Title

Subtitle

Max Sharnoff
Trinity 2022
candidate number
BA Computer Science

Abstract

Detecting changes in human lung morphology and determining its effects on lung function requires significant time committment per patient, so statistical analysis on many individuals is infeasible. Computational models of the lung are therefore a natural choice for researching the effects of altered lung morphology, with reference to existing lung function tests.

Accurate computational models also allow investigation into properties of the lungs that cannot feasibly be measured; e.g., increased internal stress in one location from damaged airways elsewhere.

This paper builds on recent advancements in modelling airflow in the lungs (⟨reference to Foy⟩) to produce an efficient, accurate model of the lungs that supports simple alterations to the simulated morphology. We then use this model to determine the strains placed on the rest of the lungs by various kinds of constricted or damaged airways.

Contents

1 Introduction

1.1 Motivation

Respiratory diseases account for more than 10% of *all* disability-adjusted life-years lost due to any medical condition, second only to cardiovascular diseases.[1] Because of this, any betterment of our understanding of the lungs and how they change from damage has immediate benefits towards understanding one of the most significant categories of disease.

In spite of this, there are relatively few existing methods for experimentation. Clinical observations on live patients are necessarily limited, and common techniques – spirometry, inert-gas washout, and fMRI imaging – all have severe limitations that render them infeasible or impossible to use for obtaining detailed data on the lungs at scale. And on top of that, the difficulty of drawing inference from these methods is enhanced by the fact that they are purely observational; in this paper, we are concerned with the effects of certain changes in lung morphology (such as: tightening of the airways, stiffness in the expansion and contraction, etc.).

Of course, it would be unethical to *induce* these changes in patients. However, sufficiently-accurate computational models present a natural solution. By designing models that can easily be arbitrarily deformed or otherwise altered, we create the opportunity to efficiently investigate how targeted changes in lung morphology affect both lung functioning as a whole and the stresses placed on individual regions.

Todo: **I'd like to add something along the lines of: "historically, computational models have been too expensive for experimentation without specialized equipment, but recent developments (i.e. Foy) have show other methods for simulating airflow are both efficient and accurate."**

1.2 Contributions to the field

This paper introduces a new tool for simulating and observing changes to the lungs, and their precise effects.

Todo: **Something about: "this is useful to people looking to find new results about *how* the lungs get impacted by various diseases." Would like to also say: "we have investigated the effects of clinically-observed symptoms from a couple diseases, to showcase the utility of this tool"**

2 Background

2.1 Physiology of the lungs

Basic points:

- air comes in through mouth, meets the lungs at the larynx
- lungs start off with the trachea, turns into bronchi/bronchial tubes
- eventually turns into bronchioles, alveoli & capillary network (although we don't care *as* much about these)

Want to talk about typical dimensions of each part here, as well as typical airflow & how that's observed (e.g. "what tests are used?") – talk about intert-gas washout, among others.

2.2 Clinical methods

- fMRI – typically very expensive & fairly low granularity: Good at getting an overview of the lung morphology, but resolution is typically low. (Not 100% sure how accurate this assessment is, but I think I saw something about it.)
- Inert gas washout

The above are good for *observing* changes in a patient, but have a couple shortcomings if we want to analyze the effects of lung morphology. Firstly, observational tests cannot be performed quickly; it would be prohibitively expensive to acquire the data for any kind of large-scale analysis. Secondly, current clinical methods do not allow forced changes to patient lung morphology (even without the risks and ethical considerations).

For those reasons, it is natural to turn to simulations – in particular, computational models – in order to gain insight into impact on physiology and overall function from isolated changes within the lungs.

2.3 Prior computational models

Talk briefly about proper fluid dynamics being too computationally expensive, use that to tie into the model from Foy.

There's definitely other models, still need to look into those.

3 Methods

Broadly speaking, this section comes in two parts; first defining the more theoretical underpinnings of the model used for simulation, followed by detail on the implementation in practice. There is relatively little theory to discuss; much of it comes from prior work.[2]

At a high level, we use a system of simultaneous equations to determine how the state of the simulated lungs evolves, updating in discrete timesteps to give a close approximation to the way a similar physical system would behave. Equation 4 governs the conservation of volume from one timestep to the next, which allows us to obtain greater numerial stability than we might otherwise, e.g., with $\text{volume}_{t+1} = \text{volume}_t + \text{flow}dt$.

