FLUX-BASED LEVEL SET METHOD:
A FINITE VOLUME METHOD FOR EVOLVING INTERFACES

PETER FROLKOVIČ [*] AND KAROL MIKULA [†]

Abstract. We introduce a new computational technique for evolving interfaces, the flux-based level set method. A nonlinear degenerate advection-diffusion level set equation is discretized by a finite volume method using a complementary volume strategy. It enables to solve the problem in an efficient and stable way. Using a flux-based method of characteristics for the advective part and a semi-implicit treatment of diffusive part, it removes the standard CFL condition on time step and it decreases CPU times significantly. The method is presented for 2D and 3D interface motions driven in normal direction by a constant driving force and (mean) curvature. Comparisons with known exact solutions and further numerical experiments, including topological changes of the interface, are presented.

Key words. level set method, finite volume method, evolving interfaces, method of characteristics

AMS subject classifications. 53C44 , 65M25, 74S10

1. Introduction. Moving interfaces (or free boundaries) arise in a broad range of applications. They represent boundaries between solid and liquid phase in the solidification of materials [1, 30], boundaries between air and liquid or immiscible liquids in free surface multiphase flows [35, 11, 17], boundaries between burnt and unburnt regions in flame front propagation in combustion [33], or free discontinuities representing edges in digital image segmentation [3, 32]. For a comprehensive overview of models, methods and applications where free boundaries occur, we refer to books by Sethian, Sapiro and Osher and Fedkiw [33, 29, 25].

An interface, represented by a closed curve in 2D or a hypersurface in 3D, can be advected by an external forcing term, given, e.g., by a velocity field of flowing liquid, by a temperature difference between phases, or by edge attracting forces, etc. Usually, a shape of the free boundary is also influenced by a principle of the minimization of surface energy related to surface tension effects. Thus, a local curvature of the interface and an anisotropy of the material (expressed in an orientation of the interface) play a role.

In a Lagrangian viewpoint (see, e.g., [18, 20, 21]), the motion of any point x of 2D interface can be represented by a space-time dependent vector field, which can be written in the form

$$(1.1) \qquad \partial_t x = \beta \vec{N} + \alpha \vec{T}$$

with $\vec{N}$ and $\vec{T}$ being the normal and tangent vector to the interface, respectively, and β, α being normal and tangential velocities (3D situation can be described similarly). Since the tangential component of the motion is related only to a reparametrization of the closed curve or hypersurface, the normal component of the velocity field influences the time image of interface evolution. Thus, the general law for the interface motion is given in the form of geometrical equation for the normal velocity $V = \partial_t x \cdot \vec{N}$ of

[*]Interdisciplinary Center for Scientific Computing, University of Heidelberg, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany, `peter.frolkovic@web.de`

[†]Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava, Slovak Republic, `mikula@vox.svf.stuba.sk`

the interface,

$$V = \beta(x, \nu, k),\tag{1.2}$$

where x is a spatial position of the moving curve, ν is a tangent angle to the interface, k is the curvature and the function β is increasing in its third argument.

In this paper we consider the simple but representative situation,

$$V = \mu k + \delta,\tag{1.3}$$

i.e., we assume a linear dependence of the evolution on the (mean) curvature k, where μ is a positive constant, and the driving force is given by a constant δ. For the modeling of 2D and 3D interface motions in the form of (1.3), we use an Eulerian approach and the so-called level set formulation [24, 33, 25]. Although the Eulerian approach passes the problem to one dimension higher case, it is a natural tool for handling complicated dynamics like topological changes of the interface. The corresponding advection-diffusion form of the level set formulation of (1.3) is given by the following level set equation

$$\partial_t \phi = \mu |\nabla \phi| \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + \delta \frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \phi,\tag{1.4}$$

where ϕ is a function of which the zero level set describes the evolving interface. The interface evolution does not depend on the choice of initial condition for ϕ, however, a signed distance function is a standard choice. As it is usual in the level set modeling of complex interface motions, we consider homogeneous Neumann boundary conditions.

Well-known finite difference computational techniques to solve level set formulations numerically were introduced by Osher and Sethian. They are based on ENO or WENO type finite difference schemes for solving Hamilton-Jacobi equations that are given for (1.3) in the form $\phi_t = \beta |\nabla \phi|$ [24, 31, 14]. The finite difference approach for solving Hamilton-Jacobi equation related to optimal control problems was developed in [27, 7]. Semi-Lagrangian schemes for solving an advective form $\phi_t = \beta \frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \phi$ of the level set formulation was introduced by Strain. He generalized so-called CIR scheme for linear advection equations to the case of advective form of the level set formulation in order to weaken standard CFL stability constrain [34].

Nowadays, applications of the finite volume methods to many physical and engineering problems, including fluid dynamics, material science, structural mechanics or image processing, are rapidly growing, cf. [16, 6, 19]. In [15] the finite volume method for solving Hamilton-Jacobi equations including level set formulation of interface motion is given and studied. An artificial diffusion term is added to get stability and, consequently, convergence of the scheme.

In this paper we introduce new flux-based finite volume scheme for solving advection-diffusion form (1.4) of the level set formulation to (1.3). In our approach, equation (1.4) is rewritten to an integral form and discretized in space by a finite volume method using a complementary volume (co-volume) technique. The advection-diffusion form (1.4) allows us to apply unconditionally stable *semi-implicit* time discretization in the curvature part of (1.4) and explicit upwind scheme with *recursive flux redistribution* (flux-based method of characteristics [9]) removing standard CFL stability constrain in advective part of (1.4), which are the main ingredients for stability and efficiency of our method.

In [9], the flux-based method of characteristics for the linear advection equation with divergence free velocity field was introduced. In this paper, we extend the

method to nonlinear case of advective part of (1.4) where the advective velocity need not be divergence free. The method can be viewed as an extension of L_∞ stable upwind schemes to allow time steps with the Courant number significantly larger than one. After solving the advective part of the motion, one can apply the semi-implicit time discretization to the curvature term, which is unconditionally stable in L_∞ and $W^{1,1}$ sense for arbitrary time step, cf. [12]. Such combination of the discretization schemes for the advective and the curvature part of (1.4) can be viewed as a two step operator splitting procedure, and, at the end, it gives our *flux-based level set method*. Computational results show that precision of the scheme is not deteriorated using larger time steps and document its applicability in complex 2D and 3D situations of curve and surface dynamics.

The paper is organized as follows. In Section §2, we present the finite volume method for the advective part of equation (1.4). In Section §3, we discuss its extension by the flux-based method of characteristics. In Section §4, we present the semi-implicit co-volume scheme for the curvature driven level set equation. In Section §5, we introduce the flux-based level set method. Finally, in Section §6, we discuss several numerical experiments.

2. Finite volume method for advective level set equation. In this section, we restrict our treatment to the advective part of the level set equation (1.4), that can be rewritten to the form

$$(2.1) \qquad \partial_t \phi + \vec{v} \cdot \nabla \phi = 0 \,,$$

where the vector velocity field $\vec{v}$ is defined on a whole computational domain (if $|\nabla \phi| \neq 0$) by

$$(2.2) \qquad \vec{v} = -\delta \, \vec{N} = -\delta \, \frac{\nabla \phi}{|\nabla \phi|} \,.$$

We describe next the solution by finite volume method for a special case of the advection equation (2.1) with the velocity $\vec{v} = \vec{v}(x)$ being a given vector function. In general, $\nabla \cdot \vec{v} \neq 0$. The differential equation (2.1) is considered for $x \in \Omega$, where $\Omega \subset R^d$ is a polygonal domain with $d = 2$ or $d = 3$, and $t \in [t^n, t^{n+1}] \subset R$.

The initial conditions are defined by

$$(2.3) \qquad \phi(t^n, x) = \phi^n(x) \,, \quad x \in \overline{\Omega} \,.$$

On the boundary $\partial \Omega$ of Ω, we consider boundary conditions $\vec{n}(\gamma) \cdot \vec{v}(\gamma) = 0$, $\gamma \in \partial \Omega$, where $\vec{n}(\gamma)$ is the outward unit normal vector to $\partial \Omega$.

Next, we formulate an integral form of the differential advective level set equation (2.1). First, we rewrite (2.1) to an equivalent divergent form,

$$(2.4) \qquad \partial_t \phi + \nabla \cdot (\vec{v} \, \phi) - \phi \nabla \cdot \vec{v} = 0 \,.$$

Further, we assume that a finite volume mesh of polygonal subsets $\Omega_i \subset \Omega$, $i = 1, \ldots, I$ is available that covers Ω, i.e.,

$$(2.5) \qquad \overline{\Omega} = \bigcup_{i=1}^{I} \overline{\Omega}_i \,, \quad \Omega_i \neq \emptyset \,, \quad \Omega_i \cap \Omega_j = \emptyset \,, \text{ if } i \neq j \,.$$

Finally, integrating (2.4) over the finite volume Ω_i and the time interval (t^n, t^{n+1}), one obtains

$$(2.6) \qquad \int\limits_{\Omega_i} \phi(t^{n+1}, x)\, dx =$$

$$\int\limits_{\Omega_i} \phi(t^n, x)\, dx - \int\limits_{t^n}^{t^{n+1}} \int\limits_{\partial\Omega_i} \phi(t, \gamma)\, \vec{n}_i(\gamma) \cdot \vec{v}(\gamma)\, d\gamma dt + \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \phi(t, x)\, \nabla \cdot \vec{v}(x)\, dx dt \,,$$

where $\vec{n}_i = \vec{n}_i(\gamma)$ is the outward unit normal vector to $\partial\Omega_i$.

Note that (2.6) can be viewed as an integral formulation of conservation laws with source terms. If one thinks of ϕ being a density then (2.6) expresses a mass balance formulation for (2.4). Consequently, if $\nabla \cdot \vec{v} = 0$ (i.e., for a divergence free velocity), the last integral in (2.6) vanishes and (2.6) describes the local mass conservation property.

If ϕ represents a level set function (e.g., a signed distance function), the conservation property for integrals of ϕ (if $\nabla \cdot \vec{v} = 0$) does not have a direct practical value for numerical simulations. Instead, one is interested in conservation of integrals for $H(u)$, where H is the Heaviside function. Nevertheless, the view of integral formulation (2.6) as conservation laws with source terms is essential for our algorithm, and, therefore, the notion of "mass balance" will be used when helpful.

