Safety Control for Driving on Forbes Avenue

Yifei Yang(yifeiy3@andrew.cmu.edu, section 824A)

December 3, 2021

Contents

Abstract

The project investigates the driving control system for autonomous vehicles under a specific situation: my daily commute down Forbes Avenue to school. Specifically, the project implements two control models under differential logic and hybrid systems that respectively reflects the driving behaviors of a my car at the start of the commute journey, where I need to make a left turn onto Forbes Avenue, and near the end of my commute journey, where I need to lane switch to stay out of the left turn lane, under simplified conditions. Then, the implemented control systems are formally verified using KeYmaeraX tool to prove their safety. While the motivation behind the project is simple and specific to my own interest, the implemented control systems could be generalized and applied to autonomous vehicle control systems, for which proving their safety is one of the most interested field of research in recent years.

Figure 1: Project Teaser Image

1 Introduction

Driving is difficult, driving in Pittsburgh is even more difficult. The project is inspired by my daily commute from my apartment to school, where I would need to one of the most crowded street in Pittsburgh: Forbes Ave. Frustrated by the amount of traffic and complicated driving situations on the street, the project designs and formally verifies a control system for an autonomous driving vehicle for my daily commute under simplified assumptions, which hopefully I can rely on to take me to school safely in the near future.

Due to the limitation in time, the project could not implement a control system for driving behavior of Forbes Avenue in its entirety. Instead, the project focuses on the two specific points that I personally found difficult to control on my daily commute: Turning onto Forbes Ave and lane switching to not go onto the left turn lane. The physical location of each point is shown on the path of my commute in Figure 2.

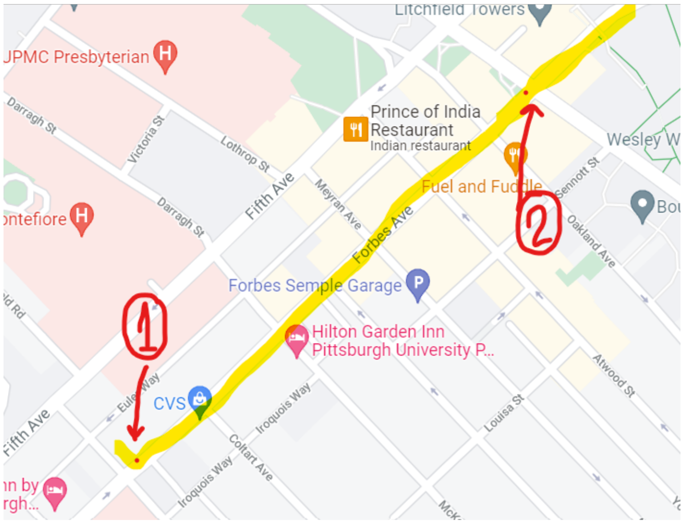

Figure 2: Part of my daily communte on Forbes Ave, labeled in yellow

For each point, the system and vehicle control is modeled under simplified assumption with Differential Logic (dL) and Hybrid Systems (HS), which it is able to be formally verified for safety with the KeYmaeraX tool. For the rest of the paper, section 2 first compares the project to the related works in the autonomous vehicle control field. Then, section 3 and 4 formally defines and proves the model for point 1, and section 5 and 6 for point 2. Finally, the paper discusses the conclusions and future works in regards to the project in section 7 and summarizes the deliverables of the project in section 8.

2 Related Works

Verifying and Validating autonomous vehicle control systems has been an major issue of CPS research in the recent decade. [4] sets up the foundation for autonomous vehicle control by introducing the main components of control being speed, distance, lane change, emergency stop, and collision avoidance, all of which are considered in our model for both points of interest. Work in [5] describes a non-conservatively defensive control strategy for autonomous driving in urban situations similar to our case driving down Forbes Ave through a logistic regression model on past driving behavior data. There are also other multiple control designs based on machine learning algorithms such as in [2] with a Hidden Markov Model and in [3] through reinforcement learning. These models trained through machine learning, however, are based on probabilistic models and are difficult if not impossible to prove completely safe, which our formally verified control system is able to guarantee.

Similar efforts in formally verifying control systems for driving environments have been experimented, with [1] utilizing dL, HS, and KeYmaeraX in formally verifying a left turn assist control and deriving static constraints for safety. However, the driving situation discussed in [1] is different to the two driving situations that our current project analyzes, and [1] focuses on keYmaeraX's ability in deriving static constraints rather than on developing and formally verifying the safety of a vehicle control system that our project does.

3 Formal Model for Point 1

From the illustration of my daily commute in Figure 2, point 1 captures the moment where the vehicle would need to leave the garage and turns onto Forbes Ave with a left turn. Since Forbes Ave's leftmost lane is packed with left turning cars that always get blocked by pedastrian crossing, turning into the left lane is never an efficient choice. As a result, the car would want to make a turn into the middle lane to start the commute journey.

