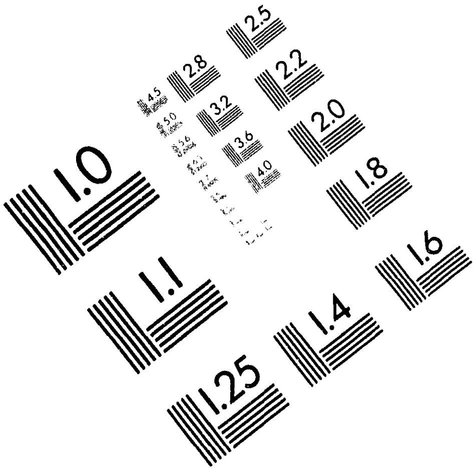
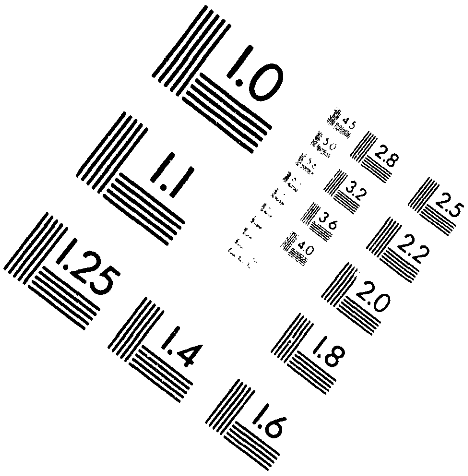


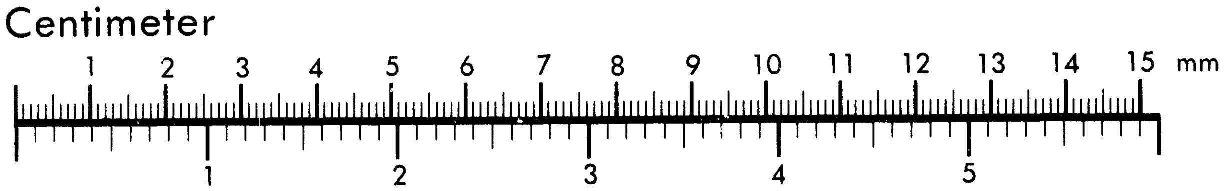
AIIM

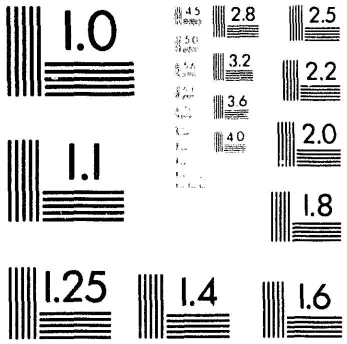
**Association for Information and Image Management**

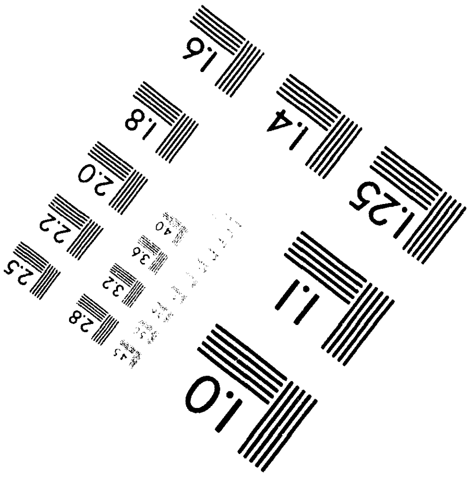
1100 Wayne Avenue, Suite 1100
Silver Spring, Maryland 20910

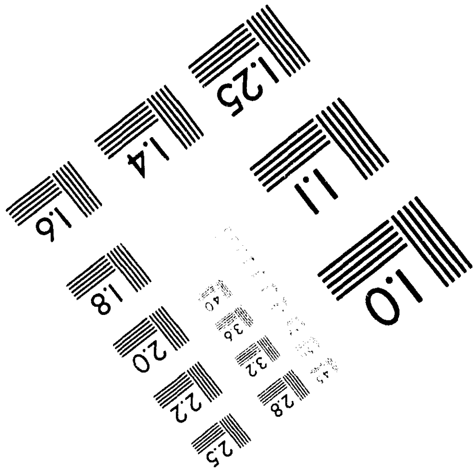
301/587-8202











MANUFACTURED TO AIIM STANDARDS
BY APPLIED IMAGE, INC.

