AIAA 2000-0272

Unsteady Separated Flow Simulations using a Cluster of Workstations

Anirudh Modi and Lyle N. Long
The Pennsylvania State University
University Park, PA

**38th Aerospace Sciences
Meeting & Exhibit**

10 – 13 January 2000 / Reno, NV

UNSTEADY SEPARATED FLOW SIMULATIONS USING A CLUSTER OF WORKSTATIONS

Anirudh Modi* and Lyle N. Long†
Department of Aerospace Engineering
The Pennsylvania State University
University Park, PA-16802

Abstract

The possibility of predicting the full three-dimensional, unsteady, separated flow around complex ship and helicopter geometries is explored using unstructured grids in a parallel flow solver. The flow solver used is a modified version of the Parallel Unstructured Maritime Aerodynamics (PUMA) software. The efficiency and accuracy of PUMA at resolving several steady state solutions and a fully three-dimensional unsteady separated flow around a sphere were studied. Since this requires immense computational resources, one has to often depend on expensive supercomputers to do the job. The *COst effective COmputing Array* (COCOA) is an inexpensive Beowulf-class supercomputer built and tested by the Aerospace Engineering Department at the Pennsylvania State University, as part of an effort to make this possible at a very economical cost. Various benchmarks were conducted on COCOA to study its performance at solving such problems. Unstructured grids were utilized in order to maximize the number of cells in the area of interest, while minimizing cells in the far field. A high level of clustering is required to solve viscous unsteady problems, and unstructured grids offer the least expensive method to ensure this. NASA's VGRID package was used to generate the unstructured grids.

Introduction

The prediction of unsteady separated, low Mach number flows over complex configurations (like

ships and helicopter fuselages) is known to be a very difficult problem. The possibility of predicting these types of flows with the aid of inexpensive parallel computers is explored in this work. A parallel, finite volume flow solver was used and efforts were made to expedite the entire solution process.

An example of a separated flow around a body with sharp edges is the air flow around ships. The increasing use of helicopter in conjunction with ships poses major problems. In the presence of high winds and rough seas, excessive ship motions and turbulent separated flow from sharp-edged, box-like ship super-structures make landing a helicopter on ships a very hazardous operation. The strong unsteady flows can cause severe rotor blade deformations as well. Recent research on ship airwakes has been conducted from several different approaches. A simple model of a ship, but rather crude, is a sharp edged blunt body called the General Ship Shape (GSS).^{1,2} More geometrically precise studies have been carried out in wind tunnels^{3,4} and full scale tests have been conducted by the US Navy.⁵ There have been several other attempts at numerically simulating ship airwakes, but no method to-date has been entirely satisfactory for predicting these flow fields.

Another class of problems that are somewhat similar to those above are the flow over bluff bodies such as helicopter fuselage. Spheres and cylinders are considered as prototype examples from the class of flows past bluff bodies. Over the decades, a lot of work has gone into the study of unsteady separated flow over spheres and cylinders at various Reynolds numbers. Since

* Graduate Research Assistant, Student Member

† Professor, Associate Fellow

these are simple geometric shapes and are easily reproducible and thus tested, they have enjoyed a lot of importance in the study and validation of numerical flow solvers designed to deal with such complex flows. Extensive experimental data is readily available for several different flow conditions for both the flow over a cylinder and over a sphere. Recently, Tomboulides⁶ has carried on a complete *Direct Numerical Simulation* (DNS) and *Large Eddy Simulation* (LES) of flow over the sphere at various Reynolds numbers ranging from 50 to 20,000.

Parallel Flow Solver

PUMA stands for **Parallel Unstructured Maritime Aerodynamics**. It is a computer program for the analysis of internal and external, non-reacting compressible flows over arbitrarily complex 3D geometries. It is written entirely in ANSI C using MPI (Message Passing Interface) libraries for message passing and hence is highly portable giving good performance.⁷ It is based on the Finite Volume Method (FVM) that solves the full three-dimensional Navier-Stokes equations, and supports mixed topology unstructured grids composed of tetrahedra, wedges, pyramids and hexahedra (bricks). PUMA may be run so as to preserve time accuracy or as a pseudo-unsteady formulation to enhance convergence to steady-state. It uses dynamic memory allocation, thus problem size is limited only by the amount of memory available on the machine. It needs 582 bytes/cell and 634 bytes/face using double precision variables (not including message passing overhead). PUMA implements a range of time-integration schemes like *Runge-Kutta*, *Jacobi* and various *Successive Over-relaxation Schemes* (SOR), as well as both *Roe* and *Van Leer* numerical flux schemes. It also implements various monotone limiters used in second-order computations.

