Numer. Math. (2003) 93: 675–695
Digital Object Identifier (DOI) 10.1007/s002110100374

Numerische Mathematik

Semi-implicit complementary volume scheme for solving level set like equations in image processing and curve evolution

Angela Handlovičová[1], **Karol Mikula**[1], **Fiorella Sgallari**[2]

[1] Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava, Slovakia; e-mail: {angela,mikula}@vox.svf.stuba.sk
[2] Department of Mathematics, University of Bologna, Piazza di Porta S. Donato 5, 40127 Bologna, Italy; e-mail: sgallari@dm.unibo.it

Received April 25, 2000 / Revised version received June 11, 2001 /
Published online November 15, 2001 –

Summary. We introduce linear semi-implicit complementary volume numerical scheme for solving level set like nonlinear degenerate diffusion equations arising in image processing and curve evolution problems. We study discretization of image selective smoothing equation of mean curvature flow type given by Alvarez, Lions and Morel ([3]). Solution of the level set equation of Osher and Sethian ([26], [30]) is also included in the study. We prove L_∞ and $W^{1,1}$ estimates for the proposed scheme and give existence of its (generalized) solution in every discrete time-scale step. Efficiency of the scheme is given by its linearity and stability. Preconditioned iterative solvers are used for computing arising linear systems. We present computational results related to image processing and plane curve evolution.

Mathematics Subject Classification (1991): 65U10

1 Introduction

The aim of this paper is to present and study numerical scheme for solving nonlinear diffusion equations arising in a wide range of applications as image processing and computer vision, phase transition, crystal growth, flame propagation, superconductivity, etc. and which are related to the curve and surface evolution problems. The scheme is based on linear semi-implicit approximation in time-scale and on the so called complementary volume method in space. We study discretization of nonlinear degenerate diffusion

equation of mean curvature flow type suggested by Alvarez, Lions and Morel in [3] for image selective smoothing

$$u_t - g(|\nabla G_\sigma * u|)|\nabla u|\nabla.(\frac{\nabla u}{|\nabla u|}) = 0, \tag{1}$$

where $u(t,x)$ is unknown function defined in $Q_T \equiv I \times \Omega$. The equation is accompanied with zero Neumann boundary conditions and initial condition

$$\frac{\partial u}{\partial \nu} = 0 \quad \text{on } I \times \partial\Omega, \tag{2}$$

$$u(0,x) = u^0(x) \quad \text{in } \Omega \tag{3}$$

where ν is unit normal to the boundary of Ω. We assume that $\Omega \subset I\!R^d$ is a bounded rectangular domain (such situation is the most typical in image processing), $I = [0,T]$ is a so called scale (time) interval, and

(4) $g : I\!R_0^+ \to I\!R^+$ is a nonincreasing function, $g(\sqrt{s})$ is smooth, $g(0) = 1$, and we admit $g(s) \to 0$ for $s \to \infty$,

(5) $G_\sigma \in C^\infty(I\!R^d)$ is a smoothing kernel (e.g. Gauss function), with $\int_{I\!R^d} G_\sigma(x)dx = 1$, $\int_{I\!R^d} |\nabla G_\sigma| dx \le C_\sigma$, $G_\sigma(x) \to \delta_x$ for $\sigma \to 0$, δ_x is the Dirac measure at point x,

(6) $u^0 \in L_\infty(\Omega)$,

and

$$\nabla G_\sigma * u = \int_{I\!R^d} \nabla G_\sigma(x-\xi)\tilde{u}(\xi)d\xi, \tag{7}$$

where $\tilde{u}$ is an extension of u to $I\!R^d$ given by periodic reflection through the boundary of Ω and for which $\|\tilde{u}\|_{L_2(I\!R^d)} \le C\|u\|_{L_2(\Omega)}$. In particular case $g \equiv 1$ we are dealing with numerical approximation of the Osher-Sethian level set equation ([26], [30], [31])

$$u_t - |\nabla u|\nabla.(\frac{\nabla u}{|\nabla u|}) = 0 \tag{8}$$

moving all level sets of u by normal mean curvature field.

In image processing, equations like (1) arise in nonlinear filtration, edge detection, image enhancement etc., when we are dealing with geometrical features of the image like silhouette of object corresponding to level line of image intensity function. The initial condition $u^0(x)$ represents greylevel intensity of the processed image and the solution $u(t,x)$ of (1) gives a family of *scaled* (filtered, smoothed) versions of $u^0(x)$. In image processing, the parameter t is understood as *scale*, in other application t represents time as

it is usual in evolutionary problems. The special case of (1), equation (8) has attracted a lot of attention in past years due to its applicability to geometrical problems, namely to the motion of hypersurfaces by mean curvature. Concerning image processing, equation (8) fulfills the so-called morphological principle which means invariance of image analysis to contrast changes which plays an important role in axiomatization of computer vision theories ([1], [2], [18]). Applying the level set equation to image yields the *intrinsic Gaussian smoothing* of its level lines in 2D or level surfaces in 3D ([8], [9], [23], [13], [24], [25]) which is equivalent to the motion of hypersurfaces by normal mean curvature field. Moreover, the level sets move independently on each other. One can interpret this fact as diffusion of the image in the direction tangential to level lines with no diffusion across them. Such idea has been used in [3], where model (1) has been suggested for computational image and shape analysis. The "stopping term" $g(|\nabla G_\sigma * u|)$ in (1) is used to strongly slow down the motion of the silhouettes which are at the same time un-spurious edges. The regions between them are smoothed by the mean curvature flow. Thus the equation (1) can be used successively for image selective smoothing with preserving edge positions in a similar way like well-known Perona-Malik equation ([27], [5]). The Perona-Malik equation has not such straightforward geometrical interpretation and its solvability is rather difficult problem ([16]). The existence of unique viscosity solution of (1) or (8) is given in [3], [11], [6].

The rest of this paper is organized as follows. In Sect. 2 we present linear semi-implicit complementary volume discretization of (1). It leads to solving of linear algebraic systems and, at the same time, gives good stability properties of numerical approximation of this strongly nonlinear problem. In Sect. 3 we derive L_∞ as well as $W^{1,1}$ stability estimates and discuss existence of a (generalized) solution of the scheme. In discussion on numerical results, in Sect. 4, we present computations in image processing and plane curve evolution driven by curvature. We also present some computational time statistics related to usage of preconditioned iterative solvers in every discrete time-scale step of the method.

2 Semi-implicit complementary volume scheme

In this section we first introduce semi-implicit discretization of equation (1) in time-scale and then give fully discrete scheme based on complementary volume method. Choosing $N \in I\!N$ we get a uniform discrete time-scale increment $\tau = \frac{T}{N}$. We replace the time-scale derivative in (1) by backward difference and the nonlinear terms of equations are treated from the previous scale step while the linear ones are considered on the current scale level. This will lead to linear problems at each discrete scale level; for other semi-

implicit approaches in image processing we refer to [5], [3], [14], [4], [33], [29], [15], [22]. Let us denote

$$g^{n-1} := g(|\nabla G_\sigma * u^{n-1}|) \tag{9}$$

for which

$$g^{n-1} \geq \nu_\sigma > 0$$

due to smoothing properties of convolution ([5], [3], [14]). Then our semi-discrete in time-scale scheme for solving equation (1) is given as follows:
Let $N \in I\!N, \tau = \frac{T}{N}$ and $\sigma > 0$ be fixed numbers and u^0 be given by (3). *Then, for every $n = 1, \ldots N$, we look for a function u^n, solution of the equation*

$$\frac{1}{g^{n-1}|\nabla u^{n-1}|}\frac{u^n - u^{n-1}}{\tau} - \nabla.\left(\frac{\nabla u^n}{|\nabla u^{n-1}|}\right) = 0. \tag{10}$$

Since in general situation there can be zero in denominator of (10), we will regularize this equation in a sense of Evans and Spruck ([11]). The precise formulation of two our regularizations as well as a behaviour of solutions for regularization parameters tending to zero will be given below. Now we are going to full discretization of (10) by means of complementary volume method.

