REPORT

MODELLING OF ANTI-LOCK BRAKING SYSTEM

Name: NOEL VARGHESE

Emp. ID: 214899

INTRODUCTION

This report shows how to model a simple model for an **Anti-Lock Braking System (ABS).** It simulates the dynamic behaviour of a vehicle under hard braking conditions. The model represents a single wheel, which may be replicated a number of times to create a model for a multi-wheel vehicle.

An anti-lock braking system (ABS) is a safety anti-skid braking system used on aircraft and on land vehicles, such as cars, motorcycles, trucks, and buses. ABS operates by preventing the wheels from locking up during braking, thereby maintaining tractive contact with the road surface and allowing the driver to maintain more control over the vehicle.

ANALYSIS AND PHYSICS

The wheel rotates with an initial angular speed that corresponds to the vehicle speed before the brakes are applied. Separate integrators were used to compute wheel angular speed and vehicle speed. Two speeds are used to calculate slip, which is determined by Equation 1. Here vehicle speed expressed as an angular velocity.

$$\omega_v = \frac{V}{R} \ (equals\ the\ wheel\ angular\ speed\ if\ there\ is\ no\ slip)$$

Equation 1:

$$\omega_v = \frac{V_v}{R_r}$$

$$slip = 1 - \frac{\omega_w}{\omega_v}$$

Where,

$\omega_v = vehicle\ speed\ divided\ by\ wheel\ radius$

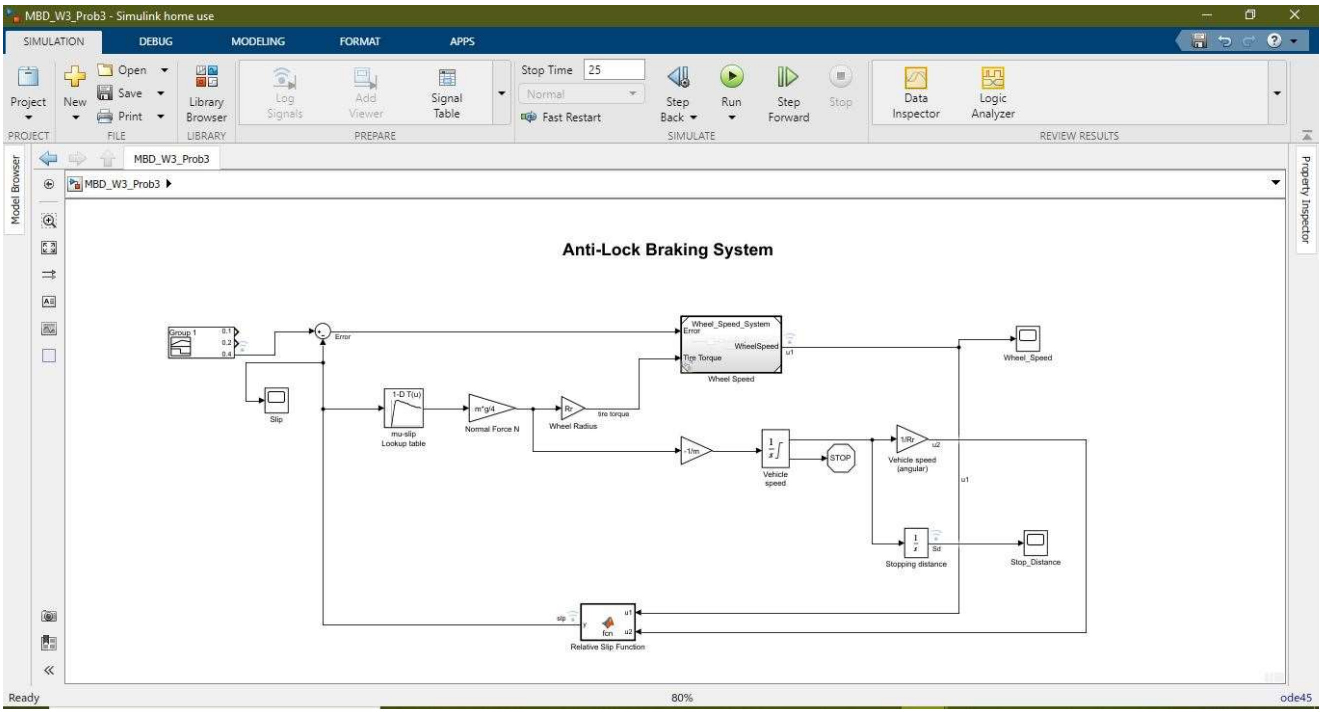
$V_v = vehicle\ linear\ velocity$

$R_r = wheel\ radius$

$\omega_w = wheel\ angular\ velocity$

From these expressions, it is observed that slip is zero when wheel speed and vehicle speed are equal, and slip equals one when the wheel is locked. A desirable slip value is 0.2, which means that the number of wheel revolutions equals 0.8 times the number of revolutions under non-braking conditions with the same vehicle velocity. This maximizes the adhesion between the tire and road and minimizes the stopping distance with the available friction.

