USE OF RADIAL BASIS FUNCTIONS AND LEVEL SET METHOD FOR SOLVING ONE-DIMENSIONAL PHASE CHANGE PROBLEMS

Leopold Vrankar[1], Edward J. Kansa[2], Franc Runovc[3], Goran Turk[4]

[1]Slovenian Nuclear Safety Administration
Železna cesta 16, Ljubljana, Slovenia
[2]Department of Mechanical and Aeronautical Engineering,
University of California, USA
[3]University of Ljubljana, Faculty of Natural Sciences and Engineering
Aškerčeva cesta 12, Ljubljana, Slovenia
[4]University of Ljubljana, Faculty of Civil and Geodetic Engineering
Jamova cesta 2, Ljubljana, Slovenia

leopold.vrankar@gov.si(Leopold Vrankar)

Abstract

Mathematically, the interface motion is expressed implicitly in an equation for the conservation of thermal energy at the interface (Stefan's conditions). This introduces a non-linear character to the system which treats each problem somehow uniquely. Due to their wide range of applications the phase change problems have drawn considerable attention of mathematicians, engineers and scientists. These problems are often called Stefan's or moving boundary value problems. One common feature of phase change problems is that the location of the solid-solid interface is not known a priori and must be determined during the course of analysis. The solution is obtained either by analytical solution or numerical methods. Recently, the numerical methods have focused on the idea of using a mesh-free methodology for the numerical solution of partial differential equations based on radial basis functions. Level set methods have become an attractive design tool for tracking, modelling and simulating the motion of free boundaries in fluid mechanics, combustion, computer animation and image processing. The surface itself is zero level set of an implicit function in a higher dimension. In our case we will study solid-solid transformation. The numerical solutions will be compared with analytical solutions. In our work we will present radial basis functions and level set methods for solving one-dimensional Stefan's problems.

Keywords: Radial Basis Functions, Level set Method, Dynamic interfaces.

Presenting Author's Biography

Leopold Vrankar was born in Trbovlje, Slovenia. He received his PhD in Geotechnology at University of Ljubljana, Faculty of Natural Sciences and Engineering in 2004, with dissertation modeling of radionuclide migration through porous material with meshless method. At the moment he is working at the Slovenian Nuclear Safety Administration.

1 Introduction

One common feature of phase change problems is that the location of the solid-liquid or solid-solid interface is not known apriori and must be determined during the course of analysis. Phase change problems can be found in the safety studies of nuclear reactors, casting of metals, semiconductor manufacturing, geophysics and industrial applications involving metals, oil, and plastics. Due to their wide range of applications the phase change problems have drawn considerable attention of mathematicians, engineers and scientists. These problems are often called Stefan's or moving boundary value problems.

The exact solution of phase change problems is limited exclusively to the cases in which e.g. the heat transfer regions are infinite or semi-infinite one dimensional-space. Therefore, solution is obtained either with approximate analytical solution or by numerical methods.

The mesh-free method has been widely investigated in the past and emerged as a new category of computational methods. One of its advantages is that no mesh generation is required to solve differential equations numerically. It is well known that in the traditional numerical methods, such as finite element methods, finite difference methods, boundary element methods, it is usually difficult and takes considerable effort to generate proper meshes for computational purposes. This is especially true for three–dimensional problems with complicated geometry in engineering applications. One of the common characteristics of all mesh-free methods is their ability to construct functional approximation or interpolation entirely based on the information given from a set of scattered nodes.

In our case we will study solid state phase transformation problem in binary metallic alloys. The numerical solutions will be compared with analytical solutions and moving data centers method [1]. This paper will show usefulness of radial basis functions for one–dimensional Stefan's problems. The position of the moving boundary will be simulated by level set method.

2 The Meshless Method

The meshless method is currently at the stage of development. Various approaches and computational procedures have been proposed in the literature. Not every method that is claimed to be meshless is really meshless. The true meshless method must provide a computational procedure without relating to any mesh point connectivity.

Three different approaches to develop meshless methods have been successfully proposed. The first one is based on the spirit of the finite element method and employs Petrov–Galerkin weak formulation. Detailed theories and formulation can be found in the book by Atluri and Shen [2].

The second approach is of boundary element type. It attempted to discretize boundary integral formulation without employing a mesh. Grid points in this approach are all on the boundaries. Several procedures [3] have been proposed with different discretization concepts.

The third approach employs radial basis functions (RBFs). The base of this approach is its employment of high-order interpolating functions to approximate solutions of differential equations. All RBFs possess the property that their values are determined only by the distance and have nothing to do with directions. Kansa [4] introduced multiquadric functions to solve hyperbolic, parabolic and elliptic differential equations with collocation methods. One of the most powerful RBF method is based on multiquadric basis functions (MQ), first used by R. L. Hardy [5].

