Simulation-based least squares parameter estimation for a low-order model of car dynamics

Aniruddha Perumalla[†]
Pennsylvania State University, University Park, PA, 16801

I. Nomenclature

x	=	Longitudinal position
v_x	=	Longitudinal speed in the body frame
y	=	Lateral position
v_y	=	Lateral speed in the body frame
ψ	=	Yaw angle
r	=	Yaw rate
m	=	Mass
I_{zz}	=	Yaw moment of inertia
a	=	Longitudinal distance from center of gravity (CG) to front axle
b	=	Longitudinal distance from center of gravity (CG) to rear axle
l	=	Total distance rear axle to front axle
$C_{\alpha f}$	=	Front tire cornering stiffness
$C_{\alpha r}$	=	Rear tire cornering stiffness
δ_f	=	Front wheel steering angle
α_f	=	Front tire sideslip angle
α_r	=	Rear tire sideslip angle

II. Introduction

In recent years, gaming applications have proven to be a strong motivator for advances in simulation. In particular, gaming engines have evolved to reproduce motion of real objects in a physically realistic manner and at very high frame rates. Such engines have provided a valuable platform for critical engineering applications to simulate physical experiments in cases where real-life experimentation may be infeasible due to physical fragility or cost. This project considers one such application, namely the ordinary least-squares estimation of parameters of a low-order, commonly-used vehicle dynamics model via data from a photorealistic, physically accurate simulator, namely Microsoft Airsim. The results of this estimation were used to judge the accuracy of this model in the context of a real SUV. All code, results, and references used in this work have been preserved in a GitHub repository.

A. Bicycle Model

The "bicycle model" is a well-known simple, low-order model of the kinematics and yaw dynamics of a moving vehicle with a front and rear axle*. The model relies on the following assumptions [1]:

- Only the front wheel(s) steer the vehicle.
- The vehicle's body-frame longitudinal (forward) speed v_x is constant.
- Both sets of tires roll without slipping.
- Aerodynamic effects are negligible.
- Small-angle approximations apply.

[†]Graduate Research Assistant, Department of Aerospace Engineering, amp7226@psu.edu

*The name "bicycle" is a misnomer and purely due to convenience since the model assumes that the vehicle has a single front wheel and single rear wheel. It does not actually describe the motion of a bicycle.

In addition, the model assumes that the force acting on each tire/wheel is a linear function of tire sideslip angle α (1, 2) and that the sideslip angle itself may be expressed in terms of the vehicle velocity components v_x and v_y and the longitudinal dimensions a and b (3, 4).

$$F_f = C_{\alpha f}\alpha_f \tag{1}$$

$$F_r = C_{\alpha r}\alpha_r \tag{2}$$

$$\alpha_f \approx \frac{v_y + ar}{U} - \delta_f \tag{3}$$

$$\alpha_r \approx \frac{v_y - br}{U} \tag{4}$$

For a given vehicle, the model can be linearized around zero yaw to present the state-space model in (5) [1] [2].

$$\begin{bmatrix} \dot{v}_y \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{C_{\alpha f}+C_{\alpha r}}{mv_x} & \frac{aC_{\alpha f}-bC_{\alpha r}}{mv_x} - v_x \\ \frac{aC_{\alpha f}-bC_{\alpha r}}{I_{zz}v_x} & \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_{zz}v_x} \end{bmatrix} \begin{bmatrix} v_y \\ r \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha f}}{m} \\ \frac{aC_{\alpha f}}{I_{zz}} \end{bmatrix} \delta_f \tag{5}$$

B. AirSim

AirSim [3] is a high-fidelity, open-source vehicle simulator developed in 2017 by Microsoft. It is built on the Unreal Engine gaming engine and is capable of simulating vehicles like drones and cars in photorealistic environments. AirSim also implements accurate models of sensors like gyroscopes, depth and infrared cameras, and LIDAR to gather noisy estimates of the state (kinematics and dynamics) of the vehicles. For the simulation accomplished in this project, AirSim's default car model was driven in an environment mimicking the soccer pitch on a Microsoft campus (the `MSBuild2018.sh` release), as depicted in 1. Originally, this project aimed to exploit AirSim's capability to simulate many different weather conditions. However, it was discovered that by default, the weather does not affect the physics unless further enhancements to the environment beyond the default are made; this unfortunately required more time than was available.

