Traffic Mitigation Using V2V Communication

Berkeley | EECS
Electrical Engineering and Computer Sciences

EECS 149/249A - 2025

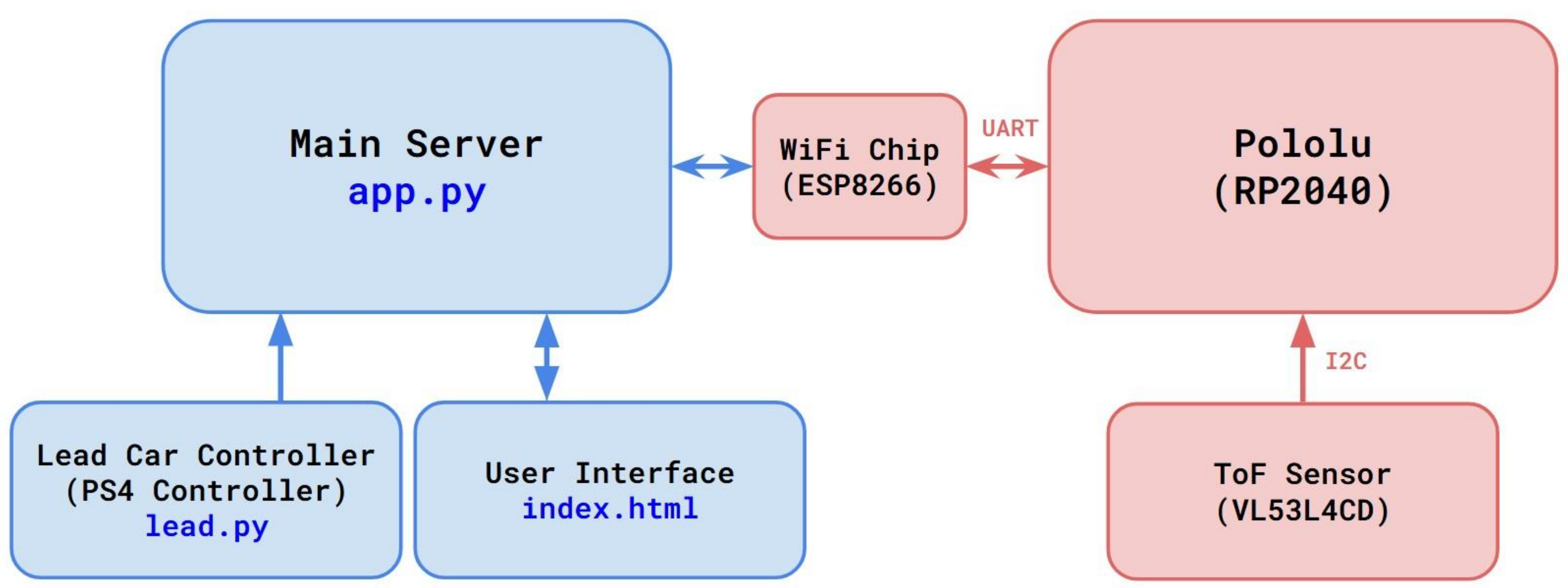
Group 5: Bazil Ahmad, Travis Miller, Lawrence Rhee, Rayyan Saiyed

Project Goals / Motivation

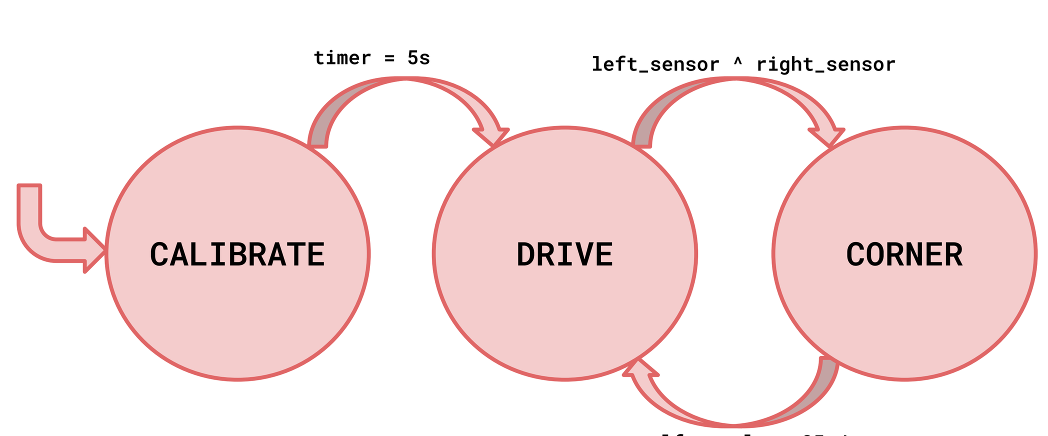
The goal of this project is to demonstrate the effectiveness of a V2V AV policy in mitigating traffic congestion with a lead adversarial [1] and human drivers on freeways. In order to achieve a physical demo, we employ the use of Pololu robots driving around an oval track (demonstrating per-lane effectiveness).