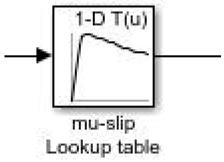
MODEL

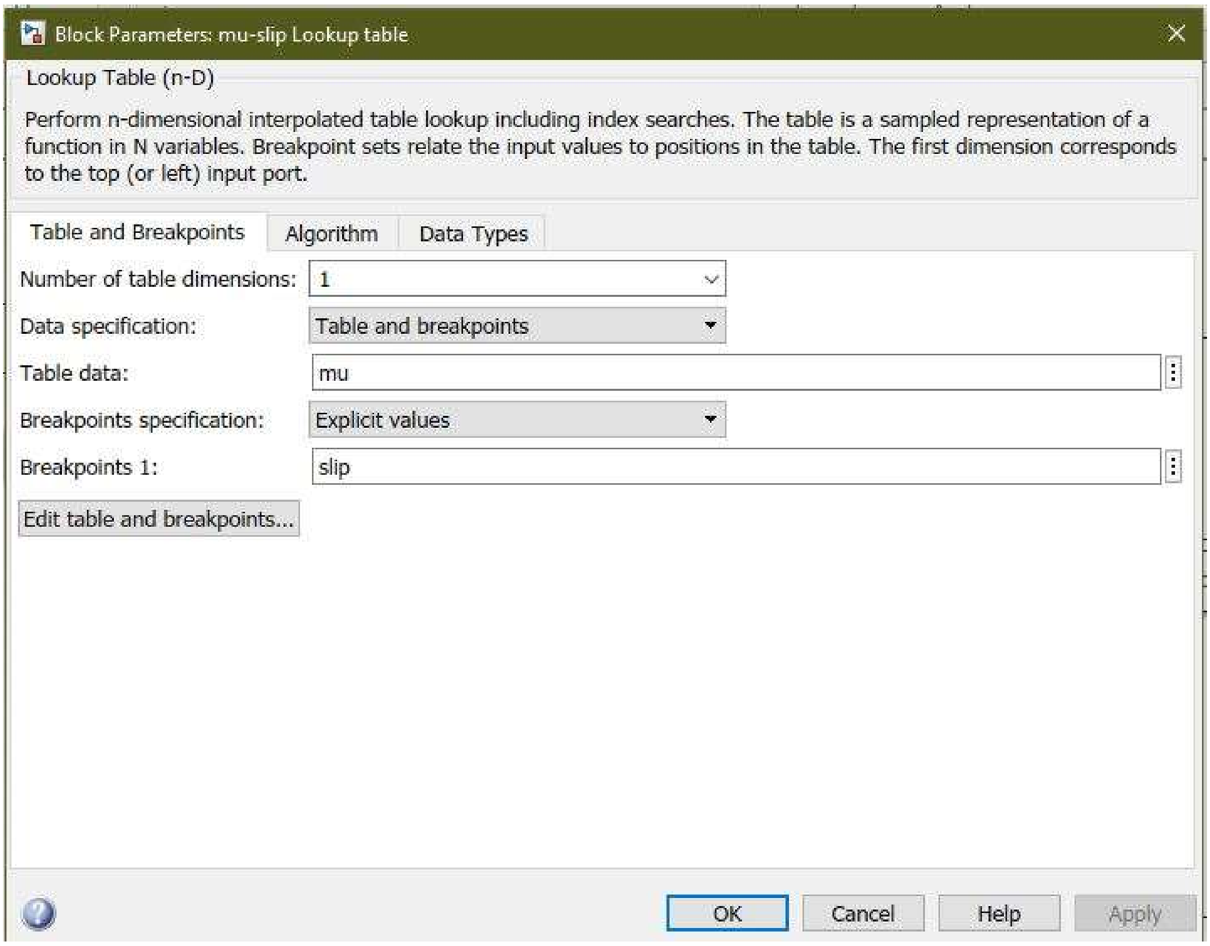


Anti-Lock Braking System Model

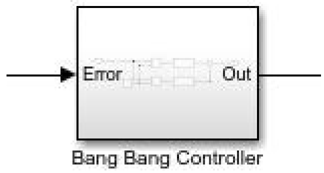
MODELING

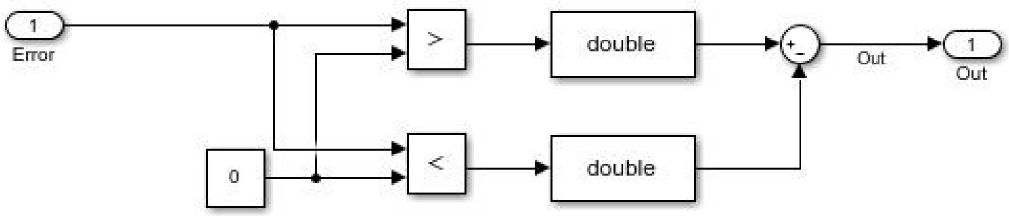
1. The friction coefficient between the tire and the road surface, mu, is an empirical function of slip, known as the mu-slip curve. A mu-slip curve was created by passing MATLAB variables into the block diagram using a **Simulink lookup table**.
   The values for the variables were obtained from the workspace. The variables were defined in a matlab script file created outside the model.



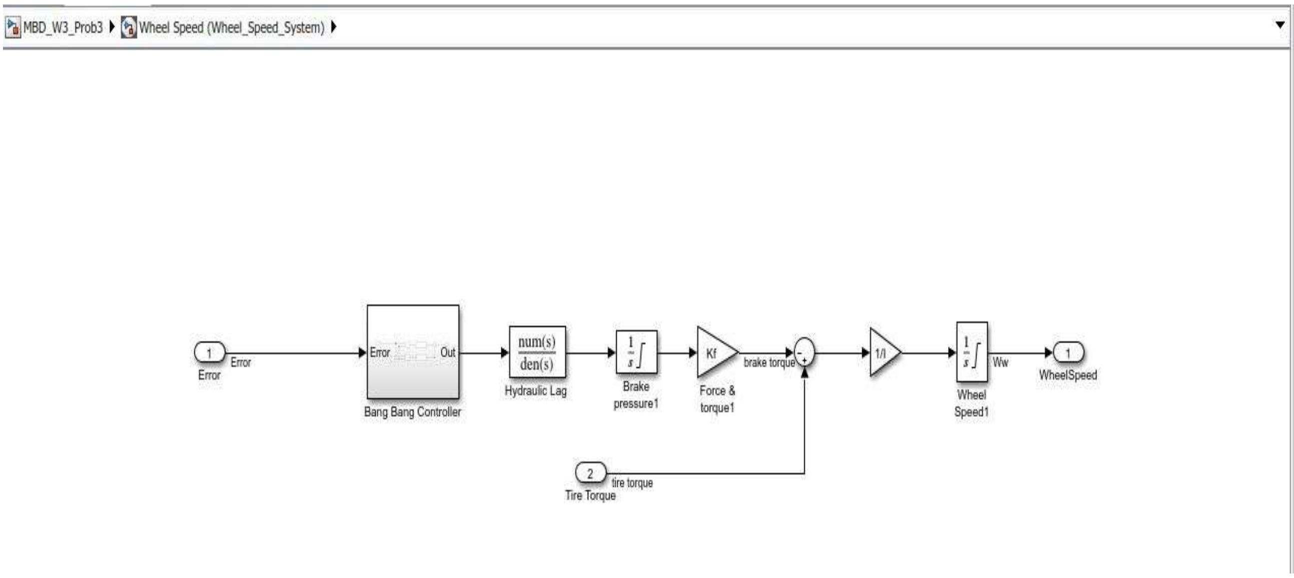


2. The model multiplies the friction coefficient, *mu*, by the weight on the wheel, *W*, to yield the frictional force, *Ff*, acting on the circumference of the tire. *Ff* is divided by the vehicle mass to produce the vehicle deceleration, which the model integrates to obtain vehicle velocity.
3. In this model, an ideal anti-lock braking controller is used, that uses **'bang-bang' control** based upon the error between actual slip and desired slip. The desired slip was set to the value of slip at which the mu-slip curve reaches a peak value, this being the optimum value for minimum braking distance.