Fig. 1 A frame of the simulated car driving in the soccer pitch environment.

III. Method

A. Collection of Data

AirSim provides an API to send control commands, namely throttle and steering angle (both dimensionless), to the simulated car. The throttle command supplied to the car must be in the range [0, 1] and the steering command must be in the range [-1, 1].

Since the bicycle model assumes constant longitudinal body-frame velocity, the throttle command supplied to the car for each simulation was a step input with a magnitude sampled from the uniform distribution in the range [0.2, 0.35]. To provide enough data about the response of the bicycle model states to the steering angle, a sinusoidal steering command was provided to the car. The magnitude of the sinusoidal steering command was sampled from the uniform distribution in the range [-0.15, 0.15], which produces a yaw response mostly within the range over which the small-angle approximation applies. The frequency of the sinsusoidal steering command was sampled from the uniform distribution in the range [0 Hz, 1 Hz]. The state responses to 10 such throttle and steering commands are displayed in 2.

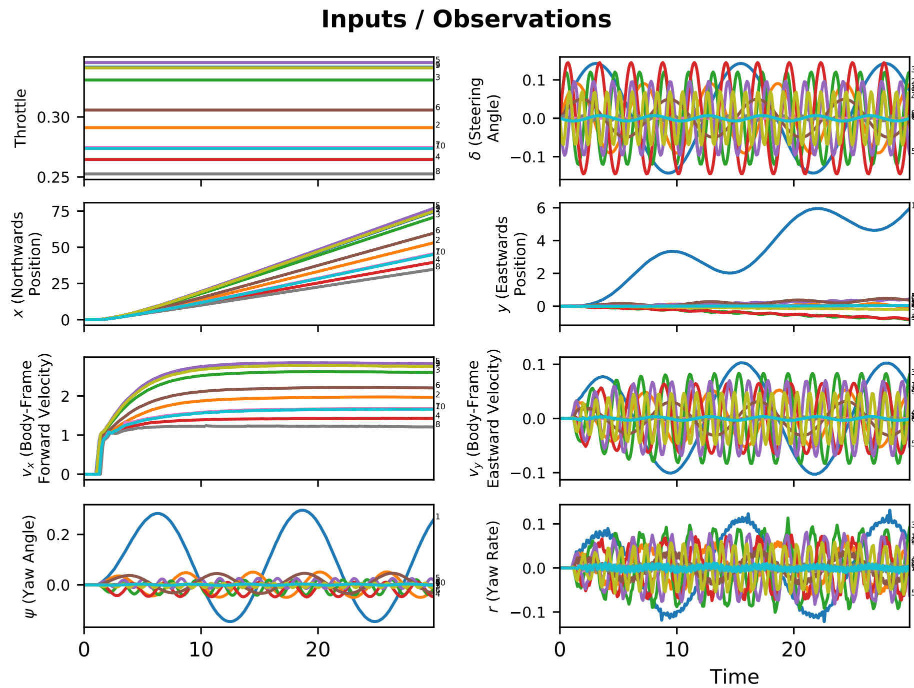

Fig. 2 The state responses to a step input in throttle and sinusoidal input in steering of 10 of the 1000 simulations. The index of each simulation is displayed to the right of each sub-plot for convenience (the 8th simulation is the bottom-most in the top-left three sub-plots).

The API was used to simulate the motion of the car one thousand times, and the entire vehicle state ($[x, y, v_x, v_y, \psi, r]^\intercal$) was measured every 0.1 s (10 Hz) for 30 s[†]. Videos of one such simulation may be viewed at the following links: back view video; side-front view video. The bicycle model assumes constant longitudinal speed; since the AirSim vehicle requires a few seconds for the longitudinal speed to rise to the constant speed commanded by the throttle, the data corresponding to this delay in each simulation was discarded. For example, in 2, only the data from 5 s and later (i.e., the last 25 s) were used for the estimation in simulation 8.