3 Radial Basis Function Methods

Radial basis function methods have been praised for their simplicity and ease of implementation in multivariate scattered data approximation, and they are becoming a viable choice as a method for the numerical solution of partial differential equations. Compared to low–order methods such as finite differences, finite volumes and finite elements, RBF–based methods offer numerous advantages, such as no need for a mesh or triangulation, simple implementation and dimensional independence, and no staircasing or polygonization for boundaries.

A radial basis function is a function $\varphi_j(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_j\|)$, which depends only on the distance between $\mathbf{x} \in \mathbf{R^d}$ and a fixed point $\mathbf{x}_j \in \mathbf{R^d}$. Here, φ is continuous and bounded on any bounded sub-domain $\Omega \subseteq \mathbf{R}^{\mathbf{d}}$. Let r denote by the Euclidean distance between any pair of points in the domain Ω.

The commonly used radial basis functions are:

$\varphi(r) = r,$	linear,
$\varphi(r) = r^3,$	cubic,
$\varphi(r) = r^2 \log r,$	thin-plate spline,
$\varphi(r) = e^{-cr^2},$	Gaussian,
$\varphi(r) = (r^2 + c^2)^{\frac{1}{2}},$	multiquadric,
$\varphi(r) = (r^2 + c^2)^{-\frac{1}{2}},$	inverse multiquadric.

4 The Level Set Method

The level set method first introduced and devised by Osher and Sethian [6]. It is one computational technique for tracking a propagating interface over time, which in many problems has proven to be more accurate in handling complexities in the evolving interface such as entropy conditions and weak solutions. It is a robust scheme that is relatively easy to implement.

4.1 Level Set formulation

In the level set formulation of moving interface, the interfaces, denoted by Γ, are represented implicitly through a level set function $\Phi(\mathbf{x}, t)$, where $\mathbf{x}$ is a position of interface, t is a moment in time. Usually, the

Φ is defined as a signed distance function to the interface. The moving interface is then captured at all time by locating the set of $\Gamma(t)$ for which Φ vanishes. The level set function is advected with time by a transport equation which is known as level set equation:

$$\frac{\partial \Phi}{\partial t} + v_n |\nabla \Phi| = 0, \quad \Phi(\mathbf{x}, 0) = \Phi_0(\mathbf{x}), \qquad (1)$$

where $\Phi_0(\mathbf{x})$ embeds the initial position of the interface and v_n is the normal component of the velocity of the interface:

$$v_n = \mathbf{v} \cdot \frac{\nabla \Phi}{|\nabla \Phi|}, \qquad (2)$$

where $\frac{\nabla \Phi}{|\nabla \Phi|}$ is the unit normal to the surface $\mathbf{N}$.

Taking into account a continuous extension of the interface velocity $\mathbf{v}$, then the evolution of the level set function can be done by the hyperbolic equation for the level set equation:

$$\frac{\partial \Phi}{\partial t} + \nabla \Phi \cdot \mathbf{v} = 0, \quad \Phi(\mathbf{x}, 0) = \Phi_0(\mathbf{x}). \qquad (3)$$

In our case a continuous extension of velocity $\mathbf{v}$ is taken as the (steady) solution of the following evolution equation [7]:

$$\frac{\partial \tilde{v}}{\partial \tau} \pm \frac{\partial \tilde{v}}{\partial \mathbf{x}} \cdot \mathbf{N} = 0, \qquad (4)$$

where τ denotes a fictitious time step not related to the main time step and the sign is determined from the normal direction of the level set function.

In this paper, the RBFs are incorporated into level set methods to construct a more efficient approach. At the initial time, all the time dependent variables should be specified over entire domain. The initial value problem (1) can be considered equivalent to an interpolation problem, and hence the starting point of the use of RBFs to solve partial differential equations is the interpolation problem. Further, the spatial portion is approximated by the RBFs and the temporal variations are approximated by the time dependent expansion coefficients.

5 Solution construction using RBFs

To introduce RBF collocation methods, we consider a PDE in the form of

$$L\,u = f(\mathbf{x}) \quad \text{in} \quad \Omega \subset \mathbf{R^d}, \qquad (5)$$

$$B\,u = g(\mathbf{x}) \quad \text{on} \quad \partial\Omega, \qquad (6)$$

where u is concentration, d is the dimension, $\partial\Omega$ denotes the boundary of the domain Ω, L is the differential operator on the interior, and B is an operator that specifies the boundary conditions of the Dirichlet, Neumann or mixed type. Both, f and g, are given functions mapping $\mathbf{R^d} \to \mathbf{R}$.

