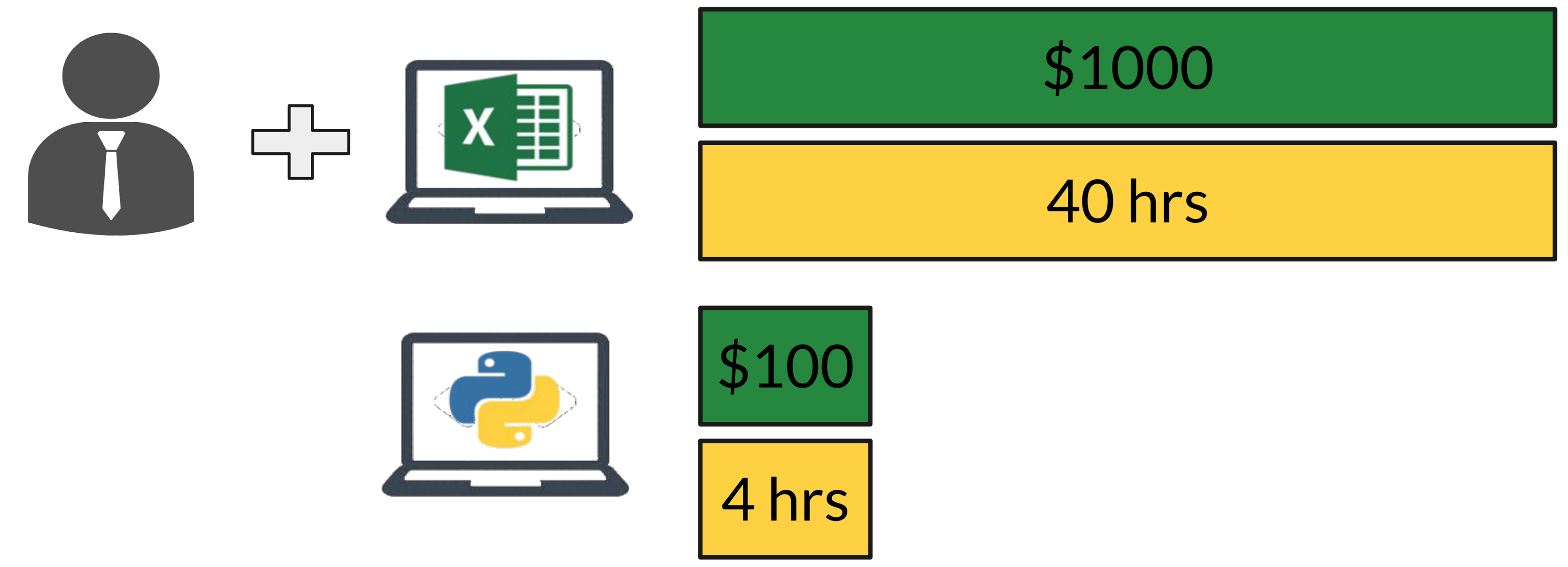
Algorithmic Automotive Fuel Delivery Tuning

Andrew Blomenberg and Dr. Jonathan Geisler

Background

Software plays a very large role in modern automobiles, especially in their engines. A finely-tuned electronic engine controller is necessary for both ideal performance and compliance with emissions laws. The complexity of these controllers is making it increasingly difficult for the aftermarket automotive community to modify their vehicles. Any aftermarket modification of the physical engine is likely to change the behavior of the engine enough that the engine controller will no longer be able to properly control the engine. This generally results in poor engine performance and may damage the engine in extreme cases.

The aftermarket community is looking for tools that allow them to continue modifying their engines without having to learn or pay for the process of manually reprogramming their engine controllers. The algorithm we wrote is designed to do the work of reprogramming an engine controller automatically.



