FINITE VOLUME SCHEMES FOR THE GENERALIZED SUBJECTIVE SURFACE EQUATION IN IMAGE SEGMENTATION

KAROL MIKULA AND MARIANA REMEŠÍKOVÁ

In this paper, we describe an efficient method for 3D image segmentation. The method uses a PDE model – the so called generalized subjective surface equation which is an equation of advection-diffusion type. The main goal is to develop an efficient and stable numerical method for solving this problem. The numerical solution is based on semi-implicit time discretization and flux-based level set finite volume space discretization. The space discretization is discussed in details and we introduce three possible alternatives of the so called diamond cell finite volume scheme for this type of 3D nonlinear diffusion equation. We test the performance of the method and all its variants introduced in the paper by determining the experimental order of convergence. Finally we show a couple of practical applications of the method.

Keywords: image segmentation, finite volume method, flux-based level set method
AMS Subject Classification: e.g. 93E12, 62A10, 62F15

1. INTRODUCTION

Image segmentation, i.e. extraction of particular objects or structures from the image, is one of the fundamental problems of image processing and has a large variety of applications in such fields as medicine or biology. The scanning devices for obtaining the images, are evolving rapidly and their using is becoming widespread and standard. This is accompanied by the demand for an automated extraction of information from the images and therefore, the imaging technique is usually supported by software tools based on image processing algorithms. Image segmentation is one of the most usual tasks. Typical examples are extraction of organs or other significant structures (tumors, aneurisms) in medical imaging or extraction of cells from microscope images in biology. As the number of images to process can be sometimes very high, the quality can vary and it is necessary to get the information or image analysis as quickly as possible, it is important to have an efficient and robust algorithm that is able to operate quickly and at the same time deal with some imperfections of the image, such as noise, incomplete borders of objects or presence of small artifacts.

In this paper, we are dealing with the generalized subjective surface method for 3D image segmentation. The method is based on the idea of evolution of a

segmentation function governed by an advection-diffusion model [7]. The paper is focused on the numerical solution of the model equation and particularly on its space discretization. This is done by using the so called flux-based level set finite volume method where, moreover, the so called diamond cell technique and its variants are applied in order to obtain the necessary approximation of average gradients on the finite volume sides.

The generalized subjective surface method was originally developed and applied to segmentation of long time series of 3D images of embryogenesis, see [7]. Here we provide all details of its spatial discretization with the comparison of various variants of the diamond cell approximation. It turns out that the scheme that we call reduced diamond cell method has comparable precision to the other variants and it is the fastest method from the computational point of view. A large number of experiments proved that the method can be succesfully used to extract the shape of cell nuclei, cells themselves or whole embryo. Some examples of this interesting application will be shown in the last section.

2. THE MATHEMATICAL MODEL

Let $I^0 : \Omega \to R$, $\Omega \subset R^3$ represent the intensity function of an image. If we want to segment an object, we need a segmentation seed - the starting point that determines the approximate position of the object in the image. Then we construct an initial segmentation function $u_0(x)$. The principle of the subjective surface method [9, 8, 10] is that the position of the seed is the main factor determining the form of this function. Having constructed the initial function, we let it evolve by solving the following GSUBSURF equation

$$u_t - w_{con}\nabla g \nabla u = w_{dif} g \sqrt{\varepsilon^2 + |\nabla u|^2}\, \nabla . \left(\frac{\nabla u}{\sqrt{\varepsilon^2 + |\nabla u|^2}} \right) \tag{1}$$

where u is the evolving function, $u(0, x) = u_0(x)$ and we consider the zero Dirichlet boundary condition on $\partial\Omega$, ε is the regularization parameter, usually $\varepsilon << 1$. The function g is the edge detector and can be of the form $g(s) = 1/(1 + Ks^2)$, $K \geq 0$. It is applied to the gradient of image intensity I^0, presmoothed by convolution with the Gaussian kernel G_σ with a small variance σ, which we denote by I_σ. The essential property of the function $g(|\nabla G_\sigma * I_\sigma|)$ is that its negative gradient points towards the edges in the image. We can also use a more general form $g(s) = f\left(1/(1 + Ks^2)\right)$ keeping in mind that f has to preserve the edge detecting property. If f is suitably chosen, this can improve the quality of the results.

The equation (1) represents a generalization of the classical subjective surface method [9] that we get if $w_{con} = 1.0$, $w_{dif} = 1.0$. Introducing two new parameters makes the model more flexible and the possibility to separately control advection and diffusion represents a chance for improvement of the performance of the method.