# FLOW VISUALIZATION FOR LAGRANGIAN PARTICLE METHODS[1]

Computational Fluid Dynamics Track

Micheal W. Glass and Steven N. Kempka
Computational Fluid Dynamics, Dept. 1511
Sandia National Laboratories
P.O. Box 5800 MS 0827
Albuquerque, NM 87185-0827 USA
(505)844-8451
mwglass@engsci.sandia.gov

## ABSTRACT

In particle methods, each particle represents a finite region over which there is a distribution of the field quantity of interest. The field value at any point is calculated by summing the distribution functions for all the particles. This summation procedure does not require the use of any connectivities to generate continuous fields. Various AVS modules and networks have been developed that enable us to visualize the results from particle methods. This will be demonstrated by visualizing a numerical simulation of a rising, chaotic bubble. In this fluid dynamics simulation, each particle represents a region with a specified vorticity distribution.

## INTRODUCTION

Lagrangian, vorticity-based methods are a subclass of Lagrangian particle methods which can be used to simulate complex flows. Vortex methods are a class of computational fluid dynamics (CFD) techniques that have distinct advantages over traditional techniques such as finite difference and finite element methods. Some advantages of vortex methods are:

- A grid is not required in the fluid volume. Only boundaries need be discretized.
- Large errors associated with discrete approximations to nonlinear convective terms in transport equations are avoided. Some vorticity formulations do not require any approximations to spatial derivatives.
- Numerical algorithms for vortex methods are simple compared to implementations of finite element or finite difference methods. This reduces the time for development and maintenance.
- Execution times can be much faster for vortex methods than for other approaches.
- Much higher spatial resolution of flow fields can be obtained since resolution is naturally focused on the areas of interest.

[1] This work performed at Sandia National Laboratories and supported by the U.S. Department of Energy under Contract DE-AC04-94AL85000.

At a particular time, a solution field from Lagrangian particle methods is represented by a collection of particles in which each particle has various field variables associated with it along with its spatial position. The number of particles does not have to be constant and their position will change during the course of a simulation. Because the particles often have no connectivity or ordering, traditional visualization tools such as those applied to grid based finite difference or finite element analysis are not applicable. In this paper we present our experience using AVS to visualize the results of a gridless vorticity-based simulation.

## A BUBBLE PROBLEM

Consider the evolution of an initially circular (two-dimensional) bubble of light fluid within an infinite domain of heavier fluid, with gravity present (see Figure 1). The bubble of lower density fluid will rise due to buoyant forces, and as it rises, the bubble interface changes from a simple round shape to a geometrically complex shape. The bubble is sufficiently large that surface tension effects are negligible. At later times in the evolution, two notable features of the bubble are the "cap" at the top of the bubble, and the highly mixed region beneath the "cap."

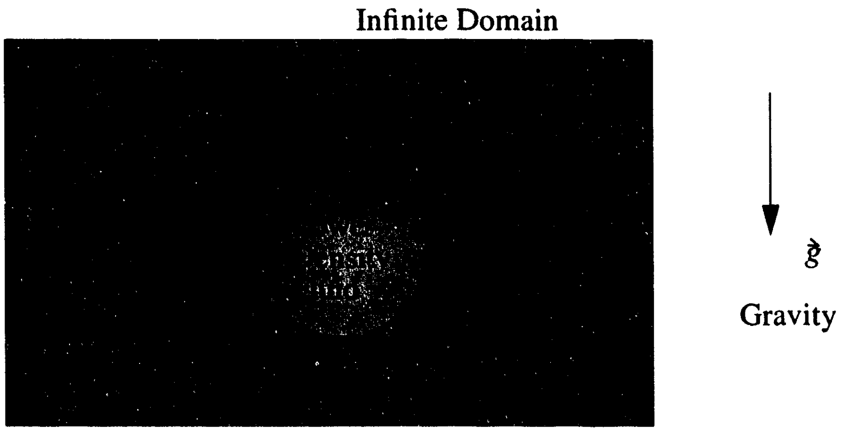


Figure 1 Two-dimensional, circular bubble of fluid within an infinite domain of higher density fluid.