In order to safely turn onto Forbes Ave, the vehicle would need to watch for incoming traffic before making the turn. To simplify the model, incoming cars are assumed driving at a constant speed, and the turn is done onto the center lane with a 90^o counterclockwise circular turn. The cars are treated as infinitesimal points at their center of masses, and collision safety is defined by cars staying at least of safety buffer distance apart. Once the car has finished the turning motion, the car is assumed to continue drive straight with constant velocity and the rest of the environment become irrelevant. The lanes are assumed to be of the same width $2l$ and the initial center of mass for the car in the garage is assumed to be at l away from the left most lane. Finally, the system representing the car's turning behavior is established on a 2-dimensional grid, with the origin being at the center of the car's rotation from start, which gives the initial position of the car to be at $(4l, 0)$, and the point of finishing the turn to be at $(0, 4l)$.

When the vehicle commits in making the turn, we assumed that it will continue making the turn until finish and can only speed up, that is, it can not have an acceleration less than 0. Since Forbes Ave is a very busy road, we assumed a continuous flow of traffic coming from both lanes with constant velocity $rogv$ in the same direction as the turn that our vehicle should not collide with. We assumed cars driving in each lane is in the middle of the lane; as a result, car $rog1$ in the left most lane would always have y coordinate set to be $2l$ while $rog2$ in the center lane to be $4l$. For simplicity of the model, incoming traffic with position already passed our vehicle is irrelevant in our turning motion. To account for the unavoidable reaction time of control systems, our control model is implemented in a time triggered style. Figure 3 below demonstrates the driving system for the turn under the assumptions listed above.

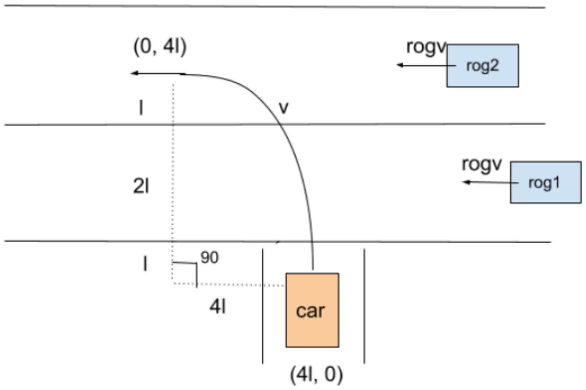

Figure 3: Driving system model at point 1, cars are treated as infinitesimal points

From the description above, we formalize our driving model and control with dL and HS in the sections below. First, we introduce the needed constants and variables, preconditions, and driving dynamics of the system. Then, from the dynamics, we define the desired safety property, from which we describe the intuition for our control system and the derived invariants. Finally, we discuss the impact and implications of our assumptions made in creating the model. The final result of our formalized model can be found in the attached deliverables.

3.1 Constants and Variables

From our system construction above, the constants and variables defined that are needed in the model are listed with their description below:

Constants

- T : Minimum time interval guaranteed between consecutive control triggering
- A : Maximum acceleration allowed for vehicle
- $rogv$: Constant velocity for incoming traffic
- l: Distance between center to side of lanes
- $buffer$: Required distance between cars to avoid collision

Variables

- x : Position of our vehicle at x direction
- y : Position of our vehicle at y direction
- $rogx1$: Position of incoming vehicle on left lane in x direction
- $rogx2$: Position of incoming vehicle on center lane in x direction
- v : Linear velocity of our vehicle
- a : Linear acceleration of our vehicle
- t : Time for control

3.2 Preconditions

Before executing our vehicle controller, the following preconditions must be met:

- $A > 0, T > 0, l > 0, buffer > 0$ by their definition
- vehicle starts at position $(4l, 0)$
- $rogv < 0$, since incoming traffic is traveling towards the negative x direction
- $v = 0$, our vehicle should be at rest to prepare for turning motion.
- $rogx1 >= 4l$ & $rogx2 >= 4l$, by our assumption that incoming traffic with position already passed our vehicle is irrelevant in our turning motion.
- $buffer < 2 * l$, so that our vehicle is not in danger for collision in its initial position and with vehicles in other lanes.

3.3 Dynamics

As described in our model setup above, our vehicle is able to accelerate with $0 \leq a \leq A$ when entering into the turn. Since by our assumption, incoming traffics' position becomes irrelevant if they exceed our vehicle's current position, we reset the position of the incoming traffic once the condition $rogx1 < x$ or $rogx2 < x$ is met (since we are traveling towards the negative direction). This can be seen as the control system is looking at the incoming car behind the car that just passed our vehicle. Our system models this behavior with the following statements:

```
{?(rogx1 < x); rogx1:=*; ?(rogx1 >= x); ++ ?(rogx1 >= x);}
{?(rogx2 < x); rogx2:=*; ?(rogx2 >= x); ++ ?(rogx2 >= x);}
a := *; ?(0<=a & a<=A);
```

Two different motion occurs for our vehicle during the turn and after the turn. When our vehicle decides to commit into the making the turn, the vehicle would follow the counterclockwise rotational motion with velocity v, acceleration a, while incoming traffic would continue travel at the constant velocity $rogv$. Since by our construction, the vehicle would finish the turn at $x = 0$, our control system would need to make sure $x \geq 0$ during this motion. This combined with our control time constraint and vehicle should only be going forward in the turn gives us the following continuous evolution motion defined below:

```
{v' = a, x'=-y*v/(4l), y'=v*x/(4l), t'=1, rogx1'=rogv,rogx2'=rogv
    & v>=0 & t<=T & x>=0}
```