Instead of a "full" fluid simulation (e.g., with the Navier-Stokes equations), we use a one-dimensional simulation as shown in **(author?)**, **(year?)**.

3.1 Simultaneous Equations

This section provides a summary and brief description of the four simultaneous equations that govern the state of our system, the first of which is the following:

$$P_{\text{parent}(i)} - P_i = R(i)Q_i \tag{1}$$

This specifies that the pressure differential between the distal end of branch i and its parent must equal the pressure from the resistance from the flow through this branch i. For the "root" branch, $P_{\text{parent}(i)}$ is the pressure at the trachea – typically atmospheric pressure.

The resistance term $R(i)$ is defined as following function, as given by Pedley et al. (1970):

$$R(i) = \frac{2\mu L_i c}{\pi r_i^4} \left(\frac{4\rho |Q_i|}{\mu \pi L_i} \right)^{\frac{1}{2}}$$

The parenthesized term corresponds to the Reynold's number of the flow, scaled by the ratio of the diameter of the branch to its length L_i. r_i is the radius of branch i, μ is the viscosity of the air, and $c = 1.85$ is a correction constant.

The second equation ensures incompressibility; the flow through a bifurcation must equal the sum of the flow through its children:

$$Q_i = \sum Q_{\text{child}} \tag{2}$$

where each *child* refers to any branch c with $\text{parent}(c) = i$.

The third equation maintains that the volume of an acinar region changes with the flow into or out of it for the given timestep:

$$V_i^t = V_i^{t-1} + dtQ_i^t \tag{3}$$

where dt is the timestep size, t refers to the current timestep, and V_i is the volume of the acinar region of branch i.

The final equation defines the elastic force of each acinar region, relating the pressure it exerts on its branch to the volume of the region itself and the pressure outside it:

$$P_i = \frac{1}{C_i} V_i + P_{pl}(t) \tag{4}$$

where $P_{pl}(t)$ is the pleural pressure (i.e. the "pressure" from the diaphragm, outside the acinar region) at the current time and C_i is the *compliance* of the acinar region of branch i. The pleural pressure changes over time to mimic human breathing patterns – hence why it is parameterised by t.

3.2 Modelling in the Abstract

We use an *implicit* Euler's method to model the system as it progresses: at each timestep, our simulation updates its state to the value of an approximate solution to the system of equations above. Equation 4 provides the necessary bounds to make the method implicit, giving us higher accuracy at the cost of implementation complexity.

To solve for an approximate solution at each timestep, we use Newton's method with $f_{\boldsymbol{S}}(\boldsymbol{x})$ as defined below, iterating until $\|f_{\boldsymbol{S}}(\boldsymbol{x})\|^2 \leq tol$ and $\|dx\|^2 \leq tol$, with a tolerance of 10^{-6}. The two "inputs" – $\boldsymbol{S}$ and $\boldsymbol{x}$ – partition the state of the model into the variables that are controlled externally (e.g.: pleural pressure, compliance) and those that are calculated from the system state (e.g.: acinar volume, airflow). The definitions of $\boldsymbol{x}$ and $f_{\boldsymbol{S}}$ are given by:

$$x = (P_i..., Q_i..., V_i...)$$

$$f_{\boldsymbol{S}}(\boldsymbol{x}) = \begin{bmatrix} P_{\text{parent}(i)} - P_i - R(i)Q_i \\ \vdots \\ Q_i - \sum Q_{\text{child}} \\ \vdots \\ V_i^t - V_i^{t-1} - dtQ_i^t \\ \vdots \\ P_i - P_{pl}(t) - \frac{1}{C_i}V_i \\ \vdots \end{bmatrix}$$

Note that the values in $\boldsymbol{x}$ and equations in f are repeated only as many times as fits; e.g., there are fewer acinar regions than total branches, so there are fewer components in $\boldsymbol{x}$ from each V_i than from each Q_i.

As $\boldsymbol{S}$ only exists in the abstract sense, we won't bother to define its structure; all that's necessary to know is that it contains every variable referred to in f that is not already given explicitly by $\boldsymbol{x}$.

