An Efficient Localized Radial Basis Function Meshless Method for Fluid Flow and Conjugate Heat Transfer

Eduardo Divo
Department of Engineering Technology,
University of Central Florida,
Orlando, FL 32816-2450

Alain J. Kassab
Department of Mechanical, Materials, and
Aerospace Engineering,
University of Central Florida,
Orlando, FL 32816-2450
e-mail: kassab@mail.ucf.edu

A localized radial basis function (RBF) meshless method is developed for coupled viscous fluid flow and convective heat transfer problems. The method is based on new localized radial-basis function (RBF) expansions using Hardy Multiquadrics for the sought-after unknowns. An efficient set of formulae are derived to compute the RBF interpolation in terms of vector products thus providing a substantial computational savings over traditional meshless methods. Moreover, the approach developed in this paper is applicable to explicit or implicit time marching schemes as well as steady-state iterative methods. We apply the method to viscous fluid flow and conjugate heat transfer (CHT) modeling. The incompressible Navier–Stokes are time marched using a Helmholtz potential decomposition for the velocity field. When CHT is considered, the same RBF expansion is used to solve the heat conduction problem in the solid regions enforcing temperature and heat flux continuity of the solid/fluid interfaces. The computation is accelerated by distributing the load over several processors via a domain decomposition along with an interface interpolation tailored to pass information through each of the domain interfaces to ensure conservation of field variables and derivatives. Numerical results are presented for several cases including channel flow, flow in a channel with a square step obstruction, and a jet flow into a square cavity. Results are compared with commercial computational fluid dynamics code predictions. The proposed localized meshless method approach is shown to produce accurate results while requiring a much-reduced effort in problem preparation in comparison to other traditional numerical methods. [DOI: 10.1115/1.2402181]

Introduction

Established numerical methods such as finite element methods (FEM) and finite volume methods (FVM) that are routinely used to resolve complex multiphysics interactions require significant effort in mesh generation. In fact, for most models of geometrically intricate components commonly encountered in engineering analysis, mesh generation is very time consuming, far from automated, and most often the most taxing part of the modeling process. The term meshless methods refers to a class of numerical techniques that rely on global interpolation on nonordered spatial point distributions that, as such, offer the hope of reducing the effort devoted to model preparation. These techniques have been under much development over the past few years [1–8]. The approach finds its origin in classical spectral or pseudo-spectral methods [9–17] that are based on global orthogonal functions such as Legendre or Chebyshev polynomials requiring a regular nodal point distribution. In contrast, meshless methods use a nodal or point distribution that is not required to be uniform or regular in their spatial distribution due to the fact that most rely on global radial-basis functions (RBF) [18–20]. These interpolating functions have proved quite successful in their application to an earlier mesh reduction method, namely the dual reciprocity boundary element method (DRBEM) [21–23]. However, global RBF based meshless methods have some drawbacks including poor conditioning of the ensuing algebraic set of equations which can be addressed to some extent by domain decomposition and appropriate pre-conditioning. Moreover, care must be taken in the evaluation of derivatives in global RBF meshless methods. Although, very promising, these techniques can also be computationally intensive. Recently, local collocation meshless method [24,25] have been proposed to address many of the issues posed by global RBF meshless methods.

In this paper, we develop a localized meshless collocation method using the Hardy Multiquadrics RBF [26] and derive new and efficient set of formulae to compute the RBF interpolation for the variable and various derivatives in terms of vector products. We show that these interpolating can be precomputed at the problem preparation stage. The formulation thus provides a substantial computational savings over traditional meshless methods, including the newer localized meshless methods. Moreover, the approach developed in this paper is flexible in that it is applicable to explicit or implicit time marching schemes as well as steady-state iterative methods.

We apply the localized meshless RBF (LMRBF) technique to the solution of the conjugate heat transfer problem (CHT). The term conjugate heat transfer refers to the coupled interaction of convective heat transfer within a fluid with conduction in the solid in which it is in contact (see Fig. 1). A variety of schemes have been proposed to numerically resolve CHT, including coupled finite volume/boundary element algorithms [27–31]. To accelerate the solution of such problems, we also adopt a parallel iterative domain decomposition as developed in the context of pure heat transfer application for the boundary element method [32] and for meshless methods in heat conduction [33–36]. Unlike global RBF meshless methods where domain decomposition is necessary to reduce the conditioning number of the resulting algebraic set, domain decomposition is used here for LMRBF only as a means of accelerating the time to solution by effectively distributing the computational burden to multiple processors. Numerical examples are presented to validate the approach by comparing the meshless

Contributed by the Heat Transfer Division of ASME for publication in the JOURNAL OF HEAT TRANSFER. Manuscript received September 8, 2005; final manuscript received May 25, 2006. Review conducted by: N. K. Anand. Paper presented at the 2005 ASME International Mechanical Engineering Congress (IMECE2005), November 5–11, 2005, Orlando, FL.

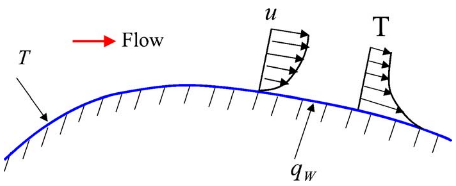


Fig. 1 CHT problem: external convective heat transfer coupled to heat conduction within the solid

solution to FVM solutions generated by a commercial computational fluid dynamics (CFD) solver (Fluent). We first review the solution algorithm for the Navier–Stokes equations, we then proceed to the localized RBF meshless collocation discretization, and we present numerical validation examples comparing predictions from our approach to those obtained by a commercial finite volume methods code.

Flow Solution by a Velocity Correction

Transient incompressible fluid flow is governed by the Navier–Stokes equations expressing the conservation of mass, linear momentum, and energy for a Newtonian fluid. In non-conservative form, these equations are

$$\nabla \cdot \mathbf{V} = 0 \tag{1a}$$

$$\rho \frac{\partial \mathbf{V}}{\partial t} + \rho(\mathbf{V}\cdot\nabla)\mathbf{V} = -\nabla p + \mu \nabla^2 \mathbf{V} + \rho \mathbf{f} \tag{1b}$$

$$\rho c \frac{\partial T}{\partial t} + \rho c(\mathbf{V}\cdot\nabla)T = k\nabla^2 T + \Phi \tag{1c}$$

Here, $\mathbf{V}$ is the flow velocity vector; ρ is the bulk density of the flow; p is the field pressure; μ is the absolute fluid viscosity; $\mathbf{f}$ is a specific body force; c is the fluid specific heat; T is the field temperature; k is the fluid thermal conductivity; and Φ is a nonlinear viscous dissipation term. All field variables are functions of space ($\boldsymbol{x}$) and time (t), in a fixed domain Ω surrounded by a closed boundary Γ. The explicit space–time dependency of each dependent variable has been omitted for simplicity of notation.

To arrive at an iterative solution algorithm for the fluid flow set of Eq. (1), a formulation must ensure coupled satisfaction of all the equations at convergence. Pressure correction or velocity correction schemes may be used to achieve this purpose in a time accurate manner or in a time-marching scheme to steady state where time stepping is then viewed and utilized as a relaxation procedure. We follow a velocity correction approach that was originally proposed by Harlow and Welch [37] in their MAC algorithm. The technique is closely related to alternative pressure-correction schemes [38]. We review the algorithm here for completeness.

The process starts from an initial velocity condition $\mathbf{V}^{(0)}$ that satisfies the continuity equation within the entire problem domain Ω, that is

$$\nabla \cdot \mathbf{V}^{(0)} = 0 \tag{2}$$

A new velocity field $\mathbf{V}^{(*)}$ may be estimated from the Navier–Stokes equations by positioning the space operators at the current time step as

$$\rho \frac{\partial \mathbf{V}^{(*)}}{\partial t} = \mu \nabla^2 \mathbf{V}^{(k)} - \nabla p^{(k)} - \rho(\mathbf{V}^{(k)}\cdot\nabla)\mathbf{V}^{(k)} \tag{3}$$

Notice that the body force term $\rho\mathbf{f}$ has been drop for simplicity. Equation (3) can also be recast in an implicit form by positioning the spatial operators at the new step (*) as

$$\mu \nabla^2 \mathbf{V}^{(*)} - \rho \frac{\partial \mathbf{V}^{(*)}}{\partial t} - \rho(\mathbf{V}^{(k)}\cdot\nabla)\mathbf{V}^{(*)} = \nabla p^{(k)} \tag{4}$$

Equations (3) or (4) can be advanced in time using a backward-difference approximation of the time derivative in a single step (first order) or multistep (second or higher order) scheme. The new velocity field $\mathbf{V}^{(*)}$ can then be determined by imposing a proper and complete set of boundary conditions generalized as

$$\alpha \frac{\partial \mathbf{V}^{(*)}}{\partial t} + \beta \frac{\partial \mathbf{V}^{(*)}}{\partial n} + \gamma \mathbf{V}^{(*)} = \sigma \tag{5}$$

As the solution of Eqs. (3) or (4) with the boundary conditions in Eq. (5) provides a velocity field that does not satisfy the continuity equation, that is

$$\nabla \cdot \mathbf{V}^{(*)} \neq 0 \tag{6}$$

Satisfaction of the continuity equation, may be accomplished by updating the velocity field $\mathbf{V}^{(*)}$ with a velocity variation field $\delta\mathbf{V}^{(k+1)}$ as

$$\mathbf{V}^{(k+1)} = \mathbf{V}^{(*)} + \delta\mathbf{V}^{(k+1)} \tag{7}$$

Furthermore, if the velocity variation field is required to be irrotational, that is

$$\nabla \times \delta\mathbf{V}^{(k+1)} = 0 \tag{8}$$

A Helmholtz potential (or velocity correction potential) $\phi^{(k+1)}$ can be defined as

$$\nabla \phi^{(k+1)} \equiv -\delta\mathbf{V}^{(k+1)} \tag{9}$$

Now, if the new velocity field is required to satisfy continuity, that is

$$\nabla \cdot \mathbf{V}^{(k+1)} = 0 \tag{10}$$

A relationship for the Helmholtz potential $\phi^{(k+1)}$ can be expressed in the form of a Poisson equation as

$$\nabla^2 \phi^{(k+1)} = (\nabla \cdot \mathbf{V}^{(*)}) \tag{11}$$

This Poisson equation for the Helmholtz potential can be solved by imposing a proper and complete set of homogeneous boundary conditions generalized as

$$\beta \frac{\partial \phi^{(k+1)}}{\partial n} + \gamma \phi^{(k+1)} = 0 \tag{12}$$

