Simulation of the Half-Car Model

Connor McPartland

Project Description

The half-car model is a simplified way to model a car. The assumption is that the road surface is uniform in the direction normal to the length of the car. Under this assumption, whatever happens to the right side of the vehicle will happen to the left side of the vehicle (if the right set of tire goes over a curb, then so does the left set of tires). Only one side of the vehicle needs to be modeled to recreate this situation. Another assumption in this model is that the car has constant velocity in the *x*-direction. The goal of the simulation is to plot the motion of the vehicle as it travels over a chosen road. The equations that govern the motion of the vehicle need to solved in order to do this, so it is necessary to find the forces acting on the different parts of the vehicle. Once they are known, the governing equations can be formed and solved using a differential equation solving-algorithm. With the model working, the vehicle's motion will be shown in the simulation as it travels over a chosen road surface.

The model consists of few parts: a vehicle body, a front suspension, a rear suspension, and a front and rear tire (this model assumes there is no metal wheel or 'rim', only an inflated rubber tire that extends radially outward from the end of the suspension spring). The suspension is a spring and damper system. The tires will be modeled in the same way, since the rubber sidewall of the tire acts like a spring and the air pressure in the tire acts like a damper. The vehicle body is subject to forces coming from its interaction with the suspension system, and the tires are subject to forces coming from both the interaction with the suspension system and interaction with the road surface. All components are subject to gravity.

To fully describe the system, 8 state-variables are necessary:

θ, $\dot{\theta}$	The angle of the vehicle body with respect to a flat road surface and its time-derivative
y_{com}, $\dot{y}_{com}$	The height of the center of mass of the body and its time-derivative
y_{rt}, $\dot{y}_{rt}$	The height of the rear tire and its time-derivative
y_{ft}, $\dot{y}_{ft}$	The height of the front tire and its time-derivative

The governing equations of the *y*-coordinate variables can be obtained by summing the forces on that vehicle part in the *y*-direction and setting them equal to $m_y\ddot{y}$. The governing equation of θ can be obtained by summing the torques on the vehicle body about the center of mass and setting them equal to $I_{car}\ddot{\theta}$, where I_{car} is the moment of inertia of the body about the center of mass. The equations are listed below:

$$m_{car}\ddot{y}_{com} = -m_{car}g - k_{fs}\delta_{fs} - b_{fs}\dot{\delta}_{fs} - k_{rs}\delta_{rs} - b_{rs}\dot{\delta}_{rs}$$

$$I_{car}\ddot{\theta} = -L_f\cos(\theta)\left[k_{fs}\delta_{fs} + b_{ft}\dot{\delta}_{fs}\right] + L_r\cos(\theta)\left[k_{rs}\delta_{rs} + b_{rs}\dot{\delta}_{rs}\right]$$

$$m_{ft}\ddot{y}_{ft} = -m_{ft}g - k_{ft}\delta_{ft} - b_{ft}\dot{\delta}_{ft} + k_{fs}\delta_{fs} + b_{fs}\dot{\delta}_{fs}$$

$$m_{rt}\ddot{y}_{rt} = -m_{rt}g - k_{rt}\delta_{rt} - b_{rt}\dot{\delta}_{rt} + k_{rs}\delta_{rs} + b_{rs}\dot{\delta}_{rs}$$

[*k* refers to spring stiffness, *b* refers to damping coefficient, δ refers to spring compression/elongation, L_f and L_r is the distance from center of mass to the front of the car and to the rear of the car, respectively, subscript *fs*/*rs* refer to front suspension/ rear suspension, and *ft/rt* refer to front tire/rear tire, $L = L_f + L_r$, the full length of the car.]

