Probabilistic Implicit Tracking Approach in Parallel Simulations of Particle Transport

Andrei Smirnov, Steven Rowan, Gouxiang Liu, and James McCormick
West Virginia University
Morgantown, WV

February 27, 2006

Abstract

A method is proposed which enables one to perform parallel simulations of particle transport in a completely decoupled (*embarrassingly parallel*) manner. In the first step the method uses the conventional domain decomposition to split the whole simulation domain into smaller sub-domains, and to assign the simulation of each subdomain to a single processing node. However, the simulations proceed completely decoupled and generate the sets of domain transfer probabilities, which can be used to analyze different aerosol dispersion scenarios. The method is appropriate for building express risk analysis systems.

Keywords: Parallel simulations, Stochastic modeling, Domain decomposition, Risk analysis, Aerosols modeling, Urban environments, Monte Carlo methods

1 Background

One of the major obstacles in the deployment of CFD solvers on distributed computing systems, such as workstation clusters or grid computing environments, is the tightly coupled nature of the CFD discretization schemes, based on continuum approximations. This is especially evident for multi-phase fluid - particle systems, including particle laden flows, aerosol transport, etc. These systems present even more formidable parallelization problems than pure fluid dynamics systems. In particular, when discrete solvers are used for the particulate phase, such as Lagrangian particle dynamics (LPD) solvers, one may encounter serious load balancing issues, related to non-uniform particle distributions inside the domain. Also, particle transport across the sub-domain boundaries may contribute significantly to the communication overhead.

However, in some cases this situation can be avoided, and a loosely coupled system can be constructed, thus enabling an efficient multi-processor implementation. In this study we propose a method, which enables a completely decoupled domain decomposition strategy based on the idea of *domain transfer probabilities* (DTP) and *probabilistic implicit tracking* (PIT) algorithm. The method is illustrated on the case of aerosol dispersion modeling in an urban environment. The method is effective when a large number of various aerosol release scenarios needs to be analyzed in a timely manner, and can form a basis for an express risk analysis system of aerosol dispersion and tracking. A prototype of such system was developed and demonstrated.

2 Method

The main idea of the method is to replace one complex tightly coupled multi-physics simulator of particle transport in a large domain with a sequence of two simpler simulators: (1) a completely decoupled (*embarrassingly parallel*) multi-physics simulator run for each sub-domain separately and independently on other sub-domains and (2) stochastic (Monte-Carlo type) simulator applied to all sub-domains using the input from the previous decoupled simulations.

Essentially the approach is based on earlier ideas of replacing complex physical modeling with a query to the database of scenarios created in prior CFD simulations [1]. However, these ideas proved to be difficult to implement, because of the enormous number of scenarios, which needs to be stored in order to cover a representative range of all possibilities. The improvement offered by the current method consists in replacing the database of scenarios with the database of domain transfer probabilities (DTPs), and replacing the query procedure with a simplified stochastic simulator. This method essentially reduces the size of the data that needs to be stored, since the set of DTPs incorporates many scenarios. Given this data set, one can emulate the results of a current scenario using stochastic simulations.

An important step in constructing the DTP set is to represent the whole simulation space as a collection of well identified objects. In the case of an urban environment this representation comes naturally, with objects representing buildings, bridges, etc. In this case a DTP is the probability for a particle to be transfered from a point A at the boundary of a domain to an object inside the domain, or to a point B at some other boundary of the neighboring domain.

Once the DTP sets are constructed for each subdomain, particle depositions on objects arising from an initial source (S) can be reproduced by stochastically generating particles and letting them transition, following the assigned DTPs. Figure 1 illustrates this concept, which we call *probabilistic implicit tracking* (PIT).

Thus, in the first stage of physical modeling the DTP data is assembled into two sets: the external transition probabilities, which represent boundary-to-boundary transition events, and internal transition probabilities for the events of particle fallout on the objects inside each domain. It should be noted, that representing each domain by objects is essential in reducing the overall size of the DTP data. This is because this representation uses point-to-object rather than point-to-point transfer probabilities, and the number of objects in the domain is smaller than the number of all possible discrete locations.