Once the Helmholtz Poisson problem is solved, the velocity field is updated and forced to satisfy continuity. An equation to update the pressure field can be easily obtained by taking the divergence of the momentum equation and applying the continuity equation where necessary to arrive at

$$\nabla^2 p^{(k+1)} = -\rho \nabla \cdot [(\mathbf{V}^{(k+1)}\cdot\nabla)\mathbf{V}^{(k+1)}] \tag{13}$$

This Poisson equation for the pressure field can be solved by imposing a proper and complete set of boundary conditions some of which can be derived directly from the Navier–Stokes equation and generalized as

$$\beta \frac{\partial p^{(k+1)}}{\partial n} + \gamma p^{(k+1)} = \sigma \tag{14}$$

With the updated velocity field, the temperature field can be solved from Eq. (1c) using an explicit scheme such as

$$\rho c \frac{\partial T^{(k+1)}}{\partial t} = k\nabla^2 T^{(k)} - \rho c(\mathbf{V}^{(k+1)}\cdot\nabla)T^{(k)} \tag{15}$$

or an implicit scheme as

$$k\nabla^2 T^{(k+1)} - \rho c \frac{\partial T^{(k+1)}}{\partial t} - \rho c(\mathbf{V}^{(k+1)}\cdot\nabla)T^{(k+1)} = 0 \tag{16}$$

Notice that in both cases the viscous dissipation term Φ has been dropped for simplicity. Equation (16) above is a linear differential equation that can be time-march using a first- or higher-order scheme starting from an initial temperature field $T^{(0)}$ by imposing a proper and complete set of boundary conditions generalized as

$$\alpha\frac{\partial T^{(k+1)}}{\partial t}+\beta\frac{\partial T^{(k+1)}}{\partial n}+\gamma T^{(k+1)}=\sigma \tag{17}$$

Typical boundary conditions imposed to solve these equations are provided in the following list:

A. Boundary conditions for $\mathbf{V}$:

1. Wall (Dirichlet)

$$\mathbf{V}=0$$

2. Inlet (Dirichlet)

$$\mathbf{V}=\hat{V}$$

3. Outlet (nonreflective advective conditions) [39]

$$\frac{\partial \mathbf{V}}{\partial t}+V_n\frac{\partial \mathbf{V}}{\partial n}=0$$

B. Boundary conditions for p:

1. Wall (Neumann from N-S equation)

$$\frac{\partial p}{\partial n}=\left[\mu(\nabla^2\mathbf{V})-\rho\frac{\partial \mathbf{V}}{\partial t}\right]\cdot\hat{n}$$

2. Inlet (Dirichlet)

$$p=\hat{p}$$

3. Outlet (Neumann from N-S equation)

$$\frac{\partial p}{\partial n}=\left[\mu(\nabla^2\mathbf{V})-\rho\frac{\partial \mathbf{V}}{\partial t}-\rho(\mathbf{V}\cdot\nabla)V\right]\cdot\hat{n}$$

C. Boundary conditions for ϕ:

1. Wall (Neumann)

$$\frac{\partial \phi}{\partial n}=0$$

2. Inlet (Neumann)

$$\frac{\partial \phi}{\partial n}=0$$

3. Outlet (Dirichlet)

$$\phi=0$$

D. Boundary conditions for T:

1. Forced

$$T=\hat{T}$$

2. Natural

$$k\frac{\partial T}{\partial n}=-\hat{q}$$

3. Convective

$$k\frac{\partial T}{\partial n}=-\hat{h}(T-T_{\text{ref}})$$

where the notation $\hat{V}$, $\hat{p}$, $\hat{T}$, $\hat{q}$, and $\hat{h}$ signifies prescribed values; V_n is the normal velocity; and $\partial/\partial n$ denotes derivative with respect to the outward-drawn normal to a boundary, $\hat{n}$. We use the following fractional step method to time march to steady state

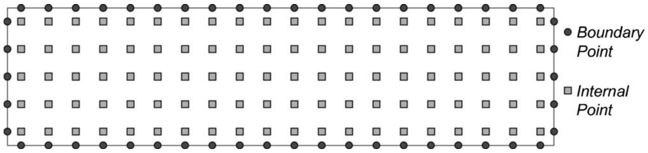


Fig. 2 Typical point collocation of data centers

$$f\left(t+\frac{\Delta t}{3}\right)=f(t)+\frac{\Delta t}{3}\left.\frac{\partial f}{\partial t}\right|_{t}$$
$$f\left(t+\frac{\Delta t}{2}\right)=f(t)+\frac{\Delta t}{2}\left.\frac{\partial f}{\partial t}\right|_{t+(\Delta t/3)}$$
$$f(t+\Delta t)=f(t)+\Delta t\left.\frac{\partial f}{\partial t}\right|_{t+(\Delta t/2)} \tag{18}$$

This method can be shown to be third-order accurate and when applied to the momentum equations leads to

$$\mathbf{V}^{(k+1/3)}=\mathbf{V}^{(k)}+\frac{\Delta t}{3}\left[\frac{\mu}{\rho}\nabla^2\mathbf{V}-(\mathbf{V}\cdot\nabla)\mathbf{V}\right]^{(k)}-\frac{\Delta t}{3\rho}\nabla p^{(k)}$$
$$\mathbf{V}^{(k+1/2)}=\mathbf{V}^{(k)}+\frac{\Delta t}{2}\left[\frac{\mu}{\rho}\nabla^2\mathbf{V}-(\mathbf{V}\cdot\nabla)\mathbf{V}\right]^{(k+1/3)}-\frac{\Delta t}{2\rho}\nabla p^{(k)}$$
$$\mathbf{V}^{(k+1)}=\mathbf{V}^{(k)}+\Delta t\left[\frac{\mu}{\rho}\nabla^2\mathbf{V}-(\mathbf{V}\cdot\nabla)\mathbf{V}\right]^{(k+1/2)}-\frac{\Delta t}{\rho}\nabla p^{(k)} \tag{19}$$

It is noted that the last update at $(k+1)$ is performed sequentially, with $\mathbf{V}^{(*)}$ computed first without the pressure gradient, and then $\mathbf{V}^{(k+1)}$ is updated with the velocity correction. These equations are iterated at each sublevel for time-accurate solutions or stepped once through if steady state is of interest. The pressure equation is subsequently updated by solving the Poisson equation and the energy equation is time-marched as well.

The use of the Helmholtz decomposition approach over the standard pressure-correction scheme [38] offers stability to the method by directly solving for the pressure field at every iteration step rather than correcting it. On the other hand, the computational efficiency of this approach is affected by the fact that two, instead of one, Poisson equations are solved at every step. This completes the review of the fluid flow solving algorithm, we now develop the localized RBF meshless collocation method.

Discretization in Space Using a Localized RBF Collocation

The meshless formulation begins by defining a set of data centers, NC, comprised of points on the boundary, NB, and points on the interior, NI. These data centers will serve as collocation points for the localized expansion of the different field variables in the domain Ω and on the boundary Γ (see Fig. 2).

A global interpolation for the field variables, $\phi(\boldsymbol{x})$ for instance, can be formulated by requiring

$$\phi(\boldsymbol{x})=\sum_{j=1}^{NB+NI}\alpha_j\chi_j(\boldsymbol{x}) \tag{20}$$

The expansion functions may be written in terms of radial-basis interpolation functions (RBF), in particular from the family of the so-called Hardy Multiquadrics [26] as

$$\chi_j(\boldsymbol{x})=[r_j^2(\boldsymbol{x})+c^2]^{n-(3/2)} \tag{21}$$

where n is a positive exponent parameter; c is a shape parameter; and $r_j(\boldsymbol{x})$ is the Euclidean distance from $(\boldsymbol{x})$ to $(\boldsymbol{x}_j)$. Several authors have shown [11,33–36] that a global interpolation of this type leads to large, fully populated, ill-conditioned coefficient matrices for the unknown expansion coefficients. This behavior is exhibited even for small scale problems with a uniform distribution of data centers, and it affects the accuracy of the interpolation of the field variable, the accuracy of the approximation of the derivatives, and hence, the stability of the solution method. Several attempts have been made to mitigate this problem including preconditioning, domain decomposition [8], finite difference approximation of derivatives [36], and moving least-squares, among others.

A promising approach was recently proposed by Sarler et al. [24] and Sarler and Vertnik [25] to reduce the burden of the global interpolation by expanding the field variable locally around each data point to obtain its derivatives in an explicit time-marching scheme for the field variables. This approach yields the generation of a different but small interpolation matrix for each data point rather than the large and fully populated global interpolation matrix of the standard global RBF. This notion gives more leverage in the selection and distribution of points as well as in the choice of the free parameters for the RBF expansion functions. Moreover, since the approach consists of expanding known values of the field variables, it is applicable as long as an explicit time-marching scheme is formulated and inapplicable directly to steady problems.

A modification to the local formulation is now presented for its implementation to a general class of problems including the steady, Poisson-like problem encountered in the iterative solution of the Navier–Stokes equations of the previous sections, as well as explicit and implicit time marching schemes. For the development of the notions underlying the method, let us consider first the Poisson equation, Eq. (13), for the pressure update. This equation is recast in the following form

$$\nabla^2 p(\boldsymbol{x}) = f(\boldsymbol{x}) \tag{22}$$

and imposed with general boundary conditions given by

$$\beta(\boldsymbol{x})\frac{\partial p(\boldsymbol{x})}{\partial n} + \gamma(\boldsymbol{x})p(\boldsymbol{x}) = \sigma(\boldsymbol{x}) \tag{23}$$

A localized expansion based on the RBF multiquadric interpolation functions may be written as

$$p(\boldsymbol{x}) \approx \sum_{j=1}^{NF} \alpha_j \chi_j(\boldsymbol{x}) \tag{24}$$

where NF is the number of points of influence around and including the data center $\boldsymbol{x_i}$. Figure 3 shows typical collocation topologies for internal, boundary, and corner data centers. In this case, all of the topologies employ five points including the data center. However, this format can be modified to include, for example, the diagonal points. In addition, in order to form the connectivity of these topologies throughout the entire domain, a smart search algorithm must be formulated to account for irregularities in the geometry and/or point distribution.

