Dynamic Optimization Project Report

# Modeling and Estimation of a Pneumatic Robot Actuator

April 16, 2022

Curtis C. Johnson

# Contents

## Cover Letter

Dr. Hedengren (and Daniel):

This project aims to model and estimate the the dynamics of a soft pneumatic actuator used for robotic manipulators. This is important because pneumatic actuation is on the same order of magnitude (in terms of speed/frequency) as the dynamics of the entire robot. In the past we have simply modeled these effects as a linear, first order system so that the higher level dynamic controllers (i.e. for the robot arm) are aware that actuation itself takes time. We've found however that this assumption is too strong and hurts the performance of upper level controllers–but these dynamics are very complex and difficult to measure. A more accurate model (but still a relatively simple one) will allow us to produce more realistic simulations of the entire robot, improve the performance of robot level control, and help us understand how to improve actuation performance when necessary.

The main contributions of this work are:

- Adaptation of a first-principles model to a soft pneumatic actuator.
- Identification and regression to develop a model for the valves.
- Development of a simpler and numerically favorable parametric model.
- Estimation of parameters in both models.
- Moving horizon estimator written for the parametric model.
- Analysis of design trade-offs and model performance in context of real-time control requirements.

Curtis Johnson

## Abstract

This paper presents the development and comparison of several different models for a pneumatic robot joint. I analyze and optimize a first-principles model based on the 'thin plate flow' assumption which captures choked and unchoked flow conditions through an orifice. I also develop and optimize a simplified parametric model which has less physical intuition but has more favorable numerical properties. These models are compared on their ability to predict the dynamic response of the robot joint using hardware data collected through a pressure-sensing PCB module. The parametric model outperforms the first-principles model but fails to capture the effects from unmodeled valve leakage. To address this, I implement a moving horizon estimator (MHE) and show that the parametric model augmented with MHE is able to capture a larger portion of the unmodeled dynamics at the cost of slightly lower nominal performance.


## 1 Introduction

Robotic manipulators typically used in industry and research are rigidly linked joints driven by motors, like the Panda in Figure 1. Pneumatic actuators are beneficial over motors in robotics because of their natural compliance, low cost, and ease of use. Often robots with pneumatic actuation use pneumatic cylinders which require a rigid mounting structure and joint as in Figure 2. The rigid mounting structure adds weight and cost to the overall design of the robot. Consider instead a 'soft' robot design where the pneumatic actuator itself is made of a flexible material like plastic like Figure 3. A robot constructed from joints like this no longer has discrete pin-like joints but instead bends in a continuum, like the trunk of an elephant. Soft pneumatic actuators like this have a much higher power to weight ratio than hard pneumatic cylinders.

**Figure 1:** Franka-Emika robot arm Panda. Typical motor-driver robot arm.

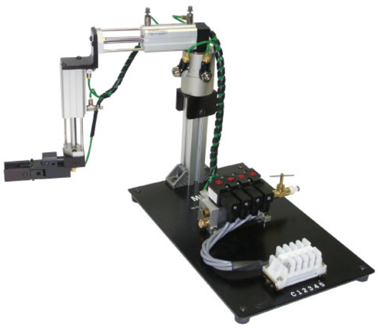

**Figure 2:** MB500 Pneumatic Robotics Module from TTI Technical Education Systems.

**Figure 3:** Soft Pneumatic Robot used in [2].

The dynamics of pneumatic actuators, however, are notoriously slow when compared with typical robot actuators (e.g. DC motors). DC motors typically settling times on the order of 10-50 ms, while the the soft pneumatic actuator used in this work (the orange joints in Figure 3) has a settling time on the order of 100-500 ms. Accordingly, the actuation dynamics should be included in the model of a pneumatic robot in order to achieve acceptable prediction performance.

The difficulty in modeling arises from the fact that the pressure dynamics in the chamber are highly nonlinear and coupled. Surely a complicated finite element model could capture these effects, but we intend to use this model for real-time control at 400 Hz, so the model must be as simple as possible while also retaining its prediction capability. At the same time, we need the model to generalize well because a typical robot arm can have up to 12 pneumatic actuators, all of which will be slightly different.