The first five hundred of the thousand simulations were used for ordinary least-squares estimation of inertial properties. The last five hundred were used for ordinary least-squares estimation of cornering stiffnesses. In both categories of least-squares estimations, a first-order forward difference operation was used to approximate the rate of change of each state variable from the discrete measurements: $\dot{x}_k \approx \frac{x_{k+1}-x_k}{T}$. The ordinary least squares estimator for this formulation is $\hat{\mathbf{q}} = (\mathbf{H}^\intercal\mathbf{H})^{-1}\mathbf{H}^\intercal\mathbf{y}$, where $\mathbf{H}$ is the feature matrix, $\mathbf{y}$ is the response vector (both populated from measurements of system state), and $\mathbf{q}$ is the parameter vector.

B. Least-Squares Estimation of Vehicle Inertial Properties

The bicycle model displayed in (5) may be re-expressed as shown in (6) to permit the inertial parameters of the vehicle, the mass m and yaw moment of inertia I_{zz}, to be identified via least-squared estimation[‡].

[†] AirSim's API provides inertial frame velocity rather than body-frame velocity, so body-frame velocity at each timestep was calculated by rotating the inertial velocity vector at the timestep through multiplication with the direction cosine matrix of the yaw angle about the downwards axis.

[‡] Note that the estimation actually identifies the inverse of each of these parameters; the resultant values are inverted to find the inertial properties.

$$\begin{bmatrix} \dot{v}_y + v_x r \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{2C_\alpha}{v_x} v_y - C_\alpha \delta_f & 0 \\ 0 & \frac{l^2 C_\alpha}{2v_x} r - \frac{l C_\alpha}{2} \delta_f \end{bmatrix} \begin{bmatrix} \frac{1}{m} \\ \frac{1}{I_{zz}} \end{bmatrix} \tag{6}$$

The estimation from measurements of lateral body-frame velocity v_y and yaw rate r, is conditional on the assumption that the remaining model parameters, namely the tire cornering stiffnesses $C_{\alpha f}$ and $C_{\alpha r}$ and longitudinal dimensions a and b, are known.

In particular, the center of gravity was assumed to be halfway between the front and rear axles, so that $a = b = \frac{l}{2}$, where l was taken to be 5 m from the AirSim car model. The front and rear cornering stiffnesses were assumed to have the same value, so that $C_{\alpha f} = C_{\alpha r} = C_\alpha$. The value of C_α was estimated from results found by Stankiewicz; for a truck driving with a speed in the range of $13.4\,\mathrm{m\,s^{-1}}$ to $26.8\,\mathrm{m\,s^{-1}}$, the cornering stiffnesses using a bicycle model with a linear tire force assumption were determined to be $-120\,000\,\mathrm{N\,rad^{-1}}$ each [1]. In the AirSim car simulations, the average longitudinal speed attained was roughly $1.5\,\mathrm{m\,s^{-1}}$. Therefore, due to the simulation speed being roughly one order of magnitude less than the speeds in [1], the cornering stiffnesses for the estimation were assumed to be $-10\,000\,\mathrm{N\,rad^{-1}}$ each, roughly one order of magnitude less than those found by Stankiewicz [1].

C. Least-Squares Estimation of Vehicle Tire Cornering Stiffnesses

The bicycle model (5) may also be re-expressed as shown in (7) to permit the tire cornering stiffnesses to be identified via least-squared estimation:

$$\begin{bmatrix} \dot{v}_y + v_x r \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{1}{m v_x} v_y + \frac{a}{m v_x} r - \frac{1}{m} \delta_f & \frac{1}{m v_x} v_y - \frac{b}{m v_x} r \\ \frac{a}{I_{zz} v_x} v_y + \frac{a^2}{I_{zz} v_x} r - \frac{a}{I_{zz}} \delta_f & -\frac{b}{I_{zz} v_x} v_y + \frac{b^2}{I_{zz} v_x} r \end{bmatrix} \begin{bmatrix} C_{\alpha f} \\ C_{\alpha r} \end{bmatrix} \tag{7}$$

This estimation, also from measurements of lateral body-frame velocity v_y and yaw rate r, is conditional on the assumption that the remaining model parameters, namely the inertial properties and the longitudinal dimensions a and b, are known.

