Q422101

Pgs 10

1999208103

GRANT
TECH. RPT.
IN-34

Implementation of Parallel Computing Technology to Vortex Flow

Final Report
for
Grant #NCC25292

Jennifer Dacles-Mariani
Mohamed Hafez
UC Davis

Aug 1, 1998 to July 31, 1999

Motivation:

Wake vortex prediction has been the focus of numerous studies in an effort to minimize aircraft separation distances during approach to landing and to determine the hazard that they posed to other following aircraft. Most flow prediction model the wake region where the vortex has been assumed to have completed its roll-up process (intermediate-wake). But it is also equally important to model the initial vortex formation/roll-up process and the near-wake for accurate drag prediction. Characterizing tip vortex flows involve several regions of interest that are distinct but interdependent. One difficulty experienced in resolving such flows is the grid density requirement. Most flow prediction model the wake region where the vortex has been assumed to have completed its roll-up process. Also, previous effort concentrated on resolving the fine details of the flow without paying too much attention to computational cpu requirement. While some success was obtained in resolving these issues, grid density requirement remain an issue. The availability of a supercomputer is needed in these computations which is not readily available to everyone.

Introduction:

Mainframe supercomputers such as the Cray C90 was invaluable in obtaining large scale computations using several millions of grid points to resolve salient features of a tip vortex flow over a lifting wing. However, real flight configurations require tracking not only of the flow over several lifting wings but its growth and decay in the near- and intermediate- wake regions, not to mention the interaction of these vortices with each other. Resolving and tracking the evolution and interaction of these vortices shed from complex bodies is computationally intensive. Parallel computing technology is an attractive option in solving these flows.

In planetary science vortical flows are also important in studying how planets and protoplanets form when cosmic dust and gases become gravitationally unstable and eventually form planets or protoplanets.The current paradigm for the formation of planetary systems maintains that the planets accreted from the nebula of gas and dust left over from the formation of the Sun. Traditional theory also indicate that such a preplanetary nebula took the form of flattened disk. The coagulation of dust led to the settling of aggregates toward the midplane of the disk, where they grew further into asteroid-like planetesimals. Some of the issues still remaining in this process are the onset of gravitational instability, the role of turbulence in the damping of particles and radial effects. In this study the focus will be with the role of turbulence and the radial effects.

Description of Work:

In this study, a parallel comutational code INS3D-MPI is developed. The validity and applicability of this code is explored by studying the 3-D steady Burgers vortex. Also, the development of a two-dimensional simulation of a solar nebula are discussed first followed by the parallel implementation in the discussion of results section.

Previous study by the author attempted to quantify to what extent numerics and turbulence models affect the accuracy of tip vortex flow prediction during the formation, growth and decay stages (i.e., wing surface and near-wake region). With a fine-grid solution using 2.5 million grid points resolving the flow around the wing and 0.75chord downstream, a total of 28 cpu hrs was required to obtain a converged solution for this case. The previous study indicated that this was the minimum grid requirement one needs to resolve the details of the flow given in this short domain. In a real flight configuration, this is just a small portion of the flowfield. One can conceivably require several million grid points to resolve the flow around the body, another 1-2 million grid points to track the interaction of the tip vortices and several million of grid points more to convect the vortices downstream. Clearly, such an undertaking will be very computationaly demanding on a mainframe computer and impossible to run on a workstation or a personal computer. The notion of parallelizing a code to parcel the flow domain into several cpus or workstations is very appealing.

The experimental results conducted along with this study indicated that the Reynolds stress and the mean strain rate are not aligned. This important finding implied that an eddy viscosity approach (constant or isotropic) will most likely not be successful. Since a full Reynolds stress model is not a realizable option at his point, two one-equation models often used in aerodynamic calculations were selected; the Baldwin-Barth one-equation model and the Spalart-Allmaras model. Preliminary computations indicate that these models though successful in predicting the overall flow pattern both over predicted the level of eddy viscosity in the core. This shortcoming was worked around through the modification of the production terms and implemented in both of the models.

