Proceedings of the International Conference on
Optimization using Exergy-Based Methods
and Computational Fluid Dynamics
Berlin, Germany, October 20-23, 2009

MESHLESS SOLUTIONS OF TEMPERATURE FIELDS FOR USE IN DENDRITIC GROWTH SIMULATIONS

Guangming Yao[1], C.S. Chen[2], Marina Jelen[3], Bozidar Šarler[4]*
[1] IMPOL Aluminium Industry, Slovenska Bistrica, Slovenia, guangmingyao@gmail.com
[2] Department of Mathematics, University of Southern Mississippi, USA, cschen.math@gmail.com
[3] IMPOL Aluminium Industry, Slovenska Bistrica, Slovenia, marina.jelen@impol.si
[4] Laboratory for Multiphase Processes, University of Nova Gorica, Slovenia, bozidar.Šarler@ung.si

Abstract

The main problem in cellular automata modeling of microstructure formulation is the dependence of the crystal shape on the orientation of the principal growth axis. This problem has been recently solved by using point automata concept [1, 2] instead of the cellular automata concept and the use of the random mesh. The temperature field in the random nodes has been recently calculated using the finite difference method, and the random mesh solution has been picked out in the randomly chosen nodes of the regular mesh. In this paper we investigate the performance of two meshless methods that can be directly used on randomly distributed node arrangements. The tools are based on the developments in [3, 4].

Keywords : *local radial basis function, diffusion equation, meshless method.*

1. Introduction

Consider a connected and bounded domain Ω with boundary Γ occupied by a substance, governed by the heat diffusion equation

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) \tag{1}$$

where ρ, c, k, T, t stand for density, specific heat, thermal conductivity, temperature, and time. We seek the solution of the above equation provided that the initial value of the temperature and the Dirichlet boundary conditions are known. During the last two decades the meshless methods have been developed and applied to solve engineering and material science problems [5-8]. More recently, various localized meshless methods have been developed [3-4, 9-11]. In this paper, two local meshless methods are compared for solving heat diffusion equations. These meshless methods will later be applied to dendritic growth model. The first method, the explicit local RBF collocation method was first introduced for solving time dependent problems in 2006 by Šarler et al in [3], the numerical results are accurate and efficient for small time steps used. The second method, the local multiquadric approximation was developed for solving Poisson equation in 2003 by Lee et al in [4] which can be extend to the modified Helmholtz equations. We compare these two local meshless methods for the Dirichlet jump problem with the analytical solution as a function of the node distribution (uniform, non-uniform), time step length and multiquadrics free parameters.

2. Governing equations

For simplicity, we assess the performance of the meshless methods by simple dimensionless diffusion equation

$$\frac{\partial T}{\partial t} = \nabla^2 T \tag{2}$$

defined on the domain Ω with the boundary Γ. The initial data at time $t = t_0$ are given by

$$T(\mathbf{p}, t_0) = T_0, \mathbf{p} \in \Omega \cup \Gamma. \tag{3}$$

The Dirichlet boundary condition is given by

$$T(\mathbf{p}, t) = T_D, t \geq t_0, \mathbf{p} \in \Gamma \tag{4}$$

Time stepping method is one of the popular methods for obtaining numerical solutions of time-dependent partial differential equations. Let Δt be the time step length and $t_n = n\Delta t, n \geq 0$ the time discretisation. For $t \in (t_{n-1}, t_n]$, let $\xi \in [0,1]$ and

$$\frac{\partial T}{\partial t} = \frac{T - T_0}{\Delta t},\ \nabla^2 T = \xi \nabla^2 T + (1-\xi)\nabla^2 T_0, \tag{5}$$

Then (2) can be written as

$$\frac{T - T_0}{\Delta t} = \xi \nabla^2 T + (1-\xi)\nabla^2 T_0. \tag{6}$$

* Corresponding author

By simplifying the above equation, we obtain

$$(1-\Delta t\xi\nabla^2)T=\left[1+\Delta t\left(1-\xi\right)\nabla^2\right]T_0. \qquad (7)$$

There are two special cases,

$$\xi=\begin{cases}0, & \text{Explicit Method,}\\ 1, & \text{Implicit Method.}\end{cases}$$

If $\xi=0,$ then (7) can be written as

$$T=T_0+\Delta t\nabla^2 T_0. \qquad (8)$$

This is called explicit time stepping method that calculates the temperatures at a later time using the temperatures at the current time.

If $\xi=1,$ (7) can be rearranged as following

$$(1-\Delta t\nabla^2)T=T_0 \qquad (9)$$

which is the case of implicit time stepping method that finds a solution by solving an equation involving both the current state of the system and the later one.

3. The two localized meshless methods

In this paper we investigate the performance of the two meshless methods that can be directly implemented on randomly distributed nodes: the explicit local RBF collocation method (ELRBFCM) and the implicit local RBF collocation method (ILRBFCM). The multiquadrics radial basis functions have been used in the interpolation in both approaches.

3.1 ELRBFCM

The local RBF collocation method was applied to explicit time stepping model in 2006 by Šarler et al [3]. The points are denoted by $\mathbf{p}_k; k=1,2,...,N$. The region $\Omega\cup\Gamma$ is divided into N overlapping subdomains ${}_{l}\Omega;\ l=1,2,...,N$. The schematics of node distribution with typical influence domains are shown in Fig.1. Each of the subdomains consists of ${}_{l}N$ points ${}_{l}\mathbf{p}_n;$ $n=1,2,...,{}_{l}N$ that coincide with some of the global points $\mathbf{p}_k; k=1,2,...,N$. There is a relation between the global and the local point with indexes on each of the subdomain. This relation is $k=k(l,n)$. The $k(l,n)$ is a function of the local subdomain index l and local index n . The following is valid

$$\mathbf{p}_{k(l,n)}={}_{l}\mathbf{p}_n; l=1,2,...,N, n=1,2,...,{}_{l}N. \qquad (10)$$

The temperature is represented on each of the subdomains by ${}_{l}N$ RBF's ${}_{l}\psi_n(\mathbf{p}), n=1,2,...,N$, and their coefficients ${}_{l}\alpha_n; n=1,2,...,{}_{l}N$, i.e.

$$T_0\left(\mathbf{p}\right)=\sum_{n=1}^{{}_{l}N}\psi_{k(l,n)}\left(\mathbf{p}\right){}_{l}\alpha_n;\ \mathbf{p}\in{}_{l}\Omega. \qquad (11)$$

The operation of the spatial partial differential operator Ξ (which can stand for example for ∇^2 , ∇ , or $\partial/\partial\mathbf{n}_\Gamma$) on temperature can be expressed as

$$\Xi T_0\left(\mathbf{p}\right)=\sum_{n=1}^{{}_{l}N}\left[\Xi\psi_{k(l,n)}\left(\mathbf{p}\right)\right]{}_{l}\alpha_n;\ \mathbf{p}\in{}_{l}\Omega. \qquad (12)$$