Before formulating a discrete form of (2.6), we introduce some notations. First,

$$(2.7) \qquad \phi_i^{n+1} := \frac{1}{|\Omega_i|} \int\limits_{\Omega_i} \phi(t^{n+1}, x)\, dx \,, \quad \phi_i^n := \frac{1}{|\Omega_i|} \int\limits_{\Omega_i} \phi(t^n, x)\, dx \,,$$

where $|\Omega_i|$ denotes the measure in R^d of Ω_i. In such a way, the term $\phi_i^n |\Omega_i|$ can represent the exact integral of ϕ over Ω_i at $t = t^n$ that can be computed for $n = 0$ from the initial conditions (2.3). Analogously, the term $\phi_i^{n+1} |\Omega_i|$ will represent the exact integral of ϕ over Ω_i at $t = t^{n+1}$ and it has to be determined.

Further, one can split the boundary $\partial\Omega_i$ to several segments Γ_{ij},

$$(2.8) \qquad \partial\Omega_i := \bigcup_{j \in \Lambda_i} \overline{\Gamma_{ij}} \,, \quad \Gamma_{ij} := \partial\Omega_i \cap \partial\Omega_j \,,$$

where the set Λ_i contains the indices of neighbouring cells Ω_j of Ω_i with Γ_{ij} having a nonzero measure in R^{d-1}.

In such a way, (2.6) can be written to the form,

$$(2.9) \qquad \phi_i^{n+1} |\Omega_i| = \phi_i^n |\Omega_i| -$$

$$\sum_{j \in \Lambda_i} \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Gamma_{ij}} \phi(t, \gamma)\, \vec{n}_i(\gamma) \cdot \vec{v}(\gamma)\, d\gamma dt + \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \phi(t, x)\, \nabla \cdot \vec{v}(x)\, dx dt \,.$$

Note that no numerical approximation has been applied so far in (2.9).

To introduce the finite volume discretization of (2.9), we define first the integrated fluxes

$$(2.10) \qquad v_{ij} := \int_{\Gamma_{ij}} \vec{n}_i(\gamma) \cdot \vec{v}(\gamma) \, d\gamma \,,$$

and we distinguish between the outflow and inflow boundaries Γ_{ij} of $\partial\Omega_i$ by defining, respectively, the sets of indices Λ_i^{out} and Λ_i^{in},

$$(2.11) \qquad \Lambda_i^{out} := \{j \in \Lambda_i \,, \, v_{ij} \geq 0\} \,, \quad \Lambda_i^{in} := \{j \in \Lambda_i \,, \, v_{ij} < 0\} \,.$$

Of course, $v_{ij} = -v_{ji}$, and if $i \in \Lambda_j^{out}$ then $j \in \Lambda_i^{in}$. In general, (2.10) is realized numerically.

Now, if one considers a piecewise constant form of $\phi^n(x)$, i.e.,

$$(2.12) \qquad \phi^n(x) = \phi_i^n \,, \, x \in \Omega_i \,,$$

and applies standard upwind arguments for the approximation of advective flux, the following discretization scheme can be derived for (2.9),

$$(2.13) \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n|\Omega_i| - \tau \sum_{j \in \Lambda_i^{out}} \phi_i^n \, v_{ij} - \tau \sum_{j \in \Lambda_i^{in}} \phi_j^n \, v_{ij} + \tau \, \phi_i^n \sum_{j \in \Lambda_i} v_{ij} \,,$$

where $\tau := t^{n+1} - t^n$.

In (2.13), the upstream values of $\phi^n(x)$ are used for the computations of advective fluxes $\phi \, \vec{n}_i \cdot \vec{v}$ at Γ_{ij}, i.e., the values $\phi = \phi_i^n$ are taken for $v_{ij} > 0$, and, consequently, $\phi = \phi_j^n$ for $v_{ij} < 0$. Moreover, for the source term in (2.6), the function ϕ is approximated by ϕ_i^n and the standard Green formula is used for the integral of $\nabla \cdot \vec{v}$ over Ω_i.

After a simple algebraic manipulation, (2.13) takes the form

$$(2.14) \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n|\Omega_i| + \tau \sum_{j \in \Lambda_i^{in}} \phi_j^n \, v_{ji} - \tau v_i \phi_i^n \,,$$

where v_i denotes the total inflow flux, i.e.,

$$(2.15) \qquad v_i := \sum_{j \in \Lambda_i^{in}} v_{ji} \,.$$

It is important to comment the differences in a possible interpretation of the standard form (2.13) and the new form (2.14). The last term in (2.13) represents the source term that can be either positive (injection) or negative (sink). In (2.14), the last term (if present) can be only negative, thus it represents the effective sink. From the point of view of mass balance formulation, the equation (2.14) would describe that the mass at the new time level (the left hand side) is equal to the mass at the previous time level (the first term on the right hand side) plus the mass coming through the inflow boundary (the second term) minus the mass leaving due to the *sink term* v_i (the last term). For a divergence free velocity field, the sink v_i is equal to the total outflow flux.

Finally, one can rewrite (2.14) to the basic finite volume scheme for the advective level set equation

$$(2.16) \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n \left(|\Omega_i| - \tau v_i\right) + \sum_{j \in \Lambda_i^{in}} \phi_j^n \, \tau v_{ji} \,.$$

Now, if one denotes the so called *critical time step* τ_i [9],

$$(2.17) \qquad \tau_i := \frac{|\Omega_i|}{v_i} \,,$$

it is clear that (2.16) is acceptable only for time steps τ that fulfill the CFL condition

$$(2.18) \qquad \tau \le \tau_{CFL} := \min\{\tau_i\,,\ i = 1, \ldots, I\}\,,$$

otherwise nonphysical oscillations of the numerical solution can occur, cf. Lemma 2.1.

Using (2.17), one can define the so called local grid Courant number

$$(2.19) \qquad \mathcal{C}_i = \mathcal{C}_i(\tau) := \frac{\tau}{\tau_i} \,,$$

and (2.18) denotes the well-known condition that $\mathcal{C}_i(\tau) \le 1$, $i = 1, \ldots, I$.

REMARK 1. *The critical time step τ_i in (2.17) and the local Courant number $\mathcal{C}_i$ in (2.19) are, formally, not defined if $v_i = 0$. This can happen when there is no inflow into Ω_i and the equation (2.14) turns to the trivial form $\phi_i^{n+1} = \phi_i^n$. It is clear that for this discrete equation there exists no restriction on the time step τ and one can, formally, define $\tau_i = \infty$ and $\mathcal{C}_i = 0$.*

LEMMA 2.1. *If (2.18) is valid, the values ϕ_i^{n+1} computed from (2.16) fulfil the local discrete minimum and maximum principle*

$$(2.20) \qquad \min\{\phi_i^n\,,\ \min_{j \in \Lambda_i^{in}}\{\phi_j^n\}\} \le \phi_i^{n+1} \le \max\{\phi_i^n\,,\ \max_{j \in \Lambda_i^{in}}\{\phi_j^n\}\}\,.$$

Proof. If we divide (2.16) by $|\Omega_i|$ we see on the right hand side that for any i the coefficients of ϕ_i^n and of all $\phi_j^n, j \in \Lambda_i^{in}$ are nonnegative provided (2.18). Moreover in the sum they are equal to 1. It means that ϕ_i^{n+1} is a convex combination (weighted average) of ϕ_i^n and $\phi_j^n, j \in \Lambda_i^{in}$, so the inequalities (2.20) are clearly satisfied. $\square$

If homogeneous Neumann boundary conditions are imposed and (2.18) is valid then clearly also global discrete L_∞ stability property is fulfilled, namely

$$(2.21) \qquad \min_{j=1,\ldots,I} \phi_j^n \le\ \phi_i^{n+1}\ \le \max_{j=1,\ldots,I} \phi_j^n, \quad i = 1, \ldots, I\,.$$

The advantage of discretization scheme (2.16) is that it can be viewed as the discrete mass balance formulation of conservation laws with source terms and that it gives physically acceptable values of the numerical solution with no unphysical oscillations. Moreover, it offers the straightforward definition (2.19) of the local grid Courant number for general finite volume computational grids. This can be used to extend this method for arbitrary large time steps (at least in a theory) by preserving the mass balance formulation and the discrete minimum-maximum principle. Such extension is explained in following Section.

3. Flux-based method of characteristics. For a clarity of presentation, we prefer to view ϕ as a density here and to treat (2.14) as a discrete mass balance formulation of conservation laws with source terms.

Before presenting a general form of the flux-based method of characteristics (FB-MOC) for the advective level set equation (2.1), we try to introduce it in simple consequent steps. To do so, let us first write (2.14) for some index $j \in \Lambda_i^{in}$,

$$(3.1) \qquad \phi_j^{n+1}|\Omega_j| = \phi_j^n|\Omega_j| + \tau \sum_{k \in \Lambda_j^{in}} v_{kj}\,\phi_k^n - \tau\, v_j\, \phi_j^n\,.$$

First, let us suppose that $\tau = \tau_j = \tau_{CFL}$, see also (2.17) and (2.18). In such a case, (3.1) takes the form

$$(3.2) \qquad \phi_j^{n+1}|\Omega_j| = \tau_j \sum_{k \in \Lambda_j^{in}} v_{kj}\, \phi_k^n = \left(\sum_{k \in \Lambda_j^{in}} \frac{v_{kj}}{v_j} \phi_k^n \right) |\Omega_j|\,.$$

Standard finite volume methods must stop here, because $\mathcal{C}_j(\tau) = 1$, and hence for a larger time step the density ϕ_j^n can not be used in the full upwind discretization at the outflow boundary of $\partial\Omega_j$ or for the sink term v_j. In fact, as one can see from (3.2), the density ϕ_j^n must be replaced for $t > t^n + \tau_j$ by the density given in the parentheses in the right hand side (r.h.s.) of (3.2), which is nothing else than a weighted average of neighbouring inflowing densities.