When vehicle completes the turn, we know that $x <= 0$. By our assumption it will continue travel toward the negative direction with velocity v. Since we set the rest of the environment as irrelevant after the turn, our continuous evolution motion can be defined as following

```
{x'=-v, t'=1 & v>=0 & t<=T & x<=0}
```

3.4 Safety Property

Since our control system is specifically for our vehicle during the turning motion and the environment after completing the turn is seen as irrelevant by our construction, our safety condition is also specifically for during the turn. During the turn, we first know that $x >= 0$, and for the vehicle to be safe, it would need to have the following properties:

- Its motion follows the counterclockwise turning track, that is, $x^2 + y^2 = (4l)^2$ $(eqn1)$
- It is safe from collision with the vehicle on the left lane. This can be guaranteed if our vehicle is at least $buffer$ distance away from the left lane traffic on x or y direction, giving us $max(abs(x - rogx1), abs(2 * l - y) > buffer$ $(eqn2)$
- It is safe from collision with the vehicle on the center lane. Similar to the left lane, we have $max(abs(x - rogx1), abs(4 * l - y) > buffer$ $(eqn3)$

Combining the three points above gives us our desired safety property

```
x>=0 -> (eqn1) & (eqn2) & (eqn3)
```

3.5 Control System Intuition

At our vehicle's initial point of entry, it is possible for the vehicle to make two decisions: Continue waiting or committing the turn. If our vehicle decide to continue waiting, it is always safe by our construction. Since our vehicle has an initial velocity of 0, the behavior for waiting can be defined as having an acceleration $a = 0$. We would also need to make sure that after the vehicle enters the turn (i.e. $y > 0$), it can not choose to wait again in the future control cycles. Thus, the condition where control decision for waiting is available can be captured by the expression:

```
y = 0 & x = 4*l & v = 0 & a <= 0
```

When vehicle commits the turn or vehicle is currently in turning motion, our control would need to make sure that after traveling with our picked acceleration a for $t \leq T$ seconds for the next control cycle, our vehicle is able to complete the turn with maximum acceleration $a = A$ before the incoming traffic collides with it by coming into $buffer$ range.

To do so, we would first need to know the time it would take for our vehicle to finish the remaining of the turn at its current condition. For an vehicle at point (x, y) on the turn, we first note the remaining distance for the turn can be upper bound by $abs(4l - x) + abs(y - 0) = 4l - y + x$. This property is a direct result of triangle inequality of convex sets, as shown in Figure 4 below. Furthermore, since we know the distance between the vehicle and our finish point $(0, 4l)$ can not be less than 0, we know that during the turning motion, $4l - y + x \geq 0$ for all coordinates (x, y) along the turn.

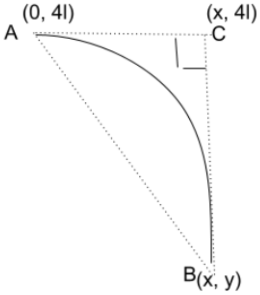

Figure 4: Upperbounding remaining distance, the length of path $|AB_{arc}|$ is bounded by the convex set defined by triangle ABC, giving that $|AC| + |CB| \geq |AB_{arc}|$

Now, let d be the maximum distance our vehicle able to travel in our current control cycle with current $v0$ and a. Then, since $a <= A$, the longest time needed for our vehicle to complete the turn by our construction would be traveling with a for T seconds in the current control cycle before switching to $a = A$. In this case, $v = v0 + aT$.

From this result, we know that the remaining time t' needed to complete the turn after switching to $a = A$ would be upperbounded by the time needed for our vehicle to travel $4l - y + x - d$ in translational motion with velocity v and acceleration A. Since $d \geq 0$, we know that t' is bounded by t, the time needed for our vehicle to travel $4l - y + x$. From translational kinematics, we get:

$$(v0 + aT)t + \frac{At^2}{2} = 4l - y + x \tag{1}$$

And solving for t in (1) with quadratic formula gives us

$$t = \frac{-(v0 + aT) + \sqrt{(v0 + aT)^2 + 2A(4l - y + x)}}{A} \tag{2}$$

Now, noticing that $v0 + aT \geq 0$ by our construction, we know that

$$-(v0 + aT) + \sqrt{(v0 + aT)^2 + 2A(4l - y + x)} <= \sqrt{2A(4l - y + x)} \tag{3}$$

from the inequality $\sqrt{a^2 + b} <= a + \sqrt{b}$ with $a \geq 0$. Giving us our final bound for t' to be

$$\frac{\sqrt{2A(4l - y + x)}}{A} \tag{4}$$

In order for the incoming traffic to be at least our safety $buffer$ distance away from our vehicle for the duration of the turn, it is safe to have our incoming traffic to be always at least $buffer$ away from our vehicle's starting x coordinate $4l$. Then, by our formulation above, our control would need to make sure

$$rogx1 + rogv * (T + \frac{\sqrt{2A(4l - y + x)}}{A}) > 4l + buffer \tag{5}$$

and

$$rogx2 + rogv * (T + \frac{\sqrt{2A(4l - y + x)}}{A}) > 4l + buffer \tag{6}$$

when we are committing the turn. We note that by our formulation, our vehicle control would never get stuck during the turn, as always choosing $a = A$ at any point would always make sure our vehicle complete the turn before incoming traffic get within $buffer$ range.