In summary, we need a model describing the dynamics of these soft pneumatic actuators so that higher level controllers on the whole robot perform well. This actuation model must be simple enough for real-time control, generalize to slightly different operating conditions, and provide reasonably accurate predictions for higher level control.

The structure of this paper is as follows: Section 2 discusses the hardware setup and various plausible model structures. Section 3 presents the simulation results of each nominal model while Section 4 shows the simulation results of all models after being optimized. Section 5 compares the performance of each model and Section 6 concludes with a discussion of the general principles learned.

## 2 Methods

### 2.1 Hardware

The hardware used for this work is a soft pneumatic joint shown in Figure 4, the same as those used in Figure 3. This joint has four bellows actuators which allow it to bend in any direction. These bellows are constrained length-wise by an inextensible steel cable.

Each bellows chamber is controlled by an Enfield LSV25 valve [3] and a custom PCB with receives pressure data and sends valve commands.

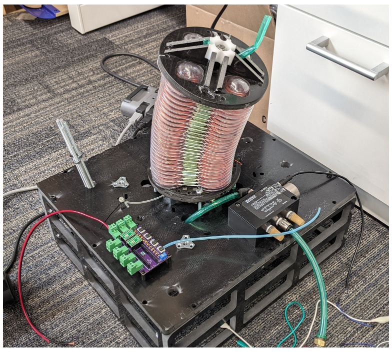

**Figure 4:** Soft pneumatic joint used in this work.

To simplify this work a little bit, only one bellows chamber/valve was used (instead of using all four). The black plates on the ends of the joint have holes which allow the joint to be bolted in place. This is important to be able to decouple the volumetric expansion from the pressure dynamics for easier analysis, as will be shown in the following sections.

### 2.2 Model Development

#### 2.2.1 First-Principles Model

The first principles model developed in this section borrows from [4] where the authors are modeling pnuematic cylinders on a humanoid robot. This paper is supplemented with [1] which delves deeper into the assumptions made in [4].

The model for the pressure dynamics $(p)$ in a chamber of volume (v) with input current $(u)$ is of the form

$$\begin{aligned} \dot{p}(p,u,\mathrm{v},\dot{\mathrm{v}}) &= \kappa\frac{RT}{\mathrm{v}}\dot{m} - \kappa\frac{\dot{\mathrm{v}}}{\mathrm{v}}p \\ \dot{m}(p,u) &= a(u)\phi\left(p_u, p_d\right) \end{aligned} \tag{1}$$

where $a(u)$ is a function mapping current applied to the valve to an orifice area,

$$\phi\left(p_u, p_d\right) = \begin{cases} z\left(p_u, p_d\right) & \text{if } p_u \geq p_d \\ -z\left(p_d, p_u\right) & \text{if } p_u < p_d \end{cases} \tag{2}$$

is a function that enforces flow direction (i.e. into or out of the chamber),

$$z\left(p_u, p_d\right) = \begin{cases} \alpha p_u \sqrt{\left(\frac{p_d}{p_u}\right)^{\frac{2}{\kappa}} - \left(\frac{p_d}{p_u}\right)^{\frac{\kappa+1}{\kappa}}} & \text{for } p_u/p_d \leq \theta \\ \beta p_u & \text{for } p_u/p_d > \theta \end{cases} \tag{3}$$

is a function that accounts for different flow regimes of a compressible fluid, and

$$\begin{aligned} \alpha &= C\sqrt{\frac{2M}{ZRT}\frac{\kappa}{\kappa-1}} \\ \beta &= C\sqrt{\frac{\kappa M}{ZRT}\left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{\kappa-1}}} \\ \theta &= \left(\frac{\kappa+1}{2}\right)^{\frac{\kappa}{\kappa-1}} \end{aligned} \tag{4}$$

which are physical parameters.