3. DISCRETIZATION OF THE MODEL

3.1. TIME DISCRETIZATION

In order to discretize (1) in time, we apply the semi-implicit approach that guarantees unconditional stability with respect to the diffusion term. Let us suppose that we solve the equation (1) in time interval $I = [0, T]$ and in N equal time steps. The time step is denoted by τ, $\tau = T/N$. The time discretization of the equation then looks as follows

$$\frac{u^n - u^{n-1}}{\tau} - w_{con} \nabla g . \nabla u^{n-1} = w_{dif} g \sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}\, \nabla . \frac{\nabla u^n}{\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}} \qquad (2)$$

3.2. SPACE DISCRETIZATION

In order to discretize (2) in space, we apply the so called flux-based level set finite volume method. First, we will explain the general principle and then we will describe three different alternatives for the implementation.

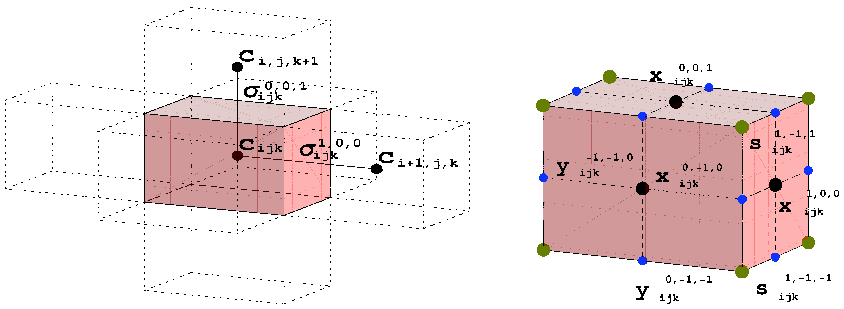

Figure 1: The finite volume mesh, illustration of our notation.

We indentify the finite volumes of the regular rectangular mesh $\mathcal{T}_h$ with the voxels of the 3D image. We denote each volume by V_{ijk}, $i = 1 \dots N_1$, $j = 1 \dots N_2$, $k = 1 \dots N_3$. Let h_1, h_2, h_3 be the size of the volumes in x_1, x_2, x_3 direction. Let $m(V_{ijk})$ denote the volume of V_{ijk} and c_{ijk} its barycenter. By u^n_{ijk} we denote the approximate value of u^n in c_{ijk}. For each $V_{ijk} \in \mathcal{T}_h$, we define three index sets. First, let N_{ijk} denote the set of all (p, q, r) such that $p, q, r \in \{-1, 0, 1\}$, $|p| + |q| + |r| = 1$. Then, let M_{ijk} be the set of all (p, q, r), $p, q, r \in \{-1, 1\}$. The third set P_{ijk} is the set of all (p, q, r) such that $p, q, r \in \{-1, 0, 1\}$, $|p| + |q| + |r| = 2$. Let us first consider $(p, q, r) \in N_{ijk}$. The line connecting the center of V_{ijk} and the center of its neighbor $V_{i+p,j+q,k+r}$ is denoted by σ^{pqr}_{ijk} and its length $m(\sigma^{pqr}_{ijk})$. The planar sides of finite volume V_{ijk} are denoted by e^{pqr}_{ijk} with area $m(e^{pqr}_{ijk})$ and normal ν^{pqr}_{ijk}. Let x^{pqr}_{ijk} be the point where the line σ^{pqr}_{ijk} crosses the side e^{pqr}_{ijk}. Further, for any $(p, q, r) \in M_{ijk}$, let s^{pqr}_{ijk} represent the vertices of the finite volume V_{ijk}, and for $(p, q, r) \in P_{ijk}$, let y^{pqr}_{ijk} denote the midpoints of the voxel edges. The approximate value of u^{n-1} in x^{pqr}_{ijk},

s_{ijk}^{pqr} and y_{ijk}^{pqr}, where (p,q,r) belongs to the corresponding index set, is denoted by u_{ijk}^{pqr}, omitting the time index, as only the values from the time level $n-1$ will be needed in these points.