To summarize, the condition available for our vehicle to have the choice to commit or continue with the turn would be

```
4*l - y + x >= 0
```

and we can only make the decision to commit the turn when

```
rogx1 + rogv*(T + (2A(4l-y+x))^(0.5)/A) > 4l + buffer
rogx2 + rogv*(T + (2A(4l-y+x))^(0.5)/A) > 4l + buffer
```

Our final control option for the vehicle happens when it completes the turn, which nothing needs to be done by our construction of considering the environment after the turn as irrelevant. Since y would remain at $4l$ and x would continue to travel in the negative direction in a straight line motion,the condition where control decision for after vehicle completing the turn is available can be captured by the expression:

```
4*l - y + x < 0
```

3.6 Derived Invariants

From our driving system and control construction, we are able to derive the following invariants at the beginning and after each control cycle:

- $x \leq 4l$ & $y \geq 0$ by our constructed 90^o rotational motion
- $x \leq 0 \implies y = 4l$ since our vehicle simply continues straight after completing the turn
- $x \geq 0 \implies x^2 + y^2 = (4l)^2$ to prove our safety condition that our vehicle always stays on the rotational track during the turn.
- $x \geq 0 \implies (cond1)|(cond2)$, where $cond1$ represents the condition for our vehicle is still waiting for entering the turn, giving us $y = 0$, $cond2$ represents the condition for our vehicle during the turn defined as $(sub1)$ & $(sub2)$ & $(sub3)$, where each sub corresponds to the control condition stated above, with $sub1 = 4l - y + x \geq 0$, $sub2 = rogx1 + rogv * (\frac{\sqrt{2A(4l-y+x)}}{A}) > 4l + buffer$, and $sub3 = rogx2 + rogv * (\frac{\sqrt{2A(4l-y+x)}}{A}) > 4l + buffer$. $(cond2)$ and $(cond3)$ makes sure that our vehicle is always safe from collision of the incoming traffic during the turn when picking acceleration A at future control cycles.

We note that safety property defined by $(eqn1)$ is trivially implied by the third invariant above, and $(eqn2)$ and $(eqn3)$ can be easily derived by the fourth invariant, since $rogv < 0$ and $A > 0$ would imply that $rogx1$ and $rogx2$ would never come into $buffer$ range within our vehicle's starting point.

3.7 Discussion of Assumptions

To derive the model, we used many oversimplifying assumptions to the driving behavior that are often not true in real life; however, the assumptions made in the model are fairly reasonable behaviors under normal driving situations. For example, cars are often traveling at a close to constant speed on the road such as following the speed limit when there is no traffic and a typical driver would not brake in the middle of the turn and prefer turning into the center lane directly rather than turning into the left lane and perform a lane switch. Since our model is only interested in the specific point of turning onto the lane, it also makes sense for the model to not consider the driving behavior after the turning motion is complete in our context.

The assumptions regarding the vehicle creating a perfect 90 degree turn, vehicles in front of our car's position are irrelevant in the control decisions, incoming vehicles traveling at the same speed, and vehicle's initial position being l away from the street are made in order for simplifying the computation needed to complete the proof due to the constraint of time, as they can add significant complications to our model. Finally, we have realized the difficulty in KeYmaeraX in proving properties regarding square roots, which we are forced to simplify the control constraint derived from quadratic formula with equation (2) and (3) in order to prove our model to be secure at the cost of some efficiency in control.

4 Proving Model for Point 1

By our vehicle's initial starting point $(4l, 0)$, our invariants are trivially proven by our precondition. And our invariant implies the post condition safety properties is discussed in 3.6. What remains to show is that our invariants remain true throughout each control cycle so we can conclude our proof with the loop rule. While the first 3 invariants in 3.6 are easily proven with our keYmaeraX's auto and differential invariant (dI) feature for all branches, the final invariant regarding the control for the turn is the most difficult part of the proof, specifically, we need to prove

```
4l - y + x >= 0&
rogx1 + rogv*((2A(4l-y+x))^(0.5)/A) > 4l + buffer &
rogx2 + rogv*((2A(4l-y+x))^(0.5)/A) > 4l + buffer
```

remains true under the continuous evolution of

```
{v' = a, x'=-y*v/(4l), y'=v*x/(4l), t'=1, rogx1'=rogv,rogx2'=rogv
& v>=0 & t<=T & x>=0}
```

First, we performed a differential cut(dC) on the first 3 invariants to the differential equation, knowing that all three will be easily proven with auto and (dI). Then, by differential weakening(dW), we are able to show $4l - y + x >= 0$ throughout the differential equation with our new evolution constraint $x \geq 0$ & $x \leq 4l$ & $y \geq 0$ & $x^2 + y^2 = (4l)^2$.

