15424 Project Proposal

Eric Yang

November 19, 2021

1 System Description

For my final project, I would like to analyze the cyberphysical system of an autonomous vehicle driving along Pittsburgh's Forbes Ave. To simplify the system for the scope of the project, I have separated the vehicle's journey into 2 specific interest points that I personally found difficult to drive myself on my daily commute to class. For the goal of the project, I would like to find a controller for each of the simplified systems of each interest point that is able to command the autonomous vehicle safely. The project was motivated by the previous labs we have completed throughout the course, where after successfully having proven the safety of straight line and rotational motions of various robots, I was fascinated by KeymaeraX's theorem proving power and wanted to utilize the tool to help building a safe vehicle to help me in the interesting and extremely relevant system of my daily commute to school.

2 Formal Model of the System

To account for the unavoidable reaction time of control systems, our models are implemented in a time triggered style. For simplicity in proving safety properties, we assumed the following properties for the system at the 2 points of interest, listed below.

1. Our car is safe once the actions are completed. In real life situations, after turning onto a street or switching lanes, unsafe behaviors can still result such as the drivers at incoming traffic behind does not pay attention and adjust speed to maintain safety distances. Such behaviors are overlooked for the simplicity of the system, and we have allocated a $buffer$ variable to account for a reasonable "safety distance" for drivers to be aware of our completed action instead.

2. The action conducted are the only available actions. In real life situations, driving can be complicated. There are multiple paths to be take for turns and for switching lanes and analyzing all possible behaviors is too complicated for the scope of the project. However, the actions the project, while oversimplified at times, all logically makes sense for a real driver to conduct, and are very relevant to real life situations.

3. Only immediate incoming traffic is considered. In real life situations, there could be multiple cars incoming, and while control models may be able to let us dodge the first car, the car immediately following the first car can also result in security risks. However, in our model, the safety control is done in a very conservative way and most if not all of such dangerous situations from multiple incoming car is implicitly avoided.

The KeymaeraX file of the models can be found with $Turning.kyx, Switching.kyx$ respectively. At the current stage of project, safety remains to be the primary concern for our controller. If time allows, we would also want to implement a efficiency measure for each of the environment; however, this is still a work in progress.

2.1 Point 1: Turning onto Forbes Ave from my apartment

2.1.1 Model Description

As shown in Figure 1, point 1 captures the moment where our autonomous vehicle leaves 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 gets blocked by pedestrian crossing, turning into the left lane is never an efficient choice to get to school. As a result, the car would want to make a turn into the middle lane to start our Forbes Ave commute.

In order to safely turn onto Forbes Ave, our car would need to watch for incoming traffics before making the turn. To simplify our model, we assume the incoming cars are lawful drivers who are following the speed limit of $rogv$ with no acceleration at all. We set each lane of width $2l$, and we assume that our vehicle is making a 90 degree counterclockwise circular motion turn onto the center of the middle lane like a perfectly skilled driver should make. Once the car turn into the lane, we assume the car drives straight with its current velocity. For simplicity, cars in the model are treated as infinitesimal points

