CONF-9610343--

ANL/MCS-P--609-0996

INTERACTIVE COMPUTATIONAL MODELS OF PARTICLE DYNAMICS USING VIRTUAL REALITY

Tom Canfield, Darin Diachin, Lori Freitag,
Daniel Heath, and Jim Herzog
MCS Division
Argonne National Laboratory
Argonne, IL 60439

William Michels
Nalco Fuel Tech
1001 Frontiac Rd
Naperville, IL 60566

RECEIVED
DEC 01 1997
OSTI

ABSTRACT

We discuss an interactive environment for the visualization, analysis, and modification of computational models used in industrial settings. In particular, we focus on interactively placing massless, massed, and evaporating particulate matter in computational fluid dynamics applications. We discuss the numerical model used to compute the particle pathlines in the fluid flow for display and analysis. We briefly describe the toolkits developed for vector and scalar field visualization, interactive particulate source placement, and a three-dimensional GUI interface. This system is currently used in two industrial applications, and we present our tools in the context of these applications. We summarize the current state of the project and offer directions for future research.

Keywords: Interactive visualization, computational steering, particle tracking, industrial applications

1 INTRODUCTION

An increasing number of industrial applications rely on computational models to reduce costs in product design, development, and testing cycles. The computational modeling of scientific applications usually involves three distinct phases:

1. The engineers determine an initial mathematical model consisting of a set of partial differential equations (PDEs), boundary conditions, and input parameters that describe the physical phenomenon of interest, for example, fluid flow or stress and strain analysis.

The submitted manuscript has been authored by a contractor of the U. S. Government under contract No. W-31-109-ENG-38. Accordingly, the U. S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U. S. Government purposes.

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.

DISCLAIMER

Portions of this document may be illegible electronic image products. Images are produced from the best available original document.

2. A discrete approximate solution to the system of PDEs is found by using appropriate numerical techniques and methods. The solution space consists of large data sets containing the vector and scalar fields corresponding to the physical variables used in the mathematical model.
3. The data sets are visualized and analyzed by the engineer who modifies the numerical model, input parameters, or boundary conditions to fine-tune the simulation or to study a variety of physical situations. The engineer then returns to Step 2 and computes a new discrete solution.

As indicated in Step 3, this process is iterative and often requires weeks or months of man hours to find the best mathematical model and computational solution for each new application area. In this article, we describe an interactive simulation and analysis tool that can considerably shorten the computational cycle described above by allowing engineers to directly interact with the model from within the visualization environment. Ideally, once the initial numerical solution is understood, the user should be able to easily and intuitively change input parameters or boundary conditions and see the results of those changes in real time. Unfortunately, the computations involved in a typical application are usually too complex to be done in real time even with the use of advanced numerical techniques on state-of-the-art computers.

One application in which real-time calculations are possible is the modeling of particle dynamics in noncoupled, two-phase flows. For this model, we have designed an interactive system that consists of three components: the display device and graphics software, the computational software, and a communication layer that passes information between the two. The display device used for the applications in this article is the immersive Cave Automatic Virtual Environment (CAVE) developed at the University of Illinois at Chicago [1]. The three-dimensional visualizations in the CAVE provides insight and understanding of three-dimensional solution spaces not available when using a two-dimensional screen display mechanism. The computations are done by using the TrackPack software package, which computes the dynamics of massless, massed, and evaporating particles on a variety of meshes for both time dependent and steady state flows. This package is linked to the CAVE visualization environment using the CAVEComm message-passing library developed at Argonne National Laboratory [4]. Using this interactive system, the engineer can easily define particle characteristics and sources from within the visualization environment and communicate these changes to a remote simulation process. The new results are calculated and returned for visualization and further modification in a matter of seconds.

The interactive particle dynamics simulation has several interesting industrial applications, two of which we focus on here. The first is a joint project between Argonne National Laboratory (ANL) and Nalco Fuel Tech (NFT) to design pollution and slag control systems for commercial boilers and incinerators. These systems use an injection-based process to spray noncatalytic reagents directly into the flue gas flows, where they react to reduce pollutants or slag. Optimal performance of the system is obtained by careful placement of the injectors in the boiler with respect to the flue gas temperatures, velocity fields, and slag buildup. Thus, a tool that provides engineers the capability to interactively place the injectors and obtain a quick evaluation of spray coverage is critical to the effective design of this system. To enable this capability, we have implemented mechanisms for the visualization of three-dimensional flue gas flow, slag distribution, and temperature distributions and for the interactive placement of injector nozzles and particulate matter in a virtual boiler.