VGRID

Before any numerical solution can be computed, the physical domain must be filled with a

computational grid. The grid must be constructed in a way to accurately preserve the geometry of interests while providing the proper resolution for the algorithm to be applied. The two major categories of grid construction are structured grids and unstructured grids. Each type of grid has its own particular advantages and disadvantages. Structured grids are easier to handle computationally because their connectivity information is stored block to block. Structured grids are however more difficult to construct and tend to waste memory with unnecessary cells in the far field. Unstructured grids are more difficult to handle computationally because their connectivity is stored for each node. Unstructured grids, however, tend to be easier to construct and do not waste memory in the far field.

The unstructured grids created around the geometries studied in this research, were generated using GridTool and VGRID, a grid generator based on the advancing front method (AFM). VGRID⁸ was developed by ViGYAN, Inc. in association with the NASA Langley Research Center, as a method of quickly and easily generating grids around complex objects. VGRID is a fully functional, user-oriented unstructured grid generator.⁹ GridTool is the program that acts as a bridge between Computer Aided Design (CAD) packages and grid generation in VGRID. The typical process starts with generating a drawing for the geometry of interest in a CAD package (e.g. ProEngineer, AutoCAD). This geometry is then exported from the CAD package to an IGES format. GridTool can then prepare the geometry for grid generation by providing VGRID with a complete and accurate definition of the geometry. This is accomplished by specifying curves along the geometry and then turning these curves into unique surface patches that define the geometry. Source terms are then added to the computational domain, which provide VGRID with the starting information for the grids in the AFM. The source terms can be freely placed anywhere in the domain and act as a control mechanism for clustering. An example of a grid generated using VGRID can be seen in Figure 1. This grid represents a part of the Apache helicopter geometry and was obtained from H.E. Jones of the U.S. Army.

Figure 1: Volume grid generation over Apache helicopter geometry

COCOA

The COst effective Computing Array (COCOA) is the Pennsylvania State University Department of Aerospace Engineering initiative to bring low cost parallel computing to the departmental level.¹⁰ COCOA is a 50 processor cluster of off-the-shelf PCs connected via fast-ethernet (100 Mbit/sec). The PCs are running RedHat Linux with MPI for parallel programming and DQS for queueing the jobs. Each node of COCOA consists of Dual 400 MHz Intel Pentium II Processors in SMP configuration and 512 MB of memory. A single Baynetworks 27-port fast-ethernet switch with a backplane bandwidth of 2.5 Gbps was used for the networking. The whole system cost approximately \$100,000 (1998 US dollars). Detailed information on how COCOA was built can be obtained from its web-site.¹¹ COCOA was built to enable the study of complex fluid dynamics problems using CFD (Computational Fluid Dynamics) techniques without depending on expensive external supercomputing resources. Some of the other projects that are using COCOA are discussed in Refs.¹²⁻¹⁵

Benchmarks

Since the cluster was primarily intended for fluid dynamics related applications, the flow solver

PUMA was chosen as one of the first benchmarks. Figure 2 shows the Mflops obtained from an inviscid run on a general ship shape (GSS) geometry on different number of processors using PUMA. For this case, an unstructured tetrahedral grid with 483,565 cells and 984,024 faces was used and the run consumed 1.2 GB of RAM. The benchmark showed that COCOA was almost twice as fast as the Penn State IBM SP (older 66 MHz RS/6000-370 nodes) for our applications.

Figure 5 shows the results obtained from the netperf test (done using: `netperf -t UDP_STREAM -l 60 -H <target-machine> -- -s 65535 -m <packet-size>`). This is indicative of the communication speed between any two nodes. It is seen that almost 96% of the peak communication speed of 100 Mbit/sec is achieved between any two nodes for packet sizes above 1,000 bytes.