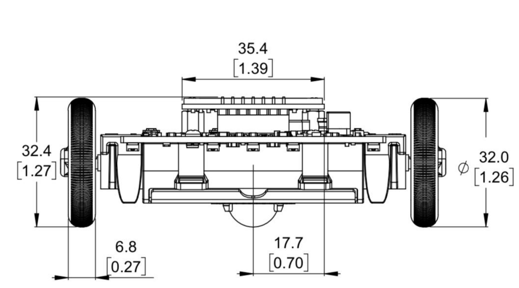
Overview / Architecture

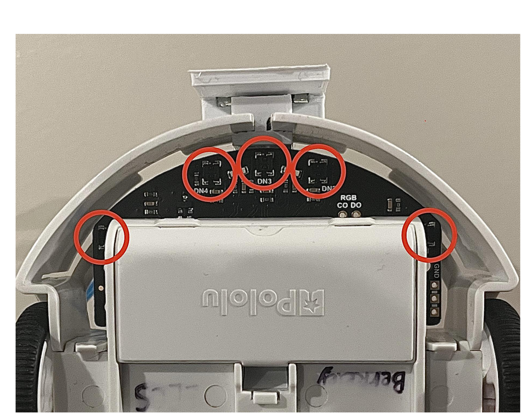
An **ESP8266 WiFi chip** links Pololu robots to a central server, enabling real-time telemetry and remote control commands.