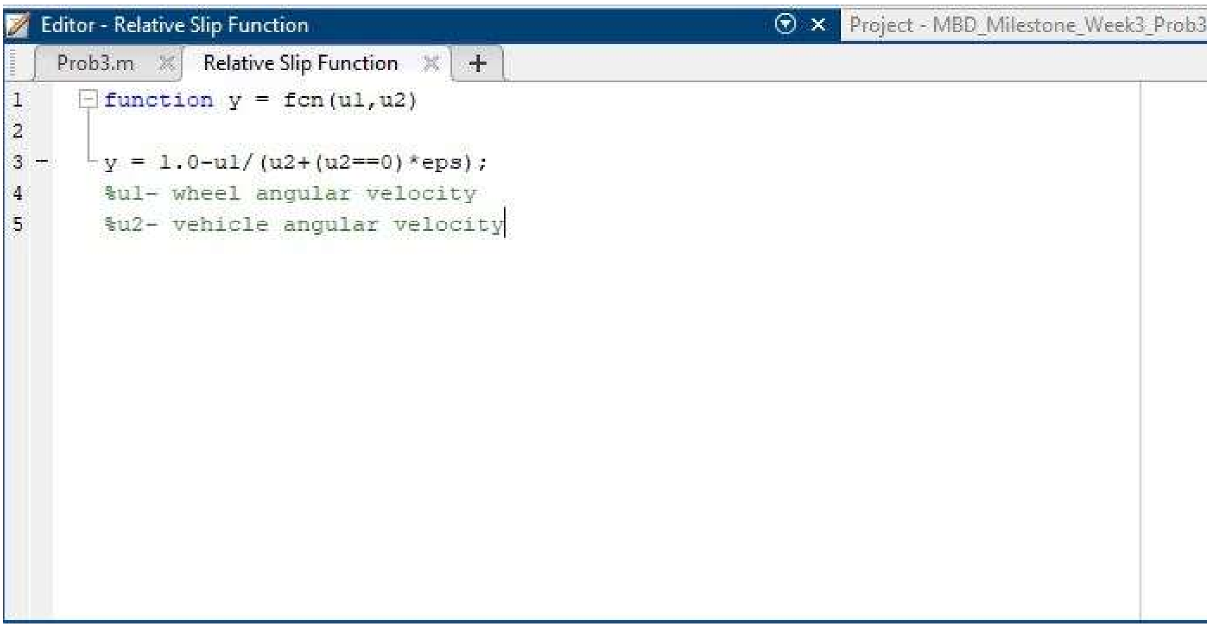
4. A wheel speed subsystem was built in which the bang bang controller was utilized.



To control the rate of change of brake pressure, the model subtracts actual slip from the desired slip and feeds this signal into a bang-bang control (+1 or -1, depending on the sign of the error). This on/off rate passes through a first-order lag that represents the delay associated with the hydraulic lines of the brake system. The model then integrates the filtered rate to yield the actual brake pressure. The resulting signal, multiplied by the piston area and radius with respect to the wheel (*Kf*), is the brake torque applied to the wheel.

The model multiplies the frictional force on the wheel by the wheel radius (*Rr*) to give the accelerating torque of the road surface on the wheel. The brake torque is subtracted to give the net torque on the wheel. Dividing the net torque by the wheel rotational inertia, *I*, yields the wheel acceleration, which is then integrated to provide wheel velocity. In order to keep the wheel speed and vehicle speed positive, limited integrators are used in this model.

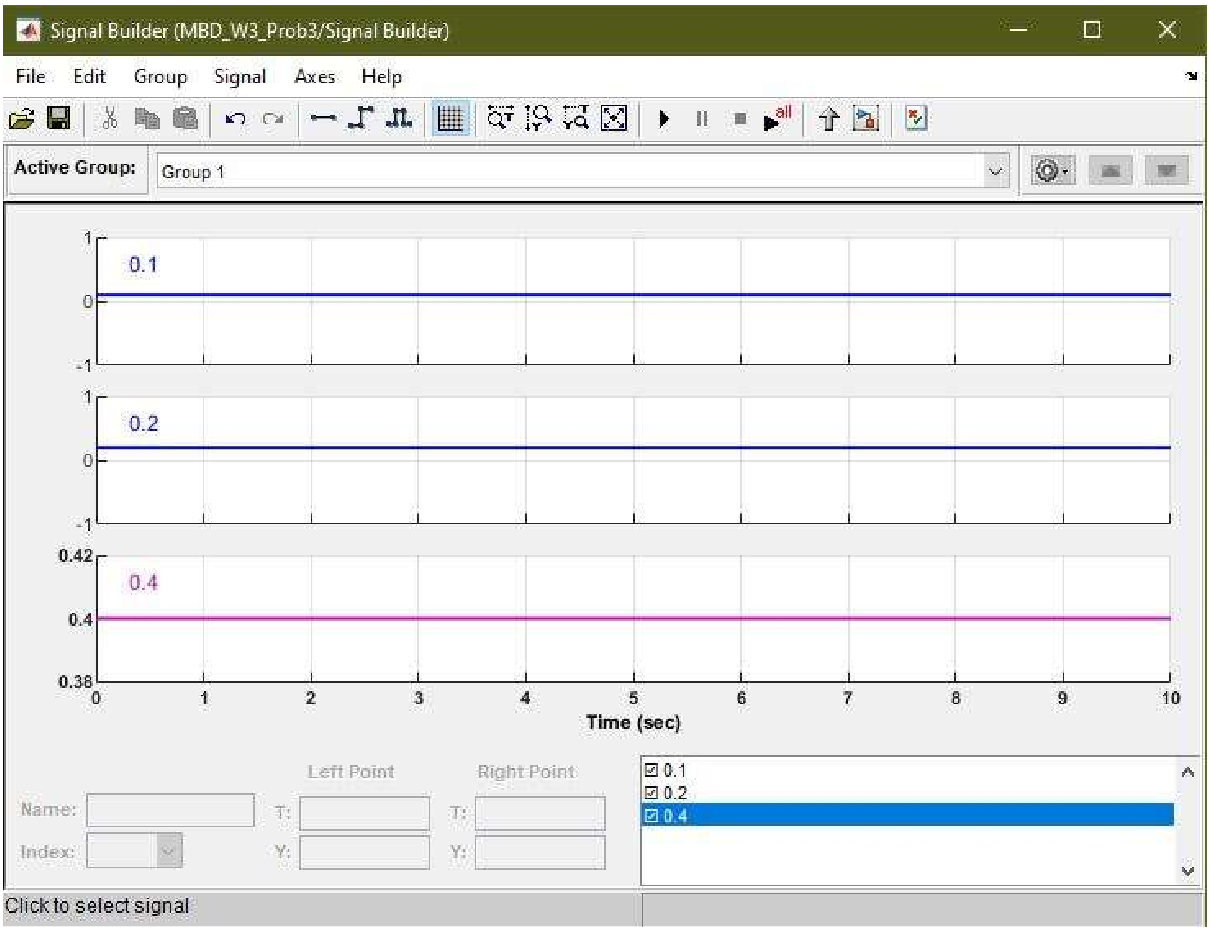
5. The relative slip of the wheel is then computed using the slip equation (Equation 1). This is realized using a **Matlab function block.**