A set of well-known parallel benchmarks related to CFD were also conducted on COCOA using the publicly available *NAS¹ Parallel Benchmarks (NPB) suite v2.3¹⁶* written in Fortran 77. Of the eight tests contained in the suite, five were *kernel* benchmarks and the other three were *simulated CFD application* benchmarks. There were four different problem sizes for each test: Class “W”, “A”, “B” and “C”. While class “W” was the workstation-size test (smallest), size “C” was supercomputer-size test (largest). Figures 3 and 4 show the performance of COCOA for each of the problem sizes for the LU solver and Multigrid solver tests, respectively.

Results

General Ship Shape (GSS)

The increasing use of helicopter in conjunction with ships poses many major problems. In the presence of high winds and rough sea, excessive ship motions and turbulent, separated flow from sharp-edged box-like ship super-structures make landing a helicopter on ships a very hazardous operation. The strong unsteady flows can cause

¹NAS stands for *Numerical Aerospace Simulation* facility, and is a part of NASA Ames Research Center

Figure 2: Total Mflops vs Number of Processors on COCOA for PUMA test case

Figure 4: NAS Parallel Benchmark on COCOA: *LU* solver (LU) test

Figure 3: NAS Parallel Benchmark on COCOA: *Multigrid* (MG) test

Figure 5: Mbits/sec vs Packet size on COCOA for netperf test

severe rotor blade deformations.¹⁷ There have been numerous incidences where the helicopter blades have actually impacted the helicopter fuselage, which is called a ‘tunnel strike’. Statistically, a helicopter can safely land on a frigate in the North Sea only 10 percent of the time in winter. In addition, flight simulators also have no adequate models for airwake. In order to avoid this and other engage/disengage problems, determining safe operating envelopes is very costly and time consuming. On the other hand, many numerical simulation attempts of this flow field

for minimizing cost have not been successful due to the inherently unsteady nature of flow and the low-speed character of the flow (which may cause numerical stiffness).

Recent research on ship airwakes has been conducted from several different approaches.¹⁸ One of the sources of relevant research is building aerodynamics which shows the general features of flow about blunt bodies of different aspect ratios, and about clusters of buildings. A simple model of a ship, but rather crude, is a sharp edged blunt body called the General Ship Shape (GSS). The GSS is a

standard shape set forth by the United States Navy as a geometry to be used to compare flow solvers. The GSS represents a typical United States Navy frigate with the area of interest being the flow field on the aft deck. The aft deck primarily acts as a landing area for helicopter. The turbulent, separated flow caused by the ship structure can make landing a helicopter extremely difficult. The strong unsteady flows can also cause severe blade deformation during start up and shut down of a helicopter. This GSS geometry unstructured grid and solution was completed in conjunction with the Pennsylvania State University ship airwake studies.² More geometrically precise studies have been carried out in wind tunnels^{3,4,19} and full scale tests have been conducted by the US Navy,⁵ which provide some important information on real ship airwakes. All these experimental tests are crucial for validating numerical models. Wind tunnel tests can be quite costly, and the flow measurements on real Naval ships is very difficult and costly to obtain.

It would be very useful to have numerical methods that could simulate ship airwakes. There have been other attempts at numerically simulating ship airwakes, e.g. a steady-state flow solver based on the 3D multi-zone, thin-layer Navier-Stokes (TLNS) method²⁰ and an unsteady, inviscid low-order method solver.²¹ No method to-date has been entirely satisfactory for predicting these flow fields.