The convention of this model is to assume that positive mass flow is from upstream pressure $p_u$ to downstream pressure $p_d$. $M$, $Z$, $R$, $T$, $\kappa$, and $C$ are defined as

| | | |
|---|---|---|
| Gas Molecular Mass | $M$ | 0.029 for air, Kg/mol |
| Temperature | $T$ | $K°$ |
| Universal Gas Constant | $R$ | $8.31\left(\text{Pa}\cdot\text{m}^3\right)/(\text{molK}°)$ |
| Discharge coefficient | $C$ | 0.72, dimensionless |
| Compressibility Factor | $Z$ | 0.99 for air, dimensionless |
| Specific Heat Ratio | $\kappa$ | 1.4 for air, dimensionless |
| Mass Flow | $\dot{m}$ | Kg/s |
| Pressure | $p$ | Pascals |
| Area | $a$ | $\text{m}^2$ |

The first term of Equation 1 is the resulting change in pressure from mass flow into or out of the chamber and the second term is the pressure change resulting from a change in the chambers volume, which naturally occurs during normal operation of the joint as it moves around.

The overall pressure model above (Equations 1-4) takes as an input an orifice area $a(u)$. This is problematic because the Enfield valves that controls the mass flow does allow the direct commanding of orifice areas. This is, in general, not a feature of pneumatic valves. The datasheet for the valves provides a functional form from input command to orifice area as seen in Figure 5. Both the command and the aperture flow area are given as percentages. 100% flow area for this valve corresponds with an actual are of 25 $mm^2$. A command of 100% corresponds with a current of 1 amp being sent to the valve, while a command of 0% corresponds with a current of -1 amps. I approximated this figure adapting the function $smax(x) = (\sqrt{x^2+1}+x)/2$ as in [4] to our specific valve configuration.

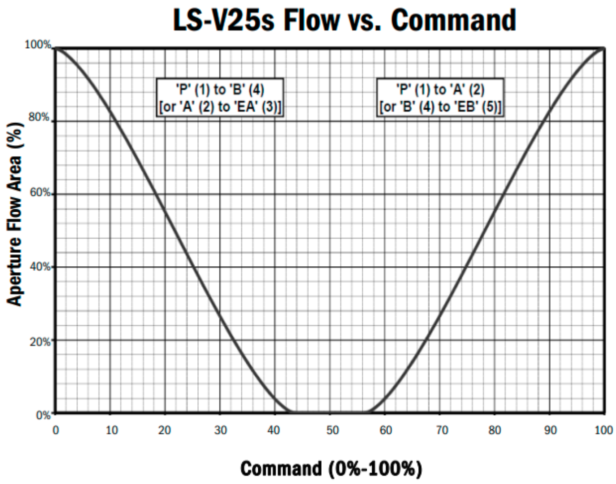


**Figure 5:** Figure provided in the Enfield Valve Datasheet [3]

In general, electronic control of current flow is a difficult and expensive task. However, controlling voltages is simple and cheap. Our custom PCB takes advantage of Ohm's Law ($V = iR$) in order to map a voltage level which we can control to a current output. The valve actuators themselves wound to a specific resistance $R$, in this case $3.8 \pm 0.304\ \Omega$.

So the remaining model that needs to be identified is a model mapping a voltage signal (i.e. a PWM) to current. I collected data on the platform and performed a simple linear regression to get the model shown in Figure 6. This model has an $R^2$ value of 0.997. Higher order polynomials provide a better fit, but were experimentally determined to over fit the data, so I use just the simple linear model.