It's worth noting that in practice, the above definitions are only *nearly* correct; a few adjustments were made to the inputs and equations to mitigate limitations from floating-point accuracy. These are discussed in the next section.

3.3 Units & Modified Equations for Numerical Stability

It is worth making explict the units used for each value in our simultaneous equations. After careful consideration, these were considered to provide the best trade-off of familiar units and those with values of magnitude close to one, where floating-point accuracy is maximized. As we will see momentarily, the spread was still quite wide. The chosen units were:

Type of thing ???	Units
Distance	m
Volume	m^3
Flow velocity	$\frac{\mathrm{m}^3}{\mathrm{s}}$
Density	$\frac{\mathrm{kg}}{\mathrm{m}^3}$
Pressure	Pascals $\left(\frac{\mathrm{kg}}{\mathrm{m}\cdot\mathrm{s}^2}\right)$
Compliance	$\frac{\mathrm{m}^3}{\mathrm{Pascal}}$ $\left(\frac{\mathrm{m}^4\cdot\mathrm{s}^2}{\mathrm{kg}}\right)$
Resistance	$\frac{\mathrm{kg}}{\mathrm{m}^4\cdot\mathrm{s}}$
Viscosity	$\frac{\mathrm{kg}}{\mathrm{m}\cdot\mathrm{s}}$

One of the challenges with using these units is that some values are at a much greater magnitude than the others. For example, the pressure inside each branch is close to atmospheric pressure – or about 10^5 Pascals, but pressure *gradients* are typically much smaller.

In practice, this can mean that if the dx from our Euler step is too small, the pressure won't change; it doesn't have the necessary precision at that magnitude.

To mitigate this issue, we define two new values: $\hat{P}$ and $\hat{V}$, which are given by:

$$\hat{P} = P - P_{\text{atm}} \tag{5}$$

where P_{atm} is is atmospheric pressure; and:

$$\hat{V} = V - V|_{P=P_{\text{atm}}} \tag{6}$$

$$= C(\hat{P} - P_{pl}) \tag{7}$$

Note that the definition of $\hat{V}$ would be the result of simply substituting $\hat{P}$ for P in 4. Applying these substitutions gives the following equations, equivalent to their counterparts above:

$$\hat{P}_{\text{parent}} - \hat{P}_i = R(i)Q_i \tag{8}$$

$$Q_i = \sum Q_{\text{child}} \tag{9}$$

$$\hat{V}_i^t = \hat{V}_i^{t-1} + dtQ_i^t \tag{10}$$

$$\hat{P}_i = \frac{1}{C_i}\hat{V}_i + P_{pl}(t) \tag{11}$$

Representing the pressure and volume by their *offset* from values at atmospheric pressure causes them to cluster much closer to zero – the magnitude of the mean is significantly decreased, relative to the variance of the values. This of course greatly improves the accuracy of each Euler step.

Note: The same substitutions also apply to our representations of the state of the model & the optimization function used for Newton's method, as shown in subsection 3.2.

3.4 Sparse Matrices

A key observation that can aid in simulation speed is that we can represent the Jacobean of our optimization function f in $\mathcal{O}(n)$ space using sparse matrices – which is necessary to allow simulation of the lungs up to a high depth without a quadratic blow-up in runtime.

Todo: **There's a small disconnect between the set of algoithms we *could* use for solving the system versus the actual set of algorithms that I can find packages implementing.**

So the remaining questions here are along the lines of: what are the typical constraints of algorithms to use here, and how does the existing structure of the matrices interact with that?

My current thinking for this section is that it should talk about the available options for organizing the sparse matrices (particularly comparing with Foy, which arranges the array in a pattern that's not quite as conducive for sparse solving algorithms). This should probably be a side-by-side comparision with some fancy-ish figures, using the ones I've laid out below.

There's some interesting points about how the structure of the equations (namely: there are always at least two variables) means that we "naturally" get an uppper/lower triangular matrix.

