Progress Towards a Medical Image through CFD Analysis Toolkit for Respiratory Function Assessment on a Clinical Time Scale

R. F. Kunz*, D. C. Haworth**, D. P. Porzio*, ***, A. Kriete[¥]
Penn State University, *Applied Research Laboratory, **Department of Mechanical Engineering, ***Department of Bioengineering
[¥]Drexel University, Department of Biomedical Engineering

ABSTRACT

A semi-automated end-to-end medical image through Computational Fluid Dynamics (CFD) analysis toolkit has been developed, with the ultimate goal of providing clinical-time-scale (hours) diagnostic information for respiratory disease and injury assessment. A software system is in place that proceeds from a standard clinical image format, through lobe and upper bronchi segmentation, upper airway "thinning", airway generation partitioning, and truncation, lower bronchiole lobe volume filling, octree-based unstructured mesh generation for the upper airways, quasi-one-dimensional geometric modeling for the lower airways, and CFD analysis of respiration. Each of these components is presented.

Index Terms — respiratory system, fluid dynamics, image segmentation, mesh generation, multi-scale modeling

1. Introduction

Recent advances in medical imaging/processing technology, CFD technology and parallel computing have enabled large-scale physiological modeling and simulation of many human and animal systems. This research focuses on time-dependent simulation of human respiration.

The authors' motivation for respiratory simulation is clinical in nature. Specifically, it is now possible, within a few hours, to process high quality CT scans of the upper respiratory system through a multi-scale cascade of geometric and flow modeling, and produce clinically relevant information, including expected pharmaceutical bio-availability, and oxygen uptake in the context of disease or aging.

This paper summarizes the authors' recent efforts in developing a semi-automated end-to-end clinical image through CFD-based diagnostic toolkit. This research is complemented by other recent work in this area (Refs. [1]-[4]).

Figure 1 shows a schematic of the Python (Ref. [5]) based driver, breathe.py, for the software package. The modular nature of the tool allows the user to supplant components with different modeling software. The paper is organized so as to proceed sequentially through the modeling components currently in place.

2. Medical Imaging and Processing

Input datasets consist of 3D CT scan image data, in our case stored according to sgi TIFF standard, compiled with a DICOM header of the clinical image slices. For image visualization and image processing, the commercially software, Amira (ref [5]), is used.

Within Amira, segmentation of the lung mask from the rest of the CT images is fully automated with an appropriate color map and threshold value. Automatic removal of "islands" (unconnected volumes) improves the final result.

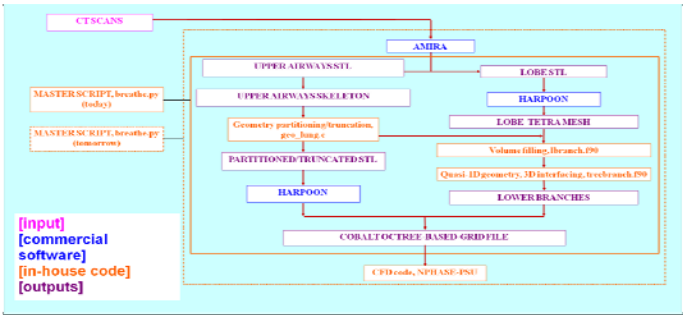


Figure 1. Structure of breathe.py

For segmentation of the five lobes, the lobar fissures separating the lung lobes can be visually discriminated (using another color map.) By following these features/curves within 2-D transverse slices, and by following the anatomic features, lobes are segmented; to date this is performed manually. The lobe surfaces are exported from Amira as standard .stl files after, typically, more required handwork: smoothing, removing islands and filling holes. Elements of the Amira lobe segmentation process are shown in Figure 2.

Region growing based segmentation within Amira is used for automated segmentation of the upper airways. The maximum generation number for automated airway segmentation depends on CT scan quality, and the lung's physical character for a particular subject. The upper airway surfaces are also exported as .stl files after, as above, smoothing and closing the geometry.