The coefficients ${}_{l}\alpha_n$ are determined by collocation

$$T_0\left(\mathbf{p}_{k(l,m)}\right)=\sum_{n=1}^{{}_{l}N}{}_{l}\psi_{k(l,n)}\left(\mathbf{p}_{k(l,m)}\right){}_{l}\alpha_n, \qquad (13)$$

for $\mathbf{p}_{k(l,m)}\in{}_{l}\Omega;\ m=1,2,...,{}_{l}N$. It can be written as

$$T_{0k(l,m)}=\sum_{n=1}^{{}_{l}N}{}_{l}\Psi_{mn}\,{}_{l}\alpha_n;\ m=1,2,...,{}_{l}N \qquad (14)$$

with the matrix element ${}_{l}\Psi_{mn}$ of the matrix ${}_{l}\boldsymbol{\Psi}$ defined as

$${}_{l}\Psi_{mn}={}_{l}\psi_{k(l,n)}\left(\mathbf{p}_{k(l,m)}\right). \qquad (15)$$

We determine the coefficients ${}_{l}\alpha_n$ by inverting the matrix ${}_{l}\boldsymbol{\Psi}$ (i,e. ${}_{l}\boldsymbol{\Psi}^{-1}\,{}_{l}\boldsymbol{\Psi}={}_{l}\mathbf{I}$)

$${}_{l}\alpha_n=\sum_{m=1}^{{}_{l}N}{}_{l}\Psi^{-1}_{nm}T_{k(l,m)} \qquad (16)$$

which implies that for $\mathbf{p}\in{}_{l}\Omega$

$$T_0\left(\mathbf{p}\right)=\sum_{n=1}^{{}_{l}N}\sum_{m=1}^{{}_{l}N}\psi_{k(l,n)}\left(\mathbf{p}\right){}_{l}\Psi^{-1}_{nm}T_{0k(l,m)}. \qquad (17)$$

The operation of the spatial partial differential operator Ξ on temperature can be expressed as

$$\Xi T_0\left(\mathbf{p}\right)=\sum_{n=1}^{{}_{l}N}\sum_{m=1}^{{}_{l}N}\Xi\psi_{k(l,n)}\left(\mathbf{p}\right){}_{l}\Psi^{-1}_{nm}T_{0k(l,m)}, \qquad (18)$$

for $\mathbf{p}\in{}_{l}\Omega$. Let $\Xi=\nabla^2$, the operation of the Laplacian on temperature at initial time at the global point $\mathbf{p}$, $\nabla^2 T_0(\mathbf{p})$, is obtained. As a result, every quantity in (8) is known except the temperature at point $\mathbf{p}_k$, $k=1,2,\cdots N$ at $t=t_0+\Delta t,$ this $T(\mathbf{p}_k)$ can be calculated by

$$T_0(\mathbf{p}_k)+\Delta t\sum_{n=1}^{{}_{l}N}\sum_{m=1}^{{}_{l}N}\nabla^2\psi_{k(l,n)}\left(\mathbf{p}_k\right){}_{l}\Psi^{-1}_{nm}T_{0k(l,m)}. \qquad (19)$$

The new values are calculated step by step. Instead of global approach, the collocation is made locally over a set of overlapping domains of influence and the time stepping is performed explicitly. Only small system of linear equations with the dimension of the number of nodes included in the domain of influence have to be solved for each node in this method.

3.2 ILRBFCM

In the implicit method, we proceed as follows: instead of calculating the temperature at current time step point by point, the temperatures at all given nodes are calculating in one step using global sparse system. The operation of the spatial partial differential operator on temperature at final time step is the issue we would like to investigate, but not at the initial time step. Thus the RBF representation of temperature on each of the subdomains becomes

$$T\left(\mathbf{p}_k\right)=\sum_{n=1}^{{}_{l}N}\sum_{m=1}^{{}_{l}N}{}_{l}\psi_{k(l,n)}\left(\mathbf{p}_k\right){}_{l}\Psi^{-1}_{nm}T_{k(l,m)}. \qquad (20)$$

For $\mathbf{p}_k\in{}_{l}\Omega,$ the operation of the spatial partial differential operator Ξ on temperature can be written as

$$\Xi T\left(\mathbf{p}_k\right)=\sum_{n=1}^{{}_lN}\sum_{m=1}^{{}_lN}\Xi\psi_{k(l,n)}\left(\mathbf{p}_k\right){}_l\Psi_{nm}^{-1}T_{k(l,m)}. \tag{21}$$

Using (9) and the boundary conditions we can write the following system of equations for each point $\mathbf{p}_l; l=1,2,...,N$, $k,l=1,2,...,N$

$$\Upsilon_{\Omega l}\sum_{n=1}^{{}_lN}\sum_{m=1}^{{}_lN}\left[1-\Delta t\nabla^2\,{}_l\psi_{ln}\right]{}_l\Psi_{nm}^{-1}T_{k(l,m)}$$

$$+\Upsilon_{\Gamma l}^{D}\sum_{n=1}^{{}_lN}\sum_{m=1}^{{}_lN}{}_l\psi_{ln}\,{}_l\Psi_{nm}^{-1}T_{k(l,m)}=\Upsilon_{\Omega l}T_0^{\Omega l}+\Upsilon_{\Gamma l}^{D}T_D^{\Gamma l}; \tag{22}$$

where

$$\Upsilon_{\Omega}\left(\mathbf{p}\right)=\begin{cases}0; \mathbf{p}\notin\Omega,\\ 1; \mathbf{p}\in\Omega,\end{cases}\quad \Upsilon_{\Gamma}^{D}\left(\mathbf{p}\right)=\begin{cases}0; \mathbf{p}\notin\Gamma^{D},\\ 1; \mathbf{p}\in\Gamma^{D}.\end{cases} \tag{23}$$

The sparse system of (22) can be written in a matrix form

$$\sum_{i=1}^{N}\Psi_{li}T_i=T_l;\ l=1,2,...,N \tag{24}$$

where

$$\Psi_{li}=\Upsilon_{\Omega l}\sum_{n=1}^{{}_lN}\sum_{m=1}^{{}_lN}\left[1-\Delta t\nabla^2\,{}_l\psi_{ln}\right]{}_l\Psi_{nm}^{-1}\delta_{k(l,m)i}$$

$$+\Upsilon_{\Gamma l}^{D}\sum_{n=1}^{{}_lN}\sum_{m=1}^{{}_lN}{}_l\psi_{ln}\,{}_l\Psi_{nm}^{-1}\delta_{k(l,m)i}, \tag{25}$$

$$T_l=\Upsilon_{\Omega l}T_0^{\Omega l}+\Upsilon_{\Gamma l}^{D}T_D^{\Gamma l}, \tag{26}$$

and Kronecker delta

$$\delta_{ki}=\begin{cases}1; & k=i,\\ 0; & k\neq i.\end{cases} \tag{27}$$

For the local RBF collocation method, the only geometrical data needed is the local configuration of nodes that fall within its influence domain. The temperatures at a later time step at all given nodes are obtained in one step by inverting a sparse system. The new values are simultaneously calculated. This is the main difference to the ELRBFCM.