What remains to show is the other two conditions to be true throughout the evolution, so we are able to conclude our result with boxAnd rule. Since the two conditions are proven in identical ways, we show the steps taken for proving $rogx1 + rogv * (\frac{\sqrt{2A(4l-y+x)}}{A}) > 4l + buffer$ here.

First, we constructed discrete ghost $x0, y0, rog10$ to represent $x, y, rogx1$ before entering the continuous evolution respectively. Then, knowing $v \geq 0, l > 0$, we are able to perform dC on $x \leq x0$ & $y \geq y0$ and prove the cut statement trivially with dI. By auto, we would also trivially prove that $rogx1 \geq rog10 + rogv * T$ since $rogv < 0$. Now, recall we have our control condition from 3.5

$$rog10 + rogv * (T + \frac{\sqrt{2A(4l - x0 + y0)}}{A}) > 4l + buffer \tag{7}$$

before entering the continuous evolution, we can then prove our invariant condition true by dW.

From (7), we can combine with the domain constraint $rogx1 <= rog10 + rogv * T$ to obtain

$$rogx1 + rogv * \frac{\sqrt{2A(4l - y0 + x0)}}{A}) > 4l + buffer \tag{8}$$

From the domain constraint $x \leq x0$ & $y \geq y0$ we are also able to obtain

$$4l - y0 + x0 \geq 4l - y + x \tag{9}$$

Now, performing a cut operation on (8) and (9) would prove $rogx1 + rogv * (\frac{\sqrt{2A(4l-y+x)}}{A}) > 4l + buffer$ to be true throughout the continuous evolution by auto.

5 Formal Model for Point 2

After successfully turning onto the center lane from point 1, our vehicle can enjoy a downtime cruising down Forbes Avenue until point 2 is reached, where a lane change to the side lane is needed since the center lane becomes left turn only before Pittsburgh's famous Cathedral of learning.

We visualize point 2 as a 2-lane system, where we assumed to have a car in front of us while we have continuous incoming traffic from the side lane that we would need to lane switch into. Similarly to our model in point 1, we assumed the lane remains the same $2l$ width, front and side-lane vehicles traveling at constant speed $rogv$, the vehicles are infinitesimal points that need to stay a $buffer$ distance apart to be safe from collision, and the rest of the environment becomes irrelevant after the lane switch is performed. We also utilizes a coordinate system for our vehicles in motion, with the center of mass for the initial position of the vehicle to be the origin and we would want a x coordinate of $2l$ upon finishing the lane switch motion. Similar to point 1, we assume that the vehicles outside of lane switching motion are driving in the center of the lanes. As a result, the x coordinate would always be at 0 for the front car and $2l$ for the side-lane car.

For simplicity of the model, our vehicle is assumed to be following the front car at constant velocity $v < rogv$ before committing the lane switch motion, and when the vehicle commits in the lane switch, it can not slow down with acceleration $a < 0$. We also assume that side-lane traffic with position already passed our vehicle is irrelevant in our lane switching motion. Finally, when our vehicle commits the lane switch, we assumed the vehicle to be executing the lane switch at a 60 degree angle direction. From the 60 degree angle assumption, we then know the total change in y during the lane switch motion would be $2l * \sqrt{3}$. Similarly to point 1, our control model is implemented in a time triggered style, and a detailed illustration of the driving system for the lane switch is shown in Figure 5 below. The steps to introduce the formalization of our control model with dL and HS follows the same format as point 1 in the sections below. The final result of our formalized model can be found in the attached deliverables.

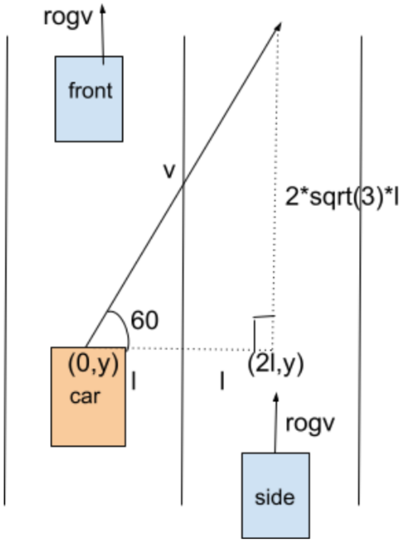

Figure 5: Driving system model at point 2, cars are treated as infinitesimal points

5.1 Constants and Variables

From our system construction above, the constants and variables defined needed for our model are listed with their description below:

Constants

- T: Minimum time interval guaranteed between consecutive control triggering

- A: Maximum acceleration allowed for vehicle
- $rogv$: Constant velocity for front car and side car
- l: Distance between the center to side of lanes
- $buffer$: Required distance between cars to avoid collision

Variables

- x : Position of our vehicle at x direction
- y : Position of our vehicle at y direction
- $sidey$: Position of side-lane vehicle in y direction
- $fronty$: Position of front vehicle in y direction
- v : Linear velocity of our vehicle
- a : Linear acceleration of our vehicle
- t : Time for control

5.2 Preconditons

Before executing our vehicle control, the following preconditions must be met:

- $A > 0, T > 0, l > 0, buffer > 0$ by their definition
- Vehicle starts at position $(0, 0)$
- $rogv > v, v > 0$, since the vehicles can only travel forward and our vehicle initially follows the front vehicle with a lesser velocity.
- $sidey <= y$, by our assumption that side-lane traffic with position already passed our vehicle is irrelevant in our lane switching motion.
- $fronty - y > buffer$, so that our vehicle is initially safe from collision
- $buffer < 2 * l$, so that our vehicle is not in danger from collision with vehicles in other lanes.

