A local adaptive multi-grid control volume method for the air bearing problem in hard disk drives

Liping Li and David B. Bogy

Computer Mechanics Laboratory

Department of Mechanical Engineering

University of California

Berkeley, CA 94720

Abstract

A new local adaptive grid-generating algorithm is developed and integrated with the multi-grid control volume method to simulate the steady flying state of the air bearing sliders in HDDs (Hard Disk Drives) accurately and efficiently. Local finer meshes (mesh dimension decreases to half) are created on the nodes of the current finest grids that have pressure gradients or geometry gradients larger than a pre-defined tolerance after the pressure distribution has been obtained on the initial uniform mesh. In this way the pressure or geometry sensitive regions have higher resolution, leading to more accurate results without inefficiently larger meshes. Two sliders are used to demonstrate the applicability of this method. It is found that this new local adaptive grid-generating method improves the stability and efficiency of the simulation scheme.

Introduction

In modern HDDs the head disk spacing has decreased to sub-3 nm to meet the demand of higher storage density. Various technologies such as TFC (Thermal Fly-height

Control), BPM (Bit Patterned Media) and HAMR (Heat-Assisted Magnetic Recording) have been proposed to achieve this objective. Besides the requirement of the small spacing the slider is also required to have a stiff air bearing and constant flying height over the entire radius of the disk. To satisfy all these requirements the ABS (air bearing surface) designs used in current hard disk drives have complicated rail shapes, multiple etch depths and highly recessed regions between the rails. Thus the pressure distribution between the slider and disk varies greatly in the head-disk interface. Accurate and efficient simulation of the air bearing pressure is a key issue in the design of sliders. Therefore a robust numerical scheme is required to efficiently solve the generalized lubrication equation, which is used to model the air bearing between the slider and disk.

There are several numerical methods available in the literature for solving the generalized lubrication equation [1-6], such as finite difference methods, finite element methods and finite volume methods. The finite difference method has been widely used because of its efficiency if uniform meshes are suitable for the ABS design. However, for sliders with quite complicated ABS designs, generating good structured meshes becomes very difficult. So the finite element or finite volume method is preferred. The finite element method is often chosen due to its easily generated unstructured meshes and its ability to capture arbitrary rail geometries. But it is hampered by its large memory required for the unstructured meshes. The finite volume method avoids this memory problem, and it is good at maintaining mass conservation of the thin film between the slider and disk. Therefore, we use the finite volume method in the present study.

Lu [3] implemented a control volume method in the CML (Computer Mechanics Laboratory) Air Bearing Simulator to solve the lubrication equation. In this scheme a

pressure profile is first obtained on an initial uniform rectangular mesh. Then the grid lines are redistributed according to the pressure gradient or geometry gradient so that the grid is more concentrated on the areas that have high gradients of pressure or recess depth. The implementation of an adaptive grid with a multi-grid method enhances significantly the accuracy and stability of the simulation. This simulator is powerful and has been widely used in the HDD industry. However, in this scheme a local grid redistribution of a discretization cell causes the modification of all the discretization cells along those grid lines, which may lead to a dense mesh on some areas where only sparse meshes are needed, and it thereby increases the computation time without benefit. Wu et al. [6] developed an unstructured adaptive triangular mesh generation technique, which is integrated with the control volume method and multi-grid method to form an efficient air bearing simulator. The unstructured triangular mesh makes it easy to locally refine the mesh at critical regions. However, the storage of the unstructured mesh requires a large computer memory, thereby reducing its efficiency. Lu et al. [7] constructed an adaptive grid-generating algorithm and integrated it with the multi-grid method to form a numerical scheme that suits slider air bearing simulation. In this scheme rectangular finer meshes are constructed over nodes of the current finest grid where the global error exceeds a predetermined tolerance. Since it uses different criteria in the multi-grid method and the local adaptive method this scheme becomes time consuming. This scheme also uses a multi-grid method but the grid level stops increasing only when the global error is smaller than a predefined tolerance, so the final grid level can be as high as 10-30, making the final mesh very dense, which requires a longer time to finish the simulation.

In this study we implement a new local adaptive grid-generating algorithm with the CML Air Bearing Simulator to study the steady state flying conditions of some current complicated slider designs. An initial uniform rectangular mesh is generated to obtain the pressure profile. Then two criteria, based on pressure gradient and geometry gradient, are applied to refine the local mesh on some critical areas. Finally, two sliders are used to demonstrate the accuracy and efficiency of this method.

Numerical Modeling

(1) The Governing Equation and the Control Volume Method (CVM)

The governing equation for the steady state gas lubricated bearing between a slider and a disk is the Generalized Reynolds equation, which can be written as:

$$\frac{\partial}{\partial X}(QPH^3\frac{\partial P}{\partial X}-\Lambda_x PH)+\frac{\partial}{\partial Y}(QPH^3\frac{\partial P}{\partial Y}-\Lambda_y PH)=0\,, \qquad (1)$$

where $P=p/p_a$, $H=h/h_m$, $X=x/L$, $Y=y/L$ are the dimensionless pressure, bearing clearance, and coordinates in the slider's length and width directions, respectively; p_a is the ambient atmospheric pressure; h_m is the reference clearance at the trailing edge center; L is the length of the slider; $\Lambda_x=\frac{6\mu UL}{p_a h_m^2}$ and $\Lambda_y=\frac{6\mu VL}{p_a h_m^2}$ are the bearing numbers in the x and y directions; μ is the viscosity of the air; U and V are the velocity components of the rotating disk surface in the x and y directions. Q is the Poiseuille flow factor, using the F-K correction [9, 10].