In image processing, a discrete image is given on a structure of pixels/voxels with rectangular shape, in general. In the complementary volume method, approximation of solution is assumed to be a piecewise linear function. The values of discrete image intensity are considered as approximations of continuous image intensity function in centers of pixels. These centers of pixels will correspond to the nodes of a triangulation. We can get such triangulation simply by connecting the centers of pixels by new rectangular mesh and then dividing every rectangle into two triangles (or six tetrahedras in 3D case). In other applications, e.g. in phase transition, the triangulation of computational domain can be more complicated. We just assume that it has no interior angle larger than $\pi/2$. Now, we will define some quantities, which will be used in description of the fully discrete scheme (see also [32]). In complementary volume method, together with the triangulation also the so-called dual mesh is used. The dual mesh consists of cells V_i (called also complementary volumes, control volumes or co-volumes) associated with the ith node, $i = 1, \ldots, M$, of the given triangulation $\mathcal{T}_h$. The co-volume V_i is bounded by the lines (planes in 3D case) that bisect and are perpendicular to the edges emanating from the node. Let us note, that in image processing the dual mesh again corresponds to pixel structure. The computational domain Ω then corresponds to image domain minus outer half of every boundary pixel - see Fig. 1.

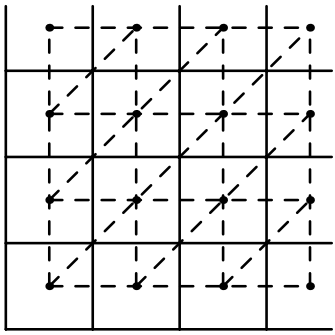

Fig. 1. The image pixels (solid lines) corresponding to the dual mesh for complementary volume method. Triangulation (dashed lines) for the complementary volume method with nodes (round points) corresponding to centers of pixels

We will denote the edge of $\mathcal{T}_h$ connecting the ith node to the jth by σ_{ij} and its length by h_{ij}. We denote by $\mathcal{E}_{ij}$ the set of simplices having σ_{ij} as an edge, i.e., $\mathcal{E}_{ij} = \{T \in \mathcal{T}_h | \sigma_{ij} \subset T\}$. Let e_{ij} denote the co-edge (co-plane) that is perpendicular bisector of σ_{ij}. For each $T \in \mathcal{E}_{ij}$ let c_{ij}^T be the length (area) of the portion of e_{ij} that is in T, i.e., $c_{ij}^T = |e_{ij} \cap T|$. For each node of $\mathcal{T}_h$ let C_i denote the set of nodes connected to the ith node by an edge.

Given a triangulation $\mathcal{T}_h$, we define the set $V_h \subset V$ of continuous piecewise linear functions, i.e., $V_h = V_h(\mathcal{T}_h) := \{v \in C^0(\bar{\Omega}) | v_{|T} \in \mathcal{P}_1 \text{ for all } T \in \mathcal{T}_h\}$. Then $|\nabla u_h|$, $u_h \in V_h$ has a constant value on every simplex $T \subset \mathcal{T}_h$. We will denote that value by $|\nabla u_T|$. For any $u_h \in V_h$ we will use notation $u_i := u_h(x_i)$ where x_i is ith node of triangulation. Let $u_h^0 = I_h(u^0) \in V_h(\mathcal{T}_h)$ be the nodal interpolant of u^0, the initial function for the computational method.

Let $\mathcal{N}_i$ be the set of simplices that have the ith node as a vertex. Let $\delta > 0$, $\varepsilon > 0$. Due to regularization we will use sets $\mathcal{N}_{i,0} = \{T \in \mathcal{N}_i, |\nabla u_T| = 0\}$, $\mathcal{N}_{i,1} = \{T \in \mathcal{N}_i, |\nabla u_T| > 0\}$ and $\mathcal{N}_{i,0}^\delta = \{T \in \mathcal{N}_i, |\nabla u_T| \leq \delta\}$, $\mathcal{N}_{i,1}^\delta = \{T \in \mathcal{N}_i, |\nabla u_T| > \delta\}$. For any $u_h \in V_h$ and each $T \in \mathcal{T}_h$ we define

$$|\nabla u_T|_\delta = \begin{cases} |\nabla u_T| \text{ if } T \in \mathcal{N}_{i,1}^\delta \\ \delta \text{ if } T \in \mathcal{N}_{i,0}^\delta, \end{cases} \tag{11}$$

and for any function $u_h \in V_h$ we denote

$$|\nabla u_h|_\varepsilon = \sqrt{\varepsilon^2 + |\nabla u_h|^2}. \tag{12}$$

The difference between regularizations (11) and (12), respectively, is that in the first case we replace by δ only vanishing or small gradients, while in the second case we add small regularization parameter to gradient everywhere. Concerning theoretical analysis of fully discrete scheme we get estimates

independent of the regularization parameter ε, δ, respectively, and thus holding also in the limit case when ε, δ tends to 0 (see Sect. 3). Such limits will be called generalized solutions of the scheme. Concerning computational point of view, we have found preconditioned iterative linear solvers, efficiency of which is not affected even when ε or δ are very small (see Sect. 4).

In order to derive complementary volume spatial discretization we integrate (10) over a co-volume V_i

$$\int_{V_i} \frac{u^n - u^{n-1}}{g^{n-1}|\nabla u^{n-1}|\tau} dx = \int_{V_i} \nabla . \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) dx. \tag{13}$$

For the right hand side using divergence theorem we get

$$\begin{aligned} \int_{V_i} \nabla . \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) dx &= \int_{\partial V_i} \frac{1}{|\nabla u^{n-1}|} \frac{\partial u^n}{\partial \nu} ds \\ &= \sum_{j \in C_i} \int_{e_{ij}} \frac{1}{|\nabla u^{n-1}|} \frac{\partial u^n}{\partial \nu} ds. \end{aligned} \tag{14}$$

If $u_h^n \in V_h$ is continuous piecewise linear function on triangulation $\mathcal{T}_h$ then

$$\sum_{j \in C_i} \int_{e_{ij}} \frac{1}{|\nabla u_h^{n-1}|} \frac{\partial u_h^n}{\partial \nu} ds = \sum_{j \in C_i} \left(\sum_{T \in \mathcal{E}_{ij}} \frac{c_{ij}^T}{|\nabla u_T^{n-1}|} \right) \frac{u_j^n - u_i^n}{h_{ij}}. \tag{15}$$

The complementary volume method approximates the left hand side of (13) by

$$\frac{|V_i|(u_i^n - u_i^{n-1})}{\tau g(|\nabla u_i^c|)|\nabla u_i^{n-1}|} \tag{16}$$

where $|\nabla u_i^{n-1}|$, $|\nabla u_i^c|$ denote an approximation of the gradient of u^{n-1} and the gradient of the result of convolution $u^c \equiv G_\sigma * u^{n-1}$, respectively, in the co-volume V_i. For that we have chosen the average value of gradients in the co-volume, i.e. for any $u_h \in V_h$ we define

$$|\nabla u_i| = \sum_{T \in \mathcal{N}_i} \frac{|T \cap V_i|}{|V_i|} |\nabla u_T| \tag{17}$$

