Introduction and Comparision of Several Numerical Schemes on Unstructured Triangular Mesh to Solve the Slider Air Bearing Problem of Hard Disk Drives

Lin Wu and D. B. Bogy

Computer Mechanics Laboratory, Department of Mechanical Engineering

University of California, Berkeley, CA 94720

Abstract

In this paper we present and compare three different multi-grid numerical schemes over unstructured triangular mesh that solve the slider air bearing problem of hard disk drives. For each fixed slider attitude the air bearing pressure is obtained by solving the generalized Reynolds equation using one of the three schemes. In the first scheme the convection part of the generalized Reynolds equation is modeled by the "flux difference splitting" technique. Higher order accuracy in space is achieved by a linear reconstruction technique with flux limiters incorporated to prevent oscillation in the high-pressure gradient regions. In the second scheme the convection part is modeled by the multi-dimensional upwind residue distribution approach. In the third scheme the SUPG finite element approach, cast in residue distribution form, is used to model the convection part. In all three schemes, a Galerkin method is used to discretize the diffusion terms. In addition, a non-nested multi-grid iteration technique is used to speed up the convergence rate. Finally, the steady state flying attitude of the slider subject to pre-applied

suspension force and torques is obtained by a Quasi-Newton iteration method, and the results of the current three schemes and the other two schemes are compared.

INTRODUCTION

Although there are several alternative ways to store digital information, the hard disk drive provides the cheapest and most efficient way to store and retrieve data. In today's hard disk drive (Fig. 1), the read-write element is attached to the trailing edge of an air bearing slider, which glides over the rotating disk with a separation (flying height) determined by the balance between the air bearing force generated by the thin squeezed air layer under the slider and the opposing force exerted by a pre-load of the suspension which links the slider and the actuator (Fig. 2). To increase the storage density and reduce the signal noise, it is desirable to achieve a low and uniform steady state flying height across the disk. The design goal of the next generation hard disk drive is to reach an areal density of 100 Gbit/in^2, which requires a flying height between 5 and 10 nm. In the hard disk drive industry the above flying attitude goal is obtained by carefully designing the rail shape of the slider, which is generally very complicated. To reduce the design cost, accurate and efficient steady state air bearing design software is required.

In the slider's manufacturing process the etch technique leaves a steep narrow wall profile region (a few microns wide for sliders with millimeter length scale) along the rail boundaries. The extension and shape of the wall profiles have profound influence on the flying attitude. As a result, it is important to capture these regions of rapidly changing elevations in the numerical model. In addition, the air bearing pressure field is characterized by irregularly distributed

regions with very high-pressure gradient. To get accurate results fine enough grids are needed to cover the above regions. Under these situations unstructured grids prove to be an economical and convenient way of decomposing the computational domain.

In our previous papers [1] [2] [3], we presented triangular mesh generation and refinement and adaptation techniques that suit the slider air bearing problem based on the Delaunay method [4] [5] and the longest-side-bisection Delaunay refinement technique [6]. An implicit non-nested multi-grid finite volume scheme was also constructed over this mesh. The numerical scheme gives comparable results with much less simulation time to these of an older code developed in the Computer Mechanics Laboratory [7], which has been widely accepted in the hard disk drive industry. But the new numerical method has two potential shortcomings. First, the scheme is an extension of the numerical schemes listed in [8] for convection and diffusion type equations on a rectangular Cartesian mesh to the unstructured triangular mesh. These schemes are quite diffusive if the convection is strong, which is true in the slider air bearing problem. Second, the control volumes of the new scheme are the dual Voronoi Polygons of the Delaunay triangulation. This requires the quality of the mesh to be extremely good, otherwise the Voronoi polygon can be highly distorted.

To improve the above potential shortcomings, in this paper we present three different numerical schemes that use the same meshes as our previous approach but take the median dual of the triangular mesh as the control volume. The median dual is different from the dual Voronoi polygons; it is obtained by connecting the neighboring centroids of the triangles. By the adoption of a less diffusive spatial discretization, we also hope to improve the accuracy of the numerical schemes themselves without significantly increasing the simulation time. In the new approaches, we use the “flux difference splitting” (FDS) technique in [9], the positive

streamwise invariant (PSI) residue distribution approach in [10][11] and the streamline upwind Petrov-Galerkin (SUPG) finite element scheme [12] cast as a residue distribution formula [13] to discretize the convection part of the generalized Reynolds equation, while a standard Galerkin type method [14] is used to model the diffusion part of the equation. A non-nested multi-grid technique based on the full storage approximation multi-grid strategy of Brandt [15] is implemented to improve the convergence rate of the Gauss-Seidel smoother used to solve the non-linear discretized equation. Mavriplis and Jameson's restriction and interpolation functions [16][17] that suit non-nested triangular meshes are used to transfer the variables and residues between the meshes.

Finally, the steady state flying height of the slider is found by a Quasi-Newton iteration method fully described in Dennis and Schnabel [18].

GOVERNING EQUATIONS AND BOUNDARY CONDITIONS

In air bearing simulation the generalized Reynolds equation is solved to get the pressure field. Because of the extremely narrow spacing between the slider and the disk (on the order of 10 nm, which is only a fraction of the mean free path of the gas molecules), the gas in the spacing is extremely rarefied and the gas molecules near the solid surfaces no longer simply adhere to them, but instead there is slipping. The usual continuity and non-slip condition assumptions are no longer an acceptable approach to the actual physics. Until now, the modified versions of the Reynolds equation that take the rarefaction and slipping effect into account give the most agreeable results with those of experiments and direct Montecarlo simulation [19-23]. The modification makes the equation appear more complicated, but from the numerical point of

view, it also gives the equation a better numerical quality than that of the traditional Reynolds equation, since the pressure fields given by the modified versions have less steep pressure profiles than those predicted by the traditional equation. The different versions of the Reynolds equation can be written in a unified dimensionless from as

$$\sigma \frac{\partial}{\partial T}(PH) = \frac{\partial}{\partial X}\left(QPH^3 \frac{\partial P}{\partial X} - \Lambda_x PH \right) + \frac{\partial}{\partial Y}\left(QPH^3 \frac{\partial P}{\partial Y} - \Lambda_y PH \right) \tag{1}$$

where $\sigma = 12\mu\omega L^2 / P_a h_m^2$ is the squeeze number, which is a constant representing the relative importance between the unsteady effect and the diffusion effect, where μ is the dynamic viscosity of the gas, ω is the angular velocity of the disk, L is the length scale of the slider (taken to be the length of the slider), p_a is the ambient pressure, h_m is the flying height of a reference point on the slider (usually taken to be the point at the trailing edge center of the slider with zero recess). $T = \omega t$ is the dimensionless time, $X = x/L$ is the dimensionless x coordinate, $Y = y/L$ is the dimensionless y coordinate, $P = p/p_a$ is the dimensionless pressure, $H = h/h_m$ is the dimensionless normal distance from the disk to the slider, $\Lambda_x = 6\mu UL / p_a h_m^2$ and $\Lambda_y = 6\mu VL / p_a h_m^2$ are the bearing numbers in the x and y directions, respectively, which represent the relative importance between the convection effect and the diffusion effect. U and V are the disk velocity components in the x and y direction. Q is the flow factor, which marks the difference between different rarefaction and slip models of the equation. Different Q for different models are briefly listed below. The details can be found in Burgdorfer [19], Hsia and Domoto [20] and Fukui and Kaneko [21].