The control volume method of Patankar [11] is employed to solve the generalized Reynolds equation. The integration of (1) over the control volume in Fig. 1 gives,

$$J_e-J_w+J_n-J_s=0\,, \qquad (2)$$

where J_e and J_w are $J_x \Delta Y$ evaluated at the control volume faces e and w respectively; J_n and J_s are $J_y \Delta X$ evaluated at the control volume faces n and s respectively; J_x and J_y are:

$$J_x = \Lambda_x PH - QPH^3 \frac{\partial P}{\partial X} , \tag{3}$$

$$J_y = \Lambda_y PH - QPH^3 \frac{\partial P}{\partial Y} . \tag{4}$$

So the final form of the equation at grid point P is written [3]:

$$a_P P_P = a_E P_E + a_W P_W + a_N P_N + a_S P_S + b . \tag{5}$$

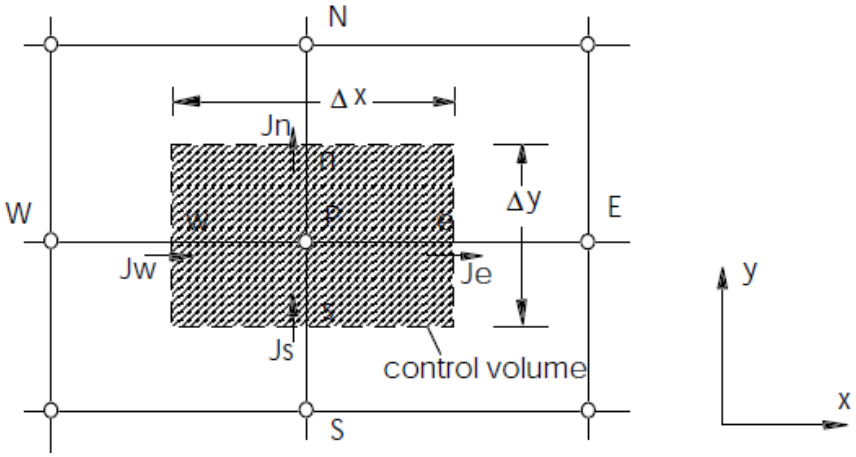

Fig. 1 Illustration of the control volume (after Patankar [11])

(2) Multi Grid Method

When solving Eq. (5), a large number of grid points is needed, in order to obtain a high accuracy the air bearing pressure, which also requires a large computation memory. So it is practical to use an iterative method. In this study we applied the FMG-FAS (full multi grid-full approximation storage) method used by Lu [3] and Shyy and Sun [8], which was designed to deal effectively with nonlinear problems.

Eq. (5) can be expressed in the matrix form:

$$[A]\langle P\rangle = \langle b\rangle . \tag{6}$$

The above equation involves two levels of iterations: the inner iteration updates $\langle P\rangle$ for fixed $[A]$ and $\langle b\rangle$, while the outer iteration updates $[A]$ and $\langle b\rangle$ using the most recent $\langle P\rangle$. The computation is carried out over a series of five grids G_k, with the corresponding solutions $\langle P_k\rangle$, where k=1, 2,3,4,5, with k=5 representing the finest mesh. The FMG-FAS method can be understood by considering two levels of grid, h (fine mesh level) and H (coarse mesh level). The solution for $\langle P_h\rangle$ on grid G_h satisfies the equation

$$[A_h]\langle P_h\rangle = \langle b_h\rangle . \tag{7}$$

At convergence $[A_h]$ and $\langle b_h\rangle$ are based on the final solutions of $\langle P_h\rangle$. During the iteration procedure they are estimated based on the most recent values of $\langle P_h\rangle$. We denote them with an over bar, and so the pressure correction is $e_h = P_h - \overline{P_h}$. Unless the approximate solution $\langle \overline{P_h}\rangle$ satisfies Eq. (7) exactly, there is a residual $\langle R_h\rangle$, which is given by:

$$\langle R_h\rangle = \langle \overline{b_h}\rangle - [\overline{A_h}]\langle \overline{P_h}\rangle . \tag{8}$$

Combining Eq. (7) and (8), we write the fine grid form as:

$$[A_h]\langle \overline{P_h} + e_h\rangle - [\overline{A_h}]\langle \overline{P_h}\rangle = \langle b_h\rangle - \langle \overline{b_h}\rangle + \langle R_h\rangle . \tag{9}$$

Then we can transfer the above equation to grid H by applying a restriction operator $[I_h^H]$ that transmits the information from a fine grid to a coarse grid. Then Eq. (9) can be written as:

$$\left[A_H\right]\left\langle I_h^H \overline{P_h} + e_H \right\rangle - \left[\overline{A_H}\right]\left\langle I_h^H \overline{P_h} \right\rangle = \left\langle b_H \right\rangle - \left\langle \overline{b_H} \right\rangle + \left\langle I_h^H R_h \right\rangle \ . \tag{10}$$

After Eq. (10) is solved, the correction e_H can be transferred to the fine grid G_h, and the fine grid pressure is updated using:

$$P_h \leftarrow \overline{P_h} + I_H^h e_H \,, \tag{11}$$

where I_H^h is the interpolation operator that transfers the correction from the coarse grid to the fine grid. This procedure can be performed recursively to form the multi-grid V-cycle.

(1) Local Adaptive Method

Different from the adaptive method in [3], in which the node number doesn't change during the mesh adaptation, this adaptive method produces new local nodes. After the pressure distribution is obtained on an initial uniform grid the pressure gradient or geometry gradient on node N is calculated on each node, and we denote it as ε(N):

$$\varepsilon(N) = \begin{cases} \max(dP_N / dx_N, dP_N / dy_N), & for \quad pressure \quad criterion \\ \max(dH_N / dx_N, dH_N / dy_N), & for \quad geometry \quad criterion \end{cases} \,, \tag{12}$$

where dP_N / dx_N is the pressure gradient in the x direction and dP_N / dy_N is the pressure gradient in the y direction, and similarly for the geometry criterion. We first set a pre-defined tolerance Tol 1, which is usually equal to a ratio number multiplied by the maximum pressure or geometry gradient:

$$Tol \quad 1 = r \bullet \max(\varepsilon(N)) \,. \tag{13}$$

If the gradient is larger than Tol 1 a finer mesh (mesh dimension decreased by half) gets created on this node. So the local adaptive criterion is:

$$\varepsilon(N) \geq Tol \quad 1 . \tag{14}$$

To demonstrated more clearly we plot a sample region in Fig.2; nodes 1, 2, 3, 4, 5 and 6 satisfy Eq. (14), so the local meshes are produced on these six nodes (see the right figure). There are three types of nodes on this refined mesh: interior nodes, interior boundary nodes and global boundary nodes. For the interior nodes the CVM form of Eq. (7) is still applicable. However, for the interior boundary nodes, Eq. (7) cannot be applied directly. Instead, an interpolation method needs to be used. The pressure on this type of node is interpolated from the pressure of those adjacent nodes.