Then if we denote by

$$|\nabla u_i|_\delta = \sum_{T \in \mathcal{N}_i} \frac{|T \cap V_i|}{|V_i|} |\nabla u_T|_\delta, \;\; |\nabla u_i|_\varepsilon = \sum_{T \in \mathcal{N}_i} \frac{|T \cap V_i|}{|V_i|} |\nabla u_T|_\varepsilon, \tag{18}$$

and

$$b_i^{n-1} = \frac{|V_i|}{g(|\nabla u_i^c|)|\nabla u_i^{n-1}|_\delta}, \tag{19}$$

$$a_{ij}^{n-1} = \frac{1}{h_{ij}} \sum_{T \in \mathcal{E}_{ij}} \frac{c_{ij}^T}{|\nabla u_T^{n-1}|_\delta}, \tag{20}$$

or

$$b_i^{n-1} = \frac{|V_i|}{g(|\nabla u_i^c|)|\nabla u_i^{n-1}|_\varepsilon}, \tag{21}$$

$$a_{ij}^{n-1} = \frac{1}{h_{ij}} \sum_{T \in \mathcal{E}_{ij}} \frac{c_{ij}^T}{|\nabla u_T^{n-1}|_\varepsilon}, \tag{22}$$

we can write

Linear semi-implicit fully discrete complementary volume scheme for solving equation (1): *For* $n = 1, \ldots, N$ *we look for* $u_i^n, i = 1, \ldots, M$, *satisfying*

$$b_i^{n-1}(u_i^n - u_i^{n-1}) + \tau \sum_{j \in C_i} a_{ij}^{n-1}(u_i^n - u_j^n) = 0. \tag{23}$$

If coefficients of (23) are regularized by (11), i.e. we use (19)-(20), the solution $(u_1^n, \ldots, u_M^n)$ of this linear system will be denoted by $u_h^{\delta,n}$. If coefficients of (23) are regularized by (12), i.e. we use (21)-(22), the solution will be denoted by $u_h^{\varepsilon,n}$. In the next section we will prove some properties of the solution of (23) which exist due to following assertions.

Proposition 1. Let coefficients of the scheme (23) be given by (19)-(20). Then there exists unique solution $u_h^{\delta,n} = (u_1^n, \ldots, u_M^n)$ of the scheme (23) for any $\delta > 0$, $n = 1, \ldots, N$.

Proof. From definition (20), it follows that off diagonal elements $-\tau a_{ij}^{n-1}$, $j \in C_i$, of the system (23) are symmetric. The positive term b_i^{n-1} given by (19) affects only diagonal which is equal to $b_i^{n-1} + \tau \sum_{j \in C_i} a_{ij}^{n-1}$. Thus, the matrix of the system (23) is symmetric and diagonally dominant M-matrix which imply that it always has unique solution.

By the same argument we get also

Proposition 2. Let coefficients of the scheme (23) be given by (21)-(22). Then there exists unique solution $u_h^{\varepsilon,n} = (u_1^n, \ldots, u_M^n)$ of the scheme (23) for any $\varepsilon > 0$, $n = 1, \ldots, N$.

Remark. With ε or δ very small the diagonal dominance of the system (23) can be weak, thus one has to be carefull with a choice of proper solver.

Before solving (23) we have to put $|\nabla u_i^c|$ into (19) or (21). For that goal, we use strategy proposed in ([4]) which is natural also for the complementary volume method. Using the Gauss function as smoothing kernel G_σ one can replace the term $G_\sigma * u^{n-1}$ by solving the linear heat equation for time σ with initial datum given by u^{n-1}. This linear equation is solved numerically at the same grid by just one implicit time step of length σ. Thus, as a realization of convolution we look for a function u^c which is a solution of linear heat equation discretized in time by backward Euler method with step σ

$$\frac{u^c - u^{n-1}}{\sigma} = \Delta u^c \tag{24}$$

where Δ denotes Laplace operator. The discrete approximation of u^c, function $u_h^c \in V_h$, is found by the same idea as given in (13)-(16) applied to equation (24), i.e. we solve (23) with u_i^n replaced by u_i^c and with $b_i^{n-1} \equiv b_i = |V_i|$, $a_{ij}^{n-1} \equiv a_{ij} = \frac{|e_{ij}|}{h_{ij}}$. Then $|\nabla u_i^c|$ is computed by (17) and we put this value into the Perona-Malik function g in (19) or (21).

3 Stability estimates and existence of generalized solution

Before analysis of our linear semi-implicit scheme, let us note another possible approach for time-scale discretization of (1) based on interesting approximation of degenerate diffusion term given in [32]. Following idea of Walkington one could, instead of (10), use the following implicit nonlinear semi-discretization in time-scale

$$\frac{1}{g^{n-1}|\nabla u^{n-1}|}\frac{u^n - u^{n-1}}{\tau} - 2\nabla.\left(\frac{\nabla u^n}{|\nabla u^n| + |\nabla u^{n-1}|}\right) = 0 \tag{25}$$

where averaging of gradient term from previous and current time-scale step is considred in denominator of divergence term. Such scheme leads to $W^{1,1}$ estimate, i.e. estimate on decay of total variation of discrete time-scale solutions. It is a basic property of the flow by mean curvature and solution of level set equation as well and can be interpreted as a curve shortening property ([9], [21], [20], [24], [25]). Any reasonable numerical approximation should also respect this fact. Following [32] one can multiply (25) by $u^n - u^{n-1}$ and integrate it over Ω. Then by integration per-partes and using zero Neumann bounadary conditions one gets

$$\int_\Omega \frac{(u^n - u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}dx + 2\int_\Omega \frac{\nabla u^n.(\nabla u^n - \nabla u^{n-1})}{|\nabla u^{n-1}| + |\nabla u^n|}dx = 0. \tag{26}$$

Using the relation

$$2a(a-b) = a^2 - b^2 + (a-b)^2 \tag{27}$$

where a, b are arbitrary real numbers, and by a simple manipulations related to the sum in denominator

$$\begin{aligned} &\int_\Omega \frac{(u^n-u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}dx + \int_\Omega \frac{|\nabla u^n|^2-|\nabla u^{n-1}|^2}{|\nabla u^{n-1}|+|\nabla u^n|}dx + \\ &+ \int_\Omega \frac{|\nabla u^n-\nabla u^{n-1}|^2}{|\nabla u^{n-1}|+|\nabla u^n|}dx = \int_\Omega \frac{(u^n-u^{n-1})^2}{\tau g^{n-1}|\nabla u^{n-1}|}dx + \\ &+ \int_\Omega \frac{|\nabla u^n-\nabla u^{n-1}|^2}{|\nabla u^{n-1}|+|\nabla u^n|}dx + \left(\int_\Omega |\nabla u^n|dx - \int_\Omega |\nabla u^{n-1}|dx\right) = 0 \end{aligned} \tag{28}$$

which means that

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^{n-1}\|_{L_1(\Omega)} \tag{29}$$

and by recursion

$$\|\nabla u^n\|_{L_1(\Omega)} \le \|\nabla u^0\|_{L_1(\Omega)}, \;\; 1 \le n \le N \tag{30}$$

which represents the important stability property of this nonlinear scheme.

However, the previous scheme leads (after any spatial discretization) to solving of nonlinear system of equations in each discrete time-scale level which is rather non-efficient approach. In order to have convergence, which is however very slow, one has to use fixed point-like nonlinear iterations; faster possibilities like Newton's method has no guarantee to converge ([32]) and are also rather complicated from implementation point of view for this type of problem.

