Project Overview

- Cooperative Adaptive Cruise Control
- Goal: provide additional convenience and safety to vehicle drivers
- Motivation for project
  - Address problem of driver constantly needing to adjust speed or disengage and reengage cruise control due to nearby vehicles
  - Facilitates safe and easy cruise control
Overview of Features

- Three main levels of functionality:
  - Cruise Control
  - Adaptive Cruise Control
  - Cooperative Adaptive Cruise Control

Cruise Control

- No sensing or communication
- Maintain driver-set speed
Adaptive Cruise Control

- Sensing with camera and radar systems
- Automatic Speed Adjustment

Cooperative Adaptive Cruise Control

- Sensing with camera and radar systems
- Automatic Speed Adjustment
- Communication with DSRC Radio Link
- GPS Data
- Platooning
Domain Research

- Investigated Adaptive Cruise Control patent
  - Provided insight to currently running sensor systems
- Investigated Adaptive Cruise Control System Overview (MIT)
  - Provided insight into the physical constraints of running systems
- Investigated Proportional-Integral-Derivative Controllers
  - Provided insight into the software constraints of the running systems

Constraints

- System must interact with the pre-existing subsystems
- System must communicate with subsystems via the CAN Bus
- System must be implemented without the use of dynamic memory allocation
Use Case Diagram

- CACC System
- Driver
- Set Following Gap
- Approve Platoon
- Form Platoon
- Set Target Speed
- Setup Functionality Level
- Turn On
- Pause
- Resume
- Turn Off
- Acknowledge Message
- Request Platoon
- Broadcast Alert
- Request ID

Domain Model Diagram

- VehicleController
  - objectID id
  - vector position
  - vector velocity
- TrackedObject
  - isOperational()
  - getRadarData()
- Radar Sensor
  - isOperational()
  - getIncomingMessage()
  - sendMessage(objectID, data)
- DSRC Radio Communication
  - isOperational()
  - getPower()
  - setPower(float value)
- Electronic Throttle Control
  - isOperational()
  - getMagnitude()
  - setMagnitude(float magnitude)
- Brake By Wire
  - isOperational()
  - sendRequest(vector location)
  - getIncomingRequest(data)
  - sendVehicleInfo(data info)
  - getIncomingResponse(data)
- Radar Transponder
  - isOperational()
  - getLocalization()
- GPS
  - isOperational()
  - getImageData()
- Camera
  - pressOnOffButton()
  - pressPauseButton()
  - pressSetPlusButton()
  - pressResumeMinusButton()
  - pressSetFollowingGap(int level)
  - pressBrakePedal(float amount)
  - pressGasPedal(float amount)
  - pressUpButton()
  - pressDownButton()
  - pressAcceptButton()
  - pressBackCancelButton()
  - displayCurrentSpeed(float speed)
  - displayTargetSpeed(float speed)
  - displayMessage(data message, bool ackRequired)
  - displayEmergencyWarning()
  - displayCruiseState(data state)

CAN bus
Scenario: Turn On

User Interface:
- pressOnOffButton()
- toggleOnOff()

Vehicle Controller:
- isOperational()

All Systems:
- bool isOperational

Driver:
- pressOnOffButton()
- toggleOnOff()

All Systems:
- bool isOperational

Scenario: Set Following Gap

User Interface:
- pressSetFollowingGap(level)

Vehicle Controller:
- setFollowingGap(level)

Brake By Wire:
- setMagnitude(value)

Condition:
- targetObject.vector.distance == level
- setMagnitude(value)
Scenario: Platoon Requested

Vehicle Controller: controller

Message Communication radioCom

User Interface: interface

getIncomingMessage()

message = (PlatoonRequest, car_ID)

displayMessage(platoon_request_message, False)

driver_response

@driver_response == True:

sendMessage(accept_platoon, car_id)

platoon_response

@platoon_response == True:

isPlatooning = true

Scenario: Broadcast Alert

Vehicle Controller: controller

Radar Sensor: radarSensor

User Interface: interface

DSRC Radio Communication radioCom

GetRadarData
data

processData(data)

broadcastAlert()

displayEmergencyWarning()
State Diagrams

[Diagram showing state transitions and conditions for different scenarios in a vehicle control system, including states like Turn On, Turn Off, Ignition Off, New Error Detected, ACC, CACC, and their respective substates and transitions based on vehicle detection and platooning status.]