$Q = 1$, continuum model.

$Q = 1 + 6aK_n / PH$, first order slip model.

$Q = 1 + {6K_n}/{PH} + 6\left({K_n}/{PH}\right)^2$, second order slip model.

$Q = f\left({K_n}/{PH}\right)$ Fukui-Kaneko model.

In these expressions $a = \frac{2-a}{a}$, α is the accomodation factor, $K_n = \frac{l}{h_m}$ is the Knudsen number, and λ is the mean free molecular path. In our simulation the Fukui-Kaneko model is used, which is believed to give the best results among different models. In the implementation of the model, the database in [21] is used to find the flow factor.

Along the outside boundary of the slider, the pressure is simply taken as the ambient pressure.

MESH GENERATION

Because of its geometric flexibility in constructing a quality mesh around complex configurations and the relative convenience of incorporating an adaptive methodology, and also its efficiency and its ability to generate optimal connections to existing node points, the so called Delaunay method as described in [4] is used as a building block in our approach for the following incremental mesh refinement and adaptation. Using different refinement techniques, we generate three sets of good quality unstructured triangular meshes. For the coarsest mesh, it's important that the boundaries of the rails be represented in the triangulation. The conforming Delaunay refinement technique in Ruppert [5] is used to generate the coarsest conformed background mesh. The longest-side bisection Delaunay refinement technique in Rivara and Inostroza [6] is adopted to cluster fine meshes in the recess wall regions with rapid geometric

change based on geometric considerations (the maximum recess depth difference in each triangle must be smaller than a prescribed value), which forms the second finer mesh. For the mesh adaptation, the maximum undivided pressure difference in a triangle is used to decide whether the triangle needs further refinement. The same longest-side bisection Delaunay refinement technique is used to refine the mesh. Details can be found in our previous papers [1][2].

THE FDS UPWIND DISCRETIZATION OF THE COVECTION EQUATION

Ignore the diffusion terms in equation (1) for the moment and divide all terms of the equation by s. After integrating the convection equation over the median dual Ω (with an area A_Ω) around each vertex, we can rewrite the equation as

$$A_\Omega \frac{\partial}{\partial T}(PH) + \oint_{\partial\Omega}\left[\frac{\Lambda_x}{s} n_x PH + \frac{\Lambda_y}{s} n_y PH\right] dl = 0. \tag{2}$$

In discretized form this can be writen as

$$\frac{A_\Omega}{\Delta T}\left[(PH)_i^{n+1} - (PH)_i^n\right] + \sum_{j=1}^{M} F_{ij} \cdot \Delta l_{ij} = 0, \tag{3}$$

where Δl_{ij} is the distance between the two centroids j' and $(j+1)'$ (Fig. 3). F_{ij} is the numerical flux across Δl_{ij} given by Roe's flux difference splitting technique [9] as

$$F_{ij} = \frac{1}{2}\left[F\left((PH)_L\right) + F\left((PH)_R\right) - \left|\bar{I}\right|\left((PH)_R - (PH)_L\right)\right]. \tag{4}$$

$(PH)_L$ and $(PH)_R$ are the left and right state values of PH to the middle of the line ij. $|\overline{l}|$ is the average wave speed defined as

$$2s|\overline{l}| = \left|(\Lambda_n)_i + (\Lambda_n)_j\right|. \tag{5}$$

$(\Lambda_n)_i$ and $(\Lambda_n)_j$ are the projections of the bearing number vector (Λ_x, Λ_y) evaluated at the vertices i and j, respectively, onto the outward normal vector $\underline{n}_{ij}$ of the boundary line connecting j' and $(j+1)'$. The bearing number vector is a continous known function of space only under a fixed flying attitude. Due to the small size of the slider compared with the radius, the bearing number vector changes slowly across the slider. F is the flux function defined as

$$sF(PH) = (\Lambda_x, \Lambda_y)\cdot \underline{n}_{ij}(PH). \tag{6}$$

Temporally, it can be linearized as

$$F(PH)^{n+1} = F(PH)^n + \frac{\partial F}{\partial(PH)}\left((PH)^{n+1} - (PH)^n\right). \tag{7}$$

It can be easily shown that the projection of the bearing vector along each line on the disk is a constant, which means the x and y components of the bearing number vector, Λ_x and Λ_y commute with the derivative operator. As a result

$$s\frac{\partial F}{\partial(PH)} = (\Lambda_x, \Lambda_y)\cdot \underline{n}_{ij} = \Lambda_n. \tag{8}$$

The final upwind scheme can be written as

$$\left[\frac{A_\Omega}{\Delta T}+\sum_{j=1}^{M}\lambda^{+}\Delta l_{ij}\right](PH)_i^{n+1}+\sum_{j=1}^{M}\lambda^{-}\Delta l_{ij}(PH)_j^{n+1}=$$
$$\left[\frac{A_\Omega}{\Delta T}+\sum_{j=1}^{M}\lambda^{+}\Delta l_{ij}\right](PH)_i^{n}+\sum_{j=1}^{M}\lambda^{-}\Delta l_{ij}(PH)_j^{n} \tag{9}$$
$$-\frac{1}{2}\sum_{j=1}^{M}\left\{F\left((PH)_L\right)+F\left((PH)_R\right)-\left|\bar{\lambda}\right|\left((PH)_R-(PH)_L\right)\right\}\Delta l_{ij},$$

where

$$2\sigma\lambda^{+}=(\Lambda_n)_i+\left|(\Lambda_n)_i\right|, \tag{10}$$

and

$$2\sigma\lambda^{-}=(\Lambda_n)_j-\left|(\Lambda_n)_j\right|. \tag{11}$$

If $(PH)_L$ and $(PH)_R$ are evaluated at vertices i and j, then the space discretization has only first order accuracy. Here the linear reconstruction technique in [24] is adopted to achieve a higher order space accuracy. The left and right state variables can be evaluated as

$$(PH)_L=(PH)_i+\Phi_i\nabla(PH)_i\cdot\underline{\Delta r_{ij}}, \tag{12}$$

and

$$(PH)_R=(PH)_j-\Phi_j\nabla(PH)_j\cdot\underline{\Delta r_{ij}}, \tag{13}$$

where $\nabla(PH)$ is the average gradient of PH at vertex i or j, which can be evaluated by use of a simple Green's theorem or least square approach [25]. In our calculation, the Green's theorem method is used. $\underline{\Delta r_{ij}}=\frac{1}{2}(\underline{r}_j-\underline{r}_i)$ is the vector pointing from i to j with half its length. To avoid the appearance of new local maximum or minimum after the reconstruction procedure, the variable Φ ranging from 0 to 1 is used to limit the reconstructed left and right hand state variables, which is determined by using the same method as in [24].