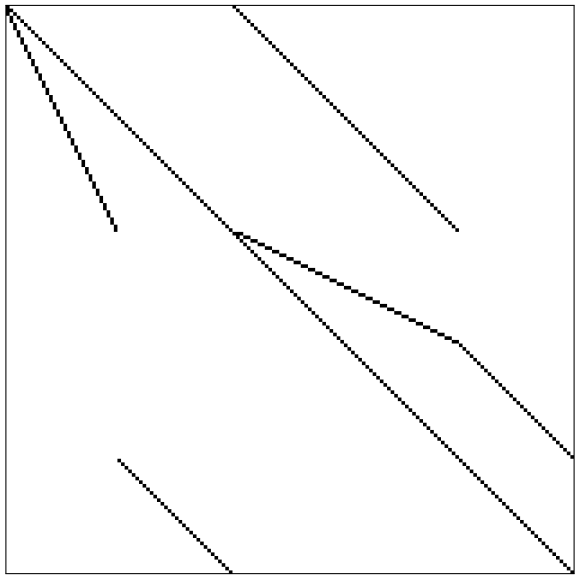

Figure 1: Values and equations separated by type (as currently implemented, and specified in Foy's paper), i.e. $(P..., Q..., V...)$.

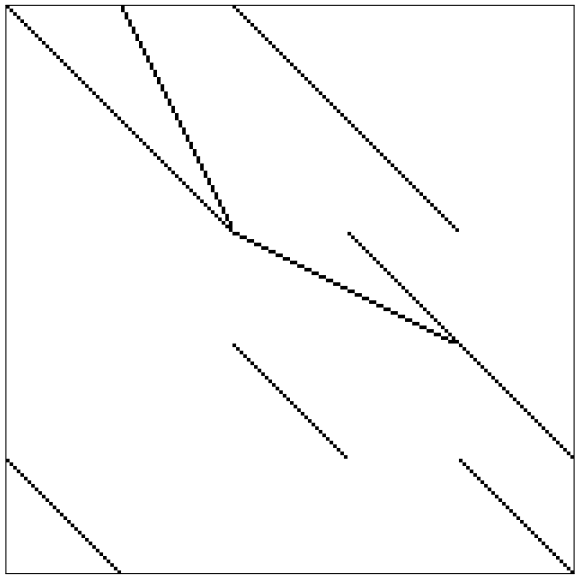

Figure 2: What's *actually* currently implemented- because of the way that we assign branch numbers. All placements of values and equations are reversed in their slots from the previous figure.

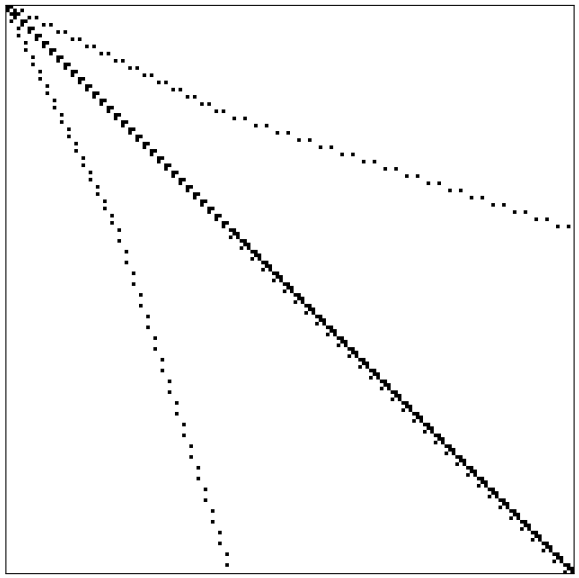

Figure 3: All values and equations for each branch followed by the next, i.e. $(..., P_i, Q_i, V_i, ...)$.

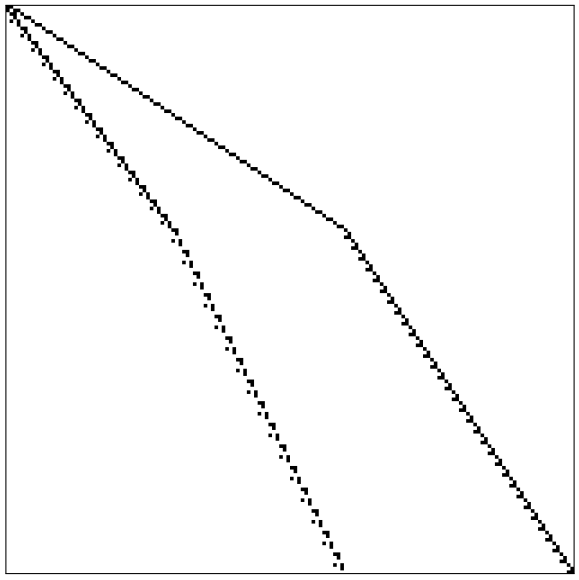

Figure 4: Like the one above, but the variable for a branch's flow is placed directly after its parent's variables, not with its own: i.e. $(..., P_i, Q_{\text{left}}, Q_{\text{right}}, ...)$