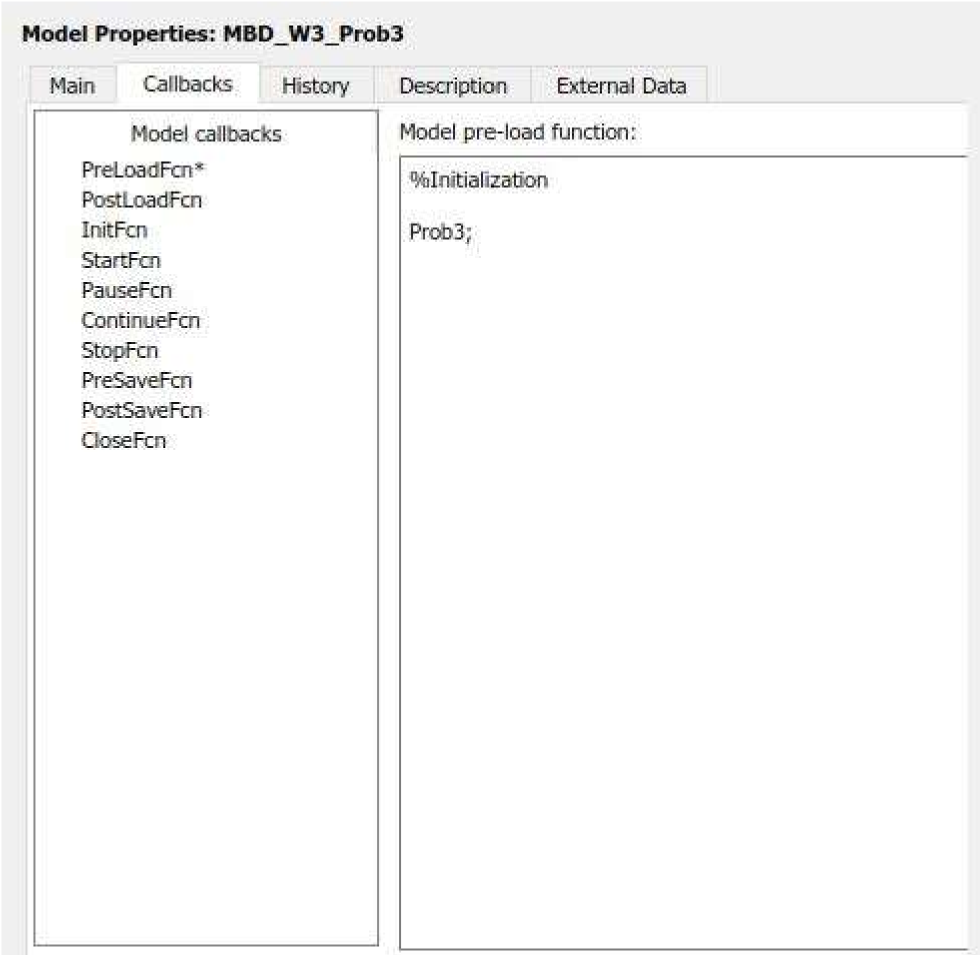

```
function y = fcn(u1,u2)

y = 1.0-u1/(u2+(u2==0)*eps);
%u1- wheel angular velocity
%u2- vehicle angular velocity
```