THE RESIDUE DISTRIBUTION DISCRETIZATION OF THE COVECTION EQUATION

In the residue distribution schemes in [10][11], the residual in triangle T is defined as

$$\begin{aligned} \boldsymbol{f}^T &= -\iint_T \frac{\partial(PH)}{\partial T} dA = \iint_T \left(\frac{\Lambda_x}{\boldsymbol{s}}, \frac{\Lambda_y}{\boldsymbol{s}} \right) \cdot \nabla(PH) dA = A_T \, \overline{\boldsymbol{l}} T \cdot \nabla(PH) \\ &= \sum_{j=1}^{3} k_j^T (PH)_j \end{aligned}, \tag{14}$$

where $\boldsymbol{s}\,\overline{\boldsymbol{l}}_T = \overline{(\Lambda_x, \Lambda_y)}$ is the averaged wave speed in each triangle. The conservation constraint gives

$$\overline{\boldsymbol{l}}_T = \frac{1}{A_T \boldsymbol{s}} \iint_T (\Lambda_x, \Lambda_y) dA = \frac{1}{3\boldsymbol{s}} \left[(\Lambda_x, \Lambda_y)_1 + (\Lambda_x, \Lambda_y)_2 + (\Lambda_x, \Lambda_y)_3 \right]. \tag{15}$$

In the above formulation, the bearing number is assumed to vary linearly in each triangle. The inflow parameter k_j^T is defined as

$$2k_j^T = \overline{\boldsymbol{l}}_T \cdot \overline{n_j}, \tag{16}$$

where $\overline{n_j}$ is the inward normal of each edge of the triangle with a magnitude equal to its length (see Fig. 4). In the air bearing problem k_j^T only needs to be calculated once and stored for later use.

The residual of each triangle is sent to its three nodes by the distribution coefficient $\boldsymbol{b}_i^T$

$$\boldsymbol{f}_i^T = \boldsymbol{b}_i^T \boldsymbol{f}^T. \tag{17}$$

The convection equation can be discretized as

$$\frac{A_\Omega}{\Delta T} \left[(PH)_i^{n+1} - (PH)_i^n \right] = -\sum_{T=1}^{M} \boldsymbol{f}_i^T = \sum_{T=1}^{M} \boldsymbol{b}_i^T \boldsymbol{f}^T = \sum_{T=1}^{M} \boldsymbol{b}_i^T \sum_{j=1}^{3} k_j^T (PH)_j^{n+1}. \tag{18}$$

In the above formulation the solution is updated by accumulating the residues at node i triangle by triangle. To put all the schemes in a unified form, it can be re-formulated such that the solution can be updated edge by edge

$$\begin{aligned}\frac{A_\Omega}{\Delta T}\left[(PH)_i^{n+1}-(PH)_i^{n}\right] &= \sum_{T=1}^{M}\boldsymbol{b}_i^{T}\sum_{j=1}^{3}k_j^{T}(PH)_j^{n+1}\\ &= \sum_{j=1}^{M}\left[\left(\boldsymbol{b}_i^{T1}k_i^{T1}+\boldsymbol{b}_i^{T2}k_i^{T2}\right)(PH)_i^{n+1}+\left(\boldsymbol{b}_i^{T1}k_j^{T1}+\boldsymbol{b}_i^{T2}k_j^{T2}\right)(PH)_j^{n+1}\right],\end{aligned} \tag{19}$$

where $T1$ and $T2$ are the two neigboring triangles sharing the edge ij.

For the PSI scheme [10][11], the distribution coefficient can be written as

$$\boldsymbol{b}_i^{T} = MinMod\left[-\frac{\max\left(0,k_i^{T}\right)\min\left(0,\left((PH)_i-(PH)_{in}\right)\boldsymbol{f}^{T}\right)}{\boldsymbol{f}^{T}}\right], \tag{20}$$

where $(PH)_{in}$ is the linearly interpolated PH value at the inflow point, which can be evaluated as

$$(PH)_{in} = \frac{\sum_{j=1}^{3}\min\left(0,k_j^{T}\right)(PH)_j}{\sum_{j=1}^{3}\min\left(0,k_j^{T}\right)}. \tag{21}$$

For the SUPG finite element scheme [12] in residue distribution form [13], the distribution coefficient can be written as

$$\boldsymbol{b}_i^{T} = \frac{1}{3}+0.5\frac{h}{\left|\overline{\boldsymbol{l}_T}\right|A_T}k_i^{T}. \tag{22}$$

The length scale h can be approximated as

$$h = \max_{i=1}^{3}\left(\left|(n_i)_x\right|,\left|(n_i)_y\right|\right). \tag{23}$$

THE FINITE ELEMENT DISCRETIZATION OF THE DIFFUSION TERM

The diffusion term can be discretized with a Galerkin weighted integral [14]

$$\iint_{A_\Omega}\left[\nabla\cdot\left(QPH^3\frac{\partial P}{\partial x},QPH^3\frac{\partial P}{\partial y}\right)\right]dA=\sum_{j=1}^{M}W_{ij}\left[(P)_j^{n+1}-(P)_i^{n+1}\right]. \tag{24}$$

Here A_Ω is the median dual, and the weights W_{ij} are defined as

$$2\sigma W_{ij}=\overline{QPH^3}^{L}\cot an(\alpha_{Lj})+\overline{QPH^3}^{R}\cot an(\alpha_{Rj}), \tag{25}$$

where $\overline{QPH^3}^{L}$ and $\overline{QPH^3}^{R}$ are the average values of QPH^3 in the triangle to the left or right of the edge ij. α_{Lj} and α_{Rj} are the opposite angles to the edge ij as shown in Fig.3.