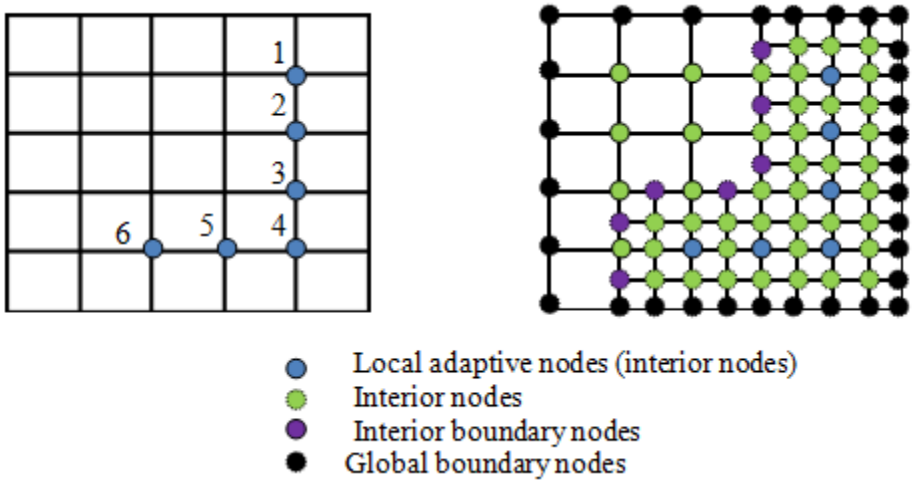

Fig. 2 A typical two consecutive mesh structure

(2) Local Adaptive Multi-grid CVM

The local adaptive method is integrated with the CVM and multi-grid method to form a complete local adaptive multi-grid control volume algorithm. The flow chart is shown in Fig. 3. After the simulation starts an initial uniform grid is generated, and the pressure is obtained after solving the Reynolds equation. The pressure gradient or geometry

gradient is calculated (depending on the criterion option) on each node. If any of the gradients is larger than a pre-defined tolerance (Tol 1), the mesh then gets locally refined.

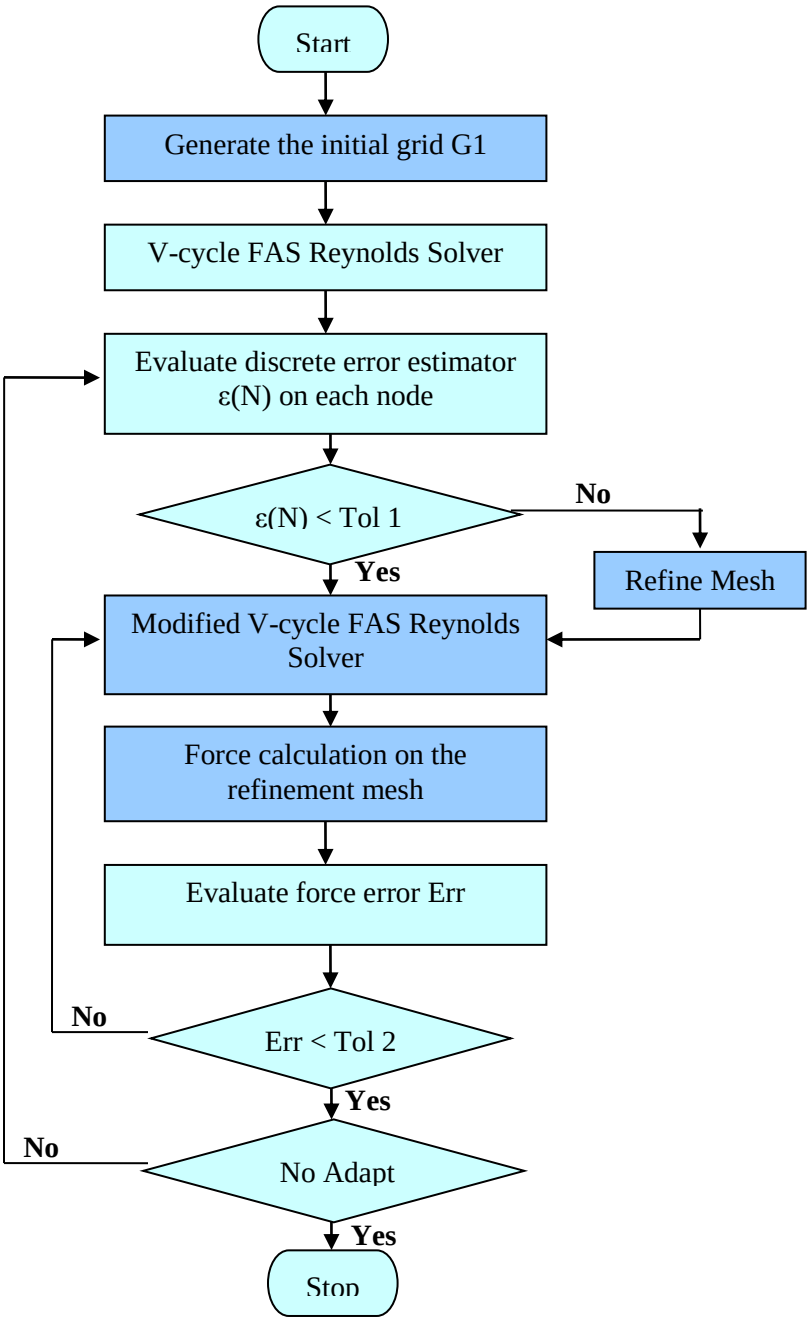

Fig. 3 A Flow chart of local adaptive multi-grid CVM

After the new mesh is created a new pressure profile is obtained by solving the Reynolds equation on this new mesh. The total force, including the air bearing force, contact force, intermolecular force, etc., is calculated to determine how well it balances the suspension force. Newton iteration is applied to reduce the difference between the simulated force and suspension force until it is smaller than a predefined tolerance (Tol 2). Then the simulation is finished after adapting the mesh only once, otherwise we again follow the same steps used in the first adaptation.

