NGSIM Data Models

For Highway Traffic Management

By: Monil Patel

What is NGSIM

- Next Generation Simulation program
- Collect vehicle trajectory data from different roadways
  - US101 Southbound, Lankershim Boulevard ⇒ Los Angeles, CA
  - I-80 ⇒ Emeryville, CA
  - Peachtree Street ⇒ Atlanta, GA
- Use data to create behavioural algorithms for modeling and simulating driving
- Need for accurate data representing multimodal traveler interactions and freeway systems

Data Collection

- Utilized synchronized video captured footage with multiple cameras on the roadway line
- Custom video processing software was applied to the raw footage to calculate features from footage
  - Speed, Vehicle, Direction etc.
- Data was collected from the four different locations at different, but equal time periods.
  - I80: April 13, 2005 | 3 15 minute videos of 4-4:15, 5-5:15, 5:15-5:30

Data Features

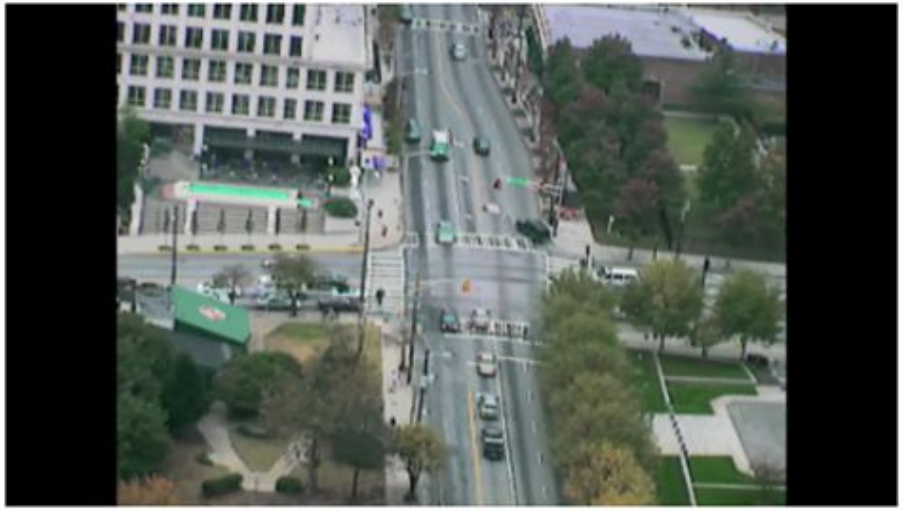
- Vehicle ID
- Global Time
- Location (X, Y)
- Lane ID
- Direction
- Speed
- Vehicle class, velocity, acceleration
- Entry, Exit zone, current section in highway
- Car in front, behind, space in front, time in front

Supplementary Data

- Detector/Tube data for occupancy
- Location images with ortho-rectified photos, CAD drawings
- Signal/ramp meter timings
- Geographic information system
- Data Analysis Files
- Raw and processed video footage

For the I80:

- Weather data from San Francisco Airport
- Signage photographs



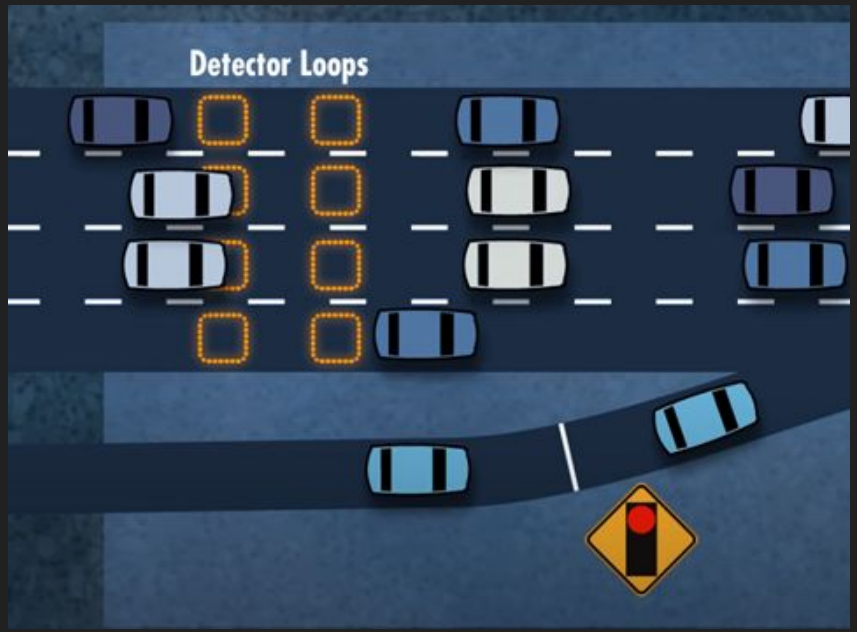

510

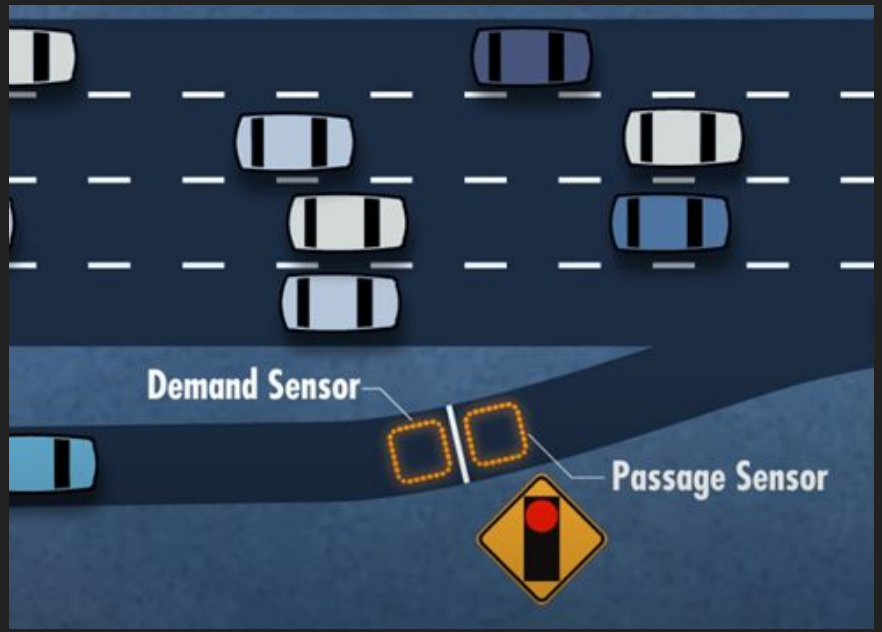
Highway Traffic Management

- Methods applied on regional networks of motorways, freeways, expressways, and other arterial roads
- Aim to enhance or increase capacity and/or stabilise traffic flow
  - Prevent stop/start conditions
  - Increase traffic safety
- Can use multiple systems to help monitor and influence traffic conditions
  - CCTV, Sensors, Time-of-day

Ramp metering

- Involves the monitoring of sensors on the highway and the merging ramp.
- Calculate the gap between passing cars to allow a car to merge or not
  - Achieve an even distribution
- Merging ramp is controlled by a traffic light to control cars proceeding
  - Sensor to check if car is waiting, sensor to check if car has passed
  - Sensor to check if queue is full ⇒ release all cars
- Increases safety by creating safe merging conditions for drivers
  - Increases traffic flow, reduces collisions
  - Can cause delay for drivers by not merging immediately