The final image processing step is thinning of the upper airways geometry; this step is fully automated using an Amira script file. The output is a text file including node coordinates of the airway "skeleton" and the diameter of each segment. This skeleton is used for the partitioning and truncation geometry processing steps. Figure 3 shows the upper airway segmentation and skeleton for an elderly female subject, hereafter Subject A, obtained from Amira.

3. Upper Airway Partitioning and Truncation

An in-house code geo_lung.c was developed to partition the airway tree by generation. This code is executed by breathe.py subsequent to the Amira processing. The .stl and skeleton files of the upper airways are input, a nearest face approach is applied, followed by an edge-based post-processing to remove "islands" of a computed surface generation number. The robustness of the algorithm is illustrated in Figures 4a and b, where the .stl of a

rubber cast of a cadaver's upper airways (which allowed for CT imaging and segmentation to generation 10/11) is partitioned.

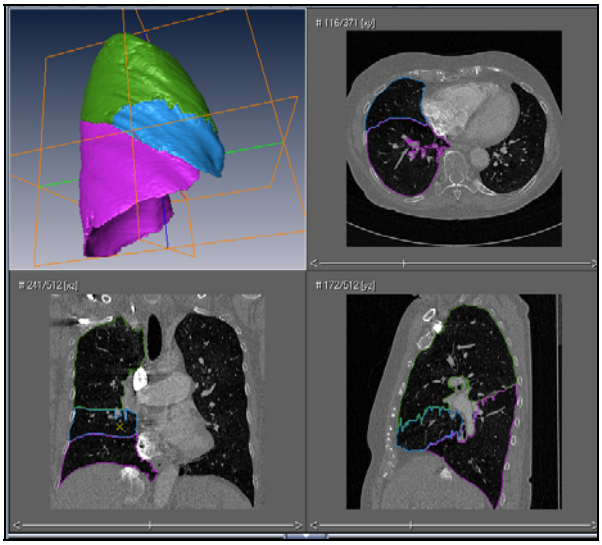


Figure 2. Elements of Amira lobe segmentation process

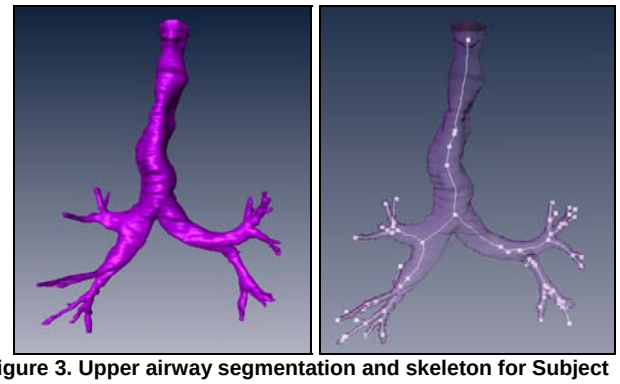


Figure 3. Upper airway segmentation and skeleton for Subject A, obtained from Amira.

The generation number attribute for each surface element is passed to the grid generator so that appropriate mesh sizes can automatically be assigned, thereby allowing the grid generator to be run in batch mode.

The interface between macro and micro-scales is here defined as the generation number where the analysis transitions from fully 3D in the upper airways (where turbulent boundary layers, secondary flows, flow splits, patient specific geometries and local particle deposition patterns can be explicitly resolved) to quasi-1D in the lower airways (where image resolution limits come into play, and where statistical descriptions of the convective regime are necessarily employed by virtue of the large number of bronchioles).

Accordingly, geo_lung.c also automatically truncates the upper bronchi geometry at a user specified location, usually generation 5-7 depending on geometric model quality for the particular lung being analyzed (see Figure 4c). This is where the quasi-1D modeling of the lower airways is interfaced.

