

Visualization Of Airflow Through The Human Respiratory System: The Sniff

Fernando Cucchietti*, Guillermo Marín†, Hadrien Calmet‡, Guillaume Houzeaux§

Barcelona Supercomputing Center

Department of Computer Applications on Science and Engineering
C/Gran Capita 2-4, Edifici Nexus I, Barcelona 08034 (Spain)

Mariano Vazquez¶

Barcelona Supercomputing Center

Department of Computer Applications on Science and Engineering
C/Gran Capita 2-4, Edifici Nexus I, Barcelona 08034 (Spain)

IIIA - CSIC, Bellaterra (Spain)

ABSTRACT

We created a visualization of a CFD simulation and transport of Lagrangian particles in the airflow through an extensive and realistic upper human airways during a rapid and short inhalation (a sniff). The simulation was performed using HPC resources with the Alya system. The output data was stored in a distributed database connected with an in-house plugin to Paraview for post-processing analysis, and then converted into proprietary Maya 2014 particle cache. The rendering was performed using HPC resources with Maya, using a multihued color scale to represent the velocity and a ghost trail to indicate direction.

1 INTRODUCTION

High performance computer simulations can now routinely reach up to tens or even hundreds of GB of data per single time step. The analysis and visualisation of such large data poses many challenges: from the practical details (*e.g.* visualisation must be done on-site as the data is too large to move to or analyse in local computers), to high level cognitive questions (*e.g.* how much and what part of the data is enough to reach meaningful conclusions). For real time analysis, visualisation is usually done with general software like Paraview, whose programmability allow users to expand its capabilities to deal with large or remote data sets. However, interactive visualisation and analysis software are typically constrained in the control of camera and rendering options. To increase this control, our approach is to use dedicated imaging software such as Maya[7] or Blender[5], for which we needed to convert scientific proprietary data formats into files that these programs can read. Because large data sets tend to produce large and complex scenes, HPC resources are needed not only for data conversion but also for rendering.

2 THE PHYSICAL PROBLEM AND SIMULATION

The flow in the human large airways is rich with complexity and variety, therefore, the geometry is extraordinarily complex with constriction zones and rapid changes in direction. According to the complex flow and geometry, a wide range of Reynolds numbers (Re) occurs in the human large airways, producing a large variety of flow patterns, potentially affecting the physiology in unexpected ways.

Due to the complexity of the flow, there exists a large variability of ways to breathe. Furthermore, during the inspiration, there are several ways to inhale. One of the most violent and high-rate-airflow way is the sniff, a rapid and short inhalation (peak 1 l s^{-1}). With the high amplitude of the inhalation and the rapid acceleration occurring during the sniff, the flow is transitional and turbulent, therefore it is the most difficult configuration to study. The inhalation flow profile used for this study was defined using a 10th order polynomial fit to experimentally measured data from a previous *in-vivo* investigation [10].

In our simulations all of the respiratory system is considered, namely from the nasal passages of nostril to the end of the seventh branch generation of the bronchopulmonary tree. The simulation is carried out with a subject-specific derived mesh up to the third branch generation obtained from a contrast-enhanced computed tomography (CT) scan of a 48-year-old male generated by D. Doorly and co-workers in Imperial College London, UK. Alternatively, we also perform simulation replacing the bronchopulmonary tree down from the trachea with an idealised lung geometry consisting of a seven-generation, nonplanar, bronchial tube model [11]. We used an unstructured mesh with a finely resolved boundary layer, see figure 1, and the Navier-Stokes equations are solved using a variational multiscale method (VMS).

A topic of high medical interest is the simulation of Lagrangian particles (transported by the fluid) that can represent an atomised drug coming from an inhaler, or pollutants in the air. Questions of interest posed by our collaborators are:

- What are the regions of most (and least) speed inside the upper airways? Where are the turbulences or vortices that cause recirculation and increase the chance of particles being adsorbed by the boundaries?
- How are particles mixed in a certain region? There are different particles types and sizes, and we want to know how they are mixed in a certain region. We also need a visual representation and a concentration percentage information.
- Where do particles in certain region come from? Smell is a directional sense, and we have a delimited section inside our noses with sensors capable of smelling. The question aims to answer where do the particles that we smell come from.