Let us use the notation $v = -w_{con}\nabla g$. According to [5], the advection term in (2) can be written in equivalent form

$$v.\nabla u^{n-1} = \nabla.(vu^{n-1}) - u^{n-1}\nabla.v\,. \tag{3}$$

Now let us make the space discretization of (2) . We substitute the advection term by (3) and integrate the resulting equation over V_{ijk}

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau}\,dx + \int\limits_{V_{ijk}} \nabla.(vu^{n-1})dx - \int\limits_{V_{ijk}} u^{n-1}\nabla.v\,dx = \int\limits_{V_{ijk}} w_{dif} g \sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}\nabla.\frac{\nabla u^n}{\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}}dx\,. \tag{4}$$

If we denote by g_{ijk} and $\bar{Q}_{\varepsilon,ijk}^{n-1}$ average values of g and $Q_\varepsilon = \sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}$ in the finite volume V_{ijk}, and we consider piecewise constant representation of u^n and u^{n-1} over the finite volume mesh, using the divergence theorem we get

$$m(V_{ijk})\frac{u_{ijk}^n - u_{ijk}^{n-1}}{\tau} + \sum_{N_{ijk}} \int\limits_{e_{ijk}^{pqr}} u^{n-1} v.\nu_{ijk}^{pqr}\,d\gamma - u_{ijk}^{n-1}\sum_{N_{ijk}} \int\limits_{e_{ijk}^{pqr}} v.\nu_{ijk}^{pqr}\,d\gamma = w_{dif}\; g_{ijk}\; \bar{Q}_{\varepsilon,ijk}^{n-1} \sum_{N_{ijk}} \int\limits_{e_{ijk}^{pqr}} \frac{\nabla u^n}{\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}}.\nu_{ijk}^{pqr}\,d\gamma. \tag{5}$$

Further, for $(p,q,r) \in N_{ijk}$ we approximate integrated fluxes $\int\limits_{e_{ijk}^{pqr}} v.\nu_{ijk}^{pqr}\,d\gamma$ by

$$v_{ijk}^{pqr} = m(e_{ijk}^{pqr})\left(-w_{con}\frac{g_{i+p,j+q,k+r} - g_{ijk}}{m(\sigma_{ijk}^{pqr})}\right),$$

and we distinguish between the outflow and inflow boundaries by defining two sets of indices $N_{ijk}^{out} = \{(p,q,r) \in N_{ijk}\,,\, v_{ijk}^{pqr} > 0\}\,, \quad N_{ijk}^{in} := \{(p,q,r) \in N_{ijk}\,,\, v_{ijk}^{pqr} \le 0\}\,.$ Using the upwind principle in the second term on the LHS of (5), approximating the normal derivative $\nabla u^n.\nu_{ijk}^{pqr}$ on the RHS by $(u_{i+p,j+q,k+r}^n - u_{ijk}^n)/m(\sigma_{ijk}^{pqr})$ and defining $Q_{\varepsilon,ijk}^{pqr;n-1}$ as an average of Q_ε on e_{ijk}^{pqr}, we get the following discrete formulation

$$m(V_{ijk})\frac{u_{ijk}^n - u_{ijk}^{n-1}}{\tau} + \sum_{N_{ijk}^{out}} u_{ijk}^{n-1} v_{ijk}^{pqr} + \sum_{N_{ijk}^{in}} u_{i+p,j+q,k+r}^{n-1} v_{ijk}^{pqr} - u_{ijk}^{n-1}\sum_{N_{ijk}} v_{ijk}^{pqr} = w_{dif}\; g_{ijk}\; \bar{Q}_{\varepsilon,ijk}^{n-1} \sum_{N_{ijk}} m(e_{ijk}^{pqr})\frac{u_{i+p,j+q,k+r}^n - u_{ijk}^n}{Q_{\varepsilon,ijk}^{pqr;n-1} m(\sigma_{ijk}^{pqr})}$$

that can be further simplified and we get our **fully discrete scheme**

$$m(V_{ijk})\frac{u^n_{ijk}-u^{n-1}_{ijk}}{\tau}+\sum_{N^{in}_{ijk}}\left(u^{n-1}_{i+p,j+q,k+r}-u^{n-1}_{ijk}\right)v^{pqr}_{ijk}=$$

$$w_{dif}\,g_{ijk}\,\bar{Q}^{n-1}_{\varepsilon,ijk}\sum_{N_{ijk}}m(e^{pqr}_{ijk})\frac{u^n_{i+p,j+q,k+r}-u^n_{ijk}}{Q^{pqr;n-1}_{\varepsilon,ijk}m(\sigma^{pqr}_{ijk})}\,. \tag{6}$$

Taking into account the zero Dirichlet boundary conditions, the scheme (6) represents a linear system of equations that can be solved efficiently for example by the SOR method. In order to be able to perform the computations, we still need to express the average values g_{ijk}, $\bar{Q}^{n-1}_{\varepsilon,ijk}$ and $Q^{pqr;n-1}_{\varepsilon,ijk}$ either in voxels or on voxel sides. In the following sections, we will describe the possibilities for obtaining these quantities by using the diamond cell strategy.