According to the physics engine file for the vehicle model used in AirSim, the mass of the vehicle is 2500 kg; this mass was assumed for this estimation. In 1999, Heydinger et al. developed a database of physical properties of various commercial car, truck, van, and SUV models [4]. Based on the AirSim vehicle's shape and mass, the 1998 Chevrolet Tahoe was identified to be the vehicle in the database most similar to the AirSim vehicle. The following parameters taken from the database entry for the 1998 Tahoe were assumed for the estimation: yaw moment of inertia I_{zz} of $5445\,\mathrm{kg\,m^2}$, front axle longitudinal distance a of 1.441 m, and rear axle longitudinal distance b of 1.536 m.

IV. Results

A. Least-Squares Estimation of Vehicle Inertial Properties

The least-squares estimation was carried out to identify the mass and yaw moment of inertia. Of the 500 simulations run to collect data for this estimation, three simulations did not meet the constant longitudinal speed requirement and were hence discarded. Figure 3 displays a histogram of the estimated values, and figure 4 displays a histogram of the RMS error in the estimation, i.e. the RMS error between the left side of equation (6) and the right side of equation (6) over all timesteps.

Comparing the estimated mass to the true value of 2500 kg as obtained from the AirSim engine data, the estimation generally meets the order of magnitude of the mass. However, most of the estimated masses are overestimates of the true mass.

The true value of the yaw moment of inertia is not available in the AirSim engine data, but comparing it to the yaw moments of inertia for SUVs like the 1998 Chevrolet Tahoe or the 1998 Toyota 4Runner from the database in [4] shows that the estimated values are overestimates by roughly one-half to one order of magnitude.

The errors in the estimation of the inertial properties most probably lie in the assumptions regarding the center of gravity of the vehicle and the stiffness coefficients, as detailed in the previous section. Judging from the model of the car, the true center of gravity of the vehicle is most probably located farther towards the rear axle than the front axle. It appears that the assumed stiffness coefficients may have resulted in the force acting on the tires being underestimated.