3.5 Procedural Lung Generation & Configuration

As part of the simluation software developed for this paper, there are a number of configurable parameters – primarily encapsulating the structure of the lungs and how they change over the course of the simulation. Listing them exhaustively, the parameters are:

- Lung structure (the tree formed by the relationship between branches)

- Branch length, as relative to the parent
- Branch radius (healthy & degraded), relative to the parent
- Acinar region *relative compliance* (healthy & degraded)
- Branch angles (only affects the lungs' appearance)
- Keyframe-based scheduling to shift the lung state between healthy & degraded
  - Multiple interpolation functions to transition between keyframes
- Pleural pressure wave characteristics (initial value, mean, amplitude, and period)

The specification of "healthy" and "degraded" allows us to define and transition between two states of the lungs, simulating the onset and treatment of the affects from many respiratory illnesses. The speed of onset or recovery is controlled by the positioning of the keyframes, which are binary "healthy" or "degraded" states. These states do not directly affect the simulation state, i.e. the flow, pressure, and acinar volume within each branch. This does mean that, for example, the pressure in constricting airways does not increase directly as a result of that constriction, but we have considered the volume inside the branches themselves largely negligible. The most important factor is that the volume within each acinar region is conserved, which *is* guaranteed, even across changing compliance.

It is also worth clarifying the meaning of *relative compliance* in the list above. In order to ensure that the total volume of the lungs remains consistent with human anatomy, the compliance of each acinar region is linearly scaled so that the total volume from all acinar regions filled at atmospheric pressure is 2.0 liters.

Todo: **This should probably not be 2 liters; looking at the actual data, the starting volume of the lung doesn't really correspond to FRC *or* FRC + tidal volume, so it should be closer. There's also the issue that the tidal volume that the simulation reports is *much*, much less than what it should be – like... 40 milliliters kind of small.**

This *probably* has something to do with the compliances being much smaller than what Foy's paper had, but I'm still really not sure why that was happening.

A comparison and analysis of transition interpolation functions is given in section 4, alongside their definitions.

The details of the configuration format itself are outside the scope of this paper, but the configurable values are described here to provide context for the parameters that were varied in the experiments we report.

4 Results

4.1 Observed numerical stability

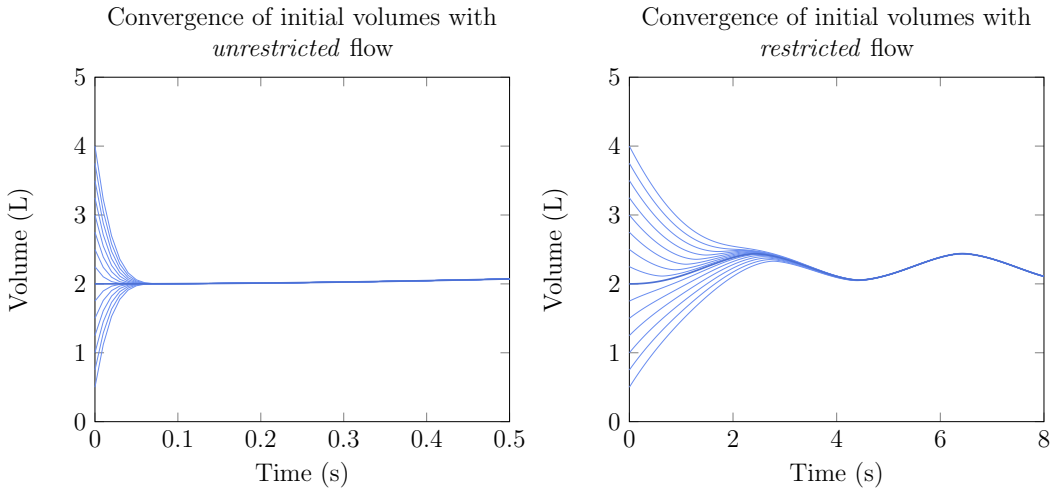

Figure 5: Simulated volume at the start of a breathing cycle, with varied initial volumes. Both experiments with a symmetric model with a depth of 6 (i.e., 63 total branches). Graphs display the distinction between unrestricted (left) vs 80% constriction (right). **Note:** displayed timespan differs between the left and right graphs.

