Received 15 May 2007
Revised 11 December 2007
Accepted 12 December 2007

# Solution of thermo-fluid problems by collocation with local pressure correction

Gregor Kosec and Božidar Šarler
*Laboratory for Multiphase Processes, University of Nova Gorica, Nova Gorica, Slovenia*

## Abstract

**Purpose** – The purpose of this paper is to explore the application of the mesh-free local radial basis function collocation method (RBFCM) in solution of coupled heat transfer and fluid-flow problems.

**Design/methodology/approach** – The involved temperature, velocity and pressure fields are represented on overlapping five nodded sub-domains through collocation by using multiquadrics radial basis functions (RBF). The involved first and second derivatives of the fields are calculated from the respective derivatives of the RBFs. The energy and momentum equations are solved through explicit time stepping.

**Findings** – The performance of the method is assessed on the classical two dimensional de Vahl Davis steady natural convection benchmark for Rayleigh numbers from $10^3$ to $10^8$ and Prandtl number 0.71. The results show good agreement with other methods at a given range.

**Originality/value** – The pressure-velocity coupling is calculated iteratively, with pressure correction, predicted from the local mass continuity equation violation. This formulation does not require solution of pressure Poisson or pressure correction Poisson equations and thus much simplifies the previous attempts in the field.

**Keywords** Flow, Convection, Pressure, Thermodynamics, Fluid dynamics

**Paper type** Research paper

## 1. Introduction

The most commonly used discrete approximate methods for solving systems of partial differential equations (PDEs) in fluid-flow problems are the finite difference method, the finite volume method, the finite element method, the spectral method and the boundary element method (BEM). Despite the suitability of the enumerated methods for solving fluid flow as well as other physical situations, there are still some substantial difficulties in applying them to realistic, geometrically complex three dimensional systems. The major problem is in creating a suitable mesh. The meshing (polygonisation) is often the most time consuming part of the solution process and is far from being fully automated. However, there is a rapidly emerging branch of numerical methods, where there is no need to create a polygonisation, neither in the domain nor on its boundary. The solution is represented on the arbitrarily distributed set of nodes without any additional topological relations between them. These meshfree methods represent a promising technique to avoid the meshing problems

The authors would like to express their gratitude to Slovenian Research Agency for support in the framework of the projects Young Researcher Programme 1000-06-310232 (G.K.), J2-6403 and L9-9508 (B.Š.).

International Journal of Numerical Methods for Heat & Fluid Flow
Vol. 18 No. 7/8, 2008
pp. 868-882

0961-5539
DOI 10.1108/09615530810898999

(Atluri and Shen, 2002a, b; Atluri, 2004; Chen, 2002; Kansa, 1990a; Liu, 2003; Liu and Gu, 2005).

A number of mesh-reduction techniques such as the dual reciprocity BEM with radial basis functions (RBF) (Šarler and Kuhn, 1999), mesh-free techniques such as the dual reciprocity method with fundamental solutions (Šarler, 2002), meshfree local Petrov Galerkin methods (MLPG) (Atluri and Shen, 2002b, Lin and Atluri, 2001) have been developed for transport phenomena and solution of the Navier-Stokes equations. This paper is focused on the simplest class of mesh-free methods in development today, the radial basis function collocation methods (RBFCM) (Buhmann, 2000; Šarler, 2002, 2007).

The fluid-flow problem is a global problem in general. In order to solve such global problem, one needs to solve the global matrix (Šarler *et al.*, 2004; Šarler, 2005). Solving matrices for global systems with fine mesh or complex geometries can become a major numerical problem, therefore completely local scheme for solving fluid-flow problems is proposed in the present paper. This method represents a local variant of the already developed global RBFCM solution (Šarler, 2005), for coupled heat transfer and fluid flow problems. This local variant has been previously developed for diffusion problems (Šarler and Vertnik, 2006), convection-diffusion solid-liquid phase change problems (Vertnik and Šarler, 2006) and subsequently successfully applied in industrial process of direct chill casting (Vertnik *et al.*, 2006). The spectra of physics coped is extended to solution of coupled mass, energy and momentum equations in this paper. Instead of solving the pressure Poisson equation or/and pressure correction Poisson equation (Divo and Kassab, 2007), a much simplified local pressure-velocity coupling (LPVC) algorithm is proposed. The new algorithm is tested on classical de Vahl Davis (de Vahl Davis, 1983; Hortmann *et al.*, 1990; Manzari, 1999) natural convection problem. The results of the method are assessed in terms of streamfunction extreme, cavity Nusselt number, and mid-plane velocity components.

## 2. Governing equations

The steady-state natural convection problem is described by three coupled PDEs and Boussinesq approximation. The PDEs are mass, momentum and energy conservation equations where all material properties are considered to be constant. The equations are given as:

$$\nabla \cdot \mathbf{v} = 0, \tag{1}$$

$$\nabla \cdot (\rho \mathbf{v}\mathbf{v}) = -\nabla P + \nabla \cdot (\mu \nabla \mathbf{v}) + \mathbf{f}, \tag{2}$$

$$\nabla \cdot (\rho c_p T \mathbf{v}) = \nabla \cdot (\lambda \nabla T), \tag{3}$$

$$\mathbf{f} = \rho[1 - \beta_B (T - T_{\text{ref}})]\mathbf{g}, \tag{4}$$

with $\mathbf{v}, P, T, \lambda, c_p, \mathbf{g}, \rho, \beta_B, T_{\text{ref}}, \mu$ and $\mathbf{f}$ standing for velocity, pressure, temperature, thermal conductivity, specific heat, gravitational acceleration, density, coefficient of thermal expansion, reference temperature for Boussinesq approximation, viscosity and body force, respectively. The problem is solved on a fixed domain $\Omega$ with boundary $\Gamma$ where Dirichlet and Neumann boundary conditions for temperature might be used and Dirichlet boundary conditions for velocity are used.