In this study we used an open source CFD solver OpenFOAM (openfoam.org) to compute the fluid phase and a simplified aerosol transport model based on the equation of particle motion, expressed in terms of particle velocity, $\mathbf{v}(\mathbf{x}, t)$, in a given mean flow field, $\mathbf{u}(\mathbf{x}, t)$:

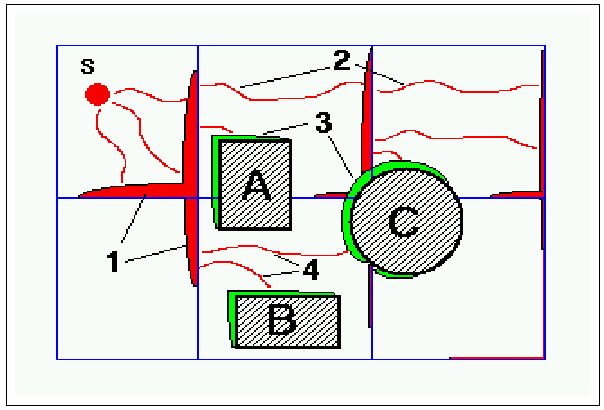

Figure 1: Implicit particle tracking using DTPs: S - initial particle source, 1 - boundary particle sources, 2 - boundary-to-boundary transfer probabilities, 3 - object sinks, 4 - boundary-to-object transfer probabilities,

$$\frac{d\mathbf{v}}{dt} = C_D(\mathbf{u} - \mathbf{v}) + C_T\mathbf{u}' - \mathbf{g} \qquad (1)$$

where C_D is the drag coefficient, C_T the turbulent diffusivity, $\mathbf{u}'$ is the instantaneous turbulent fluctuation vector, and $\mathbf{g}$ is the gravity acceleration vector [2]. The position of the particle at each time step is computed using second order Runge-Kutta time-stepping scheme. The effects of turbulent dispersion on the particles encapsulated in the second term of (1) which was modeled using the RFG technique [3].

In the physical CFD/LPD solver each aerosol particle is convected in a velocity field $\mathbf{u}$, and is traced inside the computational domain until it crosses the domain boundary or hits an object inside the domain. In the second event the hit count for that object is incremented. The final DTP is obtained by dividing all the hit counts by the number of particles released.

The statistical error of predictions will decrease with the number of particles, n, as: $\approx 1/\sqrt{n}$.

3 Results

To validate the method a generic city landscape was set up and prototyped after the Pittsburgh downtown area

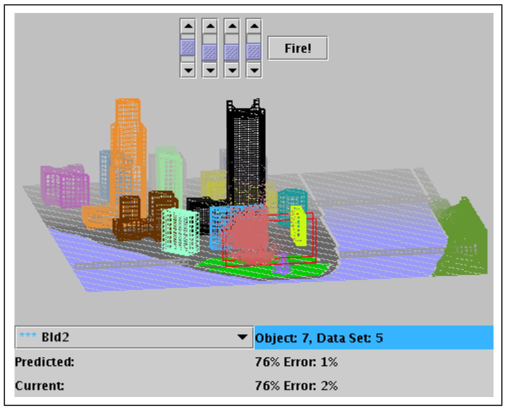

Figure 2: Web interface to simulate aerosol release in a city

http://mulphys.com/sim/demo

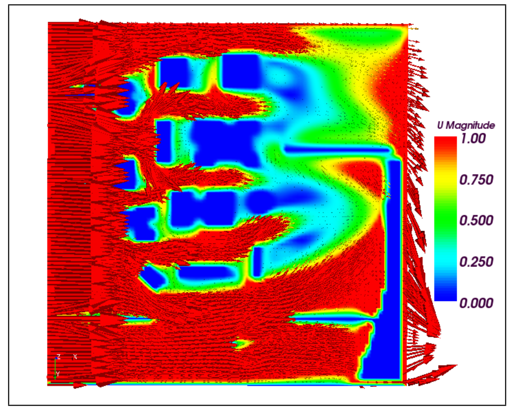

Figure 3: Velocity field

(Fig.2), using the voxel-based 3D graphics system [4, 5]. The whole domain was discretized on the $N_x = 92$, $N_y = 92$, $N_z = 32$ grid and populated with characteristic features like rivers, hills, bridges, park area, pavements and buildings.

The parallel simulator of aerosol transport and dispersion was implemented in C++ language. In the simulations the whole scene was sub-divided into 16 domains and the runs were conducted on a computer cluster with 4GB, 2GHz computing nodes (teragrid.org). One subdomain was assigned per each node. Figure 3 shows the typical velocity distribution in a horizontal cross-section.

