Scientia Iranica, Vol. 10, No. 1, pp 44–55
© Sharif University of Technology, January 2003

SCIENTIA
IRANICA

Incompressible Stokes Flow Calculation Using a Finite Point Method

S. Kazemzadeh Hannani* and F. Parsinejad[1]

In this paper, a finite point method is employed to solve the incompressible laminar Stokes flow. A moving least-squares approximation, using linear and quadratic basis functions, in conjunction with a point collocation method, has been utilized to discretize the governing equations. Two examples, including the driven cavity and the fully developed channel flow, are solved showing the accuracy and applicability of the method. In summary, the solutions for the linear basis case exhibit a large sensitivity to the size of the domain of influence of the weighting function, in contrast to the quadratic basis case.

INTRODUCTION

Standard numerical methods, such as finite element and finite volume techniques, have been widely used during the last three decades for solving complex problems in computational mechanics and its subclass, Computational Fluid Dynamics (CFD). Today, computational mechanics has become an efficient engineering tool, both in industry for design purposes and in research oriented works and one cannot preclude the important role of numerical methods in the route towards progress in science. Even turbulence simulation, that in the past seemed intractable, is today handled by efficient numerical schemes. However, it does not seem that the end of the tunnel has been reached and numerical schemes still encounter serious shortcomings. Research should be continued and innovations are needed to solve, efficiently and accurately, fluid flow problems.

Since the last two decades, advances in 3d mesh generation, computer technology and numerical algorithms have made possible the use of computational mechanics to analyze complex configurations with realistic and complex geometries. However, as configurations of greater complexity are analyzed, the task of generating a suitable grid for the analysis becomes increasingly difficult. More effort may be required to generate the grid for a complex configuration than the flow field solution. For industrial applications, the bottleneck is the mesh generation because it can absorb far more time and cost than the solution itself. In addition, grid generation is further complicated in some cases by the need to adapt the grid as the solution progresses. This is especially true for unsteady flow fields with moving components.

In the finite element method and its subclass, the finite volume technique, the computational domain is divided into finite numbers of sub-domains on which a volume integration is performed. The sub-domains are constrained by some geometrical regularity conditions, such as having positive volume (regular mapping) or a limited aspect ratio between the element dimensions (for accuracy and convergence consideration). In addition, efficient bookkeeping is needed to save the data structure regarding element connectivities. Although this poses no serious difficulties in two dimensions, the lack of robust and efficient three dimensional mesh generators makes the solution of 3D problems a more difficult task. In addition, there is a serious need to remesh economically for adaptive mesh refinement procedures and/or for problems with moving boundaries. Therefore, a method is needed where the movement of a node does not lead to numerical deficiencies and easier adaptive refinements could be performed. From a computational point, it would be desirable if only a collection of nodes, regardless of their orientation and connectivity, and a boundary were needed to discretize the equations. This idea would be exploited even more efficiently in adaptive methods, in which the user, in an interactive manner, could simply add a large number of points into the regions possessing a large gradient,

*. *Corresponding Author, Department of Mechanical Engineering, Sharif University of Technology, Tehran, I.R. Iran.*

1. *Department of Mechanical Engineering, Sharif University of Technology, Tehran, I.R. Iran.*

circumventing the construction of a new finite element or a new finite volume.

During recent years, considerable effort has been devoted to the development of the so-called mesh-free or "meshless" methods. Naylroles et al. [1] was the first to propose a meshless Galerkin method for the approximation of boundary value problems and referred it to the Diffuse Element (DE) method. In this method, the interpolants are polynomials, which are fitted to the nodal values by a local least-squares approximation, valid in the neighborhood of a point "x" and based on a given number of surrounding nodes. This kind of interpolation, although not noted by Nayroles et al., has been studied before by Lancaster and Salkauskas [2] and called Moving Least-Squares (MLS) interpolants, as a means of interpreting irregularly distributed function-value data. However, in the DE method of Nayroles et al. some kind of auxiliary grid is needed for numerical integration of Galerkin terms. Belytschko et al. [3] extended the idea of 'diffuse elements' and enhanced its accuracy by introducing three refinements: (a) Lagrange multipliers are added to the energy functional to enforce essential boundary conditions, (b) The derivatives of shape functions are evaluated exactly and (c) Spatial integration is improved by introducing a regular cell structure.

Batina [4] developed another class of meshless methods and applied it to solve compressible Navier-Stokes equations. In this method, the fluxes are approximated using a least-squares fit, based on linear polynomials. This idea is further extended and formalized by Onate et al. [5] by applying a Weighted Least Squares (WLS) interpolation. In the latter method, the Galerkin (Weighted Residual Method) formulation is approximated using a point collocation technique, circumventing the numerical integration of Galerkin terms and is referred to as the Finite Point Method. The stabilization of convection dominated flows has been also treated in this work and a residual based stabilization procedure is developed. Other varieties of Meshless methods have also been proposed in the literature (see e.g., an overview in [6]) as Generalized Finite Difference Method [7], Smoothed Particle Hydrodynamic [8], Wavelet Galerkin Method [9], Hp-clouds [10], Reproducing Kernel Particle Methods [11] and Partition of Unity Finite Element Method [12].

However, despite recent advances, the meshless method is still in its infancy and its potentials and advantages have not yet been deeply explored. In this paper, the finite Point method proposed by Onate et al. [5] has been applied to solve the incompressible Stokes flow. A moving least-squares approximation, using linear and quadratic basis interpolation functions, has been employed. The accuracy and behavior of this method is elucidated by solving the driven cavity and fully developed channel flow.

BASIC CONCEPTS OF MESHLESS METHODS

Moving Least-Squares Approximation

In the Moving Least-Square (MLS) interpolant, the function $u(x)$ is approximated by:

$$u(x) \cong \hat{u}(x) = \sum_{i=1}^{m} p_i(x)\alpha_i(x) = \mathbf{P}^T(x)\alpha(x), \qquad (1)$$

in which $p_i(x)$ are m monomials in the space coordinate x, which are chosen so that the basis is complete. For example, in one dimension,

$$m = 2 \quad \mathbf{P}^T = [1, x],$$

$$m = 3 \quad \mathbf{P}^T = [1, x, x^2], \qquad (2)$$

and in two dimensions:

$$m = 3 \quad \mathbf{P}^T = [1, x, y]$$

$$m = 6 \quad \mathbf{P}^T = [1, x, y, x^2, xy, y^2]. \qquad (3)$$

The unknown coefficients $\alpha_i(x)$ in Equation 1 are obtained by minimizing a weighted discrete L_2 error norm as follows:

$$J(x) = \sum_{j=1}^{n} w_j(x - x_j)\left(u_j^h - \mathbf{P}_j^T\alpha(x)\right)^2, \qquad (4)$$

where n is the number of points in the neighborhood of x, for which the weight function:

$$w_j(x - x_j) \neq 0. \qquad (5)$$

The neighborhood of x is called the domain of influence or the cloud of interpolation of x. u_j^h is the nodal value at $x = x_j$ (see Figure 1). It must be noted that in this procedure, $\hat{u}(x_j) \neq u_j^h$. The minimizing procedure (Equation 4) leads to the following linear relation between $\alpha(x)$ and u_i^h:

$$\mathbf{A}(x)\alpha(x) = \mathbf{B}(x)\mathbf{U}^h, \qquad (6)$$

where $\mathbf{A} \in R^{m\times m}$ and $\mathbf{B} \in R^{m\times m}$ are $m \times m$ and $m \times n$ matrices, respectively, defined by:

$$\mathbf{A}(x) \sum_{j=1}^{n} w_j(x - x_j)\mathbf{P}(x_j)\mathbf{P}^T(x_j),$$

$$\mathbf{B}(x) = \left[w_1(x - x_1)\mathbf{P}(x_1), w_2(x - x_2)\mathbf{P}(x_2), \cdots, w_n(x - x_n)\mathbf{P}(x_n)\right], \qquad (7)$$

