CruiseAuto Project – Milestone 4

Technical Brief

To: Rowan Harper, Lead Control Systems Engineer at CruiseAuto

From: Team 03, Ethan Zhang, Peter Teal, John Soares, Justin Clark

Date: 04/30/2025

RE: Technical Brief discussing the calculated parameter values for the Adapted Cruise Control (ACC) system

Introduction

Cruise Auto has asked our team to analyze the changes caused by using different tire materials with their ACC system. To do this we analyzed the changes in 4 parameters (initial velocity, final velocity, acceleration start time, and time constant), between 3 tire types (all season, winter, and summer) and different vehicle types (sedans, compacts, and SUVs). From this analysis we had to determine whether or not the changes in these parameters resulted in the ACC system operating within the given performance boundaries indicating the system's safety when used with any given tire and car type.

To ensure the accuracy of our analysis algorithm we had to compare our calculated parameters to a series of benchmark tests with our percent error and calculated SSE value indicating our accuracy. Throughout this process our team had to operate under some constraints, those being, inaccuracies in the given data measurements resulting in noisy data, variability in the reported speed and the actual speed of the vehicle due to the noise in the data, and inconsistent testing conditions causing varied acceleration start times between vehicles. However, overcoming these constraints guided our team into finding our solution by forcing us to smooth out the data, and calculate each car's acceleration start time, allowing us to generate an accurate final model. Our algorithm worked by initially smoothing out the given data set, before using an acceleration analysis algorithm to find when the vehicle's acceleration starts. This value was then used to find the average velocity before the vehicle accelerates, also the average velocity at the end of the data set was found, lastly all these values were used to solve for the time constant parameter.

To ensure the accuracy of our model we had to use a gaussian window function, which we had to edit a number of times, ultimately settling on a smaller window of 3 data points that would run 500 times. This resulted in a very smooth model, however our accuracy was improved due to a reduction in the skew of our velocity data near the start of the acceleration, this improved accuracy is evident in our calculated SSE value decreasing from 0.7842 - 0.5053 between data sets to 0.5828 - 0.5080 between data sets. Another change we made was to the method used to calculate initial velocity which was usually inaccurate due to slight error in the acceleration starting time, to account for this we included the data from 0.04 seconds after our calculated acceleration start time to account for this error, which resulted in a far lower percent error in our initial velocity (going from 40 – 20% between data sets to 15 –10 %). Lastly, we edited the window size used to calculate the acceleration at any point in the data when finding the acceleration start time to be larger. This improved the run time of our program by making it run less calculations when finding the acceleration as well as granting us a lower error in the calculated acceleration start time by counteracting the increased noise in our data by running the smoothing algorithm less times. This resulted in lowering our error from (1.31 – 5.64 % between tests to 1.45 – 0.88%), indicating an improvement in our algorithm's accuracy.

Parameter Identification Procedure

The various parameters were found by first importing the given velocity/ time data from each test. Then, the first function smooths the data by first removing any outlier or non-numerical data points, then running a moving window averaging algorithm multiple times resulting in a far smoother data set which is later used to calculate the other parameters.

After this, the first parameter to be calculated is the acceleration start time, which is calculated by comparing the relative change in velocity from data point to data point and finding the point and corresponding time in which this change exceeds a given threshold (in this case 4 m/s^2). Once this time is found, it is used in the next function to find the initial velocity by averaging the velocity data from the start of the data set until the acceleration start time, next the final velocity is found by averaging the velocity in the last 5 seconds of the data set.

Next, the time constant is calculated by using the formula describing the dynamic system given in the initial client memo (M0), using this formula the time parameter is algebraically solved for. Lastly, the previously calculated parameters are plugged into the algebraic equation calculating the value for the time constant.

Results

For details on the results of our algorithm please refer to Figures 1-3, as well as the interpretation section of the document.

Interpretation

The experiment had a great amount of variation because of obstacles such as noise, vibrations and inconsistent testing conditions which resulted in inaccuracy in the data. This would impact the model as the model was dependent and derived from the given data. A critical the initial speed values for both the benchmark data and our calculated. Since both values are extremely small any variation of more than 0.05 m/s would cause a huge error percentage. As an example, analyzing table 1 the calculated value for the Midsize Sudan initial velocity was –0.25 m/s, while the actual benchmark data value was –0.22 m/s. The difference was that of 0.03 m/s and still gave an error percentage of 15.79% being the highest of all the values. Moreover, by having a slightly inaccurate initial velocity the accuracy of the time constant is also slightly inaccurate as seen on all tables (1-3), the time constant error percentages range from 3.93% to 7.95%, this is because the formula for the time constant calculation used in the algorithm includes the initial velocity value. Thus, showing how any variation in the data would cause the model to have a slight discrepancy with the benchmark data, and how detrimental the errors in the actual experiment are to the algorithm's calculated parameter values.

The adaptive cruise control system still performs within boundaries of the benchmark data. Although the algorithm does have a slight shortcoming on one of the parameter calculations the system model can be concluded to be accurate. The data and graphical values both indicate that the performance of the ACC system, even with the change in the tire material, is still on par with the requirements imposed by the previous testing results.

Our next steps include reflecting on the inaccuracies of the algorithm, if a larger time frame was available then some possible changes would include a window of around 0.5 seconds when completing the averaging for the identification of the initial velocity value. Although we have consistently tested different window sizes this could be a more specific approach to the one that we have been taking. A higher number of windows would be tested to try and reduce the number of possible inaccuracies in the calculation. By identifying the window that produces the least amount of error overall, the algorithm will hopefully have reduced the error percentage in this shortcoming.