Consequently, for a "little larger" time step τ such that $\tau > \tau_j$, but $\tau < \tau_k$ for $k = 1, \ldots, I$ and $k \neq j$, the j-th discrete equation can be written for the time interval $(t^n, t^n + \tau_j)$ and $(t^n + \tau_j, t^{n+1})$, and the sum of both equations is equal then to

$$(3.3) \qquad \phi_j^{n+1}|\Omega_j| = \phi_j^n|\Omega_j| +$$

$$\tau \sum_{k \in \Lambda_j^{in}} v_{kj}\, \phi_k^n - v_j \left(\tau_j \phi_j^n + (\tau - \tau_j) \sum_{k \in \Lambda_j^{in}} \frac{v_{kj}}{v_j} \phi_k^n \right)\,.$$

However, after simple algebraic manipulations, one obtains that the equations (3.3) and (3.2) are identical. It is a simple consequence of the fact that for $\tau \geq \tau_j$ the r.h.s. of the j-th discrete equation can be written as the mass contribution to Ω_j through the inflow boundary during time interval $(t^{n+1} - \tau_j, t^{n+1})$.

The equations that must be changed for our special choice of the time step τ are those with indices $i \in \Lambda_j^{out}$. Particularly, using (3.2), the equation (2.14) for some index $i \in \Lambda_j^{out}$ can be written as follows,

$$(3.4) \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n|\Omega_i| +$$

$$\left(v_{ji} \left(\tau_j \phi_j^n + (\tau - \tau_j) \sum_{k \in \Lambda_j^{in}} \frac{v_{kj}}{v_j} \phi_k^n \right) + \tau \sum_{\substack{j' \in \Lambda_i^{in} \\ j' \neq j}} v_{j'i}\, \phi_{j'}^n \right) - \tau\, v_i\, \phi_i^n\,.$$

These considerations can be continued to define a general form of the FB-MOC. To extend (3.4) for arbitrary large time step τ, one has to find all possible combinations of indices starting with all $j_0 \in \Lambda_i^{in}$, continuing backwards with $j_1 \in \Lambda_{j_0}^{in}$, until the last neighbours in these chains, say $j_l \in \Lambda_{j_{l-1}}^{in}$, are found, such that the following two conditions are fulfilled,

$$(3.5) \qquad \tau_{j_0} + \tau_{j_1} + \ldots + \tau_{j_{l-1}} < \tau\,, \quad \tau_{j_0} + \tau_{j_1} + \ldots + \tau_{j_{l-1}} + \tau_{j_l} \geq \tau\,.$$

The conditions (3.5) mean that the densities $\phi_{j_0}^n$ until $\phi_{j_l}^n$ are entering the finite volume Ω_i through the inflow part of $\partial\Omega_i$ during time interval (t^n, t^{n+1}), with the densities $\phi_{j_l}^n$ being the last ones (the most far-away ones). Note that several chains of indices can start with the index j_0.
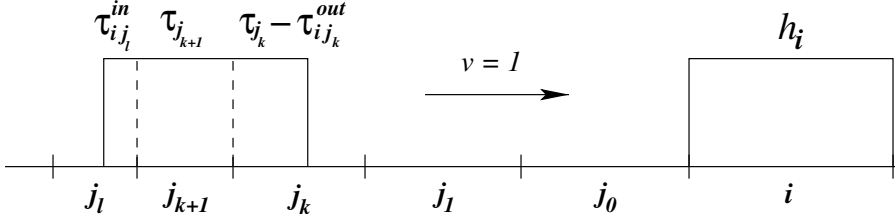
FIG. 3.1. *1D example of the flux-based method of characteristics on a nonuniform grid for (2.1) with $\vec{v} \equiv v = 1$. Note that in this case one has $|\Omega_i| =: h_i = \tau_i = (\tau_{j_k} - \tau_{ij_k}^{out}) + \tau_{j_{k+1}} + \tau_{ij_l}^{in}$.*

Further, for each particular chain of indices given by (3.5), one has to find k such that $k \leq l$ and

$$(3.6) \quad \tau_i + \tau_{j_0} + \tau_{j_1} + \ldots + \tau_{j_{k-1}} < \tau, \quad \tau_i + \tau_{j_0} + \tau_{j_1} + \ldots + \tau_{j_{k-1}} + \tau_{j_k} \geq \tau.$$

The conditions (3.6), together with (3.5), mean that the densities $\phi_{j_0}^n$ until $\phi_{j_k}^n$ are not only entering the finite volume Ω_i, but also leaving it through the sink v_i during time interval (t^n, t^{n+1}) with the densities ϕ_k^n being the most far-away ones.

The conditions (3.5) - (3.6) can be used to define the "rest" values $\tau_{ij_l}^{in}$ and $\tau_{ij_k}^{out}$,

$$\tau_{ij_l}^{in} := \tau - \tau_{j_0} - \ldots - \tau_{j_{l-1}}, \quad \tau_{ij_k}^{out} := \tau - \tau_i - \tau_{j_0} - \ldots - \tau_{j_{k-1}}.$$

Particularly, $\tau_{ij_l}^{in}$ denote the length of time intervals in which the most far-away densities $\phi_{j_l}^n$ are entering Ω_i. Of course, $\tau_{ij_l}^{in} \leq \tau_{j_l}$. Analogously, the values $\tau_{ij_k}^{out}$ denote the length of time intervals in which the most far-away density $\phi_{j_k}^n$ are leaving Ω_i due to the sink v_i. Again, $\tau_{ij_k}^{out} \leq \tau_{j_k}$. One can also see that if $k < l$, then

$$(3.7) \qquad \qquad \tau_i = (\tau_{j_k} - \tau_{ij_k}^{out}) + \tau_{j_{k+1}} + \ldots + \tau_{j_{l-1}} + \tau_{ij_l}^{in},$$

see also Figure 3.1 for an illustration, and if $k = l$, then $\tau_i = \tau_{ij_l}^{in} - \tau_{ij_k}^{out}$.

Using all above definitions, one can introduce the general form of the FB-MOC,

$$(3.8) \qquad \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n|\Omega_i| +$$

$$\sum_{j_0 \in \Lambda_i^{in}} v_{j_0 i} \left(\tau_{j_0} \phi_{j_0}^n + \sum_{j_1 \in \Lambda_{j_0}^{in}} \frac{v_{j_1 j_0}}{v_{j_0}} \left(\tau_{j_1} \phi_{j_1}^n + \ldots \sum_{j_l \in \Lambda_{j_{l-1}}^{in}} \frac{v_{j_l j_{l-1}}}{v_{j_{l-1}}} \tau_{ij_l}^{in} \phi_{j_l}^n \right) \right) -$$

$$v_i \tau_i \phi_i^n - \sum_{j_0 \in \Lambda_i^{in}} v_{j_0 i} \left(\tau_{j_0} \phi_{j_0}^n + \sum_{j_1 \in \Lambda_{j_0}^{in}} \frac{v_{j_1 j_0}}{v_{j_0}} \left(\tau_{j_1} \phi_{j_1}^n + \ldots \sum_{j_k \in \Lambda_{j_{k-1}}^{in}} \frac{v_{j_k j_{k-1}}}{v_{j_{k-1}}} \tau_{ij_k}^{out} \phi_{j_k}^n \right) \right).$$

If $\tau \geq \tau_i$, analogously to the description after (3.3), the equations (3.8) can be rewritten to the form where only the mass contribution from the inflow for the time interval $(t^{n+1} - \tau_i, t^{n+1})$ is considered,

$$(3.9) \qquad \qquad \phi_i^{n+1}|\Omega_i| = \sum_{j_0 \in \Lambda_i^{in}} v_{j_0 i} \sum_{j_1 \in \Lambda_{j_0}^{in}} \frac{v_{j_1 j_0}}{v_{j_0}} \cdots \sum_{j_k \in \Lambda_{j_{k-1}}^{in}} \frac{v_{j_k j_{k-1}}}{v_{j_{k-1}}}$$

$$\left((\tau_{j_k} - \tau_{ij_k}^{out})\phi_{j_k}^n + \sum_{j_{k+1} \in \Lambda_{j_k}^{in}} \frac{v_{j_{k+1}j_k}}{v_{j_k}} \left(\tau_{j_{k+1}}\phi_{j_{k+1}} + \dots \sum_{j_l \in \Lambda_{j_{l-1}}^{in}} \frac{v_{j_l j_{l-1}}}{v_{j_{l-1}}} \tau_{ij_l}^{in} \phi_{j_l}^n \right) \right).$$

If $\tau < \tau_i$, the term $(|\Omega_i| - \tau v_i)\phi_i^n$ must be added to the r.h.s. of (3.9), analogously to (3.8).

THEOREM 3.1. *Let us suppose homogeneous Neumann boundary conditions to* (2.1). *Then for any time step* τ *the discrete* L_∞ *stability estimate is valid for numerical solution given by the scheme* (3.9), *i.e.*

$$(3.10) \qquad \min_{j=1,\dots,I} \phi_j^n \le \phi_i^{n+1} \le \max_{j=1,\dots,I} \phi_j^n, \quad i = 1, \dots, I.$$

Proof. Dividing (3.9) by $|\Omega_i|$, one can rewrite it to

$$(3.11) \qquad \phi_i^{n+1} = \sum_{l,k} \sum_{m=k}^{l} \frac{\hat{\tau}_{j_m}}{\tau_i} \frac{v_{j_0 i}}{v_i} \frac{v_{j_1 j_0}}{v_{j_0}} \cdots \frac{v_{j_m j_{m-1}}}{v_{j_{m-1}}} \phi_{j_m}^n,$$

where the first sum in (3.11) is, analogously to (3.9), realized for all indices l and k given by (3.5) and (3.6) and value $\hat{\tau}_{j_m}$ is defined by

$$(3.12) \qquad \hat{\tau}_{j_m} = \begin{cases} \tau_{j_k} - \tau_{ij_k}^{out} & k = m < l, \\ \tau_{j_m} & k < m < l, \\ \tau_{ij_l}^{in} & k < m = l, \\ \tau_i & k = m = l. \end{cases}$$

Using (3.7) and definition of total inflow fluxes (2.15) we see that the right hand side of (3.11) is weighted avarage of all values $\phi_{j_m}^n$ contributing to ϕ_i^{n+1} by the chains given by (3.5) and (3.6). Due to homogeneous Neumann boundary conditions this convex combination is realized on a subset of all ϕ_j^n, $j = 1, \dots, I$, so the resulting value must be greater or equal then their minimum and less or equal to their maximum. □

3.1. Implementation of FB-MOC for advective level set equation. The description of the flux-based method of characteristics (3.8) can be viewed as a backward tracking form of this algorithm. This means that the characteristics (or, more precisely, the inflow boundaries) are tracked backwards to find the densities (or the fractions of mass) that are added to the r.h.s. of (3.9) (or, equivalently, to the mass $\phi_i^{n+1}|\Omega_i|$).