## 3. Solution procedure

In order to solve the problem, the time dependent variant of equations (2) and (3) is employed. The explicit time scheme is adopted to cope with the transience terms in momentum and energy equations. The Navier-Stokes equations (1) and (2) are solved iteratively. The LPVC algorithm, where pressure correction is estimated from local mass continuity violation, is used to drive the intermediate velocity towards the divergence-free velocity. The basic elements of the solution procedure are as follows:

In the first step, the velocity is estimated from the discretized transient form of equation (2):

$$\hat{\mathbf{v}} = \mathbf{v}_0 + \frac{\Delta t}{\rho}\left[-\nabla P_0 + \nabla\cdot(\mu\nabla\mathbf{v}_0) + \mathbf{f}_0 - \nabla\cdot(\rho\mathbf{v}_0\mathbf{v}_0)\right], \tag{5}$$

where $\hat{\mathbf{v}}$ denotes velocity at time $t_0 + \Delta t$, $\mathbf{v}_0$, $P_0$ denote velocity and pressure at time $t_0$ and $\Delta t$ stands for the time-step length. The calculated velocity $\hat{\mathbf{v}}$ does not satisfy the mass continuity equation (1) in general. In order to couple mass continuity equation with the momentum equation, an iteration process is used where in the first iteration, the velocity and the pressure are set to:

$$\begin{aligned} \mathbf{v}^m &= \hat{\mathbf{v}}, \\ P^m &= P_0 \end{aligned} \quad ; m = 1, \tag{6}$$

where $m$ stands for iteration index. To project the velocity into the divergence free space, a correction term $\breve{\mathbf{v}}$ is added:

$$\nabla\cdot(\mathbf{v}^m + \breve{\mathbf{v}}) = 0 \;\rightarrow\; \nabla\cdot\mathbf{v}^m = -\nabla\cdot\breve{\mathbf{v}}\,. \tag{7}$$

The velocity correction is assumed to be affected only by effect of the pressure correction:

$$\breve{\mathbf{v}} = -\frac{\Delta t}{\rho}\nabla\breve{P}, \tag{8}$$

where $\breve{P}$ stands for the pressure correction. The pressure correction Poisson equation is constructed by applying the divergence over equation (8):

$$\nabla^2\breve{P} = \frac{\rho}{\Delta t}\nabla\cdot\mathbf{v}^m. \tag{9}$$

Instead of solving the equation (9) globally with the appropriate pressure correction boundary conditions (Divo and Kassab, 2007), the pressure correction is assumed to be linearly related to the Laplace of pressure correction. In the second step, the pressure correction is therefore calculated as:

$$\breve{P} \approx L^2\nabla^2\breve{P} = L^2\frac{\rho}{\Delta t}\nabla\cdot\mathbf{v}^m, \tag{10}$$

where $L$ stands for characteristic length. The very important assumption (10) enables for solving the problem completely locally. In the third step, the intermediate pressure and velocity are corrected as:

$$P^{m+1} = P^m + \beta \breve{P},$$
$$\mathbf{v}^{m+1} = \mathbf{v}^m - \beta \frac{\Delta t}{\rho} \nabla \breve{P}, \tag{11}$$

where $\beta$ stands for suitable relaxation parameter. If the criteria:

$$\nabla \cdot \mathbf{v}^{m+1} < \varepsilon_v, \tag{12}$$

is not met than the iteration returns back to the equation (10), else the pressure-velocity iteration is completed and the calculation proceeds to the next step.

The fourth step is to solve the transient form of the energy equation (3):

$$T = T_0 + \frac{\Delta t}{\rho c_p} \left[ \nabla \cdot (\lambda \nabla T_0) - \nabla \cdot (\rho c_p T_0 \mathbf{v}_0) \right], \tag{13}$$

where $T_0$ and $T$ denote temperature at time $t_0$ and $t_0 + \Delta t$. The steady-state is achieved when the criteria:

$$\begin{gathered} \frac{|T - T_0|}{|T_0|} < \varepsilon_T; \quad T_0 \neq 0 \\ T < \varepsilon_T; \quad T_0 = 0 \end{gathered} \tag{14}$$

is met in all nodes. If the criteria (14) is not met, the body force is updated and calculation returns back to equation (5). The simulation flowchart is shown in Figure 1.

## 4. Radial basis function collocation method

The pressure, velocity and temperature fields are interpolated on the coincident grid points by Hardy's multiquadrics RBF. The arbitrary function $\theta$ is represented on each of the local sub-domains as:

$$\theta(\mathbf{p}) \approx \sum_{n=1}^{N} \alpha_n \Lambda_n(\mathbf{p}), \tag{15}$$

with $\mathbf{p}, \Lambda_n, \alpha_n$ and $N$ standing for the position vector, the basis function, the collocation coefficient and the number of the collocation points, respectively. Hardy's multiquadrics basis functions are defined as:

$$\Lambda_n(\mathbf{p}) = \sqrt{r_n^2(\mathbf{p}) + c^2 r_0^2}; \quad r_n^2 = (\mathbf{p} - \mathbf{p}_n) \cdot (\mathbf{p} - \mathbf{p}_n), \tag{16}$$