4. Mesh Generation

The commercial software HARPOON (Ref. [7]) is used for grid generation of both lobe and airway trees. Grid generation is highly automated (batch execution, interactive views shown here) and very fast: 1 million cells can be generated in approximately one minute on a single PC processor. HARPOON employs an octree based method which produces hexahedral dominant meshes with "hanging nodes" (Figure 5a), that the CFD solver used must support. For the upper airways, local grid resolution is defined by assigning different cell sizes to different generations, an attribute streamed from partitioning attributes assigned above. Prism layers are defined along wall boundaries to support adequate boundary layer resolution. For the low Reynolds numbers encountered in human respiration our experience is that grids with on the order of 10^5-10^6 elements are adequate to capture the patient specific salient physics of the upper respiratory tract.

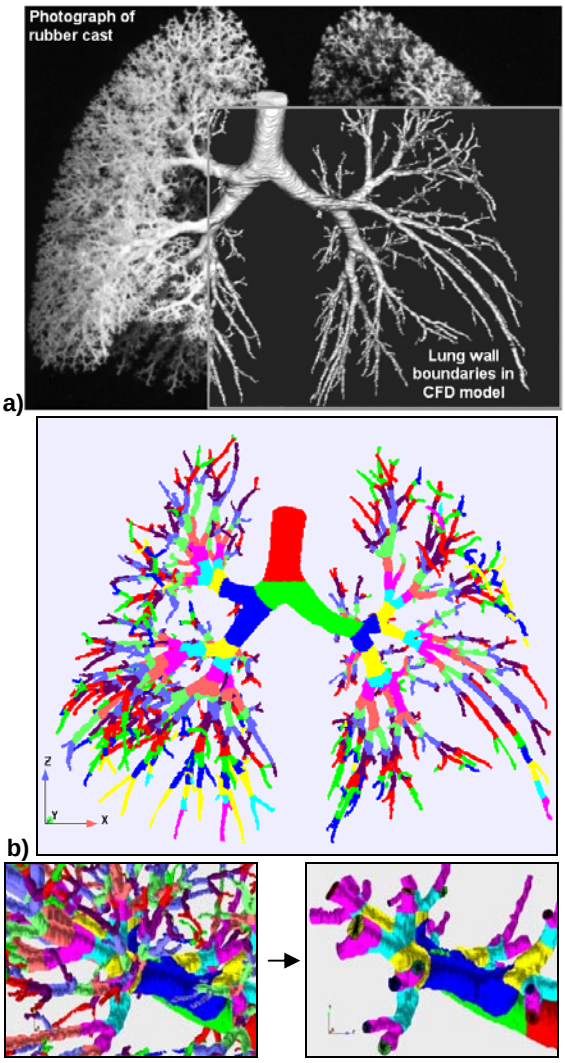


Figure 4. Upper airway partitioning and truncation of a segmented CT scan of the rubber cast of a cadaver's upper airways: a) Photograph of rubber cast overlaid by segmented .stl output, b) Partitioning colored by generation number, c) Upper airway truncation

HARPOON is also used to automatically generate lower quality all-tetrahedral meshes which are used exclusively to define lobe volumes in the volume filling algorithm.

5. Lobe Volume Filling

An in-house code, lbranch.f90 was developed to reconstruct the tracheobronchial tree from each truncated airway down to the respiratory units of each lobe. This software module is called by breathe.py after the volume lobe meshes are generated.