Formation of planets is believed to occur as a by-product of collisional accretion of comet-sized planetesimals. These particles settle toward the midplane of a flattened, rotating nebula disk which consists of interstellar dust and gas. An area of uncertainty is the transition from a gas-dominated accretion disk to a disk of comet-sized planetesimals. In this study the development of a two-dimensional gas/particle phase flow is undertaken. To start, since stability was an issue, the gas and particle equations were solved using a direct solver. However, this approach proved to be too slow. Since the particle momentum equations pose a stiff system of equations, convergence to a steady state solution took a long time. The next step was to use a semi-implicit Gauss-Seidell line relaxation scheme This algorithm proved to be stable for both the particle and gas. The viscous terms are discretized using staggered grid approach. This discretization naturally leads to second-order accuracy on both the convective and diffusion terms. The formulation is compact and so mass is conserved more accurately in the discrete level. The method of solution is as follows: the gas equations are solved first using a block tri-diagonal matrix. The domain is swept one way and then another for several subiteratons before updating the particles. The particles are solved using the same block solver but this time using a marching scheme with sub-iterations for each radial station. The boundary conditions used at the inner and outer radius are Neumann boundary conditions. At the midplane, the symmetry conditions are imposed and at the top boundary, either Nakagawa boundary conditions or Neumann conditions are used interchangeably.The initial boundary conditions user are the inviscid set of flows.

Discussion of Results:

The production code, INS3D-UP with its multi-grid, multi-block capability is a very good candidate for parallel applications on distributed memory machines. Originally, several methods of parallelizing the code were looked into including PVM and MPI message passing routines and HPF(High Performance Fortran). During the course of the study, it was decided that the best way to approach the problem is to pick one of these methods and implement the method into INS3D-UP. The MPI message passing approach was chosen over PVM since this is now the current standard in message passing utilities. HPF would involve recoding INS3D and is thought to be more labor intensive than using MPI.

The implementation of the MPI library was begun on the current serial version of INS3D-UP. The first task was to input the MPI coding in place but tested only for the serial, multi-zone version. The suite of test problems currently distributed as part of the serial INS3D-UP were used to test this along with a three-dimensional analytical Burgers vortex. The next task was to partition the grid blocks over the MPI nodes. The domain was partitioned a-priori with communication between blocks accomplished by providing interpolation stencils from PEGSUS. Next a preprocessor that maps the zones into nodes was developed (hence, the capability to do static load balancing). This also estimates the scalability of the current problem. The majority of the subroutines in INS3D-UP were modified to reflect the fact that each MPI process "owns" only a subset of the zone blocks. The boundary condition message passing was enabled by implementing MPI routines to send and receive boundary condition data for interface between zones.

Currently, INS3D-MPI has several choices of distributing(parallelizing) the boundary condition interchanges. In the next page the performance of the code in solving the 3-D Burgers Vortex problem is outlined: Table 1 has the streamwise sweeping (J-direction) performed inside the sweep loop. Here, the efficienct drops off dramatically with 4 workers. Table 2 does the j- -sweep bc interchange outside of the main relaxation loop but the results are similar to Table1. The same results can be seen in Tables 3 and 4 but this time using both j- and k-sweeps.inside and outside the mail loop, respectively. Tables 5 and 7 require sweeps in all j-, k-, and l-directions. The efficiency with 4 workers is very promising. However, more iterations is also needed to converge the solution. Finally, Table 7 shows the performance that one gets when the interchange is done only once at the end of the cycle. The bc interchange system adapted for this case is the bc-exchange 3b.

(3-D Steady Burgers Vortex Solution)

Table 1: B.C. Exchange 1a(J-sweep interchange inside the sweep loop

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4382	1.00	100.0	99
2	3616	1.21	60.5	99
4	3209	1.37	34.2	99

Table 2: B.C. Exchange 1b(J-sweep interchange bc once outside of sweep loop

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4396	1.00	100.0	99
2	3603	1.22	61.0	99
4	3209	1.37	34.2	99

Table 3: B.C. Exchange 2a(J- and K-sweeps interchange inside the sweep loop

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4395	1.00	100.0	99
2	3007	1.46	73.0	99
4	2274	1.93	48.3	99

Table 4: B.C. Exchange 2b(J- and K-sweeps interchange bc once outside sweep loops

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4393	1.00	100.0	100
2	3013	1.46	72.9	100
4	2279	1.93	48.2	100

Table 5: B.C. Exchange 3a(J-, K- and L-sweeps interchange inside the sweep loop

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4403	1.00	100.0	99
2	2229	1.98	98.8	99
4	1137	3.87	96.7	99

Table 6: B.C. Exchange 3b(J-,K- and L-sweeps interchange bc once outside sweep loops

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4466	1.00	100.0	102
2	2249	1.99	99.5	102
4	1143	3.91	97.7	102

Table 7: B.C. Exchange 4(Interchange bc once at the very end of the routine

Workers	Wall Time	Speedup	Efficiency	Ntmax
1	4844	1.00	100.0	111
2	2446	1.98	99.0	111
4	1241	3.90	97.5	111