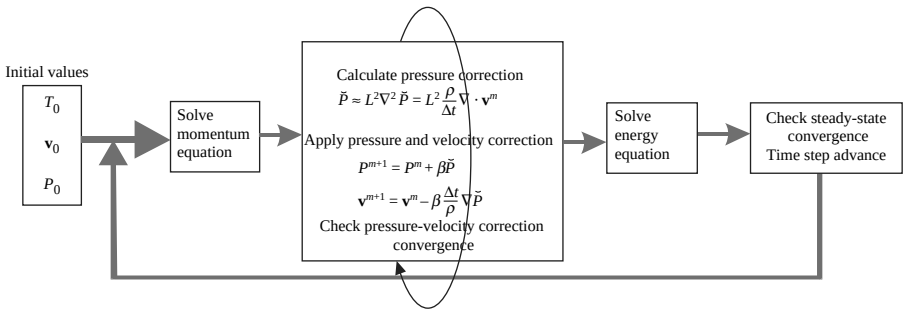


**Figure 1.**
The calculation flowchart

where $c$ represents a dimensionless shape parameter. The scaling parameter $r_0^2$ is set to the maximum nodal distance of the sub-domain. The coefficients $\alpha_n$ are obtained from the collocation condition which implies the exact satisfaction of the equation (15) in the nodal points where equation (15) must hold. In case, the number of the nodes is the same as the number of the terms in the series (15), the system simplifies to:

$$\theta(\mathbf{p}_i) = \theta_i = \sum_{n=1}^{N} \alpha_n \Lambda_n(\mathbf{p}_i), \tag{17}$$

$$\begin{bmatrix} \Lambda_{11} & \dots & \Lambda_{1N} \\ \dots & \dots & \dots \\ \Lambda_{N1} & \dots & \Lambda_{NN} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \dots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \dots \\ \theta_N \end{bmatrix}, \tag{18}$$

where $\Lambda_{ni} = \Lambda_n(\mathbf{p}_i)$. Solution of the linear system of equation (18) provides the collocation coefficients $\alpha_n$. Spatial derivatives of the function $\theta$ can be easily obtained through derivation of the equation (15):

$$\frac{\partial}{\partial p_\sigma} \theta(\mathbf{p}) \approx \sum_{n=1}^{N} \alpha_n \frac{\partial}{\partial p_\sigma} \Lambda_n(\mathbf{p}), \tag{19}$$

$$\frac{\partial^2}{\partial^2 p_\sigma} \theta(\mathbf{p}) \approx \sum_{n=1}^{N} \alpha_n \frac{\partial^2}{\partial^2 p_\sigma} \Lambda_n(\mathbf{p}), \tag{20}$$

where $p_{\sigma = x, y}$ stands for Cartesian coordinates. All necessary derivatives to construct the involved divergence, gradient and Laplace operators can be calculated through equations (19) and (20). The integral of function $\theta$ over $p_\sigma$ (used in the present context in calculation of the stream function) can be evaluated as well:

$$\int \theta(\mathbf{p}) dp_\sigma = \sum_{n=1}^{N} \alpha_n \int \Lambda_n(\mathbf{p}) dp_\sigma. \tag{21}$$

All matrix elements $\Lambda_{ni}$ and coefficients $\alpha_n$ need to be evaluated only once before time stepping begins.

Only the simplest sub-domain of five points is used within the overlapping collocation sub-domain strategy of the present paper. The described collocation method and sub-domain selection is schematically shown in Figure 2, where a five noded collocation sub-domain is used to approximate the first and the second spatial derivatives in the central node. The derivative instead of the function value is prescribed at the boundary collocation points with the Neumann boundary conditions. The equation (17) is replaced with:

$$\frac{\partial}{\partial p_\sigma} \theta(\mathbf{p}_i) = \sum_{n=1}^{N} \alpha_n \frac{\partial}{\partial p_\sigma} \Lambda_n(\mathbf{p}_i), \tag{22}$$

in such points. The index $i$ stands for node where derivative is known.

## 5. Numerical examples

The classical de Vahl Davis (1983) natural convection problem is considered for benchmarking purposes. The domain of the problem (Figure 3) is a closed air filled (Prandtl number = 0.71) square cavity with differentially heated vertical walls ($\Delta T = T_H - T_C$) and insulated horizontal walls.

The steady-state is achieved through a time transient from the initial temperature, pressure and velocity all set to zero. All results are stated in Cartesian coordinates and standard dimensionless form (Wan *et al.*, 2001):

$$x = \frac{\bar{x}}{L}, \quad y = \frac{\bar{y}}{L}, \quad u = \frac{\bar{u}L\rho c_p}{\lambda}, \quad v = \frac{\bar{v}L\rho c_p}{\lambda}, \quad \Psi = \frac{T - T_C}{T_H - T_C}, \quad \Gamma = t\frac{\lambda}{\rho c_p L^2}, \tag{23}$$

where $x$, $y$ stand for the dimensionless coordinates, $u$, $v$ stand for the dimensionless horizontal and vertical velocity components, $\Psi$ stands for the dimensionless temperature and $\tau$ stands for the dimensionless time. Prandtl and Rayleigh numbers are calculated from the expressions:

$$\mathrm{Pr} = \frac{\mu c_p}{\lambda}, \tag{24}$$

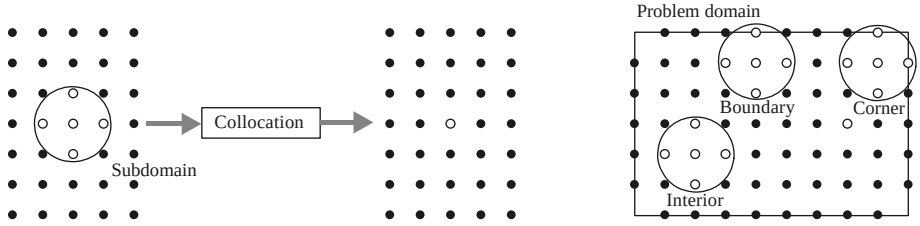


**Figure 2.**
Local collocation scheme (left) and collocation sub-domain selection strategy (right)

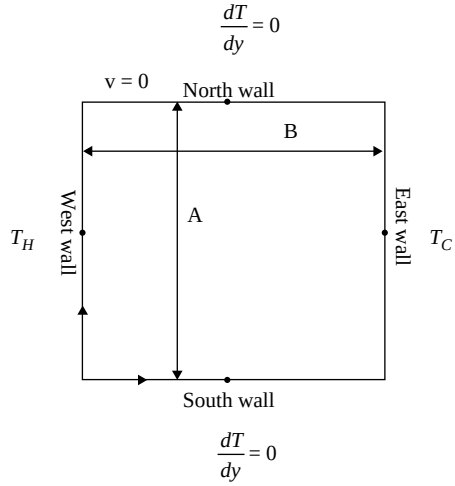


**Figure 3.**
The problem description

$$\mathrm{Ra} = \frac{g\beta\Delta T L^3 \rho^2 c_p}{\lambda\mu}, \tag{25}$$

where $\Delta T$ stands for maximum temperature difference and $L$ stands for enclosure length.