The solution of the field variable p is initialized throughout the entire domain either with the initial condition, an initial guess, or, in this case, with the pressure field from the previous time step. The pressure at the data center of the collocation topology is constrained with the governing equation as

$$\nabla^2 p(\boldsymbol{x_i}) = \sum_{j=1}^{NF} \alpha_j \nabla^2 \chi_j(\boldsymbol{x_i}) = f(\boldsymbol{x_i}) \tag{25}$$

in the case where $\boldsymbol{x_i}$ is an internal data center, while

$$\sum_{j=1}^{NF} \alpha_j \left[\beta(\boldsymbol{x_i}) \frac{l\chi_j}{\partial n}(\boldsymbol{x_i}) + \gamma(\boldsymbol{x_i})\chi_j(\boldsymbol{x_i}) \right] = \sigma(\boldsymbol{x_i}) \tag{26}$$

in the case where $\boldsymbol{x_i}$ is a boundary data center. The initial or current pressure field is imposed on the remaining points of the topology as

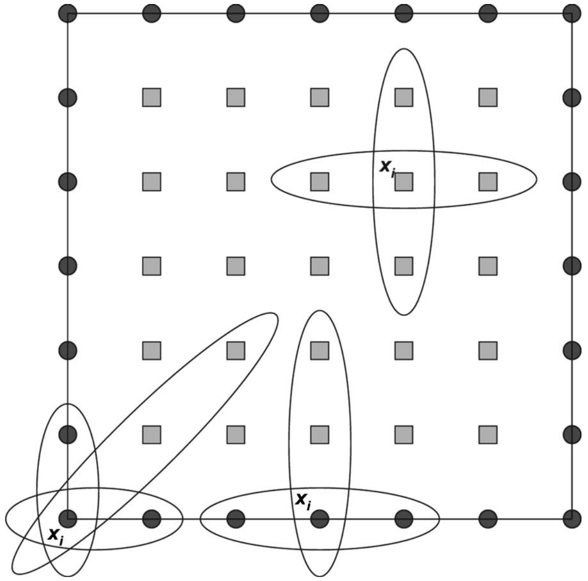


Fig. 3 Collocation topology for internal, boundary, and corner data centers

$$\sum_{j=1}^{NF} \alpha_j \chi_j(\boldsymbol{x_l}) = \hat{p}(\boldsymbol{x_l}) \tag{27}$$

where $l=2\ldots NF$. Equations (25) and (27) form a complete set of equations for the coefficients α at internal data centers while Eqs. (26) and (27) form a complete set of equations for the coefficients α at boundary data centers. This system of equations have the following form for the case of internal data centers

$$\begin{bmatrix} \nabla^2\chi_1(\boldsymbol{x_1}) & \ldots & \nabla^2\chi_{NF}(\boldsymbol{x_1}) \\ \chi_1(\boldsymbol{x_2}) & \ldots & \chi_{NF}(\boldsymbol{x_2}) \\ \vdots & \vdots & \vdots \\ \chi_1(\boldsymbol{x_{NF}}) & \ldots & \chi_{NF}(\boldsymbol{x_{NF}}) \end{bmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{\boldsymbol{NF}} \end{pmatrix} = \begin{pmatrix} f(\boldsymbol{x_1}) \\ \hat{p}(\boldsymbol{x_2}) \\ \vdots \\ \hat{p}(\boldsymbol{x_{NF}}) \end{pmatrix}$$

or in matrix-vector form

$$[C]\{\alpha\} = \{b\} \quad \text{or} \quad \{\alpha\} = [C]^{-1}\{b\} \tag{28}$$

A similar form can be obtained for boundary data centers. Combining the above with the localized expansion of the pressure p at the data center $\boldsymbol{x_i}$ yields

$$p_i = \{\chi_i\}^T\{\alpha\} = \{\chi_i\}^T[C]^{-1}\{b\} = \{g\}^T\{b\} \tag{29}$$

Therefore, evaluation of the pressure at the data center $\boldsymbol{x_i}$ is provided by a simple inner product of two vectors:$\{g\}$ which can be prebuilt and stored for every data center from the collocation matrix $[C]$ and the expansion functions $\chi_j(\boldsymbol{x_i})$ and $\{b\}$ which changes as the pressure field and/or right-hand side of the governing equation changes. Furthermore, the new pressure field may be under-relaxed with a parameter θ as

$$p^{(u+1)} = \theta p^{(\prime)} + (1-\theta)p^{(u)} \tag{30}$$

where (u) is the update step of the solution and $(\,')$ stands for the estimated field from the localized meshless expansion. Once the pressure has been updated throughout the field, a direct interpolation with the same topology may be performed from the expression in Eq. (24) as

$$\{p\} = [\chi]\{\alpha\} \Rightarrow \{\alpha\} = [\chi]^{-1}\{p\} \tag{31}$$

Then, to estimate the field variable derivatives at the data center, any linear differential operator $\mathcal{L}$ can be applied over the localized expansion equation (24) as

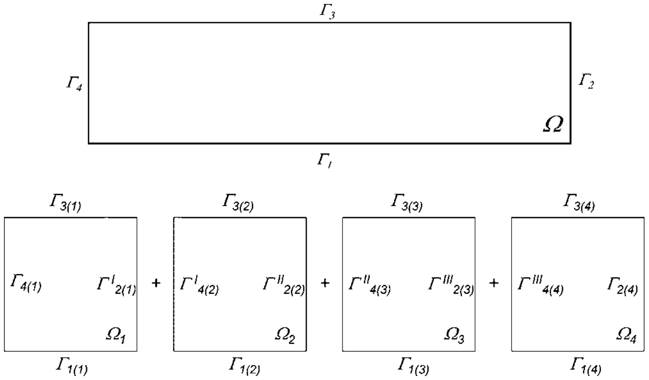


Fig. 4 Problem domain and typical decomposition

$$\mathcal{L}p(\boldsymbol{x}_i) \approx \sum_{j=1}^{NF} \alpha_j \mathcal{L}\chi_j(\boldsymbol{x}_i) \tag{32}$$

Thus, in matrix-vector form

$$\mathcal{L}p_i = \{\mathcal{L}\chi_i\}^T\{\alpha\} = \{\mathcal{L}\chi_i\}^T[\chi]^{-1}\{p\} = \{\mathcal{L}\}^T\{p\} \tag{33}$$

Therefore, evaluation of the field variable derivatives at everyone of the data centers $\boldsymbol{x}_i$ is again provided by a simple inner product of two vectors: $\{\mathcal{L}\}$ which can be prebuilt and stored and $\{p\}$ which is the updated pressure field within the topology of the data center $\boldsymbol{x}_i$. Notice that for the topologies considered herein, the length of the vectors $\{g\}$ and $\{\mathcal{L}\}$ is 5×1.

The particular feature of the localized collocation method that allows us to estimate the field variables and their derivatives by simple inner products of vectors that can be prebuilt and stored is precisely what makes this approach attractive. Notice that the Multiquadrics functions need only be evaluated at a setup stage when these vectors are being built reducing the CPU burden of having to evaluate fractional powers and complicated functions at every step of an iteration or time-marching scheme. In addition, the memory demands of this approach are minimal, as no global collocation matrix must be allocated [33], and, only very small vectors must be stored for every one of the data centers. This offers tremendous advantages in terms of data preparation over global RBF meshless methods. For example, to set up a 20,000 data point localized meshless methods problem, including establishing the topology and prebuilding all collocation vectors such as $\{g\}$ and $\{\mathcal{L}\}$, and storage of these onto disk takes less than 3 s on a 2.8 GHz P4 desktop computer with 1 GB of RAM. Thus, in addition to improved control and stability, the efficiency of the localized RBF method described in this paper is far superior to the global meshless methods reported in the literature. In the case of transient problems with moving boundaries this advantage is more pronounced in the sense that only affected topologies close to the boundary need to be recomputed as opposed to global meshless methods where the entire collocation matrix must be recomputed and stored.

Parallel Domain Decomposition Implementation

In order to further accelerate the solution process, a parallelization of the time-marching numerical algorithm is formulated and readily implemented in a parallel cluster environment. It is noted that unlike the necessity to adopt domain decomposition to reduce the conditioning number in global RBF approaches, parallel domain decomposition serves here only to reduce computing time by distributing computation to multiple processors.

The problem domain Ω is decomposed into multiple subdomains Ω_m separated by artificially created interfaces Γ^{I} (see [33–35] and Fig. 4). Each subdomain problem is rendered as a well-posed boundary value problem by imposing boundary conditions (starting from the initial conditions) at the artificially created interfaces. A simple iteration implementation can follow to force continuity conditions of the field variables and their derivatives at the interfaces. The boundary value problem will then be independently solved over each subdomain.

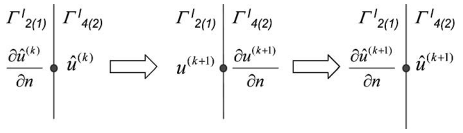


Fig. 5 Iteration process averaging across an interface

Once each boundary value problem is independently solved on each subdomain, an iteration scheme must follow to continuously ensure continuity of the field variables and their derivatives across the interfaces. This is accomplished by setting up a process in which each interface is treated differently on both sides: on one side the first kind of boundary condition is applied, while on the other side the second kind of boundary condition is applied. The resulting boundary values are then averaged with their corresponding values across the interfaces. This process is illustrated in Fig. 5 for a single point on the interface between subdomain Ω_1 and Ω_2 for the x component of the velocity field, u. In this case, the combination of boundary conditions at artificially created interfaces must be such that it ensures that each subdomain has at least one inlet (first kind) and one outlet (second kind) to satisfy physical requirements.

In Fig. 5 the imposed boundary conditions are $\partial\hat{u}^{(k)}/\partial n$ on the left-hand side of the interface and $\hat{u}^{(k)}$ on the right-hand side. After the solution of the boundary value problem on each subdomain, the resulting (provisional) values are $u^{(k+1)}$ on the left-hand side of the interface and $\partial u^{(k+1)}/\partial n$ on the right-hand side. These resulting values are averaged to generate the boundary conditions for the next time step as

$$\frac{\partial\hat{u}^{(k+1)}}{\partial n} = \frac{\partial\hat{u}^{(k)}}{\partial n} - \frac{1}{2}\left(\frac{\partial\hat{u}^{(k)}}{\partial n} + \frac{\partial u^{(k+1)}}{\partial n}\right)$$

$$\hat{u}^{(k+1)} = \frac{\hat{u}^{(k)} + u^{(k+1)}}{2} \tag{34}$$

ensuring continuity of the field variable and its normal derivative. Notice that in order to achieve truly time-accurate analysis, sublevel iterations must be performed at each time step. The iteration process may be stopped once steady state is achieved or at any time level. Steady state may be verified by defining a least-squares residual of the deviation from the satisfaction of the field equations and comparing it to that of the previous time step (k). Once the residual difference reaches a preset convergence criterion the iteration process may be stopped.