Results and Analysis

In this section we consider two different slider designs to test this local adaptive code and compare the results with Lu's method [3] to study its accuracy and efficiency.

1. First Slider

For the first slider we use three different calculations to test this Local Adaptive Multi-grid CVM.

(1.1) Case 1: uniform mesh

If we set the "r" in Eq. (13) larger than 1 in the new local adaptive mesh code, no local mesh will be generated, which means the final mesh is a uniform grid. For Lu's method we can also switch off the adaptive option so that the final mesh is a uniform grid. We compared the simulation results of Lu's method and this local adaptive method when both simulators finally employ uniform meshes. The results are shown in Table 1. From the comparison we see that the two simulation results (the slider's minimum flying height, nominal flying height, pitch and roll) have negligible differences. Thus, we conclude that this new local adaptive method code can also be used to simulate the air bearing problem in hard disk drives when there is no need for adaptive local mesh.

Table 1. Comparison with Lu's method when the final meshes are uniform grid

grid size: 145	Min FH (nm)	Nominal FH (nm)	Pitch (μrad)	Roll (μrad)
Lu's method	13.4686	10.878958	98.039114	6.5934417
Local adaptive	13.4687	10.879094	98.042139	6.5949570

(1.2) Case 2: pressure criterion

In order to test the accuracy of the local adaptive method of simulating the slider's flying attitude we first need to get a reference flying height. Fig.4 shows the minimum flying height as a function of grid size using Lu's method. It shows the minimum flying height increases as the grid size increase and finally converges to a value, which we take as the reference flying height, since the simulation result is usually more accurate when the grid size is larger. For this case, we choose the reference flying height to be 13.6 nm.

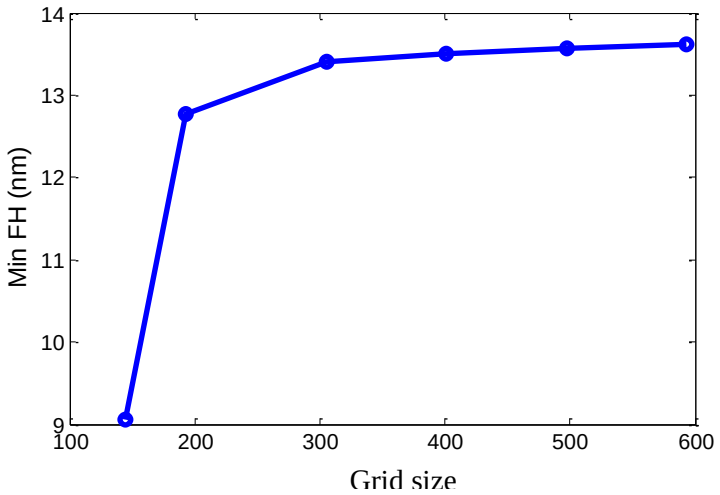

Fig. 4 Minimum flying height with different grid size using Lu's method

Next we investigate the accuracy of the local adaptive method by setting different tolerances (Eq. (13)), which leads to different local meshes. Here we use a much smaller grid size than the final mesh in Fig. 4. The result is shown in Fig. 5. The x-axis is "r" in

Eq. (13) and smaller r indicates more local meshes. The solid line shows that the minimum flying height increases as the tolerance decreases. The dashed line shows the percentage of the minimum flying height difference from the chosen reference flying height. It shows the minimum flying height difference decreases with a decrease of tolerance. And the difference is very small (less than 0.1%). These results indicate that this new local adaptive method can give very accurate results with a much smaller grid size. The corresponding final mesh for r=0.7 is plotted in Fig. 6. We see that the local meshes are generally produced close to the trailing edge center where there is high pressure and small flying height.

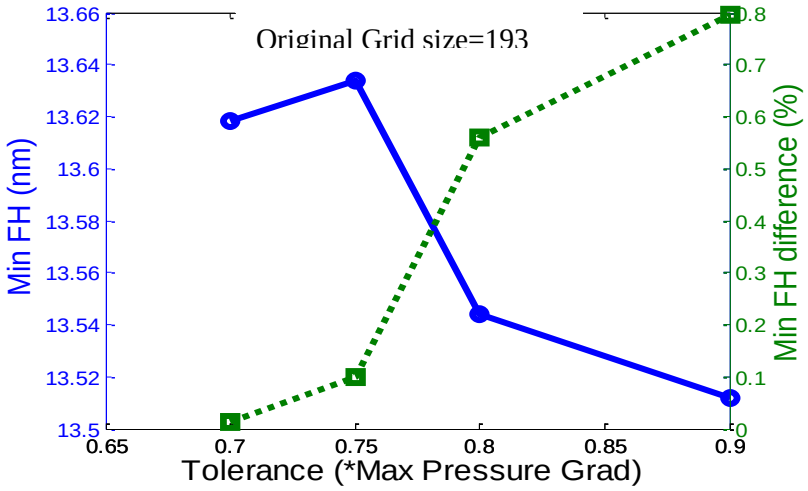

Fig. 5 Minimum flying height and its error for different tolerances (Pressure criterion)

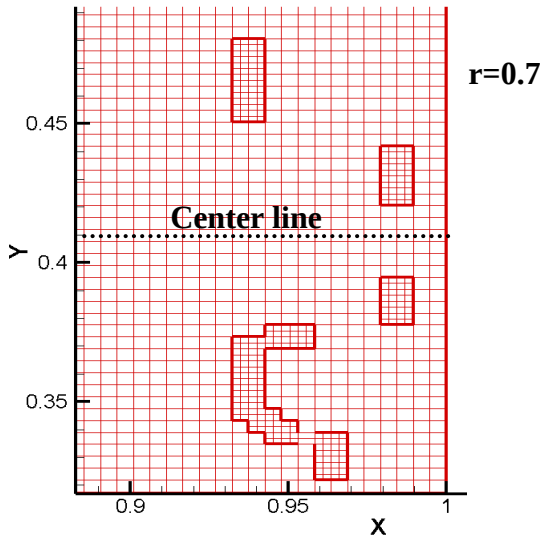

Fig. 6 Corresponding local mesh for r=0.7 in Fig. 5

(1.3) Case 3: geometry criterion

In this case we use the same reference flying height of 13.6 nm as in case 2. However, here we use the geometry criterion to produce the local meshes, which means that if the geometry gradient at a node is larger than Tol 1 then a local mesh is created on that node. The results are plotted in Fig.7. We draw the same conclusion as before, i.e., an accurate result can be obtained by using this local adaptive method with a much smaller grid size. The corresponding final mesh for r=0.7 is shown in Fig. 8. The local meshes are generally produced at the positions with a large flying height (or recess depth) change, so it captures the ABS features very well.

