m = 1500; % Vehicle mass (kg)

g = 9.81; % Acceleration due to gravity (m/s^2)

R = 0.3; % Wheel radius (m)

Iz = 300; % Moment of inertia of wheel (kg.m^2)

Vx0 = 30; % Initial velocity (m/s)

* These define physical properties of the vehicle and wheels.
* m, g, R, Iz: Used to calculate force and rotation.
* Vx0 = 30 m/s means the vehicle starts at ~108 km/h

**🔷 SECTION 2: Road Conditions**

mu\_dry = 0.9;

mu\_wet = 0.6;

mu\_ice = 0.2;

road\_condition = 'wet';

* mu is the friction coefficient.
* Depending on the surface (dry/wet/ice), friction changes.
* 'wet' is selected here.

switch road\_condition

case 'dry'

mu = mu\_dry;

case 'wet'

mu = mu\_wet;

case 'ice'

mu = mu\_ice;

end

* Sets mu based on chosen surface.

**SECTION 3: Time Configuration**

dt = 0.001;

t\_end = 5;

t = 0:dt:t\_end;

* Simulation runs from 0 to 5 seconds.
* dt = 0.001 means we calculate values every 0.001 second.

**🔷 SECTION 4: Initial Values**

vx = Vx0; % Vehicle velocity

vw = Vx0/R; % Wheel angular velocity

slip = 0;

slip\_desired = 0.2;

* vx: Vehicle speed (linear).
* Vw: Wheel speed (rotational).
* Slip: Starts at 0.
* slip\_desired = 0.2 means we want 20% controlled slip for best braking.

**🔷 SECTION 5: PID Controller Settings**

Kp = 100;

Ki = 10;

Kd = 1;

integral = 0;

prev\_error = 0;

* Standard PID control settings to adjust brake force based on slip error:
  + Kp: reacts to current error.
  + Ki: reacts to accumulated error.
  + Kd: reacts to change in error.

**🔷 SECTION 6: Load Fuzzy Controller**

fis = readfis('fuzzyABS');

* Loads a Fuzzy Inference System (.fis) file named fuzzyABS.fis.
* Fuzzy logic helps in making control decisions when the environment is uncertain (like varying road grip).

**🔷 SECTION 7: Logging Variables**

vx\_log = zeros(size(t));

slip\_log = zeros(size(t));

vw\_log = zeros(size(t));

brake\_force\_log = zeros(size(t));

* These arrays will store data during the simulation so we can later plot and analyze them.

**🔷 SECTION 8: Simulation Loop**

for i = 1:length(t)

* Start of a loop that runs for each time step (from 0 to 5 seconds).

**🔹 Inside the Loop:**

**✅ Step 1: Calculate Slip**

if vx > 0

slip = (vx - R \* vw) / vx;

else

slip = 0;

end

* Slip = how much slower the wheel rotates compared to the car’s speed.
* If the car stops (vx = 0), slip = 0.

**✅ Step 2: PID Control Output**

error = slip\_desired - slip;

integral = integral + error \* dt;

derivative = (error - prev\_error) / dt;

u\_pid = Kp \* error + Ki \* integral + Kd \* derivative;

prev\_error = error;

* The PID controller calculates how much brake force is needed to maintain the desired slip.

**✅ Step 3: Fuzzy Control Output**

u\_fuzzy = evalfis(fis, [slip vx]);

* The fuzzy controller also gives a control value based on current slip and vehicle speed.

**✅ Step 4: Combine Both Controls**

brake\_force = max(0, min(mu \* m \* g, u\_fuzzy + u\_pid));

* Adds fuzzy and PID outputs to compute final brake force.
* Clamps it to be between 0 and maximum possible brake force.

**✅ Step 5: Update Vehicle Speed**

Fx = mu \* m \* g;

ax = brake\_force / m;

vx = max(0, vx - ax \* dt);

* Calculates deceleration (ax) and updates vehicle speed.

**✅ Step 6: Update Wheel Speed**

Tw = brake\_force \* R;

alpha\_w = Tw / Iz;

vw = max(0, vw - alpha\_w \* dt);

* Applies brake torque (Tw) to the wheel.
* Updates wheel’s angular speed.

**✅ Step 7: Log Data**

vx\_log(i) = vx;

vw\_log(i) = vw;

slip\_log(i) = slip;

brake\_force\_log(i) = brake\_force;

* Saves current values for plotting later.

**🔷 SECTION 9: Plotting the Results**

figure;

subplot(3,1,1);

plot(t, vx\_log);

title('Vehicle Speed'); ylabel('m/s'); grid on;

subplot(3,1,2);

plot(t, slip\_log);

title('Slip Ratio'); ylabel('Slip'); grid on;

subplot(3,1,3);

plot(t, brake\_force\_log);

title('Brake Force'); ylabel('N'); xlabel('Time (s)'); grid on;

* 3 plots:
  1. Vehicle speed over time.
  2. Slip ratio over time.
  3. Brake force over time.

**Purpose:** To visualize how well the ABS controller performs in reducing speed while avoiding wheel lock.