The second project involves the analysis of flow in the automotive piston chamber of a four cycle engine. Experimentalists are designing a pollution control system for automotive systems based on injecting atomic nitrogen into emission flows. An understanding of the flows in both the piston chamber and the exhaust system is critical to a successful implementation of the experimental system. To visualize the flows in these systems, we expanded the interactive computational model for particulate flow mentioned above to accommodate time-dependent flows and moving meshes.

We have organized the remainder of this article as follows. In section 2, we describe the computational model used in the TrackPack software to study the dynamics of massless, massed, and evaporating particles. In Section 3, we present the interactive environment used to visualize and modify the results of the particle dynamics calculations. The complete environment consists of a number of toolkits ranging from a three-dimensional graphical user interface to visualization and particulate interaction toolkits. We describe each briefly in the context of the two applications mentioned above. Finally, in Section 4, we summarize the current state of the project and discuss directions for future research.

2 COMPUTING PARTICLE DYNAMICS

The computational model used in the two industrial applications discussed here is based on an integrated approach to calculating the dynamics of massless, massed, and evaporating particles in fluid flow. This dynamic behavior depends on a number of physical properties of the flow, including the fluid velocity vector field, fluid temperature, and density. In addition, the dynamics also depend on the particle's initial size, density, temperature, position, and velocity. These input parameters are used in the TrackPack software developed by Diachin and Herzog to allow the user to study massless, massed, and evaporating particles in a noncoupled system. The calculations are the basis for applications that study fluid flow using streamlines, massed particulate modeling in flow (such as the formation of slag on boiler walls), and injection models using evaporation sprays. We now briefly describe the ordinary differential equations governing these particles; a fuller description is available in [2].

Massless particle paths or streamlines are calculated by following the computed vector field by integrating the system

$$\frac{d\vec{x}}{dt} = \vec{V}_p \qquad \text{with} \tag{1}$$

$$\vec{V}_p = \vec{V}_g, \tag{2}$$

where $\vec{x}$ and $\vec{V}_p$ are the particle position and velocity vectors, respectively, and $\vec{V}_g$ is the fluid velocity vector given by the CFD solution data at the point $\vec{x}$. For this system Diachin and Herzog have developed an analytic solution to the system of ordinary differential equations on linear tetrahedra [3]. This approach has the efficient implementation benefits of the modified fourth-order Runge-Kutte method described in [6] but is an exact solution using linear interpolation between spatial data points. Particle dynamics on hexahedral meshes are modeled by using the same techniques after decomposing the hexahedra into five tetrahedral elements. Thus, the method is designed to be independent of the particular volume discretization used. Initial tests have been done on structured, Cartesian meshes, tetrahedral meshes, and body-fitted hexahedral meshes. Results show that this method eliminates one source of numerical error from the integration of Equations (1) and (2) and is more stable than standard numerical integration techniques [3] . It was further shown that for linear velocity fields this method is faster than forward Euler using a constant time step and is more accurate than fourth-order Runge-Kutte.

The massed particle model uses Equation (1) and includes dependencies in the formula for $\vec{V}_p$ for the forces on the particle resulting from fluid resistance and gravity. The system of equations governing massed particles is given by Equation (1) and

$$\frac{d\vec{V}_p}{dt} = \frac{18\mu_g(\vec{V}_g - \vec{V}_p)}{\rho_p D^2} + \frac{\rho_p - \rho_g}{\rho_p}\vec{g}. \tag{3}$$