The forward tracking variants of the methods of characteristics offer in general some advantages with respect to the backward tracking ones, especially concerning the treatment of boundary conditions or the control of mass balance errors [28]. Hence, we present a forward tracking (or redistributing) variant of (3.9) here.

To do so, for a chosen fixed index j one has to find all indices i such that the density ϕ_j^n occurs (at least once) in the r.h.s. of (3.11). For such i there exists then a pair of indices l and k (at least one), given by (3.5) and (3.6), such that $j = j_m$ with $k \le m \le l$. Consequently, the term in the l.h.s. of the following trivial equality

$$(3.13) \qquad v_{j_0 i} \frac{v_{j_1 j_0}}{v_{j_0}} \cdots \frac{v_{j_m j_{m-1}}}{v_{j_{m-1}}} \phi_{j_m}^n = v_{j_m j_{m-1}} \phi_{j_m}^n \frac{v_{j_{m-1} j_{m-2}}}{v_{j_{m-1}}} \cdots \frac{v_{j_1 j_0}}{v_{j_1}} \frac{v_{j_0 i}}{v_{j_0}}$$

occurs in (3.11). The order of indices in the l.h.s. of (3.13) corresponds to the notation of (3.11), and the r.h.s. of (3.13) will be used in the following algorithm.

Recall that $\tau = t^{n+1} - t^n$ is a chosen time step without the CFL condition (2.18). We denote by b_i the r.h.s. of (3.11), and we set initially $b_i = 0$ for $i = 1, \ldots, I$. We try to keep analogous notations to (3.13) and (3.11) in the following description of the algorithm.

To determine the values b_i, the following main program can be implemented:

```
for (i = 1, ..., I) {
        if (τ < τᵢ) then {
                bᵢ = bᵢ + φⁿᵢ (|Ωᵢ| − τvᵢ) ;
                τ̂ = τ ;
        }
        else
                τ̂ = τᵢ;
        jₘ = i;
        for (jₘ₋₁ ∈ Λᵒᵘᵗⱼₘ)
                DistributeMass(jₘ₋₁, tⁿ, τ̂, vⱼₘⱼₘ₋₁ φⁿⱼₘ) ;
}
```

The (recursive) procedure DistributeMass(j, t_0, $\hat{\tau}$, q) implements the consequent mass redistribution. This procedure may modify the value b_j and it may call itself, if necessary. The time $t_0 < t^{n+1}$ denotes an entering time of the mass contribution to Ω_j. The parameter $\hat{\tau}$ in DistributeMass corresponds to $\hat{\tau}_{j_m}$ in (3.12). The value q should accumulate the product on the r.h.s. of (3.13). The implementation of the procedure DistributeMass can take the following form:

```
DistributeMass(j, t₀, τ̂, q) {
        t₀ = t₀ + τⱼ;
        if (t₀ ≥ tⁿ⁺¹) then {
                bⱼ = bⱼ + τ̂ q ;
                return;
        }
        if (t₀ + τ̂ > tⁿ⁺¹) then {
                bⱼ = bⱼ + (τ̂ − (tⁿ⁺¹ − t₀)) q ;
                τ̂ = tⁿ⁺¹ − t₀ ;
        }
        jₘ₋₁ = j ;
        for (jₘ₋₂ ∈ Λᵒᵘᵗⱼₘ₋₁)
                DistributeMass(jₘ₋₂, t₀, τ̂, (vⱼₘ₋₁ⱼₘ₋₂ / vⱼₘ₋₁) q) ;
        return ;
}
```

4. Co-volume method for the motion by curvature. Although the previous finite volume method for the advective level set equation (2.1) is applicable for arbitrary computational meshes of polygonal finite volumes Ω_i, for the numerical discretization of the intrinsic diffusion part (the motion by curvature)

$$(4.1) \qquad \partial_t \phi = |\nabla \phi| \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right),$$

we apply the so called co-volume method [36, 12] (also called the vertex-centered finite volume method or the finite volume element method [2]). For a simplicity, we set $\mu = 1$ in (1.4).

More precisely, let $T^e \subset \Omega$, $e = 1, \dots, E$ be a mesh of finite elements for Ω (with the properties analogous to (2.5)), and let x_i, $i = 1, \dots, I$ be the vertices of this grid. Further, let $N_i = N_i(x)$ be the standard continuous finite element (FE) test functions that are polynomial for $x \in T^e$ and that fulfil $N_i(x_j) = \delta_{ij}$. In such a way, the FE interpolation $\hat{\phi}(t^n, x)$ of the values ϕ_i^n can be defined,

$$(4.2) \qquad \hat{\phi}(t^n, x) := \sum_{i=1}^{I} \phi_i^n N_i(x).$$

Using (4.2), the gradient $\nabla \hat{\phi}(t^n, x)$ is well defined for $x \in T^e$. An analogous definition to (4.2) is taken for $\hat{\phi}(t^{n+1}, x)$ and for $x \in T^e$ one can use

$$(4.3) \qquad \nabla \hat{\phi}(t^{n+1}, x) = \sum_{k \in \Lambda^e} \phi_k^{n+1} \nabla N_k(x),$$

where the set Λ^e contains all indices k such that $x_k \in T^e$.

The dual (complementary) mesh of finite volumes that fulfils (2.5) can be constructed by defining Ω_i around each vertex x_i. There exists some freedom in the construction of such vertex-centered finite volumes, but the most common choice is the so called barycenter-based finite volumes. In 2D case, they are obtained by connecting the edge-midpoints of elements with the barycenters of elements, see, e.g., [8] for more details.

In general, the boundary $\partial \Omega_i$ has the form

$$(4.4) \qquad \partial \Omega_i := \bigcup_{j \in \Lambda_i} \bigcup_{e \in \Lambda^{ij}} \overline{\Gamma_{ij}^e}, \quad \Gamma_{ij}^e := \partial \Omega_i \cap \partial \Omega_j \cap T^e.$$

The indices $e \in \Lambda^{ij}$ denote all elements T^e that contain the vertices x_i and x_j.

To describe a finite volume discretization of (4.1), we apply the following notations for the gradient of ϕ at $x_{ij} \in T^e$,

$$(4.5) \qquad \nabla^e \phi_{ij}^{n+1} := \nabla \hat{\phi}^{n+1}|_{T^e}(x_{ij}), \quad \nabla^e \phi_{ij}^n := \nabla \hat{\phi}^n|_{T^e}(x_{ij}),$$

where $x_{ij} := 0.5(x_i + x_j)$. Finally, we introduce

$$(4.6) \qquad |\nabla \phi_i^n| := \frac{1}{|\Omega_i|} \sum_{j \in \Lambda_i} \sum_{e \in \Lambda^{ij}} \frac{|\Omega_i \cap T^e|}{2} |\nabla^e \phi_{ij}^n|.$$

Now, one can integrate (4.1) over $x \in \Omega_i$ and $t \in (t^n, t^{n+1})$, and using (4.6) to approximate $\nabla \phi(t, x)$, one can, firstly, obtain

$$(4.7) \qquad \phi_i^{n+1}|\Omega_i| = \phi_i^n|\Omega_i| - |\nabla \phi_i^n| \int_{t^n}^{t^{n+1}} \int_{\partial \Omega_i} \frac{1}{|\nabla \phi|} \vec{n} \cdot \nabla \phi \, d\gamma dt.$$

Using (4.5) afterwards, one can derive the following discretization scheme

$$(4.8) \qquad \phi_i^{n+1}|\Omega_i| + |\nabla\phi_i^n|\tau \sum_{j\in\Lambda_i}\sum_{e\in\Lambda^{ij}} |\Gamma_{ij}^e|\frac{\vec{n}_{ij}^e \cdot \nabla^e\phi_{ij}^{n+1}}{|\nabla^e\phi_{ij}^n|} = \phi_i^n|\Omega_i| \ ,$$

where $\vec{n}_{ij}^e$ denotes the outer unit normal to Γ_{ij}^e.

The linear discrete system of equations (4.8) represents the semi-implicit co-volume discretization of the motion by curvature (4.1). In general, zero gradients in solution can occur and therefore we regularize the scheme in the sense of Evans and Spruck [5] by replacing

$$(4.9) \qquad\qquad |s| \approx \sqrt{\varepsilon + |s|^2}\,.$$

5. Flux-based level set method. The previous Section §4 described the co-volume discretization method for the intrinsic diffusion equation, i.e., the motion by curvature. This method exploits a finite volume mesh that is complementary (dual) to a standard finite element mesh. This gives us two important advantages - firstly, the gradient of numerical solution is easy to compute from the nodal values, and, secondly, the co-volume discretization can be viewed as a finite volume method that contains the terms $\phi_i^n|\Omega_i|$ representing the mass in Ω_i at $t = t^n$.

If the discretization scheme for the advective level set equation, described in Section §3, is realized on the same finite volume mesh as the co-volume method for the motion by curvature, the both discretization methods are easy to combine for the advection-diffusion level set equation (1.4).

The flux-based level set method can be straightforwardly explained using the standard operator splitting procedure. In its simplest two-step variant, the first step consists of solving the advective part (2.1) of (1.4) for a given initial condition. The second step is realized afterwards by taking the result of the first step as the initial condition and solving the diffusive part (4.1) of (1.4).

Comparing the l.h.s. of (2.14) and the r.h.s. of (4.8), these two steps can be directly combined to the basic finite volume discretization scheme of the flux-based level set method

$$(5.1) \qquad \phi_i^{n+1}|\Omega_i| + |\nabla\phi_i^n|\tau \sum_{j\in\Lambda_i}\sum_{e\in\Lambda^{ij}} |\Gamma_{ij}^e|\frac{\vec{n}_{ij}^e \cdot \nabla^e\phi_{ij}^{n+1}}{|\nabla^e\phi_{ij}^n|} =$$

$$u_i^n|\Omega_i| + \tau \sum_{j\in\Lambda_i^{in}} \phi_j^n\, v_{ji}^n - \tau v_i^n \phi_i^n\,,$$

where v_{ij}^n and v_i^n are equal to v_{ij} in (2.10) and v_i in (2.15) for the choice $\vec{v}(x) = -\delta\frac{\nabla\hat{\phi}^n(x)}{|\nabla\hat{\phi}^n(x)|}$.