3.2 Discussion of both methods

It is clear that implicit methods require solving an equation system, and they are more difficult to numerically implement. Implicit methods are used because many problems arising in real life are stiff. The use of an explicit method for such cases requires impractically small time steps Δt to obtain the stable result. For such problems, to achieve the given accuracy, it takes much less computational time using an implicit method with larger time steps, even taking into account that one needs to solve an equation of the form (9) at each time step. Whether one should use an explicit or implicit method depends upon the problem to be solved at hand.

4. Numerical results

Through this section, we numerically compare two localized meshless methods which are introduced in the last section. The similar multiquadrics scaling technique as in [11] is introduced to alleviate the difficulty of choosing shape parameter in multiquadrics. The normalization of the influence domain is performed by scaling the distance in both x and y directions, the non-scaled and scaled radial two dimension RBF ψ with corresponding $\nabla^2\psi$ are listed below. We use scaled RBF in all numerical results of the present paper.

1. Non-scaled RBF

$$\psi(x,y)=\sqrt{(x-x_c)^2+(y-y_c)^2+c^2} \tag{28}$$

$$\nabla^2\psi(x,y)=\frac{(x-x_c)^2+(y-y_c)^2+2c^2}{((x-x_c)^2+(y-y_c)^2+c^2)^{1.5}} \tag{29}$$

where (x_c,y_c) is the center of RBF ψ .

2. Scaled RBF

$$\psi(x,y)=\sqrt{x'^2+y'^2+c^2} \tag{30}$$

$$\nabla^2\psi(x,y)=\left(\frac{x'^2+c^2}{y_{\max}^2}+\frac{y'^2+c^2}{x_{\max}^2}\right)\left(x'^2+y'^2+c^2\right)^{-1.5} \tag{31}$$

where $x'=(x-x_c)/x_{\max}, y'=(y-y_c)/y_{\max}$ and $x_{\max}, y_{\max}$ represent the maximum distances along each coordinate direction in the local domain of influence. In our calculations, the given nodes are chosen as the centers of the RBFs.

The absolute error, maximum error and average error of the numerical solution at time t are defined as

$$T_{abs}(\mathbf{p}_k)=\left|T(\mathbf{p}_k,t)-T_{ana}(\mathbf{p}_k,t)\right|,\quad k=1,2,\ldots,N, \tag{32}$$

$$T_{\max}=\max\left|T(\mathbf{p}_k,t)-T_{ana}(\mathbf{p}_k,t)\right|,\quad k=1,2,...,N, \tag{33}$$

$$T_{avg}=\sum_{k=1}^{N}\frac{1}{N}\left|T(\mathbf{p}_k,t)-T_{ana}(\mathbf{p}_k,t)\right|,\quad k=1,2,\ldots,N, \tag{34}$$

where T and T_{ana} stand for numerical and analytical solution, N represents the total number of all nodes, the errors are evaluated on the random and uniform nodes.

For both uniform and non-uniform node arrangement, we leave out the corner points for simplicity. The profile of 51 by 51 uniformly distributed nodes is shown in Fig.1. The random nodes are generated from the uniform nodes through the following transformation

$$\mathbf{p}_i=\mathbf{p}_i+c_{rand}\varepsilon r_{\min}, \tag{35}$$

where $\mathbf{p}_i$ is coordinate of node $\mathbf{p}=(\mathbf{p}_i)$, c_{rand} is a random number between -1 to 1, $r_{\min}$ denotes the minimum distance among different uniform points, ε stands for a displacement factor; $\varepsilon=0.15, 0.25,$ and 0.35 are used in the numerical implementations. The profiles of 51 by 51 randomly distributed nodes with different displacement factor are shown in Fig.3-5. For larger displacement factor, the nodes are generated more randomly.

Example. Consider the following diffusion equation

$$\frac{\partial T}{\partial t}(x,y,t)=\nabla^2T(x,y,t),\quad (x,y)\in\Omega, t>0, \tag{36}$$

with initial condition :

$$T(x,y,t)=0,\quad (x,y)\in\Gamma, t>0, \tag{37}$$

and boundary condition:

$$T(x,y,0)=1,\quad (x,y)\in\Omega\cup\Gamma. \qquad (38)$$

The analytical solution $T(x,y,t)$ is given as [12]

$$T(x,y,t)=\frac{16}{\pi^2}T_{ana}(x,t)T_{ana}(y,t) \qquad (39)$$

where for $\xi = x, y,$

$$T_{ana}(\xi,t)=\sum_{i=0}^{\infty}\frac{(-1)^i\exp[-(2i+1)^2\pi^2 t]\cos[(2i+1)\pi\xi]}{2i+1}. \quad (40)$$

The unit square domain $[-0.5,0.5]^2$ with uniformly and randomly distributed nodes is considered. The parameter n stands for the number of points in the local domain of influence, Δt stands for the size of each time step, Δx stands for the minimum distance among the given nodes, the shape parameter of multiquadrics RBF c is chosen as a constant. The numerical results are obtained on uniformly and randomly distributed nodes which are generated using (35). Fig.6-8 shows the profiles of the analytical solutions at $t=10^{-3},10^{-2},10^{-1}$.

Fig.9-10 show the average errors and the maximum errors using both explicit and implicit methods with different shape parameters. We observe that better accuracy can be achieved using explicit method with larger shape parameter *c*. On the other hand, the implicit method using $c=100$ gives increasing errors when the time is longer. This is due to the increase of ill conditioning of the sparse matrix. This means that the explicit method performs better with a large range of shape parameters. In the all of following numerical results, the shape parameter $c=50$ will be used.

In Fig.11 we show the profiles of the average errors and maximum errors using both methods with 51 by 51 uniform nodes. The $n=5$ stands for the number of points in local domain of influence, the time step Δt is chosen as 10^{-5}. We compare the errors based on given interior node at time t from 0 to 0.1. One can observe the explicit method gives better accuracy than the implicit method when the time is small, but this is not the case when time becomes large. Note that when the shape parameter *c* is 50, as we have seen from Figure 8, the explicit method outperforms the implicit method using larger shape parameter. In order to achieve the same accuracy as implicit method, we can simply increase the value of the shape parameter *c*.