Formation of planets is believed to occur as a by-product of collisions accretion of comet-sized planetesimals. These particles settle toward the midplane of a flattened, rotating nebula disk which consists of interstellar dust and gas. An area of uncertainty is the transition from a gas-dominated accretion disk to a disk of comet-sized planetesimals. In this study the development of a two-dimensional gas/particle phase flow is undertaken. The Reyonlds-Averaged Navier-Stokes Equations are discretized using an implicit finite-differencing technique on a staggered mesh formulation. The staggered-grid arrangement avoids the "checker-board" pattern that one gets on a regular mesh. As a result the variables are strongly coupled and can conserve mass more compactly. The equations are second-order accurate. This code also has the capability of using the two turbulence models previous used in the early explicit version of the code. The calculation starts by forming an entire numerical matrix equation from values at the previous time level. Several choices of sweeping algorithms were investigated. It was found that since the particle equations behave in a parabolic manner similar to a boundary layer a marching scheme starting from the inner radial station marching toward the outer radial station was devised. Several sub-iterations are done at each radial station before advancing to the next one. The gas equations on the other hand can be swept using a back-and-forth sweeping procedure without the inner subiterations which was done for the particle equations. Once the gas and particle variables are obtained, the turbulence model and the Schmidt Number are implemented. The cycle is repeated until steady-state is reached.

Figures 3a-b show the particle and gas velocity profiles at 1.01 AU. In this run a grid size of 41x302 in the radial and vertical directions, respectively, are used. The radial domain is extended from 1 AU to 1.4 AU. The vertical extends to 100,000 kilometers. Here, the 60 cm size particles are first evolved from an inviscid and equilibrium state. After 480 cycles, the gas velocities behave in an Ekman -like flow while the particles follow the gas. The particles are shown to have settled below 20,000 kilometers.