The discretization scheme (5.1) corresponds to the flux-based level set method when the CFL condition (2.18) is fulfilled, otherwise, the r.h.s. of (5.1) must be replaced with the r.h.s. of (3.8).

6. Discussion on numerical experiments. In this section, we present numerical computations by the flux-based level set method in the case of curve evolution equation (1.3) in the level set formulation (1.4).
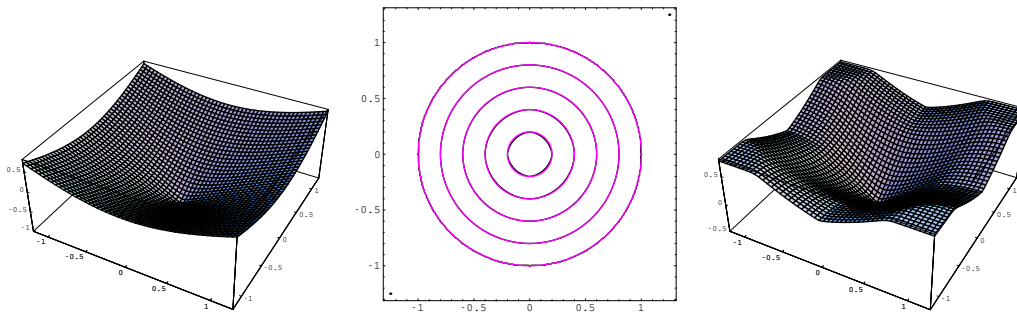
FIG. 6.1. *The initial distance function $\phi(0, x)$ (the left picture) and the numerical solution at time $t = 0.5$ (the right picture) for the pure advective case of shrinking circle. The middle picture shows the zero level lines of the exact (violet lines) and numerical solution (black lines) at the time points $t = 0, 0.2, 0.4, 0.6, 0.8$ using the time step $\tau = 0.04$ that violates the CFL condition. Note that in this plot resolution the exact and numerical solution can not be distinguished except a slight difference at $t = 0.8$.*

6.1. Comparison with exact solutions. In next examples, we give a comparison with simple known exact solutions of the curve evolution problem. Namely, in Figures 6.1–6.2 and in Tables 6.1-6.5, we compare the exact evolving circles with the corresponding zero level lines obtained by the numerical solution of the level set equation (1.4).

In all 2D experiments presented in this section, if not stated otherwise, we use the spatial domain $\Omega = [-1.25, 1.25] \times [-1.25, 1.25]$, which is split into $M \times M$ rectangular finite elements with the side length $h = 2.5/M$. The dual co-volume grid is obtained by shifting this grid by $\frac{h}{2}$. The regularization parameter in (4.9) is $\varepsilon = 10^{-4}$.

To compare numerical and exact solutions, in every discrete time step $n = 0, ..., N$ we find all zero crossing points $r_i^n, i = 1, ..., K$, of piecewise linear representation of numerical solution with finite element grid lines. Then we compute distances from origin of all $r_i^n, i = 1, ..., K$ and compare them with radius $r(n\tau)$ of the exact evolving circle. Then the formula

$$\text{Error} \ = \ \left(\sum_{n=0}^{N} \tau \frac{1}{K} \sum_{i=1}^{K} (r_i^n - r(n\tau))^2 \right)^{\frac{1}{2}}$$

is used as $L^2((0, T), L_2(S^1))$-norm, S^1 is a unit circle, $T = N\tau$, of the difference of exact and numerical curve (similar strategy can be used also in 3D). In the next two subsections we discuss comparison of numerical and exact solutions in advective and curvature driven cases.

6.1.1. Purely advective case. In the left picture of Figure 6.1, we plot the initial (distance) function $\phi(0, x)$, and, on the right, the numerical solution $\phi(t, x)$ at $t = 0.5$ for purely advective case ($\mu = 0$, $\delta = -1$) to illustrate the shape of the evolving level set function.

In the middle picture of Figure 6.1, the level lines of exact and numerical solution are plotted at the time points $t = 0, 0.2, 0.4, 0.6, 0.8$. The exact radius of a shrinking circle is given by $r(t) = r(0) - t$, $t \in [0, T]$, $T = r(0)$. The numerical solution was obtained using $M = 250$ and the time step $\tau = 0.005$ that respects the standard CFL condition.

τ	0.005	0.01	0.02	0.04
steps	160	80	40	20
Error	3.297e-3	3.2e-3	3.34e-3	3.443e-3
CPU	290 s	158 s	94 s	104 s

TABLE 6.1

Errors in $L^2((0,0.8), L_2(S^1))$-norm and CPU times for purely advective case of shrinking circle using time steps of different sizes. The latter three cases violate the standard CFL condition.

M	N	Error	EOC	N	Error	EOC	N	Error	EOC
10	4	4.253e-2		2	4.754e-2		1	6.425e-2	
20	8	1.820e-2	1.22	4	1.945e-2	1.29	2	2.394e-2	1.42
40	16	8.586e-3	1.08	8	8.812e-3	1.14	4	9.996e-3	1.26
80	32	4.148e-3	1.04	16	4.109e-3	1.10	8	4.416e-3	1.17
160	64	2.033e-3	1.03	32	2.024e-3	1.02	16	2.089e-3	1.08
320	128	1.002e-3	1.02	64	9.976e-4	1.02	32	1.017e-3	1.04

TABLE 6.2

Errors in $L^2((0,0.5), L_2(S^1))$-norm and EOC for purely advective case of shrinking circle using different time step $\tau = 0.5/N$.

In Table 6.1, we present errors and CPU times of computations by the flux-based level set method on the same grid $M = 250$ using bigger time steps that violate the CFL condition. Note that no significant differences in the precision could be observed, while the CPU times decreased on 1GHz Linux PC, and the minimum-maximum principle remained valid. Table 6.2 shows that the method is of order h for any of these time steps. The same results were obtained also in 3D case comparing numerical and exact shrinking spheres.

Further, we present more complex experiment where the initial circle is expanding ($\delta = 0.05$, $\mu = 0$) and rotating in the counterclockwise direction due to the variable external velocity vector field equals to $(-y, x)$.

The top left picture in Figure 6.7 shows numerically computed evolution starting from the smallest most right circle and ending with the largest most left circle. The evolution takes the half of rotation approximately. Other plots in Figure 6.7 show the initial circle, the exact final circle and its numerical approximation depending on a grid refinement. These images are presented in order to show the area loss of the method.

It has been reported recently [23, 4, 10] that finite volume based numerical methods (or related discontinuous Galerkin methods) exhibits a better area/volume preservation property than some standard level set methods. In our experiment the differences in encompassing area of the numerically computed interface and the exact one are clearly visible, especially in the case of coarse grids. However, the convergence to exact solution (and correspondingly diminishing of difference in exact and numerical areas) can be observed taking subsequently $M = 20, 40, 80, 160, 320$.

The convergence of the method for this experiment is reported in Table 6.3. Since the acceptable results are obtained on fine grids when the CPU time can be critical, the speed up of computation without losing stability is an important issue. In Table 6.4 we show for this example that the flux-based level set method gives the stable solution with the same error on the finest grid but with several times enlarged time steps and thus speeding up the computations (CPU times reported were obtained on

M	τ	N	Error	EOC
10	0.125	25	0.369	
20	0.0625	50	0.179	1.04
40	0.03125	100	0.0869	1.04
80	0.015625	200	0.0428	1.02
160	0.0078125	400	0.0212	1.01
320	0.00390625	800	0.0105	1.01

TABLE 6.3

Errors in $L^2((0, 3.125), L_2(S^1))$-norm and EOC for rotating and expanding circle.

τ	0.00390625	0.0078125	0.015625	0.03125
steps	800	400	200	100
Error	0.0105	0.0102	0.0103	0.0105
CPU	560 s	300 s	198 s	457 s

TABLE 6.4

Errors in $L^2((0, 3.125), L_2(S^1))$-norm and CPU for rotating and expanding circle using increasing time steps.

2.4 GHz Linux PC).
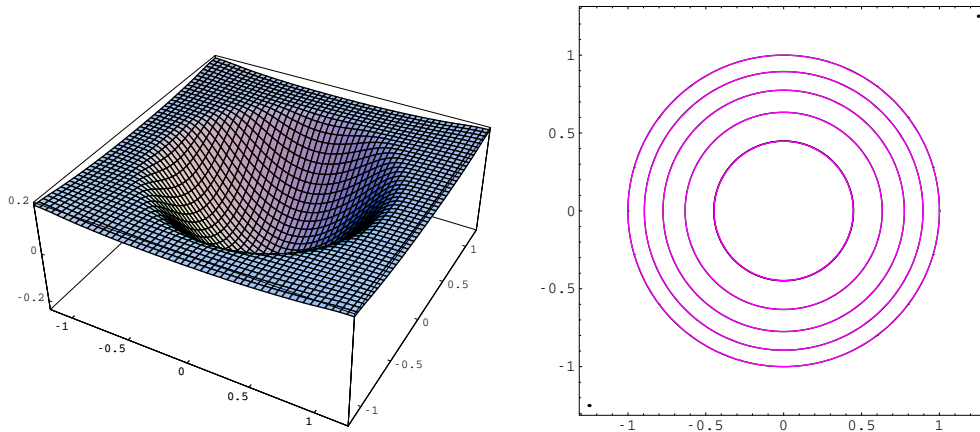
FIG. 6.2. *The left picture shows the numerical solution $\phi(t, x)$ at $t = 0.3$. The right picture compares the exact (violet lines) and the numerical zero level lines (black lines), but due to plot resolution no differences might be visible. The curvature driven case with $\mu = 1, \delta = 0$ was computed in this experiment for the initial distance function plotted in Figure 6.1.*

6.1.2. Purely curvature driven case. The level lines in Figure 6.2 represent the computed evolution of the unit circle in the curvature driven case $\mu = 1, \delta = 0$ at time points $t = 0, 0.1, 0.2, 0.3, 0.4$ using time step $\tau = 0.0005$, $M = 250$. The exact radius of a shrinking circle is described by $r(t) = \sqrt{r(0)^2 - 2t}$, $t \in [0, T]$, $T = r(0)^2/2$. Again, the evolving zero level set of the numerical and the exact solution coincide, error in $L^2((0, 0.25), L_2(S^1))$-norm is equal to 0.00006257. Although the time step τ is 10 times bigger than the usual stability constrain for explicit schemes ($\frac{h^2}{2}$, $h = 0.01$ in this experiment), the error in curve evolution is proportional to h^2.