6. **Signal Builder**

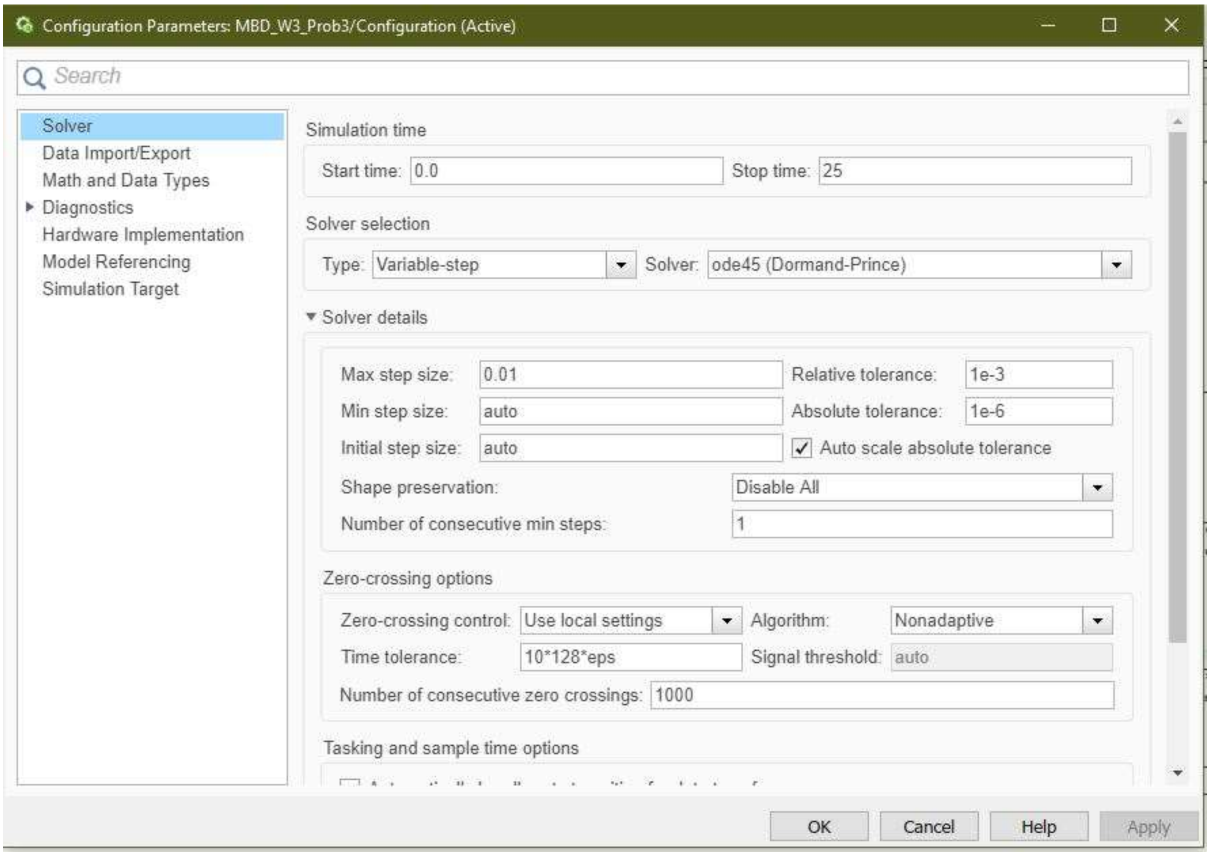
A signal builder block was used to generate 3 signals for slip (the ideal/not ideal slip values).