Although the evolution from a simple circular shape to the complex shape is complex and nonlinear, an understanding of the phenomena can be gained by considering the flow in terms of its vorticity. The bubble evolution is driven entirely by a vortex sheet that lies on the bubble interface, which includes the swirled region below the "cap." All the vorticity is generated baroclinically and all the vorticity remains on the bubble interface, assuming that diffusive transport is negligibly small. Thus, in the context of a vortex method, the bubble interface location is determined with no additional effort. This fortuitous coincidence occurs because vorticity convects at the local fluid velocity, and the bubble interface also convects at the local fluid velocity, according to the definition of a fluid interface. These features make vortex methods particularly convenient for considering bubble problems.

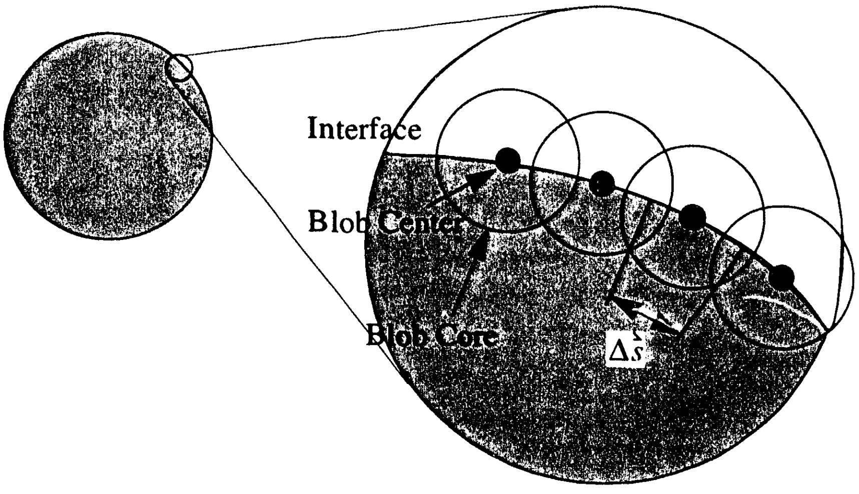


Figure 2 Vortex blob representation of the vortex sheet on the bubble interface.

A vortex model of the bubble consists of representing the vortex sheet on the bubble interface with a collection of vortex blobs. The simulation consists of calculating the time evolution of the circulation of each vortex due to baroclinic generation of vorticity, and the motion of each vortex due to the velocity field induced by all the other vortices. The vortex blob representation of a typical portion of the interface $\Delta\vec{s}$ is shown in Figure 2.

The Boussinesq form of the Navier-Stokes equations is used to describe the bubble evolution, and are:

$$\frac{\partial \rho}{\partial t} + \vec{u} \bullet \nabla \rho = 0 \tag{1}$$

$$\rho_o\left(\frac{\partial}{\partial t}\vec{u} + \vec{u} \bullet \nabla \vec{u}\right) = -\nabla P + (\rho - \rho_o)\vec{g} \tag{2}$$

Equation (1) describes the space and time evolution of the density field $\rho$, and equation (2) describes the evolution of the velocity field $\vec{u}$, which is driven by the variations in the density field. The essence of the Boussinesq approximation is that the *difference* between the local density and a reference density $\rho_o$ determines the buoyancy force, and the term $\rho_o\vec{g}$ is included in the pressure gradient $\nabla P$. This formulation is accurate only if $\rho - \rho_o \ll \rho_o$. The equivalent vortex model formulation of equations (1) and (2) is (*e.g.*, [1])

$$\frac{d\Gamma_i}{dt} = -\left(\frac{(\rho - \rho_o)\vec{g}}{\rho_o}\right) \bullet \Delta\vec{s}_i \tag{3}$$

$$\vec{u}(\vec{x}_i) = \sum_{i = 1, i \neq j}^{N} \frac{(\vec{x}_i - \vec{x}_j) \times \Gamma_j \hat{k}}{|\vec{x}_i - \vec{x}_j|^2} \tag{4}$$