Instead of using polynomial construction for a solution, u is approximated by linear combination of RBFs and polynomials:

$$u \approx U(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j \varphi_j(\mathbf{x}) + \sum_{l=1}^{M} \gamma_l p_l(\mathbf{x}), \qquad (7)$$

where φ can be any radial basis function from the list, $p_1, \ldots, p_M \in \Pi_m^d$ are polynomials of degree m or less, $M := \binom{m-1+d}{d}$ [8] and $\|\cdot\|$ indicates the Euclidean norm. Let $\{(\mathbf{x}_j)\}_{j=1}^{N}$ be the $N = N_I + N_B$ collocation points in $\Omega \cup \partial\Omega$. We assume the collocation points are arranged in such a way that the first N_I points are in Ω, whereas the last N_B points are on $\partial\Omega$. To solve for the $N + M$ unknown coefficients, $N + M$ linearly independent equations are needed. By choosing N distinct collocation points $X_I = \{\mathbf{x}_1, \ldots, \mathbf{x}_{N_I}\} \subset \Omega$ and $X_B = \{\mathbf{x}_{N_I+1}, \ldots, \mathbf{x}_N\} \subset \partial\Omega$ and ensuring that $U(\mathbf{x})$ satisfies (5) and (6) at the collocation points results in a good approximation of the solution u. The first N equations are given by

$$\sum_{j=1}^{N} \alpha_j\, L\, \varphi_j(\mathbf{x}_i) + \sum_{l=1}^{M} \gamma_l\, L\, p_l(\mathbf{x}_i) = f(\mathbf{x}_i) \quad \text{for} \quad i = 1, \ldots, N_I \qquad (8)$$

$$\sum_{j=1}^{N} \alpha_j\, B\, \varphi_j(\mathbf{x}_i) + \sum_{l=1}^{M} \gamma_l\, B\, p_l(\mathbf{x}_i) = g(\mathbf{x}_i) \quad \text{for} \quad i = N_I + 1, \ldots, N \qquad (9)$$

The last M equations could be obtained by imposing some extra condition on $p(\cdot)$:

$$\sum_{j=1}^{N} \alpha_j p_k(\mathbf{x}_j) = 0, \quad k = 1, \ldots, M. \qquad (10)$$

This leads to the equivalent matrix form: $\mathbf{A}\mathbf{x} = \mathbf{b}$ or

$$\begin{bmatrix} \mathbf{W_L} & \mathbf{v_L} \\ \mathbf{W_B} & \mathbf{v_B} \\ \mathbf{v^T} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\gamma} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \\ \mathbf{0} \end{bmatrix}, \qquad (11)$$

where

$$\mathbf{W_L} = L\,\varphi_j(\mathbf{x}_i), \quad \mathbf{x}_i \in X_I \qquad (12)$$

$$\mathbf{v_L} = L\,p_l(\mathbf{x}_i), \quad \mathbf{x}_i \in X_I \qquad (13)$$

$$\mathbf{W_B} = B\,\varphi_j(\mathbf{x}_i), \quad \mathbf{x}_i \in X_B \qquad (14)$$

ISBN 978-3-901608-32-2

$$\mathbf{v_B} = B\, p_l(\mathbf{x}_i), \quad \mathbf{x}_i \in X_B \tag{15}$$

$$\mathbf{v} = p_k(\mathbf{x}_j), \quad k = 1, \ldots, M. \tag{16}$$

For the diffusive equation (19) we used implicit scheme:

$$\frac{u^{n+1} - u^n}{\delta t} = \left(D \frac{\partial^2 u^{n+1}}{\partial x^2} \right), \tag{17}$$

where δt is the time step and u^n and u^{n+1} are the concentrations at the time t_n and t_{n+1}. The approximate solution is expressed as :

$$u(\mathbf{x}, t_{n+1}) = \sum_{j=1}^{N} \alpha_j^{n+1} \varphi_j(\mathbf{x}) + \sum_{l=1}^{M} \gamma_l p_l(\mathbf{x}). \tag{18}$$

The choice of basis function is another flexible features of RBF methods. RBFs can be globally supported, infinitely differentiable, and contain a free parameter, c, called the *shape parameter*. This leads to a full coefficient matrix.

The shape parameter affects both the accuracy of the approximation and the conditioning of the interpolation matrix. In general, for a fixed number of centers N, smaller shape parameters produce the more accurate approximations, but also are associated with a poorly conditioned matrix. The condition number also grows with N for fixed values of the shape parameter c. In practice, the shape parameter must be adjusted with the number of centers in order to produce an interpolation matrix which is well conditioned in finite precision arithemetic. Many researchers (e.g. [9]–[10]) have attempted to develop algorithms for selecting optimal values of the shape parameter c. The optimal shape parameter c is still an open question.

In our case we used an iterative mode by monitoring the spatial distribution of the residual errors in Ω and $\partial\Omega$ as a function of c. The iterations are terminated when errors are smaller then a specified bound. This map is then used to guide the search of the optimal shape parameter c that the best approximate the solution. In our study we usually used multiquadric (MQ) RBF. The generalized form of the MQ basis function is $\phi_j(\mathbf{x}) = [(\mathbf{x} - \mathbf{x}_i)^2 + c_i^2]^\beta$, where $\mathbf{x}$, $\mathbf{x}_i \in \mathbf{R^d}$, and β is a non integer $\geq -1/2$.