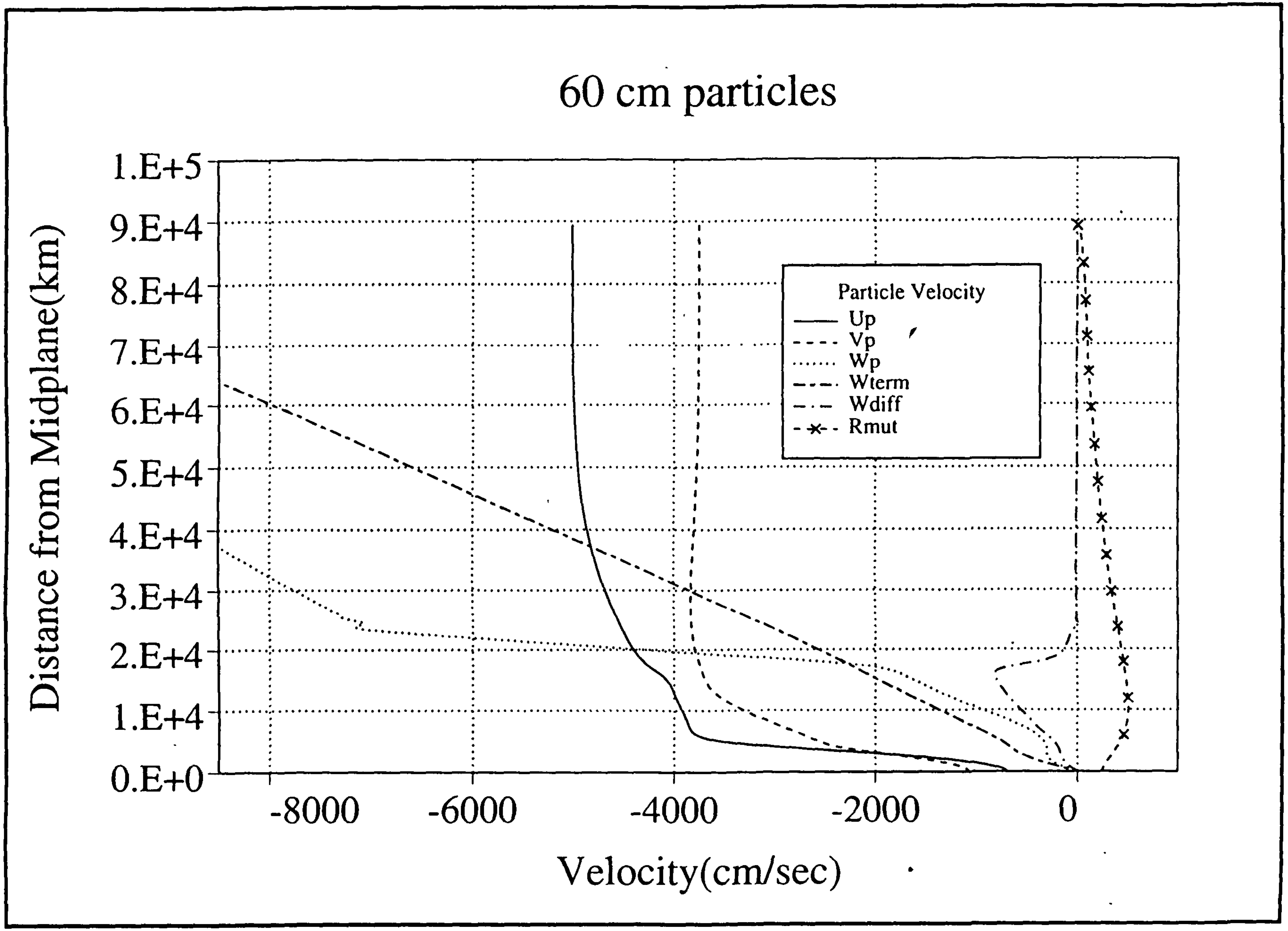

Fig.3A Particle Velocity Profiles at 1.01 AU

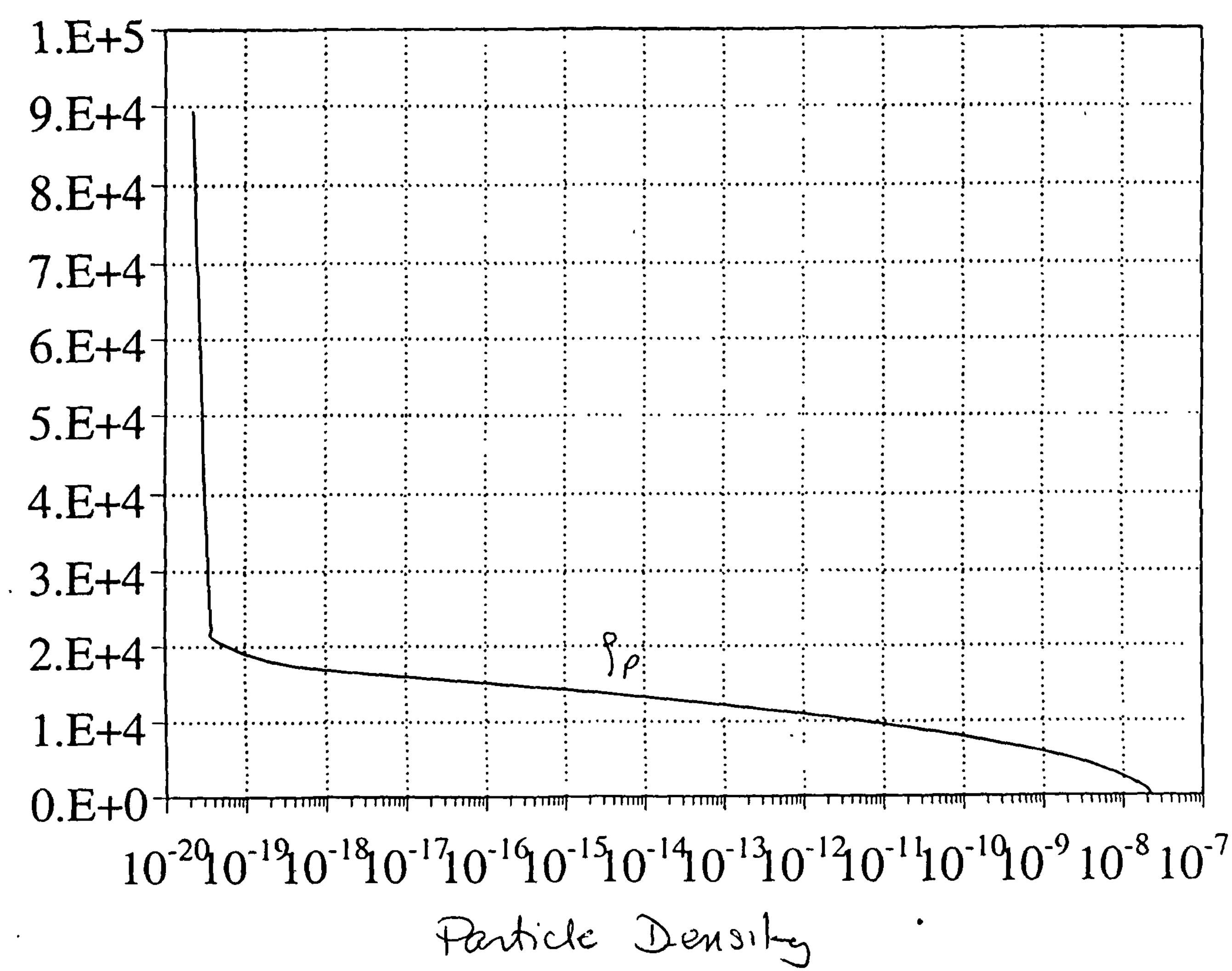