Our visualisation approach aims to answer these questions (see sections below). Other problems can only be studied by user exploration of the data set. Consider that particles concentration and movement depending on sizes become critical when designing inhalers, because each kind of a respiratory disease requires a particular drug delivery strategy to improve its treatment. Therefore:

- What are the paths followed by the particles? We want to view in the 3D environment the path that followed a single

*e-mail: fernando.cucchietti@bsc.es

†e-mail: guillermo.marin@bsc.es

‡e-mail: hadrien.calmet@bsc.es

§e-mail: guillaume.houzeaux@bsc.es

¶e-mail: mariano.vazquez@bsc.es

(selectable) particle from the first to the last simulation time step.

- What are the paths followed by particles that end up in a given region? One example we want to analyse is how the particles arrived to their final positions in the smell region of the nose, and if geometrical defects lead to odd particle paths. With this knowledge we could help improve the performance of drug delivery medical devices.
- How many particles go across a given section? We want to determine the throughput in arbitrary nose regions.
- Show only the particles that have a particular velocity or other physical properties in a certain direction.
- What are the particles deposited in certain nose parts? Of what sizes, origins, and kinds?.

For this second set of questions we developed a Paraview plugin, described in Ref. [4] and summarised in the following section. This tool is interactive and is not featured in the accompanying video, although it is available on request.

The biomedical simulation was performed using Alya [2] on the FERMI supercomputer at CINECA, Italy, on the OCCIGEN supercomputer at CINES, France, and on the Marenostrum supercomputer at BSC, Spain. We used two computational meshes: a medium one of 44 million elements (medium) and another with 350 million elements (fine). See different mesh resolutions and simulation parameters in table 1. In this work we only use flow results from the 44 million mesh, although we focus only in the particle data, which is calculated a posteriori (even in different supercomputers)

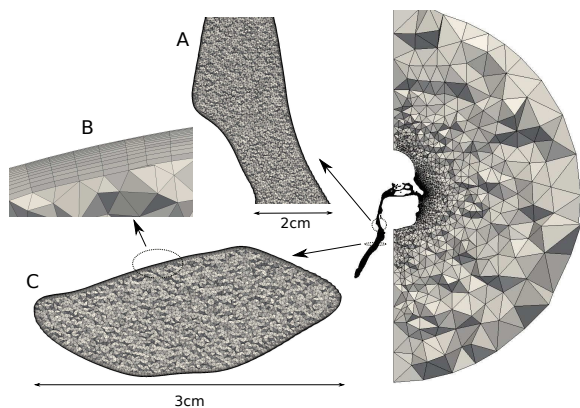


Figure 1: Sagittal view of upper airway medium mesh (44 million elements) and details: (A) section in the sagittal plane above the glottis, (B) detail of prism layers, (C) section in the axial plane in the supra-glottic region.

3 DATA STRATEGY AND INTERACTIVE ANALYSIS

The particular simulation output we worked with consists of a unique CSV file, with each row specifying the properties for one particle: time step, position, velocity, type, family, etc. A typical simulation in the smallest mesh generates a 300 GB file containing 2.8 billion rows. For visualisation purposes we worked with two subsets of the full data, a coarse one with 80 thousand particles and a fine one with 500 thousand particles, whose positions we track for 3125 time steps in the 0 to 0.5 seconds time frame. Apart from the position, velocity, and acceleration, we can track physical properties of the particles like mass, size, and chemical properties. In

Property/Mesh	M1 (coarse)	M2 (medium)	M3 (fine)
$N_N (\times 10^6)$	2	14	110
$N_E (\times 10^6)$	8	44	350
$\Delta T (\mu s)$	100	10	5
$h (mm)$	0.5	0.3	0.15
$h_{bl1} (\mu m)$	50	6	3
N_{bl}	5	13	26
R_{bl}	1.2	1.2	1.2

Table 1: Summary of different mesh resolutions and simulation parameters with N_N : number of nodes, N_E : number of elements, ΔT : time step, h : mean edge length of elements, h_{bl1} : height of the first element in the prism layer, N_{bl} : number of prism layers and R_{bl} : the prism growth ratio

figure 2 we show a 3D visualisation of our dataset obtained with Paraview [9]. In the attached vide we also show a 3D deposition map of the particles, which is computed at runtime since the particles are dropped from the simulation when they are absorbed by the walls.