To show the second order accuracy of our method for the mean curvature flow

M	τ	steps	Error	EOC
40	0.064	10	3.81e-4	
80	0.004	40	8.92e-5	2.09
160	0.001	160	2.18e-5	2.03

TABLE 6.5

Errors in $L^2((0,0.16), L_2(\Omega))$-norm, and EOC comparing numerical and exact solutions of the mean curvature flow in 3D case.

in a more detailed way, we present results for a 3D testing example where the exact solution of equation (4.1) is given by [22]

$$(6.1) \qquad \phi(x,y,z,t) = (x^2 + y^2 + z^2 - 1)/4 + t.$$

The problem is solved in $\Omega = [-1.25, 1.25]^3$ and for $T = 0.16$. The Dirichlet boundary conditions on $\partial\Omega$ given by (6.1) are used. The grid size is $h = 2.5/M$ using a subsequent grid refinement with $M = 40, 80, 160$. The time steps are chosen proportionally to $8 \times \frac{h^2}{2}$ (i.e., several times bigger than the usual stability constrain of explicit schemes), and $\varepsilon = 10^{-12}$ in (4.9). The errors are measured on the whole domain using $L_2((0,T), L_2(\Omega))$-norm, and the results are reported in Table 6.5.

6.2. Nontrivial 2D example. In the next nontrivial experiment, we evolve an initial curve in the form of a slightly rotated quatrefoil. The quatrefoil is zero level set of initial level set function constructed by the formula

$$\phi(x,y) = -1 + \sqrt{x^2 + y^2}/r_L, \quad r_L = 0.6 + 0.4\sin\left(4 \ \text{arctg}\left(\frac{y}{x}\right)\right).$$

In this experiment, $\mu = 0.0005$, $\delta = -1$, $M = 250$, and, as we see later, some topological changes of the evolving curve will occur. In Figure 6.3, the isolines and the 3D graph of the initial level set function are plotted.

The subsequent level lines, corresponding to the evolving curve, are plotted in the right picture of Figure 6.6 (the valid CFL condition, 40 time steps took 72 seconds of CPU time) and in pictures of Figure 6.4 (the CFL condition violated, the CPU times were 44 seconds and 24 seconds, see the text in caption of Figure 6.4). Note that at the time $t = 0.15$ one can observe the curve has split into 5 disconnected components which then extinct later independently.

In Figures 6.5 and 6.6 we illustrate a good behaviour of the basic finite volume scheme (2.16) when starting with a piecewise constant initial level set function. The graph of initial (binary) level set function is plotted in the right picture and the level set in the left picture of Figure 6.5. Comparing the results in Figure 6.6 for discontinuous and smooth initial level set functions, one can see a very good agreement.

In fact, this example suggests that the flux-based level set method might be used in a localized form like in [26], because the numerical gradient of level set function is zero everywhere except in a narrow band near the interface. This property results in a trivial form of the scheme (2.16) for nodes faraway from the interface that is easy to eliminate from the system of linear equations. Of course, in this case we can not directly use the time steps that violate the CFL condition.

In the last example of this subsection we evolve the same nontrivial initial quatrefoil curve by the mean curvature flow, and we compare numerical solutions given by the scheme (4.8) and by a conceptually completely different numerical method based on Lagrangean approach [20]. We use parameters $\mu = 1, \delta = 0, \tau = 0.001, M = 250$
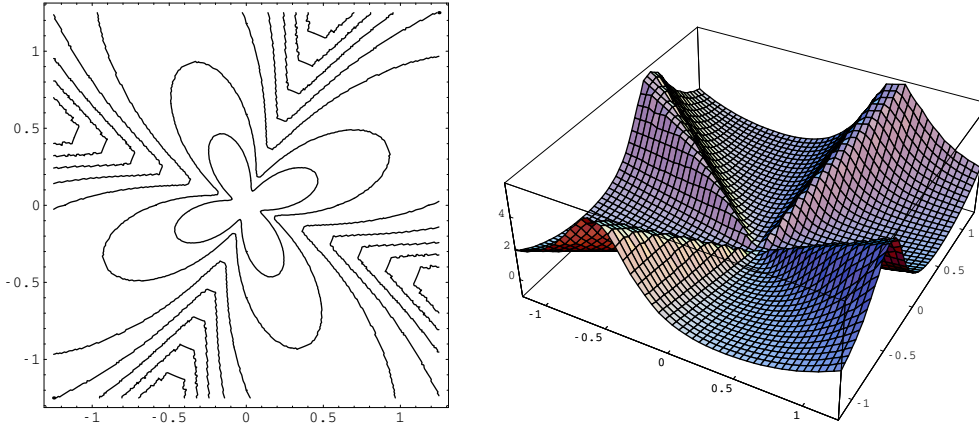
FIG. 6.3. *Level lines (the left picture) and the 3D graph (the right picture) of the initial function, the zero level line of which is a quatrefoil, which we let evolve by (1.4) with $\mu = 0.0005$ and $\delta = -1$, cf. Figure 6.4.*
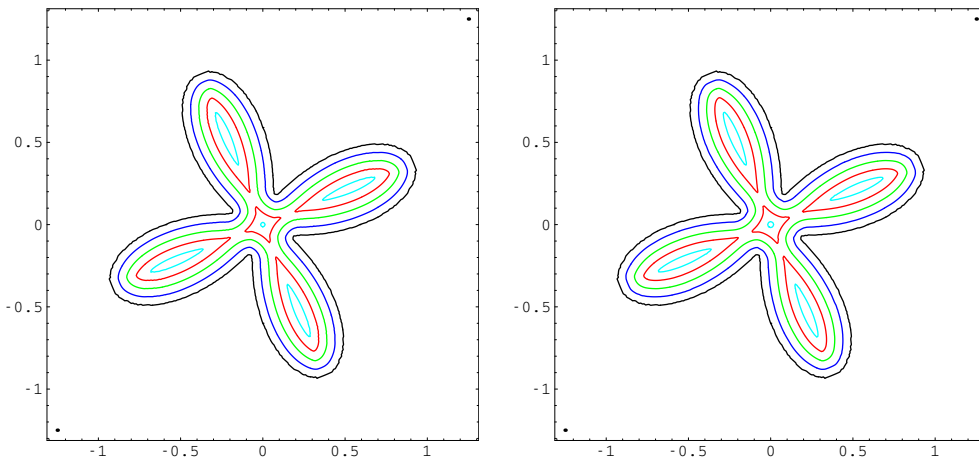

FIG. 6.4.   *The left picture shows the time evolution of zero level line at time points* $0, 0.05, 0.1, 0.15, 0.20$ *with the time step* $\tau = 0.01$, *(the CPU time for 20 time steps = 44.4 seconds) starting with smooth initial level set function, cf. Figure 6.3, and similarly for the right picture, but with the time step* $\tau = 0.025$ *(the CPU time for 8 steps = 23.84 seconds). In both cases the CFL condition was violated.*

and $\varepsilon = 10^{-6}$ in (4.9). The results are plotted in Figure 6.8. We see that although the level set function is distorted a lot with respect to its initial shape, cf. Fig. 6.3, and we do not use any redistancing strategy, both numerical solutions correspond accurately to each other. Our observation is that it is not necessary to keep strictly the unit slope along interface (and thus to do redistancing) using the flux-based level set method.

6.3. 3D experiment. In this subsection we present experiment where we evolve initial surface given by a special spherical 3D Legendre polynomial (see Figure 6.9 top
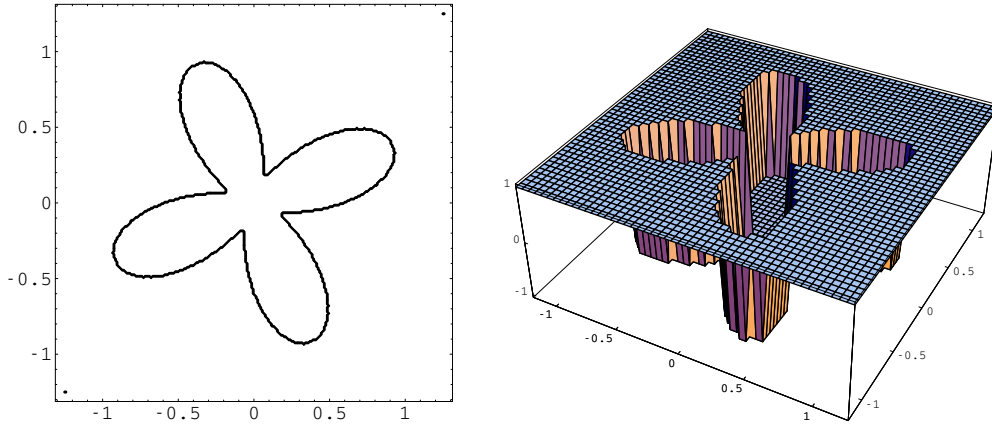
Fig. 6.5. *The isoline (the left picture) and the 3D graph (the right picture) of the piecewise constant initial level set function, the zero level line of which is a quatrefoil.*
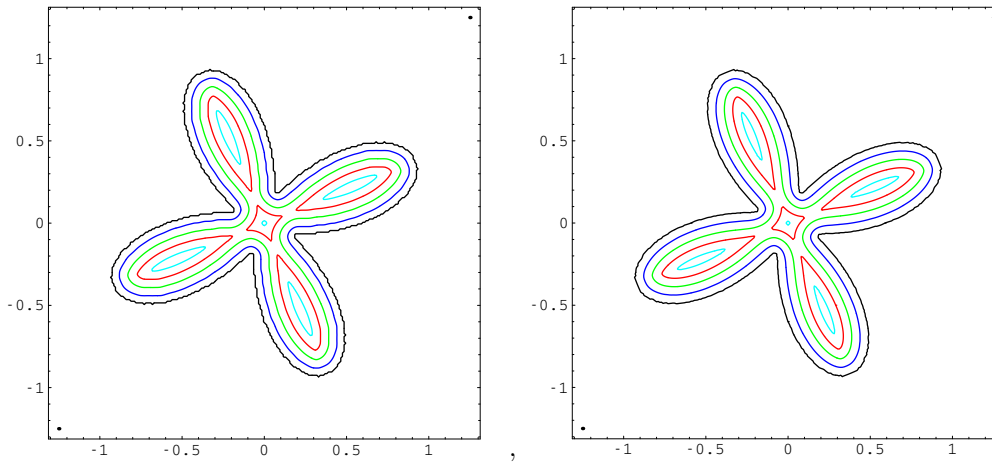

Fig. 6.6. *The left picture shows the time evolution of zero level line when starting with the piecewise constant initial function, cf. Figure 6.5, using the time step $\tau = 0.0025$. The right picture shows the evolution of zero line when starting with smooth initial level set function, cf. Figure 6.3, using the time step $\tau = 0.005$. 6.4.*

left) represented by the relation $r = r_L$, where

$$(6.2) \qquad r = \sqrt{x^2 + y^2 + z^2}, \quad r_L = 0.6 \left| 3 \left(\frac{y}{r} \right)^2 - 1 \right|.$$

We construct a piecewise constant initial level set function which is equal to -1 if $r < r_L$ and to 1 if $r \geq r_L$, and shrink this surface using $\mu = 0$, $\delta = -1$. We split the computational domain $[-1.25, 1.25]^3$ to 80^3 cells, and we use time step $\tau = 0.005$ for which the CFL condition is fulfilled. In Figure 6.9, topological changes in shrinking surface are nicely observable from 3D isosurface visualizations and from 2D cuts of evolving zero level set.