Fig3B. Particle Density Profile at 1.01 AU

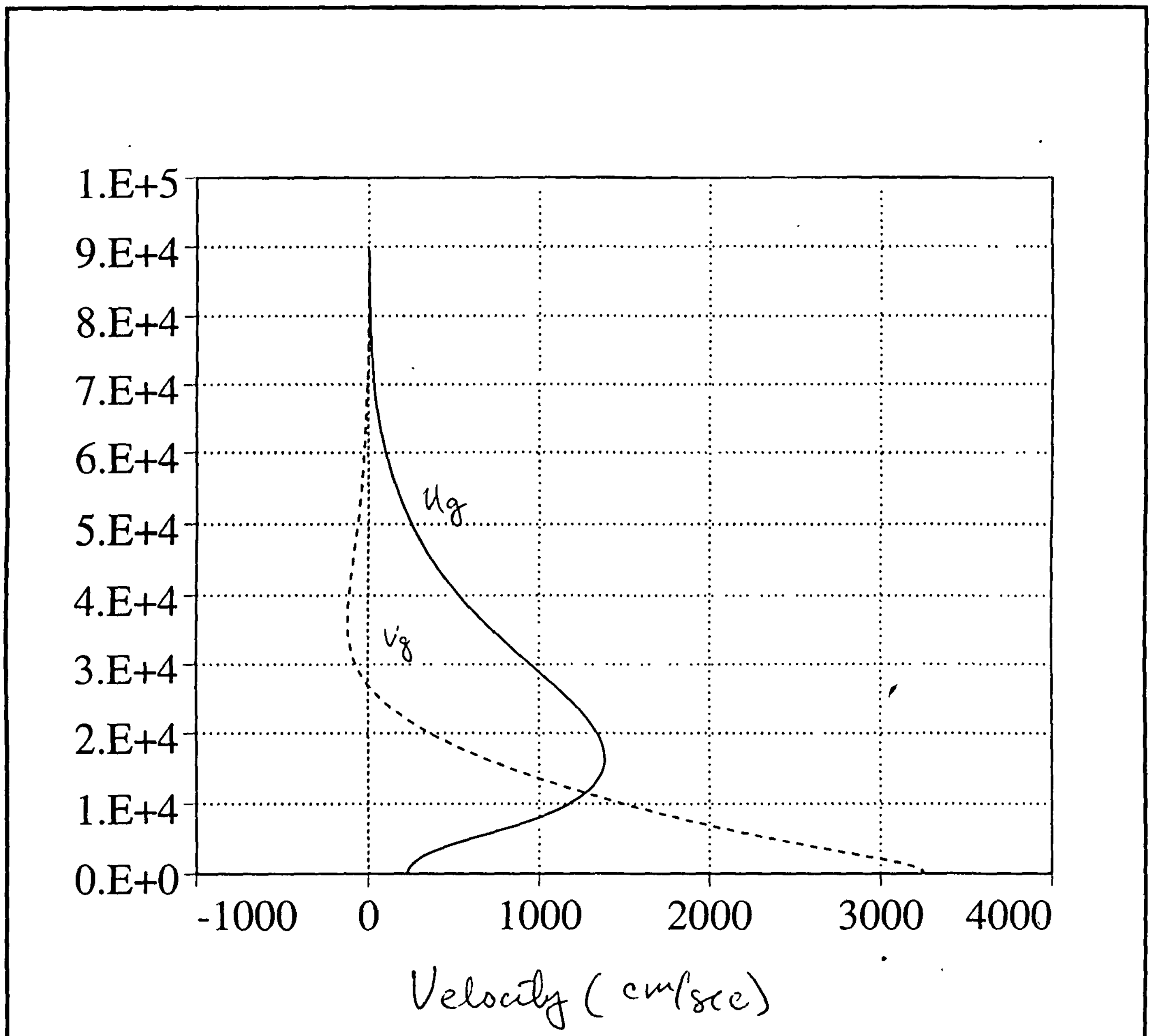

Fig 3C. Gas Velocity Profile at 1.0AU.