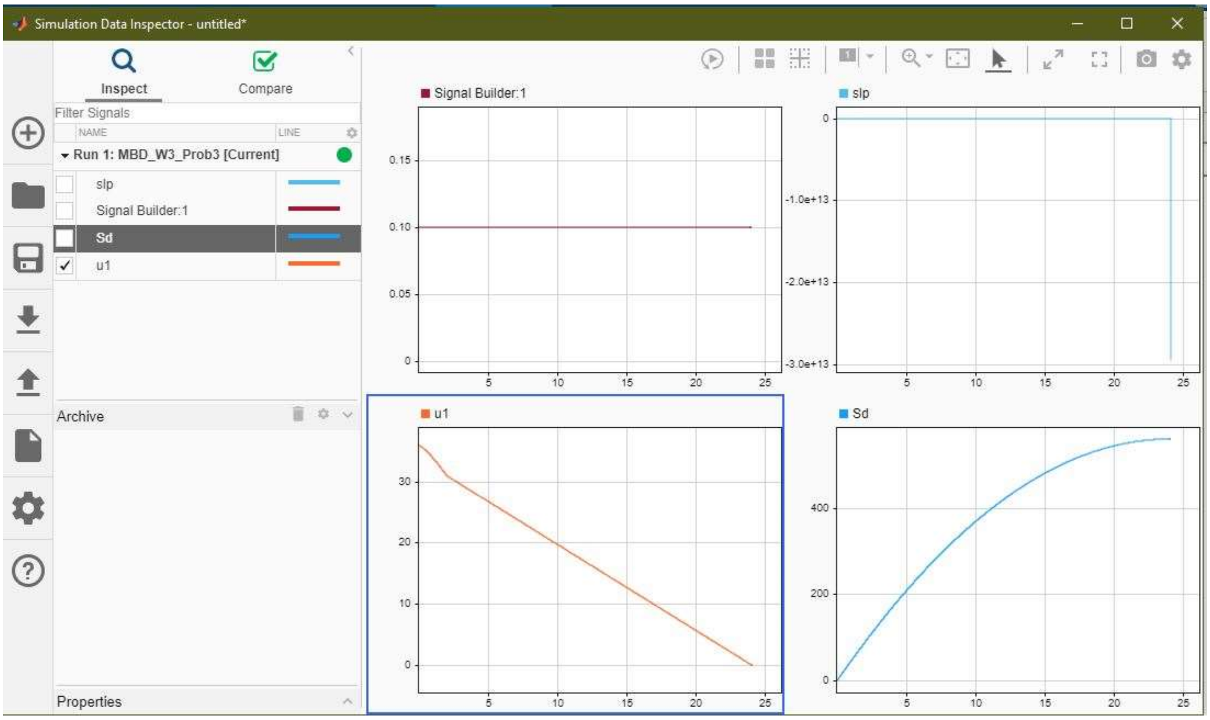
7. **Callback**
   For this model, in the model explorer – callback, a PreLoadFcn has been added to run the matlab script that containes the variables.



8. **Solver Selection:**
   For this particular model- the solver **ode45** of type **VariableStep** with max step size 0.01 was selected.