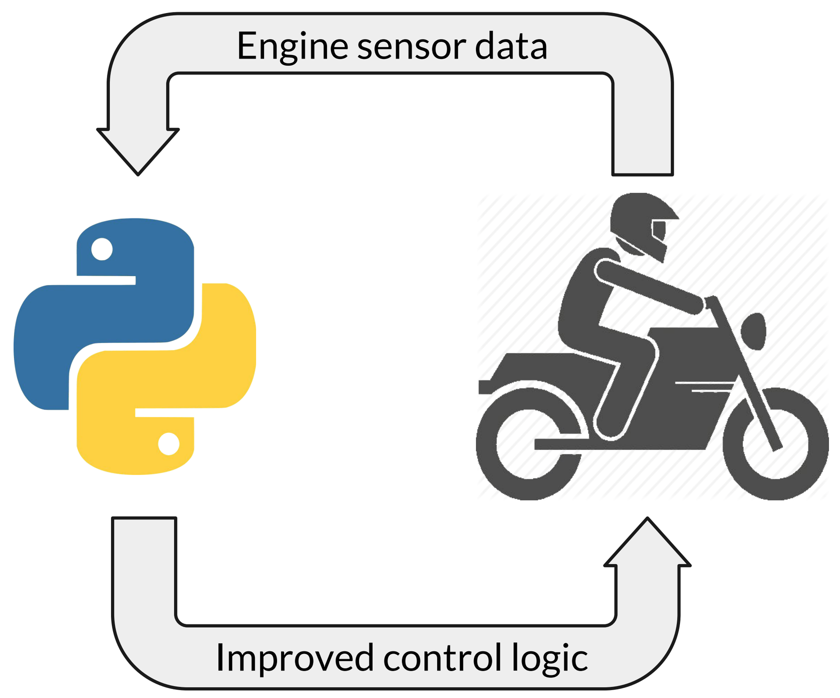
Platform

The platform we used for development was a 2010 Buell XB12 motorcycle. This motorcycle is fitted with additional engine sensors as well as a logging unit that records readings from these sensors.

Algorithm

The steps for the algorithm are described in the Algorithm Process section below.

The diagram to the right describes the algorithm's inputs and outputs. The algorithm is not intended to change the engine's electronic controller in real time for a variety of reasons including safety. Instead, the algorithm takes a chunk of logged sensor data and calculates a new configuration for the engine controller from that.



Demonstration

We tested the software in both real-life and contrived scenarios with encouraging results. The table below shows how much the algorithm changed a table within the engine controller. We intentionally made the table too high in the yellow region and you can see that the algorithm corrected for that. The algorithm also made smaller adjustments ot other regions.

0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
-1%	-1%	-3%	-3%	-2%	-2%	0%	0%	0%	0%	0%	0%	0%	0%
-1%	-1%	-1%	-3%	-1%	-3%	-8%	-4%	-2%	0%	0%	0%	0%	0%
0%	-1%	-1%	-1%	-1%	-1%	-12%	-8%	-8%	-2%	-2%	-2%	0%	0%
0%	0%	0%	0%	-1%	-2%	-11%	-10%	-10%	-1%	0%	0%	-2%	0%
0%	0%	0%	0%	0%	-3%	-3%	-2%	-4%	-1%	0%	0%	-1%	0%
0%	0%	-1%	0%	0%	-1%	-1%	-1%	-4%	-2%	-1%	-2%	-1%	0%
0%	0%	0%	0%	0%	-2%	-1%	0%	-1%	-2%	-1%	0%	-1%	0%
0%	0%	0%	0%	0%	-1%	0%	0%	-1%	-1%	-1%	0%	-1%	0%
0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	-1%	-1%	-1%	0%
0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