ITERATIVE SOLVER OF THE DISCRETIZED EQUATIONS

All the above numerical schemes can be written in a unified form as

$$C_iP_i^{n+1}+\sum_{j=1}^{M}C_{ij}P_j^{n+1}=S_i\left(P_i^n,P_j^n\right). \tag{26}$$

For the FDS scheme

$$C_i=\left[\frac{A_\Omega}{\Delta T}+\sum_{j=1}^{M}\lambda^{+}\Delta l_{ij}\right]H_i+\sum_{j=1}^{M}W_{ij}, \tag{27}$$

$$C_{ij}=\lambda^{-}\Delta l_{ij}H_j-W_{ij}, \tag{28}$$

$$\begin{aligned}S_i\left(P_i^n,P_j^n\right)=&\left[\frac{A_\Omega}{\Delta T}+\sum_{j=1}^{M}\lambda^{+}\Delta l_{ij}\right](PH)_i^n+\sum_{j=1}^{M}\lambda^{-}\Delta l_{ij}(PH)_j^n\\&-\frac{1}{2}\sum_{j=1}^{M}\left\{F\left((PH)_L\right)+F\left((PH)_R\right)-\left|\overline{\lambda}\right|\left((PH)_R-(PH)_L\right)\right\}\Delta l_{ij}\end{aligned}. \tag{29}$$

For schemes in residue distribution form

$$C_i = \frac{A_\Omega}{\Delta T} H_i + \sum_{j=1}^{M} \left(\boldsymbol{b}_i^{T1} k_i^{T1} + \boldsymbol{b}_i^{T2} k_i^{T2} \right) H_i + \sum_{j=1}^{M} W_{ij} \,, \tag{30}$$

$$C_{ij} = -\left(\boldsymbol{b}_i^{T1} k_j^{T1} + \boldsymbol{b}_i^{T2} k_j^{T2} \right) H_j - W_{ij} \,, \tag{31}$$

$$S_i \left(P_i^n, P_j^n \right) = \frac{A_\Omega}{\Delta T} \left(PH \right)_i^n \,. \tag{32}$$

Equation (26) is still nonlinear, because C_i and C_{ij} depend on P. One simple way to linearize the equation is to take P as the most recent known value of the last iteration, this is the so-called lagging technique. The resulting simultaneous equations are solved by a two sweep point Gauss-Seidel method. The first sweep starts from the beginning of the vertex list, and the second sweep starts from the end of the list. This takes into account the fact that the diffision terms in the Reynolds equation are elliptic in nature, and disturbance information is spread simultaneously in all directions.

The steady state solution for one fixed attitude is found by marching in time. For steady state problems with fixed attitude, the unsteady term is not needed physically, but it is kept here to serve as an under-relaxation term. When a relatively large time step is used, the unsteady term can be ignored, and the technique is more like a direct iteration than time marching. The implicit schemes are unconditionally stable, so an arbitrarily large CFL number such as 1.0E12 can be used. At the beginning of each time step the coefficents are updated once and stored using the solution of the previous time step, and the resulting linear algebraic equations are solved by a fixed number of two sweep Gaussian-Seidel iterations. In our code, about ten to twenty Gauss-Seidel iterations are used to find an approximate solution. The choice of the number of iteration corresponds to optimized overall convergence speed for some sliders.

The boundary condition is very simple; along the outside boundaries of the slider, the pressure is ambient.

GRID TRANSFER OPERATORS FOR THE MULTI-GRID ALGORITHM

In the implementation of the multi-grid algorithm the variables and residues are transferred frequently between different mesh levels. The transfer procedure has vital influence on the overall performance of the multi-grid algorithm. Mavripilis and Jameson's [16][17] grid transfer operators have been shown to be well suited for multi-grid algorithms over unstructured triangular meshes. Here we simply adopt their operators.

Let I_{k+1}^{k} be the operator used to transfer variables or residues from the fine mesh k+1 to the coarse mesh k, it is also called the restriction or projection operator. If I_{k+1}^{k} is operating on variables, the operation can simply be taken as a linear interpolation of the variables from the fine mesh nodes to the coarser mesh nodes. If it operates on the residue, then the residue at the vertex of a finer mesh can be distributed to the three vertices of the coaser triangle that encloses the vertex by its three area coordinates. This can guarantee the conservation of the residue in the transfer process.

Let I_{k}^{k+1} be the operator used to transfer corrections from the coarse mesh k to the fine mesh k+1, it is also called the interpolation or prolongation operator. It can simply be taken as a linear interpolation.

FAS MULTI-GRID ALGORITHM

The full approximation storage (FAS) algorithm in Brandt [15] is well suited for non-linear equations. It solves the equations by iterating over several sets of mesh. In abstract form it can be presented as follows. To simplify the expression we assume only two levels of mesh are used (k+1 represents the fine mesh and k represents the coarse mesh). Let L be the differential operator, $\underline{U}$ be the unknown vector and $\underline{F}$ be the source term, then the differential equation can be written as

$$L\underline{U} = \underline{F}. \tag{33}$$

On the fine mesh, the equation can be discretized as

$$L_{k+1}\underline{U} = \underline{F}_{k+1}. \tag{34}$$

A certain number of iterations can be carried out until the convergence rate becomes slower, then the solution $\underline{u}^{k+1}$ and the residue $\underline{F}_{k+1} - L_{k+1}\underline{u}^{k+1}$ are transferred to the coarse mesh, and the following equation

$$L_k \underline{U} = I_{k+1}^k(\underline{F}_{k+1} - L_{k+1}\underline{u}^{k+1}) + L_k\left(I_{k+1}^k \underline{u}^{k+1}\right) \tag{35}$$

is solved there with an initial guess $I_{k+1}^{k}\underline{u}^{k+1}$. A approximate solution $\underline{u}^{k}$ is found after enough iterations. Finally the correction on mesh k is transferred back to k+1, and the solution on mesh k+1, $\underline{u}^{k+1}$ is updated as

$$\underline{u}^{k+1} \leftarrow \underline{u}^{k+1} + I_{k}^{k+1}\left(\underline{u}^{k} - I_{k+1}^{k}\underline{u}^{k+1}\right), \tag{36}$$

which serves as the initial guess of the next multi-grid circle iteration. The above process continues until the error drops to an acceptible level.

For our case, the differential operator on level k+1can be written as

$$\left(L_{k+1}\underline{P}\right)_i = C_i P_i^{n+1} - \sum_{j=1}^{M} C_{ij} P_j^{n+1}. \tag{37}$$

Similar expressions hold for the other levels. The source term on the fine mesh k+1 is

$$\left(\underline{F}_{k+1}\right)_i = S_i\left(P_i^n, P_j^n\right). \tag{38}$$

On the coase level k, it can be written as

$$\underline{F}_k = I_{k+1}^{k}(\underline{F}_{k+1} - L_{k+1}\underline{u}^{k+1}) + L_k\left(I_{k+1}^{k}\underline{u}^{k+1}\right). \tag{39}$$

In our actual implementation three levels of mesh are used. Figure 5 shows the multi-grid V cycles used in the simulation. To get a good initial guess we first do forty iterations on the coarsest mesh, then we linearly interpolate the solution variables to the second level mesh.

Twenty iterations are carried out there before we transfer the solution variables to the third level mesh. After that, N V cycles are carried out before we adaptively refine the third level mesh according to the pressure distribution. The following V cycles are performed over the new finest mesh and the other two meshes until convergence is achieved. The number of iterations on each mesh level corresponds to optimized convergence speed for some sliders.