References

No external sources were used in the creation of the technical brief, please refer to the references listed on the M4 Answer sheet for any sources used during the creation of our algorithm.

Appendix: Figures and Tables

Vehicle	Tire Type	M4 Algorithm Parameters			
		Start time [s]	**Time constant [s]**	**Initial speed [m/s]**	**Final speed [m/s]**
Compact Hatchback	Summer	8.10	2.78	-0.03	25.06
Compact Hatchback	All-Season	7.76	1.87	0.00	24.92
Compact Hatchback	Winter	7.74	1.57	0.00	24.67
Midsize Sedan	Summer	8.28	2.80	-0.04	24.94
Midsize Sedan	All-Season	6.19	2.01	0.00	24.89
Midsize Sedan	Winter	7.81	1.56	0.00	24.95

Large SUV	Summer	7.65	3.77	-0.03	23.64
Large SUV	All-Season	7.27	2.46	-0.02	24.03
Large SUV	Winter	7.45	3.07	-0.02	24.06

Figure 1. Results for parameters from all tests using M4 finalized algorithm

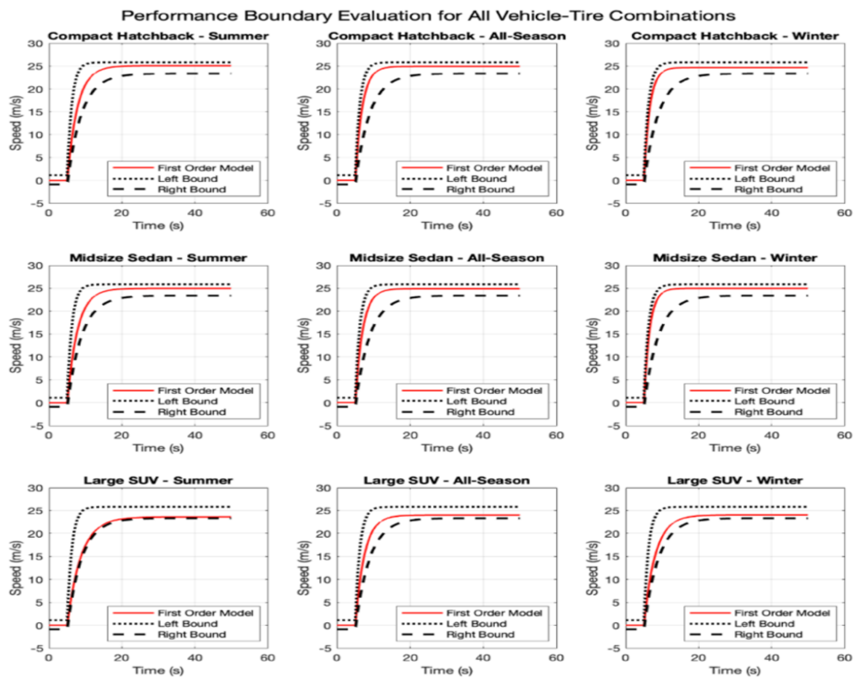

Figure 2. Performance Boundary Evaluation for All Vehicle-Tire Combinations

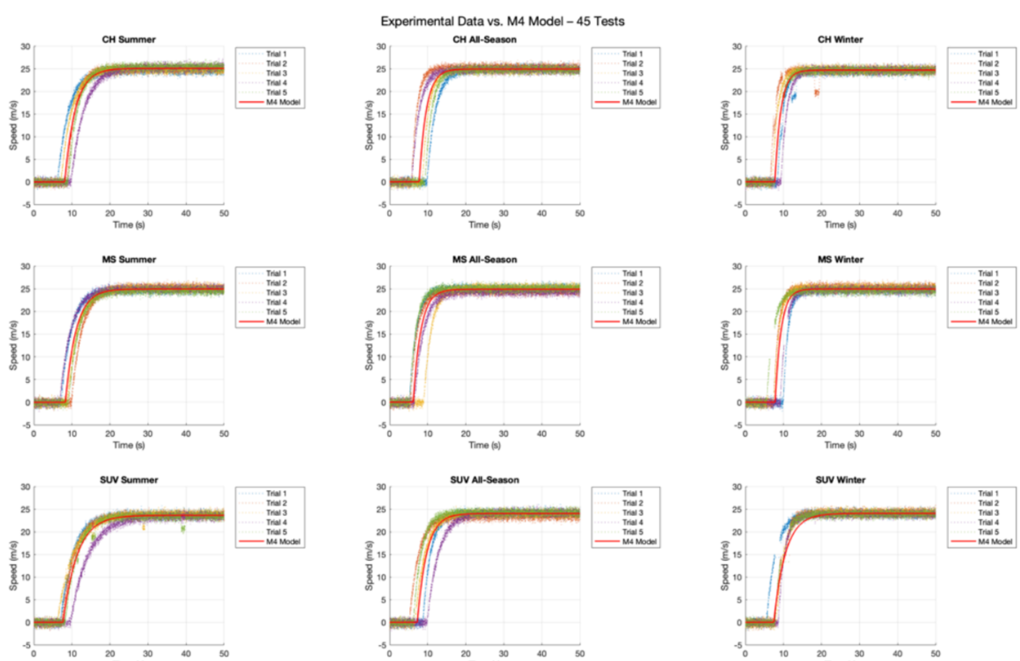

Figure 3. Comparison of Experimental Data with Algorithm