This was the first complex case which was used to validate PUMA. The flow conditions were taken to be $M_\infty = 0.065$ and the yaw angle $\beta = 30^\circ$. Both inviscid and viscous ($Re = 1.15 \times 10^8$) cases were run.¹ For the inviscid run, a structured grid with 227,120 cells and 694,556 faces was used and the run consumed 1.2 GB of memory. For the viscous run, an unstructured tetrahedral grid with 483,565 cells and 984,024 faces was used and the run consumed 1.1 GB of memory. The convergence history for the viscous run is shown in figure 6. It shows a plot of the relative residual² against the number of iterations/timesteps. The surface velocity on the GSS geometry as computed by PUMA is shown in figure 30. Qualitative

Figure 6: Convergence history for GSS viscous run for 30° yaw case

comparisons were then made with the available experimental data.²² As seen in figures 8, 11 and 14, which compare both the inviscid and viscous solutions with the experiments, it is very clear that PUMA gives very good solutions for this case. The viscous solution is seen to be clearly better in the region of flow separation.

The results, although extremely good, can be improved by implementing the atmospheric boundary layer for the water surface, which has been currently taken as a friction-less surface.

Apart from the above runs, 14 other inviscid cases were run using the unstructured grid. These cases were for yaw angles of 0° , 30° , 60° , 90° , 120° , 150° and 180° , with two different flow speeds of 40 and 50 knots, respectively. The surface U/U_∞ contours for some of these cases can be seen in figures 31, 32 and 33. These cases were run to study the effect of various flow conditions on the engage/disengage operations of the helicopter rotor. The region of interest for this study was the rear deck of the ship (figure 34). More details on this can be found in work by Keller and Smith.¹⁷

Viscous Sphere ($Re = 1000$)

A complete unsteady separated flow solution over a sphere in uniform flow has also been simulated using PUMA. Since flow over a sphere is a prototype example from the class of flows past

²Ratio of the current residual to the initial residual

Figure 7: Oil flow pattern on front part of bridge deck of GSS for 30° yaw case

Figure 10: Oil flow pattern on middle part of bridge deck of GSS for 30° yaw case

Figure 8: Vector plot on front part of bridge deck of GSS for 30° yaw case (inviscid case)

Figure 11: Vector plot on middle part of bridge deck of GSS for 30° yaw case (inviscid case)

Figure 9: Vector plot on front part of bridge deck of GSS for 30° yaw case (viscous case)

Figure 12: Vector plot on middle part of bridge deck of GSS for 30° yaw case (viscous case)

bluff bodies, and because extensive experimental data is available for it, this was considered an ideal case to use for the validation of PUMA.¹⁰

The flow domain for this case was box shaped and is shown in figure 16. The grid extended for 5 diameters in the front, the top and the bottom

Figure 13: Oil flow pattern on front deck of GSS for 30° yaw case

Figure 14: Vector plot on front deck of GSS for 30° yaw case (inviscid case)

Figure 15: Vector plot on front deck of GSS for 30° yaw case (viscous case)

of the sphere, and 20 diameters behind. The blockage ratio corresponding to these dimensions is 0.65%. Since the boundary layer on the

sphere does not become turbulent until a Reynolds number of the order of 300,000 is reached, estimates of the boundary layer thickness were derived from the laminar boundary layer solution for axisymmetric bodies.²³ For this grid, 10 layers of prismatic cells were placed within the boundary layer. The final viscous grid obtained consisted of 306,596 cells and 617,665 faces, and the run consumed 600 MB of memory on 32 nodes of COCOA (running 2-stage Runge-Kutta). The flow conditions were $M_\infty = 0.2$ ($U_\infty = 68$ m/s) and $Re = 1000$ based on the diameter. This case was initially run using SSOR to get a somewhat steady state solution (with residual reduction of about 2 orders in magnitude), and then switched over to a 2-stage time-accurate Runge-Kutta scheme. The time-accurate run commenced after 14,000 iterations, but all data before iteration 50,000 was discarded to avoid the transients. Therefore, $t = 0$ corresponds to iteration 50,000. The convergence history for the run is shown in figure 17, and the C_p and Mach contours along the planes $X = 0$ and $Z = 0$ are shown in figures 26 and 28. For the time-accurate run, each timestep corresponds to $\Delta t = 6.45911 \times 10^{-7}$ sec = 2.197×10^{-5} diameter units³. From the available experimental data, for $Re = 1000$, the Strouhal number is expected to be very close to 0.2, thus giving a vortex shedding period of approximately $T = 0.15$ sec = 5 units, for this simulation.