6 The problem

6.1 The physical model

In this study we consider classical Stefan's problem: the solid state phase transformation problem in binary metallic alloys which is described in [11]. In that problem a volume of constant composition is surrounded by a diffusive phase. In the interface between the particle and the diffusive phase a constant concentration is assumed, and the gradient of the concentration causes the movement of the interface. This problem is also called solid–solid transformation.

6.2 The mathematical model

We consider the domain Ω containing a diffusive phase Ω_{dp} and the part where the material characteristic Ω_{mat} remain of constant composition u^{mat}. The particle dissolves due to Fickian diffusion in the diffusive phase. The concentration at the interface Γ, separating Ω_{mat} and Ω_{dp}, is assumed to be given by the constant value u^{sol}. The concentration gradient on the side of Ω_{dp} at Γ causes its displacement. The governing equations and boundary conditions of this problem are:

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) = D \Delta u(\mathbf{x}, t), \quad \mathbf{x} \in \Omega_{dp}(t),\ t > 0, \tag{19}$$

$$u(\mathbf{x}, t) = u^{mat},\ \mathbf{x} \in \Omega_{mat}(t),\ t \geq 0, \tag{20}$$

$$u(\mathbf{x}, t) = u^{sol}, \mathbf{x} \in \Gamma(t),\ t \geq 0, \tag{21}$$

$$(u^{mat} - u^{sol}) v_n(\mathbf{x}, t) = D \frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma(t),\ t > 0, \tag{22}$$

where $\mathbf{x}$ is the coordinate vector of a point in Ω, D means the diffusivity constant, $\mathbf{n}$ is the unit normal vector on the interface pointing outward with respect to $\Omega_{mat}(t)$ and v_n is the normal component of the velocity of the interface. The initial concentration $u(\mathbf{x}, 0)$ inside the diffusive phase is given. We assume no flux through the boundary:

$$\frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega_{dp}(t) \backslash \Gamma(t),\ t > 0, \tag{23}$$

hence mass is conserved.

6.3 The numerical solution methods

In our model the motion of the interface is determined by the gradient of concentration, which can be computed from the solution of the diffusion equation. Here we present an interpolative moving data center method, in which the data centers are computed for each time step and the solution is interpolated from the old data centers to the new. The equations are solved with collocation methods using MQ RBF. The position of the points depend on time. An outline of the algorithm is:

- Compute the concentrations profiles solving equations (19), (20), (21) and (23);
- Predict the position of boundary s_1 at the new time–step: $s_1(t + \Delta t)$ using boundary condition (22);
- Once the boundary is moved, the concentration u can be computed in the new region using Eq. (19). The solution is interpolated from the old data center to the new.

6.4 Analytical solutions

In numerical experiments we will compare our numerical solutions with the analytical solutions that exist for the problem presented above. These solutions are expressed as functions of $\frac{x-s_0}{\sqrt{t}}$ as proved in [12], and the domain $\Omega = [0, l]$ has to be infinite or semi–infinite. The interface position is given by $s(t) = s_0 + 2\alpha\sqrt{t}$, where the constant α is obtained by solving the following equation:

$$\alpha = \frac{u^0 - u^{sol}}{u^{mat} - u^{sol}} \sqrt{\frac{D}{\pi}} \frac{\exp(-\frac{\alpha^2}{D})}{\operatorname{erfc}(\frac{\alpha}{\sqrt{D}})}. \qquad (24)$$

When α is known, the concentration is given by

$$u(x,t) = \begin{cases} u^{mat} & \text{if } x < s(t), \\ u^0 + \dfrac{(u^{sol} - u^0)\operatorname{erfc}(\frac{x-s_0}{2\sqrt{Dt}})}{\operatorname{erfc}(\frac{\alpha}{\sqrt{D}})}, & \text{if } x \geq s(t), \end{cases}$$

where u^{mat} is the concentration inside the material and u^0 is the initial concentration of the diffusive phase. s_0 is the initial position of the interface.

We assume a piecewise initial concentrations as follows:

$$u(x,0) = \begin{cases} u^{mat} & \text{if } x \in \Omega_{mat} = [0, s_0), \\ u^{sol} & \text{if } x = s_0, \\ u^0 & \text{if } x \in \Omega_{db} = (s_0, l). \end{cases}$$

7 Level Set Method Construction with RBFs

The preliminary starting point of the use of RBFs to solve PDEs is the interpolation because the advection of the level set function by a transport equation is equivalent to solving the initial value problem.

7.1 Interpolation of the level set function

In the present construction of the implicit level set function, the MQ RBFs is used to interpolate the scalar implicit level set functions $\Phi(\mathbf{x})$ with N points by using N MQs centered at these points. The resulting RBF interpolant of the implicit function can be written as

$$\Phi(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j \varphi_j(\mathbf{x}) + p(\cdot), \qquad (25)$$

where α_j is the weight of the radial basis function positioned at the jth center.