Detector Loops


Demand Sensor
Passage Sensor

Data from Analysis Report

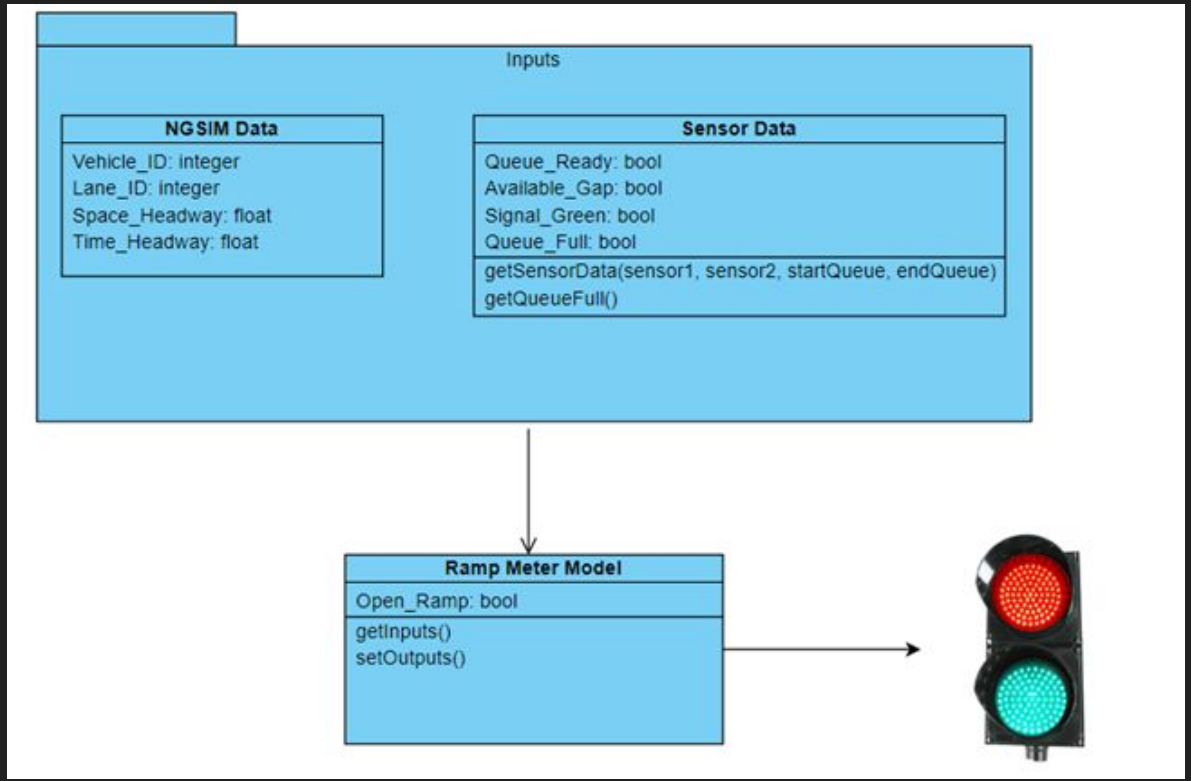
US101 Data Analysis Report

Time: 8:20 - 8:35

Table 5. Input-Output Analysis by Lane and Time Period

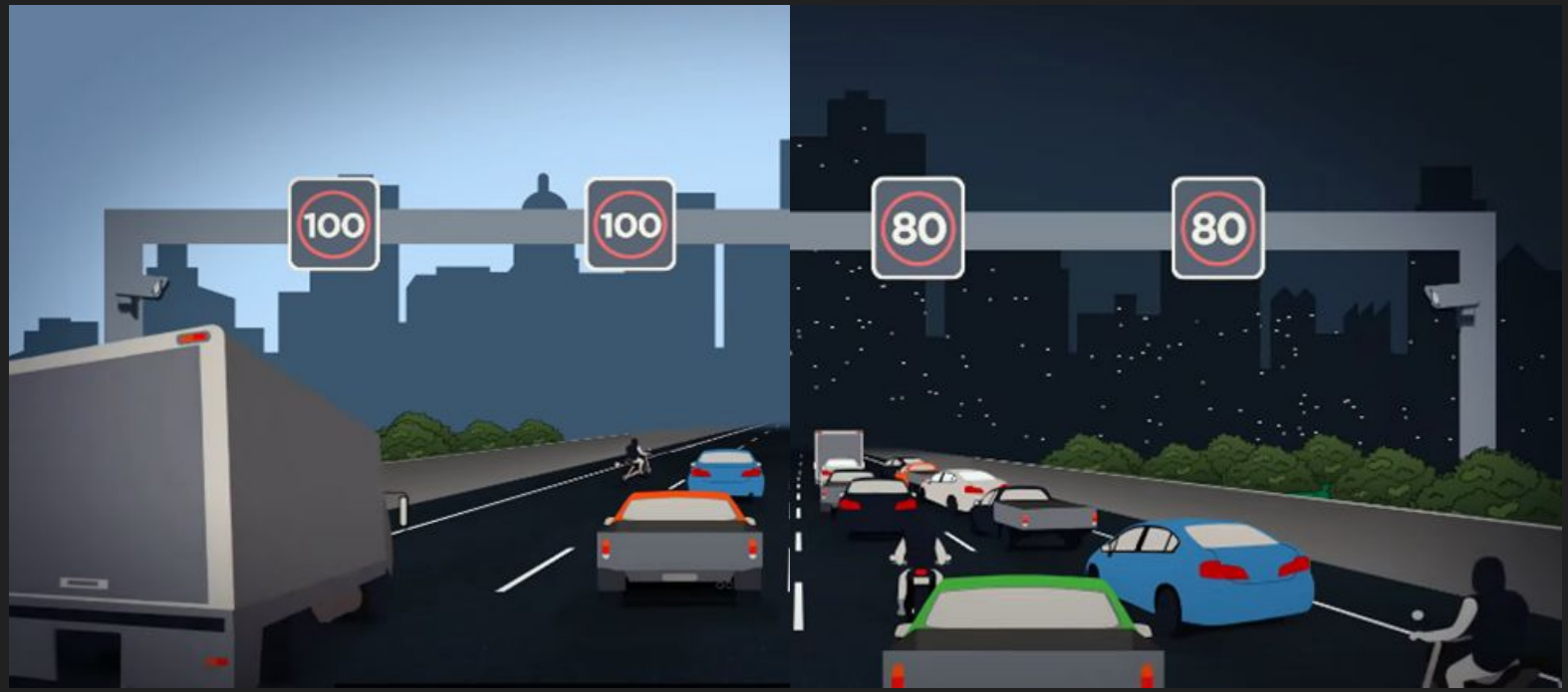
		Time Period (p.m.)					Sum (Vehicles)
		8:19:17 – 8:20	8:20 - 8:25	8:25 - 8:30	8:30 - 8:35	8:35 - 8:35:53	
Vehicles Entering (Vehicles)	Lane 1	7	121	105	109	0	342
	Lane 2	6	124	118	121	0	369
	Lane 3	6	122	109	104	0	341
	Lane 4	7	116	114	110	0	347
	Lane 5	7	124	116	122	0	369
	On-Ramp	2	46	52	47	0	147
	Sum	**35**	**653**	**614**	**613**	**0**	**1,915**
Vehicles Exiting (Vehicles)	Lane 1	0	119	108	111	7	345
	Lane 2	0	128	122	128	8	386
	Lane 3	0	121	113	117	10	361
	Lane 4	0	108	120	119	8	355
	Lane 5	0	122	119	138	8	387
	Off-Ramp	0	34	25	21	1	81
	Sum	**0**	**632**	**607**	**634**	**42**	**1,915**