In spite of our running for just two cycles of the vortex shedding period ($t = 10 = 2T$), the time-averaged C_p results (figure 18) compared quite well with the experimental data from Modi and Akutsu.²⁴ The correlation between the surface C_p values was almost perfect in the range $0^\circ \leq \theta < 110^\circ$, but seemed to vary by approximately a constant amount for the wake region ($110^\circ \leq \theta \leq 180^\circ$). Axial velocity data along the center line trailing the center of the sphere are compared to both the DNS simulation by Tomboulides,⁶ and the experimental data from Wu and Faeth²⁵ in figures 22 and 23. The comparison is not perfect, but the results are encouraging. The difference can be attributed to the fact that the

³time required by the flow to convect one diameter in the flowfield

simulation described here ran for just two cycles of vortex shedding (and 10 time units), unlike the DNS data from Tomboulides⁶ which was averaged over more than 20 shedding cycles. Due to the extremely small number of samples, our time-averaged results were very sensitive to the starting and ending points of the sampled data. Figure 19 shows the time history plot of the non-dimensional axial velocity (v/U_∞) of several points in the wake of the sphere. Axial velocity profiles for different planes in the wake of the sphere are also shown in figure 20. The nature of the flow again matches with the experiments here, as reverse flow is noticed only until about 1.8 diameters downstream of the sphere (measuring from the center of the sphere). Figure 21 shows the variation of the lift (C_L), drag (C_D) and side force (C_M) coefficients over time of the simulation.

Figure 24 shows the iso-surface at which axial velocity is 90% of U_∞ for the instant $t = 9.34$. This gives the approximate shape of the wake. Figure 25 attempts to show the three-dimensional nature of the wake region for the same instant. Figure 27 depicts the flow along the slice $X = 0$ at different instants of time. Although the streamlines shown do not signify the actual unsteady streamlines, they do give the general idea about the tendency of the flow. Streaklines are what are really needed to be shown, but they are very tedious and difficult to obtain. The figure clearly shows the alternating nature of the vortex shedding for $t = 0$ and $t = 2.75$. However, the same observation cannot be made for the latter two figures, maybe because the vortex shedding has not yet stabilized and thus cannot be considered periodic.

Other cases

Figure 35 shows the surface total velocity contours for a Landing Helicopter Aide (LHA) geometry. This case was run to study a specific spot on the LHA where a lot of problems have been seen in the event of a helicopter landing. The flow conditions were 25 knots (12.7 m/s) at 5° yaw. The inviscid grid consisted of 1,216,709 cells and 2,460,303 faces and the run consumed 2.3 GB of RAM. The initial 2,000 timesteps were traversed using SSOR and the remaining timesteps using

Figure 16: Domain for the viscous sphere grid

Figure 17: Convergence history for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

the less expensive 2-stage Jameson-style Runge-Kutta scheme. The entire run took 39 hours on 32 processors of COCOA.

Figure 29 shows surface Mach contours for an Apache helicopter geometry. The flow conditions were $U_\infty = 114$ knots. The inviscid grid consisted of 555,772 cells and 1,125,596 faces and the run consumed 1.9 GB of RAM.

Conclusions

A complete, fast and efficient unstructured grid based flow solution around several complex

Figure 18: Time averaged plot for C_p vs θ for upper surface compared with experiments,²⁴ for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$) [Note: Here, C_p is defined as $(P_\theta - P_{60^\circ}) / (P_0 - P_{60^\circ})$]

Figure 20: Instantaneous velocity profile ($t = 8.79$) showing wake region for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

Figure 19: Time history depicting axial velocity for points at fixed distances behind the center of the sphere along the y-axis

Figure 21: Variation of total C_l , C_d and C_m coefficients with time for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

geometries has been demonstrated. The objective to achieve this at a very affordable cost using inexpensive departmental level supercomputing resources like COCOA, has been fulfilled.