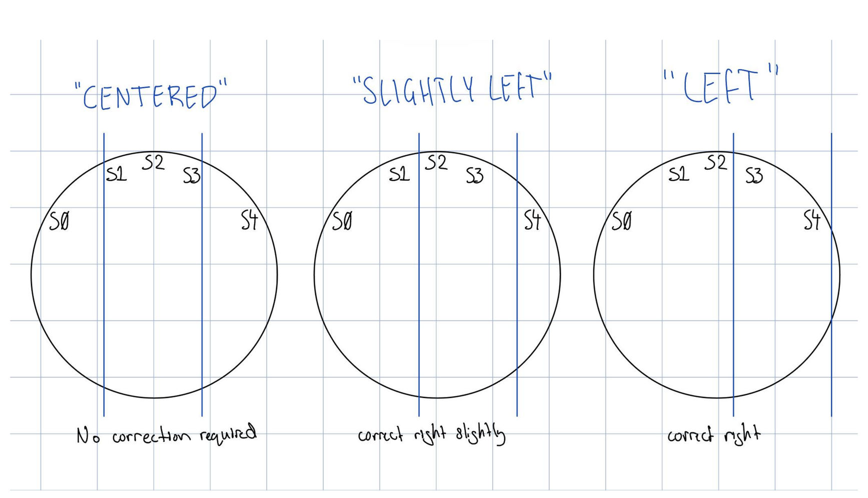


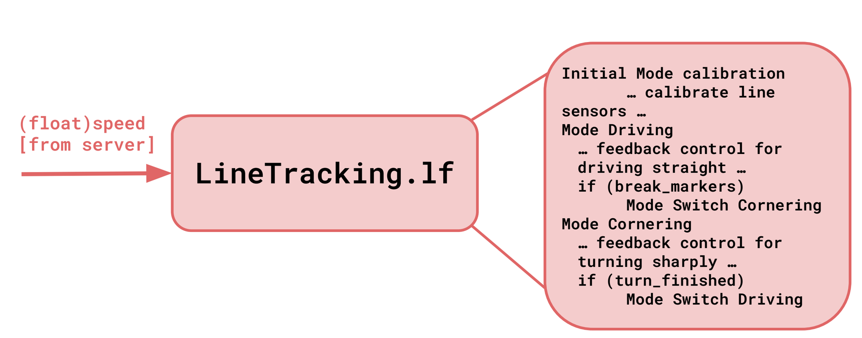
Pololu Line Tracking







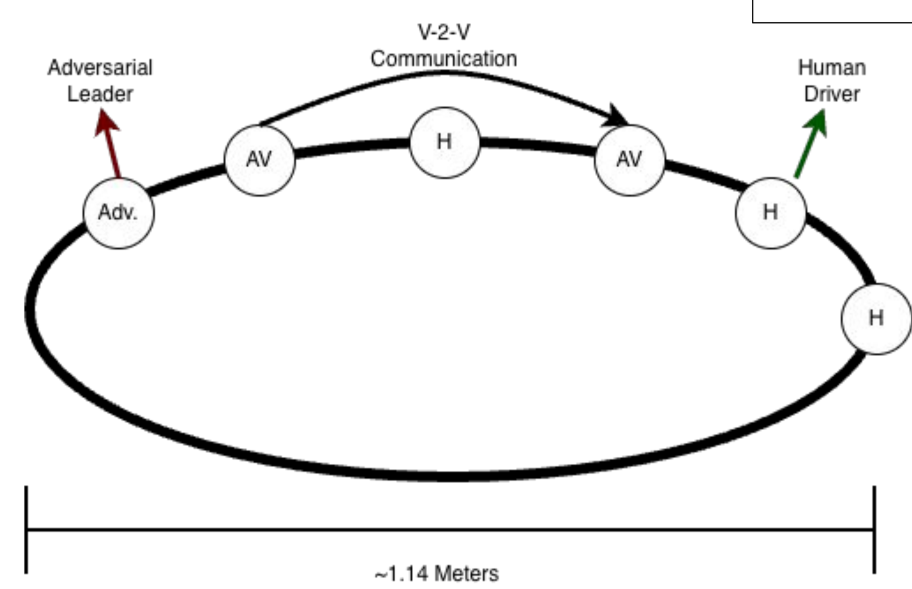




Intelligent Driving Model

$$\dot{v} = a\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \left(\frac{s^*}{s}\right)^2\right]$$

<u>IDM Parameters</u>
v: current speed.
v_0: desired speed.
s: The actual bumper-to-bumper distance to the car in front.
s^*: The desired minimum gap (dynamic, depends on speeds).
a: Maximum comfortable acceleration.



Adversarial Leader: Real driver exhibiting aggressive behavior (controlled by PS4 controller).
AV: Uses "perfect" IDM (quick reactions, optimized acceleration).
H: Modeled Human Driver using standard IDM.

We define "perfect" as quick reaction time and minimal wasted acceleration/deceleration while keeping an appropriate speed.

$$a_{\text{raw}} = f_{\text{IDM}}(v_{\text{own}},\ s,\ v_{\text{front}})$$

$$a_{\text{safe}} = f_{\text{safe}}(a_{\text{raw}},\ v_{\text{own}},\ s,\ v_{\text{front}})$$