In regard to analysis, stability and efficiency of the method, our contribution is that we get existence of generalized discrete solution (limit when regularization parameter tends to 0) in subsequent time-scale steps and decay of total variation of such discrete solutions also for the scheme (10), or more precisely for fully discrete scheme (23) which is much more simple and efficient since it is linear. As linear, it allows to use preconditioned iterative linear solvers at every time-scale level and achieve in such way fast and stable solution (we present some CPU time statistics in Sect. 4).

In order to derive an analogue of the estimate (30) for our fully discrete scheme we use the following result (see e.g. [32])

Lemma. *Let $\mathcal{T}_h$ be a two-dimensional mesh having simplicies with interior angles not exceeding $\pi/2$ and let $u, v \in V_h$, and w be piecewise constant on $\mathcal{T}_h$. Then*

$$\int_\Omega w\nabla u.\nabla v dx = \sum_{i=1}^{M} \left(\sum_{j\in C_i} \alpha_{ij}(w)(u_i - u_j) \right) v_i, \tag{31}$$

where $\alpha_{ij}(w) = \frac{1}{h_{ij}} \sum_{T \in \mathcal{E}_{ij}} w_T c_{ij}^T$ *and* w_T *denotes value of* w *in* $T \in \mathcal{T}_h$.

Then we can prove the following assertions.

Theorem 1. *There exists limit* u_h^n *of a subsequence of* $u_h^{\delta,n}$ *for* $\delta \to 0$ *where* $u_h^{\delta,n}$ *is the solution of the scheme (23) with coefficients given by (19)–(20). Moreover for this generalized solution* u_h^n *the following estimates hold*

$$\|u_h^n\|_{L_\infty(\Omega)} \le \|u_h^0\|_{L_\infty(\Omega)}, \ \ \|\nabla u_h^n\|_{L_1(\Omega)} \le \|\nabla u_h^0\|_{L_1(\Omega)}, \ \ 1 \le n \le N. \tag{32}$$

Proof. Let us rewrite (23) in the form

$$u_i^n + \frac{\tau}{b_i^{n-1}} \sum_{j \in C_i} a_{ij}^{n-1}(u_i^n - u_j^n) = u_i^{n-1} \tag{33}$$

and let $\max u_h^{\delta,n} = \max(u_1^n, \dots, u_M^n)$ be achieved in the ith node. Then the whole second term on the left hand side is nonnegative and thus value $u_i^n \le u_i^{n-1} \le \max(u_1^{n-1}, \dots, u_M^{n-1})$. In the same way we can prove the relations for minima and together we have

$$\min\ u_i^0 \le \min\ u_i^n \le \max\ u_i^n \le \max\ u_i^0, \ \ n \le N \tag{34}$$

which imply

$$\|u_h^{\delta,n}\|_{L_\infty(\Omega)} \le \|u_h^0\|_{L_\infty(\Omega)}, \ \ 1 \le n \le N. \tag{35}$$

Since estimate (35) is independent on δ we can choose convergent subsequence of $u_h^{\delta,n}$ as $\delta \to 0$. The limit of this subsequence we denote by $u_h^n \in V_h$ and it is clear that it fulfills the first estimate of the Theorem. To get the second estimate of the Theorem, let us multiply (23) by $u_i^n - u_i^{n-1}$ and sum it over all nodes. We get

$$\sum_{i=1}^{M} b_i^{n-1} \frac{(u_i^n - u_i^{n-1})^2}{\tau} + \sum_{i=1}^{M} \sum_{j \in C_i} a_{ij}^{n-1}(u_i^n - u_j^n)(u_i^n - u_i^{n-1}) = 0 \tag{36}$$

using definition of a_{ij}^{n-1} and (31) we obtain

$$\sum_{i=1}^{M} b_i^{n-1} \frac{(u_i^n - u_i^{n-1})^2}{\tau} + \int_\Omega \frac{\nabla u_h^{\delta,n} . \nabla (u_h^{\delta,n} - u_h^{n-1})}{|\nabla u_h^{n-1}|_\delta} dx = 0. \tag{37}$$

We denote second term of this equation by II and further we denote

$$\Omega_0 = \{T \in \mathcal{T}_h; |\nabla u_T^{n-1}| \le \delta\}$$

$$\Omega_1 = \{T \in \mathcal{T}_h; |\nabla u_T^{n-1}| > \delta\}.$$

Using the relation (27) we can rewrite the second term in the equation (37) as follows

$$II = \int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx - \int_{\Omega_0} \frac{\nabla u_h^{\delta,n}.\nabla u_h^{n-1}}{\delta} dx$$
$$+\frac{1}{2}\int_{\Omega_1} \frac{|\nabla u_h^{\delta,n}|^2 - |\nabla u_h^{n-1}|^2 + |\nabla u_h^{\delta,n} - \nabla u_h^{n-1}|^2}{|\nabla u_h^{n-1}|} dx. \tag{38}$$

Since the following identity holds

$$|\nabla u_h^{\delta,n} - \nabla u_h^{n-1}|^2 = (|\nabla u_h^{\delta,n}| - |\nabla u_h^{n-1}|)^2$$
$$+ \left| \frac{\nabla u_h^{\delta,n}}{|\nabla u_h^{\delta,n}|} - \frac{\nabla u_h^{n-1}}{|\nabla u_h^{n-1}|} \right|^2 |\nabla u_h^{\delta,n}||\nabla u_h^{n-1}|,$$

we get

$$II = \int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx - \int_{\Omega_0} \frac{\nabla u_h^{\delta,n}.\nabla u_h^{n-1}}{\delta} dx +$$
$$+\frac{1}{2}\int_{\Omega_1} \frac{|\nabla u_h^{\delta,n}|^2 - |\nabla u_h^{n-1}|^2 - (|\nabla u_h^{\delta,n}| - |\nabla u_h^{n-1}|)^2}{|\nabla u_h^{n-1}|} dx +$$
$$+ \int_{\Omega_1} \frac{(|\nabla u_h^{\delta,n}| - |\nabla u_h^{n-1}|)^2}{|\nabla u_h^{n-1}|} dx + \frac{1}{2}\int_{\Omega_1} \left| \frac{\nabla u_h^{\delta,n}}{|\nabla u_h^{\delta,n}|} - \frac{\nabla u_h^{n-1}}{|\nabla u_h^{n-1}|} \right|^2 |\nabla u_h^{\delta,n}| dx.$$

Because the first term of (37) and last two terms in the previous expression are nonnegative, using Cauchy-Schwartz and then Young inequality we obtain

$$\int_{\Omega_1} |\nabla u_h^{\delta,n}| dx + \int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx \le \int_{\Omega_1} |\nabla u_h^{n-1}| dx + \int_{\Omega_0} |\nabla u_h^{\delta,n}| dx \le$$
$$\int_{\Omega_1} |\nabla u_h^{n-1}| dx + \frac{1}{2}\int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx + \frac{1}{2}\int_{\Omega_0} \delta dx$$

from where

$$\frac{1}{2}\int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx + \int_{\Omega_1} |\nabla u_h^{\delta,n}| dx \le \int_{\Omega_1} |\nabla u_h^{n-1}| dx + \frac{1}{2}\delta|\Omega_0|. \tag{39}$$

If we use again

$$\int_{\Omega_0} |\nabla u_h^{\delta,n}| dx \le \frac{1}{2}\int_{\Omega_0} \frac{|\nabla u_h^{\delta,n}|^2}{\delta} dx + \frac{1}{2}\delta|\Omega_0| \tag{40}$$

we obtain

$$\|\nabla u_h^{\delta,n}\|_{L_1(\Omega)} \le \|\nabla u_h^{n-1}\|_{L_1(\Omega)} + \delta|\Omega_0|. \tag{41}$$

Let $u_h^{\delta,n}$ be subsequence converging to u_h^n corresponding to vector $u_h^n = (u_{h,1}^n, u_{h,2}^n, \ldots, u_{h,M}^n)$. It is clear that there exists subsequence of the previous one for which $|\nabla u_T^{\delta,n}| \to |\nabla u_T^n|, \forall T \in \mathcal{T}_h$ as $\delta \to 0$. Thus $\|\nabla u_h^{\delta,n}\|_{L_1(\Omega)} \to \|\nabla u_h^n\|_{L_1(\Omega)}$ for $\delta \to 0$. From the estimate (41) then follows that $\|\nabla u_h^n\|_{L_1(\Omega)} \leq \|\nabla u_h^{n-1}\|_{L_1(\Omega)}$ which gives the second estimate of the Theorem.