Here, μ_g is the viscosity of the fluid, ρ is the density, D is the particle diameter, and $\vec{g}$ is the gravitational acceleration vector. To include the effect of evaporation in the model, we start with Equations (1) and (3). To efficiently account for the processes of heat and mass transfer, we make some simplifying assumptions. We assume that the evaporation is heat transfer limited and that the droplet heating time is short compared with the droplet evaporation time. Thus, the temperature of the particle rises to near its boiling point and then begins to evaporate. This process described by the equation

$$\frac{dT_p}{dt} = \frac{N_{Nu}\pi k D(T_g - T_p)}{(m_p c_p^p)}, \tag{4}$$

where N_{Nu} is the Nusselt number, k is the thermal conductivity of the fluid, T_p and T_g are respectively the temperature of the particle and the gas, and c_p^p is the specific heat of the particle. Once a particle reaches its boiling temperature, all further heat gains from the fluid cause mass loss from evaporation without further changes in temperature. The evaporation is described by the equation

$$\frac{dm_p}{dt} = \frac{(\mathrm{Nu}\pi k D(T_g - T_p))}{H_v}, \tag{5}$$

where H_v is the heat of vaporization of the particle.

The TrackPack software comprises the analytic solutions used with streamlines and a fourth order Runge-Kutte method used to integrate the differential equations for massed and evaporating particles. We require that the user provide a query interface to the discrete data structures describing the velocity and scalar fields that returns cell location and interpolated values of discrete data.

The velocity and scalar fields used in the applications described in Section 1 were generated by using commercially available CFD packages. The query interface routines written for these applications use an interpolation algorithm based on using a weighted sum of flow data from nearby discrete points (as is done in the the finite element method). This procedure enables the interpolated velocity fields to vary continuously between volume cells, thereby ensuring that the trajectories are a smooth, accurate representation of the discrete data. The time-dependent data in the automotive problem required an additional interpolation between moving meshes. The code is written in ANSI C and C++ using a portable makefile system. We have successfully installed and run this code on a variety of computer architectures, including the IBM SP system, SGIs, and Sun workstations.

3 THE INTERACTIVE VIRTUAL ENVIRONMENT

To visualize and interact with the results of the particle dynamic calculations, we have developed a graphics package for the CAVE composed of several general-purpose toolkits. In this section we describe the visualization toolkit for interactively placing particulate matter in the flow field and a three-dimensional graphical user interface used to increase application flexibility and control. We have discussed this work in some detail in [2], and we give only a brief description here.

The visualization environment is based on the CAVE technology developed at the Electronics Visualization Laboratory at the University of Illinois at Chicago. Users are immersed in the virtual environment by stepping into a ten-foot cube that has stereo images projected onto two walls and the floor. Several users may be immersed simultaneously in the same virtual environment and interact with the same computational model. One user is tracked by an electromagnetic tracking system, and the image orientation is calculated with respect to the head position of that user. Objects in the CAVE are manipulated by the user, who uses a wand, a three-dimensional analogue of the mouse on current computer workstations.

3.1 Interactive Particulate Visualization

We have provided several options for data visualization that can be used to obtain insight into large numerical vector and scalar data fields resulting from computational simulations of physical phenomen. Of primary interest to the engineers studying the CFD applications described in Section 1 are the velocity fields and the temperature distributions, and we use these as representative vector and scalar fields. The user provides a geometry for the computational domain that serves as a frame of reference for the data visualization.

To facilitate the study of the vector fields that result from computational applications, we have developed an interactive system that allows the user to initiate a streamline from any position within the computational domain. The starting point of the streamline is given by the location of the wand at the time of initiation. This spatial coordinate is communicated to the TrackPack process, which computes the path of a massless particle. Once calculated, the entire path of the particle through the geometry is returned to the visualization process for either continuous or animated display. Each streamline requires approximately 0.2 seconds to request, compute, and visualize when the visualization process and computational process are linked by a local ethernet or fast ATM (asynchronous transfer mode) connection [2].

Animated streamlines can also be used to demonstrate the large-scale structure of the flow fields in the geometry. Rather than choosing a single starting point, we interactively define a rectangular region of interest in the computational domain. This region is filled with a uniform, three-dimensional array of grid points whose density is controlled by the user. The particle streamlines are computed from this initial array of points, and the resulting animations are displayed simultaneously. The time-dependent particle trajectories showing the circular flow in a automotive piston chamber are shown in Figure 1 for two different time steps.