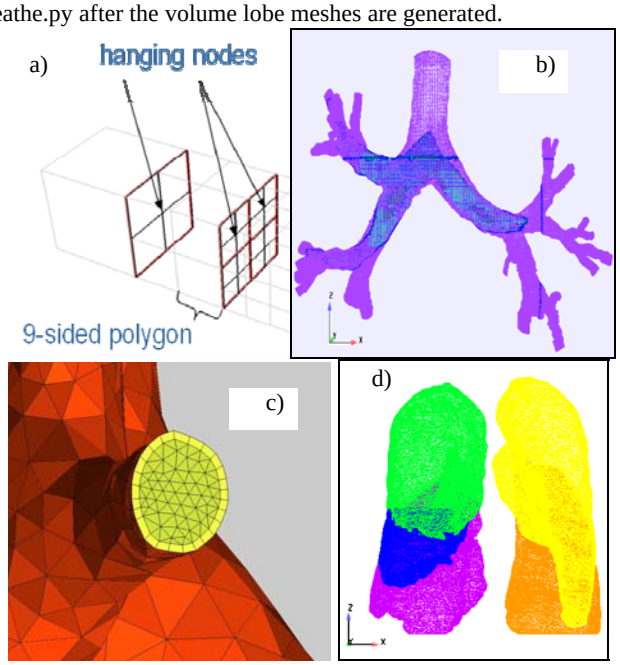


Figure 5. Elements of grid generation for Subject A: a) Hanging nodes arising from octree, b) Cuts through upper airway grid, c) Prism layer, d) Lobe meshes

A volume-filling algorithm is employed (adapted from the work of Tawhai, Ref. [8]) that enforces that the resulting spatial distribution of respiratory units is statistically uniform within each lobe. The branching algorithm reproduces defined geometric statistics (diameters, lengths, branching angles) of a human tracheobronchial tree. A typical tracheobronchial tree represents generations 6-18 of branching, and contains between 70,000 and 100,000 bronchioles, of which approximately half are terminal branches that end in a respiratory unit.

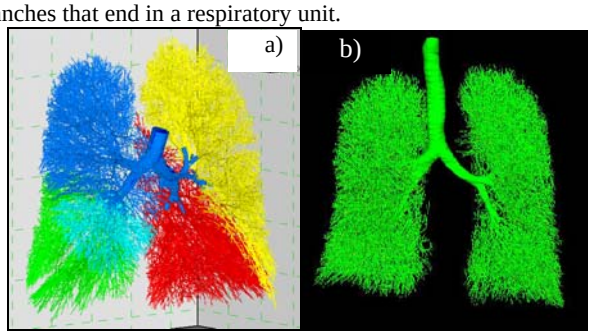


Figure 6. Full trachea-bronchial tree for Subject A. a) Volume filling algorithm results, b) Composite 3D and quasi-1D volume mesh

The algorithm allows the tree to deform to follow transient breathing cycles, with the branching topology necessarily retained. Specifically, for the unsteady breathing simulation, outer lobe surfaces (.stl data) must be available at discrete times during the breathing cycle. (For now, these are generated using prescribed volume/geometry vs. time. This approach conforms to future applications where .stl data will be provided and interpolated directly from time-dependent clinical CT data.)

For the time dependent volume filling, the location of each terminal leaf (a respiratory unit) is prescribed such that its logical position within its particular lobe remains fixed as the lobe deforms. The location of each internal node is then determined by treating the tree as a 6-DOF beam system, and using a finite element code to solve for the displacements, an approach which supports eventual specification of realistic material properties. Figure 6a shows a view of the tree generated by the algorithm for Subject A.

6. Quasi-1D Representation of Lower Airways

As discussed above, a multiscale approach is taken where the upper branches (trachea through generation 5-7) are fully resolved using 3D CFD, and the entire convective regime (generations 5-7 down through 17-19) are modeled with a quasi-1D method. An in-house code, treebranch.f90, interfaces the truncated upper branch model and the lobe-volume filling lower trachea-bronchial tree. This is done by constructing a pipe-and-branch model composed of one cylindrical section for each branch, each in turn defined by approximately 4 axial elements (depending on branch length) for each of the 70,000-100,000 lower bronchi. The treebranch.f90 module is called by breathe.py for each time-step in the breathing cycle after the volume filling algorithm (which provides the length, diameter and position of each of the branches) is executed. Figure 7 shows a close-up of the quasi-1D volumes. The "wall-less" branch elements are bounded only by through-flow faces, and the physical modeling implications of this are discussed in the next section.