Theorem 2. *There exists limit u_h^n of a subsequence of $u_h^{\varepsilon,n}$ for $\varepsilon \to 0$ where $u_h^{\varepsilon,n}$ is the solution of the scheme (23) with coefficients given by (21)-(22). Moreover for this generalized solution u_h^n the following estimates hold*

$$\|u_h^n\|_{L_\infty(\Omega)} \leq \|u_h^0\|_{L_\infty(\Omega)}, \quad \|\nabla u_h^n\|_{L_1(\Omega)} \leq \|\nabla u_h^0\|_{L_1(\Omega)}, \quad 1 \leq n \leq N. \tag{42}$$

Proof. In the same way as in the proof of Theorem 1 we get

$$\|u_h^{\varepsilon,n}\|_{L_\infty(\Omega)} \leq \|u_h^0\|_{L_\infty(\Omega)}, \quad 1 \leq n \leq N \tag{43}$$

and from there the first estimate of this Theorem. Now, we prove the second estimate. Let us again multiply (23) by $u_i^n - u_i^{n-1}$ and sum it over all nodes. As in the previous proof we obtain

$$\sum_{i=1}^{M} b_i^{n-1} \frac{(u_i^n - u_i^{n-1})^2}{\tau} + \int_\Omega \frac{\nabla u_h^{\varepsilon,n} . \nabla (u_h^{\varepsilon,n} - u_h^{n-1})}{|\nabla u_h^{n-1}|_\varepsilon} dx = 0. \tag{44}$$

Let us use the notation

$$\nabla_\varepsilon v = (v_x, v_y, \varepsilon), \quad \nabla_0 v = (v_x, v_y, 0).$$

where v_x, v_y denote partial derivatives of a function v in 2D case (in 3D analogously). Then

$$\begin{aligned} &|\nabla_\varepsilon u_h^{n-1}| = |\nabla u_h^{n-1}|_\varepsilon, \\ &|\nabla_0 u_h^{\varepsilon,n}| = |\nabla u_h^{\varepsilon,n}|, \quad \nabla_\varepsilon u_h^{n-1} . \nabla_0 u_h^{\varepsilon,n} = \nabla u_h^{\varepsilon,n} . \nabla u_h^{n-1}. \end{aligned} \tag{45}$$

We again denote second term of equation (44) by II. Using the relation (27) we have

$$II = \frac{1}{2} \int_\Omega \frac{|\nabla u_h^{\varepsilon,n}|^2 - |\nabla u_h^{n-1}|^2 + |\nabla u_h^{\varepsilon,n} - \nabla u_h^{n-1}|^2}{|\nabla u_h^{n-1}|_\varepsilon} dx. \tag{46}$$

We can compute

$$\begin{gathered} |\nabla u_h^{\varepsilon,n} - \nabla u_h^{n-1}|^2 = |\nabla u_h^{\varepsilon,n}|^2 - 2|\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon + \\ |\nabla u_h^{n-1}|_\varepsilon^2 + 2|\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon - 2\nabla u_h^{\varepsilon,n} \cdot \nabla u_h^{n-1} - \varepsilon^2 = \\ (|\nabla u_h^{\varepsilon,n}| - |\nabla u_h^{n-1}|_\varepsilon)^2 + \left(2 - \frac{2\nabla u_h^{\varepsilon,n} . \nabla u_h^{n-1}}{|\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon}\right) |\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon - \varepsilon^2 \end{gathered}$$

and using (45) we get

$$2 - \frac{2\nabla u_h^{\varepsilon,n} \cdot \nabla u_h^{n-1}}{|\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon} = \left| \frac{\nabla_0 u_h^{\varepsilon,n}}{|\nabla_0 u_h^{\varepsilon,n}|} - \frac{\nabla_\varepsilon u_h^{n-1}}{|\nabla_\varepsilon u_h^{n-1}|} \right|^2$$

which together gives us

$$II = \tfrac{1}{2}\int_\Omega \frac{|\nabla u_h^{\varepsilon,n}|^2 - |\nabla u_h^{n-1}|_\varepsilon^2}{|\nabla u_h^{n-1}|_\varepsilon} dx + \tfrac{1}{2}\int_\Omega \frac{(|\nabla u_h^{\varepsilon,n}| - |\nabla u_h^{n-1}|_\varepsilon)^2}{|\nabla u_h^{n-1}|_\varepsilon} dx +$$
$$+ \tfrac{1}{2}\int_\Omega \left| \frac{\nabla_0 u_h^{\varepsilon,n}}{|\nabla_0 u_h^{\varepsilon,n}|} - \frac{\nabla_\varepsilon u_h^{n-1}}{|\nabla_\varepsilon u_h^{n-1}|} \right|^2 |\nabla u_h^{\varepsilon,n}| dx$$

and finally

$$\sum_{i=1}^{M} b_i^{n-1} \frac{(u_i^n - u_i^{n-1})^2}{\tau} + \int_\Omega \frac{(|\nabla u_h^{\varepsilon,n}| - (|\nabla u_h^{n-1}|_\varepsilon)^2}{|\nabla u_h^{n-1}|_\varepsilon} dx +$$
$$\tfrac{1}{2}\int_\Omega \left| \frac{\nabla_0 u_h^{\varepsilon,n}}{|\nabla_0 u_h^{\varepsilon,n}|} - \frac{\nabla_\varepsilon u_h^{n-1}}{|\nabla_\varepsilon u_h^{n-1}|} \right|^2 |\nabla u_h^{\varepsilon,n}| dx +$$
$$\tfrac{1}{2}\int_\Omega \frac{|\nabla u_h^{\varepsilon,n}|^2 - |\nabla u_h^{n-1}|_\varepsilon^2 - (|\nabla u_h^{\varepsilon,n}| - |\nabla u_h^{n-1}|_\varepsilon)^2}{|\nabla u_h^{n-1}|_\varepsilon} dx = 0.$$

Since the first three terms are positive we have for the last one

$$\frac{1}{2}\int_\Omega \frac{2|\nabla u_h^{\varepsilon,n}||\nabla u_h^{n-1}|_\varepsilon - 2|\nabla u_h^{n-1}|_\varepsilon^2}{|\nabla u_h^{n-1}|_\varepsilon} dx \le 0$$

from where

$$\int_\Omega |\nabla u_h^{\varepsilon,n}| dx \le \int_\Omega |\nabla u_h^{n-1}|_\varepsilon dx = \int_\Omega \sqrt{|\nabla u_h^{n-1}|^2 + \varepsilon^2} dx$$
$$\le \int_\Omega |\nabla u_h^{n-1}| dx + \varepsilon|\Omega|,$$

holding for any $\varepsilon > 0$. The rest of the proof uses the same arguments as in the end of the proof of Theorem 1.