The visualisation of a complete simulation from the largest mesh is impractical even in workstation sized computers. Storing the data remotely can only be done using large servers, but then the workflow needs to be performed in batches. To address this need we deployed the data in a cluster using a distributed database infrastructure (Hive and Hadoop), and developed a middle-ware plugin that connects the database directly to the graphical interface of Paraview. Here we include a short summary of the system, while a full description can be found in Ref. ([4]). Our system allows the users to prepare and launch complex queries inside the system they already know, and to receive the answers to their queries directly in Paraview’s native format VTK. The database is optimised to perform distributed queries from relevant variables, enabling the users to perform real time complex analysis with a speed independent of the data size (as long as the database resources can be scaled accordingly).

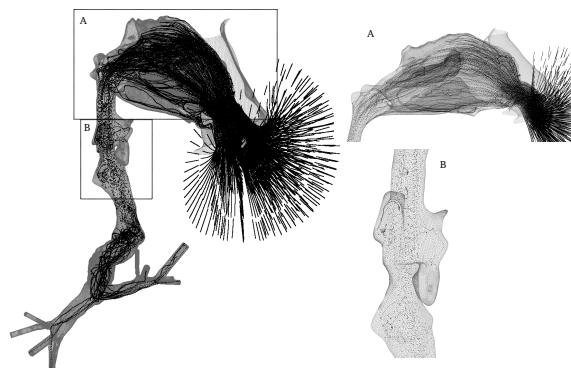


Figure 2: Example of a few selected trajectories from the simulation results visualized with Paraview. A and B are close up regions. The geometrical mesh has the nose on top, and the upper bronchia at the bottom (the computational mesh continues with fractal subdivision of the bronchi up the seventh order)

4 VISUALIZATION STRATEGY AND APPROACH

The scientific visualization team is composed of physicists, computer scientists, and graphic artists. The in-house aspect of the team, and its multi-disciplinary nature, are crucial to obtain scientifically correct and relevant visualizations.

Apart from traditional plotting tools like Paraview and Visit,

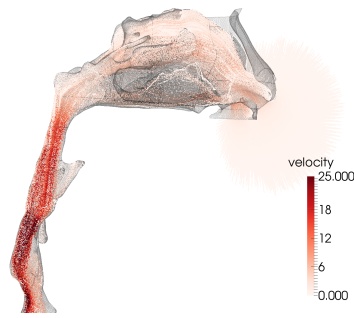


Figure 3: Close up of the nose region of the mesh, with particles coloured by velocity magnitude.

we use software commonly found in the audiovisual industry: Maya[7], Arnold Renderer[3], Blender[5], Final Cut Studio[6] and the Adobe Creative Suite[1]. We process data and render the visualizations in parallel in the Marenostrom 1.1 petaflop supercomputer.

In order to prepare the data for visualization, and to join the inputs and outputs of the different software in the workflow, we developed a suite of (mostly Python) scripts. Typically these scripts are used to convert formats or to manipulate the animation scenes to work directly with the data coming from the numerical simulation.

4.1 Data process

For the results shown in the accompanying video, data was first analysed and processed using Paraview. Then, the respiratory system surface mesh was exported and transformed into a standard 3D file format. The original data with particle trajectories was transformed into a binary Maya particle nCache file [8], which consists of an XML descriptor file and a set of files, one per frame, with the particles position, velocity, acceleration, and additional properties. The physical properties of the particles can be mapped into standard Maya properties and later used to implement color, size, transparency, or brightness scales. At each time step we randomly choose ten percent of the newly created particles, and discard the others. This helps keep the animation manageable and the visualization less crowded without significant compromise in the information representation. The deposition map uses all the particles. Particle data for each time step was independent from previous steps, thus we were able to process the data in parallel without communication burden. Deposition information was extracted into a separate cache, time synched with the particle data.

4.2 Visual encoding

The complete duration of the sniff simulation was 3125 time steps. For the sake of rhythm, it was accelerated by a factor two to last 48 seconds at 25 fps. We mapped the velocity of each particle to a linear hue change from violet to light blue to red. In order to improve readability, we also mapped velocity magnitude to the incandescence of the particles, making faster particles glow brighter. While the movement of particles already indicates direction, a single point can lead to confusion. A more elegant solution that improves readability of direction information is a vanishing trail added to the particles.

The particles are initially placed as a circular bunch right outside the nose waiting to be sniffed in by the simulation. They were removed from the visualisation for clarity using a short opacity ramp from zero to very small velocity.