For the two-dimensional modeling problem, $p(\cdot)$ can be given by

$$p(\cdot) = p_0 + p_1 x + p_2 y. \qquad (26)$$

Due to ensure a unique solution, the RBF interpolant of $\Phi(\mathbf{x})$ must be subject to the following constraints

$$\sum_{j=1}^{N} \alpha_j = 0, \quad \sum_{j=1}^{N} \alpha_j x_j = 0, \quad \sum_{j=1}^{N} \alpha_j y_j = 0. \qquad (27)$$

Knowing the interpolation data values $f_1, \ldots, f_N \in I\!R$ at the centers $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \Omega \subset I\!R$, the RBF interpolant (25) can be obtained by solving the system of $N+3$ linear equations for $N+3$ unknown coefficients:

$$\Phi(\mathbf{x}_i) = f_i \quad i = 1, \ldots, N. \qquad (28)$$

Considering the constraints (27), the equation (28) can be re-written in matrix form as

$$\mathbf{H}\boldsymbol{\alpha} = \mathbf{f}, \qquad (29)$$

where

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P^T} & \mathbf{0} \end{bmatrix} \in I\!R^{(N+3)\times(N+3)}, \qquad (30)$$

$$\mathbf{A} = \begin{bmatrix} \varphi_1(\mathbf{x}_1) & \ldots & \varphi_N(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(\mathbf{x}_N) & \ldots & \varphi_N(\mathbf{x}_N) \end{bmatrix} \in I\!R^{N\times N}, \qquad (31)$$

$$\mathbf{P} = \begin{bmatrix} 1 & x_1 & y_1 \\ \vdots & \vdots & \vdots \\ 1 & x_N & y_N \end{bmatrix} \in I\!R^{N\times 3}, \qquad (32)$$

$$\boldsymbol{\alpha} = [\alpha_1 \ \ldots \ \alpha_N \ p_0 \ p_1 \ p_2]^T \in I\!R^{N+3}, \qquad (33)$$

$$\mathbf{f} = [f_1 \ \ldots \ f_N \ 0 \ 0 \ 0]^T \in I\!R^{N+3}. \qquad (34)$$

The generalized expansion coefficients can be obtained by

$$\boldsymbol{\alpha} = \mathbf{H}^{-1}\mathbf{f}. \qquad (35)$$

The resulting RBF interpolant of the implicit function can be re-written compactly as

$$\Phi(\mathbf{x}) = \boldsymbol{\phi}^T(\mathbf{x})\boldsymbol{\alpha}, \qquad (36)$$

where

$$\boldsymbol{\phi} = [\varphi_1(\mathbf{x}) \ \ldots \ \varphi_N(\mathbf{x}) \ 1 \ x \ y]^T \in I\!R^{(N+3)\times 1}. \qquad (37)$$

7.2 Level set equation

Since the Hamilton-Jacobi PDE (1) is time dependent, it is further assumed that all knots are fixed in space and the space and time are separable, and therefore the RBF interpolant of the implicit function in equation (36) becomes time dependent as

$$\Phi(\mathbf{x}) = \boldsymbol{\phi}^T(\mathbf{x})\boldsymbol{\alpha}(t). \tag{38}$$

Substituting equation (38) in (1) yields

$$\boldsymbol{\phi}^T \frac{d\boldsymbol{\alpha}}{dt} + v_n|(\nabla\boldsymbol{\phi})^T\boldsymbol{\alpha}| = 0, \tag{39}$$

where

$$|(\nabla\boldsymbol{\phi})^T\boldsymbol{\alpha}| = \left[\left(\frac{\partial\boldsymbol{\phi}^T}{\partial x}\boldsymbol{\alpha}\right)^2 + \left(\frac{\partial\boldsymbol{\phi}^T}{\partial y}\boldsymbol{\alpha}\right)^2\right]^{1/2}, \tag{40}$$

$$\frac{\partial\boldsymbol{\phi}}{\partial x} = \left[\frac{\partial\varphi_1}{\partial x} \ \cdots \ \frac{\partial\varphi_N}{\partial x} \ 0 \ 1 \ 0\right]^T \in I\!R^{(N+3)\times 1}, \tag{41}$$

$$\frac{\partial\boldsymbol{\phi}}{\partial y} = \left[\frac{\partial\varphi_1}{\partial y} \ \cdots \ \frac{\partial\varphi_N}{\partial y} \ 0 \ 0 \ 1\right]^T \in I\!R^{(N+3)\times 1}. \tag{42}$$

The initial value problem can be considered equivalent to the interpolation problem since the expansion coefficients at the initial time are found as a solution of the interpolation problem [13]. Therefore the preliminary starting point of the use of RBFs to solve PDEs is the interpolation problem that is equivalent to solving the initial value problem. The original equation (1) is thus converted into a time-dependent interpolation problem for the initial values of expansion coefficients and the propagation of the front is governed by the time dependent equation (39).