4 Discussion on numerical results

This Section is devoted to discussion on numerical computations by the semi-implicit complementary volume scheme (23) and also on computational efficiency of used iterative solvers. In computations we have chosen $g(s) = \frac{1}{1+Ks^2}$ with a constant $K > 0$ and the convolution is realized using (24) with σ less than τ. The space step h in image processing experiments is always given as $1/n$, where n is number of pixels in vertical direction.

Fig. 2. Initial image (left), result of smoothing after 10 (middle) and 30 (right) scale steps (see Example 1)

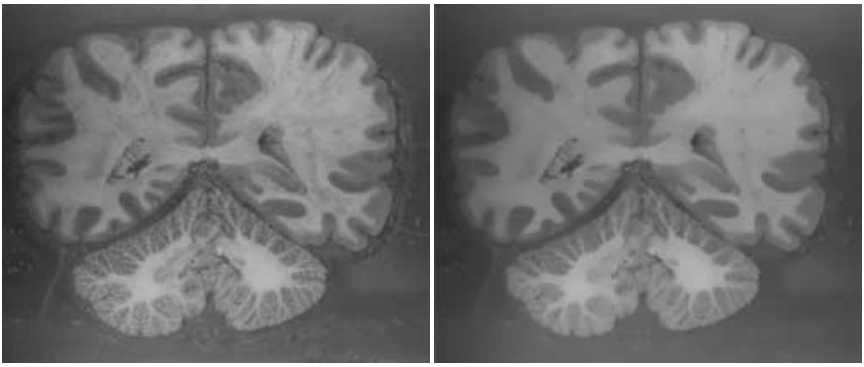

Fig. 3. Initial image (left), and its multiscale analysis (right) after 20 scale steps (see Example 2)

Example 1. In Fig. 2 we smooth an initial 321×373 pixels image (ancient coat-of-arms of Bratislava plotted in the left part of the Figure) scanned from the book with neither paper nor colors of good quality. We present results after 10 and 30 discrete scale steps, with parameters $\tau = 0.001$, $\sigma = 0.0001$, $\varepsilon^2 = 10^{-10}$, $K = 2$. In Tables 1 - 4 we give CPU-times for computing one scale step by different iterative solvers.

Example 2. In this example we process medical image (463×397 pixels) and in Fig. 3 we present result after 20 discrete steps of the scheme (23). The parameters were $\tau = 0.0001$, $\sigma = 0.00001$, $K = 4$, $\delta = 10^{-3}$.

Example 3. In this example we present multiscale analysis of mamogram (171×192 pixels). In Fig. 4 we present original (left) and results after 30 and 100 discrete steps. The parameters were $\tau = 0.0001$, $\sigma = 0.00001$, $\varepsilon^2 = 10^{-6}$, $K = 5$.

Example 4. In this example we corrupt original image plotted in Fig. 5 (150×150 pixels) by structural noise (Fig. 6, left) and by 30% salt and

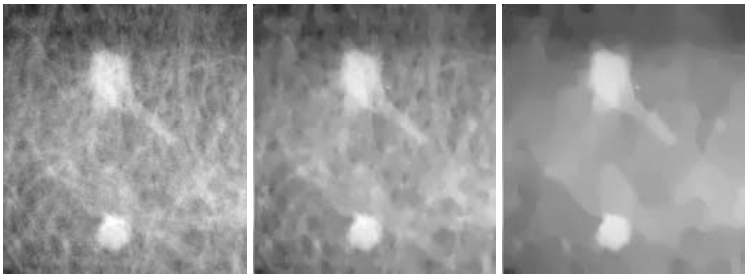

Fig. 4. Initial image (left), result of multiscale analysis after 30 (middle) and 100 (right) scale steps (see Example 3)

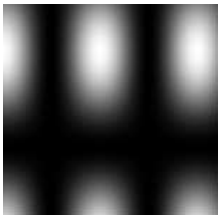

Fig. 5. Original un-noisy image (see Example 4)

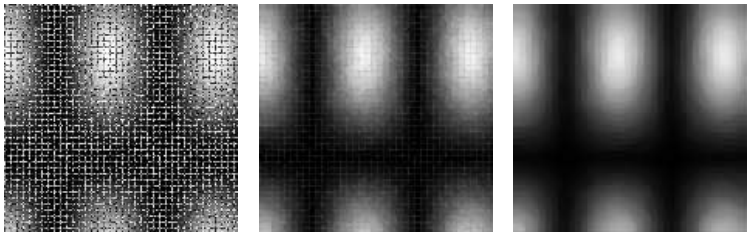

Fig. 6. Initial image corrupted by a structural noise (left), result of filtering by level set equation after 2 (middle) and 10 (right) scale steps (see Example 4)

pepper noise (Fig. 7, left). Then we reconstruct the original by level set equation (8). The parameters were $\tau = 0.00005, \sigma = 0.00001, \varepsilon^2 = 10^{-10}$, $g = 1$.

Example 5. In Fig. 8 we test our algorithm in simple situation of known exact solution of the level set equation (8) given by a shrinking circle into the point. We consider unit circle which extincts at time 0.5. From our comparison one can see precise coincidence of exact and numerical solutions and only very small error in extinction time which is for numerical solution 0.5010. The parameters were $h = 0.01$, time step $\tau = 0.0001$ and $\varepsilon^2 = 10^{-6}$.

Example 6. In Fig. 9 we test the behaviour of semi-implicit complementary volume algorithm (23) in nonconvex curve evolution by mean curvature. We

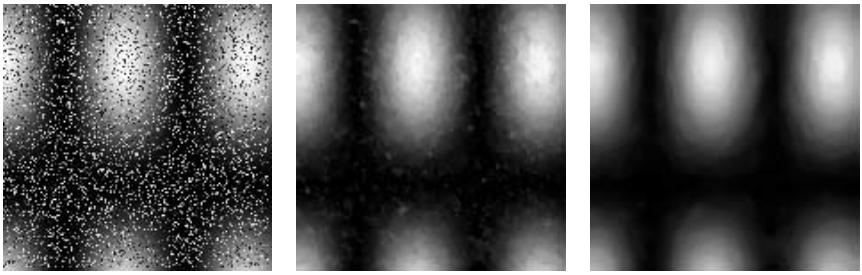

Fig. 7. Initial image corrupted by salt and pepper noise (left), result of filtering by level set equation after 2 (middle) and 10 (right) scale steps (see Example 4)

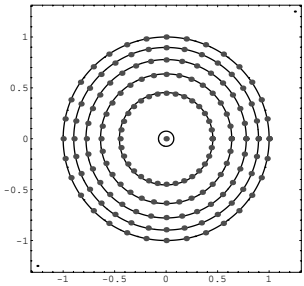

Fig. 8. Comparison of the curve evolution given by the scheme (23) (solid lines represent zero level set of numerical solution in $t = 0, 1, 2, 3, 4$) and by exact solution given by shrinking unit circle (dots) at the same time moments (see Example 5)

evolve numerically the initial nonconvex curve given in the top. In all images we plot by solid lines numerical solution by (23) and by points numerical solution given by conceptually different method based on discretization of the so-called intrinsic heat equation ([8], [9], [24], [25]). In case of scheme (23) we use space discretization parameter $h = 0.01$, time step $\tau = 0.001$, $\varepsilon^2 = 10^{-6}$ and both solutions are plotted at the same time moments ($t = 0.05, 0.1, 0.2, 0.3, 0.5, 0.8$) after which curve is shrinking in circular form into a point. From the comparison one can see a precise coincidence of two methods during evolution.