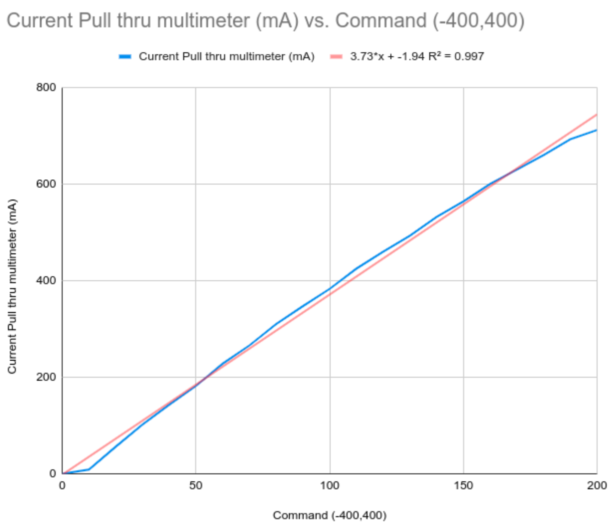


**Figure 6:** Linear Regression Model mapping PWM command to current. Blue line is data, red line is model fit.

We now have models mapping PWM to Current, Current to Orifice area, Orifice area to mass flow, and mass flow to pressures. As mentioned previously, the volume of the chamber v is fixed for this work. This means that the second term in Equation 1 is 0.

### 2.2.2 Parametric Model

The first-principles model developed in the previous section is the right form, but the dynamics (at least for physically realistic parameter values) are too slow (a plot of this will be discussed in Section 3). The analytical model also has a somewhat undesirable numerical behavior due to the switching conditions in Equations 1-4. These switches are continuously differentiable, but a large time step during numerical integration can cause the solution trajectory to chatter around the set point. There are also many parameters that can be difficult to identify.

To address these problems, I developed a parametric model inspired by the FOPDT model commonly used in the process control industry. The FOPDT model is unsuitable in this case however because of the different flow conditions. For example, the valve opening can cause air to flow into the chamber but not indefinitely. Mass flow will stop when the pressure in the chamber is equal to the source pressure, even if the valve is open. The FOPDT model cannot capture this phenomenon.

The model I developed instead is of the form

$$\tau \dot{p} = KAa(u)f(p,a) \tag{5}$$

where

$$f(p,a) = \begin{cases} (P_{src} - P) & \text{if } a \geq 0 \\ (p - P_{atm}) & \text{if } a < 0 \end{cases} \tag{6}$$

$f(p,u)$ is a function that captures the steady state pressure between a source pressure and atmospheric pressure. $K$ is a process gain, $A$ is the maximum orifice area, $a(u) \in [-1,1]$ is a percentage corresponding to the y axis of Figure 5, and $\tau$ is the process time constant. For this work, the product $KA = 1$ (i.e. $K = 1/A$), so we are left with just a single uncertain parameter: $\tau$.

This model suffers much less from numerical chattering and has only a single value to optimize, instead of several used in the first-principles model.

## 3 Simulation Results

Figure 7 shows the simulation results for the first-principles model. The parameters that I chose to optimize were the specific heat ratio $\kappa$, and the discharge coefficients $C$. I used a different coefficient for venting and filling. The results show that optimized solution is significantly better than the unoptimized version. The form of the trajectories looks correct, but clearly the model is not as fast as the data. If I allow parameters to exceed physically realistic values, the fit is very good, but a complete study of the source of this error is left for future work.

The cost function used for this simulation used a MSE objective function and the SLSQP algorithm in Scipy's optimization library. I used several other solvers as well (Nelder-Mead and BFGS) and SLSQP seems to converge the fastest for this problem. Each design variable was given a lower and upper bound as well. These bounds were tuned iteratively to help with convergence.

I observed experimentally that the cost landscape becomes multi-modal and prone to falling into local minima with an increase in parametric optimization variables. This is another challenge towards optimizing the first-principles model.

I then tried a similar optimization scheme on the parametric model by only optimizing $\tau$. The decrease in decision variables and the overall complexity of the model results in very good convergence to a minimum from several different starting conditions. The results of the optimized parametric model compared to the optimized first-principles model is shown in Figure 8.