INVERSE PROBLEM

In air bearing simulation the steady state flying attitude corresponding to a fixed prescribed load is more important than the pressure distribution of the steady state solution of one fixed attitude, because it is the former that is prescribed in the design of hard disk drives. The steady state flying attitude is defined as the one at which the pre-enforced suspension force and pitch and roll torques are balanced by their counterparts generated by the air bearing forces and moments that are functions of flying attitude. We can define a vector $\underline{R} = (R_1, R_2, R_3)$, where

$$R_1 = F_{air} - F_s, \tag{40}$$

$$R_2 = \frac{(M_{air})_p + (M_s)_p + (M_{shear})_p}{F_{air}} - X_0, \tag{41}$$

$$R_3 = \frac{(M_{air})_r + (M_s)_r + (M_{shear})_r}{F_{air}} - Y_0, \tag{42}$$

where F_{air} is the air bearing force of a certain fixed attitude, F_s is the applied suspension force, M_{air}, M_s and M_{shear} are moments caused by air bearing pressure at a fixed attitude, the applied

suspension force and viscous shear force, respectively. Subscripts p and r represent the projection in the pitch and roll directions (see Fig. 2). X_0 and Y_0 are coordinates of the position where the suspension force is applied. $\underline{R}$ is a non-linear function of the flying height, the pitch angle and the roll angle. The object is to find a particular flying attitude that makes $\underline{R}$ zero, which corresponds to the steady state flying attitude. The Quasi-Newton iteration method for non-linear problems fully described in Dennis and Schnabel [18] is implemented to find the steady state attitude. Our experience shows that generally only a few Newton steps are needed to find the steady state solution depending on the initially assumed values.

RESULTS AND DISCUSSION

Figures 6 and 7 depict the slider used in the IBM Travelstar 25 Gbit hard disk drive. The length and width in the figures have been normalized by the dimensional length (1mm). Figures 8, 9 and 10 show one set of three initial meshes for the IBM slider. Figure 11 shows one adaptively refined third level mesh. Figure 12 shows the comparison of the convergence history of the iterations on a single set of mesh and the multi-grid iteration. The 'S' at the legend end represents the iteration on a single set of mesh, while the 'M' represents multi-grid iteration. The convergence difference among different schemes is very small, and it is almost undetectable for the single mesh iteration from the figure. Pat2 is the Patankar scheme extended to triangular mesh [3]. The sudden error jump in the figure corresponds to mesh adaptation. From the figure it can be seen that for the single mesh iteration, the error initially drops very fast, only ten iterations are needed to bring the error down from about 10^{-2} to 10^{-4}. But after the high frequency error has been smoothed out, the curve flattens. It takes about another 140 iterations to further

reduce the error by about two orders of magnitude. The multi-grid curve shows that all error components can be continuely and efficently romoved. The log error drops almost linearly with the number of iterations (time steps). The figure also shows the multi-grid technique works well for all the schemes on the triangular mesh. For this particular slider only 20 multi-grid cycles are needed to get the converged solution. More than one order of magnitude simulation time is saved by use of the multi-grid technique.

Figures 13, 14, 15, 16 and 17 show the pressure contours at the steady state attitude obtained by the rectangular mesh solver [7] and the four triangular mesh solvers, respectively. The disk is rotating at 4500 RPM, and the slider is located at a 15 mm radial position with zero skew angle. The prescribed suspension force is 1.5 g at the center of the slider, the prescribed suspension torque is zero. The pressure contours differ only by very small details. From the above figures we can see that all the regions with large geometric changes or pressure gradients have been efficiently captured by the mesh generation and adaptation techniques.

Figure 18 shows the flying height grid convergence comparison between the triangular mesh solvers and the rectangular mesh solver. The difference among the grid converged flying height solution of all the schemes is less than 1 nm. All schemes show a trend to converge to a 16 nm flying height. The PSI and SUPG schemes are less diffusive than the other schemes, which predict a slightly higher flying height. Figures 19 and 20 show the pitch angle and roll angle grid convergence histories respectively. For the node number corresponding to flying height convergence, all codes reach grid convergence. Figure 21 shows a plot of the simulation time for finding the steady state attitude as a function of the grid size. For the same number of node points, the triangular mesh solvers cost more time than the rectangular solver. But this is not always true. The rail shape of the IBM slider is extremely regular. In this case, the rectangular

mesh solver is expected to do a good job. But the triangular mesh solvers will have more advantage than the rectangular mesh solver when the geometry becomes more complicated.

To demonstrate this, the NSIC slider (Fig. 22 and 23 with a characteristic length scale of 1 mm) is simulated. Figures 24, 25, 26 and 27 show one set of meshes used in the multi-grid iteration. Figure 28 shows the flying height grid convergence history. The disk is rotating at 7200 RPM, and the slider is located at a 15 mm radial position with a -1.22 *mRad* skew angle. The prescribed suspension force is 1.5 g at the center of the slider, the prescribed suspension torque is zero. Figure 28 shows the grid convergence history, again the multi-grid technique greatly improved the convergence. Only 7 multi-grid cycles are needed to achieve convergence. Figures 29, 30, 31, 32 and 33 show the pressure contours corresponding to steady state flying height for the different schemes. Figure 34 shows the flying height grid convergence comparison between the triangular mesh solvers and the rectangular mesh solver. The PSI and SUPG predict a grid converged flying height around 8 nm, while the other schemes predict a flying height around 7 nm, but all of them still have a trend to fly higher with more grid points added. This clearly shows that the PSI and the SUPG scheme introduce less numerical diffusive effects than the other schemes and are assumed to be more accurate. Figure 35 and 36 are the grid convergence history for the pitch and roll angles respectively. Figure 37 shows the simulation time used to find the steady state flying attitude as a function of node number. This time, at the beginning, when the node number is the same, the triangular mesh solver uses almost the same or less time as the rectangular mesh solver. When the grid size increases, the times needed by the PSI and the SUPG schemes increase faster than by the rectangular mesh solver. But this is due to the nature of the non-nested multi-grid technique we used. The current technique does not require any relationship between the different mesh levels, however the relationship has

influence on the performance of the multi-grid iteration. The mesh relationship is decided by several control parameters in our grid generation process [2]. These parameters are optimized to the grid size in the beginning range. When a grid size is far out of that range, a new set of parameters is needed to give optimized performance. Nevertheless from the above grid convergence figures, we can see that, the PSI and the SUPG schemes achieve grid convergence within that range for sliders as complicated as the IBM and the NSIC sliders.

Figure 38 shows the steady state flying heights of the NSIC slider at three radial positions obtained by the five numerical schemes and the preliminary experiments of NSIC. The results predicted by all numerical schemes are quite close to the experimental data. For this particular slider, all other schemes except the PSI and SUPG schemes have a tendency of under-predicting the flying height. We believe this is due to the more diffusive nature of these schemes compared with the PSI and SUPG schemes.

