

VISUALIZATION TECHNIQUES IN MATHEMATICAL ENGINEERING

MARTIN PETŘEK

ABSTRACT. In mathematical engineering visualizing the result of the experiments or simulations is needed. Scientific visualization is a field of computer science that includes creating visual representations of mathematical data, image post-processing algorithms, and also creating graphical user-interface application. For many problems in practice plenty of visualization algorithms have been developed. This article shows some basic visualization techniques like color-mapping and surface warping. These techniques were used to visualize simulation of the behaviour of biological species. Surface warping was used to visualize crystal growth (phase field model). The second part describes a frequently used technique for visualization of vector field – Line Integral Convolution.

1. NUMERICAL SIMULATION OF COMPETITION-DIFFUSION SYSTEMS

In recent years plenty of models for describing the behaviour of biological entities has been designed. These are usually described by PDE system of diffusion type. Let us show one of this model provided by [6]. On some domain the density of population is studied. The PDE system has the following form

$$(1) \quad \frac{\partial u_i}{\partial t} = D_i \Delta u_i + f_i(u_1, \dots, u_n) \quad \text{on } \Omega, \quad i \in \hat{n}$$

where u_i denotes population density of i^{th} species on examined domain, D_i is diffusion factor of species. Suitable choice of vector function f is a heart of this model. Authors designed the form

$$f_i = \left(r_i - \sum_{j=1}^n a_{ij} u_j \right) u_i$$

where r_i symbols the population-growth rate of i^{th} species, a_{ij} features the inter-specific competition rate between u_i and u_j (if $i \neq j$) and intraspecific competition rate (if $i = j$). Suppose then nonnegative values of these coefficients. If we include initial and Neumann boundary conditions we get the system

$$(2) \quad u_i(x, 0) = u_{i0}(x) \quad (i = 1, 2, \dots, n), \quad x \in \Omega$$

$$(3) \quad \frac{\partial u_i}{\partial \nu} = 0 \quad (i = 1, 2, \dots, n), \quad x \in \Omega, t > 0$$

Key words and phrases. visualization, color-mapping, surface-warping, LIC, crystal growth, phase-field model, behaviour of biological species.

Here ν denotes outer normal to $\partial\Omega$ and u_{i0} is nonnegative function. That means that no species could escape out from the defined area. The detailed analysis of the system describes non-trivial behaviour, even if we suppose only small count of species (at least three).

1.1. Color-mapping method. To perform the behaviour we have simulated this model on structured point grid using classical FDM. The computed grid had 100×100 pixels. We supposed three species (A, B, C). The initial conditions we given by “colored picture” where each point of RGB values denotated then scaled population density and the biggest value (R/G/B) appointed the species.

Then the color-mapping technique was used. Population density of all three species was scaled into $(0, 1)$, $(1, 2)$, $(2, 3)$ ranges. The common scalar function was then mapped into rain-bow color table and then visualized in OpenGL using VTK Visualization Toolkit. We computed approximately 1000 snapshots, then we used MPEG_ENCODER to create video animation.

Following pictures shows development of the system in time.

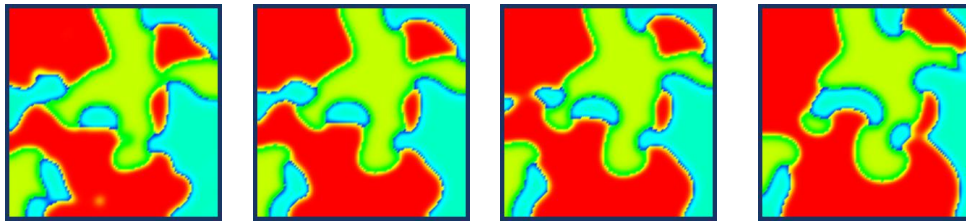


FIGURE 1. Simulation of behaviour of biological entites (Diffusion model) computed time-levels: $t = 20$ (left), $t = 30$ (second left), $t = 40$ (third left), $t = 60$ (right)

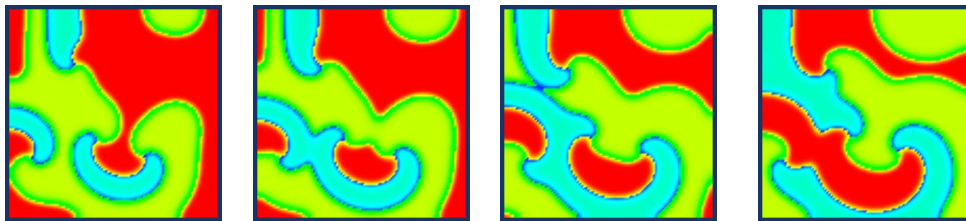


FIGURE 2. Simulation of behaviour of biological entites (Diffusion model) computed time-levels: $t = 210$ (left), $t = 230$ (second left), $t = 260$ (third left), $t = 290$ (right)

The previous figure shows the typical character of these diffusion type model. For some adjusting of directive matrices it leads to spiral-like segregation behaviour among the entities. In some cases the system settes in an equilibrium other cases leads to dominance of one species. This happen always when we suppose only two entities competing on a convex domain.