For time advance the initial values of $\boldsymbol{\alpha}$ in equation (39) we used a collocation formulation of the method of lines. The governing equation of motion of the front (39) is extended to the whole domain and the normal velocities v_n at the front are thus replaced by the extension velocities v_n^e in Ω. All nodes of domain are taken as fixed nodes of RBF interpolation. We also take into consideration constraints which must be introduced to guarantee that the generalized coefficients $\boldsymbol{\alpha}$ can be solved.

Using the present collocation method for N points and above mentioned constraints [13], a set of resulting ODEs can be compactly written as:

$$\mathbf{H}\frac{d\boldsymbol{\alpha}}{dt} + \mathbf{B}(\boldsymbol{\alpha}) = 0, \tag{43}$$

where

$$\mathbf{B}(\boldsymbol{\alpha}) = \begin{bmatrix} v_n^e(\mathbf{x}_1)|(\nabla\boldsymbol{\phi})^T(\mathbf{x}))\boldsymbol{\alpha}| \\ \vdots \\ v_n^e(\mathbf{x}_N)|(\nabla\boldsymbol{\phi})^T(\mathbf{x}))\boldsymbol{\alpha}| \\ 0 \\ 0 \\ 0 \end{bmatrix} \in I\!R^{(N+3)\times 1}. \tag{44}$$

The set of ODEs can be solved by several ODE solvers such as the first-order forward Euler's method and higher-order Runge-Kutta, Runge-Kutta-Fehlberg, Adams-Bashforth, or Adams-Moulten methods [14].

It was used the first-order forward Euler's method, an approximate solution to equation (43) is the following:

$$\boldsymbol{\alpha}(t^{n+1}) = \boldsymbol{\alpha}(t^n) - dt\mathbf{H}^{-1}\mathbf{B}(t^n) \tag{45}$$

where dt is the time step, and exact explicit time integration. Equation (43) has the following form:

$$\frac{d\boldsymbol{\alpha}}{dt} + \mathbf{E}\boldsymbol{\alpha} = 0, \tag{46}$$

where $\mathbf{E} = \mathbf{H}^{-1}(v_1\boldsymbol{\phi}_{,1} + v_2\boldsymbol{\phi}_{,2})$. The solution has the following form:

$$\boldsymbol{\alpha}(t+dt) = \mathbf{expm}(-\mathbf{E}dt)\boldsymbol{\alpha}(t-dt), \tag{47}$$

where $\mathbf{expm}$ is a MATLAB exponential matrix function represents the series expansion or a rational fraction:

$$\mathbf{expm}(-\mathbf{E}dt) = \mathbf{I} - \mathbf{E}dt + (dt^2/2!)\mathbf{E}^2 \mp \cdots. \tag{48}$$

8 Numerical Example

For the simulations we used data from [11]: the concentration inside the part where the material characteristics remain constant $u^{mat} = 0.53$, the concentration on the interface $u^{sol} = 0$, the initial concentration of the diffusive phase $u^0 = 0.1$, the diffusivity constant $D = 1$, the domain length $l = 1$ and the initial position of the interface $s_0 = 0.2$.

Let N be the total number of grid intervals, r of those lie inside constant composition and $N - r$ lie inside the diffusive phase. The grid is uniform in each phase and the interface is always located in the r^{th} node. Due to the movement of the interface, the grid is adapted at each time step.

In numerical experiments it was also included MQ exponent, β as additional parameter to be optimized. The MQ exponent, β had values 0.5 and 1.5. In fig. 1 the movement of the interface positions calculated with

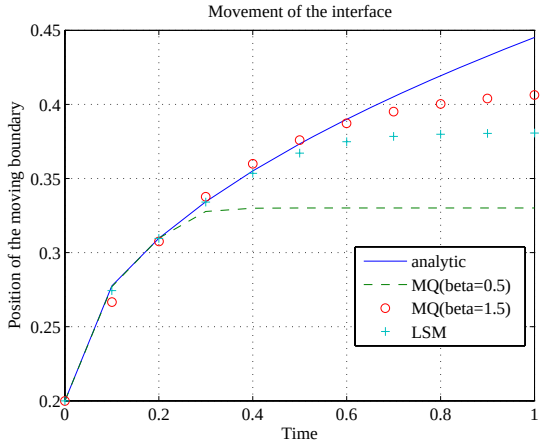

Fig. 1 Interface position vs. time

moving data centers method and level set method is presented.

The future and present studies are already headed for using RBFs in level set method for two dimensional problems. Therefore we would like present a few examples of preliminaries results. We begin with a level set evolution of the form

$$\frac{\partial \Phi}{\partial t} + |\nabla \Phi| = 0. \tag{49}$$

Using force in normal direction is presented in fig. 2, actually evolution of circle in normal direction.