$$\frac{d\vec{x}_i}{dt} = \vec{u}(\vec{x}_i) \tag{5}$$

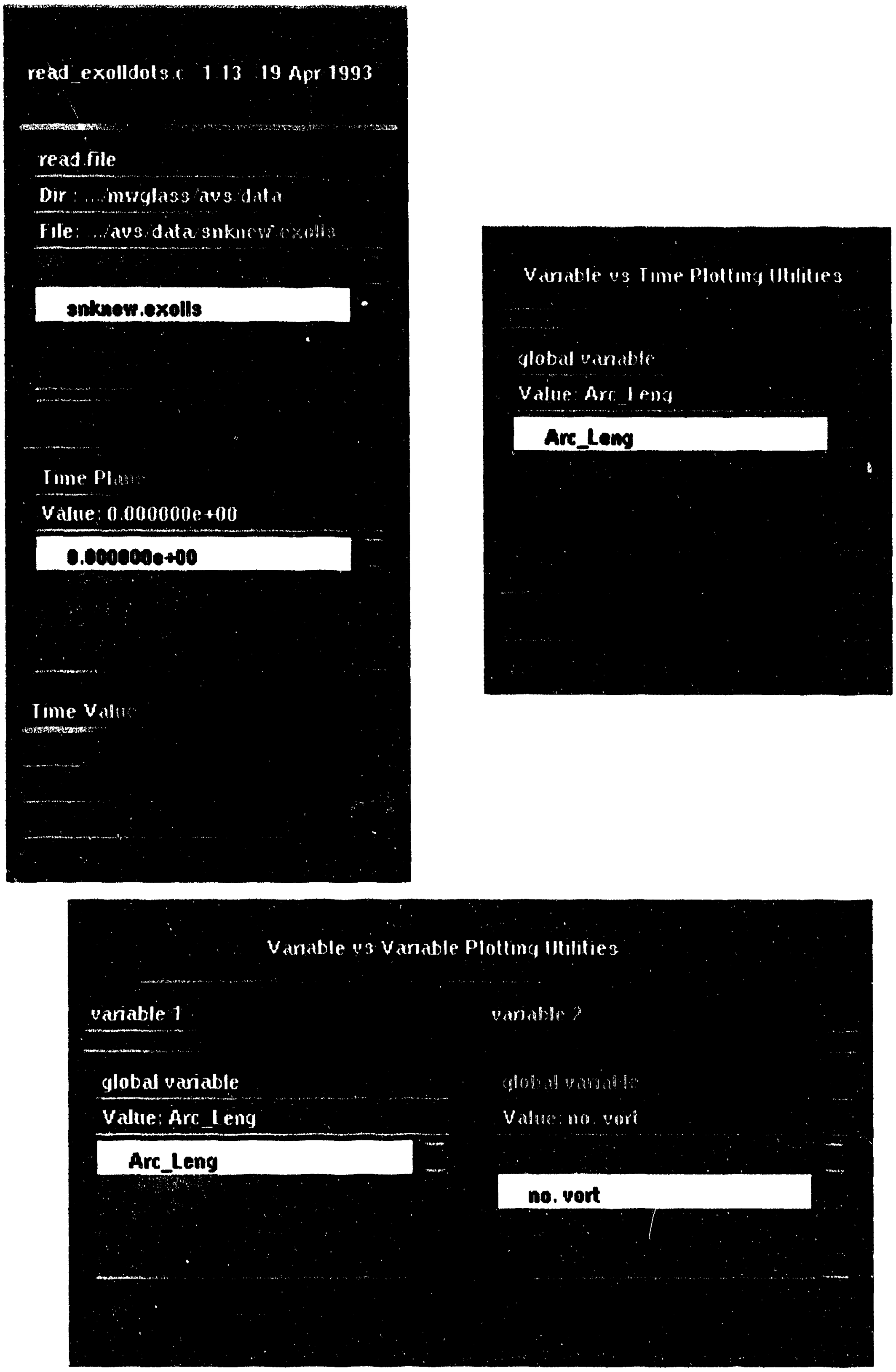
$$\Gamma_i = \int_A \vec{\omega}_i \bullet \vec{n}\, dA \tag{6}$$

$$\vec{\omega}_i(\vec{x}, t) = \frac{\Gamma_i(t)}{\pi\sigma^2} e^{-\left(\frac{|\vec{x} - \vec{x}_i|}{\sigma}\right)^2} \tag{7}$$

Equation (3) determines the circulation $\Gamma_i$ of the *i-th* vortex, and is based on the density difference $\rho - \rho_o$ between the fluid inside the bubble and the fluid outside the bubble, the fluid density inside the bubble $\rho_o$, the gravity vector $\vec{g}$, and the orientation of the arc length of the interface, $\Delta\vec{s}$. Equation (4) is a discrete form of the Biot-Savart Law which describes the velocity of each vortex, $\vec{u}(\vec{x}_i)$ , that is induced by the $N$ vortices at locations $\vec{x}_i$. Equation (4) can also be used to calculate the velocity at arbitrary locations in the domain. Equation (5) describes the translation of each vortex at the location fluid velocity. Equation (6) defines the circulation and equation (7) describes the vorticity distribution for each vortex blob where the subscript $i$ denotes a particular vortex blob, $\vec{x}_i$ is the center of the *i-th* vortex blob, $\Gamma_i$ is the circulation of the *i-th* vortex blob, and $\sigma$ is the core radius.