Finally we remark on solving of linear systems in discrete scale step of the semi-implicit method. The huge number of unknowns in each system, the sparsity pattern and the properties of the coefficient matrices suggest to use conjugate gradient (CG) method. With a good preconditioner, the total amount of steps required for convergence can be reduced dramatically, at the cost of slight increase in the number of operations per step, resulting in much more efficient algorithms in general.

It is well known that standard incomplete factorization (IC) methods exist for M-matrices ([19]) arising in our discretizations, and that modified

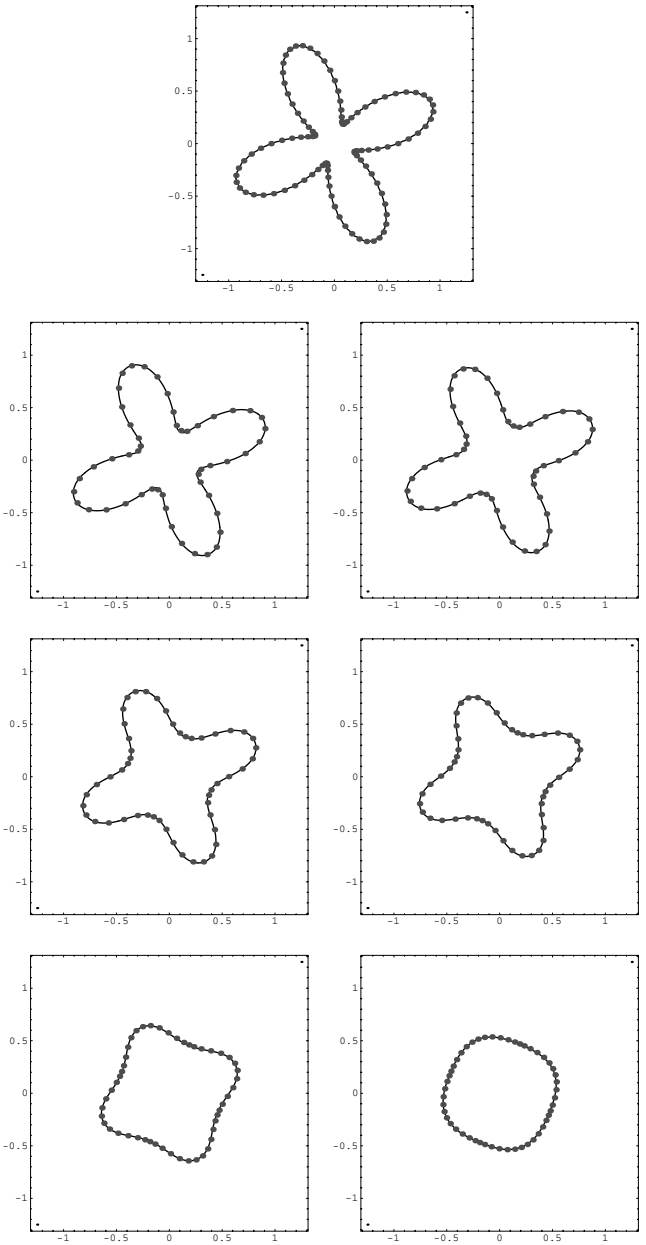

Fig. 9. Comparison of numerical solutions using semi-implicit complementary volume scheme for solving level set equation (solid lines) and Lagrangean approach based on solving the intrinsic heat equation (points) (see Example 6)

incomplete factorization (MIC) methods exist for weakly diagonally dominant matrices ([12]). The classical approaches IC(0), MIC(0) allow only to

fill entries in the Cholesky factor C where the original matrix has nonzeros. Several developments marked the last years. Two distinct ways of developing incomplete factorization preconditioners with improved accuracy were developed. The first approach is based on a symbolic factorization view, i.e. only requires the nonzero structure of the matrix to determine which fill-ins to drop. During the factorization process a *level of fill* is recursively introduced to each fill-in element from the level of fill-in of its parents. Then each fill-in that is introduced and whose level exceeds a certain threshold p is dropped, IC(p), MIC(p). The second common approach is to modify the factorization by including a dropping rule based on the numerical size T of the fill-ins introduced, ICT(p). Although the relation between the size of the dropped elements and the number of iterations required to achieve convergence is far from being completely understood, experience reveals that generally dropping small elements is more likely to produce a better quality preconditioner than dropping large elements. A great drawback of the level-of-fill approach is that it is difficult to predict the amount of fill-in that will be generated and thus memory requirements.

In our experimentations we have followed ([17]) who proposed a new incomplete Cholesky factorization depending on a parameter p that specifies the amount of additional memory that is available (in multiples of the dimension of the problem) without no need of a drop tolerance. The proposed method ([17]) retains the $n_k + p$ largest elements in the lower triangular part of the k-th column of C, where n_k is the number of elements in the k-th column of C (ICL(p) CG). The good performance of that preconditioning conjugate gradient algorithm and the growing memory requirements for big images, suggest us the following variant. Following the complementary volume discretization properties that lead to strict diagonal dominance in the matrix we allow to our algorithm to retain a variable number of $n_k + p_k$ largest elements in the lower triangular part of the k-th column of C.

In Tables 1 and 2 we report the number of matrix-vector multiplications used to obtain convergence in the solution of the linear system taken from the first scale step in Example 1, as well as CPU times in seconds on Digital Alpha XP1000 workstation. In particular in Table 1 the results were obtained by conjugate gradient algorithm without preconditioning (second column), by IC(0) and MIC(0) preconditioning (third and fourth column, respectively) and IC with threshold ICT(p,T) CG (last column). In Table 2 we compare the preconditioner ICL(p) CG proposed by ([17]) (second column), modified version MICL(p) CG (fourth column), our variable preconditioning ICL($0 \leq p \leq p_{max}$) CG and modified version of it MICL($0 \leq p \leq p_{max}$) CG (third and fifth column, respectively).

From the results of the Tables we can see that by the incomplete Cholesky preconditioned conjugate gradient we obtain a fast convergence. The prob-

lem of knowing in advance the memory requirements is solved by the proposed preconditioner. By considering $p = 6$ we need a storage of about three times the number of nonzero elements of the original matrix, but with a variable p we reduce it to twice. Moreover using variable p and modified incomplete Cholesky preconditioner we can obtain the same results of speed with only 30% more memory (see column fifth of Table 2). Tables 3 and 4 reports the same statistics, but taking a smaller scale step $\tau = 0.00001$ in Example 1. One can see that CG behaves better, but there is again an important improvement by preconditioners in case of small ε.

In other Examples the behaviour was similar. As stopping criterion we have used $\|r^k\|_2 <= tol\ \|r^0\|_2$ with tolerance $tol = 0.01$, where $\|.\|_2$ means discrete L_2 norm and r^k is residual in kth iteration.