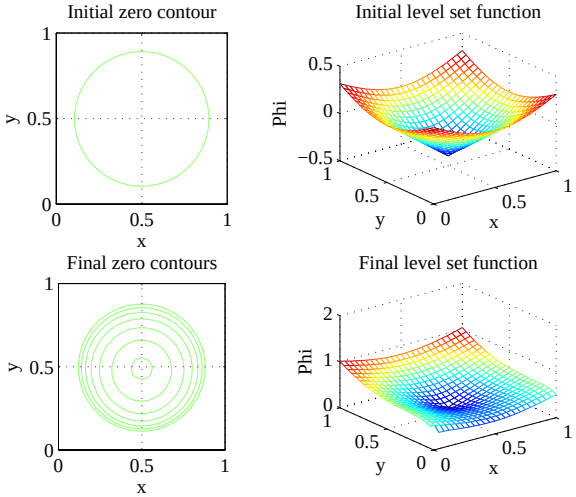

Fig. 2 Shrink in normal direction

We continue with a level set evolution of the form

$$\frac{\partial \Phi}{\partial t} - |\nabla \Phi| = 0. \tag{50}$$

The inversely situation is presented in fig. 3.

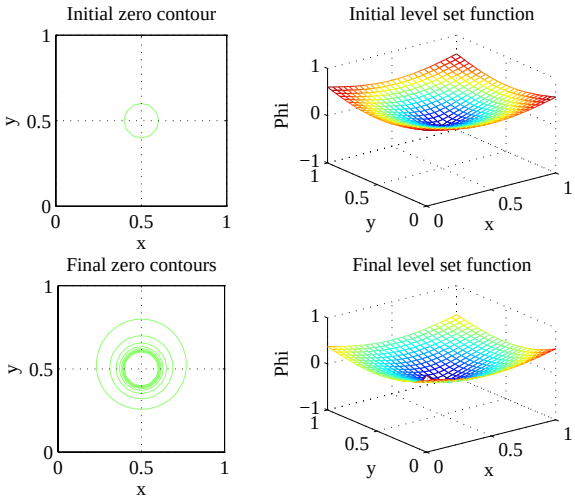

Fig. 3 Expand in normal direction

The next example is presentation of transport equation (3). We will present the rotation of the solid body. Consider the rotation of a ellipse centered at (0.0,-0.45) in a vortex flow with velocity field $(v_1, v_2) = (-y, x)$. A half cycle of rotation of zero contour of the level set function at different points and time during the rotation of a ellipse is presented in fig. 4.

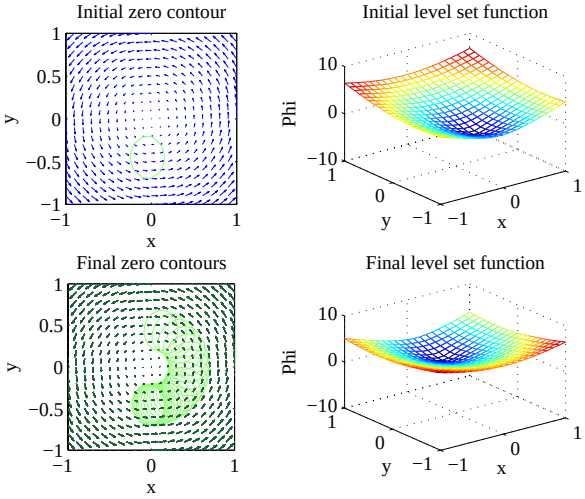

Fig. 4 Rotation of a ellipse

9 Discussion and Conclusion

Comparison of positions of the moving boundary calculated with moving data centers method and MQ ($\beta = 0.5$) and MQ ($\beta = 1.5$) (Fig. 1) shows that MQ ($\beta = 1.5$) determines the position of the interfaces much more accurately than MQ ($\beta = 0.5$). The simulations have also shown that the value of the shape parameter c which was computed by residual error procedure was in the range between 0.01 and 0.09. This confirms the fact that for a fixed number of centers N, smaller

shape parameters produce more accurate approximations. The results have shown that β should be greater than 0.5 if we want to get reasonable results.

From the results of the last chapter we can also see that RBFs can be easily included in 2D level set formulation. Figures 2, 3 and 4 show that we can get with using MQ in 2 dimensional examples logical results. In our future work we will include RBFs in three dimensional level set formulations.

Comparison of positions of the moving boundary calculated with moving data centers method $MQ(\beta = 1.5)$ and the level set method (Fig. 1) also shows that moving data centers method gives in this case better results.

It is important to mentioned that, the timestep size should be small to achieve the numerical stability due to the Courant–Friedrichs–Lewy (CFL) condition for the stability. A small timestep size together with a large number of RBF ponints can be used to achieve an accurate solution to the original Hamilton–Jacobi PDE, but the computational time will be increased significantly and the computational efficiency may pose a severe problem. In our case we have improved the efficiency like this that we constraint on the timestep size and total time number of RBF ponits.