5.3 Dynamics

Similar to point 1, by our assumption that that side-lane traffic with position already passed our vehicle is irrelevant in our lane switching motion, we reset the position of the side lane traffic once the condition $sidey > y$ is met, which adds the following statement to the system to model this behavior:

```
{?(sidey > y); sidey :=*; ?(sidey <= y); ++ ?(sidey <= y);}
```

There are three different types of motion that can occur for our vehicle: following the front car, committing for the lane switch, and continue driving after completing the lane switch. When our vehicle follows the front car, there is no change in the x coordinate, and both the front car and our vehicle travels at their set constant velocity. As a result, we have the following continuous evolution motion:

```
{x' = 0, y' = v, sidey' = rogv, fronty' = rogv, t' = 1 & t <= T}
```

During the lane switching motion, the vehicle would be traveling at a 60^o angle direction with acceleration a and velocity v. By the property of the 60^o angle, we then know the x component for the velocity would be $\frac{v}{2}$ and the y component for the velocity would be $\frac{v*\sqrt{3}}{2}$. This, combined with the constant velocity of front and side-lane car and the lane switching motion terminates when the vehicle reaches the center of the side lane with $x = 2l$ gives the following continuous evolution motion:

```
{v' = a, x' = v/2, y' = v*3^(0.5)/2. sidey' = rogv, fronty' = rogv, t' = 1
    & t<=T & v>=0 & x>=0 & x<=2*l}
```

Finally, when our vehicle completes the lane switch, the rest of the environment becomes irrelevant by our assumption, leaving us with our vehicle continuing to move forward with velocity v, giving us

```
{y'=v, t' = 1 & v >= 0 & x >= 2*l & t <= T}
```

5.4 Safety Property

To make sure our control system is safe, our vehicle would need to be safe from collision from both the front car and the side lane car. Since by our precondition $buffer < 2 * l$, we know that vehicle before committing to lane switch will not be impacted by the position of the side lane car, and the vehicle after finishing the lane switch motion will not be impacted by the position of the front car. Since the switching lane motion, by our construction, starts at $x = 0$ and finishes when $x = 2l$, our security condition can be guaranteed by the following:

- Before the vehicle commits or during the lane switching motion, which happens at ($x \geq 0$ & $x < 2l$), the vehicle is always at least $buffer$ distance behind from the front car. This property can be guaranteed with $fronty - y > buffer$ $(eqn4)$ if we only look at the distance between the y position.
- During and immediately after the lane switching motion, which happens at ($x > 0$ & $x \leq 2l$), the vehicle is always at least $buffer$ distance in front of the side-lane car. This property similarly can be guaranteed with $y - sidey > buffer$ $(eqn5)$

Combining the two points above gives us the desired safety property

```
(x >= 0 & x < 2l -> (eqn4)) & (x > 0 & x <= 2*l -> (eqn5))
```

5.5 Control System Intuition

At our vehicle's initial point of entry, it is possible for the vehicle to make two decisions: Continue driving straight down the current lane by following the front vehicle or commit to the lane switch. If our vehicle decide to continue drive straight, it is always safe by our construction since it will be driving at a constant velocity $v < rogv$ for the front vehicle by our construction. Similarly to the control condition in point 1, we would just simply need to make sure that our vehicle can not choose the option to drive straight in the middle of committing the lane switch motion (i.e. $x \neq 0$). Thus, the condition where control decision for continuing driving straight can be captured by the expression

```
    x = 0 & rogv > v
```

When the vehicle commits the lane switch or the vehicle is currently in lane switching motion, our control, by our safety condition above, need to make sure our vehicle is safe from collision with the side-lane car. Similarly to what is done for point 1, it is sufficient for our control to have the following property: After traveling with our picked acceleration a for $t \leq T$ seconds in the next control cycle, our vehicle is able to complete the lane switch with maximum acceleration $a = A$ before the side-lane traffic collides with it by coming within $buffer$ range.

To do so, we would first determine the time it would take for our vehicle to finish the remaining of the lane switch at its current condition. We note that by our construction, the vehicle completes the lane switch motion when $x = 2l$ and $x <= 2l$ during the lane change motion, which makes us able to calculate the remaining time needed by looking at the remaining distance needed to travel with respect to the x direction. Then, for a vehicle at point (x, y) on the lane switch, the remaining distance for the turn with respect to the x direction would be $2l - x$.

Now, similar to our derivation on bounding the remaining time to complete the turn in control for point 1, we first note that the longest time needed to complete the lane switch is for us to traveling with acceleration a for T seconds in the current control cycle before switching to $a = A$, since our current $a \leq A$.