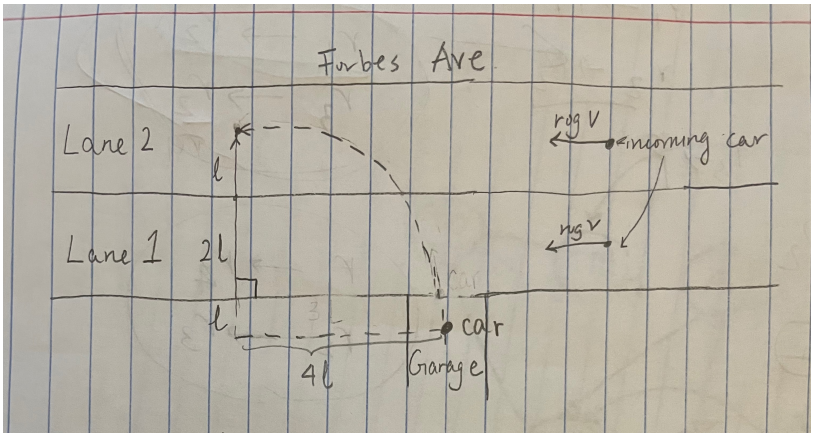

Figure 1: System diagram for point 1

at their center of mass and are required to keep at least a safety distance $buffer$ apart. Also, our initial position center of mass is assumed to be l away from the left lane for consistency of distances. As a result, if we choose the origin as the center of the rotational motion, our vehicle should start at $(4l, 0)$ and end at $(0, 4l)$.

Our car starts from rest when making the turn, and we allows it to accelerate/decelerate with $-B \leq a \leq A$ for $A, B > 0$. Finally, to reflect the random behavior of the road, the initial positions of the incoming traffic is also randomized with x-coordinate position $rogx1, rogx2 > 4l$ and we assume that each incoming car is driving in the center of the lane and thus the y-coordinate should be $rogy1 = 2l$ for left lane and $rogy2 = 4l$ for middle lane. For each iteration of the control cycle, if the current incoming traffic gets in front of our vehicle in x coordinate position, we would randomize the initial position again to reflect the behavior the current incoming car has driven past, and the next following car is the new target for consideration.

2.1.2 Desired Property

First, we would want our vehicle to be always following the 90 degrees turning track in the turning motion since complicated pathing turns is both inefficient and not good driving behavior. This is achieved with the condition

$$x > 0 \rightarrow x^2 + y^2 = (4l)^2 \tag{1}$$

Since the main motivation of the vehicle design is to drive me to school, safety is the primary property of concern. Due to the difficulty in analyzing the distances in rotational motion, I adapted the lowerbound measure of $d_1 = max(abs(rogx1, x), abs(rogy1, y))$ and $d_2 = max(abs(rogx2, x), abs(rogy2, y))$ for the distance between our car on the with incoming traffic on the left land and the middle lane correspondingly. Then, for the safety of the system, we would want

$$max(d_1, d_2) > buffer \tag{2}$$

Finally, it would be safe to assume that for vehicles that drive in different lanes, they should be perfectly safe from collision. As a result, we would want $2l > buffer$ since vehicles driving in different lanes are $2l$ distance apart.

2.1.3 Invariants and Control Intuition

In order for our safety condition to be true, we first define the our control rules to be the following

1. At initial point $(4l, 0)$, choosing $a <= 0$ is valid no matter the incoming traffic's position since that would just mean we continue waiting for the cars to pass.
2. At any point (x, y) along the turn where choosing acceleration a will not make our vehicle to complete the turn in T seconds, we must need to ensure that after driving with acceleration a for T seconds, it is possible to make the turn with maximum acceleration A before the incoming traffic come within $buffer$ range of our initial x position $4l$.
3. At any point (x, y) along the turn where choosing acceleration a will make our vehicle to complete the turn in T seconds, we ensure that incoming traffic can not come within $buffer$ range of the starting position in T seconds.

Indeed, with rule 2 and 3, if we commit in making the turn with acceleration a, there is always a way to ensure us to complete the turn before both incoming traffics reach within $buffer$ range (i.e. by choosing acceleration $a = A$ for rest of all control cycles), since we would also move farther away from initial x position $4l$ along the motion. To achieve these rules, we first note that the condition for rule 1 can be directly translated into

$$y = 0 \ \& \ a <= 0 \tag{3}$$

Determining the conditions for rule 2 and 3 is more complicated. At a fixed point (x, y) along our arc of rotation, the distance on our rotational arc from (x, y) to $(4l, 0)$ is always less than $abs(4l - x) + abs(y - 0) = 4l - x + y$ by triangle inequality of convex sets and our counterclockwise rotational motion. As a result, we can bound the time it would take for our car to complete the rest of the turn through the time for our vehicle to travel $4l - x + y$ in translational motion.