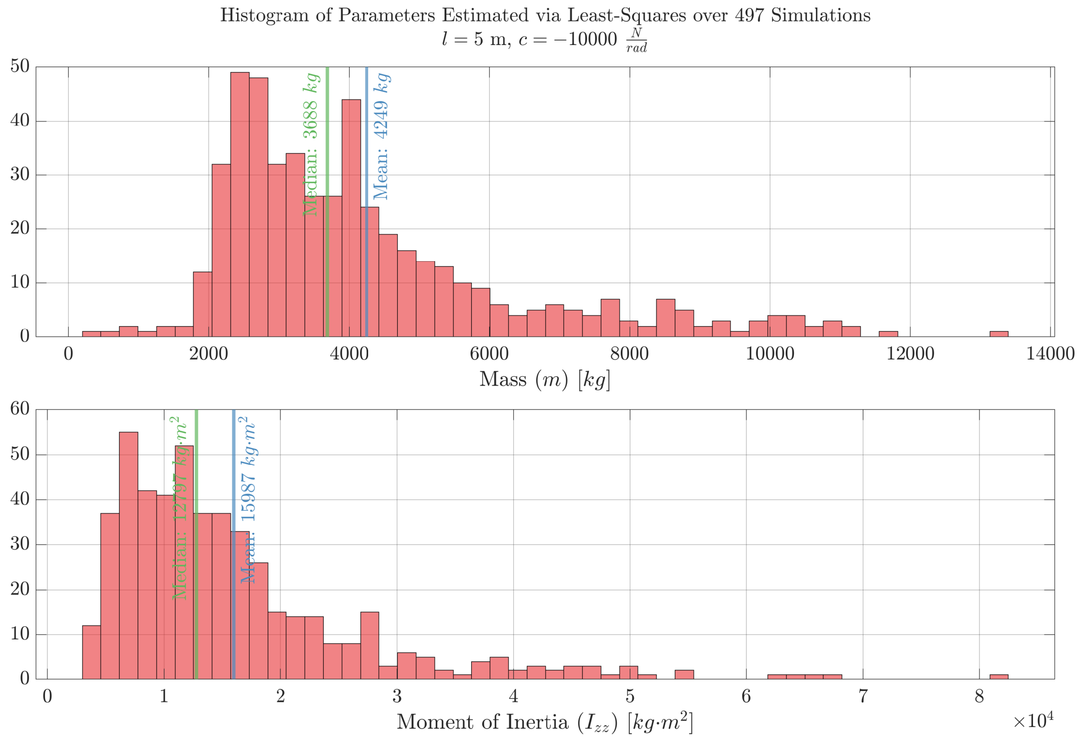

Fig. 3 Histogram of estimated values of mass m and yaw moment of inertia I_{zz} over 500 simulations. The mean and median estimated parameters are overlaid.

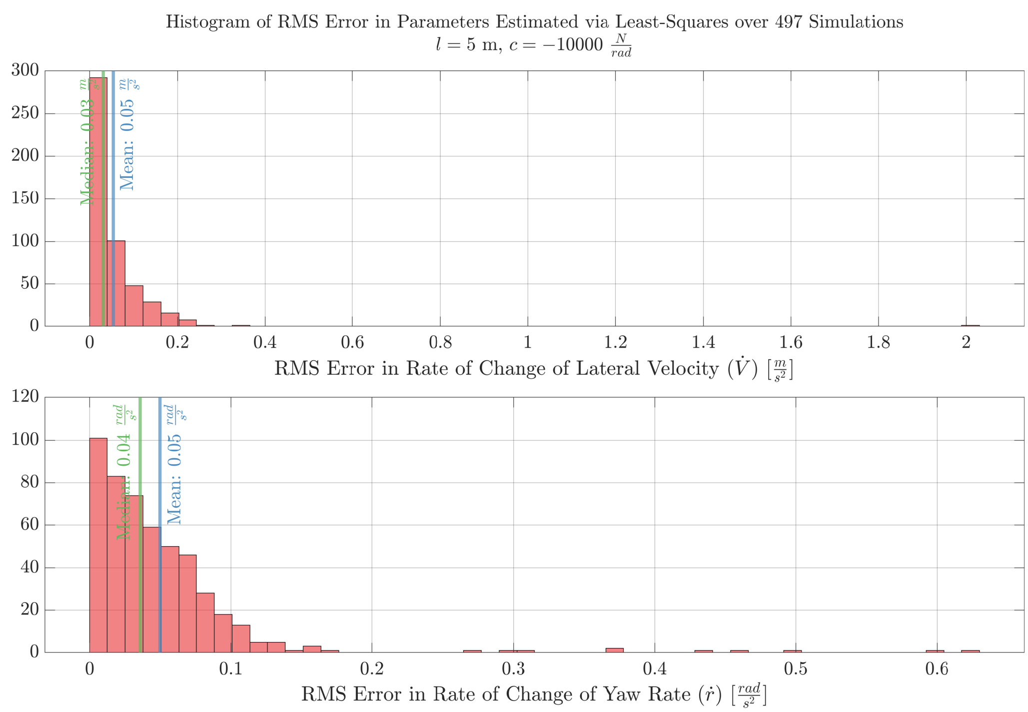

Fig. 4 Histogram of RMS error produced when substituting the estimated values shown in 3 in the model. The mean and median estimated RMS errors are overlaid.

B. Least-Squares Estimation of Vehicle Tire Cornering Stiffnesses

The least-squares estimation was carried out to identify the front and rear tire cornering stiffnesses. Of the 500 simulations run to collect data for this estimation, one simulation did not meet the constant longitudinal speed requirement and was hence discarded. Figure 5 displays a histogram of the estimated values, and figure 6 displays a histogram of the RMS error in the estimation, i.e. the RMS error between the left side of equation (7) and the right side of equation (7) over all timesteps.