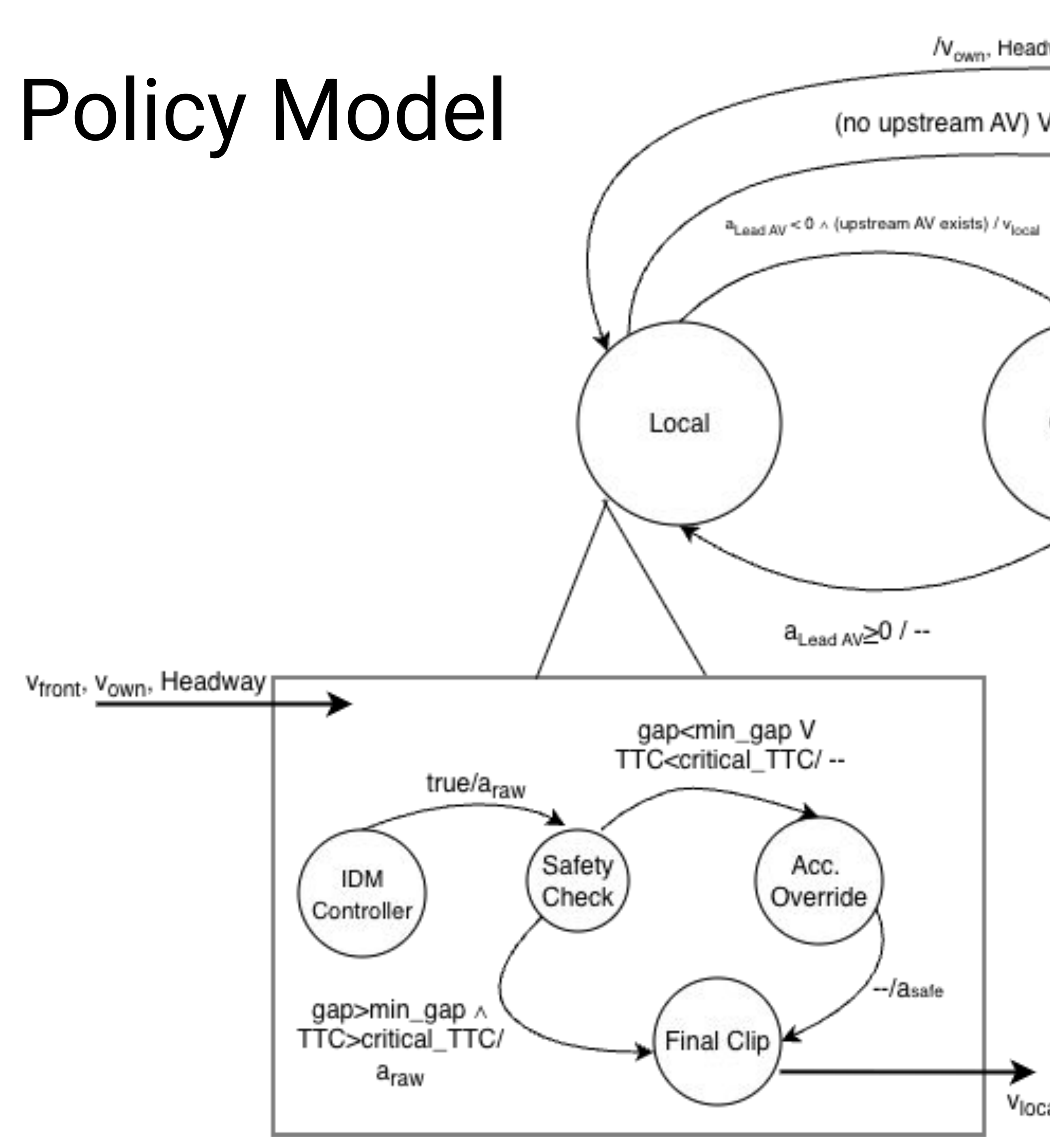
$$v_{AV_i} = \text{clip}(v_{\text{base}} + a_{\text{safe}}\Delta t,\ 0,\ v_{\max})$$

$$v_{AV_{2,\text{opt}}} = \begin{cases} v_{\text{AV}_1}, & a_{\text{leader}} > 0 \\ \beta\, v_{\text{AV}_2} + (1-\beta)\, v_{\text{AV}_1}, & a_{\text{leader}} \le 0 \end{cases}$$

$$\beta := \text{TRUST}, \qquad 0 \le \beta \le 1$$

The lead AV's velocity propagates to the follower. If the leader brakes, the follower calculates an optimal velocity using a trust factor (Beta) that ideally scales with inter-AV distance. [Value is fixed for physical demo]