2. SCALAR-WARPING TECHNIQUE

This techniques we use to visualize the numerical sollution of phase-field equations use in material research.

The PDE system of phase-fiel model has the form

$$(4) \quad \rho c \frac{\partial u}{\partial t} = \lambda \nabla^2 u + L \frac{\partial p}{\partial t}$$

$$(5) \quad \alpha(\theta) \xi^2 \frac{\partial p}{\partial t} = g(\theta) [\xi^2 \nabla^2 p + ap(1-p)(p-0.5)] + b\beta(u^* - u) \xi^2 \|\nabla p\|$$

where u means temperature and p denotes phase.

The solution (u/p) is scalar function defined on some 2D-domain. The numerical sollution is obtained on some discreet subspace of the domain. Mostly the domain is defined as a structured grid or set of triangles. To solve the equations numerical methods like FDM, FEM, FEV are we used. Each point of the grid is connected to its neighbours according to the topology. To generate surface like representation we just transform the vertices of the grid according the function values while the “connection edges” are saved. To show the second function onto the warped surface we can use previous technique of color-mapping. As we see, the PDE system is a parabolic so we can again observe time developing surface.

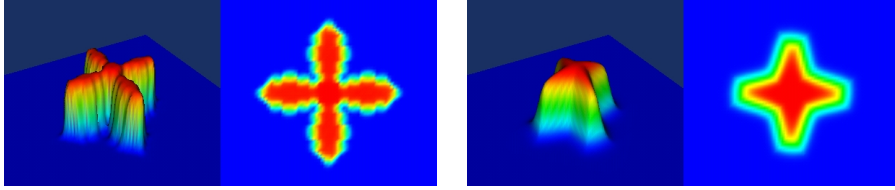


FIGURE 3. Simulation of anisotropic crystal growth – 1. part (phase-field model), time-levels: $t = 1$ (left), $t = 30$ (second left)

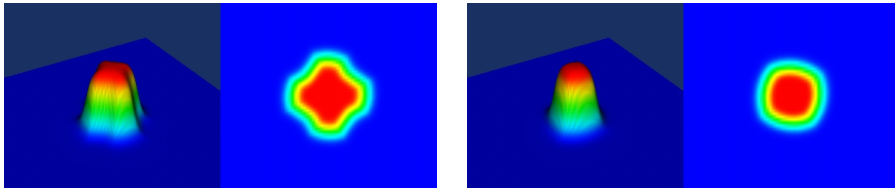


FIGURE 4. Simulation of anisotropic crystal growth – 2. part (phase-field model), time-levels $t = 60$ (left), $t = 90$ (second left) visualization technique: Surface warping and color-mapping

3. LIC METHOD

This method was first presented by Cabral and Leedom in 1993. It’s used to visualize vector field. The method is based on filtering the image texture along the

streamlines of the vector field. The filtering is provided by convolution with some filter kernel. Let $\Omega \subset \mathbf{R}^2$, with vector field $\mathbf{v} : \Omega \rightarrow \mathbf{R}^2$. Integral curve is defined as

$$\frac{d}{dt}\boldsymbol{\tau}(t) = \mathbf{v}(\boldsymbol{\tau}(t)) \quad \boldsymbol{\tau}(t_0) = \mathbf{x}_0, \quad \mathbf{x}_0 \in \Omega$$

This curve can be parametrized by arc-length by standardizing the \mathbf{v} . Let us suppose for now parametric curve $\boldsymbol{\sigma}(s)$ where s is the parameter and $s_0 = x_0$. Be T the input image texture given on Ω and h filter kernel with the finite support $[-L, L]$. Intensity $I(\mathbf{x})$ for each point \mathbf{x} of the curve can be computed as

$$(6) \quad I(\mathbf{x}) = \int_{s-L}^{s+L} T(\boldsymbol{\sigma}(s'))h(s-s')ds' \quad \text{pro } \mathbf{x} \in \boldsymbol{\sigma}(s)$$