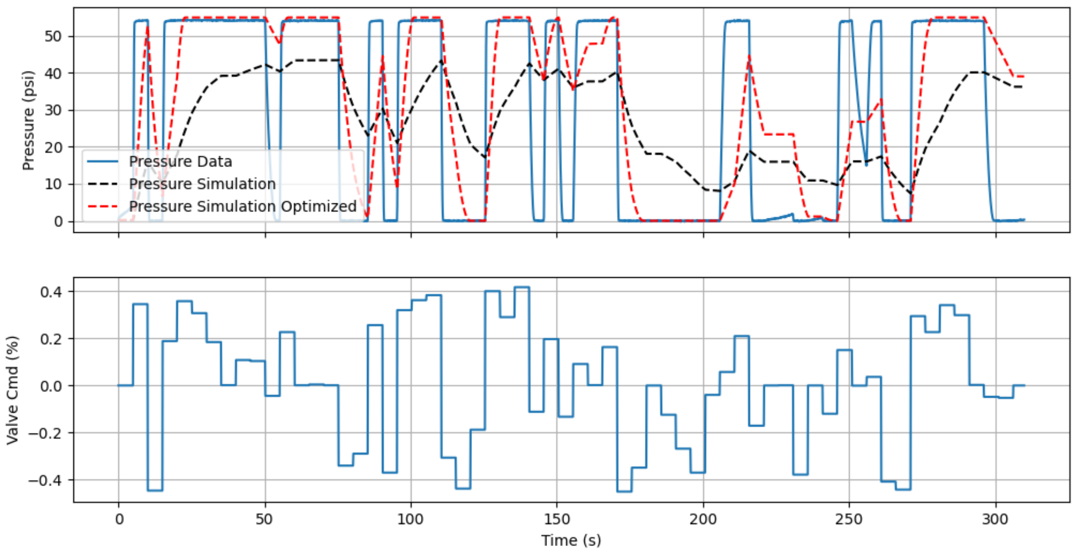


**Figure 7:** Simulation results for the First-Principles model with optimized vs unoptimized parameters with valve commands.

## 4 Estimation and Dynamic Optimization Results

I also implemented a moving horizon estimator (MHE) with the parametric model in order to compare against the single optimized value. The results of this are shown in Figure 9.

Figure 10 shows the same data zoomed in for better comparison. This MHE was also implemented using Scipy's optimize library with the SLSQP solver. I experimentally decided on using an L2 cost function with a small penalty on parameter movement (DCOST). I also found that iteratively tightening the bounds on the time constant parameter helped overall performance.

## 5 Discussion

The simulation results from the first-principles model shown in Figure 7 show that optimization improves the model significantly, but it clearly does not track very well. For example, there are several steps in quick succession around 150 s where the pressure actually reached atmospheric pressure, but the first-principles model barely reached 40 psi.

Compare these results to those in Figure 8. The parametric model tracks significantly better and was easier to optimize because of the single design variable. The most notable failure of the parametric model is the venting that occurs at just past 250 seconds. The orifice command at this moment in time is just barely negative, indicating a command to vent the chamber just slightly. The hardware data shows clearly that this was not the case and I presume it is because of leakage or stiction in the valve, where such a small command doesn't actually cause the valve to actuate. This effect is completely unmodeled (and frankly, undesirable) but occurs often in valves, especially as they age.

This effect is the major motivation for the development of the MHE introduced in Section 4. I wanted to see if allowing the time constant to change dynamically could boost the parametric model's capability to capture these unmodeled effects. Figure 9 shows these results. Though far from perfect, the simple addition of a MHE estimator reduced the modeling prediction error from around 30 psi to around 15 psi just past 250 seconds. The process of tuning the MHE revealed a few interesting design tradeoffs. The first of these is