## VISUALIZING DISCRETE FIELD VARIABLES

AVS's scatter field data format lends itself well to visualizing particle data. The output from the Lagrangian vortex code is a modified version of Sandia's ExodusII finite element database [3]. The Exodus II database format was used because: (1) it already exists; (2) it is built upon NetCDF [2] so it is portable across machine architectures; (3) it supports transient data; and, (4) an AVS module for reading Exodus II databases exists which could be easily modified for particle data. To use the finite element database format for particle data, one can think of a one-to-one mapping of each particle to a node in the database. At each time step, the database has a global variable specifying the number of particles that are present at that particular time step. The nodal data for each time step is associated with the particle data. At present, the nodal data contains the particle's: (1) spatial location; (2) an ID number that is unique to that particle throughout the simulation; (3) velocity components; (4) circulation; (5) rate of change of circulation; (6) core radius; and, (7) length. At this time, the arc length of the bubble interface is also stored as a global variable at each time step. The AVS reader for accessing the database extracts each particle's spatial location and constructs a scatter field with the other nodal data as scalar data except for the particle IDs. The particle IDs are only used by the reader when interpolating particle positions at intermediate time values. The reader's interface is shown in Figure 3. The reader also has various controls for accessing transient data and also can extract variable vs. time or variable vs. variable data for graph plotting.



read_exolldots.c 1.13 19 Apr 1993

read file

Dir: .../mvglass/avs/data

File: .../avs/data/snknew/exolls

snknew.exolls

Time Plane

Value: 0.000000e+00

0.000000e+00

Time Value

Variable vs Time Plotting Utilities

global variable

Value: Arc_Leng

Arc_Leng

Variable vs Variable Plotting Utilities

variable 1

global variable

Value: Arc_Leng

Arc_Leng

variable 2

global variable

Value: no. vort

no. vort

Figure 3 Control panel for the modified ExodusII reader.

An example AVS network for visualizing the resulting scatter field is shown in Figure 4. The righthand branch of the network extracts each particle's core radius and allows the user the option of scaling the radius for visual purposes. The lefthand branch extracts other scalar data and maps a color value to each particle depending upon its data value. The color values and radius values are then recombined into a scatter field and fed into the *scatter dots* module to generate colored spheres which are then viewed by the *geometry viewer.* Other than the reader module, two other new modules are in the network - *snl colorizer* and *snl legend. snl colorizer* is similar to the AVS *colorizer* module in that it converts a field of data to color values. However, it has the capability of overriding the data's minimum and maximum values so the data-to-color mapping can remain constant with transient data. The module also outputs the minimum and maximum values which are used by *snl legend* to create a color-to-data mapping legend for the *geometry viewer. snl legend* is similar to the AVS *color legend* module but displays the actual data values on the legend and can display the labels in either decimal or scientific notation.

Figure 5 shows the three visualization techniques possible with the network shown in Figure (4) that can be readily used on the bubble problem. In Figure (5a), the *scatter dots* module is used to connect the particles with a line. This technique allows the analyst to examine the shape of the bubble interface. For high quality output, the AVS CLI command *geom_save_postscript* is used to create a postscript file. To get a better feel of where the lighter fluid is, the output from the *geom_save_postscript* command can be easily filtered to produce a filled region as shown in Figure 5b. The same result has been achieved from within AVS by using a modified version of the *scatter dots* module to create the filled region. And lastly, a colormapped representation with a time stamp and legend can be created as shown in Figure (5c).