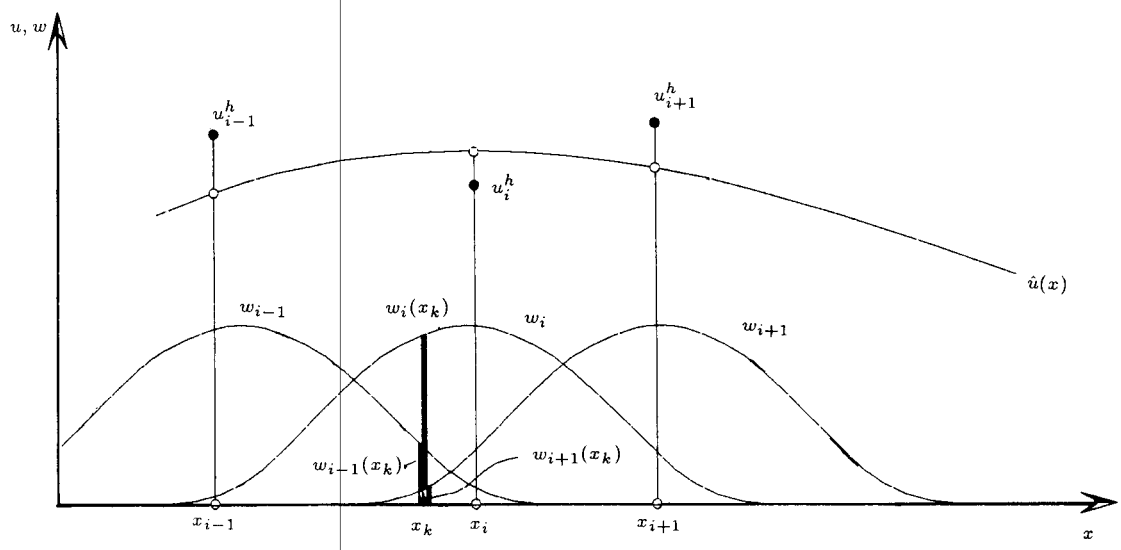


Figure 1. Moving least square procedure.

and:

$$\mathbf{U}^h = \begin{Bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_n^h \end{Bmatrix}. \tag{8}$$

Substituting the solution of Equation 6 into global approximation (Equation 1) results in:

$$u(x) \cong \hat{u}(x) = \sum_{i=1}^{n} \phi_i(x) u_i^h = \Phi^T(x)\mathbf{U}^h, \tag{9}$$

where:

$$\Phi(x) = \mathbf{P}^T(x)\mathbf{A}^{-1}(x)\mathbf{B}(x), \tag{10}$$

and is called the shape function. Relation 9 renders a local interpolation of $u(x)$ as it is represented by a combination of locally defined shape functions with the following properties:

$$\begin{cases} \phi_i(x) \neq 0 & \text{if} \quad x \in \Omega_i \\ \phi_i(x) = 0 & \text{if} \quad x \notin \Omega_i \end{cases}, \tag{11}$$

where Ω_i is the sub-domain of Ω (cloud of approximation) containing n points.

Remarks 1

It must be noted that according to the least-square character of the approximation, one has:

$$u(x_j) \cong \hat{u}(x_j) \neq u_j^h, \tag{12}$$

i.e., the local values of the approximating function do not fit the nodal unknown values. Indeed, $\hat{u}$ is the true approximation for which the satisfaction of the differential equation and boundary conditions will be sought and u_j^h are simply the unknown parameters sought.

Remarks 2

If one puts $n = m$, the matrix $\mathbf{A}(x)$ will be square and the Finite Element Method is reproduced. In this case, the shape functions satisfy the following relation:

$$\phi_j(x_i) = 1, \quad j = i$$

$$\phi_j(x_i) = 0 \quad j \neq i, \quad i, j = 1, 2, \cdots, n. \tag{13}$$

Remarks 3

Note also that in the MLS case the parameters, α, are not constants and depend on the location x. Therefore, the inversion of matrix $\mathbf{A}(x)$ is required at every point where function $\hat{u}(x)$ needs to be evaluated. It can be shown (see [13] for details) that a unique global definition of shape functions can be obtained provided:

(i) The weighting function, w_j, is continuous and differentiable in Ω_j called the interpolation domain, or set of cloud of points approximating the function $u(x)$;

(ii) The weighting function w_j, vanishes on the boundary of Ω_j;

(iii) The number of points n within Ω_j is equal to or greater than the parameters m at all points in Ω_j.

Derivatives of Shape Functions

Following Belytschko et al. [3], the first and second derivatives of shape functions are computed as:

$$\boldsymbol{\phi}_{,i} = \underline{\mathbf{P}_{,i}^T\mathbf{A}^{-1}\mathbf{B}} + \mathbf{P}^T(\mathbf{A}^{-1}\mathbf{B}_{,i} - \mathbf{G}_i\mathbf{B}), \tag{14}$$

and:

$$\begin{aligned}\phi_{,ij} =&(\mathbf{P}^T\mathbf{G}_i - \mathbf{P}^T_{,i}\mathbf{A}^{-1})(\mathbf{A}_{,j}\mathbf{A}^{-1}\mathbf{B} - \mathbf{B}_{,i})\\ &+ (\mathbf{P}^T\mathbf{G}_i - \mathbf{P}^T_{,j}\mathbf{A}^{-1})(\mathbf{A}_{,j}\mathbf{A}^{-1}\mathbf{B} - \mathbf{B}_{,i})\\ &- \mathbf{P}^T(\mathbf{G}_{ij}\mathbf{B} - \mathbf{A}^{-1}\mathbf{B}_{,ij}) + \underline{\mathbf{P}^T_{,ij}\mathbf{A}^{-1}\mathbf{B}},\end{aligned} \tag{15}$$

where:

$$\mathbf{G}_i = \mathbf{A}^{-1}\mathbf{A}_{,i}\mathbf{A}^{-1}, \quad \mathbf{G}_{ij} = \mathbf{A}^{-1}\mathbf{A}_{,ij}\mathbf{A}^{-1}. \tag{16}$$

Nayroles et al. [1], suggest that approximations ignoring the derivatives of α term may be used to define the derivatives of shape functions, in order to reduce the computational effort. The underlined terms in Equations 14 and 15 may be sufficient to compute the derivatives, using this approximation.

FINITE POINT FORMULATION

Following Onate et al. [5], consider a problem governed by the following differential equation and boundary conditions:

$$\begin{cases} A(u) = b & \text{in} \quad \Omega \\ B(u) = t & \text{in} \quad \Gamma_t \\ u - u_p = 0 & \text{in} \quad \Gamma_u, \end{cases} \tag{17}$$

where Ω is the domain of the problem, A and B are appropriate differential operators, u is the problem unknown, Γ_u and Γ_t are Dirichlet and Neumann boundary conditions, respectively and b and t represent external forces acting over the domain Ω and along the boundary Γ_t, respectively. Finally, u_p is the prescribed value of u along the boundary Γ_u.

The above system of differential equations can be solved numerically using the weighted residual method, in which the unknown function, u, is approximated by some trial approximation $\hat{u}$ and the above system of equations is replaced by:

$$\int_\Omega W_i[A\hat{u} - b]d\Omega + \int_{\Gamma_t} \overline{W}_i[B\hat{u} - t]d\Gamma + \int_{\Gamma_u} \overline{\overline{W}}_i[\hat{u} - u_p]d\Gamma = 0, \tag{18}$$

where $W_i, \overline{W}_i$ and $\overline{\overline{W}}_i$ are weighting functions defined in different ways depending on the numerical method. Onate et al. [5] observed that in order to preserve the mesh-free character of the method, the weighting domain must be defined independently of any mesh. However, approximation methods of an integral type, such as variants of the Galerkin method, necessitate the introduction of complex procedures for integration (e.g., Gauss quadrature rules used in finite element integration formulas). Batina [4] developed a class of meshless methods to solve compressible Navier-Stokes equations. In this method, the fluxes are approximated using a least-squares fit, based on linear polynomials. This idea is further extended and formalized by Onate et al. [5] by applying a Weighted Least Squares (WLS) interpolation. The Galerkin (Weighted Residual Method) formulation is approximated using a point collocation technique, circumventing the numerical integration of Galerkin terms and is referred to as the finite point method.