SUMMARY AND CONCLUSIONS

Three different schemes are used to discretize the convection part of the generalized Reynolds equation on unstructured triangular mesh, and a Galerkin finite element approach is used to model the diffusion part of the equation. The resulting numerical schemes are shown to be unconditionally stable. A non-nested FAS multi-grid algorithm has been successfully employed to speed up the convergence rate of the above schemes. The multi-grid algorithm requires no relationship between different mesh levels. On the average, nearly one order of simulation time is saved by implementing the multi-grid algorithm. In addition the steady state flying attitude is found by a Quasi-Newton method. Even though the unstructured nature of the

grid makes the situation much more complicated than it is for the structured rectangular mesh, and all the information can be stored and retrieved only through a complicated data structure, the efficiency of the current code competes favorably with the rectangular mesh counterpart with similar grid size. To get comparable results the PSI and the SUPG triangular mesh solvers are generally four to five times faster depending on the complexity of the rail shape, due to the much improved grid point position strategy and the increased accuracy of the scheme. Among the schemes, the PSI and the SUPG schemes are the most accurate and achieve grid convergence at the smallest node number.

ACKNOWLEDGEMENT

This work was supported by the Computer Mechanics Laboratory at University of California at Berkeley.

REFERENCES

1. L. Wu and D. B. Bogy, *IEEE Trans. on Magnetics,* **35,** 2421 (1999).
2. L. Wu and D. B. Bogy, submitted to *ASME J. Tribology* (1999 a).
3. L. Wu and D. B. Bogy, submitted to *ASME J. Tribology* (1999 b).
4. S. W. Sloan, *Advances in Engng Software*, **9**, 34 (1987).
5. J. Ruppert, *J. of Algorithms,* **18**, 548 (1995).
6. M. Rivara and P. Inostroza, *Int. J. Numer. Methods. Eng*., **40**, 581 (1997).
7. S. Lu, Doctoral Dissertation, Department of Mechanical Engineering, University of California, Berkeley (1997).
8. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow* (McGraw-Hill, New York, 1980).
9. P. L. Roe, *J. Comput. Phys*, **43**, 357 (1981).
10. R. Struijs, H. Deconinck and P. L. Roe, *Proc. VKI Lecture Series on Computational Fluid Dynamics, VKI LS 1991-01* (1991).
11. H. Paillere, H. Deconinck and A. Bonfiglioli, *Proc. 2nd Eur. CFD Conf*., Stuttgart (1994).
12. T. J. R. Hughes and A. N. Brooks, *Finite Element Methods for Convection Dominated Flows*, **34,** ASME, New York (1979).
13. H. Deconinck, R. Struijs, G. Bourgois and P. L. Roe, *Proc. VKI Lecture Series on Computational Fluid Dynamics, VKI LS 1993-04* (1993).
14. T. J. Barth, *AIAA paper 91-0721* (1991).
15. A. Brandt, *Math. Comp*. , **31**,333 (1977).
16. D. Mavriplis and A. Jameson, *AIAA paper 87-0353* (1987).
17. D. J. Mavriplis, *AIAA J*., **26**, 824 (1988).
18. J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Prentice-Hall, Englewood Cliffs, New Jersey, 1983).

19. A. Burgdorfer, *ASME J. of Basic Engineering*, **81**, 94 (1959).

20. Y. T. Hsia and G. A. Domoto, *ASME J. of Lubrication Technology*, **105**, 120 (1983).

21. S. Fukui and R. Kaneko, *ASME J. of Tribology*, **110**, 335 (1988).

22. F. J. Alexander, A. L. Garcia and B. J. Alder, *Phys. Fluids*, **6**, 3854 (1994).

23. W. Huang, D. B. Bogy and A. L. Garcia, *Phys. Fluids*, **9**, 1764 (1997).

24. T. J. Barth and D. C. Jespersen, *AIAA paper 89-0366* (1991).

25. T. J. Barth, *AIAA paper 93-0668* (1993).

FIG. 1. The IBM Travelstar 25GB hard disk drive.

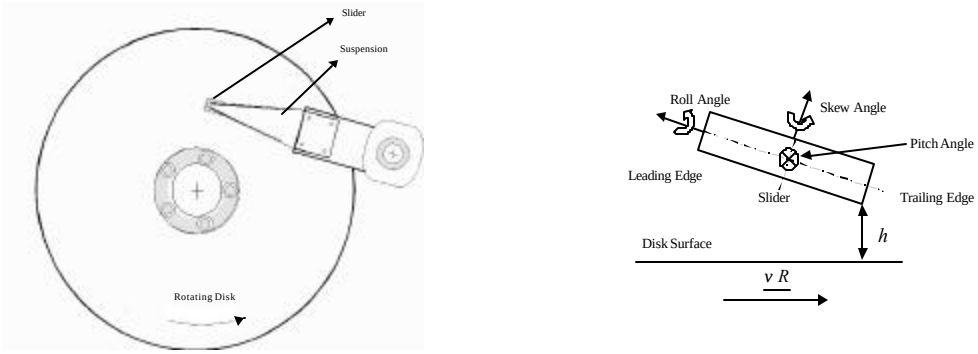

FIG. 2. A sketch of the hard disk drive assembly.

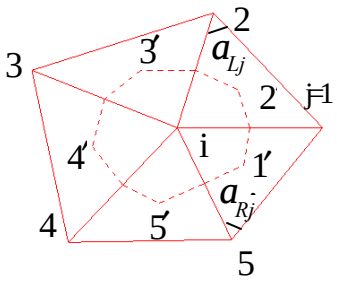

FIG. 3. The median dual control volume.

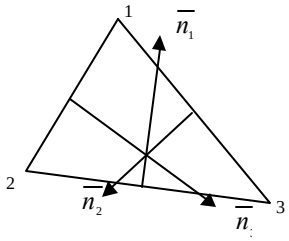

FIG. 4. The inward normals of a triangle.

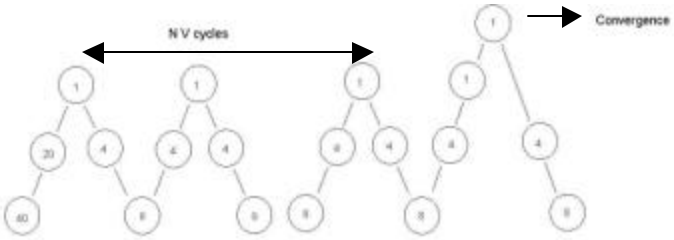

Fig. 5. The muti-grid V cycles.

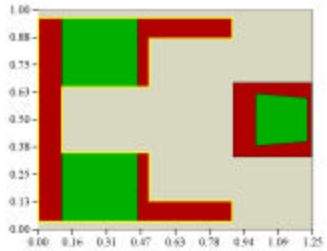

FIG. 6. A slider design close to the one used in the IBM Travelstar hard disk drive.

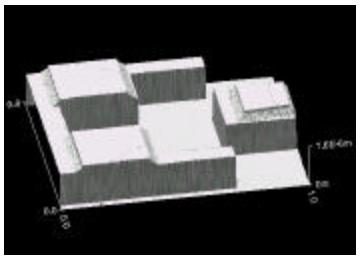

FIG. 7. The 3-D gometry of the IBM Travelstar slider.