The processor time required followed a near linear dependence to the number of particles, and for the 10^5 processor run the average time for executing the DTP calculations on a single node was $(3769 \pm 460)s$. The execution time of the stochastic algorithm on the laptop took on the average 27s, which did not significantly depend on the number of particles. This is because size of DTP set is only slightly affected by the number of particles primarily due to the increase in the number of non-zero particle hits.

To validate the method, separate simulations were performed in a conventional (non-parallel) manner, where the physical CFD/LPD solver was applied to the whole computational space without using domain decomposition. Figure 4 shows a typical comparison histogram of particle distributions between LPD and PIT methods. The

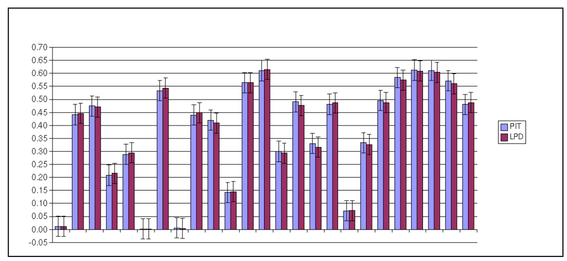

Figure 4: Comparison of aerosol deposition data computed with LPD and PIT methods

data collected for different scenarios show a very good agreement between the two methods with the average deviation of the results typically within 5%.

To demonstrate the viability of this method for express risk analysis a prototype web interface was developed, which enabled us to test different scenarios of aerosol release and dispersion (mulphys.com/sim/demo). The interface is written in Java language and provides a 3D representation of a city landscape with the possibility of navigating through the landscape, arbitrary positioning of the aerosol source, and setting wind direction (Fig.2). The applet also performs a real time simulation of aerosol propagation and dispersion for a limited number of particles as well as a web-retrieval the particle deposition data from a remote database.

4 Conclusions

The method of this study is based on information retrieval from the compressed data sets obtained in prior exhaustive simulations of different aerosol release scenarios.

Using the idea of domain transition probabilities (DTP) and implicit probabilistic tracking (PIT) it was possible to replace a complex physical simulator with a simpler and more flexible stochastic simulator for the purpose of express analysis of simulation data. The physical simulator is still needed to produce the DTP data sets, but it can be implemented in a completely decoupled (*embarrassingly parallel*) manner, since no inter-processor communication is required to produce the DTP data sets. The algorithm can then be efficiently executed in multi-processor and distributed computing environments,

Thus, the method serves dual purpose of (1) providing an *embarrassingly parallel* implementation for certain classes of transport problems on grid computing environments and (2) facilitating express risk analysis of multiple scenarios of a complex physical event. A particularly relevant problem is the express analysis of possible aerosol contamination in urban environments. The results show that this method provides a viable and efficient tool for fast analysis of different contamination scenarios.

A significant saving of retrieval time and data space was achieved by querying objects rather than particular space locations for fallout data. If a more differentiated approach is needed, this approximation can easily be refined by splitting large objects into smaller ones, like buildings can be split into floors, etc.

Acknowledgments

The authors would like to acknowledge West Virginia University Research Corporation for sponsoring this work.

References

[1] J. Thilmen. Harm's way. *Mechanical Engineering*, 127:22–24, 2005. bioterror CFD.

[2] C. Crowe, M. Sommerfeld, and Y. Tsuji. *Multiphase Flows with Droplets and Particles*. CRC Press, 1998.

[3] A. Smirnov, S. Shi, and I. Celik. Random flow generation technique for large eddy simulations and particle-dynamics modeling. *Trans. ASME. Journal of Fluids Engineering*, 123:359–371, 2001.

[4] A.V. Smirnov, H. Zhang, and B. Sowers. Voxel-based volume graphics system for multi-physics modeling. In *The 8th World Multi-Conference on Systemics, Cybernetics and Informatics*, volume 5 of *Computer Science and Engineering*, pages 144–149. International Institute of Informatics and Systemics, 2004.

[5] A.V. Smirnov, H. Zhang, A. Burt, and I. Celik. Fuel-cell simulator interface. *Journal of Power Sources*, 138/1-2:187–193, 2004.