The domain decomposition formulation is implemented in a parallel computing environment: the algorithm has been coded and implemented in a 36-node 72 CPU 3.06–3.2 GHz Xeon Dell PowerEdge Cluster running Red Hat Linux 9.0 and the MPICH implementation of the MPI standards, and compiled using GNU FORTRAN 77. The parallel code collapses to serial computation if a single processor is assigned to the execution.

Upon launching the code under MPI, the processors are identified and given a rank. A small sample problem is solved on all processors to identify their relative performance. A load balancing routine is then performed to optimally assign domains to each processor by minimizing an objective function that contains information with regard to subdomain sizes and relative computational capability. This optimization is performed using a discrete genetic algorithm [32]. This completes the development of our numerical algorithm. We now proceed to present results from several numerical verification examples.

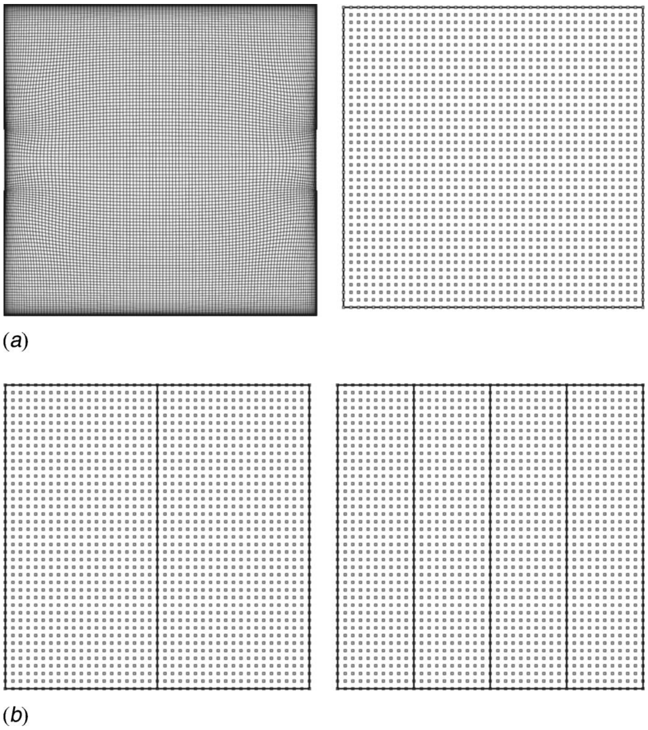


Fig. 6 FVM mesh and point collocation of cavity

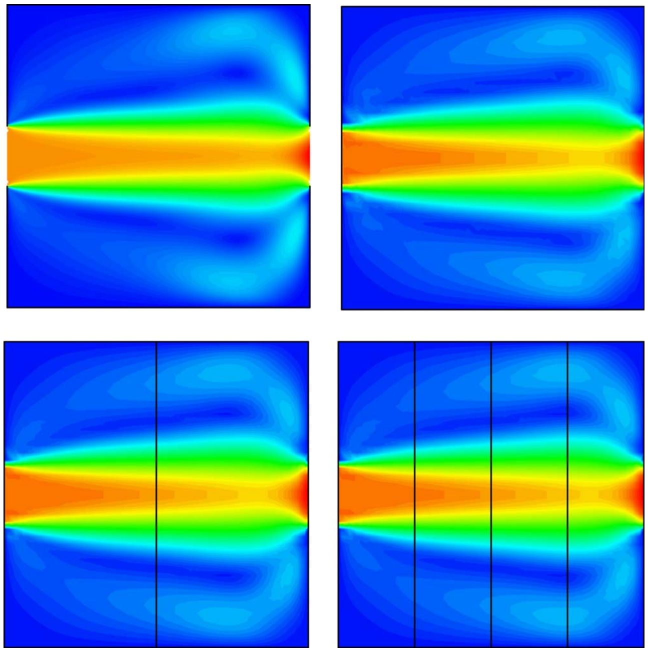


Fig. 7 FVM and meshless velocity contours

Numerical Verification

In all cases, we compare results predicted by our localized meshless RBF methods with those obtained by the commercial FVM computational fluid dynamics package Fluent 6.1. All cases considered here are laminar, and all FVM solutions are grid converged and carried out using the second-order upwinding scheme, unless otherwise stated. The efficiency of the localized meshless method is validated with these examples as the measured computational time per iteration is proportional to the ratio of data centers to FVM grid points for every case. Thus, with a much coarser meshless point distribution with respect to FVM grids, a significant improvement in computational time is obtained.

The first numerical test of the localized meshless method consists of a 1×1 m square cavity with a 20 cm air inlet at 0.01 m/s centered on the left-hand wall and a 20 cm outlet centered on the right-hand wall. The Prandtl number is taken as $\mathrm{Pr}=0.7$ and the Reynolds number based on the cavity side length is $\mathrm{Re}_L=675$. Here, the method will be tested for robustness by sequentially domain decomposing into one, two, and four regions. The meshless solution is provided by collocating 1681 (41×41) equally spaced data centers, while the FVM mesh contains 13,685 nodes as seen in Fig. 6. The velocity contours for the FVM solution as well as for the single and multiregion meshless solution is shown in Fig. 7 revealing accurate results.

The velocity profiles at $x=0$, 0.25, 0.5, 0.75, and 1.0 m are shown in Fig. 8 again revealing the accuracy of the localized meshless approach.

Next, the lid-driven square cavity problem is used as a benchmark to validate both the accuracy and efficiency of the localized meshless approach when compared to FVM solutions and results reported by Ghia et al. [40]. A square closed cavity is considered with the top wall moving at a constant speed. In addition, the bottom wall is kept at 323 K while the top wall is kept at 288 K.

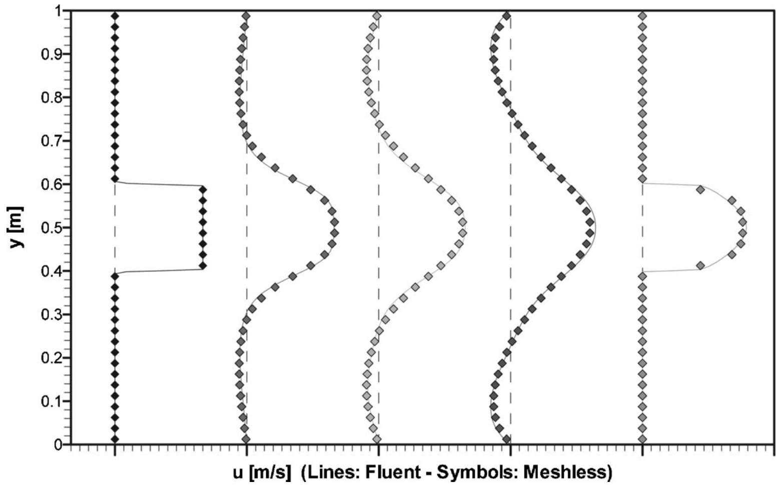


Fig. 8 FVM and meshless x velocity profiles at $x=0$, 0.25, 0.5, 0.75, and 1.0 m

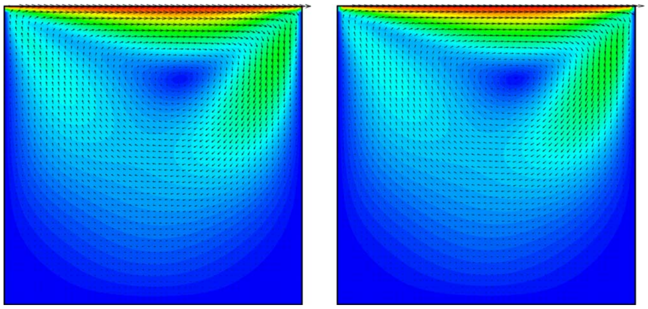


Fig. 9 FVM and meshless velocity vectors and magnitude contours for the lid-driven cavity

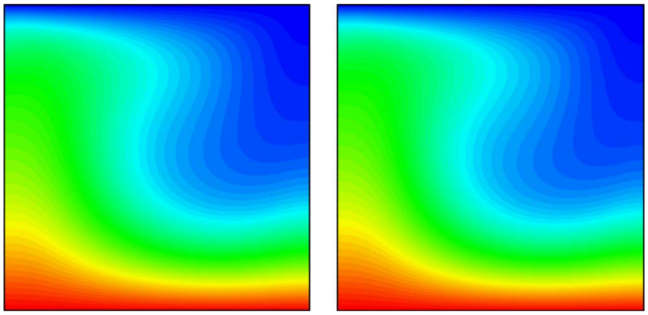


Fig. 10 FVM and meshless temperature contours for the lid-driven cavity

Table 1 Computation times for FVM and localized meshless solutions for different grid resolutions.

	FVM	Localized meshless		
Grid/Point distribution	Fluent 6.1 (1000 Δt)	Topology	RBF collocation	Solution (1000 Δt)
26×26	264 s	<1 s	<1 s	5.19 s
51×51	326 s	<2 s	<1 s	27.86 s
101×101	970 s	<8 s	<3 s	164.14 s