Ramp Meter Data Model

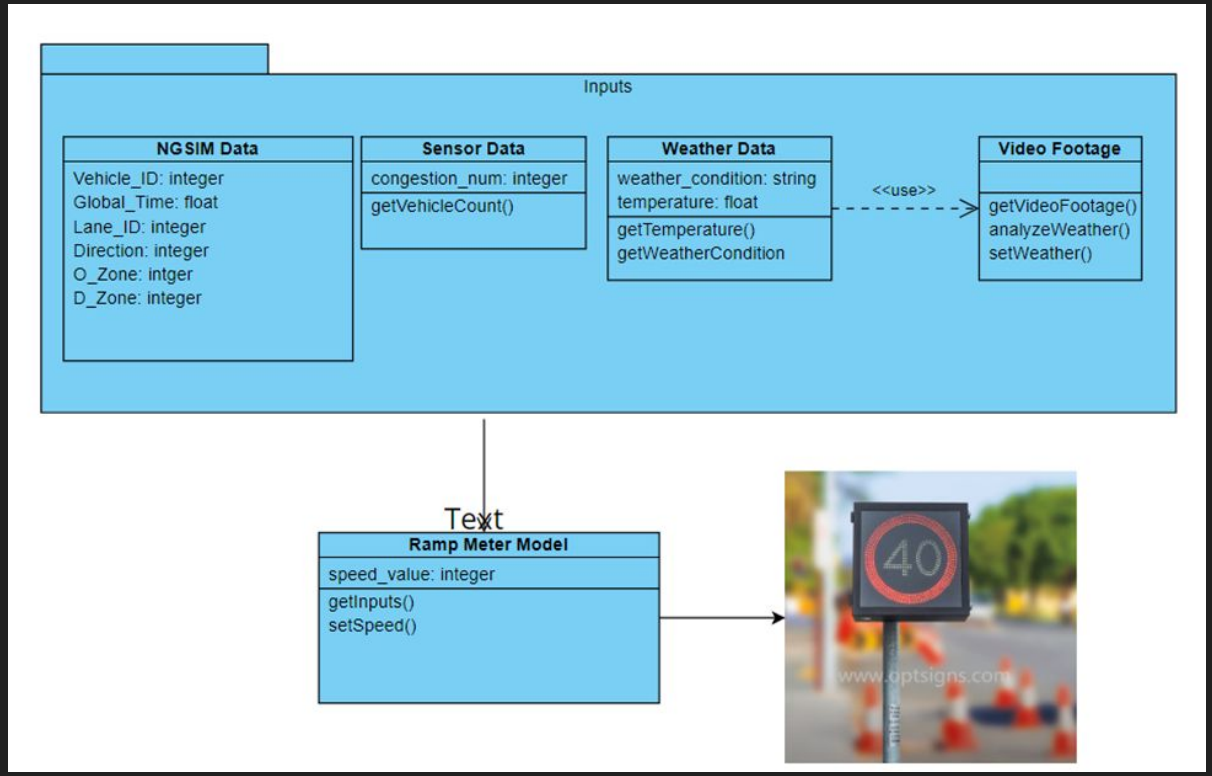


Dynamic Speed Management

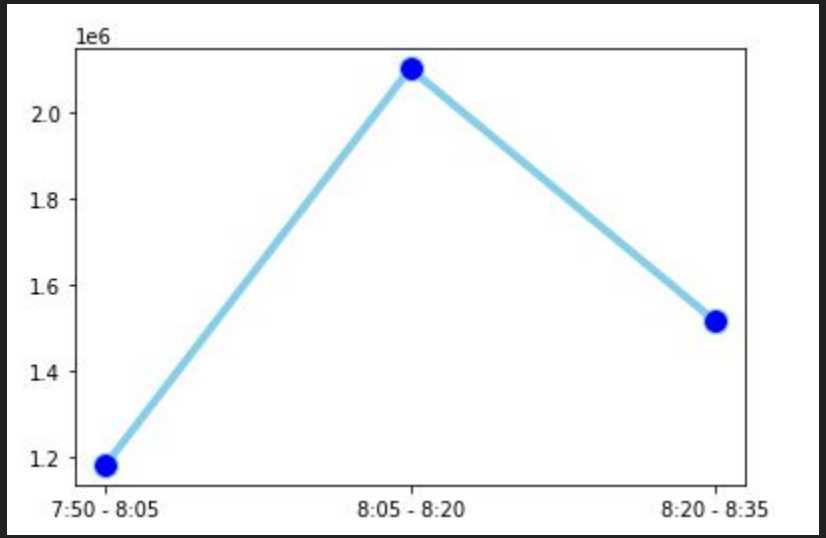
- Dynamic change of current highway speed given environment conditions
- Utilize lane detectors, sensors, and CCTV footage to monitor and manipulate speed limit
- Tries to create best outcome for all drivers
  - Less cars ⇒ speed can be greater
  - Rainy condition ⇒ lower speed is safer
  - Night-time driving ⇒ moderate speed