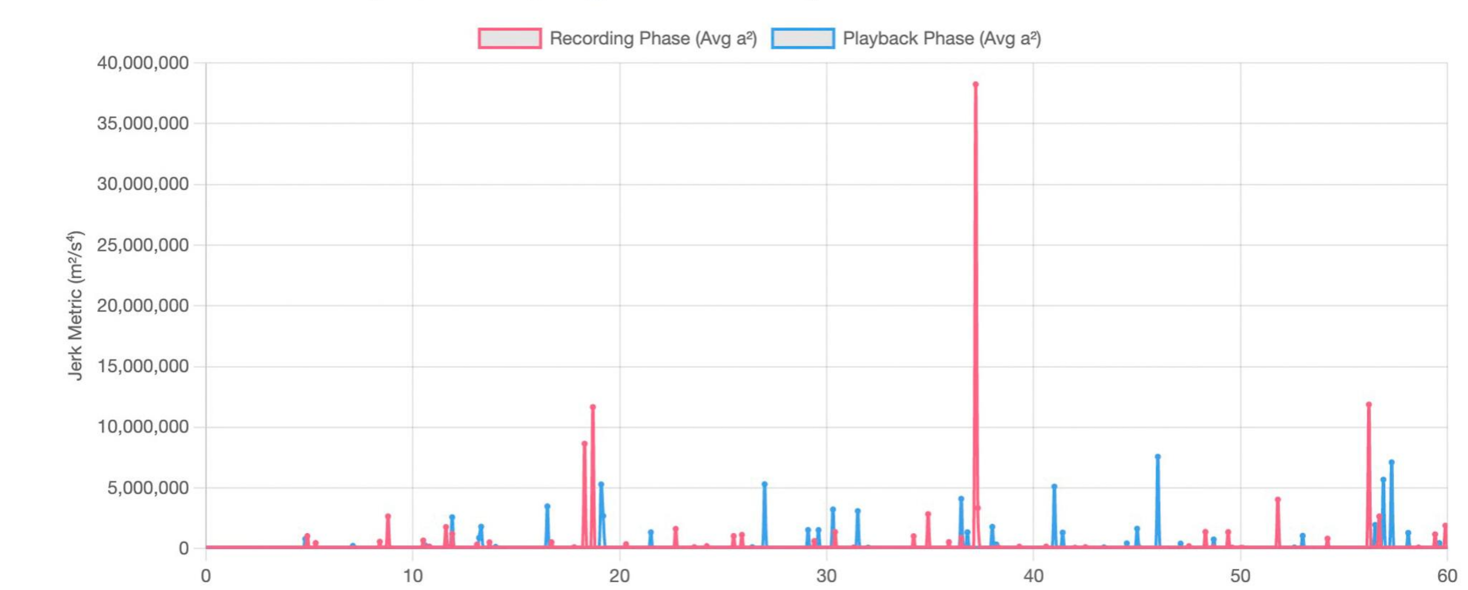
AV Policy Model



<u>Policy Overview</u>

AVs compute safe velocities using a deterministic IDM controller with safety overrides. When an upstream AV is braking, the trailing AV blends local data with communicated upstream velocity to mitigate shockwaves. Otherwise, it uses local perception. This asymmetric trust policy reduces stop-and-go traffic while preventing unsafe acceleration.

Jerk Profile Comparison (Avg a² vs Time)



Analysis Comparison

Bot ID	Recording Phase		Playback Phase	
	Avg Speed (m/s)	Jerk (m/s³)	Avg Speed (m/s)	Jerk (m/s³)
Robot #0	0.298	6719.873	0.280	7581.358
Robot #1	0.282	3739.059	0.265	4169.235
Robot #2	0.276	4307.506	0.260	4664.262
Robot #3	0.272	4163.045	0.251	2390.216
AVERAGE / TOTAL	**0.282**	**4732.371**	**0.264**	**4701.268**

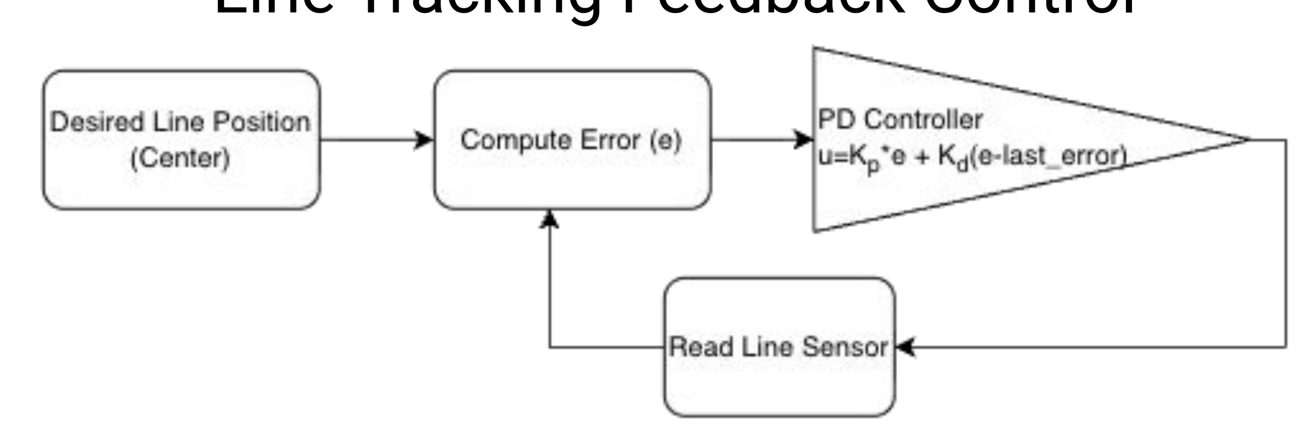
MOD-WIFI-ESP8266-DEV

Connects the server to the Pololu robot via a UART interface.

VL53L4CD Time-of-Flight Sensor

Accurately measures frontal distance (7–600 mm) via I2C, transmitting data through UART.

Line Tracking Feedback Control



Extension for trailing car to have valid ToF reading on turns