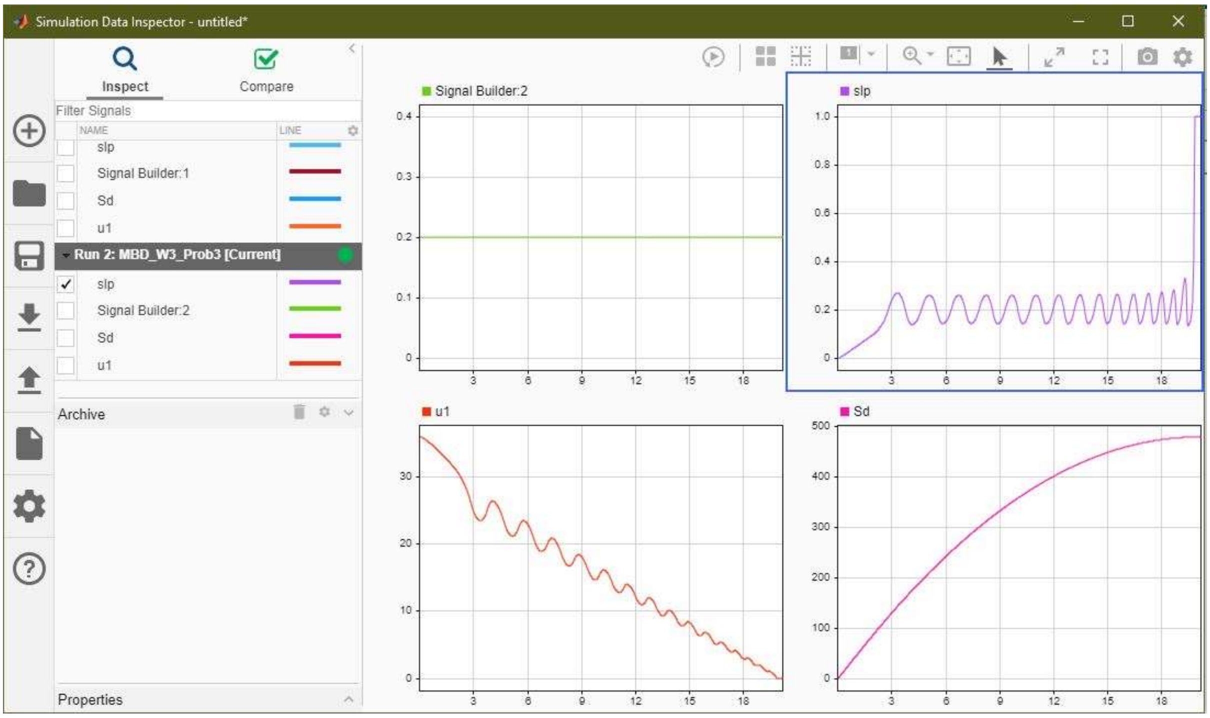


9. The model is then simulated for a stop time of 25 sec, taking initial velocity as 45Kmph. The desired relative slip (from signal builder), wheel angular velocity (u1) or wheel speed, relative slip (slp) and stopping distance (sd), signals are logged and the corresponding outputs are observed in the **Data Inspector.**
   The outputs for desired relative slip 0.1, 0.2 and 0.4 are as follows:



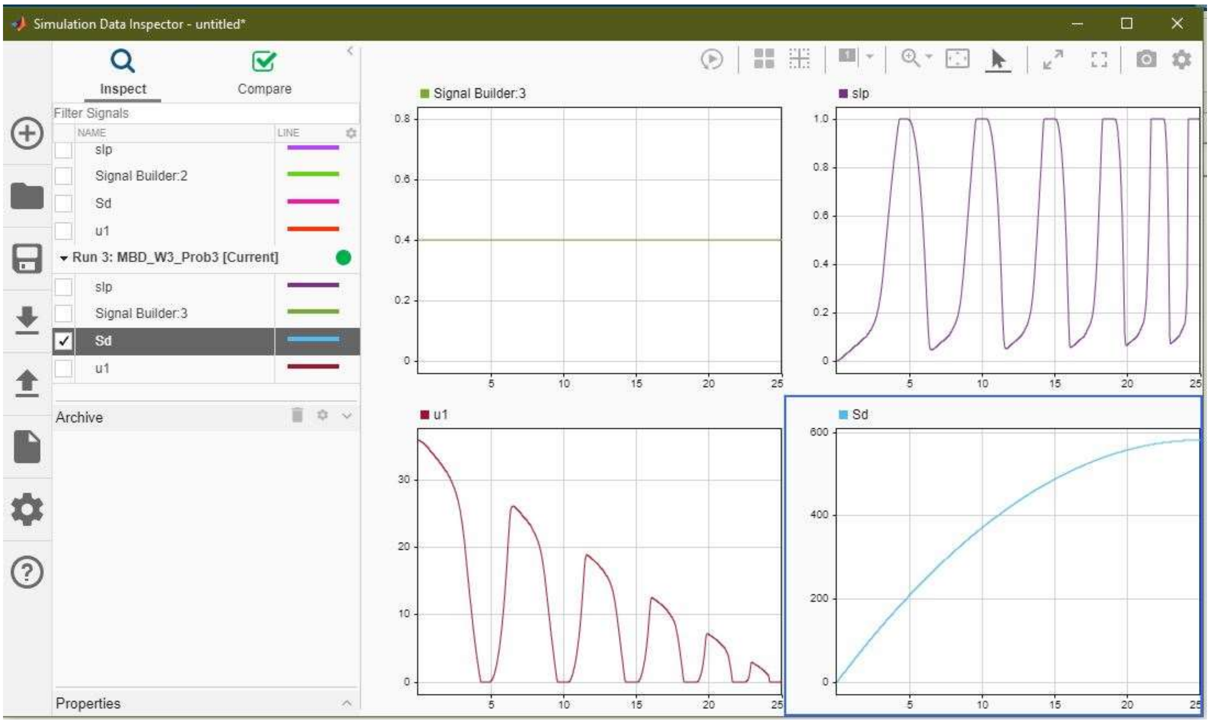
Simulation Data Inspector - untitled*
Inspect
Compare
Filter Signals
NAME
LINE
Run 1: MBD_W3_Prob3 [Current]
slp
Signal Builder:1
Sd
u1
Archive
Properties
Signal Builder:1
slp
u1
Sd

0.1



Simulation Data Inspector - untitled*
Inspect
Compare
Filter Signals
NAME
LINE
slp
Signal Builder:1
Sd
u1
Run 2: MBD_W3_Prob3 [Current]
slp
Signal Builder:2
Sd
u1
Archive
Properties
Signal Builder:2
slp
u1
Sd

0.2



Simulation Data Inspector - untitled*
Inspect
Compare
Filter Signals
NAME
LINE
slp
Signal Builder:2
Sd
u1
Run 3: MBD_W3_Prob3 [Current]
slp
Signal Builder:3
Sd
u1
Archive
Properties
Signal Builder:3
slp
u1
Sd

0.4