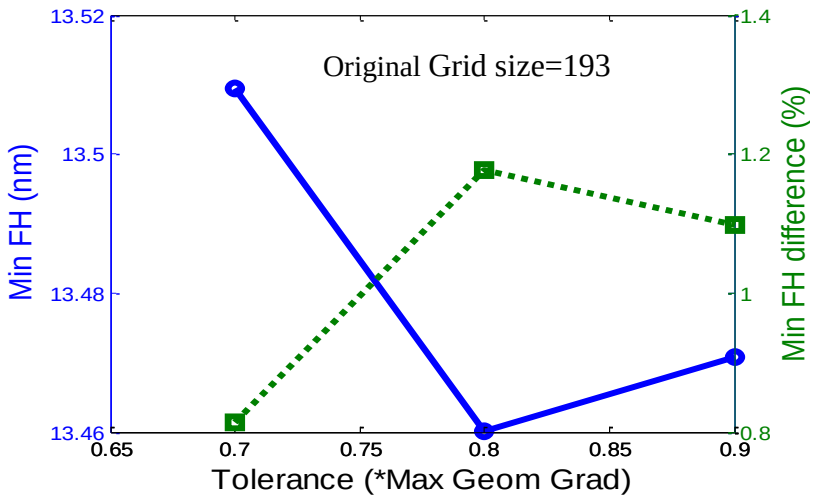

Fig. 7 Minimum flying height and its error for different tolerances (Geometry criterion)

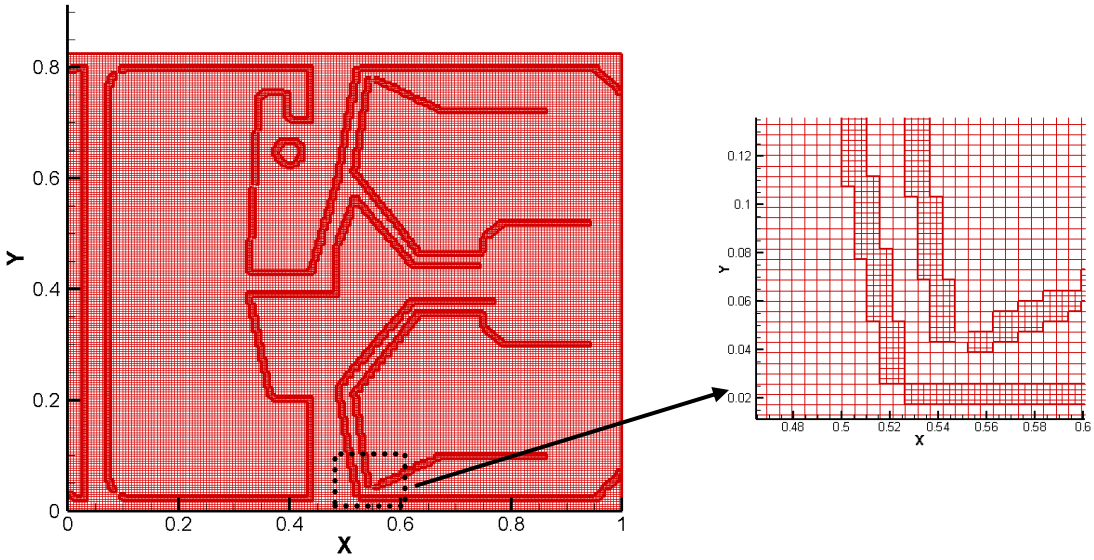

Fig. 8 Corresponding local mesh for r=0.7 in Fig. 7

(1.4) Efficiency

From the above three cases we conclude that this new local adaptive method can give very accurate results whether there are local meshes or not. So it can be used to simulate the slider air bearing problem. In this section, we examine its efficiency, and the results are shown in Table 2, which shows that the local adaptive method using the pressure criterion can save computation time compared to Lu's method, so it is more efficient for

the slider we used in this study. It also shows that when using the geometry criterion a longer time is required than when using the pressure criterion. That's mainly because the geometry criterion needs a smaller tolerance to get the same accuracy, which means more local meshes must be produced (see Fig. 6 and Fig. 8). So it is suggested to start by using the pressure criterion. However, for some very complicated ABS designs the geometry criterion may converge better than the pressure criterion. We will demonstrate this in the next section using a different slider design.

Table 2. Computation time of Lu's method and local adaptive method with the same accuracy

	Accuracy (FH difference ~1%)
Lu's method	~5 minutes
Local adaptive (pressure)	~1 minute
Local adaptive (geometry)	~6 minute

2. Second Slider

The second slider considered has a rather complicated ABS design (we call it C_slider). We first use Lu's method to simulate the slider's flying condition. The slider's minimum flying height and the corresponding computation time with grid size is shown in Fig.9. It shows that the minimum flying height doesn't converge very well as the grid size increases. Moreover, the computation time is longer than what is required for most other ABS designs. We plotted the final mesh for the case with grid size 497 in Fig. 10. It can be seen that the grid lines are concentrated near the trailing edge, slider center, outer and inner rails. For other parts there are only a few grid lines and the aspect ratio becomes very large (such as the zoom in part). This may be one of the reasons why the simulation doesn't converge very well. We also simulated the slider's flying condition using Lu's method without adaptive mesh, which means that the final mesh is uniform,

and the results are shown in Fig. 11. The minimum flying height doesn't converge very well using this method either. So Lu's method is not very suitable for this ABS design. However, from Fig. 9 and Fig. 11 it appears that this slider's final steady minimum flying height is around 13 nm because the simulation takes shorter time when the result converges to this value.

Then we used the new local adaptive method with the two criteria (pressure and geometry) to simulate the flying height of the same ABS design and analyzed its convergence.