## VISUALIZING CONTINUOUS FIELD VARIABLES

In the simulation, each vortex blob is assumed to have a Gaussian vorticity distribution as shown by equation (6). The actual vorticity field is obtained by superposing the Gaussian fields of the individual vortices by,

$$\vec{\omega}(\vec{x}, t) = \sum_{i=1}^{N} \frac{\Gamma_i(t)}{\pi\sigma^2} e^{-\left(\frac{|\vec{x}-\vec{x}_i|}{\sigma}\right)^2} \qquad (8)$$

If the vorticity field is to be calculated for a grid superimposed on the domain, then,

$$\vec{\omega}(\vec{x}_{jk}, t) = \sum_{i=1}^{N} \frac{\Gamma_i(t)}{\pi\sigma^2} e^{-\left(\frac{|\vec{x}_{jk}-\vec{x}_i|}{\sigma}\right)^2} \qquad (9)$$

An AVS module has been written to perform this calculation. For each grid point, the summation in equation (9) must be evaluated. For a high resolution grid and a large number of particles, this can be very computationally intensive. Because this module can be so computationally intensive, two features have been implemented in the module to make it more interactive. First, the grid resolution can be specified in the module control panel so low resolution images can be

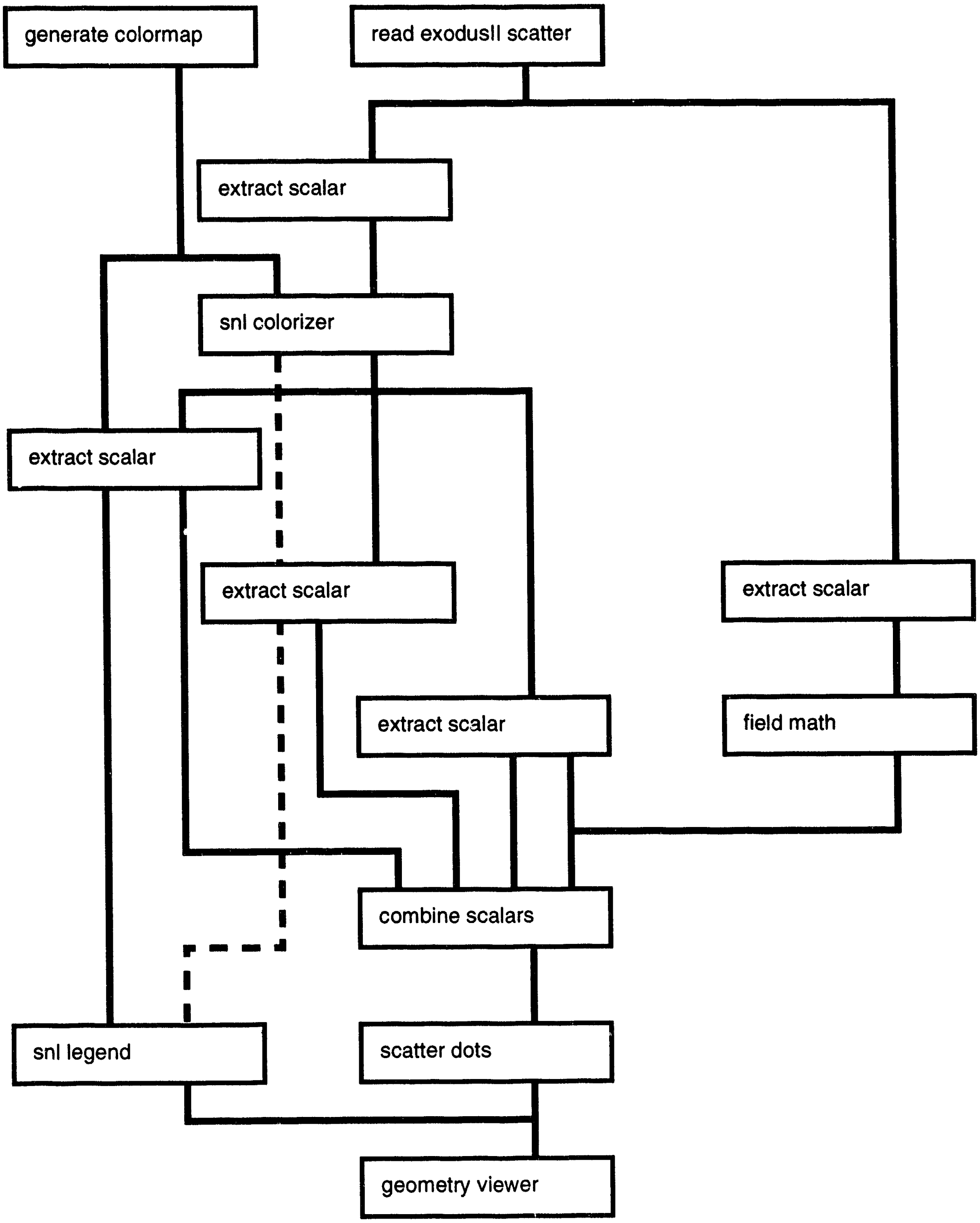


Figure 4 Sample network for visualizing particle data from a Lagrangian vortex code.