Point Collocation

Point collocation implies putting $W_i = \overline{W}_i = \overline{\overline{W}}_i = \delta_i$, in Equation 18 where δ_i, is dirac delta. This gives a set of equations:

$$\begin{aligned} &\big[A(\hat{u})\big]_i - b_i = 0 \quad \text{in } \Omega,\\ &\big[B(\hat{u})\big]_i - t_i = 0 \quad \text{in } \Gamma_t,\\ &\hat{u}_i - u_p = 0 \quad \text{in } \Gamma_u. \end{aligned} \tag{19}$$

Using a moving least-squares interpolation of u (such as Equation 9), Equations 19 can be converted to the following system of algebraic equations:

$$\mathbf{K}\mathbf{U}^h = \mathbf{f}, \tag{20}$$

where $K_{ij} = \lfloor A(\Phi_j)\big]_i + \lfloor B(\Phi_j)\rfloor_i$. Note that the symmetry of the coefficients in matrix $\mathbf{K}$ is not generally achieved. $\mathbf{f}$ is a vector containing the contributions from the force term b and t and the prescribed values u_p.

Treatment of Boundary Conditions

Following Onate et al. [5] the satisfaction of Dirichlet boundary conditions:

$$u - u_p = 0 \quad \text{in} \quad \Gamma_u, \tag{21}$$

can be approximated for points x_i on Γ_u by employing a cloud, including only the point x_i and using constant interpolating basis functions. This procedure leads to:

$$u(x) \cong \hat{u}(x) = 1.\alpha_1, \tag{22}$$

$$u(x_i) \cong \hat{u}(x_i) = u^h_i, \tag{23}$$

$$u^h_i = u_p(x_i), \quad x_i \text{ on } \Gamma_u. \tag{24}$$

The Neumann boundary conditions on Γ_t, may be approximated as:

$$B\big[\hat{u}(x_i)\big] = t(x_i), \quad x_i \text{ on } \Gamma_t. \tag{25}$$

Note that in this way the differential equation is collocated only at internal points.

INCOMPRESSIBLE STOKES EQUATIONS

Accurate and stable solution of Stokes equations is one of the most challenging problems in the domain of numerical analysis. Consequently, it is not surprising that the Stokes problem has been one of the major subjects of active research of the finite element or finite volume community during the past 30 years. The dependent variables in the Stokes flow are the velocity and pressure. In the context of the finite element method, it is well known that the finite element spaces for approximating velocity and pressure cannot be chosen independently using classical Galerkin formulations. Indeed, to obtain optimally convergent velocity and pressure solutions, the finite element spaces must satisfy a stability criterion known as the Babuska-Brezzi condition [14,15]. For example, equal-order interpolations for both dependent variables fail to satisfy this condition. The pressure solution in this case may exhibit unphysical oscillations due to the presence of spurious pressure modes [16]. Standard finite difference and finite volume methods also suffer from this pathological behavior. In these methods, stable pressure solutions can be obtained by employing staggered grid systems for pressure and velocity variables (see [17]). To circumvent the Babuska-Brezzi condition, finite element researchers have introduced modified variational formulations known as stabilized methods. In these methods, the standard Galerkin formulation is modified by the addition of mesh dependent terms which are weighted residuals of the differential equations. Various extensions of these formulations have been proposed and analyzed in the literature (see [18,19]). These formulations are known today as GLS (Galerkin/Least-Squares) methods. These methods allow the implementation of equal order interpolations for pressure and velocity.

In this paper, a finite point method based on a moving least-squares approximation is used to solve the incompressible Stokes equations. The pressure and velocity variables are defined at the same points. In addition, the domain of influence of the weight functions $w_j(x - x_j)$ used for interpolating pressure and velocity variables through Equations 1 to 10 are identical. The pressure term is stabilized by adding the Poisson equation for pressure to the continuity equations. The incompressible Stokes equations are given by:

$$-\mu\triangle\mathbf{u} + \bigtriangledown p = \mathbf{f} \quad \Omega \subset \Re^{n_{sd}}, \tag{26}$$

$$\bigtriangledown.\mathbf{u} = 0 \quad \Omega \subset \Re^{n_{sd}}, \tag{27}$$

$$\mathbf{u} = g \quad \Gamma, \tag{28}$$

where $\mathbf{u}$ is the velocity, p is the pressure, μ is the dynamic viscosity, $\triangle$ is Laplacian operator and $\mathbf{f}$ is the body force per unit mass. Ω is a bounded domain and n_{sd} is the number of space dimensions. $\Re$ is the field of real numbers and Γ is the boundary of Ω. For the sake of simplicity, only Dirichlet boundry conditions will be studied for velocity and a constant viscosity will be considered. The pressure is determined up to an additive constant.

Now, using the Finite Point Method, the incompressible Stokes equation in the two-dimensional case is written in the following discretized form:

$$-(\phi_{,x})_i P_i^h + \mu[(\phi_{,xx})_i + (\phi_{,yy})_i]u_i^h = 0,$$

$$-(\phi_{,y})_i P_i^h + \mu[(\phi_{,xx})_i + (\phi_{,yy})_i]\nu_i^h = 0,$$

$$(\phi_{,x})_i u_i^h + (\phi_{,y})_i \nu_i^h = 0. \tag{29}$$

As stated previously, the pressure term is stabilized by modifying the continuity equation using the Poisson equation for pressure in the following form:

$$\bigtriangledown.\mathbf{u} + \alpha \bigtriangledown .(-\bigtriangledown p + \mu\triangle\mathbf{u}) = 0. \tag{30}$$

The equation in the collocated form is written as:

$$(\phi_{,x})_i u_i^h + (\phi_{,y})_i \nu_i^h + \alpha\Big\{-[(\phi_{,xx})_i + (\Phi_{,yy})_i]p_i^h + \mu[(\phi_{,xxx})_i + (\phi_{,xyy})_i]u_i^h + \mu[(\phi_{,xxy})_i + (\phi_{,yyy})_i]\nu_i^h\Big\} = 0 \tag{31}$$

A third order spline weighting function in the following form is used in this investigation.

$$w(x - x_i) = w_i(d), \tag{32}$$

where:

$$d = ||x - x_i||, \tag{33}$$

$$w(r) = \begin{cases} 2/3 - 4r^2 + 4r^3 & r \le 1/2 \\ 4/3 - 4r + 4r^2 - 4/3r^3 & 1/2 < r \le 1 \\ 0 & r > 1 \end{cases}, \tag{34}$$

and:

$$r = \frac{||x - x_i||}{\max||x - x_i||}, \tag{35}$$

or:

$$r = \frac{||x - x_i||}{DM_i}, \tag{36}$$

DM_i can be written as a function of the maximum distance between two nodes ($||x_j - x_i||$) by:

$$DM_i = DMAX||x_j - x_i||, \tag{37}$$

where $DMAX$ is a parameter that controls the domain of influence of the weighting function. By simple multiplication, Equation 34 can be extended to multi-dimensions.

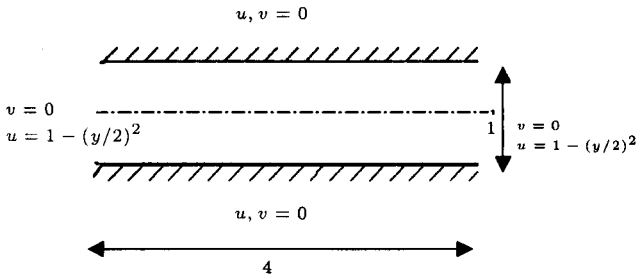


Figure 2. Domain and boundary conditions for Poiseuille flow.

NUMERICAL EXPERIMENTS

Plane Poiseuille Flow

A fully developed Poiseuille flow between two channel walls is solved and the results are compared with analytical solutions. The domain and boundary conditions are shown in Figure 2. No slip boundary conditions are imposed at solid walls and parabolic Poiseuille flow velocity profiles are assumed at inlet and outlet boundaries. Since only Stokes equation is considered in this work, the assumption of uniform velocity distribution at the inlet will not lead to a physically consistent boundary layer development. Putting $\mu = 1$, the pressure drop, Δp, for a channel length of 4 units is equal to 24. The domain consists of 15 × 8 points. Calculations are performed employing linear ($m = 3$) and quadratic ($m = 6$) basis functions. The sensitivity of solution to the stabilization parameter, α, and to the selection of the $DMAX$ parameter has been also investigated. $DMAX$ controls the size of the domain of influence of the weighting function.