3.2.1. DIAMOND CELL APPROXIMATION

The basic idea for all our schemes is to use the so called diamond cells. A diamond is an auxiliary volume in the form of an octahedron whose vertices are the barycenters of two neighboring voxels and the vertices of their common side, see Fig. 2 .

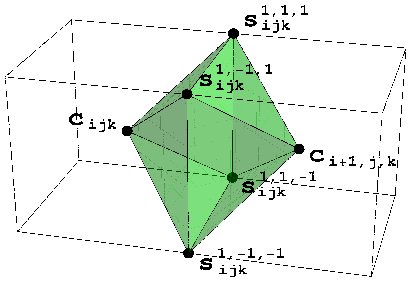

Figure 2: Construction of a diamond cell

Let us consider the diamond cell D shown in the figure. The values of u^{n-1} in the voxel centers are known. In addition, we will need to approximate the values in voxel vertices s^{pqr}_{ijk}, $(p,q,r)\in M_{ijk}$. To that goal we compute the average of neighboring voxels values, i.e., we define

$$\begin{aligned} u^{pqr}_{ijk}= \quad & \frac{1}{8}\Big(u^{n-1}_{ijk}+u^{n-1}_{i+p,j,k}+u^{n-1}_{i,j+q,k}+u^{n-1}_{i,j,k+r}+u^{n-1}_{i+p,j+q,k}+ \\ & u^{n-1}_{i+p,j,k+r}+u^{n-1}_{i,j+q,k+r}+u^{n-1}_{i+p,j+q,k+r}\Big),\ \text{for } (p,q,r)\in M_{ijk}\,. \end{aligned} \tag{7}$$

Now we can proceed to the computation of the averaged gradient of u^{n-1} in D. Using the divergence theorem, we get

$$\frac{1}{m(D)}\int_D \nabla u^{n-1}\,dx=\frac{1}{m(D)}\int_{\partial D}u^{n-1}\mathbf{n}_{\partial D}\,ds \tag{8}$$

If we assume that a side σ of D has vertices V_σ^1, V_σ^2 and V_σ^3, then the term on the RHS can be approximated as follows

$$\frac{1}{m(D)}\int\limits_{\partial D} u^{n-1}\mathbf{n}_{\partial D}\,ds \approx \frac{1}{m(D)}\sum_{\sigma\in\partial D}\frac{1}{3}\left(u_{V_\sigma^1}^{n-1}+u_{V_\sigma^2}^{n-1}+u_{V_\sigma^3}^{n-1}\right)m(\sigma)\mathbf{n}_\sigma \qquad (9)$$

The volume of D can be expressed as $m(D)=\frac{h_1h_2h_3}{3}$ and the areas of its faces as well as their normals can be easily obtained by performing the vector product od their oriented edges. Finally, we get the following expression

$$\begin{aligned}\frac{1}{m(D)}\int\limits_D \nabla u^{n-1}\,dx \;&\approx\; \frac{u_{i+1,j,k}^{n-1}-u_{ijk}^{n-1}}{h_1}\mathbf{e}_1\\ &+\; \frac{u_{ijk}^{1,1,1}+u_{ijk}^{1,1,-1}-u_{ijk}^{1,-1,1}-u_{ijk}^{1,-1,-1}}{2h_2}\mathbf{e}_2\\ &+\; \frac{u_{ijk}^{1,1,1}+u_{ijk}^{1,-1,1}-u_{ijk}^{1,-1,-1}-u_{ijk}^{1,1,-1}}{2h_3}\mathbf{e}_3 \qquad (10)\end{aligned}$$

where $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ are the unit base vectors. This expression, derived for the voxel side e_{ijk}^{100}, and its analogies for diamond cells corresponding to the other sides of the voxel V_{ijk}, can be used as the approximation of the gradient in the barycenters x_{ijk}^{pqr} of the sides e_{ijk}^{pqr}, $(p,q,r)\in N_{ijk}$. If we denote such diamond cell approximation of the gradient on voxel side e_{ijk}^{pqr} by $\nabla^{pqr}u_{ijk}^{n-1}$, and use the same strategy to compute gradients of image intensity I_σ, we can define

$$\begin{gathered}Q_{\varepsilon,ijk}^{pqr;n-1}=\sqrt{\varepsilon^2+|\nabla^{pqr}u_{ijk}^{n-1}|^2},\quad \bar{Q}_{\varepsilon,ijk}^{n-1}=\sqrt{\varepsilon^2+\tfrac{1}{6}\textstyle\sum_{N_{ijk}}|\nabla^{pqr}u_{ijk}^{n-1}|^2}\\ g_{ijk}=g\left(\tfrac{1}{6}\textstyle\sum_{N_{ijk}}|\nabla^{pqr}I_{\sigma;ijk}|\right) \qquad (11)\end{gathered}$$

and use it in the fully discrete scheme (6).