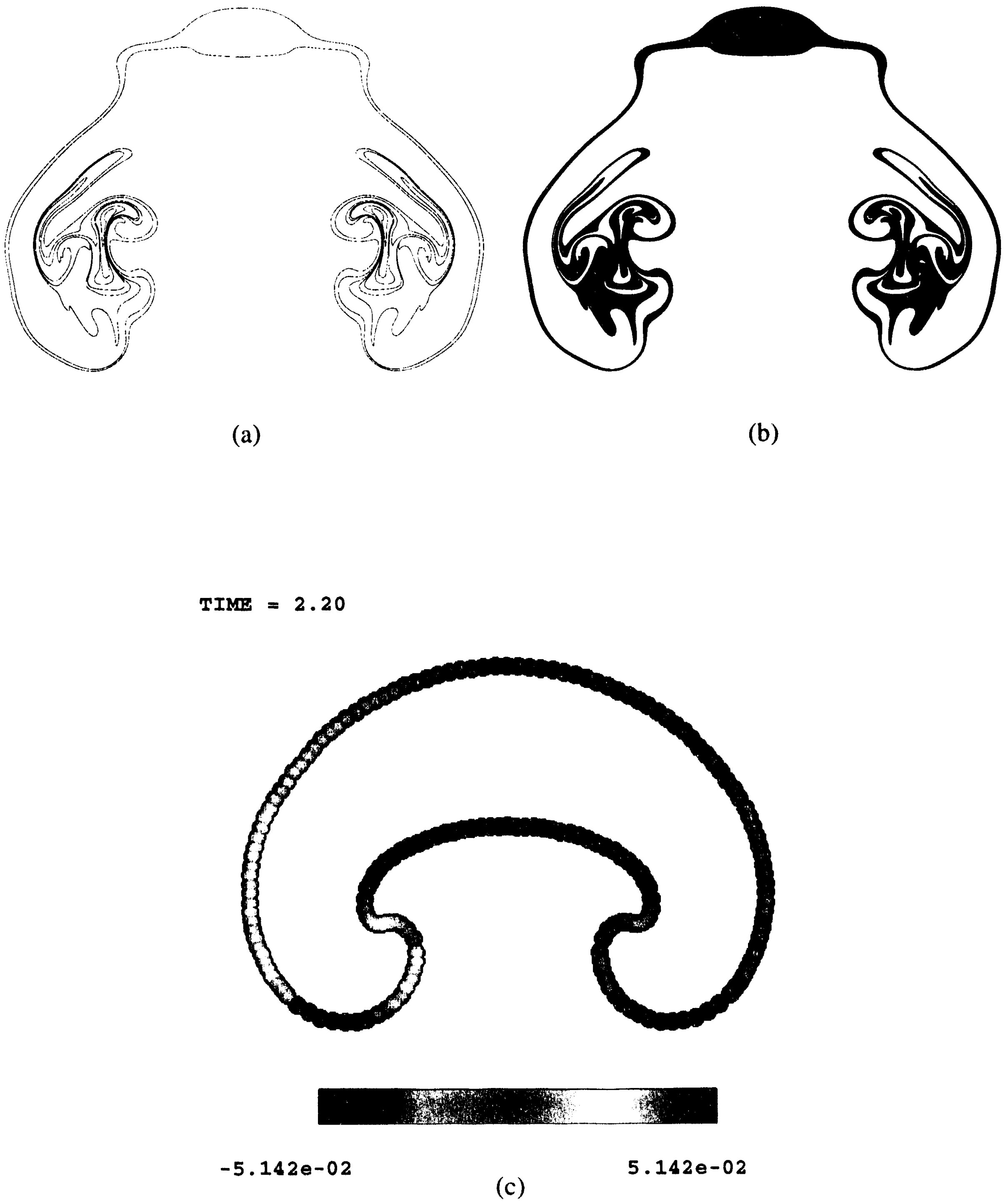
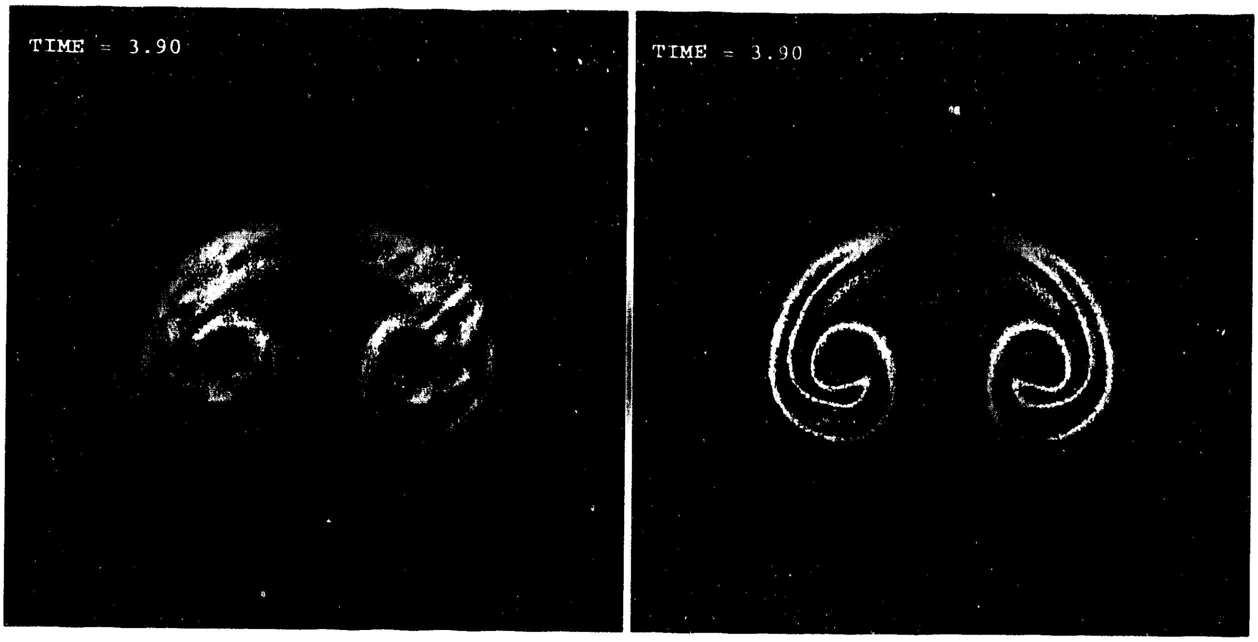