Then, let d be the distance travelled and v_x be the velocity of our vehicle after driving for T seconds in the current control cycle with respect to the x direction. Finally, we note by the 60 degree triangle, our vehicle with acceleration A would have an acceleration $\frac{A}{2}$ in the x direction. Setting t for the remaining time needed to complete the lane switch, we would have

$$v_x t + \frac{At^2}{4} = 2l - x \tag{10}$$

which gives us

$$t = \frac{-v_x + \sqrt{v_x^2 + A(2l - x)}}{\frac{A}{2}} \tag{11}$$

Similarly to our simplification in with equation (3), (4) from point 1, then, we have

$$t \leq \frac{\sqrt{A(2l - x)}}{\frac{A}{2}} \tag{12}$$

Similar to point 1, in order for the side-lane traffic to be at least our safety $buffer$ distance away from our vehicle for the duration of the lane change, it is safe to have our side-lane car to always at least $buffer$ away in the y direction with our vehicle's current y position. Then, by our formulation above, it is sufficient for our control to make sure

$$sidey + rogv * (T + \frac{\sqrt{A(2l - x)}}{\frac{A}{2}}) < y - buffer \tag{13}$$

when we are committing the lane switch.

In order for the lane switch control to be safe according to our listed safety condition, we would also need to show for the duration of the lane switch, the vehicle is safe from collision from the front vehicle. We note that at any point (x, y) during the lane switching motion, we have the remaining distance needed to travel in the x direction to be $2l - x$. Then, from our 60 degree triangle lane switch motion that our vehicle follows, we know that the remaining change in y direction for our vehicle to complete the turn can be determined with

$$\Delta y = (2l - x) * \sqrt{3} \tag{14}$$

As a result, at any point of the control cycle after commiting the lane switch, it is sufficient for our control system to make sure that

$$fronty - y >= \Delta y + buffer \tag{15}$$

We know the control would not be stuck in future control cycles since the front vehicle would move forward at a constant speed away from the point where we would finish the lane switch, i.e. $y + \Delta y$, during the rest of the lane switching motion.

To summarize, the condition available for our vehicle to have the choice to commit or continue with the turn would be

```
x <= 2*l
```

and we can only make the decision to commit the turn when

```
sidey + rogv*(T + (A(2l - x))^(0.5)\(A/2)) < y - buffer
fronty - y >= (2l - x)*3^(0.5) + buffer
```

Our final control option for the vehicle happens when it completes the lane switch motion, which nothing needs to be done by our assumption that environment after the lane switch becomes irrelevant. Since x would stay at $2l$ after the lane switch by our assumption that vehicle drives at center of the lane, the condition where control decision for after vehicle completing the turn is available can be captured by expression:

```
x = 2*l
```

5.6 Derived Invariants

From our driving system and control construction, we are then able to derive the following invariants at the beginning and after each control cycle:

- $x \geq 0$ & $x < 2l \implies fronty - y > buffer$ since by our construction above, the front vehicle should always be $buffer$ distance away from the finish point of the lane switch when we decide to commit.
- $x > 0$ & $x <= 2l \implies sidey + rogv * \frac{\sqrt{A(2l-x)}}{\frac{A}{2}} < y - buffer$ to make sure that our vehicle is always safe from collision of the side-lane car during and immediately after the lane switch when picking acceleration A at future control cycles.

We note that our first invariant trivially implies our safety condition described by $x >= 0$ & $x < 2l \implies (eqn4)$, and our second invariant $(eqn5)$ implies our second safety condition since we know $A > 0$ and $rogv = 0$ during the turn, which we can derive $sidey < y - buffer$ under $x > 0$ & $x <= 2l$.

5.7 Discussion of Assumptions

Similarly to point 1, most of the assumptions made for our system in point 2, while are very oversimplifying comparing to the driving in real life, are fairly reasonable behaviors. Both the front car and the side-lane car are assumed to travel at a constant speed which normal drivers following the speed limit would perform in a similar way. Since our car is following the front car before committing for the lane switch, it would also be reasonable for our car to travel at a speed slower than the front car in normal driving. While the 60 degree lane switch is an fairly unrealistic assumption for a car to execute, most of the lane switches comes from vehicle cutting through lanes at some specific range of angles, which our control system can be straightforwardly modified to suit the real life scenario.

Other assumptions such as side lane vehicles in front of our cars position is irrelevant and cars are traveling at the same speed are done to simplify the arithmetic for control due to the constraint of time. Finally, the simplified upper bound on time to complete the lane switch is done due to the difficulty in KeYmaeraX in proving properties regarding square roots, similar to the case for point 1.

6 Proving Model for Point 2

By our vehicle’s initial starting precondition of $fronty - y > buffer$ and $x = 0$, both of our derived invariants are trivially implied. Furthermore, from section 3.6, we have shown that our invariant implies the post condition safety properties. Now, we only need to show that our invariants remains true throughout each control cycle to prove our control system is safe through loop rule.