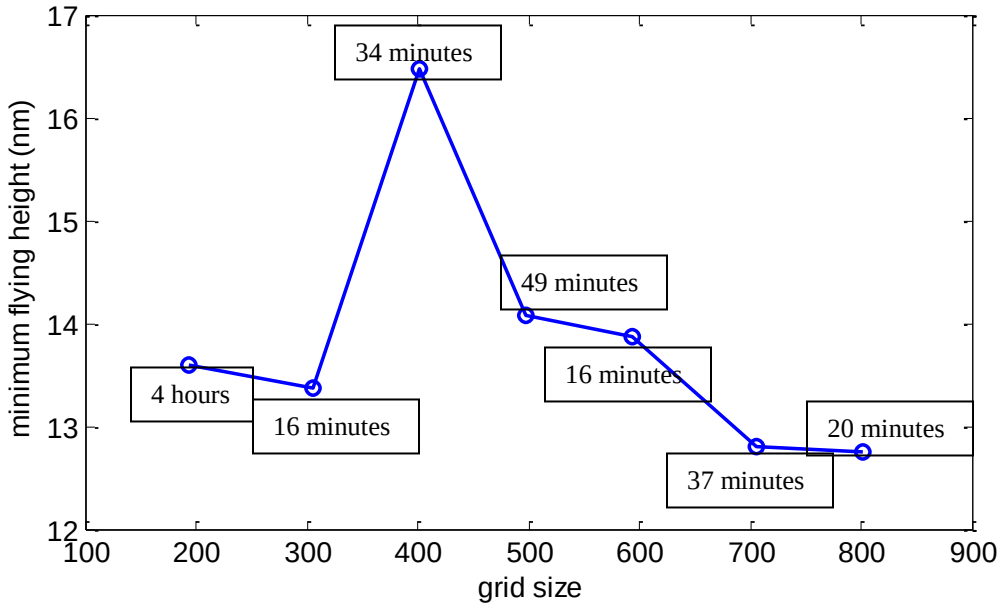

Fig. 9 Flying height and computation time with different grid size using Lu's method

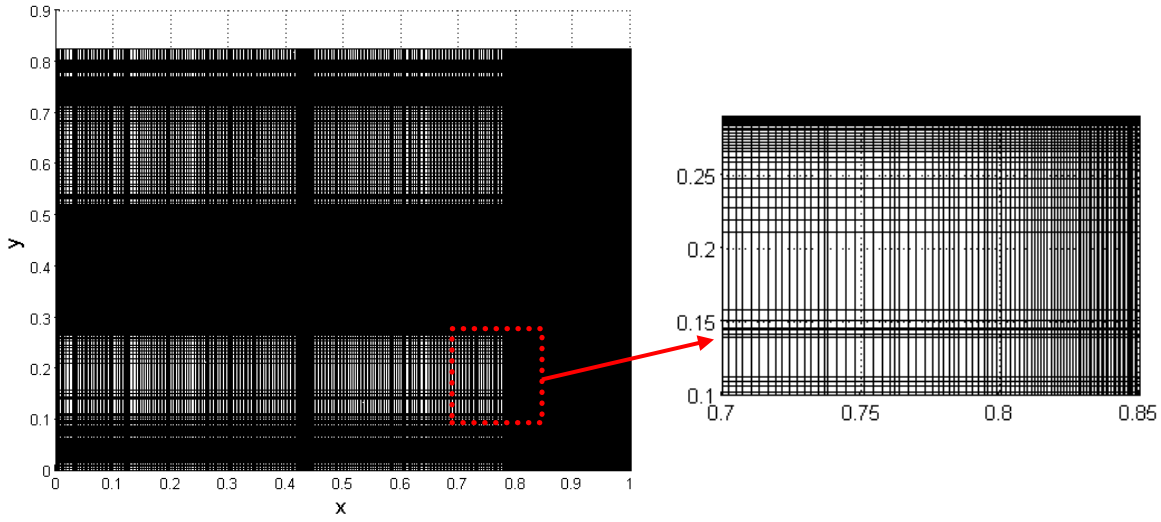

Fig. 10 Final mesh with grid size 497 for C_slider using Lu's method

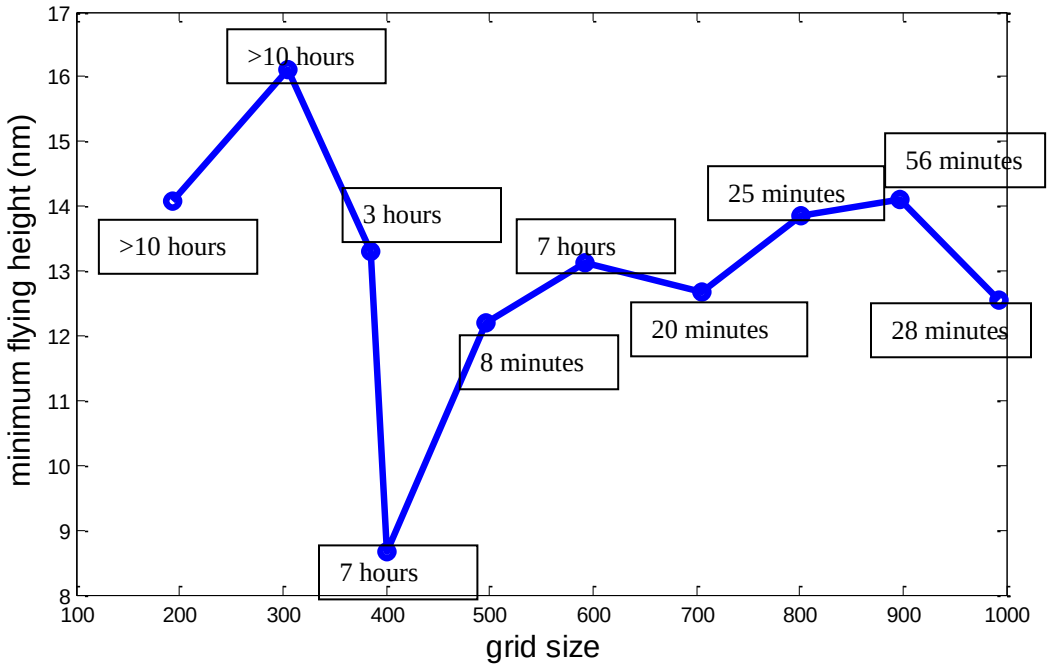

Fig. 11 Flying height and computation time with different grid size using Lu's method without adaptive mesh

(1.1) pressure criterion

The local adaptive simulation results for the minimum flying height with different tolerances using the pressure criterion for this C_slider are shown in Fig. 12. We observe that the minimum flying height still doesn't converge very well with the reduction in the

tolerance. The corresponding final mesh for r=0.2 is shown in Fig. 13. The local meshes are generally produced close to the trailing edge center where there is high pressure and small flying height. The final mesh doesn't have large aspect ratio, but it still cannot capture the high recess depth on the slider surface.

