function [F\_x, F\_y] = brush\_tire\_model(alpha, tire\_parameters)

% Brush tire model implementation

% Inputs:

% - alpha: slip angle (in radians)

% - tire\_parameters: struct containing tire parameters

% Outputs:

% - F\_x: longitudinal tire force

% - F\_y: lateral tire force

% Extract tire parameters

B = tire\_parameters.B;

C = tire\_parameters.C;

D = tire\_parameters.D;

E = tire\_parameters.E;

F = tire\_parameters.F;

R0 = tire\_parameters.R0;

F\_z = tire\_parameters.F\_z;

% Calculate slip ratio

kappa = -tan(alpha);

% Calculate longitudinal tire force

F\_x = D\*sin(C\*atan(B\*kappa - E\*(B\*kappa - atan(B\*kappa)))) + F\_z\*R0;

% Calculate lateral tire force

alpha\_eq = atan((alpha + F\*atan(B\*alpha))/D);

F\_y = C\*sin(atan(B\*alpha\_eq));

end

% Define tire parameters

tire\_parameters.B = 10;

tire\_parameters.C = 1.9;

tire\_parameters.D = 10000;

tire\_parameters.E = 0.97;

tire\_parameters.F = 0.016;

tire\_parameters.R0 = 0.3;

tire\_parameters.F\_z = 5000;

% Define slip angle range

slip\_angle\_range = -10:0.01:10;

% Initialize arrays to store tire forces

longitudinal\_force = zeros(size(slip\_angle\_range));

lateral\_force = zeros(size(slip\_angle\_range));

% Calculate tire forces

for i = 1:length(slip\_angle\_range)

[longitudinal\_force(i), lateral\_force(i)] = brush\_tire\_model(slip\_angle\_range(i), tire\_parameters);

end

% Save results to CSV file

data = [slip\_angle\_range', longitudinal\_force', lateral\_force'];

csvwrite('brush\_forces\_1.csv', data);