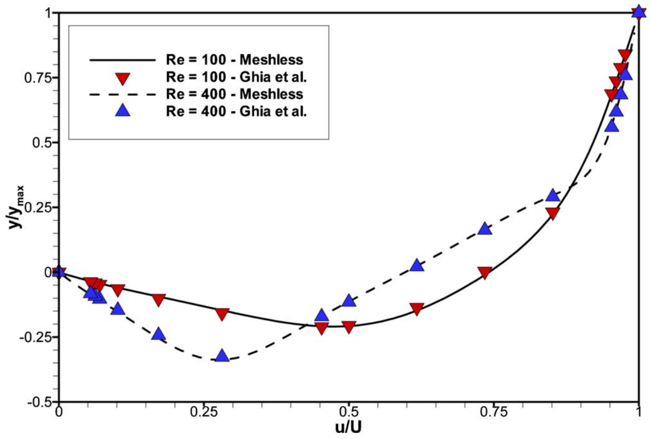


Fig. 11 FVM and meshless *x* velocity component at vertical center line

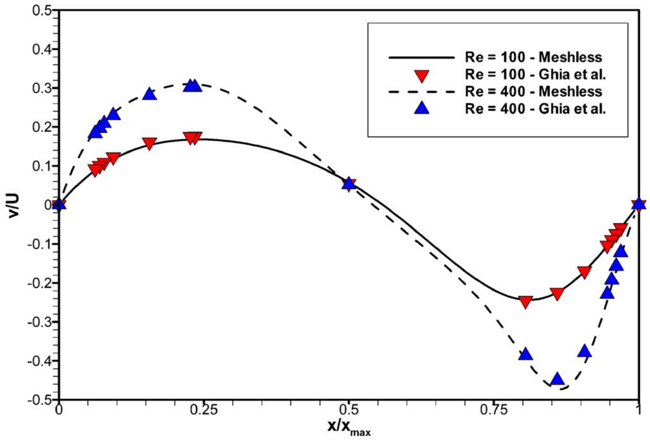


Fig. 12 FVM and meshless *y* velocity component at horizontal center line

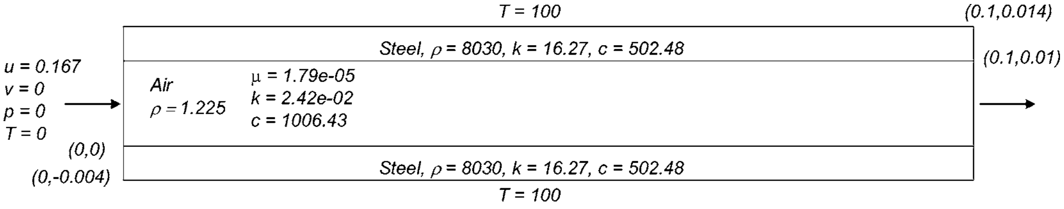


Fig. 13 Geometry and properties for parallel plates

Initially, FVM and meshless solutions are generated using three uniform point distributions $(26\times26, 51\times51, 101\times101)$ with a Reynolds number of 68 based on the side of the cavity and the Prandtl number is $\mathrm{Pr}=0.7$. Qualitative comparison of the results are shown in Figs. 9 and 10 where FVM and meshless velocity vectors and magnitude contours, as well as temperature contours are displayed for the 51×51 grid and point distribution case. A measure of the efficiency of the localized meshless method com-

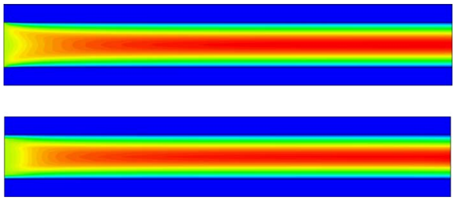


Fig. 14 FVM and meshless velocity contours

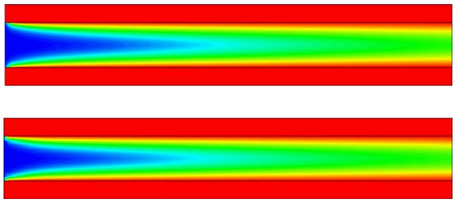


Fig. 15 FVM and meshless temperature contours

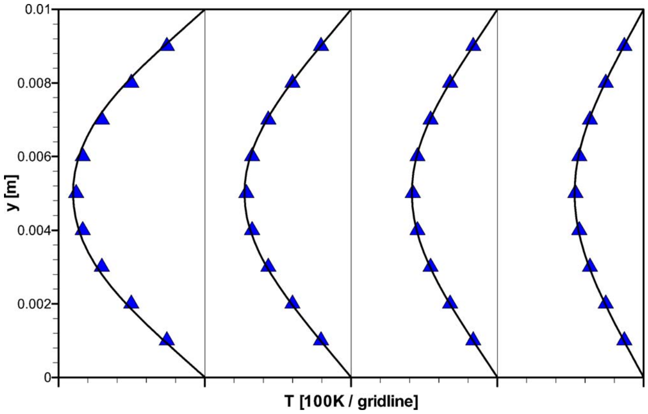


Fig. 16 FVM and meshless temperature profiles at 1/4, 1/2, 3/4, and 1/1 channel length

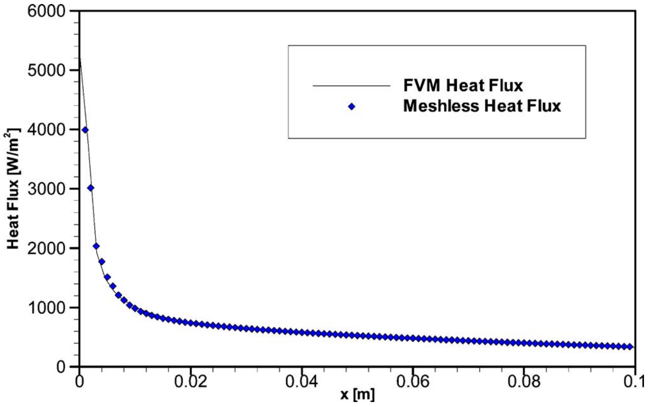


Fig. 17 FVM and meshless heat flux along interface

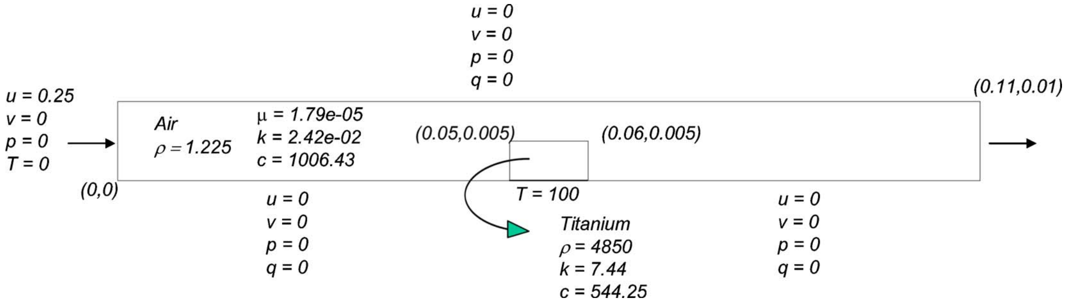
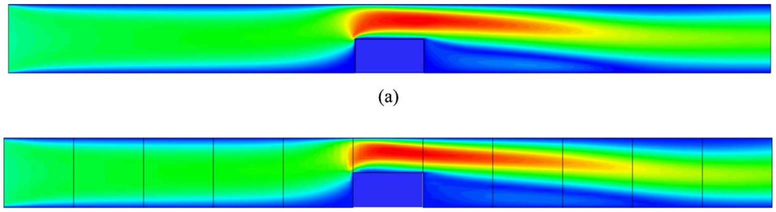


Fig. 18 Geometry and properties for square obstruction



(a)

(b)

Fig. 19 Velocity contours: (*a*) FVM, and (*b*) meshless

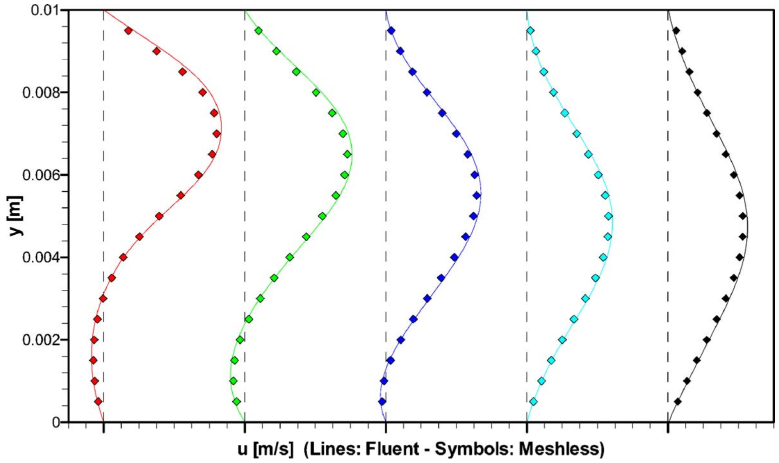


Fig. 20 FVM and meshless *x* velocity profiles at *x*=7, 8, 9, 10, and 11 cm

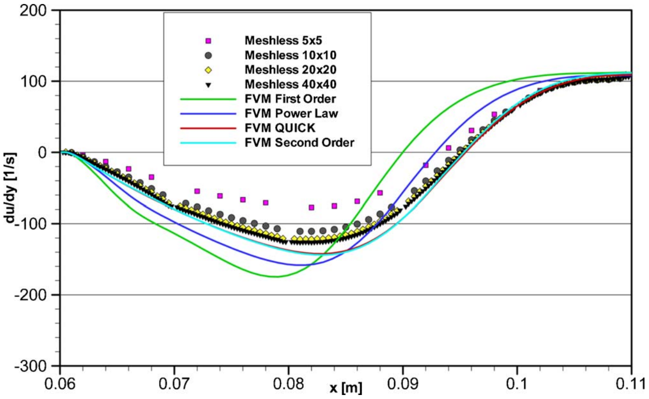


Fig. 21 FVM and meshless $\partial u/\partial y$ after the obstruction

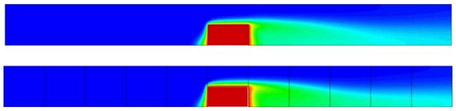


Fig. 22 FVM and meshless temperature contours

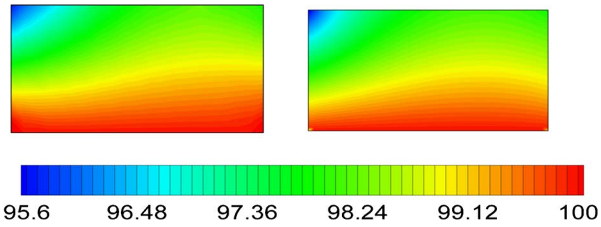


Fig. 23 Zoomed in FVM and meshless obstruction temperature contours in the square obstruction