Fig.12-13 shows the profiles of the average errors and the maximum errors which were obtained on 51 by 51 non-uniform nodes, where $\varepsilon=0.15$ is used in Fig. 11, $\varepsilon=0.25$ is used in Fig.12. Tab.1 shows the average and maximum errors at time $10^{-4},10^{-3},10^{-2}$ and 10^{-1} for 51 by 51 randomly distributed nodes using $\varepsilon=0.35$. A larger ε generates increasingly random nodes which gives slightly different numerical behaviors. The stability of both methods is similar for random nodes.

Fig.14 shows the absolute errors of the temperature at time $t=10^{-3}$ and at cross section $(0,y)$, where $y\in[-0.5,0.5]$ using both methods. This is a very challenging temperature state where the temperature keeps jumping near boundary and is difficult to approximate. As we expect, the absolute errors near boundary nodes are relatively larger than the center nodes when the time is small. Fig.15 shows the absolute errors of the temperature at time $t=10^{-2}$ at cross section $(0,y)$, where $y\in[-0.5,0.5]$ using both methods . The errors at central area are increasing as a function of time, and the errors at near boundary area are decreased as we expect since the analytical temperature field is smoother. Fig.16 shows the absolute errors for $t=10^{-1}$. The maximum errors appear at the center since the temperature at the center point is highest and the temperature field is smoother as shown in Figure 8.

Fig.17 shows accuracy of the solution as a function of the size of time step. The average errors and maximum errors are calculated on 51 by 51 uniformly distributed nodes at $t=10^{-2}$. The shape parameter $c=50,$ and the number of points in the local domain of influence is 5. The explicit method performs slightly better when the size of time step is small, but the implicit method is unconditionally stable. With larger time steps such as $10^{-2},10^{-1}$, the explicit method diverges. On the other hand, the decrease of the size of the time step, the errors of both methods are not decreasing anymore since the machine round of errors, which means that implicit method is more stable.

Fig.18 shows the stability of these two methods with respect to the minimum of distance between the given nodes, where in this figure the size of time steps is chosen as $\Delta t=10^{-5}$, and time $t=10^{-3}$, $n=5$, $c=50$. One can observe the improvement of the accuracy with denser nodes using both methods, therefore, the explicit method shows better stability for denser nodes.

Fig.19-30 represents the absolute errors using explicit method and implicit method for $t=10^{-3},10^{-2}$ $t=10^{-1}$, where the errors are calculated on 51 by 51 uniformly distributed nodes in Fig.19-24, and on 51 by 51 non-uniformly distributed nodes in Fig.25-30. Three

Table 1. The lists of the average errors and the maximum errors using explicit method and implicit method, where 51 by 51 random nodes with $\varepsilon=0.35,$ $c=50,\quad n=5,\quad \Delta t=10^{-5}.$

Average Time	Explicit Method	Implicit Method
1.0E-4	4.7432E-3	5.7891E-3
1.0E-3	5.8723E-3	6.1825E-3
1.0E-2	1.3029E-2	1.3088E-2
1.0E-1	9.9629E-3	9.9458E-3
Maximum Time	Explicit Method	Implicit Method
1.0E-4	2.7092E-1	2.6438E-1
1.0E-3	8.9737E-2	9.0123E-2
1.0E-2	4.9142E-2	4.9152E-2
1.0E-1	2.5140E-2	2.5098E-2

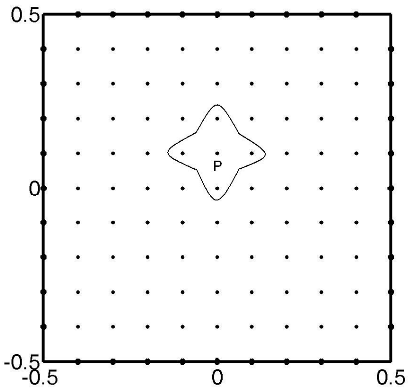

Figure 1: The 11 by 11 uniform node arrangement and the schematics of the local domain of influence using *n=5*.

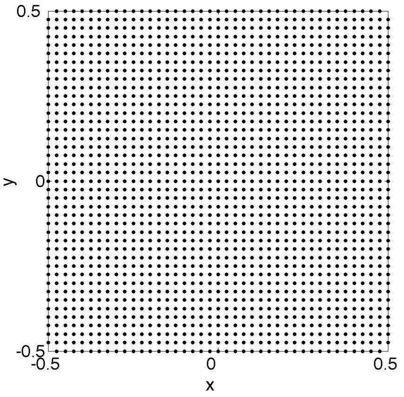

Figure 2: The 51 by 51 uniformly distributed nodes.

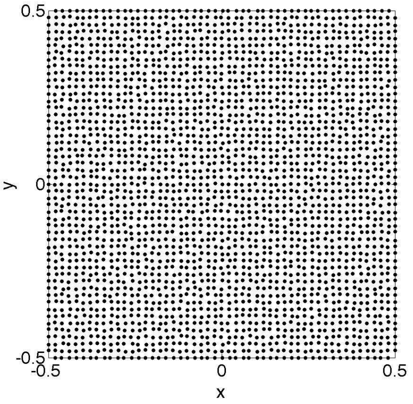

Figure 3: The 51 by 51 randomly distributed nodes with random displacement factor $\varepsilon = 0.15$.

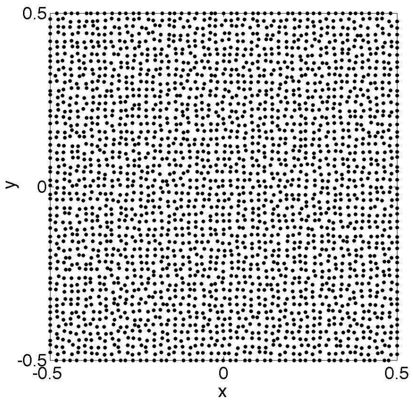

Figure 4: The 51 by 51 randomly distributed nodes with random displacement factor $\varepsilon = 0.25$.

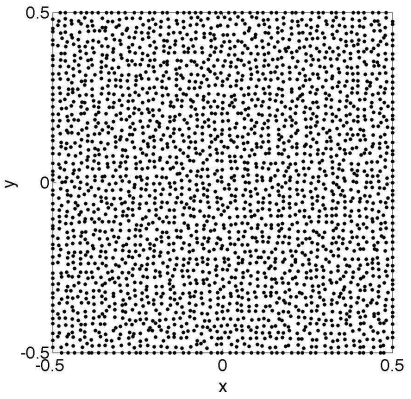

Figure 5: The 51 by 51 randomly distributed nodes with random displacement factor $\varepsilon = 0.35$.