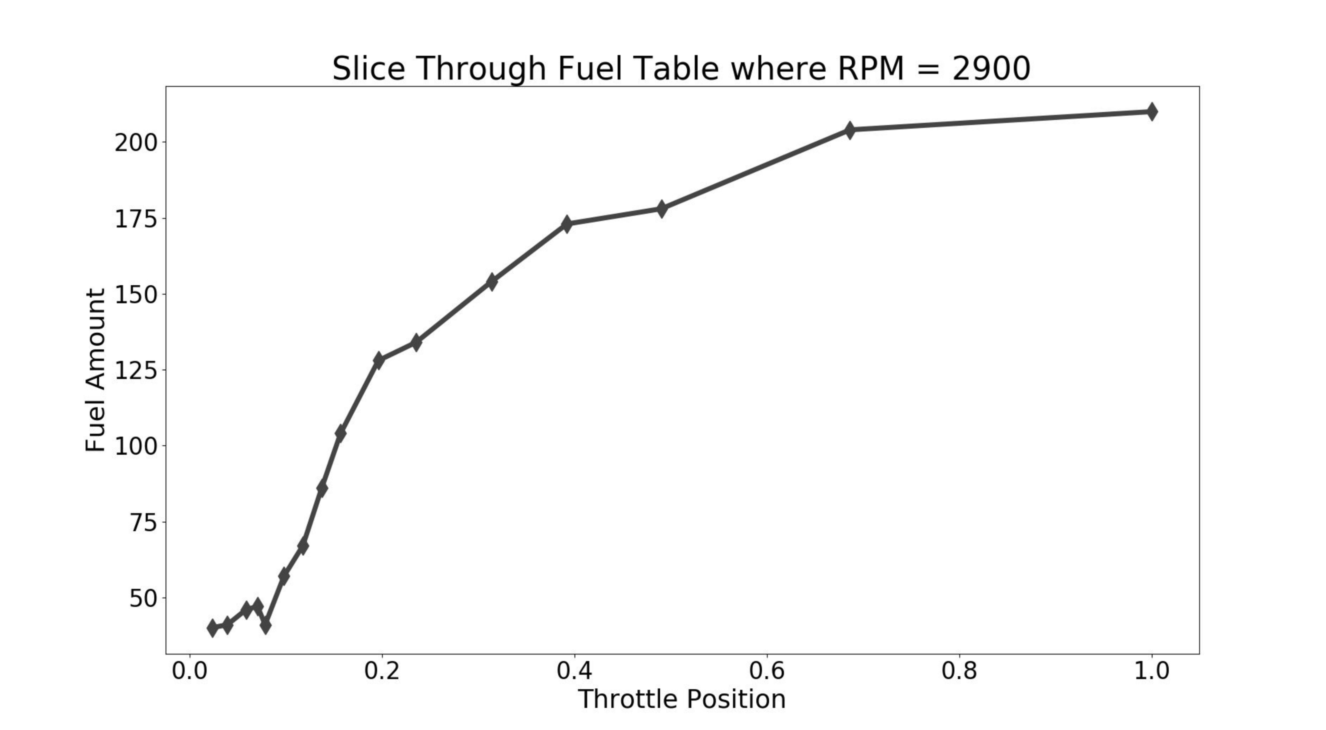
Conclusion

The software is meeting the goals we set out to achieve. It could use some fine-tuning and needs to be packaged in a way that makes it more usable for end users, however.

I would like to thank the CS department and especially Dr Geisler for giving me facilities, guidance, and ideas throughout this process. If you are a student considering research, I can tell you that making this was a lot of work but a lot of fun.