Linear Basis Functions

In Figure 3, the pressure contours are shown for $m = 3, \alpha = 1.0$ and $DMAX = 2$. For this case, the calculated pressure loss is equal to 25, which is in good agreement with the analytical solution. In Table 1, the variations of Δp with respect to $DMAX$ are given. Note that solutions depend largely on the parameter $DMAX$. Smooth pressure contours are obtained for $DMAX$ from 2 to 4, using a stabilization parameter $\alpha = 1.0$, and they resemble the $DMAX = 2$ case (hence, not shown). In order to study the sensitivity of the solution to selection of the stabilization parameter, calculations are conducted using $\alpha = 0.001, m = 3$ and $DMAX = 2.0$. The pressure contours for this case are shown in Figure 4. One can note the slight oscillations near the inlet and outlet boundary conditions.

Table 1. Pressure loss for Poiseuille flow in a channel using linear basis functions.

DMAX	2	2.5	3	4	4.5	5	6
Pressure Loss	25	25.5	31	25.5	26	32	26

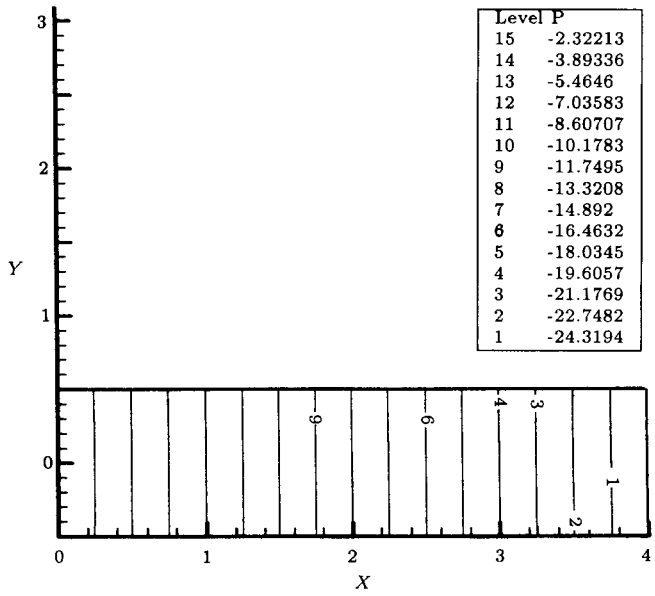


Figure 3. Pressure contours for Poiseuille flow using linear basis functions ($m = 3), \alpha = 1.0$, and $DMAX = 2.0$.

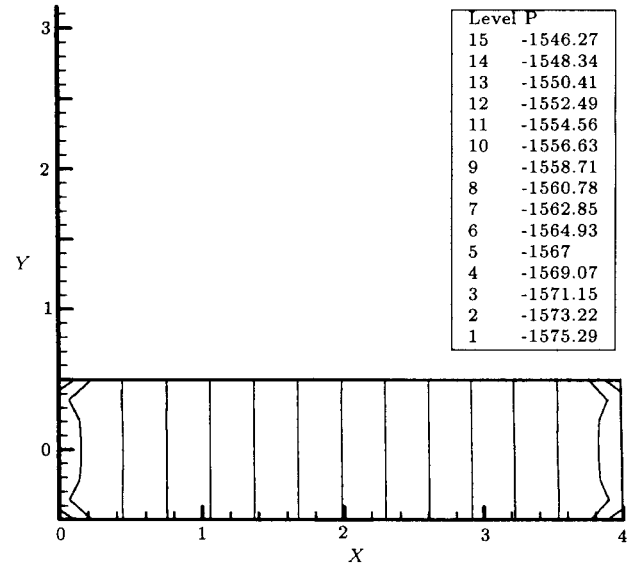


Figure 4. Pressure contours for Poiseuille flow using linear basis functions ($m = 3), \alpha = 0.001$ and $DMAX = 2.0$.

Quadratic Basis Functions

In Figure 5, the pressure contours are shown for $\alpha = 0.001, m = 6$ (quadratic basis function) and $DMAX = 2$. For $DMAX$ values between 2 to 6, the pressure loss is almost equal to 24 and is in very good agreement with the analytical solution. Pressure contours for other $DMAX$ values are not shown due to the resemblance to Figure 5. It may be concluded that the solutions are not sensitive to the $DMAX$ values. However, for quadratic basis function, employing a stabilization parameter $\alpha = 1.0$ renders a very smooth and inaccurate

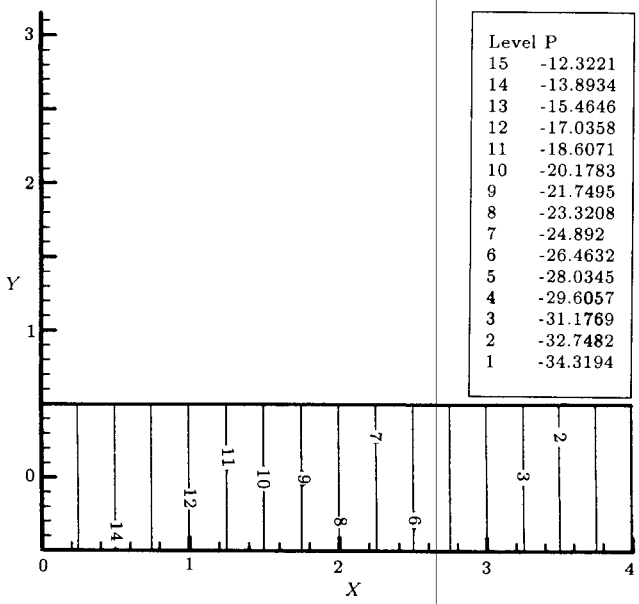


Figure 5. Pressure contours for Poiseuille flow using quadratic basis functions $(m = 6), \alpha = 0.001$, and $DMAX = 2.0$.

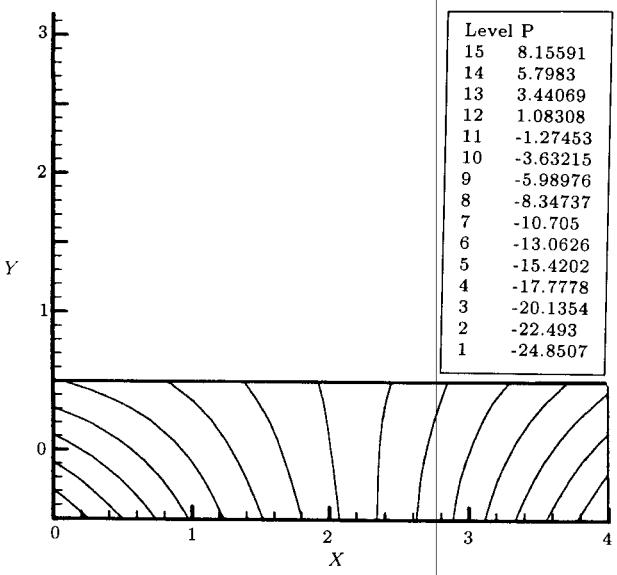


Figure 6. Pressure contours for Poiseuille flow using quadratic basis functions $(m = 6), \alpha = 1.0, DMAX = 2.0$.

pressure solution (see Figure 6). Calculations are also conducted by neglecting either the diffusion term in the stabilization part of formulation (Equation 30), or by considering only the underlined terms for the derivatives in Equations 14 and 15. For both cases, the solutions remained unchanged.

Cavity Flow

The domain and boundary conditions are presented in Figure 7. $\mu = 1.0$ was assumed for this problem. The domain consists of 19×19 regularly distributed points. For this problem, the basis of comparison is

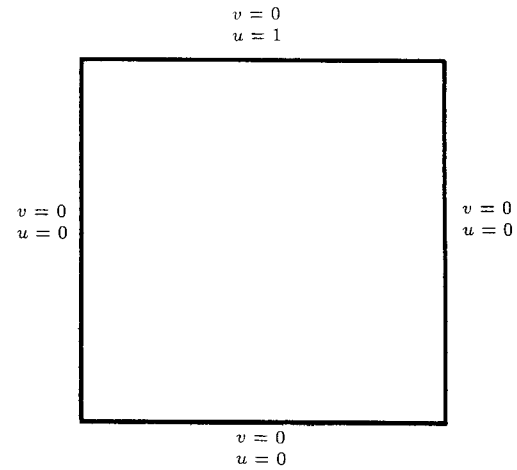


Figure 7. Domain and boundary conditions for driven cavity flow.

the accuracy of the pressure solution at upper corners of the cavity and the conservation of mass. Due to the pressure singularities, higher values of the pressure solution at the top corners indicate a more accurate result. Hence, this is a more challenging problem compared to the plane Poiseuille flow. Note that the stabilization of the pressure term may lead to accuracy degradation of mass conservation. To the best of the authors knowledge, the optimum values of stabilization parameter α are not yet known.