If we use some initial noise image as an input texture T we obtain result image as follows. The vector magnitude is usually mapped into an color to show where is the field more intensive. The colorizing is simply made by α -blending of two images where the resulting image is given by

$$O = \alpha \cdot I_1 + (1 - \alpha) \cdot I_2 \quad \alpha \in (0, 1)$$

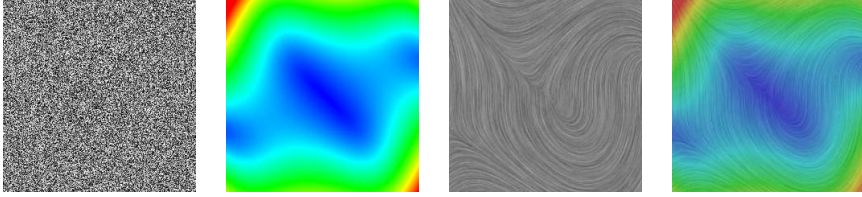


FIGURE 5. Line integral convolution, Initial noise (left), Vector field magnitude (second left) , LIC along streamlines (third left), Resulting image created by α -blending (right)

Calculating form 6 is too expensive, therefore some efficient algorithms have been designed. The first “Fast-LIC” originally presented in [4], suppose the simple box-type filter kernels. The integral simplifies and the intensities computed on the streamline can be done effective. In algorithm we walk on the streamline and keep the intensity value in then sequence of crossed points (s_0, s_1, \dots, s_n) and the actual value of convolution integral. In the next new point we just subtract the intensity $T(\boldsymbol{\sigma}(s_0))$ and add the intensity $T(\boldsymbol{\sigma}(s_{n+1}))$ of the pixel s_{n+1} . By this we walk the streamline till the boundary of the domain is reached. To accelerate the algorithm usually the number of “hits” in each pixel is saved in memory. If we get into the point where the number of hits is greater then some predefined constant, we stop and start the algorithm in some other point (mean another streamline). Finally the normalization of intensities should be done. The intensity of each pixel has to be divide by the number of hits.

The main constraint of the previous fast-lic algorithm is the special choise of filter kernel. If we want use more compound filter the procedure can’t be used. H. C. Hege and D. Stalling introduced generalization of fast-lic algorithm for special

class of filter kernels. They show in [5] that using the piecewise polynomial functions as a filter kernel leads to effective computation of convolution integral. The main idea is based on convolution theorem

$$(7) \quad f * h(x) = \int_{-\infty}^{\infty} f(y)h(x-y)dy = \int_{-\infty}^{\infty} F(y)h'(x)dy = F * h'(x)$$

Applying 7 at least n times and suppose $h^{(n)}$ is delta distribution ($h^{(n)}(x) = c\delta(x-\xi)$) we obtain $f * h(x) = cF_n(x - \xi)$. Now when we express h in terms of truncated power functions, i.e. $h = \sum_{ij} c_{ij}\varphi_{ij}$ the discrete approximation of convolution integral can be computed as

$$(8) \quad I(k\Delta s) = \sum_{ij} c_{ij}F_j(k\Delta s - \xi_i)$$

1. The filter kernel
2. The sums F_n

REFERENCES

- [1] Becker, J.; Preusser, T.; Rumpf, M., PDE Methods in Flow Simulation Post Processing, August 17, 1999
- [2] Bene, M. and Mikula, K., Simulation of Anisotropic Motion by Mean Curvature – Comparison of Phase Field and Sharp Interface Approaches, Acta Mathematica Universitatis Comenianae, Volume 67 (1998), No. 1, 17–42
- [3] Bene, M., Mathematical and computational aspects of solidification of pure substances, Acta Mathematica Universitatis Comenianae, Volume 70 (2001), 123–151
- [4] Cabral, B.; Leedom, L. C., Imaging vector fields using line integral convolution, Computer Graphics (SIGGRAPH '93 Proceedings) (J. T. Kajiya, ed.), vol. 27, August 1993, pp. 263–272.
- [5] Hege, H.-C.; Stalling D., Fast LIC with Piecewise Polynomial Filter Kernels, Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB), Germany
- [6] Shin-Ichiro Ei; Ryo Ikota; Masayasu Mimura, Segregating partition problem in competition-diffusion systems, 1999
- [7] Schroeder, W.; Martin, K.; Lorensen, B., The Visualization Toolkit, 2nd Edition, Prentice Hall, 1998, ISBN 0-13-954694-4

DEPARTMENT OF MATHEMATICS, FACULTY OF NUCLEAR SCIENCES AND PHYSICAL ENGINEERING, CZECH TECHNICAL UNIVERSITY, TROJANOVA 13, 120 00 PRAGUE 2, CZECH REPUBLIC

E-mail address: petrek@bimbo.fjfi.cvut.cz