Particles deposited against a wall disappear from the main cache and appear in the secondary one. Deposition density is visually encoded in a single hue linear scale from transparent to yellow, creating an adequate color contrast with the velocity color scale. The

deposition event is marked by a short flash of glow intensity.

For spatial reference, on top of the computational mesh we added a mesh of the back of the head and the rest of the body, although this was not used in the simulations. All meshes were rendered in light and transparent mode, producing a silhouette that gives reference and does not to obstruct the view of the particles.

Camera animation focuses on clarity and readability, displaying areas where recirculation and turbulence is clearer, or where the particles tend to accelerate more. Collaborating researchers helped us in the selection of the close up regions.

4.3 Production and post-processing

All data processing and rendering was done in parallel in the Marenostrom supercomputer, at BSC, Spain. Shading, lighting and final renderings were done using Maya, using the Mental Ray engine. Post processing and editing was done using Adobe Premiere and After Effects.

5 ANALYSIS

The video accompanying this article shows our visualisation techniques applied to answering the first set of questions mentioned in the introduction: easy location of fast speed regions, turbulences and vortices, and probability of adsorption by the walls.

Through informal interviews, the researchers that work on these simulations remarked that the strong visual appeal of the visualisation techniques shown in the video do not diminished their capability to single out technical details in the data, allowing them to use these visualisations for a broader audience spectrum (i.e. from peers to non-technical viewers). In particular, researchers easily identified the fastest regions (above $10m/s$) in the straight section of the trachea, as well as small variations in the slower regions like the inner nose channels. Vortices were deemed noticeable as clear knots or loops against the overall linear flow of particles, and particle adsorption density was labeled easily distinguishable from the dynamics thanks to its contrasting color scale.

While our techniques might be too time-expensive for interactive data exploration, we believe they are amenable for final-stage data visualisation products such as publications, conference presentations, web sites, and outreach efforts.

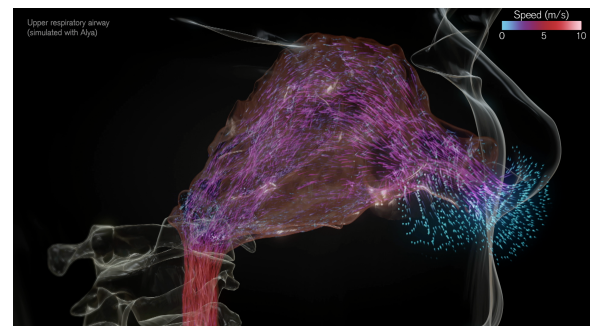


Figure 4: Final rendering of the system with particles and the computational mesh.

ACKNOWLEDGEMENTS

This work was partially supported by the Spanish Severo Ochoa Program, grant SEV-2011-00067. We acknowledge PRACE for awarding us access to the resources FERMI and OCCIGEN .

REFERENCES

- [1] Adobe CS. <http://www.adobe.com/creativecloud.html>. Developed by Adobe.

- [2] Alya system - large scale computational mechanics. <http://www.bsc.es/computer-applications/alya-system>.
- [3] Arnold global illumination renderer. <https://www.solidangle.com/>. Developed by Solid Angle.
- [4] A. Artigues, F. M. Cucchietti, C. T. Montes, D. Vicente, H. Calmet, G. Marin, G. Houzeaux, and M. Vazquez. Scientific big data visualization: a coupled tools approach. *Supercomputing Frontiers and Innovations*, 1(3), 2014.
- [5] Blender. <http://www.blender.org/>. Open source 3D animation software by the Blender Foundation.
- [6] Final Cut Pro X video editor. <https://www.apple.com/final-cut-pro/>. Developed by Apple.
- [7] Maya 3D animation software. <http://www.autodesk.com/products/maya/overview>. Developed by Autodesk.
- [8] <http://bit.ly/1mlOsJi>.
- [9] Paraview software. <http://www.paraview.org>. Paraview is developed by Kitware company. <http://www.kitware.com>.
- [10] C. E. Rennie, K. A. Gouder, D. J. Taylor, N. S. Tolley, R. C. Schroter, and D. J. Doorly. Nasal inspiratory flow: at rest and sniffing. *Int Forum Allergy Rhinol*, 1(2):128–135, 2011.
- [11] B. Soni, N. Arra, S. Aliabadi, W. Luke, and K. Walters. *Mesh Refinement Study of Flow and Particle Deposition in Human Lung Airway Model*. American Institute of Aeronautics and Astronautics, 2015/08/05 2011.