Then, at our current speed v and picked acceleration a, the distance travelled would be $d = v * T + a * T^2/2$, which means we would have at most $4l - x + y - d$ distance left to complete the turn. If we know that $4l - x + y - d \leq 0$, we know that our vehicle has completed the turn, and by our defined behavior in rule 3, we would have

$$rogx1 + rogv * T > 4l + buffer \ \& \ rogx2 + rogv * T > 4l + buffer \tag{4}$$

Otherwise, set t to be the time needed to complete the rest of the distance with acceleration A, we would then have

$$At^2/2 + (v + aT)t = 4l - x + y - d \tag{5}$$

which solve using quadatic formula and knowing that $t \geq 0$ gives

$$t = \frac{-(v + aT) + \sqrt{(v + aT)^2 + 2A(4l - x + y - d)}}{A} \tag{6}$$

And during this time, our incoming traffic would have travelled $rogv * (T + t)$ distance, which means our picked acceleration a need to satisfy

$$rogx2 + rogv * (T + t) > 4l + buffer \ \& \ rogx1 + rogv * (T + t) > 4l + buffer \tag{7}$$

From this control mechanism, we can derive the following loop invariant in the form (1)|(2)|(3), each corresponding to a rule, and they are combined through the *or* operator since our car would need to follow at least one of the rules in the turning motion.

For rule 1, trivially we have the case $y = 0$. And for rule 2, we have the precondition $y > 0$ to signify we have committed to the turning motion. Then, we know when the turn has not finsied, picking $a = A$ for the remainder of the turn will ensure the incoming traffic would not reach within buffer distance to our vehicle's starting point, shown below

$$x > 0 \ \& \ y > 0 \rightarrow rogx2 \ \& \ rogx1 + rogv * \frac{-v + \sqrt{v^2 + 2A(4l - x + y)}}{A} > 4l + buffer \tag{8}$$

where by abuse of notation we have $rogx2$ & $rogx1$ to represent that the inequality is true for substituting in both $rogx2$ and $rogx1$ as the first term of summand. And from 3, we have the invariant for the other case that when the turn gets finished,

$$x = 0 \rightarrow (rogx1 > 4l + buffer \ \& \ rogx2 > 4l + buffer) \tag{9}$$

2.2 Point 2: Switch Lane to continue onto Forbes Ave

2.2.1 Model Description

After turning onto the center lane from point 1, our autonomous vehicle control can enjoy a downtime where we can continue cruising down Forbes ave without having too much concern other than speed control until we reach point 2, where we need to make a lane change to the side lane since the center lane ends upon reaching Cathedral of Learning.

As shown in Figure 2, the system on point 2 can be visualized as a 2 lane system, where we have a car in front of us and a continuous flow of incoming traffic on the right lane. Now, our car need to make a decision on whether continue going straight, or starting to make a lane change on the other lane. Similarly to our model in point 1, the front and side cars are assumed to be at constant velocity of $rogv$, the lane are set to have width $2l$, and we would need our car to stay a safety distance $buffer$ apart from the front car and the incoming traffic from.

We use y coordinate to determine the relative distances of the cars, and x coordinate to determine the lane position of the car. We set our starting car position to be at $(0, 0)$, and have our front car to start at an arbitrary position $(0, fronty)$ where $fronty > buffer$. Due to the lane width, our side car is going to have a x coordinate of $2 * l$, and, similarly to point 1, we would randomize the y position $sidey$ of the side car and rerun the randomization every time the side car drives past our current car.

Similarly to point 1, our car is able to accelerate/decelerate with magnitude $-B \leq a \leq A$ for $A, B > 0$, and in car's regular motion, we would simply drive straight with the continuous dynamic defined by $y' = v, v' = a$. When the car decide to switch lane, we assume our car to be switching lane at a 60 degree angle direction for simplicity of the model, however, the proven result can be trivially modified for any angle direction of choice with manipulation of parameters. With the 60 degree angle assumption then, our car would be following the continuous dynamic defined by $y' = v/\sqrt{3}, x' = v/2, v' = a$.