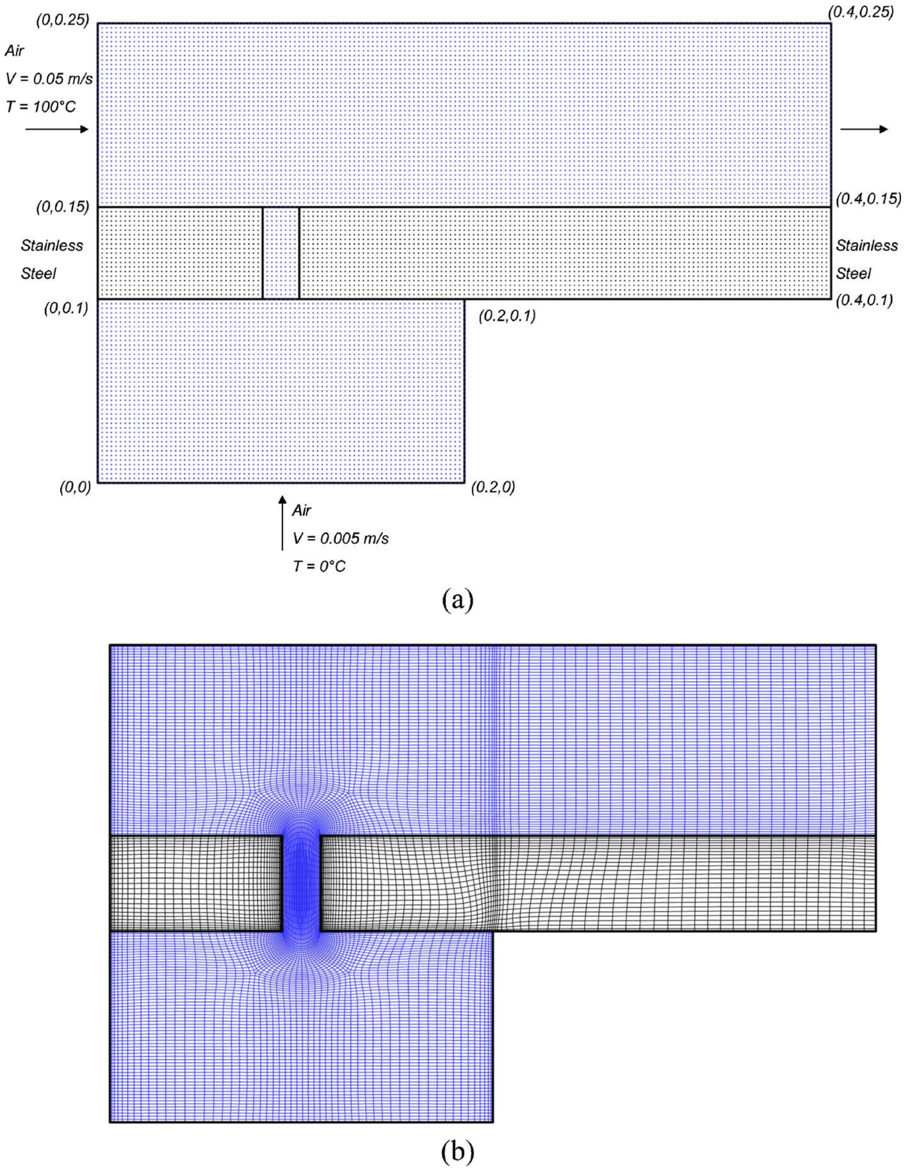


Fig. 24 Data center distribution and FVM mesh for cooling plenum, cooling hole (2 cm width), and main cooling channel:(*a*) meshless collocation (13,345 data centers) 2 cm cooling hole width and (*b*) finite volume mesh (14,859 finite volumes)

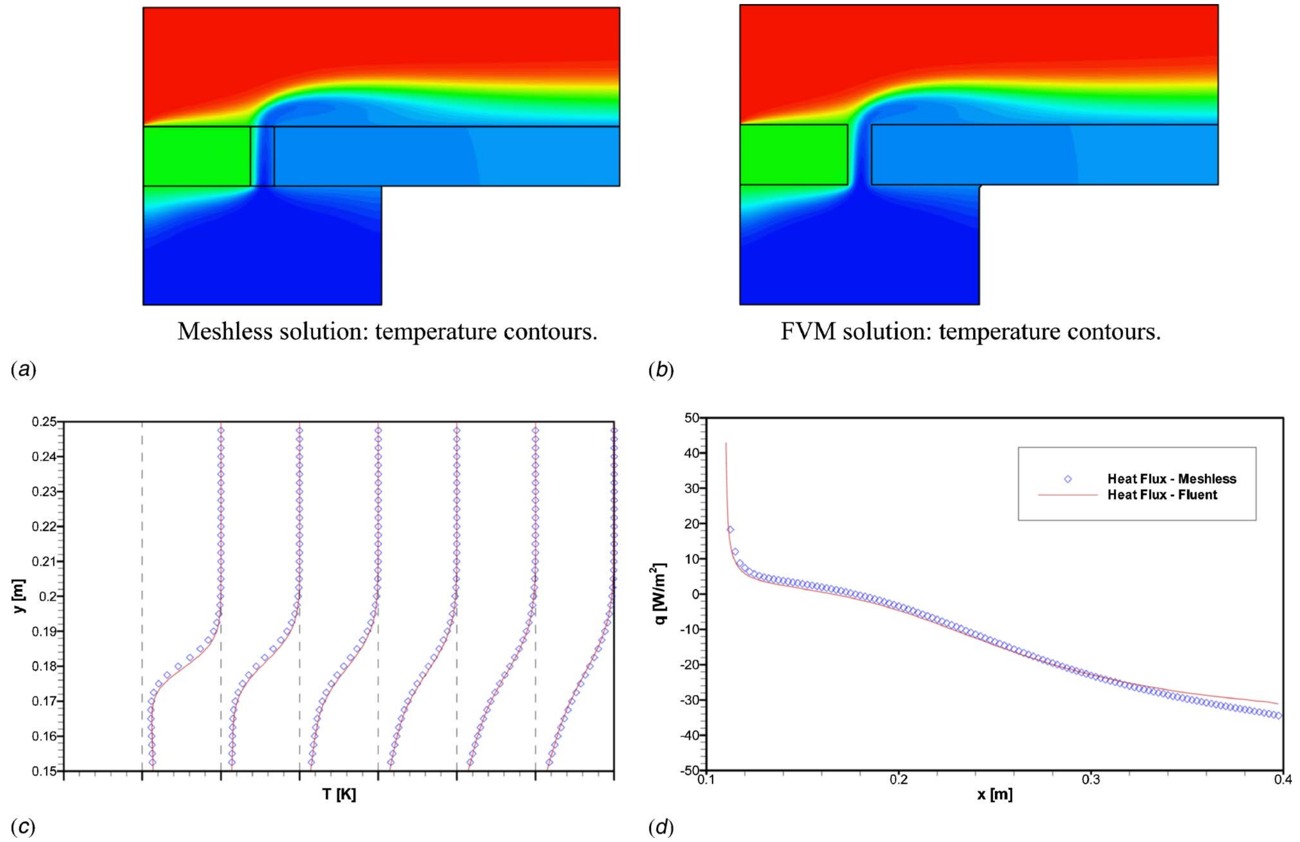


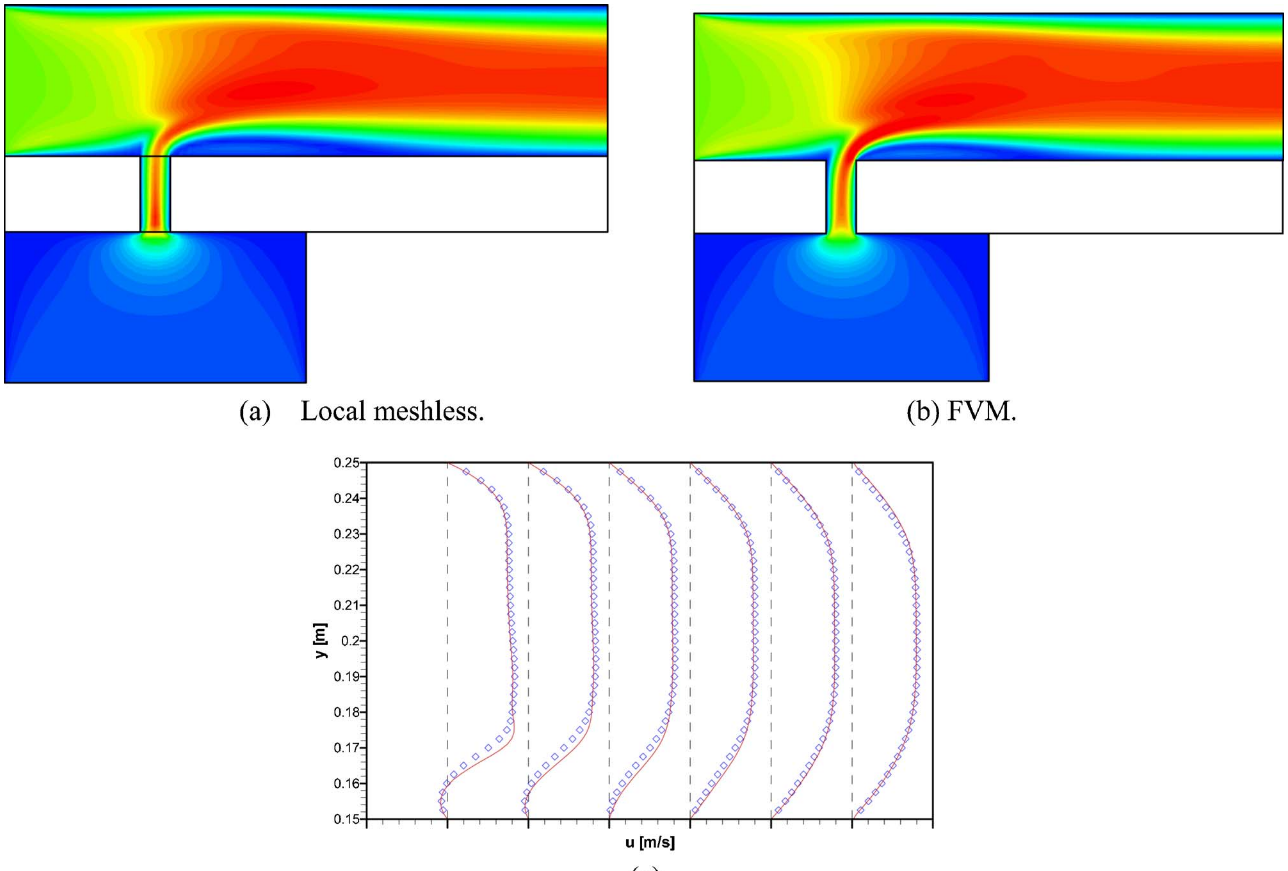
Fig. 25 Comparison of temperature distributions predicted by: (*a*) meshless and (*b*) FVM as well as (*c*) temperature profiles on top region at *x*=0.15, 0.2, 0.25, 0.3, 0.35, 0.4 m and (*d*) heat flux along top wall of stainless-steel block (*y*=0.15 m) after cooling hole

pared to FVM solutions processed in Fluent 6.1 are presented in Table 1. Here both FVM and meshless codes were implemented in the same platform (Single Xeon 64-bit, 3.2 GHz, 6 GB RAM) using the same grid and point distribution, respectively. Also, the settings of the FVM solution scheme were set to match those of the meshless solution scheme, i.e., explicit, coupled, unsteady with 1 sublevel iteration, $\Delta t=0.0001$ s. The values in Table 1 reveal that the localized meshless solution process is found to be more efficient than the FVM solution process implemented in Fluent 6.1 under the same conditions and scheme. The difference in performance is more noticable with the coarsest grid and point distribution which can be attributed to the amount of overhead involved in FVM computations. In addition to the average computation times over 1000 time steps, Table 1 also shows the localized meshless preprocessing time for generating the topologies and performing the RBF collocation and algebra, again revealing very little overhead at the startup of the process. It is worth mentioning that for this case the localized meshless topologies were configured to include up to 13 points each and despite this fact, this method was found to perform several times faster than Fluent 6.1 under the same conditions and platform.