To achieve better accuracy, the resultant system of RBF-PDE problem usually becomes badly conditioned. Several different strategies [15] have been somewhat successful in reducing the ill-conditioning problem when using RBF methods in PDE problems. The strategies include: variable shape parameters, domain decomposition, preconditioning of the interpolation matrix, and optimizing the canter locations.

We conclude that the Kansa method is a valid alternative to the analytic solutions. It has simpler implementation and we can easily use in the level set formulations. The only geometric properties that are used in RBF approximation are the pair-wise distances between points.

In the future work we will use the Gershgorin circle theorem that could be useful tool for choosing appropriate RBFs. For each value of shape parameter, eigenvalues and their distribution can be studied, therefore obtaining knowledge concerning properties of an approximation matrix and their role being played in finding better approximation of computed data to solution of equation.

Gerschgorin theorem simplifies and optimizes the calculation of eingevalus of a matrix, which can, otherwise, be quit time consuming work. The very task is limited to summation of elements in a particular row, which gives the length of an interval on which the appropriated eingenvalue lies; position of that interval is determined with its centre point and calculated from the diagonal element in row of an approximation matrix.

Sketch of the proof. The idea of looking of optimal shape parameter is actually equal to the problem of finding extreme. Since the function used in iterative algorithm as defined, satisfies a Lipschitz condition it contains a fixed point. That means the sequence of matrices, converges to the scalar matrix. Therefore, as matrix function depends on the spectra of the matrix. Such a function is used in algorithm then it is logical that shape parameter equals to an element of diagonal. If Gershgorin intervals are narrowed then eigenvalues of the matrix are also flowing together. Therefore, in the matrix sequence expansion the eigenvalues of matrix is also flowing together to the diagonal element or optimal shape parameter. So, if the intervals narrow then the value of the elements off the diagonal tends to the 0. In the limit we can get scalar matrix which eigenvalue is equal to the shape parameter.

10 References

[1] L. Vrankar, E. J. Kansa, G. Turk and F. Runovc. Solving One-Dimensional Moving-Boundary Problems with Meshless Method. *ECMI 2006-14th European Conf. on Mathematics for Industry* (in press).

[2] S. N. Atluri, and S. Shen. *The Meshless Local Petrov–Galerkin Method*. Tech Science Press, Encino, CA, 2002.

[3] G. Li, and S. N. Atluri. Boundary cloud method: a combined scattered point/boundary integral approach for boundary–only analysis. *Computer. Methods Appl. Engrg.*, 191:2337–2370, 2002.

[4] E. J. Kansa. Multiquadrics–A scattered data approximation scheme with applications to computational fluid dynamics-I. Surface approximations and partial derivative estimates. *Computers Math. Applic.*, 19, 8–9:127–145, 1990.

[5] R. L. Hardy. Multiquadric equation of topography and other irregular surfaces. *J. Geophys. Res.*, 176:1905–1915, 1971.

[6] S. Osher, J. A. Sethian. Front propagating with curvature dependent speed: algorithms base on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 78:12–49, 1988.

[7] Z. Li, and K. Ito. *The immersed interface method–Numerical solutions of PDEs involving interfaces and irregular domains*. Society for industrial and Applied Mathematics, 3600 University City Science Center, Philadelphia, USA, 2006.

[8] A. Iske. *Charakterisierung bedingt positiv definiter Funktionen für multivariate Interpolationsmethoden mit radialen Basisfunktionen*. Thesis, University Göttingen, 1994.

[9] R. E. Carlson and T. A. Foley. The parameter r^2 in multiqudric interpolation, *Comput. Math. Appl.*, 21(9):29–42, 1991.

[10] S. Rippa. An algorithm for selecting a good parameter c in radial basis function interpolation, *Advances in Computational Mathematics.*, 11:193–210, 1999.

[11] E. Javierre, C. Vuik, F. J. Vermolen and S. van der Zwaag. A comparison of numerical models for one–dimensional Stefan problems, *Reports of the Delft Institute of Applied Mathematics*, Netherlands, 2005.

[12] J. M. Hill. *One–dimensional Stefan problems: an introduction*, Longman Scientific and Technical, Harlow, 1987.

[13] S. Y. Wang, K. M. Lim, B. C. Khoo and M. Y. Wang. An extended level set method for shape and topology optimization. *Journal of Computational Physics*, 221:395-421, 2007.

[14] M. D. Greenberg. *Advanced Engineering Mathematics*, second ed., Prentice-Hall International Inc., Upper Saddle River, NJ, USA, 1988.

[15] E. J. Kansa and Y. C. Hon. Circumventing the Ill-Conditioning Problem with Multiquadric Radial Basis Functions: Applications to Elliptic Partial Differential Equations, *Computers Math. Applic*, 39, 7–8:123-137, 2000.

ISBN 978-3-901608-32-2