Non-permeable and no slip (due to consideration of viscid fluid) boundary conditions are adopted on the whole boundary $\Gamma$:

$$\mathbf{v}_\Gamma = \mathbf{0}. \tag{26}$$

The results are presented in terms of stream functions and temperature contours in Figure 4 and mid-plane velocities in Figure 5, respectively. The temperature contour plot step is 0.05 for all cases while streamfunction contour plot step is 0.1 for $\mathrm{Ra} = 10^3$, 0.5 for $\mathrm{Ra} = 10^4$, 1 for $\mathrm{Ra} = 10^5$ and $\mathrm{Ra} = 10^6$ and 5 for $\mathrm{Ra} = 10^7$ and $\mathrm{Ra} = 10^8$. In order to enable a straightforward numerical comparison, the mid-plane velocity values are stated in Table I, as well.

A comparison of our results (Table II) with the other similar attempts is done for the maximum mid-plane velocities, the mid-point stream function value, and the average Nusselt number on hot or cold wall where the results of the present study are compared with de Vahl Davis (1983), Sadat and Couturier (2000), Wan *et al.* (2001) and Šarler (2005) the streamfunction $\psi$ is calculated through integration of the velocity component:

$$\psi(x,y) = \int u(x,y)\mathrm{d}y. \tag{27}$$

The Nusselt number is calculated locally on the support of five collocation nodes through expression:

$$\mathrm{Nu}(x,y) = -\frac{\partial\Psi(x,y)}{\partial x} + u(x,y)\Psi(x,y). \tag{28}$$

Our simulations are performed on $41 \times 41$ (with $N_{\max} = 1{,}677$), $81 \times 81$ (with $N_{\max} = 6{,}557$) and $101 \times 101$ (with $N_{\max} = 10{,}197$) grid sizes, where $N_{\max}$ stands for the total number of the grid points. Additional check on the global mass conservation of the method is done by considering the time dependent mass continuity equation (1) globally, to check the numerical mass leakage. The following equation is implemented:

$$\bar{\rho}(t+\Delta t) = \bar{\rho}(t) - \Delta t \rho_0 \frac{1}{N_{\max}} \sum_{n=1}^{N_{\max}} \nabla\cdot\mathbf{v}_n\,; \quad \bar{\rho}(t=0) = \rho_0, \tag{29}$$

where $\bar{\rho}(t)$ stands for time dependent global density. The global density change is introduced as:

$$\Delta\rho = |\rho_0 - \bar{\rho}(N_t\Delta t)|, \tag{30}$$

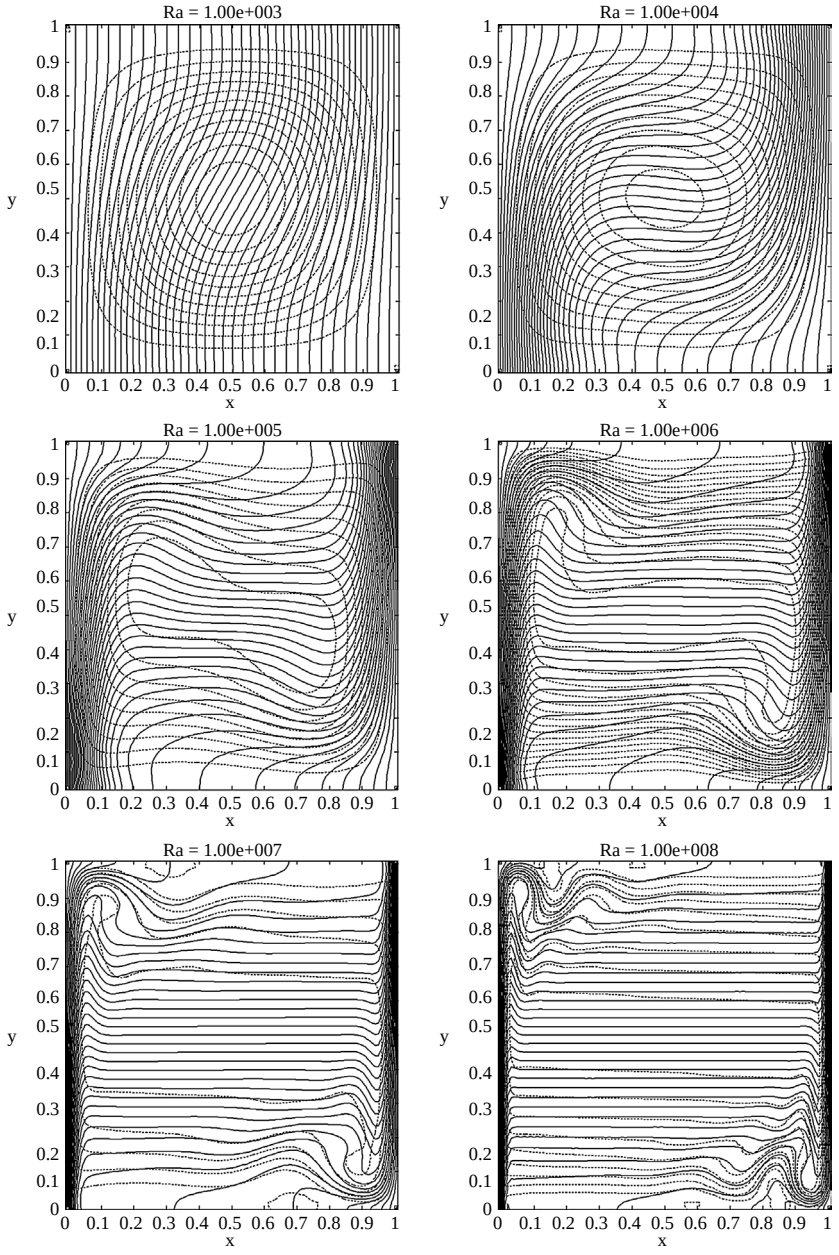


**Figure 4.**
Temperature contours (solid line) and streamline plots (dotted line)

where $N_t$ stands for the number of time-steps. The relative density changes $\Delta\rho/\rho_0$ occurring as a function of different Rayleigh numbers for calculation with the grid density 101 × 101 are stated in Table III. The time-step criteria $\epsilon_T < 10^{-5}$ is used in all cases while pressure-velocity iteration criteria varies for different Rayleigh number

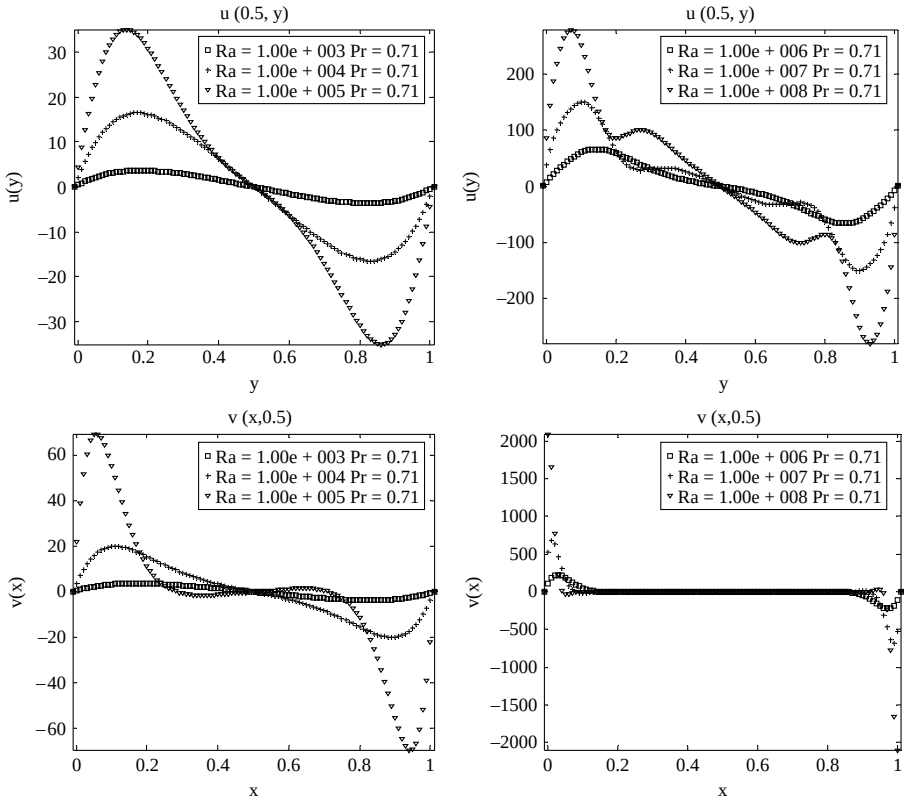


**Figure 5.**
Mid-plane velocities

(Table III). The time-step varies for different Rayleigh numbers, as well. The time-steps were determined by seeking the convergent scenario. The relaxation parameter is set to same value as dimensionless time-step in all cases. The number of pressure-velocity iterations, time-steps and actual calculation time for different Rayleigh numbers and grid sizes are stated in Table III, as well. All cases are calculated with RBF shape parameter $c = 30$. This issue is justified by the sensitivity study, given in the Appendix.

*5.1 Numerical implementation and discussion of the results*

Numerical implementation is done in C++ programming language in double precision and compiled with Intel C++9.1 compiler. The LAPACK routines are used to solve the LU decomposition. The parallelisation is implemented with OpenMP library with maximum 1.85 × speedup factor achieved for two CPU cores. All calculations are done on a laptop computer Toshiba Satellite 100, with duo core Intel 2.16 GHz processor and 1 Gb of RAM. Approximately, 1, 5 and 8 Mb of RAM storage are required for grid densities 41 × 41, 81 × 81 and 101 × 101. The pressure correction requires only one step and therefore the algorithm needs small number of calculations per iteration cycle and this makes the algorithm fast and robust. Each pressure-velocity coupling iteration

| x/y | 0.000 | 0.05 | 0.1 | 0.15 | 0.2 | 0.25 | 0.3 | 0.35 | 0.4 | 0.45 | 0.5 | Ra |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $u$ | 0.000 | 1.922 | 3.042 | 3.572 | 3.655 | 3.421 | 2.958 | 2.337 | 1.611 | 0.820 | 0.000 | $10^3$ |
| $v$ | 0.000 | 2.044 | 3.182 | 3.669 | 3.681 | 3.377 | 2.865 | 2.226 | 1.515 | 0.765 | 0.000 | |
| $u$ | 0.000 | 8.884 | 14.084 | 16.358 | 16.406 | 14.961 | 12.575 | 9.665 | 6.511 | 3.264 | 0.000 | $10^4$ |
| $v$ | 0.000 | 14.680 | 19.929 | 19.411 | 16.113 | 12.195 | 8.647 | 5.768 | 3.503 | 1.646 | 0.000 | |
| $u$ | 0.000 | 20.218 | 32.133 | 35.340 | 31.839 | 25.018 | 17.704 | 11.462 | 6.690 | 3.048 | 0.000 | $10^5$ |
| $v$ | 0.000 | 67.778 | 60.088 | 33.245 | 13.201 | 3.262 | −0.529 | −1.453 | −1.222 | −0.672 | 0.000 | |
| $u$ | 0.000 | 34.504 | 58.012 | 66.594 | 60.433 | 46.729 | 32.256 | 20.208 | 11.356 | 5.009 | 0.000 | $10^6$ |
| $v$ | 0.000 | 209.729 | 65.587 | 1.637 | −5.770 | −1.393 | 0.753 | 0.789 | 0.545 | 0.171 | 0.000 | |
| $u$ | 0.000 | 115.470 | 149.670 | 129.797 | 69.256 | 32.989 | 30.303 | 33.201 | 27.256 | 14.515 | 0.000 | $10^7$ |
| $v$ | 0.000 | 294.796 | −14.001 | −0.395 | 0.004 | 0.176 | 0.357 | 0.109 | 0.423 | 0.006 | 0.000 | |
| $u$ | 0.000 | 248.453 | 266.354 | 149.301 | 86.413 | 96.987 | 97.311 | 74.817 | 47.606 | 23.062 | 0.000 | $10^8$ |
| $v$ | 0.000 | 7.726 | −0.973 | 1.560 | −0.418 | 0.392 | 0.154 | −0.015 | 0.186 | −0.041 | 0.000 | |