3.2.2. SPLIT DIAMOND CELL APPROXIMATION

Now let us present a modification of the diamond cell strategy introduced originally in [1] for solving the classical subjective surface equation. For each side e_{ijk}^{pqr}, $(p,q,r)\in N_{ijk}$, we construct four auxiliary volumes by splitting the diamond cell in four tetrahedral parts, see Fig. 3. Let us consider the side e_{ijk}^{100} indicated in the figure. All tetrahedra have two vertices in common, c_{ijk} and $c_{i+1,j,k}$, and the remaining vertices are always two neighboring vertices s_{ijk}^{1qr} of the side e_{ijk}^{100}. For the other sides of the voxel, the tetrahedra are constructed analogously.

Let us consider the tetrahedron T displayed in the figure. Our goal is to approximate the following average gradient

$$\bar{G}_T=\left(\bar{G}_T^x,\bar{G}_T^y,\bar{G}_T^z\right)=\frac{1}{m(T)}\int\limits_T\nabla u^{n-1}\,dx$$

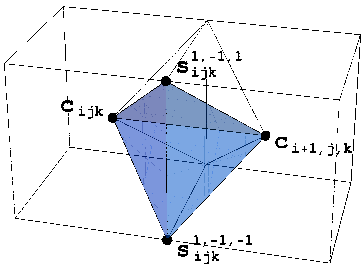

Figure 3: Construction of a tetrahedral volume

Let us use the same approximation of the values of u^{n-1} in voxel vertices as in (7). It is possible to perform similar calculations as in (9) for such tetrahedral cell, but one can get the same result using a more straightforward procedure. Let us consider the function u^{n-1} linear on T. Then we can directly write

$$\bar{G}_T^x = \frac{u_{i+1,j,k}^{n-1} - u_{ijk}^{n-1}}{h_1}, \quad \bar{G}_T^z = \frac{u_{i,j,k}^{1,-1,1} - u_{ijk}^{1,-1,-1}}{h_3}$$

$$\bar{G}_T^y = \frac{\frac{u_{i+1,j,k}^{n-1} + u_{ijk}^{n-1}}{2} - \frac{u_{ijk}^{1,-1,1} + u_{ijk}^{1,-1,-1}}{2}}{\frac{h_2}{2}}$$

This leads to the following approximation of the averaged gradient in T

$$\frac{1}{m(T)} \int_T \nabla u^{n-1}\, dx \approx \frac{u_{i+1,j,k}^{n-1} - u_{ijk}^{n-1}}{h_1} \mathbf{e}_1 \tag{12}$$
$$+ \frac{u_{i+1,j,k}^{n-1} + u_{ijk}^{n-1} - u_{ijk}^{1,-1,1} - u_{ijk}^{1,-1,-1}}{h_2} \mathbf{e}_2 + \frac{u_{ijk}^{1,-1,1} - u_{ijk}^{1,-1,-1}}{h_3} \mathbf{e}_3 \,.$$

An analogous procedure can be applied to any of the 24 tetrahedra corresponding to the 6 sides of the voxel. Let T_{ijk}^{pqrs}, $(p,q,r) \in N_{ijk}$, $s = 1 \ldots 4$, represent the four tetrahedra belonging to the side e_{ijk}^{pqr} and $\nabla^{pqrs} u_{ijk}^{n-1}$ the corresponding averaged gradients approximated according to (12) . Then we can define

$$Q_{ijk}^{pqr;n-1} = \frac{1}{4} \sum_{s=1}^{4} |\nabla^{pqrs} u_{ijk}^{n-1}|, \quad \bar{Q}_{\varepsilon,ijk}^{n-1} = \sqrt{\varepsilon^2 + \frac{1}{6} \sum_{N_{ijk}} \left(Q_{ijk}^{pqr;n-1} \right)^2} \tag{13}$$

$$Q_{\varepsilon,ijk}^{pqr;n-1} = \sqrt{\varepsilon^2 + \frac{1}{4} \sum_{s=1}^{4} |\nabla^{pqrs} u_{ijk}^{n-1}|^2}, \quad g_{ijk} = g\left(\frac{1}{6} \sum_{N_{ijk}} \frac{1}{4} \sum_{s=1}^{4} |\nabla^{pqrs} I_{\sigma;ijk}| \right) .$$