To be confident in the results of other experiments, it is first crucial to determine that the simulation remains stable after running for extended periods of time. To do this, we simulated a simpler model (fully symmetric, no constriction, depth of 6) for 1000 seconds – which required 100,000 simulation ticks.

It is at this point that we'd ideally reference some figure to show that the system is stable in this configuration, but the series of volumes at each timestamp – starting at 0, 100, and 1000 seconds – were all the same, up to nine significant figures. In other words, total volume of air in the lungs over the course of each "breath" did not change over the course of an atypically lengthy experiment.

We also considered that the initial volume used in experimentation is not guaranteed to be accurate to the "typical" volume at that point in the breathing cycle – a fact that becomes visible with higher degrees of airway constriction (discussed in subsection 4.2). Therefore, we also experimented with significantly changed initial volumes, as shown above in Figure 5. The system quickly recovers from pertubations when airflow is unrestricted, but is slower to return to the typical volume when resistance prevents the correction from being made more quickly.

4.2 Flow characteristics under varied constriction

Todo: **This section should basically just say: yeah, this is expected. It should reference whatever other papers have previously reported on these phenomena.**

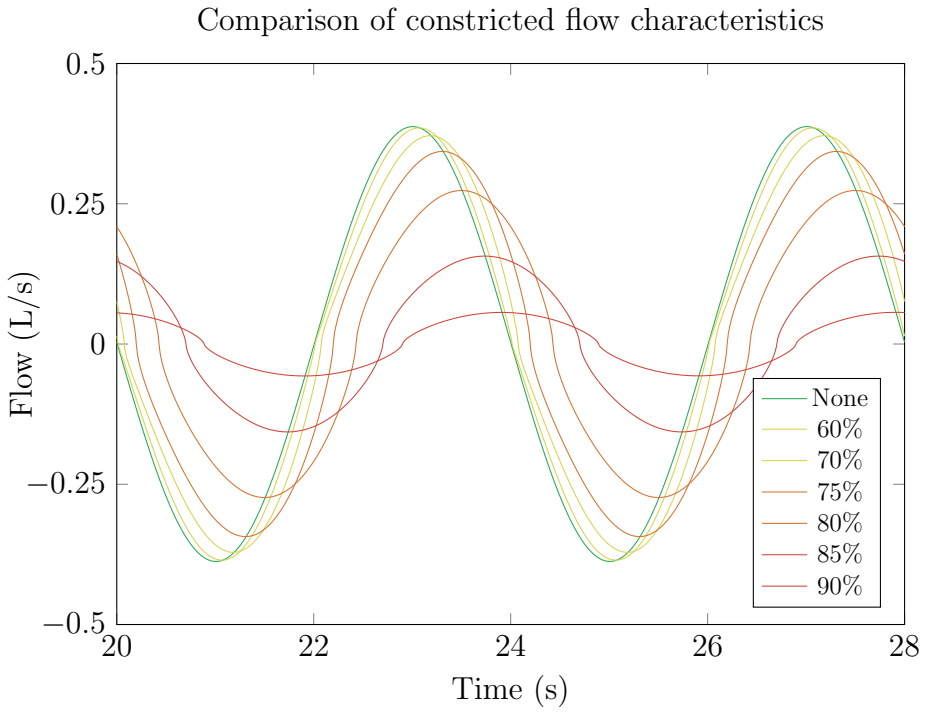

Figure 6: Stable flow during two breathing cycles with varied levels of whole-lung airway constriction

We delay until 20s after starting because we know from above that longer adjustment periods are required for constricted flow.