**Table I.**
Values of mid-plane velocity components $u$ and $v$

| Ra | $v_{max}$ | $y$ | $u_{max}$ | $x$ | $\overline{Nu}$ | $\psi_{mid}$ | References/discretization |
|---|---|---|---|---|---|---|---|
| $10^3$ | 3.679 | 0.179 | 3.634 | 0.813 | 1.116 | 1.174 | de Vahl Davis (1983) |
| | 3.686 | 0.188 | 3.489 | 0.813 | 1.117 | | Wan *et al.* (2001) |
| | 3.566 | | 3.544 | | | 1.165 | Šarler (2005) |
| | 3.991 | 0.170 | 3.931 | 0.825 | 1.101 | 1.298 | 41 × 41 |
| | 3.699 | 0.177 | 3.653 | 0.812 | 1.098 | 1.194 | 81 × 81 |
| | 3.695 | 0.179 | 3.645 | 0.820 | 1.089 | 1.196 | 101 × 101 |
| $10^4$ | 19.51 | 0.120 | 16.24 | 0.823 | 2.234 | 5.098 | de Vahl Davis (1983) |
| | 19.79 | 0.120 | 16.17 | 0.823 | 2.243 | | Wan *et al.* (2001) |
| | 19.04 | | 15.80 | | | 4.971 | Šarler (2005) |
| | 19.81 | 0.120 | 16.24 | 0.825 | 2.075 | 5.155 | 41 × 41 |
| | 19.83 | 0.120 | 16.27 | 0.825 | 2.120 | 5.167 | 81 × 81 |
| | 20.03 | 0.120 | 16.45 | 0.830 | 2.258 | 5.240 | 101 × 101 |
| $10^5$ | 68.22 | 0.066 | 34.81 | 0.855 | 4.510 | 9.142 | de Vahl Davis (1983) |
| | 68.52 | 0.064 | 34.63 | 0.852 | 4.534 | 9.092 | Sadat and Couturier (2000) |
| | 70.63 | 0.072 | 33.39 | 0.835 | 4.520 | | Wan *et al.* (2001) |
| | 67.59 | | 32.51 | | | 8.907 | Šarler (2005) |
| | 67.65 | 0.070 | 33.67 | 0.850 | 4.624 | 8.896 | 41 × 41 |
| | 68.98 | 0.062 | 34.60 | 0.850 | 4.813 | 9.135 | 81 × 81 |
| | 69.69 | 0.069 | 35.03 | 0.860 | 4.511 | 9.278 | 101 × 101 |
| 106 | 216.75 | 0.038 | 65.33 | 0.851 | 8.798 | 16.53 | de Vahl Davis (1983) |
| | 219.41 | 0.038 | 64.43 | 0.852 | 8.832 | 16.29 | Sadat and Couturier (2000) |
| | 227.11 | 0.04 | 65.4 | 0.86 | 8.8 | | Wan *et al.* (2001) |
| | 211.67 | | 61.55 | | | 15.91 | Šarler (2005) |
| | 195.98 | 0.045 | 63.73 | 0.850 | 6.1 | 15.15 | 41 × 41 |
| | 219.48 | 0.038 | 64.87 | 0.851 | 7.67 | 16.13 | 81 × 81 |
| | 221.37 | 0.039 | 65.91 | 0.860 | 8.97 | 16.51 | 101 × 101 |
| 107 | 687.43 | 0.023 | 145.68 | 0.888 | 16.59 | 28.23 | Sadat and Couturier (2000) |
| | 714.48 | 0.022 | 143.56 | 0.922 | 16.65 | | Wan *et al.* (2001) |
| | 632.60 | 0.020 | 127.70 | 0.925 | 10.43 | 24.93 | 41 × 41 |
| | 654.803 | 0.035 | 143.55 | 0.902 | 14.70 | 27.51 | 81 × 81 |
| | 687.20 | 0.021 | 149.61 | 0.900 | 16.92 | 28.61 | 101 × 101 |
| 108 | 2180.1 | 0.011 | 319.19 | 0.943 | 30.94 | 50.81 | Sadat and Couturier (2000) |
| | 2259.08 | 0.012 | 296.71 | 0.93 | 31.48 | | Wan *et al.* (2001) |
| | 2060.86 | 0.010 | 264.96 | 0.939 | 29.33 | 44.85 | 81 × 81 |
| | 2095.23 | 0.009 | 278.49 | 0.930 | 32.12 | 47.12 | 101 × 101 |

