به نام خدا

برای طراحی تابع کنترل سیستم ترمز و گاز یک اتومبیل هوشمند، باید متغیرهای کلیدی را شناسایی کرده و سپس تابعی تعریف کنیم که این متغیرها را به نحوی مدیریت کند که عملکرد خودرو بهینه باشد. در اینجا، پنج متغیر کلیدی انتخاب می‌شوند و سپس تابع کنترل پیاده‌سازی و تحلیل می‌شود.

متغیرهای کلیدی

1. \*\*سرعت فعلی خودرو (\( v \))\*\*: سرعت فعلی خودرو که به کنترل میزان گاز و ترمز تأثیر می‌گذارد.

2. \*\*موقعیت پدال گاز (\( p\_g \))\*\*: درصد فشردگی پدال گاز.

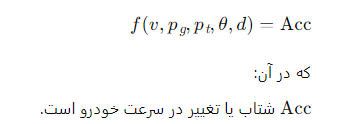
3. \*\*موقعیت پدال ترمز (\( p\_t \))\*\*: درصد فشردگی پدال ترمز.

4. \*\*شیب جاده (\( \theta \))\*\*: شیب جاده که بر نیاز به ترمز و گاز تأثیر می‌گذارد.

5. \*\*فاصله تا مانع (\( d \))\*\*: فاصله تا مانع جلو که نیاز به ترمز را تعیین می‌کند.

تابع کنترل

فرض می‌کنیم که هدف از تابع کنترل، تنظیم میزان گاز و ترمز بر اساس این متغیرها به گونه‌ای است که خودرو به‌طور بهینه سرعت را حفظ کرده و از تصادف جلوگیری کند. تابع کنترل می‌تواند به صورت زیر تعریف شود:



پیاده‌سازی تابع کنترل

در اینجا، یک تابع کنترل ساده با استفاده از زبان Python و کتابخانه‌های Numpy و Matplotlib پیاده‌سازی شده است. این تابع بر اساس متغیرهای ورودی، میزان گاز و ترمز را تعیین می‌کند.

import numpy as np

import matplotlib.pyplot as plt

# تعریف تابع کنترل

def control\_function(v, p\_g, p\_t, theta, d):

# پارامترهای ثابت برای مدل

k\_g = 0.1 # حساسیت گاز

k\_t = 0.3 # حساسیت ترمز

k\_d = 0.5 # حساسیت فاصله تا مانع

k\_theta = 0.2 # حساسیت شیب جاده

# محاسبه شتاب بر اساس ورودی‌ها

Acc = (k\_g \* p\_g – k\_t \* p\_t) – (k\_d / (d + 1)) – (k\_theta \* theta)

return Acc

# تعریف پارامترها

speed = np.linspace(0, 100, 100) # سرعت از 0 تا 100

pedal\_gas = 0.5 # موقعیت پدال گاز (50%)

pedal\_brake = 0.2 # موقعیت پدال ترمز (20%)

slope = 0.1 # شیب جاده

distance = 20 # فاصله تا مانع

# محاسبه شتاب

acceleration = control\_function(speed, pedal\_gas, pedal\_brake, slope, distance)

# رسم نمودار

plt.figure(figsize=(12, 6))

# نمودار شتاب بر حسب سرعت

plt.subplot(1, 2, 1)

plt.plot(speed, acceleration, label=’Acceleration’)

plt.xlabel(‘Speed (km/h)’)

plt.ylabel(‘Acceleration (m/s²)’)

plt.title(‘Acceleration vs Speed’)

plt.legend()

# نمودار شتاب بر حسب فاصله تا مانع

plt.subplot(1, 2, 2)

distances = np.linspace(1, 100, 100) # فاصله تا مانع از 1 تا 100

acceleration\_distance = control\_function(speed[0], pedal\_gas, pedal\_brake, slope, distances)

plt.plot(distances, acceleration\_distance, label=’Acceleration vs Distance’)

plt.xlabel(‘Distance to Obstacle (m)’)

plt.ylabel(‘Acceleration (m/s²)’)

plt.title(‘Acceleration vs Distance to Obstacle’)

plt.legend()

plt.tight\_layout()

plt.show()

----------------------------------------------------------------------------

توضیحات

1. \*\*تعریف تابع کنترل:\*\* تابع `control\_function` با استفاده از پارامترهای ورودی، شتاب خودرو را محاسبه می‌کند. میزان گاز و ترمز، شیب جاده، و فاصله تا مانع به صورت خطی بر شتاب تأثیر می‌گذارند.

2. \*\*رسم نمودار:\*\* دو نمودار رسم شده‌اند:

- \*\*شتاب بر حسب سرعت:\*\* این نمودار نشان می‌دهد که چگونه شتاب خودرو با تغییر سرعت تغییر می‌کند.

- \*\*شتاب بر حسب فاصله تا مانع:\*\* این نمودار نشان می‌دهد که چگونه شتاب خودرو با تغییر فاصله تا مانع تغییر می‌کند.

این تابع و نمودارها نمایی ساده از نحوه کنترل سیستم ترمز و گاز یک اتومبیل هوشمند را ارائه می‌دهند. در پیاده‌سازی‌های واقعی، مدل‌ها پیچیده‌تر خواهند بود و به‌طور دقیق‌تر باید شرایط مختلف رانندگی و پارامترهای اضافی را در نظر بگیرند.