See a draft of the full paper here: goo.gl/hkFK7L

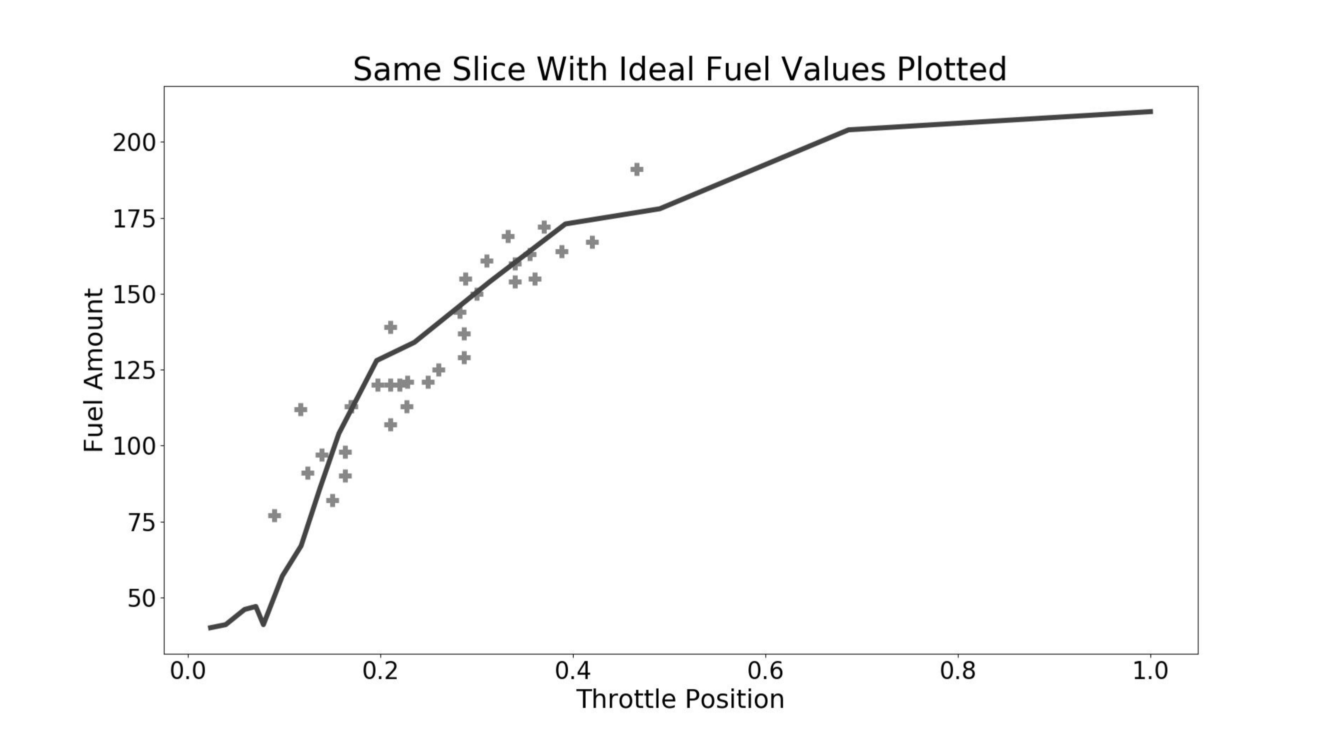
Algorithm Process



Step 0: Before the Run

The algorithm starts with a table of values that describe how much fuel to deliver to the engine given other factors like the position of the throttle.

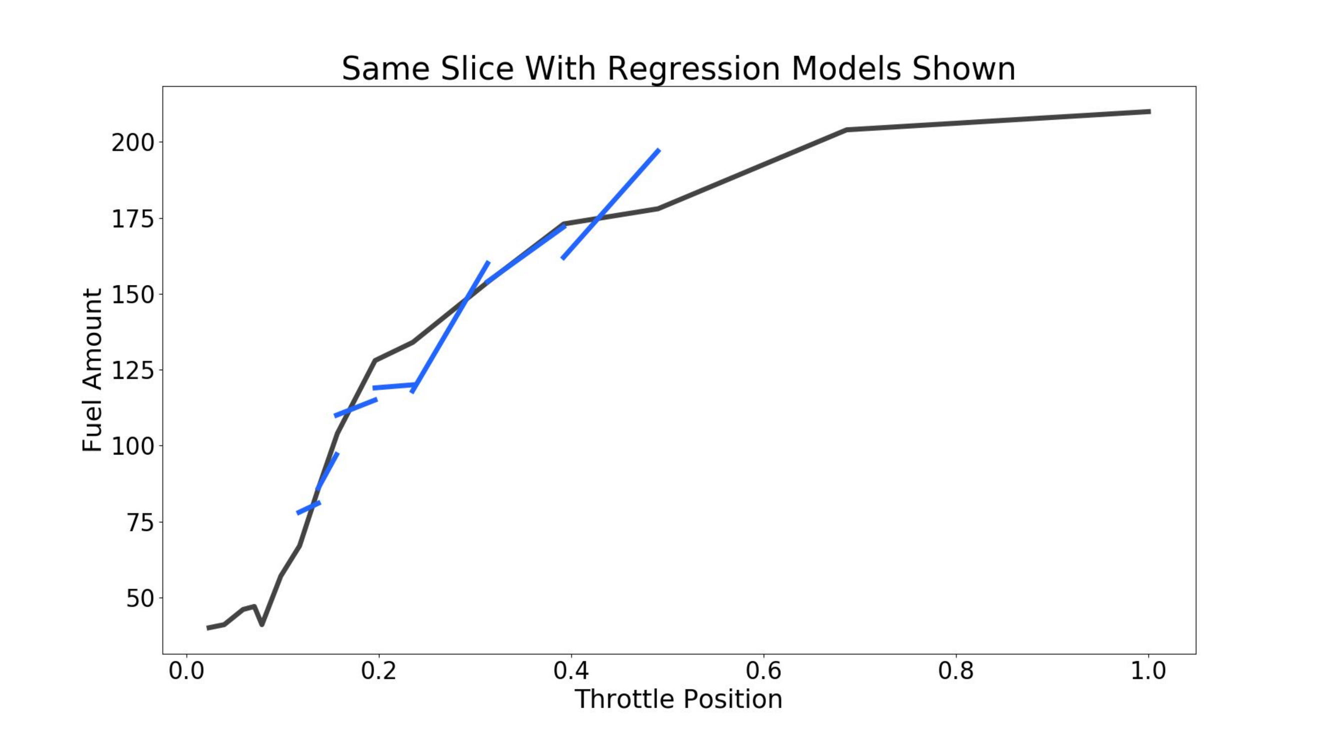
The purpose of our algorithm is to improve this table of values based on the sensor data we collected from our test vehicle.