Figure 5 Three visualization techniques for the bubble problem.



(a) 50 x 50

(b) 200 x 200

Figure 6 Vorticity field calculated with various grid resolutions.

quickly viewed to allow the analyst to get a quick feel of what the simulation results look like before using a higher resolution grid for more indepth analysis. Secondly, the module has been multithreaded using Sun Microsystems's Solaris threads. This allows the module to be remotely executed on our department's six processor Sparc Center 2000. The number of threads is also selectable from the module's control panel. Multithreading shows an almost linear speedup as the number of threads is increased. Figure 6 shows the results for different grid resolutions. The field can also be transformed into a 3D representation with the AVS *field to mesh* module. The actual quantity visualized in Figures 6 is,

$$\hat{\omega}_{jk}(\vec{x}_{jk}, t) = \left(\frac{\left|\vec{\omega}_{jk}(\vec{x}_{jk}, t)\right|}{\omega_{max}}\right)^{0.2} \qquad (10)$$

## REFERENCES

1 Anderson, C.R., "A Vortex Method for Flows with Slight Density Variations," J. Computational Physics, vol 61, pp. 417-444, 1985.

2 "NetCDF User's Guide", Unidata Program Center, University Corporation for Atmospheric Research, available from unidata.ucar.edu [128.117.140.3]:/pub/netcdf/netcdf.tar.Z.

3 Schoof, L.A. and Yarberry, V.R., "EXODUS II: A Finite Element Data Model, " Technical Report SAND92-2137, Sandia National Laboratories, Albuquerque, New Mexico, March 1994.

## DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.