Linear Basis Functions

In Figures 8 to 11, pressure solutions are shown using linear basis function $(m = 3), \alpha = 0.001$ and different $DMAX$ values. As can be seen, good accuracy is obtained for $DMAX = 2$ and 2.5. The pressure at

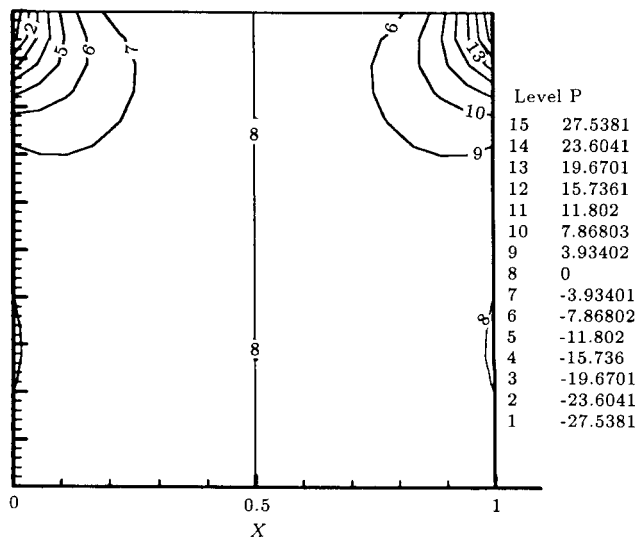


Figure 8. Pressure contours for cavity flow using linear basis functions $(m = 3), \alpha = 0.001$ and $DMAX = 2.0$.

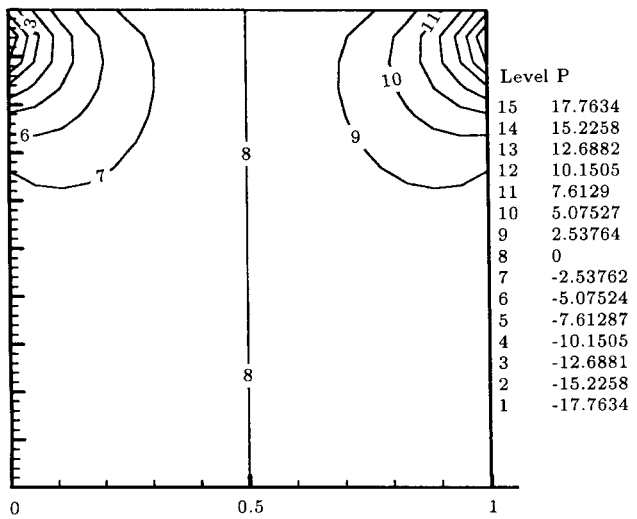


Figure 9. Pressure contours for cavity flow using linear basis functions $(m = 3), \alpha = 0.001$ and $DMAX = 2.5$.

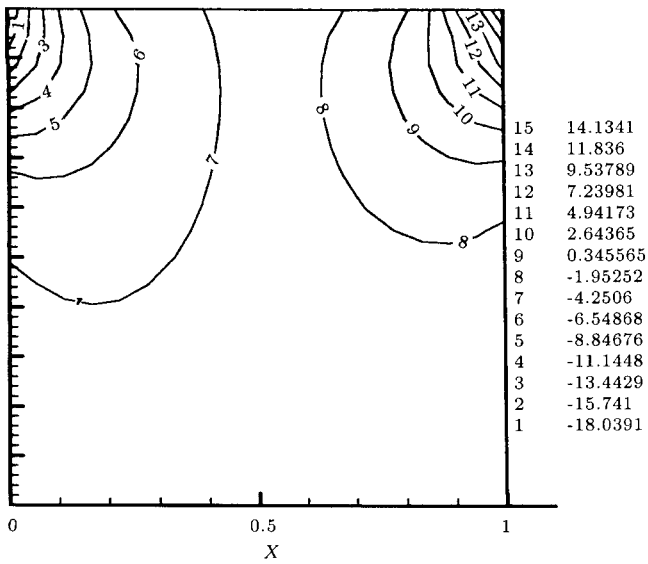


Figure 10. Pressure contours for cavity flow using linear basis functions $(m = 3), \alpha = 0.001$ and $DMAX = 3.0$.

upper corners of the cavity are, respectively, ± 30 for $DMAX = 2$. However, by increasing DMAX to 3, accuracy is considerably decreased. Even slight asymmetry in the solution is observed. Note that the K matrix is not, in general, symmetric. This behavior can be attributed to the points near boundaries which do not see, symmetrically, all points of the domain. For $DMAX = 3$, the pressure decreases to ±16. Interestingly, increasing the $DMAX$ to 4 renders a more accurate and symmetric solution. The solution for $DMAX = 3$ can be returned to the symmetric form by altering the domain of influence of the boundary nodes (see Figure 12). In Figure 13, the streamlines and pressure contour, obtained using $DMAX = 2$ and $m = 3$ for $\alpha = 0.01$ and 0.0001, are shown. As can be seen, by decreasing the α value, the continuity equation

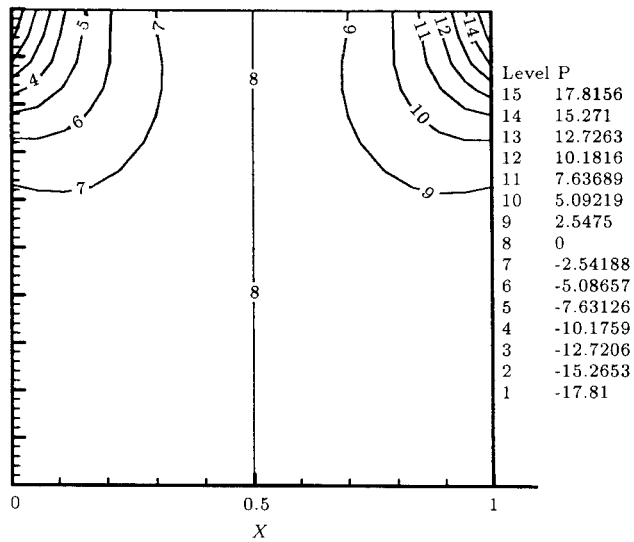


Figure 11. Pressure contours for cavity flow using linear basis functions $(m = 3), \alpha = 0.001$ and $DMAX = 4.0$.

is better satisfied. For $\alpha = 0.0001$, the pressure solution is not very smooth but is more accurate.

Quadratic Basis Functions

In Figures 14 to 18, the pressure contours for $\alpha = 0.0001, m = 6$ (quadratic basis functions) and different $DMAX$ values are shown. As can be seen, even using a very low stabilization parameter, solutions are accurate and symmetric for all $DMAX$ values in the range of 2 to 6. For $DMAX = 2.5$, pressure in the upper corners of the cavity is ± 41 and for $DMAX = 6$ they are ± 30. It should be noted that the solution is less sensitive to the selection of the $DMAX$ value than the $m = 3$ case. Finally, calculations are also conducted by neglecting either the diffusion term in the stabilization part of Equation 30, or by considering only the underlined terms for the derivatives in Equations 14 and 15. For both cases, the solutions remained unchanged.

To put in evidence the mesh-free character of the method, calculations are conducted using grid points distributed in the form shown in Figure 19. In this case, 100 nodes are added to a regular grid of 11 × 11 points. First, the 11 × 11 grid point is solved using $m = 3$ and $DMAX = 2$ and the solution is compared by adding 100 points. The accuracy of the pressure solution is enhanced in this way from ± 8 to ± 11. Note that the addition of points can be performed without any difficulty regarding computer implementation.