$$\delta_{ft} = y_{ft} - y_{Road}(x+L) - fl_{ft}$$

$$\delta_{rt} = y_{rt} - y_{Road}(x) - fl_{rt}$$

$$\delta_{fs} = y_{com} + L_f \sin(\theta) - y_{ft} - fl_{fs}$$

$$\delta_{rs} = y_{com} - L_r \sin(\theta) - y_{rt} - fl_{rs}$$

$$I_{car} = \frac{1}{3L}\left(L_r L_f^2 + L_f L_r^2\right) m_{car}$$

Simulation

Computations

The simulation solves the above second-order differential equations at discrete times using the 4th order Runge-Kutta algorithm. The initial conditions of the state-variables for this system are obtained by assuming the vehicle is resting on the road surface and is at equilibrium. Since the vehicle is in equilibrium, there are no forces are acting on the vehicle. The left-hand sides of the governing equations above are set to zero, which gives a system of equations. To make the system linear, the height of the front of the car and the height of the rear of the car are considered to be state variables instead of θ and y_{com}. Once the two heights are known, the initial conditions for θ and y_{com} are calculated from that data. The simulation solves this linear system of equations using linear algebra to get all necessary initial conditions. The initial conditions are printed to the messages box when the simulation is started.

At each discrete time step, the δ values are calculated from the values of the state-variables at the previous time step. The δ_s values are dependent on the free-length of the suspension springs and the difference between the height of the front or rear of the car (calculated from θ and y_{com}) and the height of front or rear tire. The δ_t values are trickier since the tire should never be pulled on by the road – the tire is only compressed. To handle this, the value of δ_t is clipped to have a maximum of 0 (there is no clipping in the other direction). Additionally, if the tire leaves the ground, the value of $\dot{\delta}_t$ is set to zero since the tire and the road are no longer interacting. With the δ values known, the forces and torques can be calculated from the governing equations. The Runge-Kutta4 approximately solves these equations at each time step.

Plotting

The plotting is carried out every 8 time steps to speed up plotting. The plotting includes the car and the road on a 2D plot. On another graph, the center of mass of the vehicle is optionally plotted as a function of time to show the frequency response of the vehicle to a bumpy road (there is a check-box to turn this plot on or off). The positions of the front of the car, the rear of the car, the center of mass, the front tire, and the rear tire are printed out to text boxes. The top plot will show some stretch of the chosen road. The car will travel from the left of the plot to the right of the plot. The limits will then shift over to show the next stretch of road, and this will repeat until the simulation stops.

Interface

The user can enter all the governing characteristics of the car:

- the spring stiffness and damping coefficients of the front and rear suspension;
- the spring stiffness and damping coefficients of each tire;
- the free-lengths of the suspension spring and the radii of the tire;
- the mass of the vehicle and the mass of each tire;
- the distances from the center of mass to the front and the rear of the vehicle;
- the velocity in the *x*-direction.

The user can choose one of three road surfaces:

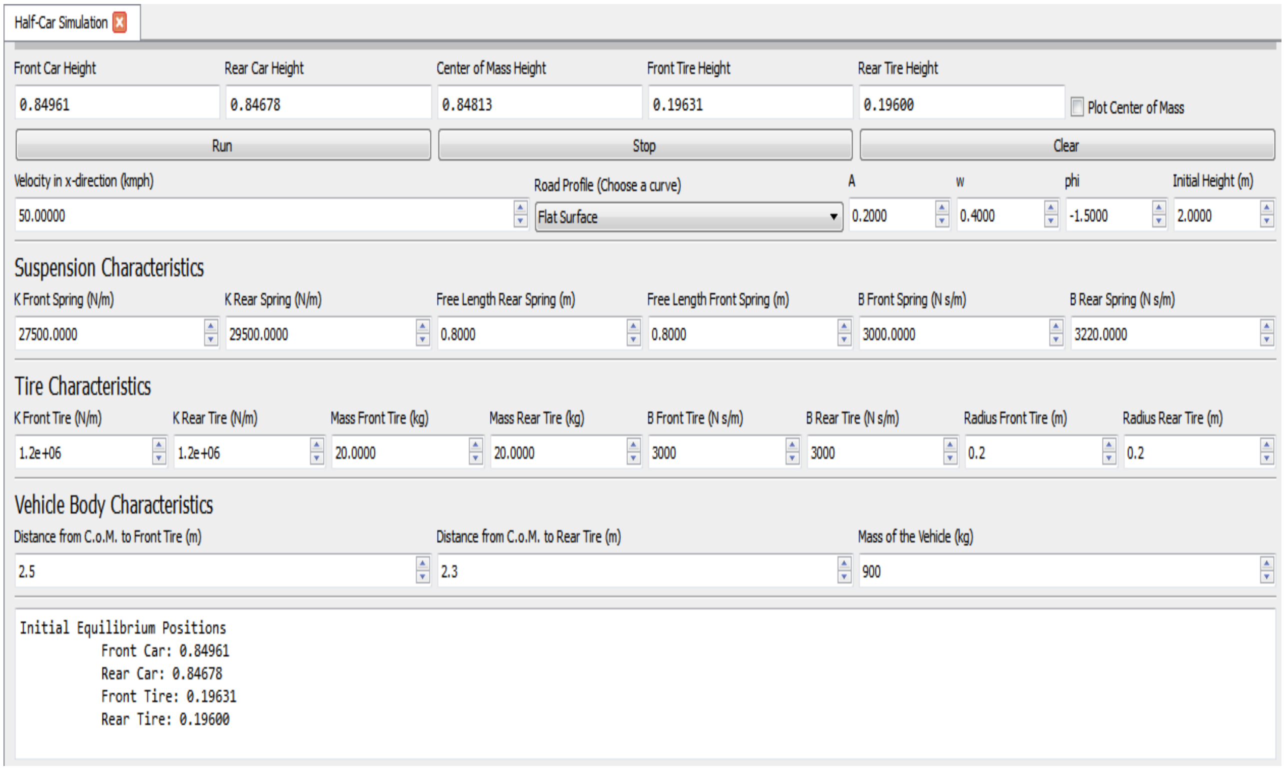
- flat surface;

- sinusoidal (user can choose amplitude, frequency, and phase shift);
- square wave (user can choose amplitude and a pseudo-frequency).

The simulation will generate the flat surface and the sinusoidal roads very easily using arrays of zeros in the first case and the numpy library in the other. The square wave is created in the program by taking the value in the 10's column of the current *x*- coordinate of the rear tire and evaluating whether it is even or odd – if it is even, the road height is 0 m, and if it is odd the road height is the user-chosen amplitude. The frequency of the square wave is simulated by multiplying the current *x*-coordinate by the user-chosen frequency value so that the frequency of the square wave is somewhat proportional to the chosen frequency.

Test Calculation

A test for the simulation is to analytically solve for the equilibrium conditions in the case of the car resting on a flat surface and then simulate this situation. To do this, the car can be dropped from a height onto a flat surface (there is an input box to set the car's initial height above the ground – the value entered into this box is simply added to the *y*-coordinates of the center of mass and tires). The car will hit the ground and the body will show damped oscillation until it reaches equilibrium. The final positions of the front of the car, the rear of the car, the front tire, and the rear tire are printed to text boxes, so they can be checked against those printed out in the messages box at the beginning of the simulation (these values are the analytic solutions that come directly from the equilibrium equations). The results of the test calculation are easiest to interpret when it is run in the simulation, but here are the results for TestCase parameter settings:



The heights of the vehicle components correctly settle into the equilibrium positions given by the analytic solution.

<u>Running the Simulation</u>

There is a parameter file called TestCase that includes the proper settings to carry out the test calculation. Otherwise, the parameter file called Basic includes the basic settings needed to run the program and show the car going over a sensible road. The spring constants, damping coefficients, and masses are realistic and come from two sources: one is a ‘handbook’ on car suspensions, and the other is an explanation of the half-car model.

References

- Dixon, John. The Shock Absorber Handbook.
- Vehicle Suspension Modeling. Kypuros, Javier A., http://mece.utpa.edu/Kypuros/teaching/mece-4305/notes/VehicleSuspensionModelingNotes.pdf
- Bus Suspension Modeling in Simulink. http://www.library.cmu.edu/ctms/ctms/simulink/examples/susp/suspsim.htm