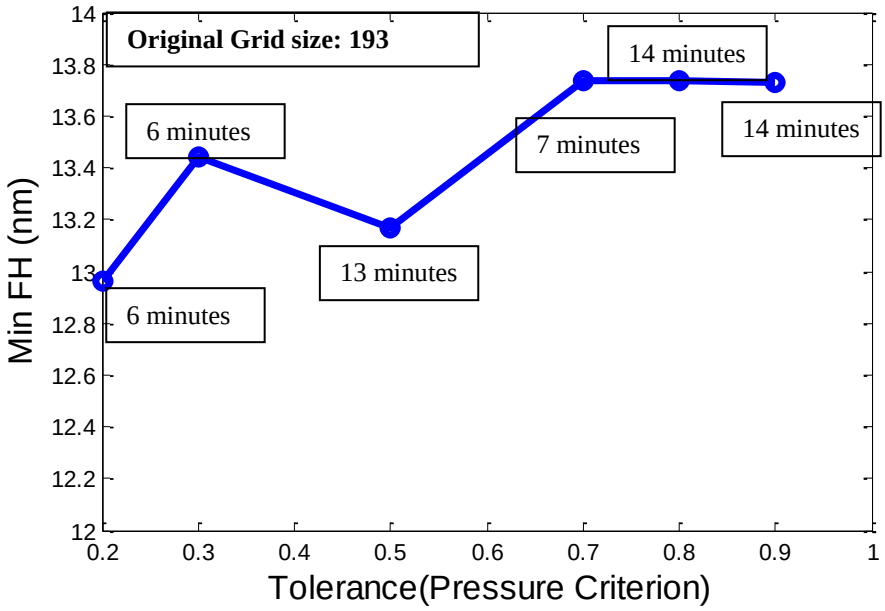

Fig. 12 Minimum flying height with different tolerances
(Pressure criterion for C_ slider)

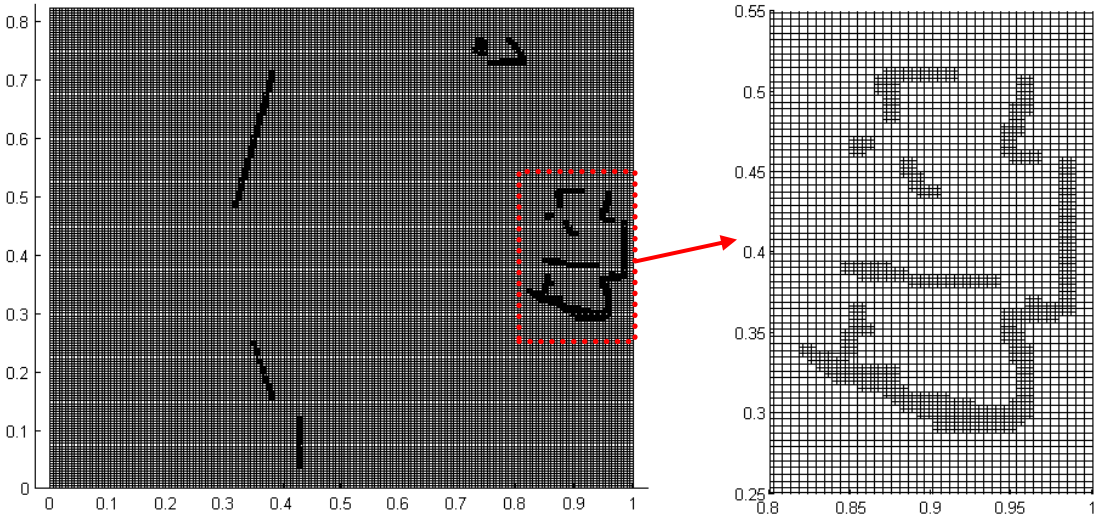

Fig. 13 Corresponding local mesh for r=0.2 in Fig. 12

(1.2) geometry criterion

The local adaptive simulated minimum flying height and the corresponding computation time with different tolerance when using the geometry criterion are shown in Fig. 14. It shows that the minimum flying height converges very well with the reduction in the tolerance. When looking at the computation times it shows that the computation efficiency improves significantly compared to the Lu's method (Fig. 9 and 11). So for this slider the local adaptive method with the geometry criterion is a better choice. The corresponding final mesh for r=0.2 is shown in Fig. 15. The local meshes are generally produced at the positions with large recess depth transitions.

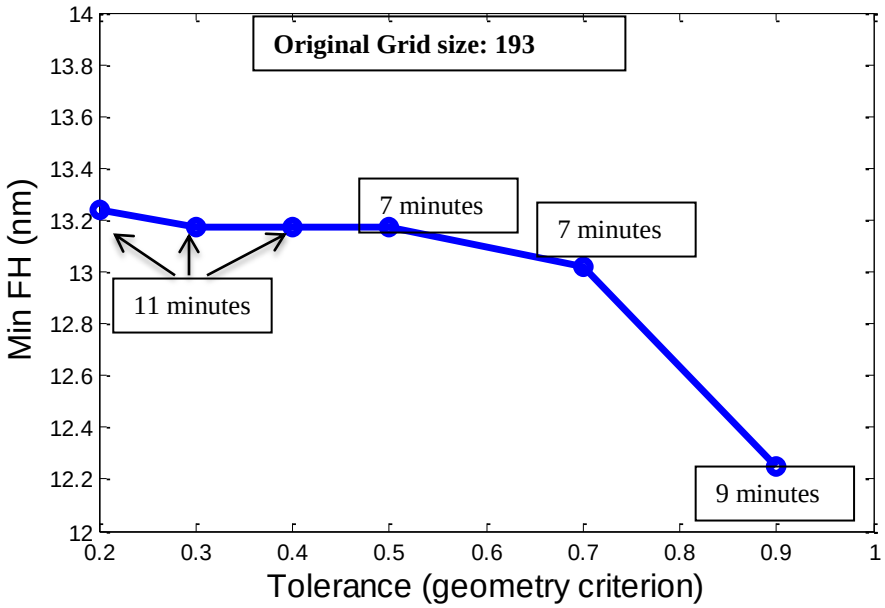

Fig. 14 Minimum flying height with different tolerances
(Geometry criterion for C_ slider)