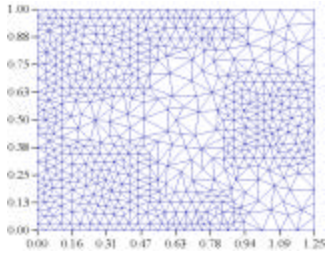

FIG. 8. The first level conforming mesh with 659 nodes

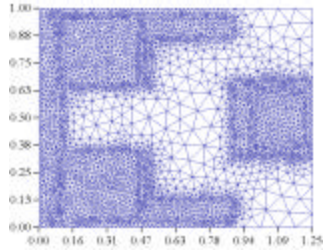

FIG. 9. The second level mesh with 4099 nodes.

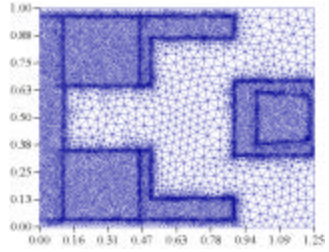

FIG. 10. The third level mesh before mesh adaptation with 12636 nodes.

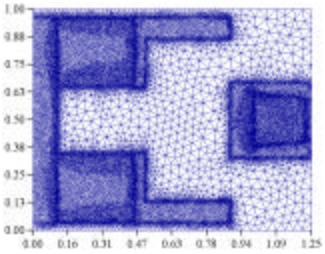

Fig. 11. The third level mesh after mesh adaptation with 18140 nodes.

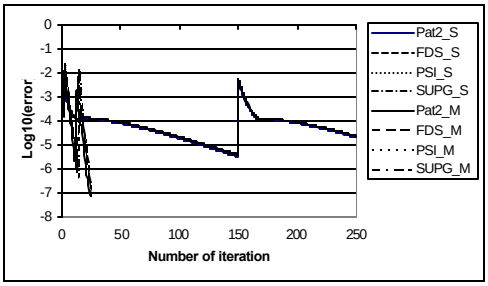

FIG. 12. The convergence history comparison between the iteration on a single set of mesh and that of the multi-grid iteration for different schemes for one fixed flying attitude .

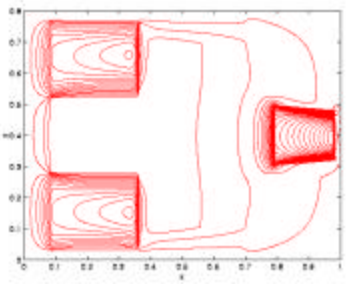

FIG. 13. The pressure contour for the steady state flying height at the 15mm radial position calculated by the rectangular mesh Patankar scheme of [7] with a 385 by 385 mesh.

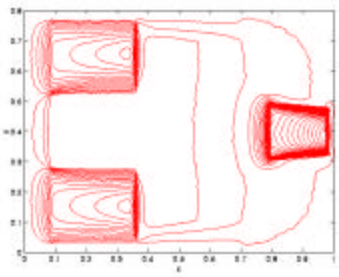

FIG. 14. The pressure contour calculated by the Patankar scheme extended to a triangular mesh [1-3] with 18140 nodes for the finest level mesh.

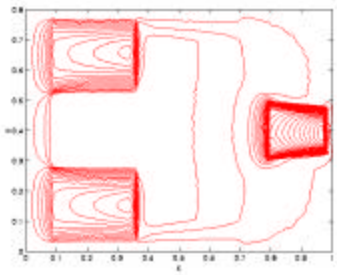

FIG. 15. The pressure contour calculated by the present FDS scheme with 18140 nodes for the finest level mesh.

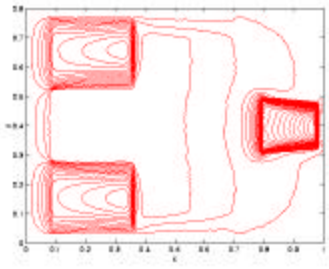

FIG. 16. The pressure contour calculated by the present PSI scheme with 18138 nodes for the finest level mesh.

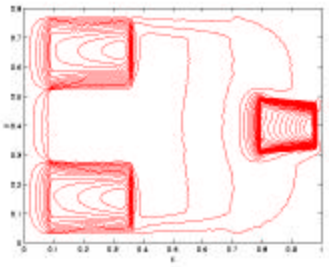

FIG. 17. The pressure contour calculated by the present SUPG scheme with 18137 nodes for the finest level mesh.

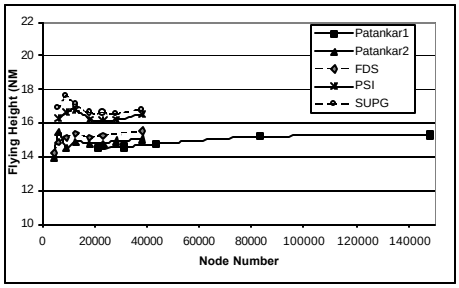

FIG. 18. The grid convergence history of nominal flying height for the different schemes.

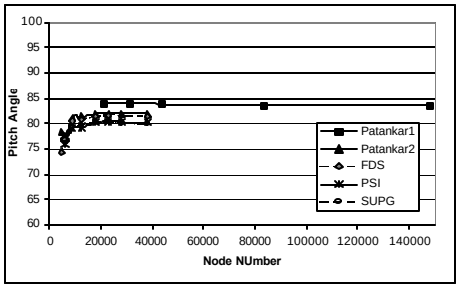

FIG.19. The grid convergence history of the pitch angle (*mRad*) for difference schemes.

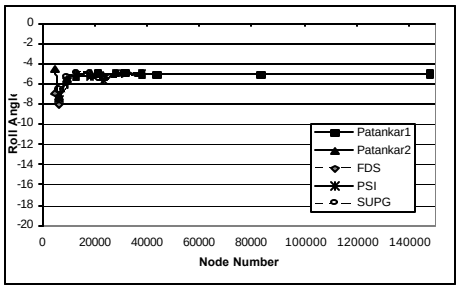

FIG.20. The grid convergence history of the roll angle (*mRad*) for difference schemes.

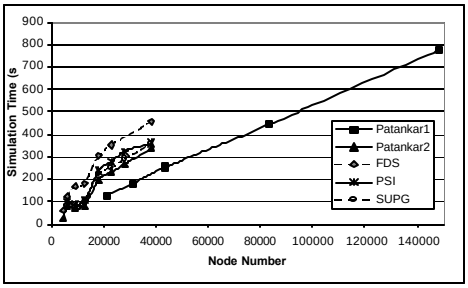

FIG. 21. The simulation time on a PII 350 PC as a function of node number.

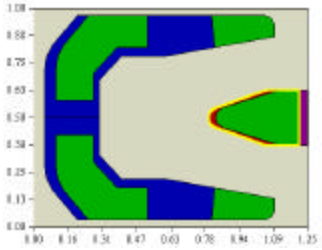

FIG. 22. The next generation NSIC slider design.

FIG. 23. The 3-D geometry of the NSIC slider.

