Vehicle Dynamics Modeling and Simulation Including Empirical Nonlinear Tire Elements with Artificial Intelligence Using Native Code MotionSim Integrated into Simulink Environment

John Dye and Hamid Lankarani

June 4, 2024

The 7th International Conference on Multibody System Dynamics
June 9-13, 2024, Madison, WI, USA

Vehicle Dynamics Modeling and Simulation Including Empirical Nonlinear Tire Elements with Artificial Intelligence using Native Code *MotionSim* Integrated into *Simulink* Environment

John Dye*, Hamid Lankarani#

*Department of Mechanical Engineering
Wichita State University
Wichita, KS 67260, USA
Jrdye10@gmail.com

#Department of Mechanical Engineering
Wichita State University
Wichita, KS 67260, USA
hamid.lankarani@wichita.edu

Abstract

Simulation of multibody systems often requires the use of commercial software's for academic research or commercial development. Vehicle dynamics simulations are often generalized as the exact nature of roads, environment conditions, and initial condition are extremely variable. To increase fidelity of these models and to better understand the transient nature of a vehicle, a multibody model will be used to simulate scenarios that generalize the vehicles intended use. Successful generalization of vehicle dynamics often requires the inclusion of multiple physical elements outside of common rigid body multibody systems [1]. Commonly these elements are the tires, dampers or bushings that introduce non-linear behavior into the multibody system. Multiple tire models have been suggested with many utilizing empirical methods such as the magic formula tire model [2]. Many non-empirical models exist as each are utilized in the scenarios where the appropriate tire dynamics are represented, however most of these are computationally expensive.

MotionSim is a collection of codes was developed in the *Matlab* language that can simulate multibody systems for kinematic and dynamic systems. This development allows the academic researcher to build multibody models, simulate and analyze these models. *MotionSim* was developed specifically to model multibody systems without physical definitions utilizing general coordinates, as well as to provide the researcher the ability to modify the source code to include any special constraints or elements that may be required. These codes utilize the *Matlab* language to vectorize the matrices to mimic equations written by hand, for simpler modification. *MotionSim* code has then been integrated into to the *Simulink* environment, which is available in *Matlab*, providing a pictorial representation of physical signals and methods. Figure 1 shows the augmented matrix form (mixed differential-algebraic equations of motion [3]) for solution of the multibody accelerations and Lagrange multipliers (constraint forces), as implemented in the *Simulink* environment.

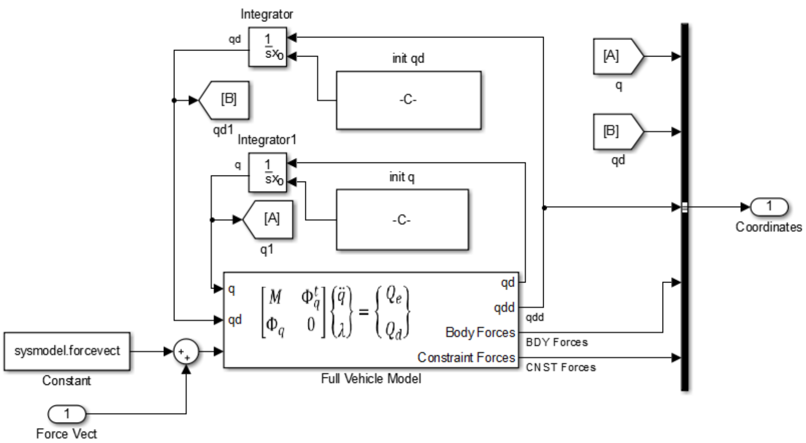

Figure 1. *Simulink* Implementation of *MotionSim* Dynamic Solver

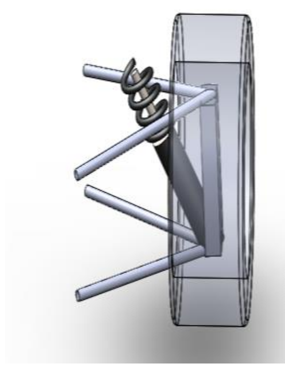

Figure 2. CAD Model for Quarter Car

A full vehicle model has been defined by creating a CAD model, and the multibody model generated using the *MotionSim* excel template. Figure 2 shows the quarter car suspension model utilized for the full vehicle model. The hard points and geometry points from CAD are used to auto-generate the multibody model utilized the *MotionSim* interfaces. Common elements such as springs and dampers are added with the model for simulating the suspension system and are available in the *MotionSim* codes. With the source of *MotionSim* easily modified the empirical Magic Tire Formula [4] has been added for simulation of slip angle and ratio effects generated from tires.

With the flexibility of *Matlab*'s toolboxes and *MotionSim* being native codes, integrating Artificial Intelligence into multibody simulations has been successful. Actual tire data by the Calspan Institute, providing tire testing data to FSAE teams, has been fitted to neural network models, which predict the lateral and longitudinal forces based on the slip angle and ratio, Figure 3. The neural network was able to generalize tire information of over 88,000 data points into a single neural network of 13 neurons in the hidden layer. This network was integrated into the multibody model allowing the tire information of the 88,000 data points to be included with no noticeable speed impact during the multibody simulation.