Let us remark that taking the average of $\nabla^{pqrs} u_{ijk}^{n-1}$ for $s = 1 \ldots 4$ would yield the same expression of the averaged gradient on the voxel side as in the original diamond cell strategy (10). The difference between the two approaches lies in averaging the absolute values of $|\nabla^{pqrs} u_{ijk}^{n-1}|$ instead of averaging gradients $\nabla^{pqrs} u_{ijk}^{n-1}$ as vectors.

3.2.3. THE REDUCED DIAMOND CELL APPROXIMATION

Finally, let us present a useful simplification of the original diamond cell approach. In both methods mentioned in the previous two sections, there were 18 points contributing to the approximation of the average gradient on one particular voxel side e_{ijk}^{pqr}. For example, as indicated in Fig. 4, for the side $S = e_{ijk}^{100}$, the values in the points $c_{i,j+q,k+r}$, $c_{i+1,j+q,k+r}$, $q, r \in \{-1, 0, 1\}$, are involved. The principle of the reduced diamond cell approach is to omit the values in the eight corner points $c_{i,j+q,k+r}$, $c_{i+1,j+q,k+r}$, $q, r \in \{-1, 1\}$ and thus reduce the stencil from 18 to 10 points and consequently the computational cost. Such reduction can be done using 2D diamond cell strategy applied in two planes containing $c_{i,j,k}$, $c_{i+1,j,k}$, and othogonal to S.

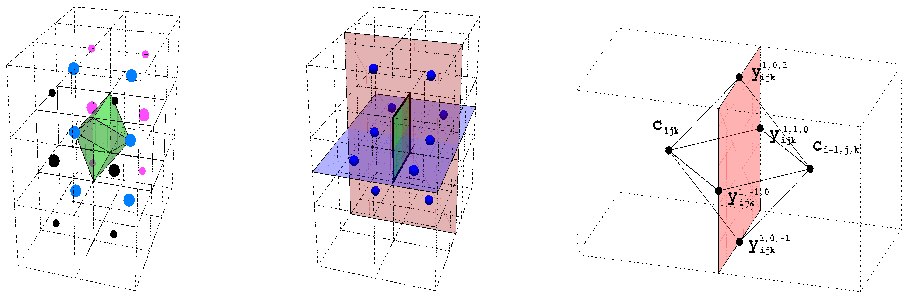

Figure 4: Reduced diamond cell scheme. On the left, the points of the diamond cell scheme corresponding to e_{ijk}^{100}. In the middle, the points of the reduced scheme. On the right, 2D analogy to the diamond cell strategy.

Our reduced scheme is using the values of u^{n-1} in the midpoints y_{ijk}^{pqr} of the voxel edges which are approximated for any $(p, q, r) \in P_{ijk}$ by

$$\begin{aligned} u_{ijk}^{pq0} &= \frac{1}{4}(u_{ijk}^{n-1} + u_{i+p,j,k}^{n-1} + u_{i,j+q,k}^{n-1} + u_{i+p,j+q,k}^{n-1}) \\ u_{ijk}^{p0r} &= \frac{1}{4}(u_{ijk}^{n-1} + u_{i+p,j,k}^{n-1} + u_{i,j,k+r}^{n-1} + u_{i+p,j,k+r}^{n-1}) \\ u_{ijk}^{0qr} &= \frac{1}{4}(u_{ijk}^{n-1} + u_{i,j+q,k}^{n-1} + u_{i,j,k+r}^{n-1} + u_{i,j+q,k+r}^{n-1}). \end{aligned}$$

The components of the averaged gradient on S are approximated by 2D diamond cell approach in the othogonal planes which use the above defined values u_{ijk}^{pqr}, cf. [2, 3]. Then on S we get

$$\frac{1}{m(S)} \int_S \nabla u^{n-1}\, dx \approx \frac{u_{i+1,j,k}^{n-1} - u_{ijk}^{n-1}}{h_1}\mathbf{e}_1 + \frac{u_{ijk}^{1,1,0} - u_{ijk}^{1,-1,0}}{h_2}\mathbf{e}_2 + \frac{u_{ijk}^{1,0,1} - u_{ijk}^{1,0,-1}}{h_3}\mathbf{e}_3 \tag{14}$$

Applying similar formulas to the other voxel sides, we get the approximations $\nabla^{pqr} u_{ijk}^{n-1}$ of the gradient in the points x_{ijk}^{pqr} and we use the same expressions as in (11) to define the required quantities.