In addition to the qualitative comparisons, the localized meshless results were compared along the geometrical center lines with those reported by Ghia et al. [40]. For this purpose, the Reynolds number was increased to 100 and 400 and the meshless resolution was increased to include 129×129 uniformly distributed points in order to match the resolution reported in Ref. [40]. Figures 11 and 12 display the x and y velocity components along the vertical and horizontal geometrical center lines respectively, revealing very close agreement between the meshless and reported benchmark FVM results.

Next, a conjugate solution is evaluated for developing air flow entering at 0°C and 0.167 m/s between parallel steel plates. The plates are 10 cm long and 1 cm apart. The plates have a finite thickness of 4 mm and the top and bottom are maintained at 100°C to test the effects of conjugate heat transfer. The geometry and thermophysical properties (in SI units) for the problem are shown in Fig. 13. The Prandtl number is taken as $\mathrm{Pr}=0.7$, and the Reynolds number based on the hydraulic diameter is $\mathrm{Re}=225$. The meshless solution was obtained by collocating 1919 (101×19) data centers while the FVM mesh contains 4091 nodes. The FVM and meshless velocity magnitude contours are shown in Fig. 14 while the temperature contours are shown in Fig. 15. A plot of the temperature profiles across the plates at 1/4, 1/2, 3/4, and 1 length of the plates is shown in Fig. 16 revealing high accuracy when compared to the FVM results. The heat flux along the interface wall between the air flow and the steel plates is shown in Fig. 17 again with very good agreement between the FVM and meshless solution. A heat balance analysis was carried out to show that the integrated heat rate into the fluid domain by conduction through the plates was $q_{\mathrm{interface}}=-112.304$ W perfectly symmetric through top and bottom, while the heat rate removed by convection through the inlet and outlet based on the mass-flow-weighted average temperatures was $q_{\mathrm{convection}}=111.797$ W, in addition to the axial conduction heat rate through inlet and outlet of $q_{\mathrm{axial}}=1.425$ W, leading to a total heat rate imbalance of $q_{\mathrm{imbalance}}=0.918$ W, or 0.82% of the total heat removed by convection, indicating remarkable precision in the analysis considering the coarseness of the point distribution.

The next numerical example examines air flowing through a channel with a titanium rectangular obstruction, all parameters are provided in the figure using SI units. The Prandtl number is taken as $\mathrm{Pr}=0.7$, and the Reynolds number based on the hydraulic di-


(a) Local meshless.

(b) FVM.

(c)

Fig. 26 Contours of velocity magnitude and velocity profiles: (*a*) LMRBF velocity contors; (*b*) FVM velocity contours; and (*c*) velocity profiles across the channel region at discrete locations *x*=0.15, 0.2, 0.25, 0.3, 0.35, and 0.4 m

ameter is Re=338. Again the effects of conjugate heat transfer will be tested as the bottom of the obstruction is kept at 100°C while the flow enters at 0°C with a velocity of 0.25 m/s. The channel is 10 cm long by 1 cm tall while the obstruction is 10 ×5 mm and is places 5 cm after the inlet as shown in Fig. 18. The meshless solution was obtained by collocation 4221 equally spaced data centers while the FVM mesh contains 9473 nodes. The FVM and meshless velocity contours are shown in Fig. 19. A plot of the velocity profiles after the obstruction at $x=7$, 8, 9, 10, and 11 cm is shown in Fig. 20 revealing very good agreement between the FVM and meshless solutions. A plot of the y derivative of the x velocity $(\partial u / \partial y)$ on the bottom wall after the obstruction is presented in Fig. 21 showing FVM results generated using different upwinding schemes as well as meshless results provided by four different resolutions of data centers (5×5, 10×10, 20 ×20, and 40×40 points per region). Results reveal that the meshless method solution tends to converge with increased density of data centers and agrees closely with the second order and QUICK FVM upwinding schemes in trend, in magnitude, and in prediction of the reattachment point location. Finally, the FVM and meshless temperature contour plots are shown in Fig. 22 as well as a zoom of the plot on the titanium obstruction in Fig. 23 clearly showing the multidimensional temperature gradients across the solid.

The next example consists of a flow of colder air from a plenum through a stainless steel plate into a hotter incoming flow. The problem is conjugate and the boundary conditions are shown in Fig. 24 along with the meshless method and finite volume discretizations consisting of 13,345 data points and 14,859 FVM nodes, while temperature solutions are compared in Fig. 25. The flow solutions and flow profiles predicted by LMRBF and FVM are compared in Fig. 26.

The last numerical example presents results for a nonuniform data center distribution for meshless methods predictions of flow over a cylinder centered in rectangular channel. The cylinder of diameter d is enclosed between two parallel plates $3 \cdot d$ apart by $10 \cdot d$ long. The center of the cylinder is located at 1/3 of the length of the channel. A uniform flow field with an entrance Reynolds number of 10 based on the cylinder diameter enters the channel from the left boundary. This example serves the purpose of demonstrating the robustness of the localized meshless approach by applying it to a nonrectangular geometry using a nonuniform point distribution using methods for automated data center distribution described in Ref. [41]. The meshless results are compared to grid-converged FVM results provided by Fluent 6.1. The meshless results are obtained using a nonuniform distribution of 8989 points while the FVM results are obtained using a clustered mesh with 7872 quadrilateral cells adapted to 31,488 quadrilateral cells to verify grid convergence. Figure 27 displays a close up of the FVM grid and meshless point distribution around the cylinder clearly showing the clustering of the grid and the nonuniformity of the point distribution. Figure 28 shows the FVM and localized meshless contours of the velocity magnitude revealing close qualitative agreement. A quantitative comparison is presented in Figs. 29 and 30 where the FVM, adapted FVM, and localized meshless pressure coefficients and viscous stress coefficients (normalized magnitude of the traction vector) are displayed

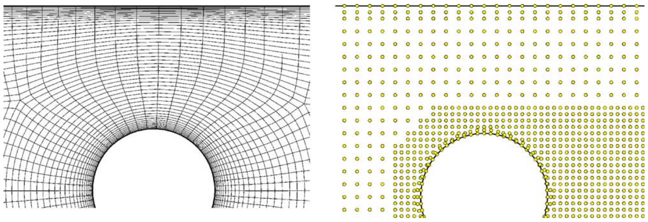


Fig. 27 FVM grid and meshless point distribution around the cylinder

around the top of the cylinder from leading edge to trailing edge, again revealing excellent agreement between grid-converged FVM results and localized meshless results.

Conclusions

In this paper we present a localized meshless method for the solution of coupled viscous fluid flow and convective heat transfer problems and implement it in parallel using domain decomposition. The method is based on localized RBF expansion using Hardy Multiquadrics for the sought-after unknowns. An efficient set of formulae are derived to compute the RBF interpolation in terms of vector products thus providing a substantial computational savings over traditional meshless methods. Moreover, the approach developed in this paper is applicable to explicit or implicit time-marching schemes as well as steady-state iterative methods. The incompressible Navier–Stokes set of equations for fluid flow and heat transfer in the fluid regions are solved using a standard time-marching scheme based on a Helmholtz potential decomposition. When conjugate heat transfer is considered, the simplified energy equation is solved in the solid regions using also a time-marching scheme.

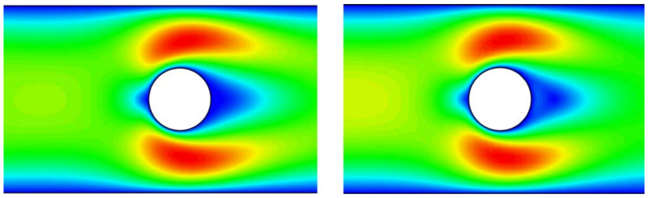


Fig. 28 FVM and meshless velocity magnitude contours around the cylinder

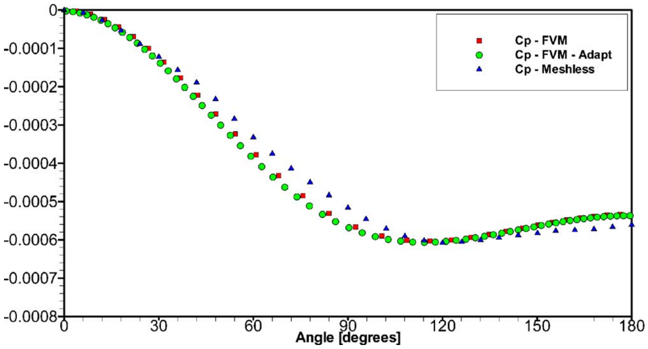


Fig. 29 Pressure coefficient around the cylinder

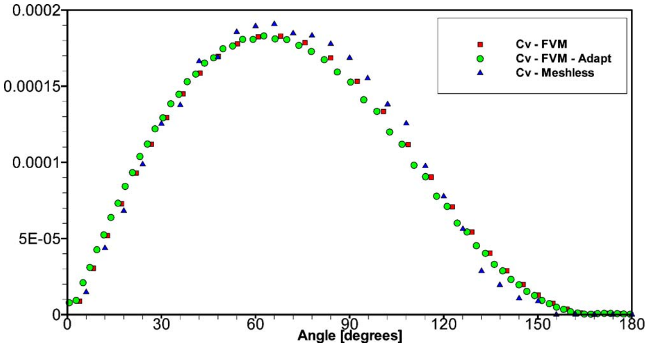


Fig. 30 Viscous stress coefficient around the cylinder

Numerical results are presented and compared to results predicted by a commercial CFD code (Fluent) for several cases including a jet flow into a square cavity, conjugate analysis of a thick-walled channel flow, conjugate analysis of flow in a channel with a square step obstruction, and conjugate analysis of a plenum-cooled plate in contact with a hot freestream. Results consistently show the localized meshless method approach produces accurate results even when using a much coarser number of data centers versus FVM nodes. The approach is accurate and efficient while requiring a much-reduced effort in problem preparation as compared to FVM. The authors are investigating several extension of the method, including alterntive solutions schemes that lend themselves to parallel computation without resort to domain decomposition, application to natural convection and species transport, and extensions to three-dimensional and incorporation of turbulence models. These forthcoming results will be reported elsewhere in the literature.