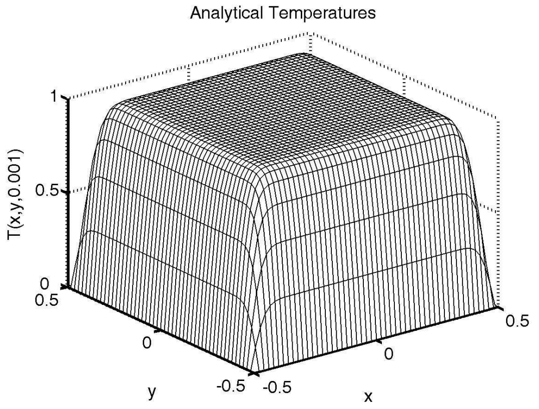

Figure 6: The analytical temperatures at time $t = 10^{-3}$.

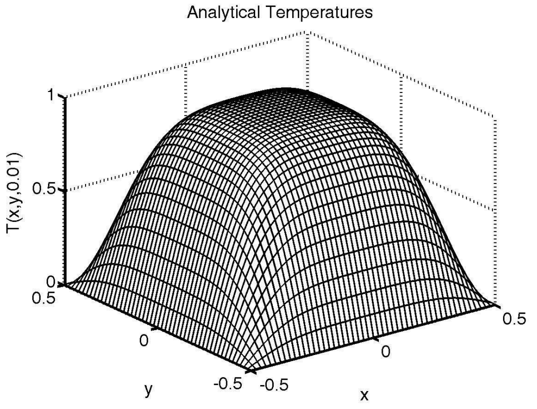

Figure 7: The analytical temperatures at time $t = 10^{-2}$.

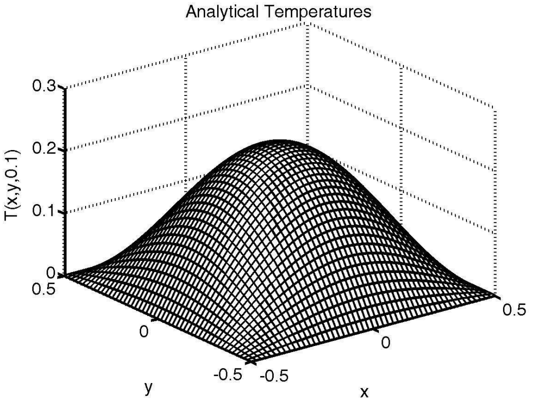

Figure 8 : The analytical temperatures at time $t = 10^{-1}$.

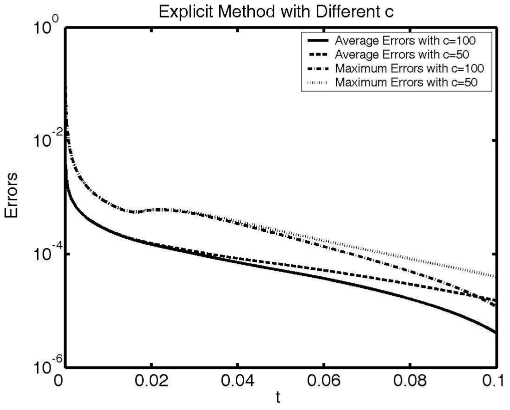

Figure 9: Comparison of the effect of shape parameter c using explicit method. The errors as a function of time are calculated on 51 by 51 uniform nodes with $n = 5$, $\Delta t = 10^{-5}$.

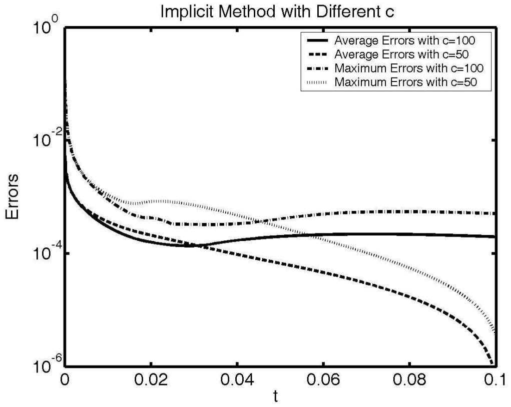

Figure 10: Comparison of the effect of shape parameter c using implicit method. The errors as a function of time are calculated on 51 by 51 uniform nodes using $n = 5$, $\Delta t = 10^{-5}$.

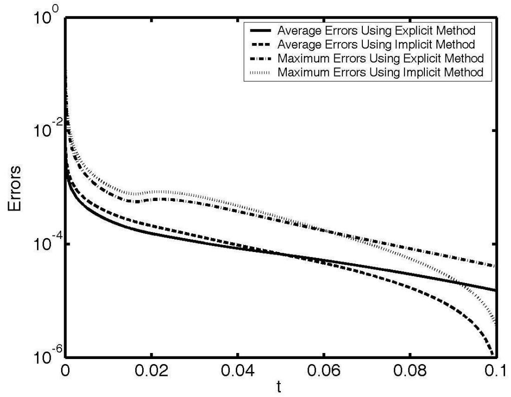

Figure 11: The errors as a function of the methods and time with 51 by 51 uniform nodes using $c = 50$, $n = 5$, $\Delta t = 10^{-5}$.

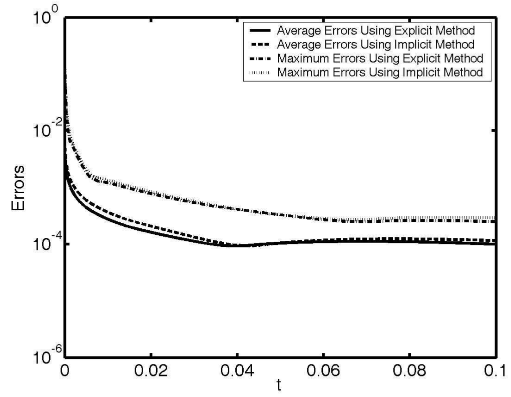

Figure 12: The errors as a function of the methods and time with 51 by 51 random nodes using $\varepsilon = 0.15$, $c = 50$, $n = 5$, $\Delta t = 10^{-5}$.

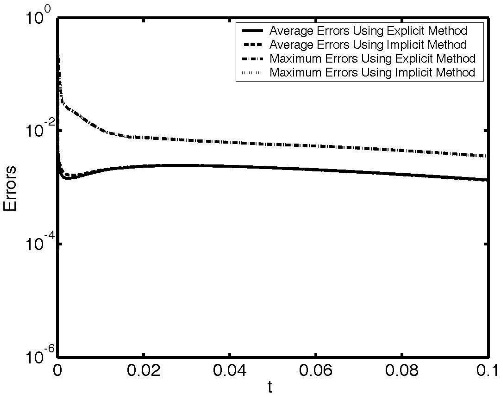

Figure 13: The errors as a function of the methods and time with 51 by 51 random nodes using $\varepsilon = 0.25,\ c = 50,\ \ n = 5,\ \Delta t = 10^{-5}$.