4. EXPERIMENTAL ORDER OF CONVERGENCE OF THE NUMERICAL METHODS

In order to inspect the convergence properties of the suggested methods, we determined the EOC for all three alternatives proposed in Sec. 3. . The analytical solution of the problem (1) is not known, so we used a modified problem with non-zero RHS.

$$\partial_t u - w_{con} \nabla g \nabla u - w_{dif} g \sqrt{\varepsilon^2 + |\nabla u|^2} \nabla . \frac{\nabla u}{\sqrt{\varepsilon^2 + |\nabla u|^2}} = f(u)\,. \tag{15}$$

We take the function $\tilde{u}(x, y, z, t) = tcos(\pi x)cos(\pi y)cos(\pi z)$ to be the analytical solution of the problem and from that we determine the term $f(\tilde{u})$. The function g is of the form $g(x, y, z) = cos^2(\pi x)cos^2(\pi y)cos^2(\pi z)$. The values of parameters were chosen as follows: $w_{con} = 1.0$, $w_{dif} = 1.0$, $\Omega = [-0.5, 0.5]^3$, $T = 0.16$, $N_1 = N_2 = N_3 = n$, $h_1 = h_2 = h_3 = h = 1/n$, $n = 10, 20, \ldots, 160$, $\varepsilon = h^2$, and we choose coupling $\tau \approx h^2$ which is typical for parabolic equations. The results for all methods are displayed in Tables 1–3 . We can see that the EOC is slightly better than 1 for all schemes which means that they are reliable and they can be used in practical applications based on solving the GSUBSURF equation.

n	τ	$L_\infty(I, L_2(\Omega))$ - error	EOC
10	0.04	2.267154e-2	
20	0.01	9.454992e-3	1.26173
40	0.0025	3.612228e-3	1.38819
80	0.000625	1.474722e-3	1.29245
160	0.00015625	6.524255e-4	1.17656

Table 1: Experimental order of convergence for diamond cell scheme

n	τ	$L_\infty(I, L_2(\Omega))$ - error	EOC
10	0.04	2.395713e-2	
20	0.01	9.878041e-3	1.27816
40	0.0025	3.782398e-3	1.38942
80	0.000625	1.531248e-3	1.30459
160	0.00015625	6.768897e-4	1.17772

Table 2: Experimental order of convergence for split diamond cell scheme

n	τ	$L_\infty(I, L_2(\Omega))$ - error	EOC
10	0.04	2.294529e-2	
20	0.01	9.546493e-3	1.26516
40	0.0025	3.636421e-3	1.39245
80	0.000625	1.478174e-3	1.29870
160	0.00015625	6.501010e-4	1.18508

Table 3: Experimental order of convergence for reduced diamond cell scheme

5. EXAMPLES OF PRACTICAL APPLICATIONS

The GSUBSURF method was extensively tested and used for segmentation of 3D images of embryogenesis coming from confocal laser microscopes. Being able to extract the shape of the individual cells, their nuclei or the shape of the whole embryo, we can perform various analyses of the process of embryonic development - determine the number of cells, volume of the embryo, density of cells, number and positions of cell divisions etc.

The reduced diamond cell variant was chosen as the most suitable one for practical computations because of its simplicity, precision comparable with the other methods and the lowest CPU time demands. Fig. 5 shows three examples of segmentation of zebrafish (danio rerio) embryo images. We segmented images of cell nuclei to get the shapes of all nuclei and cell membrane images to extract the shape of whole cells and whole embryo. All images were preprocessed by geodesic mean curvature flow filtering algorithm [6]. In all cases, we used time step $\tau = 0.1$ and voxel dimensions $h_1 = h_2 = h_3 = 1.0$. Further, $K = 1000$, $\varepsilon = 10^{-3}$, $w_{con} = 10.0$, and $w_{dif} = 5.0$, $w_{dif} = 0.1$, $w_{dif} = 2.0$ for nuclei, membrane and embryo segmentation, respectively. The edge detector was $g(s) = \frac{1}{1+Ks^2}$ for membrane and embryo segmentation, and $g(s) = G_\rho * \frac{1}{(1+Ks^2)^6}$, $\rho = 0.0001$ (G_ρ denotes the Gauss kernel) for nuclei segmentation. The segmentation seeds were represented by approximate cell centers detected by flux-based level set center detection algorithm [4]. For nuclei segmentation, all isosurfaces of the initial segmentation function were equal ellisoids centered in the detected cell center. For the whole embryo segmentation, the initial condition is obtained as the union of initial segmentation functions for all nuclei. For the membrane segmentation, we used the Voronoi diagram principle to construct the initial condition, see [7] for details.