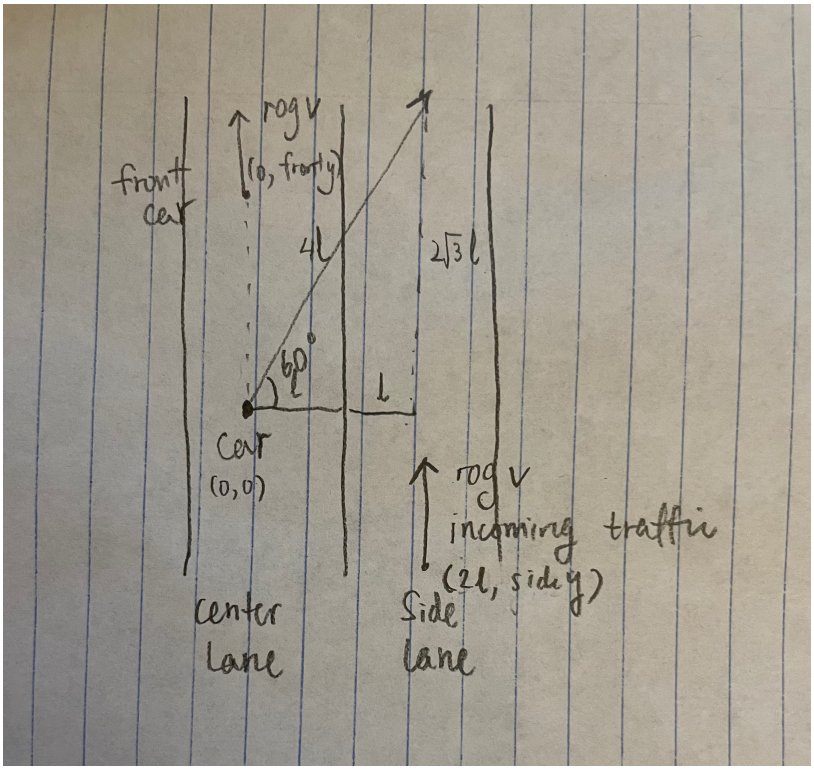

Figure 2: System diagram for section 2

2.2.2 Desired Property

In order for our vehicle control to be safe, we would want to stay the safe $buffer$ distance away from both the front car and side car during the entirety of our lane switch operation. Before our lane switch, we would also want to stay the safe $buffer$ distance away from the front car. As mentioned above, the y coordinate of our car's position is used to calculate the distance between the cars. Combining these factors, we have the following safety conditions:

$$(x >= 0 \ \& \ x <= 2 * l) \rightarrow fronty - y > buffer \tag{10}$$

and

$$(x > 0 \ \& \ x <= 2 * l) \rightarrow y - sidey > buffer \tag{11}$$

2.2.3 Invariants and Control Intuition

Since our model would require our control to make a decision on whether we continue driving down center lane or making the lane switch, we separate the control rules in the two cases.

Case 1: Continue Driving In order for this case to be an available decision, we first would need to not commit in the lane switch action yet. This can be represented by first checking $x = 0$, showing our vehicle is still in the lane. Then, in this case, we would need to make sure that our choice of acceleration would not make us come within $buffer$ with the front car. Specifically, this can be characterized by after driving T seconds with our chosen acceleration, we are able to come to a full stop with full brake on $a := -B$ before we come within range of $buffer$ with the front car. To characterize this control behavior, we have the following scenarios:

$v + aT < 0$: This means our vehicle will come in a full stop during the next T seconds, which means our control would simply need to make sure our car does not stop within $buffer$ range of the front car, giving us

$$v + a * T < 0 \rightarrow y + v^2/(2 * a) + buffer < fronty + rogv * (v/abs(a)) \tag{12}$$

where it takes $v/abs(a)$ to stop our car.