**Table II.**
Comparison of the present method with previously published results

takes the same order of CPU time ($t_{Pvi}$) as the adjacent time step calculation. The time spent for pressure correction can be estimated from $t_{Pvi} = t_c N_{Pvi}/(N_{Pvi} + N_t)$. Good agreement with other methods at a given range (Rayleigh number from $10^3$ to $10^8$ and grid density with maximum 101 × 101 grid points) is achieved with the proposed algorithm. Our method over-predicts reference results for low-Rayleigh numbers but for higher Rayleigh numbers it under-predicts reference results. For Rayleigh number $Ra = 10^3$ current method over-predicts all reference results. Similar behaviour is observed for results with $Ra = 10^4$, but for $Ra = 10^5$ and $Ra = 10^6$ our method over-predicts only two of the three reference results. Finally, for $Ra = 10^7$ and $Ra = 10^8$ current method under-predicts both comparison results. High deviance from results (Wan *et al.*, 2001) at a high Rayleigh number is due to grid selection. The present results are calculated on an entirely uniform grid. The results (Wan *et al.*, 2001) are calculated on Gauss-Lobatto grid, which is a more suitable selection due to the

| Ra | $\epsilon_v$ | 41 × 41 | | | 81 × 81 | | | | 101 × 101 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $\Delta\tau$ | $t_c$[s] | $N_{Pvi}$ | $N_t$ | $\Delta\tau$ | $t_c$[s] | $N_{Pvi}$ | $N_t$ | $\Delta\tau$ | $t_c$[s] | $N_{Pvi}$ | $N_t$ | $\Delta\rho/\rho$ |
| $10^3$ | $10e^{-4}$ | $1e^{-04}$ | 20 | 3,208 | 22,750 | $1e^{-04}$ | 190 | 26,837 | 22,750 | $5e^{-05}$ | 525 | 3662 | 45500 | $3.37e^{-007}$ |
| $10^4$ | $10e^{-3}$ | $1e^{-04}$ | 15 | 3,154 | 16,940 | $1e^{-04}$ | 92 | 3,259 | 16,940 | $5e^{-05}$ | 398 | 3706 | 33880 | $8.44e^{-006}$ |
| $10^5$ | $10e^{-2}$ | $1e^{-04}$ | 12 | 1,590 | 13,690 | $1e^{-04}$ | 70 | 1,244 | 13,690 | $5e^{-05}$ | 329 | 4090 | 27380 | $1.06e^{-005}$ |
| $10^6$ | 1 | $1e^{-04}$ | 11 | 5,527 | 10,200 | $1e^{-04}$ | 66 | 5,608 | 10,200 | $1e^{-05}$ | 1486 | 55043 | 102000 | $1.38e^{-004}$ |
| $10^7$ | 5 | $1e^{-05}$ | 19 | 18,250 | 12,500 | $1e^{-05}$ | 513 | 71,340 | 62,500 | $5e^{-06}$ | 2597 | 184697 | 125000 | $3.01e^{-004}$ |
| $10^8$ | 25 | | | | | $5e^{-06}$ | 1,225 | 193,708 | 135,000 | $5e^{-06}$ | 2938 | 219885 | 135000 | $5.90e^{-004}$ |

**Table III.**
Pressure-velocity iteration criteria $\epsilon_v$, time-step $\Delta\tau$, number of pressure-velocity correction iterations $N_{Pvi}$, number of time-steps $t_{max}$, actual CPU time consumption $t_c$ and global density loss $\Delta\rho/\rho$

highest velocities in the boundary layer (Figure 5). The effect is more intense with the higher Rayleigh numbers and so the deviance between the results grows with the higher Rayleigh numbers.

## 6. Conclusions

This paper explores the local RBFCM approach in solution of the coupled heat transfer and fluid flow problems by using the simplified entirely local pressure correction. The algorithm is very simple to implement numerically, fast and robust. The algorithm can be efficiently and straightforwardly parallelized because of its local subdomain nature and explicit time stepping. This two important features enables the exploiting of the full power of the multi core platforms. A remark should be made in the sense that there are possible difficulties when working with finer grids than in present paper. The proposed algorithm includes only few surrounding points to calculate pressure correction. The present pressure correction calculation (calculated only from closest neighbouring points) may not be efficient enough when working with finer grids ($200 \times 200$ or finer) and high-Rayleigh numbers (more than $5 \times 10^7$). Possible upgrade is to include wider domain of points in the pressure correction calculation. These topics are subject of further investigations, as well as focus on more complex geometries and more complex physical models (porous media, solidification, . . .), which seem to be quite simple to numerically implement in the present context. The investigation on the adaptive time dependent grid to enhance the accuracy and to avoid the eventual stability problems, the implementation of the characteristic-based-split algorithm (Massarotti *et al.*, 1998), the implementation of different sub-domain strategies, etc. all represents open issues, connected with the present method.

## References

Atluri, S.N. (2004), *The Meshless Method (MLPG) for Domain and BIE Discretization*, Tech Science Press, Encino, CA.

Atluri, S.N. and Shen, S. (2002a), *The Meshless Method*, Tech Science Press, Encino, CA.

Atluri, S.N. and Shen, S. (2002b), "The meshless local Petrov-Galerkin (MLPG) method: a simple & less-costly alternative to the finite element and boundary element methods", *Computer Modelling in Engineering & Sciences*, Vol. 3, pp. 11-52.

Buhmann, M.D. (2000), *Radial Basis Functions*, Cambridge University Press, Cambridge.

Chen, W. (2002), "New RBF collocation schemes and kernel RBFs with applications", *Lecture Notes in Computational Science and Engineering*, Vol. 26, pp. 75-86.

de Vahl Davis, G. (1983), "Natural convection of air in a square cavity: a bench mark numerical solution", *International Journal of Numerical Methods in Fluids*, Vol. 3, pp. 249-64.

Divo, E. and Kassab, A.J. (2007), "An efficient localized RBF meshless method for fluid flow and conjugate heat transfer", *ASME Journal of Heat Transfer*, Vol. 129, pp. 124-36.

Hortmann, M., Perić, M. and Scheuerer, G. (1990), "Finite volume multigrid prediction of laminar natural convection – bench-mark solutions", *International Journal for Numerical Methods in Fluids*, Vol. 11, pp. 189-207.