ACKNOWLEDGEMENT

This work was partially supported by the European projects Embryomics and BioEmergences, the grants APVV-RPEU-0004-06, APVV-0351-07, APVV-LPP-0020-07 and the grant of VEGA 1/3321/06.

(Received October 20, 2008.)

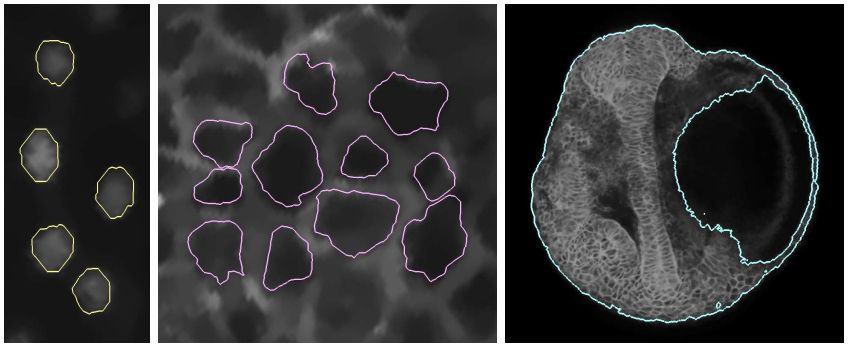

Figure 5: Examples of embryo image segmentation. On the left, segmentation of cell nuclei. In the middle, segmentation of cell shape. On the right, whole embryo segmentation.

REFERENCES

[1] S.Corsaro, K.Mikula, A.Sarti, F.Sgallari: Semi-implicit co-volume method in 3D image segmentation. SIAM Journal on Scientific Computing *28*, No. 6 (2006), 2248–2265.

[2] Y. Coudiere, J. P. Vila, P. Villedieu: Convergence rate of a finite volume scheme for a two-dimensional convection-diffusion problem, M2AN Math. Model. Numer. Anal. *33* (1999), 493–516.

[3] O. Drblíková, K. Mikula: Convergence Analysis of Finite Volume Scheme for Nonlinear Tensor Anisotropic Diffusion in Image Processing, SIAM Journal on Numerical Analysis *46*, No. 1 (2007) 37–60.

[4] P. Frolkovič, K. Mikula, N. Peyriéras, A.Sarti: A counting number of cells and cell segmentation using advection-diffusion equations. Kybernetika *43*, No. 6(2007), 817–829.

[5] P. Frolkovič, K. Mikula: Flux-based level set method: A finite volume method for evolving interfaces. Applied Num. Math. *57*, No. 4 (2007), 436–454.

[6] Z. Krivá, K.Mikula, N. Peyriéras, B.Rizzi, A.Sarti: 3D embryogenesis image filtering by non-linear partial differential equations. Submitted.

[7] K. Mikula, N. Peyriéras, M. Remešíková, A.Sarti: 3D embryogenesis image segmentation by the generalized subjective surface method using the finite volume technique. Proceedings of FVCA5 – 5th International Symposium on Finite Volumes for Complex Applications, Hermes Publ., Paris 2008.

[8] A. Sarti, G. Citti: Subjective surfaces and Riemannian mean curvature flow graphs. Acta Math. Univ. Comenian. (N.S.) *70* (2000), 85–103.

[9] A. Sarti, R. Malladi, J.A. Sethian: Subjective Surfaces: A Method for Completing Missing Boundaries. Proc. Nat. Acad. Sci. mi 12, No. 97 (2000), 6258–6263.

[10] A. Sarti, R. Malladi, J.A. Sethian: Subjective Surfaces: A Geometric Model for Boundary Completion, International Journal of Computer Vision, mi 46, No. 3 (2002), 201–221.

Karol Mikula, Mariana Remešíková, Depatment of Mathematics, Slovak University of Technology, Radlinskeho 11, 81368 Bratilsava, Slovakia.
e-mails: mikula@math.sk, remesikova@math.sk