$v + aT > 0$: Then, we would need our vehicle to come to a full stop with $a := -B$ after T seconds outside of the $buffer$ range of the front car. This gives us

$$v + a * T \geq 0 \rightarrow y + v * T + a * T^2/2 + (v + a * T)^2/(2 * B) + buffer < fronty + rogv * (T + (v + a * T)/B) \tag{13}$$

Since after the T seconds, our vechicle will be at position $y + v * T + a * T^2/2$ with velocity $v + a * T$, while our front car will be at position $fronty + rogv * T$. Combining equation (12) and (13) with or operator gives us our control condition for case.

Case 2: Commiting the Lane Switch We set our control condition for committing lane switch in a fairly conservative way. By our assumption of committing the lane switch in a 60 degree angle, at any point (x, y), we would need a gain of $(2l - x) * \sqrt{3}$ in y coordinate to arrive at the center of the side lane by property of similar triangles. As a result, our control first ensures that there is enough space between the car and the front car to make such coordinate gains with the following conservative precondition.

$$fronty - y >= (2l - x) * \sqrt{3} + buffer \tag{14}$$

We did not take into account the speed of the front car since the actual time it takes to conduct the lane switch is unknown. Then, the safety of our car can be ensured through rules identical to 2 and 3 for point 1 described in 2.1.3, stated below:

1. At any point (x, y) along the lane switch where choosing acceleration a will not make our vehicle to complete the switch in T seconds, we must need to ensure that after driving with acceleration a for T seconds, it is possible to make the switch with maximum acceleration A before the incoming traffic $sidey$ come within $buffer$ range of our current y.
2. At any point (x, y) along the turn where choosing acceleration a will make our vehicle to complete the turn in T seconds, we ensure that incoming traffic can not come within $buffer$ range of our current position y within the T seconds.

In this case, the total distance we would need to travel from our initial point to the final point of lane switch would be $4l$ by our triangle property, and at point (x, y), the remaining distance would be of $4l - \sqrt{x^2 + y^2}$ before driving with acceleration a for T seconds. Now, similarly to point 1, we define $d = v * T + a * T^2/2$, we can then separate into two scenarios:

$4l - \sqrt{x^2 + y^2} - d <= 0$, then we know that our vehicle will completed the lane switch within the next T seconds. According to rule 2, we would have

$$sidey + rogv * T < y - buffer \tag{15}$$

$4l - \sqrt{x^2 + y^2} - d > 0$, then similarly to point 1, we are able to compute the time t for the time needed to complete the rest of the lane switch with acceleration A as

$$t = \frac{-(v + aT) + \sqrt{(v + aT)^2 + 2A(4l - \sqrt{x^2 + y^2} - d)}}{A} \tag{16}$$

And giving us rule

$$sidey + rogv * (T + t) < y - buffer \tag{17}$$

From this control mechanism, we can derive the loop invariant in the following form (1) | (2), where (1) corresponds to case 1 and (2) corresponds to case 2. For (1), we know that picking $a := -B$ will make sure that our car can stop outside $buffer$ range of the front car, if our car has not stopped already. The condition for (1) requires us to not commit in our lane switching action yet, giving us the precondition that $x = 0$. This gives us (1):

$$x = 0 \rightarrow y + v^2/(2 * B) + buffer < fronty + rogv * (v/B) \tag{18}$$

Where for (2), similarly to what we have derived for point 1, we would have (2) = (1′) | (2′), where $1', 2'$ corresponds to each rule for case 2. Identically to what we have in point 1 with the exception that we complete our lane switch action upon x coordinate reach $2l$, we have for (1′),

$$x > 0 \ \& \ x < 2l \rightarrow sidey + rogv * \frac{-v + \sqrt{v^2 + 2A(4l - \sqrt{x^2 + y^2})}}{A} < y - buffer \tag{19}$$

and for (2′) we have

$$x = 2l \rightarrow sidey < y - buffer \tag{20}$$

3 Progress of Proofs

So far, developing the model of the two points of interest above has been taking the majority of my time working on the project. For both models, I have proven simpler safety properties such as for point 1, our vehicle would always be on the 90 degree counterclockwise rotational track when turning into Forbes Ave, and for point 2, our vehicle, before committing to making the lane change, would always be at the safety distance apart from the front car. Both proofs are done in a similar

fashion as we have completed for Lab2 and Lab3 of the course.