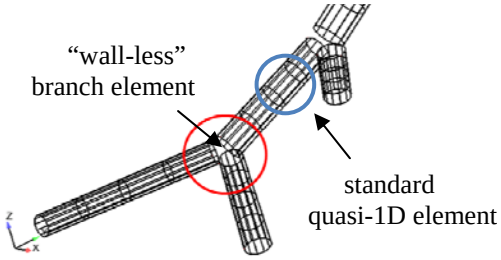


Figure 7. Sketch of quasi-1D pipe and branch topology for lower airways

7. CFD Method

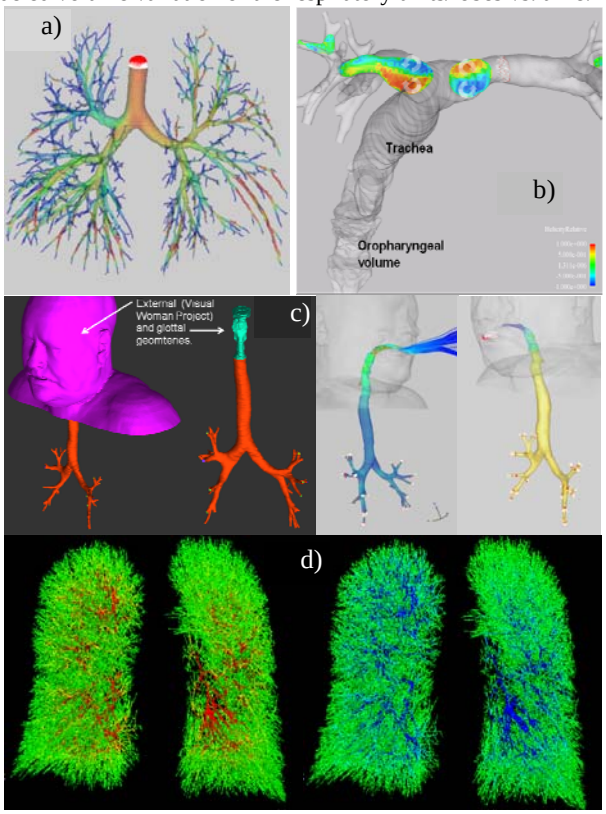
The CFD solver used is a time-accurate finite-volume code. It employs a face-based cell-centered discretization strategy and supports arbitrary polyhedral element types, as required by both the octree-based 3D mesher (hanging nodes) and the n-faced elements that define the branching structure of the lower airways (i.e., nface=nc+2 for the standard quasi-1D element in Figure 7, where nc=#of edges approximating the circular cross-section). The code is parallelized based on domain decomposition/MPI. The code is multiphase capable as discussed in Ref. [4] for respiratory particle transport and deposition simulation.

In this effort, standard high Reynolds number form turbulence modeling is employed for the upper airways (k-ε model), and the lower airway flows are assumed to be laminar. In the quasi-1D regions, body forces are added that model axial skin friction pressure drop, flow turning in the branches, and branch total pressure losses (which must include accommodation that no shear is imparted to the flow in the "wall-less" branch elements.)

Boundary conditions are applied in one of two fashions. A standard pressure drop condition can be applied between the upper-most boundary faces (e.g., the trachea for geometry truncated there), and the leaves of either the upper airways (for 3D upper-lung-only simulations) or the numerous terminal bronchioles in simulations that incorporate the quasi-1D modeling of the lower airways). An oscillatory pressure drop corresponding to nominal breathing cycles and tidal volume are typically applied in these cases. More generally, the entire upper respiratory system is modeled, including the oro-pharyngeal regions and near-subject atmosphere, so a far-field gauge pressure of zero is set. For the quasi-1D terminal bronchioles, a "piston" volume variation boundary condition can be applied to represent the variation of respiratory unit volume within the breathing cycle.