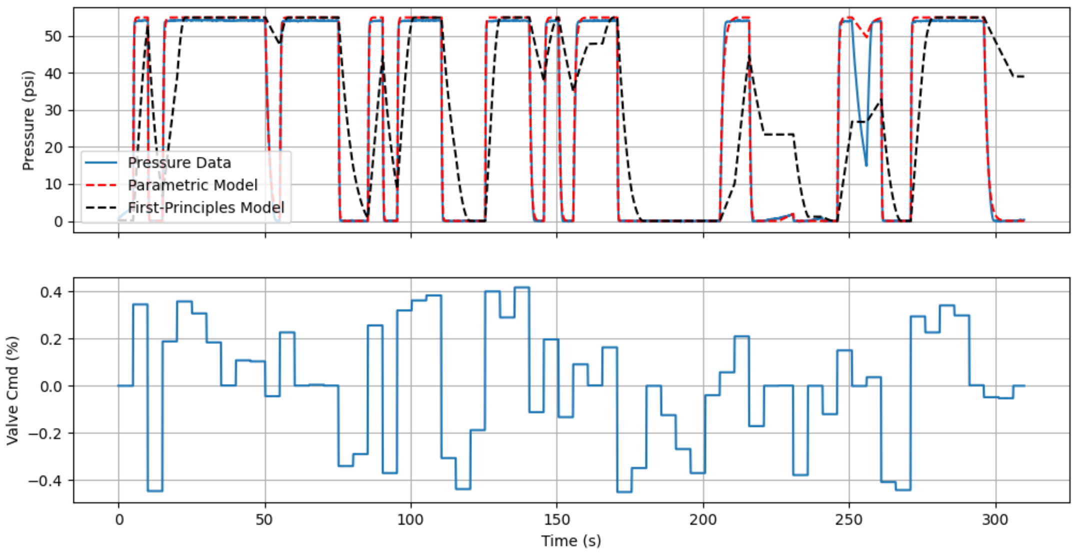


**Figure 8:** Comparison between first-principles and parametric models, both with optimized parameters.

that if I keep the DCOST low and allow larger bounds on the time constant, the error from the unmodeled dynamics can almost be completely eliminated. However, this causes a phenomenon that is easily seen in Figure 10, which is zoomed in significantly. From the perspective of this figure, it seems that the parametric model *without* the MHE performs better. The MHE adds a sort of 'parameter lag', where rising edges are too fast and falling edges are too slow. This occurs because it takes a few iterations for the optimal time constant to change, which are indicated by rapid value changes in the middle subplot, which is the optimal value for the time constant at that moment in time. Perhaps this effect can be tuned away using a longer horizon that includes more steps (the horizon was set to 1 second for this work). But doing so decreases the available control bandwidth because the optimization problem is slower. A higher DCOST did not seem to help alleviate this problem in this case either.

As the parameter bounds increase (to try and capture off-nominal/unmodeled dynamics), the parameter is allowed to move farther. This exacerbates the parameter lag problem mentioned above because the parameter now has to move a farther distance. In essence, there is a trade-off in tuning: tuning to capture unmodeled dynamics decreases nominal performance while tuning to track modeled dynamics closely decreases its ability to adapt to unmodeled dynamics. This loosely parallels the L1 vs L2 loss functions, where L1 is known to be much less sensitive to outliers. Generally we want to ignore outliers, but in this case, we care about the outliers – the unmodeled dynamics.

## 6 Conclusion

In summary, this work develops several models for a soft pneumatic actuator. Physics-based modeling, regression, and parameter estimation yielded a first principles model which did not track the hardware data well. To address this, a simpler parametric model was developed and optimized and shown to track hardware data well, with the exception of unmodeled dynamics. Further, a MHE was implemented and the design trade offs between nominal tracking and capturing unmodeled dynamics were explored.

Future work will include the volumetric expansion term in Equation 1. Identifying the reason why the first-principles model is too slow with realistic parameters is also an important area to explore because the form of the first-principles model is generally more accurate than that of the parametric model.