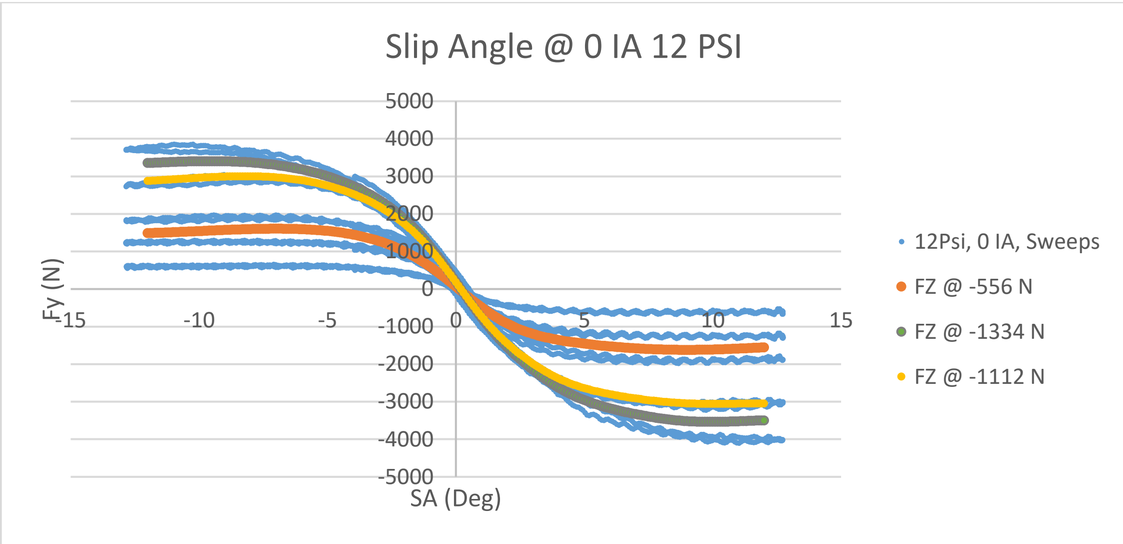

Figure 3. Neural Network Fitting of Tire Data [3]

All this information is put together and can be simulated in the *Simulink* environment. Instead of custom *Matlab* functions to represent tires, tire signals can be modeled as force elements that are added into the multibody solver, Figure 4. The general multibody solver provided by Figure 1, is encapsulated into the full vehicle *Simulink* model block. Using the *Simulink* environment, it is possible to create and break out force elements that affect the multibody model visually.

This allows the integration of control systems into the model, which can include simple elements such as a driver model utilizing a simple PID loop that may control the vehicle in the specified scenario. It also allows the inclusion of control and simulation of active suspension elements such as continuously variable dampers and stability control. The tire element block in Figure 4 reads the dynamic information from the rigid body tire element, and it then provides the forces that act upon that element. Figure 5 represents acceleration data of the chassis from a virtual driver in an acceleration and breaking scenario.

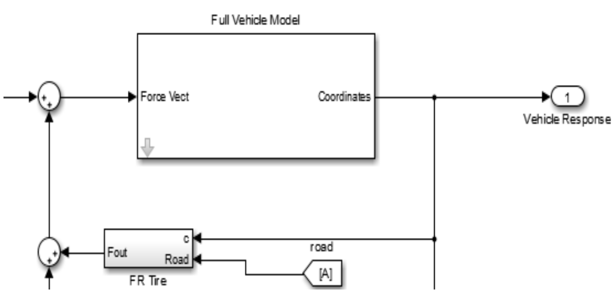

Figure 4. *Simulink* Vehicle Model Expanded

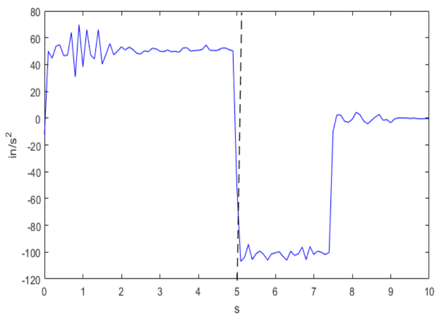

Figure 5. Acceleration Data of the Chassis

References

[1] M. Blundell, and D. Harty, *The Multibody Systems Approach to Vehicle Dynamics*, 2004.

[2] Calspan Tire Research Facility (TIRF), "TIRF Project 1464 Tire Data," FSAE Tire Test Consortium (TTC), February 2012.

[3] P. Nikravesh, Computer-Aided Analysis of Mechanical Systems, Prentice Hall, 1988.

[4] H. Pacejka, *Tire and Vehicle Dynamics*. Elsevier, 3 ed., 2012.