100
100
80
80

Data Model

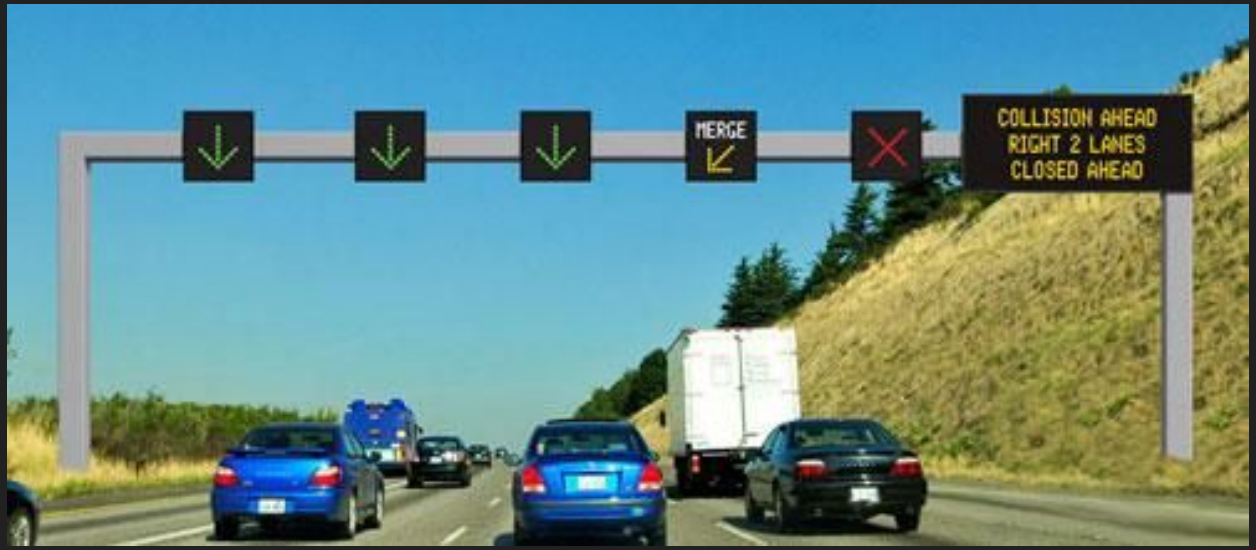

Inputs
NGSIM Data
Vehicle_ID: integer
Global_Time: float
Lane_ID: integer
Direction: integer
O_Zone: intger
D_Zone: integer
Sensor Data
congestion_num: integer
getVehicleCount()
Weather Data
weather_condition: string
temperature: float
getTemperature()
getWeatherCondition
<<use>>
Video Footage
getVideoFootage()
analyzeWeather()
setWeather()
Text
Ramp Meter Model
speed_value: integer
getInputs()
setSpeed()
40
www.optsigns.com