For the case where our vehicle has not committed the lane switching motion, the invariants are proven trivially knowing $v < rogv$ before the turn and our assumption that the environment becomes irrelevant after the turn. What is left to show is that our invariants persists after each control cycle during the lane switching motion. Specifically, we would want to prove

```
sidey + rogv*(A(2l - x))^(0.5)\(A/2)) < y - buffer
fronty - y >= (2l - x)*3^(0.5) + buffer
```

remains true under the continuous evolution of

```
{v’ = a, x’ = v/2, y’ = v*3^(0.5)/2. sidey’ = rogv, fronty’ = rogv, t’ = 1
& t<=T & v>=0 & x>=0 & x<=2*l}
```

Showing the first invariant is true follows similarly to our proof in point 1, where we first use discrete ghost $x0$ and $sidey0$ to represent $x, sidey$ before entering the continuous evolution. Trivially, we are able to prove statements $x0 \leq x$ and $sidey \leq sidey0 + rogv * T$ with auto for the evolution, which we perform dC on the two statements with our original formulation. Now, recall by our control condition in 5.5, we have initially

$$sidey0 + rogv * (T + \frac{\sqrt{A(2l - x0)}}{\frac{A}{2}}) < y - buffer \tag{16}$$

Similar to point 1, we use dW on our formulation, which we are able to prove our first invariant by a combination of $x0 \leq x$ and an additional cut stating $sidey + \frac{\sqrt{A(2l-x0)}}{\frac{A}{2}}) < y - buffer$ that can be proved through $sidey \leq sidey0 + rogv * T$ and equation (16).

What remains to show is our second invariant and we can conclude our result through boxAnd. This can be done directly through dI, where we have

$$(fronty - y >= (2l - x) * \sqrt{3} + buffer)' = (-y' >= -x' * \sqrt{3}) \tag{17}$$

which is true by our continuous evolution of $x' = \frac{v}{2}, y' = \frac{v*\sqrt{3}}{2}$.

7 Conclusion and Future Work

In this project, I have successfully implemented a control model for each of the two different driving scenarios on my daily commute to school along Forbes Ave: Left turning onto the street and lane switching on the street. The control models, while greatly oversimplified comparing to actual driving behaviors under the assumptions of their implementations, are able to be formally verified to be safe using KeYmaeraX, which met my first two goals described in the proposal.

During the completion of the project, I have learned that modeling and verifying control for autonomous vehicles is very difficult. Even only considering the two specific driving scenarios on my commute to school takes a tremendous amount of work, oversimplification, and overestimation in order to achieve a provable control model. It is unimaginable for the amount of work needed and difficulties experienced for modern autonomous car company to get their vehicle on the road under drastically more diverse scenarios. Safety on autonomous driving has always been a topic of debate in the recent years, and completing this project has made me learn that having a provably safe autonomous vehicle control system requires a significant amount of work and still has a long way to go. However, this project has convinced me that, with formal verification tools such as KeYmaeraX, such control system is achievable. Completing this project has also further increased my passion and motivation in the field of implementing autonomous driving control. Maybe one day, there will be a safe vehicle that can carry me to school without needing me to deal with the annoyance and complications of driving on crowded streets like Forbes Ave.

Due to the limitations of time and the unexpected difficulty I have experienced in proving square-root related formulas in KeYmaeraX, I was forced to make some oversimplifications to the control models in my proposal and did not complete my final goal of implementing a simulation for a vehicle under our control model. In the future, I would like to reduce these oversimplifications through measures such as allowing incoming traffic to travel at different speeds and reduce the overestimation for the time needed to complete the turn or the lane switch so that I would prove safety in a more realistic control model. I would also like to finish implementing a simulation of the control model as described in my final proposal goal to show that our control, while provably safe, is also reasonably efficient in that the vehicle would actually commit the turns and lane switch instead of waiting for a very long time.

8 Deliverables

Turning onto Forbes.kyx: The full implementation of control system for point 1 in dL and HS, with documentation.
Turning onto Forbes 2_Proof.kyx: The proof for the control model of our vehicle at point 1.
Switching Lane.kyx: The full implementation of control system for point 2 in dL and HS, with documentation.
Switching Lane Model 2_Proof.kyx: The proof for the control model of our vehicle at point 2.

References

[1] Nikos Aréchiga et al. “Using Theorem Provers to Guarantee Closed-Loop System Properties”. In: *ACC*. Ed. by Dawn Tilbury. 2012, pp. 3573–3580. DOI: 10.1109/ACC.2012.6315388.

[2] Stephanie Lefevre et al. “Lane Keeping Assistance with Learning-Based Driver Model and Model Predictive Control”. In: Sept. 2014.

[3] Nan Li et al. “Game Theoretic Modeling of Driver and Vehicle Interactions for Verification and Validation of Autonomous Vehicle Control Systems”. In: *IEEE Transactions on Control Systems Technology* 26.5 (2018), pp. 1782–1797. DOI: 10.1109/TCST.2017.2723574.

[4] Umit Ozguner, Christoph Stiller, and Keith Redmill. “Systems for Safety and Autonomous Behavior in Cars: The DARPA Grand Challenge Experience”. In: *Proceedings of the IEEE* 95.2 (2007), pp. 397–412. DOI: 10.1109/JPROC.2006.888394.

[5] Wei Zhan et al. “A non-conservatively defensive strategy for urban autonomous driving”. In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. 2016, pp. 459–464. DOI: 10.1109/ITSC.2016.7795595.