The TrackPack software can also be used to compute the paths for massed and evaporating particulate matter for real-time visualization. This capability is particularly useful in the Nalco Fuel Tech collaboration in analyzing injector placement for pollution control systems. In this application, an initial injector placement is determined by using the knowledge of temperature distribution and flow field data from the computational model and the prior experience of an engineer who has worked with similar boilers. The spray coverage for this initial system is evaluated, and the injector configuration is iteratively refined until the maximal coverage has been achieved.

Once an initial configuration is selected, the position and orientation of each injector are communicated to the remote particle tracking process. The sprays are calculated by using a statistical model consisting of 500 massed evaporating particles for each injector [2]. The results for each injector are requested interactively and communicated back to the visualization process for display using continuous paths, which can be colored by source or by a user-defined scalar quantity. Injectors can be relocated by selecting them with the wand and dragging them to the new location. In addition, the specific spray configuration for each injector can be modified to study the effects of changes in the initial particle size, speed, and distribution. In Figure 2 we show the spray from an injector placed on an exterior boiler wall colored by temperature.

3.2 The Graphical User Interface

To assist in the management of information in the virtual environment, we have designed a three-dimensional graphical user interface (GUI) that gives the user greater flexibility and control of the application. To allow a number of different operating modes to exist in an application. we use a menuing system to enhance the functionality of the wand. For example, in the applications studied here, we use a data visualization mode, an injector placement mode, and a features mode. The last allows the user to toggle the visualization of complex and computationally expensive features in the geometry, such as superheaters and economizers in boilers, change navigation schemes, and/or toggle wire frame mode. In addition to the menuing interface, we have designed three-dimensional scrollbars that allow the user to modify numerical parameters or adjust visualization environment parameters. In Figure 3 we show the scrollbars used to adjust the playback speed and direction of the time-dependent flows in the automotive piston. In the boiler application, we use scrollbars to adjust initial particle size, speed, and distribution from the injector source when studying particulate matter.

4 SUMMARY

In this paper we have described an interactive system that industrial engineers can routinely use in the modeling and analysis of particle dynamics in computational models. Portions of this system have already been incorporated into the Nalco Fuel Tech production environment and have significantly reduced the time required to analyze injector configurations for pollution control systems.

Several enhancements are being incorporated into the current software environment to improve its usefulness and increase the accuracy of the computational model. We are generalizing the libraries for vector and scalar field visualization to allow them to be easily incorporated into other applications. In addition, we are building templates for typical CAVE application interactions, which should greatly reduce the start-up time required by each application. To increase the capabilities of the computational model, we are incorporating unstructured mesh techniques [5] and parallel processing to allow real-time interaction with more complex models. Advanced computational techniques such as boot-strapping the solution from a coarse grid to a fine grid and adaptive refinement methods are also being investigated.

ACKNOWLEDGEMENTS

This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Computational and Technology Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

REFERENCES

[1] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *ACM SIGGRAPH 93 Proceedings*, pages 135–142. ACM, 1993.

[2] Darin Diachin, Lori Freitag, Daniel Heath, Jim Herzog, Bill Michels, and Paul Plassmann. Remote engineering tools for the design of pollution control systems for commercial boilers. To appear *International Journal of Supercomputing Applications*, 10.2, 1996.

[3] Darin P. Diachin and James A. Herzog. Analytic streamline calculations for linear tetrahedra. Submitted to *13th AIAA Computational Fluid Dynamics Conference*, 1996.

[4] T. L. Disz, M. E. Papka, M. Pellegrino, and R. L. Stevens. Sharing visualization experiences among remote virtual environments. In *Proceedings of the International Workshop*

on High Peformance Computing for Computer Graphics and Visualization, pages 135–142. Springer-Verlag, 1995.

[5] Mark T. Jones and Paul E. Plassmann. Computational results for parallel unstructured mesh computations. *Computing Systems in Engineering*, 5(4-6):297–309, 1994.

[6] S.K. Ueng, K. Sikorski, and Kwan-Lui Ma. Fast algorithms for visualizing fluid motion in steady flow on unstructured grids. In *Proceedings of the Visualization '95 Conference*, pages 313–319. IEEE Computer Society Press, 1995.