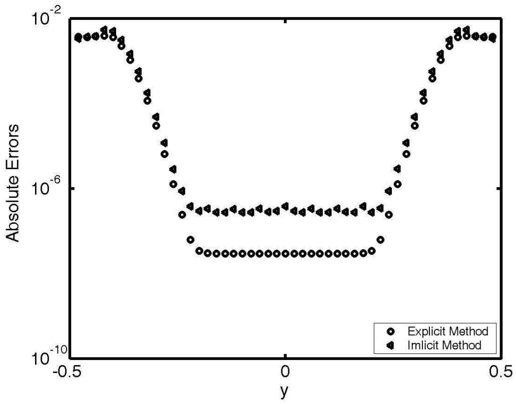

Figure 14: The absolute errors at y direction for $x = 0.0,\ \ \Delta t = 10^{-5}$, with 51 by 51 uniform nodes and $n = 5,\ \ c = 50$ at $t = 10^{-3}$.

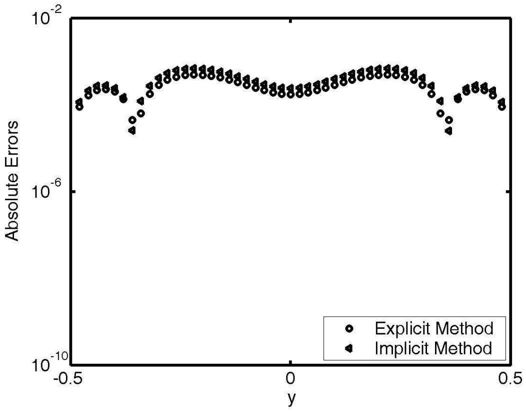

Figure 15: The absolute errors at y direction for $x = 0.0,\ \ \Delta t = 10^{-5}$, with 51 by 51 uniform nodes and $n = 5,\ \ c = 50,$ at $t = 10^{-2}$.

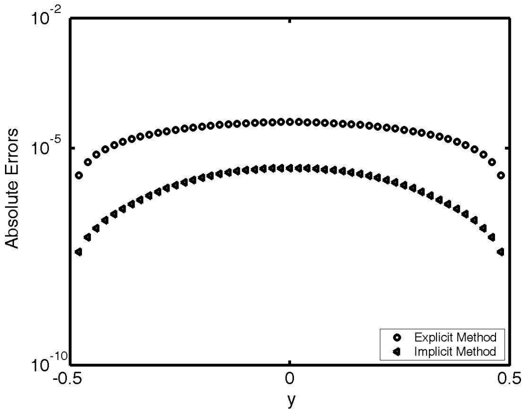

Figure 16: The absolute errors at y direction for $x = 0.0,\ \ \Delta t = 10^{-5}$ with 51 by 51 uniform nodes and $n = 5,\ \ c = 50,$ at $t = 10^{-1}$.

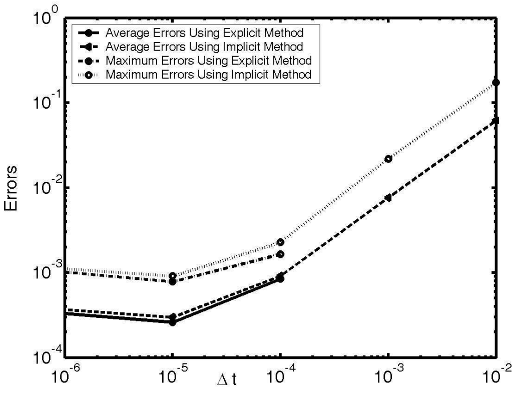

Figure 17: The errors as a function of size of time step with 51 by 51 uniform node arrangement with $\Delta t = 10^{-5},\ \ n = 5,\ \ c = 50,\ \ t = 10^{-2}$.

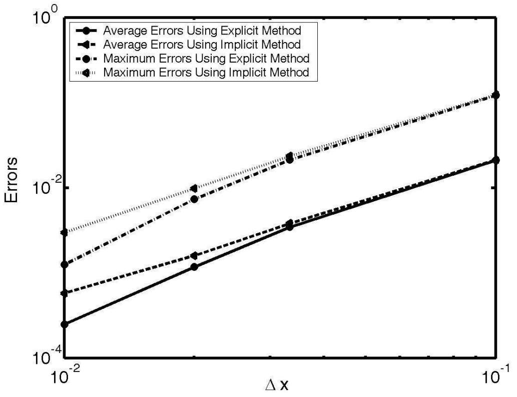

Figure 18: The errors as a function minimum node distance with $\Delta t = 10^{-5},\ \ n = 5,\ \ c = 50,\ \ t = 10^{-3}$.

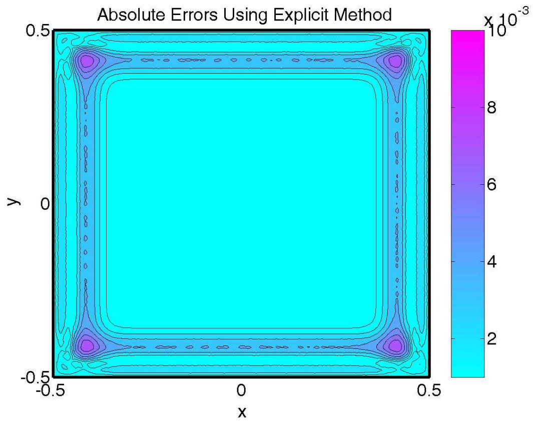

Figure 19: The absolute errors using explicit method with 51 by 51 uniform nodes, where $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-3}$.

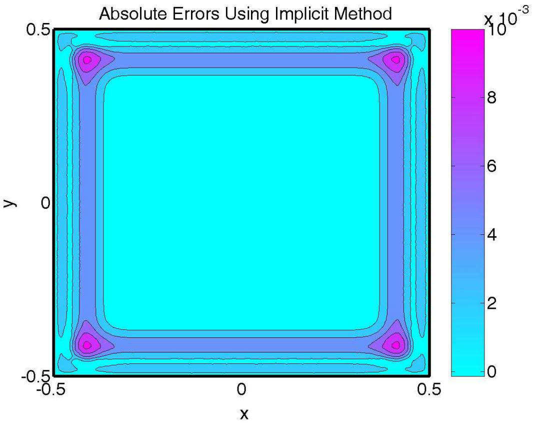

Figure 20: The absolute errors using implicit method with 51 by 51 uniform nodes using $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-3}$.

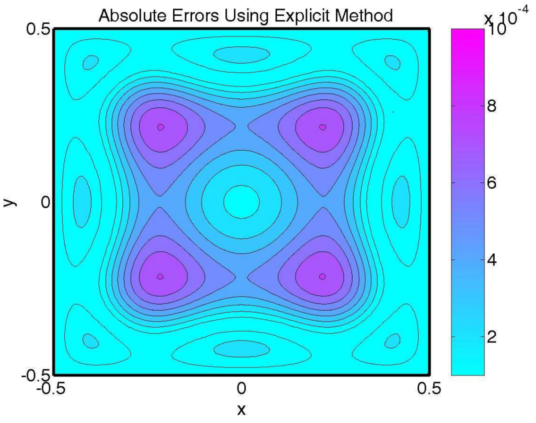

Figure 21: The absolute errors using explicit method and implicit method with 51 by 51 uniform nodes using $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-2}$.

