th Pacific Conference on Computer Graphics and Applications (PG'02)
0-7695-1784-6/02 $17.00 © 2002 IEEE

between each node is governed by the following diffusion equation:

$$\frac{\partial \omega_i}{\partial t} = D_\omega \frac{\partial^2 \omega_i}{\partial^2 \mathbf{x}_i} \tag{2}$$

where the diffusion constant D_ω is to be determined experimentally. In this paper the above equation is solved by the finite difference method (FDM).

2.4 Link between Mechanical Model and Moisture Model

It is understood that some parameters in the mechanical model are governed by the moisture condition derived from the moisture model. Specifically three parameters: the spring constant ($k(\omega)$), maximum strain ($\Gamma(\omega)$), and contraction ratio ($\gamma(\omega)$) included in the mechanical model are considered to be functions of the moisture content (ω) of a node. These functions are defined on the basis of measurements on real clay beforehand.

3 Simulation and Experiment

The crack simulation is carried out as follows. There are two time scales: long time (t) and short time (τ). First, we start by setting the long time to be zero. Assume that at the end of long time t, all the parameters are set. Then we go to the moisture model and compute the moisture content at each node of the network model. From the computed moisture content, we then derive the spring constant and the maximum strain. Next, we start the short time iterations with the new values for each node and each edge in the mechanical model until the system becomes stable. After the short time iterations end, then we increment the long time by one as (t+1). In this fashion, we alternatively repeat the process of reading the moisture content and that of initiating the short time iterations.

Figure 1 shows a temporal change of simulated cracks on a cube ($4 \times 4 \times 4$[cm]). From this figure, it is confirmed that the temporal agreement between the simulation and real is excellent. The next examples are a sphere (diameter:2.2[cm]) and a bell (bottom radius:1.45[cm], height:2.9[cm]) shape object shown in Figure 2. The resulting cracks are reasonable. Therefore, the technique for a 3D freeform object as described above is shown to be effective. The total computation time for the $4 \times 4 \times 4$[cm] cube was 8 hours on a PC with a Pentium 2G CPU. The sphere and the bell shape model required 0.7 and 1 hours, respectively.

4 Conclusion and Future Work

We proposed a new comprehensive crack model which consists of a mechanical model and a moisture model. The simulation runs with alternating turns of executing the two models. This new model can provide crack patterns which resemble actual cracks reasonably well. One of the strengths of this new method is excellent temporal agreement in crack pattern formation. We also devised a new technique to simulate for a freeform solid with its quantized version.

Future work includes the effect of cracks on the moisture model, which is not considered in this paper. Another work is to employ texture mapping on the surface in order to illustrate useful applications.

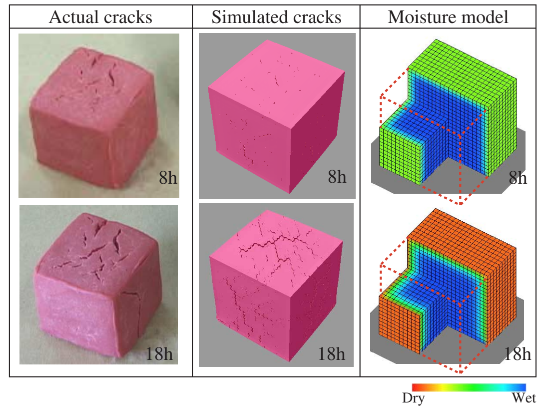
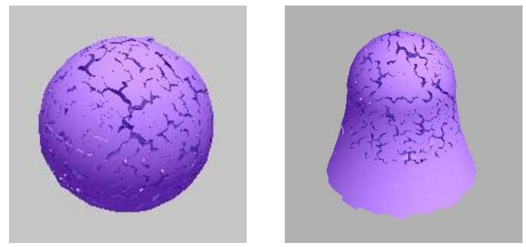


Figure 1: Crack patterns and moisture models



(a) Sphere (b) Bell

Figure 2: Crack pattern on freeform object

References

[1] N. Chiba, K. Kaino, S. Wada, and K. Muraoka. Behavioral model of cracks and its application to computer graphics (in english). *Systems and Computers in Japan*, Vol. 22, pp. 82–91, 1991.

[2] T. Agui T, K. Moriyama, T. Nagao, and M. Nakajima. A cg representation method of cracks on a plate glas. *Trans IPS*, Vol. 33, No. 10, pp. 1235–1244, 1992.

[3] A. T. Skjeltorp and P. Meakin. Fracture in microsphere monolayers studied by experiment and computer simulation. *Nature*, Vol. 335, pp. 424–426, 1988.

[4] K. Hirota, Y. Tanoue, and T. Kaneko. Simulation of three-dimensional cracks. *Visual Computer*, Vol. 16, pp. 371–378, 2000.

[5] S. Gobron and N. Chiba. 3d surface cellular automata and their applications. *J. of Visualization and Computer Animation*, Vol. 10, pp. 143–158, 1999.

[6] A. Norton, G. Turk, B. Bacon, J. Gerth, and P. Sweeney. Animation of fractures by physical model. *Visual Computer*, Vol. 7, pp. 210–219, 1991.

[7] J.F. O'Brien and J. K. Hodgin. Graphical modeling and animation of brittle fracture. *Computer Graphics*, Vol. 34, pp. 137–146, 1999.

Proceedings of the 10 th Pacific Conference on Computer Graphics and Applications (PG'02)
0-7695-1784-6/02 $17.00 © 2002 **IEEE**