Table 1. Comparison of preconditioned conjugate gradient solvers

♯ Mult / time	CG no prec	IC(0) CG	MIC(0) CG	ICT(6,0.0001) CG
$\varepsilon^2 = 10^{-3}$	103/3.29	18/1.43	17/1.36	11/1.23
$\varepsilon^2 = 10^{-4}$	180/5.71	27/1.97	25/2.05	12/1.34
$\varepsilon^2 = 10^{-5}$	328/10.35	40/2.94	39/2.95	11/1.16
$\varepsilon^2 = 10^{-6}$	559/17.57	68/4.86	68/4.88	9/0.91
$\varepsilon^2 = 10^{-10}$	No Conv.	79/5.23	79/5.34	9/0.91

Table 2. Comparison of preconditioned conjugate gradient solvers with limited memory

♯ Mult / time	ICL(6) CG	ICL($0 \leq p \leq 6$) CG	MICL(3) CG	MICL($0 \leq p \leq 3$) CG
$\varepsilon^2 = 10^{-3}$	4/0.87	4/0.78	4/0.90	6/0.87
$\varepsilon^2 = 10^{-4}$	5/0.91	5/0.87	6/0.96	3/1.05
$\varepsilon^2 = 10^{-5}$	6/0.97	6/0.97	8/1.07	10/1.22
$\varepsilon^2 = 10^{-6}$	5/0.94	7/1.08	10/1.22	11/1.31
$\varepsilon^2 = 10^{-10}$	6/1.07	6/1.05	11/1.29	11/1.30

Table 3. Comparison of preconditioned conjugate gradient solvers in case of smaller scale step

♯ Mult / time	CG no prec	IC(0) CG	MIC(0) CG	ICT(6,0.0001) CG
$\varepsilon^2 = 10^{-6}$	55/1.8	8/0.6	8/0.68	4/0.57
$\varepsilon^2 = 10^{-10}$	163/5.15	15/1.2	14/1.24	7/0.82

Table 4. Comparison of preconditioned conjugate gradient solvers with limited memory in case of smaller scale step

♯ Mult / time	ICL(6) CG	ICL($0 \leq p \leq 6$) CG	MICL(3) CG	MICL($0 \leq p \leq 3$) CG
$\varepsilon^2 = 10^{-6}$	3/0.6	3/0.63	6/0.7	3/0.51
$\varepsilon^2 = 10^{-10}$	5/0.66	5/0.68	3/0.53	3/0.52

Acknowledgements. This work was supported by grant No. 1/7132/20 of the Slovak Scientific Grant Agency VEGA and by Italian MURST and GNCS-INDAM grants.

References

1. L. Alvarez, F. Guichard, P.L. Lions, J.M. Morel, Axioms and Fundamental Equations of Image Processing, Arch. Rat. Mech. Anal. **123** (1993) 200–257
2. L. Alvarez, J.M. Morel, Formalization and computational aspects of image analysis, Acta Numerica (1994) 1–59
3. L. Alvarez, P.L. Lions, J.M. Morel, Image selective smoothing and edge detection by nonlinear diffusion II, SIAM J. Numer. Anal. **29** (1992) 845–866
4. E. Bänsch, K. Mikula, A coarsening finite element strategy in image selective smoothing, Comput. Vis. Sci. **1**(1) (1997) 53–61
5. F. Catté, P.L. Lions, J.M. Morel, T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, SIAM J. Numer. Anal. **129** (1992) 182–193
6. Y.-G. Chen, Y. Giga, S. Goto, Uniqueness and existence of viscosity solutions of generalized mean curvature flow equation, J. Diff. Geom. **33** (1991) 749–786
7. M.G. Crandall, H. Ishii, P.L. Lions, User's guide to viscosity solutions of second order partial differential equations, Bull.(NS) Amer. Math. Soc. **27** (1992) 1 –67
8. G. Dziuk, Algorithm for evolutionary surfaces, Numer. Math. **58** (1991) 603–611
9. G. Dziuk, Convergence of a semidiscrete scheme for the curve shortening flow, Math. Models Methods in Appl. Sci. **4** (1994) 589–606
10. G. Dziuk, Numerical schemes for the mean curvature flow of graphs, in P.Argoul, M.Frémond, Q.S.Nguyen (Eds): IUTAM Symposium on Variations of Domains and Free Boundary Problems in Solid Mechanics, Kluwer Academic Publishers, Dodrecht-Boston-London (1999) pp. 63–70
11. L.C. Evans, J. Spruck, Motion of level sets by mean curvature I, J. Diff. Geom. **33** (1991) 635–681
12. I. Gustafsson, Modified incomplete Cholesky (MIC) Methods, in Preconditioning methods, theory and applications, Ed. D.J.Evans, Gordon and Breach, New York (1983) 265–293
13. A. Handlovičová, K. Mikula, A. Sarti, Numerical solution of parabolic equations related to level set formulation of mean curvature flow, Comput. Vis. Sci. **1**(2) (1999) 179–182
14. J. Kačur, K. Mikula, Solution of nonlinear diffusion appearing in image smoothing and edge detection, Appl. Numer. Math. **17** (1995) 47–59
15. J. Kačur, K. Mikula, Slow and fast diffusion effects in image processing, Comput. Vis. Sci. **3**(4) (2001) 185–195
16. S. Kichenassamy, The Perona-Malik paradox, SIAM J. Appl. Math. **57**(5) (1997) 1328–1342
17. C.J. Lin, J.J. Moré, Incomplete Cholesky factorizations with limited memory , SIAM. J. Sci. Comput. **21** (1999) 24–45
18. P.L. Lions, Axiomatic derivation of image processing models, Math. Models Methods in Appl. Sci. **4** (1994) 467–475
19. J.A Meijerink, H.A. van der Vorst, An iterative solution method for systems of which the coefficient matrix is a symmetric M-matrix, Math. Comp. **31** (1977) 148–162
20. K. Mikula, Solution of nonlinear curvature driven evolution of plane convex curves, App. Numer. Math. **23**(3) (1997) 347–360
21. K. Mikula, J. Kačur, Evolution of convex plane curves describing anisotropic motions of phase interfaces, SIAM J. Sci. Comput. **17** (1996) 1302–1327

22. K. Mikula, N. Ramarosy, Semi-implicit finite volume scheme for solving nonlinear diffusion equations in image processing, Numer. Math., DOI 10.1007/s002110100264
23. K. Mikula, A. Sarti, C. Lamberti, Geometrical diffusion in 3D-echocardiography, Proceedings of ALGORITMY'97 – Conference on Scientific Computing, West Tatra Mountains – Zuberec (1997) 167–181
24. K. Mikula, D. Ševčovič, Solution of nonlinearly curvature driven evolution of plane curves, Appl. Numer. Math. **31**(2) (1999) 191–207
25. K. Mikula, D. Ševčovič, Evolution of plane curves driven by a nonlinear function of curvature and anisotropy, SIAM J. Appl. Math. **61**(5) (2001) 1473–1501
26. S. Osher, J. Sethian, Front propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formulation, J. Comput Phys. **79** (1988) 12–49
27. P. Perona, J. Malik, Scale space and edge detection using anisotropic diffusion. In Proc. IEEE Computer Society Workshop on Computer Vision (1987) 16–22
28. Y.Saad, Iterative methods for sparse linear systems. PWS Publ. Comp. (1996)
29. A.Sarti, K.Mikula, F.Sgallari, Nonlinear multiscale analysis of 3D echocardiographic sequences, IEEE Transactions on Medical Imaging **18**(6) (1999) 453–466
30. J.A. Sethian, Numerical algotithm for propagating interfaces: Hamilton-Jacobi equations and conservation laws, J. Diff. Geom. **31** (1990) 131–161
31. J.A. Sethian, Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Material Science, Cambridge University Press 1999
32. N.J. Walkington, Algorithms for computing motion by mean curvature, SIAM J. Numer. Anal. **33**(6) (1996) 2215–2238
33. J. Weickert, B.M.t.H. Romeny, M.A. Viergever: Efficient and reliable schemes for nonlinear diffusion filtering, IEEE Trans. Image Processing **7**(3) (1998) 398–410