In Figure 20, the pressure solution obtained, using $m = 3, DMAX = 2$ and $\alpha = 0.001$ on a domain consisting of 19 × 19 points, is compared to the Galerkin/least-squares FEM formulation [18], employing a mesh system of 20 × 20 Q1Q1 velocity-pressure elements. For FEM and FPM the pressure at the upper corners is, respectively, ± 17 and ± 31. Evidently, the FPM results are superior, based on the pressure values

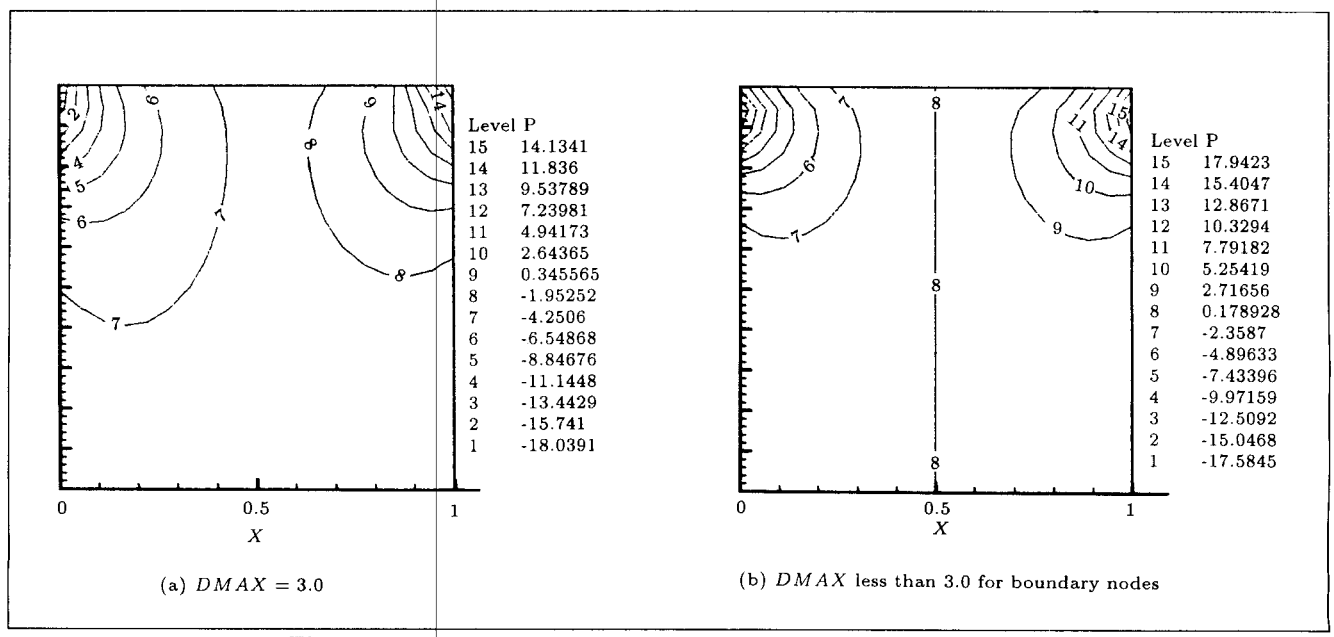


Figure 12. Pressure contours for cavity flow using $m = 3, \alpha = 0.001$.

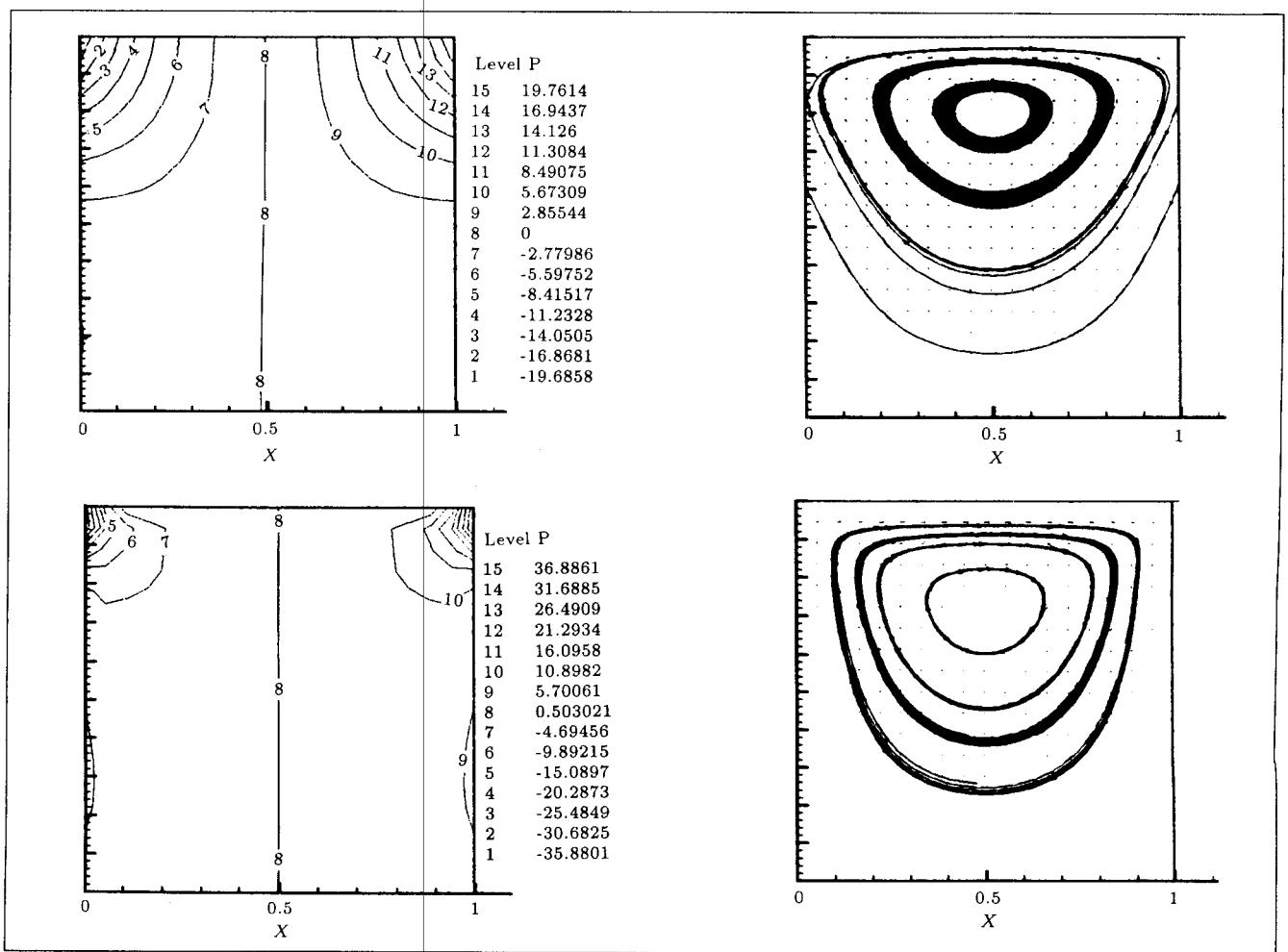


Figure 13. Streamlines and pressure contours for cavity flow using $m = 3.0$ and $DMAX = 2.0$. Upper and lower figures are obtained using $\alpha = 0.01$ and 0.0001, respectively.

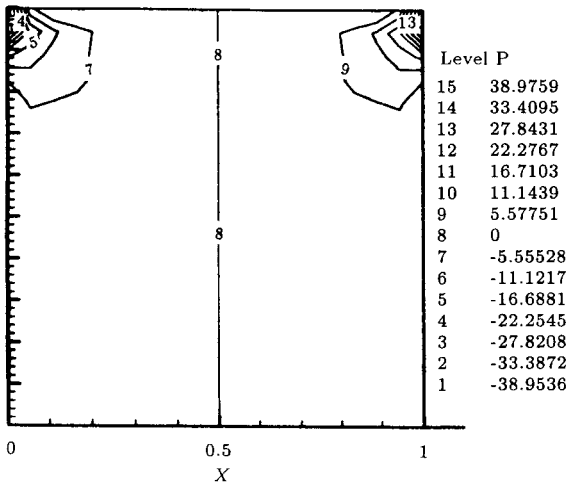


Figure 14. Pressure contours for cavity flow using quadratic basis functions $(m = 6), \alpha = 0.0001$, and $DMAX = 2.0$.

