TRAFFICSENSE: Intersection Simulation

Bryan Healy, Justin Maggio, Samara Reddy,
Eilif Mikkelsen Gaitz Soponski, Joe Curcie

Introduction

People spend a significant amount of their lives in cars. Much of this can be spent in traffic resulting from backed up intersections. Frustrating situations such as pulling up to an empty intersection and the traffic light turning red can arise from outdated and inefficient designs. The aim of the TrafficSense team is to ameliorate these situations by building a simulation environment that can simulate smart traffic light systems. These devices can use sensors such as induction loops or cameras to detect cars approaching an intersection. Based on this data, there is an algorithm in place that will move the vehicles through the intersection in the most efficient manner. Over time, the feedback that the signals receive enables them to be more accurate and rely on historical data to identify traffic trends for further research.

Customer Requirements

Category	Description
Primary problem	Reduce emissions by reduction of acceleration at intersections
Use scenarios	Traffic system must be suitable for most common intersections in the U.S.
Tech	Model must be compatible across several relevant tech platforms
Tech	Model must simulate all lanes of the given intersection
Primary problem	Reduce idling time at intersections

Engineering Requirements

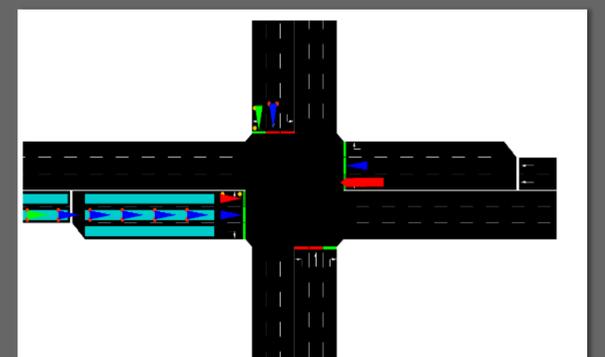| Description	Initial Value
Smart Traffic Light must reduce emissions	20% emission reduction
Tool must support common intersections	80% of all intersections
Must be compatible with common technology platforms	Tools should be cross-platform between Windows, Mac, and Linux
Must Simulate all lanes of a given intersection (pass/fail)	Pass

Simulation Environment

The team used Simulation of Urban Mobility (SUMO) to design the intersection and model traffic flow. This environment was chosen due to its robust capabilities and accessibility. SUMO is a free software that can generate customizable intersections and provide tools that help generate traffic within the network based on given parameters. What makes SUMO highly valuable to the team is its high level of customization both in terms of creation and analysis.

TS_CONFIG

A general cross intersection was generated that had three inbound and outbound lanes going in each direction. The generation of this network required three separate files: the edge file, node file, and connection file.. The connection file details the possible lane that a vehicle in a given lane can turn into as shown below.

Features of Our Design

- Lane-area detection: Simulates camera sensors.
- Easily configurable
- Emissions report
- User-friendly Excel input
- Intersection Customization
- Traffic Customization

TS_RUNNER

Once the configuration files are created based on the user's specifications, the runtime program is started from the command prompt and is passed the location of the configuration files. The runtime program then starts the SUMO simulation using the standard interface, TraCI. The simulation can be ran with or without the SUMO GUI visible. Data collection occurs in this phase.

TS_OPTIMISER

The optimizer is trained after every tick of the simulation by the runtime program. To facilitate the data collection in ts_runner storage and manipulation requirements of a machine learning traffic optimizer, the Rolodex class was created. The Rolodex provides an interface for the user to easily collect data for any object within the simulation which is then used in the optimizer.
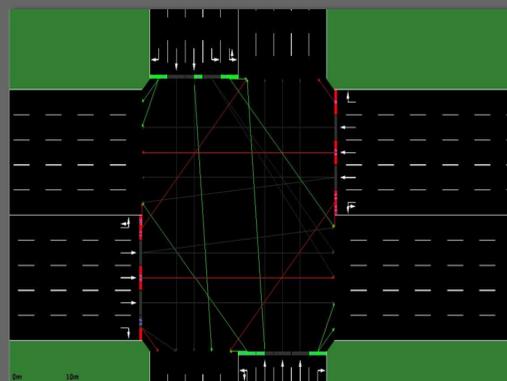
Conclusions

This project will have a wide range of benefits including reducing emissions, lowering time that drivers are on the road, and reducing waiting at traffic lights. This project lays down the foundation for future teams to build upon. Some features to be included are lane splits, lane area detectors, and different types of intersections.

R·I·T

MULTIDISCIPLINARY SENIOR DESIGN