Step 1: Ideal Fuel

The algorithm calculates a value called ideal fuel for every data point it was given. This value represents how much fuel should have ideally been injected into the engine.

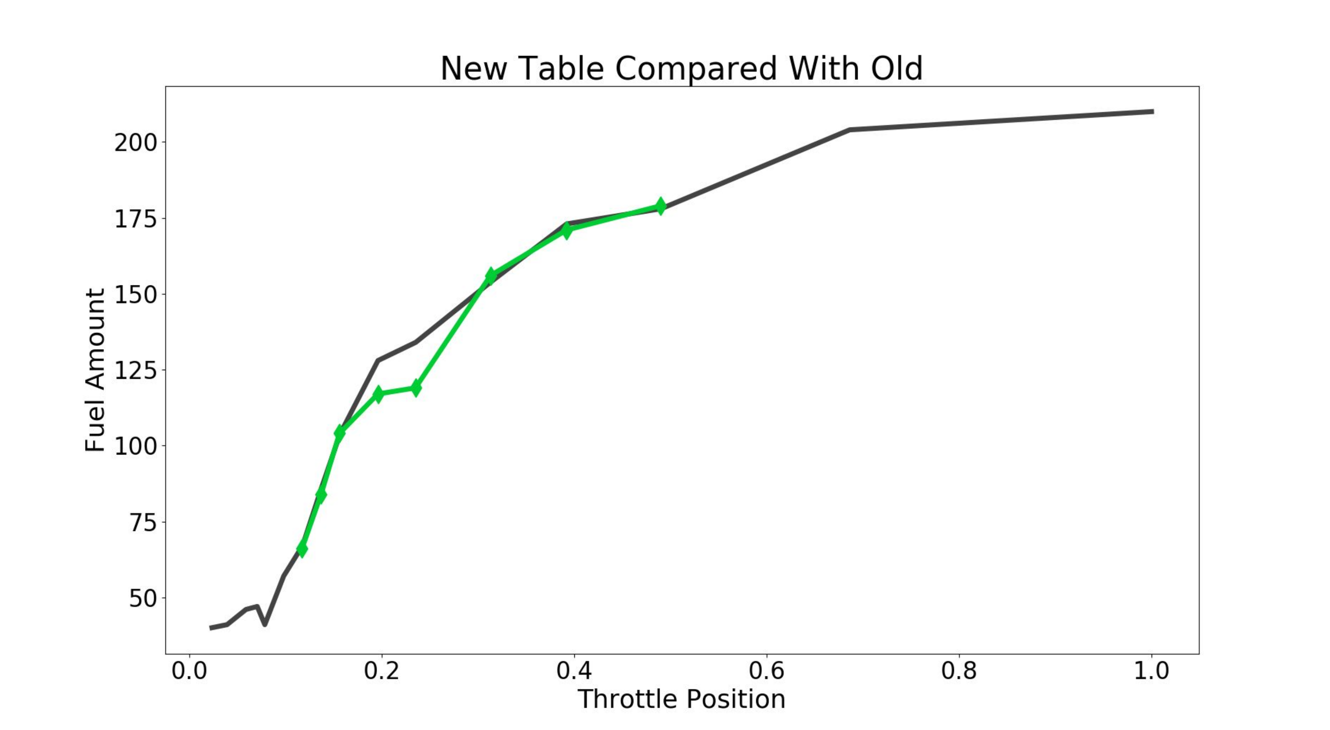
We are able to calculate one ideal fuel value for each combustion event.



Step 2: Regression

The algorithm models the ideal fuel values with linear regression. This turns the separate and often significantly different ideal fuel values into more cohesive models.

However, the regression models do not yet form a smooth curve,



Step 3: Weighted Average

The algorithm stitches the regression models together using a weighted average. This weighted average gives a higher weight to regression models that represent more ideal fuel values and have a higher r^2 values. At the end of this step, we have a new fuel table.