Kansa, E.J. (1990a), "Multiquadrics – a scattered data approximation scheme with application to computational fluid dynamics, Part I", *Computers and Mathematics with Applications*, Vol. 19, pp. 127-45.

Lin, H. and Atluri, S.N. (2001), "Meshless local Petrov Galerkin method (MLPG) for convection-diffusion problems", *Computer Modelling in Engineering & Sciences*, Vol. 1, pp. 45-60.

Liu, G.R. (2003), *Mesh Free Methods*, CRC Press, Boca Raton, FL.

Liu, G.R. and Gu, Y.T. (2005), *An Introduction to Meshfree Methods and Their Programming*, Springer, Dordrecht.

Manzari, M.T. (1999), "An explicit finite element algorithm for convection heat transfer problems", *International Journal of Numerical Methods for Heat and Fluid Flow*, Vol. 9, pp. 860-77.

Massarotti, N., Nithiarasu, P. and Zienkiewicz, O.C. (1998), "Characteristic-based-split (CBS) algorithm for incompressible flow problems with heat transfer", *International Journal of Numerical Methods for Heat and Fluid Flow*, Vol. 8, p. 969.

Sadat, H. and Couturier, S. (2000), "Performance and accuracy of a meshless method for laminar natural convection", *Numerical Heat Transfer, Part B*, Vol. 37.

Šarler, B. (2002), "Towards a mesh-free computation of transport phenomena", *Engineering Analysis with Boundary Elements*, Vol. 26, pp. 731-8.

Šarler, B. (2005), "A radial basis function collocation approach in computational fluid dynamics", *Computer Modelling in Engineering & Sciences*, Vol. 7, pp. 185-93.

Šarler, B. (2007), *From Global to Local Radial Basis Function Collocation Method for Transport Phenomena*, Springer, Berlin, pp. 257-82.

Šarler, B. and Kuhn, G. (1999), "Primitive variable dual reciprocity boundary element method solution of incompressible Navier-Stokes equations", *Engineering Analysis with Boundary Elements*, Vol. 23, pp. 443-55.

Šarler, B. and Vertnik, R. (2006), "Meshfree explicit local radial basis function collocation method for diffusion problems", *Computers and Mathematics with Applications*, Vol. 51, pp. 1269-82.

Šarler, B., Perko, J. and Chen, C.S. (2004), "Radial basis function collocation method solution of natural convection in porous media", *International Journal of Numerical Methods for Heat & Fluid Flow*, Vol. 14, pp. 187-212.

Vertnik, R. and Šarler, B. (2006), "Meshless local radial basis function collocation method for convective-diffusive solid-liquid phase change problems", *International Journal of Numerical Methods for Heat and Fluid Flow*, Vol. 16, pp. 617-40.

Vertnik, R., Založnik, M. and Šarler, B. (2006), "Solution of transient direct-chill aluminium billet casting problem with simultaneous material and interphase moving boundaries by a meshless method", *Engineering Analysis with Boundary Elements*, Vol. 30, pp. 847-55.

Wan, D.C., Patnaik, B.S.V. and Wei, G.W. (2001), "A new benchmark quality solution for the buoyancy-driven cavity by discrete singular convolution", *Numerical Heat Transfer, Part B*, Vol. 40, pp. 199-228.

## Appendix. The influence of the dimensionless shape parameter on the results

The convergence of the present RBF method was systematically studied in previous works for diffusive and convective diffusive problems (Šarler and Vertnik, 2006; Vertnik and Šarler, 2006), where it has been shown that the method converges.

The dimensionless shape parameter is set to 30 in all calculations in the preset paper. In order to justify this selection, a preliminary analysis has been done, described in this Appendix. For grid density $81 \times 81$, all cases have been recalculated with different selection of the shape parameter. In order to measure the influence of the shape parameter on the results, the following deviation has been introduced:

$$D = \frac{\max|\mathbf{v}|}{\max|\mathbf{v}_{120}|} \tag{A.1}$$

where max stands for maximum and $\mathbf{v}_{120}$ stands for velocity calculated with $c = 120$. From Figure A1, it is evident that $c = 30$ represents a reasonable choice for the shape parameter in all cases. The results are not sensitive to increase of the value of the shape parameter over 30, however, setting high-shape parameter produces ill-conditioned matrix in equation (18). The choice of shape parameter represents a compromise between accuracy and ill-condition of the matrices, respectively. This behaviour has been observed also in our previous works (Šarler and Vertnik, 2006; Vertnik and Šarler, 2006).

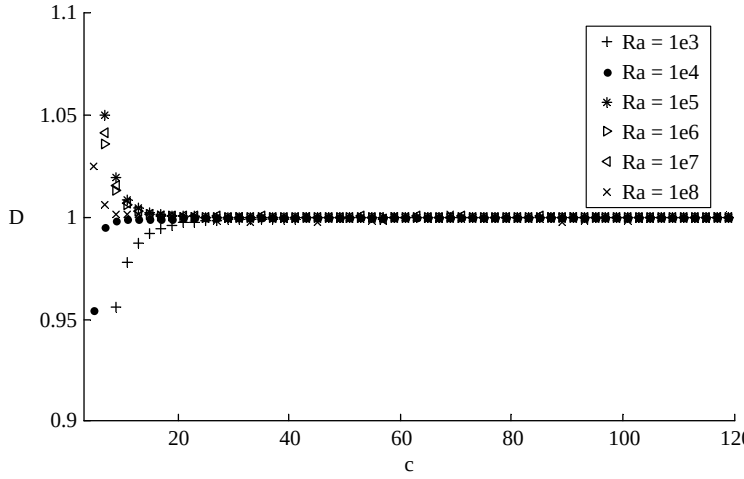


**Figure A1.**
Deviation of the final result with respect to the shape parameter

**Corresponding author**
Božidar Šarler can be contacted at: bozidar.sarler@p-ng.si