Since the AirSim car model does not assume that tire force is a linear function of tire sideslip angle as the bicycle model does, the engine data for the model does not contain any data about the cornering stiffnesses. However, real tire cornering stiffnesses as used in the bicycle model have often been determined to have magnitude of at least roughly $-10\,000\,\mathrm{N\,rad^{-1}}$ [5], ranging up until roughly $-200\,000\,\mathrm{N\,rad^{-1}}$ for heavy-duty tires used for large vehicles like trailers [6]. For example, Sienel found a vehicle moving on dry road at $50\,\mathrm{km\,s^{-1}}$ to have a cornering stiffness in the range of $-10\,000\,\mathrm{N\,rad^{-1}}$ and $-70\,000\,\mathrm{N\,rad^{-1}}$ from low to high sideslip angle, respectively. Therefore, we can conclude that the estimated stiffnesses are of a reasonable magnitude, given that the details of the true tire model used in the simulation were unknown.

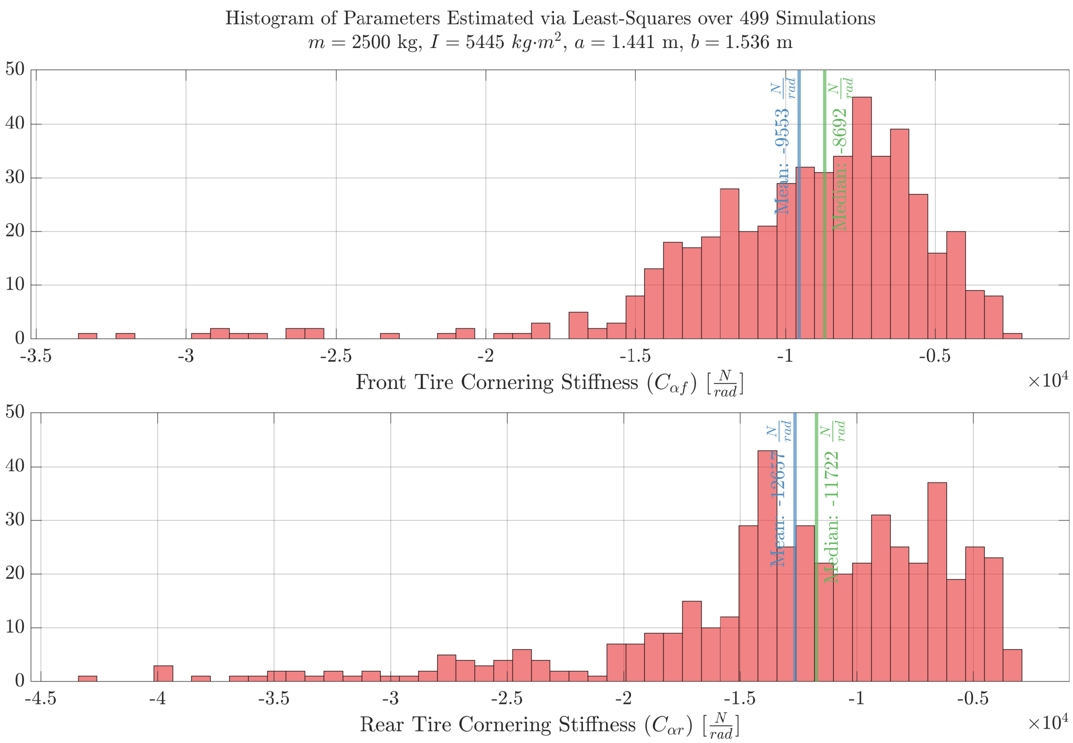

Fig. 5 Histogram of estimated values of front tire cornering stiffness $C_{\alpha f}$ and rear tire cornering stiffness $C_{\alpha r}$. The mean and median estimated parameters are overlaid.