The flow solver PUMA has been verified with experimental data from inviscid and viscous solutions around the GSS geometry. Although the experimental data was qualitative in nature, the excellent correlation between the two for

such a complex flow was extremely encouraging. Both the inviscid and viscous solutions compared very well, with the viscous solution being better in the region of flow separation. Recent work by Schweitzer⁹ has also compared inviscid solutions from PUMA around the ROBIN body with experiments, and the outcome was very favorable. Several other large problems for flow around different ship configurations (LHA, aircraft carrier CVN-75) and helicopter fuselages

Figure 22: Time averaged axial velocity plot (v/U_∞) for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

Figure 23: Time averaged RMS velocity plot (v'/U_∞) for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

(AH-64 Apache, Boeing generic fuselage) have been run, but lack of any experimental data has prevented the measure of PUMA's success for these cases. Viscous, unsteady runs for $Re = 1000$ around a finite cylinder and sphere have also been conducted. For the sphere case, in spite of our running for just two cycles of the vortex shedding period, the time-averaged C_p results compared quite well with the experimental data. The correlation between the surface C_p values

Figure 24: Iso-surface at which axial velocity is 90% of U_∞ ($t = 9.34$)

was almost perfect in the range $0^\circ \leq \theta < 110^\circ$, but seemed to vary by approximately a constant amount for the wake region ($110^\circ \leq \theta \leq 180^\circ$). The code also was able to predict the trends in the wake axial velocity, although there were some discrepancies. This can be attributed to the fact that the simulation described here ran for just two cycles of vortex shedding (10 time units), which made the data statistically stiff, as compared to the DNS data which was averaged over more than 20 cycles. Based on the validation and the experience from the above runs, PUMA is seen to be a very good starting point for modeling such complex unsteady separated flows. We plan to run LES simulations in the near future.

COCOA was found to be extremely suitable for our numerical simulations. One of the real benefits of inexpensive machines is that they do not have to be shared with hundreds of other users, and we do not have to wait days in a queueing system. We quite often have to wait several days at a supercomputer center just to use 16 processors. Also, while it is quite difficult to get 50,000 CPU hours at a supercomputer center, our Beowulf cluster gives us more than 400,000 CPU hours per year. COCOA was also found to have good scalability with most of the MPI applications used. Although Beowulf clusters have very high latency as compared to conventional supercomputers, this did not affect our applications

Figure 25: Mach contour slices every 2 diameters in the wake region for the viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

as most of our codes had only a few large messages being communicated at every timestep. For those codes that have high communication to computation ratios, COCOA was not found to be an ideal platform because of the high latency. With several more enhancements planned for COCOA in the near future, including the addition of several more nodes and increasing the networking bandwidth by addition of more fast-ethernet cards to every node, the performance is expected to increase. Faster interconnect networks could also be used, such as Myrinet, Gigabit, or ATM; but these would increase the cost of the system by approximately 50%.

References

- [1] Long, L. N., Liu, J., and Modi, A., "Higher Order Accurate Solutions of Ship Airwake Flow Fields Using Parallel Computers," *RTO AVT Symposium on Fluid Dynamics Problems of Vehicles Operating near or in the Air-Sea Interface*, Oct. 1998.
- [2] Liu, J. and Long, L. N., "Higher Order Accurate Ship Airwake Predictions for the Helicopter/Ship Interface Problem," *54th Annual American Helicopter Society Forum*, 1998.

Figure 26: C_p contours for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

- [3] Rhoades, M. M. and Healey, J. V., "Flight Deck Aerodynamics of a Non-aviation Ship," *Journal of Aircraft*, Vol. 29, No. 4, 1992, pp. 619–626.
- [4] Hurst, D. W. and Newman, S., "Wind Tunnel Measurements of Ship Induced Turbulence and the Prediction of Helicopter Rotor Blade Response," *Vertica*, Vol. 12, No. 3, 1988, pp. 267–278.
- [5] Williams, S. and Long, K. R., "Dynamic Interface Testing and the Pilots Rating Scale," *The 53rd AHS Forum*, Apr. 1997.
- [6] Tomboulides, A. G., *Direct and Large-Eddy Simulation of Wake Flows: Flow Past a Sphere*, PhD Dissertation, Princeton University, Jun. 1993.
- [7] Bruner, C. W. S., *Parallelization of the Euler equations on unstructured grids*, PhD Dissertation, Virginia Polytechnic Institute and State University, May 1996.
- [8] Pirzadeh, S., "Progress Toward a User-Oriented Unstructured Viscous Grid Generator," *AIAA Paper 96-0031*, Jan. 1996.
- [9] Schweitzer, S., *Computational Simulation of Flow around Helicopter Fuselages*, M.S.