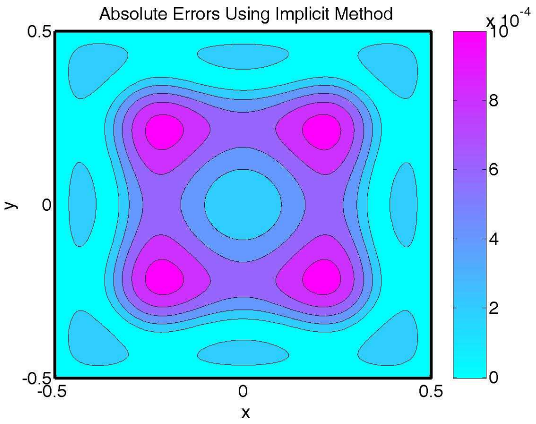

Figure 22: The absolute errors using implicit method with 51 by 51 uniform node arrangement using $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-2}$.

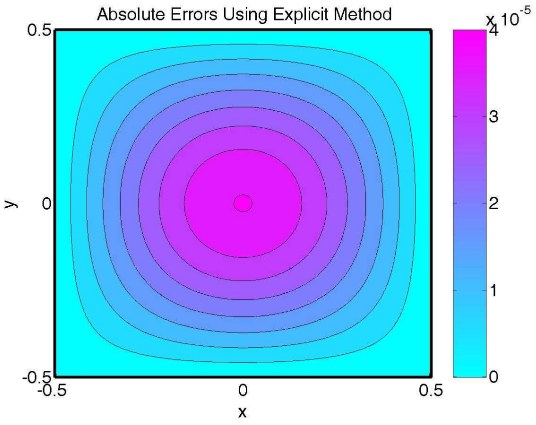

Figure 23: The absolute errors using explicit method with 51 by 51 uniform nodes and $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-1}$.

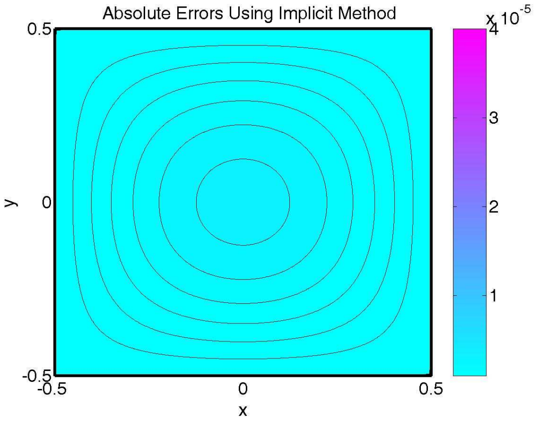

Figure 24: The absolute errors using implicit method with 51 by 51 uniform nodes using $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-1}$.

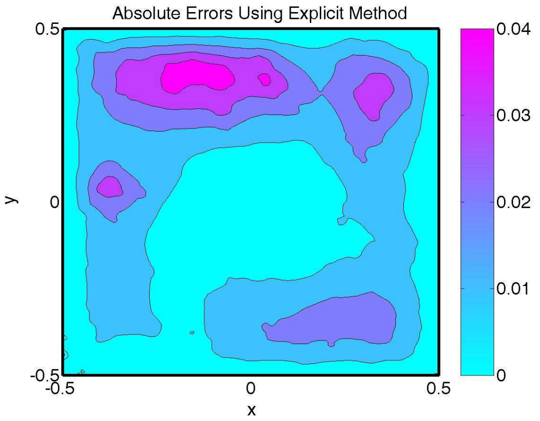

Figure 25: The absolute errors using explicit method with 51 by 51 random nodes using $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-3}$.

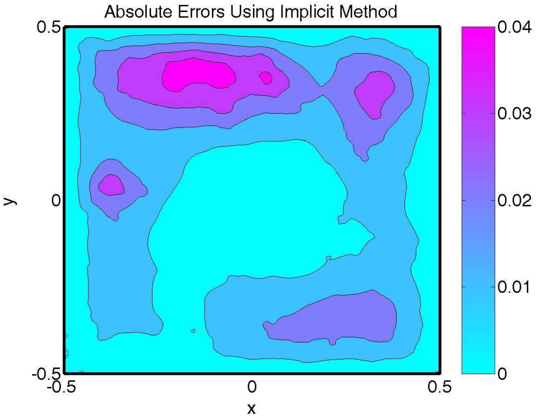

Figure 26: The absolute errors using implicit method with 51 by 51 random node arrangement, where $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-3}$.

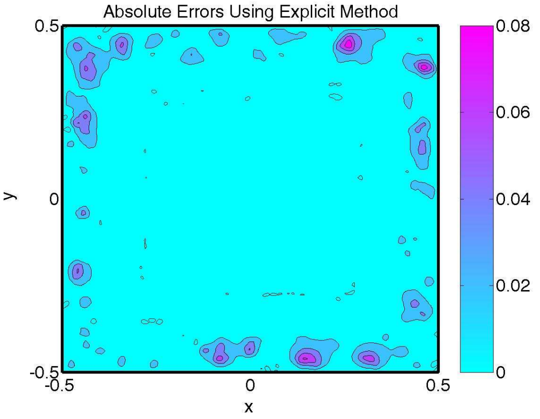

Figure 27: The absolute errors using explicit method with 51 by 51 random node arrangement using $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-2}$.

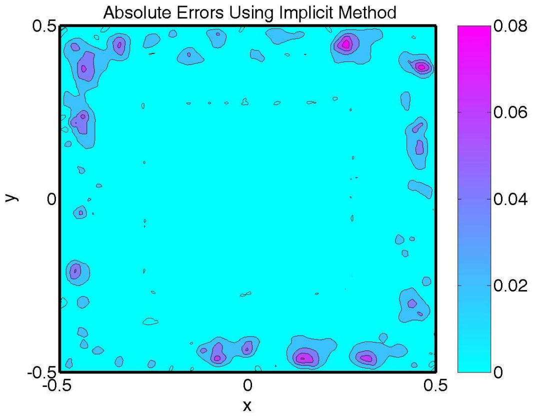

Figure 28: The absolute errors using implicit method with 51 by 51 random node arrangement, where $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-2}$.