7. Conclusions. We introduced the new flux-based level set method for evolving interfaces. It is based on the finite volume discretization of the nonlinear degenerate

advection-diffusion level set equation. It enables to solve the problem in a stable and efficient way. Using the recursive flux redistribution among control volumes and the semi-implicit treatment of intrinsic diffusion, we remove the standard CFL condition on time step and improve significantly the CPU times without deteriorating the precision of the method. The method was presented for 2D and 3D curve and surface evolutions depending on curvature and subject to constant driving force and spatially varying velocity field. We use first order upwind scheme in advective part of (1.4) which is known as L_∞ stable method. Although higher order schemes are well-known for these type of problems, in many applications, see e.g. [33, 29, 32], the first order schemes are widely used in practice. Removing the CFL stability condition for them is then highly desirable, e.g., when advective part dominates the curvature part since the precision of the semi-implicit scheme for the (mean) curvature motion is reasonable also for larger time steps.

We describe our method for general grids, but in this paper we restricted our computations only to structured meshes. An extension of our finite volume method to second order approximation for general unstructured and adaptively refined grids was recently done in [10]. Development of the second order method has been motivated by the second order accuracy in intrinsic diffusion part and ambition to get the same order of approximation also in the advective part and by a requirement to get better area (volume) conservation properties. Moreover, the adaptive grid refinement is natural tool for our method to concentrate computational effort locally along a moving interface as it is usual in other froms of local level set methods [26].

REFERENCES

[1] M. Beneš, *Mathematical and computational aspects of solidification of pure crystallic materials,* Acta Math. Univ. Comenian., 70 (2001) pp. 123–151.

[2] Z. Cai, *On the finite volume element method,* Numer. Math., 58 (1991) 713–735.

[3] V. Caselles, R. Kimmel, G. Sapiro, *Geodesic active contours,* International Journal of Computer Vision, 22 (1997) pp. 61–79.

[4] D.A. Di Pietro, S. Lo Forte, N. Parolini, *Mass preserving finite element implementations of the level set method,* Applied Numerical Mathematics, (2006), in press.

[5] L. C. Evans, J. Spruck, *Motion of level sets by mean curvature I,* J. Differential Geometry, 33 (1991) pp. 635–681.

[6] R. Eymard, T. Gallouet, R. Herbin, *The finite volume method,* in: Handbook for Numerical Analysis, Vol.7 (Ph. Ciarlet, J. L. Lions, eds.), Elsevier, 2000.

[7] M. Falcone, T. Giorgi, P. Loreti, *Level sets of viscosity solutions: some applications to fronts and randez-vous problems,* SIAM J. Applied Math., 54 (1994) pp. 1335–1354.

[8] P. Frolkovič, *Discretization,* in D^3F - ein Programmpaket zur Modellierung von Dichteströmungen (E. Fein, Ed.), Braunschweig, 1999. GRS-139, ISBN 3-923875-97-5, (also available at http://www.iwr.uni-heidelelberg.de/∼Peter.Frolkovic).

[9] P. Frolkovič, *Flux-based method of characteristics for contaminant transport in flowing groundwater,* Computing and Visualization in Science, 5(2) (2002) pp. 73–83.

[10] P. Frolkovič, K.Mikula, *High-resolution flux-based level set method.* Preprint 2005-12, Department of Mathematics and Descriptive Geometry, Slovak University of Technology, Bratislava (2005)

[11] I. GINZBURG, G. WITTUM *Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolants.* J. Comput. Phys., 166 (2001) pp. 302–335.

[12] A. HANDLOVIČOVÁ, K. MIKULA, F. SGALLARI, *Semi-implicit complementary volume scheme for solving level set like equations in image processing and curve evolution,* Numer. Math., 93 (2003) pp. 675–695.

[13] T. IKEDA, Maximum principle in finite element models for convection-diffusion phenomena, North-Holland, Amsterdam, New York, Oxford, 1983.

[14] G.S. JIANG, D. PENG, *Weighted ENO schemes for Hamilton-Jacobi equations,* SIAM J. Sci. Comput., 21 (2000) pp. 2126–2143.

[15] G. KOSSIORIS, CH MAKRIDAKIS, P.E. SOUGANIDIS, *Finite volume schemes for Hamilton-Jacobi equations,* Numer. Math., 83 (1999) pp. 427–442.

[16] R.J. LEVEQUE, Finite Volume Methods for Hyperbolic Problems, Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002.

[17] X.Y. LUO, M.J. NI, A. YING, M.A. ABDOU *Numerical Modeling for Multiphase Incompressible Flow with Phase Change,* Numerical Heat Transfer Part B: Fundamentals, 48 (2005) pp. 425–444.

[18] K. MIKULA, J. KAČUR, *Evolution of convex plane curves describing anisotropic motions of phase interfaces,* SIAM J. Sci. Comput., 17 (1996) pp. 1302–1327.

[19] K. MIKULA, N. RAMAROSY, *Semi-implicit finite volume scheme for solving nonlinear diffusion equations in image processing,* Numer. Math., Vol.89, No.3 (2001) pp. 561–590.

[20] K. MIKULA, D. ŠEVČOVIČ, *Evolution of plane curves driven by a nonlinear function of curvature and anisotropy,* SIAM J. Appl. Math., 61 (2001) pp. 1473–1501.

[21] K. MIKULA, D. ŠEVČOVIČ, *Computational and qualitative aspects of evolution of curves driven by curvature and external force,* Computing and Visualization in Science, Vol. 6, No. 4 (2004) pp. 211–225.

[22] A.M. OBERMAN, *A convergent monotone difference scheme for motion of level sets by mean curvature,* Numer. Math., DOI: 10.1007/s00211-004-0566-1, 2004.

[23] E. OLSSON, G. KREISS *A conservative level set method for two phase flow* J. Comput. Phys., 210 (2005) pp. 225–246.

[24] S. OSHER, J. SETHIAN, *Fronts propagating with curvature dependent speed: algorithm based on Hamilton-Jacobi formulation,* J. Comput. Phys., 79 (1988) pp. 12–49.

[25] S. OSHER, R. FEDKIW, Level set methods and dynamic implicit surfaces Springer-Verlag, 2003.

[26] D. PENG, B. MERRIMAN, S. OSHER, H.K. ZHAO, M. KANG, *A PDE based fast local level set method,* J. Comput. Phys., 155 (1999) pp. 410–438.

[27] E. ROUY, A. TOURIN, *A viscosity solutions approach to shape-from-shading,* SIAM J. Numer. Anal., 29 (1992) pp. 867–884.

[28] T.F. RUSSELL, M.A. CELIA, An overview of research on Eulerian-Lagrangian localized adjoint methods (ELLAM), Advances in Water Resources, 25 (2002), pp. 1215–1231.

[29] G. SAPIRO, Geometric Partial Differential Equations and Image Analysis, Cambridge University Press, 2001

[30] A. SCHMIDT, *Computation of three dimensional dendrites with finite elements,* J. Comput. Phys., 125 (1996) pp. 293–312.

[31] C.W. SHU, S. OSHER, *Efficient Implementation of Essentially Nonoscillatory Shock Capturing Schemes,* J. Comput. Phys., 77 (1988) pp. 439–471.

[32] A. SARTI, R. MALLADI, J.A. SETHIAN, *Subjective Surfaces: A Method for Completing Missing Boundaries,* Proceedings of the National Academy of Sciences of the United States of America, Vol. 12, No. 97 (2000) pp. 6258-6263.

[33] 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, New York, 1999.

[34] J. STRAIN, *Semi-Lagrangian methods for level set equations,* J. Comput. Phys., 151 (1999) pp. 498–533.

[35] A.-K. TORNBERG, B. ENGQUIST, *A Finite Element Based Level Set Method for Multiphase Flow Applications,* Computing and Visualization in Science, 3 (2000) pp. 93-101.