8. Representative Results

Figures 8a-e illustrate a range of calculations performed with the tool. Figure 8a shows a time-step for an unsteady, full 3D pressure forced simulation through the lung configuration show in Figure 4. Shown are contours of near-wall oxygen concentration at near full exhale. Figure 8b shows normalized relative helicity contours at a cross section immediately below the first branch of the upper airways. This was from an unsteady model of the upper airways of Subject A, with a representative oro-pharyngeal and external subject geometry adapted from the Visual Woman dataset (Ref. [9]). Clearly evidenced at this near full inspiration time-step are the strong turbulent secondary flows induced by the branching. Figure 8c shows a view of the geometric components of this model as well as streamlines colored by velocity magnitude and pressure contours at a near full expiration time-step. Figure 8d shows pressure contours at full exhale and inhale time-steps in a quasi-1D simulation of all 74734 lower bronchioles of subject A, including a modeled volume variation of the respiratory units/lobes vs. time.



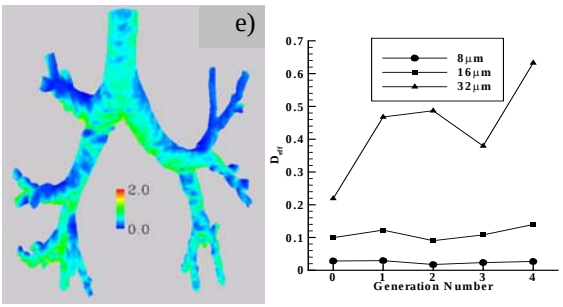


Figure 8. Representative results. Description in section 8.

Figure 8e shows particle deposition results carried out using the standalone CFD tool, reproduced here from Ref. [4], where more details are available. Shown are steady (max inspiration) contours of relative particle concentration (volume fraction/inlet volume fraction) in near wall cell for a 16μm particle distribution. Also shown is a plot of predicted deposition efficiency vs. generation number for a range of particle sizes.

9. Conclusions

This paper has briefly summarized all of the components of a medical image through CFD analysis toolkit developed by the authors. As our research continues we are refining the method in a number of areas including fluid-structure interaction, aging and disease modeling, improved respiratory unit modeling, and improved geometric and flow coupling of the macro/micro scales.

10. Acknowledgements

This work was sponsored by NIH Grant 5-R01-ES014483-02, with technical monitors, Dr. David Balshaw and Dr. Grace Peng.

11. References

[1] Lin, C.L. and E.A. Hoffman, "A numerical study of gas transport in human lung models," *SPIE Medical Imaging: Physiology, Function, and Structure from Medical Images*, 5746:92-100, 2005.

[2] Nazridoust, K. and Asgharian, B., "Unsteady-State Airflow and Particle Deposition in a Three-Generation Human Lung Geometry," *Inhalation Toxicology*, 20:595-610, 2008.

[3] Minard, K.R., Einstein, D.R., Jacob, R.E., Kabilan, S., Kuprat, A.P., Timchalk, C., Trease, L.L., Corley, R.A., "Application of Magnetic Resonance (MR) Imaging for the Development and Validation of Computational Fluid Dynamic (CFD) Models of the Rat Respiratory System." *Inhalation Toxicology* 18(10):787-794, 2006.

[4] Kunz, R.F., Haworth, D.C., Leemhuis, L.S., Davison, A.C., Zidowitz, S., Kriete,A., "Eulerian multiphase CFD analysis of particle transport and deposition in the human lung, " Simulations in Biomedicine V, 121-134, WIT Press, 2004.

[5] http://www.python.org

[6] http://www.amiravis.com

[7] http://www.sharc.co.uk

[8] Tawhai, M.H., Hunter, P.J., Tschirren, J., Reinhardt, J.M., McLennan, G. and Hoffman, E.A., "CT-based geometry analysis and finite element models of the human and ovine bronchial tree." *J. Appl. Physiol*. 97:2310–2321, 2004.

[9] http://www.crd.ge.com/esl/cgsp/projects/vw/#thevisiblewoman