For the rest of the safety proofs for when our vehicle chooses to commit the respective actions, I am a bit hesitant on starting the proofs before the proposal checkpoint, as I am not very confident on the efficiency of our model since a lot of controls are done very conservatively. For example, on point 1, our car would only starts turning when both lane's cars are very far away from the initial starting point, where in reality the turning motion also makes our car farther away from the incoming traffic, and cars being in different lanes are not considered differently while in reality we do not need to care about the traffic on left lane when we are in turning motion in the middle lane. If possible, some feedbacks on whether my proposed model control is efficient enough would be extremely helpful before I start attempting the proof work.

One difficulty I can forsee in proving the safety of both systems is the wide use of square roots and complicated mathematical expression for control due to the usage of quadratic formula. I am confident in KeymaeraX's ability in handling such mathematical computations, but if such computation becomes troublesome in the proving process, I would look into simplifying the model further by providing an upper bound on the time for our car to finish the turn or lane switch to complete the proof. Any help is greatly appreciated!

4 Stepping Stones

To complete the final deliverable of the project, I have planned the following four stepping stones in helping me reach my goal, listed below:

1. Proving the safety property of vehicle control in environment 1 using KeymaeraX
2. Proving the safety property of vehicle control in environment 2 using KeymaeraX
3. Constructing a visualization on simulating our vehicle with implemented control on the two interest point environments
4. If there is time, add in efficiency conditions to the environments and possibly prove the results

Proving the safety properties through KeymaeraX can be achieved straightforwardly, and I believe both models can be solved through the sequent rules we have learned in class. Simulation visualization can be achieved through drawing simple figures representing cars and lanes on a coordinate system, though determining the details and packages needed are still currently in progress. The exact details of the efficiency conditions for both model's control are also a work in progress, and one significant difficulty in determining a provable condition is that the road conditions of each interest points are also generated randomly (such as distance of incoming cars for the turn in point 1) and we can have very bad conditions for our control to be inefficient. One idea I currently have was to incorporate the simulation implemented from step 3 to have a measure on the average wait time of our car to make a turn in model for point 1 and switch lane for model in point 2 throughout simulations.

5 Related Works

Verifying and Validating autonomous vehicle control systems has been an major issue of CPS research in the recent decade. [1] 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 [2] 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 [3] with a Hidden Markov Model while in [4] through reinforcement learning. These models trained through machine learning, however, are based on probabilitic models and are difficult if not impossible to prove completely safe, which makes our provable control system with hybrid systems novel and particularly useful in modern autonomous vehicle control designs.

6 Citations

[1] U. Ozguner, C. Stiller and K. Redmill, "Systems for Safety and Autonomous Behavior in Cars: The DARPA Grand Challenge Experience," in Proceedings of the IEEE, vol. 95, no. 2, pp. 397-412, Feb. 2007, doi: 10.1109/JPROC.2006.888394.

[2] W. Zhan, C. Liu, C. Chan and M. Tomizuka, "A non-conservatively defensive strategy for urban autonomous driving," 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), 2016, pp. 459-464, doi: 10.1109/ITSC.2016.7795595.

[3] S. Lefevre, Y. Gao, D. Vasquez, E. Tseng, R. Bajcsy, and F. Borrelli,"Lane keeping assistance with learning-based driver model and modelpredictive control," inProceedings of the 12th international symposiumon advanced vehicle control, 2014.

[4] N. Li, D. W. Oyler, M. Zhang, Y. Yildiz, I. Kolmanovsky and A. R. Girard, "Game Theoretic Modeling of Driver and Vehicle Interactions for Verification and Validation of Autonomous Vehicle Control Systems," in IEEE Transactions on Control Systems Technology, vol. 26, no. 5, pp. 1782-1797, Sept. 2018, doi: 10.1109/TCST.2017.2723574.