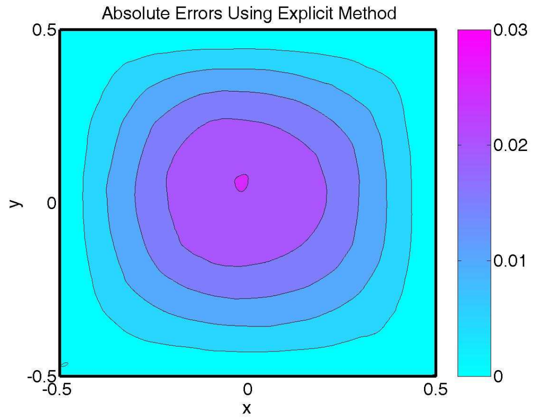

Figure 29: The absolute errors using explicit method with 51 by 51 random node arrangement using $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-1}$.

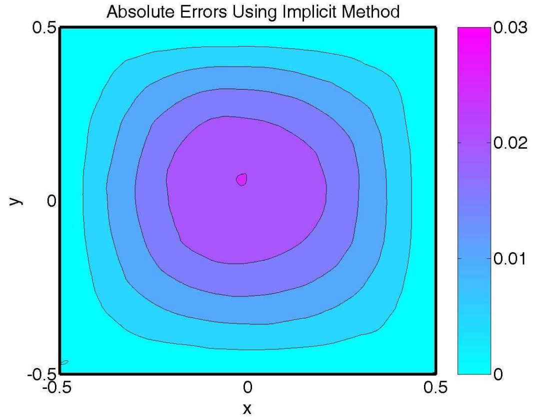

Figure 30: The absolute errors using implicit method with 51 by 51 random node arrangement, where $\varepsilon = 0.35$, $\Delta t = 10^{-5}$, $n = 5$, $c = 50$, $t = 10^{-1}$.

time fields $t=10^{-3},10^{-2},10^{-1}$ are graphed for both kinds of nodes. For the shorter time such as $t=10^{-3}$, the errors at the corner points are relatively large since the neighbor of the corner points includes two boundary nodes in the uniform node arrangement, which decreased the accuracy. For non-uniformly distributed nodes, both methods have slightly different, and the error fields are not symmetric as uniformly distributed nodes.

The computer program has been coded in FORTRAN with double precision. The computational cost of both methods is primarily dependent on two factors: the computational time of each time step to find the approximate temperatures at all given node, denoted by R, and the number of time steps m. Total time cost can be represented by $\mathrm{R}\cdot\mathrm{m}$. The computational costs of explicit method and implicit method at each time step using 101 by 101 random node arrangements are $0.0560\cdot\mathrm{m}$ and $0.1120+0.3720\cdot\mathrm{m}$ seconds, respectively, where 0.1120 is the time for setup sparse matrix in implicit method which does not change at different time step. The implicit method takes much longer time than explicit method at each time step, but implicit method works fine with larger size of time step Δt, by the inverse proportion relationship between Δt and m the computational time of implicit method can be less than explicit method if Δt is chosen large enough.

5. Conclusions

Two kind of localized meshless methods were compared for Dirichlet jump problem for diffusion equation: ELRBFCM and ILRBFCM. The ELRBFCM is made locally over a set of overlapping domains of influence and the time stepping is performed in an explicit way, small systems of linear equations have to be solved in each time step for each node and associated domain of influence. The ILRBFCM is performed in an implicit way and resultant matrix is large sparse matrix which is solved by sparse system package Y12MA (for more detailed discussion, see [13]). The numerical results show high accuracies and the improvement of the accuracies with denser nodes and the smaller time step length for both methods. On the other hand, ELRBFCM performs best with small time step and larger range of shape parameters for shorter time, but ILRBFCM gives better results for longer time and converges even with large time step length. For random node arrangement problems, both methods perform with similar accuracy with small time step length. Our ongoing research is focused on inclusion of the phase-change effect to the present simple diffusion equations.

Acknowledgement

This paper represents activities within project INSPIRE by the early stage researcher Guangming Yao in the period from September 2009 till December 2009 in the IMPOL d.d. Aluminium Industry, Slovenia. The financial support from EU is kindly acknowledged.

In addition, the second and the fourth author would like to acknowledge the financial support within SLO-US bilateral project “Advanced Meshless Methods”.

References

[1] Lorbiecka A. Z., and Šarler B., *Point Automata Method for Prediction of Grain Structure in the Continuous Casting of Steel,* Proceedings, 3rd International Conference on Simulation and Modelling of Metallurgical Processes in Steelmaking, Ludwig A. eds., Leoben: ASMET, Austria, 2009, pp. 192-197.

[2] Lorbiecka A. Z., and Šarler B., *Meshless Point Automata Method for Simulation of Dendritic Growth*, Book of abstracts, 5th ICCES International Symposium on Meshless and Other Novel Computational Methods, Atluri, S. N. and Šarler, B. eds., Ljubljana, Slovenia, University of Nova Gorica, 2009, pp. 8.

[3] Šarler B., Vertnik R., *Meshfree Explicit Local Radial Basis Function Collocation Method for Diffusion Problems*, Computers and Mathematics with Applications, 2006, pp.1269-1282.

[4] Lee C. K., Liu X., and Fan S. C., *Local Multiquadric Approximation for Solving Boundary Value Problems*, Computational Mechanics, Vol. 30, 2003, pp. 396-409.

[5] Liu G. R., *Mesh Free Methods: Moving Beyond the Finite Element Method*, CRC Press, Boca Raton, 2003.

[6] Atluri S. N. and Shen S., *The Meshless Local Petrov-Galerkin (MLPG) Method*, Tech Science Press, Encino, 2002.

[7] Fasshauer G. E., *Meshfree Application Method with Matlab*, Interdiscilinary Mathematical Sciences 6, 2007.

[8] Atluri S. N., Shen S., *The Meshless Method*, Tech Science Press, Forsyth, 2002.

[9] Šarler B., *From Global to Local Radial Basis Function Collocation Method for Transport Phenomena, in: V .M. A. Leitao, C. J. S. Alves, C. Armando-Duarte*, Advances in Meshfree Techniques, Computational Methods in Applied Sciences, Vol. 5, 2007, pp.257-282.

[10] Chen C. S., Yao G. M., *A Localized Approach for the Method of Approximate Particular Solutions*, Book of abstracts, 5th ICCES International Symposium on Meshless and Other Novel Computational Methods, Atluri, S. N. and Šarler, B. eds., Ljubljana, Slovenia, University of Nova Gorica, 2009, pp. 2.

[11] Vertnik R., *Local Collocation Method for Phase-Change Problems*, Master’s Thesis, University of Nova Gorica, Nova Grica, 2007.

[12] Carlsaw H. S., Jaeger J. C., *Condition of Heat in Solids*, Clarendon Press, Oxford, 1995.

[13] Zlatev Z., Wasniewski J., Schaumburg K., *Y12M, Solution of Large and Sparse Systems of Linear Algebraic Equations,* Lecture Notes in Computer Science, Springer, Vol. 121, 1981.