Nomenclature

$\mathbf{V}(\boldsymbol{x})$	=	velocity field
$p(\boldsymbol{x})$	=	pressure field
$T(\boldsymbol{x})$	=	temperature field
$\phi(\boldsymbol{x})$	=	Helmholtz potential
$\partial/\partial n$	=	outward normal derivative
$\mathbf{f}$	=	specific body force
ρ	=	density
μ	=	dynamic viscosity
k	=	thermal conductivity
c	=	specific heat
Φ	=	viscous dissipation
NC	=	number of data centers
NB	=	number of boundary collocation points
NI	=	number of internal collocation points
α_j	=	expansion coefficients
$\chi_j(x)$	=	expansion functions
$r_j(x)$	=	Euclidean distance to point j

References

[1] Belytscho, T., Lu, Y. Y., and Gu, L., 1994, "Element-Free Galerkin Methods," Int. J. Numer. Methods Eng., **37**, pp. 229–256.

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

[3] Atluri, S. N., and Zhu, T., 1998, "A New Meshless Local Petrov-Galerkin (MLPG) Approach in Computational Mechanics," Comput. Mech., **22**, pp. 117–127.

[4] Liu, G. R., 2003, *Mesh Free Methods*, CRC Press, Boca Raton, FL.

[5] Melenk, J. M., and Babuska, I., 1996, "The Partition of Unity Finite Element Method: Basic Theory and Application," Comput. Methods Appl. Mech. Eng., **139**, pp. 289–316.

[6] Kansa, E. J., 1990, "Multiquadrics—A Scattered Data Approximation Scheme With Applications to Computational Fluid Dynamics I—Surface Approximations and Partial Derivative Estimates," Comput. Math. Appl., **19**, pp. 127–145.

[7] Kansa, E. J., 1990, "Multiquadrics—A Scattered Data Approximation Scheme With Applications to Computational Fluid Dynamics II—Solutions to Parabolic, Hyperbolic and Elliptic Partial Differential Equations," Comput. Math. Appl., **19**, pp. 147–161.

[8] Kansa, E. J., and Hon, Y. C., 2000, "Circumventing the Ill—Conditioning Problem with Multiquadric Radial Basis Functions: Applications to Elliptic Partial Differential Equations," Comput. Math. Appl., **39**, pp. 123–137.

[9] Franke, R., 1982, "Scattered Data Interpolation: Test of Some Methods," Math. Comput., **38**, pp. 181–200.

[10] Mai-Duy, N., and Tran-Cong, T., 2002, "Mesh-Free Radial Basis Function Network Methods With Domain Decomposition for Approximation of Functions and Numerical Solution of Poisson's Equation," Eng. Anal. Boundary

Elem., **26**, pp. 133–156.

[11] Cheng, A. H.-D., Golberg, M. A., Kansa, E. J., and Zammito, G., 2003, "Exponential Convergence and H-c Multiquadric Collocation Method for Partial Differential Equations," Numer. Methods Partial Differ. Equ., **19**, pp. 571–594.

[12] Gottlieb, D., and Orzag, S. A., 1977, *Numerical Analysis of Spectral Methods: Theory and Applications*, Society for Industrial and Applied Mathematics, Bristol, UK.

[13] Maday, Y., and Quateroni, A., 1982, "Spectral and Pseudo-Spectral Approximations of the Navier-Stokes Equations," SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal., **19**, pp. 761–780.

[14] Patera, A., 1984, "A Spectral Element Method of Fluid Dynamics: Laminar Flow in a Channel Expansion," J. Comput. Phys., **54**, pp. 468–488.

[15] Macaraeg, M., and Street, C. L., 1986, "Improvement in Spectral Collocation Discretization Through a Multiple Domain Technique," Appl. Numer. Math., **1989**, pp. 95–108.

[16] Hwar, C. K., Hirsch, R., Taylor, T., and Rosenberg, A. P., 1989, "A Pseudo-Spectral Matrix Element Method for Solution of Three Dimensional Incompressible Flows and its Parallel Implementation," J. Comput. Phys., **83**, pp. 260–291.

[17] Fasshauer, G., 2005, "RBF Collocation Methods as Pseudo-Spectral Methods," *Boundary Elements XVII*, A. Kassab, C. A. Brebbia, and E. Divo, eds., WIT Press, Southampton, UK, pp. 47–57.

[18] Powell, M. J. D., 1992, "The Theory of Radial Basis Function Approximation," *Advances in Numerical Analysis*, W. Light, ed., Oxford Science Publications, Oxford, Vol. II, pp. 143–167.

[19] Buhmann, M. D., 2003, *Radial Basis Functions: Theory and Implementation*, Cambridge University Press, Cambridge, MA.

[20] Dyn, N., Levin, D., and Rippa, S., 1986, "Numerical Procedures for Surface Fitting of Scattered Data by Radial Basis Functions," SIAM (Soc. Ind. Appl. Math.) J. Sci. Stat. Comput., **7**, pp. 639–659.

[21] Brebbia, C. A., Partridge, P., and Wrobel, L. C., 1992, *The Dual Reciprocity Boundary Element Method*, Computational Mechanics and Elsevier Applied Science, Southampton, UK.

[22] Golberg, M., Chen, C. S., and Bowman, H., 1999, "Some Recent Results and Proposals for the Use of Radial Basis Functions in the BEM," Eng. Anal. Boundary Elem., **23**, pp. 285–296.

[23] Rahaim, C. P., and Kassab, A. J., 1996, "Pressure Correction DRBEM Solution for Heat Transfer and Fluid Flow in Incompressible Viscous Fluids," Eng. Anal. Boundary Elem., **18**, pp. 265–272.

[24] Sarler, B., Tran-Cong, T., and Chen, C. S., 2005, "Meshfree Direct and Indirect Local Radial Basis Function Collocation Formulations for Transport Phenomena," *Boundary Elements XVII*, A. Kassab, C. A. Brebbia, and E. Divo, eds., WIT Press, Southampton, UK, pp. 417–428.

[25] Sarler, B., and Vertnik, R., 2005, "Local Explicit Radial Basis Function Collocation Method for Diffusion Problems," Comput. Math. Appl., **51**(8), pp. 1269–1282.

[26] Hardy, R. L., 1971, "Multiquadric Equations of Topography and Other Irregular Surfaces," J. Geophys. Res., **176**, pp. 1905–1915.

[27] Kassab, A., Divo, E., Heidmann, J., Steinthorsson, E., and Rodriguez, F., 2003, "BEM/FVM Conjugate Heat Transfer Analysis of a Three-Dimensional Film Cooled Turbine Blade," Int. J. Numer. Methods Heat Fluid Flow, **13**, pp. 581–610.

[28] Kassab, A. J., and Aliabadi, M. H., eds., 2001, *Advances in Boundary Elements: Coupled Field Problems*, WIT Press, Southampton, UK.

[29] Rahaim, C. P., Kassab, A. J., and Cavalleri, R., 2000, "A Coupled Dual Reciprocity Boundary Element/Finite Volume Method for Transient Conjugate Heat Transfer," AIAA J. Thermophys. Heat Transfer, **14**, pp. 27–38.

[30] He, M., Bishop, P., Kassab, A. J., and Minardi, A., 1995, "A Coupled FDM/BEM Solution for the Conjugate Heat Transfer Problem," Numer. Heat Transfer, Part B, **28**, pp. 139–154.

[31] Li, H., and Kassab, A. J., 1981, "A Coupled FVM/BEM Solution to Conjugate Heat Transfer in Turbine Blades," AIAA Paper No. 94–1981.

[32] Divo, E. A., Kassab, A. J., and Rodriguez, F., 2003, "Parallel Domain Decomposition Approach for Large-Scale 3D Boundary Element Models in Linear and Non-Linear Heat Conduction," Numer. Heat Transfer, Part B, **44**, pp. 417–437.

[33] Divo, E., Kassab, A. J., Mitteff, E., and Quintana, L., 2004, "A Parallel Domain Decomposition Technique for Meshless Methods Applications to Large-Scale Heat Transfer Problems," ASME Paper No. HT-FED2004-56004.

[34] Divo, E., and Kassab, A. J., 2005, "Effective Domain Decomposition Meshless Formulation of Fully-Viscous Incompressible Fluid Flows," *Boundary Elements XVII*, A. Kassab, C. A. Brebbia, and E. Divo, eds., WIT Press, Southampton, UK, pp. 67–77.

[35] Divo, E., and Kassab, A. J., 2005, "A Meshless Method for Conjugate Heat Transfer," *Proceedings of ECCOMAS Coupled Problems 2005*, M. Papadrakakis, E. Oñate, and B. Schrefler, eds., Santorini, Greece, April.

[36] Divo, E., and Kassab, A. J., 2005, "A Meshless Method for Conjugate Heat Transfer Problems," Eng. Anal., **29**, pp. 136–149.

[37] Harlow, F. H., and Welch, J. E., 1965, Numerical Calculation of Time Dependent Viscous Incompressible Flow of Fluids With a Free Surface, Phys. Fluids, **8**, pp. 2182–2189.

[38] Patankar, S. V., 1980, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, Washington, DC.

[39] Orlanski, I., 1976, "A Simple Boundary Condition for Unbounded Hyperbolic Flows," J. Comput. Phys., **21**, pp. 251–269.

[40] Ghia, U., Ghia, K. N., and Shin, C. T., 1982, "High-RE Solutions for Incompressible-Flow Using the Navier Stokes Equations and a Multigrid Method," J. Comput. Phys., **48**(3), pp. 387–411.

[41] Mitteff, E., Divo, E., and Kassab, A. J., 2006, "Automated Point Distribution and Parallel Segmentation for Meshless Methods," *Proceedings of CIMENICS 2006, 8th International Congress of Numerical Methods in Engineering and Applied Sciences*, B. Gamez, D. Ojeda, G. Larrazabal, and M. Cerrolaza, eds., Sociedad Venezuelana de Methodos Numericos En Engineria, Valencia, Venezuela, Margarita Island, Venezuela, March 20–24, pp. 93–100.