[36] N.J. WALKINGTON, *Algorithms for computing motion by mean curvature,* SIAM J. Numer. Anal., 33 (6) (1996) pp. 2215–2238.
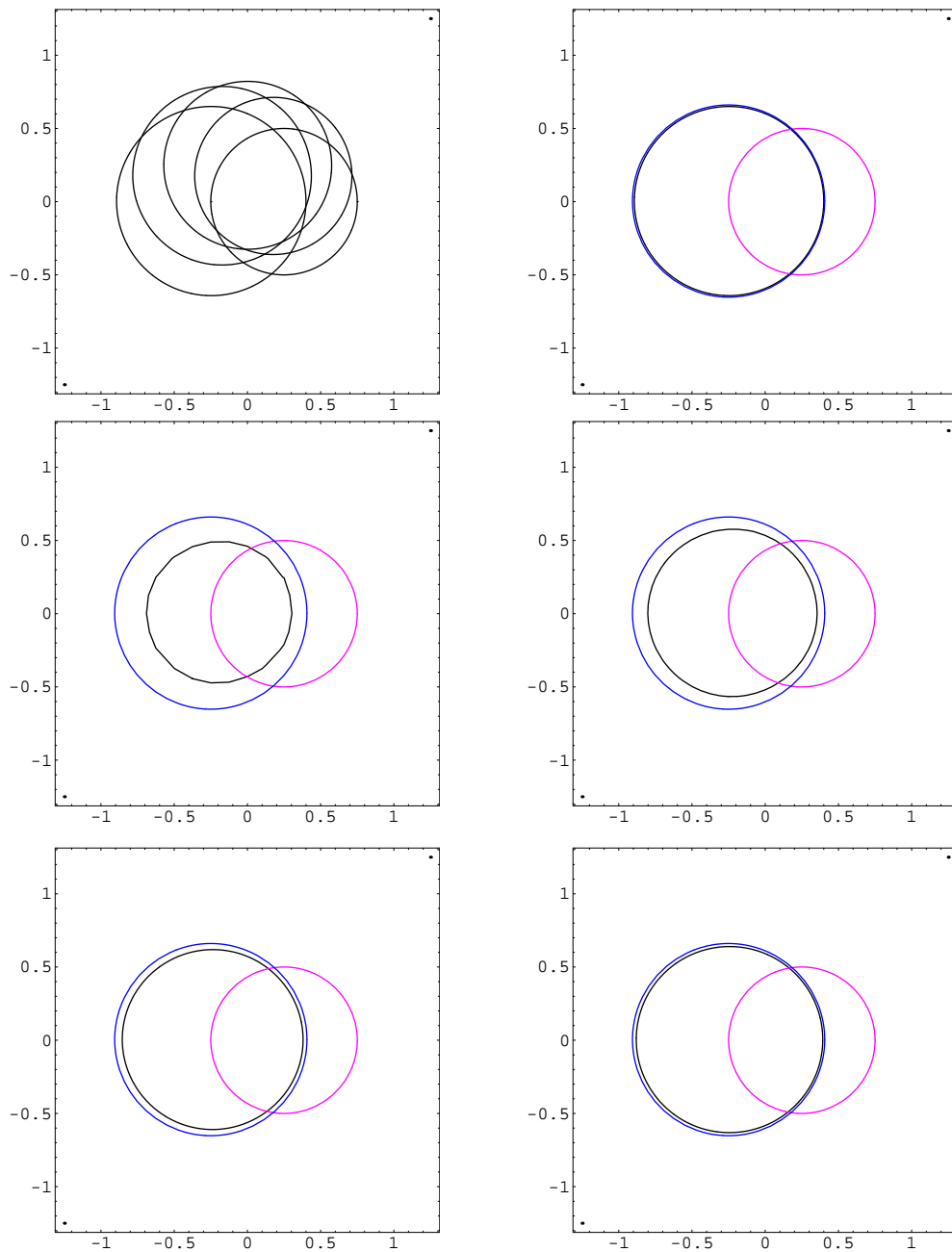
Fig. 6.7. *Numerical solutions of the example with expanding and rotating circle. The top left picture shows the evolution of the zero level line starting with the smallest circle (right) and ending with the largest circle (left) using the grid with $M = 320$. The top right picture shows for the same grid the initial zero line (the left circle) and the final zero lines of the exact and numerical solution. All other pictures illustrate the area loss of the method on coarser grids, particularly for $M = 20$ (middle left), $M = 40$ (middle right), $M = 80$ (bottom left), and $M = 160$ (bottom right).*
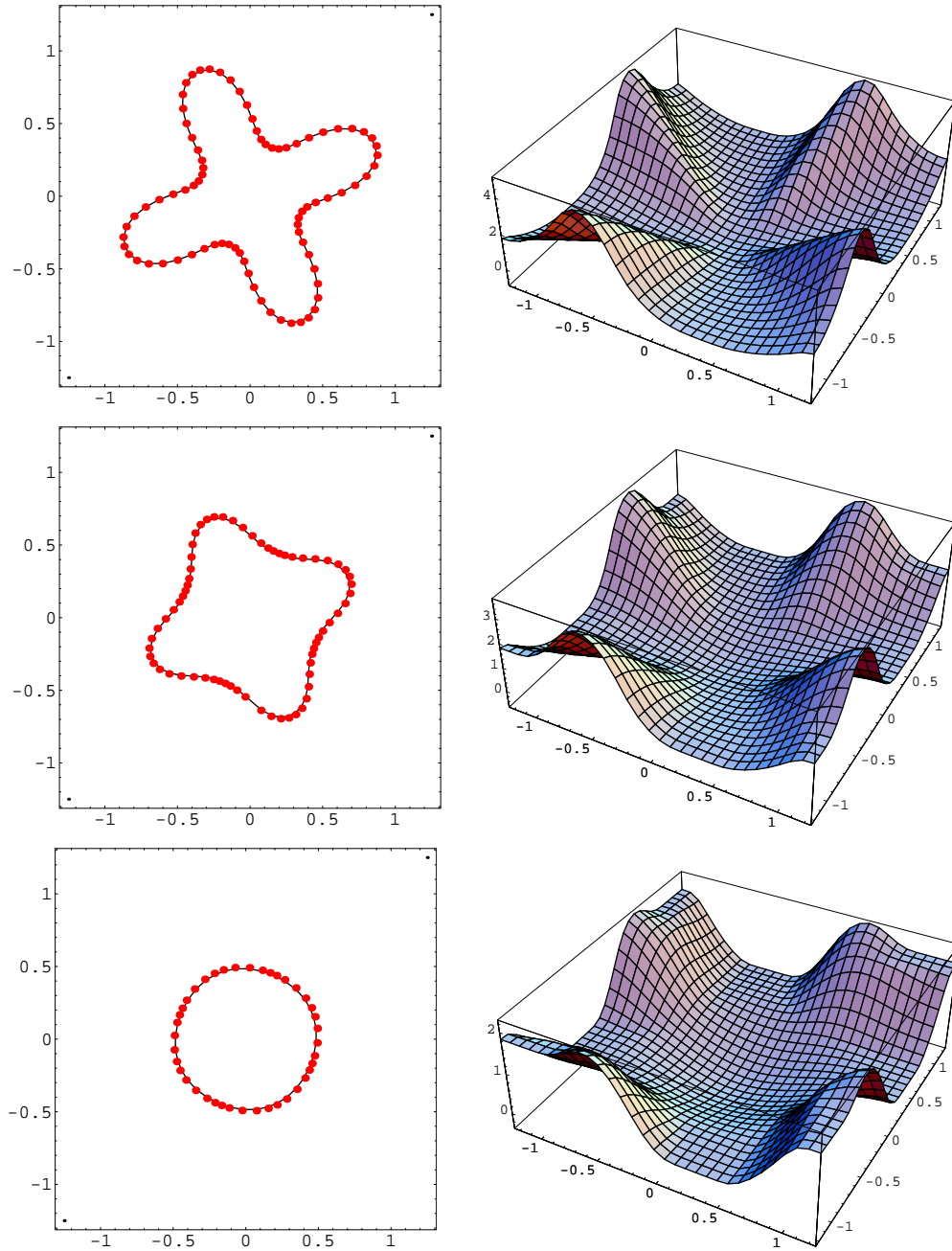
Fig. 6.8. *The left pictures show the comparison of numerical solutions to the mean curvature flow obtained by Lagrangian method (the points) [20] and by the flux-based level set method (the lines) plotted in time points* 0.01, 0.04 *and* 0.1. *The right pictures show the shapes of evolving level set function.*
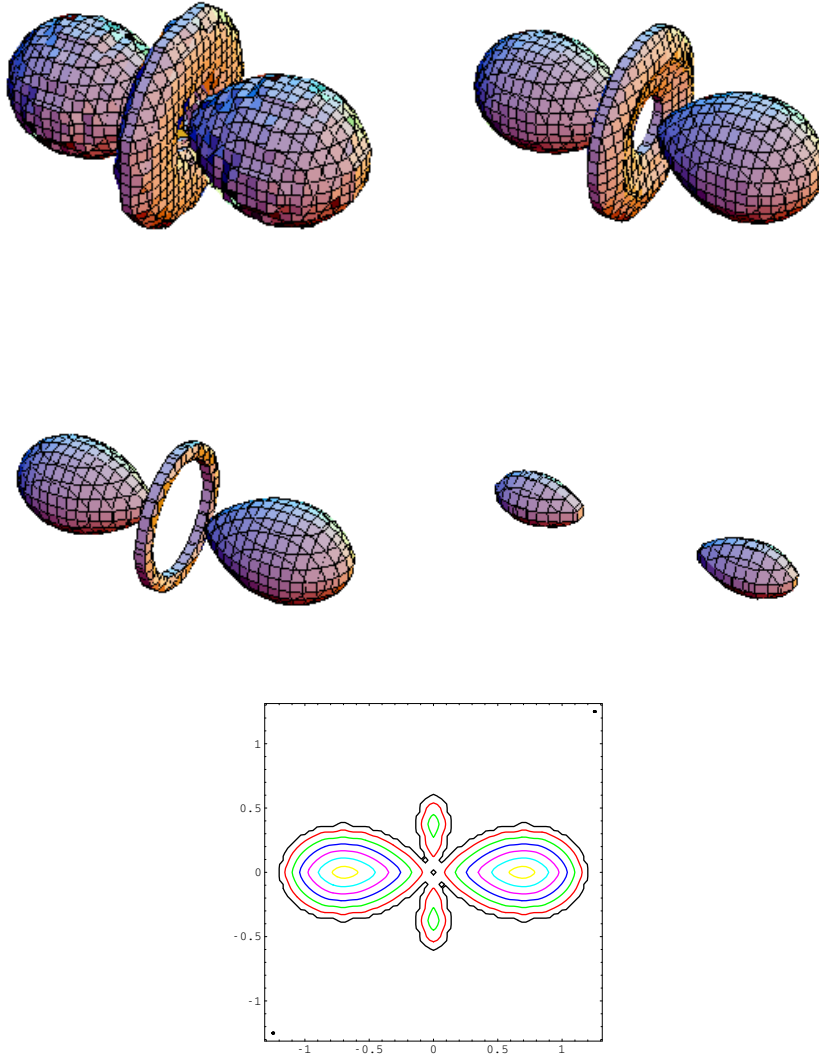
FIG. 6.9. *Evolution of 3D Legendre function, $F = -1$, $h = 2.5/80$, $\tau = 0.005$, time points $0, 10, 20, 40$ visualized as 3D isosurfaces, and 2D cuts of evolving isosurface at time points $0, 10, 20, 30, 40, 50, 60$.*