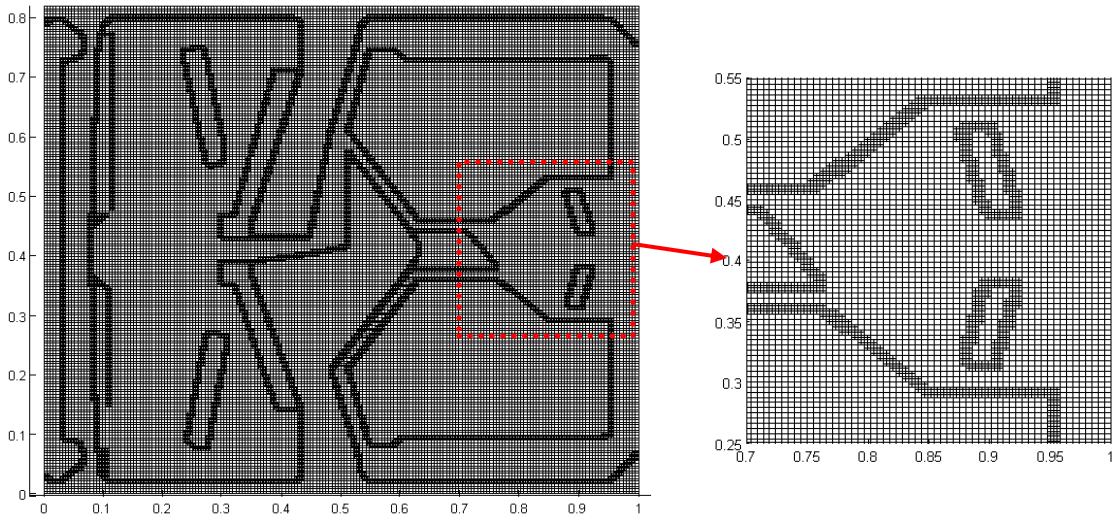

Fig. 15 Corresponding local mesh for r=0.5 in Fig. 14

Conclusion

In this study we first developed a new Local Adaptive Multi-grid Control Volume Method using multi-grids to study the air bearing problem in hard disk drives. Then two ABS designs are used to test this method. For the first ABS design three cases are considered: without a local adaptive mesh, with a local adaptive mesh using the pressure criterion and with a local adaptive mesh using the geometry criterion. These simulations indicate that this local adaptive method saves computation time as well as produces a higher accuracy. In most cases it is suggested to not use the geometry gradient, because a smaller tolerance is required in order to obtain an accurate result and thus a longer computation time is required. However, for some designs, such as the second ABS (more complicated design), it can be seen that the local adaptive method with the geometry criterion is a better choice since the produced local meshes can capture most of the geometry sensitive regions.

Acknowledgments

This research was supported by the Computer Mechanics Laboratory (CML) at University of California at Berkeley.

References

1. White J. W., and Nigam, A., 1980, ‘‘A Factored Implicit Scheme for the Numerical Solution of the Reynolds Equation at Very Low Spacing”, *ASME J. Lubr. Technol*., Vo1.102, pp. 80–85.
2. Garcia-Suarez C., Bogy D. B., and Talke F. E., 1984, ‘‘Use of an Upwind Finite Element Scheme for Air Bearing Calculations’’, *ASLE* SP-16, pp. 90–96.
3. Lu S., 1997, “Numerical Simulation of Slider Air Bearings”, *PhD thesis*, Department of Mechanical Engineering, University of California.
4. Cha E. and Bogy D. B., 1995, “A Numerical Scheme for Static and Dynamic Simulation of Subambient Pressure Shaped Rail Sliders”, *ASME Journal of Tribology*, Vo1.117:36–46.
5. Bhargava P. and Bogy D. B., 2009, “An Efficient FE Analysis for Complex Low Flying Air-Bearing Slider Designs in Hard Disk Drives—Part I: Static Solution”, *ASME Journal of Tribology*, Vo1.131: 031902.
6. Wu L, and Bogy D. B., 2000, “Unstructured adaptive triangular mesh generation techniques and finite volume schemes for the air bearing problem in hard disk drives”, *ASME Journal of Tribology*, Vo1.122:761–70.
7. Lu C.-J., Chiou S.-S. and Wang T.-K, 2004, “ Adaptive multilevel method for the air bearing problem in hard disk drives”, *Tribology International*, Vo1.37, pp. 473–480

8. Shyy W. and Sun C.S., 1993, “Development of a Pressure-correction/Staggered-grid Based Multi grid Solver for Incompressible Recirculating Flows”, *Computers and Fluids,* Vo1.22, No. 1, pp.5 l-76.
9. Fukui S. and Kaneko R., 1988, “Analysis of Ultra-Thin Gas Film Lubrication Based on Linearized Boltzmann-Equation: First Report-Derivation of a Generalized Lubrication Equation Including Thermal Creep Flow”, *ASME Journal of Tribology*, Vol. 110(2): 253-262.
10. Fukui S. and Kaneko R., 1990, “A Database for Interpolation of Poiseuille Flow-Rates for High Knudsen Number Lubrication Problems”, *ASME Journal of Tribology*, Vol. 112(1):78-83.
11. Patankar S. V., 1980, “Numerical Heat Transfer and Fluid Flow”, McGRAW-Hill, New York.
12. Brandt A., 1977, “Multi-level adaptive solutions to boundary-value problems”, *Mathematics of Computation,* Vol. 31: 333-390.