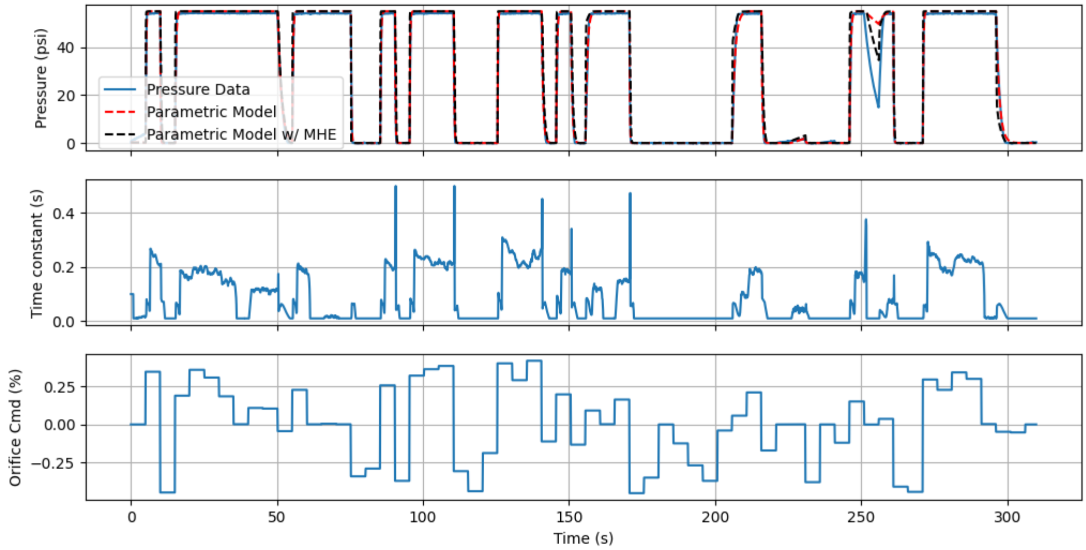


**Figure 9:** Comparison between the parametric model optimized with just a single time constant vs performing MHE on the time constant.

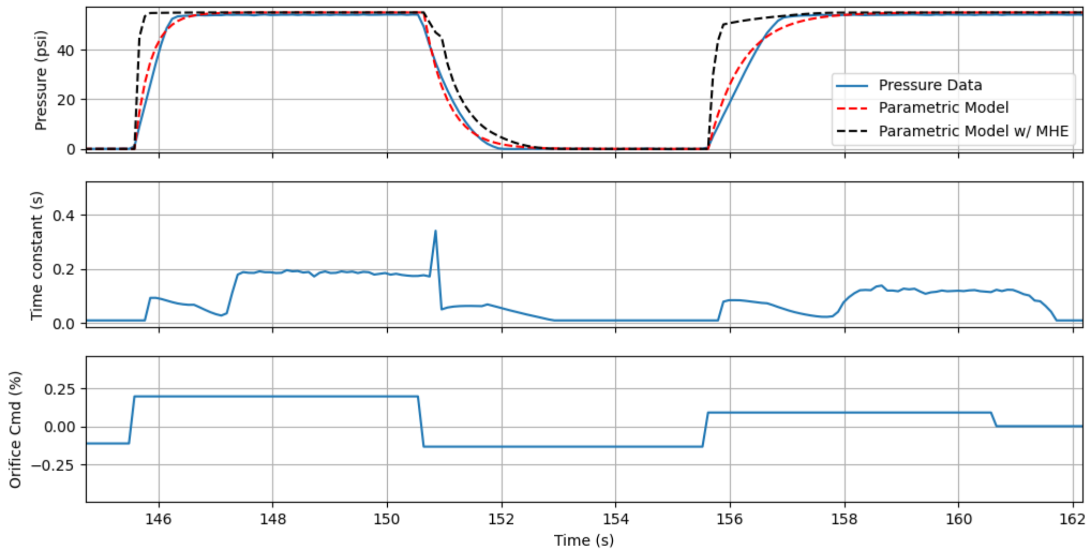


**Figure 10:** Same data as Figure 9, but zoomed in for better temporal resolution

## References


[1] B.W. Andersen. *Analysis and Design of Pneumatic Systems*. Wiley, 1967. ISBN: 978-0-471-02635-8. URL: https://books.google.com/books?id=FawYAQAAIAAJ.

[2] Wyatt Felt et al. "An Inductance-Based Sensing System for Bellows-Driven Continuum Joints in Soft Robots". In: July 2017. DOI: 10.15607/RSS.2017.XIII.027.

[3] *Proportional Pnuematic Control Valve*. LS-V25s. Rev. 20140203. Enfield Technologies. 2014.

[4] Yuval Tassa et al. "Modeling and identification of pneumatic actuators". In: *2013 IEEE International Conference on Mechatronics and Automation*. IEEE. 2013, pp. 437–443.