Figure 27: Mach contours and streamlines (at $t = 0.0, 2.75$) for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

Thesis, The Pennsylvania State University, Jan. 1999.

- [10] Modi, A., *Unsteady Separated Flow Simulations using a Cluster of Workstations*, M.S. Thesis, The Pennsylvania State University, May 1999.
- [11] Modi, A., "COst effective COmputing Array," <http://cocoa.ihpca.psu.edu>, 1998.
- [12] Kantor, A., Long, L. N., and Micci, M. M., "Molecular Dynamics Simulation of Dissociation Kinetics," *AIAA Paper 2000-0213*, Jan. 2000.

Figure 28: Mach contours for viscous sphere run ($M_\infty = 0.2$, $Re = 1000$)

Figure 29: Surface Mach contours for Apache helicopter geometry ($U_\infty = 114$ knots)

- [13] Long, L. N. and Brentner, K. A., "Self Scheduling Parallel Version of Wopwop," *AIAA Paper 2000-0346*, Jan. 2000.
- [14] Hall, C. M. and Long, L. N., "Higher Order Accurate Methods for Vortex Flow Simulations," *AHS Forum, Montreal, Canada*, May 1999.
- [15] Morris, P. J., L. L. N. and Scheidegger, T., "Parallel Computations of High Speed Jet Noise," *AIAA Paper 99-1873*, May 1999.

Figure 30: Surface U/U_∞ contours for GSS geometry for 30° yaw case

Figure 32: Surface U/U_∞ contours for GSS geometry for flow speeds of 40 knots and yaw angle of 90°

Figure 31: Surface U/U_∞ contours for GSS geometry for flow speeds of 40 knots and yaw angle of 30°

Figure 33: Surface U/U_∞ contours for GSS geometry for flow speeds of 40 knots and yaw angle of 150°

- [16] Woo, A., "NAS Parallel Benchmarks," <http://science.nas.nasa.gov/Software/NPB/>, 1997.
- [17] Keller, J. and Smith, E., "Analysis and Control of the Transient Shipboard Engagement Behavior of Rotor Systems," *Proceedings of the 55th Annual National Forum of the American Helicopter Society (Montreal, Canada)*, May 1999.
- [18] Healey, J. V., "The Prospects for Simulating

the Helicopter/Ship Interface," *Naval Engineers Journal*, Mar. 1987, pp. 45–63.

- [19] Healey, J. V., "Establishing a Database for Flight in the Wake of Structures," *Journal of Aircraft*, Vol. 29, No. 4, 1992, pp. 559–564.
- [20] Tai, T. C. and Cario, D., "Simulation of DD-963 Ship Airwake by Navier-Stokes Method," *AIAA Paper 93-3002*, 1993.

Figure 34: Region of interest on the GSS for the helicopter rotor problem

Figure 35: Surface U/U_{∞} contours for LHA ship geometry (Yaw angle $\beta = 5^\circ$, $U_{\infty} = 25$ knots)

Number Range of 30-2000,” *Trans. of ASME I: Journal of Fluids Engg*, Vol. 106, Mar. 1984, pp. 66–73.

- [25] Wu, J. and Faeth, G., “Sphere wakes in still surroundings at intermediate Reynolds numbers,” *AIAA Journal*, 1993.

- [21] Landsberg, A. M., Young, T. R., J., and Boris, J. P., “An efficient, Parallel Method for Solving Flows in Complex Three-Dimensional Geometries,” *AIAA Paper 94-0413*, 1994.

- [22] Zan, S., “Private communication,” *NRC, Canada*, 1998.

- [23] Schlichting, H., *Boundary-Layer Theory*, McGraw Hill Co., 7th ed., 1979.

- [24] Modi, V. J. and Akutsu, T., “Wall Confinement Effects for Spheres in the Reynolds