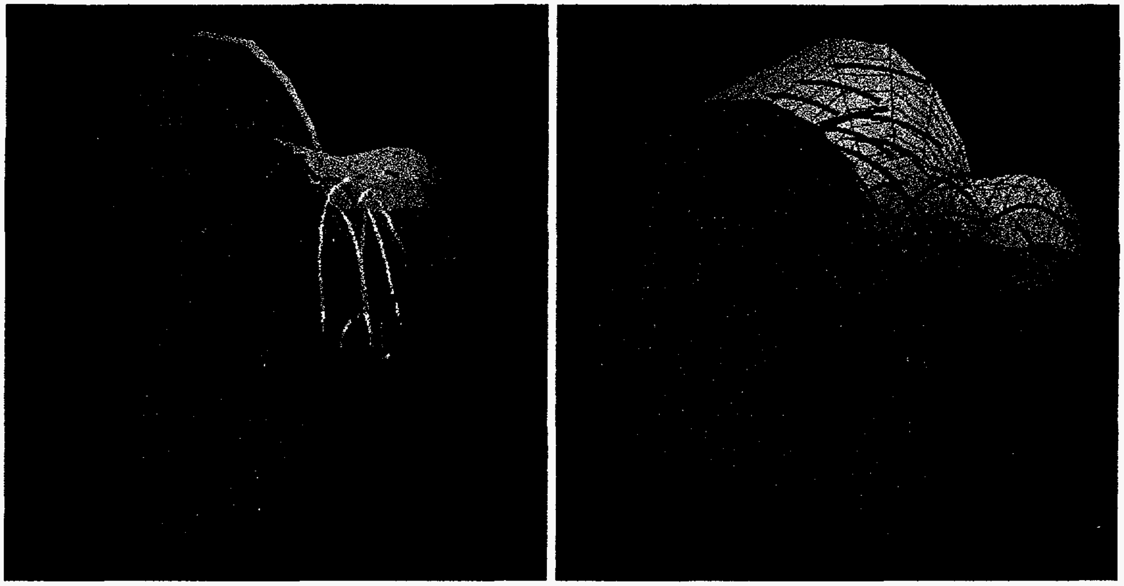

Figure 1: The flow field illustrated through the use of animated streamlines

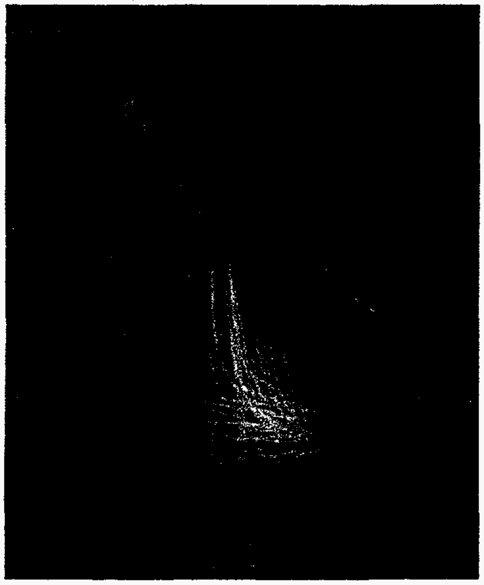

Figure 2: An injector placed on an exterior wall showing an evaporating particle flow

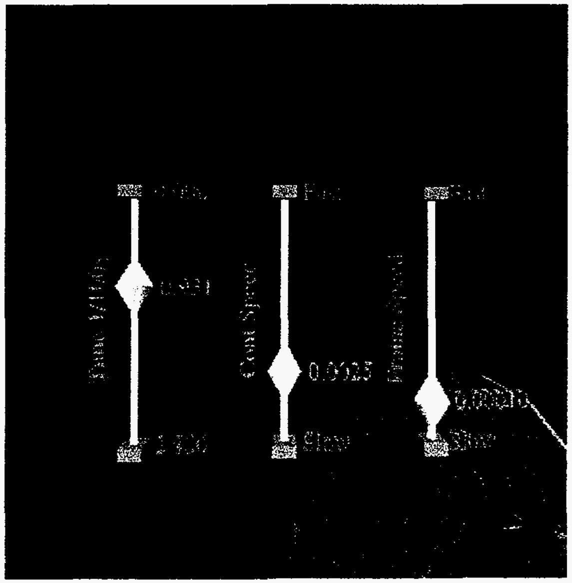

Figure 3: The use of scrollbars and a VCR style interface to control playback speed and direction of time-dependent flow visualizations