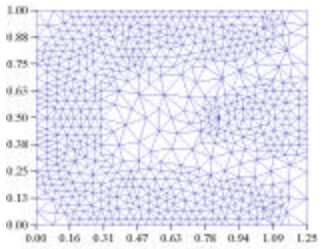

FIG. 24. The first level conforming mesh with 717 nodes

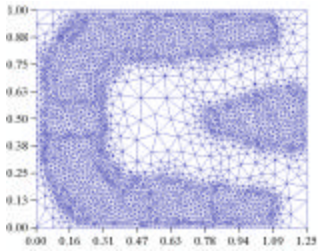

FIG. 25. The second level mesh with 4245 nodes.

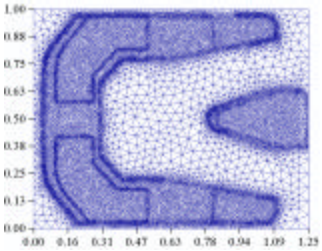

FIG. 26. The third level mesh before mesh adaptation with 12612 nodes.

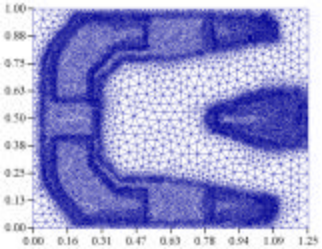

Fig. 27. The third level mesh after mesh adaptation with 18112 nodes.

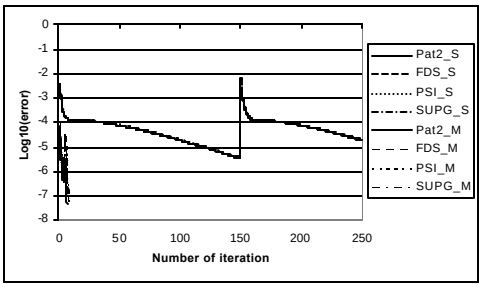

FIG. 28. The convergence history comparison between the iteration on a single set of mesh and that of the multi-grid iteration for different schemes.

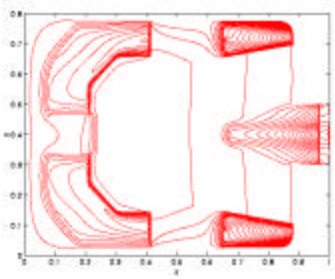

FIG. 29. The pressure contour for the steady state flying height at the 15mm radial position, calculated by the Patankar scheme with a 385 by 385 mesh.

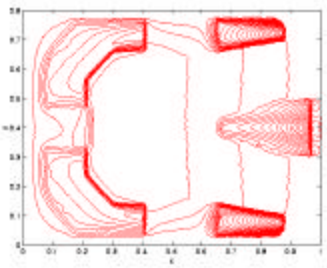

FIG. 30. The pressure contour calculated by the Patankar scheme extended to a triangular mesh with 18112 nodes for the finest level mesh.

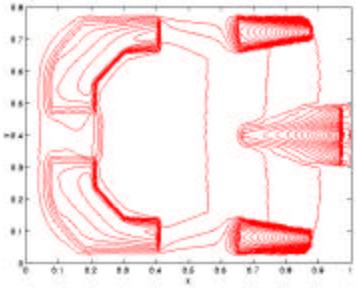

FIG. 31. The pressure contour calculated by the FDS scheme with 18111 nodes for the finest level mesh.

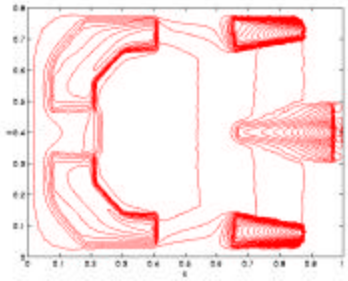

FIG. 32. The pressure contour calculated by the PSI scheme with 18110 nodes for the finest level mesh.

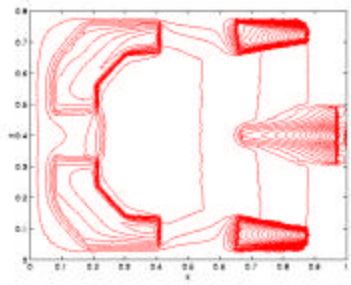

FIG. 33. The pressure contour calculated by the SUPG scheme with 18110 nodes for the finest level mesh.

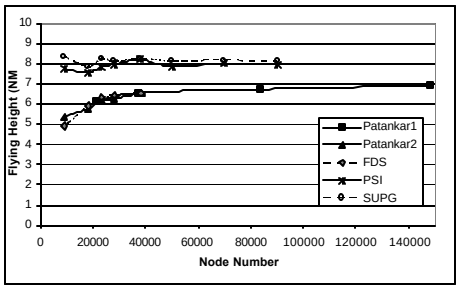

FIG. 34. The grid convergence history of nominal flying height for different schemes.

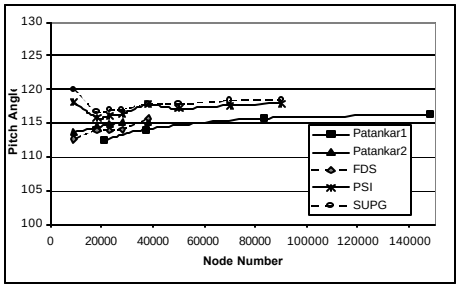

FIG. 35. The grid convergence history of the pitch angle (*mRad*) for difference schemes.

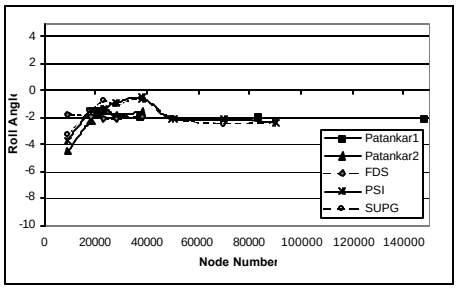

FIG. 36. The grid convergence history of the roll angle (*mRad*) for difference schemes.

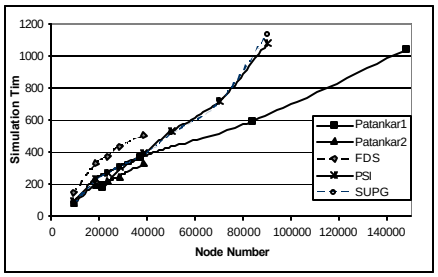

FIG. 37. The simulation time on a PII 350 PC as a function of node number.

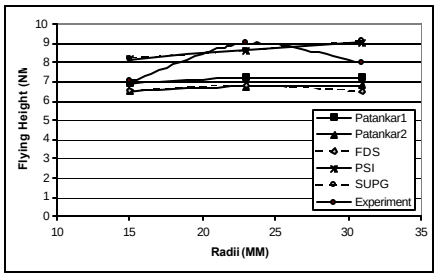

FIG. 38. The flying height at different radial position for the NSIC slider.