Autonomous Car Intersection Simulation Project

Ivan Naumovski, inau@itu.dk
Martino Secchi, msec@itu.dk

supervised by
Prof. Kasper Støy

IT University of Copenhagen

December 3, 2016

Contents

Abstract

This project aims to investigate some of the aspects involving traffic management with autonomous vehicles. In particular, we want to know how it can be possible to regulate self driving cars through an intersection, and if it is possible to achieve this without a centralised controller and the use of communication. We will run a simulation of an intersection with self driving cars and discuss our findings, comparing different methodologies and related work.

1 Introduction

A presentation of previous research (related work), different paragdigms of control (basic theory), and our angle.

2 Related work

Most of the research that has been conducted on intersection management systems for autonomous vehicles tends to prefer centralised systems that can handle traffic requests. Particularly relevant is the research of two major groups in this area: the researchers at the University of Texas at Austin, and an international group of researchers composed by members of the Massachusetts Institute of Technology (MIT), the Swiss Institute of Technology (ETHZ), and the Italian National Research Council (CNR).

At the University of Texas, Kurt Dresner and Peter Stone developed AIM, Autonomous Intersection Management, a reservation-based system built around a detailed communication protocol able to coordinate movement of self driving cars through intersections [1]. Through a simulation they are able to demonstrate the potential of this system to outperform current intersection control mechanism: traffic lights and stop signs. In the simulation, the intersection center is divided into a n x n grid of reservation tiles. Through a "first come, first served" policy, approaching vehicles make a request to the system to reserve the space-time they need to cross the intersection. Their trajectory is then computed by the system and if the requesting vehicle at any time occupies a reservation tile that is already in use, the request is rejected. The vehicle will then continue requesting until it can pass.

A similar approach is also followed by the international group of MIT, ETHZ and CNR researchers. They developed a centralised slot-based intersection [2]. In their simulations, cars adjust their velocities in approaching the intersection in order to arrive and cross at a given slot of time that is made available for them. This system also involves communication between vehicles and a centrlised controller.

3 Theory

Reactive vs. deliberate vs. hybrid.

maybe theory makes more sense before related work

The theory we have touched upon is from multiple fields of engineering. One field is within traffic engineering, namely intersection management. The other is within the field of robotics, namely robot control.

We use the term agents for robots or other actors moving within an environment.

We use the term environment as being synonimous to the world in which the agent is confined. The following will contain short presentations of the three canonical ways of doing robot control.

3.1 Reactive Control

This paragdim is also coined Sense-Act. This is due to the fact that it can be explained as simply as sensing the environment and reacting instantly to the registered values. Consider a simple robot with only one distance sensor in the front and the goal of avoiding collisions. While there is nothing infront of it for a certain distance it moves forward, else it does not.

3.2 Deliberate Control

This paragdim is also known as Sense-Plan-Act. Compared to the reactive control it introduces a layer of planning. This however requires alot more information about the environment. Any change to the environment would also invalidate the current plan as this is not factored in. This is computationally heavy since it senses, plans a step, executes the step and starts over. A plan can consist of multiple steps which are required to reach a goal. Consider a simple robot as the one previously mentioned needing to go from point A to B. While its tempting to just point it in the direction and hope it succeeds, any obstacle in the way would make this difficult. However translating the goal to multiple steps of moving in a direction, and handling the case of being unable to perform a step by invalidating the current plan and recalculating it, makes this approach better suited for complex tasks.

3.3 Hybrid Control

This paragdigm is a hybrid of the two previous control paragdigms. It can be explained as decomposing a complex goal into sub tasks. Lets say a robot needing to pickup a object and moving it to a position. It has one behaviour for gripping the object once the sensor has identified it, and another for actually going from A to B.

4 System description

- Presentation of our model.
- Intersection w/wo traffic lights. Cars going straight vs. cars turning.
- Car model and sensor modelling (sick vs simplified directional).
- Graph for navigation.
- Reactive controller.
- Special rules (right hand).
- Sensor range limitation based on braking distance considerations.
- SimScale to RealWorldScale

4.1 The World Model

As observed in the real world, multiple intersection models exist, three lane and 4 lane intersections are the most common ones. The style can be either with the roundabout approach, which essentially rules out the left turn complication or a traditional 4 lane intersection where going left crosses the opposing traffics lane.

In our model we mainly focus on a single lane 4 way intersection.

insert simple intersection images - maybe graph images from unity

We represent these intersection models using an underlying directed graph. A car agent in our environment is seeded with information about its origin and its destination. These points are vertices in our graph. Every combination of origin and destination pair has one path.

While this does not provide alot of flexibility in straying from the path, the simplicity makes the model really easy to implement.

4.1.1 The traffic lights

This is an attempt to mimic traffic light regulated intersections. The lights change at intervals chosen in regards to our experiences with intersections. These values can vary alot from intersection to intersection - and are highly influenced by traffic flow during the day. We settled on 20 second windows.

4.2 The Car Model

The way the car has been modelled is using a interface which essentially contains actions such as accellerating or decellerating, braking, and turning left or right. This abstraction simplifies the step of going from a virtual model to a physical one with robot actors. This interface is the used by the controller. This seperation makes it simple to develop multiple types of controllers.

5 Experiments

we have some fixed parameters such as light control on or off. Mimics intersection lights.
Whether simulated vehicles are able to turn or not.
We have different max speeds - 60 or 40 km per hour approach. This translates to lower speeds during turns (roughly half).

tests are different combinations of above parameters. And results are measured in collisions, cars spawned, cars reaching destination, and time before deadlock.
Deadlock is the situation where cars have been stuck for a period of time in the 'intersection zone' without entering or leaving.

6 Discussion

Discuss experimental results. Shortcomings of model. Improvements for further tests.

7 Conclusion

References

[1] K. Dresner and P. Stone. A multiagent approach to autonomous intersection management. 1905.

[2] R. Tachet, P. Santi, S. Sobolevsky, L. I. Reyes-Castro, E. Frazzoli, D. Helbing, and C. Ratti. Revisiting street intersections using slot-based systems. 1905.