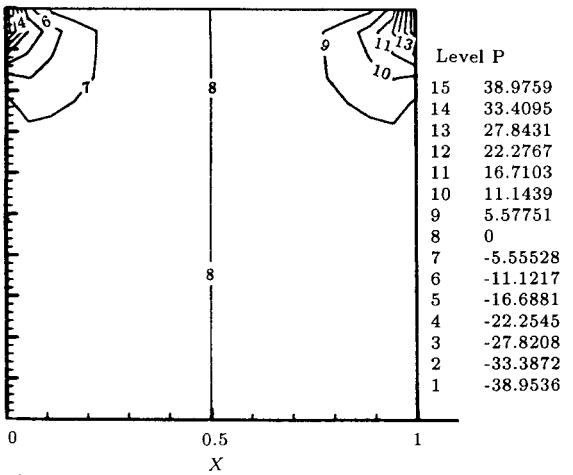


Figure 15. Pressure contours for cavity flow using quadratic basis functions $(m = 6), \alpha = 0.0001$ and $DMAX = 2.5$.

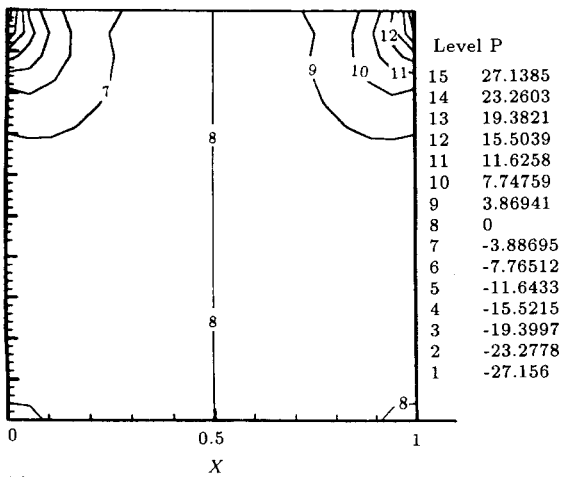


Figure 16. Pressure contours for cavity flow using quadratic basis functions $(m = 6), \alpha = 0.0001$ and $DMAX = 3.0$.

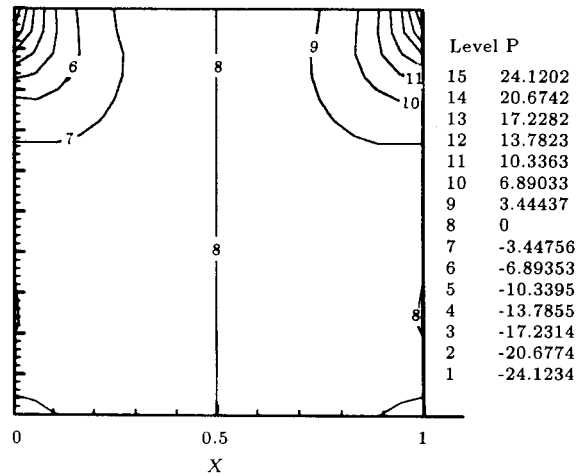


Figure 17. Pressure contours for cavity flow using quadratic basis functions $(m = 6), \alpha = 0.0001$ and $DMAX = 4.0$.

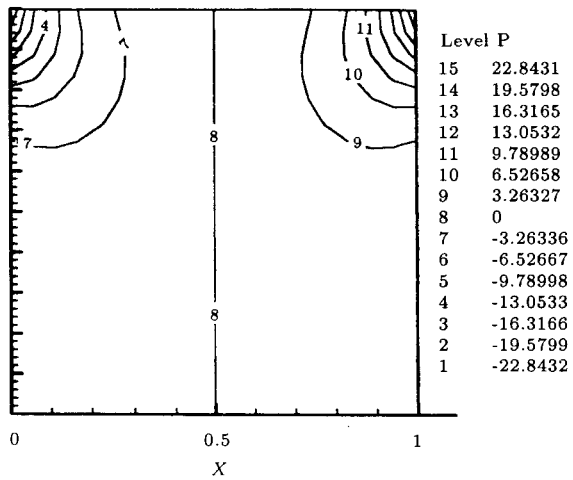


Figure 18. Pressure contours for cavity flow using quadratic basis functions $(m = 6), \alpha = 0.0001$ and $DMAX = 6.0$.

at the upper corners. However, the FPM matrix system is neither symmetric (even though the Stokes equation is symmetric), nor banded. Therefore, this result does not necessarily show the overall computational superiority of FPM results. Obviously, more work should be performed in this context. For example, more efficient solvers adapted for FPM strategies have to be worked out before drawing consistent conclusions regarding the advantages and drawbacks of meshless methods.

CONCLUSION

A finite point method, based on a moving least square interpolation and a point collocation technique, is employed to solve the incompressible Stokes flow. Calculations are conducted using linear and quadratic

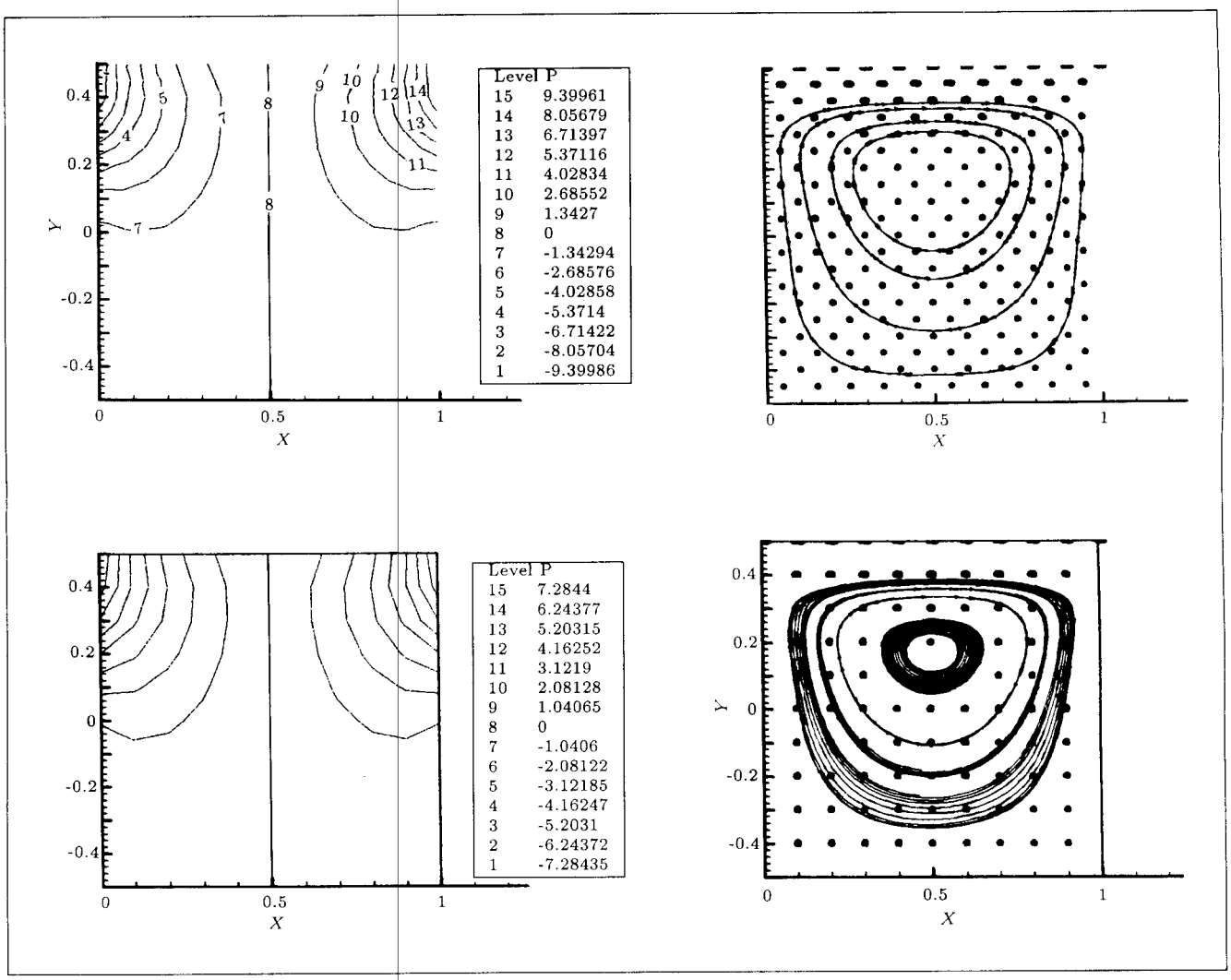
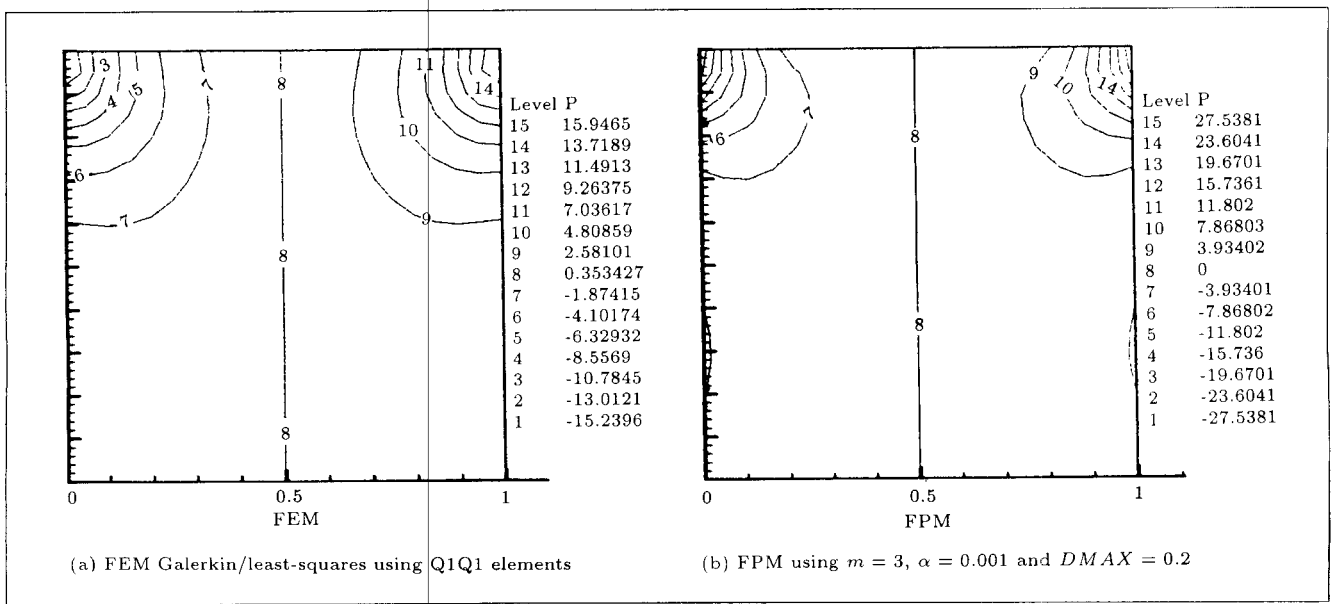