Example with data

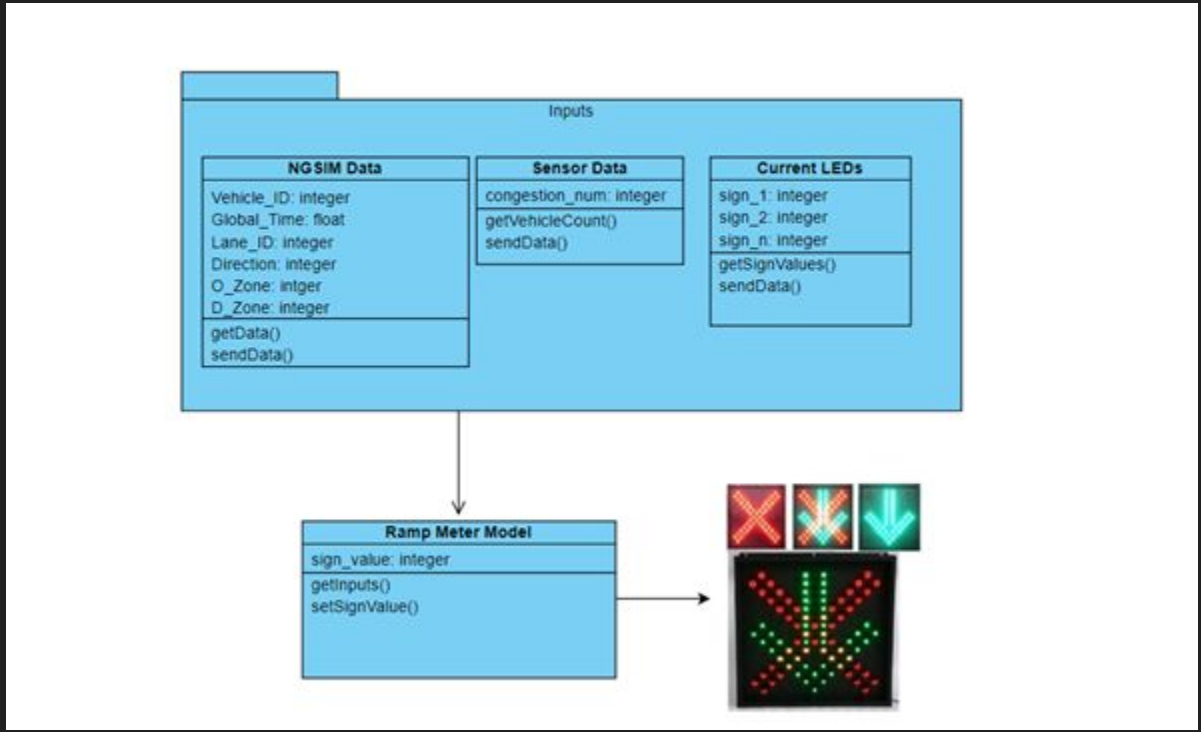


Dynamic Lane Management

- Similar to dynamic speed management, we monitor what lanes are needed for each direction of traffic
- A reversible lane is controlled to be utilized by either side of traffic
  - With LED sign or movable barrier
- Beneficial to special vehicles
  - Heavy goods vehicle
  - High occupancy vehicles
  - Emergency medical service, law enforcement


MERGE
COLLISION AHEAD
RIGHT 2 LANES
CLOSED AHEAD

Data Model


Inputs
NGSIM Data
Vehicle_ID: integer
Global_Time: float
Lane_ID: integer
Direction: integer
O_Zone: intger
D_Zone: integer
getData()
sendData()
Sensor Data
congestion_num: integer
getVehicleCount()
sendData()
Current LEDs
sign_1: integer
sign_2: integer
sign_n: integer
getSignValues()
sendData()
Ramp Meter Model
sign_value: integer
getInputs()
setSignValue()

Example With NGSIM Data

- US101 Data segmented in 3 15min consecutive groups

Vehicle Type	7:50	8:05	8:20
Auto	1141718	2055900	1482316
Truck	26689	40402	32516
Motorcycle	15016	5650	2726
Total	1183423	2101952	1517558

- Potential use at from 8:05 to 8:20, use a HGV lane

Conclusion

- NGSIM data is robust, and there is analysis that can be performed to provide consistent results
- With robust models, the data can be beneficial in creating algorithms that understand traffic behaviour
- With advancements in technology, the data collection can be replicated and made much more accurate
- The NGSIM data can effectively be utilized to implement various highway traffic management systems

Thank you for listening