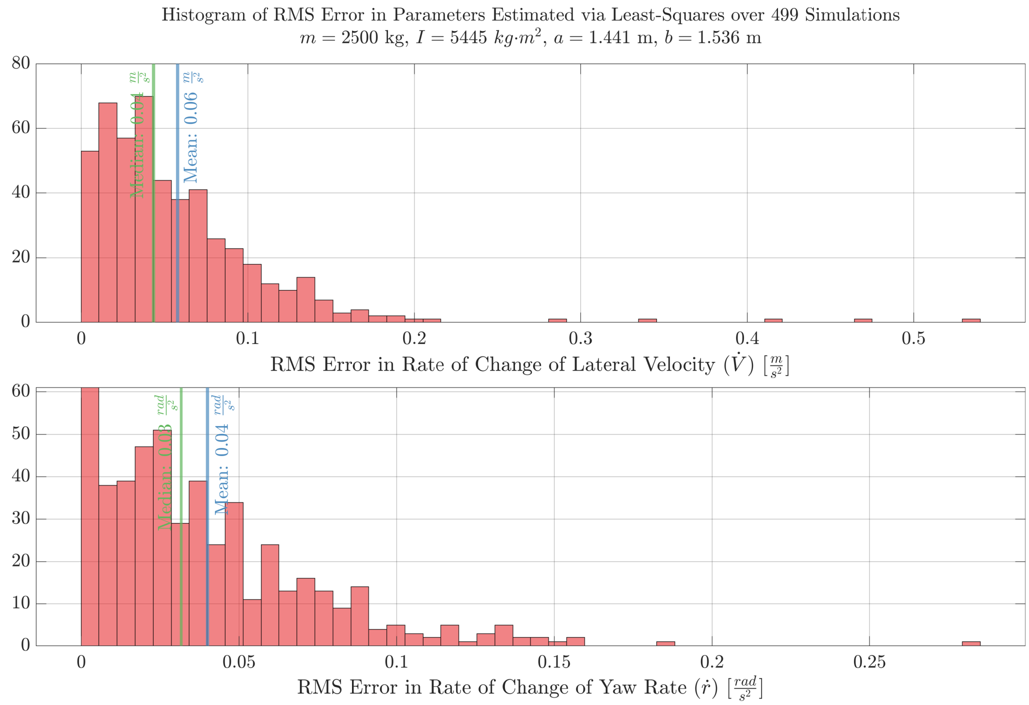

Fig. 6 Histogram of RMS error produced when substituting the estimated values shown in 5 in the model. The mean and median estimated RMS errors are overlaid.

V. Conclusion

Microsoft AirSim, a high-fidelity flight and vehicle simulator built upon the Unreal Engine gaming engine, was used to collect data about the motion of a car resembling an SUV in a simulated photorealistic and physically accurate environment. The data collected through this simulation was used to judge the accuracy of a low-order bicycle model as applied to the car through ordinary least-squares estimation of the parameters in the model. First, least-squares estimation found that the estimated mass of the vehicle was of a similar magnitude to the true magnitude, and that the estimated yaw moment of inertia was approximately double or triple that of the approximate true yaw moment of inertia. Next, least-squares estimation of the tire cornering stiffnesses found that the true values were of a reasonable magnitude compared to the experimentally determined stiffnesses of real tires. Overall, the use of the bicycle model was found to be approximately accurate to describe the motion of the simulated car.

References

[1] Stankiewicz, P. G., "Vehicle Control for Collision Avoidance and Rollover Prevention using the Zero-moment Point," 2015.

[2] Limroth, J. C., and Kurfess, T., "Real-time vehicle parameter estimation and equivalent moment electronic stability control," *International Journal of Vehicle Design*, Vol. 68, No. 1/2/3, 2015, p. 221. doi:10.1504/ijvd.2015.071072, URL `https://doi.org/10.1504/ijvd.2015.071072`.

[3] Shah, S., Dey, D., Lovett, C., and Kapoor, A., "AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles," *Field and Service Robotics*, 2017. URL `https://arxiv.org/abs/1705.05065`.

[4] Heydinger, G. J., Bixel, R. A., Garrott, W. R., Pyne, M., Howe, J. G., and Guenther, D. A., "Measured Vehicle Inertial Parameters-NHTSA's Data Through November 1998," *SAE Technical Paper Series*, SAE International, 1999. doi:10.4271/1999-01-1336, URL `https://doi.org/10.4271/1999-01-1336`.

[5] Vorotović, G. S., Rakicević, B. B., Mitić, S., and Stamenković, D. D., "Determination of Cornering Stiffness Through Integration of A Mathematical Model and Real Vehicle Exploitation Parameters," *FME Transactions*, Vol. 41, 2013, pp. 66–71.

[6] McDowell, J., "Comparison of Modern Controls and Reinforcement Learning for Robust Control of Autonomously Backing Up Tractor-Trailers to Loading Docks," 2019.