Figure 19. Streamlines and pressure contours for cavity flow using $m = 3, \alpha = 0.001$ and $DMAX = 2.0$. Upper figures are obtained using irregularly distributed nodes.



(a) FEM Galerkin/least-squares using Q1Q1 elements

(b) FPM using $m = 3$, $\alpha = 0.001$ and $DMAX = 0.2$

Figure 20. Pressure contours for cavity flow.

basis functions. The pressure and velocity variables are interpolated employing identical weighting functions (same cloud points) and identical basis functions. The formulation is stabilized by perturbing the continuity equation by the Poisson equation for pressure. In summary, one may conclude:

1. Based on numerical experiments, solutions for the linear case exhibit a large sensitivity to the size of the domain of influence of weighting functions, in contrast to the quadratic basis case.
2. Stabilized methods can be extended to finite point formulation in a straight-forward manner, leading to stable solutions, circumventing staggered type grids needed for velocity and pressure variables.
3. The solution of a full Navier-Stokes equation will be the subject of future work. The main goal of the present work was to study, in detail, certain computational properties of the FPM method, especially its extension to the Stokes equation and its behavior when the pressure term is stabilized. If a complex flow were solved with this new scheme, it would have been very difficult to identify the origin of errors or discrepancies, due to various numerical noises or errors which are naturally present in any approximate numerical scheme. The full capabilities of the scheme, its mesh free character and advantages, with respect to computational effort and execution time, must be studied later.

REFERENCES

1. Nayroles, B., Touzot, G. and Villon, P. "Generalizing the finite element method: Diffuse approximation and diffuse elements", *Comput. Mech.*, **10**, pp 307-318 (1992).
2. Lancaster, P. and Salkauskas, K. "Surfaces generated by moving least squares methods", *Math. Comput.*, **37**, pp 141-158 (1981).
3. Belytschko, T., Lu, Y.Y. and Gu, L. "Element-free Galerkin methods", *Int. J. Num. Methods Engrg.*, **37**, pp 229-256 (1994).
4. Batina, J. "A gridless Euler/Navier-Stokes solution algorithm for complex aircraft applications", **AIAA** 93-0333, Reno, NV, pp 11-14 (Jan. 1993).
5. Onate, E. Idelsohn, S., Zienkiewicz, O.C. and Fisher, T. "A stabilized finite point method for analysis of fluid mechanics problems", *Comput. Methods Appl. Mech. Engrg.*, **139**, pp 315-346 (1996).
6. Duarte, C.A. "A review of some meshless methods to solve partial differential equations", Technical Report 95-06, Texas Institute for Computational and Applied Mathematics, Austin (1995).
7. Liszka, T. and Orkisz, J. "The finite difference method at arbitrary irregular grids and its application in applied mechanics", *Comput. Struct.*, **11**, pp 83-95 (1980).
8. Monaghan, J.J. "An introduction to SPH", *Comput. Phus. Commun.*, **45**, pp 89-96 (1982).
9. Qian, S. and Weiss, J. "Wavelet and the numerical solution of partial differential equations", *J. Comput. Phys.*, **106**, pp 155-175 (1993).
10. Duarte, C.A. and Oden, J.T. "Hp clouds -A meshless method to solve boundary-value problems", Technical Report 95-05, Texas Institute for Computational and Applied Mathematics, Austin (1995).
11. Liu, W.K., Sukkey, Jun, Li, S., Adee, J. and Belytschko, T. "Reproducing kernel particle methods for structural dynamics", *Int. J. Numer. Methods Engrg.*, **38**, pp 1655-1679 (1995).
12. Babuska, I. and Melenk, J.M. "The partition of unity finite element method", *Int. J. Numer. Methods Engrg.*, **40**, pp 727-758 (1997).
13. Onate, E., Idelsohn, S., Zienkiewicz, O.C. and Taylor, R. "A finite point method in computational mechanics. Applications to convective transport and fluid flow", *Int. J. Numer. Methods Engrg.*, **39**, pp 3839-3866 (1996).
14. Babuska, I. "The finite elements with Lagrangian multipliers", *Num. Math.*, **20**, pp 179-192 (1973).
15. Brezzi, F. "On the existence, uniqueness and approximation of saddle point problem arising from Lagrange multipliers", Revue Francaise Automatique Informatique Recherche Operationnelle, Ser. Rouge Anal. Numer., **8**, R-2, pp 129-151 (1974).
16. Sani, R.L., Gresho, P.M., Lee, R.L. and Griffiths, D.F. "The cause and cure of the spurious pressures generated by certain FEM solutions of the incompressible Navier-Stokes equation: Part 1", *Int. J. Num. Methods Fluids*, **1**(1), pp 17-43 (1981).
17. Patankar, S.V., *Numerical Heat Transfer*, McGraw Hill (1980).
18. Hughes, T.J.R., Franca, L.P. and Balestra, M. "A new finite element formulation for computational fluid dynamics: Circumventing the Babuska-Brezzi Condition", *Comput. Methods Appl. Mech, Engrg.*, **59**, pp 85-99 (1986).
19. Douglas, J. and Wang, J. "An absolutely stabilized finite element for the stokes problem", *Math. Comput.*, **52**, pp 495-508 (1989).