4.3 Comparison of interpolation functions

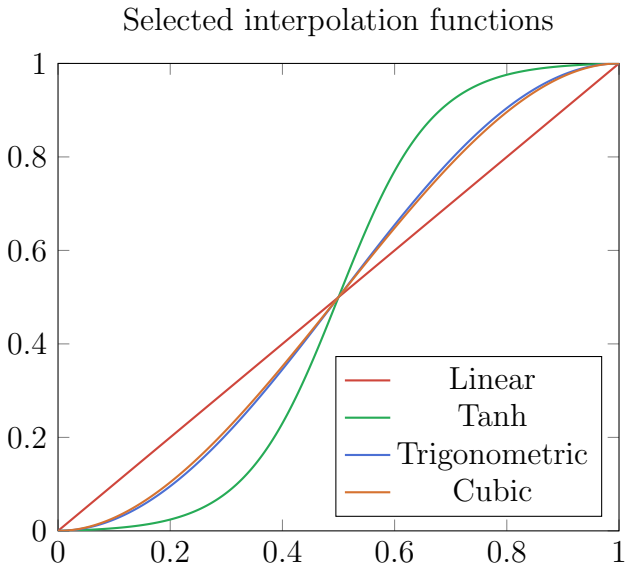

Figure 7: Comparison of the four selected interpolation functions

Later experiments explore the effects of rapid changes in airway constriction. To move between these "constricted" and "unconstricted" states, we have four different interpolation functions:

- *Linear*: $f(x) = x$
- *Tanh*: $f(x) = \frac{1}{2}\frac{\tanh(6x-3)}{\tanh(3)} + \frac{1}{2}$
- *Trigonometric*: $f(x) = \frac{1}{2}(1 - \cos(\pi x))$
- *Cubic*: $f(x) = 3x^2 - 2x^3$

A visual comparison of these functions is provided in Figure 7.

These functions were selected for their relative simplicity and variety of curvature.

Todo: **Explaining exactly why the curvature is necessary is escaping me right now. It's basically that the flow behaves extra weirdly without curvy interpolation, but that's not very scientific.**

There's an alternate angle, which is that curvy interpolation intentionally allows the significant changes at high constriction to happen more slowly (e.g. going 10% to 15% is much slower than 15% to 20%). This is "good" because flow characteristics change rapidly as you change constriction *from* high constrictions, so moving slowly ensures that the flow is less erratic.

On the other hand, maybe this is biasing the results and *actually* the erratic flow is more realistic. Would be good to discuss.

Todo: **The other piece of writing in this section is going to be a comparison of the flow characteristics when when switching on various interpolation methods.**

We eventually choose *Tanh*, basically because it looks the best (subject to change!). See above notes.

Finally, it is worth noting that the correction factor of tanh(3) in the denominator of the *Tanh* interpolation function is nearly equal to 1, but still necessary. The value of tanh(3) is only 0.995, but experiments without that correction factor showed clearly visible instantaneous changes in flow from that small change. While these changes were most likely harmless, we still elected to remove them.

Todo: **Here's an outline of the remaining items I'd like to put in this section, in a rough order:**

- Stable flow characteristics of lungs with varying degradation
  - Basically just want to point out the ways that the curve starts to become lopsided and out of phase from the pleural pressure, with particular emphasis on the way that the effect starts rapidly heightening as the constriction approaches 10% (about – I haven't tested beyond that, which I should).
- Onset and recovery from airflow constriction. Key things are:
  - Just showing off the simple graphs, to get a sense of what's going on
  - Slow recovery reduces strain – we can get abnormally high airflow if pleural pressure lines up with suddenly-reduced constriction
  - Relating to above: I'd like to have a version of the crazy graph – the one where it's all just "degrade and recover", with each startpoint being offset by 0.5s from the previous one. It would be really good to align that with some kind of chart showing the minimum and maximum flow velocity reached (on top of the normal min/max), to show the kind of chance of risk we run – and how much that reduces as the recovery becomes slower.
  - Flow characteristics align with those from the current level of constriction (i.e. on the way to 10%, the area around 20% constriction tends to mimic the flow characteristics from 20% at that point)
  - The above points are probably three distinct sub-sections.

References

[1] The global impact of respiratory disease – second edition.

[2] BH Foy. Computational models of pulmonary function tests, 2018.